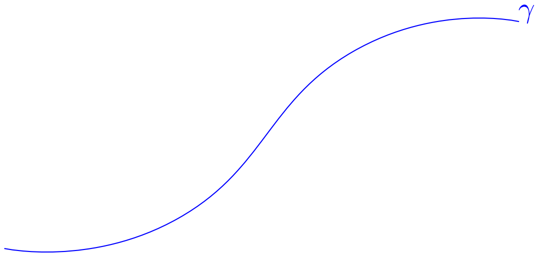
The complexity of tangent words

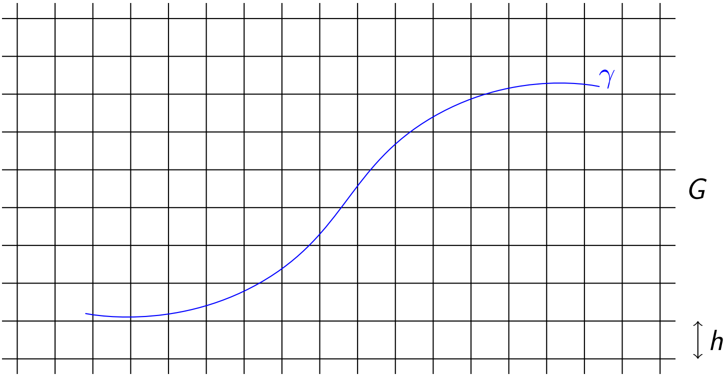
Thierry Monteil

WORDS 2011

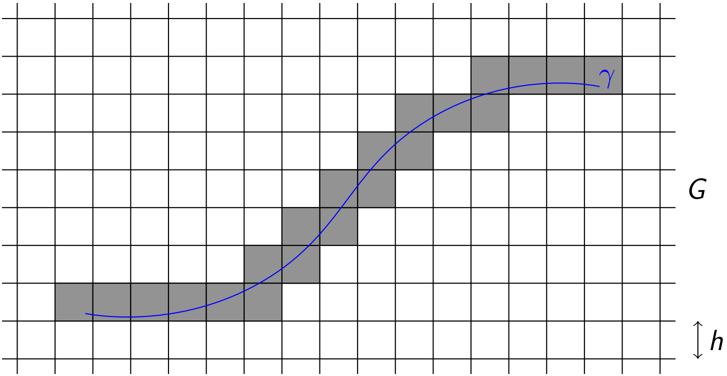
Coding of a smooth curve



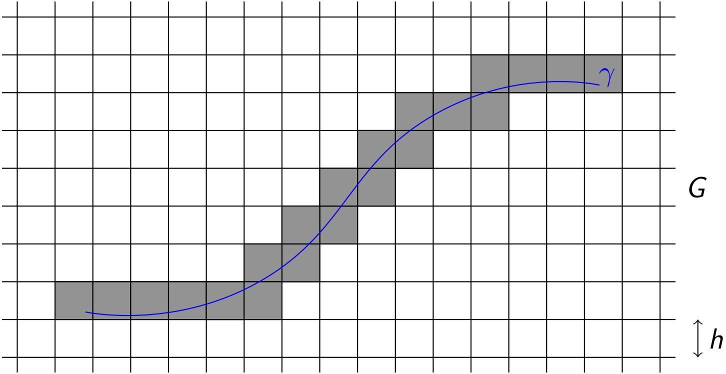
Coding of a smooth curve



Coding of a smooth curve

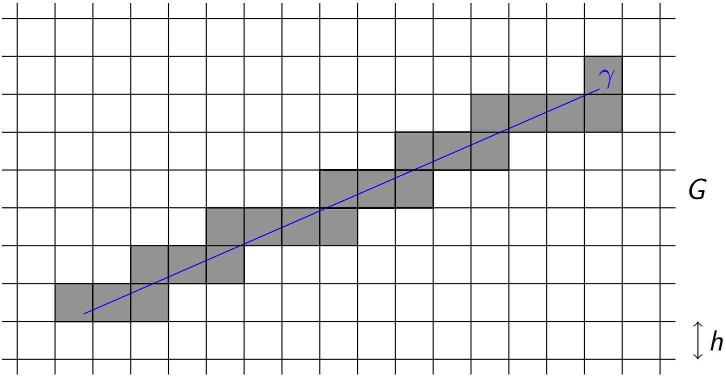


Coding of a smooth curve



$$F(\gamma, G) = 000001010101001000$$

The case of digital straight segments: picture



$$w = F(\gamma, G) = 0010010001001001001$$

The case of digital straight segments: characterisations

There are three types of characterisations of the codings of digital straight segments, corresponding to three properties of the lines:

- *Lines are the curves of constant slope*
  A word w is 1-*balanced* if
  $\forall u, v \in Fact(w) \quad |u| = |v| \Rightarrow ||u|_1 - |v|_1| \leq 1$
