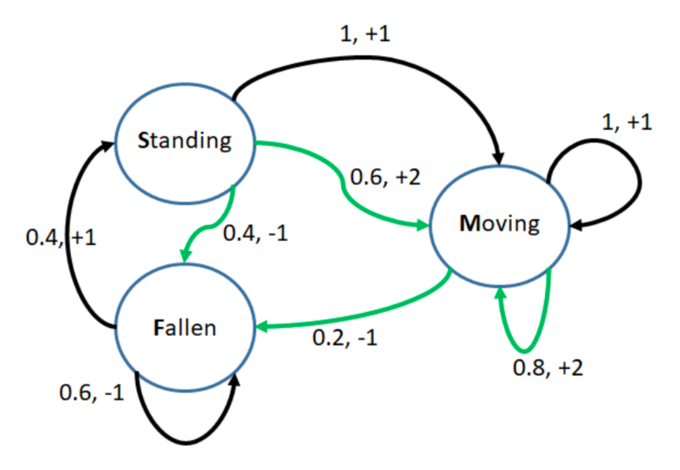
Lecture 2
Section 4: Markov Decision Processes with Imprecise Probabilities (MDPIPs)

Fabio G. Cozman
Universidade de São Paulo - Brazil

Markov decision process (MDP)



Markov decision processes (MDPs)

- Popular in economics, management, operations research.
- An MDP consists of
  1. A state space S.
  2. An action space A.
  3. Transition probabilities $p_a(r|s) = P_a(s_{t+1} = r|s_t = s)$.
  4. Rewards/costs $c_a(s)$.

Policies and their costs

- A *policy* specifies an action for each state (act-state dependence).
- A *stationary policy* is a policy that does not depend on t.
- A policy π_1 dominates policy π_2 if π_1 has total cost smaller than π_2.
- But how to measure "cost" of a policy?

Costs

Additive cost: just add costs for all transitions.

Discounted cost: add costs, but with discount γ:

$$c(s_0) + \gamma c(s_1) + \gamma^2 c(s_2) + \dots$$

Average cost: add costs, divide by number of transitions.

Goal state: all costs are ignored, what matters is to reach some state.

Most popular: Discounted cost

- We must find the optimal policy π^*:

$$\pi^* = \arg\min_{\pi} E\left[\sum_{t=0}^{\infty} \gamma^t c_{\pi(s_t)}(s_t)\right].$$

- For discounted cost, the optimal policy always exists (not necessarily true for other costs!).

Basic relation about discounted cost

- Denote by $E[\pi|s]$ the expected cost when the state is s at $t = 0$.
- Then:

$$E[\pi|s] = c_{\pi(s)}(s) + \gamma \sum_{r \in S} p_{\pi(s)}(r|s) E[\pi|r] \, .$$

- How about the optimal policy and the optimal expected cost?

Bellman equation

- Denote by $E^*[s]$
  - the optimal expected cost when the state is s at $t = 0$;
  - called the *value function* (it depends only on s!).
- By dynamic programming we have the Bellman equation:

$$E^*[s] = \min_{a \in A} \left(c_a(s) + \gamma \sum_{r \in S} p_a(r|s) E^*[r] \right) .$$

- From the optimal cost, we obtain:

$$\pi^*(s) = \arg\min_{a \in A} \left(c_a(s) + \gamma \sum_{r \in S} p_a(r|s) E^*[r] \right) .$$

Algorithms

1. Linear programming: polynomial algorithm, but rarely used.
2. Value iteration.
3. Policy iteration.
4. ...and many variants of those.

Value iteration

- Start with some function $E_0[s]$ for all $s \in S$ (may even be equal to zero!).
- Now repeat untill convergence:
  For each $s \in S$,

$$E_{i+1}[s] = \min_{a \in A} \left(c_a(s) + \gamma \sum_{r \in S} p_a(r|s) E_i[r] \right).$$

- Take, from the last iteration:

$$\pi^*(s) = \arg\min_{a \in A} E_N[s].$$

Convergence of value iteration

- It always converges to the unique optimal policy.
- Convergence is exponentially fast:

$$||E_{i+1}[s] - E^*[s]|| \leq \gamma||E_i[s] - E^*[s]||$$

(where $||f(x)|| = \max_x |f(x)|$).

Policy iteration

- Start with some policy π_0.
- Repeat:
  1. Solve (note that this is a linear system):

$$E[\pi_i|s] = c_{\pi_i(s)}(s) + \gamma \sum_{r\in S} p_{\pi_i(s)}(r|s)E[\pi_i|r]\,.$$

  2. Find $a \in A$ such that, for some $s \in S$,

$$c_a(s) + \gamma \sum_{r\in S} p_a(r|s)E_i[r] \leq E_i[s]\,.$$

    - If there is such a, then make $\pi_{i+1}(s) = a$.
    - Otherwise, stop policy iteration.

Convergence of policy iteration

- It always converges to the unique optimal policy.
- Speed of convergence is not known, but empirically observed to be quite fast.

Factored representations

- Usually MDPs represent states explicitly.
- However, representations in terms of variables are more compact.
- Factored representations use Bayesian networks to represent $P_a(r|s)$ (a dynamic Bayesian network indexed by actions).
- There are graphical representations for costs and policies as well.

Planning languages: Distinguishing feature of AI

- Many representation languages: STRIPS, PDDL, PPDDL....
- PPDDL:
  (:action buy-coffee
  :effect
  (when (not (in-office)) (probabilistic 0.8 (has-coffee))))

Obvious problem: specifying probabilities

- One solution: estimate them from observed/experimental data (back to Silver (1963)).
- Another solution: run the system, estimate ("learn") transitions (reinforcement learning).
- Also natural to consider explicit representation for uncertainty about probability values.
- MDPIPs: MDPs with imprecision in transition probabilities.

MDPIPs

- An MDPIP consists of
  1. A state space S.
  2. An action space A.
  3. Transition credal sets $K_a(r|s) = K_a(s_{t+1} = r|s_t = s)$.
  4. Rewards/costs $c_a(s)$.

Bellman-like equation (Satia and Lave 1973)

- Γ-minimax solution:

$$E^*[s] = \min_{a \in A} \left(c_a(s) + \max_{p \in K} \gamma \sum_{r \in S} p_a(r|s) E^*[r] \right) .$$

- Policy and value iteration have been adapted to this setting.
- Equation can be solved by bilinear programming; in some cases, by integer linear programming.

A few notes

- *Bounded-parameter MDPs* were proposed to abstract complex transitions (Givan, Leach, Dean 1997).
- Within ISIPTA: MDPIPs under E-admissibility by Harmanec (1999/2001); also work by Troffaes (2005).
- Imprecise transition probabilities have been specified by credal networks (Delgado et al. 2008).
- MDPSTs have been proposed to unify various kinds of planning.

Set-valued Markov Decision Processes (MDPSTs)

- A unified representation for probabilistic *and* "nondeterministic" planning (Trevisan et al. 2007).
- Operations are closer to MDPs than to generic MDPIPs.