A semi-algorithm to explore the set of imbalances in S-adic systems

Mélodie Andrieu

Advisor : Julien Cassaigne
Institut de Mathématiques de Marseille

Dyadisc3, Amiens, July 2019

I - MOTIVATION : understand the dynamics of [multiD] continued fraction algorithms.

II - OUR TOOL : a semi-algorithm to explore imbalances in S-adic systems.

III - APPLICATION : to Cassaigne-Selmer continued fraction algorithm.

definitions & notations

... "S-adic" ?

Be $\mathcal{A}$ a finite set ⟵ "alphabet"

"Substitution" : $\sigma \in End(\mathcal{A}^*)$

ex :
- $\sigma_0 : 1 \mapsto 2; 2 \mapsto 1$
- $\sigma_1 : 1 \mapsto 11; 2 \mapsto \epsilon$
- Thue-Morse : $\sigma_{TM} : 1 \mapsto 12; 2 \mapsto 21$
- Fibonacci : $\sigma_{Fib} : 1 \mapsto 12; 2 \mapsto 1$
- Tribonacci : $\sigma_{Trib} : 1 \mapsto 12; 2 \mapsto 13; 3 \mapsto 1$

"Substitutive word" :

an *infinite word* that can be written : $w = \lim_{n\to\infty} \sigma^n(a)$ with $a \in \mathcal{A}$ ⟵ "seed"

ex :
- no word for σ_0
- $w_{\sigma_1} = 1111111111111111...$
- Thue-Morse : $\begin{cases} w_{TM} = 1221211221121221... \\ w'_{TM} = 2112122112212112... \end{cases}$
- $w_{Fib} = 1211212112112121...$
- $w_{Trib} = 1213121121312...$

definitions & notations

... "S-adic" ?

Be S a *finite* set of *non-erasing* substitutions on a *common alphabet* $\mathcal{A}$.

"S-adic word" : an *infinite word* that can be written : $w = \lim_{n\to\infty} \sigma_0 \circ \ldots \circ \sigma_{n-1}(a)$

with : - $a \in \mathcal{A} \longleftarrow$ "seed"
- $(\sigma_n) \in S^{\mathbb{N}} \longleftarrow$ "directive sequence"

ex : - substitutive words $S = \{\sigma\}$
- but not only

"S-adic system" : given S, the set of *all* S-adic words

definitions & notations

... "imbalance" ?

Be w a finite or infinite word on $\mathcal{A}$.

"imbalance of w" : $Imb(w) := \sup_{n\in\mathbb{N}} \sup_{u,v\in F_n(w)} \max_{l\in A} \left| |u|_l - |v|_l \right|$ $\in \mathbb{N}$ or ∞

where : $|u|_l$ = number of 'l' in u ex : $|banana|_a = 3$

ex : Imb('1221') = 1

ex :
- Thue-Morse : $w_{TM} = 1221211221121221...$ $Imb(w_{TM}) = 2$
- Fibonacci : $w_{Fib} = 1211212112112121...$ $Imb(w_{Fib}) = 1$
- Tribonacci : $w_{Trib} = 1213121121312...$ $Imb(w_{Trib}) = 2$

Motivations ?

Why are S-adic systems of interest ?

What should we study their imbalances ?

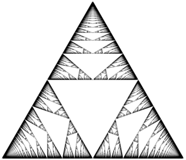
$\longrightarrow$ fruitful results for understanding the dynamics of (multiD) continued fraction algorithm.

1D : Euclid's algorithm

Dynamical system	$\begin{array}{ccl} \mathbb{R}_+^2 & \longrightarrow & \mathbb{R}_+^2 \\ (x, y) & \longmapsto & (x-y, y) \text{ if } x \geq y \\ & & (x, y-x) \text{ otherwise.} \end{array}$
Associated words	sturmian words
Substitutions set	$\tau_1 : \begin{array}{ccc} 1 & \mapsto & 1 \\ 2 & \mapsto & 21 \end{array}$ $\tau_2 : \begin{array}{ccc} 1 & \mapsto & 12 \\ 2 & \mapsto & 2 \end{array}$
(Im)balance	1-balanced
Rotation coding	✓

