TD 02 – M_u, castor affairé, jeu de la vie

Exercice 1. M_u

✎ Choisir un codage des machines de Turing et des entrées des machines de Turing, puis donner le code d'une machine de Turing universelle M_u, c'est-à-dire telle que :

$$\text{pour tous } \langle M \rangle \text{ et } \langle w \rangle \text{ on ait } M_u(\langle M \rangle \# \langle w \rangle) = M(w).$$

Exercice 2. *Le concours du castor affairé*

Cet exercice est basé sur l'article de Tibor Radó, « On Non-Computable Functions », *Bell Systems Technology Journal*, vol. 41, no 3, mai 1962, p. 877-884.

<u>Concours du castor affairé</u>

Considérons des machines de Turing sur l'alphabet binaire $\Gamma = \{1, B\}$ dont la fonction de transition est $\delta : Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$. Ces machines n'ont pas d'état final, elles s'arrêtent uniquement lorsqu'une transition est indéfinie. Dans cet exercice, les nombres seront représentés en unaire sur le ruban.

Voici les règles du concours du castor affairé.

(a) Le participant sélectionne un entier $n \in \mathbb{N}$, et construit sa propre machine de Turing binaire M à n états.

(b) Il lance sa machine M dans son état initial sur un ruban vide (uniquement des symboles B), et annonce qu'elle s'arrête après s étapes.

(c) Il soumet son entrée (M, s) à un membre du *International Busy Beaver Club*.

(d) Le membre vérifie que la machine M s'arrête après exactement s étapes. Cette vérification est décidable, il suffit de lancer la machine pour s étapes : si la machine ne s'est pas arrêtée après s étapes alors elle est rejetée, et si elle s'arrête en moins de s étapes alors elle est retournée au participant pour correction. Après qu'une machine ait été vérifiée, son score est le nombre de 1 écrits sur le ruban lorsqu'elle s'arrête.

Le champion BB-n est le participant qui obtient le plus grand score avec une machine à n états.

1. Quel score a obtenu le champion BB-2 ?

2. Sauriez-vous obtenir le score 6 dans la catégorie BB-3 ?

3. Comment prouver que 6 est le meilleur score possible de la catégorie BB-3 ?

Le problème du castor affairé consiste à déterminer le plus grand score possible pour BB-n. Soient

— $N(n)$ le nombre de machines de Turing à n états,
— E_n l'ensemble des entrées (M, s) valides pour BB-n,
— $N_e(n)$ la taille de E_n.

4. Que vaut $N(n)$?

5. Argumenter que $0 < N_e(n) < N(n)$.

6. Etant donnée une entrée (M, s), peut-on décider si $(M, s) \in E_n$?

L'ensemble E_n possède une définition claire et précise, est non-vide et fini, pour tout n. Nous verrons à la fin de cet exercice que pourtant, $N_e(n)$, le nombre d'éléments de E_n, n'est pas une fonction calculable.

Croissance de $\Sigma(n)$
Chaque entrée valide (M, s) possède un score bien défini $\sigma(M, s)$. Soit

$$\Sigma(n) = \max\big\{\sigma(M, s) \mid (M, s) \in E_n\big\}.$$

$\Sigma(n)$ est la fonction du castor affairé, et nous allons voir qu'elle n'est pas calculable. Attention cependant, il est tout a fait possible d'en calculer certaines valeurs, comme $\Sigma(2) = 4$.

7. Regarder sur wikipedia les valeurs de $\Sigma(4)$, $\Sigma(5)$, $\Sigma(6)$.

Soient $f(x)$ et $g(x)$ deux fonctions de $\mathbb{N}$ dans $\mathbb{N}$, nous écrirons

$$f(x) \succ g(x) \text{ si et seulement si } \exists x_0 : \forall x > x_0 : f(x) > g(x).$$

Nous allons maintenant démontrer le théorème suivant :

Théorème. Pour toute fonction calculable $f : \mathbb{N} \to \mathbb{N}$, on a $\Sigma(n) \succ f(n)$.
Donc $\Sigma(n)$ n'est pas calculable.

Démonstration. Soit $f(x)$ une fonction calculable, et soit la fonction $F(x) = \sum_{i=0}^{x} \big(f(i) + i^2\big)$.

8. Argumenter que $F(x)$ est calculable par une machine de Turing.

Soit M_F une telle machine et c son nombre d'états. On considère que M_F, placée sur le premier symbole blanc à droite d'une entrée x (en unaire), s'arrête dans la configuration dont la description instantanée est $xBF(x)qB$ pour un certain état q.

9. Est-ce que $F(x) \geq f(x)$ pour tout x ?

10. Est-ce que $F(x) \geq x^2$ pour tout x ?

11. Est-ce que $F(x + 1) > F(x)$ pour tout x ?

12. Donner la fonction de transition d'une machine de Turing M_x à $x + 1$ états qui, à partir d'une entrée vide, écrit x symboles 1 consécutifs sur le ruban de façon à être composée avec M_F.