- *The set of lines is stable under linear maps*
  Multi-scale structure (desubstitution, continued fractions):
  $w = 0010010001001001001 \mapsto 11011101 \mapsto 00 \mapsto \varepsilon$
- *Lines are the most predictible curves*
  A word w is a *Sturmian factor* if
  w is a factor of an infinite word with complexity $n + 1$

The case of digital straight segments: complexity

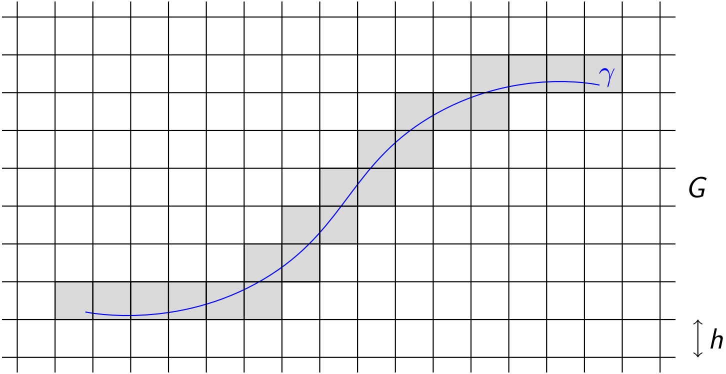
The complexity of balanced words was studied [Lipatov 1982], [Mignosi 1991] and [Berstel Pocchiola 1993], where it was proved to be equal to:

$$p_n(B) = 1 + \sum_{i=1}^{n} \sum_{j=1}^{i} \varphi(j) = 1 + \sum_{i=1}^{n} (n - i + 1)\varphi(i)$$

where φ denotes the Euler totient function:

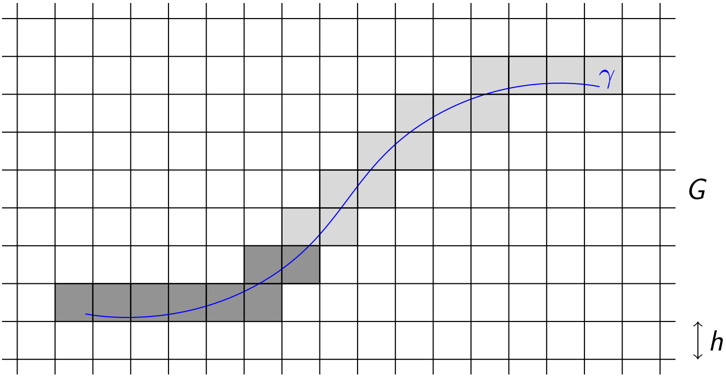
$$\varphi(n) = card\{k \leq n \mid \gcd(k, n) = 1\}$$

