

Résolution de problèmes

Recherche avec heuristique

Rappel

Les algorithmes de recherche sont un mécanisme général de résolution de problèmes très utilisé en intelligence artificielle.

- Le problème est décrit par un espace d'états contenant :
 - Un état initial
 - Un ou plusieurs états finaux (évalués par une fonction test-but)
 - Un ensemble d'actions possibles permettant de transiter d'un état à un autre pour un coût donné.
- Une résolution correspond à une suite d'actions (appelée chemin) qui mène de l'état initial à l'état final.
- La meilleure solution est la résolution la moins coûteuse.

1 Le problème des missionnaires

1.1 Présentation

Trois missionnaires et trois cannibales sont d'un même côté d'une rivière qu'ils doivent traverser à l'aide d'une barque à deux places nécessitant au moins un homme pour la manœuvrer. Si sur un côté (et la barque est toujours considérée comme étant sur un des côtés) il y a strictement plus de cannibales que de missionnaires, ces derniers sont mangés. Comment faire traverser tout le monde sans que les missionnaires ne soient mangés ?

1.2 Formalisation

1. Choisir un formalisme permettant de représenter les états possibles.
2. Combien y a-t-il d'états possibles ?
3. Identifier, dans ce formalisme, l'état initial et la fonction test-but.
4. Donner les opérateurs (conditions et transitions).

2 Problème du taquin

2.1 Présentation

Ce jeu est composé de 8 pièces numérotées et d'une case vide que l'on peut déplacer. Le but est partant d'une configuration mélangée de retrouver la configuration *but* illustrée par la figure 1.

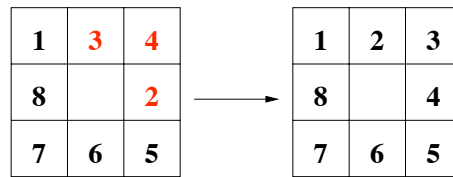


FIGURE 1 – Un état initial (à gauche) et l'état-but (à droite) du jeu du taquin

2.2 Formalisation

1. Choisir un formalisme permettant de représenter les états possibles.
2. Combien y a-t-il d'états possibles ?
3. Identifier, dans ce formalisme, l'état initial et la fonction test-but.
4. Donner les opérateurs (conditions et transitions).

2.3 Application de l'algorithme A*

A* est un algorithme de recherche heuristique (voir Algorithme 1), il doit donc être guidé par une fonction heuristique qui estime la qualité d'un nœud. Pour un nœud n , on utilise une fonction de la forme $f(n) = g(n) + h(n)$ où g est le coût du chemin depuis l'état initial jusqu'au nœud et h est une estimation du coût du chemin entre le nœud et l'état final.

1. Proposer au moins trois heuristiques pour résoudre le jeu de taquin avec un algorithme de type A*.
2. Appliquer l'algorithme A* avec la meilleure heuristique que vous ayez trouvée sur le taquin.

Algorithme 1 A (A^* si h est admissible)

s_0 est l'état initial
OUVERTS est la liste des nœuds à étudier
FERMES est la liste des nœuds étudiés
la fonction tête(OUVERTS) rend le premier élément de la liste OUVERTS
la fonction chemin(s) appelle chemin(pere(s)) si $s \neq s_0$ puis rend s

OUVERTS $\leftarrow \{s_0\}$
FERMES $\leftarrow \emptyset$
 $g(s_0) \leftarrow 0$
 $s \leftarrow s_0$
tant que OUVERTS $\neq \emptyset$ et tête(OUVERTS) non terminal **faire**
 Supprimer s de OUVERTS et le mettre dans FERMES
 pour tout nœud s' successeur de s **faire**
 si $s' \notin (OUVERTS \cup FERMES)$ ou $g(s') > g(s) + \text{cout}(s, s')$ **alors**
 $g(s') \leftarrow g(s) + \text{cout}(s, s')$
 $f(s') \leftarrow g(s') + h(s')$
 $\text{pere}(s') \leftarrow s$
 Ranger s' dans OUVERTS dans l'ordre f croissant
 si $s' \in FERMES$ **alors**
 FERMES $\leftarrow FERMES \setminus \{s'\}$
 fin si
 fin si
 fin pour
 si OUVERTS $\neq \emptyset$ **alors**
 $s \leftarrow \text{tete}(OUVERTS)$
 fin si
fin tant que
si OUVERTS = $\emptyset$ **alors**
 le problème n'admet pas de solution
sinon
 le chemin(s) est solution
fin si
