facebook
Artificial Intelligence Research

# How to solve an MDP: Dynamic Programming

**Alessandro Lazaric**

Facebook AI Research (on leave from Inria Lille)

*Notice*

From now on we mostly work in the *discounted infinite horizon* setting.

Most results (*not always so smoothly*) extend to other settings.

How to solve *exactly* an MDP

# Dynamic Programming

**Bellman Equations**

Value Iteration

Policy Iteration

# The Optimization Problem

$$\max_{\pi} \; V^{\pi}(s_0) =$$

$$= \max_{\pi} \; \mathbb{E}\big[r(s_0, d_0(a_0|s_0)) + \gamma r(s_1, d_1(a_1|s_0, s_1)) + \gamma^2 r(s_2, d_2(a_2|s_0, s_1, s_2)) + \ldots\big]$$

# Plan to Simplify the Optimization Problem



1. Reduce the search space
   - i. History-based $\Rightarrow$ Markov decision rules
   - ii. Non-stationary $\Rightarrow$ Stationary policies
   - $\Rightarrow$ Focus on *stationary policies with Markov decision rules*
2. Leverage Markov property of the MDP to “simplify” the value function
3. Stochastic $\Rightarrow$ Deterministic decision rules

$\Rightarrow$ Focus on stationary policies with *deterministic* Markov decision rules

# From History-Based to Markov Policies



**Theorem ([1])**

*Consider an MDP with $|A| < \infty$ and an initial distribution $\beta$ over states such that $\big|\{s \in S : \beta(s) > 0\}\big| < \infty$. For any policy $\pi$, let*

$$p_t^{\pi}(s,a) = \mathbb{P}\big[S_t = s, A_t = a\big]; \qquad p_t^{\pi}(s) = \mathbb{P}\big[S_t = s\big].$$

*Then for any history-based policy $\pi$ there exists a Markov policy $\overline{\pi}$ such that*

$$p_t^{\overline{\pi}}(s,a) = p_t^{\pi}(s,a); \qquad p_t^{\overline{\pi}}(s) = p_t^{\pi}(s)$$

*for any $s \in S$, $a \in A$ and $t \in \mathbb{N}^+$.*

$\Rightarrow$ Markov policies are as "expressive" as history-based policies

# Proof: From History-Based to Markov Policies



For any $\pi = (d_0, d_1, \ldots)$ with $d_t$ a randomized history-dependent decision rule, let $\overline{\pi} = (\overline{d}_0, \overline{d}_1, \ldots)$ be a randomized Markov policy such that

$$\overline{d}_t(a|s) = \frac{p_t^{\pi}(s,a)}{p_t^{\pi}(s)}$$

*Base case.* For any $s$, $p_0^{\overline{\pi}}(s) = p_0^{\pi}(s)$ by definition. And

$$p_0^{\overline{\pi}}(s,a) = p_0^{\overline{\pi}}(s)\overline{d}_0(a|s) = p_0^{\overline{\pi}}(s)\frac{p_0^{\pi}(s,a)}{p_0^{\pi}(s)} = p_0^{\overline{\pi}}(s)\frac{p_0^{\pi}(s,a)}{p_0^{\overline{\pi}}(s)} = p_0^{\pi}(s,a)$$

# Proof: From History-Based to Markov Policies

*Induction.* For any $s$ and some $t > 0$, $p_t^{\overline{\pi}}(s) = p_t^{\pi}(s)$ and $p_t^{\overline{\pi}}(s,a) = p_t^{\pi}(s,a)$ by inductive assumption. Then

$$
\begin{aligned}
p_{t+1}^{\overline{\pi}}(s_{t+1}) &= \sum_{s_t,a_t} p_t^{\overline{\pi}}(s_t,a_t)p(s_{t+1}|s_t,a_t) \\
&= \sum_{s_t,a_t} p_t^{\overline{\pi}}(s_t)\overline{d}_t(a_t|s_t)p(s_{t+1}|s_t,a_t) \\
&= \sum_{s_t,a_t} p_t^{\overline{\pi}}(s_t)\frac{p_t^{\pi}(s_t,a_t)}{p_t^{\pi}(s_t)}p(s_{t+1}|s_t,a_t) \\
&= \sum_{s_t,a_t} p_t^{\overline{\pi}}(s_t)\frac{p_t^{\pi}(s_t,a_t)}{p_t^{\overline{\pi}}(s_t)}p(s_{t+1}|s_t,a_t) \\
&= \sum_{s_t,a_t} p_t^{\pi}(s_t,a_t)p(s_{t+1}|s_t,a_t) \\
&= p_{t+1}^{\pi}(s_{t+1})
\end{aligned}
$$

Similar for $p_{t+1}^{\overline{\pi}}(s_{t+1},a_{t+1}) = p_{t+1}^{\pi}(s_{t+1},a_{t+1})$.

# The Discounted Occupancy Measure



$$
\begin{aligned}
V^\pi(s) &= \mathbb{E}\Big[\sum_{t=0}^{\infty} \gamma^t r(s_t, d_t(s_t))\Big] \\
&= \sum_{t=0}^{\infty} \gamma^t \mathbb{E}\Big[r(s_t, d_t(s_t))\Big] \\
&= \sum_{t=0}^{\infty} \gamma^t \sum_{s,a} \mathbb{P}[S_t = s, A_t = a] r(s,a) \\
&= \frac{1}{1-\gamma} \sum_{s,a} (1-\gamma) \sum_{t=0}^{\infty} \gamma^t \mathbb{P}[S_t = s, A_t = a] r(s,a) \\
&= \frac{1}{1-\gamma} \sum_{s,a} \rho_\gamma^\pi(s,a) r(s,a)
\end{aligned}
$$

# From Non-Stationary to Stationary Policies



**Theorem ([? ])**

*Consider an MDP with* $|A| < \infty$ *and an initial distribution* $\beta$ *over states such that* $\big|\{s \in S : \beta(s) > 0\}\big| < \infty$.
*Then for any non-stationary policy* $\pi$ *there exists a stationary policy* $\overline{\pi}$ *such that*

$$\rho_\gamma^{\overline{\pi}}(s,a) = \rho_\gamma^{\pi}(s,a); \qquad \rho_\gamma^{\overline{\pi}}(s) = \rho_\gamma^{\pi}(s)$$

*for any* $s \in S$, $a \in A$ *and* $t \in \mathbb{N}^+$.

