

DES AUTOMATES AUX EXPRESSIONS RATIONNELLES

À la fin de ce chapitre, je sais :

- expliquer ce qu'est un automate généralisé
- fusionner des transitions multiples
- éliminer des états
- passer d'un automate à une expression régulière

Ce chapitre permet d'apporter une démonstration à la réciproque du théorème de Kleene, à savoir qu'un langage reconnaissable est un langage régulier. On s'appuie pour cela sur l'algorithme d'élimination des états qui nécessite le concept d'automate généralisé.

A Automate généralisé

■ **Définition 1 — Automate généralisé.** Soit Σ un alphabet et \mathcal{E}_R l'ensemble des expressions régulières sur un Σ . Un automate généralisé est un automate tel que $\mathcal{A} = (Q, \mathcal{E}_R, Q_i, \Delta, F)$, c'est-à-dire un automate dont les étiquettes des arcs sont des expressions régulières.

B D'un automate généralisé à une expression régulière

Soit un automate \mathcal{A} . On souhaite trouver une expression régulière e telle que $\mathcal{L}(\mathcal{A}) = \mathcal{L}_{ER}(e)$, c'est-à-dire le langage reconnu par l'automate \mathcal{A} est le même que celui dénoté par e .

La construction de Brzozowski et McCluskey est intuitive et facile à programmer. Il s'agit

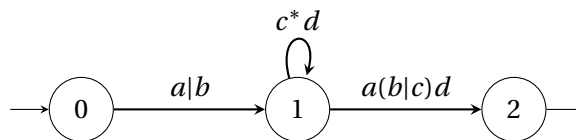


FIGURE 1 – Exemple d'automate généralisé sur l'ensemble des expressions régulières sur $\Sigma = \{a, b, c, d\}$

d'éliminer un à un les états de l'automate généralisé associé à \mathcal{A} . À la fin de la procédure, il ne reste plus que deux états reliés par un seul arc étiqueté par une seule expression régulière e . Le langage dénoté par cette dernière est le langage reconnu par l'automate.

M **Méthode 1 — Construire l'expression régulière équivalent à un automate** Deux grandes étapes sont nécessaires pour construire l'expression régulière équivalent à un automate. **Pour chaque état q à éliminer**, c'est-à-dire les états autres que l'état initial ou l'état final,

1. fusionner les expressions régulières des transitions au départ de q_s et à destination du même état q_n comme illustré sur la figure 2. Formellement, si on a les transitions (q_s, e_1, q_n) et (q_s, e_2, q_n) , alors on fusionne les deux expressions en faisant leur somme : $(q_s, e_1|e_2, q_n)$. On ne conserve ainsi qu'une seule expression par destination au départ de q_s .
2. éliminer l'état q_s en mettant à jour les transitions au départ des états précédents comme l'illustre la figure 3. Considérons chaque transition de type (q_p, e_1, q_s) et (q_s, e_2, q_n) , c'est-à-dire les transitions pour lesquelles q_s intervient. Si on souhaite éliminer q_s , il faut considérer à chaque fois deux cas :
 - (a) une transition boucle (q_s, e_b, q_s) existe : alors il est nécessaire d'ajouter la transition $(q_p, e_1 e_b^* e_2, q_n)$,
 - (b) dans le cas contraire, il suffit d'ajouter la transition $(q_p, e_1 e_2, q_n)$.

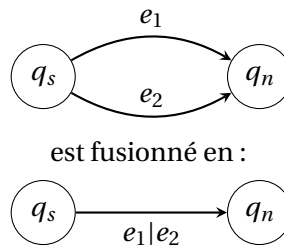
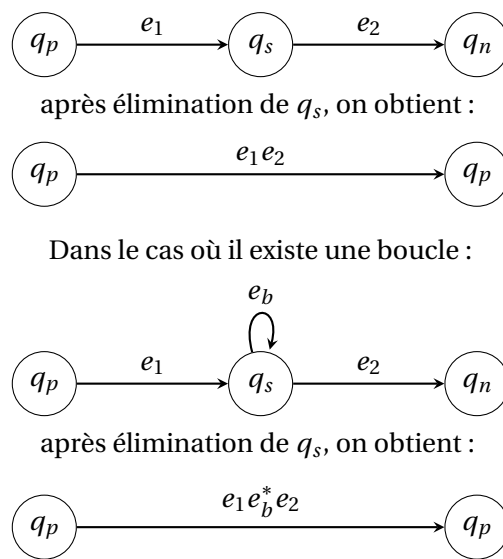


FIGURE 2 – Fusion de deux arcs au départ d'un état q_s et à destination du même état q_n

FIGURE 3 – Élimination d'un état q_s .