2D : Arnoux-Rauzy algorithm [Arnoux, Rauzy, 1991]

Dynamical system	$\begin{array}{lcl} \mathcal{G} \subseteq \mathbb{R}^3_+ & \longrightarrow & \mathcal{G} \\ (x,y,z) & \longmapsto & (x-y-z,y,z) \text{ if } x \geq y+z \\ & & (x,y-x-z,z) \text{ if } y \geq x+z \\ & & (x,y,z-x-y) \text{ if } z \geq x+y. \end{array}$
Associated words	Arnoux-Rauzy words
Substitutions set	$\sigma_1 : 1 \mapsto 1,\ 2 \mapsto 21,\ 3 \mapsto 31$; $\sigma_2 : 1 \mapsto 12,\ 2 \mapsto 2,\ 3 \mapsto 32$; $\sigma_3 : 1 \mapsto 13,\ 2 \mapsto 23,\ 3 \mapsto 3$
(Im)balance	There exist AR words with infinite imbalance. [Cassaigne, Ferenczi, Zamboni, 2001]
Rotation coding	X [Cassaigne, Ferenczi, Messaoudi, 2008]



The Rauzy gasket $\mathcal{G}$.

A new candidate for 2D : Cassaigne-Selmer algorithm

Dynamical system	$\begin{array}{rcl} \mathbb{R}_+^3 & \longrightarrow & \mathbb{R}_+^3 \\ (x,y,z) & \longmapsto & (x-z,z,y) \text{ if } x \geq z \\ & & (y,x,z-x) \text{ otherwise.} \end{array}$
Associated words	C-adic words
Substitutions set	$c_1 : \begin{array}{rcl} 1 & \mapsto & 1 \\ 2 & \mapsto & 13 \\ 3 & \mapsto & 2 \end{array}$ $c_2 : \begin{array}{rcl} 1 & \mapsto & 2 \\ 2 & \mapsto & 13 \\ 3 & \mapsto & 3 \end{array}$
(Im)balance	?
Rotation coding	?

$C := \{c_1, c_2\}$.
What can be said about the imbalance of C-adic words ?

A new candidate for 2D : Cassaigne-Selmer algorithm

Dynamical system	$\begin{array}{rcl} \mathbb{R}_+^3 & \longrightarrow & \mathbb{R}_+^3 \\ (x,y,z) & \longmapsto & (x-z,z,y) \text{ if } x \geq z \\ & & (y,x,z-x) \text{ otherwise.} \end{array}$
Associated words	C-adic words
Substitutions set	$c_1 : \begin{array}{rcl} 1 & \mapsto & 1 \\ 2 & \mapsto & 13 \\ 3 & \mapsto & 2 \end{array}$ $\quad c_2 : \begin{array}{rcl} 1 & \mapsto & 2 \\ 2 & \mapsto & 13 \\ 3 & \mapsto & 3 \end{array}$
(Im)balance	There exists C-adic words with infinite imbalance.
Rotation coding	?

$C := \{c_1, c_2\}$.

What can be said about the imbalance of C-adic words ?

Foretaste

Theorem

Be S a finite set of non-erasing substitutions over $\mathcal{A}$, $n \in \mathbb{N}$.
Assume (H) that each $a \in \mathcal{A}$ appear in at least one S-adic word.
*Then "There exists w S-adic such that $Imb(w) \geq n$" is **semi-decidable**.*

$\longrightarrow$ If such a w exists, we'll find it !

Our semi-algorithm "imbalances automaton" : a *growing* family of automata, for which :

imbalances	$\approx$ final states
directive sequences	$\approx$ paths labels

II - TOOL : imbalances automaton

Everything hinge on a finiteness lemma !

Be $C \in \mathbb{N}$

Lemma

$\exists w$ *S-adic s.t.* $Imb(w) \geq C \quad \Rightarrow \quad \exists \sigma_0, \ldots, \sigma_{n_0} \in S, a \in \mathcal{A}$
$\exists u, v \in F(\sigma_0 \circ \ldots \circ \sigma_{n_0}(a))$ *s.t* :
$|u| = |v|$ *and* $\max_{l \in A} ||u|_l - |v|_l| \geq C$

Is the **converse** true ?