Segmentation



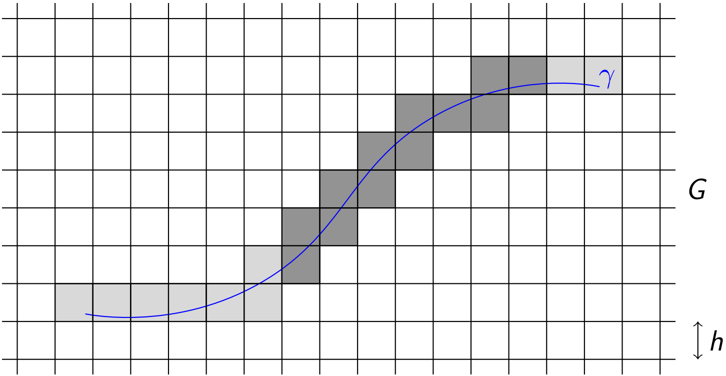
$$F(\gamma, G) = 0000010.10101010010.00$$

Segmentation



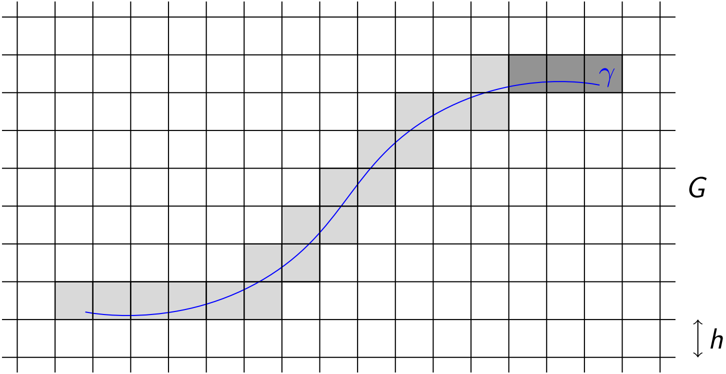
$$F(\gamma, G) = 0000010.10101010010.00$$

Segmentation



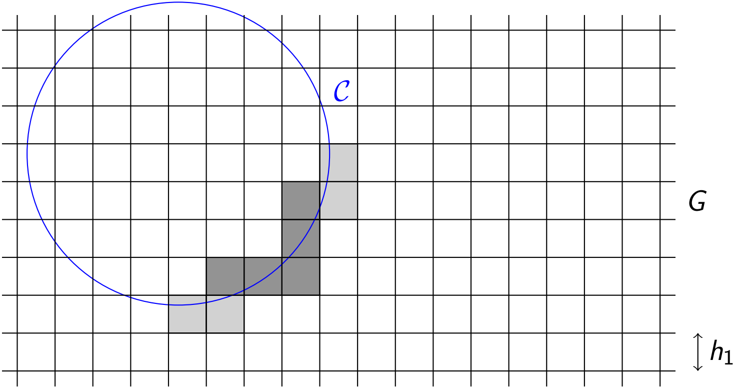
$$F(\gamma, G) = 0000010.10101010010.00$$

Segmentation



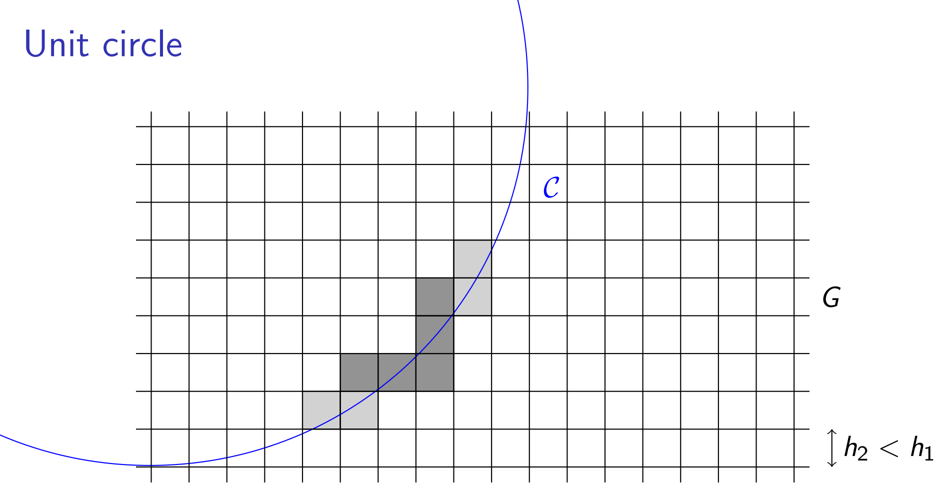
$$F(\gamma, G) = 0000010.10101010010.\textcolor{blue}{00}$$

