Partially Observable MDPs (POMDPs)

CSCI 2951-F

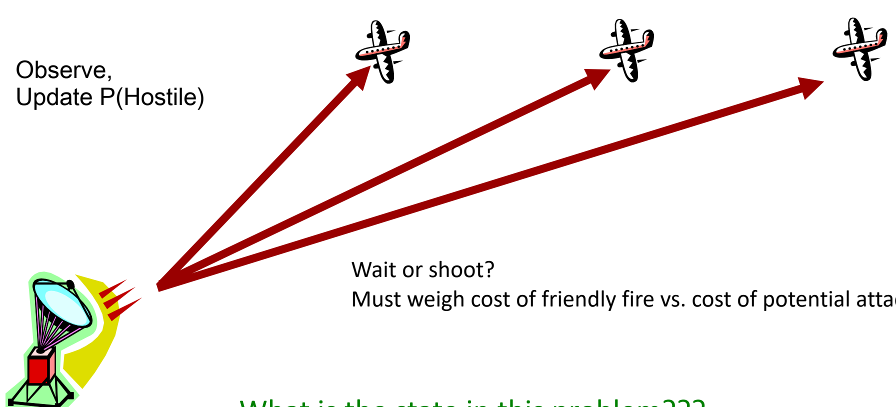
Ron Parr

With thanks to Christopher Painter-Wakefield

Example POMDP

Unidentified incoming target:

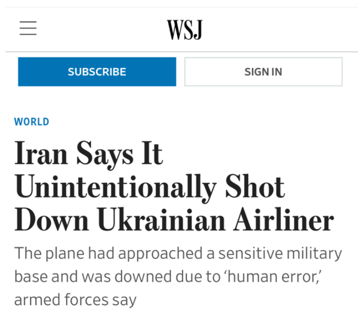
Observe,
Update P(Hostile)



Wait or shoot?
Must weigh cost of friendly fire vs. cost of potential attack

What is the state in this problem???

This Is A Real Problem!



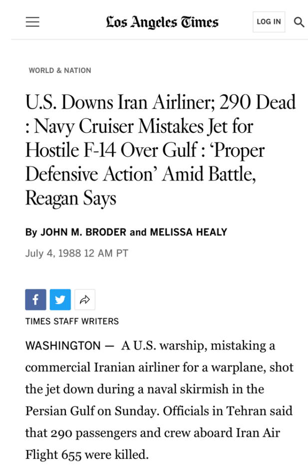
WSJ

SUBSCRIBE SIGN IN

WORLD

Iran Says It Unintentionally Shot Down Ukrainian Airliner

The plane had approached a sensitive military base and was downed due to 'human error,' armed forces say



Los Angeles Times LOG IN

WORLD & NATION

U.S. Downs Iran Airliner; 290 Dead : Navy Cruiser Mistakes Jet for Hostile F-14 Over Gulf : 'Proper Defensive Action' Amid Battle, Reagan Says

By JOHN M. BRODER and MELISSA HEALY

July 4, 1988 12 AM PT

TIMES STAFF WRITERS

WASHINGTON — A U.S. warship, mistaking a commercial Iranian airliner for a warplane, shot the jet down during a naval skirmish in the Persian Gulf on Sunday. Officials in Tehran said that 290 passengers and crew aboard Iran Air Flight 655 were killed.

Many other tragic examples since 1940s

Other Example POMDPs

- Patient diagnosis/treatment (patient state?)
- Machine maintenance (machine state?)
- Robotic search problems, e.g., de-mining (object my sensors detected?)
- Robot navigation (robot's true location?)
- Assistive technologies (user's intent/needs?)

