Reinforcement Learning

Lecture 11:
Partially Observable MDP

Yishay Mansour, Tel-Aviv University

Lecture 11: outline

- POMDP
  - Example
  - Model
- Belief state
  - Definition
  - Computation
- Value Iteration
- Policy
  - Policy Tree
  - Automata
- Reusable trajectories

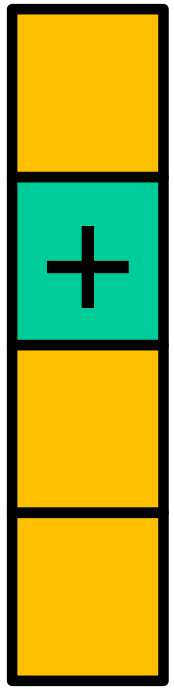
Partially Observable MDP

- MDP:
  - Observes the state
  - History not important
- Optimal policy:
  - Deterministic

- POMDP:
  - Observes a signal
    - Signal has only partial information
  - History has an influence
- Optimal policy:
  - Well, we will see …

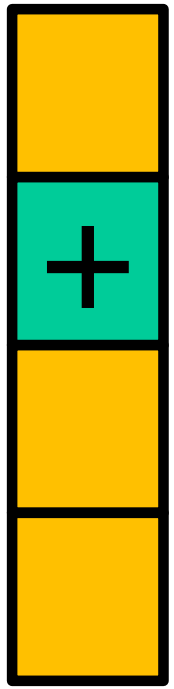
Example

- Goal: reach the green state
- Initially:
  - Random yellow state
- Action
  - Up/Down
    - Done with prob. 0.9
- Observation
  - Green/Yellow



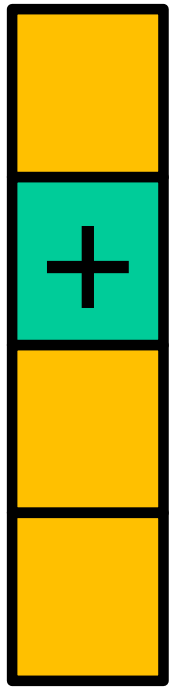
Example

- Initially
  - [0.33,0,0.33,0.33]
- After UP:
  - Observe Yellow
- What is the posterior?
  - [0.45,0,0.45,0.1]



Example

- Computing posterior:
  - [0.9,0.1,0,0]
  - [0,0.9,0,0.1]
  - [0,0,0.9,0.1]
- Averaging:
  - [0.3,0.333,0.3,0.066]
- Observe Yellow
  - [0.45,0,0.45,0.1]



POMDP: Model

- States
  - S
- Actions
  - A
- Transition
  - $p(s'|s,a)$
- Reward
  - $r(s,a)$
- Initial state s_0

- Observation
  - O
    - Finite or infinite
  - Observation probability
  - $Ob(o|s',a)$
  - For an MDP
    - $O = S$
    - $Ob(s'|s',a) = 1$

Lecture 11: outline

- POMDP
  - Example
  - Model
- Belief state
  - Definition
  - Computation
- Value Iteration
- Policy
  - Policy Tree
  - Automata
- Reusable trajectories

Belief state

- Where are we now?
  - Probability distribution over states
- Computation:
  - Compute posterior
    - Given action and observation
      - Bayes rule
- Sufficient statistics
  - All the information needed from the history

Belief state: definition

- Belief state:
  - Distribution over states
  - $b \in [0,1]^{|S|}, b^\top 1 = 1$
- Given:
  - belief state b,
  - Action a
  - Observation o

- Next belief state b'
  - $b'(s') = \Pr[s'|o,a,b]$
  - Define:
  - $b' = T(b,a,o)$

- Sufficient Statistics!

Belief state: Computation

- $b'(s') = \Pr[s'|o,a,b]$
- $= \dfrac{\Pr[o|s',a,b]\,\Pr[s'|a,b]}{\Pr[o|a,b]}$
- $= \dfrac{\Pr[o|s',a]\,\sum_{s\in S}\Pr[s'|a,b,s]\,\Pr[s|a,b]}{\Pr[o|a,b]}$
- $= \dfrac{\mathrm{Ob}(o|s',a)\,\sum_{s\in S} p(s'|s,a)b(s)}{\Pr[o|a,b]}$

From POMDP to MDP

- Consider MDP:
  - States: B
    - Belief states
    - Infinite!
  - Actions: A
  - Rewards:
    - $r(b,a) = \sum_s b(s)r(s,a)$
  - Initial state
    - s_0
- Transition:
  - $\Pr[b'|b,a,o]$
    - Deterministic!
  - $\Pr[b'|b,a]$
    - stochastic

