

Projet apprentissage par renforcement - Snake

Exercice 1 : implémentation de la stratégie Tabular Q-learning

Voici les scores obtenus pour des niveaux de différentes tailles avec un seul Snake.

	Very Small No Wall alone			Very Small alone			Small Alone		
	Temps (sec)								
	5	20	60	5	20	60	5	20	60
Test	5.01	5.82	6.81	2.81	4.09	5.30	1.33	2.27	3.22
Train	4.72	5.46	6.05	1.62	2.24	2.57	1.01	1.76	2.40
Augmentation	Test : 1.80 Train : 1.33			Test : 2.49 Train : 0.95			Test : 1.89 Train : 1.39		

On observe qu'au fil du temps le score augmente. La moyenne des scores en phase d'entraînement est inférieure à celles en phase de test. Lors de la phase d'apprentissage, l'agent explore différentes stratégies et essaie d'apprendre à maximiser le score. L'agent peut donc être amené à prendre des décisions aléatoires ou expérimenter des stratégies qui ne sont pas optimales. Lors du test, l'agent utilise la stratégie qu'il a apprise pendant la phase d'apprentissage et doit être plus performant, car il utilise la stratégie optimale qu'il a apprise.

L'algorithme semble efficace pour des niveaux avec des petits plateaux. En effet, la phase de test voit son score augmenter d'environ 2,5 pour un petit plateau contre 1,8 pour un grand plateau.

Exercice 2 : implémentation de la stratégie Approximate Q-learning

Afin d'encoder un couple (état, action), j'ai implémenter trois features:

- La distance entre la tête du snake et la pomme.
- La distance entre la tête du snake et son corps le plus proche.

Les distances sont calculé en utilisé un algorithme A*. Une troisième features pourrait être ajouter, la distance entre la tête du snake et le mur le plus proche. Cependant l'algorithme était moins performant.

Voici les scores obtenus pour des niveaux de différentes tailles avec un seul Snake.

	Very Small No Wall alone			Very Small alone			Small No Wall Alone		
	Temps (sec)								
	5	20	60	5	20	60	5	20	60
Test	5.82	6.10	6.32	2.03	2.17	2.54	-	5.87	9.97
Train	5.09	5.21	5.75	1.25	0.96	1.00	-	3.14	5.80
Augmentation	Test : 0.50 Train : 0.66			Test : 0.51 Train : -0,25			Test : 4.10 Train : 2.66		

Pour les très petits niveaux, le score augmente au fil du temps durant l'apprentissage. Par rapport à l'exercice 1 on obtient des scores plus petits. Les limites de cet algorithme sont le nombre de features. Plus il y en a, plus l'algorithme prendra du temps à trouver une stratégie optimale.

Exercice 3 : implémentation de la stratégie Tabular Qlearning duel

Voici les résultats obtenus en donnant la stratégie *Tabular Qlearning Duel* à l'un des serpents et la stratégie *Strategy Random* à l'autre serpent et en mode test.

	Very Small No Wall Fair Duel			Small Arena No Wall			Arena No Wall		
	Temps (sec)								
	5	20	60	5	20	60	5	20	60
Tabular Qlearning Duel (1)	11.77	12.00	13.24	10.65	10.78	12.61	2.79	6.27	7.20
Strategy Random (2)	2.59	5.13	1.84	3.03	2.64	1.33	3.87	1.97	1.00
Augmentation	Stratégie (1) : 1.47 Stratégie (2) : -0.75			Stratégie (1) : 1.96 Stratégie (2) : -1.70			Stratégie (1) : 4.41 Stratégie (2) : -2,8		

La différence réside dans la méthode encode State qui prend en compte le snake ennemi. Les caractères utilisés sont V (vide), H (tête), B (corps) et A (pomme) auxquels on ajoute E (ennemi de taille supérieure ou égal) ou R (ennemi de taille inférieure) On observe que le score augmente grandement au cours du temps et que l' algorithme est efficace sur les petites cartse.

Voici les résultats obtenus en mode test en donnant la stratégie *Tabular Qlearning Duel* aux deux serpents sur les niveaux *Very Small No Wall Fair Duel* et *Very Small No Wall Unfair Duel* et en prenant en compte le champ *Random First Apple*.

		Very Small No Wall Fair Duel			Very Small No Wall Unfair Duel		
		Temps (sec)					
		5	20	60	5	20	60
Random first apple: TRUE	Serpent (1)	11.05	11.57	12.07	11.54	12.06	12.47
	Serpent (2)	5.07	3.39	4.11	5.70	4.33	4.03
Random first apple: FALSE	Serpent (3)	12.83	13.56	14.44	12.37	13.67	13.42
	Serpent (4)	1.99	1.75	0.92	2.65	1.55	1.64
Augmentation		Serpent (1) : 1.02 Serpent (2): -0.96 Serpent (3): 1.61 Serpent (4): -1.07			Serpent (1) : 0.93 Serpent (2): -1.67 Serpent (3): 1.05 Serpent (4): -1.01		

Les scores du premier serpent sont plus élevés avec random first apple a true et sur la carte: Very Small No Wall Fair Duel

Exercice 4 : implémentation de la stratégie Approximate Qlearning duel

Afin d'encoder un couple (état, action), j'ai implémenter une feature supplémentaire par rapport à l'exercice 2:

- La distance entre la tête du snake et le corps ennemi le plus proche en utilisant un algorithme A*.

Voici les résultats obtenus en donnant la stratégie *Tabular Qlearning Duel* à l'un des serpents et la stratégie *Approximate Qlearning Duel* à l'autre serpent et le champ *Random First Apple* à true .

	Very Small No Wall Fair Duel			Small Arena No Wall			Arena No Wall		
	Temps (sec)								
	5	20	60	60	120	180	60	120	180
Approximate Qlearning	11.99	11.50	12.13	7.60	6.80	7.03	2.68	3.22	2.72
Tabular Qlearning	4.17	4.06	3.66	7.80	6.97	7.18	1.99	3.11	3.15
Augmentation	Approximate : 0.14 Tabular : -0.51			Approximate : -0.57 Tabular : -0.62			Approximate : 0.4 Tabular : 1.16		

Pour les petites cartes les scores de l'approximate sont très élevés mais très faibles pour les grandes cartes. Le tabular a des scores faibles pour les deux types de carte. L'algorithme Approximate voit sa stratégie évoluer au fur et à mesure des parties et tente de manger le snake ennemi le plus rapidement possible.