Improving Human-Robot Interaction: Efficient Model Learning for Dialog Management

Finale Doshi
Nick Roy

CSAIL

Outline

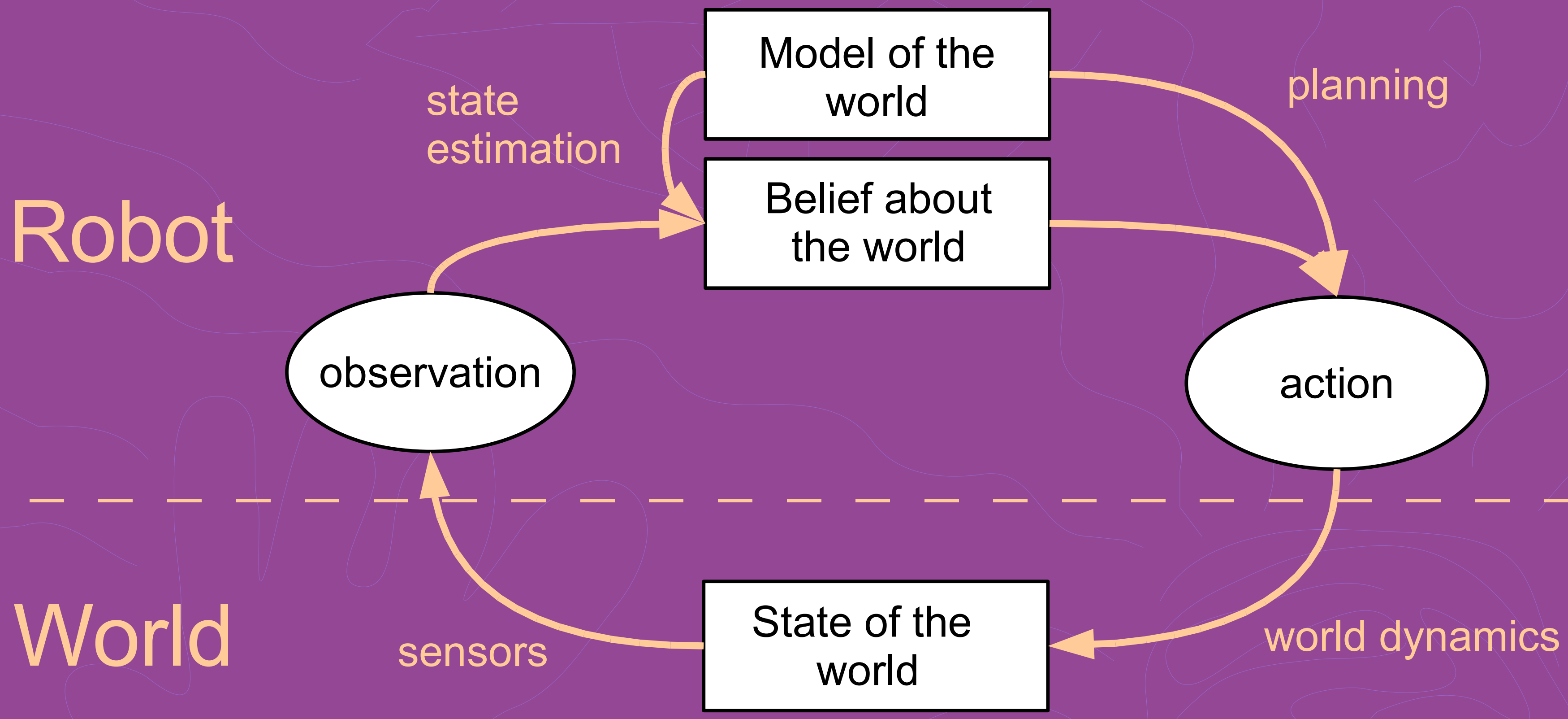
- What can our robots do now?
- What would we like our robots to do?
- Specific Scenario: Dialog Management
  - Using Partially Observable Markov Decision Processes (POMDPs) to handle dialog uncertainty
  - Learning from human-robot interactions

What can our robots do now?

CSAIL

What can our robots do now?

How do they do they do this?



How do they do they do this?

Examples of sensing and planning.

