LifLF – Théorie des langages formels

Sylvain Brandel

2019 – 2020

sylvain.brandel@univ-lyon1.fr

CM 7

LANGAGES RATIONNELS RATIONALITÉ

Langages rationnels

- Langage : ensemble de mots sur Σ
  - Éléments de base
    - L'ensemble $\emptyset$
    - Le mot vide ε
    - Les singletons sur Σ
  - Opérations
    - La concaténation de langages
    - La réunion de deux langages
    - L'intersection de deux langages
    - La fermeture de Kleene

- Langages rationnels
  - Représentation finie :
    - Éléments de base,
    - concaténation,
    - union,
    - fermeture de Kleene

Langages rationnels

- Expressions régulières sur Σ : plus petit ensemble E tel que
  - $\emptyset \in E$
  - $\varepsilon \in E$
  - Si $\sigma \in \Sigma$, alors $\sigma \in E$
  - Si $e_1, e_2 \in E$, alors $e_1 + e_2 \in E$ et $e_1 . e_2 \in E$
  - Si $e \in E$, alors $(e) \in E$, $e^* \in E$ et $e^+ \in E$

- Priorité : * > . > +

- Exemple
  - $\Sigma = \{a, b\}$ $(a + a . b)^* . a^* + \varepsilon$

Langages rationnels

- Langage représenté :
  - $[\![\emptyset]\!] = \emptyset$
  - $[\![\sigma]\!] = \{\sigma\}$ pour $\sigma \in \Sigma$
  - $[\![\varepsilon]\!] = \{\varepsilon\}$
  - $[\![e_1 + e_2]\!] = [\![e_1]\!] \cup [\![e_2]\!]$
  - $[\![e_1 . e_2]\!] = \{ w_1 w_2 \mid w_1 \in [\![e_1]\!]$ et $w_2 \in [\![e_2]\!] \}$
  - $[\![e^*]\!] = \bigcup_{n=0}^{\infty} [\![e^n]\!]$ où $e^0 = \{\varepsilon\}, e^1 = e, e^{n+1} = e . e^n$
  - $[\![e^+]\!] = \bigcup_{n=1}^{\infty} [\![e^n]\!]$ où $e^1 = e, e^{n+1} = e . e^n$

Langages rationnels

- Exemples

  - $\Sigma = \{a, b, c\}$ $\quad (a \,.\, a^{*} \,.\, c + (b + c))^{*} \,.\, a^{*}$ $\quad$ Mots ne contenant pas ab

  - $\Sigma = \{0, 1\}$ $\quad 0 + 1 \,.\, (0 + 1)^{*}$ $\quad$ Entiers en binaire

  - $\Sigma = \{0, 1\}$ $\quad 0 + 1 \,.\, (0 + 1)^{*} \,.\, 0$ $\quad$ Entiers en binaire pairs

  - $\Sigma = \{0, 1\}$ $\quad 0^{*} + (((0^{*} \,.\, (1 + (1.1))) \,.\, ((0 \,.\, 0^{*}) \,.\, (1 + (1 \,.\, 1)))^{*}) \,.\, 0^{*})$

    Mots ne contenant pas 111

Langages rationnels

- Système d'équations linéaires gauche :

$$\begin{cases} X_1 &= e_1^1 X_1 + \cdots + e_n^1 X_n + f^1 \\ & \vdots \\ X_n &= e_1^n X_1 + \cdots + e_n^n X_n + f^n \end{cases}$$

avec e_i^j expressions régulières $\in \Sigma \cup \{\varepsilon\} \cup \emptyset$

- Solution

$$X_1, \cdots, X_n \ si \ X_1 = [\![e_1^1]\!] \,.\, X_1 \cup \cdots \cup [\![e_n^1]\!] \,.\, X_n \cup [\![f^1]\!]$$

$$\vdots$$

$$X_n = [\![e_1^n]\!] \,.\, X_1 \cup \cdots \cup [\![e_n^n]\!] \,.\, X_n \cup [\![f^n]\!]$$

Langages rationnels

- Exemple

$$\begin{cases} X_1 & = & aX_2 + bX_3 + \varepsilon \\ X_2 & = & aX_1 + bX_4 \\ X_3 & = & bX_1 + aX_4 \\ X_4 & = & bX_2 + a\,X_3 \end{cases}$$

X_1 = { mots ayant un nombre pair de a et pair de b }

X_2 = { mots ayant un nombre impair de a et pair de b }

X_3 = { mots ayant un nombre pair de a et impair de b }

X_4 = { mots ayant un nombre impair de a et impair de b }

Langages rationnels

- Soit L un langage.

