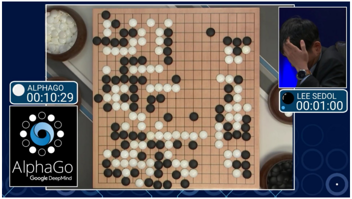
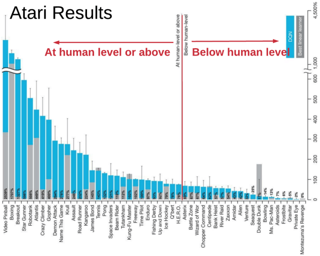
A Human-Centered Data-Driven Planner-Actor-Critic Architecture via Logic Programming

Daoming Lyu[1], Fangkai Yang[2], Bo Liu[1], Steven Gustafson[3]

[1]Auburn University, Auburn, AL, USA
[2]NVIDIA Corporation, Redmond, WA, USA
[3]Maana Inc., Bellevue, WA, USA

Sequential Decision-Making





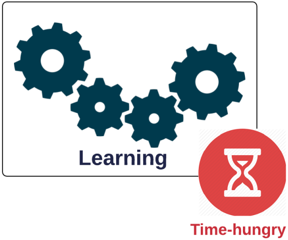
- **Sequential decision-making**: concerns an agent making a sequence of actions based on its behavior in the environment.
- **Reinforcement learning** has achieved tremendous success on sequential decision-making problems, i.e., training agent to play games on Atari 2600, which enables to learn human-level control policy *(Mnih et al., 2015)*.

Difficulties

- “Data-hungry” and “time-hungry”.
- Slow initial learning process with bad performance level of the initial policy, due to learning from scratch.

Difficulties

- "Data-hungry" and "time-hungry".
- Slow initial learning process with bad performance level of the initial policy, due to learning from scratch.
- By contrast, human learning can be faster.



Our Solution

Human Learning

- Embodied with prior and abstract knowledge.
- Learn from multiple information resources, including environmental reward signals, human feedback, or demonstrations.

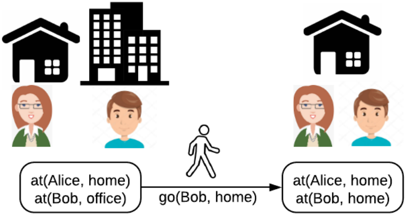
Solution

- A unified framework where knowledge-based **planning**, reinforcement **learning**, and **human feedback** jointly contribute to the policy learning of an agent.

Background: Action Language

Action language *(Gelfond & Lifschitz, 1998)*: a formal, declarative, logic-based language that describes dynamic domains.

- Dynamic domains can be represented as a transition system.



Action Language $\mathcal{BC}$

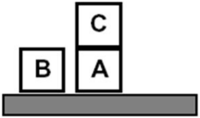
Action Language $\mathcal{BC}$ (Lee et al., 2013) is a language that describes the transition system using a set of *causal laws*.

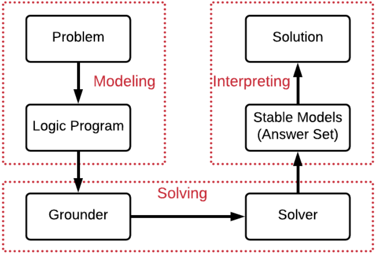
- *dynamic laws* describe transition of states

$$move(x, y_1, y_2) \text{ causes } on(x, y_2) \text{ if } on(x, y_1).$$

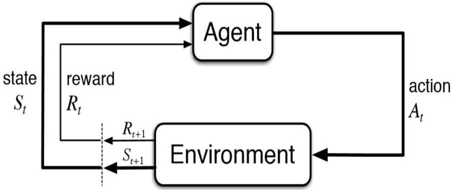
- *static laws* describe value of fluents inside a state

