Reinforcement Learning
Lectures 4 and 5

Gillian Hayes

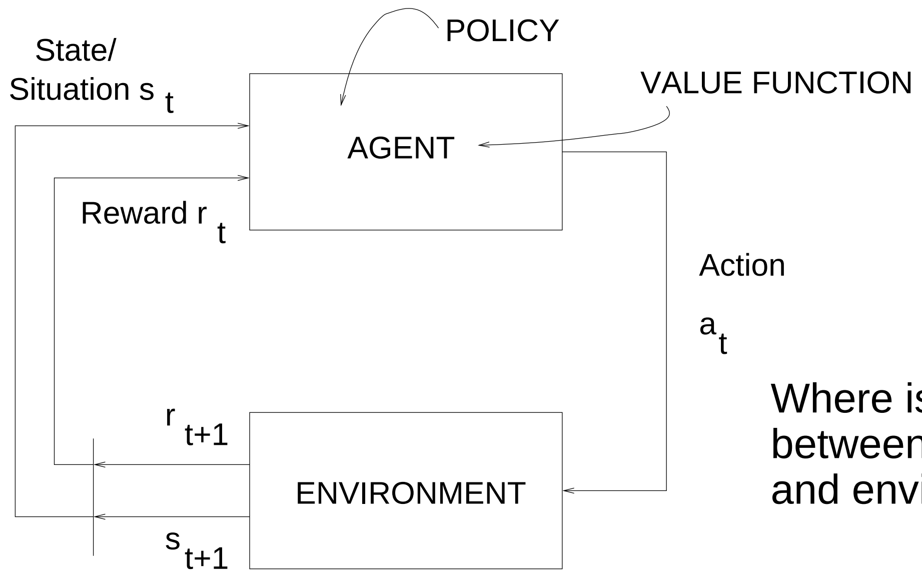
18th January 2007

School of informatics

Reinforcement Learning

- Framework
- Rewards, Returns
- Environment Dynamics
- Components of a Problem
- Values and Action Values, V and Q
- Optimal Policies
- Bellman Optimality Equations

Framework Again



Where is boundary between agent and environment?

Task: one instance of an RL problem – one problem set-up

Learning: how should agent change policy?

Overall goal: maximise amount of reward received over time

Goals and Rewards

Goal: maximise total reward received

Immediate reward r at each step. We must maximise expected cumulative reward:

Return = Total reward $R_t = r_{t+1} + r_{t+2} + r_{t+3} + \cdots + r_\tau$

$\tau =$ final time step (episodes/trials) But what if $\tau = \infty$?

Discounted Reward

$$\begin{aligned} R_t &= r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \cdots \\ &= \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \end{aligned}$$

$0 \leq \gamma < 1$ discount factor $\rightarrow$ discounted reward finite if reward sequence $\{r_k\}$ bounded

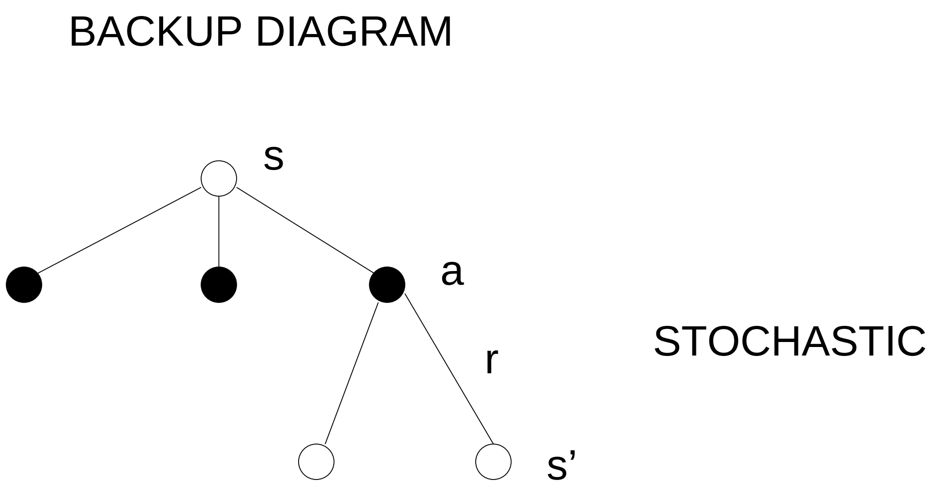
$\gamma = 0$: myopic $\gamma \rightarrow 1$: agent far-sighted. Future rewards count for more

Dynamics of Environment

Choose action a in situation s: what is the probability of ending up in state s'?

