Initiation à la combinatoire des mots - GMIN314

Gwenaël Richomme

gwenael.richomme@lirmm.fr

Cours 5

Prévisionnel des prochaines séances

- 29 octobre : Vacances d'enseignement
- 8 novembre : Pascal Ochem
- 12 novembre : Pascal Ochem
- 19 novembre : GR
- 26 novembre : Pascal Ochem
- 3 décembre : Pascal Ochem
- 10 décembre : Pascal Ochem
- 17 décembre : examen final

Retours sur exercices pour séance 3

- Attention !
  - $f^{\omega}(x) \neq f(x)^{\omega}$ (général en différent)

Retours sur exercices pour séance 3

- Attention !
  - $f^\omega(x) \neq f(x)^\omega$ (général en différent)
  - $f^\omega(a) = axf(x)f^2(x)f^3(x)\ldots$ avec $f^i(x) \neq \varepsilon$ pour tout i

Retours sur exercices pour séance 3

- Attention !
  - $f^{\omega}(x) \neq f(x)^{\omega}$ (général en différent)
  - $f^{\omega}(a) = axf(x)f^2(x)f^3(x)\dots$ avec $f^i(x) \neq \varepsilon$ pour tout i
- Chevauchement
  - Le seul mot binaire sans chevauchement généré par morphisme est le mot de Thue-Morse.

Retours sur exercices pour séance 3

- Attention !
  - $f^{\omega}(x) \neq f(x)^{\omega}$ (général en différent)
  - $f^{\omega}(a) = axf(x)f^2(x)f^3(x)\dots$ avec $f^i(x) \neq \varepsilon$ pour tout i
- Chevauchement
  - Le seul mot binaire sans chevauchement généré par morphisme est le mot de Thue-Morse.
  - Fibonacci contient des chevauchements.

Retours sur exercices pour séance 3

- Attention !
  - $f^\omega(x) \neq f(x)^\omega$ (général en différent)
  - $f^\omega(a) = axf(x)f^2(x)f^3(x)\ldots$ avec $f^i(x) \neq \varepsilon$ pour tout i
- Chevauchement
  - Le seul mot binaire sans chevauchement généré par morphisme est le mot de Thue-Morse.
  - Fibonacci contient des chevauchements.
  - Le morphisme de Fibonacci ne préserve pas l'absence de chevauchement.

Retours sur exercices pour séance 3

- Attention !
  - $f^\omega(x) \neq f(x)^\omega$ (général en différent)
  - $f^\omega(a) = axf(x)f^2(x)f^3(x)\dots$ avec $f^i(x) \neq \varepsilon$ pour tout i
- Chevauchement
  - Le seul mot binaire sans chevauchement généré par morphisme est le mot de Thue-Morse.
  - Fibonacci contient des chevauchements.
  - Le morphisme de Fibonacci ne préserve pas l'absence de chevauchement.
  - Mot sans chevauchement = mot sans $\alpha u \alpha u \alpha$

Retours sur exercices pour séance 3

- Attention !
  - $f^\omega(x) \neq f(x)^\omega$ (général en différent)
  - $f^\omega(a) = axf(x)f^2(x)f^3(x)\ldots$ avec $f^i(x) \neq \varepsilon$ pour tout i
- Chevauchement
  - Le seul mot binaire sans chevauchement généré par morphisme est le mot de Thue-Morse.
  - Fibonacci contient des chevauchements.
  - Le morphisme de Fibonacci ne préserve pas l'absence de chevauchement.
  - Mot sans chevauchement = mot sans $\alpha u \alpha u \alpha$
  - $\alpha u \alpha u \alpha \neq \alpha u \alpha u \alpha u \alpha$

Retours sur exercices pour séance 3

- Attention !
  - $f^\omega(x) \neq f(x)^\omega$ (général en différent)
  - $f^\omega(a) = axf(x)f^2(x)f^3(x)\ldots$ avec $f^i(x) \neq \varepsilon$ pour tout i
- Chevauchement
  - Le seul mot binaire sans chevauchement généré par morphisme est le mot de Thue-Morse.
  - Fibonacci contient des chevauchements.
  - Le morphisme de Fibonacci ne préserve pas l'absence de chevauchement.
  - Mot sans chevauchement = mot sans $\alpha u \alpha u \alpha$
  - $\alpha u \alpha u \alpha \neq \alpha u \alpha u \alpha u \alpha$
  - φ ne conserve pas l'absence de $\alpha u \alpha u \alpha u \alpha$

