TD 5
Complexité

Luc Lapointe
luc.lapointe@ens-paris-saclay.fr
home.lmf.cnrs.fr/LucLapointe/

Exercice 1 – Planification

On note $I = [\![1, n]\!]$. Un *problème de planification* est donné par :
- n variables booléennes $\{x_i\}_{i \in I}$,
- m *opérations*, où une opération est définie par :
  - une condition de la forme $\bigwedge_{i \in I'} (x_i = \alpha_i)$ avec $I' \subseteq I$,
  - et une mise à jour de la forme $\{x_i \leftarrow \beta_i\}_{i \in I''}$ avec $I'' \subseteq I$.

  Voici un exemple d'opération : Si $x_1 = \top \wedge x_3 = \bot$ alors $x_1 \leftarrow \bot; x_2 \leftarrow \top$.
- Une configuration initiale s_{init} et une configuration finale s_{fin}.
- Une opération est applicable à une configuration si la condition de l'opération s'évalue à $\top$. Son application consiste à effectuer les mises à jour. Par exemple, l'opération précédente est applicable à la configuration $(\top, \bot, \bot)$ et conduit à la configuration $(\bot, \top, \bot)$.

Le problème consiste à déterminer s'il existe une suite d'applications des opérations, avec éventuellement plusieurs applications d'une même opération, qui conduit de la configuration initiale à la configuration finale.

1. Montrer que le problème de planification est dans PSpace.
2. Montrer que le problème de planification est PSpace-dur. Pour cela, on peut exploiter le problème ci-dessous qui est PSpace-dur :

Entrée: Le code d'une machine de Turing M, un mot w, un entier t écrit en unaire.
Question: Est-ce que M accepte w en espace t ?

Pour simplifier, on supposera que M a une seul bande de travail, et l'efface à l'issue du calcul (une fois dans l'état acceptant ou rejetant) ainsi que replace les têtes de lecture au début des bandes correspondantes.

Exercice 2 – Géographie

Le jeu Géographie se joue de la manière suivante :

- Le jeu commence avec un nom de ville donné, par exemple *Gometz-le-Châtel.*
- La première joueuse donne le nom d'une ville dont la première lettre coïncide avec la dernière lettre de la ville précédente, par exemple *Les Ulis.*
- Le deuxième joueur donne un autre nom de ville, commençant également par la dernière lettre de la ville précédente, par exemple *Saclay.*
- Le premier joueur joue à nouveau et ainsi avec la restriction qu'il est interdit de donner le nom d'une ville déjà vue dans le jeu.
- La première personne qui ne peut pas trouver de nom de ville a perdu.

Ce jeu peut être décrit à l'aide d'un graphe orienté dont les noeuds représentent les villes et où une arête (X, Y) signifie que la dernière lettre de X est la première lettre de Y. Ce graphe a également un noeud distingué qui correspond à la ville initiale. Une partie consiste à se déplacer d'un sommet courant en choisissant alternativement l'arête à emprunter, en s'interdisant de repasser par un sommet déjà visité, et il faut réussir à bloquer l'autre.

Géographie Généralisée, abrégé GG, correspond au problème suivant :

Entrée: un graphe orienté G, un noeud initial s.

Question: La joueuse 1 a une stratégie gagnante pour GG sur G depuis s ?

1. Montrer que GG est dans PSPACE.
2. Montrer que GG est PSPACE-dur via une réduction depuis QBF.

Exercice 3 – Espace polylogarithmique

On rappelle le théorème de hiérarchie en espace :

Théorème *Pour deux fonctions constructibles en espace f et g telles que $f(n) = o(g(n))$, on a DSPACE $(f) \subsetneq$ DSPACE (g).*

On définit à présent la classe de complexité :

$$\text{PolyLog} = \bigcup_{k>0} \text{SPACE}\left(\log^k(n)\right)$$

1. Montrer que PolyLog n'a pas de problème complet pour des réductions en espace logarithmique. Que peut-on en déduire quant à la comparaison entre les classes PTime et PolyLog ?
2. On rappelle que $\text{PSpace} = \bigcup_{k>0} \text{SPACE}\left(n^k\right)$. Est-ce que PSpace a un problème complet pour des réductions en espace logarithmique ? Pourquoi est-ce que la preuve de la question précédente ne s'applique pas à PSpace ?