Informatique théorique III
(Automates, langages & calculabilité)

Formulaire du Cours 2004-2005
Partie 3
EPFL – I&C

Uwe Nestmann
Sébastien Briais
Daniel C. Bünzli

14 février 2005

Bibliographie

[HMU03] John Hopcroft, Rajeev Motwani, and Jeffrey Ullman. *Introduction to Automata Theory, Languages and Computation*. Pearson Education International, 2003. ISBN 0321210298.

Table des matières

$Id: notes-3.tex,v 1.15 2004/11/17 09:49:58 dbuenzli Exp $

semaine 3
lundi

Voir [HMU03], §2.3–§2.5.

2.2 Automates finis non déterministes

2.2.1 Définition (AFN)
Un automate fini non déterministe (AFN) est un quintuplet

$$M = (Q, \Sigma, \Delta, s, F)$$

où
- Q est un ensemble fini d'*états*,
- Σ est un ensemble fini de symboles, *l'alphabet (d'entrée)*,
- $\Delta : Q \times \Sigma \to \mathcal{P}(Q)$ est la *fonction (totale) de transition*,
- $s \in Q$ est l'*état initial* (ou de *départ*),
- $F \subseteq Q$ est l'ensemble des *états accepteurs* (ou *finaux*).

2.2.2 Définition (Fonction de transition sur les mots) Étant donné un AFN $(Q, \Sigma, \Delta, s, F)$ on étend la fonction de transition Δ en une fonction de transition $\widehat{\Delta} : Q \times \Sigma^* \to \mathcal{P}(Q)$ sur les mots définie par :

$$\widehat{\Delta}(p, \epsilon) \triangleq \{\, p \,\} \qquad \widehat{\Delta}(q, wa) \triangleq \bigcup_{p \in \widehat{\Delta}(q,w)} \Delta(p, a)$$

2.2.3 Définition (Langage accepté par un AFN)
Soit $M = (Q, \Sigma, \Delta, s, F)$ un AFN.

1. M *accepte* le mot $w \in \Sigma^*$ si $\widehat{\Delta}(s, w) \cap F \neq \emptyset$.
   Dans le cas contraire on dit que M *rejette* w.
2. Le *langage accepté par* M est défini par :

$$\mathrm{L}(M) \triangleq \{\, w \in \Sigma^* \mid \widehat{\Delta}(s, w) \cap F \neq \emptyset \,\}$$

Tout automate fini non déterministe peut être converti en un automate fini déterministe en utilisant la *construction des sous-ensembles*.

2.2.4 Définition (Conversion d'un AFN en AFD) On définit la fonction D de *déterminisation* des AFN de la manière suivante :

$$\begin{aligned} \mathrm{D} : \quad & \text{AFN} \to \text{AFD} \\ & (Q, \Sigma, \Delta, s, F) \mapsto (Q_{\mathrm{D}}, \Sigma, \delta, s_{\mathrm{D}}, F_{\mathrm{D}}) \end{aligned}$$

avec

$$\begin{aligned} Q_{\mathrm{D}} &\triangleq \mathcal{P}(Q) \\ \delta(P, a) &\triangleq \textstyle\bigcup_{q \in P} \Delta(q, a) \\ s_{\mathrm{D}} &\triangleq \{\, s \,\} \\ F_{\mathrm{D}} &\triangleq \{\, P \subseteq Q \mid P \cap F \neq \emptyset \,\} \end{aligned}$$

2.2.5 Lemme Soit $M = (Q, \Sigma, \Delta, s, F)$ un AFN et $\mathrm{D}(M) = (\ldots, \delta, \ldots)$ son AFD associé. Alors, on a $\forall w \in \Sigma^* : \widehat{\delta}(\{s\}, w) = \widehat{\Delta}(s, w)$.

2.2.6 Théorème Soit M un AFN, on a $\mathrm{L}(M) = \mathrm{L}(\mathrm{D}(M)))$.

En pratique la fonction de déterminisation des automates D génère un AFD avec un grand nombre d'états inaccesibles depuis l'état de départ et donc inutiles. C'est pourquoi on utilise plutôt l'algorithme suivant qui, étant donné un AFN, construit le tableau d'un AFD en ne générant que les états accessibles.

2.2.7 Définition (Algorithme de déterminisation)
Soit $M = (Q, \Sigma, \Delta, s, F)$ un AFN. On calcule le tableau de la fonction de transition δ d'un AFD M' tel que $\mathrm{L}(M) = \mathrm{L}(M')$ de la façon suivante.

1. Dans le tableau, insérer une ligne pour l'état inital $\{s\}$. Pour tout $a \in \Sigma$, calculer les valeurs du tableau $\delta(\{s\}, a) = \Delta(s, a)$. Chacune de ces valeurs définit un état P pour lequel on rajoute une ligne au tableau (s'il n'est pas déjà défini).
2. Pour chacun des états P rajoutés, pour tout $a \in \Sigma$, il faut calculer les valeurs du tableau $\delta(P, a) = \bigcup_{q \in P} \Delta(q, a)$. Chacune de ces valeurs définit un état P' pour lequel on rajoute une ligne au tableau (s'il n'est pas déjà défini) dont on calcule les valeurs $\delta(P', a)$ pour tout $a \in \Sigma$ comme indiqué ci-dessus. On répète ce processus jusqu'à ce que le tableau se stabilise, c.-à-d., jusqu'à ce qu'il n'y ait plus de nouvel état (ligne) à insérer dans le tableau.
3. Marquer l'état $\{s\}$ par un S. Et les états P tels que $P \cap F \neq \emptyset$ par un F.