$\Rightarrow$ Stationary policies are as "expressive" as non-stationary policies
$\Rightarrow$ Stationary policies can "generate" any value function

# Proof: From Non-Stationary to Stationary Policies

*State* discounted occupancy measure for stationary policy $\overline{\pi}$ (with Markov decision rules)

$$
\begin{aligned}
\rho_\gamma^{\overline{\pi}}(s) &= (1-\gamma)\sum_{t=0}^{\infty}\gamma^t \mathbb{P}[S_t = s] \\
&= (1-\gamma)\beta(s) + (1-\gamma)\sum_{t=1}^{\infty}\gamma^t \mathbb{P}[S_t = s] \\
&= (1-\gamma)\beta(s) + (1-\gamma)\gamma\sum_{t=1}^{\infty}\gamma^{t-1}\sum_{s'}\sum_{a}\mathbb{P}[S_{t-1} = s', A_{t-1} = a]p(s|s',a) \\
&= (1-\gamma)\beta(s) + \gamma\sum_{s'}(1-\gamma)\sum_{t=1}^{\infty}\gamma^{t-1}\mathbb{P}[S_{t-1} = s']\sum_{a}\overline{\pi}(a|s')p(s|s',a) \\
&= (1-\gamma)\beta(s) + \gamma\sum_{s'}(1-\gamma)\sum_{t=1}^{\infty}\gamma^{t-1}\mathbb{P}[S_{t-1} = s']p^{\overline{\pi}}(s|s') \\
&= (1-\gamma)\beta(s) + \gamma\sum_{s'}\rho_\gamma^{\overline{\pi}}(s')p^{\overline{\pi}}(s|s')
\end{aligned}
$$

# Proof: From Non-Stationary to Stationary Policies

Moving to *matrix* formulation

$$[\boldsymbol{\rho}_\gamma^{\overline{\pi}}]_s = \rho_\gamma^{\overline{\pi}}(s)$$
$$[P^{\overline{\pi}}]_{s,s'} = p^{\overline{\pi}}(s'|s)$$

$$\rho_\gamma^{\overline{\pi}}(s) = (1-\gamma)\beta(s) + \gamma \sum_{s'} \rho_\gamma^{\overline{\pi}}(s') p^{\overline{\pi}}(s|s')$$
$$\Rightarrow \boldsymbol{\rho}_\gamma^{\overline{\pi}} = (1-\gamma)\beta(s) + \gamma \boldsymbol{\rho}_\gamma^{\overline{\pi}} P^{\overline{\pi}}$$
$$\Rightarrow \boldsymbol{\rho}_\gamma^{\overline{\pi}} = (1-\gamma)\beta(s)\big(I - \gamma P^{\overline{\pi}}\big)^{-1}$$

# Proof: From Non-Stationary to Stationary Policies

For any non-stationary policy $\pi$ define a stationary policy $\overline{\pi}$

$$\overline{\pi}(a|s') = \frac{\rho_\gamma^\pi(s,a)}{\rho_\gamma^\pi(s)}$$

$$\begin{aligned}
\rho_\gamma^\pi(s) &= (1-\gamma)\beta(s) + \gamma \sum_{s'}\sum_{a}(1-\gamma)\sum_{t=1}^{\infty}\gamma^{t-1}\mathbb{P}\big[S_{t-1}=s', A_{t-1}=a\big]p(s|s',a) \\
&= (1-\gamma)\beta(s) + \gamma \sum_{s'}\sum_{a}\rho_\gamma^\pi(s',a)p(s|s',a) \\
&= (1-\gamma)\beta(s) + \gamma \sum_{s'}\sum_{a}\overline{\pi}(a|s')\rho_\gamma^\pi(s')p(s|s',a) \\
&= (1-\gamma)\beta(s) + \gamma \sum_{s'}\rho_\gamma^\pi(s')\sum_{a}\overline{\pi}(a|s')p(s|s',a) \\
&= (1-\gamma)\beta(s) + \gamma \sum_{s'}\rho_\gamma^\pi(s')p^{\overline{\pi}}(s|s')
\end{aligned}$$

# Proof: From Non-Stationary to Stationary Policies



Moving to the *matrix* formulation

$$
\begin{aligned}
\rho_\gamma^\pi(s) &= (1-\gamma)\beta(s) + \gamma \sum_{s'} \rho_\gamma^\pi(s') p^{\overline{\pi}}(s|s') \\
\Rightarrow \boldsymbol{\rho}_\gamma^\pi &= (1-\gamma)\beta(s)\big(I - \gamma P^{\overline{\pi}}\big)^{-1} \\
\Rightarrow \boldsymbol{\rho}_\gamma^\pi &= \boldsymbol{\rho}_\gamma^{\overline{\pi}}
\end{aligned}
$$

# The Optimization Problem



$$\max_{\pi} V^{\pi}(x_0) =$$

$$= \max_{\pi} \mathbb{E}\big[r(s_0, d_0(a_0|s_0)) + \gamma r(s_1, d_1(a_1|s_0, s_1)) + \gamma^2 r(s_2, d_2(a_2|s_0, s_1, s_2)) + \ldots\big]$$

$$= \max_{\pi \in \Pi^{MRS}} \mathbb{E}\big[r(s_0, \pi(a_0, s_0)) + \gamma r(s_1, \pi(a_1|s_1)) + \gamma^2 r(s_2, \pi(a_2|s_2)) + \ldots\big]$$

# The Optimization Problem



$$\max_{\pi} \; V^{\pi}(x_0) =$$

$$= \max_{\pi} \; \mathbb{E}\big[r(s_0, d_0(a_0|s_0)) + \gamma r(s_1, d_1(a_1|s_0, s_1)) + \gamma^2 r(s_2, d_2(a_2|s_0, s_1, s_2)) + \ldots\big]$$

$$= \max_{\pi \in \Pi^{MRS}} \; \mathbb{E}\big[r(s_0, \pi(a_0, s_0)) + \gamma r(s_1, \pi(a_1|s_1)) + \gamma^2 r(s_2, \pi(a_2|s_2)) + \ldots\big]$$

☞ Even if we restrict to deterministic policies we still have $|A|^{|S|}$ policies to check