Straw Man

- What if we treat the observation as the state?
- Violates Markov assumption
- Can't distinguish between two states that coincidentally produce similar observations
- Leads to suboptimal policies and/or can cause oscillation in many algorithms
  (though not pure policy gradient)

Partially Observable MDP (POMDP)

- State space: $s \in S$
- Action space: $a \in A$
- Observation space: $z \in Z$
- Reward model: $R(s,a,s')$
- Transition model: $P(s'|s,a)$
- Observation model: $P(z|s',a)$
- Discount: $\gamma \in [0,1]$

- MDP dynamics (transitions, rewards) are unchanged
- After a state transition, agent observes z w.p. $P(z|s',a)$
- Underlying Markovian process **BUT** state is hidden; agent only sees observation
- Like HMMs with actions and reward

Belief States

True state is only *partially* observable

- b = belief state
- b[s] = probability of state s
- At each step, the agent
  - takes some action a
  - transitions to some state s' with probability p(s'|s,a)
  - makes observation z with probability p(z|s',a)
- Posterior belief given z, a, b:

Same as HMM tracking/monitoring equations

$$b'(s') = \alpha\, p(z \mid s',a) \sum_{s} p(s' \mid s,a)\, b(s)$$

Understanding Belief States

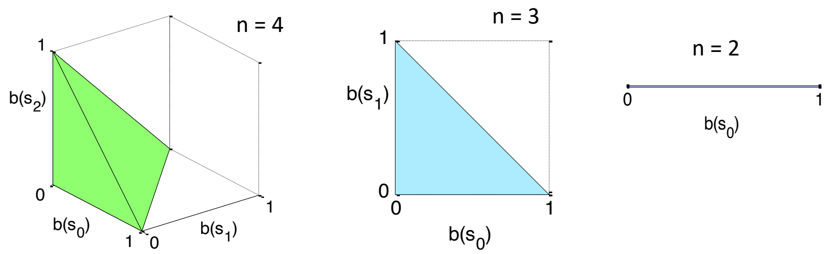
- A problem with n underlying states (discrete state space of size n) has:
  - A continuous belief space
  - Each element of the belief space is a distribution over the n underlying states
  - Belief states that are vectors of length n
- Partial observability turns discrete problems into continuous problems
- A POMDP with n states induces an n-dimensional ***belief MDP***

Belief Space

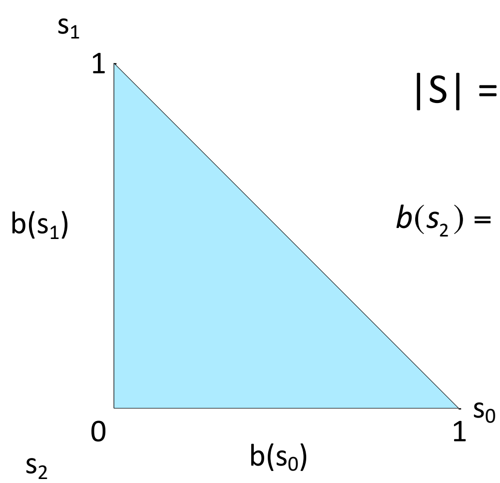
- Since belief is a probability distribution:

$$\sum_s b[s] = 1$$

  - For n states, belief has n-1 degrees of freedom
  - Beliefs live in a n-1 dimensional *simplex*



Belief Space Illustrated



|S| = 3

$$b(s_2) = 1 - b(s_1) - b(s_0)$$

POMDP Value Functions

- Bellman equation for POMDPs:

$$V^*(b) = \max_a \left[\underbrace{\rho(b,a)}_{} + \gamma \int_{b'} \underbrace{p(b' \mid a,b) V^*(b')}_{} db' \right]$$

Expectation of R given b, a:

$$= \sum_s R(s,a)b(s)$$

Need to compute a probability for an infinite number of belief states ☹

- How do we compute this integral? We don't!

POMDP Value Functions

- Bellman equation for POMDPs:

$$V^*(b) = \max_a \left[\underbrace{\rho(b,a)}_{} + \gamma \sum_{b'} \underbrace{P(b' \mid a,b) V^*(b')}_{} \right]$$