  L rationnel si ∃ S tq (**L**, L_1, L_2, …) est solution minimale de S

- Lemme

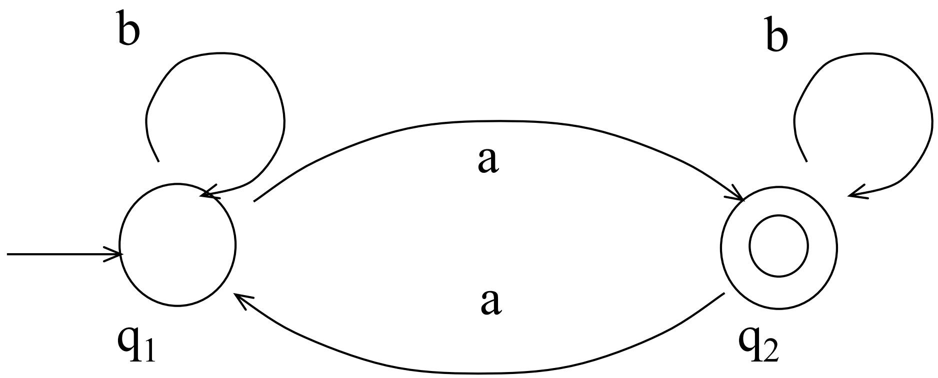
$X = e\,X + f$	avec e, f : expressions régulières	
$e^*.\,f$	est solution minimale de $X = e\,X + f$	si $\varepsilon \in$ e
	est solution unique de $X = e\,X + f$	si $\varepsilon \notin$ e

Preuve

- $e^*.\,f$ solution
- $e^*.\,f$ solution minimale
- $e^*.\,f$ solution unique si $\varepsilon \notin$ e

Langages rationnels

- Exemple



$$\begin{cases} X_1 = & bX_1 + aX_2 \\ X_2 = & bX_2 + aX_1 + \varepsilon \end{cases}$$

X_1 = b* a (b + a b* a)*

X_2 = (b + a b* a)*

Langages rationnels

- Exemple

$$\begin{cases} X_1 = & 1X_1 + 1X_2 + \varepsilon \\ X_2 = & 0X_1 \\ X_3 = & 0X_2 + 0X_3 + 1X_3 \end{cases}$$

X_1 = (1 + 10)*

X_2 = 0(1 + 10)*

X_3 = (0 + 1)* 00 (1 + 10)*

Rationalité

- Montrer qu'un langage est rationnel

(1) Stabilité (rappel : la classe des langages acceptés par un automate est stable par union, concaténation, fermeture itérative, complément, intersection)

(2) Caractérisation (rappel : un langage est rationnel ssi il est accepté par un automate)

(3) Un langage est rationnel ssi il peut être décrit par une expression rationnelle.

(2) et (3) → équivalence entre automate et ER

→ il existe des algorithmes

automate → ER

ER → automate

Rationalité

- Montrer qu'un langage est rationnel

  – A partir de (1) : utiliser les propriétés de stabilité

    → décomposer le langage en sous ensembles par union, intersection, concaténation,

    et montrer que ces sous ensembles sont rationnels.