# The Bellman Equation



**Theorem**

*For any stationary policy $\pi = (\pi, \pi, \dots)$, at any state $s \in S$, the state value function satisfies the Bellman equation:*

$$V^\pi(s) = r(s, \pi(s)) + \gamma \sum_y p(y|s, \pi(s)) V^\pi(y).$$

# Proof: The Bellman Equation

For any stationary policy $\pi = (\pi, \pi, \dots)$,

$$V^\pi(s) = \mathbb{E}\Big[\sum_{t=0}^\infty \gamma^t r(s_t, \pi(s_t)) \,|\, s_0 = s; \pi\Big]$$ *[value function]*

$$= r(s, \pi(s)) + \mathbb{E}\Big[\sum_{t=1}^\infty \gamma^t r(s_t, \pi(s_t)) \,|\, s_0 = s; \pi\Big]$$

$$= r(s, \pi(s))$$ *[Markov property]*

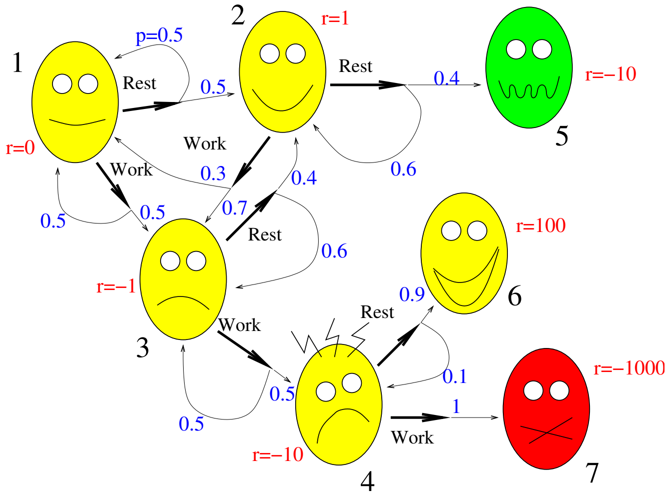
$$+ \gamma \sum_{s'} \mathbb{P}(s_1 = s' \,|\, s_0 = s; \pi(s_0)) \mathbb{E}\Big[\sum_{t=1}^\infty \gamma^{t-1} r(s_t, \pi(s_t)) \,|\, s_1 = s'; \pi\Big]$$

$$= r(s, \pi(s))$$ *[MDP and change of "time"]*

$$+ \gamma \sum_{s'} p(s'|s, \pi(s)) \mathbb{E}\Big[\sum_{t'=0}^\infty \gamma^{t'} r(s_{t'}, \pi(s_{t'})) \,|\, s_{0'} = s'; \pi\Big]$$

$$= r(s, \pi(s)) + \gamma \sum_{s'} p(s'|s, \pi(s)) V^\pi(s')$$ *[value function]*

# The student dilemma

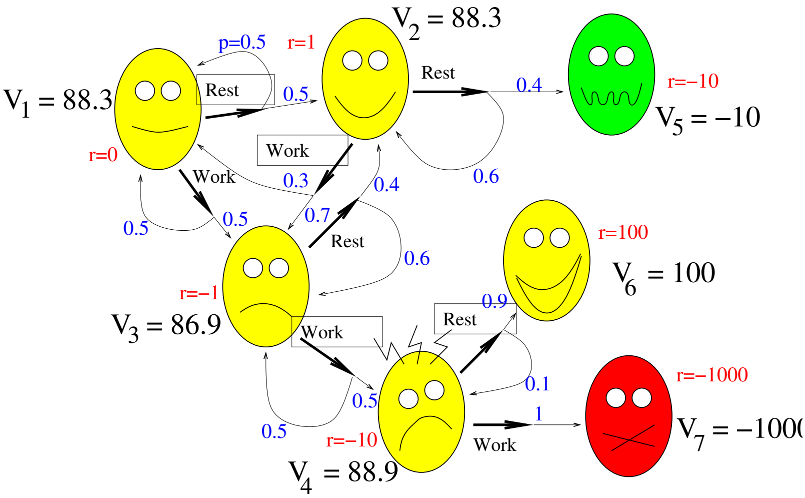


# The student dilemma

- *Model*: all the transitions are Markov, states $x_5, x_6, x_7$ are terminal.
- *Setting*: infinite horizon with terminal states.
- *Objective*: find the policy that maximizes the expected sum of rewards before achieving a terminal state.

*Notice*: not a discounted infinite horizon setting! But the Bellman equations hold unchanged.

# The student dilemma



# The student dilemma

Computing $V_4$:

$$V_6 = 100$$
$$V_4 = -10 + (0.9V_6 + 0.1V_4)$$

$$\Rightarrow V_4 = \frac{-10 + 0.9V_6}{0.9} = 88.8$$

# *The student dilemma*

Computing $V_3$: *no need* to consider all possible trajectories

$$V_4 = 88.8$$
$$V_3 = -1 + (0.5V_4 + 0.5V_3)$$

$$\Rightarrow V_3 = \frac{-1 + 0.5V_4}{0.5} = 86.8$$

# The student dilemma

Computing $V_3$: *no need* to consider all possible trajectories

$$\begin{aligned} V_4 &= 88.8 \\ V_3 &= -1 + (0.5V_4 + 0.5V_3) \end{aligned}$$

$$\Rightarrow V_3 = \frac{-1 + 0.5V_4}{0.5} = 86.8$$

and so on for the rest...

# Bellman Equation: a System of Equations

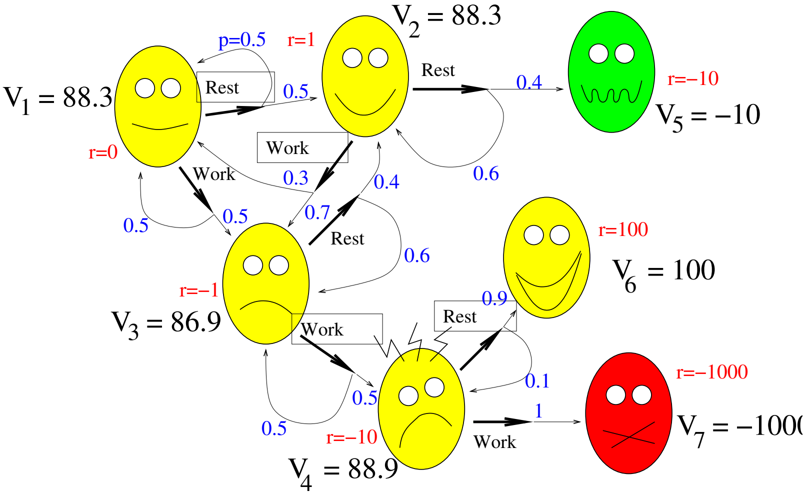


The Bellman equation

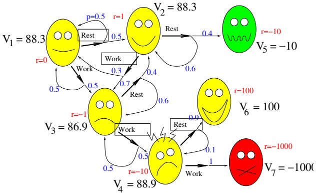
$$V^\pi(x) = r(x, \pi(x)) + \gamma \sum_y p(y|x, \pi(x)) V^\pi(y).$$

is a **linear** system of equations with $N$ unknowns and $N$ linear constraints.

