Chapter 3: Markov decision processes (MDP)

SB: Chap 3

3.1. Model

- Environment state: $S_t \in \mathcal{S}$ $\quad t = 0, 1, \ldots$
  - System to control
  - Info for making decision

- Agent state: $A_t \in \mathcal{A}(s)$ $\quad t = 0, 1, \ldots$
  - Controller
  - Action state/decision — available actions from given state
  - State space can depend on current state
  - Simplification: $\mathcal{A}(s) = \mathcal{A} \quad \forall s$

- Reward: $R_t \in \mathbb{R}$
  - Signal from system/environment
  - Defines good/bad actions ($+$ / $-$)
  - Zero/baseline not important
  - RV correlated with states/actions
  - Sometimes expressed as $r(s, a, s')$

- Transition probability (dynamics):

$$p(s', r \mid s, a) = P(S_{t+1} = s', R_{t+1} = r \mid S_t = s, A_t = a)$$

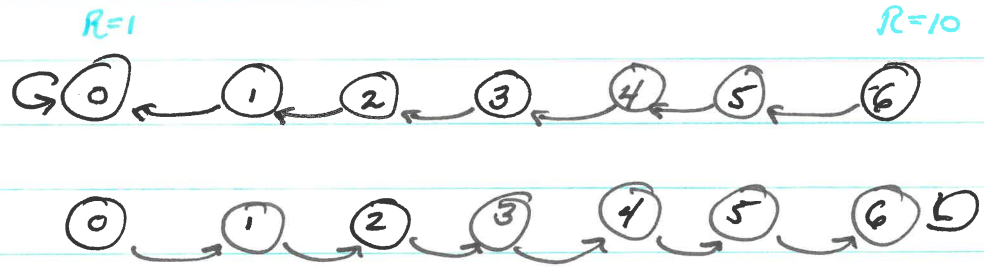
$$\begin{tikzcd}
\text{Reward} & & R_1 & R_2 & \\
\text{System} & S_0 \arrow[r] & S_1 \arrow[r] & S_2 \arrow[r] & \cdots \\
\text{Agent} & A_0 \arrow[ur] & A_1 \arrow[ur] & &
\end{tikzcd}$$

- Assumed homogeneous (time-independent, stationary)

· Example:

a = 0

a = 1



- 2 deterministic actions: left, right
- Deterministic reward given state — and doesn't depend on action (same for a=0,1)

· State transition probability:

$$p(s'|s,a) = \sum_r p(s',r|s,a)$$

Transition matrix for each a

$p_a(j|i)$ $\quad P_a$

· Reward probability:

$$p(r|s,a) = \sum_{s'} p(s',r|s,a)$$

↑ action

compare with MRP

· Expected reward: $\rho(s,a) = E[R_{t+1} | S_t = s, A_t = a]$

$$\begin{array}{ll} & R_{t+1} \\ s = S_t \longrightarrow & S_{t+1} \\ a = A_t \nearrow & \end{array}$$

$$\rho(s,a) = \sum_r r \, p(r|s,a)$$

$$= \sum_{r,s'} r \, p(s',r|s,a)$$

- One-step reward from state s and action a
- Doesn't depend on time (stationary)

3.2. Policy

- How to choose action/control/decision at given state?

- Policy transition probability:

$$\underset{\substack{\uparrow \\ \text{action}}}{} \pi(a|s) = P(A_t = a \,|\, S_t = s)$$

$$\pi(a|s) = P(\underset{\substack{\uparrow\\ \text{action}}}{A_t} = a \,|\, \underset{\substack{\uparrow\\ \text{state}}}{S_t} = s)$$

$$\begin{tikzcd}
S_0 \arrow[rr] \arrow[dr, "\pi"'] & & S_1 \arrow[rr] \arrow[dr, "\pi"'] & & S_2 \arrow[r] & \cdots \\
& A_0 \arrow[ur, "p"] & & A_1 \arrow[ur, "p"] & &
\end{tikzcd}$$

  - Mapping state → action
  - Probabilistic policy: actions can be are random
  - Assumed homogeneous (time-independent, stationary)

$\left\{ \begin{array}{l} \text{noise in environment} \\ \text{exploration} \end{array} \right.$

- Deterministic policy: $\underset{\text{one state}}{\pi(s)} = \underset{\text{one action}}{a}$