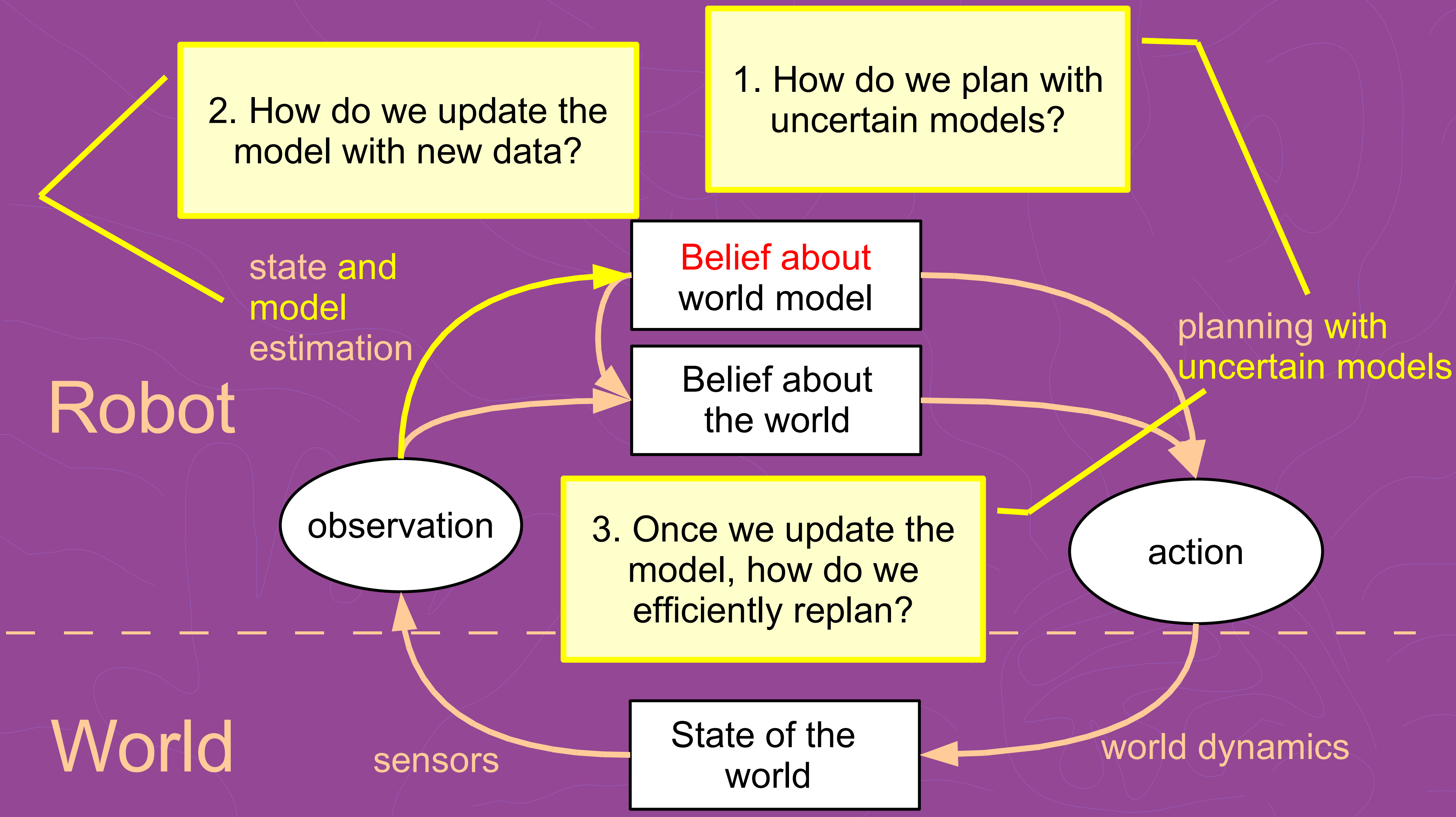
CSAIL

What would we like our robots to do?

What would we like our robots to do?

- People think in terms of trajectories, corridors and goals – not occupancy grids or laser scans.
- Wouldn't it be nice if we could simply tell the robot
  - to follow behind us?
  - to go to a location and fetch something?
  - to remember the name of a new room or person?
- However, models of humans are very difficult to specify! We must learn to work with uncertain models.

What would we like our robots to do?



Dialog Management (with uncertain models)

- Background on Partially Observable Markov Decisions Processes (POMDPs)
- Working with uncertain models in dialog management

CSAIL

Background on POMDPs (for dialog management)

The Problem

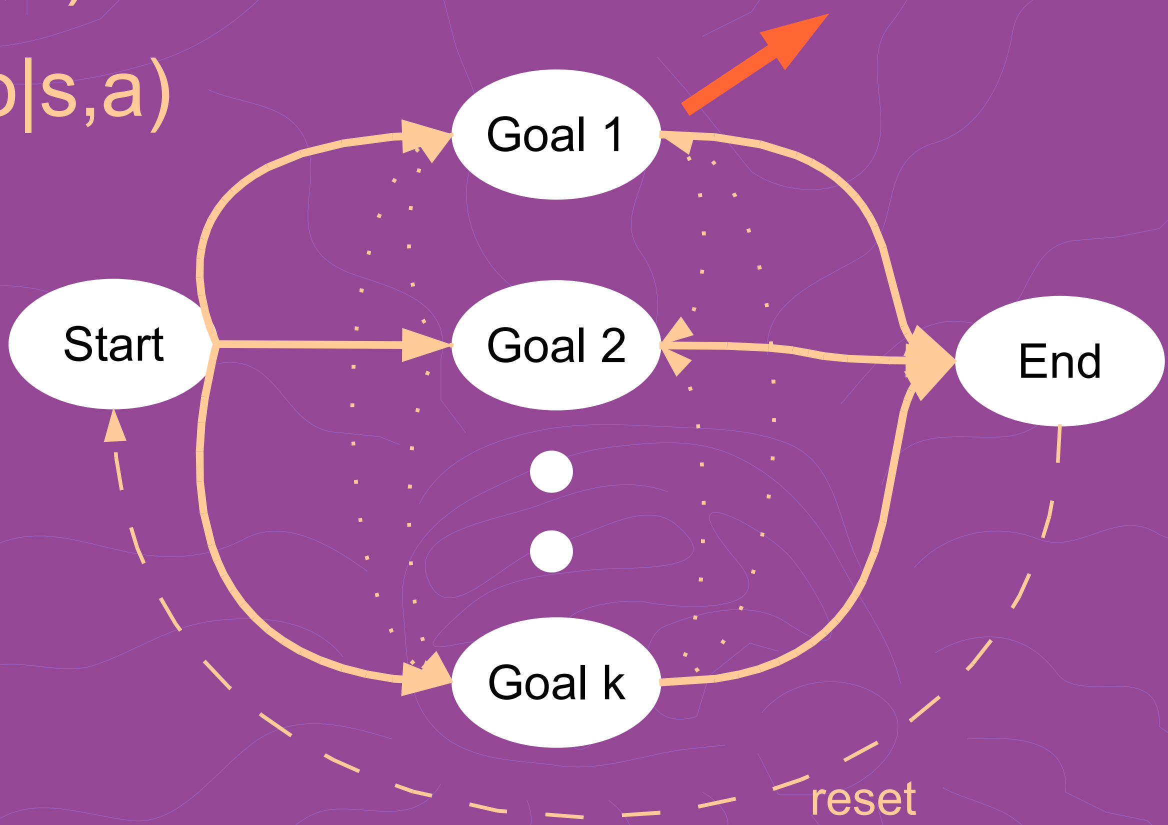
Spoken language allows for natural human robot interaction, but there are several challenges:

- Noisy speech recognition
  - ex. “Gates” becomes “Good”
- Linguistic ambiguities
  - multiple “elevators” may exist
  - the robot must know that an “elevator” is a location

Dialog Model: POMDPs

- Set of states (S), actions (A), and observations (O)
- Transition Model T(s'|s,a)
- Observation Model O(o|s,a)
- Reward Model R(s,a)

Large number of parameters difficult to specify a priori!