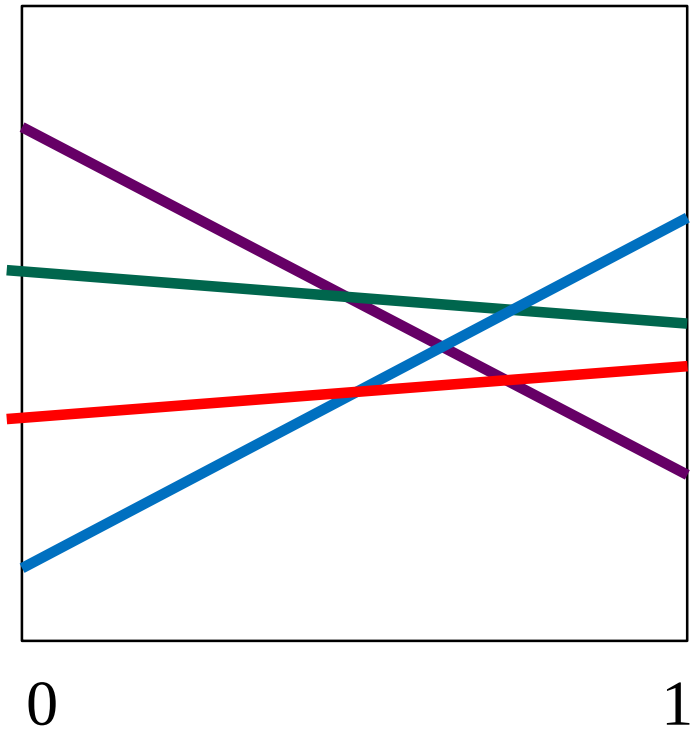
Value Function

- Mapping from belief states to expected return
  - Finite horizon
  - Discounted
  - Average
- Characterization:
  - Value functions will be piece-wise-linear and convex
    - Finite horizon
- Algorithm
  - How do we handle an infinite state space?

Value Iteration: Finite horizon

- Start with $H = 1$
  - Immediate reward
  - $r(b, a) = \sum_s b(s) r(s, a)$
    - Value per action, linear!

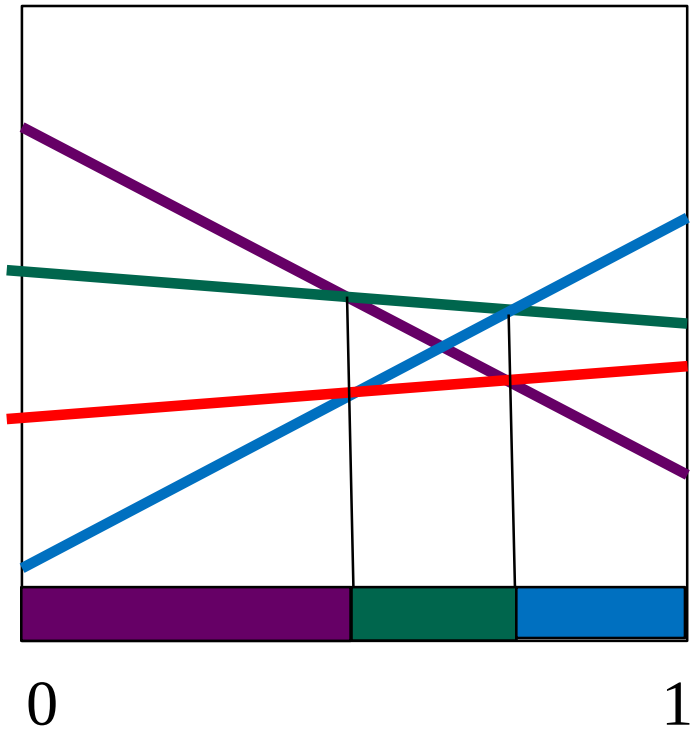
- Two state: $b = (x, 1 - x)$



Value Iteration: Finite horizon

- Start with $H = 1$
  - Immediate reward
  - $r(b, a) = \sum_s b(s) r(s, a)$
    - Value per action, linear!
  - Optimal action
  - $V^*(b) = \max_a r(b, a)$

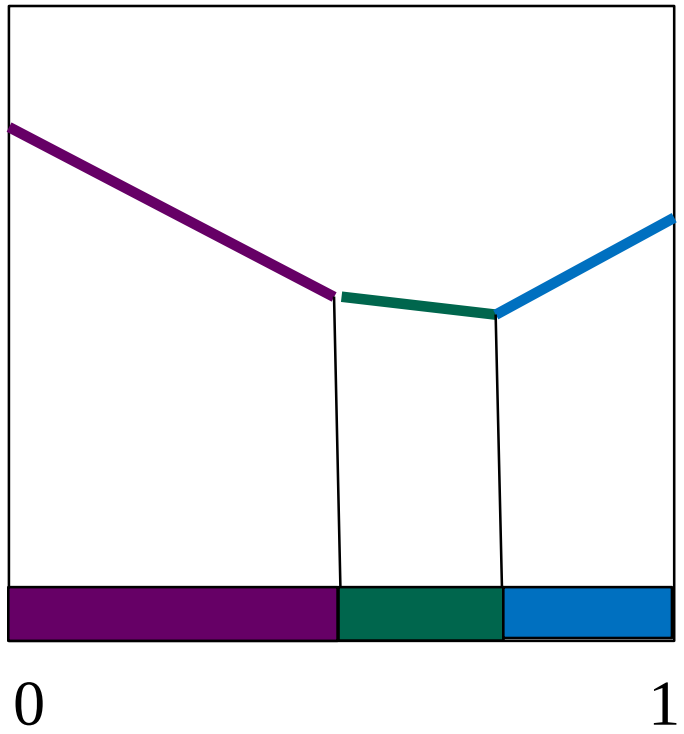
- Two state: $b = (x, 1 - x)$



Value Iteration: Finite horizon

- Start with $H = 1$
  - Immediate reward
  - $r(b,a) = \sum_s b(s) r(s,a)$
    - Value per action, linear!
  - Optimal action
  - $V^*(b) = \max_a r(b,a)$
- Finite representation
  - Intersection of hyperplans

