TD 02 – Machines de Turing et limites du calcul

Exercice 1. *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 2. *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 3. *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* ?

Exercice 4. *MT et pseudo-code*

Soit le pseudo-code suivant.

```
h(x : tableau de bits de taille n)
  i <- n
  tant que ( x[i] == 1 et i >= 1 ) faire
    i <- i-1
  fin tant que
  si i == 0 alors
    accepter
  sinon
    rejeter
  fin si
```

1. Donner une machine de Turing qui décide le même langage que h, et qui s'arrête toujours.
2. Quel est le langage décidé par l'algorithme de la question 1 ?

Soit le pseudo-code suivant.

```
f(n : entier)
  si n%2==0 alors
    return n/2
  sinon
    return n
```

3. Donner une machine de Turing qui calcule f.

Soit le pseudo-code suivant.

```
g(n : entier)
  return 2^n
```

4. Donner une machine de Turing qui calcule g.
5. Sauriez-vous construire une machine de Turing pour convertir une entrée codée en binaire, en une sortie de même valeur codée en unaire ?

Exercice 5. *MT : décider et calculer*

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

1. $L = \{ww \mid w \in \Sigma^*\}$ sur $\Sigma = \{a, b\}$.
2. $L = \{w_1 w_2 \ldots w_n \mid w_i \in \Sigma$ et $w_2 = a$ et $w_4 = b\}$ sur $\Sigma = \{a, b\}$.
3. $f : \mathbb{N} \to \mathbb{N}$ telle que $f(n) = 4 * n$ avec n donné en binaire.

4. $f : \Sigma^* \to \Gamma^*$ telle que $f(w) = w\#w$ avec $\Sigma = \{a, b\}$
5. $L = \{x_1\#x_2 \mid x_1 < x_2\}$ avec x_1 et x_2 donnés en binaire.

Exercice 6. *Infinis*

Argumenter chacune des égalités suivantes.

1. $|\mathbb{N}| = |\mathbb{Z}|$.
2. $|\mathbb{N}| = |\mathbb{Q}|$.
3. $|\mathbb{R}| = |[0, 1]|$.
4. $|\mathbb{R}| = |\mathbb{R} \times \mathbb{R}|$.

Exercice 7. $\langle M \rangle$

Donner le code $\langle M \rangle$ complet de la machine de Turing M suivante.
$Q = \{q_0, q_1, q_F\}$, $\Sigma = \{0, 1\}$, $\Gamma = \{0, 1, B\}$ et
$\delta(q_0, 0) = (q_1, 1, R)$, $\delta(q_1, 1) = (q_1, 0, R)$, $\delta(q_1, B) = (q_F, B, L)$.

Exercice 8. *Récursivement énumérable*

Une machine de Turing qui énumère un langage récursivement énumérable (r.e.) possède un « état d'énumération » supplémentaire, $q_e \in Q$, dans lequel elle va pour énumérer un nouvel élément du langage. On peut choisir les conventions d'énumération, et l'usage de plusieurs rubans est judicieux. Dans cet exercice, on énumère le mot qui se trouve **à droite de la tête**.

1. Montrer que le langage $L = (1\,\{0, 1\}^*) \setminus \{1, 10, 11\}$ est r.e.
2. Donner une machine de Turing qui énumère les éléments du langage L.
3. L est-il récursif ?
4. Donner un exemple de langage ou fonction r.e. mais pas récursif.

Exercice 9. *Espace et temps*

Supposons qu'une machine de Turing s'arrête au bout de t étapes de calcul en consommant s cases du ruban.

1. Quelle(s) relation(s) existe(nt) entre t et s ?