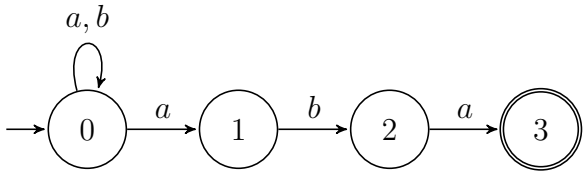
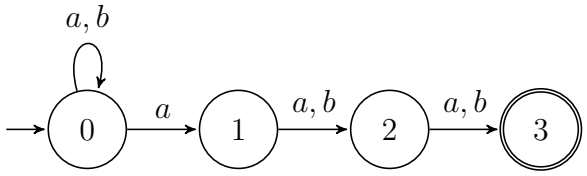
TD1 19 janvier 2017

Langages Formels

Exercice 1 (Déterminisation)

1. Pour chacun des automates suivants, déterminer le langage reconnu et donner un automate déterministe équivalent :





2. Si $\Sigma = \{a, b\}$, montrer que le langage $\Sigma^* a \Sigma^{n-1}$ ne peut être reconnu par un automate de moins de 2^n états.
3. Conclure que tout automate non-déterministe à n états admet un automate déterministe équivalent à au plus 2^n états, et que cette borne est précise.

Exercice 2 (Lemme de l'étoile)
Des conditions nécessaires pour qu'un langage soit reconnaissable sont connues sous le nom de « lemme de l'étoile » ou lemmes d'itération :

Lemme 1 (Lemme de l'étoile)**.** *Si L est un langage reconnaissable de Σ^*, alors il existe un entier n tel que*

1. *pour tout mot w de L avec $|w| \geq n$, il existe une factorisation $w = u_1 u_2 u_3$ avec u_2 non vide, telle que $u_1 u_2^* u_3 \subseteq L$,*

2. *pour tout mot w de L et pour toute factorisation de $w = v_1v_2v_3$ avec $|v_2| \geq n$, il existe une factorisation $v_2 = u_1u_2u_3$ avec u_2 non vide, telle que $v_1u_1u_2^*u_3v_3 \subseteq L$,*
3. *pour tout mot w de L et pour toute factorisation $w = u_0u_1 \cdots u_nu_{n+1}$ avec u_i non vide pour i de 1 à n, il existe deux positions $0 \leq j < k \leq n$ telles que $u_0u_1 \cdots u_j(u_{j+1} \cdots u_k)^*u_{k+1} \cdots u_nu_{n+1} \subseteq L$.*

1. Démontrez le lemme de l'étoile.
2. Montrez que $\{a^nb^n \mid n \geq 0\}$ n'est pas reconnaissable sur $\{a, b\}^*$.
3. Soit L un langage quelconque sur Σ et $\#$ un symbole qui n'est pas dans Σ.
   (a) Montrer que le langage $L_\# = (\#^+L) \cup \Sigma^*$ satisfait les conditions de la première version du lemme de l'étoile.
   (b) Montrer à l'aide de propriétés de clôture que si L n'est pas reconnaissable, alors $L_\#$ n'est pas reconnaissable.
   (c) Montrer que si L ne satisfait pas les conditions de la première version du lemme de l'étoile, alors $L_\#$ ne satisfait pas les conditions de la seconde version du lemme.
4. Soit L un langage quelconque sur Σ et \$ un symbole qui n'est pas dans Σ. On définit

$$L_1 = \{\$^+a_1\$^+a_2\$^+ \ldots \$^+a_n\$^+ \mid n \geq 0, \forall i\ a_i \in \Sigma, \text{ et } w = a_1 \ldots a_n \in L\}$$
$$L_2 = \{\$^+v_1\$^+v_2\$^+ \ldots \$^+v_n\$^+ \mid n \geq 0, \forall i\ v_i \in \Sigma^* \text{ et } \exists j\ |v_j| > 1\}$$
$$L_\$ = L_1 \cup L_2\ .$$

   (a) Montrer que $L_\$$ satisfait les conditions de la seconde version du lemme de l'étoile.
   (b) Montrer à l'aide de propriétés de clôture que si L n'est pas reconnaissable, alors $L_\$$ non plus.
   (c) Montrer que si L ne satisfait pas les conditions de la troisième version du lemme de l'étoile, alors $L_\$$ ne les vérifie pas non plus.

