Analyse Syntaxique – CTD - Partie 1

Septembre 2018

Langages Rationnels

1. Langages : généralités

Un **alphabet** est un ensemble non vide dont les éléments sont appelés **lettres**.

EXEMPLES :

$$\begin{aligned} \mathcal{A} &= \{a, b, d\} \\ \mathcal{B} &= \{0, 1\} \\ \mathcal{G} &= \{\text{alpha}, \text{delta}, \text{sigma}, \text{ksi}\} \end{aligned}$$

Un **mot** est une suite finie de lettres.

EXEMPLES :
$m_1 = aba$ est un mot sur l'alphabet $\mathcal{A}$.
$m_2 = 00101$ mot sur $\mathcal{B}$.
$m_3 = $ sigma.delta.delta mot sur $\mathcal{G}$

Le **mot vide** est la suite vide de lettre. On le note ε. (Attention, ne pas confondre ε avec une lettre : c'est un mot).

Opération de concaténation : Soient deux mots $m_1 = \alpha_1 \cdots \alpha_n$ et $m_2 = \beta_1, \cdots, \beta_p$, avec les α_i et les β_j lettres d'un alphabet donnée X. Alors, le résultat de la concaténation de m_1 et m_2, noté $m_1.m_2$ est le mot :

$$m_1.m_2 = \alpha_1 \cdots \alpha_n \beta_1, \cdots, \beta_p$$

Cette opération est associative, n'est pas commutative, et admet le mot vide pour élément neutre.

La **longueur** du mot m, notée $\mid m \mid$ est le nombre de lettres du mot m.

EXEMPLES :
$|m_1| = |aba| = 3 = |m_3|$
$|m_2| = |00101| = 5$
$|\varepsilon| = 0$

Un **langage** sur un alphabet X est un ensemble de mots sur X.

EXEMPLES : Sur l'alphabet $\mathcal{B}$

- $L_1 = \{00, 010, 11, 1010\}$
- $L_2 = \{m \ / \ m \text{ est un multiple de 3 en base 2}\}$
- $L_3 = \{m \ / \ m \text{ se termine par } 0\}$

2. Les langages rationnels

Les **langages rationnels** sont les langages définissables par une **expression rationnelle**.

On définit les expressions rationnelles par induction :

Soit X un alphabet.
- $\emptyset$ est une expression rationnelle.
- ε est une expression rationnelle.
- si a lettre de X alors a est une expression rationnelle.
- si e_1 et e_2 sont des expressions rationnelles, alors $e_1.e_2$, $e_1 + e_2$ sont des expressions rationnelles.
- si e expression rationnelle, alors (e) et e^* sont des expressions rationnelles.

concaténation de langages :

$$L_1.L_2 = \{m_1.m_2 \ / \ m_1 \in L_1, m_2 \in L_2\}$$

puissance d'un langage :
Soit L un langage sur un alphabet X
- $L^0 = \{\varepsilon\}$
- $L^1 = L$
- $L^i = L^{i-1}.L$ pour $i > 0$
- $L^* = \bigcup_{i \geq 0} L^i$
- $L^+ = \bigcup_{i > 0} L^i = L.L^*$

Les expressions rationnelles définissent les langages rationnels, de la façon suivante :
- $\emptyset$ définit le langage vide.
- ε définit le langage $\{\varepsilon\}$.
- si a lettre de X alors a définit le langage $\{a\}$.
- si e_1 et e_2 sont des expressions rationnelles qui définissent respectivement L_1 et L_2, alors $e_1.e_2$ définit $L_1.L_2$, et $e_1 + e_2$ définit $L_1 \cup L_2$.
- si e expression rationnelle qui définit L, alors (e) définit L et e^* définit L^*.

Langages reconnaissables

3. Automates finis

Un **automate fini de mots** est un quintuplet (A, Q, I, F, δ) avec
- — A alphabet
- — Q ensemble d'états
- — I ensemble non vide d'états initiaux. $I \subseteq Q$.
- — F ensemble d'états finaux. $F \subseteq Q$
- — δ fonction de transition. $\delta : Q \times A \cup \{\varepsilon\} \to 2^Q$

2^Q désigne ici l'ensemble des parties de Q (dénoté aussi $\mathcal{P}(Q)$). La définition ci-dessus concerne les automates **non déterministes** : plusieurs états initiaux, transitions par le mot vide ε, plusieurs états image pour un couple (q, a) fixé (car $\delta(q, a)$ est une partie de Q).