# The student dilemma



# The student dilemma




$$V^\pi(x) = r(x, \pi(x)) + \gamma \sum_y p(y|x, \pi(x)) V^\pi(y)$$

System of equations

$$\begin{cases} V_1 &= 0 + 0.5V_1 + 0.5V_2 \\ V_2 &= 1 + 0.3V_1 + 0.7V_3 \\ V_3 &= -1 + 0.5V_4 + 0.5V_3 \\ V_4 &= -10 + 0.9V_6 + 0.1V_4 \\ V_5 &= -10 \\ V_6 &= 100 \\ V_7 &= -1000 \end{cases} \quad \Rightarrow$$

$$(V, R \in \mathbb{R}^7,\ P \in \mathbb{R}^{7\times 7})$$

$$V = R + PV$$

$$\Downarrow$$

$$V = (I - P)^{-1} R$$

# The Optimal Bellman Equation



**Bellman's Principle of Optimality** [? ]:

> *"An optimal policy has the property that, whatever the initial state and the initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision."*

# The Optimal Bellman Equation



**Theorem**

*The optimal value function $V^*$ (i.e., $V^* = \max_\pi V^\pi$) is the solution to the optimal Bellman equation:*

$$V^*(s) = \max_{a \in A} \Big[ r(s,a) + \gamma \sum_{s'} p(s'|s,a) V^*(s') \Big].$$

*and any optimal policy is such that*

$$\pi^*(a|s) \geq 0 \Leftrightarrow a \in \arg\max_{a' \in A} \Big[ r(s,a') + \gamma \sum_{s'} p(s'|s,a) V^*(s') \Big].$$

# The Optimal Bellman Equation



**Theorem**

*The optimal value function $V^*$ (i.e., $V^* = \max_\pi V^\pi$) is the solution to the optimal Bellman equation:*

$$V^*(s) = \max_{a \in A} \Big[ r(s,a) + \gamma \sum_{s'} p(s'|s,a) V^*(s') \Big].$$

*and any optimal policy is such that*

$$\pi^*(a|s) \geq 0 \Leftrightarrow a \in \arg\max_{a' \in A} \Big[ r(s,a') + \gamma \sum_{s'} p(s'|s,a) V^*(s') \Big].$$

There is always a deterministic policy

# Proof: The Optimal Bellman Equation



For any policy $\pi = (a, \pi')$ (possibly non-stationary),

$$
\begin{aligned}
V^*(x) &= \max_{\pi} \mathbb{E}\Big[\sum_{t\geq 0} \gamma^t r(x_t, \pi(x_t)) \,|\, x_0 = x; \pi\Big] \\
&= \max_{(a,\pi')} \Big[r(x,a) + \gamma \sum_y p(y|x,a) V^{\pi'}(y)\Big] \\
&= \max_{a} \Big[r(x,a) + \gamma \sum_y p(y|x,a) \max_{\pi'} V^{\pi'}(y)\Big] \\
&= \max_{a} \Big[r(x,a) + \gamma \sum_y p(y|x,a) V^*(y)\Big].
\end{aligned}
$$

# Proof: The Optimal Bellman Equation



We have

$$\max_{\pi'} \sum_y p(y|x,a) V^{\pi'}(y) \leq \sum_y p(y|x,a) \max_{\pi'} V^{\pi'}(y)$$

But, let $\overline{\pi}(y) = \arg\max_{\pi'} V^{\pi'}(y)$

$$\sum_y p(y|x,a) \max_{\pi'} V^{\pi'}(y) = \sum_y p(y|x,a) V^{\overline{\pi}}(y) \leq \max_{\pi'} \sum_y p(y|x,a) V^{\pi'}(y)$$

■

# System of Equations

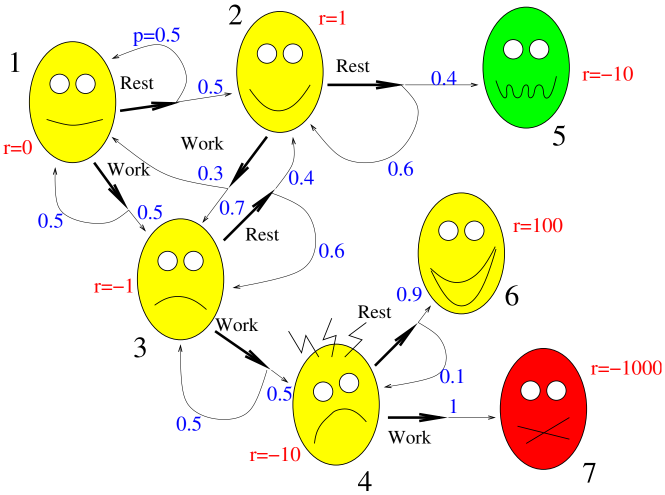


The optimal Bellman equation

$$V^*(s) = \max_{a \in A} \big[ r(s,a) + \gamma \sum_{s'} p(y|s,a) V^*(s') \big].$$

is a **non-linear** system of equations with $N$ unknowns and $N$ non-linear constraints (i.e., the $\max$ operator).

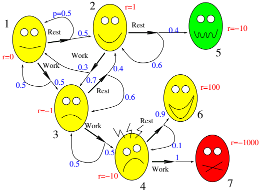
# The Student Dilemma



# The Student Dilemma



$$V^*(x) = \max_{a \in A} \Big[ r(x,a) + \gamma \sum_y p(y|x,a) V^*(y) \Big]$$




*System of equations*

$$\begin{cases} V_1 & = \max\{0 + 0.5V_1 + 0.5V_2;\ \ 0 + 0.5V_1 + 0.5V_3\} \\ V_2 & = \max\{1 + 0.4V_5 + 0.6V_2;\ \ 1 + 0.3V_1 + 0.7V_3\} \\ V_3 & = \max\{-1 + 0.4V_2 + 0.6V_3;\ \ -1 + 0.5V_4 + 0.5V_3\} \\ V_4 & = \max\{-10 + 0.9V_6 + 0.1V_4;\ \ -10 + V_7\} \\ V_5 & = -10 \\ V_6 & = 100 \\ V_7 & = -1000 \end{cases}$$

$\Rightarrow$ too complicated, we need to find an alternative solution.

# The Bellman Operators



*Notation.* w.l.o.g. a discrete state space $|S| = N$ and $V^\pi \in \mathbb{R}^N$.

