String attractors de préfixes de mots infinis

France Gheeraert

30 mai 2024

Radboud Universiteit

String attractors : rappels

Définition

Un *string attractor* pour un mot fini $w = w_1 \cdots w_n$ est un ensemble $\Gamma \subseteq \{1, \ldots, n\}$ tel que tout facteur $u \in \mathrm{Fac}(w)$ a une occurrence croisant une position dans Γ.

Exemple :
$\{3, 6, 9\}$ est un string attractor de $01\underline{0}01\underline{2}10\underline{1}0$ de taille minimale

String attractors et préfixes

Pourquoi ?

- pour étudier les mots infinis via des string attractors finis
- pour étudier une infinité de mots finis “liés”

String attractors et préfixes

Pourquoi ?

- pour étudier les mots infinis via des string attractors finis
- pour étudier une infinité de mots finis “liés”

Problèmes ?

- la taille des string attractors peut augmenter avec la taille des préfixes
- on veut une méthode/description qui convienne pour tous/une infinité de préfixes

Taille des string attractors

Fonction de profil

$\gamma^*(w)$: taille minimale d'un string attractor de w

Définition

Si $x \in \mathcal{A}^{\mathbb{N}}$, alors la *fonction de profil* de x est

$$s_x : \mathbb{N} \to \mathbb{N} \quad n \mapsto \gamma^*(x_{[1,n]}).$$

Fonction de profil bornée

Thm (Schaeffer, Shallit)

Si

- *x est ultimement périodique*
- *ou x est linéairement récur., i.e., il existe K tel que si $|u| \geq K|w|$, $u, w \in \mathrm{Fac}(x)$, alors $w \in \mathrm{Fac}(u)$*

alors s_x est borné.

Fonction de profil bornée

Thm (Schaeffer, Shallit)

Si

- *x est ultimement périodique*
- *ou x est linéairement récur.*, *i.e., il existe K tel que si $|u| \geq K|w|$, $u, w \in \mathrm{Fac}(x)$, alors $w \in \mathrm{Fac}(u)$*

alors s_x est borné.

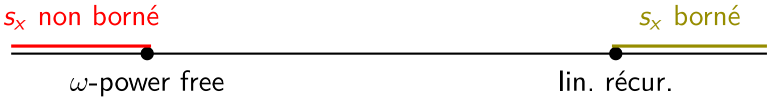
Thm (Restivo, Romana, Sciortino)

Si s_x est borné, alors

- *x est ultimement périodique*
- *ou $p_x(n) \in \Theta(n)$ et x est ω-power free*, *i.e., pour tout $w \in \mathrm{Fac}(x)$, il existe $k \in \mathbb{N}$ tel que $w^k \notin \mathrm{Fac}(x)$.*

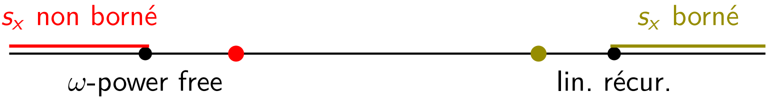
Caractérisation ?

Si $p_x(n) \in \Theta(n)$:



Caractérisation ?

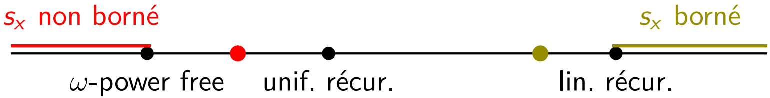
Si $p_x(n) \in \Theta(n)$:



Aucun des théorèmes n'est une caractérisation.

Caractérisation ?

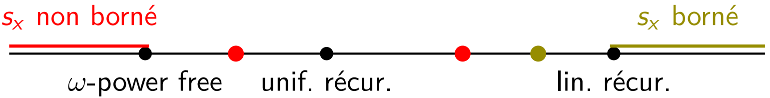
Si $p_x(n) \in \Theta(n)$:



Aucun des théorèmes n'est une caractérisation.

Caractérisation ?

Si $p_x(n) \in \Theta(n)$:



Aucun des théorèmes n'est une caractérisation.

Si $\sigma : \begin{cases} 0 \mapsto 000 \\ 1 \mapsto 101 \end{cases}$, $\mu : \begin{cases} 0 \mapsto 010 \\ 1 \mapsto 111 \end{cases}$ et

$$x = \lim_n \sigma\mu\, \sigma^2\mu^2 \cdots \sigma^n\mu^n(0),$$

alors $p_x(n) \in \Theta(n)$, x est uniformément récurrent et $s_x(n)$ n'est pas borné.

Construction explicite de string attractors

Techniques

Le but est de trouver **1** string attractor (petit) par préfixe.

Comment ?