Retours sur exercices pour séance 3

- Attention !
  - $f^\omega(x) \neq f(x)^\omega$ (général en différent)
  - $f^\omega(a) = axf(x)f^2(x)f^3(x)\dots$ avec $f^i(x) \neq \varepsilon$ pour tout i
- Chevauchement
  - Le seul mot binaire sans chevauchement généré par morphisme est le mot de Thue-Morse.
  - Fibonacci contient des chevauchements.
  - Le morphisme de Fibonacci ne préserve pas l'absence de chevauchement.
  - Mot sans chevauchement = mot sans $\alpha u \alpha u \alpha$
  - $\alpha u \alpha u \alpha \neq \alpha u \alpha u \alpha u \alpha$
  - φ ne conserve pas l'absence de $\alpha u \alpha u \alpha u \alpha$
- Quand vous bloquez sur une question, pensez à regarder les suivantes !

Correction des exercices à faire pour la séance 3

Exercice

Montrer que le mot de Fibonacci ne contient aucun facteur de la forme

$$u\alpha u\alpha u\alpha u$$

avec u mot et α lettre

Correction des exercices à faire pour la séance 3

Exercice

Montrer que le mot de Fibonacci ne contient aucun facteur de la forme

$$u\alpha u\alpha u\alpha u$$

avec u mot et α lettre

Remarques :
Énoncé traité (mais pas dans transparents) : $\alpha u\alpha u\alpha u\alpha$

Correction des exercices à faire pour la séance 3

Exercice

Montrer que le mot de Fibonacci ne contient aucun facteur de la forme

$$u\alpha u\alpha u\alpha u$$

avec u mot et α lettre

Remarques :
Énoncé traité (mais pas dans transparents) : $\alpha u\alpha u\alpha u\alpha$
C'est faux dans ce cas !

$$\mathbf{F} = abaababaabaababaababaabaababaabaa$$

Correction des exercices à faire pour la séance 3 (suite)

Exercice

Quel est le plus grand entier k tel qu'il existe un facteur de la forme u^k dans le mot infini de Fibonacci ?

Correction des exercices à faire pour la séance 3 (suite)

Exercice

Quel est le plus grand entier k tel qu'il existe un facteur de la forme u^k dans le mot infini de Fibonacci ?

Exercice

Quel est le plus grand entier k tel qu'il existe un préfixe de la forme u^k dans le mot infini de Fibonacci ?

Correction des exercices à faire pour la séance 3 (suite)

Exercice

Quel est le plus grand entier k tel qu'il existe un facteur de la forme u^k dans le mot infini de Fibonacci ?

Exercice

Quel est le plus grand entier k tel qu'il existe un préfixe de la forme u^k dans le mot infini de Fibonacci ?

A faire pour la séance 6

Corrections d'exercices

Montrer

Si $w = L_a(w')$ avec w infini alors
w équilibré $\Leftrightarrow$ w' équilibré

Montrer

Si $w = L_a(w')$ avec w infini alors
w équilibré $\Leftrightarrow$ w' équilibré

Montrer

Si $w = R_a(bw')$ avec w infini alors
w équilibré $\Leftrightarrow$ bw' équilibré

Exercices

Montrer que si $\mathbf{w}$ est équilibré et périodique, alors $\mathbf{w} = f(a^\omega)$ ou $\mathbf{w} = f(b^\omega)$ avec $f \in \{L_a, L_b, R_a, R_b\}^*$.

Exercices

Montrer que si $\mathbf{w}$ est équilibré et périodique, alors $\mathbf{w} = f(a^\omega)$ ou $\mathbf{w} = f(b^\omega)$ avec $f \in \{L_a, L_b, R_a, R_b\}^*$.