Soit la machine M_{FFx} qui correspond à la composition $M_F \circ M_F \circ M_x$.

13. Combien d'états possède la machine M_{FFx} ?

14. Que peut-on en déduire sur Σ(réponse à la question précédente) ?

15. Démonter que $F(F(x)) \succ F(1 + x + 2c)$.

16. En déduire que $\Sigma(1 + x + 2c) \succ F(1 + x + 2c)$.

17. Conclure la preuve.

Fonction $S(n)$
Remarquons que E_n coïncide avec l'ensemble des machines de Turing à n états qui s'arrêtent sur l'entrée vide. Soit $S(n) = \max\big\{s \mid (M, s) \in E_n\big\}$.

18. La fonction $S(n)$ est-elle calculable ?

Fonction $N_e(n)$
Rappelons que $N_e(n) = |E_n|$. Soit $N(s,n) = \big|\{(M,s') \in E_n \mid s' = s\}\big|$.

19. La fonction $N(s,n)$ est-elle calculable ?

Soient $G(s,n) = \sum\limits_{i=1}^{s} N(i,n)$ et $\Phi(s,n) = N_e(n) - G(s,n)$.

20. Exprimer avec des mots la fonction $G(n,s)$.

21. Que peut-on dire de $\Phi(s,n)$?

22. Quelle est la valeur minimale de s telle que $\Phi(s,n) = 0$?

23. En déduire que $N_e(n)$ n'est pas calculable.

Remarque
Supposons que pour un entier n_0 nous arrivons à déterminer $N_e(n_0)$.

24. Peut-on alors déterminer $S(n_0)$ et $\Sigma(n_0)$?

25. Montrer que $S(n) \leq \Sigma(20n)$ (ou un peu plus que 20 si vous voulez).

Exercice 3. *Automate cellulaire : le jeu de la vie*

Un *automate cellulaire* en dimension d sur l'alphabet fini A est une fonction $F : A^{\mathbb{Z}^d} \to A^{\mathbb{Z}^d}$ définie par un voisinage fini $U \subset \mathbb{Z}^d$ et une fonction locale $f : A^U \to A$ par

$$\forall i \in \mathbb{Z}^d : F(x)(i) = f(x_{i+U}),$$

où x_{i+U} est la fonction $j \in U \mapsto x(i+j)$. Un automate cellulaire est un système dynamique discret (en espace et en temps) sur l'ensemble de configurations $A^{\mathbb{Z}^d}$ (une configuration associe à chaque cellule de $\mathbb{Z}^d$ une lettre de A. Un automate cellulaire est défini par un triplet (A, U, f) (la dynamique est définie par la fonction locale f qui permet de calculer l'image de chaque cellule par F).
Le *jeu de la vie* (*game of life*) est un exemple très fameux d'automate cellulaire, défini par Conway dans les années 1970. Il est défini en deux dimensions sur l'alphabet $\{0,1\}$ avec le voisinage de Moore (9 voisins) $U = \bigcup\{(i,j) \mid i,j \in \{-1,0,1\} \text{ et } |i| + |j| \leq 2\}$ par la fonction locale

$$f : x_U \mapsto \begin{cases} 1 \text{ si } x_{(0,0)} = 0 \text{ et } \sum\limits_{i \in U} x_i = 3 \\ 1 \text{ si } x_{(0,0)} = 1 \text{ et } 3 \leq \sum\limits_{i \in U} x_i \leq 4 \\ 0 \text{ sinon.} \end{cases}$$

Avec des mots, on dit qu'une cellule est *vivante* si son état est 1 et *morte* si son état est 0. Alors une cellule morte peu naître si elle a exactement 3 voisines vivantes (il faut 3 parents pour naître), et une cellule vivante reste vivante si elle a entre 2 et 3 voisines vivantes (trop peu et elle meurt d'isolation, trop et elle meurt de surpopulation). Une configuration est *finie* si elle contient un nombre fini de cellules vivantes, et on appelle *motif* la projection/restriction d'une configuration sur un support fini.
On peut voir le jeu de la vie comme un modèle de calcul, et considérer les problèmes :

- étant donnés deux motifs A et B, le motif B va-t-il apparaître au cours de l'évolution depuis A ?
- étant donnée une configuration initiale finie, toutes les cellules vont-elles mourir ?

— étant donnée une configuration initiale finie et une cellule morte, cette cellule va-t-elle naître au cours de l'évolution ?

qui sont tous indécidables.

Pour jouer :

— simulateur du jeu de la vie : `http://golly.sourceforge.net/`,
— bibliothèque de motifs : `http://conwaylife.com/wiki/Main_Page`.