

# Logique et langages de données

Manuel Baclet-Patricia Bouyer

Laboratoire Spécification et Vérification

UMR 8643 CNRS & ENS de Cachan

<b>Langages réguliers</b>	<b>Langages de données</b>
Automates finis	Automates de données
Expressions rationnelles	Adaptation du théorème de Kleene
<i>MSO</i>	Logique du second ordre
Monoïdes	Monoïdes

# Pour les langages réguliers

▷ Un langage  $L \subset \Sigma^*$  est **régulier** si et seulement si il existe un monoïde fini  $M$ , un morphisme de monoïde  $f : \Sigma^* \rightarrow M$  et une partie  $F \subset M$  tels que:

$$L = f^{-1}(F)$$

▷ Syntaxe de MSO:

$$\psi ::= x < y \mid x = y \mid \exists x.\psi \mid \forall x.\psi \mid R_a x \mid Xx \mid \neg\psi \mid \psi \wedge \psi \mid \psi \vee \psi$$

▷ Un langage est régulier si et seulement si il est définissable par une formule close de MSO.

# Mécanismes à monoïde

▷ Une **mise à jour** est une partie de  $\{1..k\}$ .

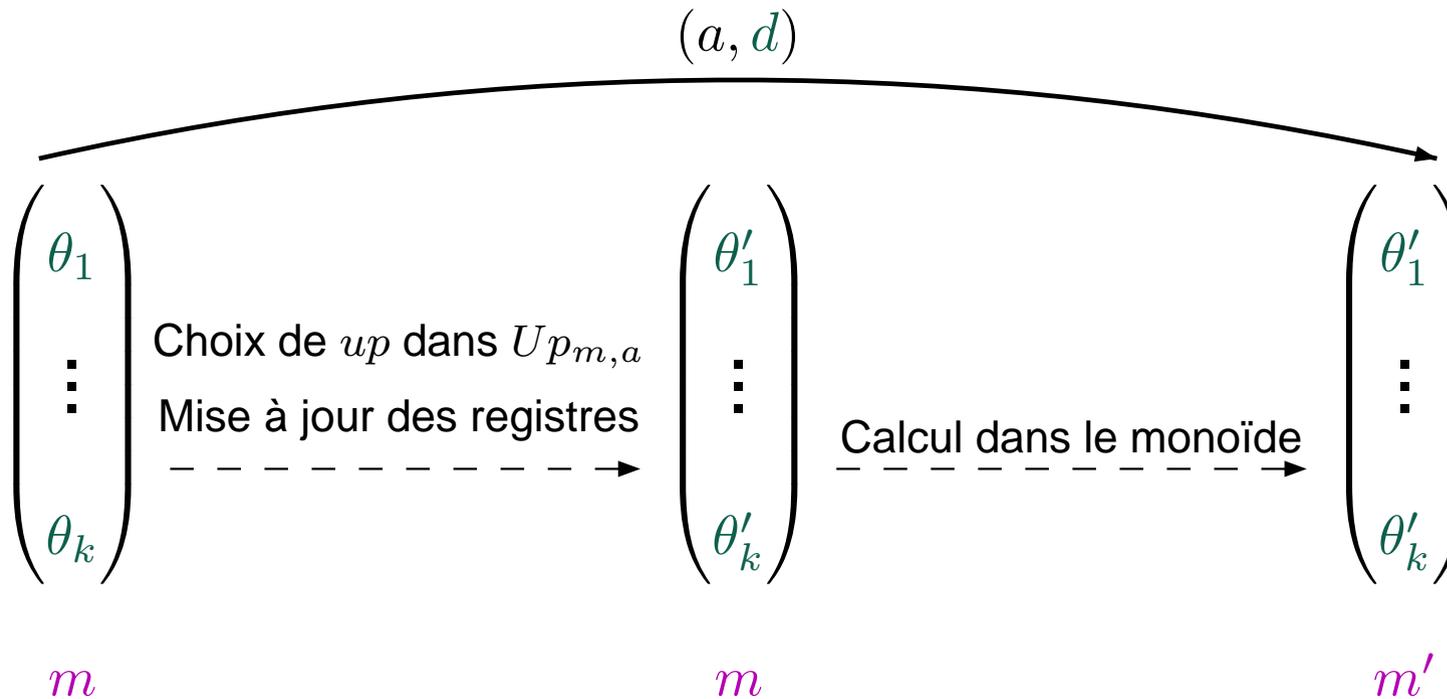
$$\begin{pmatrix} r_1 \\ \vdots \\ r_k \end{pmatrix} \xrightarrow[d]{up} \begin{pmatrix} r'_1 \\ \vdots \\ r'_k \end{pmatrix} \quad \text{où } r'_i = \begin{cases} d & \text{si } i \in up \\ r_i & \text{sinon.} \end{cases}$$