Unit circle



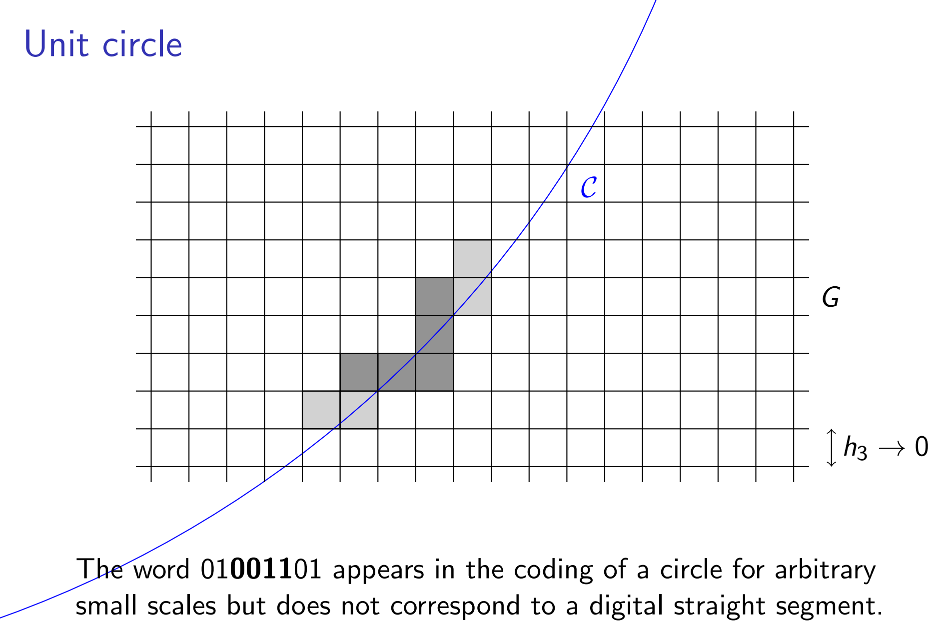
The word 01**0011**01 appears in the coding of a circle for arbitrary small scales but does not correspond to a digital straight segment.

Unit circle



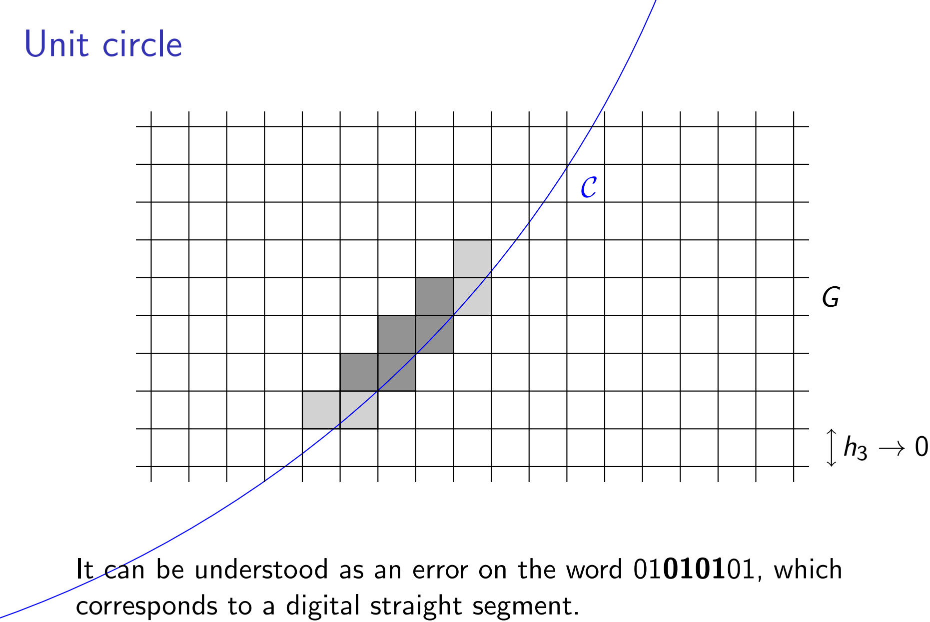
The word 01**0011**01 appears in the coding of a circle for arbitrary small scales but does not correspond to a digital straight segment.

