

Énumération et numération

Victor Marsault

Telecom-ParisTech

Soutenance de thèse
2016-03-01

- 1 Qu'est-ce qu'un système de numération
- 2 Sur la base rationnelle
- 3 Sur la notion de signature
- 4 Sur la base entière
- 5 Perspectives

Nombres

- Quantités abstraites
- Liés aux mathématiques

Ex: 8, $\sqrt{2}$, π , etc.

Représentation

Mots

- Suite de lettres/chiffres
- Liés à l'informatique

Ex: *abbac*, 3.141592... ,

Évaluation

Alphabet

Les chiffres autorisés.

Une fonction d'évaluation

- mot \rightarrow nombre
- la valeur d'un mot u est notée $\pi(u)$

Une fonction ou algorithme de représentation

- nombre \rightarrow mot
- la représentation d'un nombre n est noté $\langle n \rangle$

Alphabet

$\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$

Alphabet

{0, 1, 2, 3, 4, 5, 6, 7, 8, 9}

Représentation

 → 19 (le chiffre 1 suivi du chiffre 9)

Alphabet

{0, 1, 2, 3, 4, 5, 6, 7, 8, 9}

Représentation

 → 19 (le chiffre 1 suivi du chiffre 9)

Évaluation

2 3 5 →

Alphabet

{0, 1, 2, 3, 4, 5, 6, 7, 8, 9}

Représentation

 → 19 (le chiffre 1 suivi du chiffre 9)

Évaluation

2 3 5 →

o o o o o

Alphabet

{0, 1, 2, 3, 4, 5, 6, 7, 8, 9}

Représentation

 → 19 (le chiffre 1 suivi du chiffre 9)

Évaluation

2 3 5 → 

Alphabet

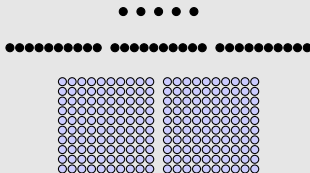
{0, 1, 2, 3, 4, 5, 6, 7, 8, 9}

Représentation

 → 19 (le chiffre 1 suivi du chiffre 9)

Évaluation

2 3 5 →



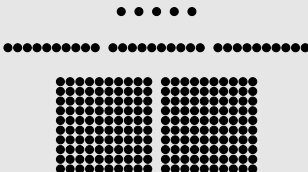
Alphabet

{0, 1, 2, 3, 4, 5, 6, 7, 8, 9}

Représentation

 → 19 (le chiffre 1 suivi du chiffre 9)

Évaluation

2 3 5 → 

Mot

Un *mot* sur un alphabet A est une suite de lettres/chiffres de A .

Exemples: 19 ; 01032016 .

Mot

Un *mot* sur un alphabet A est une suite de lettres/chiffres de A .

Exemples: 19 ; 01032016 .

Langage

Un langage sur un alphabet A est un ensemble de mots.

- Exemples:
- les mots dont la valeur est divisible par 7 :
 $\{7, 14, 21, 28, 35, \dots\}$;
 - les mots dont la valeur est une puissance de 10 :
 $\{10, 100, 1000, 10000, \dots\}$.

Mot

Un *mot* sur un alphabet A est une suite de lettres/chiffres de A .

Exemples: 19 ; 01032016 .

Langage

Un langage sur un alphabet A est un ensemble de mots.

Exemples:

- les mots dont la valeur est divisible par 7 :
 $\{7, 14, 21, 28, 35, \dots\}$;
- les mots dont la valeur est une puissance de 10 :
 $\{10, 100, 1000, 10000, \dots\}$.

Un système de numération fait correspondre un langage et un ensemble de nombres.

1 Qu'est-ce qu'un système de numération

2 Sur la base rationnelle

- ▶ Compter en base rationnelle
- ▶ Le langage $L_{\frac{p}{q}}$
- ▶ La notion de langage FLIP
- ▶ La fonction successeur sur les mots minimaux

3 Sur la notion de signature

4 Sur la base entière

5 Perspectives

	milliers	centaines	dizaines	unités
	1	9	8	8
+			1	2
+				2
<hr/>				

	milliers	centaines	dizaines	unités
	1	9	8	8
+			1	2
+				2
<hr/>				
	1	9	9	12

	milliers	centaines	dizaines	unités
	1	9	8	8
+			1	2
+				2
<hr/>				
	1	9	9	12
			+1	-10
			10	2

	milliers	centaines	dizaines	unités
	1	9	8	8
+			1	2
+				2
<hr/>				
	1	9	9	12
		9	10	2

Diagram illustrating the addition process in base 10 with carrying (retenue). The columns represent thousands (milliers), hundreds (centaines), tens (dizaines), and units (unités). The numbers being added are 1988, 12, and 2. The sum is 2002. The process shows that 8 + 2 + 2 = 12 in the units column, which is written as 2 with a carry of 1 to the tens column. Then, 9 + 1 + 1 = 10 in the tens column, which is written as 0 with a carry of 1 to the hundreds column. Finally, 9 + 1 = 10 in the hundreds column, which is written as 0 with a carry of 1 to the thousands column. The final result is 2002.

	milliers	centaines	dizaines	unités
	1	9	8	8
+			1	2
+				2
<hr/>				
	1	9	9	12
		9	10	2
		10	0	

The diagram illustrates the carrying process in base 10 addition. It shows the addition of three numbers: 1988, 12, and 2. The result is shown in two rows below the horizontal line. The first row shows the initial sum: 19912. The second row shows the result after carrying: 19102. Red arrows indicate the flow of values: from 12 to 10 and 2, and from 10 to 10 and 0. Green labels '+1' and '-10' indicate the adjustments made during the carry process.

	milliers	centaines	dizaines	unités
	1	9	8	8
+			1	2
+				2
<hr style="border: 1px solid black;"/>				
	1	9	9	12
		9	10	2
	1	10	0	
	2	0		

The diagram illustrates the carrying process in base 10. Red arrows show the flow of values from right to left. Green annotations '+1' and '-10' indicate the carry and the subtraction of 10 from the current column's sum.

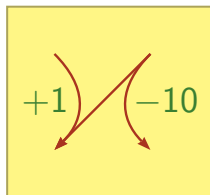
- From the units column (12), a red arrow points to the tens column with '+1', and another red arrow points to the units column with '-10'.
- From the tens column (10), a red arrow points to the hundreds column with '+1', and another red arrow points to the tens column with '-10'.
- From the hundreds column (10), a red arrow points to the thousands column with '+1', and another red arrow points to the hundreds column with '-10'.

	milliers	centaines	dizaines	unités
	1	9	8	8
+			1	2
+				2
<hr style="border: 1px solid black;"/>				
	1	9	9	12
		9	10	2
	1	10	0	
	2	0	0	2

Diagram illustrating the carrying process in base 10 addition. Red arrows show the flow of values from right to left, with green annotations indicating the carry (+1) and the subtraction of 10 (-10) from the current column.

	milliers	centaines	dizaines	unités
	1	9	8	8
+			1	2
+				2
<hr/>				
	1	9	9	12
		9	10	2
	1	10	0	
	2	0	0	2

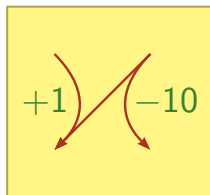
Diagram illustrating the carrying process in base 10. Red arrows show the flow of values between columns. Green annotations indicate the adjustments: $+1$ (adding 1 to the next higher column) and -10 (subtracting 10 from the current column). The final result is circled at the bottom: 2 0 0 2.



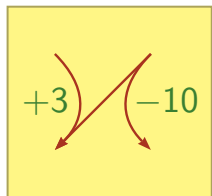
	milliers	centaines	dizaines	unités
	1	9	8	8
+			1	2
+				2
<hr style="border: 1px solid black;"/>				
	1	9	9	12
		9	10	2
	1	10	0	
	2	0	0	2

Diagram illustrating the addition process in base 10 with carrying (retenue). The numbers are aligned by place value (milliers, centaines, dizaines, unités). The sum is calculated from right to left, showing the carry-over process. Red arrows indicate the carry-over of 1 unit from the units column to the tens column, and 1 unit from the tens column to the hundreds column. Green annotations show the carry-over of 1 unit and the subtraction of 10 units from the current column.

Base $\frac{10}{1}$



Base $\frac{10}{3}$



La retenue en base $\frac{10}{3}$

 $\frac{1000}{27}$

-aines

 $\frac{100}{9}$

-aines

dix-tiers-aines

unités

.....

.....

.....

.....

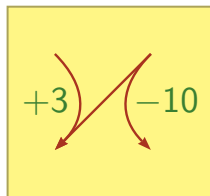
Base $\frac{10}{3}$

$+3$ -10

La retenue en base $\frac{10}{3}$

	$\frac{1000}{27}$ -aines	$\frac{100}{9}$ -aines	dix-tiers-aines	unités
	1	9	8	8
+			1	2
+				2
<hr/>				
	1	9	9	12

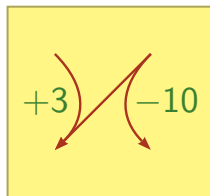
Base $\frac{10}{3}$



La retenue en base $\frac{10}{3}$

	$\frac{1000}{27}$ -aines	$\frac{100}{9}$ -aines	dix-tiers-aines	unités
	1	9	8	8
+			1	2
+				2
<hr/>				
	1	9	9	12
			+3	-10
			12	2

Base $\frac{10}{3}$

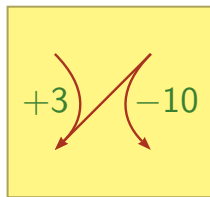


La retenue en base $\frac{10}{3}$

	$\frac{1000}{27}$ -aines	$\frac{100}{9}$ -aines	dix-tiers-aines	unités
	1	9	8	8
+			1	2
+				2
<hr style="border: 1px solid black;"/>				
	1	9	9	12
		9	12	2
		12	2	

+3 (from 8 to 9 in the 1000/27-aines column)
-10 (from 8 to 2 in the 100/9-aines column)
+3 (from 9 to 12 in the dix-tiers-aines column)
-10 (from 9 to 2 in the unités column)

Base $\frac{10}{3}$

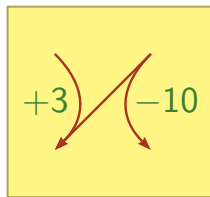


La retenue en base $\frac{10}{3}$

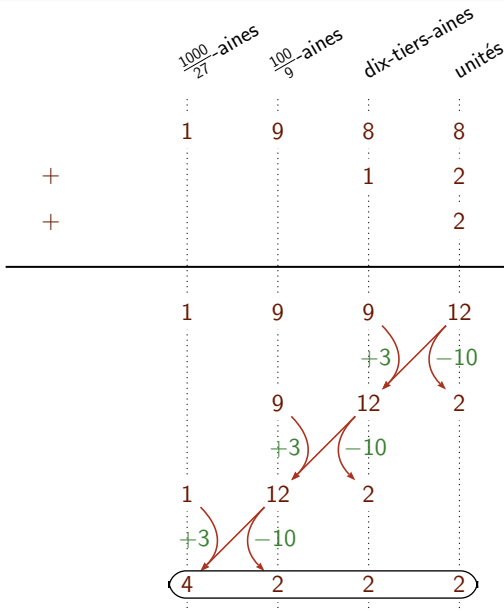
	$\frac{1000}{27}$ -aines	$\frac{100}{9}$ -aines	dix-tiers-aines	unités
	1	9	8	8
+			1	2
+				2
<hr style="border: 1px solid black;"/>				
	1	9	9	12
		9	12	2
	1	12	2	
	4	2		

Diagram illustrating the addition process in base $\frac{10}{3}$. Red arrows show the carry propagation from right to left. Green annotations indicate the carry values: $+3$ (downward) and -10 (upward).

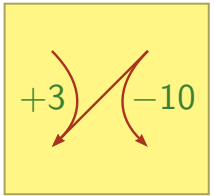
Base $\frac{10}{3}$



La retenue en base $\frac{10}{3}$



Base $\frac{10}{3}$



Calculer la représentation de 42 en base $\frac{10}{3}$

$\frac{1000}{27}$ -aines

$\frac{100}{9}$ -aines

dix-tiers-aines

unités

.....