▷ Un monoïde fini  $M$ , une partie  $F \subset M$  et un quadruplet

$(Up_{m,a}(m, a) \in M \times \Sigma, k, \sim, \phi)$  où:

- i.  $k$  est le nombre de registres.
- ii.  $Up_{m,a}$  un ensemble de mises à jour
- iii.  $\sim$  est une relation d'équivalence sur  $\mathcal{D}^k$  d'index fini.
- iv.  $\phi : (\Sigma \times \mathcal{D}^k / \sim)^* \rightarrow M$  un morphisme de monoïde

# Exemple de calcul



$\triangleright m' = \phi(a, \overline{\theta'})$ .

# Une première tentative

On considère les formules closes de la forme:  $\exists X_1 \dots \exists X_k. \psi$ .

▷ Syntaxe:

$$\psi ::= x < y \mid x = y \mid \exists x. \psi \mid \forall x. \psi \mid R_a x \mid X x \mid g(x_1, \dots, x_l) \mid \neg \psi \mid \psi \wedge \psi \mid \psi \vee \psi$$

$$::= MSO \mid g(x_1, \dots, x_l)$$

▷ Sémantique:

$$\text{Si } g \subset \mathcal{D}^l, (u; \mu. \nu) \models g(x_1, \dots, x_l) \iff (d_{\mu(x_1)}, \dots, d_{\mu(x_l)}) \in g$$

Mais ça ne marche pas: cette logique est trop expressive!

# Contre-exemple

Le langage de données

$$\{(a_1, d_1) \dots (a_n, d_n) \mid n \in \mathbb{N} \text{ et } \forall 1 \leq i < j \leq n, (a_i, d_i) \neq (a_j, d_j)\}$$

peut être décrit par la formule

$$\forall x. \forall y. (x < y \Rightarrow g_{\neq}(x, y) \vee \bigwedge_{a \in \Sigma} (\neg R_a x \vee \neg R_a y))$$

où  $g_{\neq} = \{(d, d') \mid d \neq d'\}$ , mais aucun mécanisme ne reconnaît ce langage.

# Caractérisation logique des langages reconnaissables

$\mathcal{L} = \{\text{formules closes de la forme } \exists X_1 \dots \exists X_k. \psi\}$

▷ Syntaxe:

$\psi ::= x < y \mid x = y \mid \exists x. \psi \mid \forall x. \psi \mid R_a x \mid X x \mid g(X_1, \dots, X_k, x) \mid \neg \psi \mid \psi \wedge \psi \mid \psi \vee \psi$

$::= MSO \mid g(X_1, \dots, X_k, x)$

▷ Sémantique:

$$(u; \mu.\nu) \models g(X_1, \dots, X_k, x) \iff (d_{\max(X_1 \leq x)}^{(\mu, \nu)}, \dots, d_{\max(X_k \leq x)}^{(\mu, \nu)}) \in g$$

Intuitivement, on a  $k$  registres.

La variable  $X_i$  contient les dates auxquelles le  $i$ ème registre est mis à jour.

A la date  $x$ , on regarde si le vecteur de registre est dans l'ensemble  $g$ .

**Théorème 1 (Caractérisation logique)** *Un langage de données est reconnaissable par un mécanisme à monoïdes si et seulement si il constitue l'ensemble des modèles d'une formule close de  $\mathcal{L}$ .*

# Décidabilité de la logique

- ▷ Décider du vide pour une formule de  $\mathcal{L}$  est un problème **indécidable**.
- ▷ On dispose d'une condition simple qui, dès qu'elle est vérifiée par une formule, assure qu'on peut décider du vide.

# Langages temporisés

▷ Tout langage temporisé reconnaissable (par un automate temporisé) peut être décrit par une formule de  $\mathcal{L}$  vérifiant cette condition.

## Langages réguliers

## Langages de données

Automates finis

Automates de données

Automates sans compteur

???