- utiliser les propriétés combinatoires du mot infini
  → périodicité, répétitions, palindromes
- au début : 2 familles particulières de mots considérées
  → 2 grandes familles d'approches

Famille 1 : mots Sturmiens standard

Mots Sturmiens standard finis : $u_n = (u_{n-1})^{q_n} u_{n-2}$

Proposition (Mantaci, Restivo, Romana, Rosone, Sciortino)

*Si w est un mot **Sturmien standard fini**, alors w a un string attractor composé de deux positions consécutives.*

Famille 1 : mots Sturmiens standard

Mots Sturmiens standard finis : $u_n = (u_{n-1})^{q_n} u_{n-2}$

Proposition (Mantaci, Restivo, Romana, Rosone, Sciortino)

*Si w est un mot **Sturmien standard fini**, alors w a un string attractor composé de deux positions consécutives.*

Proposition (Restivo, Romana, Sciortino)

*Si w est un **préf. d'un mot Sturmien caractéristique**, alors un string attractor de w est donné par*

$$\begin{array}{rl} \{1\} & \textit{si } |w| \leq |u_2|, \\ \{|u_k|-1, |u_k|\} & \textit{si } |u_{k-1}| + |u_k| - 1 \leq |w| \leq |u_k| + |u_{k+1}| - 2. \end{array}$$

Clôture palindromique et mots episturmiens

Définition

Si w est un mot fini, alors $w^{(p)}$ est le plus court palindrome commençant par w.

Exemple : $(01020)^{(p)} = 0102010$

Clôture palindromique et mots episturmiens

Définition

Si w est un mot fini, alors $w^{(p)}$ est le plus court palindrome commençant par w.

Exemple : $(01020)^{(p)} = 0102010$

Mots episturmiens standard finis : $u_n = (u_{n-1}a_n)^{(p)}$
Mot episturmien : même language qu'un $\lim_n u_n$

Proposition (Dvořáková)

Si w est un ***préf. d'un mot episturmien****, alors w a un string attractor de taille* $\mathrm{alph}(w)$ *décrit à l'aide de préfixes palindromiques.*

Clôture anti-palindromique

Définition

Si w est un mot fini sur $\{0, 1\}$, alors $w^{(a)}$ est le plus court anti-palindrome commençant par w.

Exemple : $(0010)^{(a)} = 001011$

Clôture anti-palindromique

Définition

Si w est un mot fini sur $\{0, 1\}$, alors $w^{(a)}$ est le plus court anti-palindrome commençant par w.

Exemple : $(0010)^{(a)} = 001011$

Proposition (Dvořáková, Hendrychová)

1. *Si w est un **préf. anti-palindromique d'un mot pseudo-standard**, alors w a un string attractor de taille 3 décrit à l'aide de préfixes anti-palindromiques.*

Clôture anti-palindromique

Définition

Si w est un mot fini sur $\{0, 1\}$, alors $w^{(a)}$ est le plus court anti-palindrome commençant par w.

Exemple : $(0010)^{(a)} = 001011$

Proposition (Dvořáková, Hendrychová)

1. *Si w est un* ***préf. anti-palindromique d'un mot pseudo-standard****, alors w a un string attractor de taille 3 décrit à l'aide de préfixes anti-palindromiques.*
2. *Si w est un* ***préf. (anti-)palindromique d'un mot de Rote CS standard****, alors w a un string attractor de taille 2 décrit à l'aide de préfixes (anti-)palindromiques.*

Famille 2 : mots de Thue-Morse finis

Mots de Thue-Morse fini : $\tau^n(0)$ où $\tau(0) = 01$ et $\tau(1) = 10$

Proposition (Kutsukake, Matsumoto, Nakashima, Inenaga, Bannai, Takeda)

Si w est un mot de ***Thue-Morse fini****, alors*

$$\{2^{n-2}, 3 \cdot 2^{n-3}, 2^{n-1}, 3 \cdot 2^{n-2}\}$$

est un string attractor de w.

Utilisation de `Walnut`

Théorème (Schaeffer, Shallit)

Si x est un mot morphique “Walnut-friendly”, alors

1. *on peut décider si $s_x \leq c$, pour chaque $c \in \mathbb{N}$ fixé ;*
2. *on peut alors calculer un string attractor minimal pour chaque préfixe.*

Utilisation de `Walnut`

Théorème (Schaeffer, Shallit)

Si x est un mot morphique "Walnut-friendly", alors

1. *on peut décider si $s_x \leq c$, pour chaque $c \in \mathbb{N}$ fixé ;*
2. *on peut alors calculer un string attractor minimal pour chaque préfixe.*

Proposition (Schaeffer, Shallit)

Si x est le mot de ***period-doubling****, i.e., le point fixe de $0 \mapsto 01$, $1 \mapsto 00$, alors $s_x(n) = 2$ pour tout $n \geq 2$ et un string attractor de $x_{[1,n]}$ est donné par*

$$\begin{cases} \{3 \cdot 2^{i-3}, 3 \cdot 2^{i-2}\} & \text{si } 2^i \leq n < 3 \cdot 2^{i-1}, \\ \{2^i, 2^{i+1}\} & \text{si } 3 \cdot 2^{i-1} \leq n < 2^{i+1}. \end{cases}$$

Limitations de `Walnut`

- x doit être "`Walnut`-friendly"

Limitations de `Walnut`

- x doit être "`Walnut`-friendly"
  $\rightarrow$ OK pour les k-automatiques, moins facile pour les morphiques