.....

.....

.....

Calculer la représentation de 42 en base $\frac{10}{3}$

 $\frac{1000}{27}$ -aines $\frac{100}{9}$ -aines

dix-tiers-aines

unités

42

Calculer la représentation de 42 en base $\frac{10}{3}$

 $\frac{1000}{27}$ -aines $\frac{100}{9}$ -aines

dix-tiers-aines

unités

0

42

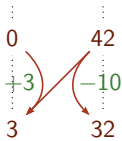
Calculer la représentation de 42 en base $\frac{10}{3}$

$\frac{1000}{27}$ -aines

$\frac{100}{9}$ -aines

dix-tiers-aines

unités



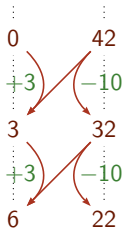
Calculer la représentation de 42 en base $\frac{10}{3}$

$\frac{1000}{27}$ -aines

$\frac{100}{9}$ -aines

dix-tiers-aines

unités



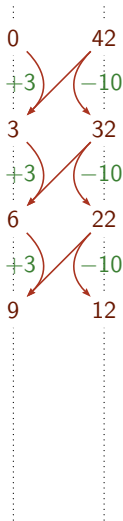
Calculer la représentation de 42 en base $\frac{10}{3}$

$\frac{1000}{27}$ -aines

$\frac{100}{9}$ -aines

dix-tiers-aines

unités



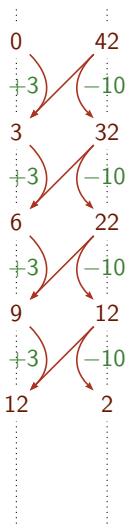
Calculer la représentation de 42 en base $\frac{10}{3}$

$\frac{1000}{27}$ -aines

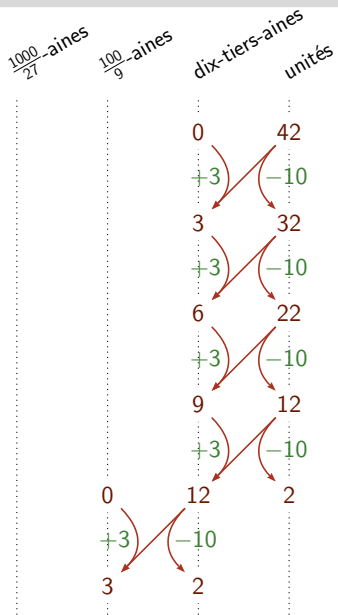
$\frac{100}{9}$ -aines

dix-tiers-aines

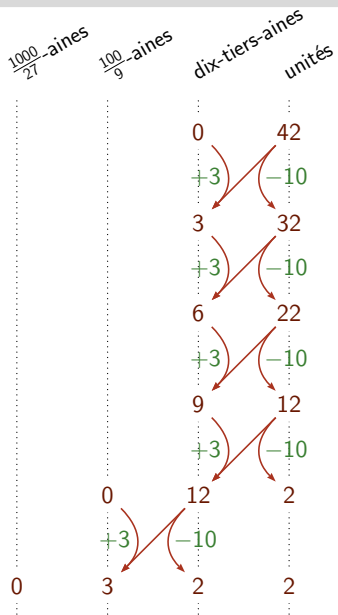
unités



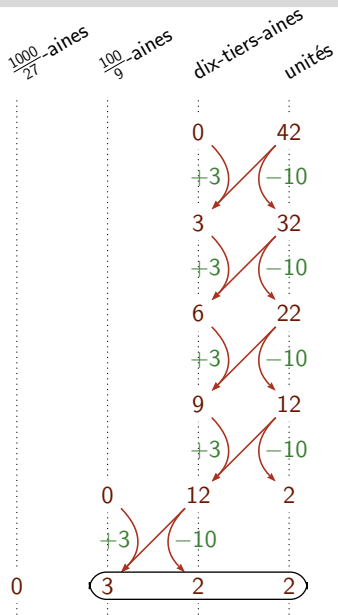
Calculer la représentation de 42 en base $\frac{10}{3}$



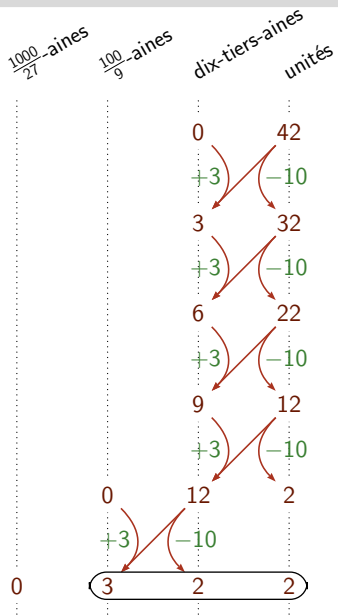
Calculer la représentation de 42 en base $\frac{10}{3}$



Calculer la représentation de 42 en base $\frac{10}{3}$



Calculer la représentation de 42 en base $\frac{10}{3}$



Vérification: calcul de $\pi(322)$

$$\begin{aligned}
 &= 3 \times \frac{100}{9} + 2 \times \frac{10}{3} + 2 \times 1 \\
 &= \frac{300}{9} + \frac{20}{3} + 2 \\
 &= \frac{300 + 60 + 18}{9} \\
 &= \frac{378}{9} \\
 &= 42
 \end{aligned}$$

- Si $n < 10$, alors sa représentation est lui-même : $\langle n \rangle = n$.

- Si $n < 10$, alors sa représentation est lui-même : $\langle n \rangle = n$.
- Sinon, on calcule le chiffre a le plus à droite de $\langle n \rangle$:

$$\langle n \rangle = \langle 3 \times k \rangle a$$

- où k est le quotient de la div. euclidienne de n par 10
(le nombre de retenues à appliquer);
- et a est le reste de la div. euclidienne de n par 10
($a = n - k \times 10$).

Données

Deux entiers p et q , premiers entre eux, tels que $p > q > 1$.

Alphabet

$$A_p = \{0, 1, 2, \dots, p-1\}$$

Évaluation

$$\pi(a_n \cdots a_1 a_0) = a_n \left(\frac{p}{q}\right)^n + \cdots + a_2 \left(\frac{p}{q}\right)^2 + a_1 \left(\frac{p}{q}\right) + a_0$$

Représentation

Calculée de droite à gauche, grâce à la formule:

$$\langle n \rangle = \langle k \times q \rangle a \quad \text{où } n = (p \times k) + a \text{ et } 0 < a < p .$$

Données

Deux entiers p et q , premiers entre eux, tels que $p > q > 1$.

Alphabet

$$A_p = \{0, 1, 2, \dots, p-1\}$$

Évaluation

$$\pi(a_n \cdots a_1 a_0) = \frac{1}{q} \left[a_n \left(\frac{p}{q}\right)^n + \cdots + a_2 \left(\frac{p}{q}\right)^2 + a_1 \left(\frac{p}{q}\right) + a_0 \right]$$

Représentation

Calculée de droite à gauche, grâce à la formule:

$$\langle n \rangle = \langle k \times q \rangle a \quad \text{où} \quad (n \times q) = (p \times k) + a \quad \text{et} \quad 0 < a < p .$$

$L_{\frac{p}{q}}$: l'ensemble des représentations des nombres entiers en base $\frac{p}{q}$.

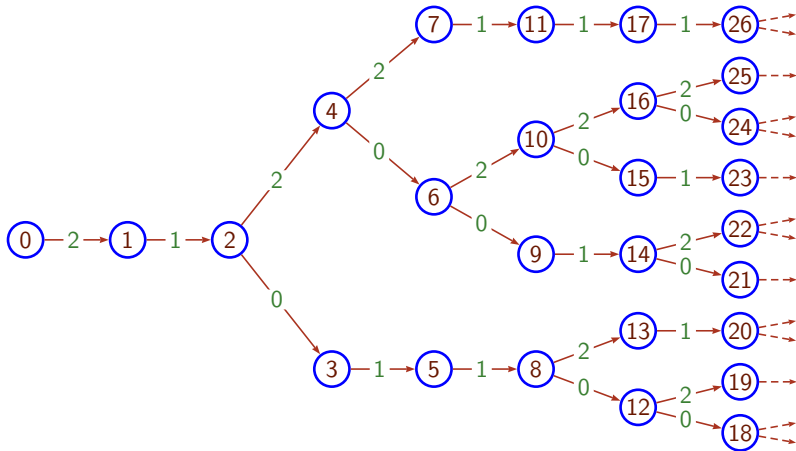


Figure: Arbre étiqueté qui représente le langage $L_{\frac{3}{2}}$

$L_{\frac{p}{q}}$: l'ensemble des représentations des nombres entiers en base $\frac{p}{q}$.

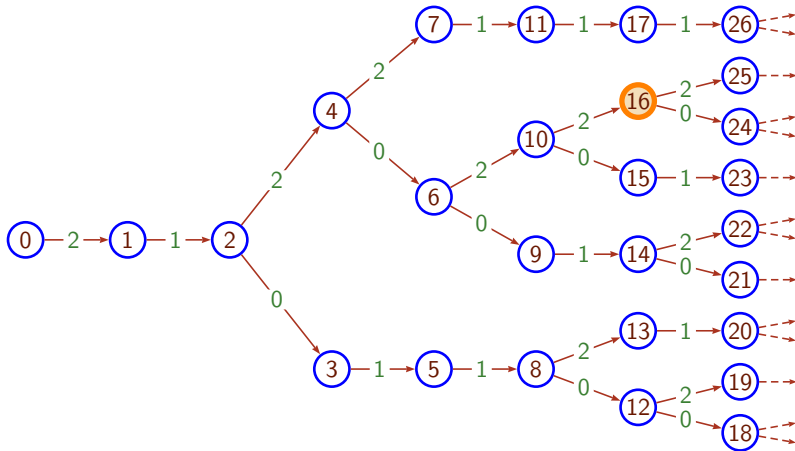


Figure: Arbre étiqueté qui représente le langage $L_{\frac{3}{2}}$

$L_{\frac{p}{q}}$: l'ensemble des représentations des nombres entiers en base $\frac{p}{q}$.

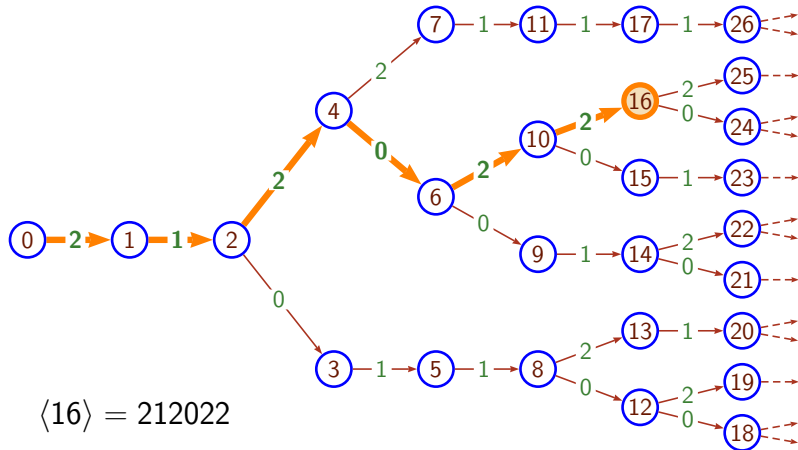


Figure: Arbre étiqueté qui représente le langage $L_{\frac{3}{2}}$

$L_{\frac{p}{q}}$: l'ensemble des représentations des nombres entiers en base $\frac{p}{q}$.

