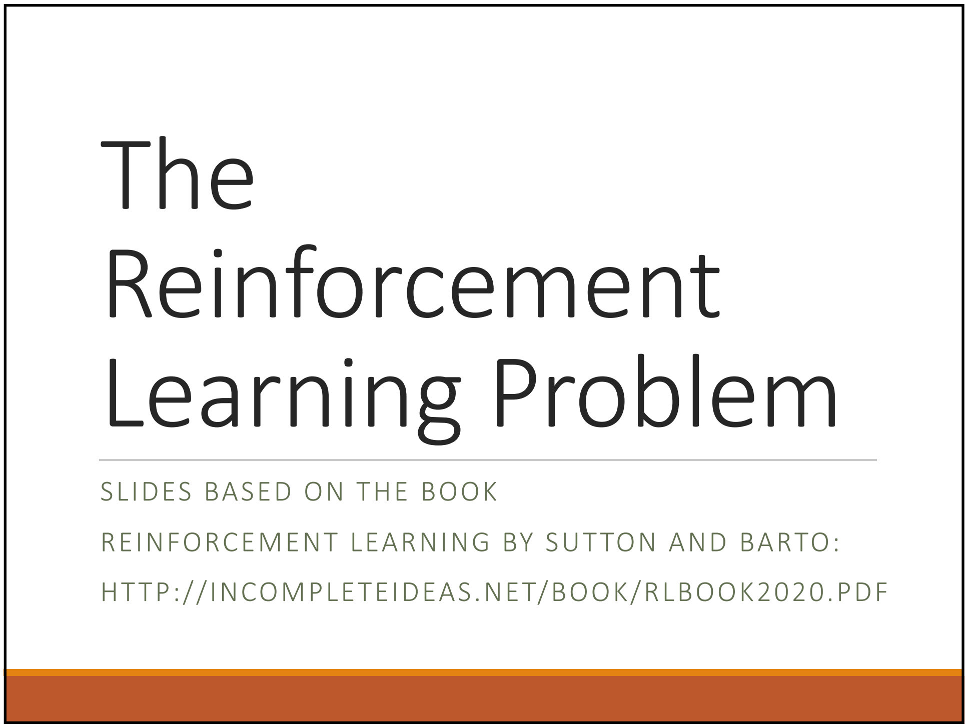
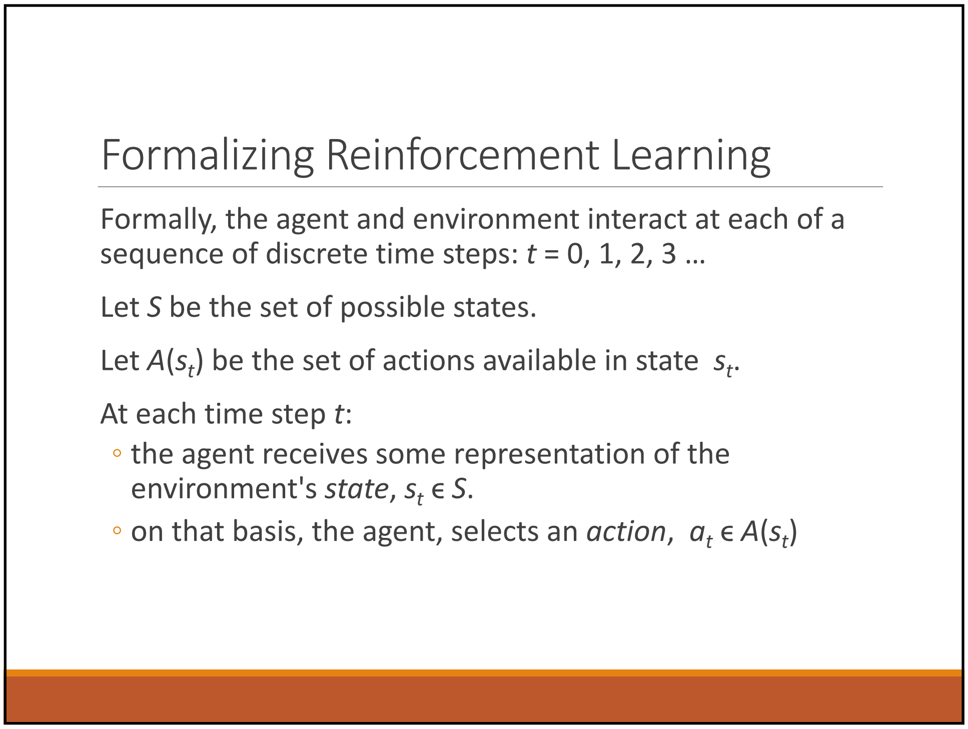