Exercice 3 (Construction d'automates, ou pas)
Pour chacun des langages suivants sur $\Sigma = \{0, 1\}$, dire s'il est régulier (reconnaissable par un automate fini) ou pas. Justifier en donnant un automate ou un argument.

1. $\{\ u\ :\ |u|_0 = |u|_1\ \}$ où $|u|_a$ est le nombre d'occurrences de a dans u ;
2. $\{\ 0^p 1^q\ :\ p \geq q\ \}$;
3. $\{\ 0^p 1^q\ :\ p \geq q,\ q \leq 1337\ \}$;
4. $\{\ u\ :\ \overline{u}^2 = 0 \mod 3\}$ où $\overline{u}^2$ est l'interprétation de u en binaire, avec le bit de poids fort à droite.

Exercice 4 (Dérivées partielles, automate d'Antimirov)
La dérivée partielle est aux expressions rationnelles ce que le résiduel est aux langages : on va définir une opération $\partial_a(E)$ qui va correspondre (modulo interprétation) à $a^{-1}\mathcal{L}(E)$.

Formellement, pour une expression E et une lettre a, on définit l'*ensemble d'expressions* $\partial_a(E)$ comme suit, où l'opération de concaténation est naturellement étendue sur les ensembles d'expressions :

$$\begin{array}{rcl}
\partial_a(\underline{\emptyset}) & = & \partial_a(\underline{\varepsilon}) \ = \ \emptyset \\
\partial_a(\underline{b}) & = & \left\{ \begin{array}{ll} \emptyset & \text{si } a \neq b \\ \{\underline{\varepsilon}\} & \text{sinon} \end{array} \right. \\
\partial_a(E + E') & = & \partial_a(E) \cup \partial_a(E') \\
\partial_a(E^*) & = & \partial_a(E) \cdot \{E^*\} \\
\partial_a(E \cdot E') & = & \left\{ \begin{array}{ll} \partial_a(E) \cdot \{E'\} & \text{si } \varepsilon \notin E \\ (\partial_a(E) \cdot \{E'\}) \cup \partial_a(E') & \text{sinon} \end{array} \right.
\end{array}$$

On définit ensuite $\partial_w(E)$ pour un mot w quelconque en posant $\partial_\varepsilon(E) = \{E\}$ et $\partial_{wa}(E) = \partial_a(\partial_w(E))$, où $\partial_w(S) = \bigcup_{E \in S} \partial_w(E)$ quand S est un ensemble d'expressions.

1. Calculer les dérivées partielles de $(ab + b)^* ba$ par a et b.
2. Montrer que $\mathcal{L}(\partial_w(E)) = w^{-1}\mathcal{L}(E)$.
3. Utiliser cette construction pour traduire une expression rationnelle en un automate acceptant le même langage, en ne s'inquiétant pas de la finitude de cet automate pour l'instant.
4. On définit l'ensemble des suffixes propres d'un mot :

$$\mathrm{Suf}(w) = \{\ v \in \Sigma^+\ :\ \exists u, w = uv\ \}$$

Montrer les résultats suivants pour $w \in \Sigma^+$:

$$\begin{aligned}
\partial_w(E + E') &= \partial_w(E) \cup \partial_w(E') \\
\partial_w(E \cdot E') &\subseteq (\partial_w(E) \cdot E') \cup \bigcup_{v \in \mathrm{Suf}(w)} \partial_v(E') \\
\partial_w(E^*) &\subseteq \bigcup_{v \in \mathrm{Suf}(w)} \partial_v(E) \cdot E^*
\end{aligned}$$

5. Soit $\|E\|$ le nombre d'occurrences de lettres de Σ dans E. Montrer que l'ensemble des dérivées partielles différentes de E contient au plus $\|E\| + 1$ éléments.