**Definition**

For any $W \in \mathbb{R}^N$, the *Bellman operator* $\mathcal{T}^\pi : \mathbb{R}^N \to \mathbb{R}^N$ is

$$\mathcal{T}^\pi W(s) = r(s, \pi(s)) + \gamma \sum_{s'} p(s'|s, \pi(s)) W(s'),$$

and the *optimal Bellman operator* (or dynamic programming operator) is

$$\mathcal{T} W(s) = \max_{a \in A} \big[ r(s, a) + \gamma \sum_{s'} p(s'|s, a) W(s) \big].$$

# The Bellman Operators



**Proposition**

*Properties of the Bellman operators*

1. *Monotonicity: for any $W_1, W_2 \in \mathbb{R}^N$, if $W_1 \leq W_2$ component-wise, then*

$$\begin{aligned} \mathcal{T}^\pi W_1 &\leq \mathcal{T}^\pi W_2, \\ \mathcal{T} W_1 &\leq \mathcal{T} W_2. \end{aligned}$$

# The Bellman Operators



**Proposition**

*Properties of the Bellman operators*

1. *Monotonicity*: *for any* $W_1, W_2 \in \mathbb{R}^N$, *if* $W_1 \leq W_2$ *component-wise, then*

$$\begin{aligned} \mathcal{T}^\pi W_1 &\leq \mathcal{T}^\pi W_2, \\ \mathcal{T} W_1 &\leq \mathcal{T} W_2. \end{aligned}$$

2. *Offset*: *for any scalar* $c \in \mathbb{R}$,

$$\begin{aligned} \mathcal{T}^\pi (W + cI_N) &= \mathcal{T}^\pi W + \gamma c I_N, \\ \mathcal{T} (W + cI_N) &= \mathcal{T} W + \gamma c I_N, \end{aligned}$$

# The Bellman Operators

## Proposition

3. *Contraction in $L_\infty$-norm:* for any $W_1, W_2 \in \mathbb{R}^N$

$$
\begin{aligned}
||\mathcal{T}^\pi W_1 - \mathcal{T}^\pi W_2||_\infty &\leq \gamma ||W_1 - W_2||_\infty, \\
||\mathcal{T} W_1 - \mathcal{T} W_2||_\infty &\leq \gamma ||W_1 - W_2||_\infty.
\end{aligned}
$$

# The Bellman Operators

## Proposition

3. *Contraction in $L_\infty$-norm*: *for any* $W_1, W_2 \in \mathbb{R}^N$

$$\begin{aligned} ||\mathcal{T}^\pi W_1 - \mathcal{T}^\pi W_2||_\infty &\leq \gamma ||W_1 - W_2||_\infty, \\ ||\mathcal{T} W_1 - \mathcal{T} W_2||_\infty &\leq \gamma ||W_1 - W_2||_\infty. \end{aligned}$$

4. *Fixed point*: *For any policy* $\pi$

$$V^\pi \text{ is the unique fixed point of } \mathcal{T}^\pi,$$
$$V^* \text{ is the unique fixed point of } \mathcal{T}.$$

# The Bellman Operators



**Proposition**

3. *Contraction in $L_\infty$-norm*: *for any* $W_1, W_2 \in \mathbb{R}^N$

$$
\begin{aligned}
||\mathcal{T}^\pi W_1 - \mathcal{T}^\pi W_2||_\infty &\leq \gamma ||W_1 - W_2||_\infty, \\
||\mathcal{T} W_1 - \mathcal{T} W_2||_\infty &\leq \gamma ||W_1 - W_2||_\infty.
\end{aligned}
$$

4. *Fixed point*: *For any policy* $\pi$

$V^\pi$ *is the unique fixed point of* $\mathcal{T}^\pi$,

$V^*$ *is the unique fixed point of* $\mathcal{T}$.

For any $W \in \mathbb{R}^N$ and any stationary policy $\pi$

$$
\begin{aligned}
\lim_{k\to\infty} (\mathcal{T}^\pi)^k W &= V^\pi, \\
\lim_{k\to\infty} (\mathcal{T})^k W &= V^*.
\end{aligned}
$$

# Proof: Contraction of the Bellman Operator

For any $s \in S$

$$
\begin{aligned}
&|\mathcal{T}W_1(s) - \mathcal{T}W_2(s)| \\
&= \Big| \max_a \big[ r(s,a) + \gamma \sum_{s'} p(s'|s,a) W_1(s') \big] - \max_{a'} \big[ r(s,a') + \gamma \sum_{s'} p(s'|s,a') W_2(s') \big] \Big| \\
&\leq \max_a \Big| \big[ r(s,a) + \gamma \sum_{s'} p(s'|s,a) W_1(s') \big] - \big[ r(s,a) + \gamma \sum_{s'} p(s'|s,a) W_2(s') \big] \Big| \\
&= \gamma \max_a \sum_{s'} p(s'|s,a) |W_1(s') - W_2(s')| \\
&\leq \gamma ||W_1 - W_2||_\infty \max_a \sum_{s'} p(s'|s,a) = \gamma ||W_1 - W_2||_\infty,
\end{aligned}
$$

■

👍 Same proof applies for $\mathcal{T}^\pi$

# Dynamic Programming

Bellman Equations

**Value Iteration**

Policy Iteration

# Value Iteration: the Idea

1. Let $V_0$ be *any* vector in $R^N$

# Value Iteration: the Idea

1. Let $V_0$ be *any* vector in $R^N$
2. At each iteration $k = 1, 2, \dots, K$

# Value Iteration: the Idea

1. Let $V_0$ be *any* vector in $R^N$
2. At each iteration $k = 1, 2, \ldots, K$
   - Compute $V_{k+1} = \mathcal{T} V_k$

# Value Iteration: the Idea

1. Let $V_0$ be *any* vector in $R^N$
2. At each iteration $k = 1, 2, \ldots, K$
   - Compute $V_{k+1} = \mathcal{T} V_k$
3. Return the *greedy* policy

$$\pi_K(s) \in \arg\max_{a \in A} \Big[ r(s,a) + \gamma \sum_{s'} p(s'|s,a) V_K(s') \Big].$$

# Value Iteration: the Guarantees



**Theorem**

*Let $V_0 \in \mathbb{R}^N$ be an arbitrary function, then the sequence of functions $\{V_k\}_k$ generated by value iteration* ***converges*** *to the optimal value function $V^*$.*