Montrer que si $\mathbf{w}$ est équilibré et ultimement périodique mais non périodique, alors $\mathbf{w} = f(ab^\omega)$ ou $\mathbf{w} = f(ba^\omega)$ avec $f \in \{L_a, L_b, R_a, R_b\}^*$.

Exercices

Montrer que si $\mathbf{w}$ est équilibré et périodique, alors $\mathbf{w} = f(a^\omega)$ ou $\mathbf{w} = f(b^\omega)$ avec $f \in \{L_a, L_b, R_a, R_b\}^*$.

Montrer que si $\mathbf{w}$ est équilibré et ultimement périodique mais non périodique, alors $\mathbf{w} = f(ab^\omega)$ ou $\mathbf{w} = f(ba^\omega)$ avec $f \in \{L_a, L_b, R_a, R_b\}^*$.

Montrer que si $\mathbf{w}$ est équilibré et apériodique (non ultimement périodique), alors dans toute décomposition infinie de $\mathbf{w}$ sur $\{L_a, L_b, R_a, R_b\}$, les morphismes L_a ou R_a apparaissent infiniment souvent, et, les morphismes L_b ou R_b apparaissent infiniment souvent.

Périodicité (rappel de la dernière fois)

Montrer (lemme 8.2) corrigé

Pour $\mathbf{w}$ un mot infini sur $\{a, b\}$:

- Si w est périodique alors $f(\mathbf{w})$ est périodique pour tout morphisme f non effaçant ;
- Si $L_a(\mathbf{w})$ ou $L_b(\mathbf{w})$ est périodique alors $\mathbf{w}$ est périodique ;
- Si $R_a(b\mathbf{w})$ (resp. $L_b(a\mathbf{w})$) est périodique alors $b\mathbf{w}$ (resp. $a\mathbf{w}$) est périodique.

Mots sturmiens

Theoreme 8.8

Les assertions suivantes sont équivalentes.

1. Un mot $\mathbf{w}$ est sturmien.
2. Il existe une suite infinie de mots $(\mathbf{w}_n)_{n\geq 0}$, et une suite infinie de morphismes $(\varphi_n)_{n\geq 0} \in \{L_a, L_b, R_a, R_b\}$ telles que :
   1. $\mathbf{w}_0 = \mathbf{w}$,
   2. $\mathbf{w}_n = \varphi_n(\mathbf{w}_{n+1})$ pour tout entier $n \geq 0$,
   3. un des morphismes L_a et R_a apparaît infiniment souvent, et un des morphismes L_b ou R_b apparaît infiniment souvent,
   4. pour un entier n et une lettre α, si $\varphi_n = R_\alpha$, alors $\varphi_{n+1} \neq L_\alpha$.
3. Il existe deux suites d'entiers $(a_n)_{n\geq 0}$ et $(c_n)_{n\geq 0}$ telles que :
   1. $a_0 \geq 0$ et $a_n \geq 1$ pour $n \geq 1$,
   2. $0 \leq c_n \leq a_n$ pour $n \geq 0$,
   3. $\mathbf{w} = \lim_{n\to\infty} L_a^{c_0} R_a^{a_0 - c_0} L_b^{c_1} R_b^{a_1 - c_1} \cdots L_a^{c_{2n}} R_a^{a_{2n} - c_{2n}} L_b^{c_{2n+1}} R_b^{a_{2n+1} - c_{2n+1}}(a)$

Mots finis équilibrés et désubstitution

Objectif

Algorithme linéaire en temps pour tester si un mot fini est équilibré

Mots finis équilibrés et désubstitution

Objectif

Algorithme linéaire en temps pour tester si un mot fini est équilibré

Remarque

w équilibré $\Rightarrow \exists n \geq 0$,

entre 2 occurrences successives de b, a^n ou a^{n+1}

Mots finis équilibrés et désubstitution

Objectif

Algorithme linéaire en temps pour tester si un mot fini est équilibré

Remarque

w équilibré $\Rightarrow \exists n \geq 0$,

entre 2 occurrences successives de b, a^n ou a^{n+1}

Si $|w|_b \geq 1$, $w = SA_1A_2 \ldots A_kP$ avec

$$\begin{cases} k \geq 0 \\ S \in \bigcup_{\ell=0}^{n+1} a^\ell, \\ A_i \in \{ba^n, ba^{n+1}\} \ (1 \leq i \leq k) \\ P \in \bigcup_{\ell=0}^{n+1} ba^\ell \end{cases}$$