Dialog Model: Solving the POMDP

Value of a belief

Value of belief, action pair

$$V_n(b) = \max_{a \in A} Q(b, a),$$

$$Q_n(b, a) = R(b, a) + \gamma \sum_{b' \in B} T(b'|b, a) V_{n-1}(b'),$$

$$Q_n(b, a) = R(b, a) + \gamma \sum_{o \in O} \Omega(o|b, a) V_{n-1}(b_a^o),$$

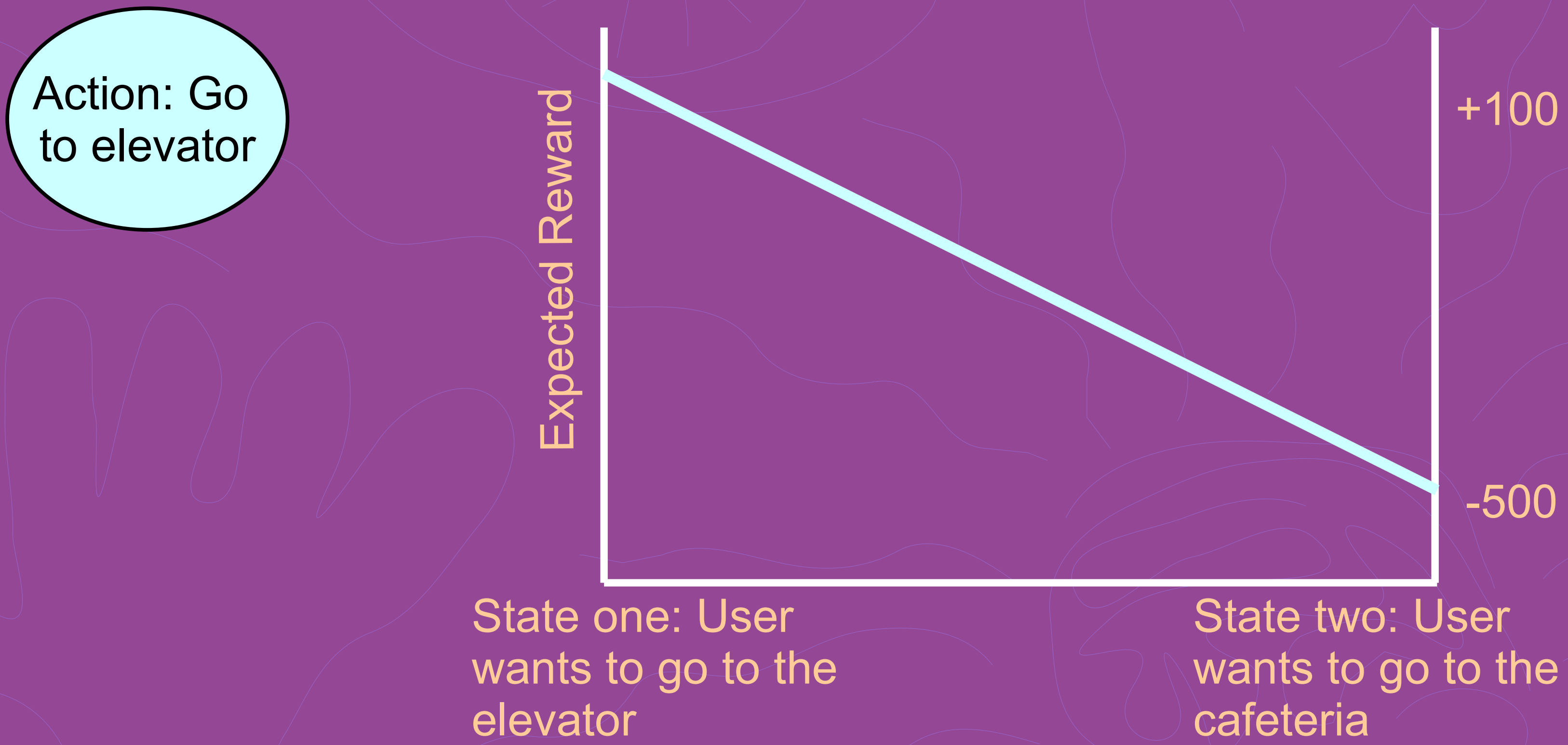
Current reward

Future Reward

We can apply these equations to solve for V(b) recursively.

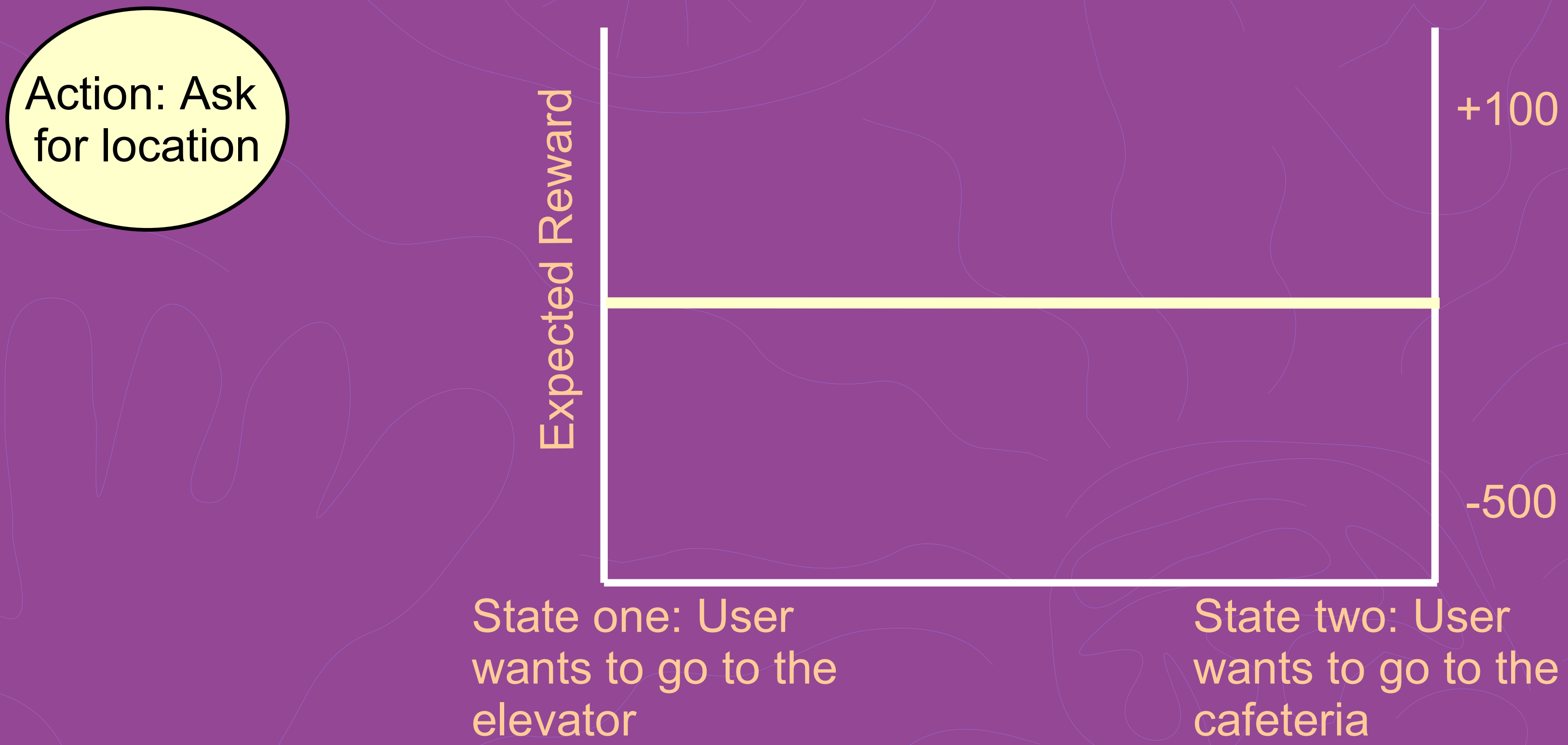
Dialog Model: Solving the POMDP

We think of the previous recursions as building a policy tree...



