Sequential decision problems and MDPs

Chapter 17, Section 1

Adapted from slides kindly shared by Stuart Russell

Appreciations

♢ StackOverflow Q&A web site - great programming answers!

♢ Book "Abundance" by Diamondis and Kotler - "The Future is Better than you Think"

Share some of yours?

Announcements

Project P2 Multi-Agent Pac-Man is out, due Thu Nov 1

Reformatted, with a more clarity about grading, e.g. point thresholds for q1

Outline

- ♢ Non-Deterministic Search
- ♢ Sequential decision problems and Markov Decision Processes (MDPs)
- ♢ GridWorld (part of P3, Reinforcement Learning)

Rational preferences

Idea: preferences of a rational agent must obey constraints.

The axioms of rationality:

Orderability

Exactly one of $(A \succ B) \vee (B \succ A) \vee (A \sim B)$ holds

Transitivity

$(A \succ B) \wedge (B \succ C) \;\Rightarrow\; (A \succ C)$

Continuity

$A \succ B \succ C \;\Rightarrow\; \exists\, p \;\; [p, A;\; 1-p, C] \sim B$

Substitutability

$A \sim B \;\Rightarrow\; [p, A;\; 1-p, C] \sim [p, B; 1-p, C]$

Monotonicity

$A \succ B \;\Rightarrow\; (p \geq q \;\Leftrightarrow\; [p, A;\; 1-p, B] \succsim [q, A;\; 1-q, B])$

Rational preferences $\;\Rightarrow$

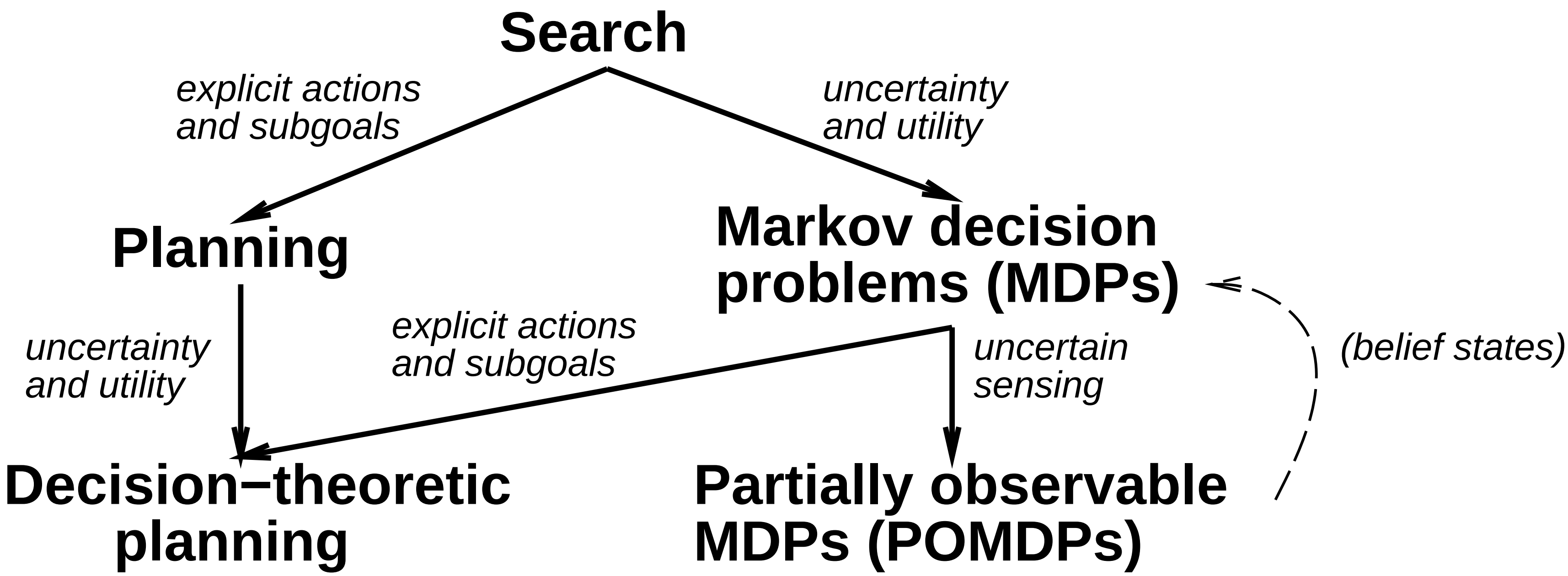
behavior describable as maximization of expected utility