Reduction

Si $|w|_b \geq 1$, $w = SA_1A_2\dots A_kP$ avec

$$\begin{cases} k \geq 0 \\ S \in \bigcup_{\ell=0}^{n+1} a^\ell, \\ A_i \in \{ba^n, ba^{n+1}\}\ (1 \leq i \leq k) \\ P \in \bigcup_{\ell=0}^{n+1} ba^\ell \end{cases}$$

Définition

$reduced_n(w) = sa_1a_2\dots a_kp$ such that :

$$\begin{cases} s = \varepsilon \text{ if } S \neq a^{n+1}, \text{ else } s = b, \\ a_i = a \text{ if } A_i = ba^n, \\ a_i = b \text{ if } A_i = ba^{n+1}\ (1 \leq i \leq k), \\ p = \varepsilon \text{ if } P \neq ba^{n+1}, \text{ else } p = b. \end{cases}$$

Réduction et morphismes

Posons $r = reduced_n(w) = s a_1 a_2 \ldots a_k p$:

$$\begin{cases} s = \varepsilon \text{ if } S \neq a^{n+1}, \text{ else } s = b, \\ a_i = a \text{ if } A_i = ba^n, \\ a_i = b \text{ if } A_i = ba^{n+1} \ (1 \leq i \leq k), \\ p = \varepsilon \text{ if } P \neq ba^{n+1}, \text{ else } p = b. \end{cases}$$

Définition

- E : exchange morphim $a \mapsto b$, $b \mapsto a$
- $h_n = R_a^n \circ L_b \circ E \begin{cases} h_n(a) = ba^n \\ h_n(b) = ba^{n+1} \end{cases}$

$$w = \begin{cases} Sh_n(r)P & \text{si } s = \varepsilon, \ p = \varepsilon \\ b^{-1}Sh_n(r)P & \text{si } s \neq \varepsilon, \ p = \varepsilon \\ Sh_n(r) & \text{si } s = \varepsilon, \ p \neq \varepsilon \\ b^{-1}Sh_n(r) & \text{si } s \neq \varepsilon, \ p \neq \varepsilon \end{cases}$$

Principe actif de l'algo

Lemme

Let w be a word over A containing at least one b. The following are equivalent :

1. w is balanced
2. w is n-reducible (for an integer $n \geq 0$) and $reduced_n(w)$ is balanced.

Principe actif de l'algo

Lemme

Let w be a word over A containing at least one b. The following are equivalent :

1. w is balanced
2. w is n-reducible (for an integer $n \geq 0$) and $reduced_n(w)$ is balanced.

Preuve similaire aux résultats précédemment vus

Algo

If $|w|_a \leq 1$ or $|w|_b \leq 1$, **Then** w is balanced
Else

  If w or $E(w)$ is not reducible,
  Then w is not balanced
  Else

    Let n be the greater integer such that w is n-reducible
    Let m be the greater integer such that $E(w)$ is m-reducible
    If $n \geq 1$ **Then** let $w' = reduced_n(w)$
      Else let $w' = reduced_m(E(w))$
    w is balanced iff w' is balanced.

Algo

If $|w|_a \leq 1$ or $|w|_b \leq 1$, **Then** w is balanced
Else
  If w or $E(w)$ is not reducible,
  Then w is not balanced
  Else
    Let n be the greater integer such that w is n-reducible
    Let m be the greater integer such that $E(w)$ is m-reducible
    If $n \geq 1$ **Then** let $w' = reduced_n(w)$
      Else let $w' = reduced_m(E(w))$
    w is balanced iff w' is balanced.