- Two state: $b = (x, 1-x)$

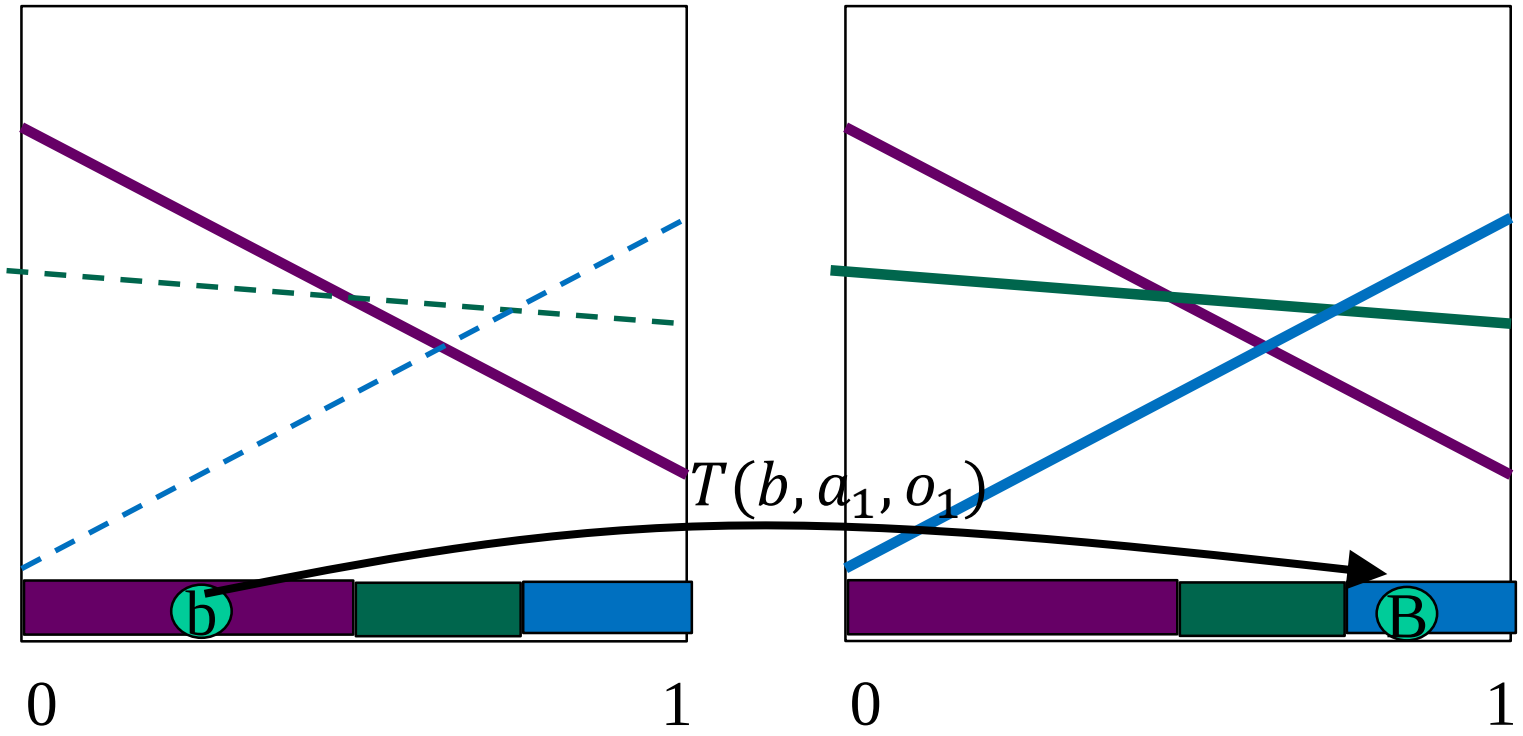


Value Iteration: Finite horizon

- Next $H = 2$
- Three steps …
- Compute value function
  - Given (b, a_1, o_1)
    - Belief state b
    - Action a_1
    - Observation o_1
- Value function
  - Given (b, a_1)
    - Belief state b
    - Action a_1
- Optimal value function
  - Given b
    - Belief state b

Belief state, action, observation

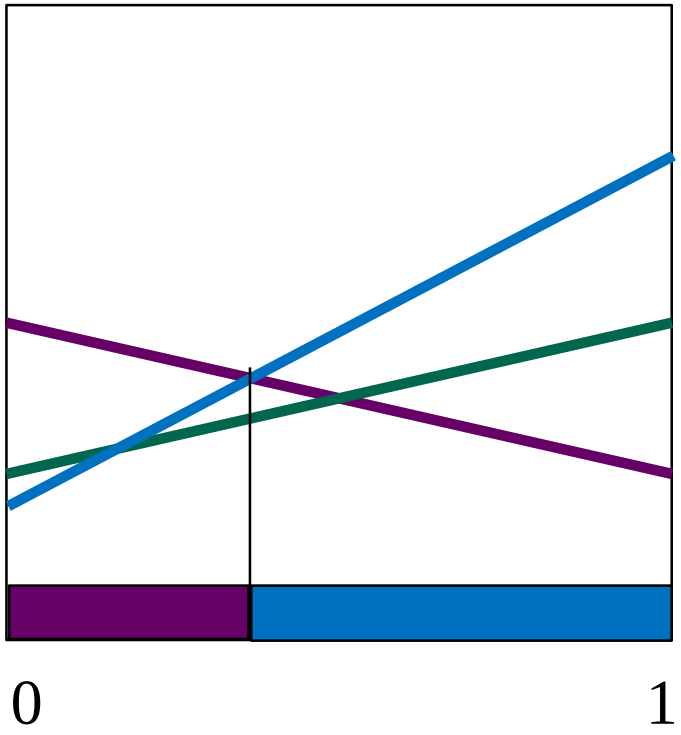
- Given (b, a_1, o_1)
  - Compute $V_2(b|a_1, o_1)$
- Given $\boldsymbol{B} = T(b, a_1, o_1)$
  - $V_2(b|a_1, o_1) = r(b, a_1) + V_1^*(B)$



