强化学习 HW3

— 饶翔云 520030910366

Problem 1

Proof:

$E_{\pi_\theta}[\frac{\partial \log \pi_\theta(a|s)}{\partial \theta} f(s)] = \sum_{s\in S} d(s) \sum_{a\in A} \frac{\partial \pi_\theta(a|s)}{\partial \theta} = \sum_{s\in S} \rho^{\pi_\theta}(s) \sum_{a\in A} \pi_\theta(a|s) \frac{\partial \log \pi_\theta(a|s)}{\partial \theta} f(s)$

$\because$ 可生成ρ的唯一策略是$\pi_\rho(a|s) = \frac{\rho(s,a)}{\sum_{a'} \rho(s,a')}$，而且$\rho(s) = \sum_{a'} \rho(s,a')$ (占用度量第二定理)

$\therefore E_{\pi_\theta}[\frac{\partial \log \pi_\theta(a|s)}{\partial \theta} f(s)] = \sum_{s\in S} \rho(s) \sum_{a\in A} \frac{\rho(s,a)}{\sum_{a'} \rho(s,a')} \frac{\partial \log \frac{\rho(s,a)}{\sum_{a'} \rho(s,a')}}{\partial \theta} f(s)$

$= \sum_{s\in S} \sum_{a\in A} \rho(s,a) \frac{\partial \log \frac{\rho(s,a)}{\sum_{a'} \rho(s,a')}}{\partial \theta} f(s)$

$= \sum_{s\in S} \sum_{a\in A} \rho(s,a) \frac{\partial \log \rho(s,a)}{\partial \theta} f(s) - \sum_{s\in S} \sum_{a\in A} \rho(s,a) \frac{\partial \log \sum_{a'} \rho(s,a')}{\partial \theta} f(s)$

$= \sum_{s\in S} \sum_{a\in A} \frac{\partial \rho(s,a)}{\partial \theta} f(s) - \sum_{s\in S} (\sum_{a\in A} \rho(s,a)) \frac{\partial \log \sum_{a'} \rho(s,a')}{\partial \theta} f(s)$

$= \sum_{s\in S} \sum_{a\in A} \frac{\partial \rho(s,a)}{\partial \theta} f(s) - \sum_{s\in S} \frac{\partial \sum_{a'} \rho(s,a')}{\partial \theta} f(s)$

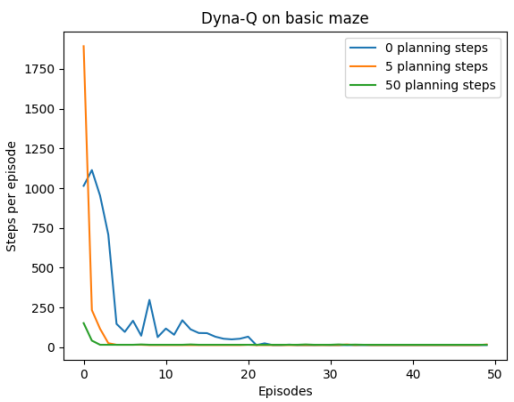
$= \sum_{s\in S} \sum_{a\in A} \frac{\partial \rho(s,a)}{\partial \theta} f(s) - \sum_{s\in S} \sum_{a'} \frac{\partial \rho(s,a')}{\partial \theta} f(s)$

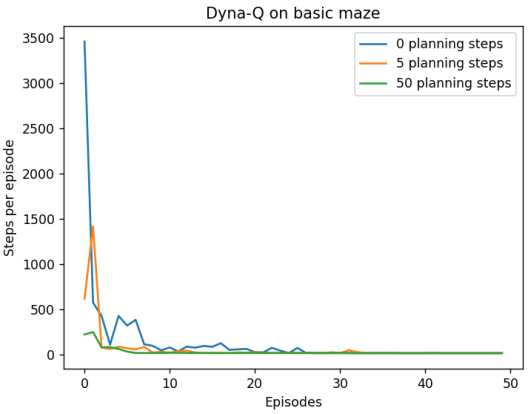
$= 0$

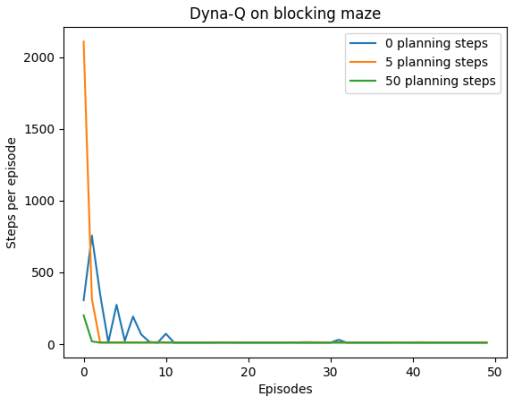
Problem 2

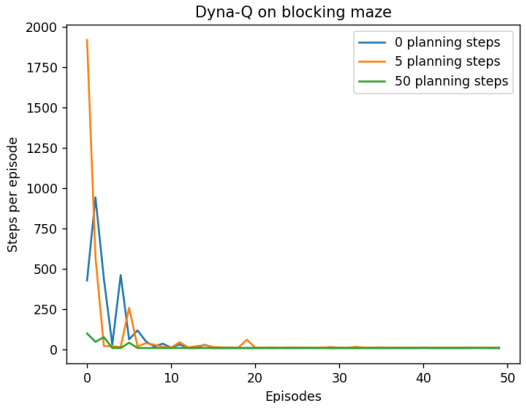
(a) What are the impacts of the number of planning steps on the performances of algorithms and what is the reason?