Expectation of R given b, a:

$$= \sum_s R(s,a)b(s)$$

Belief transition probability derived from POMDP transition/observation models:

$$= \sum_{z:b_a^z=b'} \sum_{s'} P(z \mid s',a) \sum_s P(s' \mid s,a)$$

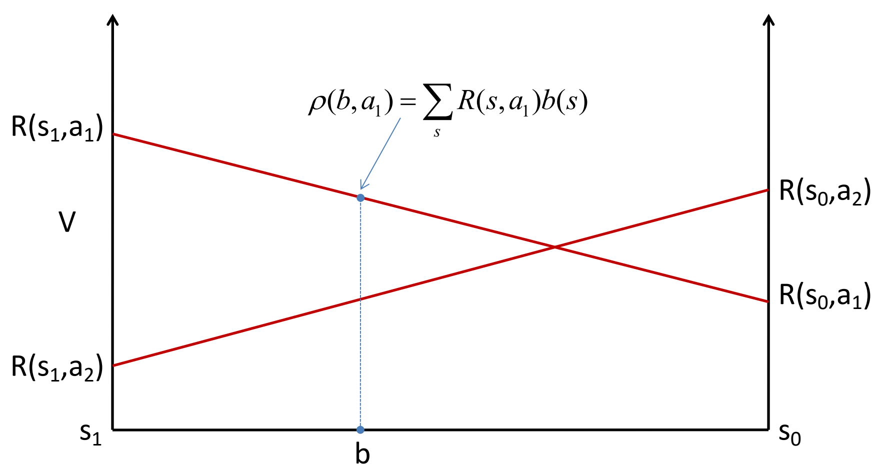
- Why sum and not integral?

Representing V

- Good news: Computing RHS of the Bellman equation for a particular V(b) takes a reasonable amount of time given some method of querying V(b')

- Bad news: V is still defined over a continuous domain – how do we represent V tractably?

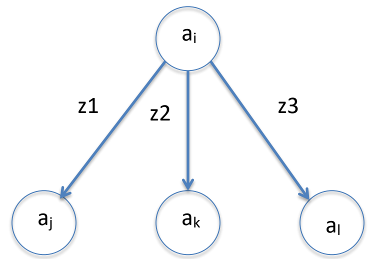
1-Step POMDP Value Function



With one step to go, we can take an action and receive a reward.

2-Step POMDP Policy

- How many 2-step policies are there?
- Exponential in |Z|

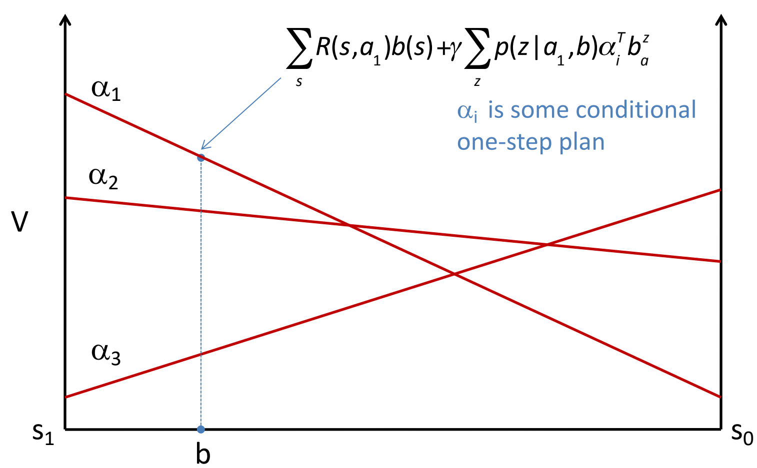


What is the value of the root node, as a function of the (unknown) starting state?

It is the immediate reward + expected discounted value of next action

Call $\alpha[s]$ the value of being in node s, starting at the root
$\alpha^T b$ = value of a belief state b under this policy

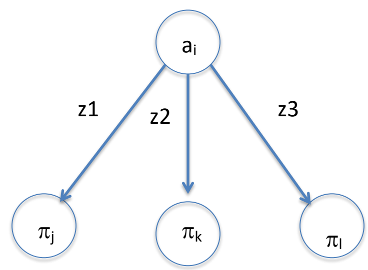
POMDP Value Function



With two steps to go, we can take an action and make an observation, then take another action.

