Automates et langages

L'examen corrigé — RICM1— 9 janvier 2002

1 Grammaire → Automate → Expression

On considère la grammaire régulière $G = (\Gamma,\Sigma,S,\Pi)$ avec $\Gamma = \{S,P,R\}$, $\Sigma = \{a,b\}$ et $\Pi = \{S \to P, P \to baR, P \to aS, R \to bb, R \to aP\}$.

1. Construire un automate A acceptant le langage défini par la grammaire G. Donner explicitement A sous la forme (Q,Σ,q_0,F,Δ).
2. Trouver une expression régulière pour ce même langage. *Indication: 3 équations suffisent.*
3. Trouver un automate déterministe minimal acceptant ce même langage.

2 Automates et Arithmétique

1. Pour le langage $M = \overline{(a^3)^* \cdot (a^4)^*}$ sur l'alphabet $\{a\}$ construire un automate qui le reconnaisse.
2. Est-ce que M est vide, non-vide et fini, ou bien infini? Si le langage M est fini, donner la liste de tous ses mots.
3. Appliquer ce résultat pour trouver tous les entiers naturels non représentables sous la forme $3m + 4n$ avec $m,n \in \mathbb{N}$.

3 Simplification

En passant par les automates transformer l'expression régulière étendue $(00+01)^* \cap 0(10+01)^*$ en forme régulière (non étendue).

4 Expressions arithmétiques polonaises

Dans les expressions arithmétiques en notation préfixée (ou polonaise) on met le symbole de l'opération avant les deux opérands et on ne met pas de parenthèses. Par exemple, au lieu de $2+3$ on écrit $+23$, au lieu de $(1+2) \times (3+4)$ on écrit $\times + 12 + 34$.

1. Décrire toutes les expressions polonaises sur 1,2,3,4,5 avec les opérations $+,\times$ par une BNF (Backus-Naur Form) ou une grammaire hors contexte.
2. Trouver des arbres syntaxiques pour $\times + 12 + 34$ et $\times + \times 1123$

5 Un langage non-régulier, mais hors-contexte

On considère le langage $L = \{a^{2k}ba^{3k} | k \in \mathbb{N}\}$

1. Prouver que L n'est pas régulier.
2. Trouver une grammaire hors-contexte G qui génère L. Donner cette grammaire explicitement sous la forme $G = (\Gamma,\Sigma,S,\Pi)$. Dériver les mots a^4ba^6 et b
3. Construire un automate à pile acceptant L.

6 Automates et Arithmétique Décimale

Construire les automates sur l'alphabet $\{0,1,2,3,\ldots,9\}$ qui acceptent tous les entiers naturels représentés en système décimal qui sont

1. multiples de 5;
2. multiples de 3;

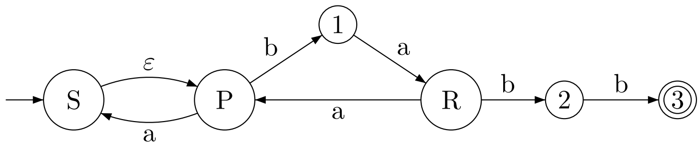
Indication: Un nombre décimal est multiple de 3 si et seulement si la somme de ses chiffres est multiple de 3.

Solutions

S1. Grammaire → Automate → Expression

On considère la grammaire régulière $G = (\Gamma,\Sigma,S,\Pi)$ avec $\Gamma = \{S,P,R\}$, $\Sigma = \{a,b\}$ et $\Pi = \{S \to P, P \to baR, P \to aS, R \to bb, R \to aP\}$.

1. En appliquant la méthode vue en cours on obtient



$A = (Q,\Sigma,q_0,F,\Delta)$ avec

$$
\begin{array}{rcl}
Q & = & \{P,R,S,1,2,3\} \\
\Sigma & = & \{a,b\} \\
q_0 & = & P \\
F & = & \{3\}, \Delta = \{S\varepsilon P, PaS, Pb1, 1aR, RaP, Rb2, 2b3\}
\end{array}
$$

2. On utilise les mêmes lettres S, P et R pour les langages accepté à partir des états S, P et R. Ces langages satisfont le système d'équations:

$$
\left\{
\begin{array}{rcl}
S & = & \varepsilon P \\
P & = & aS + baR \ ) \\
R & = & aP + bb
\end{array}
\right.
$$