$$intower(x, y_2) \text{ if } intower(x, y_1), on(y_1, y_2).$$



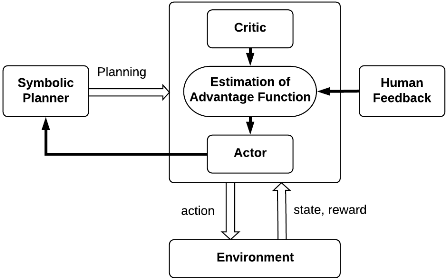


Background: Reinforcement Learning



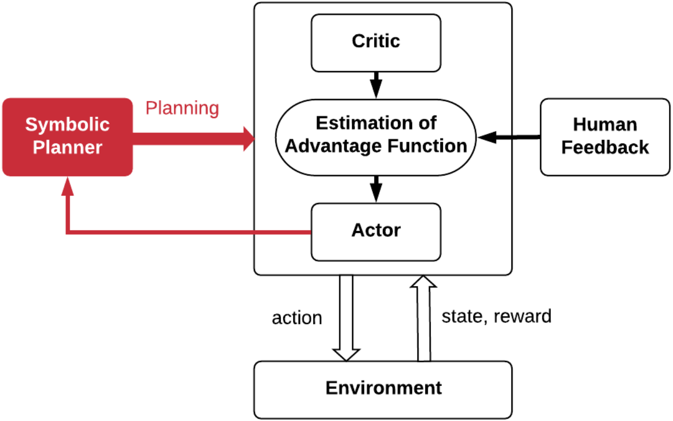
- **Reinforcement learning** is defined on a Markov Decision Process $(\mathcal{S}, \mathcal{A}, P^a_{ss'}, r, \gamma)$.
  - $\mathcal{S}, \mathcal{A}$ denote the state and action spaces.
  - transition probability model $P^a_{ss'}$.
  - reward function r.
  - discount factor γ.
- To achieve the maximal cumulative reward, a policy $\pi : \mathcal{S} \times \mathcal{A} \mapsto [0, 1]$ is learned by the agent.

PACMAN: Planner-Actor-Critic architecture for huMAN-centered planning and learning



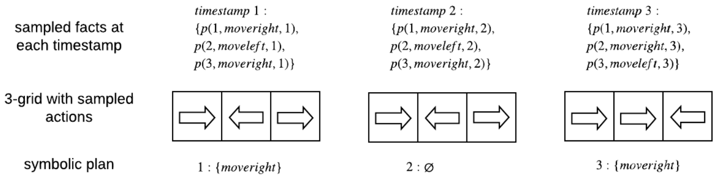
- **Symbolic Planner**: generates the symbolic plan based on the sampled facts.
- **Actor-Critic Learner**: learns from the experience by executing the symbolic plan.
- **Human Feedback**: interpreted as an estimation of advantage function.

Symbolic Planner

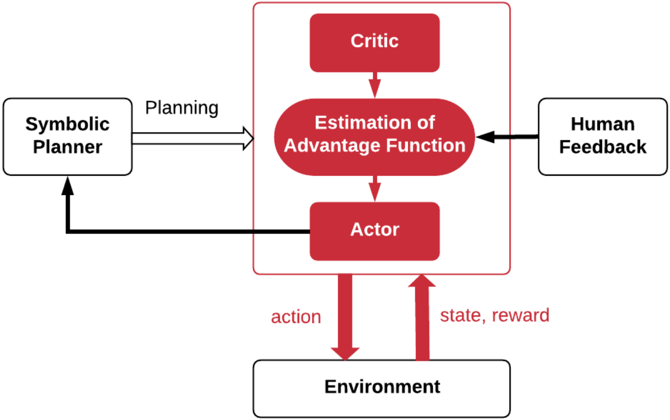


Sample-based Symbolic Planning

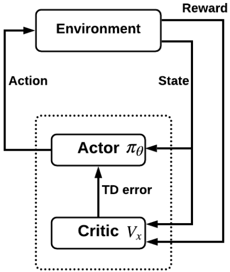
- Sample-based planning problem is defined on tuple (I, G, π_θ, D):
  - initial state condition I.
  - goal state condition G.
  - a stochastic policy function π_θ.
  - action description D, which contains a set of facts sampled from π_θ.
- A simple planning example.



Actor-Critic Learner

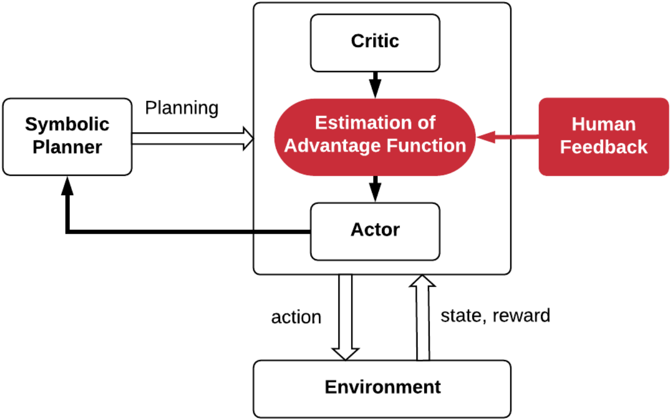


Actor-Critic Architecture



- Critic: state-value function V_x that criticizes the action taken by the learner.
- Actor: policy function π_θ that is used for action selection.
- Advantage function: how much better or worse an action a is compared to the current policy at state s.
- Temporal difference(TD) error: $r(s,a) + \gamma V_x(s') - V_x(s)$.

