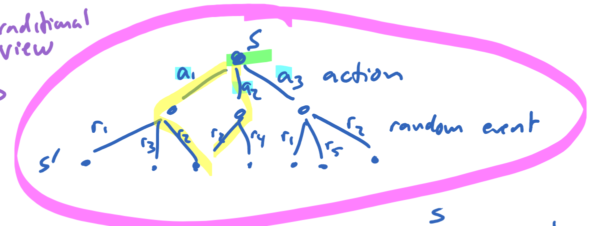
Markov Decision Process

Markov Decision Process : (S, A, p, R)

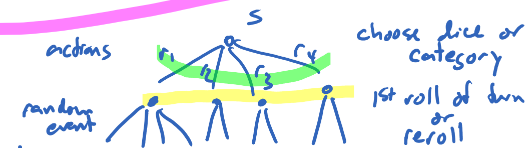
states (positions) — rewards — probability — actions (moves)

traditional view of MDP



$p: S \times A \times S \times R \rightarrow [0,1]$

state, action, new state, reward



p (initial state w/ 1st rolled 16666 , keep 4 6's , initial state after 1st reroll 56666 , 0) $= \frac{1}{6}$

for any s, a

$$\sum_{s', r} p(s, a, s', r) = 1$$

p (initial state w/ 1st roll 16666 , keep 4 6's, end of game , 0) $= 0$

because you can't do this in one step

because you never get 39 points when rolling

p (initial state w/ 1st rolled 16666 , keep 4 6's , initial state after 1st reroll 56666 , 39) $= 0$

initial state — action — resulting state — reward

Episode : $(S_0, a_0, S_1, R_1), (S_1, a_1, S_2, R_2), \ldots, (S_{T-1}, a_{T-1}, S_T, R_T), [\ldots\ldots]$

terminal state

$G = R_1 + R_2 + \cdots + R_T$

(S_0, a_0, S_1', R_1')

goal: maximize total reward (for finite process) ← take $\gamma = 1$

maximize total discounted reward

↳ value immediate reward over future

$$\sum_{t=1}^{\infty} R_t \cdot \gamma^{t-1} \qquad R_1 + \underbrace{\gamma}_{\text{discount factor}} R_2 + \gamma^2 R_3 + \ldots$$

Policy : $\pi : S \times A \rightarrow [0,1]$ s.t. $\sum_a \underbrace{\pi(a|s)}_{\pi(s,a)} = 1$

deterministic policy $\pi(a|s) = 1$ for one action

$\pi_{determ} : S \rightarrow A$

$\pi_{determ}(s) = a$

expected future reward given curr state s, following π

Value : $v_\pi(s) = \sum_a \pi(a|s) \cdot \sum_{s'} \sum_r p(s', r | s, a) \cdot (r + \gamma v_\pi(s'))$

outcomes of reroll — next state — rewards — immediate reward — discounted future reward given new state s'

expected future reward from s, choosing a, then following policy π — $q_\pi(s,a)$

value of state under policy π

optimal value

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

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

$v^*(s) = \max_{a \in A(s)} q^*(s,a) = \max_a \max_\pi q_\pi(s,a)$

$q^*(s, a_1) = 5$ — $\frac{1}{2}$

$q^*(s, a_2) = 5$ — $\frac{1}{2}$

$q^*(s, a_3) = 5$

$\pi(s, a_1) = \frac{1}{2}$

$\pi(s, a_2) = \frac{1}{2}$

$$v^*(s) = \max_{a \in A(s)} q^*(s,a) = \max_a \max_\pi q_\pi(s,a)$$

$$= \max_a \sum_{s',r} p(s',r|s,a) \cdot [r + \gamma v^*(s')]$$

Bellman equation

recurrence

so for a finite process compute recursively — base cases = terminal states $v^*(t) = 0$