Dialog Model: Solving the POMDP

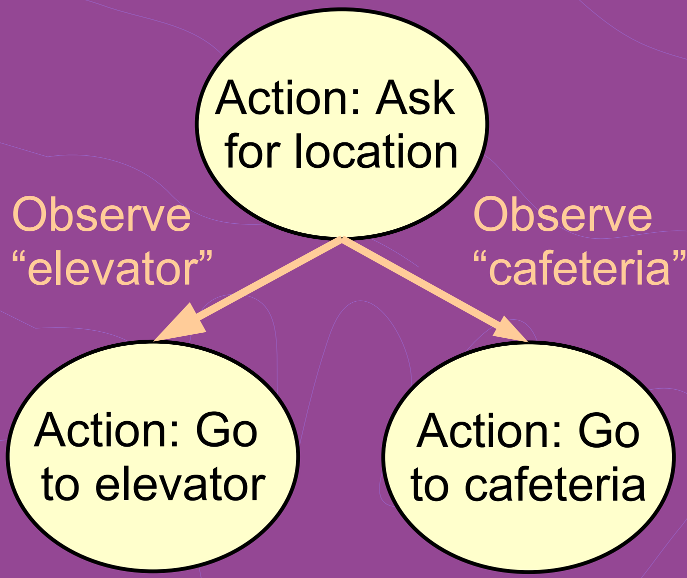
We think of the previous recursions as building a policy tree...

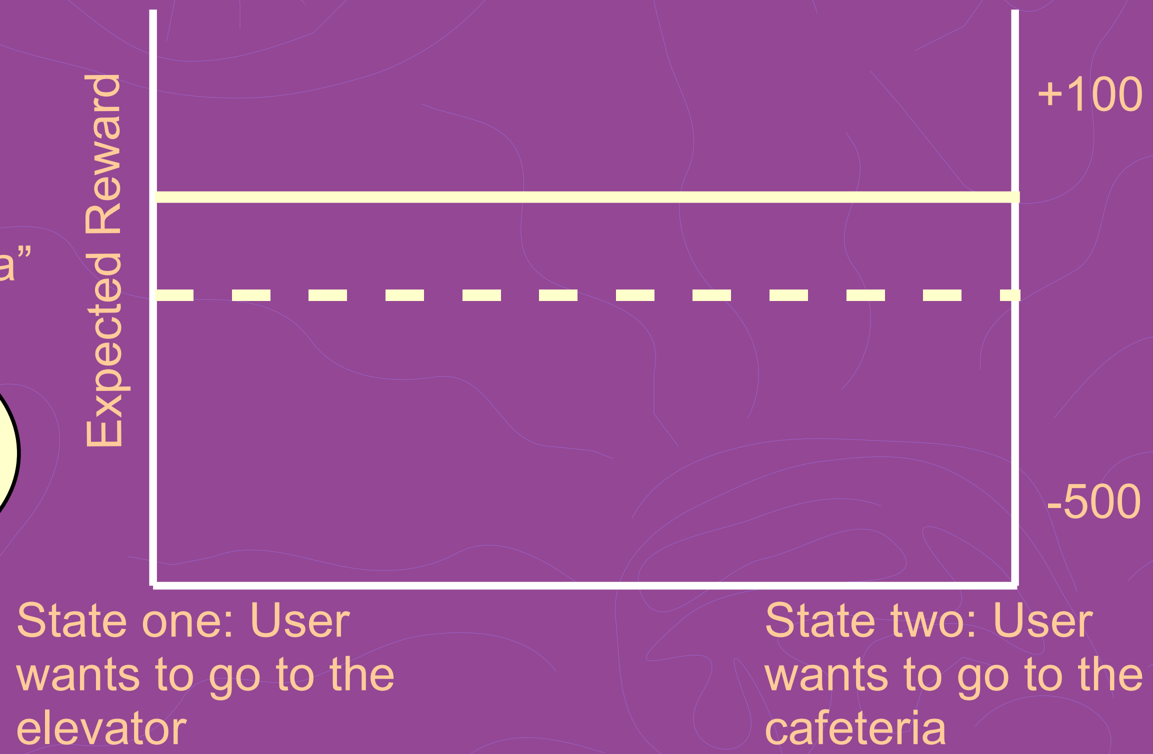


CSAIL

Dialog Model: Solving the POMDP

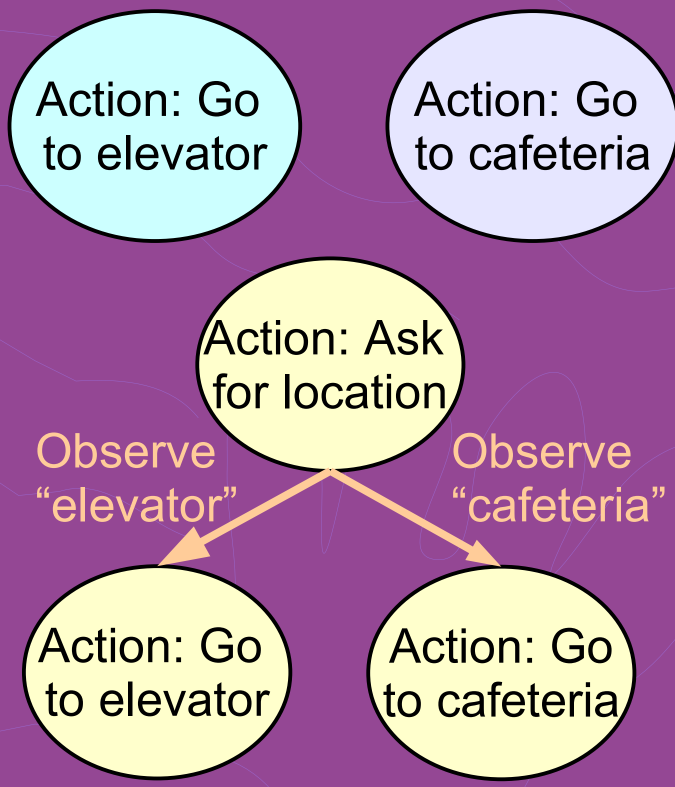
We think of the previous recursions as building a policy tree...

