TD 05 – Réels, Castor affairé, FO : non calculables !

Exercice 1. *Calculabilité des nombres réels*

On dit qu'un nombre réel est calculable si et seulement si il existe une machine de Turing qui énumère ses digits (en base 2 pour simplifier). Quelques exemples de nombres réels calculables : 0, $\frac{1}{3}$, $\sqrt{2}$, π, $\frac{1+\sqrt{5}}{2}$.

1. Donner la partie entière et les trois premiers bits après la virgule de chacun de ces nombres.
2. Est-ce qu'il y a une différence entre 1 et $0.111111\ldots$?
3. Est-ce que tous les nombres réels sont calculables ?

Soit

$$h_i = \begin{cases} 1 & \text{si la MT numéro } i \text{ s'arrête sur l'entrée vide} \\ 0 & \text{sinon} \end{cases} .$$

Voici une variante du nombre Ω de Chaitin :

$$\Omega = 0.h_0h_1h_2h_3h_4h_5h_6h_7h_8h_9\ldots$$

4. Le nombre Ω est-il calculable (justifier) ?

Soit

$$x_i(t) = \begin{cases} 1 & \text{si la MT numéro } i \text{ s'arrête sur l'entrée vide en moins de } t \text{ étapes} \\ 0 & \text{sinon} \end{cases} ,$$

et soit la suite :

$$(X_n)_{n>0} = 0.x_1(n)x_2(n)x_3(n)x_4(n)x_5(n)x_6(n)x_7(n)x_8(n)\ldots$$

5. Cette suite est-elle calculable (c'est-à-dire, chaque X_n est-il calculable) ?
6. Cette suite est-elle croissante ?
7. Cette suite est-elle bornée supérieurement ?
8. La limite supérieure (plus petite borne supérieure) de X_i est-elle calculable ?

Exercice 2. *Entraînement pour l'examen*

Montrer que les deux langages suivants ne sont pas récursivement énumérables, sans utiliser le théorème de Rice.

1. $\{\langle M\rangle \mid M$ s'arrête sur toutes les entrées sauf $\epsilon\}$.
2. $\left\{\langle M_1\rangle\#\langle M_2\rangle \mid L(M_1) \cap \{0,1\}^{42} = L(M_2) \cap \{0,1\}^{42}\right\}$.

Montrer que le langage suivant est récursivement énumérable.

3. $\{\langle M\rangle \mid M$ s'arrête sur toutes les entrées de taille au plus $1000\}$.

Montrer que les deux langages suivants ne sont pas récursifs, en utilisant le théorème de Rice.

4. $\{\langle M \rangle \mid (a^* + b^*) \subseteq L(M)\}$.

5. $\{\langle M \rangle \mid M$ ne s'arrête sur aucune entrée$\}$

Montrer que le langage suivant est récursif.

6. $\{w \mid \forall M : w \notin L(M)\}$.

Exercice 3. *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.

Concours du castor affairé

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

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

$\Sigma(n)$ 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.

<u>**Fonction** $S(n)$</u>
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 ?

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

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

Soient $G(s, n) = \sum_{i=1}^{s} N(i, n)$ 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 4. *La logique du premier ordre est indécidable*

Le but de cet exercice est de démontrer le résultat suivant :

Théorème. Le problème de savoir si $\Gamma \models \phi$ dans la logique du premier ordre est indécidable.

Via une réduction depuis L_u.

1. Quel est le type de l'entrée de notre réduction, quel est le type de la sortie, et quelle est la relation ($\iff$) qui doit exister entre l'entrée et la sortie ?

Soit $M = (Q^M, \Sigma^M, \Gamma^M, \delta^M, q_0^M, B^M, q_F^M)$ et w une instance du problème d'appartenance à L_u. Nous définissons le langage $S = (S_f, S_r)$ avec $S_f = \{(\epsilon, 0)\} \cup \{(a, 1) \mid a \in \Gamma^M\}$ et $S_r = \{(f_q, 2) \mid q \in Q^M\}$. Voici l'idée que nous allons suivre : les variables x correspondront à des mots sur Γ^M, ϵ sera utilisé pour le mot vide, $a(w)$ sera le mot aw, et $f_q(x, y)$ signifiera que M, sur l'entrée w, peut atteindre la configuration $\bar{x}qy$ (avec $\bar{x}$ le mot miroir de x).

2. Quel terme représente le mot $abba$?

3. Quelle formule représente le fait que la machine M atteint l'état q avec la tête de lecture sur la première lettre du mot $abba$?

4. Quelle formule représente la configuration initiale de M ? Doit-on l'ajouter à Γ ?

5. Quel est le but ϕ ?

Nous allons maintenant ajouter des formules dans Γ, de façon à ce que la formule correspondant à la configuration $uqav$ soit vraie si et seulement si $\iota_w \vdash^* uqav$.

6. Supposons que M possède la transition suivante : $\delta^M(q_0^M, a) = (q_1^M, b, R)$. Quelle formule faut-il ajouter pour que l'on puisse appliquer la règle de résolution afin d'en déduire que la formule représentant la configuration $bq_1^M w_1 w'$ (avec $w = w_0 w_1 w_2 \dots w_n$ et $w' = w_2 \dots w_n$) est conséquence logique de Γ si $w_0 = a$?

7. Etant donnée M, donner l'ensemble des règles à ajouter pour encoder toutes ses transitions.

Il faut également rajouter les règles suivantes à Γ pour permettre d'écrire sur les symboles initialement blancs :

pour tout $q \in Q^M$ on ajoute la formule $\forall x : f_q(x, \epsilon) \iff f_q(x, B)$
pour tout $q \in Q^M$ on ajoute la formule $\forall y : f_q(\epsilon, y) \iff f_q(B, \epsilon)$

8. Donner Γ, ϕ et les étapes du calcul de la résolution qui permettent de conclure que la machine suivante atteint l'état $bq_F^M bb$ sur l'entrée ab :

$$\delta^M(q_0^M, a) = (q_1^M, b, R) \qquad \delta^M(q_1^M, b) = (q_2^M, b, R) \qquad \delta^M(q_2^M, B) = (q_F^M, b, L)$$

9. Argumenter le fait que cette réduction est correcte (c'est-à-dire que la relation attendue entre l'entrée et la sortie est bien vérifiée).