Transition probability

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



If action a chosen in state s and subsequent state reached is s' what's the expected reward?

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

If we know P and R then have complete information about environment – may need to learn them

$R^a_{ss'}$ and $\rho(s,a)$

Reward functions

$R^a_{ss'}$	expected next reward given current state s and action a and next state s'
$\rho(s,a)$	expected next reward given current state s and action a

$$\rho(s,a) = \sum_{s'} P^a_{ss'} R^a_{ss'}$$

Sometimes you will see $\rho(s,a)$ in the literature, especially that prior to 1998 when S+B was published.

Sometimes you'll also see $\rho(s)$. This is the reward for being in state s and is equivalent to a "bag of treasure" sitting on a grid-world square (e.g. computer games – weapons, health).

Sutton and Barto's Recycling Robot 1

- At each step, robot has choice of three actions:
  - go out and search for a can
  - wait till a human brings it a can
  - go to charging station to recharge
- Searching is better (higher reward), but runs down battery. Running out of battery power is very bad and robot needs to be rescued
- Decision based on current state – is energy high or low
- Reward is no. cans (expected to be) collected, negative reward for needing rescue

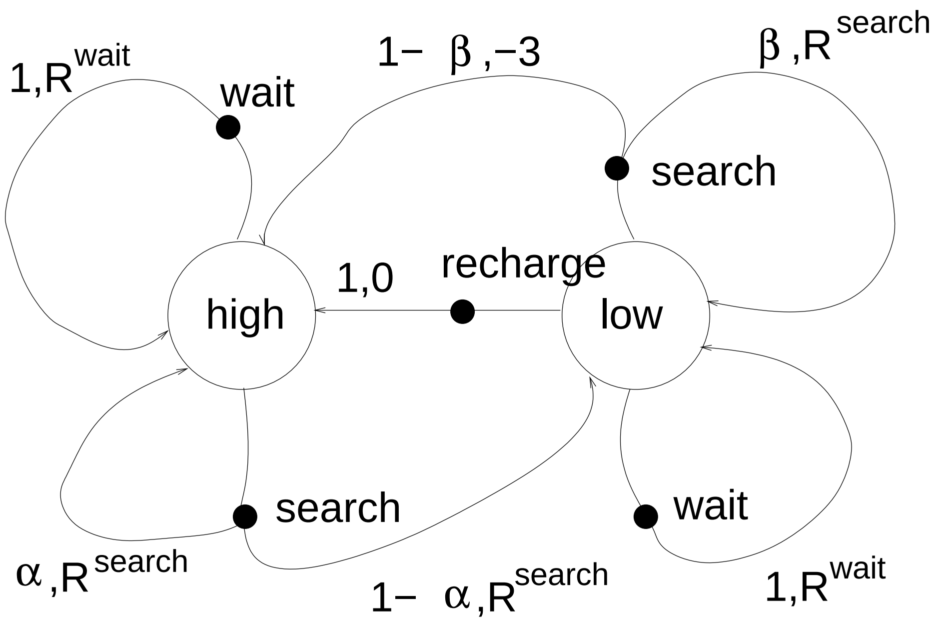
This slide and the next based on an earlier version of Sutton and Barto's own slides from a previous Sutton web resource.

Sutton and Barto's Recycling Robot 2

S={high, low}   A(high) = {search, wait}   A(low) = {search, wait, recharge}

R^{search} expected no. cans when searching R^{wait} expected no. cans when waiting

$R^{search} > R^{wait}$



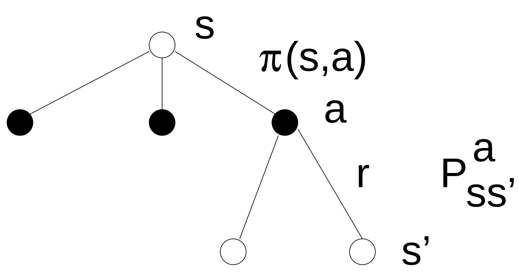
Values V

Policy π maps situations $s \in S$ to (probability distribution over) actions $a \in A(s)$