Non-Deterministic Search

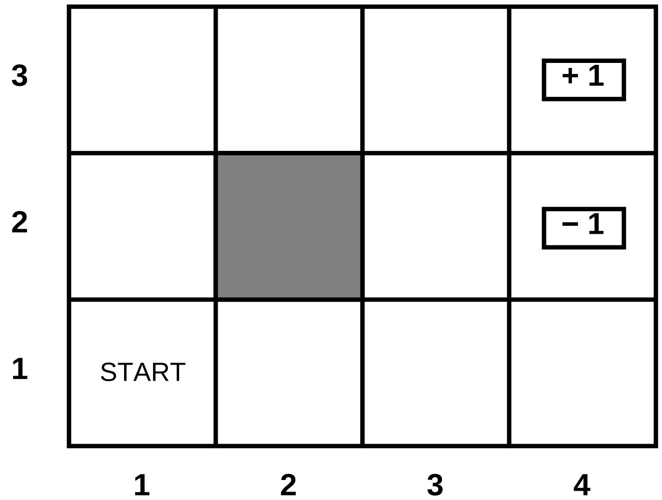
How do you plan when your actions might fail?

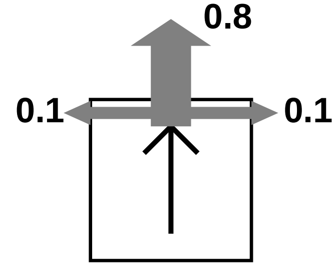
Sequential decision problems

Agent's utility depends on a sequence of of decisions, incorporating utilities, uncertainty and sensing.



Example MDP: Gridworld





Agent in a grid with obstacles, uncertain transitions (think robots)

20% chance of not going in chosen direction

Rewards: combination of per-move reward (positive or negative) and terminal state reward

Gridworld Demo

Gridworld Search Tree

MDP trees vs Expectimax

Markov Decision Processes - a family of non-deterministic search problems

Expectimax will solve non-deterministic search problems (often badly)

Better techniques coming later

States $s \in S$, actions $a \in A$

Model $T(s, a, s') \equiv P(s'|s, a) =$ probability that a in s leads to s'

Transition function, like Successor function

Q-states (like choice nodes)

Reward function $R(s)$ (or $R(s, a)$, $R(s, a, s')$)

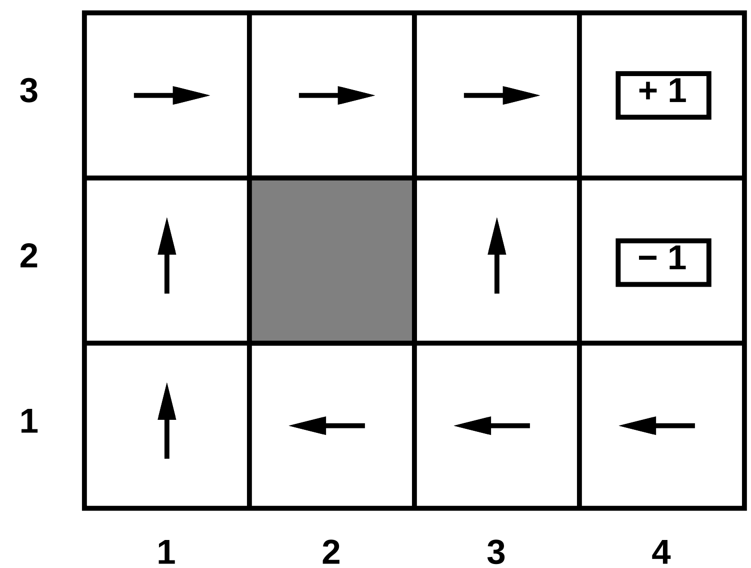
$$= \begin{cases} -0.04 & \text{(small penalty) for nonterminal states} \\ \pm 1 & \text{for terminal states} \end{cases}$$

Solving MDPs

In search problems, aim is to find an optimal *sequence* (luxury!)

In MDPs, aim is to find an optimal *policy* $\pi(s)$
i.e., best action for every possible state s
(because can't predict where one will end up)
The optimal policy maximizes (say) the *expected sum of rewards*

Optimal policy when state penalty $R(s)$ is –0.04:



Solving in Non-Deterministic Search

For now, calculate the whole policy at the beginning - it's small

Then just follow it

More techniques for big state spaces later

Markov Assumptions

Where does this term "Markov" fit in?

