Constructions sur les langages réguliers

1 L'automate miroir

Soit L un langage d'états finis. On note $\widetilde{L}$ l'ensemble des mots qui s'écrivent comme le miroir d'un mot de L, c'est-à-dire :

$$\widetilde{L} = \{\alpha_n \dots \alpha_1 | \alpha_1 \dots \alpha_n \in L\}$$

Montrons que $\widetilde{L}$ est d'états finis.

Remarque 1 L'intuition de la construction est que si un mot est dans le langage miroir, alors il se lit "à l'envers´´ dans l'automate initial, en partant d'un état final et en terminant dans l'état q_0.

Plus formellement Comme L est un langage d'états finis, alors il existe un automate fini le reconnaissant, donc, quitte à le déterminiser, on peut dire qu'il existe un certain automate déterministe $\mathcal{A}$ tel que $\mathcal{L}(\mathcal{A}) = L$. Notons $(Q, \Sigma, q_0, \delta, F)$ cet automate (δ est une fonction, c'est à dire qu'à tout couple (q, a) elle associe un unique état successeur q'). Sur la figure 2, on décrit les flèches à l'envers (flèches en bleu).

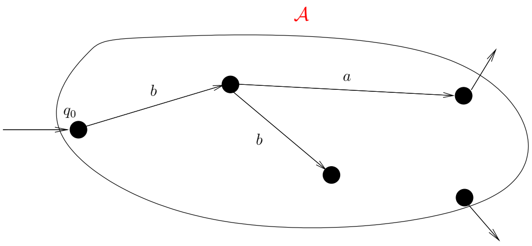

Fig. 1 – Automate reconnaissant L

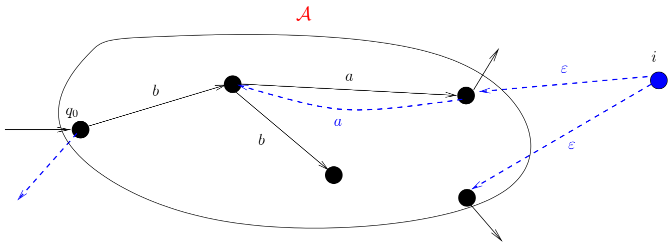

Fig. 2 – L, à l'envers

On voit immédiatement que pour avoir un unique état initial, il faut créer un nouvel état, avec une variable fraiche, disons i, qui pointe par epsilon-transition vers les états finaux de $\mathcal{A}$. Lorsqu'on lit une lettre (disons a) "à l'envers" à partir d'un certain état q de $\mathcal{A}$,alors on peut obtenir plusieurs états q' tels que $\delta(q', a) = q$. Mais ce n'en est pas un problème, au pire on obtient un automate non déterministe. donc, on obtient l'automate $\mathcal{A}' = (Q, \Sigma, i, \Delta, \{q_0\})$ avec :

$$\Delta = \big\{(q, \alpha, q') \mid \delta(q', \alpha) = q\big\} \cup \{(i, \varepsilon, q_f) \mid q_f \in F\}$$

Il reste maintenant à montrer que $\mathcal{A}'$ reconnaît effectivement $\widetilde{L}$. On montre par récurrence sur la taille de w qu'il existe un chemin étiqueté par w dans $\mathcal{A}$ ssi il existe un chemin étiqueté par $\widetilde{w}$ dans $\mathcal{A}'$.

2 Les langages racine(L) et 1/2L

Soit $\frac{1}{2}L = \{u \mid \exists v, |v| = |u|, u.v \in L\}$ et $racine(L) = \{u \mid u^2 \in L\}$.

Pour L rationnel, on va montrer que ces deux langages sont rationnels. Pour l'intuition, on va se référer à $racine(L)$. L'idée est de lire sur l'automate initial $(Q, \Sigma, q_0, \delta, F)$ le mot u à partir de q_0, et de tous les états q. On va donc contruire $n = |Q|$ automates $\mathcal{A}'_q$ de la façon suivante :

- Les états seront des 2-vecteurs d'éléments de Q, c'est-à-dire des $\begin{pmatrix} q_i \\ q_j \end{pmatrix}$.
- L'état initial de l'automate $\mathcal{A}'_q$ est l'état $\begin{pmatrix} q_0 \\ q \end{pmatrix}$.
- Les états terminaux de de l'automate $\mathcal{A}'_q$ sont les états $\begin{pmatrix} q \\ q_f \end{pmatrix}$ avec $q_f \in F$.

– Une transition étiquetée par α existe entre les deux vecteurs $\begin{pmatrix} q_i \\ q_j \end{pmatrix}$ et $\begin{pmatrix} q'_i \\ q'_j \end{pmatrix}$ ssi $\delta(q_i, \alpha) = q'_i$ et $\delta(q_j, \alpha) = q'_j$.

Finalement, si on prend $\mathcal{A}' = \bigcup_{q \in Q} \mathcal{A}'_q$, alors on obtient un automate reconnaissant $racine(L)$. Pourquoi ? Parce que si un mot est reconnu par $\mathcal{A}'$, alors il existe un q tel que l'automate $\mathcal{A}'_q$ reconnaît u (construction de l'union). Par construction, $\delta^*(q_0, u) = q$ et $\delta(q, u) \in F$, ce qui signifie que $u.u \in L$, c'est ce qu'on voulait. La réciproque se fait de façon similaire.

Pour $1/2L$, on fait de même sauf pour les fonctions de transition : une transition étiquetée par α existe entre les deux vecteurs $\begin{pmatrix} q_i \\ q_j \end{pmatrix}$ et $\begin{pmatrix} q'_i \\ q'_j \end{pmatrix}$ ssi $\delta(q_i, \alpha) = q'_i$ et il existe $\beta \in \Sigma$, tel que $\delta(q_j, \beta) = q'_j$. (pour garder la même longueur).