

Projet Foot – 2I013

***Baskiotis Nicolas***

***Benadjal Adame / Luvisamo Jessy***

***Groupe 1***

***Licence Informatique (2016/2017)***

Sommaire

Introduction

Présentation

## Objectifs

## Difficultés

## Organisation du code

Stratégies

## 2 contre 2

## 4 contre 4

## Innovations

Analyses

## Résultats

## Conclusion

Introduction

**Nous avons choisi ce projet foot car nous aimons le football (forcément) mais aussi parce que nous étions fortement interéssés par l’idée de participer à la conception d’un jeu de foot. Dans cette Unité d’Enseignements (UE) nous avons appris, grâce à l’apprentissage statistique, à coder une Intelligence Artificielle (IA) pour créer des joueurs pouvant évoluer de façon intelligente à travers une plateforme de simulation qu’il nous a fallu maîtriser en quelques séances.**

Présentation

Objectifs

**Nous avions pour objectif de créer les meilleures stratégies pour nos joueurs afin d’être bien classé chaque semaine dans le tournoi hebdomadaire. Amateur de jeu de foot(et de foot en général) que nous sommes, nous posions sur papiers des stratégies plus farfelues et ingénieuses les unes des autres. Cependant, il y avait un écart énorme entre ce qu’on arrivait à mettre sur papier et ce que l’on savait coder.**

Difficultés

En effet, cela s’annonçait plus difficile que prévu de coder nos « super » stratégies. La première consigne du professeur fût de coder une stratégie simple dite de « fonceur », c’est-à-dire, une stratégie pour laquelle le joueur court vers le ballon et tire droit devant lui le plus fort possible. Coder cette stratégie simpliste nous fit prendre conscience de plusieurs choses que l’on pouvait améliorer: hormis le fait qu’elle ne sois pas difficile à contrer, on remarqua qu’il faudrait créer un test pour indiquer au joueur s’il est, oui ou non, en position de tirer. Effectivement, lorsqu’il arrive devant la balle, ce dernier la dépasse car il doit attendre 10 pas avant de pouvoir tirer à nouveau et vu qu’il tire en permanence pendant sa course, ce problème intervient. Au fur et à mesure de l’implémentation de notre code, apparût le problème du « Domicile/Extérieur »: effectuer un test, afin de savoir de quel côté on attaque/défend, pour chacune de nos fonctions le rendait extrêmement et inutilement long.

Organisation du code

Nous nous sommes donc penché sur la question de l’organisation de notre code. Au départ, nous utilisions un seul et unique fichier pour tout: nos stratégies (AttaqueStrategy, PasseStrategy, ...), et nos fonctions (aller(), shoot(), ...). Mais suite aux conseils et réflexions des professeurs, nous les avons séparé en deux fichiers. Le fichier strategy.py contenant toutes nos stratégies. Et le fichier toolbox.py contenant nos fonctions. D’ailleurs, pour ce fichier, nous avons choisi de coder nos fonctions à travers plusieurs classes en suivant le principe d’héritage (la classe fille hérite de toutes les fonctions de la classe mère): la classe Action hérite de la classe Deplacement qui hérite elle-même de la classe Position. Nous avons organisé le code de cette façon afin de le rendre plus clair et facilement lisible; c’est pourquoi nous avons aussi classé les fonctions dans chaque classe par ordre alphabétique.

