



Langages et automates (2ème partie)

Principales définitions et «recettes» pour rédiger des démonstrations correctes

Langages rationnels

Définition Soit A un alphabet. L'ensemble $Rat(A^*)$ des langages rationnels sur A est défini par induction structurelle par :

- (B) \emptyset , $\{\varepsilon\}$ et $\{a\}$, pour tout $a \in A$ sont des langages rationnels,
- (I) Si L_1 et L_2 sont des langages rationnels, alors $L_1 \cup L_2$ et $L_1.L_2$ sont aussi des langages rationnels, et si M est un langage rationnel, alors M^* est aussi un langage rationnel.

Montrons que L est rationnel.

Montrons que $L = \emptyset$

...

Montrons que L est rationnel.

Montrons que $L = \{\varepsilon\}$.

...

Montrons que L est rationnel.

Montrons que $L = \{a\}$, pour $a \in A$.

...

Montrons que L est rationnel.

Montrons que $L = L_1 \cup L_2$ à trouver

...

Montrons que L_1 est rationnel.

...

Montrons que L_2 est rationnel.

...

De l'hypothèse « L est un langage rationnel » on peut déduire
que $L = L_1 \cup L_2$ ou $L = L_1.L_2$ avec L_1 et L_2 rationnels
ou $L = M^*$ avec M rationnel
ou $L = \emptyset$ ou $L = \{\varepsilon\}$ ou $L = \{a\}$ pour $a \in A$

Montrons que L est rationnel.

Montrons que $L = L_1.L_2$ à trouver

...

Montrons que L_1 est rationnel.

...

Montrons que L_2 est rationnel.

...

Montrons que L est rationnel.

Montrons que $L = M^*$ à trouver

...

Montrons que M est rationnel.

...

Expressions rationnelles

Définition Soit A un alphabet. L'ensemble des expressions rationnelles sur A est défini par induction structurelle par

- (B) 0 est une expression rationnelle, et, pour tout $a \in A$, a est une expression rationnelle.
- (I) Si E est une expression rationnelle, alors $(E)^*$ est une expression rationnelle, et si E_1 et E_2 sont deux expressions rationnelles, alors $(E_1 + E_2)$ et $(E_1 \cdot E_2)$ sont des expressions rationnelles.

Définition Le langage associé à une expression rationnelle sur A est défini par induction structurale par

(B) $L(0) = \emptyset$, $L(a) = \{a\}$, pour tout $a \in A$.

(I) $L(E_1 + E_2) = L(E_1) \cup L(E_2)$, $L(E_1 \cdot E_2) = L(E_1) \cdot L(E_2)$ et $L(E^*) = L(E)^*$.

Théorème Un langage $L \subseteq A^*$ est *rationnel* si et seulement si il existe une expression rationnelle E telle que $L(E) = L$.

Montrons que L est rationnel.

Soit $E = \boxed{\text{à trouver}}$ une expression rationnelle.

Montrons que $L = L(E)$

...

Langages rationnels et langages reconnaissables

Théorème [Kleene (1956)] Un langage est rationnel si et seulement si il est reconnaissable.

Théorème [lemme d'Arden] Soient K et M deux langages tels que $\varepsilon \notin K$. L'équation $X = KX \cup M$ a pour unique solution $X = K^*M$

Soit $E = E_1 + E_2$ une expression rationnelle.

Construisons l'automate \mathcal{A} tel que $L(\mathcal{A}) = L(E)$

1- Construisons l'automate \mathcal{A}_1 tel que $L(\mathcal{A}_1) = L(E_1)$

...

2- Construisons l'automate \mathcal{A}_2 tel que $L(\mathcal{A}_2) = L(E_2)$

...

3- Construisons l'automate \mathcal{A} tel que $L(\mathcal{A}) = L(\mathcal{A}_1) \cup L(\mathcal{A}_2)$

...

Soit $E = E_1 \cdot E_2$ une expression rationnelle.

Construisons l'automate \mathcal{A} tel que $L(\mathcal{A}) = L(E)$

1- Construisons l'automate \mathcal{A}_1 tel que $L(\mathcal{A}_1) = L(E_1)$

...

2- Construisons l'automate \mathcal{A}_2 tel que $L(\mathcal{A}_2) = L(E_2)$

...

3- Construisons l'automate \mathcal{A} tel que $L(\mathcal{A}) = L(\mathcal{A}_1) \cdot L(\mathcal{A}_2)$

...

Soit $E = E_1^*$ une expression rationnelle.

Construisons l'automate \mathcal{A} tel que $L(\mathcal{A}) = L(E)$

1- Construisons l'automate \mathcal{A}_1 tel que $L(\mathcal{A}_1) = L(E_1)$

...

2- Construisons l'automate \mathcal{A} tel que $L(\mathcal{A}) = L(\mathcal{A}_1)^*$

...

Soit $E = 0$ une expression rationnelle.

1- Construisons l'automate \mathcal{A} tel que $L(\mathcal{A}) = \emptyset$

...

Soit $E = \varepsilon$ une expression rationnelle.