*Furthermore, let $\varepsilon > 0$ and $\max_{s,a} |r(s,a)| \le r_{\max} < \infty$, then after* ***at most***

$$K = \frac{\log(r_{\max}/\varepsilon)}{\log(1/\gamma)}$$

*iterations $\|V_K - V^*\|_\infty \le \varepsilon$.*

# Proof: Value Iteration

- From the *fixed point* property of $\mathcal{T}$ and $V_k = \mathcal{T} V_{k-1}$

$$\lim_{k \to \infty} V_k = \lim_{k \to \infty} \mathcal{T}^k V_0 = V^*$$

# Proof: Value Iteration

- From the *fixed point* property of $\mathcal{T}$ and $V_k = \mathcal{T} V_{k-1}$

$$\lim_{k \to \infty} V_k = \lim_{k \to \infty} \mathcal{T}^k V_0 = V^*$$

- From the *contraction* property of $\mathcal{T}$

$$\begin{aligned}
||V^* - V_{k+1}||_\infty & \\
&= ||\mathcal{T} V^* - \mathcal{T} V_k||_\infty && \text{[value iteration and Bellman eq.]} \\
&\leq \gamma ||V_k - V^*||_\infty && \text{[contraction]} \\
&\leq \gamma^{k+1} ||V^* - V_0||_\infty && \text{[recursion.]} \\
&\to 0
\end{aligned}$$

# Proof: Value Iteration

- From the *fixed point* property of $\mathcal{T}$ and $V_k = \mathcal{T} V_{k-1}$

$$\lim_{k\to\infty} V_k = \lim_{k\to\infty} \mathcal{T}^k V_0 = V^*$$

- From the *contraction* property of $\mathcal{T}$

$$\begin{aligned}
||V^* - V_{k+1}||_\infty & \\
&= ||\mathcal{T}V^* - \mathcal{T}V_k||_\infty && \textit{[value iteration and Bellman eq.]} \\
&\leq \gamma||V_k - V^*||_\infty && \textit{[contraction]} \\
&\leq \gamma^{k+1}||V^* - V_0||_\infty && \textit{[recursion.]} \\
&\to 0
\end{aligned}$$

- *Convergence rate*. Let $\varepsilon > 0$ and $||r||_\infty \leq r_{\max}$, then after *at most*

$$||V^* - V_{k+1}||_\infty \leq \gamma^{k+1}||V^* - V_0||_\infty < \varepsilon \quad \Rightarrow \quad K \geq \frac{\log(r_{\max}/\varepsilon)}{\log(1/\gamma)}$$

# Value Iteration: the Guarantees

## Corollary

*Let $V_K$ the function computed after $K$ iterations by value iteration, then the greedy policy*

$$\pi_K(x) \in \arg\max_{a \in A} \Big[ r(x,a) + \gamma \sum_y p(y|x,a) V_K(y) \Big]$$

*is such that*

$$\underbrace{\|V^* - V^{\pi_K}\|_\infty}_{\text{performance loss}} \leq \frac{2\gamma}{1-\gamma} \underbrace{\|V^* - V_K\|_\infty}_{\text{approx. error}}.$$

*Furthermore, there exists $\epsilon > 0$ such that if $\|V_K - V^*\|_\infty \leq \epsilon$, then $\pi_K$ is optimal.*

# Proof: Performance Loss

$$
\begin{aligned}
\|V^* - V^\pi\|_\infty &\le \|\mathcal{T}V^* - \mathcal{T}^\pi V\|_\infty + \|\mathcal{T}^\pi V - \mathcal{T}^\pi V^\pi\|_\infty \\
&\le \|\mathcal{T}V^* - \mathcal{T}V\|_\infty + \gamma\|V - V^\pi\|_\infty \\
&\le \gamma\|V^* - V\|_\infty + \gamma(\|V - V^*\|_\infty + \|V^* - V^\pi\|_\infty) \\
&\le \frac{2\gamma}{1-\gamma}\|V^* - V\|_\infty.
\end{aligned}
$$

■

# Value Iteration: the Idea

*Termination condition*

$$\max_s |V_k(s) - V_{k-1}(s)| - \min_s |V_k(s) - V_{k-1}(s)| \leq \varepsilon$$

*Performance guarantees*

$$\underbrace{\|V^* - V^\pi\|_\infty}_{\text{performance loss}} \leq \frac{\gamma}{1-\gamma}\varepsilon$$

# Value Iteration: the Idea

At each iteration $k = 1, 2, \ldots, K$

$\Rightarrow$ *Compute* $V_{k+1} = \mathcal{T} V_k$

1. For all $s \in S$
   1. Set maxV $= -\infty$
   2. For all $a \in A$
      1. Compute $q(s,a) = r(s,a) + \gamma \sum_{s'} p(s'|s,a) V_k(s)$
      2. If $q(s,a) >$ maxV, set maxV $= q(s,a)$
   3. Set $V_{k+1}(s) =$ maxV

# Value Iteration: the Complexity

*Time* complexity

- Each iteration takes $O(S^2 A)$ operations

$$V_{k+1}(s) = \mathcal{T} V_k(s) = \max_{a \in A} \Big[ r(s,a) + \gamma \sum_{s'} p(s'|s,a) V_k(s') \Big]$$

- The computation of the greedy policy takes $O(S^2 A)$ operations

$$\pi_K(s) \in \arg\max_{a \in A} \Big[ r(s,a) + \gamma \sum_{s'} p(s'|s,a) V_K(s') \Big]$$

- Total time complexity $O(KS^2 A)$

*Space* complexity

- Storing the MDP: dynamics $O(S^2 A)$ and reward $O(SA)$.
- Storing the value function and the optimal policy $O(S)$.

# Value Iteration: Extensions and Implementations

*Asynchronous VI.*

1. Let $V_0$ be any vector in $R^N$
2. At each iteration $k = 1, 2, \ldots, K$
   - **Choose a state $s_k$**
   - Compute $V_{k+1}(s_k) = \mathcal{T}V_k(s_k)$
3. Return the greedy policy

$$\pi_K(s) \in \arg\max_{a \in A} \Big[r(s,a) + \gamma \sum_{s'} p(s'|s,a) V_K(s')\Big].$$

*Comparison*

- Reduced time complexity to $O(SA)$
- Using round-robin, number of iterations increased by at most $O(KS)$ but much smaller in practice if states are properly *prioritized*
- Convergence guarantees if no *starvation*

# State-Action Value Function



**Definition**

In discounted infinite horizon problems, for any policy $\pi$, the *state-action value function* (or Q-function) $Q^\pi : S \times A \mapsto \mathbb{R}$ is

$$Q^\pi(s,a) = \mathbb{E}\Big[\sum_{t=0}^{\infty} \gamma^t r(s_t,a_t) | s_0 = s, a_0 = a, a_t = \pi(s_t), \forall t \geq 1\Big],$$

