Université Bordeaux 1
Mastère d'Informatique 1, 2008/2009

Modèles de Calcul : Corrigé du CC 2

Réductions-Simulations

PB1 : Donnée : $n \in \mathbb{N}$ et $w \in \{a, b\}^*$; Question : $w \in \mathrm{L}(M_n)$?
Réponse : PB1 est indécidable
Justification 1 : cela découle du théorème de Rice :
fixons $w = \varepsilon$ (par exemple).
- la propriété à tester ne dépend que du langage $\mathrm{L}(M_n)$
- la propriété est non-triviale :
  – le langage $\{a\}$ est récursivement énumérable et ne vérifie pas la propriété.
  – le langage $\{\varepsilon\}$ est récursivement énumérable et vérifie la propriété.

PB2 : Donnée : $n \in \mathbb{N}$; Question : $\mathrm{L}(M_n) = \emptyset$?
Réponse : PB2 est indécidable
Justification : cela découle du théorème de Rice :
- la propriété à tester ne dépend que du langage $\mathrm{L}(M_n)$
- la propriété est non-triviale :
  – le langage $\emptyset$ est récursivement énumérable et vérifie la propriété.
  – le langage $\{a\}$ est récursivement énumérable et ne vérifie pas la propriété.

PB3 : Donnée : $n \in \mathbb{N}$; Question : Le calcul de la machine M_n, partant de son état initial et d'une bande entièrement blanche, utilise-t-il au moins une transition qui déplace la tête de lecture vers la *gauche* ?
Réponse : PB3 est décidable
Justification : La sous-machine de Turing $D(M_n)$ obtenue à partir de M_n en lui ôtant toutes ses transitions gauches, peut être simulée par un automate fini. Comme le problème de l' accessibilité d' un état à partir de l'état initial est décidable pour les automates finis, on peut décider si $D(M_n)$, partant de la bande entièrement blanche, atteint l' état de départ d' une transition gauche lisant la lettre $\square$. Si oui, alors la réponse au PB 3 est oui, sinon, la réponse au PB 3 est non.

PB4 : Donnée : $n \in \mathbb{N}$; Question : Le calcul de la machine M_n, partant de son état initial et d'une bande entièrement blanche, utilise-t-il *toutes* les transitions qui déplacent la tête de lecture vers la *gauche* ?
Réponse :PB4 est indécidable
Justification :
Elle est un peu longue (donc vaut 3 points au lieu du maigre point initialement annoncé).
Soit $M = \langle Q, q_-, F, \Sigma, \Gamma, \delta \rangle$ une machine de Turing déterministe sur $\{a, b\}$. On fait deux hypothèse techniques : (H1) M ne peut s' arrêter que dans des états de F et aucune transition ne part d'un état de F.
(H2) M ne peut insérer un blanc entre des lettres non-blanches.
On note $(q_i, x_i, r_i, y_i, \leftarrow)_{1 \leq i \leq p}$ une liste de toutes les transitions gauches de M. On construit une nouvelle machine

$$M' := \langle Q \cup Q' \cup \{\perp, \top\}, q'_1, F, \Sigma, \Gamma, \delta \cup \delta' \rangle$$

comme suit.
L'ensemble Q' est formé de $3(p+1)$ nouveaux états tous distincts entre eux

$$Q' := \bigcup_{1 \leq i \leq p+1} \{q'_i, q''_i, q'''_i\}$$

L'ensemble δ' est formé des nouvelles transitions

$$(q'_i, \Box, q''_i, x_i, \leftarrow), (q''_i, \Box, q_i, \Box, \rightarrow), (r_i, \Box, q'''_i, \Box, \rightarrow), (q'''_i, y_i, q'_{i+1}, \Box, \leftarrow)$$

où $i \in [1, p]$, de la transition

$$(q'_{p+1}, \Box, q_-, \Box, \rightarrow)$$

et des transitions

$$(q, \gamma, \bot, \gamma, \rightarrow), (\bot, x, \bot, x, \rightarrow), (\bot, \Box, \top, \Box, \leftarrow)$$

où $q \in F, \gamma \in \Gamma, x \in \Gamma \setminus \{\Box\}$.
M' se comporte de la façon suivante (en partant de ε) :
Phase 1 : M' commence par enchaîner toutes les transitions gauches de M (combinées avec des nouvelles transitions) :

$$q'_1\Box \vdash q''_1\Box x_1 \vdash \Box q_1 x_1 \vdash r_1\Box y_1 \vdash \Box q'''_1 y_1 \vdash q'_2\Box$$

puis

$$q'_2\Box \vdash q''_2\Box x_2 \vdash \Box q_2 x_2 \vdash r_2\Box y_2 \vdash \Box q'''_2 y_2 \vdash q'_3\Box$$

et ainsi de suite, pour tous les indices $i \in [1, p-1]$, puis

$$q'_p\Box \vdash q''_p\Box x_p \vdash \Box q_p x_p \vdash r_p\Box y_p \vdash \Box q'''_p y_p \vdash q'_{p+1}\Box \vdash q_-\Box$$