Unit circle



The word 01**0011**01 appears in the coding of a circle for arbitrary small scales but does not correspond to a digital straight segment.

Unit circle



It can be understood as an error on the word 01**0101**01, which corresponds to a digital straight segment.

Tangent words

The aim of this talk is to describe and count those words that survive in the coding of a smooth curve when the mesh of the grid goes to 0.

If γ is a smooth curve, we define the set of tangent words to γ as

$$T(\gamma) = \limsup_{mesh(G)\to 0} L(\gamma, G) = \bigcap_{\varepsilon > 0} \bigcup_{mesh(G)\leq\varepsilon} L(\gamma, G),$$

where $L(G, \gamma)$ denotes the set of words appearing in the coding of γ by the grid G.

We denote by T^∞ the set of all tangent words obtained for all smooth curves and call them *tangent words*.

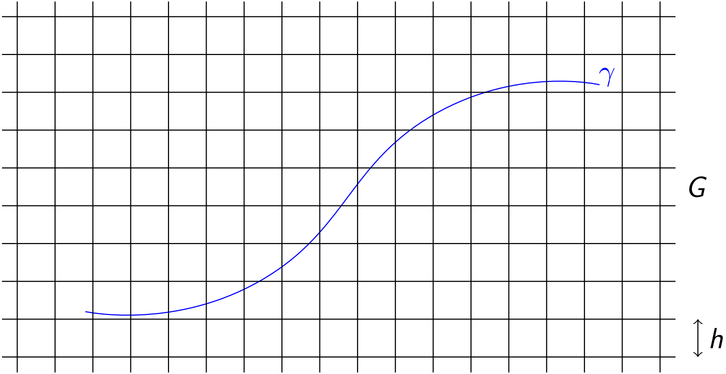
First examples

Any coding of a digital straight segment is tangent.

We just saw that the word 0011 is tangent.

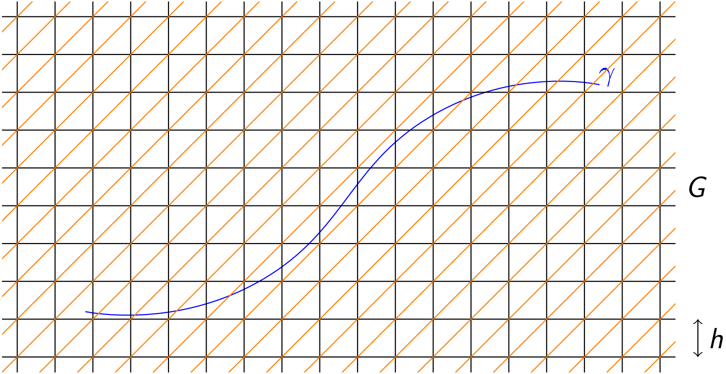
The word 00011 is not tangent.

