

TP6 - IFT2105

par Ilan Elbaz

17 juin 2019

Machine de Turing - Rappel

Une machine de Turing est un 6-tuplet $(Q, \Sigma, \Gamma, \delta, q_0, F)$ où:

- Q est un ensemble fini d'états,
- Σ est l'alphabet,
- Γ est l'alphabet de ruban. $\sqcup \in \Gamma$ et $\Sigma \subset \Gamma$,
- $\delta : Q \times \Gamma \rightarrow Q \times \Gamma \times \{D, G\}$ fonction de transition,
- q_0 l'état initial,
- $F = \{q_A, q_R\}$ les états finaux.

Une configuration d'une machine de Turing M est un triplet (u, q, v) où $q \in Q$ et $u, v \in \Gamma^*$. On l'interprète par " M contient uv , suivi d'une chaîne infinie de \sqcup et la tête de lecture est au-dessus du premier symbole de v ".

Dans les sections suivantes, on va montrer que le modèle précédent est robuste, i.e. on va montrer que ses relaxations sont polynomialement équivalentes au modèle standard.

1. Machine de Turing avec possibilité de tête immobile

Étudions le modèle de machine de Turing dans lequel en plus de pouvoir déplacer la tête à droite et à gauche, on lui permet de rester immobile.

Ses transitions sont de la forme: $\delta : Q \times \Gamma \rightarrow Q \times \Gamma \times \{D, G, S\}$

Soit M une machine de Turing avec possibilité de tête immobile. Montrons qu'on peut simuler M sur une machine de Turing standard.

On bâtit M' une machine de Turing standard. Elle est identique à M avec l'exception que les transitions qui ne bougent pas la tête sont remplacées par une paire de transitions qui déplace la tête à gauche puis à droite.

On remplace les transitions $\delta(q, s) = (q', s', S)$ de M par la paire $\delta(q, s) = (q'_g, s', G)$ et $\delta(q'_g, s') = (q', s0, D)$. Ceci ajoute Q états et autant de transitions. Avec cela on simule l'immobilité avec une opération élémentaire supplémentaire. Si M fonctionne en temps polynomial, notre construction fonctionne aussi en temps polynomial.

Soit M une machine de Turing standard. Montrons qu'on peut simuler M sur une machine de Turing avec possibilité de tête immobile.

On pose $M' := M$. On n'a aucune modification à faire. M' n'utilisera simplement jamais la possibilité de rester immobile.

2. Machine de Turing avec un changement d'alphabet

On définit un encodage binaire E des symboles de Σ dans lequel chaque symbole est encodé avec $\log(|\Sigma|)$ bits.

Supposons qu'il existe une machine de Turing M qui reconnaît un langage L sur alphabet Σ . Montrez que $E(L)$ peut être reconnu par une machine de Turing M' d'alphabet de ruban $\Gamma = 0, 1, t$. De plus, montrez que si M fonctionne en temps polynomial, M' est également polynomial.

Pour chaque paire état/symbole (q, s) , on simule sur M' la transition $\delta(q, s) = (q', s', d)$ de M de la façon suivante :

Lecture

Pour la lecture de s encodé en binaire sur le ruban de la machine dans l'état $q \in Q$, on doit ajouter $\log|\Sigma|$ états. $(q_{E(s)_1}, q_{E(s)_1, E(s)_2}, \dots, q_{E(s)_1, E(s)_2, \dots, E(s)_{\log|\Sigma|}})$ et ajouter autant de transitions. À la fin de la lecture de $E(s)$, faite en $\log|\Sigma|$ opérations, on connaît s et la tête se situe dans la dernière case encodant s .

Écriture

On veut remplacer les $\log|\Sigma|$ derniers symboles par $E(s')$. Pour remplacer $\log|\Sigma|$ symboles, on a besoin d'ajouter $\log|\Sigma|$ états $(q_{s', \text{dernier}}, \dots, q_{s', \text{premier}})$ et autant de transitions et de faire $\log|\Sigma|$ opérations. La tête est maintenant dans la première case encodant s' . On doit maintenant déplacer la tête dans la direction donnée par d .