On note $\delta^*(q, m)$ l'ensemble des états atteints en lisant m à partir de q. Si $m = a.m'$ avec $a \in A \cup \{\varepsilon\}$, alors $\delta^*(q, m) = \bigcup_{e \in \delta(q,a)} \delta^*(e, m')$.

Un automate $M = (A, Q, I, F, \delta)$ est **complet** si $\forall q \in Q,\ \ \forall a \in A,\ \delta(q, a)$ a au moins 1 élément.

Un mot m est **reconnu** par l'automate $M = (A, Q, I, F, \delta)$ si et seulement si

$$\exists q_0 \in I \quad \delta^*(q_0, m) \cap F \neq \emptyset$$

Le **langage** $L(M)$ **reconnu par un automate** M est l'ensemble des mots reconnus par M.

4. Langages reconnaissables

Un langage est **reconnaissable** s'il existe un automate qui le reconnaît.

Propriétés de clôture :

On note *Rec(X^*)* la famille des langages reconnaissables sur un alphabet X. Par définition, un langage L sur l'alphabet X est dans cette famille s'il existe un automate M qui le reconnaît.
- — *Rec(X^*)* est fermée par produit de concaténation
- — *Rec(X^*)* est fermée par étoile
- — *Rec(X^*)* est fermée par complémentaire
- — *Rec(X^*)* est fermée par union
- — *Rec(X^*)* est fermée par intersection.

Théorème de Kleene

On note *Rat(X^*)* la famille des langages rationnels sur l'alphabet X.

$$Rat(X^*) = Rec(X^*)$$

Automates finis

5. Description des automates

On donne les quatres ensembles A, Q, I et F : par exemple $A = \{0,1\}$, $Q = \{q_1, q_2, q_3, q_4\}$, $I = \{q_1\}$, $F = \{q_4\}$. La fonction de transition peut se représenter de 4 manières différentes :

— **extensive** :
$$\begin{array}{lll} \delta(q_1,0) = \{q_1\} & \delta(q_1,1) = \{q_1,q_2\} & \delta(q_1,\varepsilon) = \emptyset \\ \delta(q_2,0) = \{q_3\} & \delta(q_2,1) = \emptyset & \delta(q_2,\varepsilon) = \{q_3\} \\ \delta(q_3,0) = \emptyset & \delta(q_3,1) = \{q_4\} & \delta(q_3,\varepsilon) = \emptyset \\ \delta(q_4,0) = \{q_4\} & \delta(q_4,1) = \{q_4\} & \delta(q_4,\varepsilon) = \emptyset \end{array}$$

— **table de transition** : une ligne i par état q_i, une colonne j par symbole a_j de A plus une colone pour le mot vide ε, la cellule (i,j) contient l'ensemble $\delta(q_i, a_j)$.

— **table de passage** : une ligne i par état q_i, une colonne j par état q_j, la cellule (i,j) contient l'ensemble des symboles permettant de passer de q_i à q_j

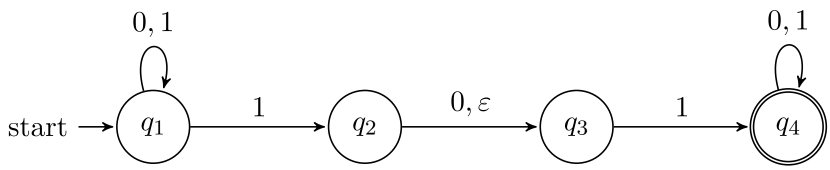
— **graphe** : un nœud par état q_i, une arête orientée et étiquettée a de q_i vers q_j si $q_j \in \delta(q_i, a)$. Une flèche entrante sans étiquette pour les états initiaux, un "double cerclage" pour les états finaux.