other cases $v^* =$

or dynamic programming

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

$\pi(s, a_1) = \frac{1}{2}$

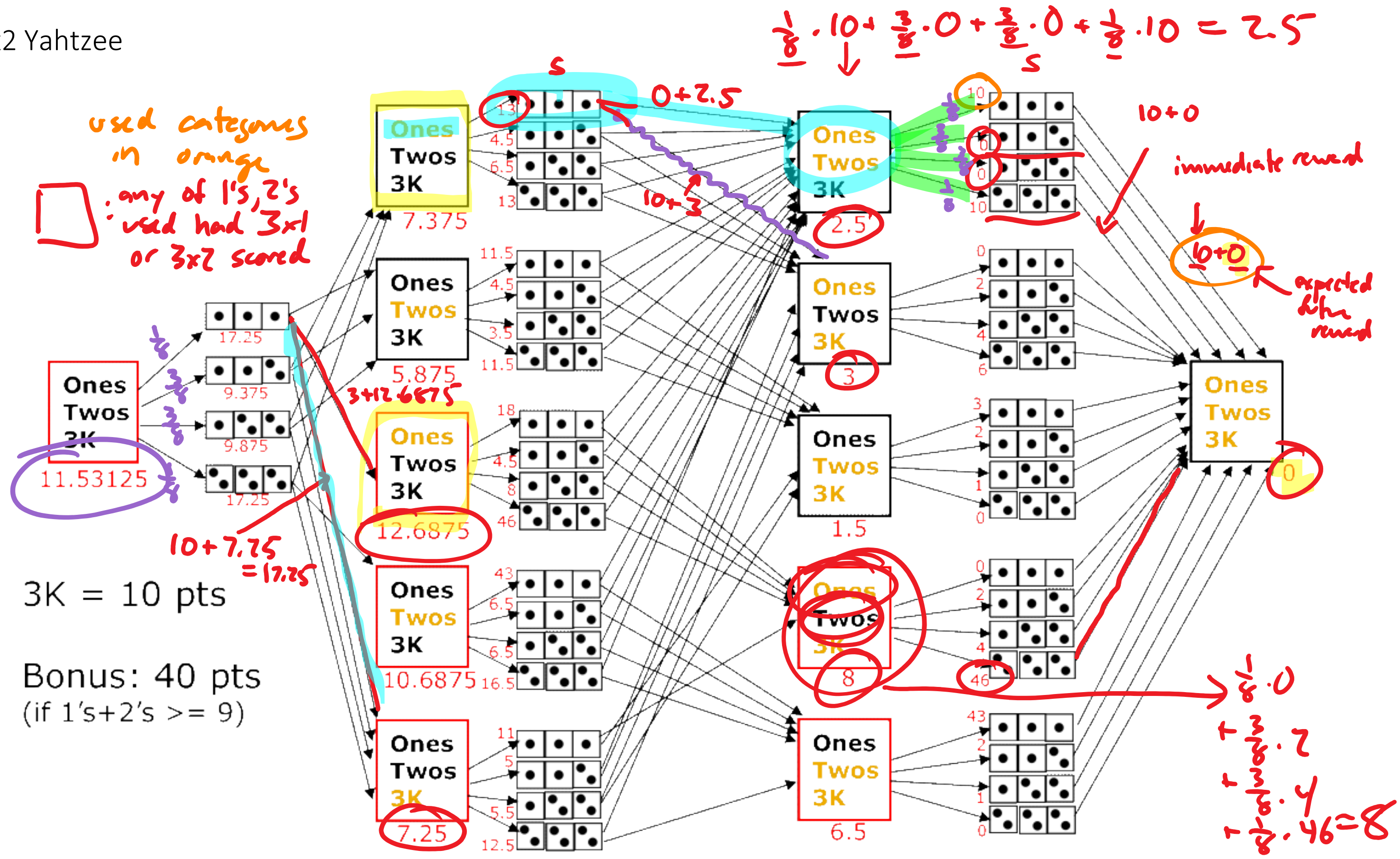
$\pi(s, a_2) = \frac{1}{2}$

$v_\pi(s) = \frac{1}{2} \cdot q^*(s,a_1) = \frac{5}{2}$

$+ \frac{1}{2} \cdot q^*(s,a_2) = \frac{3}{2}$

$\pi'(s,a_1) = 1 = 5$

3x2 Yahtzee



3K = 10 pts

Bonus: 40 pts
(if 1's+2's >= 9)

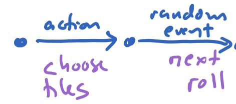
Shut The Box

8   1 ~~2~~ ~~3~~ ~~4~~ ~~5~~ ~~6~~ 7 ~~8~~ ~~9~~   P1

1   1 ~~2~~ ~~3~~ ~~4~~ ~~5~~ ~~6~~ ~~7~~ ~~8~~ ~~9~~

you win!