La première équation donne $S = P$, en substituant les expressions pour S et R dans la deuxième équation on obtient

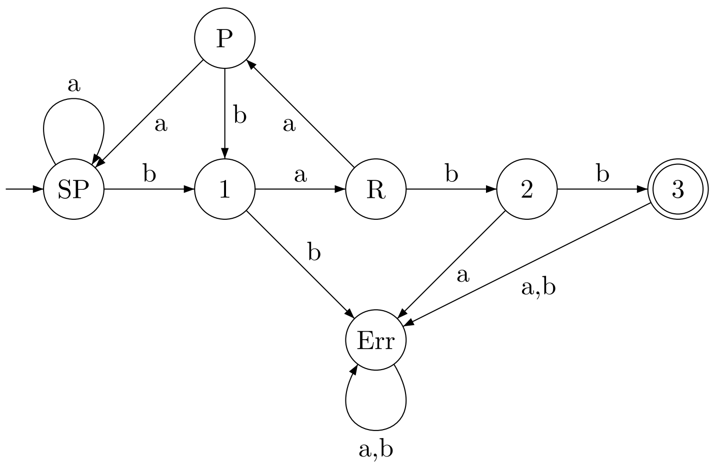
$$P = aP + ba(aP + bb)$$

ce qui est équivalent à

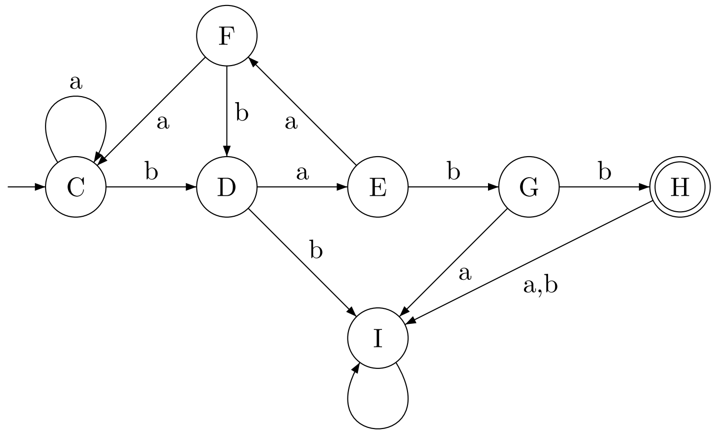
$$P = (a + baa)P + babb$$

On résout cette dernière équation: $P = (a+baa)^*babb$, d'où $L(A) = S = P = (a+baa)^*babb$.

3. En déterminisant l'automate A on obtient B:



Avant de minimiser on renomme les états



On applique l'algorithme de minimisation.

Initialisation:

	D	E	F	G	H	I
C					X	
D	-				X	
E	-	-			X	
F	-	-	-		X	
G	-	-	-	-	X	
H	-	-	-	-	-	X

Première itération:

	D	E	F	G	H	I
C				X	X	
D	-			X	X	
E	-	-		X	X	
F	-	-	-	X	X	
G	-	-	-	-	X	X
H	-	-	-	-	-	X

Deuxième itération:

	D	E	F	G	H	I
C		X		X	X	
D	-	X		X	X	
E	-	-	X	X	X	X
F	-	-	-	X	X	
G	-	-	-	-	X	X
H	-	-	-	-	-	X

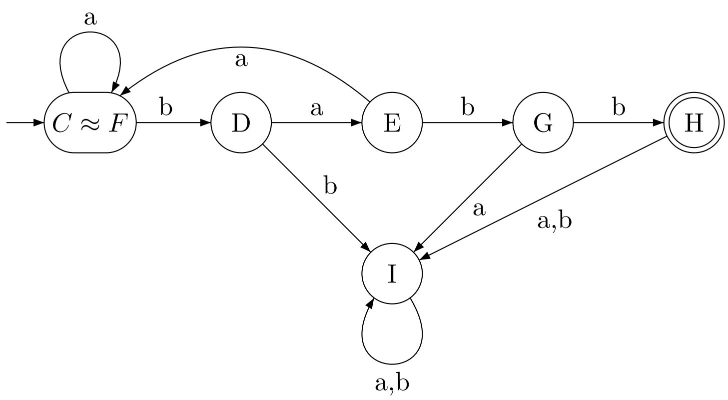
Troisième itération:

	D	E	F	G	H	I
C	X	X		X	X	
D	-	X	X	X	X	X
E	-	-	X	X	X	X
F	-	-	-	X	X	
G	-	-	-	-	X	X
H	-	-	-	-	-	X

Quatrième itération:

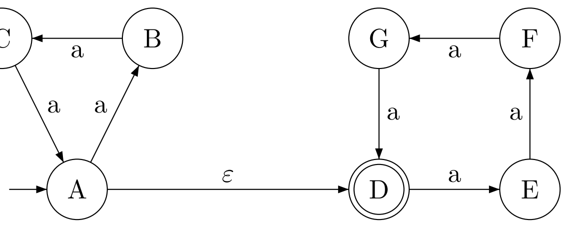
	D	E	F	G	H	I
C	X	X		X	X	X
D	-	X	X	X	X	X
E	-	-	X	X	X	X
F	-	-	-	X	X	X
G	-	-	-	-	X	X
H	-	-	-	-	-	X

L'itération suivante ne modifie pas le tableau. En observant les cases non-cochées on trouve la relation d'équivalence : $C \approx F$, les autres états ne sont pas équivalents. On construit l'automate minimal:

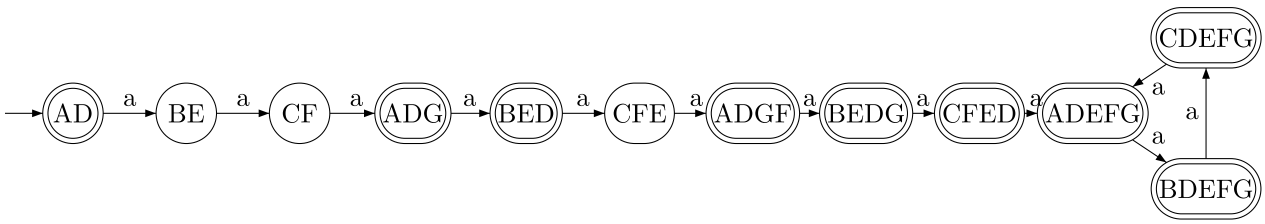


S2. Automates et Arithmétique

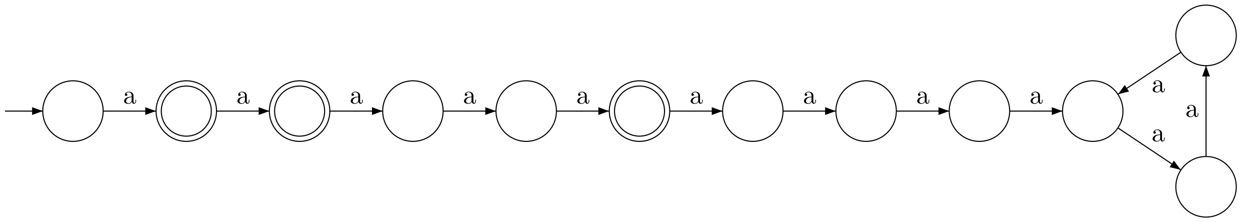
1. On construit d'abord un automate acceptant $(a^3)^* \cdot (a^4)^*$



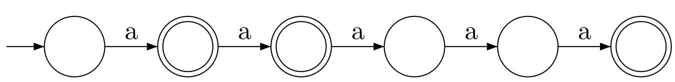
En le déterminisant on obtient:



En complémentant on obtient un automate pour M:



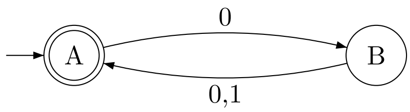
Finalement on le simplifie en supprimant les états à partir desquels les états accepteurs sont inaccessibles.



