TD 01 – Machines de Turing

Exercice 1. *Bijections*

Soient A et B deux ensembles, une fonction $f : A \to B$ est

— injective ssi $\forall\, a, a' \in A : f(a) = f(a') \implies a = a'$ (ou la contraposee),

— surjective ssi $\forall\, b \in B : \exists\, a \in A : b = f(a)$,

— bijective ssi elle est à la fois injective et surjective.

1. Donner cinq éléments de l'ensemble $\mathbb{N} \times \{0, 1, a\}$.
2. Donner une bijection de $\mathbb{N} = \{0, 1, 2, \ldots\}$ dans $2\mathbb{N} = \{0, 2, 4, \ldots\}$.
3. Donner une bijection de $\mathbb{N}$ dans $\mathbb{N} \times \mathbb{N}$.
4. Donner une bijection de $\mathbb{N}$ dans $\mathbb{N} \times \mathbb{N} \times \mathbb{N}$.
5. Donner une bijection de Σ^* dans $\mathbb{N}$.

Exercice 2. *MT : états = mémoire finie*

Objectif : voir que l'on peut sauvegarder un symbole (ou plusieurs) dans l'état.

Soit $M = (Q, \Sigma, \Gamma, \delta, q_0, B, q_F)$ la machine de Turing où

— $Q = \{q_0, q_a, q_b, q_F\}$,

— $\Sigma = \{a, b\}$, $\Gamma = \{a, b, B\}$,

— δ est donnée par

$(q_0, a) \mapsto (q_a, a, R)$		$(q_0, b) \mapsto (q_b, b, R)$	
$(q_a, a) \mapsto (q_a, a, R)$	$(q_a, b) \mapsto (q_a, b, R)$	$(q_b, a) \mapsto (q_b, a, R)$	$(q_b, b) \mapsto (q_b, b, R)$
$(q_a, B) \mapsto (q_F, a, R)$		$(q_b, B) \mapsto (q_F, b, R)$	

1. Dessiner cette machine sous la forme d'un automate.
2. Dans quelle configuration est-on à la fin de l'exécution de M sur le mot d'entrée $abab$?
3. Quelle fonction est calculée par cette machine ?

Exercice 3. *MT : décalage*

Objectif : voir que l'on peut facilement faire un décalage avec une MT.

Soit $\Sigma = \{a, b\}$, et la fonction $f : \Sigma^* \to \Sigma^*$ définie par

$$\text{pour tout } w \in \Sigma^* \text{ on a } f(w) = aw$$

Le but étant de décaler tous les symboles d'une case sur la droite (afin de réutiliser cette machine ensuite).

1. Dessiner l'automate d'une machine qui calcule f.

Exercice 4. *MT : composition*

Objectif : voir qu'une MT peut utiliser (simuler) une autre MT.

Soit $\Sigma = \{a, b\}$, et la fonction $g : \Sigma^* \to \Sigma^*$ définie par

$$\text{pour tout } w = w_1 w_2 \ldots w_{2k} \in \Sigma^* \text{ on a } g(w) = w_1 \ldots w_k a w_{k+1} \ldots w_{2k}$$

Pour simplifier on supposera que les entrées sont toujours de longueur paire.

1. Dessiner l'automate d'une machine qui calcule g.

Exercice 5. *MT : décider et calculer*

Donner des machines de Turing pour décider/calculer les langages/fonctions suivant(e)s.

1. $L = \{aw \mid w \in \Sigma^*\}$ avec $\Sigma = \{a, b, c, d, e\}$.
2. $L = \{w \in \Sigma^* \mid |w| \equiv 0 \mod 3\}$ avec $\Sigma = \{a, b\}$.
3. $L = \{w \in \Sigma^* \mid w$ est un palindrome$\}$ avec $\Sigma = \{a, b, c\}$.
4. La fonction d'incrémentation $(+1)$ en binaire sur $\Sigma = \{0, 1\}$.
5. La fonction de décrémentation (-1) en binaire sur $\Sigma = \{0, 1\}$.
6. $L = \{a^n b^n c^n \mid n \in \mathbb{N}\}$ avec $\Sigma = \{a, b, c\}$.
7. $L = \{a^n b^{\lfloor \frac{n}{2} \rfloor} \mid n \in \mathbb{N}\}$ avec $\Sigma = \{a, b\}$.
8. La fonction d'addition de deux nombres binaires sur $\{0, 1, \#\}$, donnés par $w\#w'$. On supposera que les entrées sont toujours bien formées, par exemple $1010\#111$ donnera 10001.