Combinatorial characterisation: recoding



$w = 000001010101001000$

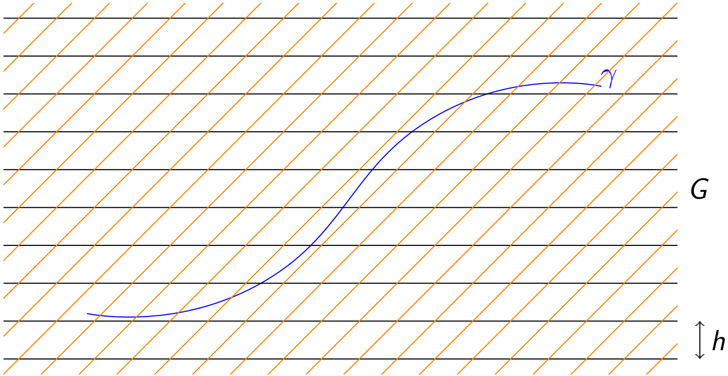
Combinatorial characterisation: recoding



$w = 00000101010101001000$

02020202010101010102010202 0

Combinatorial characterisation: recoding

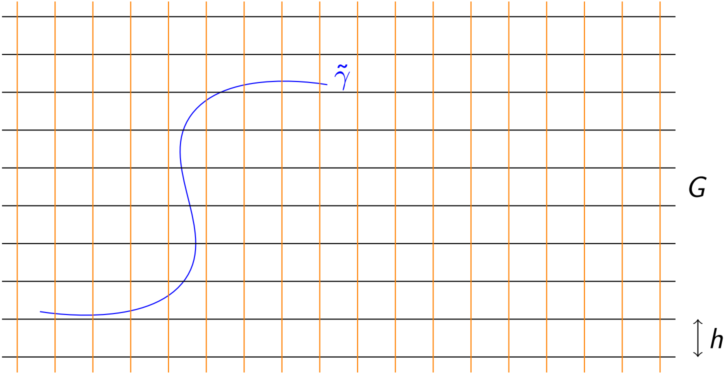


$w = 00000101010101001000$

0202020201010101010201020 20

2222111112122

Combinatorial characterisation: recoding



$w = 0000010101010101001000$

0202020201010101010201020200

2222111112122

$\delta(w) = 0000111110100$

We removed one letter to each run of the non-isolated letter.

Combinatorial characterisation: multi-scale structure

A word w is a coding of a curve γ if, and only if, $\delta(w)$ is a coding of $\tilde{\gamma} = \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix} \circ \gamma$ (resp. $\begin{pmatrix} 1 & 0 \\ -1 & 1 \end{pmatrix} \circ \gamma$ if 0 is isolated).

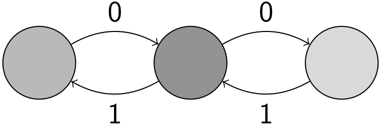
Hence, a word w is tangent if, and only if, $\delta(w)$ is tangent.

Since $\delta(w)$ is shorter than w, we can repeat the process until we are stuck: we get a word $d(w)$.

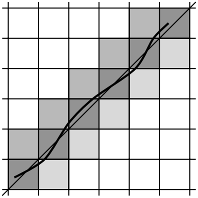
- $d(w)$ is empty if, and only if, w is a coding of a digital straight segment.
- Otherwise, $d(w)$ contains 00 and 11: we have to characterize tangent words whose slope is 1 (called *diagonal words*).

Combinatorial characterisation: slope 1

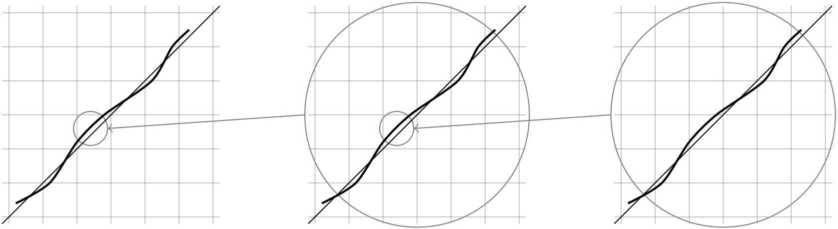
A word w is tangent if, and only if, $d(w)$ is recognised by the following automaton with three states, which are all considered as initial and accepting:



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



Slope 1: how to produce such oscillations ?