Andrey Markov (1856-1922)

Markov processes, Markov chains

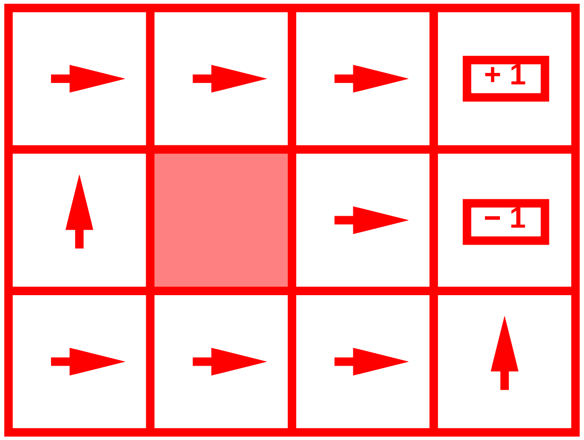
"Markov assumption" is that given the present state, the future and the past are independent

Or at most a finite fixed number of previous states

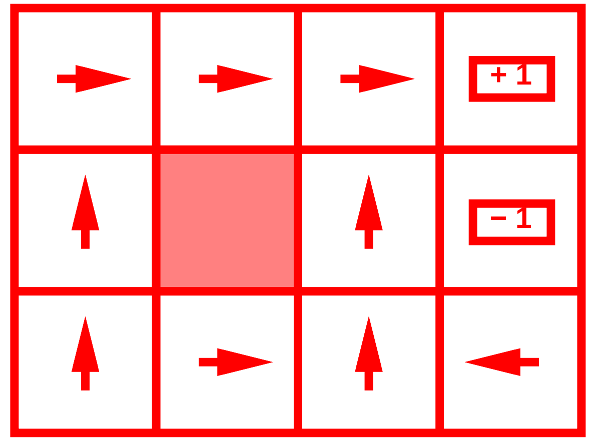
Optimal choice doesn't depend on previous actions

Gridworld Policy Demos

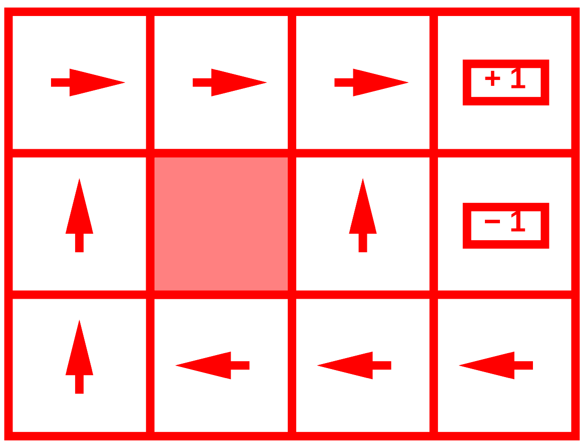
Risk and reward



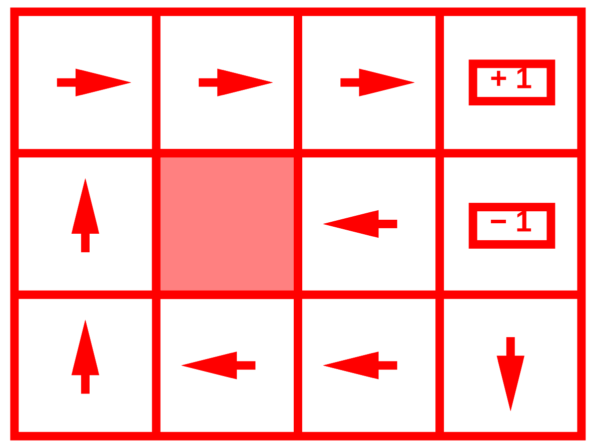
r = [$-\infty$: −1.6284]



r = [−0.4278 : −0.0850]



r = [−0.0480 : −0.0274]



r = [−0.0218 : 0.0000]

Example: High-Low

Utility of state sequences

Need to understand preferences between *sequences* of states

Typically consider stationary preferences on reward sequences:

$$[r, r_0, r_1, r_2, \ldots] \succ [r, r'_0, r'_1, r'_2, \ldots] \;\Leftrightarrow\; [r_0, r_1, r_2, \ldots] \succ [r'_0, r'_1, r'_2, \ldots]$$

Theorem: there are only two ways to combine rewards over time.