Exemples :

- *abababbabaab*

Algo

If $|w|_a \leq 1$ or $|w|_b \leq 1$, **Then** w is balanced
Else
  If w or $E(w)$ is not reducible,
  Then w is not balanced
  Else
    Let n be the greater integer such that w is n-reducible
    Let m be the greater integer such that $E(w)$ is m-reducible
    If $n \geq 1$ **Then** let $w' = reduced_n(w)$
      Else let $w' = reduced_m(E(w))$
    w is balanced iff w' is balanced.

Exemples :

- *abababbabaab* non équilibré
- *aabaababaabaababaababaabab*

Algo

If $|w|_a \leq 1$ or $|w|_b \leq 1$, **Then** w is balanced
Else
  If w or $E(w)$ is not reducible,
  Then w is not balanced
  Else
    Let n be the greater integer such that w is n-reducible
    Let m be the greater integer such that $E(w)$ is m-reducible
    If $n \geq 1$ **Then** let $w' = reduced_n(w)$
      Else let $w' = reduced_m(E(w))$
    w is balanced iff w' is balanced.

Exemples :

- *abababbabaab* non équilibré
- *aabaababaabaababaababaabab* non équilibré
- *abaababaabaababaababa*

Algo

If $|w|_a \leq 1$ or $|w|_b \leq 1$, **Then** w is balanced
Else
  If w or $E(w)$ is not reducible,
  Then w is not balanced
  Else
    Let n be the greater integer such that w is n-reducible
    Let m be the greater integer such that $E(w)$ is m-reducible
    If $n \geq 1$ **Then** let $w' = \mathit{reduced}_n(w)$
      Else let $w' = \mathit{reduced}_m(E(w))$
    w is balanced iff w' is balanced.

Exemples :

- *abababbabaab* non équilibré
- *aabaababaabaababaababaabab* non équilibré
- *abaababaabaababaababa* équilibré

Complexité

Preuve complexité linéaire en temps :

- une étape récursive en temps $O(|w|)$
  Soit λ tel que temps $\leq \lambda|w|$

Complexité

Preuve complexité linéaire en temps :

- une étape récursive en temps $O(|w|)$
  Soit λ tel que temps $\leq \lambda|w|$
- If $n \geq 1$, $|reduced_n(w)| \leq \frac{|w|}{2}$

Complexité

Preuve complexité linéaire en temps :

- une étape récursive en temps $O(|w|)$
  Soit λ tel que temps $\leq \lambda|w|$
- If $n \geq 1$, $|reduced_n(w)| \leq \frac{|w|}{2}$
- if $2^{k-1} < |w| \leq 2^k$,
  complexité en temps borné par
  $\sum_{i=0}^{k} \lambda 2^i = \lambda(2^{k+1} - 1) < 4\lambda|w|$

Complexité

Preuve complexité linéaire en temps :

- une étape récursive en temps $O(|w|)$
  Soit λ tel que temps $\leq \lambda|w|$
- If $n \geq 1$, $|reduced_n(w)| \leq \frac{|w|}{2}$
- if $2^{k-1} < |w| \leq 2^k$,
  complexité en temps borné par
  $\sum_{i=0}^{k} \lambda 2^i = \lambda(2^{k+1} - 1) < 4\lambda|w|$
- Complexité en $O(|w|)$.

Au passage

- tout mot fini équilibré est facteur d'un mot de la forme

$$h_{n_1} h_{n_2} \dots h_{n_i}(\alpha^j)$$

avec α lettre

Au passage

- tout mot fini équilibré est facteur d'un mot de la forme

$$h_{n_1} h_{n_2} \ldots h_{n_i}(\alpha^j)$$

avec α lettre

- tout mot fini équilibré est facteur d'un mot infini équilibré apériodique

Au passage

- tout mot fini équilibré est facteur d'un mot de la forme

$$h_{n_1} h_{n_2} \dots h_{n_i}(\alpha^j)$$

  avec α lettre

- tout mot fini équilibré est facteur d'un mot infini équilibré apériodique

- tout mot fini équilibré se prolonge en un mot équilibré plus long

Complexité en facteur

Objectif

Montrer que :

- le nombre de facteurs de longueur n est $n+1$ pour tout mot
- cela est caractéristique

Complexité en facteur

Objectif

Montrer que :