$\longrightarrow$ In general : NO
$\longrightarrow$ If (H) : YES

Consequence : a naive semi-algorithm to detect imbalances.

We can do better !

1) Be methodic :
understand where does an imbalance come from, to reinvest what we learnt at a previous step.

$\longrightarrow$ desubstitution

2) Eco-friendly computing :
do we really need to know everything about u and v ?

$\longrightarrow$ semi-abelianization

1) Be methodic

Let's desubstitute !

Be $u,v \in F(\sigma_0 \circ \ldots \circ \sigma_{n_0}(a))$, s.t. $|u| = |v|$.

$\longrightarrow$ Where do u and v come from ?

$$\begin{aligned} \sigma_1 \circ \ldots \circ \sigma_{n_0-1}(a) &= l_0 \ldots\ldots \overbrace{l_i \ldots\ldots l_j}^{\tilde{u}} \ldots\ldots l_k \\ \sigma_0 \circ \sigma_1 \circ \ldots \circ \sigma_{n-1}(a) &= \sigma_0(l_0) \ldots \sigma_0 \underbrace{(l_i) \ldots \sigma_0(l_j)}_{u} \ldots \sigma_0(l_k) \end{aligned}$$

Good point : directive sequence is shorter

Bad point : $\widetilde{u}$ and $\widetilde{v}$ may not have the same length !

Compromise : study the couples of factors in S-adic words, with no consideration on their lengths.

1) Be methodic

Towards an "infinite" automaton

$\mathfrak{S} := \{(u, v) \mid \exists w \text{ S-adic s.t. } u, v \in F(w)\}$ states

$\mathfrak{F} := \{(u, v) \in \mathfrak{S} \mid |u| = |v|\}$ final states

...transitions ?

$\longrightarrow$ **converse of the desubstitution**

which is NOT the substitution

...but the "substitute and cut" operation.

1) Be methodic : the **substitute** and **cut** operation

Be $u, \widetilde{u} \in \mathcal{A}^*$.

Notations : - $\alpha(u)$ and $\omega(u)$ the first and last letter of u ;
- $p_k(u)$ [$s_k(u)$] the prefix [suffix] of u of length k.

- A **substitute and cut operation from $\widetilde{u}$ to u** is a triplet $(\sigma, \beta, \gamma) \in S \times \mathbb{N}^2$ s.t. :
  - $p_\beta(\sigma(\widetilde{u})).u.s_\gamma(\sigma(\widetilde{u})) = \sigma(\widetilde{u})$
  - Cutting conditions : $\begin{cases} \beta = \gamma = 0 \text{ if } \widetilde{u} = \epsilon \text{ (empty word)} \\ \beta + \gamma \leq |\sigma(\widetilde{u})| \text{ and } \beta, \gamma < |\sigma(\widetilde{u})| \text{ if } |\widetilde{u}| = 1 \\ \beta < |\sigma(\alpha(\widetilde{u}))| \text{ and } \gamma < |\sigma(\omega(\widetilde{u}))| \text{ otherwise.} \end{cases}$ (✂)

ex : $S = \left\{ \begin{array}{llll} c1 : & 1 \mapsto 1 & & c2 : \ 1 \mapsto 2 \\ & 2 \mapsto 13 & & \ 2 \mapsto 13 \\ & 3 \mapsto 2 & & \ 3 \mapsto 3 \end{array} \right\}$

From $u = 23$, there are 4 allowed S&C operations :
$(c_1, 0, 0), (c_1, 1, 0), (c_2, 0, 0)$ and $(c_2, 1, 0)$.
which give respectively $132, 32, 133$ and 33.

1) Be methodic : the substitute and cut operation

Be $u, \widetilde{u}, v, \widetilde{v} \in \mathcal{A}^*$.