Belief state, action, observation

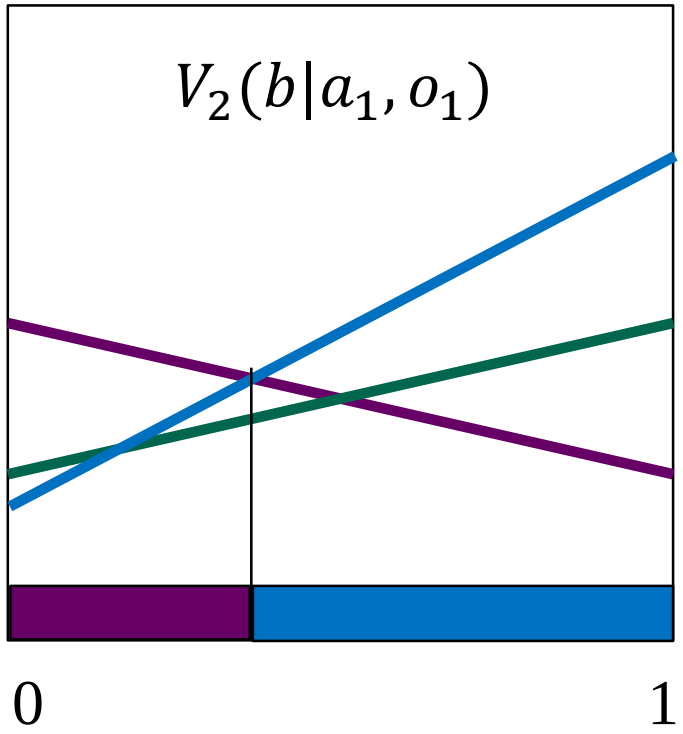
- Given:
  - First action a_1
    - Observation o_1
- Compute $V_2(b|a_1, o_1)$
  - Piece-wise linear