V-Value of s under policy π is $V^\pi(s)$ = expected return starting in s and following policy π

$$V^\pi(s) = E_\pi\{R_t \mid s_t = s\} = E_\pi\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \mid s_t = s\}$$

BACKUP DIAGRAM FOR V(s)



Convention:
open circle = state
filled circle = action

Action Values Q

Q-Action Value of taking action a in state s under policy π is $Q^\pi(s,a) =$ expected return starting in s, taking a and then following policy π

$$\begin{aligned} Q^\pi(s,a) &= E_\pi\{R_t \mid s_t = s, a_t = a\} \\ &= E_\pi\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \mid s_t = s, a_t = a\} \end{aligned}$$

What is the backup diagram?

Recursive Relationship for V

$$
\begin{aligned}
V^\pi(s) &= E_\pi\{R_t \mid s_t = s\} \\
&= E_\pi\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \mid s_t = s\} \\
&= E_\pi\{r_{t+1} + \gamma \sum_{k=0}^{\infty} \gamma^k r_{t+k+2} \mid s_t = s\} \\
&= \sum_a \pi(s, a,) \sum_{s'} P^a_{ss'}[R^a_{ss'} + \gamma E_\pi\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+2} \mid s_{t+1} = s'\}] \\
&= \sum_a \pi(s, a,) \sum_{s'} P^a_{ss'}[R^a_{ss'} + \gamma V^\pi(s')]
\end{aligned}
$$

This is the BELLMAN EQUATION. How does it relate to backup diagram?

Recursive Relationship for Q

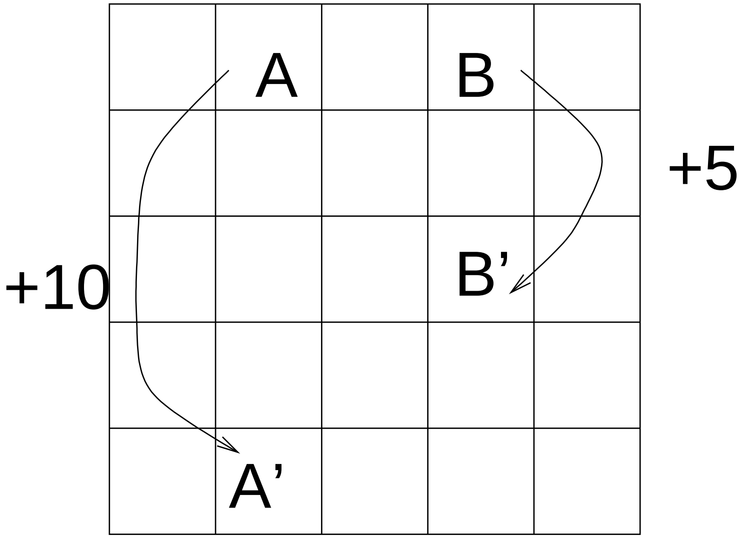
$$Q^\pi(s,a) = \sum_{s'} P^a_{ss'}[R^a_{ss'} + \gamma \sum_{a'} \pi(s',a')Q(s',a')]$$

Relate to backup diagram

Grid World Example

Check the V's comply with Bellman Equation

From Sutton and Barto P. 71, Fig. 3.5



3.3	8.8	4.4	5.3	1.5
1.5	3.0	2.3	1.9	0.5
0.1	0.7	0.7	0.4	-0.4
-1.0	-0.4	-0.4	-0.6	-1.2
-1.9	-1.3	-1.2	-1.4	-2.0

Relating Q and V

$$
\begin{aligned}
Q^{\pi}(s,a) &= E_{\pi}\{\sum_{k=0}^{\infty}\gamma^{k}r_{t+k+1} \mid s_t = s, a_t = a\} \\
&= E_{\pi}\{r_{t+1} + \gamma \sum_{k=0}^{\infty}\gamma^{k}r_{t+k+2} \mid s_t = s, a_t = a\} \\
&= \sum_{s'} P^{a}_{ss'}[R^{a}_{ss'} + \gamma E_{\pi}\{\sum_{k=0}^{\infty}\gamma^{k}r_{t+k+2} \mid s_{t+1} = s'\}] \\
&= \sum_{s'} P^{a}_{ss'}[R^{a}_{ss'} + \gamma V_{\pi}(s')]
\end{aligned}
$$