- A **substitute and cut operation from** $(\widetilde{u}, \widetilde{v})$ **to** (u, v) is $(\sigma, \beta, \gamma, \delta, \eta) \in S \times \mathbb{N}^4$ s.t. :
  - (σ, β, γ) is a S&C operation from $\widetilde{u}$ to u
  - (σ, δ, η) is a S&C operation from $\widetilde{v}$ to v.

  We denote it ${}^{\beta}_{\delta}\sigma^{\gamma}_{\eta}$.
- A quintuplet $(\sigma, \beta, \gamma, \delta, \eta)$ which satisfy (✂) is **"allowed"**.

ex : $S = \left\{ \begin{array}{llll} c1: & 1 & \mapsto & 1 \\ & 2 & \mapsto & 13 \\ & 3 & \mapsto & 2 \end{array} \quad \begin{array}{llll} c2: & 1 & \mapsto & 2 \\ & 2 & \mapsto & 13 \\ & 3 & \mapsto & 3 \end{array} \right\}$ $u = 23$ $v = 33$

There are 4 allowed S&C operations from (u,v) : ${}^{0}_{0}c1^{0}_{0}$, ${}^{1}_{0}c1^{0}_{0}$, ${}^{0}_{0}c2^{0}_{0}$ and ${}^{1}_{0}c2^{0}_{0}$ which give respectively $(132, 22), (32, 22), (133, 33)$ and $(33, 33)$.

1) Be methodic

Closer to the "infinite" automaton

$\mathfrak{S}$	:=	$\{(u,v) \mid \exists w$ S-adic s.t. $u,v \in F(w)\}$	states
$\mathfrak{F}$	:=	$\{(u,v) \in \mathfrak{S} \mid \lvert u\rvert = \lvert v\rvert\}$	final states
$(\widetilde{u},\widetilde{v}) \to (u,v)$		allowed S&C operations	transitions

Good points :

- Given $(u,v) \in \mathfrak{S}$, the number of allowed S & C operations from (u,v) is finite.
- Their image remain in $\mathfrak{S}$.

...Initial states ?

$\longrightarrow$ Does there exist a finite set $\mathfrak{I} \subset \mathfrak{S}$ from which any state can be reached ?

1) Be methodic

A finite initial set

Theorem

If $\forall a \in \mathcal{A}$ there exists a S-adic word w s.t. $a \in F(w)$ (H), then there exists a finite set $\mathfrak{I} \in \mathfrak{S}$ s.t. :

*$\forall (u,v) \in \mathfrak{S}, \exists (u_0, v_0) \in \mathfrak{I}, \exists (T_i)_{i \in \{0,n_0\}} = ({}^{\beta i}_{\delta i}\sigma i^{\gamma i}_{\eta i})_{i \in \{0,n_0\}}$ a **finite** sequence of **allowed** substitute and cut operations s.t. :*

$$(u,v) = T_{n_0 - 1} \circ \ldots \circ T_0(u_0, v_0).$$

We can take

$$\mathfrak{I} = \{(\epsilon,\epsilon)\} \cup \bigcup_{a \in \mathcal{A}} \{(a,\epsilon),(\epsilon,a),(a,a)\}.$$

elegance is coming

Be $u, v \in \mathcal{A}^*$.

"abelianized of u" : $ab(u) := (|u|_l)_{l\in\mathcal{A}} \in \mathbb{N}^{\mathcal{A}}$.

ex : $ab('12111') = (4, 1)$

Properties :

1. $ab(u).(1)_{l\in\mathcal{A}} = |u|$
2. $[ab(u) - ab(v)].(1)_{l\in\mathcal{A}} = 0 \iff |u| = |v|$.
   then $ab(u) - ab(v)$ is the **"imbalance vector "** of u and v.
3. If $w \in \mathcal{A}^{\mathbb{N}}$, $Imb(w) = \sup_{u,v\in F(w),|u|=|v|} ||ab(u) - ab(v)||_\infty$

Lemma : Given S, the imbalance of S-adic words is bounded iff the set of imbalance vectors $\{$ab(u)-ab(v) $|$ $(u, v) \in \mathfrak{F}\}$ is finite.

$\longrightarrow$ We want to explore this set !

But abelianization breaks the transitions !

