An Introduction to reinforcement learning

**Reinforcement Learnings**

Reinforcement learning is learning what to do—how to map situations to actions—so

as to maximize a numerical reward signal.

## Markov Decision Process

RL problems can be mathematically formulated as a finite Markov Decision Process(MDP). This is one approach to formulate a reinforcement learning problem.

Finite MDPs can be solved by multiple methods: dynamic programming, Monte Carlo method, Temporal difference methods.

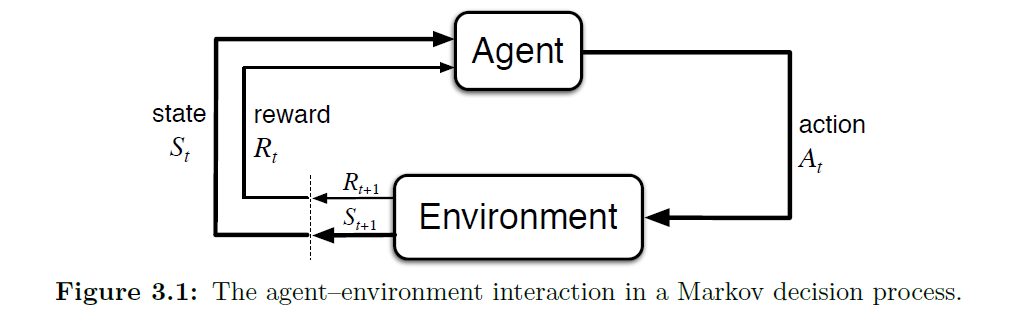
* **Agent**: The learner and decision maker is called the agent.  
  *Ex, a self-driving car, a house cleaning robot, etc.*
* **Environment**: Everything outside the agent is called the environment. It is the surroundings the Agent interacts with.  
  *Ex, road, warehouse, etc.*
* **State**: state as a signal conveying to the agent some sense of “how the environment

is” at a particular time.

*Ex, position/orientation of a robot, climate of a particular day, etc.*

* **Action**: It is the decision the Agent takes at a particular time.

*Ex, move forward, lift something, get back to the charging point, etc*



**Reward (Rt)**

* The numerical signal that the agent receives from the environment at each time step is called the reward.
* Agent’s goal is to maximize the total amount of reward it receives. This means maximizing not immediate reward, but cumulative reward in the long run.
* We must provide rewards to it in such a way that in maximizing them the agent will achieve the final goal.

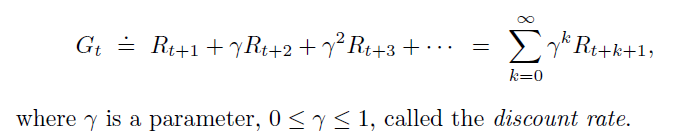
**Return (Gt)**

It is the total reward that the Agent receives over a long run.



**Discounting**

The agent tries to select actions so that the sum of the discounted rewards it receives over the future is maximized. In particular, it chooses At to maximize the expected discounted return

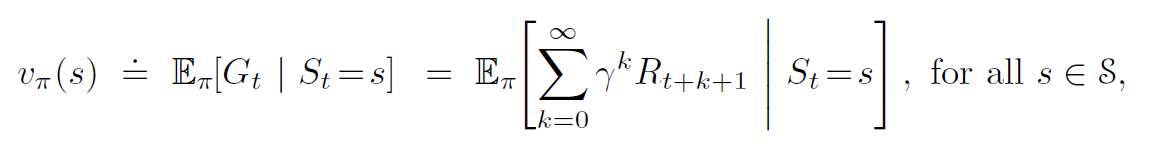
****

As 𝛾 approaches 1, the return objective takes future rewards into account more strongly; the agent becomes more farsighted.

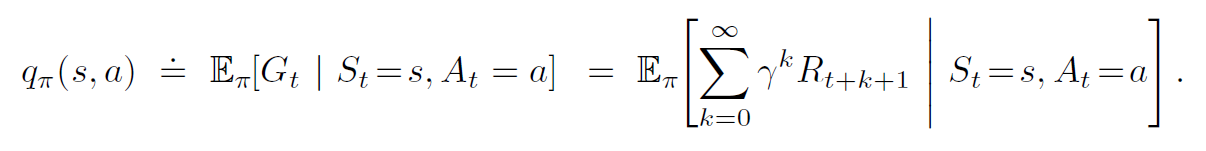
**Value functions**

functions of states (or of state–action pairs) that estimate how good it is for the agent to be in a given state (or how good it is to perform a given action in a given state).

* State value functions



* Action value functions



**Policy**

a policy is a mapping from states to probabilities of selecting each possible action.

If the agent is following policy at time t, then (a|s) is the probability that At = a if St = s.