Automates finis déterministes

OPTION INFORMATIQUE - TP nº 3.9 - Olivier Reynet

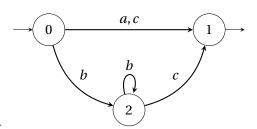
À la fin de ce chapitre, je sais :

- 🕼 définir un automate fini déterministe
- représenter un automate fini déterministe
- qualifier les états d'un automates (accessibilité)
- 🎏 compléter un AFD
- 🎏 complémenter un AFD
- faire le produit de deux AFD

A Construction d'automates simples

- A1. On considère l'alphabet $\Sigma = \{a, b, c\}$. Construire les automates suivants :
 - (a) A_0 reconnaissant les mots commençant par deux occurrences de a.
 - (b) A_1 reconnaissant le langage défini par l'expression rationnelle $a|b^*c$.
 - (c) A_2 reconnaissant les mots contenant un nombre de a égal à 1 modulo 3, sans contrainte sur les autres lettres.
 - (d) \mathcal{A}_3 reconnaissant les mots contenant un nombre pair de a et un nombre impair de b, c'est à dire $\{b, baa, aab, bbb, bababbb, aabbb, aabbb, ababaab, ...\}$. On suppose que l'alphabet est $\Sigma = \{a, b\}$.
- A2. Donner les représentations tabulaires des automates A_0 , A_1 , A_2 et A_3 .
- A3. Les automates que vous avez dessinés sont-ils complets?
- A4. Combien d'automates complets différents à n états peut-on construire? On cherchera à exprimer la réponse en fonction de n et $|\Sigma|$

B Complété et complémentaire d'automates finis déterministes



- B1. Compléter l'automate fini déterministe $\mathcal A$ suivant.
- B2. Quels sont les états co-accessibles de l'automate complété de A?
- B3. Dessiner le complémentaire de l'automate \mathcal{A} précédent.
- B4. Le complémentaire de l'automate \mathcal{A} reconnaît-il le mot vide ϵ ?

OPTION INFORMATIQUE TP no 3.9

C Modélisation d'un automate en OCaml

type get alphabet successors predecessors

D Calcul d'un mot par un automate

up_to match_word

E Algorithmes de transformation simple d'un automate

complétion : tester la complétude puis algo de complétion p106 107 complémentaire : facile p108

F États accessibles et co-accessibles d'un automate

p95 falcone p96 Produire l'algo d'un automate émondé p111

G Problème de la finitude d'un langage reconnaissable

p100