2. a. Describe what Temporal Difference (TD) methods are. What are their potential

advantages over both Monte Carlo methods and policy and value iteration?

Learns models of reward and environment. Builds from bootstrapping and Monte Carlo methods. Uses Monte Carlo scheme for non-stationary environments using a bootstrapped estimate of the return which doesn't require a model of a complete episode.

Bootstrapping uses estimates of successor estimates of state and action values --> predicts future returns at a certain state to estimate the return.

Advantages over both

Monte Carlo and PI/VI have to have episodic tasks

- episodes can be very long, or just not episodic

...over stationary environments

- environments continuously change

New addition:

Temporal difference methods builds from bootstrapping and Monte Carlo methods. It uses the property of not needing to wait until a full episode is truncated through from bootstrapping, and it uses the property of not requiring a pre-specified model from Monte Carlo to learn a models environments and rewards.

From the Monte Carlo methods, to account for non-stationary environments, it updated the value function estimation from calculating the average of returns (Equation (1)), which is very unsensitive if changes to the property of the environment occurs when the number of states is large:

(1)

To a new value function estimation procedure (Equation (2)):

(2)

Where is the value function, is the return and is some constant scalar, denoting a step-size parameter between (0,1) to ensure the algorithm can converge. The can be treated as an error signal, which we want to minimize and this error is what makes the estimation of the value function react to changes in the system property as alpha, which is constant, keeps the value function fluctuating but also keeping to the trend of the optimal value function.

However from equation (2) is still dependent on , where the definition of and so it can be seen that the value function estimation is still reliant on knowing every return of future states, limiting Monte Carlo methods to only run episodic tasks. Bootstrapping is introduced to remove the need for these future terms Bootstrapping updates value estimates of states based off the estimate of future values of successor states.

The expected return is therefore replaced with:

(3)

Where is the estimated value of return, is the reward in the next step, is the bootstrap estimate of the value function at the next step .

As such, substituting in equation (3) into (2), we obtain equation (4) which is the temporal difference learning estimate, specifically, TD(0) also known as 1 step TD.

(4)

And so the key points are that the allows the temporal difference method to work with non-stationary systems and allows the system to be non-episodic.

Therefore, it is clear an advantage over Monte Carlo methods is that it does not need the environment to be episodic. Monte Carlo methods must have episodic tasks that run to a terminal state, however the issue would be that if episodes are extremely long, the time to run these episodes would take too long. Furthermore, some tasks may not be episodic at all and so Monte Carlo method would no be able to run to completion. Therefore Temporal Difference methods are able to overcome as they only need to wait until the next steps where would be self defined.

Its advantage over policy and value iteration is that as well as the advantages it achieves over Monte Carlo, it also doesn’t need a pre-specified model of the environment, or rewards as it learns through experience and repeating episodes over time, this means it can run in non-stationary environments as it supports online learning to continually learn about its environment properties.