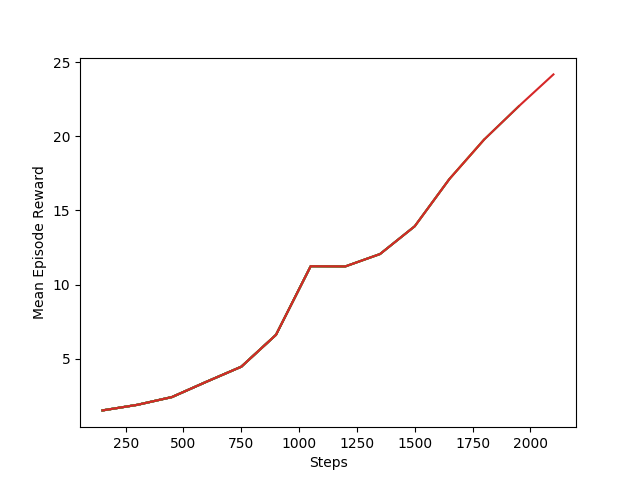
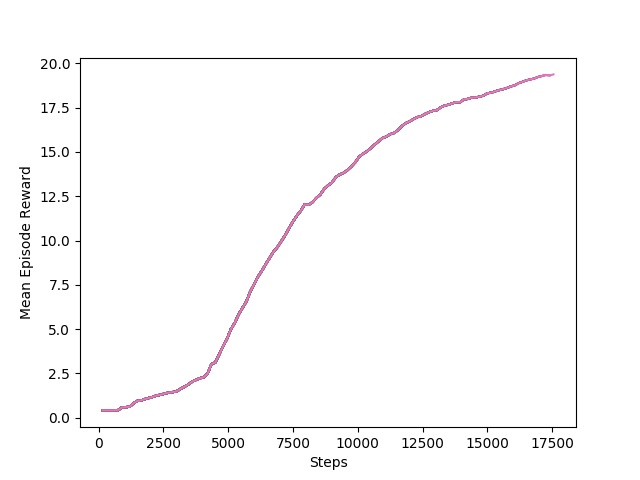
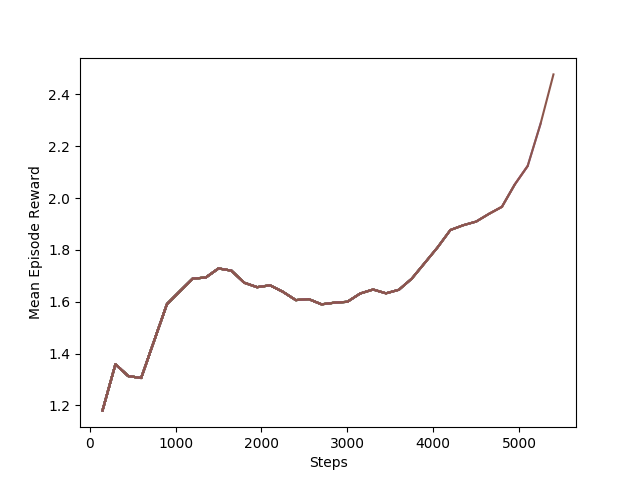
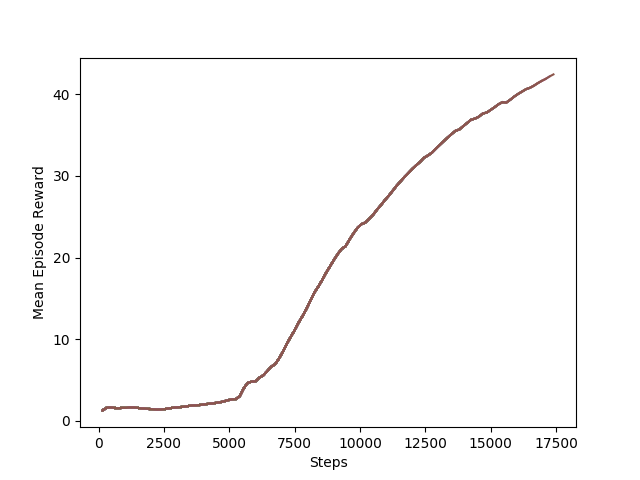
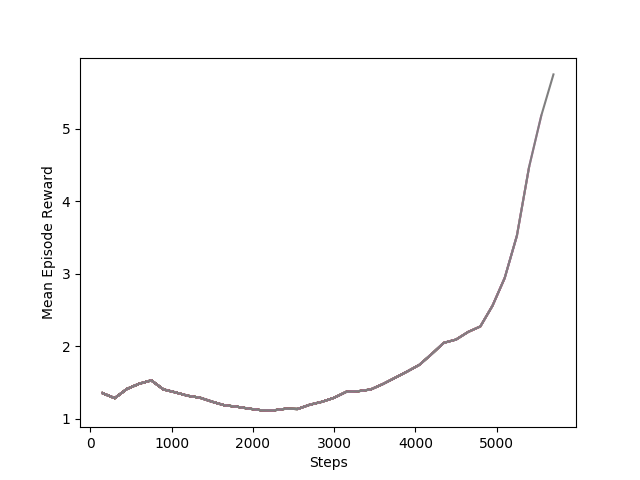
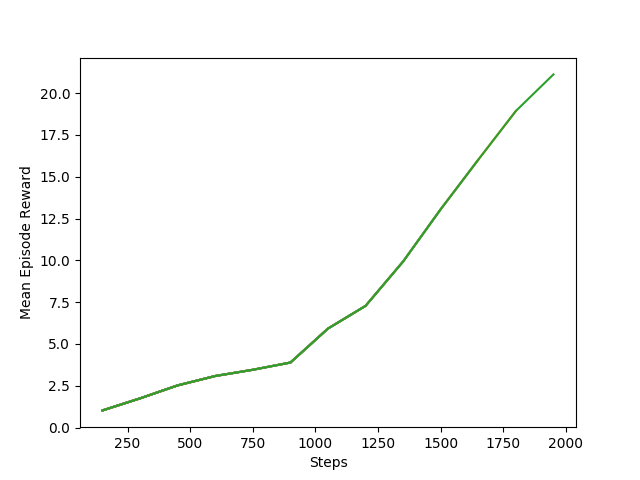