The Reinforcement Learning Problem

SLIDES BASED ON THE BOOK

REINFORCEMENT LEARNING BY SUTTON AND BARTO:

HTTP://INCOMPLETEIDEAS.NET/BOOK/RLBOOK2020.PDF



Formalizing Reinforcement Learning

Formally, the agent and environment interact at each of a sequence of discrete time steps: $t = 0, 1, 2, 3$ …

Let S be the set of possible states.

Let $A(s_t)$ be the set of actions available in state s_t.

At each time step t:

- the agent receives some representation of the environment's *state*, $s_t \in S$.
- on that basis, the agent, selects an *action*, $a_t \in A(s_t)$

Agent-Environment Interface

One time step later, in part as a consequence of its action:

- the agent receives a numerical *reward*, $r_{t+1} \in R$ and
- finds itself in a new state, s_{t+1}.

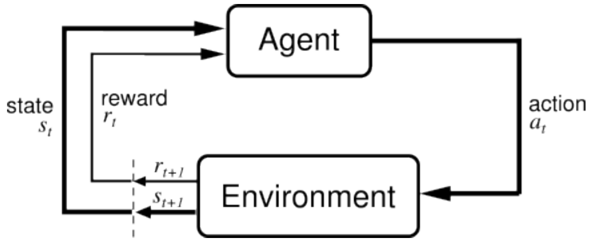


Image source: Fig. 3.1 in Sutton and Barto: Reinforcement Learning

Returns

At each time step, the reward is a simple number, $r_t \in R$.

Let the sequence of rewards received after time step *t* be denoted be r_{t+1}, r_{t+2}, r_{t+3}, …

We seek to maximize the *expected return,* where the return, R_t, is defined as some specific function of the reward sequence.

Returns

In the simplest case the return is the sum of the rewards:

$R_t = r_{t+1} + r_{t+2} + r_{t+3} + \ldots + r_T$

T is the final time step.

This approach makes sense in applications in which there is a final time step.

Alternatively, the approach works if we can separate interactions into subsequences, called *episodes,* such that each episode has a terminal state.

