Automates finis

- Automate fini déterministe (AFD)
- Automates finis non déterministes AFN
- Construction d'automates
- Langage d'un AFD
- Théorème de Kleene
- Équivalence d'automates – Minimisation

Automate complet

Définition

Un automate est dit complet si à partir de tout état, toute lecture de lettre provoque une transition.

Définition

Un état q est un puits si, $\forall \ell \in \Sigma$, $T(q, \ell) = q$

Dit autrement, quand on y est on y reste...

Définition

On appelle piège (ou poubelle) un puits refusant. Toute lecture qui tombe dans le piège ne peut plus être acceptante.

Proposition

Tout automate fini peut être complété (remplacé par un automate complet équivalent) en ajoutant un état piège vers lequel on redirige toutes les transitions manquantes.

(Voir l'animation compléterUnAFD)

Automate émondé

Rappel : émonder un arbre, c'est le débarrasser de ses branches mortes... Dans le cas d'un automate, cela consiste à le débarrasser de tous les états ne servant à accepter aucun mot.

Définition

- Un état q est dit accessible s'il existe un mot dont la lecture fait passer de l'état initial à l'état q.
- Un état q est dit co-accessible s'il existe un mot dont la lecture fait passer de l'état q à un état acceptant.

Il est clair que si un état n'est pas accessible ET co-accessible, aucune lecture acceptante d'un mot ne passe par cet état. Il est donc inutile à l'automate et doit être émondé.

Définition

Un automate est émondé ssi tous ses états sont accessibles ET co-accessibles.

Calcul pratique de l'émondé d'un automate

1. Un parcours en profondeur à partir de l'état initial permet de déterminer l'ensemble des états accessibles.

2. Des parcours en profondeur du graphe inverse à partir des états acceptants permettent de calculer l'ensemble des états co-accessibles.

3. Par intersection on détermine l'ensemble des états à conserver. On élimine les états superflus et les arcs adjacents.

(Voir l'animation EmonderUnAFD)

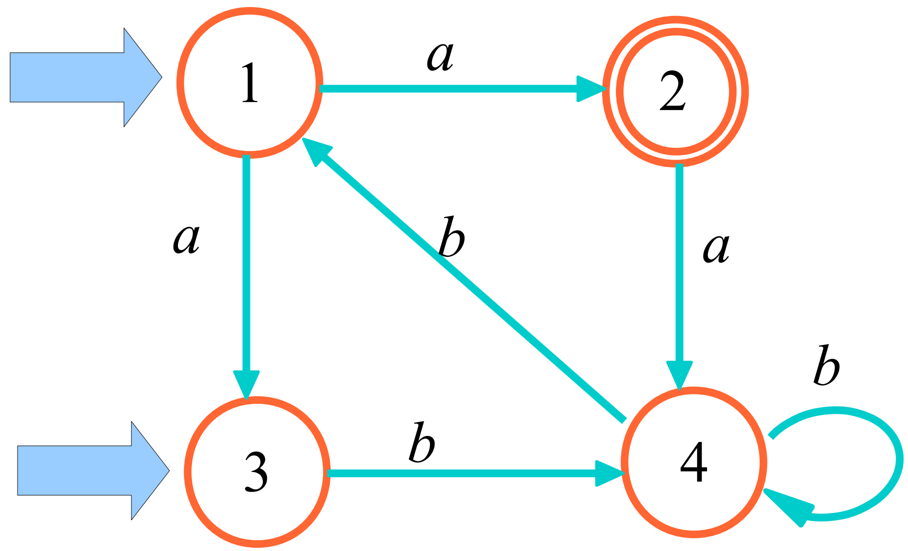
Remarque : Cet automate n'est pas encore minimal comme nous le verrons au dernier paragraphe mais est débarrassé des états « grossièrement inutiles »

TP : Comment pourrions nous résister à l'envie de programmer le calcul de l'automate complet et de l'automate émondé associé à un automate donné.

Automates finis non déterministes AFN

- Le but est d'assouplir la définition d'un automate de manière à en faciliter la construction.
- Dans un premier temps on va autoriser plusieurs états initiaux et plusieurs transitions possibles à partir d'un même état.

