

Projet IA - Problème de la patrouille - Approche EVAP

Timothé Rios - Nicolas Venot

mars 2021

Table des matières

1	Principe général	3
2	Implémentation	3
2.1	Les cases	3
2.2	Les agents	3
3	Algorithme	4
3.1	4
4	étude des paramètres	4
4.1	4
5	Conclusion	4
5.1	4
A	Récapitulatif des types et fonctions	5

Introduction

Ce projet a pour objectif de modéliser aussi précisément que possible le problème de patrouille en utilisant l'approche d'évaporation des phéromones. Cette approche, inspirée par l'étude des colonies d'insectes sociaux, a pour particularité d'attribuer à chaque zone géographique un taux de phéromones spécifique qui diminue avec le temps. Les agents ont alors pour fonction de maximiser autant que possible le taux de phéromones des zones autour d'eux, d'où la nécessité de patrouiller et la pertinence de ce modèle pour résoudre ce problème.

1 Principe général

Dans ce modèle, le temps est discrétisé et l'environnement est modélisé par une grille de cases. Chaque agent avance d'une case par pas de temps, sur la case à gauche, à droite, devant ou derrière lui qui a la plus petite quantité de phéromones. Chaque fois qu'un agent arrive sur une case, il dépose une quantité P_{max} de phéromones sur celle-ci.

À chaque pas de temps, toutes les cases perdent une certaine quantité de phéromones : si elles ont P_t phéromones à l'instant t , elles ont $P_{t+1} = P_t * (1 - p)$ à l'instant $t + 1$, avec p compris entre 0 et 1 exclus.

Une catégorie de cases ne peut pas recevoir de phéromones et ne peut pas être traversée par les agents. Cette catégorie permet de représenter des murs ou des obstacles dans ce modèle. Évidemment, les agents en contact avec des cases de type mur ne vont pas prendre en compte ces dernières dans leurs déplacements.

Enfin, si plusieurs cases atteignables par un agent au prochain pas de temps ont toutes deux le niveau de phéromones minimum parmi les voisins de cet agent, il choisit au hasard parmi celles-ci. Cependant, si une de ces cases est la case qui lui fait face, il a la probabilité p_{avant} de garder sa trajectoire. Ceci permet d'éviter des mouvements trop erratiques.

2 Implémentation

2.1 Les cases

Chaque case voit attribuer un booléen indiquant si celle-ci est un mur ou non (aussi distingué par une couleur bleue qui leur est spécifique), un paramètre phéromone chargé d'indiquer le niveau de phéromones de chaque case ainsi qu'un paramètre ini destiné à être utilisé à des fins d'observation statistiques. Les cases se voient attribuer une couleur à chaque tick, celle-ci varie selon leur niveau de phéromones : plus celui-ci est proche de 1 plus la case apparaît verte, lorsqu'il se rapproche de 0 la case redevient noire. Le niveau de phéromones est multiplié par $1 - p$ à chaque tick (p modulable) si la case n'est pas un mur afin de simuler l'évaporation des phéromones.

2.2 Les agents

Les agents sont chargés de maintenir le taux de phéromones de tous les cases le plus proche de 1 possible. Ils fonctionnent de la manière suivante : à chaque tick, l'agent dresse la liste des cases ayant le niveau de phéromones le plus faible parmi ses 4 voisins (uniquement sur la même ligne ou colonne). Une fois la liste créée, l'agent va effectuer un mouvement ayant un pourcentage de chance défini par le paramètre 'go-ahead' de privilégier la case en face de lui si celle-ci fait partie de la liste, sinon l'agent fait simplement le mouvement vers une autre case de la liste en se tournant vers elle au préalable.

Après avoir atteint une nouvelle case, l'agent indique à celle-ci son nouveau niveau de phéromones (égal à 1, la case devient verte) et remet son paramètre ini à 0. Les agents ignorent les cases correspondant à des murs car ceux-ci se voient attribuer une valeur fixe de phéromones par défaut de 2, il ne peuvent donc jamais correspondre au minimum de la liste, ce qui empêche l'agent d'engager un déplacement vers un mur.

3 Algorithme

3.1

4 étude des paramètres

4.1

5 Conclusion

5.1

A Récapitulatif des types et fonctions