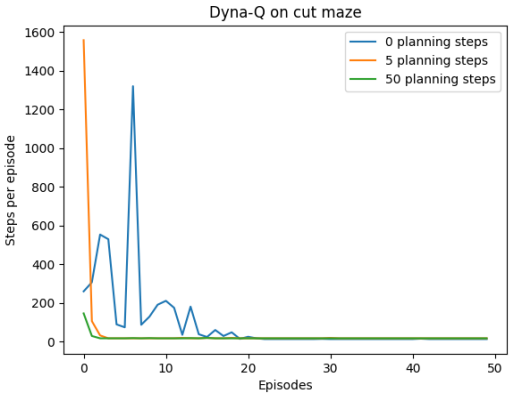
首先先把图贴在这（type：p1）

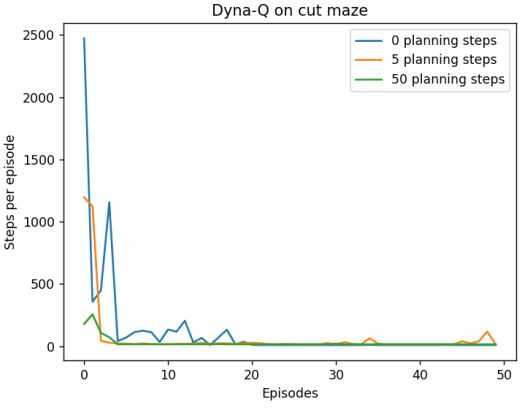










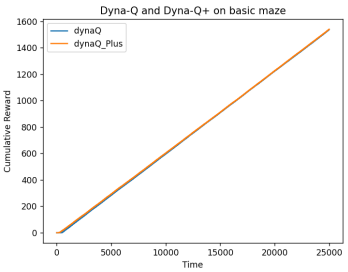


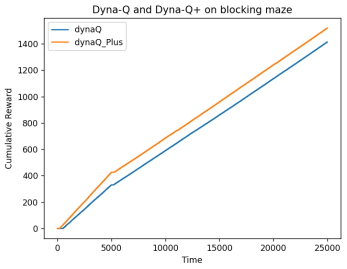
可以从图中看出，当planning step小的时候，在最初的几轮迭代中需要较多步数才能到达终点，即agent的行动具有较强的随机性，而当planning step多了之后，只需要几轮迭代就能很快的让agent找到通往终点的最短路径。

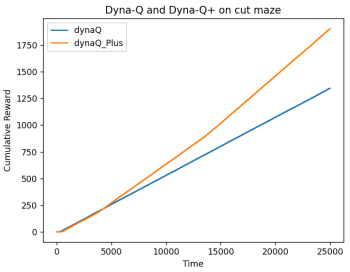
这是因为planning step的设计意图就是让agent不仅能够从当前状态中学习更新Qtable，还能通过充分利用之前经历过的状态和行动来学习更新Qtable。planning step越多，相当于agent利用过去的经历越充分，从而达到更快的收敛。而且由于DynaQ+的鼓励探索的特性。它收敛的比DynaQ快得多。

(b) What are the differences between the performance of Dyna-Q and that of Dyna-Q+ in three environments? Please discuss the reason for these differences.

图见下，从左到右依次是base，blocking和cut。（type：p2）







由于base中环境不会发生变化，所以在找到终点后，DynaQ和DynaQ+策略的agent都基本会按找好的路径进行运动，所以斜率相对稳定。

在blocking中，环境在第5000步后发生变化，原来的路被截断，留下了一条新的路，而且比原来更长。这点从DynaQ策略的agent的cumulative reward曲线斜率在5000步后明显变小可见一斑。但拉大后可以发现，DyanQ+的agent可以更早的发现新路径，这和他鼓励探索的reward公式有很大关系。

在cut中，环境在第3000步后发生变化，原来的路还在，但是出现了一条捷径。由于DynaQ+策略的agent鼓励探索的特性，它可以很轻易的发现哪里是捷径，并很痛快的采用了这条捷径。而反观DynaQ策略的agent，由于它已经找到了一条路，再加上ϵ-greedy的低随机性，它很难发现捷径，至少在666作为random.seed的时候，在25000步内，DynaQ策略的agent并不能找到捷径。所以在图中，会出现DynaQ+策略的agent的cumulative reward曲线斜率先是小的，然后变得比DynaQ策略的agent的cumulative reward曲线大，且越来越大，并最终DynaQ+策略的agent的cumulative reward比DynaQ策略的agent的大。