- le nombre de facteurs de longueur n est $n+1$ pour tout mot
- cela est caractéristique

Définition

Pour $\mathbf{w}$ mot infini, fonction de complexité en facteurs =
complexité = complexité en facteurs =
$p_{\mathbf{w}}(n)$ = fonction qui associe n, le nombre de facteurs de longueur n

Facteurs spéciaux droits

Définition

Un mot u est un facteur spécial droit de **w** si :
$\exists \alpha, \beta$ lettres, $\alpha \neq \beta$ et
$u\alpha$, $u\beta$ facteurs de **w**

Facteurs spéciaux droits

Définition

Un mot u est un facteur spécial droit de $\mathbf{w}$ si :
$\exists \alpha, \beta$ lettres, $\alpha \neq \beta$ et
$u\alpha$, $u\beta$ facteurs de $\mathbf{w}$

Pour des mots binaires infini à droite :
$p_{\mathbf{w}}(n+1) = p_{\mathbf{w}}(n)$ + nombres facteurs spéciaux droits longueur n

Facteurs spéciaux droits

Définition

Un mot u est un facteur spécial droit de $\mathbf{w}$ si :
$\exists \alpha, \beta$ lettres, $\alpha \neq \beta$ et
$u\alpha$, $u\beta$ facteurs de $\mathbf{w}$

Pour des mots binaires infini à droite :
$p_{\mathbf{w}}(n+1) = p_{\mathbf{w}}(n)$ + nombres facteurs spéciaux droits longueur n

Conséquence : $p_{\mathbf{w}}$ est croissante
(strictement croissante pour un non ultimement périodique)

Facteurs spéciaux droits

Définition

Un mot u est un facteur spécial droit de $\mathbf{w}$ si :
$\exists \alpha, \beta$ lettres, $\alpha \neq \beta$ et
$u\alpha$, $u\beta$ facteurs de $\mathbf{w}$

Pour des mots binaires infini à droite :
$p_{\mathbf{w}}(n+1) = p_{\mathbf{w}}(n)$ + nombres facteurs spéciaux droits longueur n

Conséquence : $p_{\mathbf{w}}$ est croissante
(strictement croissante pour un non ultimement périodique)

Se généralise à des alphabets quelconques

Facteurs spéciaux gauches

Définition

Un mot u est un facteur spécial gauche de **w** si :
$\exists \alpha, \beta$ lettres, $\alpha \neq \beta$ et
$u\alpha$, $u\beta$ facteurs de **w**

Facteurs spéciaux gauches

Définition

Un mot u est un facteur spécial gauche de $\mathbf{w}$ si :
$\exists \alpha, \beta$ lettres, $\alpha \neq \beta$ et
$u\alpha$, $u\beta$ facteurs de $\mathbf{w}$

Pour des mots binaires infini à droite récurrents :
$p_{\mathbf{w}}(n+1) = p_{\mathbf{w}}(n)$ + nb facteurs spéciaux gauches longueur n

Facteurs spéciaux gauches

Définition

Un mot u est un facteur spécial gauche de $\mathbf{w}$ si :
$\exists \alpha, \beta$ lettres, $\alpha \neq \beta$ et
$u\alpha$, $u\beta$ facteurs de $\mathbf{w}$

Pour des mots binaires infini à droite récurrents :
$p_{\mathbf{w}}(n+1) = p_{\mathbf{w}}(n)$ + nb facteurs spéciaux gauches longueur n

Se généralise à des alphabets quelconques

Facteurs spéciaux gauches

Définition

Un mot u est un facteur spécial gauche de $\mathbf{w}$ si :
$\exists \alpha, \beta$ lettres, $\alpha \neq \beta$ et
$u\alpha$, $u\beta$ facteurs de $\mathbf{w}$

Pour des mots binaires infini à droite récurrents :
$p_{\mathbf{w}}(n+1) = p_{\mathbf{w}}(n)$ + nb facteurs spéciaux gauches longueur n

Se généralise à des alphabets quelconques

Mots bispéciaux

= Spéciaux à gauche et à droite

Theoreme 8.15

Sont équivalents pour un mot infini $\mathbf{w}$ sur $\{a, b\}$:

1. $\mathbf{w}$ est sturmien ;
2. $\mathbf{w}$ a exactement un mot spécial droit de chaque longueur ;
3. $p_{\mathbf{w}}(n) = n + 1$ pour tout entier $n \geq 0$.