- Joint probability: $p(s', r, a|s) = p(s', r|s, a)\,\pi(a|s)$

- System transition probability:

$$P_\pi(s'|s) = \sum_a p(s'|s,a)\,\pi(a|s)$$

Average over actions

$$= \sum_{a,r} p(s', r|s,a)\,\pi(a|s)$$

  - Effective dynamics under policy π
  - Transition matrix: P_π $\quad (P_\pi)_{ij} = p_\pi(j|i)$

$$S_0 \xrightarrow{P_\pi} S_1 \xrightarrow{P_\pi} S_2 \xrightarrow{P_\pi} \cdots$$

· Comparison:

· No control: $S_0 \xrightarrow{P} S_1 \xrightarrow{P} S_2 \to \cdots$ $\qquad p(s'|s)$

· Open-loop control:

$$\begin{tikzcd} S_0 \arrow[rr, "P"'] & & S_1 \arrow[rr, "P"'] & & S_2 \arrow[r] & \cdots \\ A_0 \arrow[ur] & & A_1 \arrow[ur] & & & \end{tikzcd}$$

$p(s'|s,a)$

· Closed-loop control: feedback

$$\begin{tikzcd} S_0 \arrow[rr] \arrow[d, "\pi"'] & & S_1 \arrow[rr] \arrow[d, "\pi"'] & & S_2 \arrow[r] & \cdots \\ A_0 \arrow[urr, "P"] & & A_1 \arrow[urr, "P"] & & & \end{tikzcd}$$

$p(s'|s,a)$
$\pi(a|s)$

· Reduced dynamics: $S_0 \xrightarrow{P_\pi} S_1 \xrightarrow{P_\pi} S_2 \xrightarrow{P_\pi} \cdots$ $\qquad p_\pi(s'|s)$

· Expected reward under policy π:

$$\rho_\pi(s) = E_\pi[R_{t+1} | S_t = s]$$
$$= E_\pi \, E[R_{t+1} | S_t = s, A_t]$$
$$= E_\pi[\rho(s, A_t)]$$

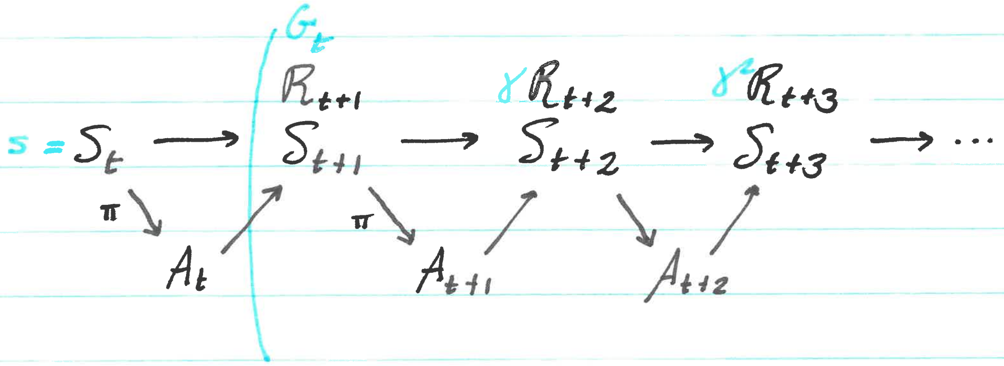
$$\rho_\pi(s) = \sum_a \rho(s,a)\, \pi(a|s)$$ average over actions
$$= \sum_{a,r} r \; p(r|s,a)\, \pi(a|s)$$
$$= \sum_{a,r,s'} r \; p(s',r|s,a)\, \pi(a|s)$$

· One-time 1-step reward from state s under policy π
· Doesn't depend on time (stationary)
· $\rho(s,a)$ doesn't depend on π
· Deterministic: $\rho_\pi(s) = \rho(s, \pi(s))$

3.3. Value functions

· State value function:

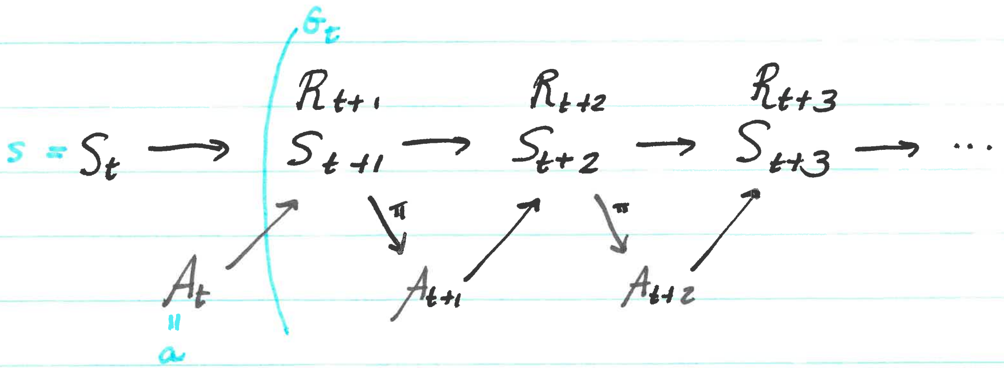
$$v_\pi(s) = E_\pi[\overset{\text{return}}{G_t} \mid S_t = s]$$



