Examen Intelligence Artificielle
ING2 - mai 2019
Documents non autorisés - Durée 2 heures

Exercice 1 :

Soit $R(x, y)$ un prédicat binaire qui a une interprétation liée à un graphe. Les variables dans ce prédicat représentent les sommets du graphe et la relation $R(x, y)$ signifie qu'il y a un arc du sommet x au sommet y (l'arc est orienté).
Soient les quatre formules de la logique du premier ordre suivantes :

- $F1 : \forall x((\exists y \neg R(x, y)) \Rightarrow \exists y(R(x, y) \wedge R(y, x)))$
- $F2 : \forall x \exists y(R(x, y) \vee R(y, x))$
- $F3 : \forall xyz((R(x, y) \wedge R(y, z)) \Rightarrow R(x, z))$
- $F4 : \exists x R(x, x)$

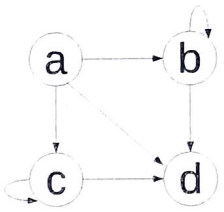

Figure 1 – graphe

- Expliquer la signification de chaque formule $\{F1, F2, F3, F4\}$ en langage naturel.
- Lesquelles des formules sont vraies et lesquelles sont fausses si on considère le graphe de la Figure 1. Justifier votre réponse.
- Montrez en utilisant la méthode de résolution que :
  - la formule F2 est conséquence logique de la formule F1.
  - la formule F4 est conséquence des deux formules F1 et F3.

Exercice 2 :

On veut programmer une grue pour réarranger une pile de conteneurs. Nous proposons de résoudre ce problème en utilisant l'algorithme A^*. Pour cela, nous utiliserons la fonction heuristique h : le nombre de conteneurs mal placés.

- Est ce que la fonction h est admissible ? Justifiez votre réponse.

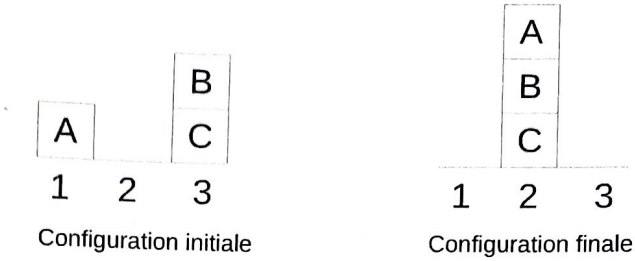

FIGURE 2 – Problème de repositionnement des conteneurs

- L'algorithme $A*$ est il complet ? optimal ?
- Quel sera le facteur de branchement de l'arbre de recherche appliqué à l'exemple de la Figure 2.
- Déroulez l'algorithme A^* avec la fonction heuristique h sur l'exemple de la Figure 2 consituté de trois conteneurs $\{A, B, C\}$. L'objectif est de trouver le minimum d'action que la grue doit faire pour passer de la configuration initiale à la configuration finale. Lors de la génération d'états successeurs, il faudrait respecter les contraintes suivantes :
  - La grue ne peut déplacer qu'un conteneur à la fois. Le coût de cette action est égal à 1.
  - On peut déplacer un conteneur uniquement s'il n'a pas un autre au dessus de lui.
  - On peut poser un conteneur dans une pile vide ou sur un autre conteneur.
  - Les états déjà explorés dans l'arbre de recherche ne doivent pas être développé une deuxième fois (votre arbre ne doit pas contenir deux états identiques).

Exercice 3 :

On s'intéresse ici au jeu de Nim simplifié : on dispose au départ de 5 allumettes et chacun des deux joueurs peut à son tour enlever 1 ou 2 allumettes. Le joueur perdant est celui qui ramasse la dernière allumette.

- Construire l'arbre complet du jeu en partant d'une rangée de 5 allumettes.
- Supposons que le joueur 1 commence. Appliquez l'algorithme Min-Max sur l'arbre du jeu construit précédemment. Pour les valeurs des états finaux (les noeuds feuilles), vous utiliserez les valeurs suivantes : 1 si le joueur 1 gagne, -1 s'il perd.

Exercice 4 :

On considère un jeu à deux joueurs dont l'arbre de jeu est donné par le schéma ci-dessous :

1. Sachant que la racine est un noeud MAX. Appliquez l'algorithme $\alpha - \beta$ à ce jeu . Pensez à faire apparaître les élagages.