Déplacement

Pour se déplacer à droite on doit avancer de $\log|\Sigma|$ cases. On le fait avec $\log|\Sigma|$ états, autant de transitions et $\log|\Sigma|$ opérations. Pour se déplacer à gauche on doit reculer de $\log|\Sigma|$ cases. On le fait avec $\log|\Sigma|$ états, autant de transitions et $\log|\Sigma|$ opérations. On peut maintenant passer à l'état q' et effectuer l'opération suivante.

Complexité

Il y a $|\Sigma| \times Q$ paires (q, s) possibles. La lecture, l'écriture et le déplacement

ajoutent au total $3 \times |\Sigma| \times |Q| \times \log|\Sigma|$ états et autant de transitions à la description de la machine M' .

On simule chaque transition de M avec $3|\Sigma|\log|\Sigma|$ opérations dans M' . Si M est polynomial, M' l'est aussi.

3. Machine de Turing avec plusieurs têtes et plusieurs rubans

On étudie le modèle de machine de Turing avec un nombre fini k de rubans et une tête par ruban.

Ses transitions sont de la forme: $\delta : Q \times \Gamma^k \rightarrow Q \times \Gamma^k \times \{D, G\}^k$

Une configuration de M donne la position de chaque tête, le contenu de chaque ruban et l'état de la machine. On le représente par le tuple:

$$(u^1, v^1, u^2, v^2, \dots, u^k, v^k, q)$$

Où $u^i v^i$ est le contenu du i^e ruban, où la tête est située au premier symbole de v et q est l'état.

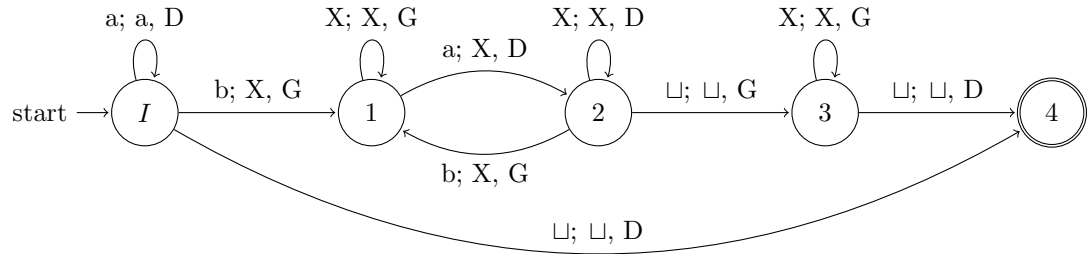
Soit M une machine de Turing à k rubans. Montrons qu'on peut simuler M sur une machine de Turing standard.

On écrit le contenu des k rubans sur le seul ruban de M' de la manière suivante : la première case du premier ruban va dans la première case, la première case du deuxième ruban va dans la deuxième case,... , la première case du k^e ruban va dans la k^e case, la 2^e case du premier ruban va dans la $k + 1^e$ case et ainsi de suite. Si $i = m(\text{mod } k)$ et $i = nk + m$, la i^e case du ruban de M' est associée à la n^e case du m^e ruban de M .

