Learning Algorithm

Deterministic Deep Policy Gradient (DDPG) is based on actor-critic method that as Figure 1 shows below where actor provide the best action to take while critic provide the best policy through update for actor to feedback on [1].

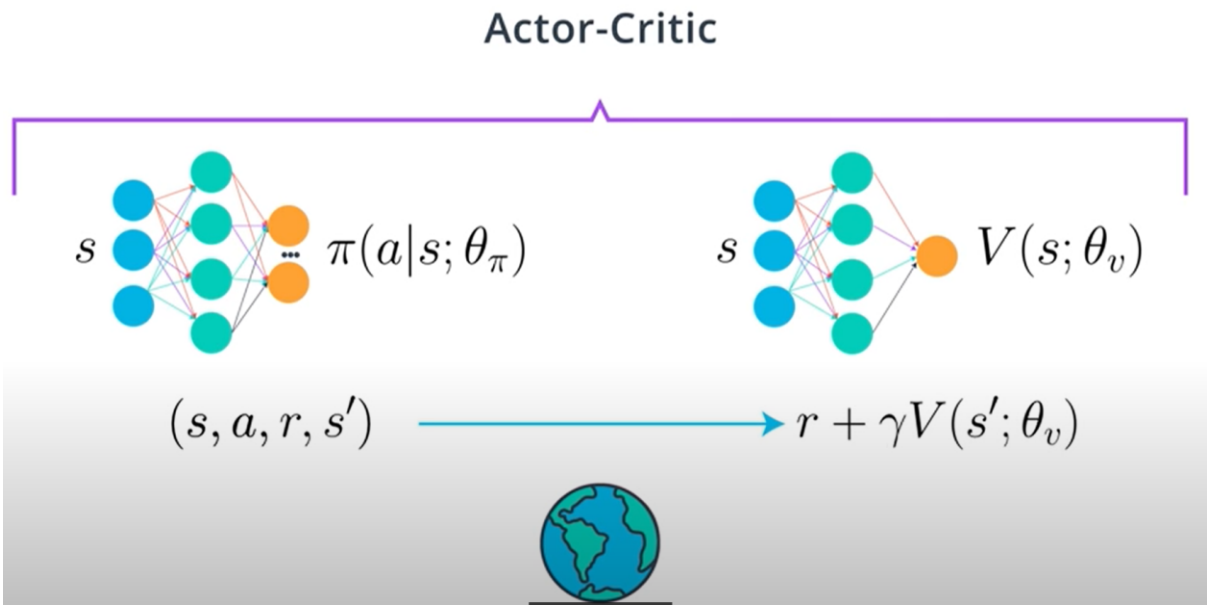


Figure 1 Actor-critic method for DDPG [Source: Udacity Deep Reinforcement Learning Course]

The network update run on the same base as DQN where soft update will be updating the target network from the local/regular network every single step by a small percentage. Beside that, replay buffer from DQN has been carried forward into DDPG as well.

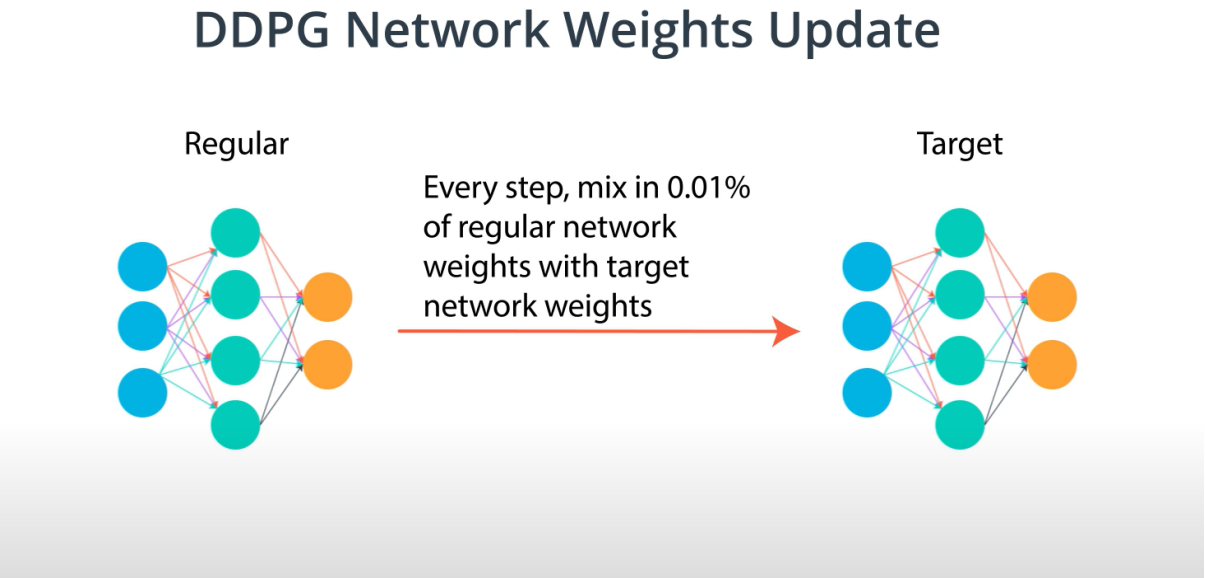


Figure 2 DDPG soft update mechanism

	Actor			Critic		
	Activation	**In node**	**Out Node**	**Activation**	**In node**	**Out Node**
Input Layer	**Relu**	33	256	**Relu**	33	260
Hidden Layer	**Relu**	256	128	**Relu**	260	128
Output Layer	**Tanh**	128	4	**Sigmoid**	128	1

