Bases formelles du TAL
DM sur les ϵ-transitions

Pierre-Léo Bégay

À me rendre le 6 mars 2020

1 ϵ-transitions

1.1 Définitions

On donne parfois la définition suivante d'un AFND :

$$A = \langle Q, \Sigma, q_0, F, \delta \rangle$$

Q ensemble fini d'états

Σ l'alphabet (ensemble de lettres)

q_0 l'état initial

$F \subseteq Q$, les états terminaux

δ fonction de $(Q \times (\Sigma \cup \{\epsilon\}))$ dans 2^Q

Par rapport à la définition du cours, on revient à un seul état initial et qu'on permet d'étiqueter des transitions par ϵ. Ces transitions, appelées ϵ-transitions, sont *gratuites*, par contraste avec les transitions normales qui *consomment* une lettre chaque fois qu'on les emprunte. La notion d'acceptation est sinon la même que pour les AFND qu'on a déjà vus.

1.2 Exemples

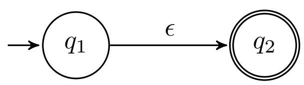

Figure 1: Automate A_1

Dans l'automate A_1, aucune transition par lettre n'est possible, ce qui empêche d'accepter tout mot autre que le mot vide. Ce dernier est cependant reconnu car on peut emprunter *gratuitement* l'unique transition et atterrir dans un état terminal.

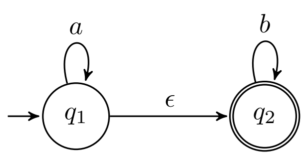

Figure 2: Automate A_2

L'automate A_2 reconnait quant à lui le langage a^*b^*. En effet, on peut boucler avec des a sur q_1 puis, une fois qu'on a fini, on passe gratuitement à q_2 (sans consommer de a ou de b) où on peut boucler avec des b jusqu'à avoir fini le mot.

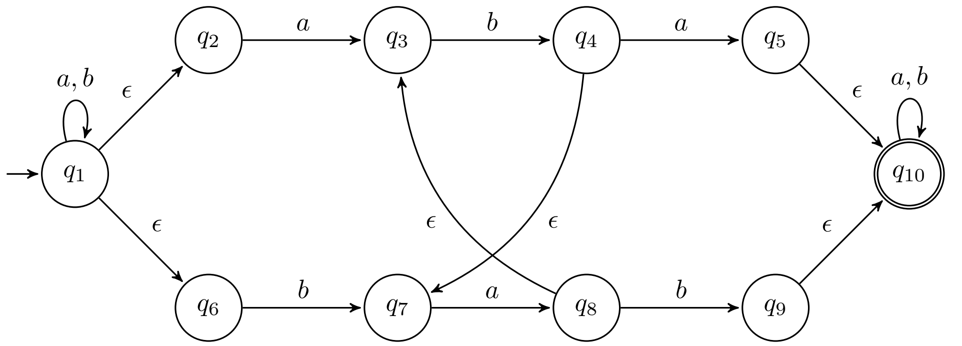

Figure 3: Automate A_3

Enfin, l'automate A_3, proche d'un qu'on a vu en cours, reconnaît quant à lui le langage $\Sigma^* aba \Sigma^* + \Sigma^* bab \Sigma^*$ (les deux ϵ-transitions en croix ne permettent pas d'accepter plus de mots).

2 Lecture d'automates avec ϵ-transitions

Décrivez les langages reconnus par les automates A_4, A_5 et A_6 à l'aide d'une expression rationelle. Essayez de justifier, au moins informellement, votre réponse.