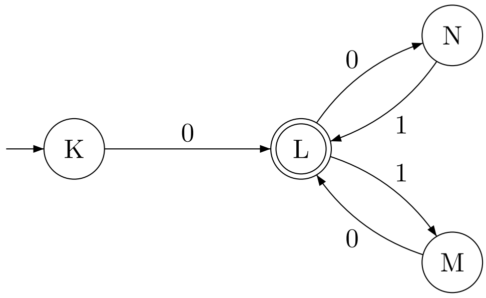
2. En observant l'automate on voit que $M = \{a,aa,aaaaa\}$ - non-vide et fini.
3. Un naturel k n'est pas représentable sous la forme $3m + 4n$ si et seulement si $a^k \in M$, d'où la réponse: 1,2,5.

S3. Simplification

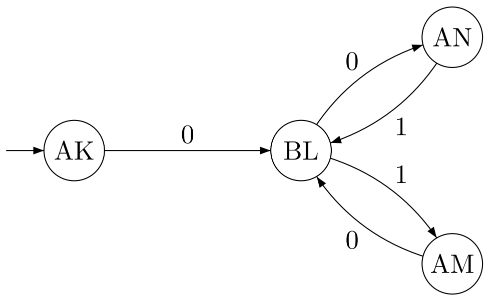
Un automate A1 pour $(00+01)^*$:



Un automate A2 pour $0(10+01)^*$:



L'automate produit de A1 et A2 accepte l'intersection $(00+01)^* \cap 0(10+01)^*$:



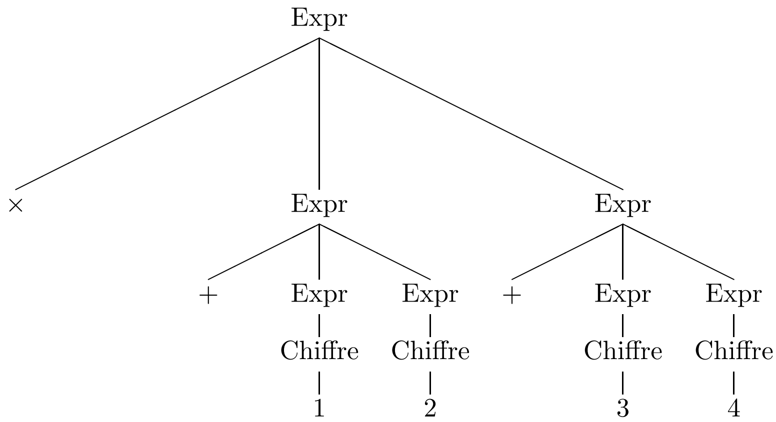
Comme il n'a pas d'états accepteurs, son langage est vide, d'où l'expresion régulière : $\emptyset$

S4. Expressions arithmétiques polonaises

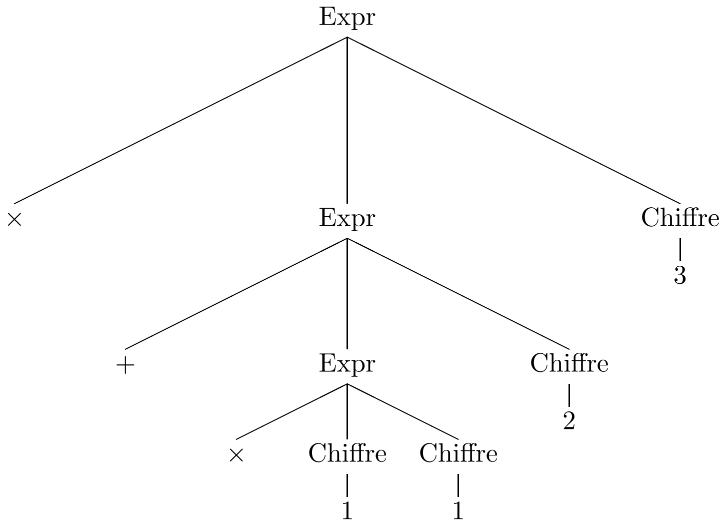
1. La BNF:

$$\begin{array}{rcl} \langle \text{Expr} \rangle & ::== & \langle \text{Chiffre} \rangle \mid +\langle \text{Expr} \rangle\langle \text{Expr} \rangle \mid \times\langle \text{Expr} \rangle\langle \text{Expr} \rangle \\ \langle \text{Chiffre} \rangle & ::== & 1|2|3|4|5 \end{array}$$