- Future return / cost to go from state s + policy π
- Long term value of s under π
- Function of $\pi(\cdot \mid s)$
- Doesn't depend on time t (stationary dynamics, infinite return)

· State-action value function:

$$q_\pi(s,a) = E_\pi[G_t \mid S_t = s, A_t = a]$$



- Return / cost to go if action a taken from state s
- Value of action a when in s if policy π followed after
- Doesn't depend on time t

· Connection :

$$v_\pi(s) = E_\pi[\, q_\pi(s, A_t) \mid S_t = s \,]$$ average over actions

$$= \sum_a q_\pi(s,a)\, \pi(a|s)$$

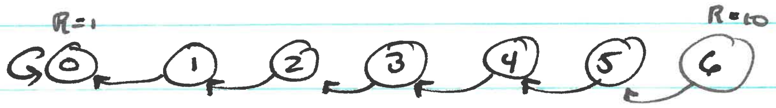
$$v_\pi(s) = q_\pi(s, \pi(s))$$ for deterministic policy

$a = \pi(s)$

· Example :

$a = 0$

Move Left

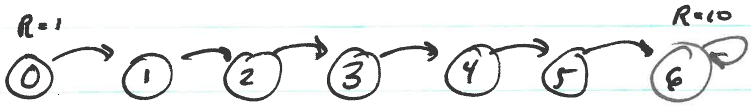


$p(s'|s, a=0)$

$$\begin{array}{cc} & j \\ i & \begin{pmatrix} \quad \end{pmatrix} \end{array}$$

$a = 1$

Move Right



$p(s'|s, a=1)$

$$\begin{array}{cc} & j \\ i & \begin{pmatrix} \quad \end{pmatrix} \end{array}$$

· Policy 1 : $\pi_1(s) = 0 \quad \forall s$ move left from all state deterministic

· Policy 2 : $\pi_2(s) = 1 \quad \forall s$ move right "

· Policy 3 : $\pi_3 = \frac{1}{2}\pi_1 + \frac{1}{2}\pi_2$

random policy/control

CW : Calculate $v_{\pi_1}, v_{\pi_2}, v_{\pi_3}$

· Note : $\rho_\pi(s) = 1$ or 10 $\quad s=0, s=6 \quad \forall \pi$

$\rho(s,a) = 1$ or 10 $\quad s=0, s=6 \quad \forall a$

reward only depends on S

· Note : 2 actions per state $\Rightarrow$ 2^7 possible policies

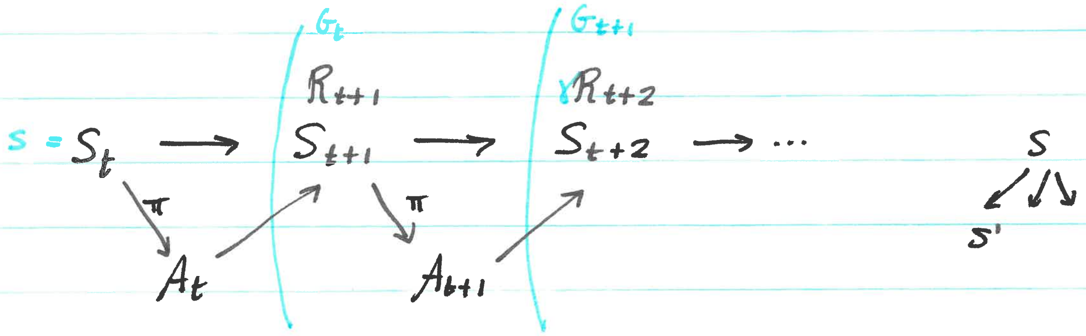
3.4. Bellman equations (BE)

· Return : $G_t = R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \cdots = \underline{R_{t+1} + \gamma G_{t+1}}$