1- Construisons l'automate \mathcal{A} tel que $L(\mathcal{A}) = \{\varepsilon\}$

...

Soit $E = \{a\}$ une expression rationnelle.

1- Construisons l'automate \mathcal{A} tel que $L(\mathcal{A}) = \{a\}$

...

Soit $\mathcal{A} = (S, T, I, F)$ un automate, construisons une expression rationnelle E telle que $L(E) = L(\mathcal{A})$.

1- Pour tout $s \in S$ avec

$\{s \xrightarrow{a_1} s_1, s \xrightarrow{a_2} s_2, \dots, s \xrightarrow{a_p} s_p\}$,

l'ensemble des transitions sortant de s ,

on pose

$$L_s = \begin{cases} a_1 L_{s_1} \cdots + a_p L_{s_p} + \{\varepsilon\} & \text{si } s \text{ est un état final} \\ a_1 L_{s_1} + \cdots + a_p L_{s_p} & \text{sinon} \end{cases}$$

2- Résoudre le système d'équations à l'aide du lemme d'Arden

3- $L = \sum_{s \in I} L_s$

Langages non reconnaissables

Théorème Le langage $L = \{a^n b^n, n \in \mathbb{N}\}$ n'est pas reconnaissable.

Montrons que L n'est pas reconnaissable.

1- Supposons qu'il existe un automate $\mathcal{A} = (S, T, I, F)$
tel que $L(\mathcal{A}) = L$

1-1 Soit $w =$ à trouver
(dépend souvent du nombre d'états de S).

Montrons que $w \in L$.

...

Donc \mathcal{A} a une exécution acceptante sur w .

1-2 Soit $w' =$ à trouver
(vient d'une boucle dans l'exécution de \mathcal{A} sur w)

Montrons que $w' \in L(\mathcal{A})$.

...

1-3 Montrons que $w' \notin L$.

...

Montrons que L n'est pas reconnaissable.

1- Supposons que L est reconnaissable.

1-1 Montrons que $L_2 = \text{op}(L, L_1)$

op étant une opération préservant la reconnaissabilité

op, L_1, L_2 à trouver

...

1-2 Montrons que L_1 est reconnaissable.

...

1-2 Montrons que L_2 n'est pas reconnaissable.

...

Minimisation d'automates

Soit $\mathcal{A} = (S, T, i, F)$ un automate déterministe et complet. Pour tout $s \in S, u \in A^*$, on note $s \cdot u$ l'unique état s' tel que $s \xrightarrow{u} s'$.

Définition Soit $\mathcal{A} = (S, T, i, F)$ un automate déterministe et complet. Alors, pour tous $s, s' \in S$, on note $s \sim s'$ si et seulement si $L_s = L_{s'}$. Autrement dit, si, pour tout $u \in A^*$, $s \cdot u \in F$ si et seulement si $s' \cdot u \in F$.

Montrons que $s \not\sim s'$

Soit $u \in A^*$ à trouver.

Montrons que $s \cdot u \in F$ et $s' \cdot u \notin F$

...

Montrons que $s \not\sim s'$

Soit $u \in A^*$ à trouver.

Montrons que $s \cdot u \notin F$ et $s' \cdot u \in F$

...

De l'hypothèse « $s \sim s'$ », on peut déduire

que pour tout mot $u \in A^*$, $u \in L_s$ si et seulement si $u \in L_{s'}$

Théorème La relation $\sim \subseteq S \times S$ est une relation d'équivalence (l'équivalence de Nérode). De plus, pour tout $a \in A$, si $s_1 \sim s_2$, alors $s_1 \cdot a \sim s_2 \cdot a$, et si $s_1 \sim s_2$ alors $s_1 \in F$ si et seulement si $s_2 \in F$.

Algorithme de Moore pour calculer les classes d'équivalence

— Pour tous $s, s' \in S$, $s \sim_0 s'$ si $s \in F$ si et seulement si $s' \in F$.

— Pour tous $s, s' \in S$, pour tout $k \geq 0$, $s \sim_{k+1} s'$ si (i) $s \sim_k s'$ et (ii) pour tout $a \in A$, $s \cdot a \sim_k s' \cdot a$.

Il existe un entier $p \in \mathbb{N}$ tel que $\sim_{p+1} = \sim_p$. Alors l'équivalence de Nérode $\sim = \sim_p$.

Définition Soit $\mathcal{A} = (S, T, i, F)$ un automate déterministe et complet. Soit \sim l'équivalence de Nérode. L'automate quotient de \mathcal{A} par \sim est défini par $\mathcal{A}/\sim = (S/\sim, [i], \hat{T}, F/\sim)$, avec $\hat{T} = \{([s], a, [s']) \mid (s, a, s') \in T\}$, et $[s]$ représentant la classe d'équivalence de s pour \sim .

Théorème Soit \mathcal{A} un automate déterministe, complet et *monogène* (i.e. dont tous les états sont accessibles depuis l'état initial). Alors $L(\mathcal{A}/\sim) = L$ et \mathcal{A}/\sim est l'automate minimal de \mathcal{A} .