Theoreme 8.15

Sont équivalents pour un mot infini $\mathbf{w}$ sur $\{a, b\}$:

1. $\mathbf{w}$ est sturmien ;
2. $\mathbf{w}$ a exactement un mot spécial droit de chaque longueur ;
3. $p_{\mathbf{w}}(n) = n + 1$ pour tout entier $n \geq 0$.

Preuve ?

C'est un exemple d'extrémalité des mots sturmiens !

Lemme 8.12 [Morse Hedlund 1940]

Sont équivalents pour un mot $\mathbf{w}$ infini :

1. $\mathbf{w}$ est ultimement périodique ;
2. la fonction de complexité de $\mathbf{w}$ est bornée ;
3. il existe n tel que $p_{\mathbf{w}}(n) \leq n$;
4. il existe n tel que $p_{\mathbf{w}}(n) = p_{\mathbf{w}}(n+1)$;
5. il existe n_0 tel que pour tout entier $n \geq n_0$, $p_{\mathbf{w}}(n) = p_{\mathbf{w}}(n+1)$.

Quelques autres résultats classiques

Remarque : beaucoup de survols de la notion de complexité.
Le dernier en date : Cassaigne, Nicolas, "Factor complexity", dans livre "Combinatorics, automata, and number theory" édité par V. Berthé et M. Rigo 2010

Quelques autres résultats classiques

Remarque : beaucoup de survols de la notion de complexité.
Le dernier en date : Cassaigne, Nicolas, "Factor complexity", dans livre "Combinatorics, automata, and number theory" édité par V. Berthé et M. Rigo 2010

Pansiot 1984

La complexité d'un mot infini engendré par morphisme est en $\Theta(1)$, $\Theta(n)$, $\Theta(nloglogn)$, $\Theta(nlogn)$ ou $\Theta(n^2)$.

Quelques autres résultats classiques

Remarque : beaucoup de survols de la notion de complexité.
Le dernier en date : Cassaigne, Nicolas, "Factor complexity", dans livre "Combinatorics, automata, and number theory" édité par V. Berthé et M. Rigo 2010

Pansiot 1984

La complexité d'un mot infini engendré par morphisme est en $\Theta(1)$, $\Theta(n)$, $\Theta(nloglogn)$, $\Theta(nlogn)$ ou $\Theta(n^2)$.

Pour les mots morphiques, pas de caractérisation complète (mais complexité en $O(n^2)$).

Problème ouvert

Conjecture S-adic

Soit S un ensemble fini de morphismes.
Il existe une condition sur S tel que tout mot S-adic est de complexité linéaire.

Récemment :
la condition devrait aussi porter sur la suite infinie de décomposition.

Et sur 3 lettres

Équilibré apériodique [Hubert 2000]

Soit $k \geq 3$.
Un mot infini apériodique $\mathbf{w}$ sur k lettres est apériodique
si et seulement si

- l'alphabet peut être partitionné en deux sous-alphabets A et B
- la projection sur A est $(a_1 \dots a_{\#A})^\omega$ avec $\{a_1, \dots, a_{\#A}\} = A$
- la projection sur B est $(b_1 \dots b_{\#B})^\omega$ avec $\{b_1, \dots, b_{\#A}\} = B$
- l'image de $\mathbf{w}$ par $a_i \mapsto a$, $b_i \mapsto b$ $(a \neq b)$ est un mot sturmien

Et sur 3 lettres (suite)

Conjecture de Fraenkel

Soit $k \geq 3$. Un mot infini périodique équilibré $\mathbf{w}$ sur k lettres est un suffixe de $(Fr_k)^\omega$ où

- $Fr_1 = 1$
- $Fr_n = Fr_{n-1} n Fr_n$ pour $n \geq 2$

Résolue pour $k = 3$ à 7