δ	0	1	ε
q_1	$\{q_1\}$	$\{q_1,q_2\}$	$\emptyset$
q_2	$\{q_3\}$	$\emptyset$	$\{q_3\}$
q_3	$\emptyset$	$\{q_4\}$	$\emptyset$
q_4	$\{q_4\}$	$\{q_4\}$	$\emptyset$

(a) Table de transition

$\nearrow$	q_1	q_2	q_3	q_4
q_1	$0,1$	1	-	-
q_2	-	-	$0,\varepsilon$	-
q_3	-	-	-	1
q_4	-	-	-	$0,1$

(b) Table de passage



(c) Graphe

FIGURE 1 – Représentations de la fonction de transition pour l'exemple d'automate.

6. Automate déterministe

Un automate $M = (A, Q, I, F, \delta)$ est **déterministe** si
— I est un singleton $\{q_0\}$
— $\forall q \in Q, \quad \forall a \in A$, $\delta(q, a)$ a au plus 1 élément.

Ainsi un automate déterministe est un quintuplet (A, Q, q_0, F, δ) avec
— A alphabet
— Q ensemble d'états
— $q_0 \in Q$ l'état initial.
— F ensemble d'états finaux. $F \subseteq Q$
— δ fonction de transition. $\delta : Q \times A \to Q$

$\delta^*(q, m)$, l'ensemble des états atteints en lisant m à partir de q est alors défini par

— Si $m = \varepsilon$, $\delta^*(q, \varepsilon) = \{q\}$
— Sinon, $m = a.m'$ et alors $\delta^*(q, m) = \delta^*(e, m')$ avec $e = \delta(q, a)$.

Un mot m est **reconnu** par l'automate déterministe $M = (A, Q, q_0, F, \delta)$ si et seulement si

$$\delta^*(q_0, m) \in F$$

7. Déterminisation

Théorème : Soit M un automate non déterministe.
Il existe un automate M' déterministe qui reconnaît $L(M)$.

Soit $M = (A, Q, I, F, \delta)$ un automate non déterministe.

a-successeurs

Par définition $\delta(q, a)$ représente l'ensemble des successeurs immédiats de q par le symbole a. On étend cette notion aux parties de Q. Soit $T \subseteq Q$ une partie de Q,

$$succ(T, a) = \bigcup_{t \in T} \delta(t, a)$$

C'est l'ensemble des états que l'on peut atteindre à partir des états de T par le symbole a.

ε-fermeture

Soit $q \in Q$, l'ε-fermeture de q, $\varepsilon\text{-}clos(q) = \delta^*(q, \varepsilon^*)$ est l'ensemble des états que l'on peut atteindre en faisant 0, 1 ou plusieurs ε-transitions. Pratiquemment $\varepsilon\text{-}clos(q) = \{q\} \cup \delta(q, \varepsilon) \cup succ(\delta(q, \varepsilon), \varepsilon) \ldots$ et se calcule incrémentalement jusqu'à ne plus rajouter de nouvel état.
Comme précédemment on étend cette notion aux parties de Q. Soit $T \subseteq Q$ une partie de Q,

$$\varepsilon\text{-}clos(T) = \bigcup_{t \in T} \varepsilon\text{-}clos(t)$$

L'**automate déterministe** M' **qui reconnaît** $L(M)$ est défini par le quintuplet $(A', Q', q'_0, F', \delta')$:

$A' = A$	même alphabet
$Q' = 2^Q$	ensemble d'états = ensemble des parties de Q
$q'_0 = \varepsilon\text{-}clos(I)$	état initial = ε-fermeture de I
$F' = \{T \mid T \cap F \neq \emptyset\}$	$T \in 2^Q$ est final si il contient un état final de M
$\delta'(T, a) = \varepsilon\text{-}clos(succ(T, a))$	$T \in 2^Q$, son successeur par a dans M' est l'ε-fermeture des successeurs de T par a dans M

En pratique on commence avec $Q' = \emptyset$ et on calcule l'état initial $q'_0 = \varepsilon\text{-}clos(I)$ que l'on ajoute à Q'. Puis, pour chaque état q' de Q' non traité, et pour chaque symbole a de A, on calcule $\delta'(q', a) = \varepsilon\text{-}clos(succ(q', a))$ et on l'ajoute à Q'. On itère jusqu'à avoir traité tous les états.

