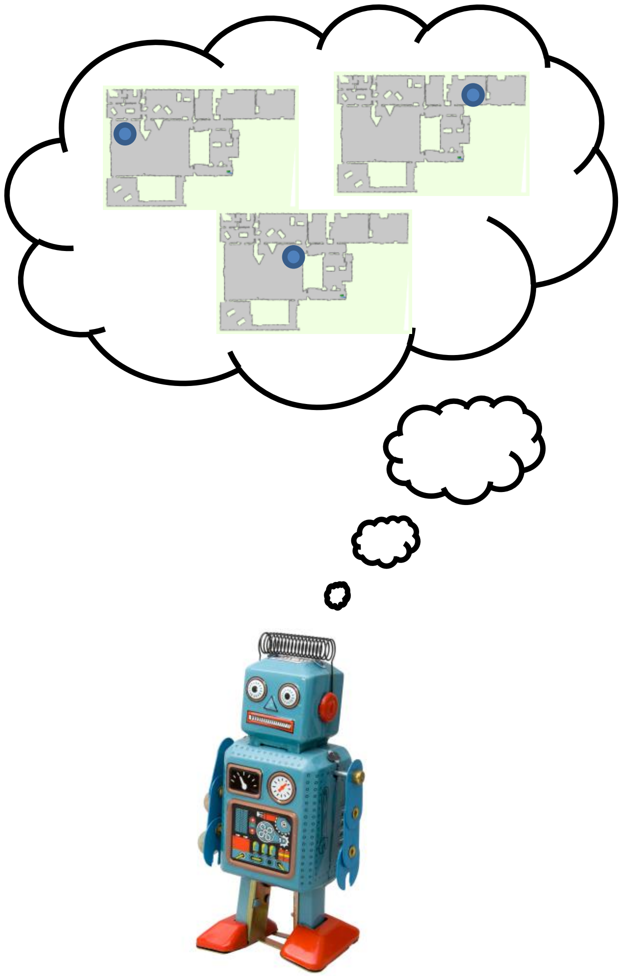
Partially Observable Markov Decision Process

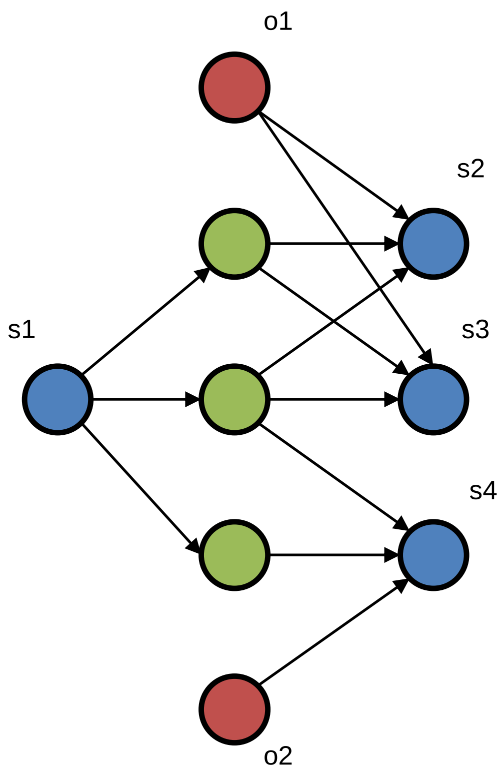
based on lecture slides of Branislav Bošanský and a POMDP tutorial of H. Huang

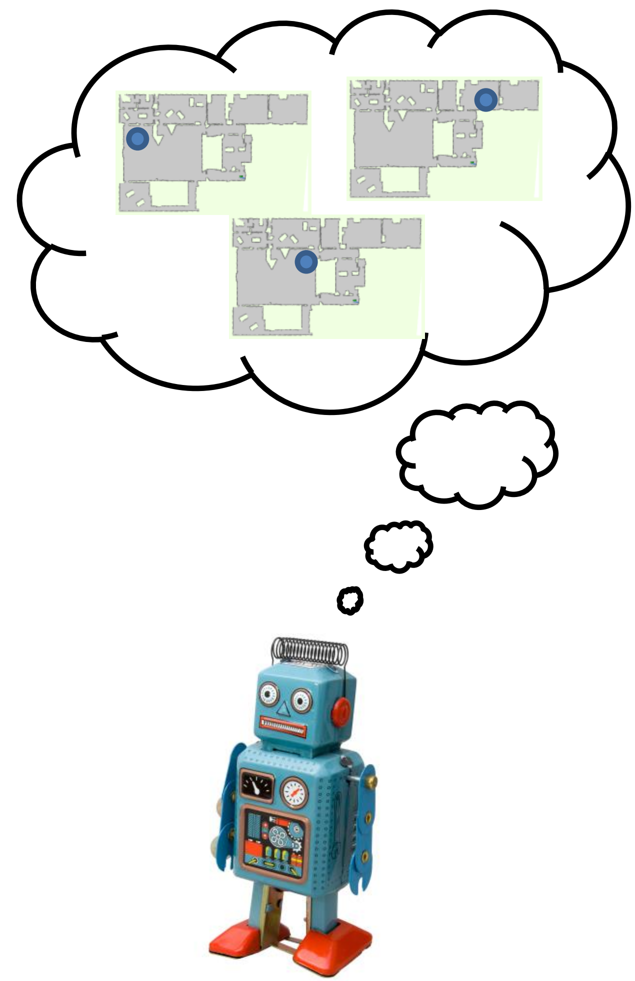
Partial Observability

- the world is not perfect
  - actions take some time to execute
  - actions may fail or yield unexpected results
  - the environment may change due to other agents
  - the agent does not have knowledge about whole situation
  - sensors are not precise