An example

$$100100010010010010001001000100 = w$$

An example

$100100010010010010001001000100 = w$

$1\cancel{00}1\cancel{00}01\cancel{00}1\cancel{00}1\cancel{00}1\cancel{00}01\cancel{00}1\cancel{00}01\cancel{00}$

An example

$100100010010010001001000100 = w$

1~~00~~1~~00~~01~~00~~1~~00~~1~~00~~1~~00~~01~~00~~1~~00~~01~~00~~

110111101101

An example

$100100010010010001001000100 = w$

1~~00~~1~~000~~1~~00~~1~~00~~1~~000~~1~~00~~1~~000~~1~~00~~

110111101101

~~11~~0~~11~~110~~11~~0~~1~~

An example

$1001000100100100010010001000100 = w$

$1\not{0}\not{0}1\not{0}\not{0}\not{0}1\not{0}\not{0}1\not{0}\not{0}1\not{0}\not{0}1\not{0}\not{0}\not{0}1\not{0}\not{0}1\not{0}\not{0}\not{0}1\not{0}\not{0}$

110111101101

$\not{1}\not{1}0\not{1}\not{1}110\not{1}\not{1}0\not{1}$

$01100 = d(w)$

An example

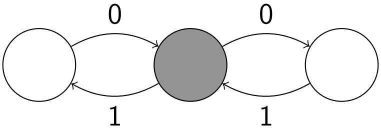
$1001000100100100010010001 00 = w$

1001000100100100010010001 00

110111101101

110111101101

$01100 = d(w)$



An example

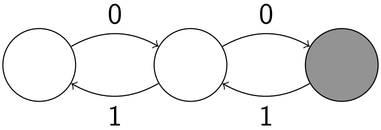
$1001000100100100010010001 00 = w$

1001000100100100010010001 00

110111101101

110111101101

$01100 = d(w)$



An example

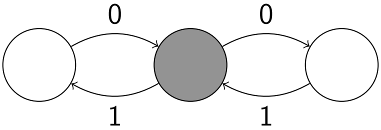
$1001000100100100010010001 00 = w$

1001000100100100010010001 00

110111101101

110111101101

$01100 = d(w)$



An example

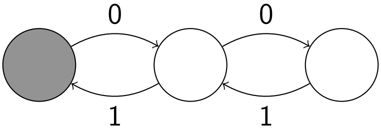
$1001000100100100010010001000100 = w$

$1001000100100100010010001000100$

110111101101

110111101101

$01100 = d(w)$



An example

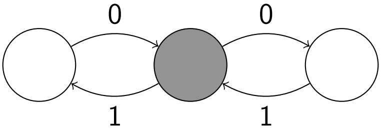
$1001000100100100010010001 00 = w$

$10010001001001000100100 0100$

110111101101

110111101101

$01100 = d(w)$



An example

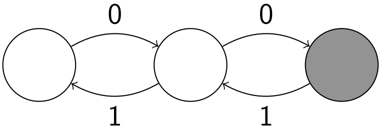
$1001000100100100010010001 00 = w$

1001000100100100010010001 00

110111101101

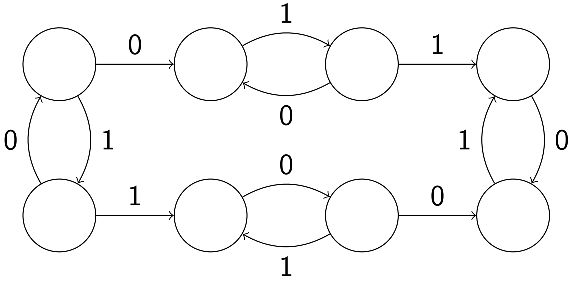
110111101101

$01100 = d(w)$



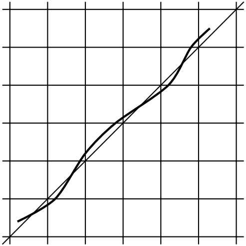
No bad vibration: analytic curves

A word w is a tangent word of an analytic curve if, and only if, $d(w)$ is recognised by the following automaton with eight states, which are all considered as initial and accepting:

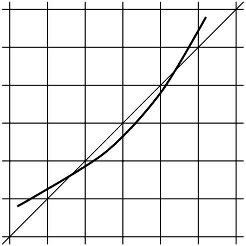


The *tangent analytic words* also correspond to the tangent words of the smooth curves with nowhere zero curvature, we denote this set by T^{ω}.