Sur l'exemple précédent, on aura ainsi :
— $q'_0 = \varepsilon\text{-}clos(\{q_1\}) = \{q_1\}$
— $\delta'(\{q_1\}, 0) = \varepsilon\text{-}clos(succ(\{q_1\}, 0)) = \{q_1\}$
— $\delta'(\{q_1\}, 1) = \varepsilon\text{-}clos(succ(\{q_1\}, 1)) = \varepsilon\text{-}clos((\{q_1, q_2\}) = \{q_1, q_2, q_3\}$
— $\delta'(\{q_1, q_2, q_3\}, 0) = \varepsilon\text{-}clos(succ(\{q_1, q_2, q_3\}, 0)) = \{q_1, q_3\}$
— $\delta'(\{q_1, q_2, q_3\}, 1) = \varepsilon\text{-}clos(succ(\{q_1, q_2, q_3\}, 1)) = \{q_1, q_2, q_3, q_4\}$
— $\delta'(\{q_1, q_3\}, 0) = \varepsilon\text{-}clos(succ(\{q_1, q_3\}, 0)) = \{q_1\}$
— $\delta'(\{q_1, q_3\}, 1) = \varepsilon\text{-}clos(succ(\{q_1, q_3\}, 1)) = \{q_1, q_2, q_3, q_4\}$
— $\delta'(\{q_1, q_2, q_3, q_4\}, 0) = \varepsilon\text{-}clos(succ(\{q_1, q_2, q_3, q_4\}, 0)) = \{q_1, q_3, q_4\}$

— $\delta'(\{q_1, q_2, q_3, q_4\}, 1) = \varepsilon\text{-}clos(succ(\{q_1, q_2, q_3, q_4\}, 1)) = \{q_1, q_2, q_3, q_4\}$
— $\delta'(\{q_1, q_3, q_4\}, 0) = \varepsilon\text{-}clos(succ(\{q_1, q_3, q_4\}, 0)) = \{q_1, q_4\}$
— $\delta'(\{q_1, q_3, q_4\}, 1) = \varepsilon\text{-}clos(succ(\{q_1, q_3, q_4\}, 1)) = \{q_1, q_2, q_3, q_4\}$
— $\delta'(\{q_1, q_4\}, 0) = \varepsilon\text{-}clos(succ(\{q_1, q_4\}, 0)) = \{q_1, q_4\}$
— $\delta'(\{q_1, q_4\}, 1) = \varepsilon\text{-}clos(succ(\{q_1, q_4\}, 1)) = \{q_1, q_2, q_3, q_4\}$

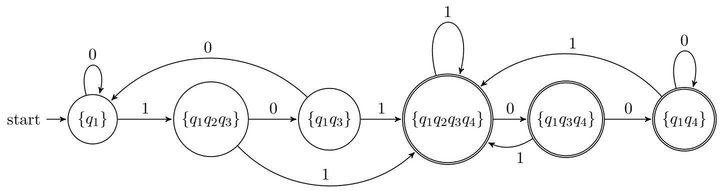


FIGURE 2 – Le déterminisé de l'exemple

8. Minimisation

Relation d'équivalence Une relation d'équivalence sur un ensemble X est une relation binaire sur X, c'est-à-dire une partie R de $X \times X$, qui est :
— réflexive : $x\,R\,x$ pour tout $x \in X$;
— symétrique : $x\,R\,y \implies y\,R\,x$ pour tous $x, y \in X$;
— transitive : $(x\,R\,y$ et $y\,R\,z) \implies x\,R\,z$ pour tous $x, y, z \in X$.

Congruence Soit $M = (A, Q, q_0, F, \delta)$ un automate déterministe. Une relation d'équivalence $\sim$ sur Q est une congruence sur M si :

i elle est compatible avec δ : $(p \sim q) \implies (\forall a \in A : \delta(p, a) \sim \delta(q, a))$;

ii elle sature F : $(p \sim q) \implies (p \in F \Leftrightarrow q \in F)$.

