Université de Paris | UFR Informatique
Automates et Analyse Lexicale | Année 2021–2022

Handout Cours 3

Automates Déterministes et Non Déterministes

1 Complétion d'un automate

Un automate est *complet* quand sa fonction de transition est une fonction totale.

Étant donné un automate $A = (\Sigma, Q, q_0, F, \delta)$ on définit sa *complétion* par $A_c = (\Sigma, Q \cup \{\perp\}, q_0, F, \delta_c)$ avec pour tous $a \in \Sigma$:

$$\begin{array}{rcll} \delta_c(q,a) & = & \delta(q,a) & \text{si } q \in Q,\ \delta(q,a) \in Q \\ \delta_c(q,a) & = & \perp & \text{si } q \in Q,\ \delta(q,a) \text{ indéfini} \\ \delta_c(\perp,a) & = & \perp & \end{array}$$

Il est sous-entendu que $\perp \notin Q$ (sinon il faut choisir un autre symbole). On a que $\mathcal{L}(A) = \mathcal{L}(A_c)$.

2 Complément d'un langage reconnaissable

La classe *Rec* des langages reconnaissables est close sous complément : Soit $A = (\Sigma, Q, q_0, F, \delta)$ un automate déterministe *complet* (il est important qu'il est complet !), alors on peut définir son complément $A^{comp} = (\Sigma, Q, q_0, Q - F, \delta)$. On a que $(\mathcal{L}(A))^{comp} = \mathcal{L}(A^{comp})$.

Par contre il n'est pas évident pour le moment si *Rec* est close sous union, concaténation, et étoile de Kleene.

3 Automates Non Déterministes

Un *automate fini non déterministe* (AFND) est un quintuplet $(\Sigma, Q, I, F, \delta)$ tel que

- Σ est un alphabet fini ;
- Q est un ensemble fini, appelé l'ensemble des *états* ;
- $I \subseteq Q$, appelé *ensemble des états initiaux* ;
- $F \subseteq Q$, appelé *ensemble des états acceptants* ;
- $\delta \colon Q \times \Sigma \mapsto 2^Q$ est une fonction, appelée la *fonction de transition.*

Dans un automate non déterministe, la fonction de transition associe à un état et une lettre un *ensemble* d'états. Cet ensemble peut être vide.

On se permet de considérer les AFD comme un cas particulier des AFND, même s'il y a formellement une différence dans les types.

Intuitivement, pour exécuter un AFND sur un mot il faut exploiter *toutes* les possibilités de suivre les flèches. Un AFND accepte un mot quand il y a *au moins une* possibilité d'atteindre un état acceptant à partir d'un état initial.

Formellement : Étant donné un AFND $(\Sigma, Q, I, F, \delta)$, on définit une fonction $\delta^*\colon Q \times \Sigma^* \mapsto 2^Q$ par récurrence sur le deuxième argument :

$$\begin{array}{rcl} \delta^*(q, \epsilon) & = & \{q\} \\ \delta^*(q, aw) & = & \bigcup_{p \in \delta(q,a)} \delta^*(p, w) \end{array}$$

Le langage *reconnu* par l'AFND A est

$$\mathcal{L}(A) = \{w \in \Sigma^* \mid \text{il existe } q_0 \in I \text{ tel que } \delta^*(q_0, w) \cap F \neq \emptyset\}$$

L'AFND A *accepte* un mot w si $w \in \mathcal{L}(A)$.

4 Déterminisation

Pour tout AFND $A = (\Sigma, Q, I, F, \delta)$ il existe un AFD $A' = (\Sigma, Q', q_0', F', \delta')$ tel que $\mathcal{L}(A') = \mathcal{L}(A)$.

La construction de A' est comme suit :

- $Q' = 2^Q$, donc tout sous-ensemble de Q' est un élément de Q.
- $q_0' = I$
- $F' = \{P \subseteq Q \mid P \cap F \neq \emptyset\}$
- $\delta'(P, a) = \bigcup_{p \in Q} \delta(p, a)$

Donc, si l'automate non déterministe a n état, l'automate déterministe correspondant en a 2^n.

On obtient comme corollaire que *Rec* est close sous union : étant donné deux AFD $A_1 = (\Sigma, Q_1, q_1, F_1, \delta_1)$ et $A_2 = (\Sigma, Q_2, q_2, F_2, \delta_2)$, on s'assure d'abord que Q_1 et Q_2 sont disjoints. Puis on défini un AFND

$$A_{12} = (\Sigma, Q_1 \cup Q_2, \{q_1, q_2\}, F_1 \cup F_2, \delta_{12})$$

où

$$\delta_{12}(q, a) = \begin{cases} \delta_1(q, a) & \text{si } q \in Q_1 \\ \delta_2(q, a) & \text{si } q \in Q_2 \end{cases}$$

puis on déterminise l'automate A_{12}.

Puisque *Rec* est close sous complément et union, elle est aussi close sous intersection, car $L_1 \cap L_2 = \left(L_1^{comp} \cup L_2^{comp}\right)^{comp}$.