

MOSIMA
Rapport du projet Duel

Gaspard Ducamp (3200233)
Yoann Taillé (3200171)

UPMC - 2017

Table des matières

1	Introduction	3
2	Agent	3
2.1	Description et attributs	3
2.2	Comportements	3
2.2.1	FelixBaumgartnerBehaviour	3
2.2.2	BonnieTylerBehaviour	4
2.2.3	CliffhangerBehaviour	4
2.2.4	FollowIfVisibleBehaviour	5
2.2.5	NotSoLuckyLukeBehaviour	5
2.2.6	RainManBehaviour	5
2.2.7	RondeBehaviour	6
3	Apprentissage	7
3.1	De l'utilisation de Weka	7
4	Prolog et épilogue	8
4.1	Prolog	8
4.2	Améliorations	8
5	Annexe	9

1 Introduction

Dans ce projet deux agents apparaissent sur une même carte. Un agent a pour but de tuer l'autre en lui tirant dessus. Il convient d'implémenter différents comportements afin d'atteindre cet objectif et d'apprendre de ses expériences passées afin de devenir le meilleur chasseur.

2 Agent

2.1 Description et attributs

Notre agent, fièrement nommé d'après le meilleur ancien assassin de la CIA, hérite des caractéristiques d'un `AbstractAgent`. Il faut noter que nous avons modifié ses attributs afin qu'il soit en mesure de mémoriser différentes situations vécues, une sorte d'anamnèse de ses déplacements, dont nous détaillerons l'intérêt dans la 3ème partie. On peut dire qu'il a "La mémoire dans la peau".

2.2 Comportements

Notre agent pourra être augmenté de plusieurs comportements, certains seront concurrents, d'autres s'enchaîneront séquentiellement. Un graphe récapitulatif est disponible en annexe, chouette !

2.2.1 `FelixBaumgartnerBehaviour`

"Il faut bien que nous vivions malgré la chute de tant de cieux."
– David Herbert Lawrence, *L'amant de Lady Chatterley*

Ce comportement, ajouté lors de l'initialisation de l'agent, permet de vérifier que celui-ci n'est pas en train de tomber (ce qui est le cas lorsqu'il "entre en jeu"). Ce faisant on l'empêchera de "suivre" ou de tirer sur l'adversaire avant que les deux se soient effectivement posés sur la carte, on n'est pas dans Point Break.

Une fois que le différentiel d'altitude entre deux positions de notre agent sera nul on le considérera comme à terre, seront alors activés plusieurs nouveaux comportements, à savoir :

- `BonnieTylerBehaviour`
- `RainManBehaviour`
- `NotSoLuckyLukeBehaviour`

Un déplacement est effectué au hasard, de manière à laisser à l'adversaire le temps de s'éloigner

2.2.2 BonnieTylerBehaviour

*“Turn around, every now and then I get a little bit terrified
And then I see the look in your eyes”*
– Bonnie Tyler, *Total Eclipse of the Heart*

Une fois ce comportement activé notre agent tournera sur lui même, comme une toupie, pour observer les différentes vues disponibles via un appel à LookAt avec pour paramètres les différents points cardinaux accessibles depuis la classe LegalAction.

Pour chaque nouvelle vue :

- Soit on repère l’ennemi, auquel cas on active le FollowIfVisibleBehaviour.
- Soit on regarde si un point de vue plus élevé que le nôtre est visible.

Une fois notre tour achevé nous activons soit le CliffhangerBehaviour afin de grimper sur le plus haut sommet vu, soit nous établissons un circuit via nos situations enregistrées au travers du RondeBehaviour.

2.2.3 CliffhangerBehaviour

“Puisque la montagne ne vient pas à nous, allons à la montagne”
– Mahomet

Nous y sommes, face à notre montagne, à la recherche d’un point de vue imprenable sur notre terrain de chasse.

Déterminés ? Certainement, nous avançons vers le sommet d’un pas sûr. Seuls ? Peut-être. Et nous le saurons bien assez tôt. Bien assez tôt car toutes les 3 secondes nous relancerons un BonnieTylerBehaviour, on n’est jamais trop prudent, le traqueur est vite devenu traqué par ici.

Au pire on découvrira une nouvelle destination, un nouvel objectif. Au mieux on entamera une course poursuite avec notre cible.

2.2.4 FollowIfVisibleBehaviour

*“Même si tu pars plus loin que ne portent mes yeux
Où tu iras je te suivrai,
Je te suivrai...”*
– Francis Cabrel, *Je te suivrai*

Lorsqu’il suit l’ennemi notre agent ne fait rien d’autre, il observe son environnement, cible sa proie, se dirige vers elle, machinalement. Mais, contrairement à Cabrel, il ne la suivra pas après l’avoir perdue de vue. Loin des yeux, loin du coeur, qu’on dit.

Déboussolé par cette poursuite haletante notre agent relancera un BonnieTylerBehaviour.

2.2.5 NotSoLuckyLukeBehaviour

“Quand tu dois tirer, tire, cause pas !”
– Sergio Leone, *Le bon, la Brute et le truand*

Pendant tout ce temps notre agent garde son arme contre lui, chargée. La règle est simple : voir l’ennemi à moins de 35 mètres -et le moteur d’inférence nous le dira- c’est faire feu, quitte à faire mouche.