Automate quotient Soit $M = (A, Q, q_0, F, \delta)$ un automate déterministe et $\sim$ une congruence sur M . L'automate quotient de M par $\sim$ est noté $M/\sim$ et défini par $M/\sim = (A, [Q], [q0], [F], [\delta])$ où :

i $[Q] = Q/\sim = \{[q], q \in Q\}$;

ii $[\delta]([q], a) = [\delta(q, a)]$ pour tout $[q] \in [Q]$ et tout $a \in A$;

iii $[q_0]$ est la classe de q_0 modulo $\sim$;

iv $[F] = F/\sim = \{[q], q \in F\}$

Lemme Pour toute congruence $\sim$ sur un automate déterministe M on a :

$$L(M/\sim) = L(M)$$

Langages L_q Pour chaque état $q \in Q$ d'un automate $M = (A, Q, I, F, \delta)$, on note L_q l'ensemble des mots w tels que le calcul de M sur w à partir de q termine dans un état acceptant. Autrement dit : $L_q = \{w \in A^* | \delta^*(q, w) \cap F \neq \emptyset\}$. En particulier, $L_{q_0} = L(M)$.

Congruence de Nérode Soit $M = (A, Q, q_0, F, \delta)$ un automate déterministe. La congruence de Nérode de M est la relation d'équivalence $\cong$ définie sur Q par :

$$p \cong q \text{ si } L_p = L_q$$

Automate minimal Soit $L \subseteq A^*$ un langage reconnaissable. L'automate minimal de L est, parmi les automates déterministes complets reconnaissant L, celui qui minimise le nombre d'états.

Proposition Soit M un automate déterministe complet reconnaissant un langage L. L'automate minimal M_L de L est égal au quotient $M/\cong$, où $\cong$ est la congruence de Nérode de M.

Calcul de $M/\cong$ On calcule la congruence de Nérode d'un automate déterminisite $M = (A, Q, q_0, F, \delta)$ en considérant la suite de relations d'équivalence suivante :

— $p \sim_0 q$ ssi $p, q \in F$ ou $p, q \notin F$. Ainsi, $\sim_0$ définit deux classes d'équivalences exactement : F et $Q \setminus F$.

— $p \sim_{n+1} q$ ssi $(p \sim_n q \; et \; \forall a \in A : \delta(p, a) \sim_n \delta(q, a))$

Autrement dit, les classes de $\sim_{n+1}$ sont obtenues en séparant, dans chaque classe de $\sim_n$, les états qui sont "envoyés" par une même lettre sur des $\sim_n$-classes différentes.

Sur l'exemple précédent, on aura ainsi, en renommant de gauche à droite les états de q_1 à q_6 :

— $\sim_0$: $\{q_1, q_2, q_3\}, \{q_4, q_5, q_6\}$
— $\sim_1$: $\{q_1\}, \{q_2, q_3\}, \{q_4, q_5, q_6\}$
— $\sim_2$: $\{q_1\}, \{q_2\}, \{q_3\}, \{q_4, q_5, q_6\}$
— $\sim_3$: $=\sim_2$

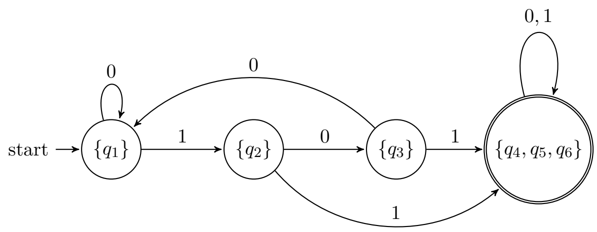


FIGURE 3 – L'automate minimal de l'exemple

Calcul de $M/\cong$ en utilisant un tableau Il est possible de formaliser l'algorithme ci-dessus en utilisant un tableau représentant les paires d'états :

— Marquer toutes les paires (p, q) avec p un état final et q un état non final, ou inversement. On utilise en fait qu'une moitié du tableau par symétrie.

