Examen partiel, L3 Calculabilité

13 novembre 2014

Durée: 3h. Tous les documents papier sont autorisés. Les ordinateurs sont interdits. Les résultats vus en cours (même non démontrés en cours) peuvent être utilisés avec une simple référence. Les autres résultats vus en TD doivent être redémontrés. Il est conseillé de ne traiter la dernière question qu'après toutes les autres, car sa solution est significativement plus longue et difficile que toutes les autres.

Dans les 9 cas suivants dire (en le justifiant) si le probème est décidable.

De plus, pour les 5 premiers problèmes, dire (en le justifiant) si l'ensemble des données pour lesquelles la réponse à la question est oui est un ensemble récursivement énumérable, co-récursivement énumérable, les deux, ou aucun des deux.

1. **Donnée:** Une machine de Turing M

   Question: il existe w_1, w_2 tels que ((M s'arrête sur w_1) ssi (M s'arrête sur w_2))

2. **Donnée:** Une machine de Turing M

   Question: pour tous w_1, w_2, ((M s'arrête sur w_1) ssi (M s'arrête sur w_2))

3. **Donnée:** deux machines de Turing M_1, M_2 et un mot w

   Question: $w \notin L(M_1) \cap L(M_2)$

4. **Donnée:** Deux suites finies de mots $u_1, \ldots, u_n$, $v_1, \ldots, v_n$ et un entier k

   Question: il existe une suite d'incides $i_1, \ldots, i_k$ telle que $u_{i_1} \cdots u_{i_k} = v_{i_1} \cdots v_{i_k}$

5. **Donnée:** Deux suites finies de mots $u_1, \ldots, u_n$, $v_1, \ldots, v_n$

   Question: il existe un entier k et il existe une suite d'incides $i_1, \ldots, i_k$ telle que $u_{i_1} \cdots u_{i_k} = v_{i_1} \cdots v_{i_k}$

6. **Donnée:** Une machine de Turing M qui s'arrête sur toute entrée

   Question: $L(M)$ est fini.

7. **Donnée:** deux machines de Turing M_1, M_2 qui s'arrêtent sur toutes les entrées

   Question: pour tout w, le temps de calcul de M_1 sur w est inférieur ou égal au temps de calcul de M_2 sur w

8. **Donnée:** une fonction récursive primitive f à un argument

   Question il existe un entier n tel que $f(n) = 0$

9. Si $S = \frac{p_1}{q_1}, \dots \frac{p_k}{q_k} \in \mathbb{Q}$ est une séquence finie de fractions positives (par nécessairement irréductibles), et $m \in \mathbb{N}$, on définit la suite $u_n^{S,m}$ par récurrence: $u_0^{S,m} = m$ et

   - si aucun aucun des entiers $q_1, \dots q_k$ ne divise $u_n^{S,m}$ alors $u_{n+1}^{S,m}$ est indéfini (la suite s'arrête)
   - sinon, $u_{n+1}^{S,m} = u_n^{S,m} \times \frac{p_i}{q_i}$ où i est le plus petit indice tel que q_i divise $u_n^{S,m}$

   Donnée: Une suite de fractions positives S et un entier m

   Question: La suite $u_n^{S,m}$ est infinie.

   Pour cette question, on admettra le lemme suivant:

   Lemme:

   (a) Etant donné $N \in \mathbb{N}$, Il existe une suite S_{shift} de 4 fractions telles que, pour tout α, $u_n^{S_{\mathsf{shift}}, 3\times 2^{\alpha} \times 7^{\beta}}$ est finie et la dernière valeur est $5 \times 2^{\alpha \times N + \beta}$. (On peut ici remplacer $2, 3, 5, 7$ par des nombres entiers supérieurs à 2 et premiers entre eux deux à deux).

   (b) Étant donné $N \in \mathbb{N}$, il existe une suite S_{div} de 4 fractions telles que, pour tout α, $u_n^{S_{\mathsf{div}}, 3\times 2^{\alpha}}$ est finie et la dernière valeur est $5 \times 7^r \times 2^q$ où q et r sont respectivement le quotient et le reste de la division euclidienne de α par N (On peut remplacer $2, 3, 5, 7$ par 4 nombres entiers supérieurs à 2 et premiers entre eux deux à deux)

   Question bonus (hors barême): prouver le lemme.