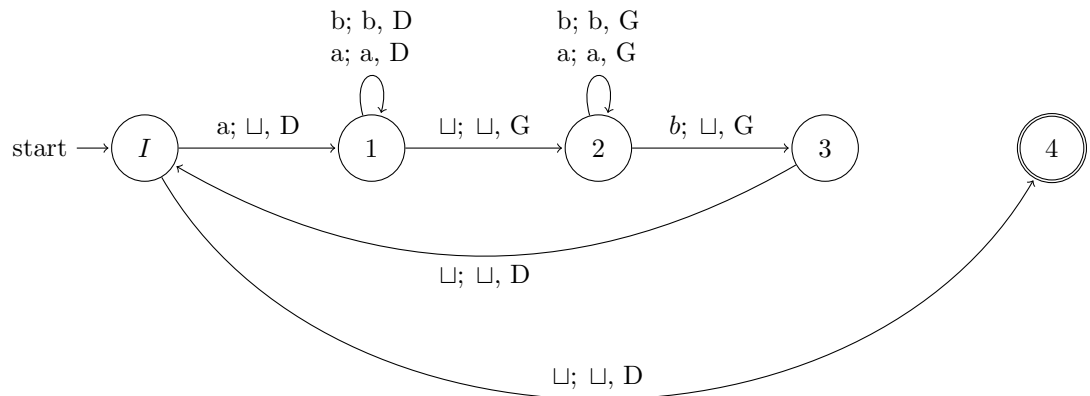
On représente la position de la tête en créant $|\Gamma|$ symboles spéciaux (un pour chaque lettre). Si la tête du m^e ruban pointe à la position n contenant le symbole a , on écrit \hat{a} à la case $nk + m$ de M'

Pour effectuer ses transitions, M' dans l'état q commence par trouver les k symboles qui ont un point au-dessus, disons $\hat{s}_1, \dots, \hat{s}_k$. Pour simuler la transition $(s_1, \dots, s_k, q) \rightarrow (s'_1, \dots, s'_k, q, d_1, \dots, d_k)$ M' remplace chaque \hat{s}_i par s'_i , met un point au-dessus de la k^e case de droite ou la k^e case de gauche selon d_i . À la fin, M' passe à l'état q' .

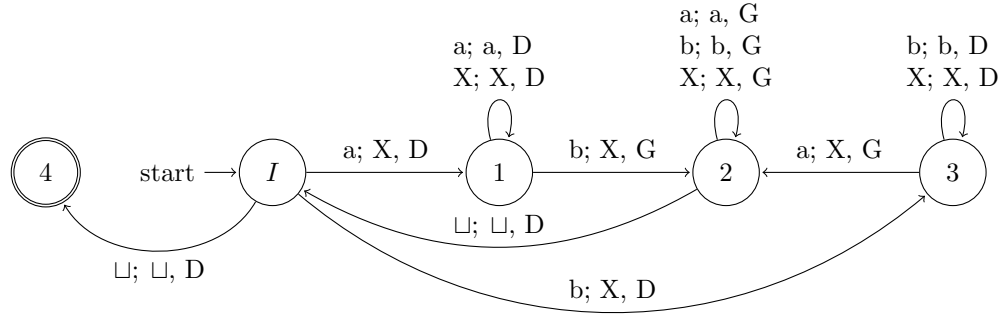
4. Soit $\Sigma = \{a, b\}$. Définir une machine de Turing permettant de reconnaître le langage $L = \{a^n b^b\}$. Que retourne la machine sur "aaabbb" et "aaabb"?



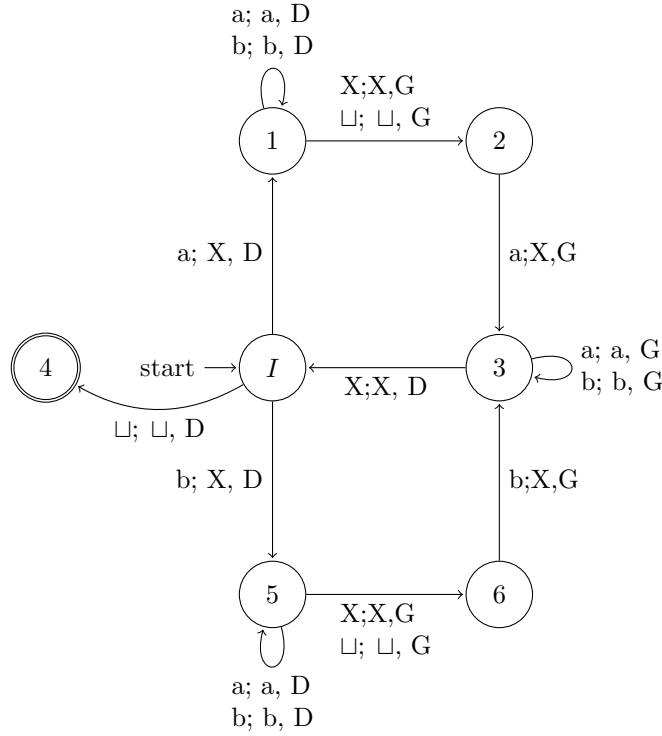
Autre possibilité:



