TD 11

Langages Formels, Calculabilité et Complexité

Exercice 1: Le problème *d'acceptation sur place* pour les machines de Turing, ASP_{TM} consiste à décider si une MT (à une bande, non-déterministe) accepte une entrée w sans déplacer sa tête de lecture hors de la portion de bande où est écrite l'entrée.

(a) Montrer que ASP_{TM} est PSPACE-complet.
(b) On dit qu'un langage est context-sensitive (CSL) ssi il est reconnu par un automate linéairement borné (LBA), c'est-à-dire une MT à une bande, non-déterministe qui sur l'entrée w ne se déplace jamais en dehors de la $c|w|$ case la plus à droite, pour une constante c. Montrer que le problème de décider si un mot w appartient à CSL (représenté par un LBA), est PSPACE-complet.

Exercice 2: Jeu artificiel

On considère le jeu artificiel, appelé JEU-FORMULE, à deux joueurs qui consiste étant donné une formule booléenne complètement quantifiée en forme prénexe à choisir des valeurs booléennes pour chaque variable. La formule commence par un quantificateur existentiel et se termine par l'un des deux quantificateurs.

Le joueur U choisit les valeurs pour les variables quantifiées universellement et le joueur E pour les variables quantifiées existentiellement. L'ordre des joueurs est le même que l'ordre des quantificateurs. À la fin du jeu, on utilise les valeurs choisies par les joueurs pour évaluer la formule et on déclare que E a gagné le jeu si l'évaluation est à VRAI. Sinon, U gagne.

Un joueur a une *stratégie gagnante* pour ce jeu, s'il gagne quand les deux joueurs jouent de manière optimale.

(a) Montrer que le joueur E a une stratégie gagnante sur la formule

$$\exists x_1 \forall x_2 \exists x_3 [(x_1 \vee x_2) \wedge (x_2 \vee x_3) \wedge (\overline{x_2} \vee \overline{(}x_3))]$$

(b) Montrer que le joueur U a une stratégie gagnante sur la formule

$$\exists x_1 \forall x_2 \exists x_3 [(x_1 \vee x_2) \wedge (x_2 \vee x_3) \wedge (x_2 \vee \overline{(}x_3))]$$

(c) On considère le problème

$$\text{JEU-FORMULE} = \{\langle \phi \rangle : E \text{ a une strategie gagnante pour la formule } \phi\}$$

Montrer que JEU-FORMULE est PSPACE-complet.

Exercice 3: Jeu de Géographie Généralisé

On considère le jeu de géographie à deux joueurs qui consiste à donner des noms de villes chacun à son tour de telle sorte que la première lettre de la n-ième ville soit la dernière lettre de la $(n-1)$-ième ville. Le premier joueur choisit une ville au hasard. Il ne faut pas redonner la même ville deux fois et le premier qui ne peut pas continuer a perdu.

On peut voir ce problème sous la forme d'un graphe orienté où les noms des villes seraient les sommets et les arcs entre deux sommets si la contrainte est respectée (la ville de départ à sa dernière lettre qui est également la première lettre du mot d'arrivée).

Si on part d'un graphe orienté, on dit qu'il s'agit du jeu de géographie généralisé. On considère alors le langage suivant :

$GG = \{\langle G, b\rangle : \text{Joueur } I \text{ a une strategie gagnante sur le graphe } G \text{ a partir du sommet } b\}$

(a) Montrer que ce langage est PSPACE-complet à partir de JEU-FORMULE. (Montrer qu'on peut supposer que la formule commence et se termine par un quantificateur existentiel et qu'elle alterne entre les deux quantificateurs. Coder le choix de chaque variable, imposer à chaque joueur de sélectionner à tour de rôle et coder les clauses.)

(b) Que se passe-t-il si le graphe est acyclique ?

Exercice 4: Isomorphisme de Graphes

Soit deux graphes $G_1(V, E_1)$ et $G_2 = (V, E_2)$ qui ont le même ensemble de sommets $V = \{1, 2, \ldots, n\}$. On dit que les deux graphes sont *isomorphes* s'il existe une permutation $\pi \in \mathbb{S}_n$ telle que $(i, j) \in E_1$ ssi $(\pi(i), \pi(j)) \in E_2$. Deux graphes ne sont pas isomorphes s'il n'existe pas d'isomorphisme d'un graphe vers l'autre.

(a) Montrer que le langage

$$GI = \{\langle G, H\rangle : G \text{ et } H \text{ sont isomorphes}\}$$

est dans NP.

(b) Que pouvez-vous dire pour le complémentaire de ce langage, appelé GNI ?

(c) On dit qu'un *système de preuve* est un protocole (échange de messages) entre deux joueurs, un vérifieur V et un prouveur P qui sont des machines de Turing.

  (a) Le vérifieur est en plus randomisé et fonctionne en temps polynomial. (Il a accès à une bande qui contient des bits aléatoires.)

  (b) Le prouveur est tout puissant mais n'a pas accès aux bits aléatoires du vérifieur, seulement ceux révélés au cours des échanges.

  À la fin du protocole, le vérifieur doit être convaincu d'un théorème par exemple que deux graphes G_1 et G_2 ne sont pas isomorphes : quand G_1 et G_2 ne sont pas isomorphes, il est possible pour P de convaincre V d'accepter, si les deux graphes sont isomorphes, même un prouveur malhonnête ne peut pas persuader V d'accepter avec probabilité plus grande que $1/2$. Enfin, on appelle IP la classe des langages qui ont un système de preuve.
  Montrer que GNI $\in$ IP.

On peut montrer, mais c'est un problème difficile, que PSPACE=IP.