Partial Observability

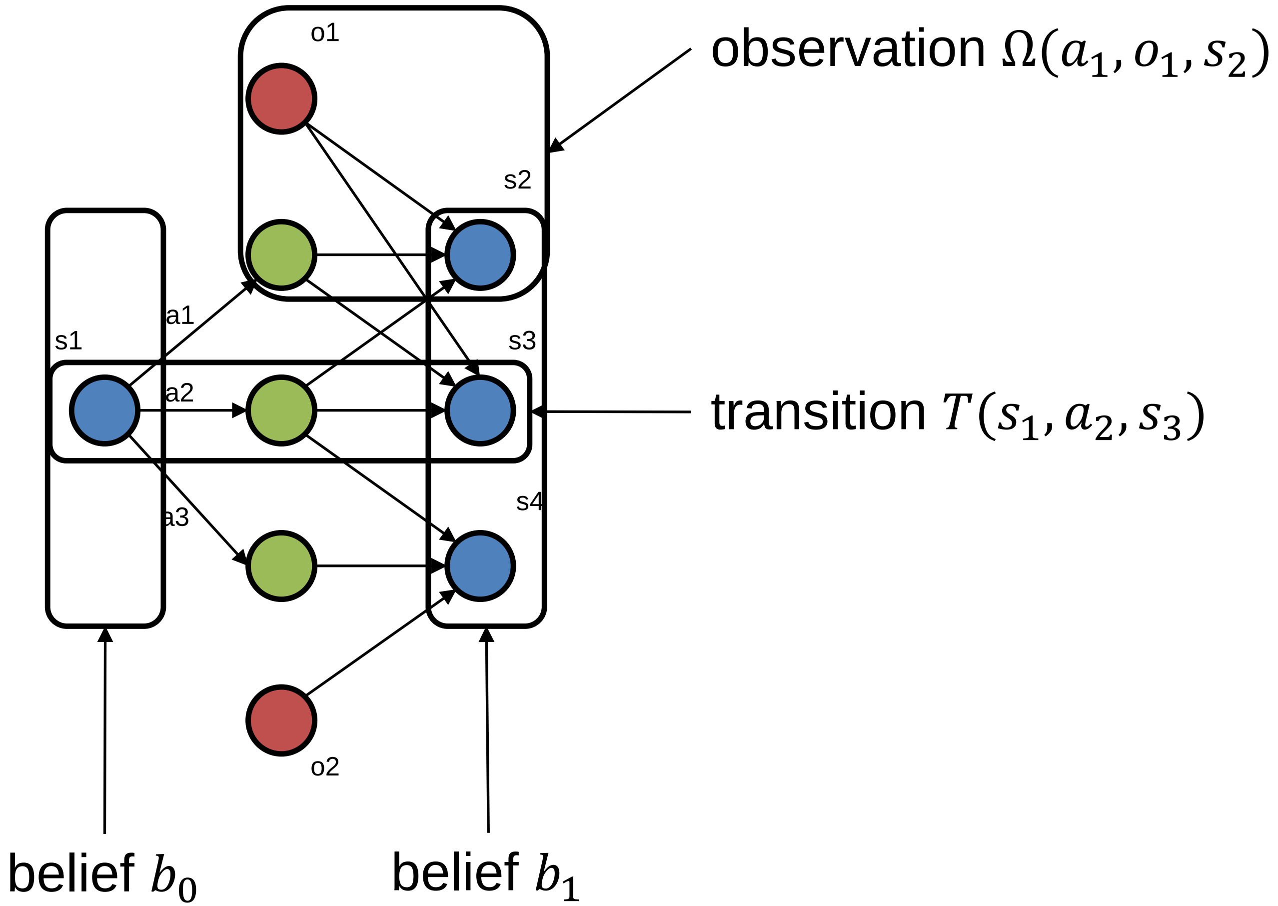




Partially Observable MDPs

- main formal model for scenarios with uncertain observations
- $\langle S, A, D, O, b_0, T, \Omega, R, \gamma \rangle$
  - states – finite set of states of the world
  - actions – finite set of actions the agent can perform
  - time steps
  - observations – finite set of possible observations
  - initial belief function $b_0: S \rightarrow [0,1]$
  - transition function $T: S \times A \times S \rightarrow [0,1]$
  - observation probability $\Omega: A \times O \times S \rightarrow [0,1]$
  - reward function $R: S \times A \rightarrow \mathbb{R}$
  - discount factor $0 \leq \gamma < 1$