Multistep POMDP Value Functions

- Build (i+1)-step policies by considering all ways of adding on to i-step policies



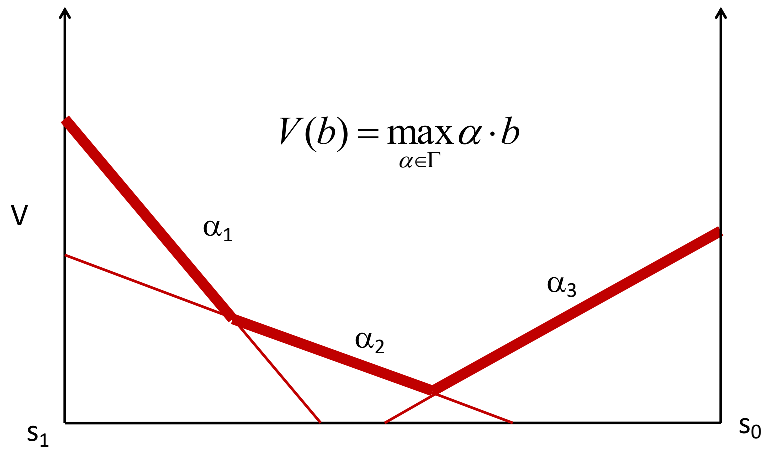
- How many (i+1)-step policies are there?
- n i-step policies, a actions z obsersvations -> an^z

POMDP Value Functions

- Any finite horizon conditional plan has value that is linear in the belief state
- α[s]=value of starting plan in state s
- $\Gamma=[\alpha_0\ldots\alpha_m]$: Set of vectors corresponding to values of conditional plans
- Value of following plan i from belief state b:

$$\sum_s \alpha_i[s]b[s] = \alpha_i \cdot b$$

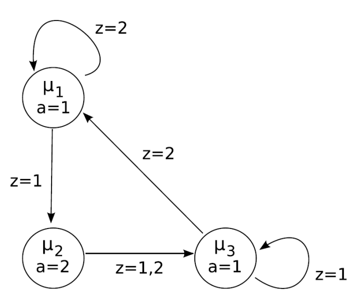
POMDP Value Functions



Finite horizon POMDP value function is piecewise linear and convex (assume we follow best plan for each belief state)

Infinite Horizon Policies

- Conditional policies represented as finite state machines
  - States $\mu_1 \ldots \mu_m$ labeled with actions
  - Deterministic transition function $\delta(\mu,z)$
  - Belief state not used in following policy



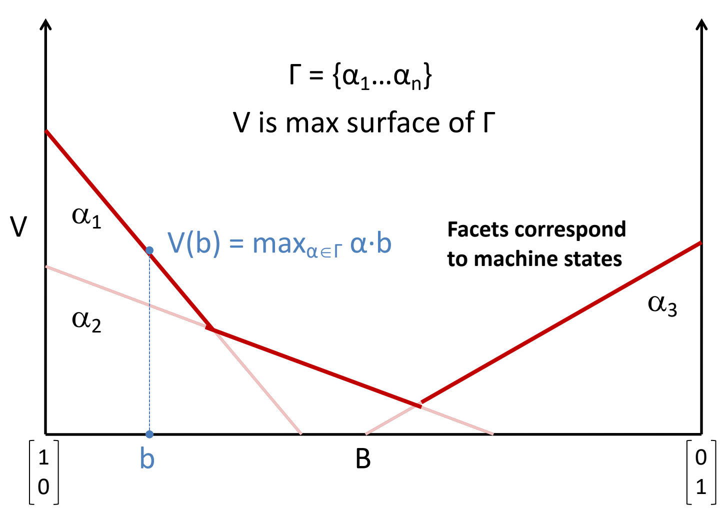
FSM Policy Evaluation

- Policy x POMDP induces a Markov chain
  - States: $\sigma_{\mu,s}$ $\quad$ ($\forall$ s $\in$ S, $\mu \in$ FSM)
  - Reward function: $\rho_{\mu,s} = R(s,a_\mu)$
  - Transition function:

$$\underbrace{\tau(\sigma_{\mu,s}, \sigma_{\mu',s'})}_{Pr(\mu',s' \mid \mu,s)} = \underbrace{P(s' \mid s,a_\mu)}_{Pr(s' \mid \mu,s)} \underbrace{\sum_{\{z:\delta(\mu,z)=\mu'\}} P(z \mid s',a_\mu)}_{Pr(\mu' \mid s',\mu,s)}$$

  - Discount factor: γ
- POMDP value function can be extracted from Markov chain value function