- $V_2(b|a_1, o_1)$

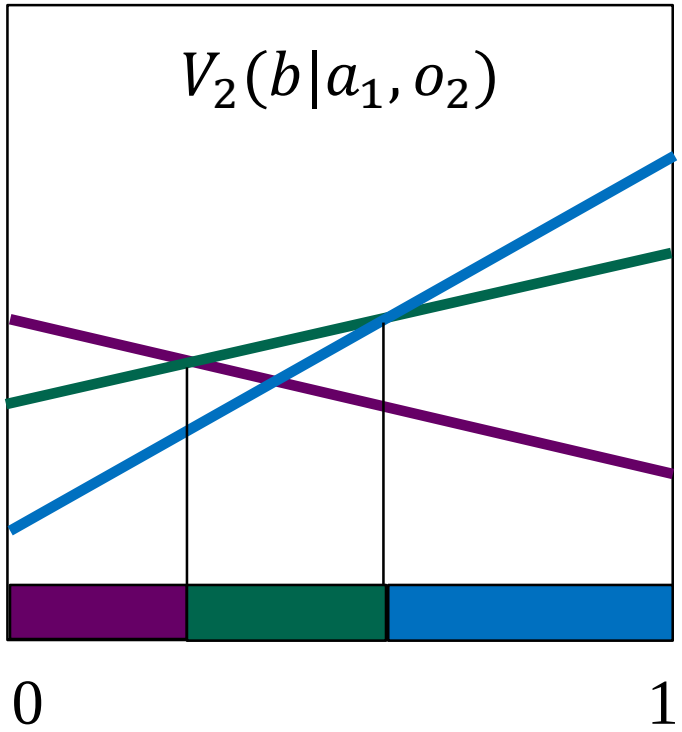


Belief state, action

- Compute $V_2(b|a_1)$
  - Multiple observations



  - For observation o_2

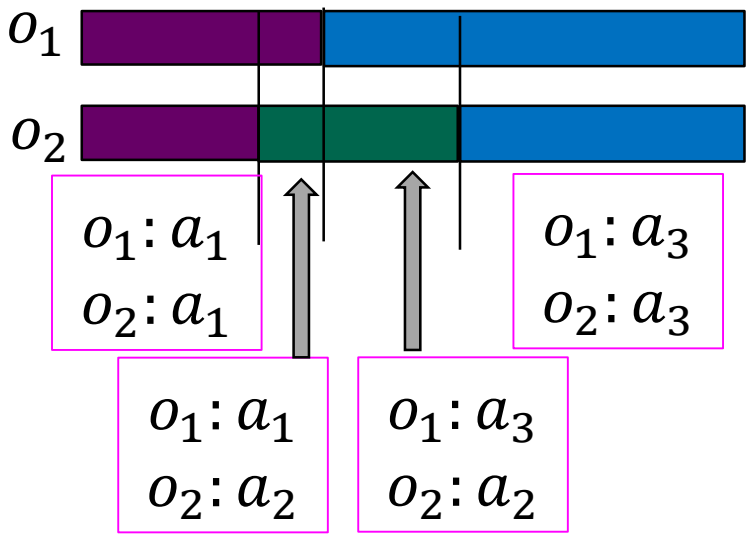


Belief state, action

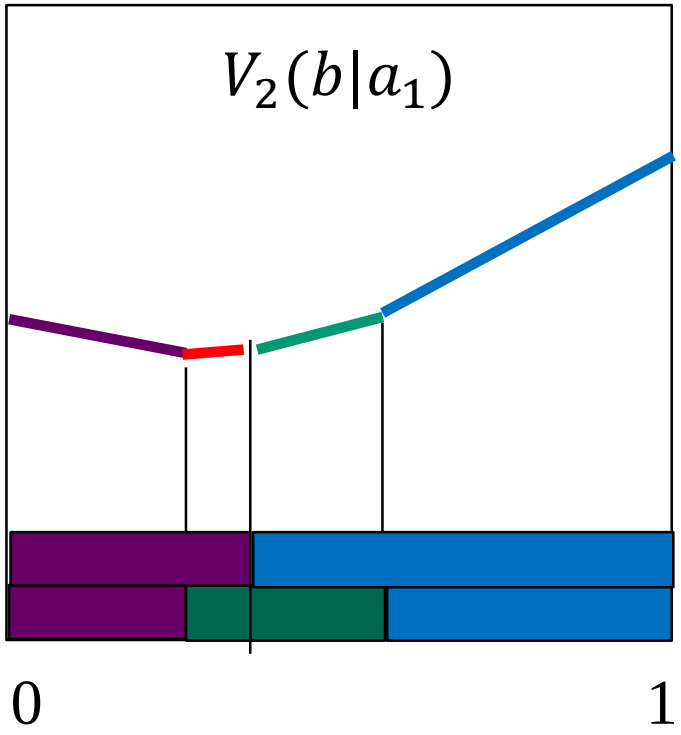
- Given
  - Belief state b and action a_1
- Value function
  - $V_2(b|a_1) = \sum_o \Pr[o|b, a_1]\, V_2(b|a_1, o)$
  - $\Pr[o|b, a_1] = \sum_{s,s'} b(s)p(s'|s, a)Ob(o|s', a)$
    - Linear in b

