Exemple

Construire un automate qui accepte le langage

$$\begin{aligned} A = \{x \mid x \in \Sigma^* \ \wedge & \ \ (\exists n \in \mathbb{N} : \#a(x) = 2 * n) \\ \wedge & \ \ (\exists n \in \mathbb{N} : \#b(x) = 3 * n) \ \ \} \end{aligned}$$

Solution

Un bon candidat automate et $M = (\{ 0, 1, 2, 3, 4, 5 \}, \{ a, b \}, \delta_M, 0, \{ 0 \})$ avec :

δ_M	a	b
$SF0$	3	1
1	4	2
2	5	0
3	0	4
4	1	5
5	2	3

Verification ?

Pour comprendre le fonctionnement de l'automate ci-dessus, nous allons profiter d'une redéfinition en donnant plus de structure à son ensemble d'états par

$$\mathcal{Q}_M \stackrel{\text{def}}{=} \{ (i, j) \mid i \in \{ 0, 1 \} \wedge j \in \{ 0, 1, 2 \} \} \quad \text{où} \quad s_M = (0, 0) \quad \text{et} \quad F_M = \{ (0, 0) \}$$

suivant l'idée que i correspond au nombre de a (modulo 2) déjà vu en traversant un mot, et que j correspond au nombre de b (modulo 3) déjà vu de la même manière. Par conséquent, un état précédent $q \in \{ 0 \ldots, 5 \}$ correspond à un état nouveau calculable par $(q \text{ div } 3, q -_3 3)$.

Ainsi, on peut exploiter cette structure pour aussi redéfinir δ_M de « manière mathématique » et uniforme par

$$\begin{aligned} \delta_M : \mathcal{Q}_M \times \Sigma & \rightarrow \mathcal{Q}_M \\ ((i, j), a) & \mapsto (i +_2 1, j) \\ ((i, j), b) & \mapsto (i, j +_3 1) \end{aligned}$$

où $+_2$ et $+_3$ dénotent l'*addition modulo* 2 et 3. On untilisera aussi la notation $=_2$ et $=_3$ dénotant l'*égalité modulo* 2 et 3.

Il faut démontrer que $L(M) = A$.
Pour ce but, on observe que $x \in L(M)$ ssi $\widehat{\delta}_M((0, 0), x) = (0, 0)$ qui et le seul état final.
L'intuition de la redéfinition était que, dans ce cas, on a $\#a(x) =_2 0$ et $\#b(x) =_3 0$.
Ceux-ci sont, en fait, equivalent aux deux propriétés qui définissent le langage A.

Preuve.

Pour démontrer que $L(M) = A$ on va prouver que

$$\widehat{\delta}((0,0), x) = (i, j) \qquad \text{ssi} \qquad (\#a(x) =_2 i \wedge \#b(x) =_3 j) \tag{1}$$

ce qui implique la proposition de la page précédente où $i = 0 = j$.

Pour démontrer (1), on procède par induction sur la structure de x, suivant la définition de $\widehat{\delta}$.

base ϵ :

On a, par définition de $\widehat{\delta}$, que $\widehat{\delta}((0,0), \epsilon) = (0, 0)$.
On observe que $\#a(\epsilon) = 0 = \#b(\epsilon)$. CQFD.

hypothèse pour x :

On suppose que (1) est vrai pour le x choisi.

Par la définition inductive de $\widehat{\delta}$, on peut procéder par analyse des cas : il faut que l'on considère deux cas différents dans l'induction selon lequel des deux symboles de l'alphabet $\{ a, b \}$ étend le mot x.

cas $x \cdot a$:

$$\begin{array}{rcl} & \widehat{\delta}((0,0), xa) & \text{par définition de } \widehat{\delta} \\ = & \delta(\widehat{\delta}((0,0), x), a) & \text{par (1)} \\ = & \delta((i,j), a) & \text{par définition de } \delta \\ = & (i +_2 1, j) & \end{array}$$

Selon (1), on a $\#a(x) =_2 i$.
Par définition, $\#a(xa) = \#a(x) + 1$.
Par les règles de l'addition modulo, on a alors $\#a(xa) =_2 i + 1$. CQFD.

Selon (1), on a $\#b(x) =_3 j$.
Trivialement, on a $\#b(xa) = \#b(x) =_3 j$. CQFD.

cas $x \cdot b$:

$$\begin{array}{rcl} & \widehat{\delta}((0,0), xb) & \text{par définition de } \widehat{\delta} \\ = & \delta(\widehat{\delta}((0,0), x), b) & \text{par (1)} \\ = & \delta((i,j), b) & \text{par définition de } \delta \\ = & (i, j +_3 1) & \end{array}$$

Selon (1), on a $\#a(x) =_2 i$.
Par définition, $\#a(xb) = \#a(x) =_2 i$. CQFD.

Selon (1), on a $\#b(x) =_3 j$.
Trivialement, on a $\#b(xb) = \#b(x) + 1$.
Par les règles de l'addition modulo, on a alors $\#b(xb) =_3 i + 1$. CQFD.

$\blacksquare$