Solution

1. Ce problème est décidable (et donc récursivement énumérable et co-récursivement énumérable): la réponse est toujours oui, en choisissant $w_1 = w_2$. Il suffit donc de tester si la donnée est le code d'une machine de Turing.

   Taux de réussite: 84%

2. La question est équivalente à "M s'arrête sur toute entrée ou M ne s'arrête sur aucune entrée".

   Le langage n'est pas co-récursivement énumérable (et donc le prblème n'est pas décidable) par réduction du problème de l'arrêt: à partir de M, w on construit un mot $w_2 \neq w$ et une machine M' tels que

   (a) M' commence par tester si son entrée est w_2 et, si c'est le cas, elle s'arrête

   (b) Sinon, elle simule M sur w.

   M' s'arrête toujours sur w_2 et s'arrête sur $w_1 \neq w_2$ ssi M s'arrête sur w. Donc "M' s'arrête sur toute entrée ou M' ne s'arrête sur aucune entrée"ssi M' s'arrête sur toute entrée ssi M s'arrête sur w.

   Le langage n'est pas non plus récursivement énumérab;e, par réduction du complémetaire du probème de l'arrêt: à partir de M, w on construit un mot $w_2 \neq w$ et une machine M' tels que

   (a) M' commence par tester si son entrée est w_2 et, si c'est le cas, elle boucle

   (b) Sinon, elle simule M sur w.

   "M' s'arrête sur w_1 ssi M' s'arrête sur w_2" ssi M' ne s'arrête pas sur w_1 ssi M ne s'arrête pas sur w.

   Taux de réussite: 61%

3. Ce n'est pas récursivement énumérable, par réduction du non arrêt de M sur w:on choisit pour M_1 une machine qui accepte sans aucune transition ($L(M_1) = \Sigma^*$). M_2 est la machine M qui, quand celle-ci va s'arrêter, accepte. M ne s'arrête pas sur w ssi $w \notin L(M_2) = L(M_1) \cap L(M_2)$.

   Le langage est co-récursivement énumérable: La machine M simule M_1 sur w sur son premier ruban et M_2 sur w sur son deuxième ruban. Elle accepte quand les deux simulations se sont arrêtées en acceptant: $w \in L(M_1) \cap L(M_2)$ ssi $(M_1, M_2, w) \in L(M)$.

   Taux de récussite: 64%

4. C'est décidable (et donc récursivement énumérable et co-récursivement énumérable): il suffit d'énumérer toutes kes séquences $i_1, \ldots, i_k$ (en nombre borné par n^k) et de tester l'égalité pour chacune d'elles.

   Par exemple, on maintient sur un ruban annexe un indice i compris entre 1 et n^k, sur un autre ruban annexe la séquence $j_1, ...j_k$ d'indices dans $[1..n]$ qui est la $i$ème dans l'ordre lexicographique (initialement 1....1), et sur deux autres rubans, les séquences $u_{j_1} \cdots u_{j_k}$ et $v_{j_1} \cdots v_{j_k}$.

   La machine, pour i de 1 à n^k:

(a) construit, à partir de la séquence d'indices $j_1, \cdots j_k$ du ruban 3, les séquences $u_{j_1} \cdots u_{j_k}$ et $v_{j_1} \cdots v_{j_k}$ et teste leur égalité (Si c'est le cas, la machine s'arrête en acceptant)

(b) incrémente i

(c) calcule le successeur dans l'ordre lexicographique du contenu du ruban 3.

NB: il était inutile de détailler la machine.

Taux de réussite: 76%

5. C'est le problème de correspondance de Post, donc indécidable d'après le cours. Le langage est en revanche récursivement énumérable; il suffit pour $k := 1$ à $+\infty$ d'appliquer l'algorithme de la question précédente.

   Comme le problème est indécidable et récursivement énumérable, le langage n'est pas co-récursivement énumérable.