Partially Observable MDPs - probabilities



Partially Observable MDPs - beliefs

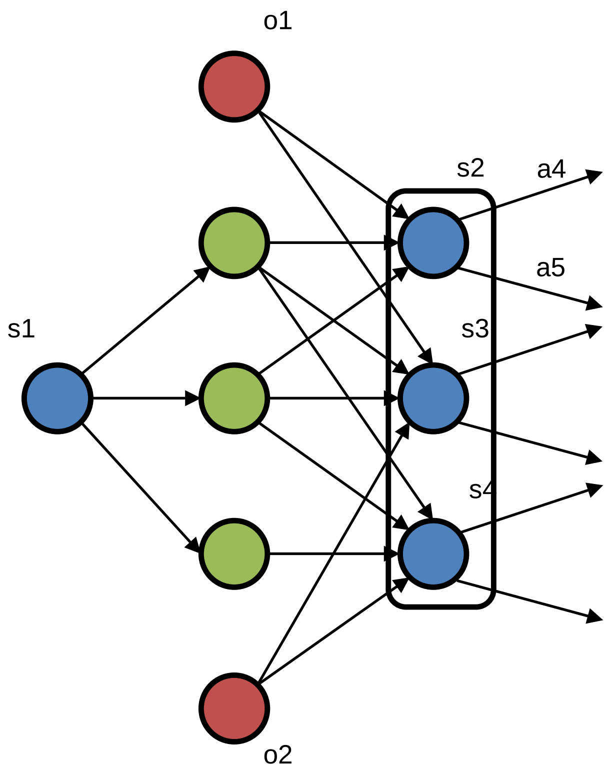
- beliefs represent a probability distribution over states
- beliefs are uniquely identified by the history
  - b_1 - probability distribution over states after playing one action
  - $b_t \leftarrow \Pr(s_t | b_0, a_0, o_1, \dots, o_{t-1}, a_{t-1}, o_t)$
- we can exploit dynamic programming (define transformation of beliefs, belief update)
  - $b_t(s') = \mu\Omega(a, o, s'). \sum_{s \in S} T(s, a, s') b_{t-1}(s)$
  - where
    - o is the last observation
    - a is the last action
    - μ is the normalizing constant

Partially Observable MDPs - values

- beliefs determine new values
  - $V(b) = \max_{a \in A}[R(b,a) + \gamma \sum_{b' \in B} T(b,a,b')V(b')]$
- what we have done …
  - we have transformed a POMDP to a continuous state MDP
  - belief state is a simplex
    - $|S| - 1$ dimensions
- in theory we can use all the algorithms for MDPs (value iteration)
  - but B is infinite

Solving Continuous State MDPs

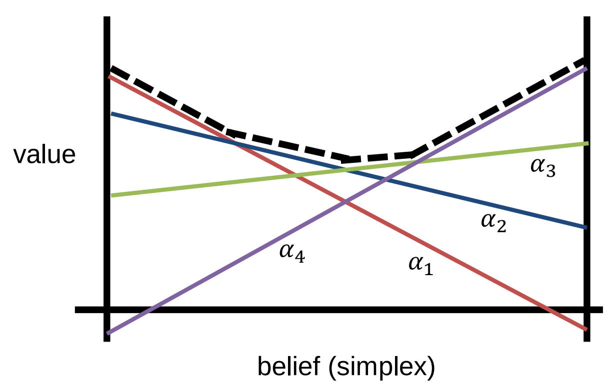
- in value iteration we take max of actions
- the belief space can be partitioned depending on the fact, which action is the best one

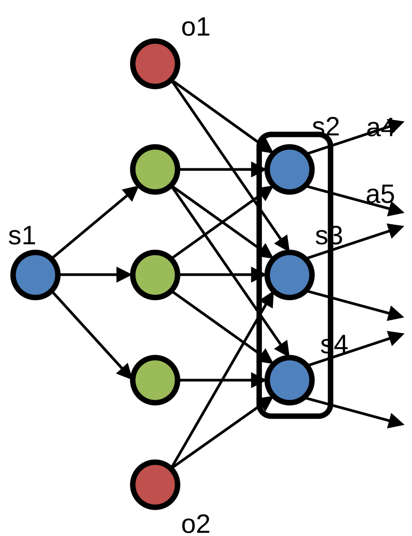


s2	s3	s4	V(a4)	V(a5)
0.2	0.1	0.7	3	2
0.7	0.1	0.2	1	7

Solving Continuous State MDPs

- values can be compactly represented as a finite set of α vectors; $V = \{\alpha_0, \dots, \alpha_m\}$
  - α vector is an $|S|$ dimensional hyper-plane
    - a linear function representing utility values after selecting some fixed action
  - defines the value function over a bounded region of the belief
  - $V(b) = \max_{\alpha \in V} \sum_{s \in S} \alpha(s) b(s)$
  - V is a piece-wise linear convex function

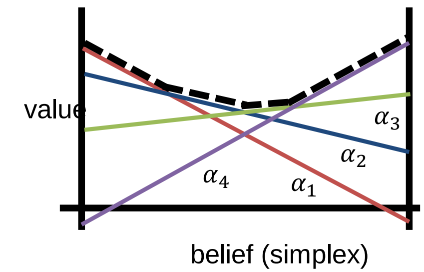




Solving Continuous State MDPs

- **Q: Can we modify value iteration algorithm to work with α functions?**

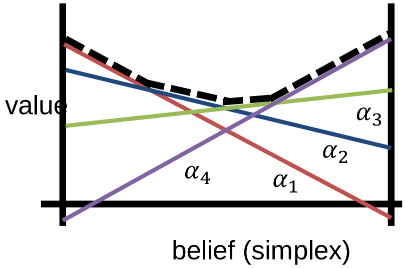
- exact value iteration for POMDPs
  - $V^t(b) = \max_{a \in A} [\sum_{s \in S} R(s,a) b(s) +$
  - $+\gamma \sum_{o \in O} \max_{\alpha' \in V^{t-1}} \sum_{s \in S} \sum_{s' \in S} T(s,a,s') \Omega(o,s',a) \alpha'(s') b(s)]$



