TP IAR
Reinforcement Learning

ARBERET Antonin et SCHOTT Lucas

Sorbonne Université Sciences

Pour régler les soucis d'importations, les classes et méthodes à importer sont dans des fichiers .py.
Les tests de convergence sont effectués sur l'instance suivante :

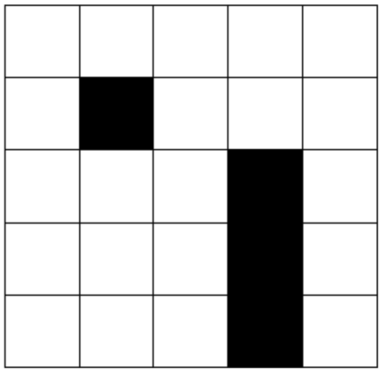

Fig. 1. Caption

1 MDPs and mazes

La version stochastique de mdp se trouve dans le fichier mdp_stoch_trans.py.

2 Dynamic Programming

2.1 Policy Iteration with the Q function

Why do we have to loop twice over states in the former?

On boucle sur le etats x, puis sur chaque état potentiellement accessible depuis x par une action.

Convergence

	V function	Q function
Value Iteration	17	18
Policy Iteration	14	16

Les valeurs sont assez proches les unes des autres, on note tout de même une léger avantage en terme de convergence de V function sur Q function et de Policy Iteration sur Value Iteration sur cette instance.

Voici la politique et la valeur Q calculée par Value iteration sur les Q :

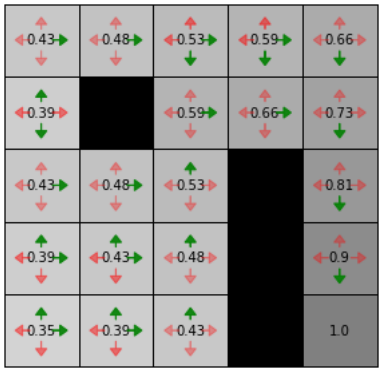

Fig. 2. Caption

3 (Model-Free) Reinforcement Learning

Ici (et par la suite) nous évaluons la vitesse de convergence de nos algorithmes vers la valeur retourné par Value Iteration.

3.1 Convergence

	Softmax	Epsilon greedy
Q-Learning	172432	292511
SARSA	-	-

SARSA ne converge pas suffisamment vite pour pouvoir l'observer, quelque soit les valeurs de epsilon ou de tau choisis. En outre, Q-Learning converge en environ 172432 itérations avec les softmax. La version epsilon greedy de Q-Learning en demande 292511 avec epsilon élevé (0.5) ce qui n'est pas surprenant puisque si on ne pousse pas l'algorithme à explorer beaucoup il risque de s'enfermer dans des solutions localement optimales. Mais si l'on baisse epsilon alors il n'explore pas assez, mettant trop peu à jour les valeurs Q[x,u] faible, et donc celles-ci ne convergent pas vers leur valeur réel, en ravanche avec un epsilon plus faible la politique converge bien plus rapidement.

4 Model based reinforcement Learning

	Deterministic rewards	Stochastic rewards
Q-Learning	2437	2502

On note que l'approche model based semble nettement plus performante que l'approche model free dans ce cas particulier, car le model est très facile à apprendre. Pour la version avec des reward stochastique l'algorithme est tout aussi performant on obtient en moyenne les mêmes performances.

5 On-policy, off-policy

5.1 Convergence

Q-learning	SARSA
132365	-

Q-learning converge alors que SARSA ne converge pas. Ce qui fait sense car SARSA est on-policy et ne peut donc pas apprendre sur des etats-actions samplé uniformément.

6 N-step Q-learning

N-step Q-learning	29447

On observe que N-step Q learning converse extremement rapidement

7 Actor-Critic

7.1 Convergence

Q-value	4214
V-value	-