POMDP FSM Value Functions



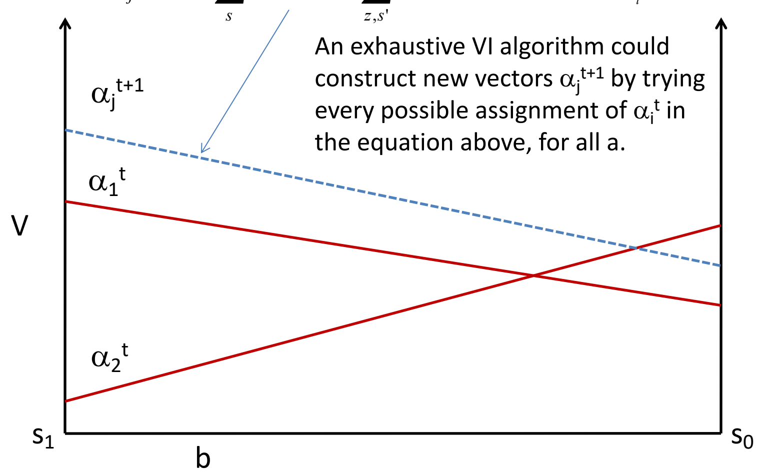
Solving POMDPs by Value Iteration

- Basic outline of an exact VI algorithm
  - Given $V_i=\Gamma_i$
  - Generate Γ_{i+1} as one step extensions from Γ_i
  - Note: ($|A||\Gamma_i|^{|Z|}$ extensions!)
  - Prune vectors in Γ_{i+1} which are not maximal for any b
  - $V_{i+1}=\Gamma_{i+1}$
- Challenges:
  - Potentially large number of new vectors
  - Exponential growth with number of iterations

POMDP Value Iteration

$$\alpha_j^{t+1}(s)=\sum_s R(s,a)+\gamma\sum_{z,s'} p(z\,|\,s',a)\,p(s'\,|\,s,a)\,\alpha_i^t(s')$$

An exhaustive VI algorithm could construct new vectors α_j^{t+1} by trying every possible assignment of α_i^t in the equation above, for all a.



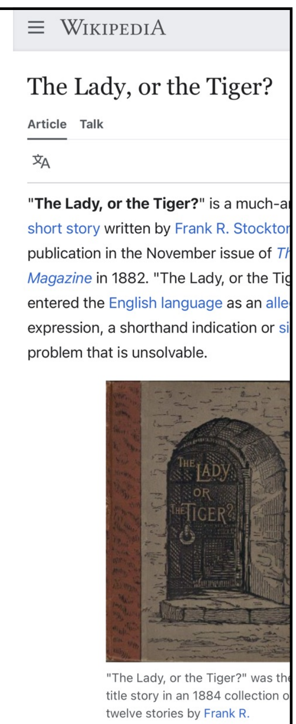
Policy Iteration for POMDPs

- Basic idea of MDP policy iteration carries over to POMDPs
- Policies = FSMs
- Implementation is slightly tricky
- Highlights:
  - Evaluate FSM (generate alpha vectors)
  - Do one step of value iteration (policy evaluation)
  - Modify FSM based on value iteration results (policy improvement)
  - Alternate between policy evaluation, policy improvement
- Good news: Can be more efficient than VI
- Bad news: FSM complexity can grow exponentially

Example: Tiger Problem

- Tiger behind one door, prize behind another
- Agent doesn't know which is which (2 states)
- Listening gives a noisy indicator
- Intuitive solution: Listen until you are confident, then open the door

- What does the value function for this problem look like? (discussion)



POMDP Computational Complexity

- Size of value function can grow exponentially with number of iterations of value iteration
- Pruning can help, but no guarantees
- In practice, exact value iteration algorithms are practical for POMDPs with **ones** of states

- Doesn't necessarily imply problem is intractable, but…
- POMDPs are, in fact, PSPACE hard ☹

POMDP Conclusions

- Generalize MDPs
- Like HMMs, track distribution over underlying states
- Every POMDP is a continuous state MDP, where MDP states correspond to POMDP belief states
- Tricky and computationally expensive in practice