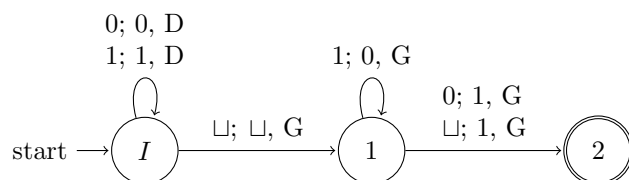
5. Soit $\Sigma = \{a, b\}$. Définir une machine de Turing permettant de reconnaître le langage $L = \{|w|_a = |w|_b\}$.



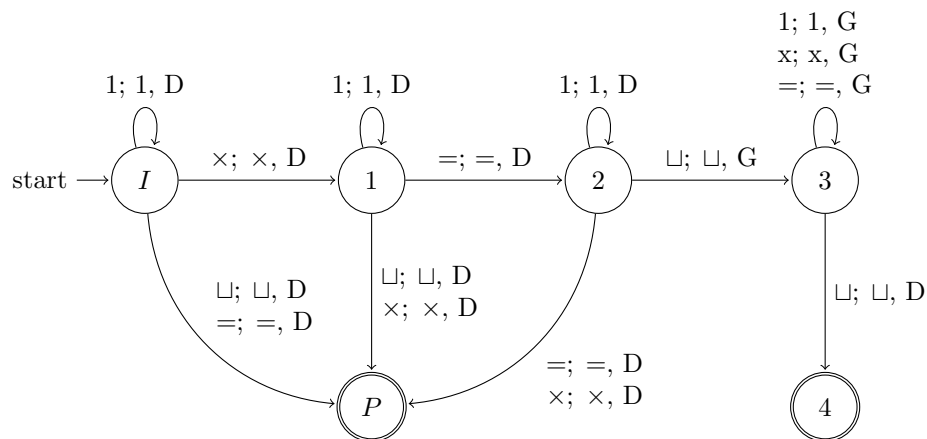
6. Soit $\Sigma = \{a, b\}$. Définir une machine de Turing permettant de reconnaître le langage $L = \{ww^R\}$.



6. Définir une machine de Turing permettant d'ajouter 1 aux mots binaires.



7. Définir une machine de Turing permettant de reconnaître les mots de la forme $1^* \times 1^* = 1^*$



8. Soit M la machine de Turing définie par la fonction de transition suivante:

δ	a	b	c
q_0	q_0, a, D	q_0, b, D	q_1, c, G
q_1	q_1, b, G	q_1, a, G	

Pour M :

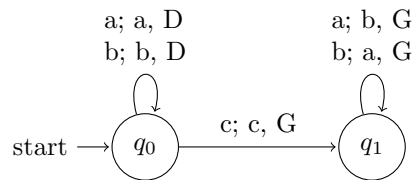
1. Donnez la trace d'exécution de M sur l'entrée $abcab$

$$\begin{aligned}
 q_0 abcab &\vdash aq_0 bcab \\
 &\vdash abq_0 cab \\
 &\vdash aq_1 bcab \\
 &\vdash q_1 aacab \\
 &\vdash q_1 bacab
 \end{aligned}$$

2. Donnez la trace d'exécution de M sur l'entrée $abab$

$$\begin{aligned}
 q_0 abab &\vdash aq_0 bab \\
 &\vdash abq_0 ab \\
 &\vdash abaq_0 b \\
 &\vdash ababq_0
 \end{aligned}$$

3. Dessiner le diagramme d'état de M



4. Que fait la machine M ?
Le résultat de l'exécution de M correspond à remplacer tout les a en b et inversement avant le premier c. Si il n'y en pas, la chaîne reste identique.