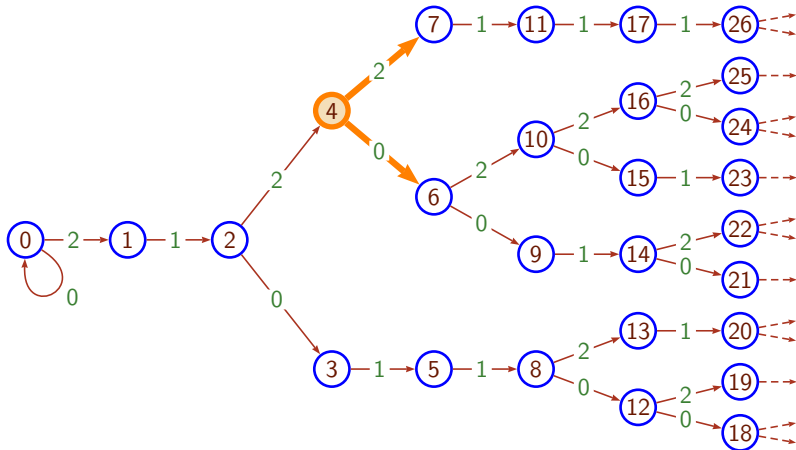


Figure: I-Arbre étiqueté qui représente le langage calable $0^*L_{\frac{3}{2}}$

Lemme (Akiyama, Frougny, Sakarovitch, 2008)

- Le langage $L_{\frac{p}{q}}$ est clos par préfixe.
- Le langage $L_{\frac{p}{q}}$ est prolongeable à droite.

Théorème (Akiyama, Frougny, Sakarovitch, 2008)

$L_{\frac{p}{q}}$ n'est pas un langage régulier, ni même algébrique.

Lemme (Akiyama, Frougny, Sakarovitch, 2008)

- Le langage $L_{\frac{p}{q}}$ est clos par préfixe.
- Le langage $L_{\frac{p}{q}}$ est prolongeable à droite.

Théorème (Akiyama, Frougny, Sakarovitch, 2008)

$L_{\frac{p}{q}}$ n'est pas un langage régulier, ni même algébrique.

$L_{\frac{p}{q}}$ possède la propriété d'itération préfixe finie (FLIP).

Definition

A : un alphabet.

L : un langage sur A .

Le langage L est dit *FLIP* si

$\forall u, v \in A^*, \exists$ qu'un nombre fini d'entiers i
tels que $u v^i$ est le préfixe d'un mot de L .

Exemple : les préfixes d'un mot infini apériodique.

Definition

A : un alphabet.

L : un langage sur A .

Le langage L est dit *FLIP* si

$\forall u, v \in A^*, \exists$ qu'un nombre fini d'entiers i
tels que $u v^i$ est le préfixe d'un mot de L .

Exemple : les préfixes d'un mot infini apériodique.

Remarques

Hormis $L_{\frac{p}{q}}$, on ne connaît pas d'exemples naturels de langage FLIP.

Definition

A : un alphabet.

L : un langage sur A .

Le langage L est dit *FLIP* si

$\forall u, v \in A^*, \exists$ qu'un nombre fini d'entiers i
tels que $u v^i$ est le préfixe d'un mot de L .

Exemple : les préfixes d'un mot infini apériodique.

Remarques

Hormis $L_{\frac{p}{q}}$, on ne connaît pas d'exemples naturels de langage FLIP.

Chaque branche infinie de L (représenté comme un arbre) est étiquetée par un mot apériodique.

Théorème

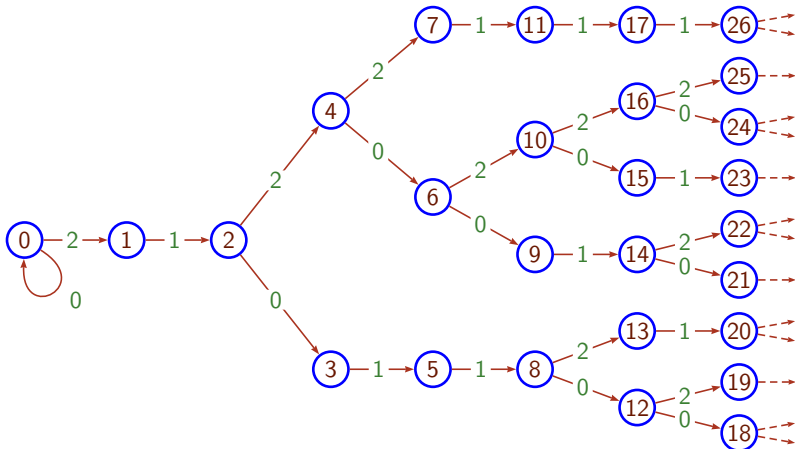
$\frac{p}{q}$: une base rationnelle.

M : sous-monoïde additif de \mathbb{Q} .

M est finiment engendré
tous les nombres de M sont représentables } $\implies \langle M \rangle$ est FLIP.

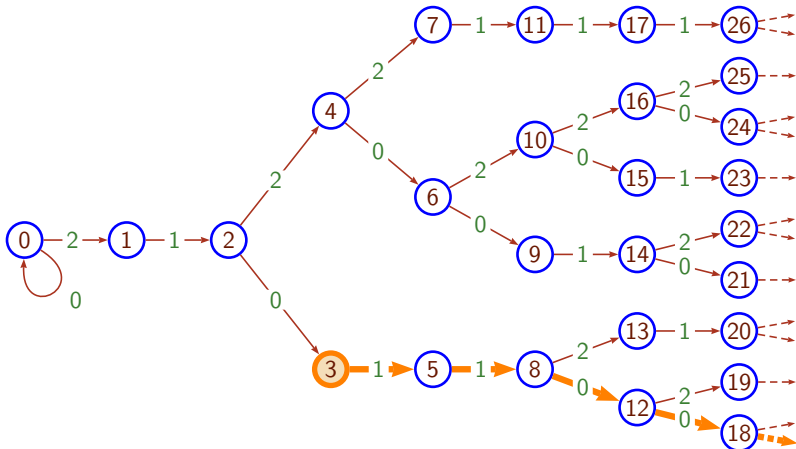
Mot minimal de n : w_n^-

Le plus petit mot qui étiquette une branche partant de n .



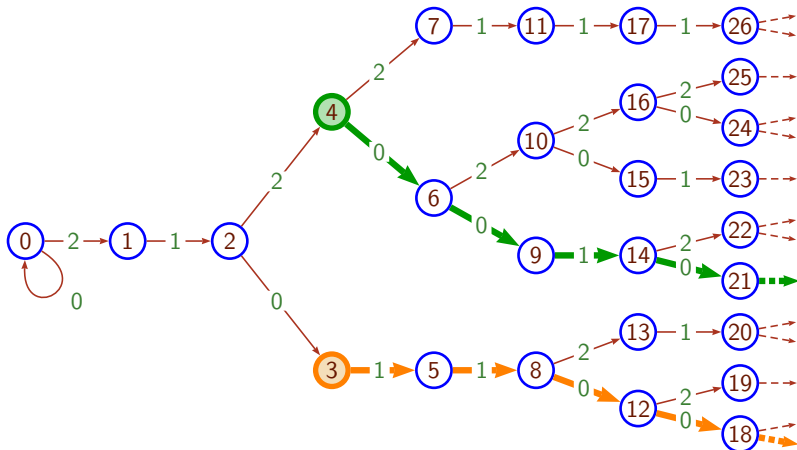
Mot minimal de n : w_n^-

Le plus petit mot qui étiquette une branche partant de n .

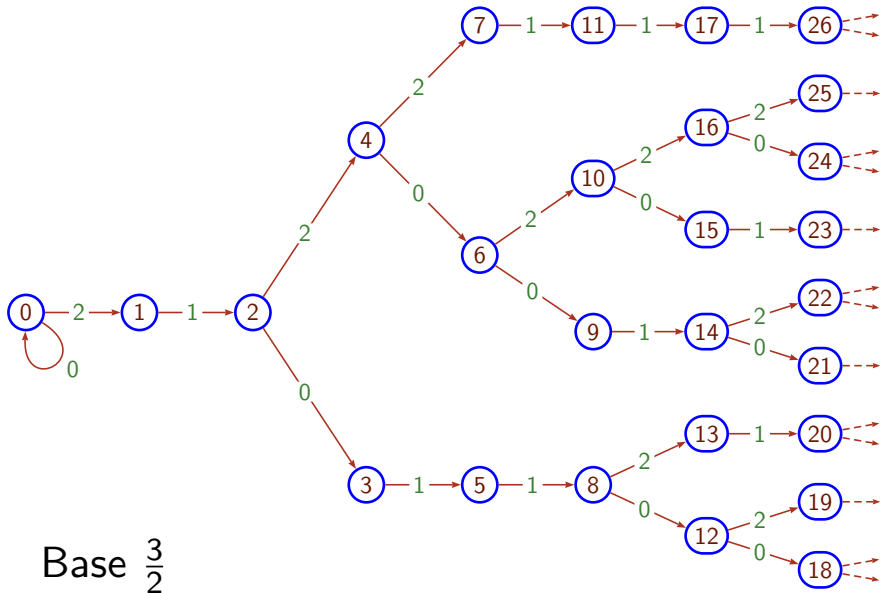


Mot minimal de n : w_n^-

Le plus petit mot qui étiquette une branche partant de n .



Ex. 1: quand la base est moyenne ($p = 2q - 1$)



Ex. 1: quand la base est moyenne ($p = 2q - 1$)

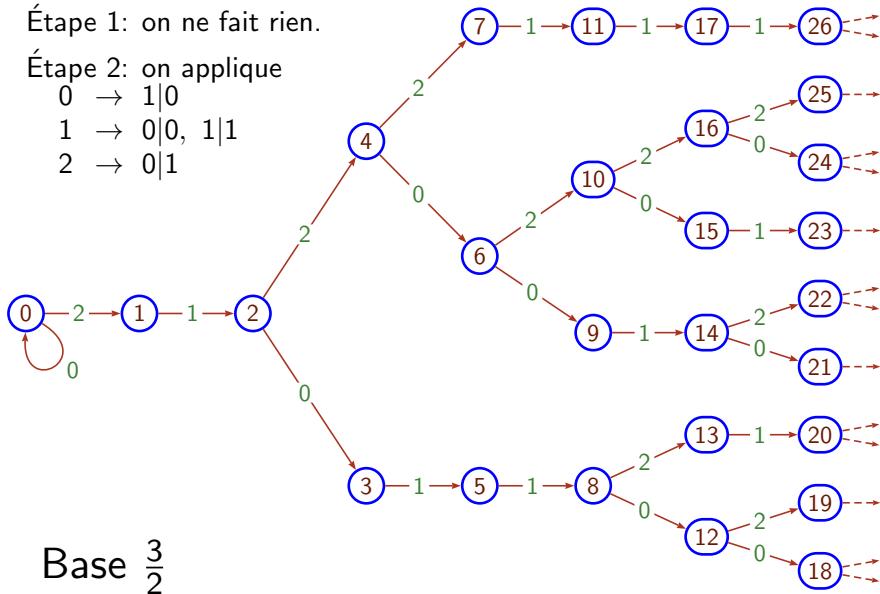
Étape 1: on ne fait rien.