2. L'expression $\times + 12 + 34$ "signifie" $(1+2) \times (3+4)$. Son arbre syntaxique:



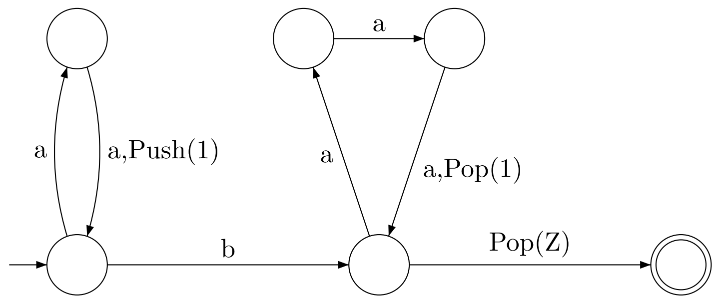
L'expression $\times + \times 1123$ "signifie" $((1 \times 1) + 2) \times 3$. Son arbre syntaxique:



S5. Un langage non-régulier, mais hors-contexte

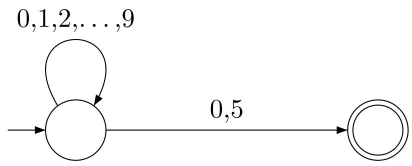
1. Pour démontrer que $L = \{a^{2k}ba^{3k} | k \in \mathbb{N}\}$ n'est pas régulier on utilise la méthode de preuve par contradiction.
   Supposons que le langage L est régulier. Donc il existe un automate déterministe fini qui accepte S. Soit M le nombre d'états de cet automate.
   On choisit un mot particulier $w = a^{2M}ba^{3M}$. Par définition du langage L on a $w \in L$.
   Comme $|w| = 5M + 1 > M$, on peut appliquer le lemme de gonflement. Ce lemme dit, qu'il existe une décomposition $w = xuy$ avec $u \neq \varepsilon$ et $|xu| \leq M$, telle que tous les mots de la forme xu^iy appartiennent aussi au langage S. *On ne sait pas quels sont les 3 morceaux x, u et y de la décomposition, mais pumping lemma garantit leur existence.*
   Comme $|xu| \leq M$, les morceaux x et u sont dans les M premiers caractères du mot w et ne peuvent contenir que des a. Soient m et $n \neq 0$ les nombres de lettres a dans x et u respectivement.
   Donc, on a $x = a^m$, $u = a^n$, et, comme $w = xuy$, le dernier morceau y ne peut être autre chose que $a^{2M-m-n}ba^{3M}$.
   Le mot "gonflé" $w' = xu^2y$ est de la forme xu^iy et **doit appartenir** au langage L. Mais
   $$w' = xu^2y = a^m(a^n)^2a^{2M-m-n}ba^{3M} = a^{m+2n+(2M-m-n)}ba^{3M} = a^{2M+n}ba^{3M}.$$
   Comme $\dfrac{2M+n}{3M} > \dfrac{2}{3}$ (on a utilisé le fait que $n > 0$) et par définition du langage L, ce mot **ne peut pas appartenir** au même langage L. La contradiction obtenue conclue la preuve.
2. $G = (\Gamma, \Sigma, S, \Pi)$ avec
   $$\begin{aligned} \Gamma &= \{S\} \\ \Sigma &= \{a,b\} \\ \Pi &= \{S \to aaSaaa, S \to b\} \end{aligned}$$
   Dérivation de a^4ba^6: $S \to aaSaaa \to aaaaSaaaaaa \to aaaabaaaaaa$
   Dérivation de b: $S \to b$

3. L'automate à pile:



S6. Automates et Arithmétique Décimale

1. Un nombre décimal est multiple de 5 s'il se termine par 0 ou par 5, d'où l'expression régulière $(0+1+2+\ldots+9)^*(0+5)$ et l'automate



2. L'idée est de calculer la somme des chiffres modulo 3, on aura besoin de trois états 0,1 et 2 pour représenter la somme mod 3, on accepte lorsque la somme est multiple de 3 (c-à-d dans l'état 0). Pour ne pas accepter le mot vide ε (qui n'est pas un nombre décimal) on utilise encore un état I.