Stratégies

2 contre 2

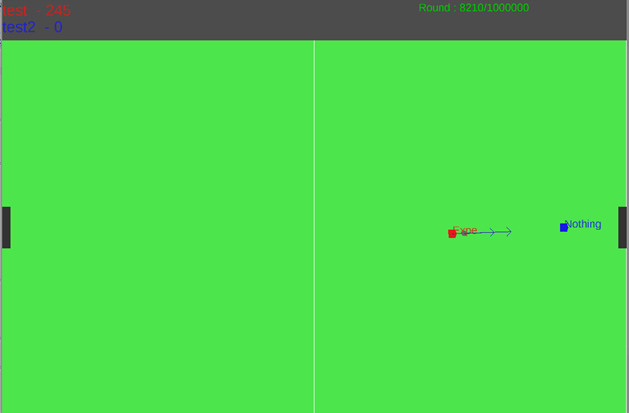
Quant aux stratégies, nous n’avons pas jugé utile de développer plus en profondeur notre stratégie pour le contre 1 contre 1. Nous avons juste ajouter une fonction dribbler à notre attaquant et améliorer la précision de son tir. En ce qui concerne la stratégie du 2 contre 2, nous avons décidé, contrairement à la plupart des groupes que nous avons observé, de jouer tout pour l’attaque: en d’autres termes, nous avons deux joueurs qui se concentrent sur les tâches offensives plutôt que d’avoir un joueur qui se consacre à l’attaque et un autre à la défense. Ainsi, notre attaquant 2v2 réalise un appel sur un coté du terrain tandis que son coéquipier va s’emparer du ballon et dribbler vers l’avant jusqu’à ce que l’attaquant soit en position de tirer pour lui faire une passe. Pour cela, nous avons découpé le terrain en 3 zones: attaque, milieu, défense. La zone attaque correspondant évidemment à la zone où l’attaquant doit se situer pour que la passe soit déclenchée. Une fois la passe effectuée, le passeur se replace au centre du terrain pour récupérer la balle si celle-ci est renvoyée par un défenseur.

4 contre 4

En revanche, la tactique « tout pour l’attaque » semblait plus difficile à appliquer en 4 contre 4. Cela s’explique par le fait que lorsqu’on augmente le nombre de joueurs sur le terrain, les possibilités d’offensives augmentent considérablement et vu qu’on encaissait déjà beaucoup de buts en 2 contre 2, reproduire cette tactique aurait été synonyme de défaite assurée. Par conséquent, nous avons mis en place une formation comprenant: un gardien, un passeur, et deux attaquants. Le passeur et le premier attaquant ont les mêmes rôles et fonctions que ceux en 2 contre 2. Le gardien quant à lui, n’intervient que lorsque la balle est dans la zone défense que nous avons défini dans toolbox.py. Sinon, il reste en position devant ses buts, il adopte, en somme, le comportement d’un véritable gardien de but. Enfin, le 4e et dernier joueur, ou 2e attaquant, n’est ni plus ni moins que notre « fonceur » du 1 contre 1.

Innovations

A la base, les joueurs étaient seulement capables de tirer et de se déplacer. Grâce à recherche\_params.py, un fichier fourni par le professeur qui permet de tester les actions des joueurs afin d’en extraire une position, un angle, une distance, etc... de tir ou de passe optimal; nous avons optimiser ces deux actions triviales pour en faire d’autres fonctions (passe, dribble, ...).



Chaque semaines nous effectuions nos propres matchs contre des camarades pour tester nos équipes en conditions réelles et prendre note de ce qui devait être améliorer. Cela nous à permis de corriger les éventuels défauts de nos stratégies.

Analyses

Résultats

C:\Users\JESSY - LVS\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.Word\Resultats.png

C:\Users\JESSY - LVS\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.Word\Resultats.png

Au vu de nos résultats, on peut considérer que notre stratégie en 1 contre 1 est efficace mais que celle du « tout pour l’attaque » en 2 contre 2, est perfectible.

Conclusion

En dernier lieu, on a pu constaté, grâce à cette UE, la difficulté de mener un projet dans son intégralité: que ce soit au niveau du travail en équipe ou encore au niveau de la réalisation du projet en lui même. Cependant, nous avons apprécier ce travail d’équipe car cela rends la conception plus intéressante tout comme le classement hebdomadaire qui entretenait notre esprit de compétition et notre envie d’améliorer nos stratégies. En définitive, cette expérience qui nous à familiariser avec le domaine de l’IA et la découverte de la plateforme de partage Github, ne pourrons que nous être bénéfiques pour nos projets futurs en Intelligence Artificielle.