Belief state, action

- Second action

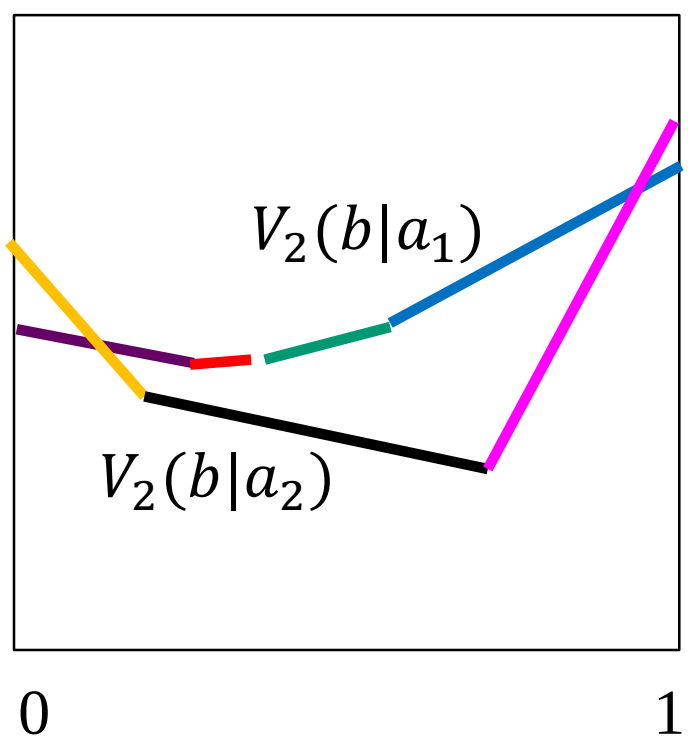


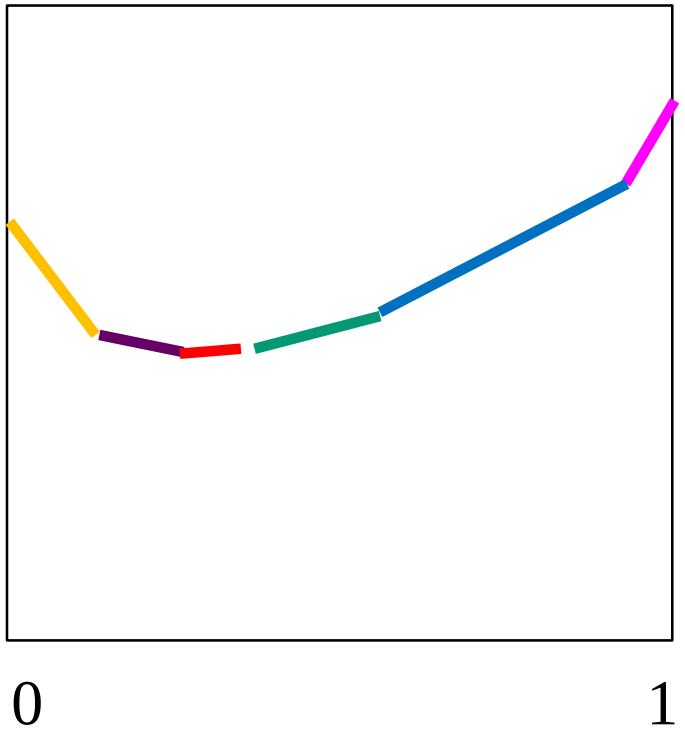
- $V_2(b|a_1)$



Belief state

- Compute $V_2(b|a_i)$
- $V_2(b)$





Longer horizon

- Horizon $H = 3$
- First compute $V_2(b)$
- For each action a_i
  - And observation o_j
  - Compute:
    - $r(b, a_i)$
    - $B = T(b, a_i, o_j)$
  - $V_3\left(b\middle|a_i, o_j\right) = r(b, a_i) + V_2^*(B)$

- Sum over o_j
  - $V_3(b|a_i) =$

    $\sum_j \Pr[o_j|b, a_i] V_3(b|a_i, o_j) =$

    $\sum_{s,s',j} Ob\left(o_j\middle|a_i, s'\right) p(s'|s, a_i)$

    $b(s) \quad V_3(b|a_i, o_j)$

- Over actions:
  - $V_3^*(b) = \max_a V_3\,(b|a)$

Optimal Value function

- Finite Horizon
  - $V_H^*(b) = \max_a [r(b,a) + \sum_o \Pr[o|b,a]\, V_{H-1}^* (T(b,a,o))]$
  - Optimal policy
    - $\pi_H^*(b) = \arg\max_a [r(b,a) + \sum_o \Pr[o|b,a]\, V_{H-1}^* (T(b,a,o))]$
- Discounted
  - $V^*(b) = \max_a [r(b,a) + \gamma \sum_o \Pr[o|b,a]\, V^* (T(b,a,o))]$

Optimal Value function

- Theorem:
  - Finite horizon:
    - V_H^* is convex and piece-wise-linear
    - Exists a collection $\Theta = \{\theta_i \in \mathbb{R}^{|S|}\}$
      - $V_H^*(b) = \max_{\theta_i \in \Theta} b^\top \theta_i$
  - Discounted
    - V^* is convex

Representation of Value function

- $V_H^*(b) = \max_{\theta_i \in \Theta} b^\top \theta_i$
- Geometric view
  - $R(\theta, \Theta) = \{b : b^\top \theta > b^\top \theta' , \theta' \in \Theta - \{\theta\}\}$
  - Convex regions
    - Need to resolve boundary issue
  - The number of regions is finite!

- Proof: By induction on the horizon
  - Base: $H = 1, \quad V_1^*(b) = \max_a \sum_s b(s)\, r(s,a)$
  - Inductive step:
    - Assume that for $i-1$ we have Θ_{i-1} such that
    - $V_{i-1}^*(b_{i-1}) = \max_{\theta \in \Theta_{i-1}} b_{i-1}^\top \theta$
  - For belief state b_i, action a, observation o:
    - $b_{i-1}(s'|b_i, a, o) = \Pr[s'|b_i, a, o]$
  - For $V_i^*(b_i)$:
    - $= \max_a [r(b_i, a) + \sum_o \Pr[o|b_i, a]\, V_{t-1}^*\,(T(b_i, a, o))]$
    - $= \max_a \left[r(b_i, a) + \sum_o \Pr[o|b_i, a] \max_{\theta_{j-1} \in \Theta_{i-1}} b_{i-1}^\top \theta_{j-1} \right]$

- $\sum_o \Pr[o|b_i, a] \max_{\theta_{j-1} \in \Theta_{i-1}} b_{i-1}^{\top} \theta_{j-1} = \sum_o \max_{\theta_{j-1} \in \Theta_{i-1}} \Pr[o|b_i, a]\, b_{i-1}^{\top} \theta_{j-1}$
- $= \sum_o \max_{\theta_{j-1} \in \Theta_{i-1}} \Pr[o|b_i, a] \sum_{s'} \Pr[s'|b_i, a, o]\, \theta_{j-1}(s')$
- $= \sum_o \max_{\theta_{j-1} \in \Theta_{i-1}} \sum_{s'} \Pr[s', o|b_i, a]\, \theta_{j-1}(s')$
- $= \sum_o \max_{\theta_{j-1} \in \Theta_{i-1}} \sum_s b_i(s) \sum_{s'} \Pr[s', o|s, a]\, \theta_{j-1}\ (s')$

