Vie Artificielle

NOM : NAIT-LARBI , AMRANE, NADJAR

Prénom : Takfarinas ,Lydia, Farid

Double-licence : Math-Informatique

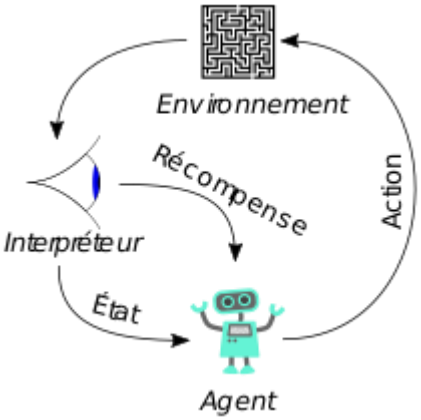
Groupe : 01

Chargé cours et TD : Mr. DECELLE Aurélien

Compte rendu du projet :

« Apprentissage par renforcement »

I. Introduction :



Apres avoir vu dans la première partie l'apprentissage supervisé dans ce projet on s'intéressera a l'apprentissage par renforcement.

En intelligence Artificielle, et plus précisément en apprentissage automatique, l'apprentissage par renforcement consiste, pour un agent autonome à apprendre les actions, à apprendre à partir de l'expérience de façon à optimiser une récompense quantitative au cours du temps .L'agent est plongé au sein d'un environnement, et prend ses décisions en fonction de son état courant, en retour l'environnement procure à l'agent une récompense qui peut être soit positive ou négative.

II. Structure du projet :

Notre projet se décompose en deux Package différents :

1. Package Bellman :

Dans ce package on s'intéressera à la résolution de l'équation de Bellman d'une façon exacte (Matrice), et d'une façon itérative. Dans ce package on a deux classe :

- **GridWord_Sql :**

  Dans cette classe on trouve le code et l'algorithme de la résolution de l'équation de Bellman .Les fonctions principales de cette classe sont :

  - InitRdmPol : qui initialise la politique aléatoirement

- IniTransitionMatrice : qui donne la probabilité d'arriver dans l'état S' en partant de l'état Sen choisissant l'action A.
- Iterative : cette fonction calcule V (Pi) d'une façon itérative
- ImprouvePlicy : Dans cette fonction, on calcule une politique déterministes, pour chaque état on met une probabilité égale à 1 pour la meilleures action et une probabilité égale à zéro sur le restes des actions

A la fin de cette classe on trouve un petit exemple d'application (exemple TP 02) ou on a affiché les récompenses de la Grille et V(Pi) calculé avec les deux méthodes (exacte et itérative)

- **GridWord :**

  Cette classe est l'exemple d'application (exemple donnée dans le projet) dans cet exemple on inclut les murs dans notre grille ou on a affiché les récompenses de la de la Grille et V(Pi) calculé avec les deux méthodes (exacte et itérative), ce qui nous permet de comparer les deux méthodes.

***Remarque** : on remarque que le calcule exacte et itérative de V(Pi) donne le même résultat*

2. Package RL :

Dans ce package on s'intéressera aux jeux de pack man, on a utilisé le squelette fourni dans le sujet et on a modifié deux classe à savoir :

- Qlearn : Dans cette classe on a implémenté deux sous-fonctions :
  - Intialize : qui initialise Q(S, A)
  - Argmax : qui renvoie la meilleure action pour un état S donné

Et deux fonctions principales à savoir :

- Qlearning () : implémente la stratégie « Qlearning »
- Sarsa() : implémente la stratégie « Sarsa »

- Pacman : Dans cette classe on a modifié une principale fonction qui est Update(), ou on a mis à jour Q(S,A) à chaque itération , et on choisis une action suivant la Stratégie E-Greedy, et aussi on a mis l'attribut use_ia à « Vrai »afin d'utiliser l'intelligence Artificielle.

III. Conclusion :

L'apprentissage par renforcement est paradigme d'apprentissage qui peut être une solution à plusieurs problèmes, à condition de l'adapter aux contraintes liées au cadre d'application

Pour conclure, ce deuxième projet il nous a permet d'appliquer ce qu'on a vu dans le deuxième chapitre de Vie Artificielle dont l'objectif est d'apprendre aux agents autonome (par exemple robots) des actions (en s'inspirant des systèmes vivants comme le montre l'image) .