Expressions rationnelles

Adaptation du théorème de Kleene

Langages sans étoile

???

*MSO*

Logique du second ordre

*FO/LTL*

Logique du "premier ordre"

Monoïdes

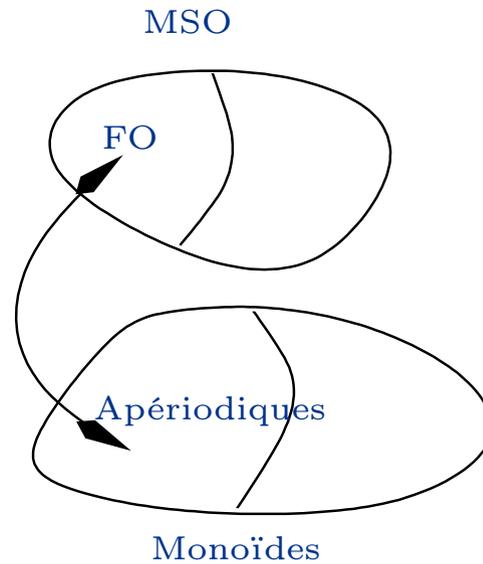
Monoïdes

Monoïdes apériodiques

Monoïdes apériodiques

# Langages réguliers

▷ Un monoïde  $M$  est **apériodique** si  $\forall x \in M, \exists n \in \mathbb{N}, x^n = x^{(n+1)}$ .



# Pour commencer

**Théorème 2** Soit  $L$  un langage de données, les deux énoncés suivants sont équivalents:

- i.  $L$  est définissable par une formule ne contenant pas de terme du type  $Xx$ .
- ii.  $L$  est reconnu par un mécanisme à monoïde apériodique avec  $Up_{m,a} = \mathcal{P}(1..k)$ .

▷ Si  $\phi = \exists X_1 \dots \exists X_k. Q_1 x_1 \dots Q_p x_p. \psi$  est une formule de  $\mathcal{L}$  définissant  $L$ .

▷ On transforme  $\psi$  en  $\overline{\psi}$  suivant les règles suivantes:

$$\begin{array}{ll} \overline{x < y} = x < y & \overline{g(X_1, \dots, X_k, x)} = R_g x \\ \overline{x = y} = x = y & \overline{\psi_1 \wedge \psi_2} = \overline{\psi_1} \wedge \overline{\psi_2} \\ \overline{R_a x} = R_a x & \overline{\neg \psi} = \neg \overline{\psi} \end{array}$$

▷ On considère  $Q_1 x_1 \dots Q_p x_p. \overline{\psi}$  qui s'interprète sur des mots de  $(\Sigma \times \mathcal{D}^k / \sim)$ .

- ▷ Soit  $\rho = (Up_{m,a}, k, f)$  sur le monoïde apériodique  $M$  tel que  $\rho$  reconnait  $L$ , en utilisant la partie acceptante  $F \subset M$ .
  
- ▷ Le morphisme  $f$  avec la partie  $F$  reconnait un langage  $\tilde{L} \subset (\Sigma \times \mathcal{D}^k / \sim)^*$  qui est définissable par une formule logique du premier ordre de la forme  $Q_1x_1..Q_px_p\bar{\psi}$ . On transforme  $\bar{\psi}$  en inversant les règles précédentes pour obtenir  $\psi$ .
  
- ▷ La formule  $\exists X_1..\exists X_k.Q_1x_1...Q_px_p\psi$  a comme ensemble de modèles les mots de  $L$ .

# Caractérisation des langages reconnus par un monoïde apériodique

Nous noterons  $\mathcal{L}_a$  l'ensemble des formules de la forme suivante:

$$\exists X_1 \dots \exists X_k. \psi \wedge \bigwedge_i \left( \forall x. \psi_{i,x} \Rightarrow \left( \bigwedge_{a \in \Sigma} R_a(x+1) \Rightarrow f_{a,i}(X_1(x+1), \dots, X_k(x+1)) \right) \right)$$

où  $f_{a,i}$  est une formule booléenne,  $\psi$  et les  $\psi_i$  sont des formules du premier ordre.

