
Travaux pratiques d'IA

SÉRIE 3 : RECHERCHE HEURISTIQUE

À rendre le 22 octobre 2020.

1 Simulation de l'algorithme A^*

Supposez que vous deviez trouver un chemin entre S et G dans l'espace d'état défini par la graphe dirigé suivant. Le nombre attaché à chaque arc représente le coût pour traverser cet arc.

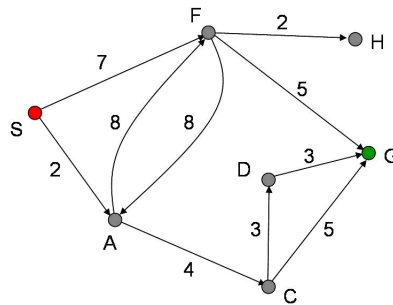


FIGURE 1 – Espace d'état

1.1 Recherche aveugle

pour chacune des méthodes de recherche ci-dessous :

- recherche en largeur
- recherche en profondeur

Dessiner l'arbre de recherche au moment où l'état but est atteint. Noter la longueur du chemin parcouru de l'état initial à l'état final. Indiquez aussi l'ordre d'extension des noeuds entre parenthèse à côté des noeuds étendus. Lorsque qu'aucun ordre de priorité existe sur l'extension des noeuds, suivez l'ordre alphabétique. Ne développez pas les noeuds qui font partie d'une boucle.

1.2 Recherche heuristique

la fonction de recherche heuristique $h(N)$ estime à chaque noeud N de l'arbre de recherche, le coût pour atteindre G depuis l'état associé à N . Cela dépend seulement de l'état associé avec N et non pas de la position de N

dans l'arbre de recherche.
 Soit h définit par la table suivante

Etat	S	A	C	D	F	G	H
h	10	5	4	3	4	0	2

- Est ce que cette fonction est admissible ? pourquoi ?
- Quelle est la fonction d'évaluation utilisée par la méthode greedy best first search et A^* ?
- Pour chacune de ces deux méthodes de recherche, listez les noeuds de l'arbre de recherche dans l'ordre dans lesquels ils sont étendus (contrairement au point précédant, il n'est pas nécessaire de dessiner l'arbre de recherche). Pour chaque noeud étendu, indiquez les noeuds qui sont créés ainsi que la valeur de la fonction d'évaluation. Utiliser le format suivant :
 1. *Greedy best first search* crée la racine de l'arbre de recherche $N0$ avec l'état S et $f(N0) = 10$ Ensuite, il va étendre successivement les noeuds suivants :
 2. $N0$, pour créer $N1$ avec l'état A et $f(N1) = 5$, et $N2$ avec l'état F et $f(N2) = 4$.
 3. $N2$, pour créer ...
 4. ...

Comme pour la question précédente, Lorsque qu'aucun ordre de priorité existe sur l'extension des noeuds suivez l'ordre alphabétique. Ne développez pas les noeuds qui font partie d'une boucle (c'est à dire un noeud dont l'état associé apparaît déjà dans la branche en cours de traitement). Les deux méthodes de recherche, assument que le test du noeud solution est appliqué lorsqu'un noeud est retiré de la pile.

Finalement, implémenter Greedy best first search pour se problème.