ESIEE Paris

PR-3602

Résolution de problèmes en intelligence artificielle et optimisation combinatoire : les algorithmes A*

Auteurs :
Boris Ghidaglia
Augustin Prolongeau

Encadrant :
M.Couprie

28 mars 2018

Sujet : https ://perso.esiee.fr/ coupriem/PR3602/

Table des matières

1 Algorithme naïf **2**

1.1 Concept . . . 2

1.2 Questions . . . 2

2 Algorithme glouton **3**

2.1 Concept . . . 3

2.2 Questions . . . 3

3 Algorithme A* **4**

3.1 Concept . . . 4

3.2 Questions . . . 4

1 Algorithme naïf

1.1 Concept

Tester toutes les solutions possibles et choisir la moins couteuse.

1.2 Questions

Combien y'a-t il de solutions possibles ?
Le sujet stipule que nous sommes dans le cas d'une matrice $N \times N$. Cela signifie que chaque agent peut être attribué à N postes différents. Cependant, à chaque fois qu'un agent est affecté à un poste, c'est une combinaison en moins à tester pour tous les autres. Ainsi, il y a $N!$ solutions possibles.

Si l'on suppose qu'une affectation peut être évaluée en une microseconde, et que l'on dispose de trois mois pour faire le calcul, quelle est la valeur maximum de N possible pour envisager d'appliquer cette méthode ?
Calculons combien de microsecondes trois mois représentent (on considèrera qu'un mois dure environ 30.5 jours) :

$$3 \text{ mois} = 3 \times 30.5 \times 24 \times 3600 \times 10^6 = 7.9056 \times 10^{12} \mu s$$

Il suffit alors de prendre la plus grande factorielle inférieure à cette valeur pour connaître notre N maximal théorique :

$$16! = 2.0922789888 \times 10^{13}$$

$$15! = 1.307674368 \times 10^{12}$$

Notre N maximum théorique est donc : 15. Si nous devons gérer une équipe de plus de 15 agents à affecter à plus de 15 postes, il nous sera impossible de calculer le résultat optimal via cet algorithme en trois mois ou moins.

2 Algorithme glouton

2.1 Concept

Il s'agit de sélectionner la valeur minimale de la matrice des coûts, d'effectuer l'affectation correspondante et de retirer le poste et l'agent qui sont concernés, puis de recommencer jusqu'à affectation de la totalité de l'effectif.

2.2 Questions

Montrez par un contre-exemple simple que l'algorithme glouton ne trouve pas toujours la solution optimale pour ce problème

Posons les matrices 3×3 suivantes : situation initiale, solution glouton et solution optimale :

$$\begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 5 \\ 3 & 5 & 100 \end{bmatrix} \begin{bmatrix} {\color{red}1} & 2 & 3 \\ 2 & {\color{red}4} & 5 \\ 3 & 5 & {\color{red}100} \end{bmatrix} \begin{bmatrix} 1 & 2 & {\color{green}3} \\ {\color{green}2} & 4 & 5 \\ 3 & {\color{green}5} & 100 \end{bmatrix}$$

On constate bien que l'algorithme glouton dévore la plus petite valeur de la matrice C_k à chaque étape k, sans se soucier des conséquences de ses actes sur ses choix futurs.

3 Algorithme A*

3.1 Concept

3.2 Questions