2	2
6	5
6	6
4	6
2	2
5	3
5	2
1	2
5	4
5	1
6	5



for P_2 : $V_2(s)$ = expected wins for P_2 (draw = $\frac{1}{2}$ win)

s = tuple (S, r, t)

S = set of remaining numbers; r = current roll; t = P1's score

$$V_2(S, r, t) = \begin{cases} 0.0 & \text{if no subset } S' \subseteq S \text{ has } \mathrm{sum}(S') = r \text{ and } \mathrm{sum}(S) > t \quad \text{can't move, higher score} \\ 0.5 & \text{if no subset } S' \subseteq S \text{ has } \mathrm{sum}(S') = r \text{ and } \mathrm{sum}(S) = t \quad \text{can't move, tie score} \\ 1.0 & \text{if no subset } S' \subseteq S \text{ has } \mathrm{sum}(S') = r \text{ and } \mathrm{sum}(S) < t \\ \max\limits_{\substack{S' \subseteq S \\ \mathrm{sum}(S') = r}} \underbrace{\sum_{r'=2}^{12} P(\text{roll } r')}_{\text{roll 2 dice on next turn}} \cdot \underbrace{V_2(S - S', r', t)}_{\text{value of resulting state}} & \text{if } \mathrm{sum}(S - S') > 6 \\ \max\limits_{\substack{S' \subseteq S \\ \mathrm{sum}(S') = r}} \underbrace{\sum_{r'=1}^{6} \frac{1}{6}}_{\text{roll 1 die}} \cdot V_2(S - S', r', t) & \text{if } \mathrm{sum}(S - S') \le 6 \end{cases}$$

max over $S' \subseteq S$, $\mathrm{sum}(S') = r$: all possible actions

(reward only at end)

for P_1 : $V_1(s)$ = expected wins for P_1

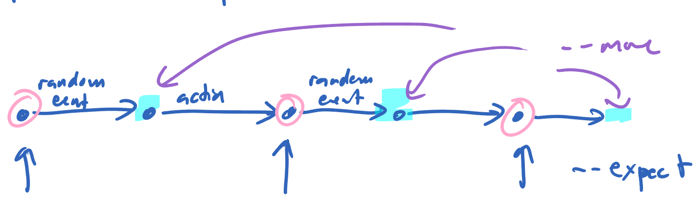
$$V_1(S, r) = \begin{cases} 1.0 & \text{if } S = \emptyset \quad \text{automatic win} \\ \max\limits_{\substack{S' \subseteq S \\ \mathrm{sum}(S' = r)}} \sum_{r'=2}^{12} P(\text{roll } r') \cdot V_1(S - S', r') & \text{if } \mathrm{sum}(S - S') > 6 \\ \max\limits_{\substack{S' \subseteq S \\ \mathrm{sum}(S' = r)}} \sum_{r'=1}^{6} P(\text{roll } r') \cdot V_1(S - S', r') & \text{if } \mathrm{sum}(S - S') \le 6 \\ \sum_{r'=2}^{12} P(\text{roll } r') \cdot \left(1 - V_2(\{1, .., 9\}, r', \mathrm{sum}(S))\right) & \text{if no } S' \subseteq S \text{ has } \mathrm{sum}(S') = r \end{cases}$$

r': outcomes of P2's 1st roll

$\{1, .., 9\}$: P2 starts w/ all tiles

$$P(\text{P1 wins} \mid \text{P1 starts turn w/ tiles } S) = \sum_r P(\mathrm{sum}(\text{roll}) = r) \cdot V_1(S, r)$$

$$P(\text{P2 wins} \mid \text{P2 starts turn w/ tiles } S, \text{P1 score } t) = \sum_r P(\mathrm{sum}(\text{roll}) = r) \cdot V_2(S, r, t)$$



$\uparrow$ $\uparrow$ $\uparrow$ -- expect