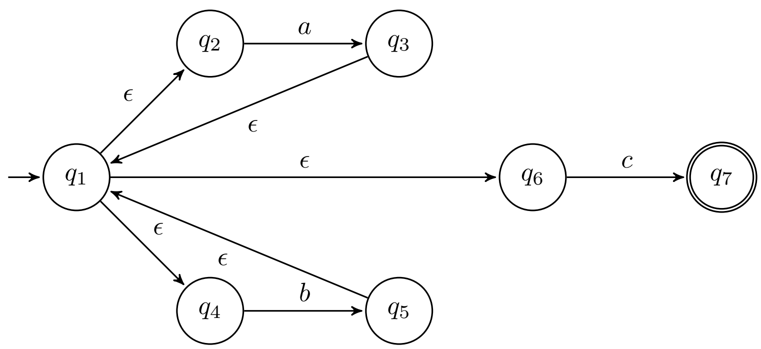

Figure 4: Automate A_4

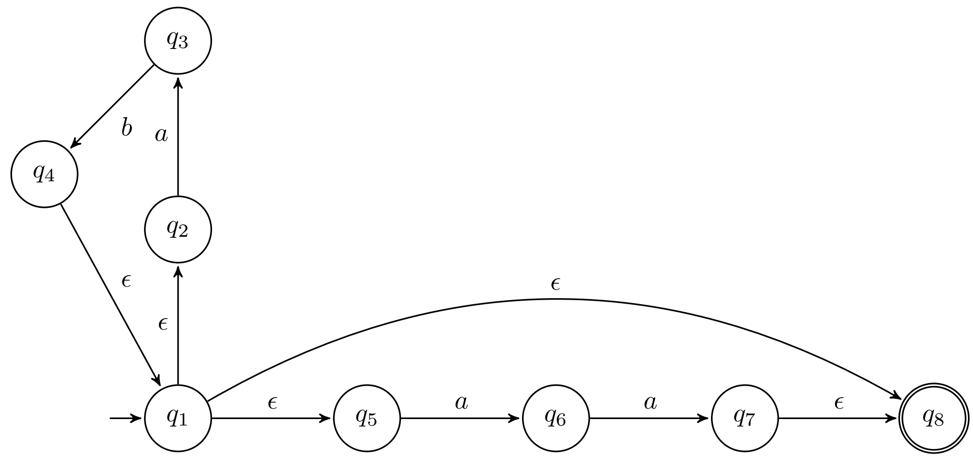

Figure 5: Automate A_5

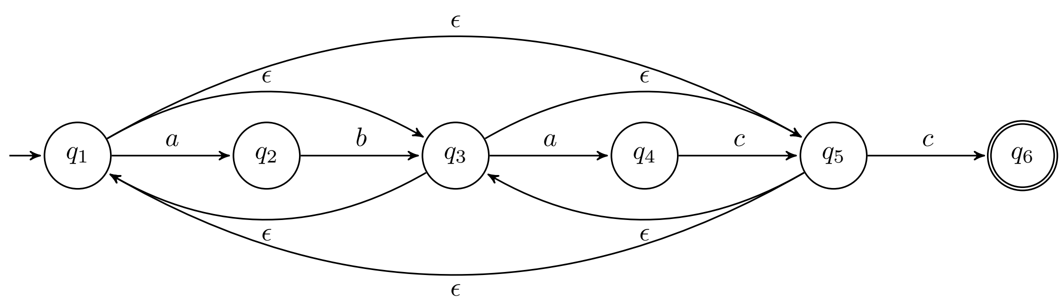

Figure 6: Automate A_6

3 Elimination d'ϵ-transitions

On propose une méthode pour éliminer les ϵ-transitions s'appuyant sur la fonction ϵ^+ de type $Q \to 2^Q$ définie de la façon suivante :

1. Si $q_j \in \delta(q_i, \epsilon)$, alors $q_j \in \epsilon^+(q_i)$
2. Si $q_j \in \epsilon^+(q_i)$ et $q_k \in \delta(q_j, \epsilon)$, alors $q_k \in \epsilon^+(q_i)$

Figure 7: Définition de ϵ^+

A partir d'un automate non-déterministe avec ϵ-transitions $\langle Q, \Sigma, q_0, F, \delta \rangle$, on génère un automate non-déterministe équivalent sans ϵ-transitions $\langle Q, \Sigma, q_0, F', \delta' \rangle$ avec l'algorithme suivant :

```
F' := F
for all q_i ∈ Q do
    for all c ∈ Σ do δ'(q_i, c) := δ(q_i, c)
    end for
end for
for all q_i ∈ Q tels que ε⁺(q_i) ≠ ∅ do
    for all q_j ∈ ε⁺(q_i) do
        for all c ∈ Σ et q_r ∈ Q tels que q_r ∈ δ(q_j, c) do
            δ'(q_i, c) := δ'(q_i, c) ∪ {q_r}
        end for
        if q_j ∈ F then
            F' := F' ∪ {q_i}
        end if
    end for
end for
```

Figure 8: Algorithme d'élimination des ϵ-transitions

Question 1 Pour chaque automate (A_1 à A_6), calculez la fonction ϵ^+ en vous servant de la définition en figure 7. Vous devriez donner l'image de chaque état, par exemple $\epsilon^+(q_1) = \{q_1, q_2\}$, $\epsilon^+(q_2) = \emptyset$ etc.

Question 2 Pour chaque automate (A_1 à A_6), appliquez l'algorithme de la figure 8. Vous pouvez dessiner le résultat. Pas besoin de détailler tous les calculs.

Question 3 Essayez d'expliquer en français la fonction ϵ^+ et l'algorithme comme si vous vouliez me convaincre qu'ils font correctement leur boulot (ce qui est le cas)[1]. Vous pouvez vous aider d'exemples, soit tirés de la question précédente, soit originaux.

4 Formalisation

Question 4 Donnez une formalisation de l'acceptation d'un mot dans le contexte des AFND avec ϵ-transtion en adaptant la définition de δ^* donnée dans le cours.

Question bonus Les AFND avec ϵ-transitions sont-ils plus ou moins expressifs[2] que ceux vus en cours, où on pouvait avoir plusieurs états initiaux et plusieurs transitions "concurrentes" ?

5 Propriétés de clôture

Question 5 Etant donnés des AFND (version ϵ) représentant deux expressions rationnelles quelconques e_1 et e_2, expliquez, en vous aidant de schémas, comment construire des automates reconnaissant les expressions $e_1 + e_2$, $e_1.e_2$ et e_1^*.

Question bonus Répondre à la question précédente en utilisant la formalisation des automates.

[1] Notez que rien n'est gratuit dans l'algorithme et que chaque *morceau* a un sens. Vous devriez donc tout mentionner.

[2] cad. permettent-ils de décrire plus ou moins de langages ?