

## Rapport LI 351

### I – Algorithme

#### 1 - Recherche Aléatoire

La recherche du meilleur perceptron dans l'algorithme aléatoire passe par l'évaluation d'une grosse quantité de perceptron pour à la fin rendre celui qui rend le meilleur score.

#### 2 - FittedQ

L'algorithme étudie les différents cas qu'il peut rencontrer et en fonction des résultats (gain / perte de point), il ajuste ses paramètres pour par la suite prendre la meilleurs option.

#### 3 - Perceptron T

Ce perceptron se sert de l'algorithme fittedQ comme base de départ puis avec un panel de test il confronte ses paramètres de base (ceux de fittedQ) avec ceux de son panel de test et choisi celui qui rend le meilleur comportement.

### II – Le fonctionnement

Le pacman regarde ce qu'il se passe dans les cases autour avec le sensor et obtient une suite de valeur 0 ou 1 (suivant la présence d'élément), ensuite il applique un certain coefficient à chaque valeur. Puis il choisit la direction ou le score rendu est le plus élevé. Le score pour chaque case est donné par le reward, il associe un score à chaque état : mur / nourriture / capsule ou bien victoire ou défaite.

### III – Les différents tests

Le reward de base -100 | 100 | 0 | -1 | 0 (défaite | victoire | nourriture | rien | capsule) ne me donne pas de résultat très satisfaisant, le pacman évolue sur des cases vides et ne cherche pas à éviter les fantômes.

Je suis passé à un reward de -1000 | 100 | 10 | -1 | 25, le pacman a tendance à manger un petit peu plus de nourriture et à esquiver un fantôme de temps à autre.

Globalement que je sois en aléatoire, en fittedQ ou avec le perceptron T, il n'y a pas de nette différence, j'ai l'impression que mon pacman n'apprends rien ou vraiment pas grand-chose.

J'ai réalisé des essais avec de nombreuses itérations et/ou avec des panels très grands pour arriver à un agent fixe ou suicidaire (j'ai l'impression qu'il finit par croire que quitte à mourir un jour autant

mourir dès le départ).

Je n'ai d'ailleurs jamais eu de score positif signe qu'il aurait gagné la partie.

#### IV – Résultats

Mon pacman refusant de fonctionner comme il devrait et n'arrivant pas à trouver l'erreur dans le code, je n'ai que de piètres résultats.

En imaginant un pacman qui fonctionne, je pense qu'un des soucis qu'il peut exister est lorsqu'on permet au pacman de voir assez loin autour de lui, il doit finir par faire des aller-retour sur une cases d'à côté de peur de se mettre en danger en allant plus loin et il faudrait dans ce cas pondéré les scores des cases en fonction de leur éloignement par rapport au pacman.