Limitations de `Walnut`

- x doit être "`Walnut`-friendly"
  $\rightarrow$ OK pour les k-automatiques, moins facile pour les morphiques
- les calculs deviennent vite longs

Limitations de `Walnut`

- x doit être "`Walnut`-friendly"
  $\rightarrow$ OK pour les k-automatiques, moins facile pour les morphiques
- les calculs deviennent vite longs
  $\rightarrow$ "deviner" les string attractors et ne faire que la vérification via `Walnut`

Limitations de Walnut

- x doit être "Walnut-friendly"
  $\rightarrow$ OK pour les k-automatiques, moins facile pour les morphiques
- les calculs deviennent vite longs
  $\rightarrow$ "deviner" les string attractors et ne faire que la vérification via Walnut

Proposition (Schaeffer, Shallit)

1. *Si x est le mot de **Thue-Morse**, alors $s_x(n) = 4$ pour tout $n \geq 25$ et les string attractors utilisent des positions de la forme 2^i, $3 \cdot 2^i$ et $5 \cdot 2^i$.*

Limitations de Walnut

- x doit être "Walnut-friendly"
  → OK pour les k-automatiques, moins facile pour les morphiques
- les calculs deviennent vite longs
  → "deviner" les string attractors et ne faire que la vérification via Walnut

Proposition (Schaeffer, Shallit)

1. *Si x est le mot de* ***Thue-Morse****, alors $s_x(n) = 4$ pour tout $n \geq 25$ et les string attractors utilisent des positions de la forme 2^i, $3 \cdot 2^i$ et $5 \cdot 2^i$.*
2. *Si x est le mot de* ***Tribonacci****, alors $s_x(n) = 3$ pour tout $n \geq 4$ et les string attractors sont formés de trois nombres de Tribonacci consécutifs.*

Lien avec les systèmes de numération

Mot	Sys. de numération	Positions des s.a.
Period-doubling	base 2	$2^i, 3 \cdot 2^i$
Thue-Morse	base 2	$2^i, 3 \cdot 2^i, 5 \cdot 2^i$
Tribonacci	Tribonacci	T_i

Lien avec les systèmes de numération

Mot	Sys. de numération	Positions des s.a.
Period-doubling	base 2	$2^i, 3 \cdot 2^i$
Thue-Morse	base 2	$2^i, 3 \cdot 2^i, 5 \cdot 2^i$
Tribonacci	Tribonacci	T_i
`Walnut`-friendly	$\mathcal{S}_x$	donnés par un automate basé sur $\mathcal{S}_x$

Lien avec les systèmes de numération

Mot	Sys. de numération	Positions des s.a.
Period-doubling	base 2	$2^i, 3 \cdot 2^i$
Thue-Morse	base 2	$2^i, 3 \cdot 2^i, 5 \cdot 2^i$
Tribonacci	Tribonacci	T_i
`Walnut`-friendly	$\mathcal{S}_x$	donnés par un automate basé sur $\mathcal{S}_x$

Peut-on expliciter le lien avec les systèmes de numération pour d'autres mots morphiques ?

Mots associés aux nombres de Parry simples

Soient $c_0 \cdots c_k$ strictement maximal parmi ses conjugués (i.e., $d_\beta(1) = c_0 \cdots c_k 0^\omega$ pour un β Parry simple) et

$$\mu_{\mathbf{c}} : \begin{cases} 0 & \mapsto 0^{c_0} 1 \\ \vdots & \\ k-1 & \mapsto 0^{c_{k-1}} k \\ k & \mapsto 0^{c_k} \end{cases}$$

Proposition (G., Stipulanti, Romana)

Si w est un ***préf. de*** $\mu_{\mathbf{c}}^\omega(\mathbf{0})$*, alors w a un string attractor de taille* $\mathrm{alph}(w) + 1$ *décrit à l'aide du système de numération U_β.*

Conclusions et perspectives

- j'aurais pu parler de :
  - caractérisation des Sturmiens et quasi-Sturmiens mono-infinis,
  - identification du Sturmien caractéristique sur base des string attractors,
  - propriétés de $s_x(n)$,
  - complexité liée à l'envergure
- encore beaucoup de questions ouvertes :
  - caractérisation de $s_x(n)$ borné,
  - description de string attractors pour d'autres mots morphiques,
  - caractérisation d'autres familles de mots à l'aide de string attractors,
  - reconstruction du mot sur base des string attractors

Conclusions et perspectives

- j'aurais pu parler de :
  - caractérisation des Sturmiens et quasi-Sturmiens mono-infinis,
  - identification du Sturmien caractéristique sur base des string attractors,
  - propriétés de $s_x(n)$,
  - complexité liée à l'envergure

- encore beaucoup de questions ouvertes :
  - caractérisation de $s_x(n)$ borné,
  - description de string attractors pour d'autres mots morphiques,
  - caractérisation d'autres familles de mots à l'aide de string attractors,
  - reconstruction du mot sur base des string attractors

Merci pour votre attention !