Étape 2: on applique

0 \rightarrow 1|0

1 \rightarrow 0|0, 1|1

2 \rightarrow 0|1



Ex. 1: quand la base est moyenne ($p = 2q - 1$)

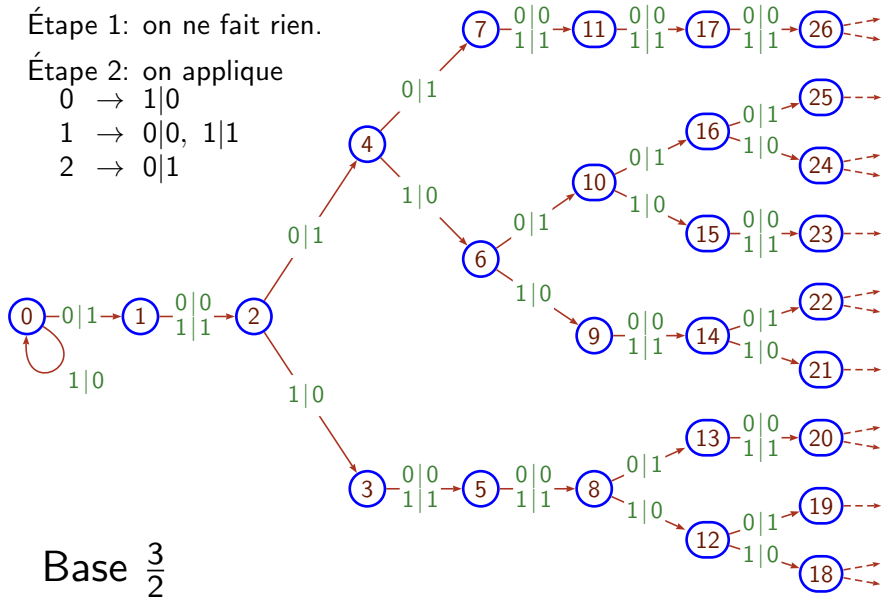
Étape 1: on ne fait rien.

Étape 2: on applique

$$0 \rightarrow 1|0$$

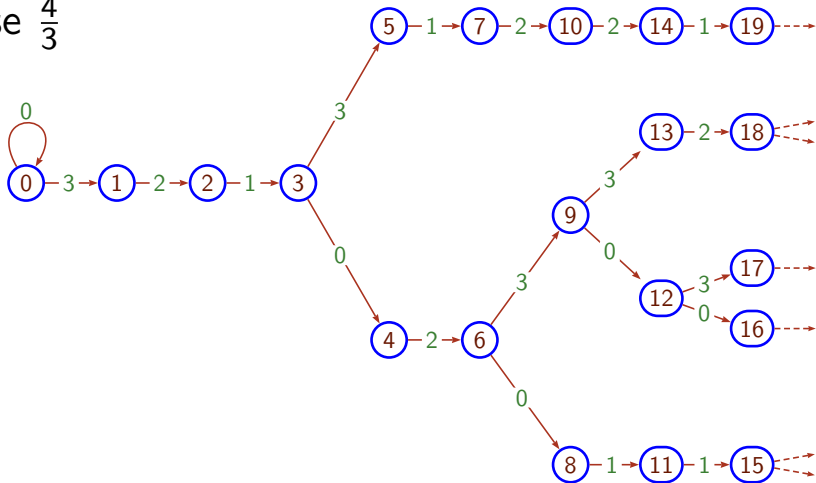
$$1 \rightarrow 0|0, 1|1$$

$$2 \rightarrow 0|1$$

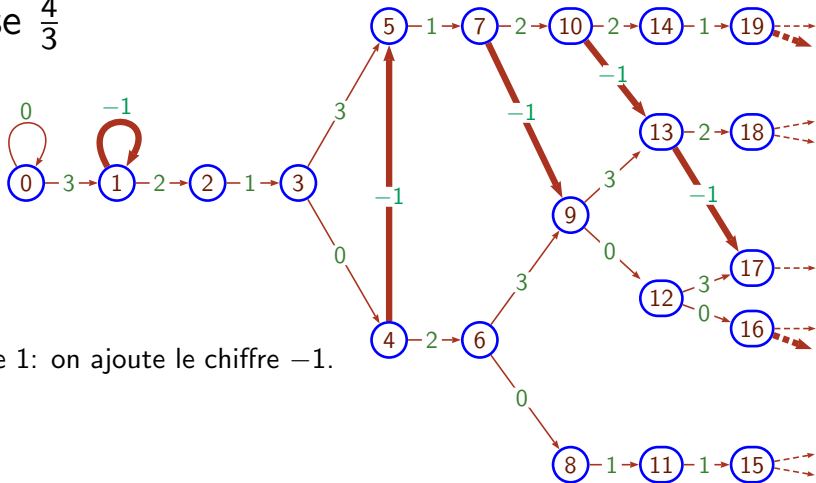


Ex. 2: quand la base est petite ($p < 2q - 1$)

Base $\frac{4}{3}$

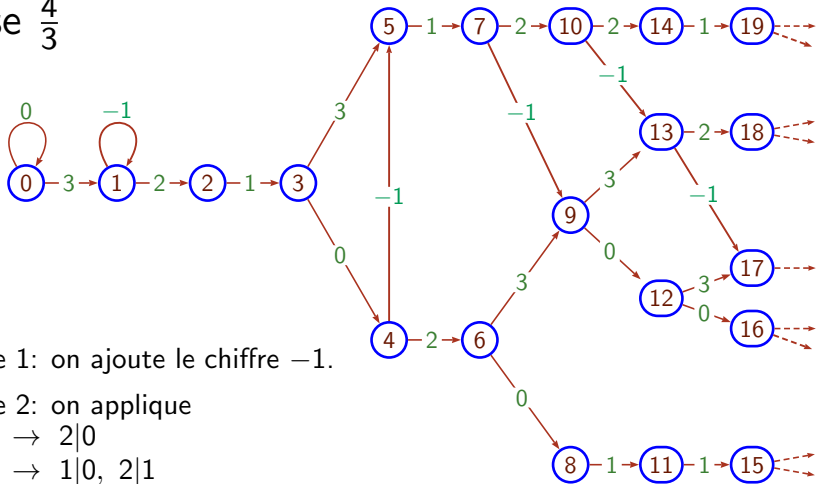


Base $\frac{4}{3}$



Étape 1: on ajoute le chiffre -1.

Base $\frac{4}{3}$



Étape 1: on ajoute le chiffre -1 .

Étape 2: on applique

$$-1 \rightarrow 2|0$$

$$0 \rightarrow 1|0, 2|1$$

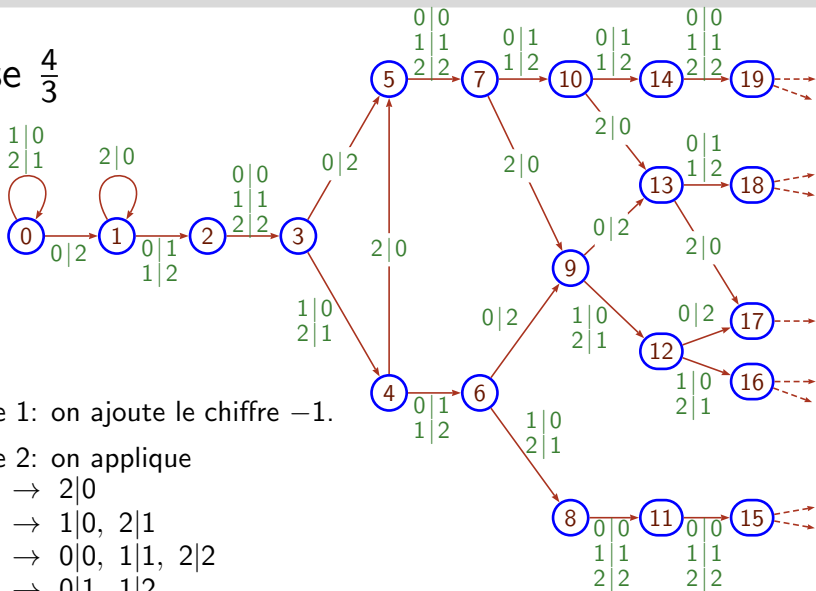
$$1 \rightarrow 0|0, 1|1, 2|2$$

$$2 \rightarrow 0|1, 1|2$$

$$3 \rightarrow 0|2$$

Ex. 2: quand la base est petite ($p < 2q - 1$)

Base $\frac{4}{3}$



Étape 1: on ajoute le chiffre -1 .

Étape 2: on applique

$$-1 \rightarrow 2|0$$

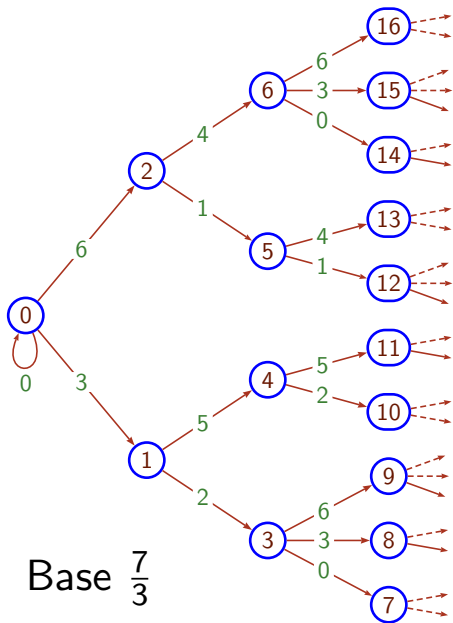
$$0 \rightarrow 1|0, 2|1$$

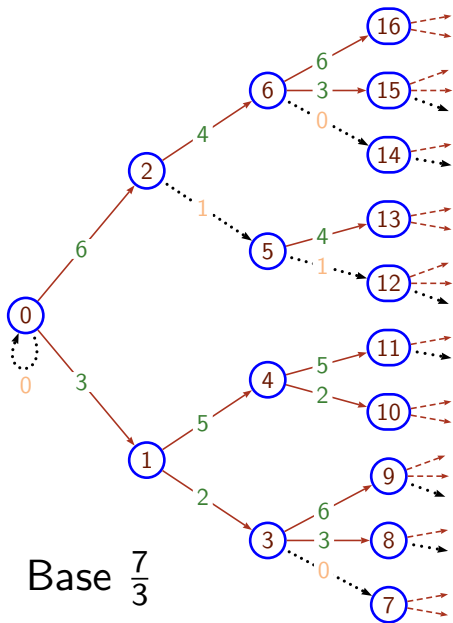
$$1 \rightarrow 0|0, 1|1, 2|2$$

$$2 \rightarrow 0|1, 1|2$$

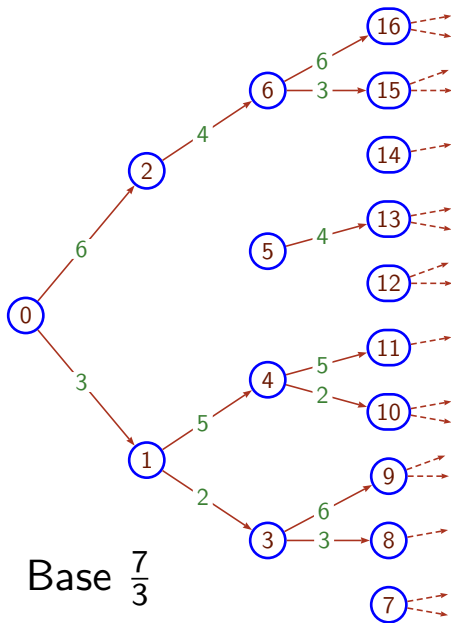
$$3 \rightarrow 0|2$$

Ex. 3: quand la base est grande ($p > 2q - 1$)





Étape 1: on supprime les 0 et 1.



Étape 1: on supprime les 0 et 1.

Étape 2: on applique

$$2 \rightarrow 2|0$$

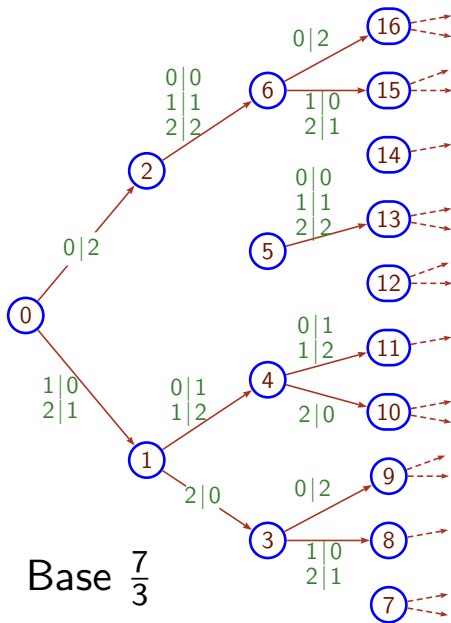
$$3 \rightarrow 1|0, 2|1$$

$$4 \rightarrow 0|0, 1|1, 2|2$$

$$5 \rightarrow 0|1, 1|2$$

$$6 \rightarrow 0|2$$

Ex. 3: quand la base est grande ($p > 2q - 1$)



Étape 1: on supprime les 0 et 1.