   Taux de réussite: 78%

6. C'est indécidable par réduction du complémentaire du problème de l'arrêt: Soit (M, w) une instance du problème de l'arrêt, on considère la machine M' qui, sur la donnée x, simule $|x|$ étapes du calcul de M sur w. Si M s'arrête en moins de $|x|$ étapes, alors M' accepte.

   M' s'arrête toujours sur la donnée x (en temps linéaire). De plus, si M ne s'arrête pas sur w, alors $L(M') = \emptyset$ et est donc fini. R'eciproquement, si M s'arrête sur w en N étapes, alors $L(M') = \{x \in \Sigma^* | \, |x| \geq N\}$ est infini.

   Et $L(M')$ est fini (en fait, vide) ssi M ne s'arrête pas sur w.

   NB: attention, on ne peut pas appliquer le théorème de Rice, qui ne s'applique qu'aux langages récursivement énumérables, et pas aux langages récursifs comme c'est le cas ici.

   Taux de réussite: 30%

7. C'est indécidable (pas semi-décidable) par réduction du complémentaire du problème de l'arrêt: M_1 et M_2, sur la donnée x, simulent $|x|$ étapes du calcul de M sur w (typiquement, en écrivant sur le ruban, à la suite de x, la configuration initiale de M sur w, puis en répétant une étape de calcul de M et retirer un symbole à x). Si M s'arrête en moins de $|x|$ étapes, alors M_2 s'arrête et M_1 effectue une étape inutile, puis s'arrête.

   Les deux machines M_1, M_2 s'arrêtent sur toutes données.

   Si M ne s'arrête pas sur w, alors les calculs de M_1 et M_2 sont identiques sur toutes les données et donc le temps de calcul de M_1 sur x est inférieur où égal au temps de calcul de M_2 sur x.

   Si M s'arrête sur w en N étapes, alors, pour une donnée de longueur N, M_1 effectue une étape de plus que M_2: le temps de calcul de M_1 est supérieur strictement à celui de M_2 sur cette donnée.

   En conclusion: M ne s'arrête pas sur w ssi pour tout x, le temps de calcul de M_1 sur x est inférieur ou égal au temps de calcul de M_2 sur x.

   Taux de réussite: 55%

8. C'est indécidable. Par réduction du problème de l'arrêt: étant donnés M, w, la fonction f associe à n (le codage de) la $n$ième configuration de la machine de Turing M sur w, ou 0 si la machine s'arrête avant la $n$ième étape. f est récursive primitive, comme vu en cours. M ne s'arrête pas sur w ssi f ne s'annule pas.

   Taux de réussite: 60%

9. C'est indécidable par réduction du complémentaire du problème de l'arrêt; Soit M une machine de Turing et w_0 un mot. On suppose sans perte de généralité que les états sont $1, ..., |Q|$ et les symboles de l'alphabet Σ sont les entiers de 1 à $|\Sigma| = N$.

   Si $\gamma = (q, w, aw')$ est une configuration de la machine de Turing, on note $\overline{\gamma}$ l'entier $p_{q+20} \times 2^{c(w)} \times 3^{c'(aw')} \times 5$ où p_n est le $n$ième nombre premier, $c'(w')$ est le mot w' considéré comme un entier en base $N+1$ et $c(w)$ est l'image miroir du mot w, considéré comme un entier en base $N+1$.

   Les étapes principales consistent à donner des suites de fractions qui permettent successivement d'atteindre à partir de $\overline{\gamma}$:

   (a) $p_{q+20} \times 2^{c(w)} \times 3^{c'(w')} \times 7^a \times 11$

   (b) $p_{q'+20} \times 2^{c(w)} \times 3^{c'(w')} \times 7^b \times p_d \times 23$ où
      - $p_d = 13$ si $d = \rightarrow$, $p_d = 17$ si $d = \leftarrow$ et $p_d = 19$ si $d = \downarrow$.
      - $\delta(q, a) = (q', b, d)$

   (c) $p_{q'+20} \times 2^{c(wb)} \times 3^{c'(w')} \times p_d \times 29$ si $d = \rightarrow$,
   $p_{q'+20} \times 2^{c(w'')} \times 3^{c'(a'bw')} \times p_d \times 29$ si $d = \leftarrow$ et $w' = w''a'$
   $p_{q'+20} \times 2^{c(w')} \times 3^{c'(bw')} \times p_d \times 29$ si $d = \downarrow$

   (d) $\overline{\gamma'}$ si $\gamma \vdash_M \gamma'$.