Returns

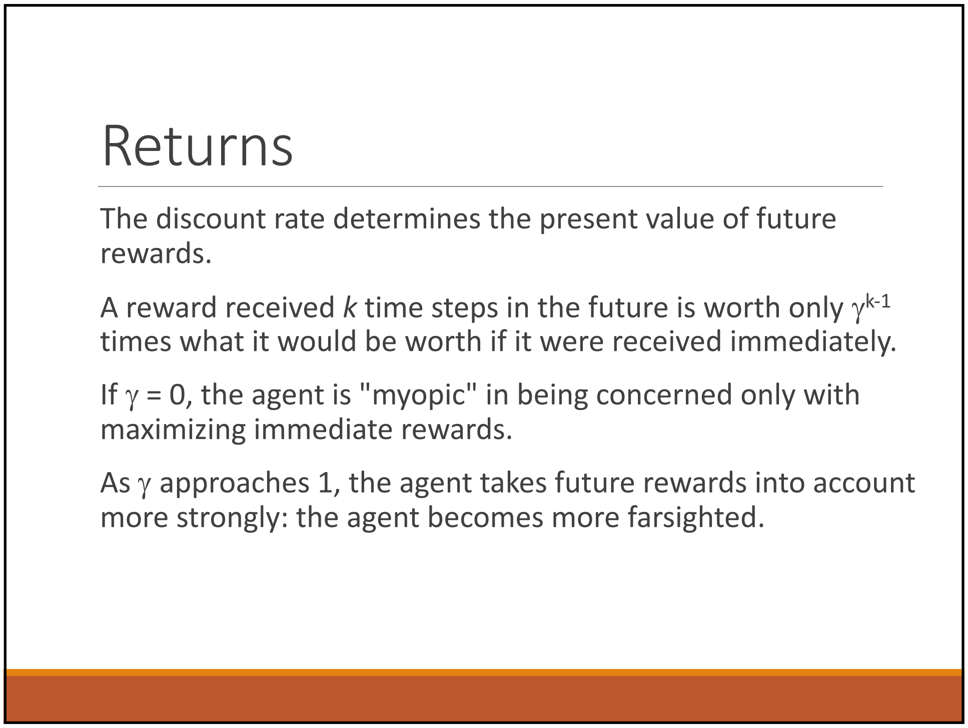
Continuing tasks are interactions that do not break naturally into identifiable episodes, but go on continually without limit.

The return formulation from the prior slide is problematic because the final time step would be $T = \infty$.

We introduce an additional concept, that of *discounting*.

$$R_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \cdots \quad = \quad \sum_{k=0}^{\infty} \gamma^k r_{t+k+1},$$

γ is a parameter, $0 <= \gamma < 1$, called the *discount rate*



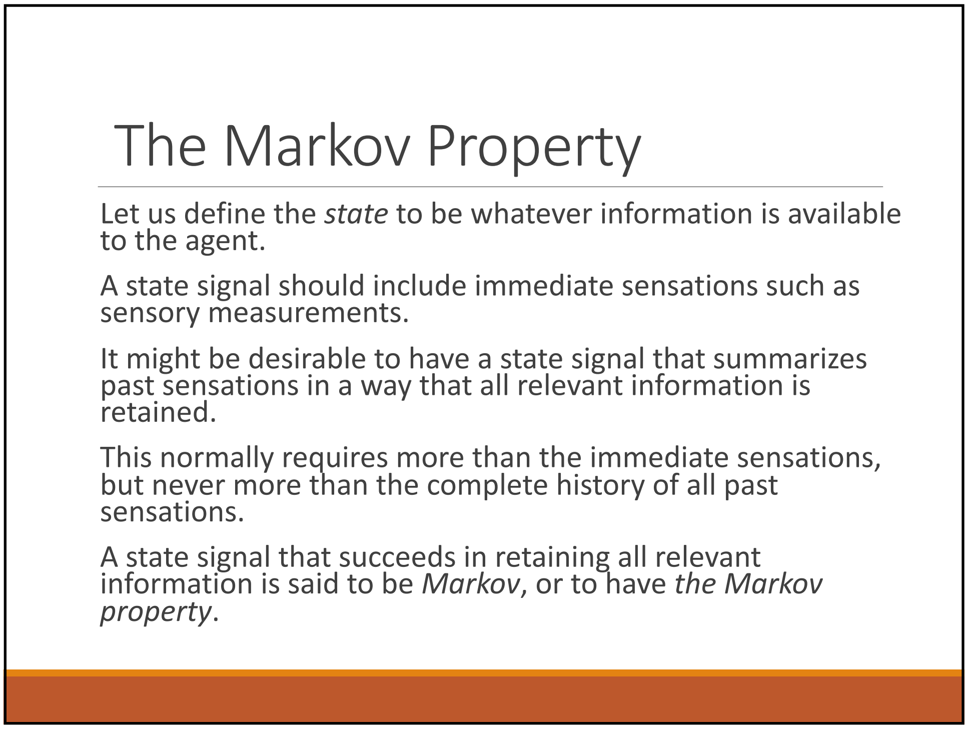
Returns

The discount rate determines the present value of future rewards.

