Markov Decision Processes (MDPs)

- S: state
- A: action
- T: state transition probability matrix
- R: reward function
- gamma: discount factor

$$Cumulative\ Reward:\ \sum_{t=0}^{\infty} \gamma^t R_{a_t}(s_t, s_{t+1})$$

Optimal Policy

- Provides optimal action for any state
- Maximize rewards

$$Optimal\ Policy = \pi^*, \pi^*(S) = A^*$$

Bellman Equation

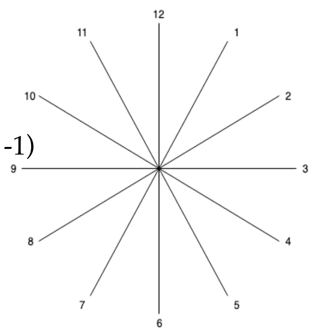
- Used to update value
- For both policy & value iteration

$$V^*(S) = \max_a E[R(s,a) + \gamma V^*(S')]$$

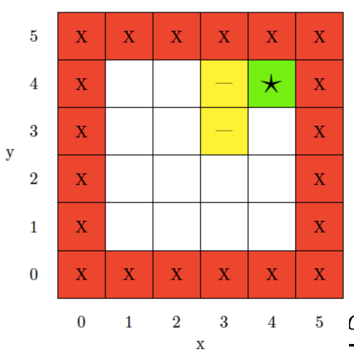
Robot Environment

7 Possible robot actions
Movement (Forward 1, None 0, Backward -1)
Rotation (Left -1, None 0, Right 1)

$$\begin{bmatrix} Movement \\ Rotation \end{bmatrix}_{action\#} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}_1, \begin{bmatrix} 1 \\ 0 \end{bmatrix}_2, \begin{bmatrix} 1 \\ 1 \end{bmatrix}_3, \begin{bmatrix} -1 \\ -1 \end{bmatrix}_4, \begin{bmatrix} -1 \\ 0 \end{bmatrix}_5, \begin{bmatrix} -1 \\ 1 \end{bmatrix}_6, \begin{bmatrix} 0 \\ 0 \end{bmatrix}_7$$

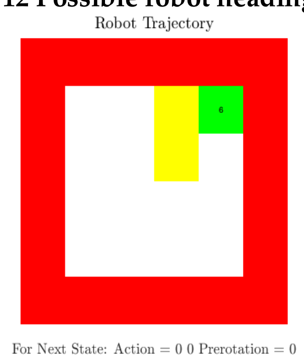


12 Possible robot headings



Grid World in Lab Manual

$\xrightarrow{Grid\ World}$



Matlab robot live trajectory heading and next state shown

Policy Iteration

Psuedo Code
$\pi_0(s) = a\ \forall\ s \in S$
Loop
$\pi \xrightarrow{corresponds\ to} \pi_0$
Compute values of π using Bellman Equation
$V^\pi(S) = E[r|s, \pi(s)] + \gamma \sum_{s' \in S} P(s'|s, \pi(s)) V^\pi(s')$
Improve policy at each state
$\pi'(S) \leftarrow argmax_a [E[r|s,a] + \gamma \sum_{s' \in S} P(s'|s,a) V^\pi(s')]$
Until $\pi = \pi'$

Value Iteration

Psuedo Code
$V_0(s) = 0\ \forall\ s \in S$
Assign arbitrary values to $V(s)$
Loop
$\forall\ s \in S$
$\forall\ a \in A$
$Q(s,a) \leftarrow Q(s,a) = E[r|s,a] + \gamma \sum_{s' \in S} P(s'|s,a) V(s')$
$V(s) \leftarrow \max_a Q(s,a)$
Until $V(s)$ converges

IOU-SHENG (DANNY) CHANG UID:804743003