- the above formula compute values (we need α-vectors)
  - $\alpha^{a,*}(s) = R(s,a)$
  - $\alpha_i^{a,o}(s) = \gamma \sum_{s' \in S} T(s,a,s')\, \Omega(o,s',a) \alpha_i'(s') \qquad \forall \alpha'_i \in V'$
  - $V^a = \alpha^{a,*} \oplus \alpha^{a,o_1} \oplus \alpha^{a,o_2} \oplus \cdots$
  - $V = \cup_{a \in A} V^a$

Exact Value Iteration for POMDPs

- exact baseline algorithm, however has several disadvantages
- complexity
  - exponential in size of observations $|O|$
  - base of the exponent is $|V|$
  - it is important to remove dominated alpha-vectors
  - useful only for very small domains

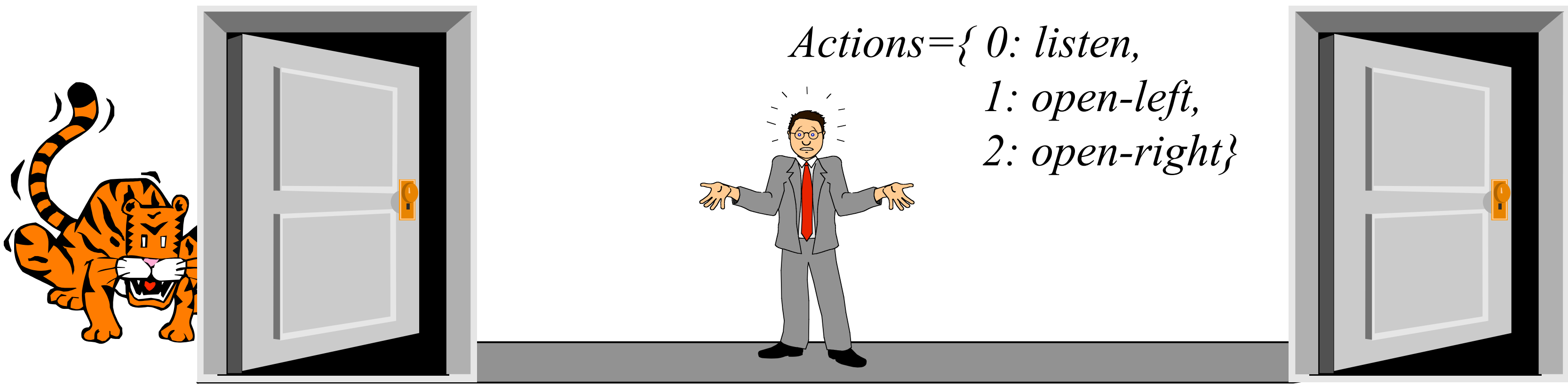


- Tiger example

A POMDP example: The tiger problem

S0
"tiger-left"
Pr(o=TL | S0, listen)=0.85
Pr(o=TR | S1, listen)=0.15

S1
"tiger-right"
Pr(o=TL | S0, listen)=0.15
Pr(o=TR | S1, listen)=0.85



Actions={ 0: listen,
1: open-left,
2: open-right}

Reward Function

- *Penalty for wrong opening: -100*
- *Reward for correct opening: +10*
- *Cost for listening action: -1*

Observations

- *to hear the tiger on the left (TL)*
- *to hear the tiger on the right(TR)*

Tiger Problem (Transition Probabilities)

•

Prob. (LISTEN)	Tiger: left	Tiger: right
Tiger: left	1.0	0.0
Tiger: right	0.0	1.0

Doesn't change

Tiger location

Prob. (LEFT)	Tiger: left	Tiger: right
Tiger: left	0.5	0.5
Tiger: right	0.5	0.5

Problem reset

Prob. (RIGHT)	Tiger: left	Tiger: right
Tiger: left	0.5	0.5
Tiger: right	0.5	0.5

Tiger Problem (Observation Probabilities)

Prob. (LISTEN)	O: TL	O: TR
Tiger: left	0.85	0.15
Tiger: right	0.15	0.85

Prob. (LEFT)	O: TL	O: TR
Tiger: left	0.5	0.5
Tiger: right	0.5	0.5

Any observation

Without the listen action

Is uninformative

Prob. (LEFT)	O: TL	O: TR
Tiger: left	0.5	0.5
Tiger: right	0.5	0.5

Tiger Problem (Immediate Rewards)

