Décidabilité T.D. N^o 3

3 décembre 2010

La non-calculabilité

1 Le complément du langages L0

Démontrer le Lemme 4 :
Le langage défini par :

$$\bar{L_0} = \{w : w = w_i \wedge M_i \text{ accepte } w_i\}$$

est dans la classe RE.

Corrigé Considérons une machine de Turing qui accepte $\bar{L_0}$. Pour un mot w, elle procède ainsi :

1. Elle énumère tous les mots jusqu'à ce qu'elle trouve l'indice i $w = w_i$;
2. Elle détermine la machine M_i ;
3. Elle simule l'execution de M_i sur w_i et accepte si M_i accepte w_i.

2 Le problème d'arrêt et ses variantes

Démontre que les problèmes suivants sont indécidables :

1. Le problème d'arrêt : H={<M,w>|M s'arrête sur w}.
2. Le problème d'arrêt sur un mot vide.
3. Le problème d'arrêt existentiel.

Corrigé

1. Le problème d'arrêt : H={<M,w>|M s'arrête sur w} : nous appliquons la réduction à partir de LU (langage universelle vu en cours). Soit <M,w> une instance de LU, nous supposons qu'il existe une procédure de décision hypothétique pour H.

(a) Appliquer l'algorithme décidant H à <M,w>

(b) Si H répond non (ca veut dire que la machine M ne s'arrête pas sur w) répondre non ($< M, w > \notin LU$)

(c) Si H répond oui simuler l'execution de M sur w et donner la réponse obtenue.

Donc on arrive à décider LU à partir de la procédure qui décide H, donc contradiction. Il n'existe donc pas une procédure qui décide H.

2. Le problème d'arrêt sur un mot vide : la réduction se fait à partir du problème d'arrêt précédent. Pour une instance <M,w> du problème de l'arrêt, on construit une machine M' :

   (a) elle écrit w sur son ruban d'entrée ;

   (b) elle se comporte ensuite comme M.

   On résout le problème de l'arrêt sur mot vide pour M' et on transmet la réponse obtenue.

3. Le problème d'arrêt existentiel : déterminer si une machine de Turing s'arrêt pour au moins un mot d'entrée. La réduction se fait à partir du problème de l'arrêt sur un mot vide. Pour une instance M du problème de l'arrêt sur un mot vide, on construit une machine de Turing avec le comportement suivant :

   (a) Elle efface le contenu de son ruban d'entrée ;

   (b) elle se comporte ensuite comme M.

   On résout le problème de l'arrêt existentiel pour M' et on transmet la réponse obtenue.

3 Problèmes relatifs aux ensembles récursivement enumerable

Démontrer que les problème suivants sont indécidables :

1. Déterminer si le langage accepté par une machine de Turing est vide.
2. Déterminer si le langage accepté par une machine de Turing est récursif.
3. Déterminer si le langage accepté par une machine de Turing est indécidable.

Corrigé

1. Déterminer si le langage accepté par une machine de Turing est vide. Une instance de ce problème est une machine de Turing M et le problème est de déterminer si $L(M) = \Phi$ (M n'accepte aucun mot). Nous procédons par déduction à partir de $\bar{LU}$ Pour une instance <M,w> de $\bar{LU}$, on construit la machine M' :

   (a) Elle simule l'exécution de M sur w sans tenir compte de son propre mot d'entrée.

   (b) Si M accepte w, elle accepte son mot d'entrée quel qu'il soit : $L(M') = \Sigma^*$ quand $< M, w > \notin \bar{LU}$

   (c) Si M n'accepte pas w, elle n'accepte aucun mot : $L(M') = \Phi$ quand $< M, w > \in \bar{LU}$

On résout le problème du langage accepté vide pour M' et on transmet la réponse obtenue

2. Déterminer si le langage accepté par une machine de Turing est récursif. Une instance de ce problème est une machine de Turing M et le problème est de déterminer si L(M$^\circ$ est récursif. Nous procédons par réduction à partir de $L\bar{U}$. On résout donc $L\bar{U}$ ainsi : pour une instance <M,w> de $L\bar{U}$, on construit M' qui la le comportement suivant :
   - Elle simule l'exécution de M sur w sans tenir compte de son propre entrée x.
   - Si M accepte w ,elle se comporte sur sont mot d'entrée x comme une machine de Turing universelle.
   - Si M n'accepte pas w (le rejette ou a une exécution infinie), elle n'accepte aucun mot.

   On transmet la réponse.
   - $L(M') = \phi$ et est récursif exactement quand M n'accepte pas w à savoir quand $< M, w > \in L\bar{U}$
   - $L(M') = LU$ et n'est pas récursif exactement quand M accepte w à savoir quand $< M, w > \notin L\bar{U}$
3. Déterminer si le langage accepté par une machine de Turing est indécidable.