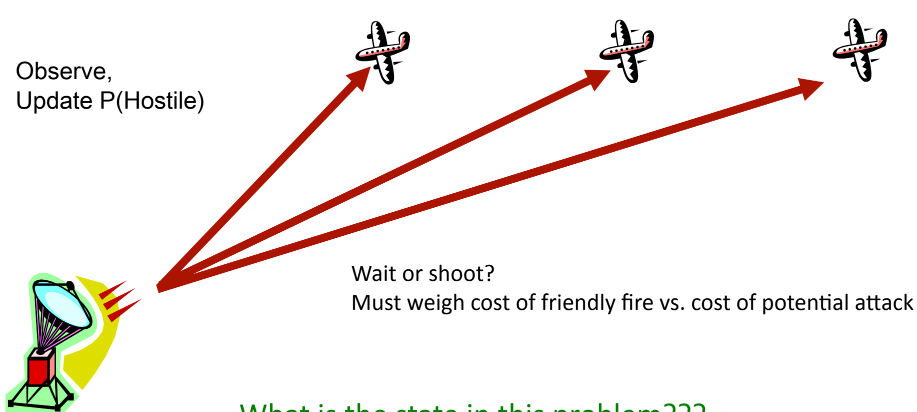
Partially Observable MDPs (POMDPs)

CPS 170
Ron Parr
With thanks to Christopher Painter-Wakefield

Example POMDP

Unidentified incoming target:



What is the state in this problem???

Other Example POMPs

- Patient diagnosis/treatment
- Machine maintenance
- Robotic search problems (e.g., de-mining)

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 (no way to improve your estimate of what's going on over time)
- Leads to suboptimal policies

Partially Observable MDP (POMDP)

- State space: $s \in S$
- Action space: $a \in A$
- Observation space: $z \in Z$
- Reward model: $R(s,a)$
- 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 with probability $P(z|s',a)$.
- State is hidden; agent only sees observation.

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:

$$b'(s') = \alpha\, p(z \mid s', a) \sum_s p(s' \mid s, a)\, b(s)$$

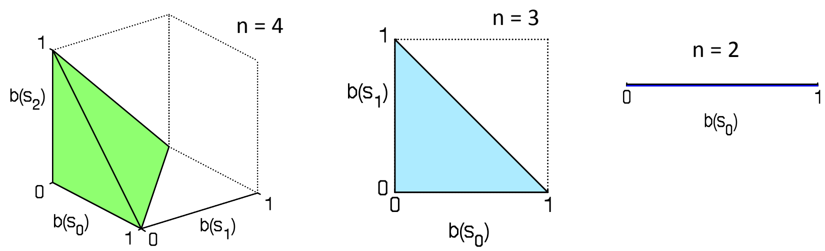
Compare with HMMs!

Belief Space

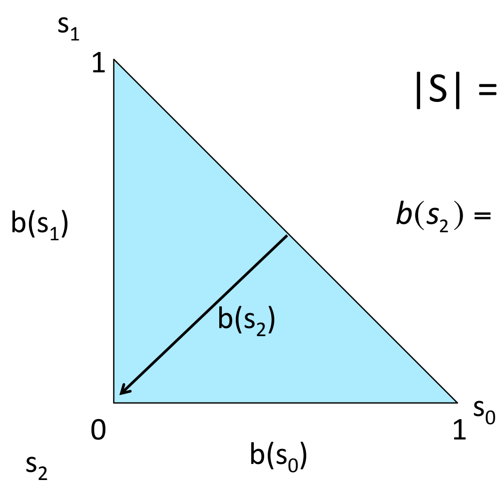
- Since belief is a probability distribution:

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

  - 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:

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

Expectation of R given b, a:

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

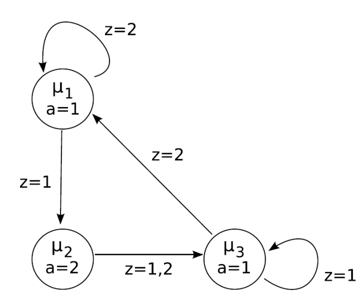
Belief transition probability derived from POMDP transition/observation models:

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

- Why sum and not integral?

Finite State Machine Policies

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



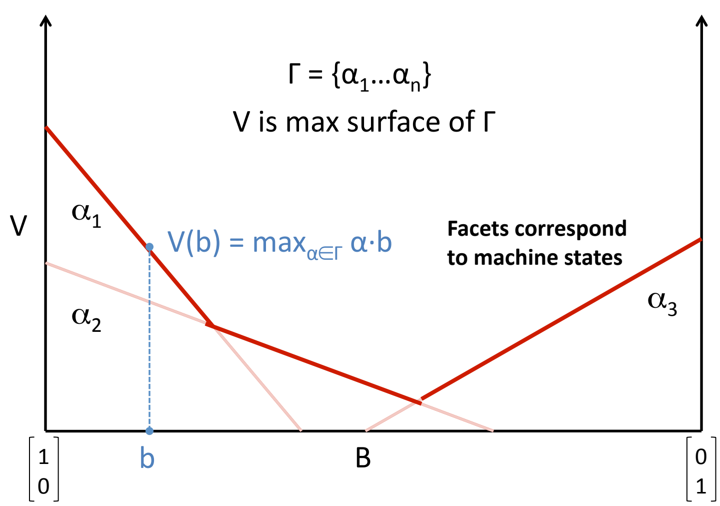
POMDP Policy Evaluation

- Policy x POMDP induces a Markov chain
  - States: $\sigma_{\mu,s}$ $(\forall\, s \in S,\ \mu \in FSM)$
  - 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 Value Functions



Policy Iteration for POMDPs
(one of several possible methods)

- Basic idea of MDP policy iteration carries over to POMDPs
- Implementation is tricky
- Highlights:
  - Set of rules for adding new machine states to finite state controller, such that new controller is guaranteed to improve on old one
  - Alternate between policy evaluation phases and policy improvement phases
- Good news: Turns a nasty, continuous problem into a somewhat manageable discrete one
- Bad news: May add $O(m^{\#Z})$ new FSC states per iteration
  (m = current number of states, #Z = number of possible observations)
- In practice, it is possible to find optimal solutions only for fairly small POMDPs (high 10's to low 100's of states)

POMDP Conclusions

- Generalize MDPs to include imperfect information about the state
- Like HMMs in that we track a distribution over underlying states
- Every POMDP is a continuous state MDP, where MDP states correspond to POMDP belief states
- POMDPs are quite tricky and computationally expensive to solve in practice