Étape 2: on applique

$$2 \rightarrow 2|0$$

$$3 \rightarrow 1|0, 2|1$$

$$4 \rightarrow 0|0, 1|1, 2|2$$

$$5 \rightarrow 0|1, 1|2$$

$$6 \rightarrow 0|2$$

1 Qu'est-ce qu'un système de numération

2 Sur la base rationnelle

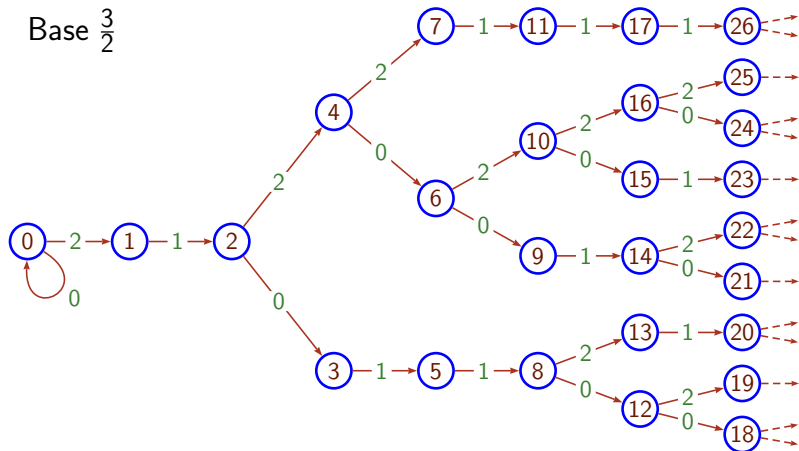
3 Sur la notion de signature

- ▶ Origine et définition
- ▶ Signatures périodiques
- ▶ Signatures des langages réguliers
- ▶ Surminimisation

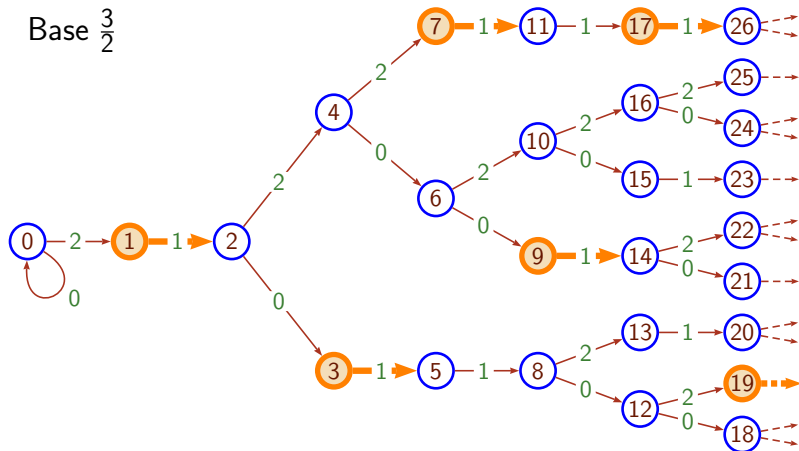
4 Sur la base entière

5 Perspectives

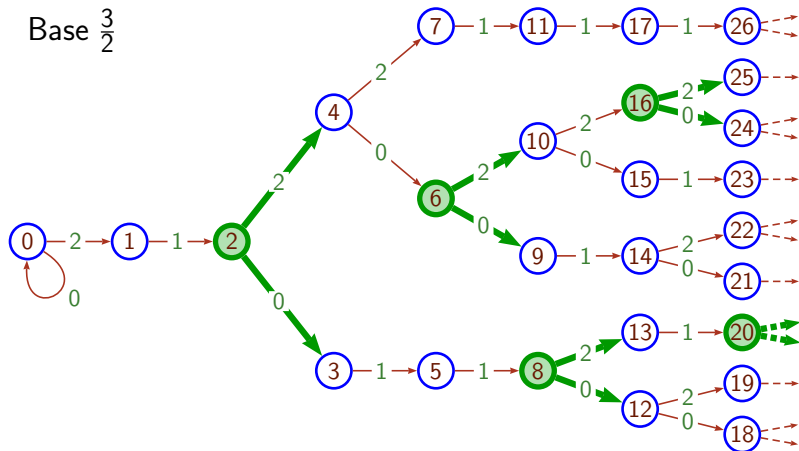
Base $\frac{3}{2}$



Base $\frac{3}{2}$

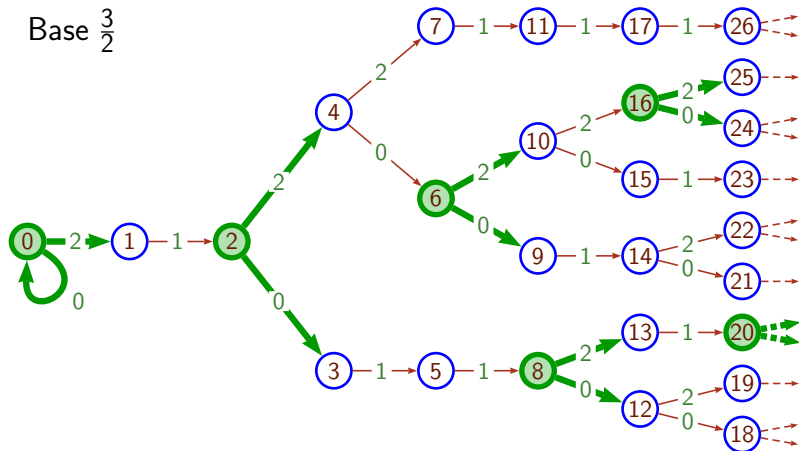


- Les nœuds impairs ont un arc sortant étiqueté par 1.



- Les nœuds impairs ont un arc sortant étiqueté par 1.
- Les nœuds pairs ont deux arcs sortants étiquetés par 0 et 2.

Base $\frac{3}{2}$



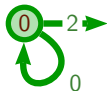
- Les nœuds impairs ont un arc sortant étiqueté par 1.
- Les nœuds pairs ont deux arcs sortants étiquetés par 0 et 2.

Base $\frac{3}{2}$

0

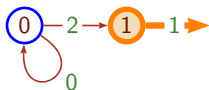
- Les nœuds impairs ont un arc sortant étiqueté par 1.
- Les nœuds pairs ont deux arcs sortants étiquetés par 0 et 2.

Base $\frac{3}{2}$



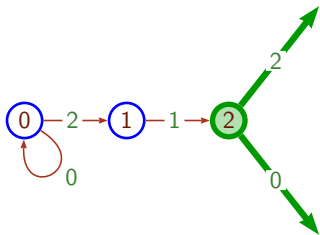
- Les nœuds impairs ont un arc sortant étiqueté par 1.
- Les nœuds pairs ont deux arcs sortants étiquetés par 0 et 2.

Base $\frac{3}{2}$



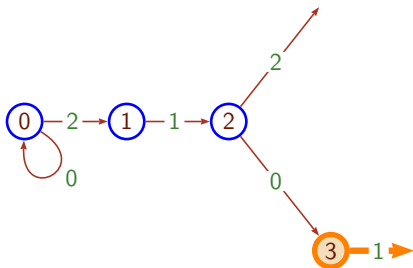
- Les nœuds impairs ont un arc sortant étiqueté par 1.
- Les nœuds pairs ont deux arcs sortants étiquetés par 0 et 2.

Base $\frac{3}{2}$



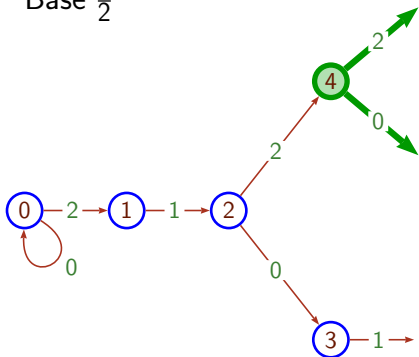
- Les nœuds impairs ont un arc sortant étiqueté par 1.
- Les nœuds pairs ont deux arcs sortants étiquetés par 0 et 2.

Base $\frac{3}{2}$



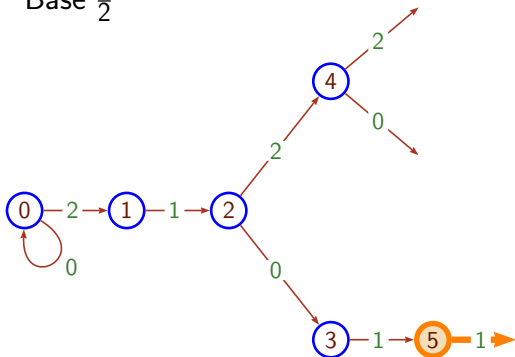
- Les nœuds impairs ont un arc sortant étiqueté par 1.
- Les nœuds pairs ont deux arcs sortants étiquetés par 0 et 2.

Base $\frac{3}{2}$



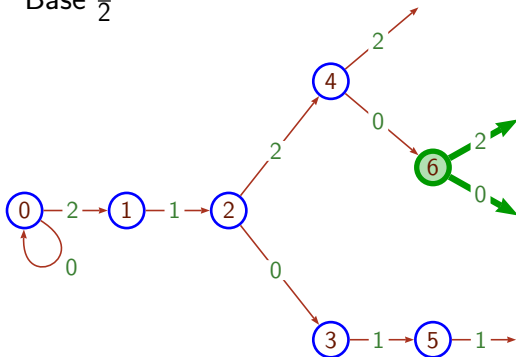
- Les nœuds impairs ont un arc sortant étiqueté par 1.
- Les nœuds pairs ont deux arcs sortants étiquetés par 0 et 2.

Base $\frac{3}{2}$



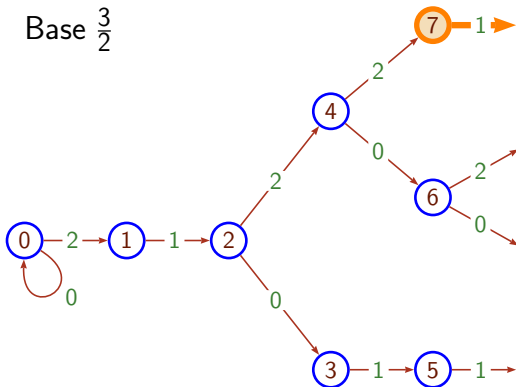
- Les nœuds impairs ont un arc sortant étiqueté par 1.
- Les nœuds pairs ont deux arcs sortants étiquetés par 0 et 2.

Base $\frac{3}{2}$



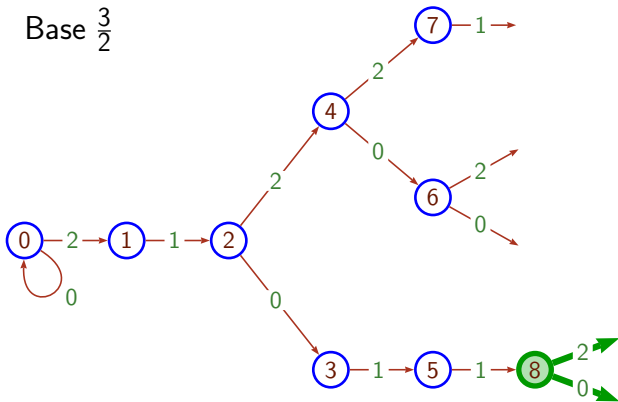
- Les nœuds impairs ont un arc sortant étiqueté par 1.
- Les nœuds pairs ont deux arcs sortants étiquetés par 0 et 2.