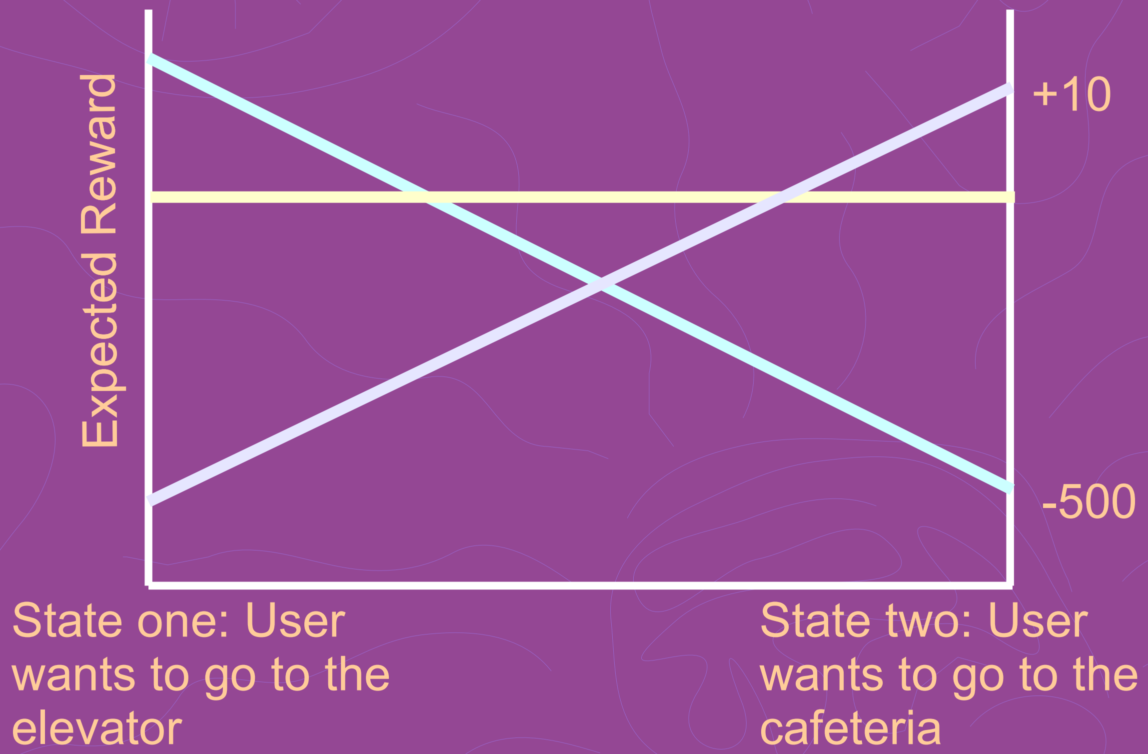




Dialog Model: Solving the POMDP

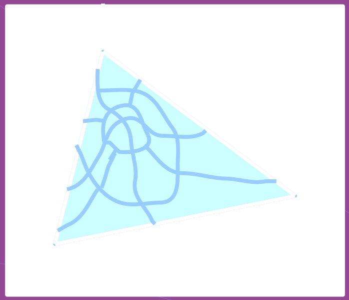
Given multiple trees, we can determine the most appropriate action:



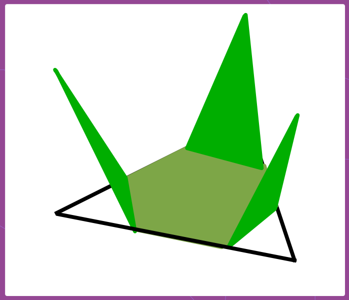


Working with Uncertain Models

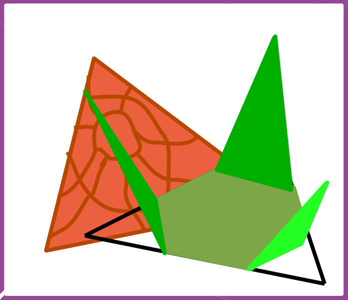
Our Approach



- Place priors over parameters
  - “Expert” provides estimate and confidence
  - We convert to Dirichlet/Gaussian prior



- Find a policy that maximizes reward given the uncertainty in the parameters



- After each completed interaction
  - Update priors on the parameters
  - Update POMDP solution with additional backups

Planning with uncertain parameters

If parameters are uncertain, solve POMDP with the expected value of the parameters to optimize reward.

Expectation over states

$$Q(b,a) = \max_i \vec{q_a} \cdot b,$$

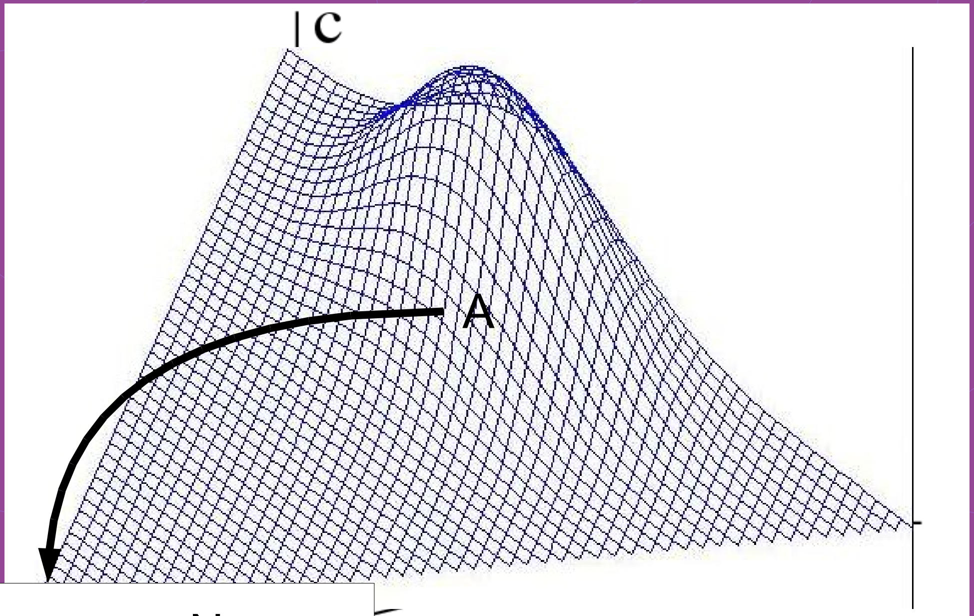
$$q_a(s) = \mathbf{E}[R(s,a)] + \gamma \sum_{o \in O} \sum_{s' \in S} \mathbf{E}[T(s'|s,a)\Omega(o|s',a)] V_{n-1,i}(s)$$