Exercice 6. *Propriétés de clôture*

Donner un exemple puis démontrer chacun des résultats suivants.

1. la famille des langages récursifs est close par complémentation ;
2. les familles des langages récursifs et r.e. sont closes par union et intersection ;
3. Un langage $L \subseteq \Sigma^*$ est récursif si et seulement si L et $\Sigma^* \setminus L$ sont r.e.

Exercice 7. *MT : conventions*

Objectif : voir que l'on peut utiliser d'autres conventions.

1. Peut-on calculer exactement les mêmes fonctions / décider exactement les mêmes langages si la tête de lecture est initialement placée sur la case la plus à droite du mot d'entrée ? (Justifier)
2. Peut-on calculer les mêmes fonctions / décider les mêmes langages si l'on autorise des transitions pour lesquelles la tête de lecture/écriture ne bouge pas (ni L ni R) ? (Justifier)
3. Peut-on calculer les mêmes fonctions / décider les mêmes langages si l'on ajoute la restriction $\Sigma = \{0, 1\}$? Et si en plus $\Gamma = \{0, 1, B\}$? (Justifier)

Exercice 8. *MT : non-déterministes*

Objectif : montrer que le modèle des machines de Turing non-déterministes est équivalent au modèle des machines de Turing.

Une machine de Turing non-déterministe peut, à une étape de temps donnée (c'est à dire dans un état et pour un symbole lu), avoir plusieurs transitions possibles. Un calcul d'une machine de Turing est une succession de choix parmi l'ensemble des transitions possibles. Un mot est accepté si et seulement si il existe au moins un calcul qui mène à l'état final.

1. Donner le type de la fonction de transition δ, et donner un exemple de transition.
2. Donner une machine de Turing non déterministe qui reconnaisse le langage $L = \{ww \mid w \in \{a, b\}^*\}$.
3. Soit $r = \max\{|\delta(q, x)| \mid q \in Q \text{ et } x \in \Gamma\}$. Que représente r ?

4. Que pourrait représenter une suite finie de lettres sur l'alphabet $R = \{1, \ldots, r\}$?
5. Peut-on construire une machine de Turing qui énumère (c'est-à-dire écrit un à un et se place dans un état particulier q_e à chaque fois que son ruban contient un mot à énumérer) tous les mots finis sur l'alphabet R ?
6. Montrer que tout langage reconnu par une MT non-déterministe est reconnu par une MT déterministe (indication : trois rubans avec chacun leur tête semble un bon choix, ou bien on déterminise comme on le fait avec les automates finis en prenant $Q' = P(Q)$).

Exercice 9. *MT : multi-ruban*

Objectif : montrer que le modèle des machines de Turing à plusieurs rubans et plusieurs têtes de lecture (une tête de lecture indépendante par ruban, et un seul état pour toute la machine) est équivalent au modèle des machines de Turing.
En fonction de l'état et du symbole lu sur chacun des rubans, la machine peut
- changer d'état,
- écrire un symbole sur chaque ruban,
- déplacer chaque tête de lecture vers la droite ou la gauche indépendamment les unes des autres.

1. Donner le type de la fonction de transition δ, et donner un exemple de transition.

Dans l'état initial, l'entrée est écrite sur le premier ruban et tous les autres rubans sont vides. Un mot est accepté si et seulement si la machine entre dans l'état final q_F au cours du calcul.

2. Comment simuler une MT *multi-tape* avec une MT *mono-tape* ?