



SORBONNE UNIVERSITÉ

2I013 : PROJET (APPLICATION)

Sujet :
IA Football

Fangzhou Ye
Angelo Ortiz

Table des matières

Introduction	3
I Architecture logicielle	4
II Stratégies	5
III Méthodes d'optimisation	7
1 Recherche en grille	7
2 Algorithmes génétiques	7
3 Arbres de décision	7
4 Apprentissage automatique	7
Conclusion	8

Introduction

Présentation

Le projet a consisté en le développement d'intelligences artificielles de joueurs de football. Le simulateur du jeu étant déjà fourni, notre travail a été d'implémenter des stratégies de jeu pour mieux réussir les matches. L'objectif de ce travail a été d'apprendre à bien mener un projet collaboratif long sur un nouvel environnement de développement, à savoir Python.

Aperçu

Voici un aperçu du jeu... ...

Arborescence

Le projet a été organisé en un module **ia** dirigeant le comportement de nos joueurs et en divers fichiers de test.

ia/tools

Ce fichier contient une classe représentant l'état du jeu qui comporte des méthodes nous facilitant l'accès aux diverses caractéristiques de cet état, ainsi que plusieurs fonctions utilitaires.

ia/conditions

Comme le nom l'indique, ce fichier a été dédié aux diverses conditions qui déclenchent les différents types de comportement.

ia/actions

Ce fichier comporte les actions de base que l'on peut réaliser à chaque instant du jeu.

ia/behaviour

En ce qui concerne ce fichier, ici se trouvent les actions complexes, i.e. composées de plusieurs actions de base. En effet, pour une certaine phase de jeu, un joueur peut prendre différentes décisions.

ia/strategies

C'est dans ce fichier que les stratégies ont été réunies. Ces stratégies dirigent le comportement des joueurs tout au long d'un match.

ia/gene_optimisation

Ce fichier contient les classes nécessaires pour la mise en œuvre d'une implémentation des algorithmes génétiques.

ia/gs_optimisation

On retrouve ici le code utilisé pour suivre la méthode de recherche en grille.

ia/

init ? ML_strats ? DT_strats ?

Il reste
encore
des fi-
chiers

Première partie

Architecture logicielle

Dans cette première partie, on vous expliquera les choix faits concernant l'architecture logicielle.

Deuxième partie

Stratégies

Dans cette deuxième partie, on vous détaille le comportement de nos différents joueurs.

Compte tenu de la taille du terrain de jeu, le nombre maximum de joueurs par équipe est limité à quatre. Ainsi, il y a trois catégories de matches selon le nombre de joueurs par équipe : un, deux ou quatre.

Un joueur

Pour l'équipe composée uniquement d'un joueur, on a développé un seul joueur que l'on appelle *Fonceur*. Au tout début du projet, il faisait tout au maximum : il doit s'approcher de la balle à toute vitesse lorsqu'il ne la contrôle pas et il frappe avec toute sa puissance dans l'autre cas. On a décidé de garder le même nom, même si son comportement diffère. En effet, à présent il avance avec la balle de petites distances et lorsqu'un adversaire lui fait opposition il essaie de le dribbler ; il frappe vers la cage adverse s'il se trouve dans la surface de réparation.

Deux joueurs

Cette catégorie a été celle sur laquelle on a consacré la plupart du projet. On compte trois joueurs fonctionnels que l'on appelle *Attaquant*, *Gardien* et *Défenseur*.

Attaquant

Il a une vocation plutôt offensive. Il se comporte différemment selon s'il contrôle la balle ou pas. Dans la première situation, il essaie systématiquement d'avancer sur le terrain avec la balle et de faire un tir vers la cage dès qu'il se trouve dans la surface de réparation adverse. Cependant, s'il rencontre un adversaire lui bloquant le chemin, il essaie de faire une passe vers son coéquipier s'il est sans marquage, il dribble le joueur sinon. Lorsqu'il se trouve sans le contrôle de la balle ...

Gardien

Ce joueur se comporte exactement pareil que l'attaquant lorsqu'il s'agit d'attaquer. En revanche, en situation défensive, il reste à une distance de sa cage lorsque l'équipe adverse contrôle la balle et essaie d'intercepter la balle lorsque le porteur du ballon se trouve dans sa surface de réparation. S'il voit que son coéquipier va bientôt contrôler la balle ou il en a déjà le contrôle, il monte dans le terrain pour lui proposer une possibilité de passe.