A reward received *k* time steps in the future is worth only γ^{k-1} times what it would be worth if it were received immediately.

If $\gamma = 0$, the agent is "myopic" in being concerned only with maximizing immediate rewards.

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



The Markov Property

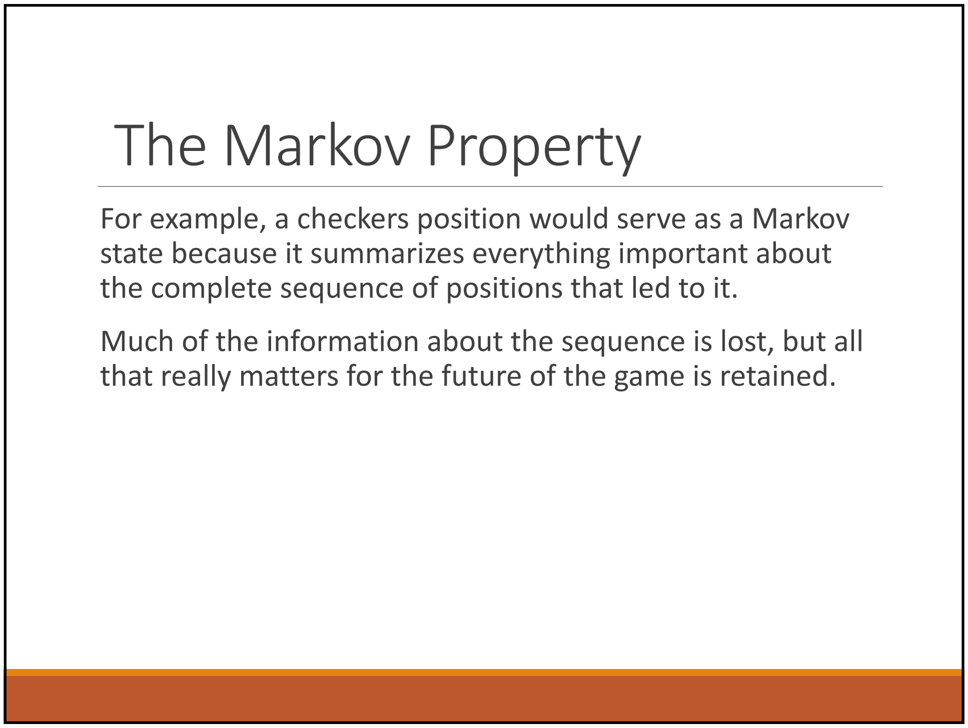
Let us define the *state* to be whatever information is available to the agent.

A state signal should include immediate sensations such as sensory measurements.

It might be desirable to have a state signal that summarizes past sensations in a way that all relevant information is retained.

This normally requires more than the immediate sensations, but never more than the complete history of all past sensations.