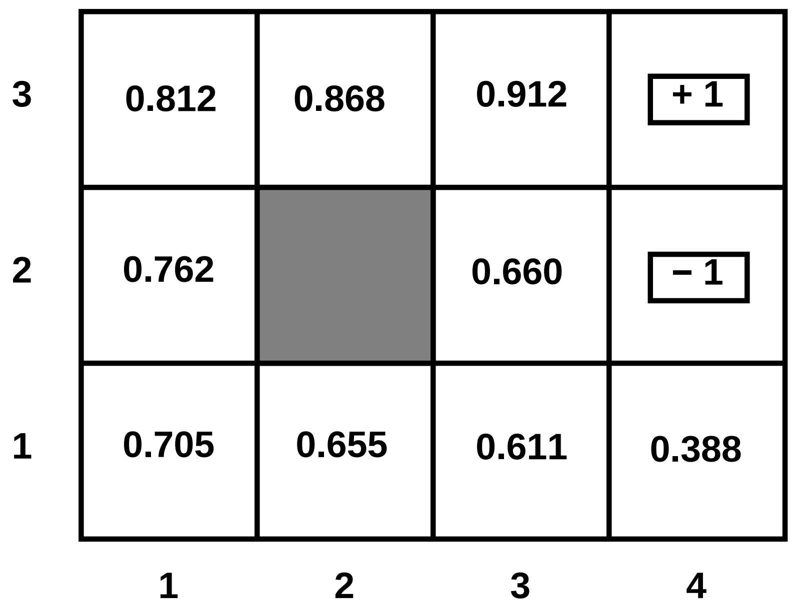
1) *Additive* utility function:
   $U([s_0, s_1, s_2, \ldots]) = R(s_0) + R(s_1) + R(s_2) + \cdots$
2) *Discounted* utility function:
   $U([s_0, s_1, s_2, \ldots]) = R(s_0) + \gamma R(s_1) + \gamma^2 R(s_2) + \cdots$
   where γ is the discount factor

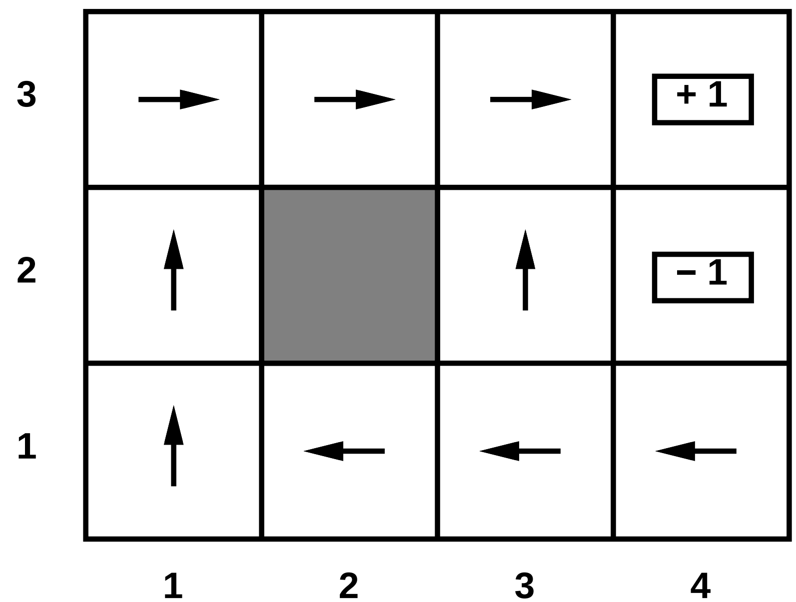
Utility of states

Utility of a *state* (a.k.a. its *value*) is defined to be

$$U(s) = \underline{\text{expected (discounted) sum of rewards (until termination)}}$$
$$\underline{\text{assuming optimal actions}}$$

Given the utilities of the states, choosing the best action is just MEU:
maximize the expected utility of the immediate successors





Utilities contd.

Problem: infinite lifetimes $\Rightarrow$ additive utilities are infinite

1) Finite horizon: termination at a *fixed time* T
$\Rightarrow$ nonstationary policy: $\pi(s)$ depends on time left

2) Absorbing state(s): w/ prob. 1, agent eventually "dies" for any π
$\Rightarrow$ expected utility of every state is finite

3) Discounting: assuming $\gamma < 1$, $R(s) \leq R_{\max}$,

$$U([s_0, \ldots s_\infty]) = \Sigma_{t=0}^{\infty} \gamma^t R(s_t) \leq R_{\max}/(1-\gamma)$$

Smaller $\gamma \Rightarrow$ shorter horizon

4) Maximize system gain = average reward per time step
Theorem: optimal policy has constant gain after initial transient
E.g., taxi driver's daily scheme cruising for passengers