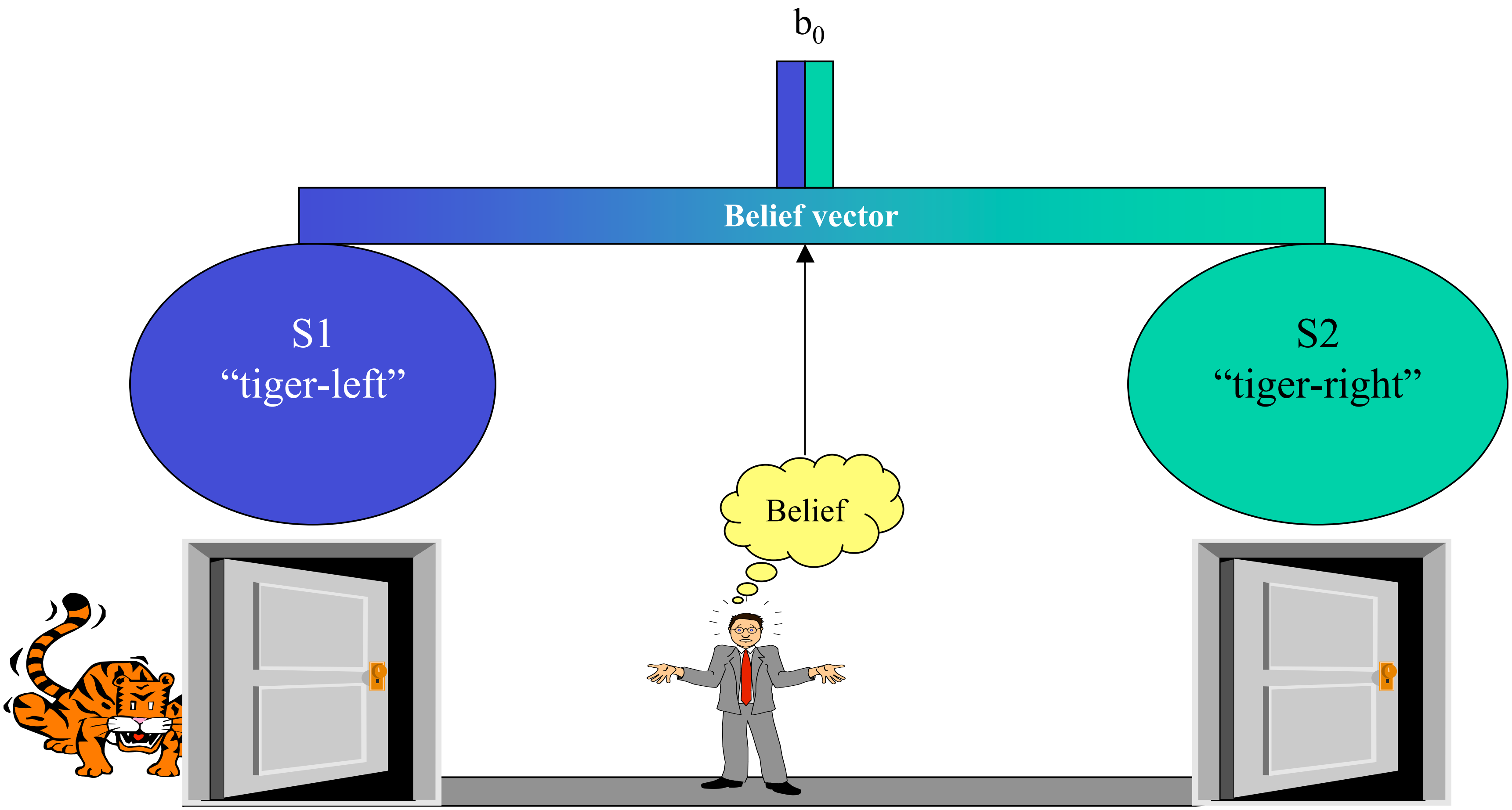
•

Reward (LISTEN)	
Tiger: left	-1
Tiger: right	-1

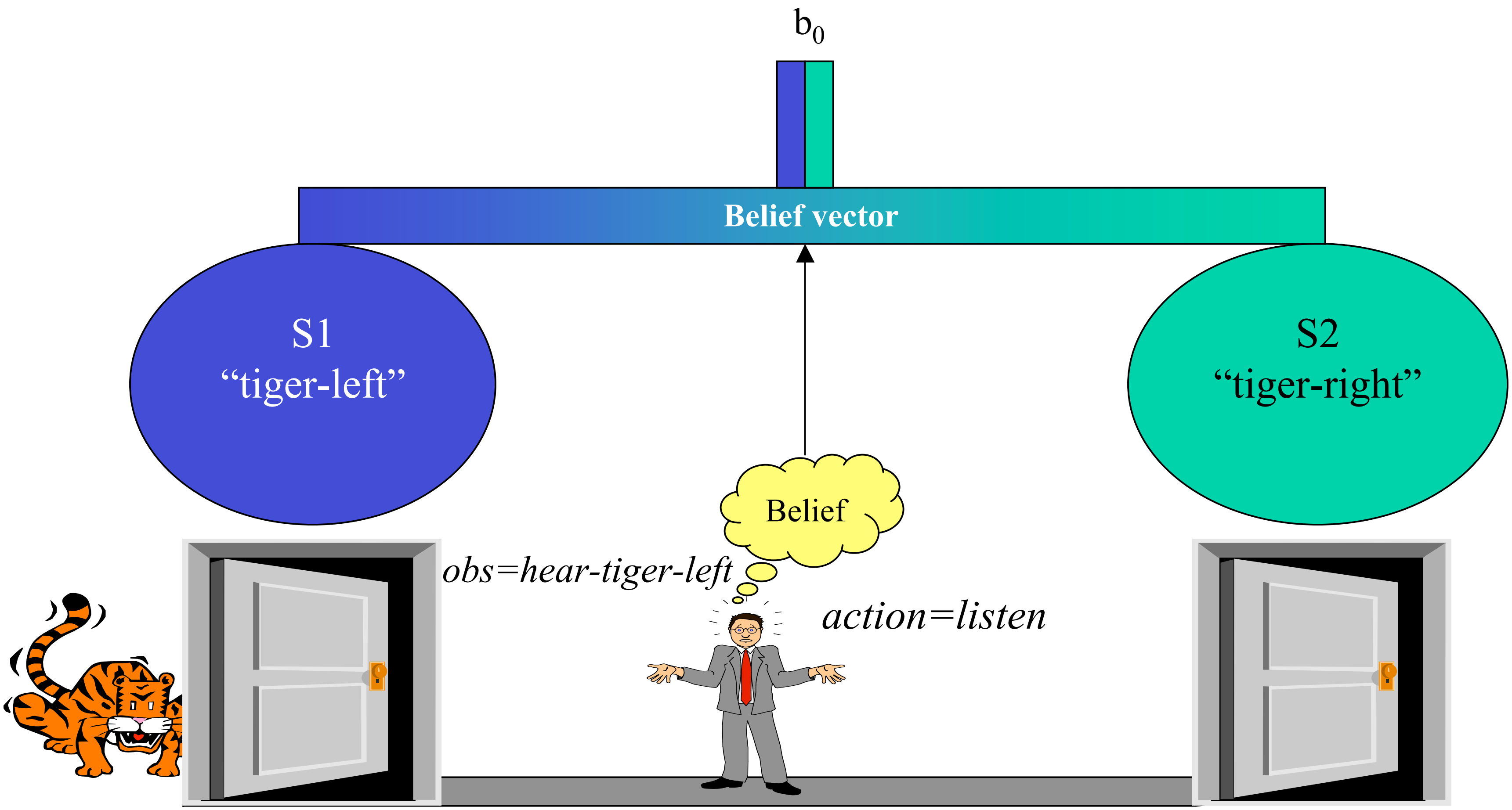
Reward (LEFT)	
Tiger: left	-100
Tiger: right	+10