The optimal Q-function is

$$Q^*(s,a) = \max_\pi Q^\pi(s,a),$$

and the optimal policy can be obtained as

$$\pi^*(s) = \arg\max_a Q^*(s,a).$$

# State-Action Value Function Operators*

$$\mathcal{T}^\pi Q(s,a) = r(s,a) + \gamma \sum_{s'} p(s'|s,a) Q(s', \pi(s'))$$

$$\mathcal{T} Q(s,a) = r(s,a) + \gamma \sum_{s'} p(s'|s,a) \max_{a} Q(s', a)$$

*Abuse of notation for the operators

# State-Action and State Value Function



$$Q^\pi(s,a) = r(s,a) + \gamma \sum_{s'} p(s'|s,a) V^\pi(s')$$

$$V^\pi(s) = Q^\pi(s,\pi(s))$$

$$Q^*(s,a) = r(s,a) + \gamma \sum_{s'} p(s'|s,a) V^*(s')$$

$$V^*(s) = Q^*(s,\pi^*(s')) = \max_{a \in A} Q^*(s,a)$$

# Value Iteration: Extensions and Implementations

*Q-iteration.*

1. Let $Q_0$ be any Q-function
2. At each iteration $k = 1, 2, \ldots, K$
   - Compute $Q_{k+1} = \mathcal{T} Q_k$
3. Return the greedy policy

$$\pi_K(s) \in \arg\max_{a \in A} Q_K(s, a)$$

*Comparison*

- Increased space and time complexity to $O(SA)$ and $O(S^2 A^2)$
- Reduced time complexity to compute the greedy policy $O(SA)$
- *Bonus:* computing the greedy policy from the $Q$-function **does not require** the MDP

# Dynamic Programming

Bellman Equations

Value Iteration

**Policy Iteration**

# Policy Iteration: the Idea

1. Let $\pi_0$ be *any* stationary policy

# Policy Iteration: the Idea

1. Let $\pi_0$ be *any* stationary policy
2. At each iteration $k = 1, 2, \ldots, K$

# Policy Iteration: the Idea



1. Let $\pi_0$ be *any* stationary policy
2. At each iteration $k = 1, 2, \ldots, K$
   - *Policy evaluation* given $\pi_k$, compute $V^{\pi_k}$.

# Policy Iteration: the Idea

1. Let $\pi_0$ be *any* stationary policy
2. At each iteration $k = 1, 2, \ldots, K$
   - *Policy evaluation* given $\pi_k$, compute $V^{\pi_k}$.
   - *Policy improvement*: compute the *greedy* policy

$$\pi_{k+1}(s) \in \arg\max_{a \in A}\Big[r(s,a) + \gamma \sum_{s'} p(s'|s,a) V^{\pi_k}(s')\Big].$$

# Policy Iteration: the Idea

1. Let $\pi_0$ be *any* stationary policy
2. At each iteration $k = 1, 2, \ldots, K$
   - *Policy evaluation* given $\pi_k$, compute $V^{\pi_k}$.
   - *Policy improvement*: compute the *greedy* policy

$$\pi_{k+1}(s) \in \arg\max_{a \in A} \Big[ r(s,a) + \gamma \sum_{s'} p(s'|s,a) V^{\pi_k}(s') \Big].$$

3. Stop if $V^{\pi_k} = V^{\pi_{k-1}}$

# Policy Iteration: the Idea

1. Let $\pi_0$ be *any* stationary policy
2. At each iteration $k = 1, 2, \ldots, K$
   - *Policy evaluation* given $\pi_k$, compute $V^{\pi_k}$.
   - *Policy improvement*: compute the *greedy* policy

$$\pi_{k+1}(s) \in \arg\max_{a \in A}\Big[r(s,a) + \gamma \sum_{s'} p(s'|s,a) V^{\pi_k}(s')\Big].$$

3. Stop if $V^{\pi_k} = V^{\pi_{k-1}}$
4. Return the last policy $\pi_K$

# Policy Iteration: the Guarantees



**Proposition**

*The policy iteration algorithm generates a sequences of policies with non-decreasing performance*

$$V^{\pi_{k+1}} \geq V^{\pi_k},$$

*and it converges to $\pi^*$ in a finite number of iterations.*

# Proof: Policy Iteration

From the definition of the Bellman operators and the greedy policy $\pi_{k+1}$

$$V^{\pi_k} = \mathcal{T}^{\pi_k} V^{\pi_k} \leq \mathcal{T} V^{\pi_k} = \mathcal{T}^{\pi_{k+1}} V^{\pi_k}, \tag{1}$$

# Proof: Policy Iteration

From the definition of the Bellman operators and the greedy policy $\pi_{k+1}$

$$V^{\pi_k} = \mathcal{T}^{\pi_k} V^{\pi_k} \le \mathcal{T} V^{\pi_k} = \mathcal{T}^{\pi_{k+1}} V^{\pi_k}, \tag{1}$$

and from the monotonicity property of $\mathcal{T}^{\pi_{k+1}}$, it follows that

$$\begin{aligned} V^{\pi_k} &\le \mathcal{T}^{\pi_{k+1}} V^{\pi_k}, \\ \mathcal{T}^{\pi_{k+1}} V^{\pi_k} &\le (\mathcal{T}^{\pi_{k+1}})^2 V^{\pi_k}, \\ &\dots \\ (\mathcal{T}^{\pi_{k+1}})^{n-1} V^{\pi_k} &\le (\mathcal{T}^{\pi_{k+1}})^n V^{\pi_k}, \\ &\dots \end{aligned}$$

# Proof: Policy Iteration

From the definition of the Bellman operators and the greedy policy $\pi_{k+1}$

$$V^{\pi_k} = \mathcal{T}^{\pi_k} V^{\pi_k} \leq \mathcal{T} V^{\pi_k} = \mathcal{T}^{\pi_{k+1}} V^{\pi_k}, \tag{1}$$

and from the monotonicity property of $\mathcal{T}^{\pi_{k+1}}$, it follows that

$$\begin{aligned} V^{\pi_k} &\leq \mathcal{T}^{\pi_{k+1}} V^{\pi_k}, \\ \mathcal{T}^{\pi_{k+1}} V^{\pi_k} &\leq (\mathcal{T}^{\pi_{k+1}})^2 V^{\pi_k}, \\ &\dots \\ (\mathcal{T}^{\pi_{k+1}})^{n-1} V^{\pi_k} &\leq (\mathcal{T}^{\pi_{k+1}})^n V^{\pi_k}, \\ &\dots \end{aligned}$$

