Calculabilité avancée (SINBU06) – Examen – Session 1

Durée : 2 heures *(Barème indicatif)*
Documents : non autorisés
Ce sujet comporte 1 page et 4 exercices

Soit $A = \{\langle M \rangle \mid L(M) = \{a, ab, aba\}\}$ l'ensemble des codes de machines de Turing dont le langage contient les trois mots a, ab, aba, et aucun autre mot.

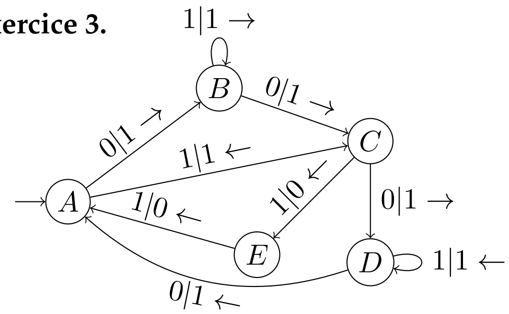
Exercice 1. *Rice (2 points)*

✎ Appliquer le théorème de Rice pour montrer que A n'est pas un langage décidable.

Exercice 2. *Réduction (8 points)*
Soit $L_u = \{(\langle M \rangle, w) \mid w \in L(M)\}$ l'ensemble des couples (code de machine de Turing, mot) tels que la machine de Turing accepte le mot.

1. Donner l'idée d'un algorithme pour montrer que L_u est semi-décidable.
Décrire brièvement son comportement sur les couples $(\langle M \rangle, w)$ qui sont dans L_u, et sur ceux qui ne sont pas dans L_u.
2. Montrer que $L_u \leq_m^T A$ (réduction many-one Turing).

Exercice 3. *Automate cellulaire affairé (5 points)*



La machine ci-contre détient le record actuel du castor affairé à 5 états sur l'alphabet $\{0, 1\}$ (où 0 est le symbole blanc), qui s'arrête sur l'entrée vide en $47\,176\,870$ étapes de calcul.
Elle s'arrête sur l'état E en lisant 0.

1. Pour un automate cellulaire en dimension $d = 1$, avec $|Q| = n$ états et voisinage $\{-1, 0, 1\}$ (la cellule elle-même et ses deux plus proches voisines), quel est le *type* de la *règle locale* ? (c'est-à-dire, la règle locale est une fonction de quel ensemble vers quel ensemble ?)
2. En vous inspirant de la machine de Turing ci-dessus, définir la règle locale d'un automate cellulaire en dimension $d = 1$, avec 12 états et voisinage $\{-1, 0, 1\}$, telle que dans l'automate cellulaire associé il existe une configuration (on vous demande de préciser laquelle) qui atteint un point fixe en $47\,176\,870$ étapes de calcul et pas avant.
Indication : l'automate cellulaire peut *simuler* la machine de Turing.

Exercice 4. *Auto-référence (5 points)*
Pour une machine de Turing M, on note $t(M) = \min\{|\langle M' \rangle| \mid L(M') = L(M)\}$ la taille du plus court code de machines de Turing qui accepte le même langage que M.
Soit $T = \{\langle M \rangle \mid \langle M \rangle = t(M)\}$ l'ensemble des codes de machines de Turing telles qu'il n'existe pas de machine dont le code est strictement plus court et qui accepte le même langage.

✎ Montrer que le langage T n'est pas semi-décidable (c'est-à-dire **n'est pas récursivement énumérable**), avec un raisonnement par l'absurde utilisant le fait qu'un programme peut avoir accès à son propre code source, qui lui est retourné sous la forme d'une chaîne de caractères par l'appel à `MONCODE()`.