Contrôle continu – Automates Avancés et Applications
Master 1 Informatique

28 octobre 2021

Durée : 2h00.

Documents autorisés : Deux feuilles A4 manuscrites recto-verso.

Notation : On rappelle que pour un mot $w \in \Sigma^*$, sa longueur est notée $|w|$ et si $a \in \Sigma$ alors $|w|_a$ correspond au nombre de fois que la lettre a apparaît dans le mot w.

Exercice 1 : [5 points] *Langages rationnels*

On considère l'alphabet $\Sigma = \{a, b, c\}$. Dire en justifiant la réponse si les langages suivants sont rationnels ou non.

1. $L_1 = \{w \in \Sigma^* \mid \text{tout } a \text{ est suivi par } bc \text{ dans } w\}$.
2. $L_2 = \{w = a^n b^n c^n \mid n \geq 0\} \cap L_1$.
3. $L_3 = \{a^n b^m a^n b^m \mid n \geq 1 \text{ et } m \geq 1\}$.
4. $L_4 = \{w \in \Sigma^* \mid |w|_a \text{ est paire ssi } |w|_b + |w|_c \text{ est paire}\}$.

Exercice 2 : [3 points] *De l'expression rationnelle à l'automate et vice-versa*

On considère l'alphabet $\Sigma = \{a, b, c\}$.

1. Utiliser la méthode par automate normalisé pour donner l'automate reconnaissant le langage rationnel suivant : $((a \cdot b)^* + c)^*.(a + b + c)^*$.
2. Donner l'expression rationnelle correspondant au langage reconnu par l'automate A_1 représenté à la Figure 1.

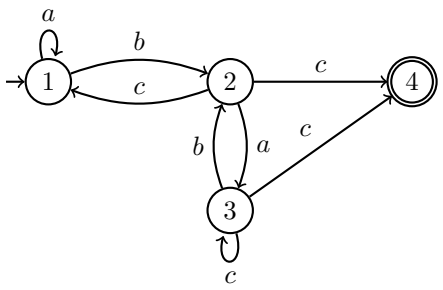

FIGURE 1 – Automate A_1

Exercice 3 : [3 points] *Grammaires algébriques*

On considère l'alphabet $\Sigma = \{a, b, c\}$.

1. Donner une grammaire algébrique G reconnaissant le langage $L_1 = \{a^n b^m c^\ell \mid n \geq 0 \text{ et } m \geq 0 \text{ et } \ell \geq 0 \text{ et } (n = m + \ell \text{ ou } m = n + \ell \text{ ou } \ell = m + n)\}$.
2. Donner une grammaire algébrique reconnaissant le langage $L_2 = \{wcw' \in \Sigma^* \mid w \in \{a, b\}^* \text{ et } |w| = |w'|\}$.

Exercice 4 : [3 points] *Normalisation de grammaires*

On considère l'alphabet $\Sigma = \{a, b, c, d\}$.

1. Réduire la grammaire suivante pour S :

$$\begin{array}{lcl} S & \rightarrow & aDb \ + \ bCa \ + \ cSd \\ A & \rightarrow & aaA \ + \ Sbb \ + \ b \\ B & \rightarrow & bBb \ + \ cCc \\ C & \rightarrow & aBa \ + \ dCd \\ D & \rightarrow & bbA \ + \ ccS + ddB \end{array}$$

2. Rendre la grammaire suivante propre :

$$\begin{array}{lcl} S & \rightarrow & aVbU \ + \ VU \ + \ aTb \\ T & \rightarrow & cTd \ + \ TaT \\ U & \rightarrow & UaU \ + \ aVb \ + \ V \\ V & \rightarrow & a + b + \epsilon \end{array}$$

Exercice 5 : [3points] *Automates à pile*

Pour chacun des langages présentés ci-dessous, donner un automate à pile les reconnaissant avec acceptation par pile vide.

1. $L_1 = \{a^n b^m \mid 0 \leq n \leq m \leq 2n\}$.
2. $L_2 = \{w \in \{a,b\}^* \mid \nexists x \in \{a,b\}^*.w = xx\}$.

Exercice 6 : [3points] *Automates à pile et langages réguliers*

On considère un nouveau mode d'acceptation pour les automates à pile : si Γ est l'alphabet de pile, ce mode d'acceptation est défini par un langage régulier $L_F \subseteq \Gamma^*$ et une configuration $(q, w) \in Q \times \Gamma^*$ (où Q est l'ensemble des états de l'automate à pile) est acceptante si, et seulement si, $w \in L_F$. On appelle ce mode d'acceptation, acceptation par langage régulier.

1. Montrer que si un langage algébrique est accepté par un automate à pile avec acceptation par état final, alors il est accepté par un automate à pile avec acceptation par langage régulier.
2. Montrer que si un langage algébrique est accepté par un automate à pile avec acceptation par langage régulier, alors il est accepté par un automate à pile avec acceptation par pile vide.