Joining all the inequalities in the chain we obtain

$$V^{\pi_k} \leq \lim_{n \to \infty} (\mathcal{T}^{\pi_{k+1}})^n V^{\pi_k} = V^{\pi_{k+1}}.$$

# Proof: Policy Iteration

From the definition of the Bellman operators and the greedy policy $\pi_{k+1}$

$$V^{\pi_k} = \mathcal{T}^{\pi_k} V^{\pi_k} \le \mathcal{T} V^{\pi_k} = \mathcal{T}^{\pi_{k+1}} V^{\pi_k}, \tag{1}$$

and from the monotonicity property of $\mathcal{T}^{\pi_{k+1}}$, it follows that

$$\begin{aligned} V^{\pi_k} &\le \mathcal{T}^{\pi_{k+1}} V^{\pi_k}, \\ \mathcal{T}^{\pi_{k+1}} V^{\pi_k} &\le (\mathcal{T}^{\pi_{k+1}})^2 V^{\pi_k}, \\ &\dots \\ (\mathcal{T}^{\pi_{k+1}})^{n-1} V^{\pi_k} &\le (\mathcal{T}^{\pi_{k+1}})^n V^{\pi_k}, \\ &\dots \end{aligned}$$

Joining all the inequalities in the chain we obtain

$$V^{\pi_k} \le \lim_{n\to\infty} (\mathcal{T}^{\pi_{k+1}})^n V^{\pi_k} = V^{\pi_{k+1}}.$$

Then $(V^{\pi_k})_k$ is a non-decreasing sequence.

# Policy Iteration: the Guarantees

Since a finite MDP admits a finite number of policies, then the termination condition is eventually met for a specific $k$.
Thus eq. 1 holds with an equality and we obtain

$$V^{\pi_k} = \mathcal{T} V^{\pi_k}$$

and $V^{\pi_k} = V^*$ which implies that $\pi_k$ is an optimal policy. ■

# Policy Iteration: Complexity



*Notation.* For any policy $\pi$ the reward *vector* is $r^\pi(x) = r(x, \pi(x))$ and the transition *matrix* is $[P^\pi]_{x,y} = p(y|x, \pi(x))$

# Policy Iteration: Complexity

Policy Evaluation Step

- *Direct computation.* For any policy $\pi$ compute

$$V^\pi = (I - \gamma P^\pi)^{-1} r^\pi.$$

*Complexity:* $O(S^3)$.

# Policy Iteration: Complexity



Policy Evaluation Step

- *Direct computation.* For any policy $\pi$ compute

$$V^\pi = (I - \gamma P^\pi)^{-1} r^\pi.$$

*Complexity:* $O(S^3)$.

- *Iterative policy evaluation.* For any policy $\pi$

$$\lim_{n\to\infty} \mathcal{T}^\pi V_0 = V^\pi.$$

*Complexity:* An $\varepsilon$-approximation of $V^\pi$ requires $O\Big(S^2 \frac{\log(1/\epsilon)}{\log(1/\gamma)}\Big)$ steps.

# Policy Iteration: Complexity

Policy Evaluation Step

- *Direct computation.* For any policy $\pi$ compute

$$V^\pi = (I - \gamma P^\pi)^{-1} r^\pi.$$

*Complexity:* $O(S^3)$.

- *Iterative policy evaluation.* For any policy $\pi$

$$\lim_{n\to\infty} \mathcal{T}^\pi V_0 = V^\pi.$$

*Complexity:* An $\varepsilon$-approximation of $V^\pi$ requires $O\Big(S^2 \frac{\log(1/\epsilon)}{\log(1/\gamma)}\Big)$ steps.

- *Monte-Carlo simulation.* In each state $s$, simulate $n$ trajectories $((s_t^i)_{t\geq 0,})_{1\leq i\leq n}$ following policy $\pi$ and compute

$$\hat{V}^\pi(s) \simeq \frac{1}{n}\sum_{i=1}^{n}\sum_{t\geq 0}\gamma^t r(s_t^i, \pi(s_t^i)).$$

*Complexity:* In each state, the approximation error is $O\Big(\frac{r_{\max}}{1-\gamma}\sqrt{\frac{1}{n}}\Big)$

# Policy Iteration: Complexity

Policy Improvement Step

- If the policy is *evaluated with $V$*, then complexity $O(SA)$

# Policy Iteration: Complexity

Policy Improvement Step

- If the policy is *evaluated with* $V$, then complexity $O(SA)$
- If the policy is *evaluated with* $Q$, then complexity $O(A)$

$$\pi_{k+1}(s) \in \arg\max_{a\in A} Q^{\pi_k}(s,a),$$

Number of Iterations

- At most $O\left(\frac{SA}{1-\gamma}\log(\frac{1}{1-\gamma})\right)$
- Other results exist that do not depend on $\gamma$

# Comparison between Value and Policy Iteration



*Value Iteration*

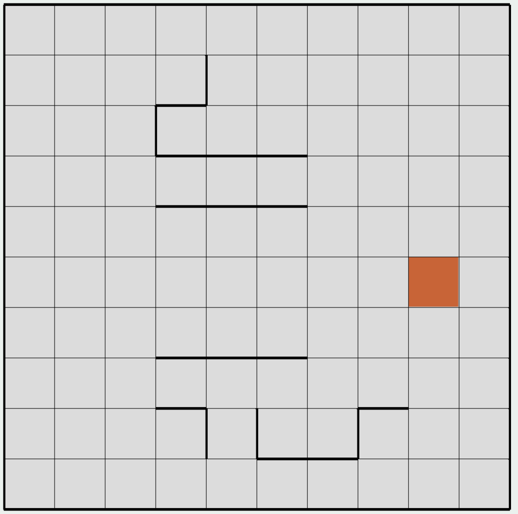
- *Pros:* each iteration is very *computationally efficient*.
- *Cons:* convergence is only *asymptotic*.

*Policy Iteration*

- *Pros:* converge in a *finite* number of iterations (often small in practice).
- *Cons:* each iteration requires a full *policy evaluation* and it might be expensive.

# The Grid-World Problem





# Other Algorithms

- Linear programming
- Modified Policy Iteration
- $\lambda$-Policy Iteration
- Primal-dual formulations

# Summary

- Bellman equations and Bellman operators (and their properties)
- Value iteration (algorithm, guarantees, and complexity)
- Policy iteration (algorithm, guarantees, and complexity)

# Bibliography

[1] Dimitri P. Bertsekas. *Dynamic Programming and Optimal Control, Vol. II*. Athena Scientific, 3rd edition, 2007.

# Thank you!

facebook
Artificial Intelligence Research