TD 4
Complexité

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

Vous pouvez reprendre les exercices des TD précédents si vous ne les aviez pas déjà tous traités.

Rappel Un langage est dans PSPACE s'il existe une machine de Turing M déterministe décidant le langage L qui termine sur toute entrée et telle qu'il existe un polynôme p tel que l'espace pris par la machine M sur toute entrée x de taille n prend un espace au plus $p(n)$. En particulier, le temps pris peut être exponentiel en n.

Exercice 1 – Mon tout premier problème PSPACE-complet

Montrer que le problème suivant est PSPACE-complet :

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 ?

Exercice 2 – QBF

On rappelle la définition de QBF (*Quantified Boolean Formula*), qui est PSPACE-complet :

Entrée: Une formule booléenne quantifiée

$$\varphi = Q_1 x_1 ... Q_n x_n : \psi(x_1, ..., x_n)$$

avec ψ une formule de la logique propositionnelle sans variable libre, et Q_i valant $\exists$ ou $\forall$.
Question: Est-ce que φ est satisfiable ?

Que dire si tous les quantificateurs sont existentiels ? Et s'ils sont tous universels ?

Exercice 3 – Jeu et complexité

Exercice tiré de [1].

Dans cet exercice, un *jeu* est un graphe orienté $G = (S, A)$ où l'ensemble des sommets $S = S_A \uplus S_B$ est partitionné en sommets appartenant à deux adversaires A et B, avec un sommet distingué s_0 qui sert de sommet initial. Le graphe est tel que chaque sommet a un successeur. Une *partie* correspond alors à un chemin fini ou infini $\rho = s_0 s_1 s_2 ... \in S^* \cup S^\omega$ avec s_0 le sommet initial. Si la partie en est au sommet $s_i \in S_A$ alors c'est le joueur A qui choisit le prochain sommet $s_{\{i+1\}} \in \mathrm{Succ}(s_i)$, sinon c'est la joueuse B. Une *condition de victoire* détermine lorsqu'une partie infinie est gagnée par A. On considère que si A ne gagne pas, c'est B qui gagne. Un joueur $J \in \{A, B\}$ a une *stratégie gagnante* à partir d'un sommet $s \in S$ si, à partir de toute partie qui a mené jusqu'à s, J a une manière de jouer pour gagner à coup sûr toute partie infinie.

1. Supposons que la condition de victoire soit une condition d'atteignabilité : étant donné un ensemble cible $T \subseteq S$, le joueur A gagne si et seulement si un sommet de T est vu. Montrer que décider le vainqueur d'un jeu d'atteignabilité à partir du sommet $s_0 \in S$ peut se faire en temps polynomial.
2. Considérons un entier $k \in \mathbb{N}$. Une condition d'atteignabilité k-généralisée est définie comme suit : étant donné k ensembles cibles $T_1, ..., T_k \subseteq S$, le joueur A gagne si et seulement si, pour tout $i \in [\![1, k]\!]$, un sommet de T_i est vu. Montrer que décider le vainqueur d'un jeu d'atteignabilité k-généralisé à partir du sommet $s_0 \in S$ peut être fait en temps polynomial.
3. Une condition d'atteignabilité généralisée est définie comme suit : étant donné plusieurs ensembles cibles $T_1, ..., T_k \subseteq S$, le joueur A gagne si et seulement si, pour tout $i \in [\![1, k]\!]$, un sommet de T_i est vu. La différence avec l'objectif k-généralisé est que le nombre de cibles n'est pas borné a priori. Montrer que décider le vainqueur d'un jeu d'atteignabilité généralisé est $\mathrm{PS{\scriptsize PACE}}$-complet.
   Indice : Pour l'appartenance à PSPACE, on pourra utiliser que si le joueur A peut gagner avec k cibles, il peut voir toutes les cibles en au plus $k \cdot n$ étapes avec $n := |S|$.
4. Montrer que décider le vainqueur d'un jeu d'atteignabilité généralisé lorsque $S_B = \emptyset$ (i.e. le joueur A joue tout seul) est NP-complet.

Bibliographie

[1] N. Fijalkow et F. Horn, « The surprizing complexity of generalized reachability games », *arXiv preprint arXiv:1010.2420*, 2010.