



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	6
1 Recherche en grille	6
2 Algorithmes génétiques	6
3 Arbres de décision	6
4 Apprentissage automatique	6
Conclusion	7

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*. Les deux premiers présentent le même comportement lorsqu'ils se trouvent en situation d'attaque : _____

reprendre
ICI

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

...

2 Algorithmes génétiques

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

...