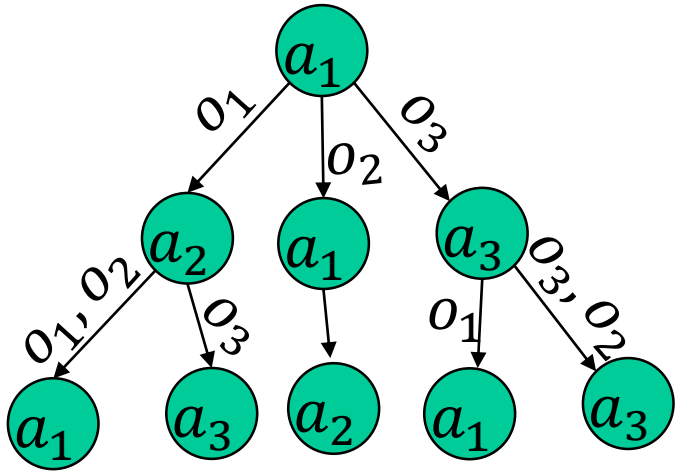
- $V_i^*(b_i) = \max_a [r(b_i, a) + \sum_o \Pr[o|b_i, a]\, V_{t-1}^*\,(T(b_i, a, o))]$
  - $= \max_a \max_{\theta_{j-1}^{a,o} \in \Theta_{i-1}^a} \left[r(b_i, a) + \sum_s b_i(s) \sum_{s',o} \Pr[s', o|s, a]\, \theta_{j-1}^{a,o}\ (s')\right]$
  - $= \max_a \max_{\theta_{j-1}^{a,o} \in \Theta_{i-1}^a} \left[\sum_s b_i(s)(R(s, a) + \sum_{s',o} \Pr[s', o|s, a]\, \theta_{j-1}^{a,o}\ (s'))\right]$

Lecture 11: outline

- POMDP
  - Example
  - Model
- Belief state
  - Definition
  - Computation
- Value Iteration
- Policy
  - Policy Tree
  - Automata
- Reusable trajectories

Policy Tree

- How does the policy look like?
- We can use a decision tree.
  - Nodes labeled by action
    - Implicitly: belief state
  - Edges by observations



Value Iteration

- Update:
  - $V_{t+1}(b) = \max_a \{r(b,a) + \sum_o Ob(o|b,a)\, V_t(b')\}$
    - $b'(s') = \Pr[s'|b,a,o]$
  - $V_{t+1}(b) = \max_a V_t^a(b)$
  - $V_t^a(b) = \sum_o V_t^{a,o}(b)$
  - $V_t^{a,o}(b) = \frac{r(b,a)}{|O|} + \Pr[o|b,a]\, V_t(b')$

Value Iteration

- Assume: $\exists \Theta_t$
  - $V_t(b) = \max_{\theta \in \Theta_t} b^\top \theta$
- For $V_t^{a,o}$:
  - b' linear in b
    - $b' = Mb$
  - $V_t^{a,o}(b') = \max_{\theta \in \Theta_t} \theta^\top b'$
  - $= \max_{\theta \in \Theta_t} b^\top M \theta$
  - $\Theta_t^{a,o} = M\theta$
- For V_t^a:
  - $V_t^a(b) = \sum_o V_t^{a,o}(b)$
  - $= \sum_o \max_{\theta \in \Theta_t^{a,o}} b^\top \theta$
  - $= \max_{\theta \in \Theta_t^a} \sum_o b^\top \theta$
  - $\Theta_t^a = \sum_o \Theta_t^{a,o}$
- For V_{t+1}
  - $V_{t+1}(b) = \max_a V_t^a(b)$
  - $= \max_{\theta \in \Theta_{t+1}} b^\top \theta$
    - $\Theta_{t+1} = \cup_a \Theta_t^a$

Value Iteration: complexity

- Depends on Θ_t
- For $\Theta_t^{a,o}$
  - $\left|\Theta_t^{a,o}\right| \leq |\Theta_t|$
- For Θ_t^a
  - $|\Theta_t^a| \leq \left|\Theta_t^{a,o}\right|^{|O|} \leq |\Theta_t|^{|O|}$

- For Θ_{t+1}
  - $|\Theta_{t+1}| \leq \sum_a |\Theta_t^a|$
  - $\leq |A| \cdot |\Theta_t|^{|O|}$

- Complexity
  - Exponential in $|O|$
    - Each iteration
  - Pruning can help
  - Exponential time unavoidable

Hardness

- Computing the optimal policy for
  - Finite horizon
    - Unique start state
  - P-SPACE complete
- Computing the optimal policy for
  - Finite horizon
    - Unobservable
  - NP-complete

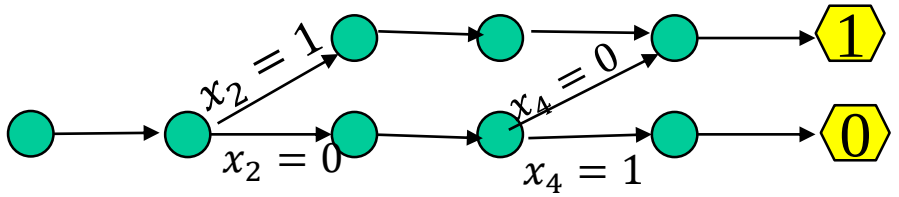
Unobservable MDP

- No observations
- Need to plan a sequence of actions
  - No feedback …
- Optimize finite horizon

- We will show:
  - Simple reduction from SAT

Reduction from SAT

- For each clause:
  - Create a sequence of *n* steps
  - Step *i* action is the value of x_i
  - If clause satisfiable, reward 1 (else 0)
- Example
  - Clause: $x_2 \vee \bar{x}_4$