Relating V and Q

$$
\begin{aligned}
V^\pi(s) &= E_\pi\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \mid s_t = s\} \\
&= \sum_a \pi(s,a) Q^\pi(s,a)
\end{aligned}
$$

Optimal Policies π^*

An optimal policy has the highest/optimal value function $V^*(s)$

It chooses the action in each state which will result in the highest return

Optimal Q-value $Q^*(s,a)$ is reward received from executing action a in state s and following optimal policy π^* thereafter

$$V^*(s) = \max_\pi V^\pi(s)$$

$$Q^*(s,a) = \max_\pi Q^\pi(s,a)$$
$$Q^*(s,a) = E\{r_{t+1} + \gamma V^*(s_{t+1}) \mid s_t = s, a_t = a\}$$

Bellman Optimality Equations 1

Bellman equations for the optimal values and Q-values

$$
\begin{aligned}
V^*(s) &= \max_a Q^{\pi^*}(s,a) \\
&= \max_a E_{\pi^*}\{R_t \mid s_t = s, a_t = a\} \\
&= \max_a E_{\pi^*}\{r_{t+1} + \gamma \sum_k \gamma^k r_{t+k+2} \mid s_t = s, a_t = a\} \\
&= \max_a E\{r_{t+1} + \gamma V^*(s_{t+1}) \mid s_t = s, a_t = a\} \\
&= \max_a \sum_{s'} P^a_{ss'}[R^a_{ss'} + \gamma V^*(s')]
\end{aligned}
$$

$$
\begin{aligned}
Q^*(s,a) &= E\{r_{t+1} + \gamma \max_{a'} Q^*(s_{t+1},a') \mid s_t = s, a_t = a\} \\
&= \sum_{s'} P^a_{ss'}[R^a_{ss'} + \gamma \max_{a'} Q^*(s',a')]
\end{aligned}
$$

Value under optimal policy = expected return for best action from that state.

Bellman Optimality Equations 2

If dynamics of environment $R^a_{ss'}, P^a_{ss'}$ known, then can solve equations for V^* (or Q^*).

Given V^*, what then is optimal policy? I.e. which action a do you pick in state s?

The one which maximises expected $r_{t+1} + \gamma V^*(s_{t+1})$, i.e. the one which gives the biggest

$$\sum_{s'} (\text{instant reward} + \text{discounted future maximum reward}) * P^a_{ss'}$$

So need to do one-step search

There may be more than one action doing this $\rightarrow$ all OK

All GREEDY actions

Given Q^*, what's the optimal policy?

The one which gives the biggest $Q^*(s,a)$, i.e. in state s, you have various Q values, one per action. Pick (an) action with largest Q.

Assumptions for Solving Bellman Optimality Equations

1. Know dynamics of environment $P^a_{ss'}, R^a_{ss'}$
2. Sufficient computational resources (time, memory)

BUT

Example: Backgammon

1. OK
2. 10^{20} states $\Rightarrow 10^{20}$ equations in 10^{20} unknowns, nonlinear equations (max)

Often use a neural network to approximate value functions, policies and models $\Rightarrow$ compact representation

Optimal policy? Only needs to be optimal in situations we encounter – some very rarely/never encountered. So a policy that is only optimal in those states we encounter may do

Components of an RL Problem

Agent, task, environment

States, actions, rewards

Policy $\pi(s, a) \rightarrow$ probability of doing a in s

Value $V(s) \rightarrow$ number – Value of a state

Action value $Q(s, a)$ – Value of a state-action pair

Model $P^a_{ss'} \rightarrow$ probability of going from $s \rightarrow s'$ if do a

Reward function $R^a_{ss'}$ from doing a in s and reaching s'

Return $R \rightarrow$ sum of future rewards

Total future discounted reward $r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \cdots = \sum_{k=0}^{\infty} r_{t+k+1} \gamma^k$

Learning strategy to learn... (continued)

- value – V or Q
- policy
- model

sometimes subject to conditions, e.g. learn best policy you can within given time

Learn to maximise total future discounted reward

RL Buzzwords

Agent, task, environment

Actions, situations/states, rewards

Policy

Environment dynamics and model

Return, total reward, discounted rewards

Value function V, action-value function Q

Optimal value functions and optimal policy

Complete and incomplete environment information

Transition probabilities and reward function

Model-based and model-free learning methods

Temporal and spatial credit assignment

Exploration/exploitation tradeoff