No bad vibration: analytic curves



0110100110
tangent



1001010110
tangent analytic

Complexity of tangent words

The use of bispecial factors

To compute the complexity of T^∞ and T^ω, we will use the following formula introduced in [Cassaigne 1994] using bispecial factors:

$$p_n(L) = 1 + n + \sum_{i=0}^{n-1} \sum_{j=0}^{i-1} (sb_j(L) - wb_j(L))$$

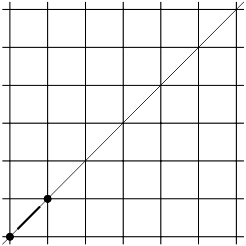
where $wb_j(L)$ (resp. $sb_j(L)$) denotes the number of weak (resp. strong) bispecial factors of length j in L.

Desubstitution

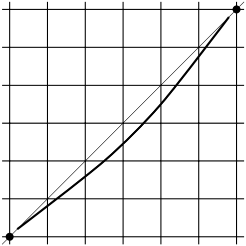
If w is a non-diagonal bispecial tangent word, then it can be desubstituted (in a single way) and $\delta(w)$ is a bispecial factor of the same kind.

Diagonal strong bispecials

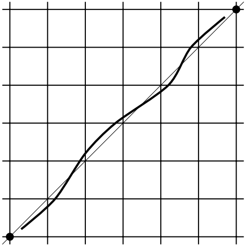
Balanced



Tangent analytic

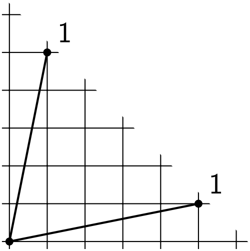


Tangent



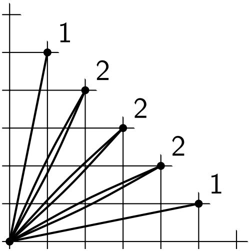
Resubstitution ($SL(2,\mathbb{Z})$-action)

Balanced



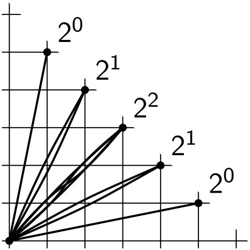
$\varphi(j)$

Tangent analytic



$2j - \varphi(j)$

Tangent



$$\sum_{\substack{d|j \\ d\neq 1}} \varphi(j) 2^{j/d-1}$$

Result

$$p_n(B) = 1 + \sum_{i=1}^{n} \sum_{j=1}^{i} \varphi(j) \quad \text{[Lipatov 1982]}$$

$$p_n(T^\omega) = 1 + n + \sum_{i=1}^{n} \sum_{j=2}^{i} (2j - \varphi(j) - 1)$$

$$p_n(T^\infty) = 1 + n + \frac{1}{2} \sum_{i=1}^{n} \sum_{j=2}^{i} \sum_{\substack{d \mid j \\ d \neq 1}} \varphi(j) 2^{j/d}$$

Conclusion: balance and complexity

Each class of words is strictly included in the next one:

- 1-balanced words (digital straight segments)
- tangent analytic words
- tangent (smooth) words
- 2-balanced words

For the first two classes, the complexity is cubical whereas for the last two classes, the complexity is exponential.

(Recall that a word is k-balanced if $\forall u, v \in Fact(w) \quad |u| = |v| \Rightarrow ||u|_1 - |v|_1| \leq k$)