Consider the equivalence relation on $(\mathcal{A}^*)^2$:

$$(u,v)\sim_{ab}(u',v') \iff ab(u)-ab(v)=ab(u')-ab(v')$$

$\longrightarrow$ **Factorize** $\mathfrak{S}$ by $\sim_{ab}$?

Problem : $x \in \mathfrak{S}/\sim_{ab}$ can represent in the same time (u,v) and (u',v') for which the sets of allowed S&C operations (and the result they give) are different !

ex : $x=(0,0,0)$ represents $u=v=132$ as well as $u'=v'=\epsilon$.
${}^0_0c1^1_0$ is allowed for (u,v) but not for (u',v').

—— Recall ——

Cutting conditions : $\begin{cases} \beta=\gamma=0 \text{ if } \widetilde{u}=\epsilon \text{ (empty word)} \\ \beta+\gamma \leq |\sigma(\widetilde{u})| \text{ and } \beta,\gamma<|\sigma(\widetilde{u})| \text{ if } |\widetilde{u}|=1 \\ \beta<|\sigma(\alpha(\widetilde{u}))| \text{ and } \gamma<|\sigma(\omega(\widetilde{u}))| \text{ otherwise.} \end{cases}$ (✂)

$\longrightarrow$ We need information about extremal letters !

Semi-abelianizaton

Solution : append a **"matrix of extremities"** containing the first/last letter of (u, v).

A more restrictive equivalence relation :

$$(u,v)\sim_{sab}(u',v') \iff \begin{cases} ab(u)-ab(v)=ab(u')-ab(v') \\ M_{ext}(u,v)=M_{ext}(u',v') \end{cases}$$

ex : $\begin{pmatrix} a & b \\ c & c \end{pmatrix}, (1,1,-1)$ represents $(acb, cc), \ldots$

$\begin{pmatrix} a & . \\ c & c \end{pmatrix}, (1,0,-2)$ represents (a, cc).

$\begin{pmatrix} . & . \\ . & . \end{pmatrix}, (0,0,0)$ represents (ϵ, ϵ).

Theorem : The S&C operation is compatible with $\sim_{sab}$.

Hence the quotient graph :

$\mathcal{S}$:= $\{(M,x) \in Ext \times \mathbb{Z}^{\mathcal{A}} \mid \exists$w S-adic $, \exists u, v \in F(w)$ s.t. $M = M_{ext}(u,v)$ and $x = ab(u) - ab(v)\}$ — states

$\mathcal{F}$:= $\{(M,x) \in \mathcal{S} \mid x.(1)_{\mathcal{A}} = 0\}$ — final states

$\rightarrow$ allowed S & C op. — transitions

$\mathcal{I}$:= $\mathfrak{I}/\sim_{sab}$ — initial states

Good properties :

- Given $S \in \mathcal{S}$, the number of allowed S & C operations from S is finite.
- Their image remain in $\mathcal{S}$.
- From $\mathcal{I}$ (which is **finite** !) you can reach every $S \in \mathcal{S}$.

$\longrightarrow$ **From $\mathcal{I}$, we can explore $\mathcal{S}$, and thus, $\mathcal{F}$.**

Our semi-algorithm : the "imbalances automaton"

Input : S a finite set of substitutions, $n \in \mathbb{N}$
Output : - YES if there exist w S-adic s. t. $Imb(w) \geq n$
- keep running otherwise.

It consists of :

$\longrightarrow$ A **B**readth **F**irst **S**earch of the quotient graph, starting from $\mathcal{I}$...