2.2.6 RainManBehaviour

“Ne faut-il pas que chaque homme ait un endroit quelconque où il puisse aller ? Car il y a de ces moments où il faut absolument aller quelque part, n’importe où !”
– Dostoïevski, *Crime et châtiment*

N’importe où ? Vraiment ?

Grâce à ce comportement notre agent enregistrera des situations, après un certain pas de temps. Il fera de rapides tours sur lui-même pour observer ses différents panoramas et enregistrera le plus intéressant. Pour déterminer si une situation est intéressante nous aurons plusieurs méthodes, par exemple pondérer ses caractéristiques (le champ de vision, la hauteur moyenne, la profondeur de champ, ect) et, via une fonction d’évaluation, lui attribuera une fitness caractéristique (*float RainManBehaviour.evalSituation(Situation)*).

L’apprentissage de ces méta-paramètres pourrait-être intéressant, on ne nécessiterait pas de connaissance *a priori* sur la topologie de la carte employée.

Une autre façon d’évaluer une situation est d’utiliser un classifieur, ici un réseau bayésien grâce à Weka (*float RainManBehaviour.evalSituationWeka(Situation)*).

2.2.7 RondeBehaviour

Ce comportement, censé prendre avantage des résultats de l'apprentissage, n'est pas implémenté.

L'agent aurait été capable de récupérer les points d'intérêt déterminés (ou du moins les 20 premiers) et de faire le tour de ceux-ci.

On aurait pu imaginer qu'il change de cap à certains moments de manière à continuer son exploration.

3 Apprentissage

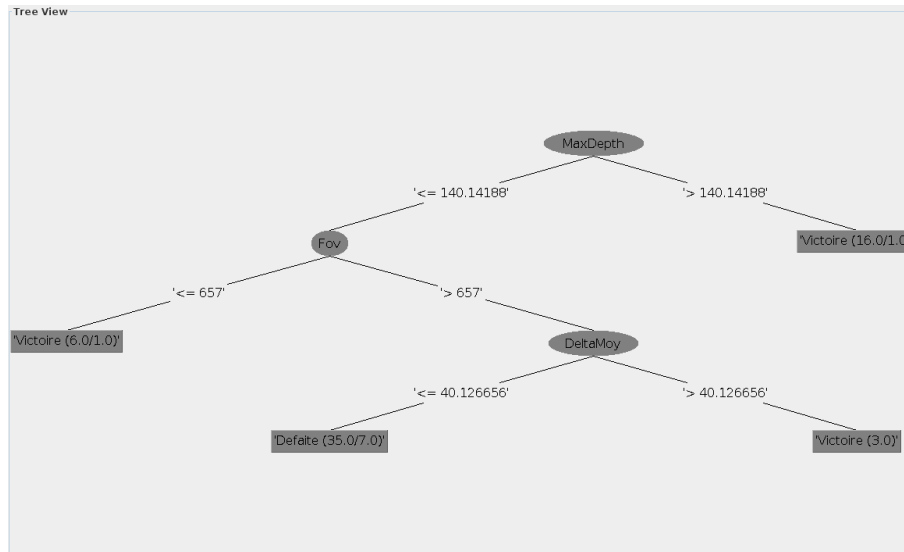
3.1 De l'utilisation de Weka

"Tsamina mina eh eh. Weka Weka eh eh."

– Shakira, *Waka Waka (This Time for Africa)*

Pour notre utilisation de Weka nous générons à la main une "base de données" de 60 situations de victoires ou de défaites. Présente dans le fichier "apprentissage3.csv" celle-ci dépend pour l'instant de la différence d'altitude entre notre position et la position moyenne ("sommes-nous plus sur un pic ou dans un creux?"), de la profondeur de vue, du champ de vision et de la différence d'altitude entre notre position et la position la plus haute ("sommes-nous dominés?"). Il pourrait-être intéressant d'utiliser d'autres paramètres, tels que la distance à l'ennemi ou une meilleure caractérisation de notre environnement.

Voici l'arbre produit par Weka (ici un J48, plus parlant qu'un RB) :



Voir "loin" serait donc un critère particulièrement important pour réussir à tuer l'agent.

Nous préférons utiliser un Bayesian Network (Réseau Bayésien) dans Weka afin d'avoir des probabilités sur les distributions des classes. Une instance aura par exemple une probabilité de 0.80085691337 d'être associée à une situation de victoire et ergo, 0.19914308663 d'être associée à une défaite. Nous utiliserons donc ce résultat de Weka pour évaluer les situations visitées (dans le RainMan-Behaviour).

4 Prolog et épilogue

4.1 Prolog

*“Exception in thread “main” java.lang.UnsatisfiedLinkError : no
jpl in java.library.path”*

– Eclipse

Comme des personnes parcourant notre projet auront pu le constater, le

Prolog n’est pas à son centre.

L’utilisation principale que nous en faisons intervient dans le cadre de la vérification de la position de l’ennemi. S’il est à une distance satisfaisante, notre agent n’aura cure de miséricorde et tirera.

Nous aurions pu utiliser celui-ci pour comparer deux instances de situations ou pour lancer un changement de comportement.

4.2 Améliorations

“C’est une perfection de n’aspirer point à être parfait.”

– Fénelon

Trois points nous semblent améliorables : l’implémentation du comportement de ronde, une utilisation peut-être plus poussée de Prolog et tenter d’apprendre sur les méta-paramètres pondérant les différentes caractéristiques des situations (voire mieux détailler celles-ci).

5 Annexe