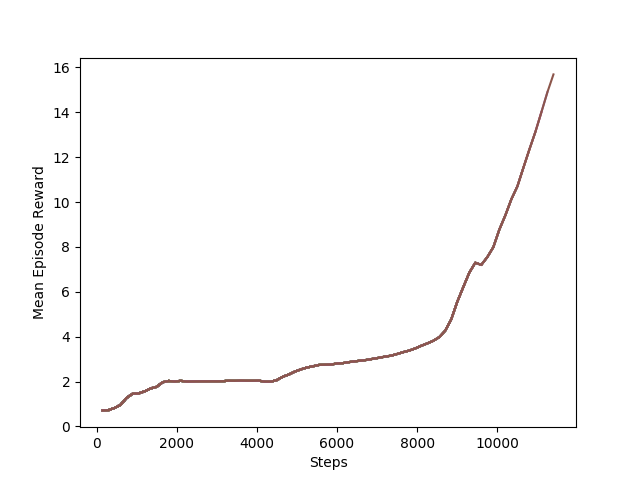
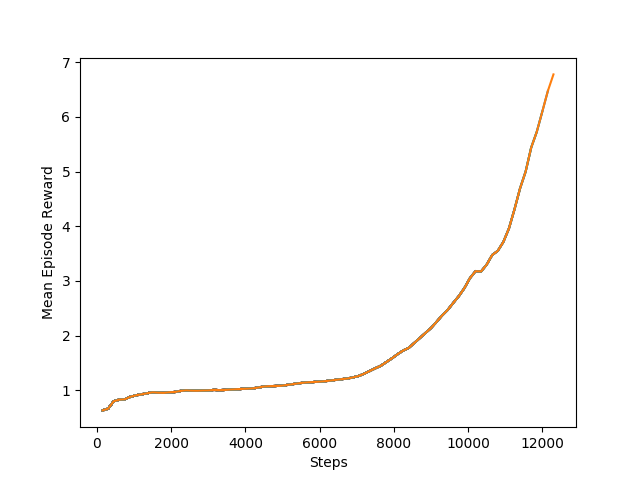
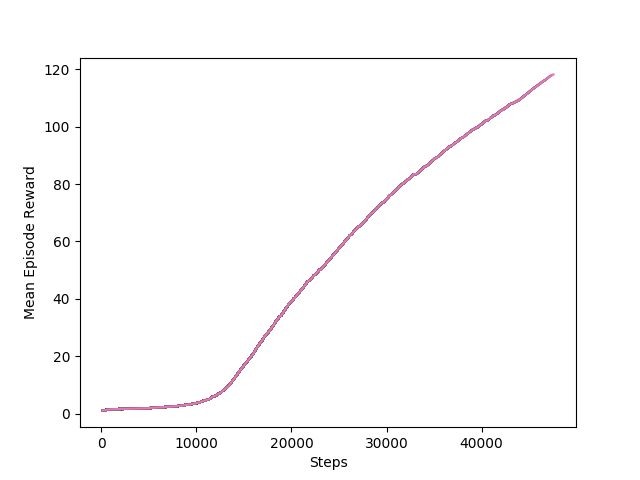
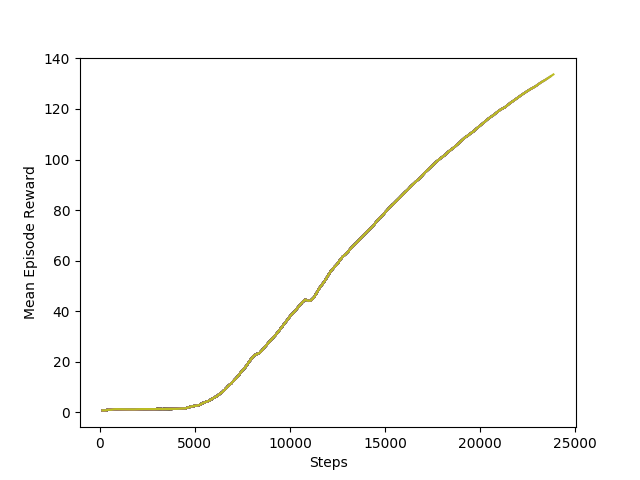
Comparisons of RL for *reacher* with initialized and non-initialized policy