· Bellman equation for value function :

$$v_\pi(s) = E_\pi\left[R_{t+1} + \gamma v_\pi(S_{t+1}) \mid S_t = s \right]$$

Compare with MRP



$$v_\pi(s) = \sum_{\cancel{s'},a} p(s,a)\,\pi(a|s) + \gamma \sum_{s',a} v_\pi(s')\, p(s'|s,a)\, \pi(a|s$$

$$= \sum_{s',r,a} r\, p(s',r|s,a)\, \pi(a|s)$$

$$+ \gamma \sum_{s',r,a} v_\pi(s')\, p(s',r|s,a)\, \pi(a|s)$$

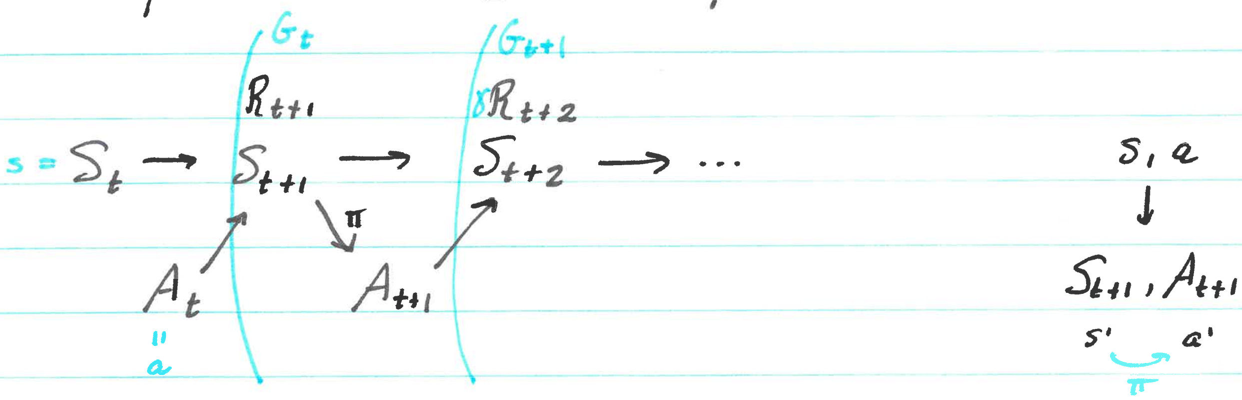
· $\underset{s}{\text{Value}}$ = $\underset{s}{\text{immediate reward}}$ + discounted value of successor state s'

· Linear equation : $v_\pi = \rho_\pi + \gamma P_\pi v_\pi$

· Solution : $v_\pi = (I - \gamma P_\pi)^{-1} \rho_\pi$

· BE for action-value function:

$$q_\pi(s,a) = E_\pi\left[R_{t+1} + \gamma\, q_\pi(S_{t+1}, A_{t+1}) \mid S_t = s, A_t = a \right]$$



$q_\pi(s,a)$ $\qquad\qquad$ $q_\pi(S_{t+1}, A_{t+1})$

$$q_\pi(s,a) = \rho(s,a) + \gamma \sum_{a',s'} q_\pi(s',a')\, p(s'|s,a)\, \pi(a'|s')$$

$$= \sum_{s',r} r\, p(s',r|s,a)$$

$$+ \gamma \sum_{a',s',r} q_\pi(s',a')\, p(s',r|s,a)\, \pi(a'|s')$$

- Value (s,a) = immediate reward (s,a) + discounted value of successor state, action (s',a')