Expectations over model parameters

Expectation over model stochasticity

Modeling Uncertainty: Placing Priors

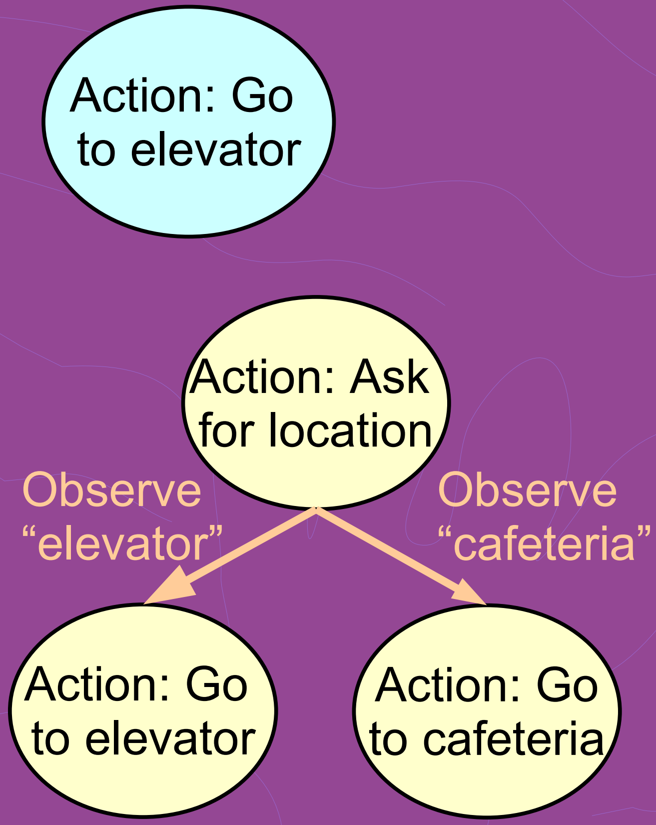
- Place Gaussian priors over each reward R(s,a)
- Place Dirichlet priors over each transition T(.|s,a) and O(.|s,a)
- Expert specifies a mode value and “pre-observation count” confidence score for each prior.

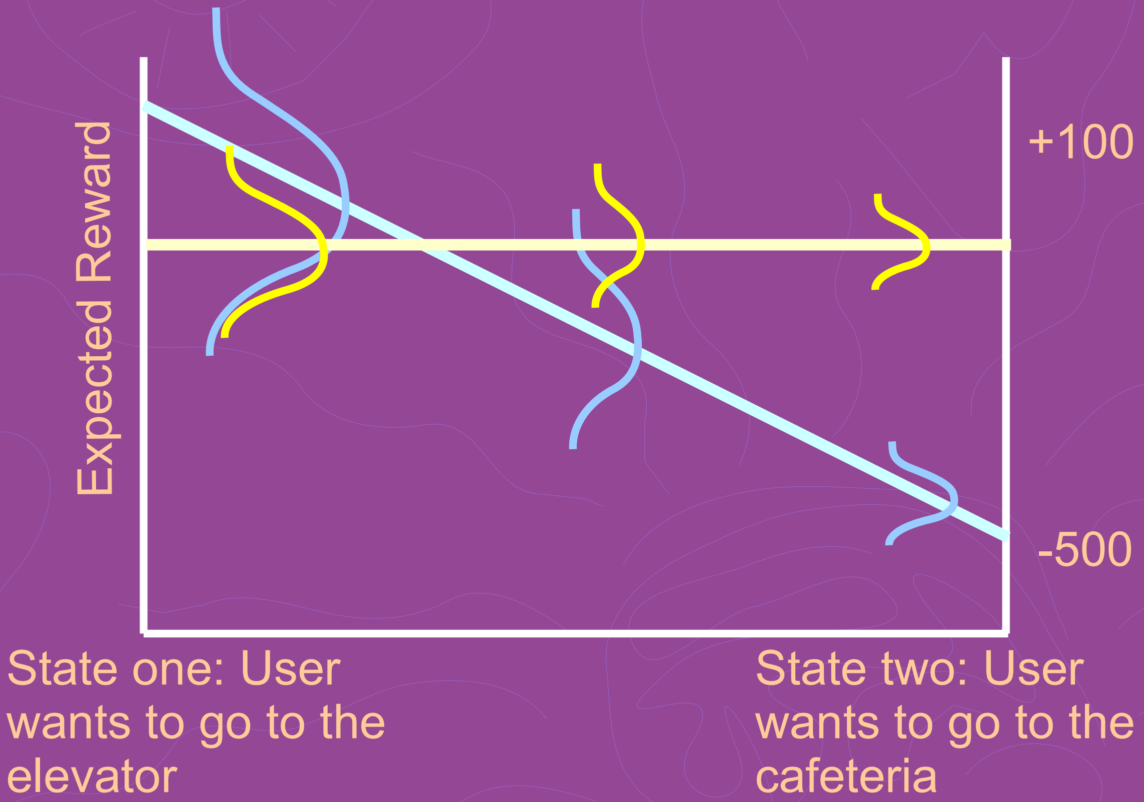


$$P(A) = 1/B(\boldsymbol{\alpha}) \prod_{k=1}^{N} p_k^{\alpha_k}$$

Modeling Uncertainty: Placing Priors

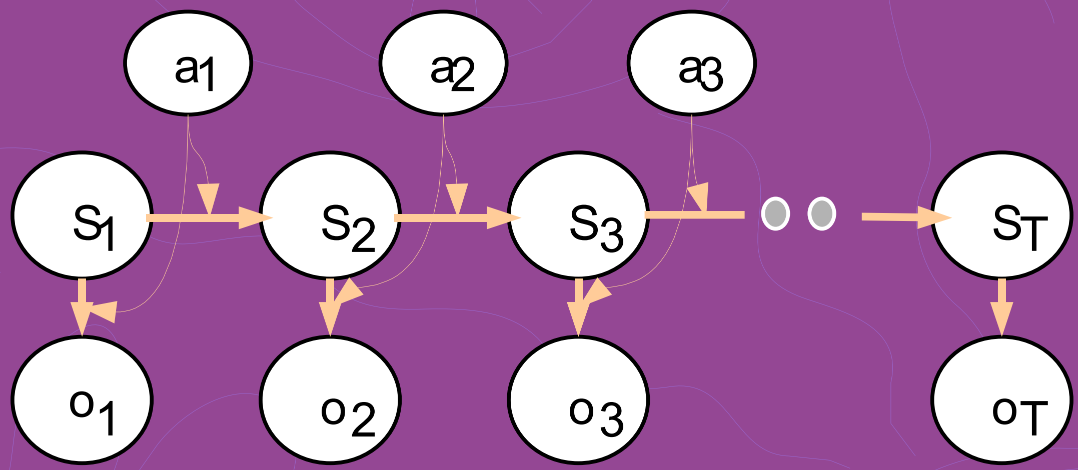
Intuitively, parameter uncertainty induces uncertainty in the value function:





Modeling Uncertainty: Updating Priors

- Given an action, observation history, the POMDP reduces to a Hidden Markov Model:



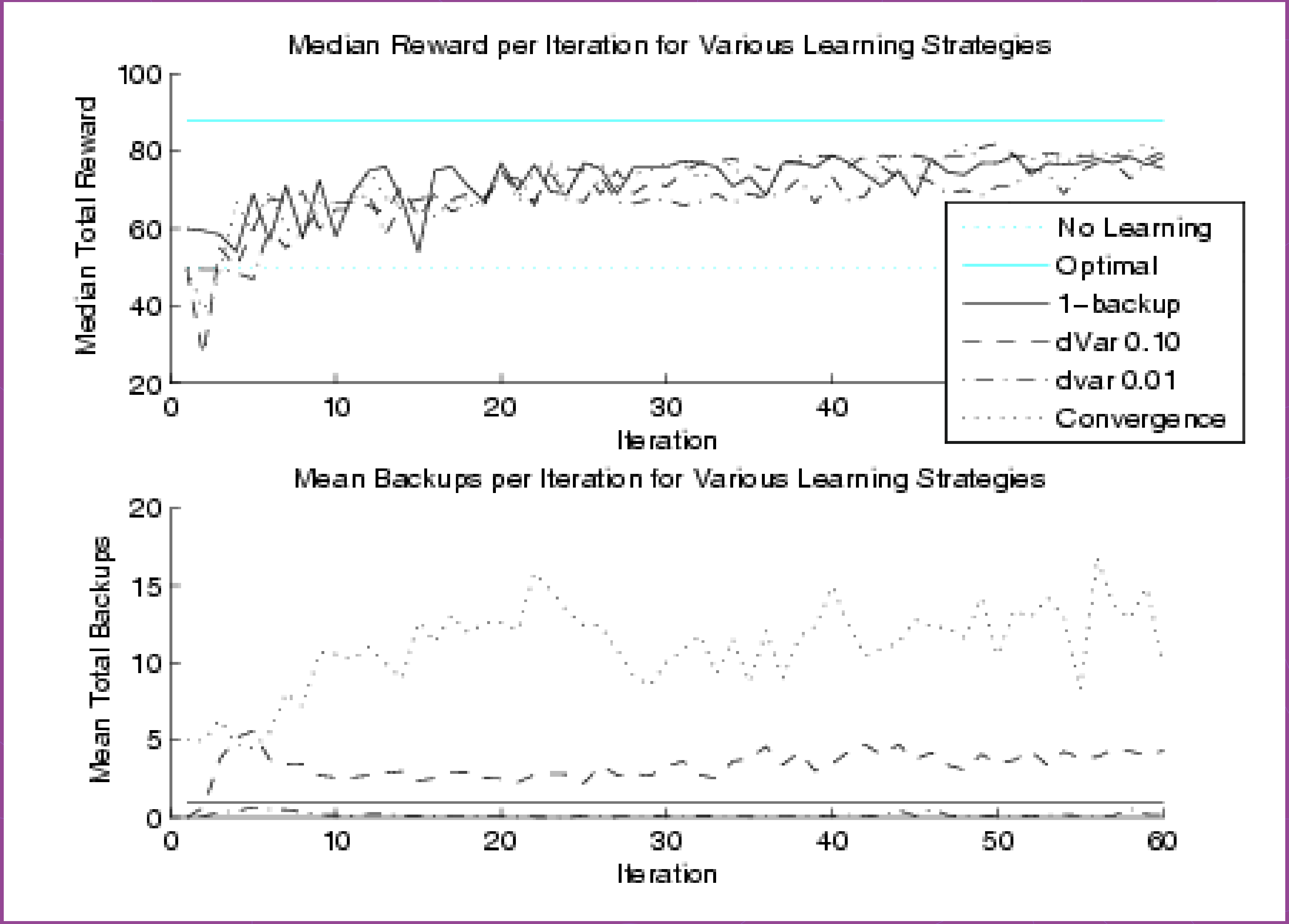
- Solve for most likely state sequence and use this to update Gaussian, Dirichlet distributions.

Updating the Policy

- How? Compute additional backups
  - POMDP solution will converge given new expected values for the parameters.
- How many backups?
  - Backup to convergence
  - Backup a fixed number of times
  - Backup proportionally to variance reduction in the parameters: $b = k \, dVar$, where $dVar = \sum_i \sigma^2_{i,t+1} - \sigma^2_{i,t}$