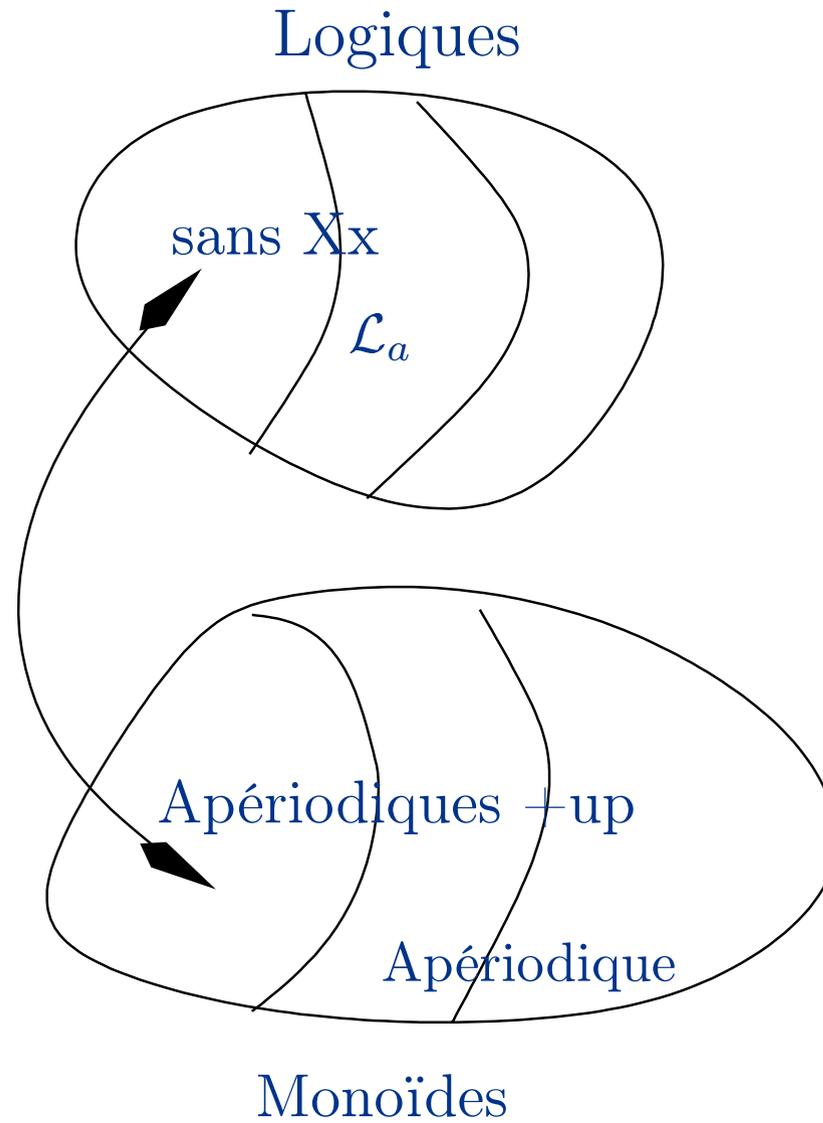
**Théorème 3** *Soit  $L$  un langage de données, les deux énoncés suivants sont équivalents:*

- i.  $L$  est définissable par une formule de  $\mathcal{L}_a$ .*
- ii.  $L$  est reconnu par un mécanisme à monoïde apériodique.*

▷ Le langage  $L = \{u = (a_1, d_1) \dots (a_n, d_n) \mid \exists i < j, a_i = a_j = a \text{ et } d_i = d_j \neq \perp\}$  sur l'alphabet  $\Sigma = \{a, b\}$  et sur un ensemble de données  $\mathcal{D}$  infini n'est reconnu par aucun mécanisme à monoïde apériodique où  $Up_{m,a} = \mathcal{P}(1..k)$  alors qu'il est reconnaissable par un mécanisme à monoïde apériodique.

▷ Le langage  $L = \{u = (a, d_1) \dots (a, d_n) \in (\{a\} \times \mathcal{D})^* \mid n \text{ est pair}\}$  n'est reconnu par aucun mécanisme à monoïde apériodique.

# Conclusion



# Conclusion

## Langages réguliers

## Langages de données

Automates finis

Automates de données

Automates sans compteur

???

Expressions rationnelles

Adaptation du théorème de Kleene

Langages sans étoile

???

*MSO*

Logique du second ordre

*FO/LTL*

Logique du "premier ordre"

Monoïdes

Monoïdes

Monoïdes apériodiques

Monoïdes apériodiques