

Rapport de Mini-Projet 3I025

Recherche de chemins coopératifs entre agents

Castellon Clément

Zhang Noé

22 mars 2019

Première partie

introduction

1 Projet

Nous considérons plusieurs agents en compétition qui cherchent chacun à atteindre leur propre objectif et nous souhaitons éviter qu'il y ait des collisions entre eux. Nous cherchons à maximiser la coopération entre les agents et ainsi minimiser le temps total nécessaire à ce que chacun atteigne son objectif.

2 Notions de base

2.1 Agents Intelligents

Un agent intelligent est une entité autonome qui peut-être caractérisé par plusieurs caractéristique, celui-ci doit être capable de s'adapter à son environnement, aussi il doit pouvoir être configurable afin de répondre aux besoins précis de l'utilisateur. De plus, dans un cadre plus avancé, il doit savoir profiter de ses expériences passées afin de mieux comprendre les souhaits de l'utilisateur. Lors du déploiement de plusieurs agents la communication et l'interaction entre eux est primordiale afin de maximiser la coopération.

2.2 Collisions

Il existe 2 type de collisions différentes, la première occure lorsque deux agents différents veulent accéder à la meme case au même moment, la seconde à lieu lorsque ceux-ci sont face à face et vont se croiser.

2.3 Solutions

Afin d'empêcher les différentes collisions qui pourrait arriver, différentes solutions peuvent être implémenté afin que les agents puissent réagir de manière

autonome .

Deuxième partie

Stratégies

3 Path-Splicing

3.1 Idée globale

Cette première implémentation consiste à modifier le chemin emprunté par un agent seulement lorsque celui-ci détecte une collision c'est-à-dire à l'instant précédant la collision, en effet lorsqu'un agent détecte une collision avec un autre, il va considérer le lieux de collision comme un obstacle et va calculer en conséquence un nouveau chemin qui lui permettra d'atteindre un état postérieur de son chemin en tenant compte du lieux de collision.

3.2 Implémentation

Dans un premier temps chaque agents calcule indépendamment des autres un chemin lui permettant d'atteindre son objectif, pour cela nous avons utilisé l'algorithme A étoile qui permet de calculer le plus court chemin entre deux coordonnées Une fois chaque chemin calculé, on garde en mémoire chaque chemin et on associe un compteur à chaque agent qui correspond à l'étape où se trouve l'agent par rapport à son chemin associé. Un ordre de passage est associé à chaque agent cela à des conséquences sur le chemin des agents. En effet chaque agent calcul les collisions par rapport à ceux qui ont déjà effectuer leur déplacements.

3.3 Résultat

Graphique à mettre

4 Coopérative Basique

4.1 Idée globale

Pour cette stratégie, les agents se déplacent par groupes formés à partir des chemins qui ne produisent pas de collisions qui sont alors executé en différés, différentes stratégies peuvent être implémentés afin de définir l'ordre de passage.

4.2 Implémentation

Pour chaque agent, un chemin est calculé à partir de l'algorithme A étoile sans tenir compte des autres agents. On forme ensuite des groupes d'agents en

fonction des chemins qui forment pas de collisions s'ils ont lieux en même temps. L'ordre de passage des groupes est alors organisé grâce à une stratégie choisie, la stratégie de base consistant à prendre le premier groupe formé et ainsi de suite. Lorsque tout les agents d'un groupe ont atteint leur objectif, on recalcule un chemin vers leur nouvel objectif et on leur associe un groupe. Enfin, avant d'envoyer le prochain groupe on va recalculer leur chemin en tenant compte de la position courante des agents des autres groupes. Si deux nouveaux chemins forme produisent une collision on en enlève un des deux et on envoie le groupe de chemin.

4.3 Résultat

Graphique à mettre

5 Coopérative avancée

5.1 Idée globale

On souhaite utiliser une structure de données partagée qui correspond à une table de réservation où l'on va stocker des triplets qui correspondent à la position à un instant donnée, si une case est réservée, les autres agents ne pourront pas y accéder évitant ainsi les collisions.

5.2 Implémentation

5.3 Résultat

6 Discussion

Troisième partie

Conclusion