Base $\frac{3}{2}$



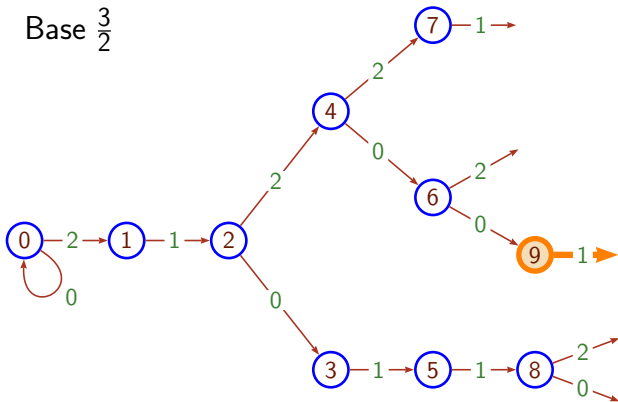
- Les nœuds impairs ont un arc sortant étiqueté par 1.
- Les nœuds pairs ont deux arcs sortants étiquetés par 0 et 2.

Base $\frac{3}{2}$



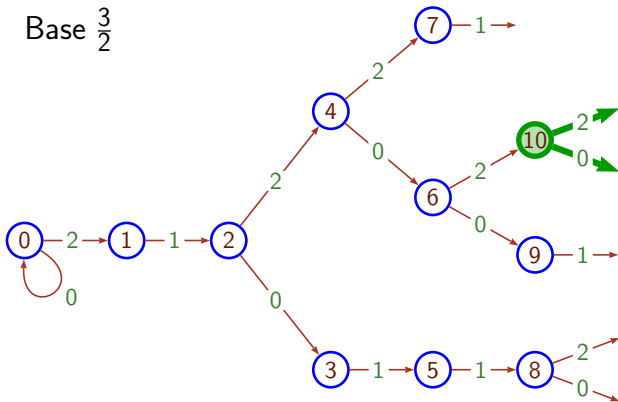
- Les nœuds impairs ont un arc sortant étiqueté par 1.
- Les nœuds pairs ont deux arcs sortants étiquetés par 0 et 2.

Base $\frac{3}{2}$



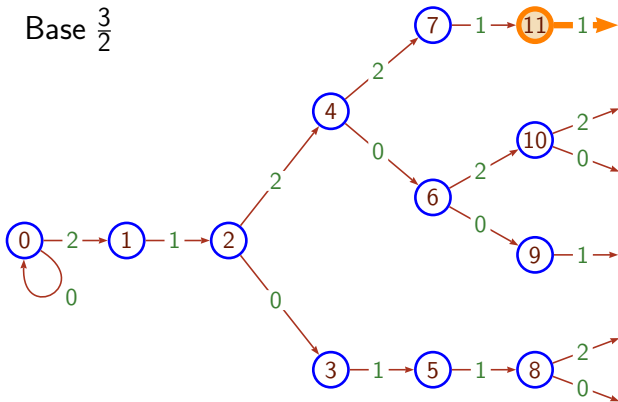
- Les nœuds impairs ont un arc sortant étiqueté par 1.
- Les nœuds pairs ont deux arcs sortants étiquetés par 0 et 2.

Base $\frac{3}{2}$



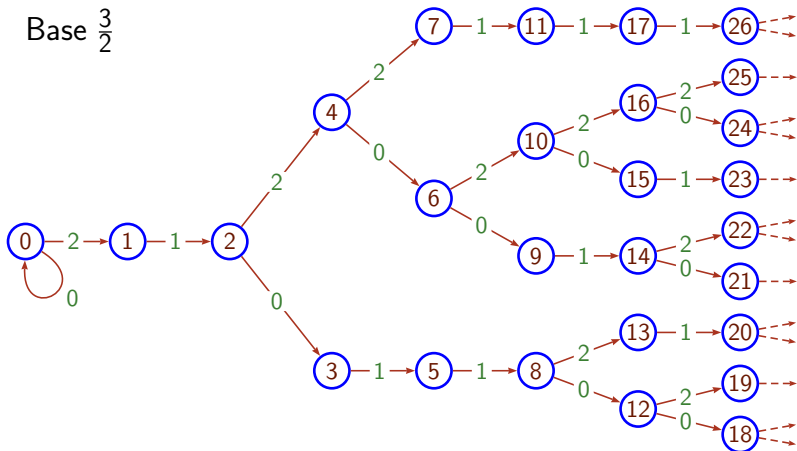
- Les nœuds impairs ont un arc sortant étiqueté par 1.
- Les nœuds pairs ont deux arcs sortants étiquetés par 0 et 2.

Base $\frac{3}{2}$



- Les nœuds impairs ont un arc sortant étiqueté par 1.
- Les nœuds pairs ont deux arcs sortants étiquetés par 0 et 2.

Construire $L_{\frac{3}{2}}$ périodiquement avec 2, 1, 2, 1, ...

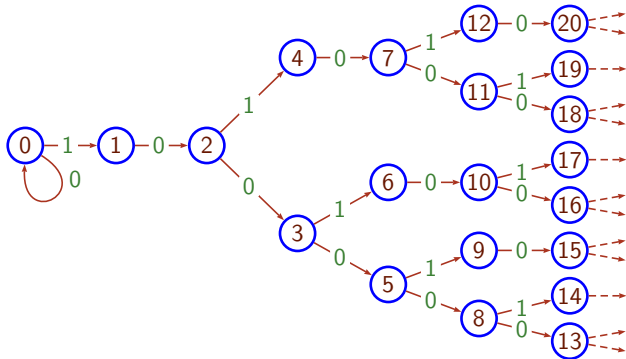


- Les nœuds impairs ont un arc sortant étiqueté par 1.
- Les nœuds pairs ont deux arcs sortants étiquetés par 0 et 2.

A : un alphabet totalement ordonné.

T : un i-arbre étiqueté sur A (ou langage calable, clos par préfixe).

L'ordre de A induit un parcours en largeur canonique de T .



• Signature : $\mathbf{s} =$

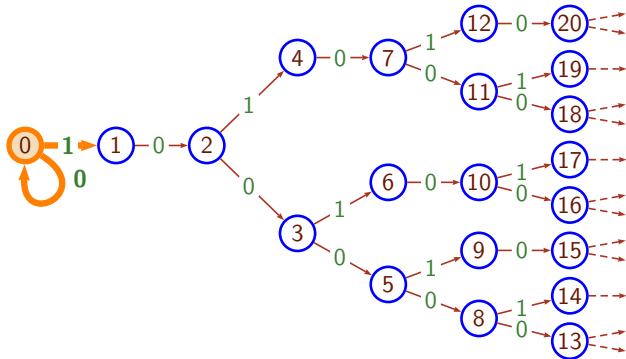
• Étiquetage : $\lambda =$

Signature et étiquetage d'un arbre étiqueté

A : un alphabet totalement ordonné.

T : un i-arbre étiqueté sur A (ou langage calable, clos par préfixe).

L'ordre de A induit un parcours en largeur canonique de T .



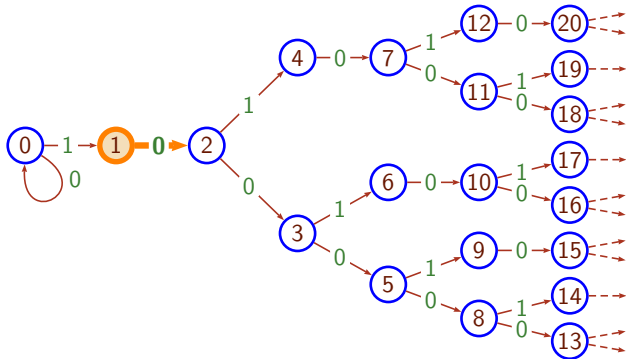
- Signature : $s = 2$
- Étiquetage : $\lambda = 01$

Signature et étiquetage d'un arbre étiqueté

A : un alphabet totalement ordonné.

T : un i-arbre étiqueté sur A (ou langage calable, clos par préfixe).

L'ordre de A induit un parcours en largeur canonique de T .



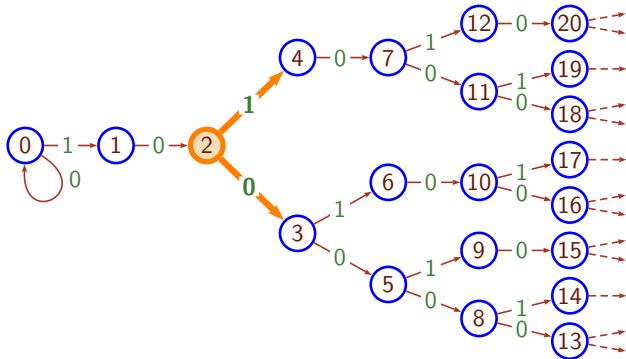
- Signature : $s = 21$
- Étiquetage : $\lambda = 010$

Signature et étiquetage d'un arbre étiqueté

A : un alphabet totalement ordonné.

T : un i -arbre étiqueté sur A (ou langage calable, clos par préfixe).

L'ordre de A induit un parcours en largeur canonique de T .



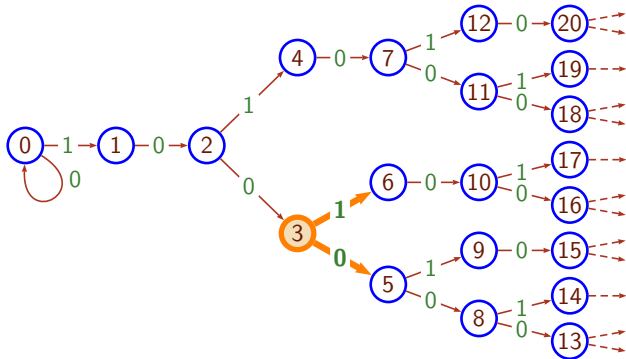
• Signature : $\mathbf{s} = 2\ 1\ 2$

• Étiquetage : $\lambda = 01\ 0\ 01$

A : un alphabet totalement ordonné.

T : un i-arbre étiqueté sur A (ou langage calable, clos par préfixe).

L'ordre de A induit un parcours en largeur canonique de T .



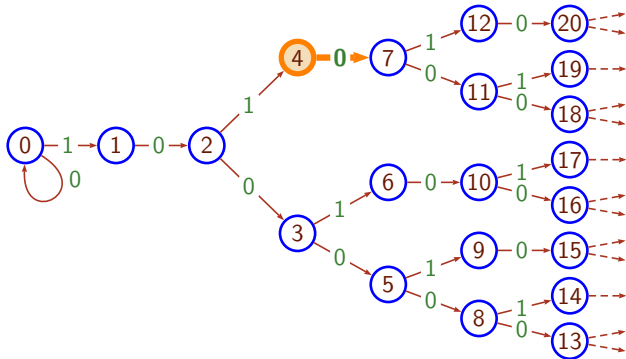
• Signature : $\mathbf{s} = 2\ 1\ 2\ 2$

• Étiquetage : $\lambda = 01\ 0\ 01\ 01$

A : un alphabet totalement ordonné.

T : un i -arbre étiqueté sur A (ou langage calable, clos par préfixe).

L'ordre de A induit un parcours en largeur canonique de T .



• Signature : $\mathbf{s} = 2\ 1\ 2\ 2\ 1$

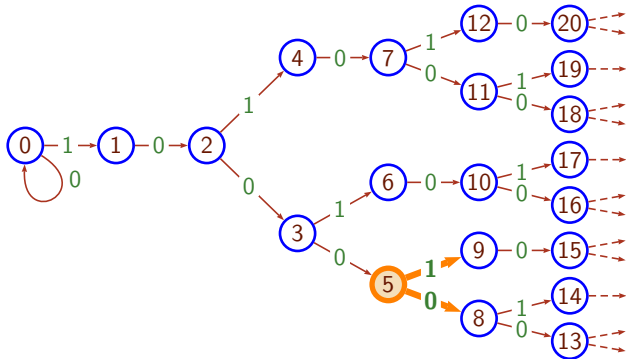
• Étiquetage : $\lambda = 01\ 0\ 01\ 01\ 0$

Signature et étiquetage d'un arbre étiqueté

A : un alphabet totalement ordonné.

T : un i-arbre étiqueté sur A (ou langage calable, clos par préfixe).

L'ordre de A induit un parcours en largeur canonique de T .