$\longrightarrow$ If it finds a vertex $S = (M, x)$ s. t. :

- sum of coordinates of $x = 0$ $\longleftarrow S \in \mathcal{F}$
- $||x||_\infty \geq n$

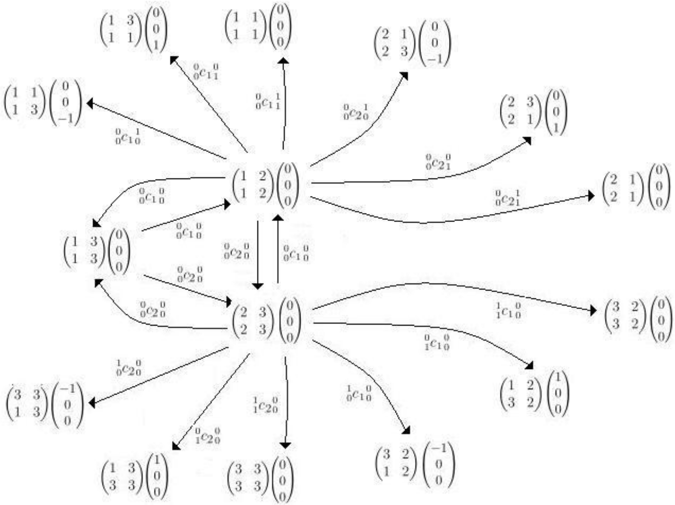
then return **YES**.

First steps of computation with Cassaigne-Selmer substitutions

$C = \{c_1, c_2\}$

$c_1 : 1 \mapsto 1,\ 2 \mapsto 13,\ 3 \mapsto 2$

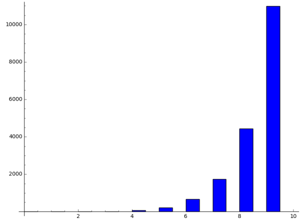
$c_2 : 1 \mapsto 2,\ 2 \mapsto 13,\ 3 \mapsto 3$



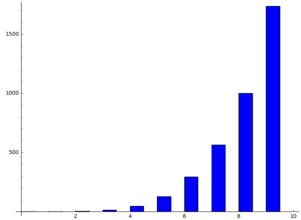
III - Applications

Experimentation (sad) reality

Problem : the tree grows too fast !



Number of vertices in function of depth



Growth after cuttings

Solution : cut branches with no hope to reach new final states...

At depth 9, among 1 000 vertices, we found the first imbalance 3...
At depth 16, among 80 000 vertices, we found the first imbalance 4...

Results for Cassaigne-Selmer words

$C = \{c_1, c_2\}$

c_1 :	1	$\mapsto$	1	c_2 :	1	$\mapsto$	2
	2	$\mapsto$	13		2	$\mapsto$	13
	3	$\mapsto$	2		3	$\mapsto$	3

Be w_0 any CS word, e.g. $c_1 \circ c_2 \circ c_1 \circ c_2 \circ ...(1)$. Consider $w_1 = c_2 \circ c_2 \circ c_2(w_0)$ and for each $n \geq 1$:

$$\begin{cases} w_{n+1} = c_1^{2n+2} \circ c_2(w_n) & \text{if } n \text{ is odd} \\ w_{n+1} = c_2^{2n+2} \circ c_1(w_n) & \text{otherwise.} \end{cases}$$

Theorem 1 : For every n, w_n is a CS word satisfying $D(w_n) \geq n$.

$\longrightarrow$ The imbalance of CS words is not bounded.

Theorem 2 : There exists a CS word with infinite imbalance.

This is a construction from $(w_n)_n$ using the following lemma :

Lemma : If w is a C-adic word s.t. $D(w) \geq 3n$, then $c_1(w)$ (resp. $c_2(w)$) is a CS word satisfying $D(w) \geq n$.

Moral

- Given S and n, a "realistic" algorithm to semi-answer the question : "$\exists w$ S-adic s.t. $Imb(w) \geq n$?"
- The imbalance automaton gives intuitions on the nature **(bounded/ unbounded)** of the set of imbalances of S-adic words - and on **rules to construct** these imbalances.
- OTHER USES :

  - behaviour with bounded partial quotient ?
  - abelian complexity
  - ...
- Difficulty : the growth speed.

- Miscellaneous questions :

  - what about sets containing erasing substitutions ?

  - what if the substitutions work on different alphabets ?

  - more generally, what if we put restrictions on the language of directives sequences ?

  - Does there exist S such that imbalances of S-adic words are finite but not bounded ?

  - Choice of an initial set $\mathcal{I}$ when the condition (H) is not satisfied ?

Thank you !