Below are 5 pairs of test (same settings) with different time scales,on the left is with initialization policy from inverse kinematics (just called initialized) and on the right is with general random initialization (just called non-initialized). Generally initialized Reinforcement Learning surely performs better than non-initialized one.









One worth noticing point is that the initialized RL doesn’t show a significant better performance than non-initialized one in the starting stage in training, like first 300 steps. This does not meet my expectation quite much. I guess the the initialized policy only improves the performance a little bit at the beginning, but the effects accumulate after a long run.

I think the limit of improvement with initialized policy for RL mainly lies in the accuracy of supervised learning policy. In my test, the accuracy of supervised learning policy is not very high through my observation, as it needs to generate to all possible goal positions.

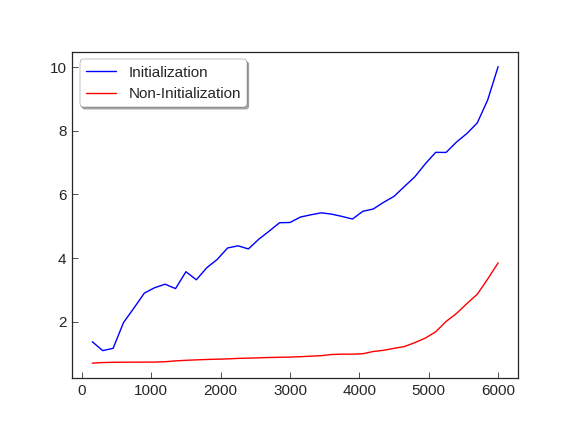
1.11:

In order to improve supervised learning accuracy, use more samples and larger neural network for both supervised learning and actor network in RL: 5 layers of 1000 nodes each, with relu as hidden layer activation and leakyrelu as output layer activation (range from -inf to +inf, not like relu only 0 to +inf and tanh between +-1).

Comparison of training with and without initialization is as follow. The point that start reward of initialization is larger than non-initialization is important, which testifies the effects of initialization with supervised learning policy.

However, with smaller network for supervised learning (with inefficient activation like tanh), the difference of starting is not obvious, which makes the improvement of initialization to non-initialization only work in stochastic meanings, i.e not guarantee every initialization is better than non-initialized learning.

Larger network: 4\*1000+1\*1000, relu, leakyrelu



Smaller network: 2\*200+1\*400, mainly tanh