CSAIL

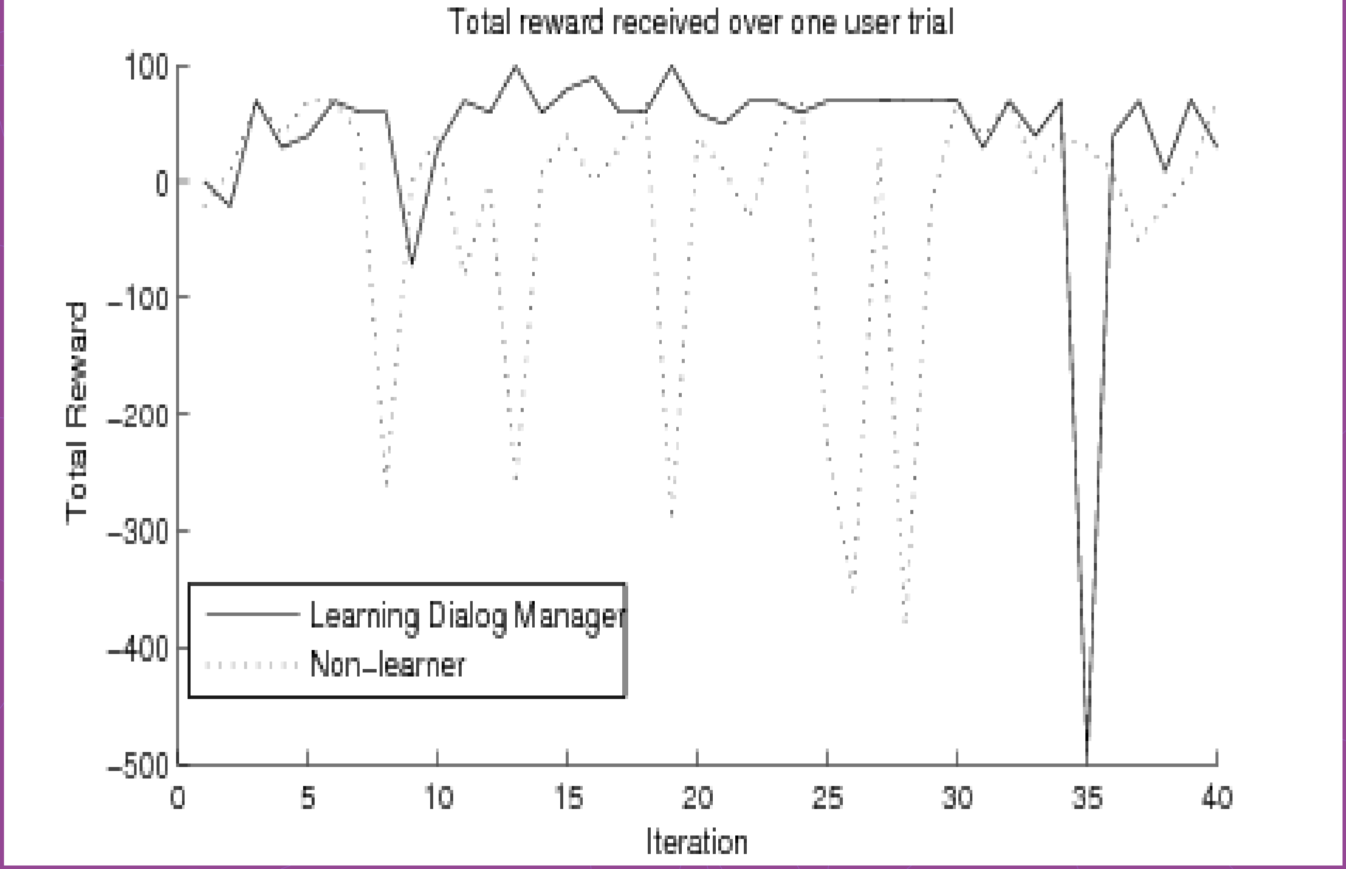
Simulation Results



Wheelchair Results

Non-learner
User: Take me to the elevator.
Robot: Where did you want to go?
User: The Gates elevator please.
Robot: Do you want to go to the Gates Elevator?
User: Yes.
Robot: Going to Gates.

Learner
User: Take me to the elevator.
Robot: Do you want to go to the Gates Tower?
User: Yes
Robot: Going to Gates



Current work: Improve the Heuristic

- Three sources of variation in solutions
  - Stochasticity inherent in model
  - Uncertainty in parameters
  - Incomplete solution convergence
- Refine solution until variation due to incomplete convergence is not significant compared to other sources.

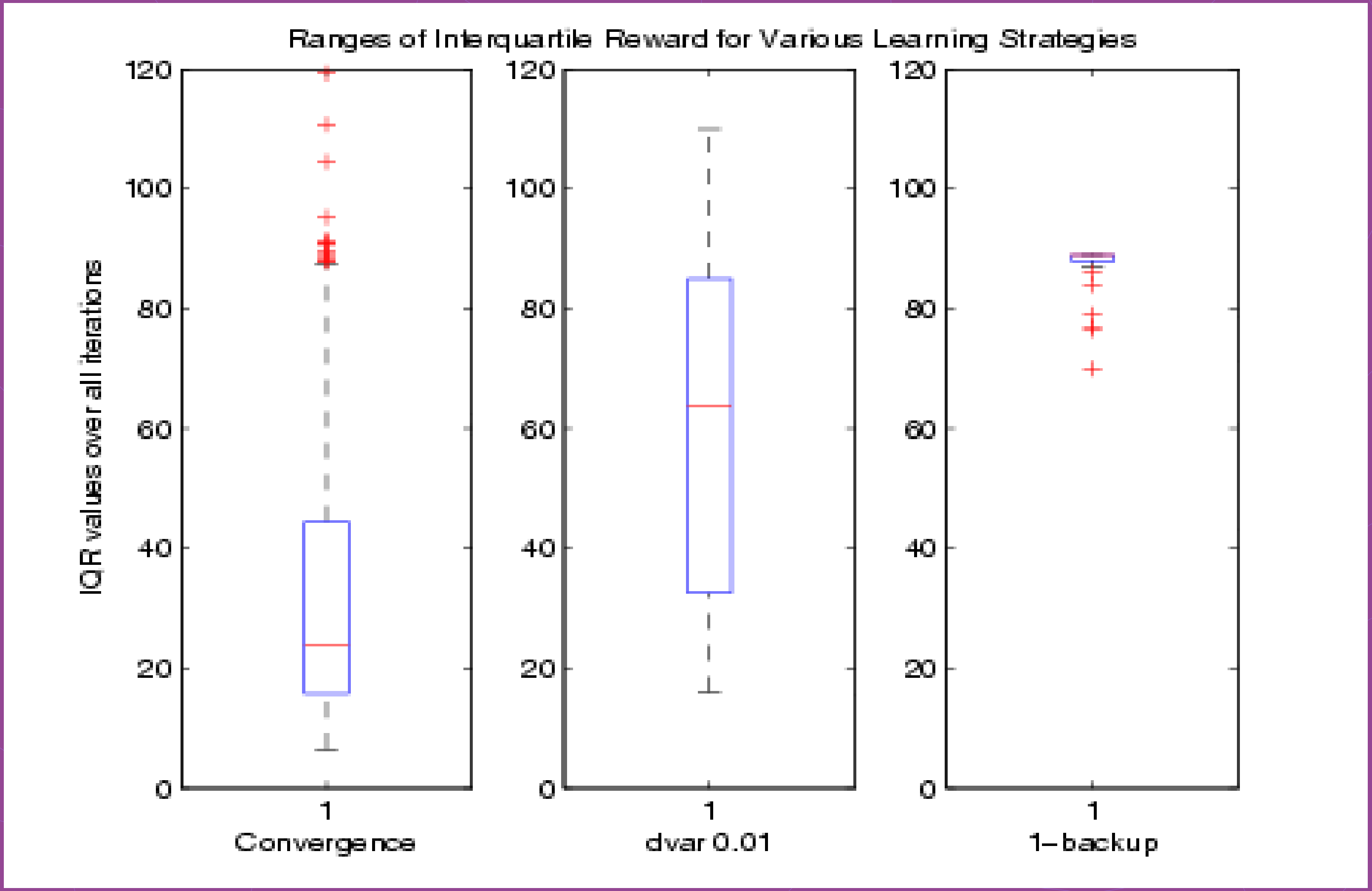
Current Work: Larger Questions

- How can we guarantee convergence of the important parameters?
- How can we make the POMDP “aware” of parameter uncertainty?
  - Should take exploratory actions *and*
  - Should act robustly
- How can more sophisticated language understanding improve interactions?
- What aspects of the dialog manager are most important to wheelchair users?

CSAIL

Thank-you

Simulation Results



Estimating Problem Uncertainty

$$var(V(b)) = f(b) + \sum_{o \in O} \eta_o(b) var(V(b_a^o))$$