Reward (RIGHT)	
Tiger: left	+10
Tiger: right	-100

The tiger problem: State tracking

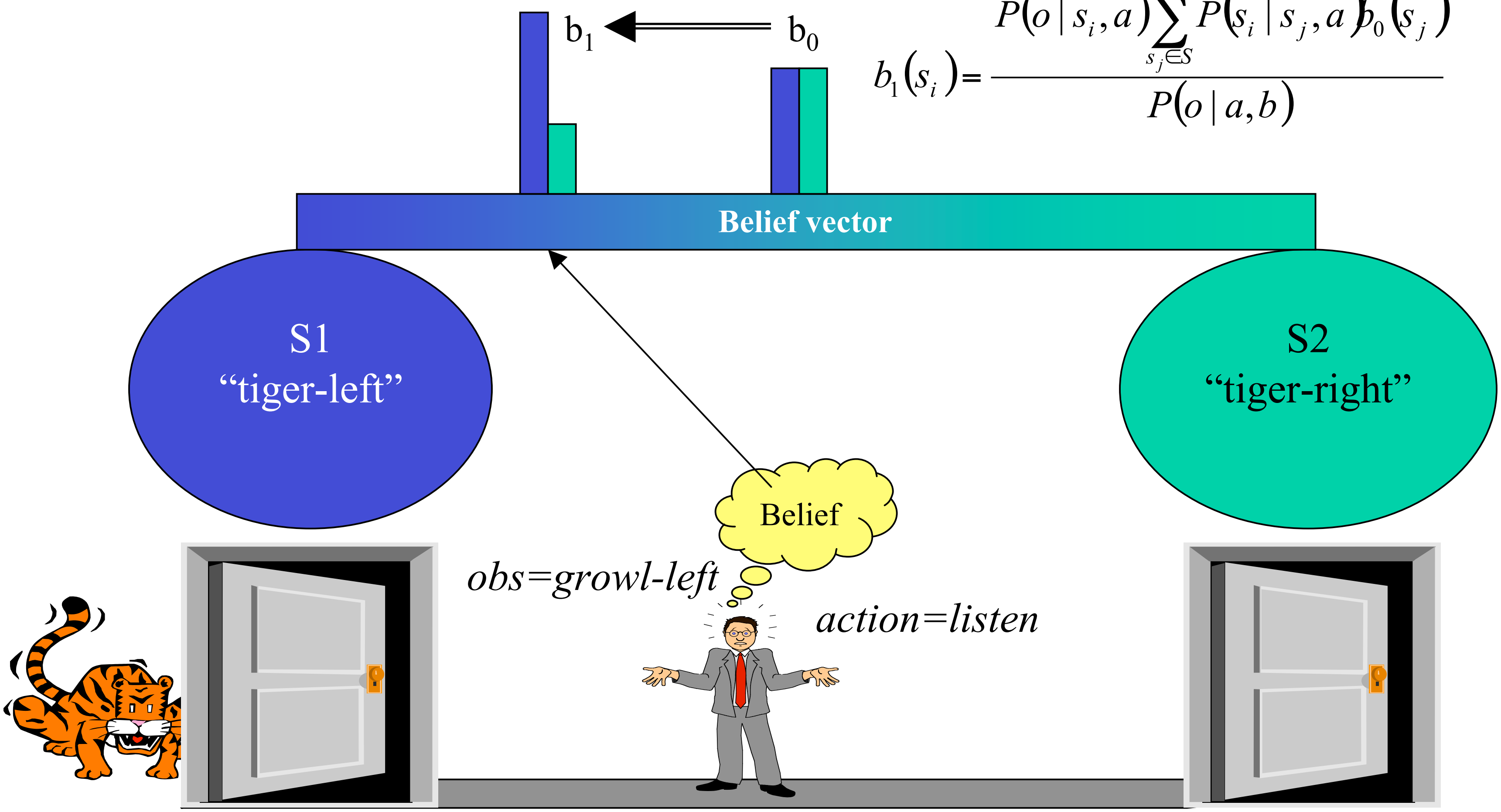


The tiger problem: State tracking