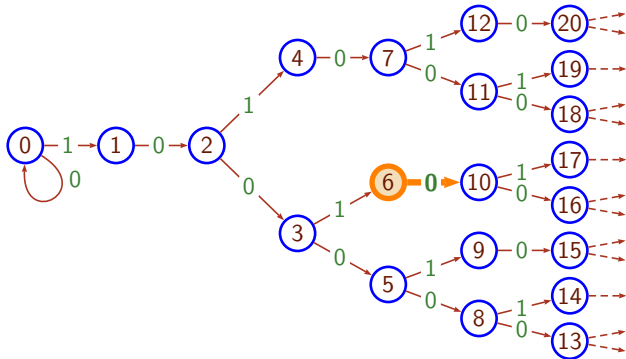
• Signature : $s = 2\ 1\ 2\ 2\ 1\ 2$

• Étiquetage : $\lambda = 01\ 0\ 01\ 01\ 0\ 01$

A : un alphabet totalement ordonné.

T : un i-arbre étiqueté sur A (ou langage calable, clos par préfixe).

L'ordre de A induit un parcours en largeur canonique de T .



• Signature : $\mathbf{s} = 2\ 1\ 2\ 2\ 1\ 2\ 1$

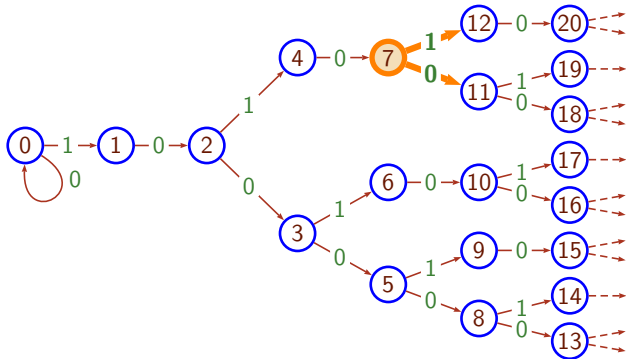
• Étiquetage : $\lambda = 01\ 0\ 01\ 01\ 0\ 01\ 0$

Signature et étiquetage d'un arbre étiqueté

A : un alphabet totalement ordonné.

T : un i-arbre étiqueté sur A (ou langage calable, clos par préfixe).

L'ordre de A induit un parcours en largeur canonique de T .



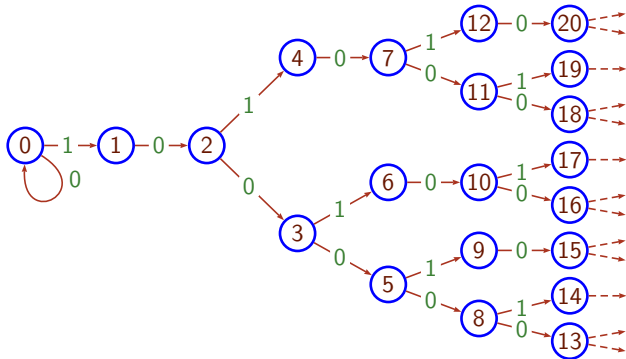
• Signature : $\mathbf{s} = 2\ 1\ 2\ 2\ 1\ 2\ 1\ 2$

• Étiquetage : $\lambda = 01\ 0\ 01\ 01\ 0\ 01\ 0\ 01$

A : un alphabet totalement ordonné.

T : un i -arbre étiqueté sur A (ou langage calable, clos par préfixe).

L'ordre de A induit un parcours en largeur canonique de T .



• Signature : $\mathbf{s} = 2\ 1\ 2\ 2\ 1\ 2\ 1\ 2\ 2\ 1\ 2\ 2\ 1\ \dots$

• Étiquetage : $\lambda = 01\ 0\ 01\ 01\ 0\ 01\ 0\ 01\ 01\ 0\ 01\ 01\ 0\ \dots$

Théorème

$\frac{p}{q}$: une base entière ou rationnelle.

Le Langage $0^* L_{\frac{p}{q}}$ a

- pour signature le mot périodique $(\mathbf{r}_{\frac{p}{q}})^\omega$,
où $\mathbf{r}_{\frac{p}{q}}$ est le *rythme de Christoffel* de pente $\frac{p}{q}$;
- et pour étiquetage le mot périodique $(\boldsymbol{\gamma}_{\frac{p}{q}})^\omega$, où
 $\boldsymbol{\gamma}_{\frac{p}{q}} = (0, (q \bmod p), (2q \bmod p), \dots, ((p-1)q \bmod p)$.

Théorème

$\forall \mathbf{r}$: un mot fini sur un alphabets de chiffres.

$\exists \gamma_{\mathbf{r}}$: l'*étiquetage spécial* associé à \mathbf{r} .

0^*L : le langage calable de signature \mathbf{r}^ω et d'étiquetage $(\gamma_{\mathbf{r}})^\omega$.

$\frac{p}{q}$: la moyenne des lettres de \mathbf{r} .

Alors, L est une représentation des entiers en base $\frac{p}{q}$ sur un alphabet (fini) non-canonique.

Théorème

z^*L : un langage calable et clos-par-préfixe.

L est régulier \iff sa signature étiquetée est *s-morphique*[†].

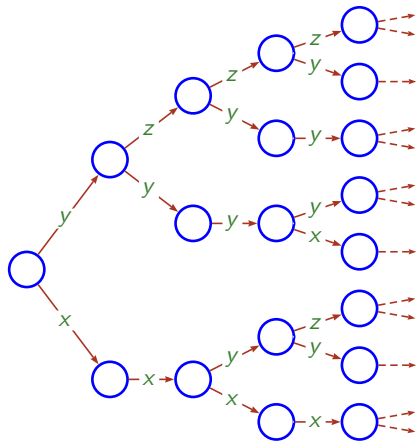
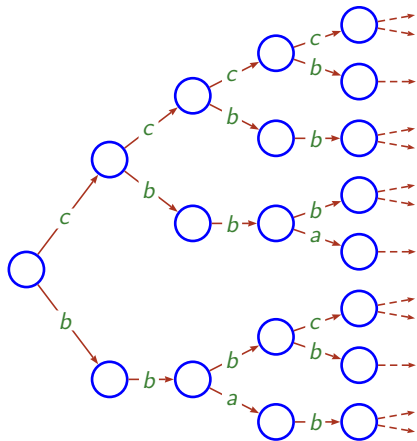
La démonstration utilise une transformation
morphismes de mots \longleftrightarrow automates
que l'on retrouve dans les travaux de

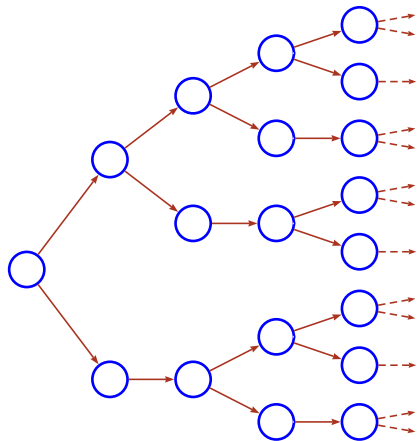
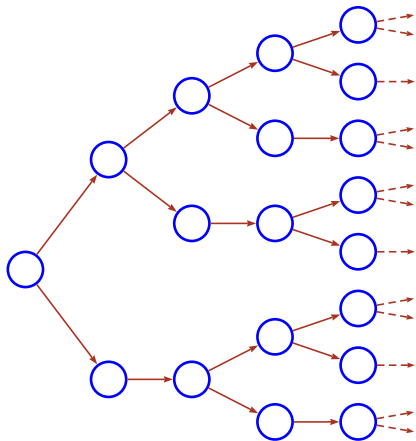
- Dumont et Thomas (1989, 1991, 1993);
- Rigo et Maes (2002).

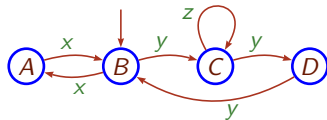
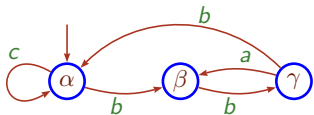
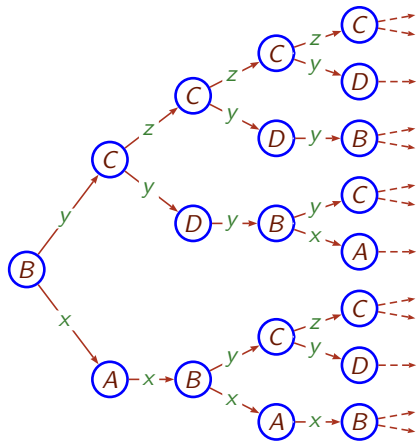
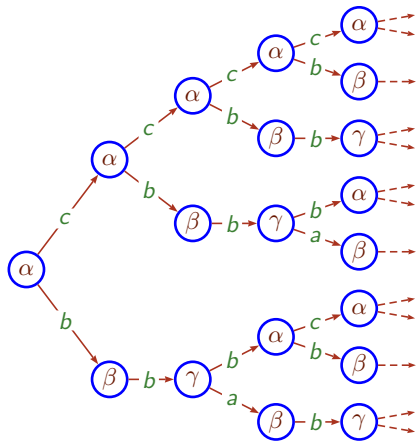
† Une signature étiquetée $(\mathbf{s}, \boldsymbol{\lambda})$ est *s-morphique* si

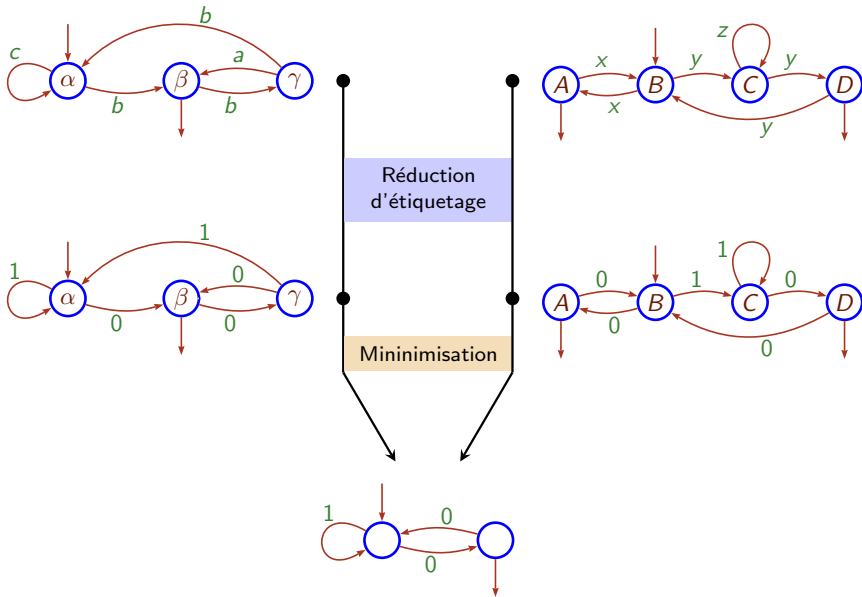
$$\mathbf{s} = f_{\sigma}(\sigma^{\omega}(a)) \quad \text{et} \quad \boldsymbol{\lambda} = g(\sigma^{\omega}(a))$$

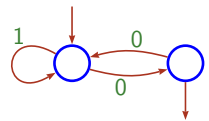
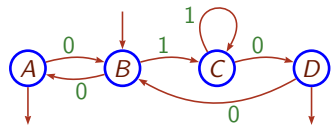
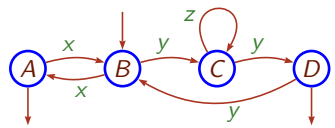
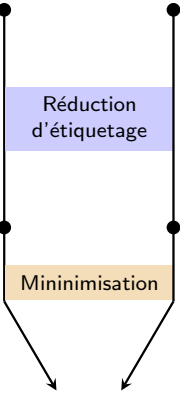
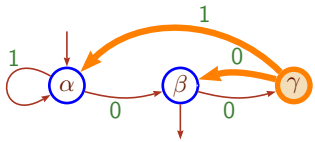
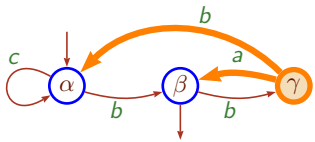
- σ est un endomorphisme de mots;
- f_{σ} est le morphisme défini par: pour tout b , $f_{\sigma}(b) = |\sigma(b)|$;
- g est un morphisme vérifiant: pour tout b , $|g(b)| = |\sigma(b)|$.