Table 1 Continuous control network configuration

DDPG Hyperparameters	
Batch size	1000000
Buffer size	1000000
gamma	0.99
actor learning rate	0.001
soft update	0.001
critic learning rate	0.001
noise decay rate	0.999

Table 2 DDPG Hyperparameters

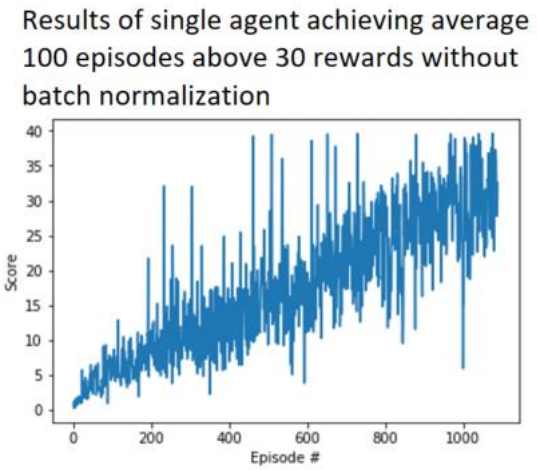
Table 1 network setting for the actor and critic network configuration, where for critic input layer batch normalization is used to speed up training. Table 2 is overall DDPG hyperparameters that was used in setting up the continuous control environment.

Results

Different setting on the network for agent reward behaviour experiment has been carrier out.

1. Learning with actor critic of 400 nodes of first hidden layer and 300 nodes of second hidden layer.

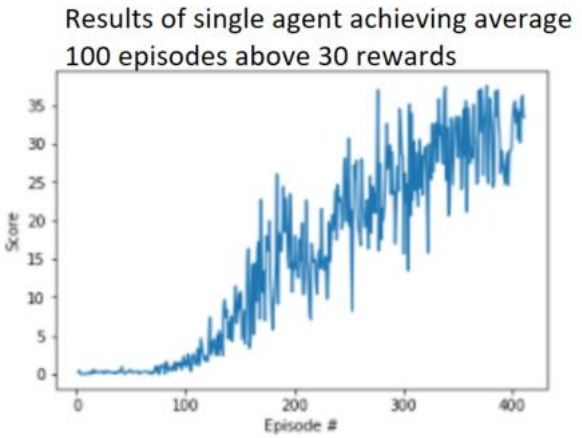
   Experiment result:

   Agent able to achieve +30 rewards average rewards over 1200 episodes with significant instability observed

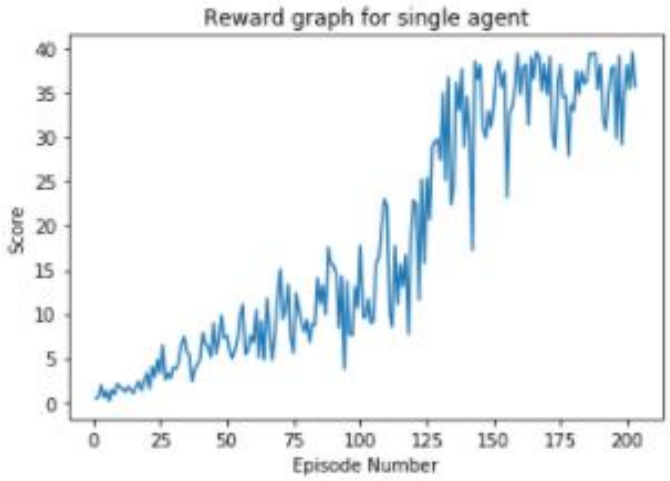
2. Learning with actor critic of 400 nodes of first hidden layer and 300 nodes of second hidden layer. Additional setting is by adding batch normalization at the output of first layer.

   Experiment result:

   Agent able to achieve +30 rewards average over 400 epidoes with significant improve in stability. Stability can be observed over sudden spike in score happened before achieving the objective score of +30 rewards average.

3. Final setting is having batch normalization with reduced number of nodes on the network. First hidden layer was set to have 256 nodes, second hidden layer was set to have 128 nodes.



Experiment result:

Agent able to achieve +30 rewards average within 200 episodes. Number of input and output nodes to hidden layer is crucial setting for agent performance as policy gradient is directly learning through the network that experiment is setting up.

Future Work

Using Distributed Distributional DDPG (D4PG) with parallel actors and distributional critic update as first upgrade to DDPG could further improve and speed up learning process of agent. With prioritized experience replay can further improve agent learning.

Prioritized experience replay could be first future work to replace current replay buffer for agent learning improvement in DDPG.

References

[1] Lillicrap, T.P., Hunt, J.J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D. and Wierstra, D., 2015. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*