Défenseur

Le défenseur n'a par contre qu'une vocation défensive. En effet, s'il a la balle, il s'en défait le plus rapidement possible soit à l'aide d'une passe vers son coéquipier si démarqué, soit via un dégagement le moins risqué possible. Quant à son effort défensif, il demeure à une certaine distance radial à partir de sa cage et s'approche de la balle en vu d'une interception s'il en suffisamment proche.

Quatre joueurs

Il s'agit de la catégorie que l'on a trouvée la plus intéressante en raison de la difficulté de la gestion des espaces et la répartition des fonctions. Tout comme pour la catégorie précédente, on compte trois joueurs fonctionnels que l'on repère d'ailleurs par les mêmes noms.

Attaquant

Il y a deux différences en son comportement par rapport à son pair de la catégorie ci-dessus et celles-ci concernent toutes les deux la situation où il se trouve dans sa surface de réparation. La première fait référence à la récupération de la balle : il ne se précipite plus et se comporte tel qu'il le fait dans le milieu du terrain. La deuxième correspond à l'effort de marquer les adversaires sans marquage.

Gardien

Ce joueur se comporte exactement pareil que l'attaquant lorsqu'il s'agit d'attaquer. En revanche, en situation défensive, il reste à une distance de sa cage lorsque l'équipe adverse contrôle la balle et essaie d'intercepter la balle lorsque le porteur du ballon se trouve dans sa surface de réparation. S'il voit que son coéquipier va bientôt contrôler la balle ou il en a déjà le contrôle, il monte dans le terrain pour lui proposer une possibilité de passe.

Apparemment c'est le même gardien qu'en 2v2

Défenseur

Le défenseur n'a par contre qu'une vocation défensive. En effet, s'il a la balle, il s'en défait le plus rapidement possible soit à l'aide d'une passe vers son coéquipier si démarqué, soit via un dégagement le moins risqué possible. Quant à son effort défensif, il demeure à une certaine distance radial à partir de sa cage et s'approche de la balle en vue d'une interception s'il en suffisamment proche.

plus d'interception s'il y a un coéquipier plus proche de la balle que lui (sans la balle) (quid de la sit ou il la controle ?)

Troisième partie

Méthodes d'optimisation

Dans cette troisième partie, on vous détaille les différents algorithmes appris tout au long du semestre pour l'amélioration des stratégies initialement proposées.

1 Recherche en grille

La première méthode d'optimisation présentée en cours a été la recherche en grille. Celle-ci est utilisée pour l'optimisation d'une action.

Tout d'abord, on doit définir l'action à optimiser et le critère d'optimalité, et repérer les paramètres associés. Puis on obtient un ensemble de valeurs pour chaque paramètre. Pour les paramètres discrets, il suffit de prendre toutes les valeurs possibles, alors que pour les paramètres continus il faudra faire en amont une discrétisation, i.e. on divise l'intervalle délimité par des bornes définies selon un pas de précision. On compte ainsi des vecteurs dont les coordonnées correspondent chacune à un paramètre. On fait ensuite une recherche exhaustive sur tous les vecteurs obtenus : on teste chaque vecteur sous différentes conditions environnementales et on moyenne les évaluations du critère dans lesdites conditions. Finalement, on prend la valeur optimale et son vecteur associé.

Il est important de remarquer que la plupart de nos paramètres sont continus. Pour obtenir des valeurs plus précises et un meilleur comportement, on a besoin d'un pas de discrétisation très petit. Ainsi, on réalise qu'une augmentation du nombre de paramètres à optimiser a une très forte influence sur le temps d'exécution de l'algorithme mis en place. On en déduit que le minuscule gain relatif nombre de paramètres-temps d'exécution ne convient pas au moyen terme.

2 Algorithmes génétiques

Dans un deuxième temps, on est passés aux algorithmes génétiques.

TODO

3 Arbres de décision

4 Apprentissage automatique

Dans la dernière partie du semestre, on s'est concentré sur l'apprentissage automatique. On nous a présenté les deux grandes classes d'algorithmes d'apprentissage : supervisés et non supervisés, ainsi que l'apprentissage par renforcement. En effet, ce dernier type d'apprentissage implémenté par l'algorithme de **Q-learning** a été mise en place.

Conclusion

...