Exemple :



Définition

Un <u>automate fini non déterministe</u> (AFN) est un quintuplet :

$$\mathcal{A} = (Q, \Sigma, T, I, A)$$

où deux points sont modifiés par rapport à un AFD

- I est une partie de Q appelée ensemble des états initiaux
- la fonction de transition est maintenant une application

$$T : Q.\Sigma \rightarrow \mathcal{P}(Q)$$

qui à un état et une lettre associe un ensemble d'états vers lesquels une transition est possible.

- Cette extension de la définition va donner beaucoup de souplesse dans la construction des automates.

- Nous montrerons que cela ne changera pas théoriquement l'ensemble des langages automatiques.

- La différence fondamentale (qui justifie l'appellation non-déterministe) est que, pouvant partir de différents états initiaux et pouvant lire différemment les lettres du mot, il y a différentes façons de lire un mot, pouvant conduire à des états acceptants ou non...

- La convention qui est choisie est d'accepter un mot si au moins une façon de le lire conduit à un état acceptant.

Lecture d'un mot

Soit $\mathcal{A} = (Q, \Sigma, T, I, A)$ un automate non déterministe, on définit :

UNE lecture du mot $w=\ell_1..\ell_n$ par $\mathcal{A}$ est UNE suite d'états q_0, q_2, q_n telle que

- $q_0 \in I$
- $q_0 \xrightarrow{\ell_1} q_1 \xrightarrow{\ell_2} q_2 \xrightarrow{\ell_3} \qquad \xrightarrow{\ell_n} q_n$

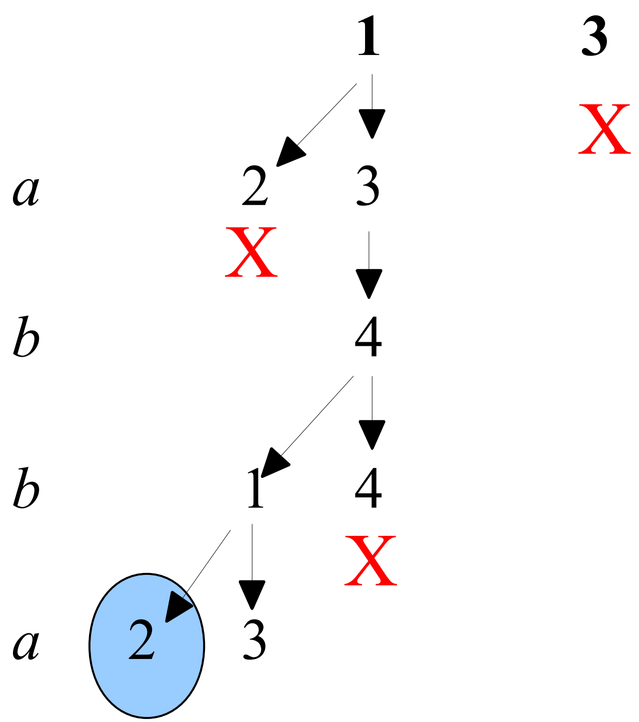
- La lecture d'une lettre pouvant conduire à différents états, la lecture d'un mot à partir d'un état initial donné conduit à **un arbre de lecture.**

- La lecture d'un mot conduit donc à une forêt de lecture et d'après la définition donnée, une lecture de ce mot est un chemin dans cette forêt, allant d'une des racines à une feuille du même arbre.

Exemple : L'AFN donné en exemple, a pour table de transition :

T	*a*	*b*
1	{2,3}	
2	4	
3		4
4		{1,4}

Les lectures possibles du mot *abba* par cet automate sont données par la forêt suivante :



Il suffit qu'une des lectures possibles soit acceptante pour que le mot « abba » soit accepté, comme le confirme la définition suivante.

Langage d'un automate ND

Définition

On dit que le mot w est accepté par l'automate non déterministe $\mathcal{A} = (Q, \Sigma, T, I, A)$ s'il existe une lecture du mot w conduisant à un état acceptant $q \in A$.

Définition

On appelle langage de l'automate $\mathcal{A} = (Q, \Sigma, T, I, A)$ l'ensemble des mots acceptés par cet automate.

On le note $L(\mathcal{A})$.

On dit encore que $\mathcal{A}$ reconnaît ou accepte ce langage.

Définition

Deux automates sont dits équivalents si ils reconnaissent le même langage.