2.3 Transitions instantanées

semaine 3
mercredi

2.3.1 Notation (Transition instantanée) On utilise le symbole distingué $\mathbf{e}$ pour représenter une transition instantanée. Une *transition instantanée* (ou *transition epsilon*) est une transition d'automate qui ne consomme pas de symbole.

2.3.2 Définition (AFN$_\mathbf{e}$) Un *automate fini non déterministe avec transitions instantanées* (AFN$_\mathbf{e}$) est un quintuplet

$$M = (Q, \Sigma, \Delta, s, F)$$

où

- Q est un ensemble fini d'*états*,
- Σ est un ensemble fini de symboles, *l'alphabet (d'entrée)*,
- $\Delta : Q \times (\Sigma \cup \{\mathbf{e}\}) \to \mathcal{P}(Q)$ est la *fonction (totale) de transition*,
- $s \in Q$ est l'*état initial* (ou de *départ*),
- $F \subseteq Q$ est l'ensemble des *états accepteurs* (ou *finaux*).

2.3.3 Définition (e-fermeture) Soit un AFN$_\mathbf{e}$ $(Q, \Sigma, \Delta, s, F)$.
On définit l'**e**-*fermeture* $\mathrm{C}_\mathbf{e}(q)$ d'un état $q \in Q$, comme étant le plus petit ensemble satisfaisant les règles :

$$\frac{}{q \in \mathrm{C}_\mathbf{e}(q)} \qquad\qquad \frac{p \in \mathrm{C}_\mathbf{e}(q) \qquad p' \in \Delta(p, \mathbf{e})}{p' \in \mathrm{C}_\mathbf{e}(q)}$$

L' **e**-fermeture d'un ensemble d'états est définie par :

$$\mathrm{C}_\mathbf{e}(P) = \bigcup_{q \in P} \mathrm{C}_\mathbf{e}(q)$$

Noter que $\mathrm{C}_\mathbf{e}(q) \subseteq Q$ et $\mathrm{C}_\mathbf{e}(P) \subseteq Q$. □

Intuitivement, l'**e**-fermeture d'un état q donne l'ensemble de tous les états atteignables par zéro, une, ou plusieurs **e**-transitions à partir de q.

2.3.4 Définition (Fonction de transition sur les mots) Étant donné un AFN$_\mathbf{e}$ $M = (Q, \Sigma, \Delta, s, F)$ on étend la fonction de transition Δ en une fonction de transition $\widehat{\Delta}_\mathbf{e} : Q \times \Sigma^* \to \mathcal{P}(Q)$ sur les mots définie par :

$$\widehat{\Delta}_\mathbf{e}(q, \epsilon) \triangleq \mathrm{C}_\mathbf{e}(q) \qquad\qquad \widehat{\Delta}_\mathbf{e}(q, xa) \triangleq \bigcup\nolimits_{p \in \widehat{\Delta}_\mathbf{e}(q,x)} \mathrm{C}_\mathbf{e}\left(\Delta(p, a)\right)$$

2.3.5 Définition (Langage accepté par un AFN$_\mathbf{e}$)
Soit $M = (Q, \Sigma, \Delta, s, F)$ un AFN$_\mathbf{e}$.

1. L'automate M *accepte* le mot $w \in \Sigma^*$ si $\widehat{\Delta}_\mathbf{e}(s, w) \cap F \neq \emptyset$. Dans le cas contraire on dit que M *rejette* w.
2. Le *langage accepté par* M est défini par :

$$\mathrm{L}_\mathbf{e}(M) \triangleq \{\, w \in \Sigma^* \mid \widehat{\Delta}_\mathbf{e}(s, w) \cap F \neq \emptyset \,\}$$

2.3.8 Définition (Conversion de AFN$_\mathbf{e}$ en AFD) La fonction $\mathrm{D}_\mathbf{e}$ d'*élimination des* **e**-*transitions et de déterminisation* est définie par :

$$\begin{aligned} \mathrm{D}_\mathbf{e} : \qquad & \mathrm{AFN}_\mathbf{e} \to \mathrm{AFD} \\ & (Q, \Sigma, \Delta, s, F) \mapsto (Q_{\mathrm{D}_\mathbf{e}}, \Sigma, \delta, s_{\mathrm{D}_\mathbf{e}}, F_{\mathrm{D}_\mathbf{e}}) \end{aligned}$$

avec :

$$\begin{aligned} Q_{\mathrm{D}_\mathbf{e}} &\triangleq \mathcal{P}(Q) \\ \delta(P, a) &\triangleq \bigcup\nolimits_{q \in P} \mathrm{C}_\mathbf{e}(\Delta(q, a)) \\ s_{\mathrm{D}_\mathbf{e}} &\triangleq \mathrm{C}_\mathbf{e}(s) \\ F_{\mathrm{D}_\mathbf{e}} &\triangleq \{\, P \subseteq Q \mid P \cap F \neq \emptyset \,\} \end{aligned}$$

On peut raffiner cette définition par $Q_{\mathrm{D}_\mathbf{e}} \triangleq \{\, P \in \mathcal{P}(Q) \mid P = \mathrm{C}_\mathbf{e}(P) \,\}$ et une adaptation correspondante de $F_{\mathrm{D}_\mathbf{e}}$.

2.3.9 Théorème Soit M un AFN$_\mathbf{e}$, on a $\mathrm{L}_\mathbf{e}(M) = \mathrm{L}(\mathrm{D}_\mathbf{e}(M))$.