— Répéter
  pour toute paire (p, q) non marquée
    pour toute lettre a de A
      soit $r = \delta(p, a)$ et $s = \delta(q, a)$
      si la paire (r, s) est marquée, alors marquer la paire (p, q)
  jusqu'à ce que plus aucune paire ne soit marquée

— les paires d'état non marquées forment les classes d'équivalence

Sur l'exemple précédent, on construit le tableau étape par étape de cette façon

	q_1	q_2	q_3	q_4	q_5	q_6
q_1	–	–	–	–	–	–
q_2		–	–	–	–	–
q_3			–	–	–	–
q_4	X	X	X	–	–	–
q_5	X	X	X		–	–
q_6	X	X	X			–

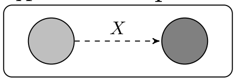
	q_1	q_2	q_3	q_4	q_5	q_6
q_1	–	–	–	–	–	–
q_2	X	–	–	–	–	–
q_3	X		–	–	–	–
q_4	X	X	X	–	–	–
q_5	X	X	X		–	–
q_6	X	X	X			–

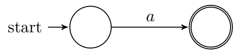
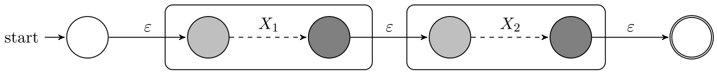
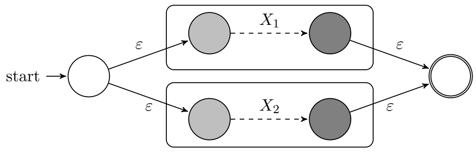
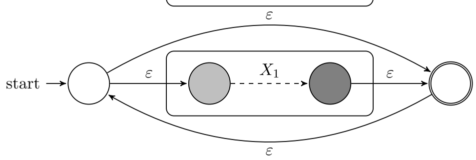
	q_1	q_2	q_3	q_4	q_5	q_6
q_1	–	–	–	–	–	–
q_2	X	–	–	–	–	–
q_3	X	X	–	–	–	–
q_4	X	X	X	–	–	–
q_5	X	X	X		–	–
q_6	X	X	X			–

En énumérant les paires non marquées, (q_4, q_5), (q_4, q_6) et (q_5, q_6), que l'on regroupe par transitivité dans un seul état de l'automate minimal, on obtient la même partition $\{q_1\}, \{q_2\}, \{q_3\}, \{q_4, q_5, q_6\}$ que précédemment.

9. Des expressions aux automates

Soit une expression régulière X, on construit un **automate non-déterministe** M_X par induction sur la structure de X. Dans cette construction on fait en sorte que les automates M_X n'aient qu'un seul état initial et un seul état final. Nous les représenterons sous la forme



Expression régulière	Automate correspondant
$X = a$ avec $a \in A$	
$X = X_1.X_2$	
$X = X_1 + X_2$	
$X = X_1^*$	

10. Des automates aux expressions

Soit $M = (A, Q, q_0, F, \delta)$ un automate déterministe.

Equation associée au langage L_q Pour tout état q de Q, on définit l'équation associée à L_q par

- $L_q = \bigcup_{(a_i,p_i)|\delta(q,a_i)=p_i} a_i L_{p_i}$ si $q \notin F$
- $L_q = \bigcup_{(a_i,p_i)|\delta(q,a_i)=p_i} a_i L_{p_i} \cup \{\varepsilon\}$ si $q \in F$

Système d'équations associé à un automate Le système d'équations associé à M est le système composé des équations associées à chacun des L_q pour $q \in Q$.

Lemme d'Arden Soient L_1, L_2 deux langages sur un alphabet A tels que $\varepsilon \notin L_1$. Alors l'équation $X = L_1 X \cup L_2$ admet $L_1^* L_2$ comme unique solution.
En pratique, cherchant des expressions rationnelles que l'on notera X_i, on remplace les langages L_i par X_i, l'union $\cup$ des langages par $+$, ...

Sur l'exemple de l'automate minimal précédent, on obtient le système suivant (en renommant q_4 le dernier état) :

- $X_1 = 0X_1 + 1X_2$
- $X_2 = 0X_3 + 1X_4$
- $X_3 = 0X_1 + 1X_4$
- $X_4 = 0X_4 + 1X_4 + \varepsilon$

que l'on résout ainsi :

- $X_4 = (0+1)X_4 + \varepsilon$ soit, par le lemme d'Arden, $X_4 = (0+1)^*$
- $X_1 = 0X_1 + 1X_2 = 0X_1 + 10X_3 + 11X_4 = 0X_1 + 100X_1 + 101X_4 + 11X_4 = (0+100)X_1 + (11+101)X_4$ soit, par le lemme d'Arden, $X_1 = (0+100)^*(11+101)X_4 = (0+100)^*(11+101)(0+1)^* = L(M)$