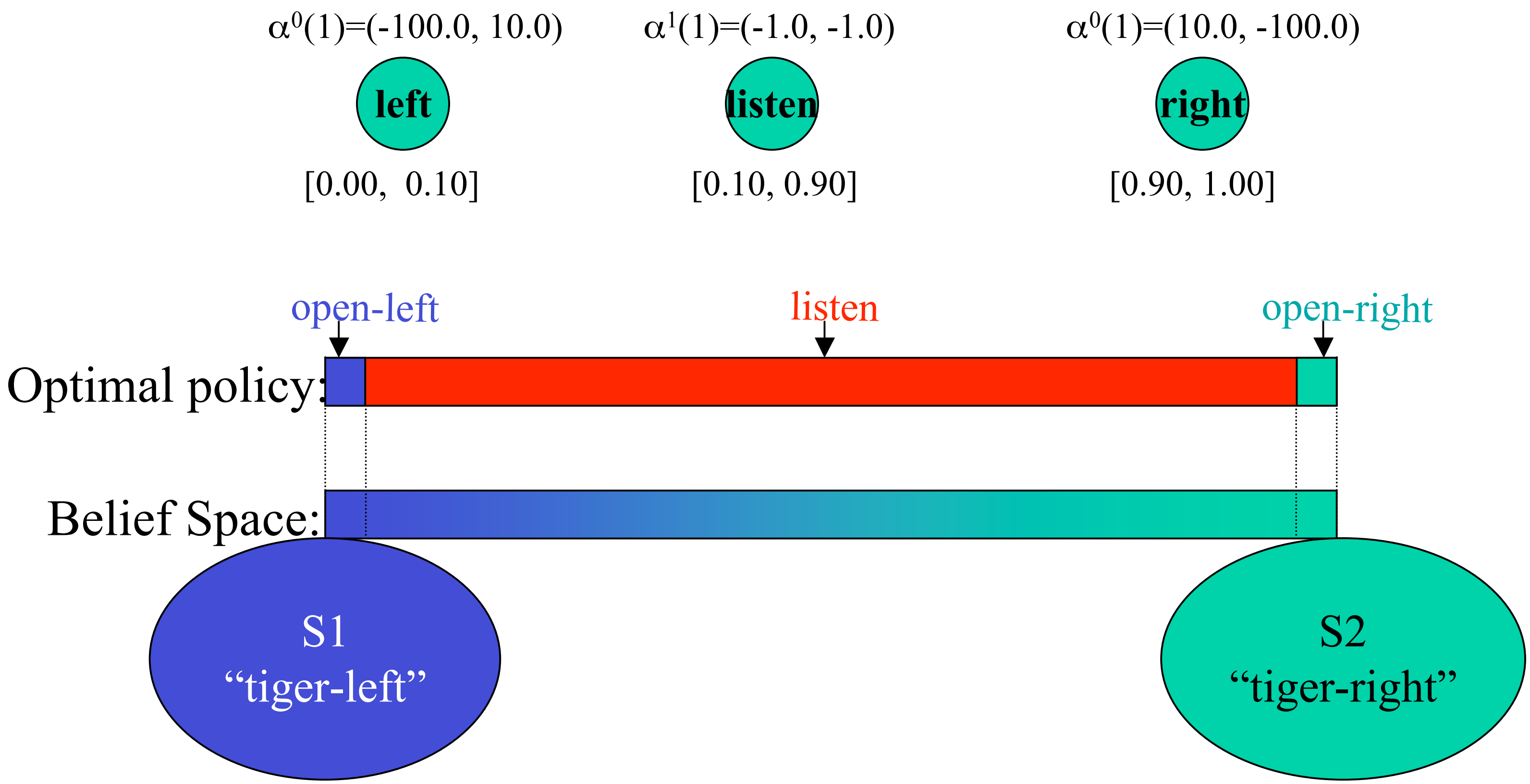
The tiger problem: State tracking

$$b_1(s_i) = \frac{P(o \mid s_i, a) \sum_{s_j \in S} P(s_i \mid s_j, a)\, b_0(s_j)}{P(o \mid a, b)}$$



