Partie théorique - Projet d'Informatique Théorique

Nicolas Endredi - Baptiste Oruezabal - Rémi Schiano

8 mai 2014

1 Question 1

Soit A un automate et m un mot du langage reconnu par ce dernier. Alors, m peut s'écrire de la façon suivante :

$$m = ik_0q_1k_1...k_nf$$

avec K ensemble des lettres du mot de longueur n, q ensemble des états du mots, i état initial et f état final.

Or par définition, le miroir de l'automate inverse toutes les transitions ainsi que les états finaux et initiaux. On obtient alors le mot m' suivant

$$m' = fk_n....k_1q_1k_0i$$

On en déduit que

$$m' = MIR(m)$$

2 Question 2

Soit w un mot du langage de l'automate et MIR(w) son miroir.

Prenons

$$w = uv$$

et

$$MIR(w) = u'v'$$

tels que u est le langage à gauche, v le langage à droite et u' langage à droite et v' langage à gauche. De plus |u| = |u'| et |v| = |v'|.

Supposons que

$$MIR(v') \neq v$$

On a par définition,

$$MIR(MIR(w)) = w$$

Donc

$$MIR(v'u') = uv$$

Or

$$MIR(v') \neq v$$

Par conséquent,

$$MIR(v'u') \neq uv$$

Donc MIR(w) n'est pas w. Ce qui est absurde.

Conclusion : le miroir du miroir du langage à gauche est le langage à droite. Même raisonnement pour le langage à droite du miroir.

3 Question 3

On suppose

$$L_d^{DET(A)}(R) \neq \bigcup_{r \in R} L_d^A(r)$$

Soit

$$q_1 x q_2$$

une transition de l'automate de

$$L_d^A(R)$$

alors

$$\exists q_1 x q_2 \in L_d^A(R) | q_1 x q_2' \notin L_d^{DET(A)}$$

avec

$$q_2 \subseteq q_2'$$

Donc DET(A) n'est pas l'automate déterministe de A. Ce qui est absurde.

Conclusion :

$$L_d^{DET(A)}(R) = \bigcup_{r \in R} L_d^A(r)$$

4 Question 4

Par définition, un automate déterministe ne dispose que d'un seul état initial et chaque état dispose d'au plus une transition avec la même lettre. Par conséquent, à partir d'un état, on ne peut pas arriver sur deux états différents avec la même lettre.

$$\forall q, r \in Q | q \neq r, DET(A) \Rightarrow |I| = 1 \wedge L_g^A(q) \bigcap L_g^A(r) = \emptyset$$

Ce qui est vrai par définition.

Supposons que

$$DET(A) \Leftarrow |I| \neq 1 \vee L_g^A(q) \bigcap L_g^A(r) \neq \emptyset$$

Or, par définition, un automate déterministe ne peut avoir qu'un seul état initial. Donc on a

$$DET(A) \Leftarrow \forall q, r \in Q | q \neq r, L_g^A(q) \bigcap L_g^A(r) \neq \emptyset$$

Si l'intersection des langages gauche de deux états différents est non vide, alors il dispose d'au moins un mot en commun. Par conséquent, à un moment donné il y a eu deux transitions avec la même lettre partant d'un état. Ceci étant en contradiction avec la définition d'un automate déterministe.

On a donc

$$\forall q, r \in Q | q \neq r, DET(A) \Leftarrow |I| = 1 \wedge L_g^A(q) \bigcap L_g^A(r) = \emptyset$$

On en déduit que

$$\forall q, r \in Q | q \neq r, DET(A) \iff |I| = 1 \wedge L_g^A(q) \bigcap L_g^A(r) = \emptyset$$

5 Question 5

Soit A un automate déterministe dont tous les états sont accessibles. Ainsi, pour que A soit minimal, il est nécessaire et suffisant que

$$\forall q, r \in Q | q \neq r, L_d^A(q) \bigcap L_d^A(r) = \emptyset$$

En effet, si les intersection de langages ne sont pas vides, cela signifie que l'on peut obtenir au moins un mot de deux façons différentes. On pourrait donc fusionner des états et l'automate ne serait pas minimal.

6 Question 6

Soit A un automate. Lorsqu'on déterminise le miroir de A, on obtient d'après la question 4

$$DET(A) \Leftarrow \forall q, r \in Q | q \neq r, L_g^A(q) \bigcap L_g^A(r) \neq \emptyset$$

Si l'on fait le miroir de l'automate déterministe précédent, alors grâce à la question 2 on a

$$DET(A) \Leftarrow \forall q, r \in Q | q \neq r, L_d^A(q) \bigcap L_d^A(r) \neq \emptyset$$

puisque

$$L_g^A(q) = L_d^{MIR(A)}(q)$$

Lors de la dernière déterminisation, on obtient

$$DET(A) \Leftarrow \forall q, r \in Q | q \neq r, L_d^A(q) \bigcap L_d^A(r) \neq \emptyset \wedge L_g^A(q) \bigcap L_g^A(r) = \emptyset$$

On en déduit d'après la question 5 que
DET(MIR(DET(MIR(A)))) est minimal.

7 Question 7

Si l'on considère le pire des cas, la déterminisation ne change ni le nombre d'états ni le nombre de transitions et sera donc de complexité

$$O(2^n)$$

où n est le nombre d'états de l'automate.

D'autre part, l'opération miroir ne changeant pas non plus le nombre d'états (il s'agit d'une simple inversion d'ensembles ou de transitions), la complexité sera de

$$O(an^2)$$

où a est la taille de l'alphabet. La complexité de la déterminisation du miroir sera donc de

$$O(an^2 + 2^n)$$

Etant donné que nous effectuons deux fois la déterminisation du miroir au final, la complexité dans le pire des cas sera de

$$O(4^n + an^2)$$