Phase 2 : M' exécute le calcul de M
Phase 3 : si le calcul de M ne s' arrête pas, alors la phase 2 ne s' arrête pas ; le calcul de M' n' utilise donc pas la transition gauche $(\bot, \Box, \top, \Box, \leftarrow)$
si le calcul de M s' arrête, alors la phase 2 s' arrête dans une configuration $uqxv$ avec $q \in F, u \in \Gamma^*, x \in \Gamma, v \in (\Gamma \setminus \{\Box\})^*$; M' exécute alors

$$uqxv \vdash ux\bot v \vdash^* uxv\bot\Box \vdash uxv'\top x'$$

où $v = v'x'$. Donc le calcul de M' utilise *toutes* les transitions gauches de M'.
Finalement : (M s' arrête sur la bande vide) ssi (le PB4 a la réponse oui sur M').
Comme le PB de l' arrêt sur la bande vide est indécidable, PB4 est indécidable.
PB5 : Donnée : un automate à un compteur, déterministe, A, qui lit des mots de $\{a, b\}^*$ et $w \in \{a, b\}^*$;Question : $w \in \mathrm{L}(A)$?
Réponse :PB5 est décidable
Justification : Tout automate à un compteur est aussi un automate à pile. Or le problème de l' appartenance au langage reconnu par un automate à pile est décidable (voir le cours d' INFOT2).
PB6 : Donnée : un automate à deux compteurs, déterministe, A, qui lit des mots de $\{a, b\}^*$ et un mot $w \in \{a, b\}^*$; Question : $w \in \mathrm{L}(A)$?
Réponse : PB6 est indécidable

Justification : Pour toute machine de Turing M, on peut construire un automate à 2 compteurs A qui simule M. Notons $\rho(M)$ l'automate ainsi construit à partir de M. Le problème " $w \in \mathrm{L}(M)$?" se réduit donc au PB6 sur l'automate $A = \rho(M)$. L'application ρ est donc une réduction du PB1 au PB6. Comme le PB1 est indécidable, le PB6 est indécidable.
PB7 : Donnée : une grammaire algébrique $G = \langle \{a, b\}, V, P, \sigma \rangle$; Question : La grammaire G est-elle ambiguë ?
Réponse :PB7 est indécidable.
Justification : On peut réduire le Problème de correspondance de Post au PB7 (vu en cours et en TD).
PB8 : Donnée : une grammaire algébrique $G = \langle \{a, b\}, V, P, \sigma \rangle$.
Question : $\forall v \in V, \forall w \in \mathrm{L}_G(v), \exists T, T'$, arbres de dérivation de racine v, tels que $T \neq T'$ et $fr(T) = fr(T') = w$?
Réponse : PB8 est décidable.
Justification : Considérons une donnée $G = \langle \{a, b\}, V, P, \sigma \rangle$. 1- Pour que la réponse soit oui, il *faut* que : pour toute règle terminale $(v, w) \in P \mid w \in \{a, b\}^*$, il existe deux arbres de dérivation T, T', de racine v, tels que $T \neq T'$ et $fr(T) = fr(T') = w$. On en connaît déjà un : celui de hauteur 1 qui correspond à la règle $v \to w$. Appelons T cet arbre. Il suffit de déterminer s'il existe un arbre de dérivation T' de racine étiquetée par v, de frontière w, et différents de T. Notons $\mathcal{T}(v, w)$ l'ensemble des arbres de dérivation T' de racine étiquetée par v et de frontière w. La question à résoudre est donc :

$$\mathcal{T}(v, w) \setminus \{T\} \neq \emptyset ?$$

Mais $\mathcal{T}(v, w)$ est langage rationnel d'arbres , donc $\mathcal{T}(v, w) \setminus \{T\}$ est aussi rationnel, et on peut tester si un langage rationnel d' arbres est vide (voir INFOT2).
2- D'autre part, pour que la reponse soit oui, il *suffit* que : pour toute règle terminale $(v, w) \in P \mid w \in \{a, b\}^*$, il existe deux arbres de dérivation T, T', de racine v, tels que $T \neq T'$ et $fr(T) = fr(T') = w$.
Pour le montrer, supposons que cette condition est vérifiée. Soit T_1 un arbre de dérivation terminal et notons $w_1 := fr(T_1), v_1 := rt(T_1)$ avec $v_1 \in V, w_1 \in \{a, b\}^*$. L'arbre T_1 admet au moins un sous arbre T de hauteur 1, correspondant à une règle (v, w) terminale i.e. où $v \in V, w \in \{a, b\}^*$. Notre supposition est que, il existe un autre arbre de dérivation T', tel que $w = fr(T'), v = rt(T')$. En remplaçant T par T' dans T_1 on obtient un arbre de dérivation T'_1, de racine étiquetée par v_1, tel que $T_1 \neq T'_1$ et $w_1 = fr(T'_1)$.