   On procède comme suit pour chacune des étapes (les séquences de fractions sont données dans cet ordre):

   (a) Utiliser S_{div} (avec les changements de nombres premiers appropriés)

   (b) Si $1, ..., |\Sigma|$ sont les lettres de Σ, on note $\delta(q, i) = (q_i, b_i, d_i)$ et on utilise la séquence:

   $$\frac{p_{q_N+20} \times 7^{b_N} \times p_{d_N} \times 23}{p_{q+20} \times 7^N \times 11}; \quad \cdots \quad \frac{p_{q_1+20} \times 7^{b_1} \times p_{d_1} \times 23}{p_{q+20} \times 7 \times 11}$$

   C'est important de les donner dans l'ordre décroissant des puissances de 7, pour être sûr d'appliquer la transition de i et pas celle de $j < i$.

   (c) On utilise 3 fois $S_{\mathsf{shift}}, S_{\mathsf{div}}$ avec des changement de nombres premiers appropriés.

   (d) On termine par la séquence

   $$\frac{15}{3 \times 13 \times 29}; \quad \frac{15}{3 \times 17 \times 29}; \quad \frac{15}{3 \times 19 \times 29}; \quad \frac{5 \times 3^B}{13 \times 29}$$

   qui permet d'obtenir $\overline{\gamma'}$ et de s'assurer que le mot w' n'est pas vide: c'est seulement dans le cas où 3 ne divise pas le résultat que la dernière fraction s'applique. Elle rajoute un blanc en fin de ruban.

La suite finie de fractions ainsi construite est donc telle que, si $u_n^{S,m} = \overline{\gamma}$ et que $\gamma \vdash_M \gamma'$, alors $u_{n+1}^{S,m} = \overline{\gamma'}$. Réciproquement, si γ est une configuration finale, comme tous les dénominateurs des séquences (b,c,d) ont un facteur premier distinct de 2,3,7,11, la séquence $u_n^{S,m}$ ne peut se prolonger après l'étape (a).

En conclusion, si $m = \overline{\gamma_0}$, la séquence $u_n^{S,\overline{\gamma_0}}$ est finie ssi la machine s'arrête sur w. Le problème est donc indécidable.

Taux de réussite: 0%

Question bonus:

1. On considère la suite de 4 fractions:

$$\frac{10}{55}; \quad \frac{3 \times 11^N}{6}; \quad \frac{33}{21}; \quad \frac{5}{3}$$

$$\begin{array}{rlcl}
3 \times 2^\alpha \times 7^\beta & \to^* \; (\frac{3 \times 11^N}{6})^\alpha \times 3 \times 2^\alpha \times 7^\beta & = & 11^{N \times \alpha} \times 3 \times 7^\beta \\
& \to \; (\frac{33}{21})^\beta \times 3 \times 11^{N\alpha} \times 7^\beta & = & 3 \times 11^{N\alpha+\beta} \\
& \to \; \frac{5}{3} \times 3 \times 11^{N\alpha+\beta} & = & 5 \times 11^{N\alpha+\beta} \\
& \to^* \; (\frac{10}{55})^{N\alpha+\beta} \times 5 \times 11^{N\alpha+\beta} & = & 5 \times 2^{N\alpha+\beta}
\end{array}$$

2.

$$\frac{33}{3 \times 2^N}; \quad \frac{21}{6}; \quad \frac{5}{3}; \quad \frac{2}{11}$$

$$\begin{array}{rlllll}
3 \times 2^\alpha & \to^* & 3 \times 2^r \times 11^q & \to^* \; 3 \times 7^r \times 11^q & \to & 5 \times 7^r \times 11^q \\
& \to^* & 5 \times 7^r \times 2^q & & &
\end{array}$$