Tiger Example Optimal Policy t=1

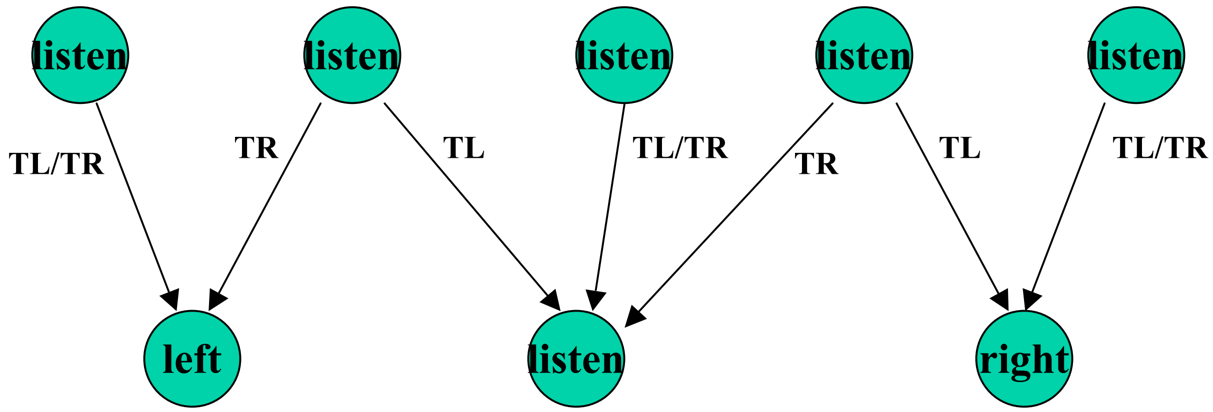
- Optimal Policy for t=1



Tiger Example Optimal Policy for t=2

- For t=2

[0.00, 0.02] [0.02, 0.39] [0.39, 0.61] [0.61, 0.98] [0.98, 1.00]

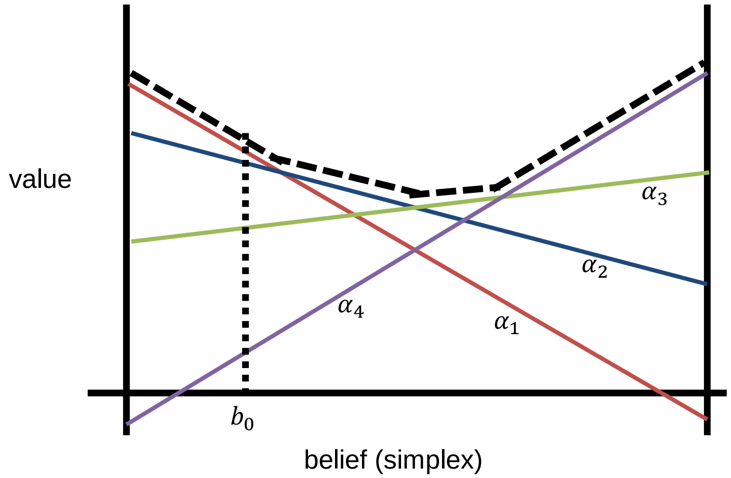


Exact Value Iteration for POMDPs

- can we do better than full value iteration?
- only a fraction of all belief state is actually achievable in POMDP
  - we can sample the belief state

Point Based Value Iteration for POMDPs

- instead of the complete belief space we use a limited set
  - $B = \{b_0, \dots, b_q\}$
- the algorithm keeps only a single alpha vector for one belief point
- anytime algorithm altering 2 main steps
  - belief point value update
  - belief point set expansion

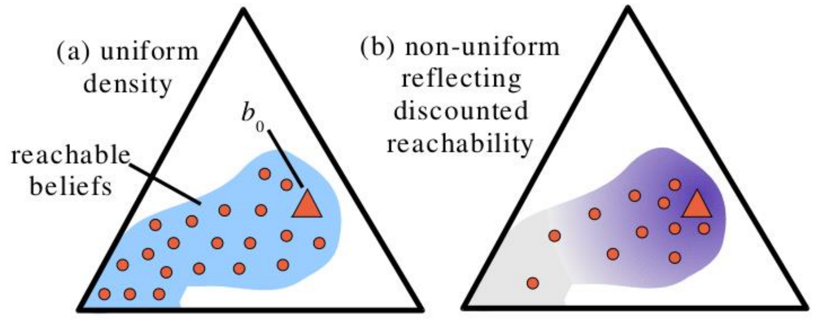


Point Based Value Iteration for POMDPs

- belief value update
  - $V_b^a = \alpha^{a,*} + \gamma \sum_{o \in O} \arg \max_{\alpha \in \alpha_i^{a,o}} (\alpha . b)$
  - $V \leftarrow \arg \max_{V_b^a, \forall a \in A} V_b^a . b \quad \forall b \in B$
- removes the exponential complexity
- VI state ends after h iterations
  - finite horizon / the error is smaller than ε
- belief point set expansion
  - sampling new beliefs from existing beliefs
  - trying to uniformly cover reachable belief space

Point Based Value Iteration for POMDPs

- further improvements
- exploiting heuristics
  - for setting initial values
  - selecting belief points



- current scalability
  - up to 10^5 states of POMDP
- further reading
  - Shani, Pineau, Kaplow: A survey of point-based POMDP solvers (2012)

Beyond (PO)MDPs

- many other models
- specific variants of MDPs / generalization
  - AND/OR graphs
  - influence diagrams
  - dynamic Bayesian networks
- multiple agents
  - decentralized (PO)MDPs - DEC-(PO)MDPs
    - theoretical framework for multi-agent planning
  - partially observable stochastic games (POSG)
    - theoretical framework for interaction of rational agents