  – A partir de (2) : construire un automate acceptant ce langage
    (on peut éventuellement déterminiser / minimiser cet automate)

  – A partir de (3) : construire une expression rationnelle décrivant ce langage

Non rationalité

- Il existe des langages non rationnels
  - L'ensemble des expressions régulières est dénombrable
  - L'ensemble des langages est non dénombrable

- Tout langage fini est rationnel (il peut être décrit par une ER composée de l'union de tous les mots du langage)

→ La question de non rationalité ne se pose que pour les langages infinis

- Montrer la non rationalité
  - Stabilité et raisonnement par l'absurde
  - Lemme de l'étoile

Non rationalité

Propriétés de stabilité

- Pour montrer que L est non rationnel :

  on pose l'hypothèse que L est rationnel

  et on détermine L_0 non rationnel et L_1 rationnel tels que

  $$L_0 = L \, \theta \, L_1 \quad (\theta \in \{\cap, \cup, .\}$$

- L supposé rationnel
- L_1 rationnel $\quad$ $L \, \theta \, L_1$ rationnel $\quad$ (stabilité de la classe des langages rationnels par θ)
- Or $L \, \theta \, L_1 = L_0$ avec L_0 connu (démontré) non rationnel

  ➔ Contradiction

  → l'hypothèse (L rationnel) est fausse

Non rationalité
Lemme de l'étoile

- Théorème Lemme de l'étoile

Soit L un langage rationnel infini accepté par un automate déterministe M à k états.
Soit z un mot quelconque de L tel que $|z| \geq k$.

Alors z peut être décomposé en uvw
avec $|uv| \leq k$, $|v| \neq 0$ et $uv^iw \in L$, $\forall i \geq 0$.

Non rationalité
Lemme de l'étoile

- Exemple : Montrons que $L = \{a^n b^n \mid n \geq 0\}$ est non rationnel.

Supposons que L est rationnel. L est reconnu par un automate M à k états.
D'après le lemme de l'étoile, $\forall\, z \in L, |z| \geq k, \exists\, u, v, w \in \Sigma^*$ tels que

$$z = uvw,\ |uv| \leq k,\ |v| > 0 \text{ et } \forall\, i \geq 0,\ uv^iw \in L$$

Soit $z_0 = a^k b^k$.
On a bien $z_0 \in L$ et $|z_0| = 2k \geq k$.
Toutes les décompositions possibles $z_0 = uvw$ telles que $|uv| \leq k$, $|v| > 0$
sont de la forme $u = a^p$, $v = a^q$, $w = a^r b^k$ avec $q > 0$ et $p+q+r = k$.

Or $uv^iw = a^p a^{qi} a^r b^k = a^{p+qi+r} b^k$
On a $\forall\, i \neq 1,\ p + qi + r \neq k$
Donc $\forall\, i \neq 1,\ uv^iw \notin L$
Donc contradiction dans la propriété
Donc l'hypothèse (L rationnel) est fausse
Donc L non rationnel

Complexité d'algorithmes pour les automates

- Théorèmes

(i) Il existe un algorithme exponentiel (en le nombre d'états)
Entrée : un automate fini non déterministe
Sortie : un automate fini déterministe équivalent

(ii) Il existe un algorithme polynomial (en fonction de la taille de l'expression ou du nombre d'opérateurs)
Entrée : une expression régulière
Sortie : un automate non déterministe équivalent

Complexité d'algorithmes pour les automates

(iii) Il existe un algorithme exponentiel (en fonction du nombre d'états)

Entrée : un automate non déterministe

Sortie : une expression régulière équivalente

(la taille des R(i, j, k) est multipliée par 4 à chaque incrément de k)

$$R(i, j, k) = R(i, j, k-1) \cup R(i, k, k-1)\, R(k, k, k-1)^*\, R(k, j, k-1)$$

(iv) Il existe un algorithme polynomial (en fonction du nombre d'états)

Entrée : un automate déterministe

Sortie : l'automate déterministe minimal (standard) équivalent

Complexité d'algorithmes pour les automates

(v) il existe un algorithme polynomial pour décider si deux automates déterministes sont équivalents

(Passe par l'automate standard)

(vi) Il existe un algorithme exponentiel pour déterminer si deux automates non déterministes son équivalents

Complexité d'algorithmes pour les automates

- Théorème

  L = langage rationnel (donné par un automate ou une expression régulière)

  et $w \in \Sigma^*$

  Il existe un algorithme qui teste si $w \in L$ avec une complexité en temps de $O(|w|)$

- Théorème

  $M = (K, \Sigma, \Delta, s, F)$ non déterministe et $w \in \Sigma^*$

  Il existe un algorithme qui teste si $w \in L(M)$ avec une complexité en temps de $O(|K|^2\,|w|)$