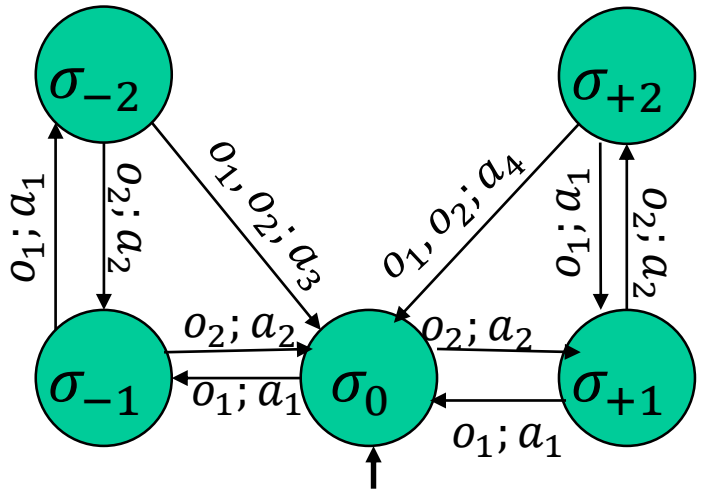


Reduction from SAT

- Initial state s_0
  - Selects a random Clause C_i, for $1 \leq i \leq m$
- If formula satisfiable
  - $\exists x_1, \ldots, x_n$ get always reward 1
- If formula not satisfiable
  - $\forall x_1, \ldots, x_n$ expected rewards $\leq 1 - \frac{1}{m}$
- Hardness of approximation

Policy Plan: Automata

- A simple policy class
  - Finite Automata
    - with output
  - Cannot represent any strategy!
- Moore Automata
  - Inputs:
    - Observations
  - Outputs:
    - Actions



Policy Plan: Automata

- We can compute V^{π}
  - When π is an automata
  - $V^{\pi}(s;\sigma) = r(s,a) + \sum_{o',s'} \gamma\, p(s'|s,a) Ob(o'|s',a) V^{\pi}(s';\sigma')$
    - $a = \pi(\sigma, o),\ \sigma' = \delta(\sigma, o')$
  - Set of linear equations in $V^{\pi}(s;\sigma)$
    - Can be computed easily

Lecture 11: outline

- POMDP
  - Example
  - Model
- Belief state
  - Definition
  - Computation
- Value Iteration
- Policy
  - Policy Tree
  - Automata
- Reusable trajectories

POMDP: reusable trajectories

- Goal:
  - For policies $\pi \in \Pi$
    - Estimate $V^{\pi}(s_0)$
      - expected return
  - A trajectory set Γ
    - Small
- Random trajectory:
  - Follow random policy
    - $\pi_R(a|b) = \frac{1}{|A|}$
  - Generate m histrories

- PAC criteria:
  - For $\pi \in \Pi$
    - Estimate $\hat{V}^{\pi}(s_0)$
  - With probability $1 - \delta$
  - For every $\pi \in \Pi$
    - $\left|\hat{V}^{\pi}(s_0) - \mathrm{V}^{\pi}(s_0)\right| \leq \epsilon$
- How large is m?
  - $\epsilon, \delta, \Pi, V_{max}$
    - Finite horizon/discounted

POMDP: reusable trajectories

- Define
  - For history h and $\pi \in \Pi$
  - $acc_\pi(h) = 1$ iff
    - $\forall t: a_t = \pi(h_t)$
    - h_t prefix up to time t
      - π deterministic
  - For $\pi \in \Pi$
    - $\Gamma_\pi = \{h \in \Gamma: acc_\pi(h) = 1\}$

- Estimating return
  - $\hat{V}^\pi(s_0) = \frac{1}{|\Gamma_\pi|} \sum_{h \in \Gamma_\pi} retrun(h)$

- Unbiased:
  - $E[\hat{V}^\pi(s_0)] = V^\pi(s_0)$

Reusable trajectories: Analysis

- Lemma:
  - For a finite horizon H
  - For any strategy π
  - $\Pr_{h}[acc_{\pi}(h) = 1] = \frac{1}{|A|^{H}}$
    - $h \sim \pi_R$
- Lemma:
  - $D_{\pi}(h) = D_{\pi_R}(h | acc_{\pi}(h) = 1)$

Reusable trajectories: Analysis

- Lemma
  - For $m > 8|A|^H \frac{V_{max}^2}{\epsilon^2} \log(2|\Pi|/\delta)$
  - With probability $1 - \delta/2$
    - $|\Gamma_\pi| \geq \frac{m}{2|A|^H} > 4\frac{V_{max}^2}{\epsilon^2} \log(2|\Pi|/\delta)$

Reusable trajectories: Analysis

- Lemma
  - If for every $\pi \in \Pi$ we have
    - $|\Gamma_\pi| \geq \frac{m}{2|A|^H} > 4\frac{V_{max}^2}{\epsilon^2}\log(2|\Pi|/\delta)$
  - Then with probability $1 - \delta/2$
    - $\left|\hat{V}^\pi(s_0) - V^\pi(s_0)\right| \leq \epsilon$
      - For every $\pi \in \Pi$

Reusable trajectories: Analysis

- Theorem:
  - For $m > 8|A|^H \frac{V_{max}^2}{\epsilon^2} \log(2|\Pi|/\delta)$
  - With probability $1 - \delta$, for every $\pi \in \Pi$,
$$\left|\hat{V}^{\pi}(s_0) - V^{\pi}(s_0)\right| \leq \epsilon$$