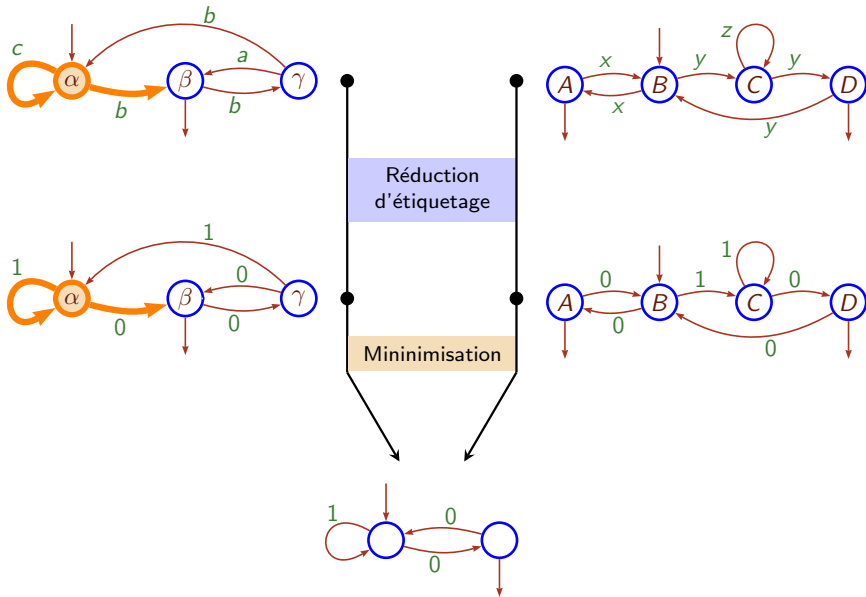












1 Qu'est-ce qu'un système de numération

2 Sur la base rationnelle

3 Sur la notion de signature

4 Sur la base entière

▶ Définition

▶ Le problème ULTIME PÉRIODICITÉ

▶ Solutions antérieures

▶ Amélioration obtenue

5 Perspectives

Un entier $p > 1$

Alphabet: $A_p = \{0, 1, \dots, (p-1)\}$

Représentation (par divisions euclidiennes successives)

$$\langle n \rangle = \langle n' \rangle a \quad ,$$

où, (n', a) est le résultat de la div. euclidienne de n par p .

Évaluation

$$\pi(a_k \cdots a_1 a_0) = (a_k \times p^k) + \cdots + (a_1 \times p^1) + a_0.$$

Un entier $p > 1$

Alphabet: $A_p = \{0, 1, \dots, (p-1)\}$

Représentation (par divisions euclidiennes successives)

$$\langle n \rangle = \langle n' \rangle a \quad ,$$

où, (n', a) est le résultat de la div. euclidienne de n par p .

Évaluation

$$\pi(a_k \cdots a_1 a_0) = (a_k \times p^k) + \cdots + (a_1 \times p^1) + a_0.$$

Convention (LSDF)

On écrit le chiffre de poids faible en premier.

Exemple: La quantité  s'écrit donc 91 (au lieu de 19).

Un entier $p > 1$

Alphabet: $A_p = \{0, 1, \dots, (p-1)\}$

Représentation (par divisions euclidiennes successives)

$$\langle n \rangle = a \langle n' \rangle ,$$

où, (n', a) est le résultat de la div. euclidienne de n par p .

Évaluation

$$\pi(a_0 a_1 \cdots a_k) = (a_k \times p^k) + \cdots + (a_1 \times p^1) + a_0.$$

Convention (LSDF)

On écrit le chiffre de poids faible en premier.

Exemple: La quantité  s'écrit donc 91 (au lieu de 19).

Langage régulier

Un langage est dit *régulier* s'il est accepté par un automate.

Un tel langage est considéré *simple*.

Langage régulier

Un langage est dit *régulier* s'il est accepté par un automate.

Un tel langage est considéré *simple*.

Ensemble p -reconnaisable

p : une base entière.

E : un ensemble d'entiers.

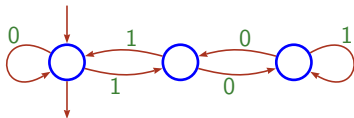
$$E \text{ est } p\text{-reconnaisable} \iff \langle E \rangle \text{ est régulier}$$

Un tel ensemble est considéré *simple* relativement à p .

Théorème (Pascal?)

Les ensembles ultimement périodiques d'entiers sont p -reconnaissables pour toute base p .

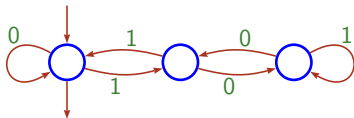
Exemple: l'ensemble des entiers divisibles par 3, écrits en base 2:



Théorème (Pascal?)

Les ensembles ultimement périodiques d'entiers sont p -reconnaisables pour toute base p .

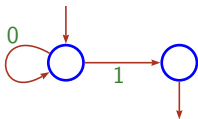
Exemple: l'ensemble des entiers divisibles par 3, écrits en base 2:



Théorème (?)

Il existe des ensembles qui sont p -reconnaisables pour certaines bases p mais pas pour d'autres.

Exemple: l'ensemble des puissances de 10, écrits en base 10:



Théorème (Cobham, 1969)

p, p' : deux bases entières multiplicativement indépendantes[†].

E : un ensemble d'entiers.

Si E est à la fois p - et p' -reconnaisable, alors E est ultimement périodique.

Corollaire

Si un ensemble d'entiers E est p -reconnaisable dans toutes les bases p , alors E est ultimement périodique.

[†] Deux entiers b et b' sont multiplicativement indépendants si aucune puissance de l'un n'est égale à une puissance de l'autre.

Donnée

- une base p
- un automate \mathcal{A}

Question

Le langage $L(\mathcal{A})$ est-il la représentation, en base p , d'un ensemble ultimement périodique d'entiers ?

Théorème (Honkala, 1986)

Le problème ULTIME PÉRIODICITÉ est décidable.

Théorème (Honkala, 1986)

Le problème ULTIME PÉRIODICITÉ est décidable.

Théorème (Muchnik, 1991)

Le problème SOUS-ENSEMBLE RATIONNEL[†] est décidable.

Théorème (Leroux, 2005)

Le problème SOUS-ENSEMBLE RATIONNEL[†] est décidable en complexité quadratique.

[†] SOUS-ENSEMBLE RATIONNEL est une généralisation d'ULTIME PÉRIODICITÉ: on considère un ensemble de d -uplets d'entiers.

Théorème (Allouche, Rampersad, Shallit, 2009)

Le problème ULTIME PÉRIODICITÉ GÉNÉRALISÉ[‡] est décidable en complexité exponentielle pour les U -systèmes dans lequel l'addition est réalisable par un automate.

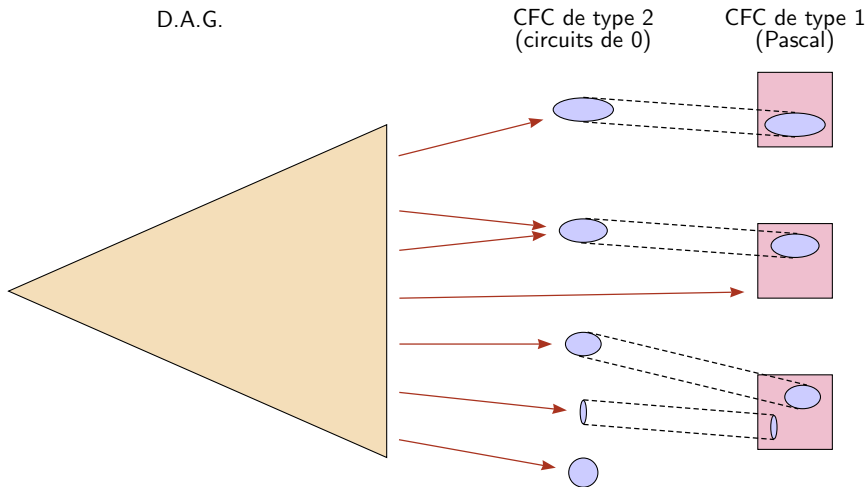
Théorème (Bell, Charlier, Fraenkel, Rigo, 2009)

Le problème ULTIME PÉRIODICITÉ GÉNÉRALISÉ[‡] est décidable pour une large classe de U -systèmes.

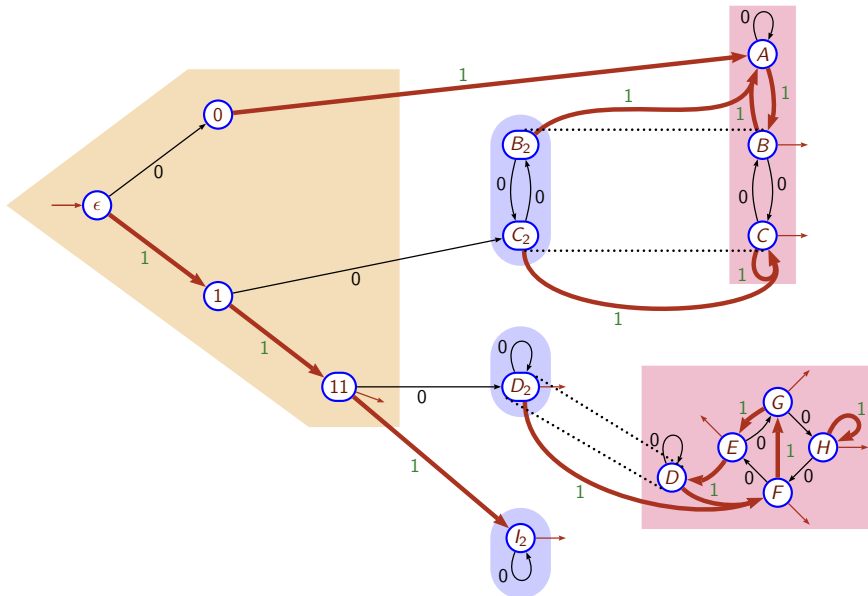
[‡] Ce problème généralise d'ULTIME PÉRIODICITÉ à d'autres systèmes de numérations.

Théorème

Le problème ULTIME PÉRIODICITÉ est décidable en complexité quasi-linéaire.



Un automate satisfaisant le critère (UP)



- 1 Qu'est-ce qu'un système de numération
- 2 Sur la base rationnelle
- 3 Sur la notion de signature
- 4 Sur la base entière
- 5 Perspectives

Sur ULTIME PÉRIODICITÉ en base entière

- En utilisant la convention LSDF : quasi-linéaire.
- Sinon : exponentielle [ARS 2009].
- Pour des problèmes plus généraux :
 - en dimension supérieure : quadratique [L 2005] ;
 - aux U-systèmes : exponentielle [ARS 2009] ;
 - aux SNA réguliers : décidable [D 2013, M 2011].

Sur les mots minimaux en base rationnelle

- La fonction successeur : transducteur infini proche de $L_{\frac{p}{q}}$.
- Si restreinte à une seule entrée : inconnu, transducteur fini?

Sur les signatures

- Signatures s-morphique \leftrightarrow SNA réguliers.
- Signatures périodique \leftrightarrow bases entières et rationnelles.
- Classification des systèmes de numérations?
 - ? \leftrightarrow U-systèmes.
 - ? \leftrightarrow SNA algébrique.

Sur la surminimisation

- Réduit le nombre d'états davantage que la minimisation.
- Conserve uniquement l'arbre ordonné sous-jacent.
- Induit une "forme normale" sur les SNA réguliers.