- Linear equation for "matrix" q_π

· Two ways of defining MDPs

① ②

$p(s', r \mid s, a)$

$\hookrightarrow p(r \mid s, a)$

$\hookrightarrow \rho(s,a) = E[R_{t+1} \mid S_t = s, A_t = a]$ ρ

$\hookrightarrow p(s' \mid s, a)$

$\hookrightarrow P_a$ P_a

$\pi(a \mid s)$

$\hookrightarrow \rho_\pi(s) = E_\pi[R_{t+1} \mid S_t = s]$ ρ_π

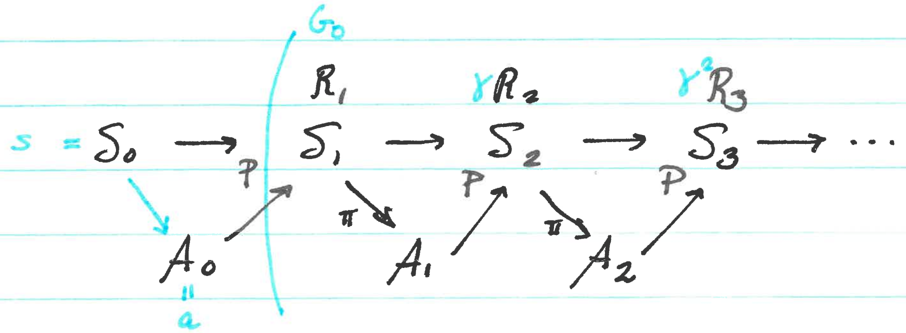
$\hookrightarrow p_\pi(s' \mid s)$

$\hookrightarrow P_\pi$ P_π

$$v_\pi = \rho_\pi + \gamma P_\pi v_\pi$$

$$q_\pi = \rho + \gamma (P_a \pi q_\pi)$$

3.5 Optimal policies



Dynamics	Policy/control	Reward
$p(s' \mid s,a)$	$\pi(a \mid s)$	$p(s',r \mid s,a)$

$$v_\pi(s) = E_\pi[G_t \mid S_t = s] = E_\pi[G_0 \mid S_0 = s]$$

$$q_\pi(s,a) = E_\pi[G_t \mid S_t = s, A_t = a] = E_\pi[G_0 \mid S_0 = s, A_0 = a]$$

infinite horizon return

- Goal: Find π with max expected return
- Optimal value function:

$$v_*(s) = \max_\pi v_\pi(s)$$

  - Best actions/policies from s
  - At least one solution

- Optimal action-value function:

$$q_*(s,a) = \max_\pi q_\pi(s,a)$$

  - Optimal policy for successor states from s,a
  - Same solution π_* as v_*

- Optimal policy: $\pi_* = \arg\max_\pi v_\pi(s)$

- <u>General results</u> (Silver: L2): For any finite MDPs
  - There exists at least one optimal policy π_*
  - All optimal π_* achieve v_*
  - " " " " q_*
  - Optimal policy is deterministic (one action per state)
  - " " " stationary

$$\Rightarrow \quad \pi_*(s) = a$$

- Optimal policy:

$$\pi_*(s) = \arg\max_a \; q_*(s,a)$$ = best action from s

- Optimal value function:

best action

$$v_*(s) = v_{\pi_*}(s) = q_*(s, \pi_*(s))$$

or

$$v_*(s) = \max_a \; q_*(s,a)$$ best action

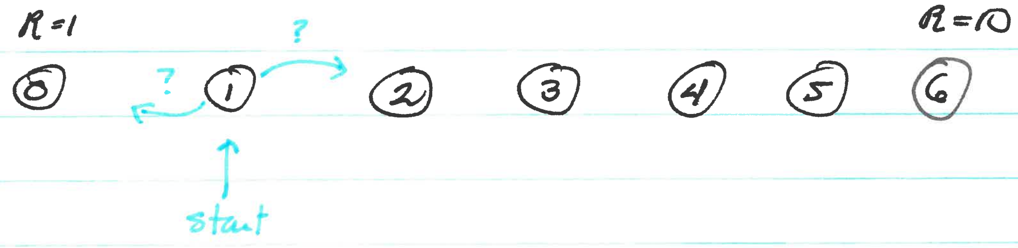
$$\pi_* \Longleftrightarrow q_* \Longleftrightarrow v_*$$

- Note: $v_\pi(s) = \sum_a q_\pi(s,a)\,\pi(a|s)$

See p. 3-6

$$= q_\pi(s, \pi(s))$$ deterministic action

- Example:



$\gamma \approx 0$: Better to move left to get $R=1$ reward

$\gamma \approx 1$: " " " right " " $R=10$ "