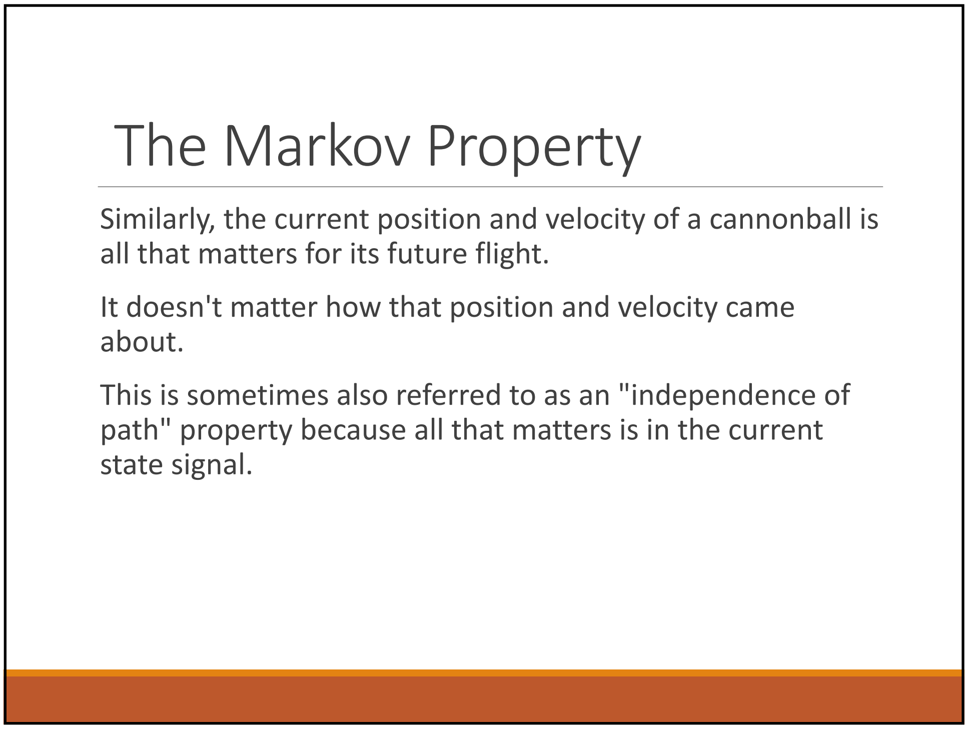
A state signal that succeeds in retaining all relevant information is said to be *Markov*, or to have *the Markov property*.



The Markov Property

For example, a checkers position would serve as a Markov state because it summarizes everything important about the complete sequence of positions that led to it.

Much of the information about the sequence is lost, but all that really matters for the future of the game is retained.



The Markov Property

Similarly, the current position and velocity of a cannonball is all that matters for its future flight.

It doesn't matter how that position and velocity came about.

This is sometimes also referred to as an "independence of path" property because all that matters is in the current state signal.

The Markov Property - Not

Consider how a general environment might respond at time *t+1* to the action taken at time *t*.

In the non-Markov case, this response may depend on everything that has happened earlier.

In this case the dynamics can be defined only by specifying the complete probability distribution:

$$Pr\left\{s_{t+1} = s', r_{t+1} = r \mid s_t, a_t, r_t, s_{t-1}, a_{t-1}, \ldots, r_1, s_0, a_0\right\},$$

for all *s'*, *r*, and all possible values of the past events: s_t, a_t, r_t, …, r_1, s_0, a_0.

The Markov Property

If the state signal has the *Markov property*, then the environment's response at *t* + 1 depends only on the state and action representations at *t*.

In this case, the environment's dynamics can be defined by specifying only

$$Pr\left\{s_{t+1} = s', r_{t+1} = r \mid s_t, a_t\right\},$$

for all *s'*, *r*, s_t *and* a_t

Markov Decision Processes

A reinforcement learning task that satisfies the Markov property is called a *Markov decision process*, or *MDP*.

If the state and action spaces are finite, then it is called a *finite Markov decision process (finite MDP)*.

Finite MDPs are particularly important to the theory of reinforcement learning.

Markov Decision Processes

A particular finite MDP is defined by its state and action sets and by the one-step dynamics of the environment.

Given any state and action, *s* and *a*, the probability of each possible next state, *s'*, is:

$$\mathcal{P}^a_{ss'} = Pr\left\{s_{t+1} = s' \mid s_t = s, a_t = a\right\}.$$

Similarly, given any current state and action, *s* and *a*, together with any next state, *s'*, the expected value of the next reward is:

$$\mathcal{R}^a_{ss'} = E\left\{r_{t+1} \mid s_t = s, a_t = a, s_{t+1} = s'\right\}.$$