TD 01 – Rappels de calculabilité

Rappel : 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 contraposée),
- surjective ssi $\forall\, b \in B : \exists\, a \in A : b = f(a)$,
- bijective ssi elle est à la fois injective et surjective.

Utile : Théorème de Cantor-Schröder-Bernstein : soient A et B deux ensembles, si il existe une fonction injective de A vers B (intuitivement $|A| \leq |B|$), et une fonction injective de B vers A (intuitivement $|B| \leq |A|$), alors il existe une bijection entre A et B (intuitivement $|A| = |B|$).

Exercice 1. *Ensembles infinis dénombrables*

1. Donner une bijection entre $\mathbb{N} = \{0, 1, 2, \ldots\}$ et $\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, \ldots\}$.
2. Donner une bijection entre $\mathbb{N}$ et $\mathbb{N} \times \mathbb{N}$.
3. Peut-on en déduire que $|\mathbb{Q}| = |\mathbb{N}|$?
4. Donner une bijection entre $\mathbb{N}$ et $\mathbb{N} \times \mathbb{N} \times \mathbb{N}$.
5. Donner une bijection entre Σ^* et $\mathbb{N}$, pour Σ un alphabet fini.

Exercice 2. *Ensembles infinis indénombrables*

1. Donner une bijection entre $\mathcal{P}(\mathbb{N})$ (l'ensemble des parties de $\mathbb{N}$) et $[0, 1]$.
2. Donner une bijection entre l'ensemble des langages sur un alphabet fini Σ, et $[0, 1]$.
3. Donner une bijection entre $[0, 1]$ et $\mathbb{R}$.

Exercice 3. *MT*

Donner des machines de Turing pour décider les langages suivants.
1. $L = \{aw \mid w \in \Sigma^*\}$ avec $\Sigma = \{a, b, c\}$.
2. $L = \{w \in \Sigma^* \mid |w| \equiv 0 \mod 3\}$ avec $\Sigma = \{a\}$.
3. $L = \{w \in \Sigma^* \mid w$ est un palindrome$\}$ avec $\Sigma = \{a, b\}$.

Exercice 4. *L'arrêt*

Indiquer si chacun des énoncés qui suit est vrai ou faux, en justifiant.
1. $\nexists M_{halt}, \forall M, \forall w : M_{halt}(\langle M \rangle, w) = halt(\langle M \rangle, w)$.
2. $\forall M, \forall w, \nexists M_{halt} : M_{halt}(\langle M \rangle, w) = halt(\langle M \rangle, w)$.

Exercice 5. *Espace et temps*

Supposons qu'une machine de Turing s'arrête au bout de t étapes de calcul en visitant (possiblement plusieurs fois chacune) s cases du ruban. Quelle(s) relation(s) existe(nt) entre t et s ?

Exercice 6. *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. Le multi-ruban est très pratique !

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 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-ruban* avec une MT *mono-ruban* ?

Exercice 7. *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 $r = \max\{|\delta(q, x)| \mid q \in Q$ 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 : on pourra utiliser trois rubans, le premier pour l'entrée, le second pour énumérer les choix non-déterministes, le troisième pour simuler une exécution particulière de la machine non-déterministe).

Exercice 8. *Utilité des réductions*

Démontrer les quatre énoncés suivants.

1. Si $A \leq_m^T B$ et B est décidable alors A est décidable.

2. Si $A \leq_m^T B$ et B est semi-décidable alors A est semi-décidable.

3. Si $A \leq_m^T B$ et A n'est pas décidable alors B n'est pas décidable.

4. Si $A \leq_m^T B$ et A n'est semi-décidable alors B n'est pas semi-décidable.

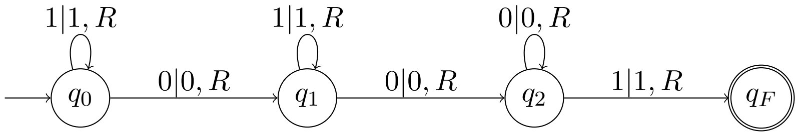
Exercice 9. *Ma première réduction Turing many-one*

1. Réduire $L_{halt\epsilon} = \{\langle M\rangle \mid M$ s'arrête quand on la lance sur l'entrée vide $\}$
à $A = \{\langle M\rangle \mid M$ s'arrête quand on la lance sur l'entrée $aa\}$.

2. En utilisant l'exercice 8, que peut-on en déduire ?

Exercice 10. *Complémentaire*

Soit la machine de Turing M suivante sur l'alphabet d'entrée $\Sigma = \{0, 1\}$.



1. Cette machine de Turing s'arrête-elle sur toute entrée ?

2. Construire une machine de Turing M' telles que $L(M') = \Sigma^* \setminus L(M)$.

Exercice 11. *Réductions Turing many-one*

Ecrire chacune des réductions (Turing many-one) suivantes, et indiquer ce que l'on peut en déduire quant à la récursivité de ces langages.

1. Réduire $L_u = \{\langle M\rangle\#w \mid M$ accepte le mot $w\}$ à $B = \{\langle M\rangle \mid a \in L(M)\}$.

2. Réduire L à $aL = \{aw \mid w \in L\}$ pour tout langage L.

3. Réduire aL à L pour tout langage L.

4. Réduire $L_{\bar{u}} = \{\langle M\rangle\#w \mid M$ n'accepte pas $w\}$
à $C = \{\langle M\rangle\#w \mid M$ n'accepte pas w mais accepte $bbw\}$.

5. Réduire $L_{stupide} = \{a\}$ à L_u.

Exercice 12. *Lancer deux MT en parallèle*

Soient deux machines mono-ruban $M_A = (Q_A, \Gamma_A, \Sigma_A, q_0^A, B_A, q_F^A, \delta_A)$,
et $M_B = (Q_B, \Gamma_B, \Sigma_B, q_0^B, B_B, q_F^B, \delta_B)$.

1. Définir une machine M à deux rubans, telle que $L(M) = L(M_A) \cup L(M_B)$.

2. Dans le cas où $\Sigma_A = \Sigma_B$ et $L(M_A) = \Sigma_A^* \setminus L(M_B)$, comment adapter la construction pour créer une machine de Turing M' à deux rubans qui décide le langage $L(M_A)$?

Exercice 13. *Semi-décidable mais pas décidable*

Donner un exemple de langage semi-décidable, mais pas décidable (justifier).