Optimal policy: for each s, either left or right
$p(s'|s,a=0)$ $p(s'|s,a=1)$

π_* among 2^7 possible policies

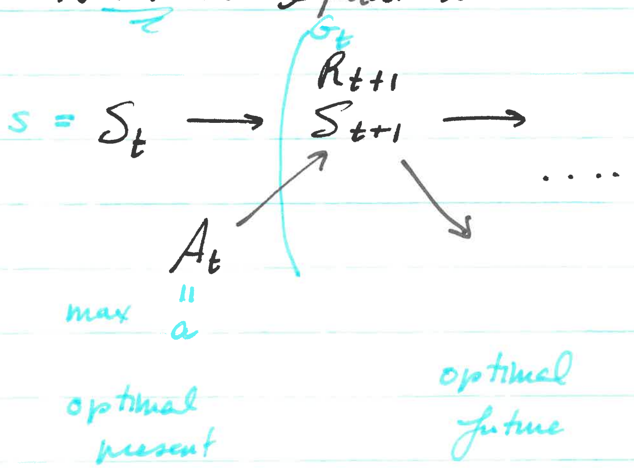
3.6 Bellman optimality equations

See 3-12b

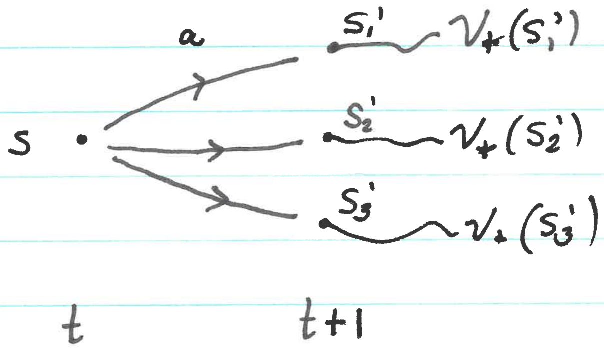
- Value function:

$$v_*(s) = \max_a E\left[R_{t+1} + \gamma v_*(S_{t+1}) \mid S_t = s, A_t = a\right]$$

- Optimal from s = optimal action from s and then optimal policy after
- Expectation on one step – no E_π
- max outside expectation – influences reward
- arg max defines $\pi_*(s)$
- Nonlinear equation because of max



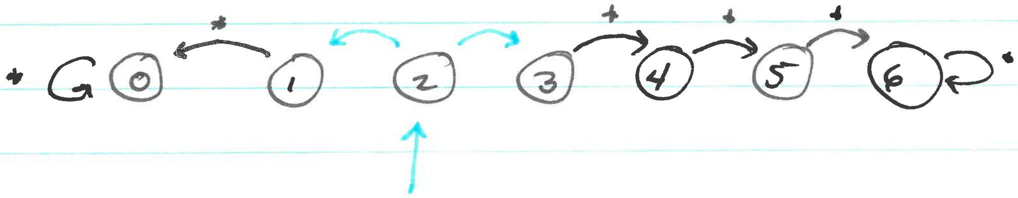
$$v_*(s) = \max_a p(s,a) + \gamma \sum_{s'} v_*(s') p(s'|s,a)$$



- Whatever action chosen at time t, rest of dynamics is optimal
- To be optimal at $S_t = s$, choose optimal action and then follow optimal policy from S_{t+1}

$$\Rightarrow \quad v_*(s) = \max_a E[R_{t+1} + \gamma v_*(S_{t+1}) \mid S_t = s, A_t = a]$$

- Example:



· Action value function:

$$q_*(s,a) = E[R_{t+1} + \gamma \max_{a'} q_*(S_{t+1}, a') \mid S_t = s, A_t = a]$$

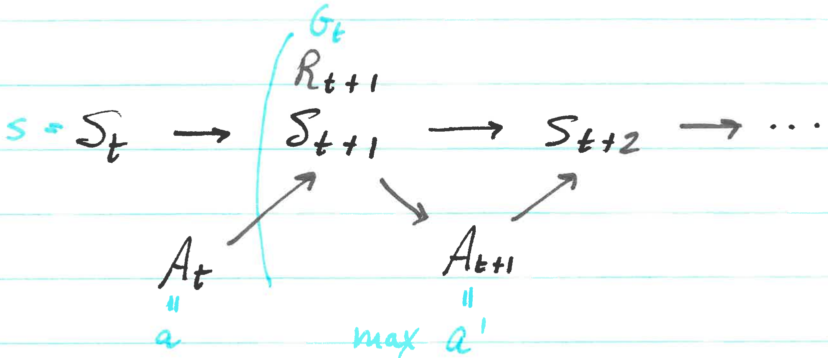
action after

$$= E[R_{t+1} + \gamma v_*(S_{t+1}) \mid S_t = s, A_t = a]$$

i.e.

$$q_*(s,a) = \rho(s,a) + \gamma \max_{a'} \sum_{s'} q(s',a')\, p(s'|s,a)$$

$$= \rho(s,a) + \gamma \sum_{s'} v_*(s')\, p(s'|s,a)$$



· Optimal from s,a = optimal after reaching S_{t+1} by taking optimal action after

· Max inside – doesn't influence reward

· arg max defines $\pi_*(s')$

· Nonlinear equation because of max

· Bellman's optimality principle:
An optimal policy has the property that whatever the initial state / decision are, the remaining decisions constitute an optimal policy with regard to the state resulting from the first decision.