Human Feedback



Human Feedback

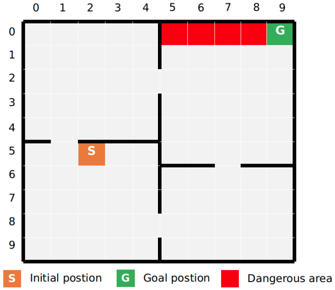
- Human feedback for making decision is dependent on learner's current policy *(MacGlashan et al., 2017)*.
- Advantage function provides a better model of human feedback.
- Guide exploration towards human preferred behaviors.

Experiment Setting

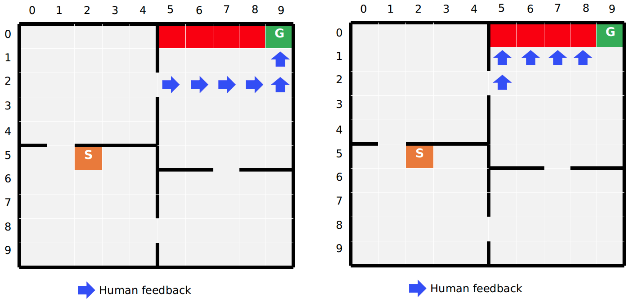
- Domains: four rooms and taxi.
- Baseline methods:
  - BQL *(Griffith et al., 2017)*.
  - TAMER+RL *(Knox & Stone, 2012)*.
  - Actor-critic with human feedback.
- Two scenarios: helpful or misleading human feedback.
  - ideal case.
  - inconsistent case.
  - infrequent case.
  - infrequent+inconsistent case.

Four Rooms Domain

- Task: navigate from the initial position to the goal position.

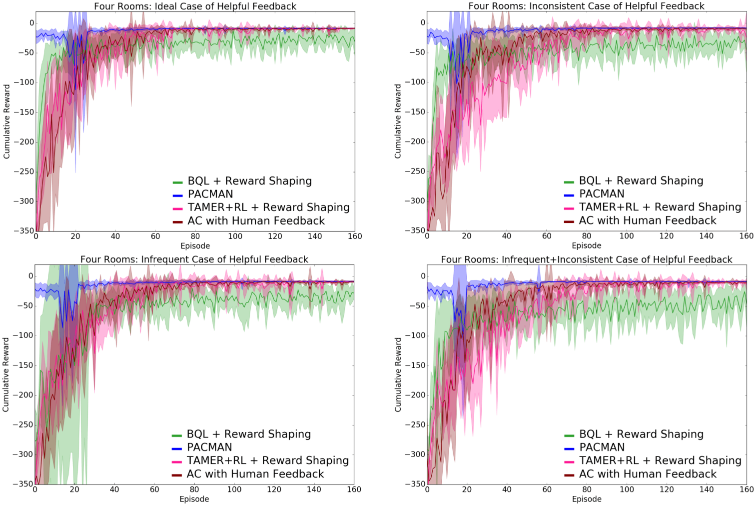


Scenarios on Four Rooms

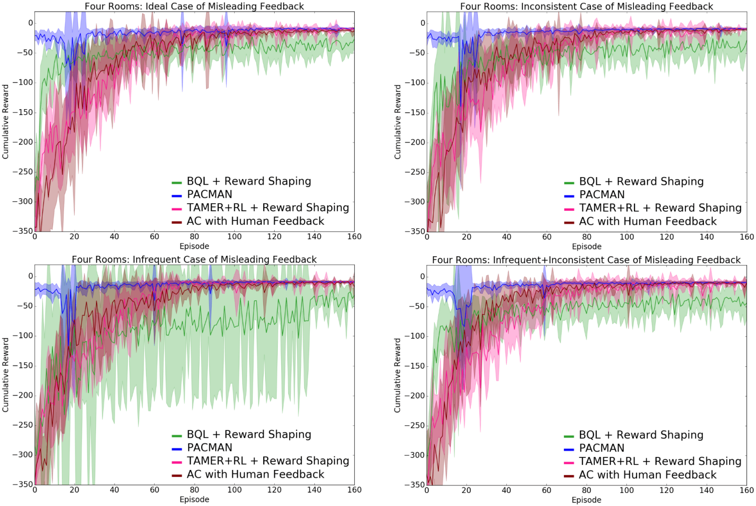


- **Helpful feedback**: consider an experienced user that wants to help the agent to navigate safer and better.
- **Misleading feedback**: consider an inexperienced user who doesn't know there is a dangerous area, but mistakenly wants the agent to step into those red grids.

Results about Helpful Feedback

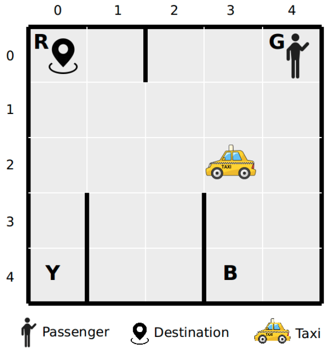


Results about Misleading Feedback

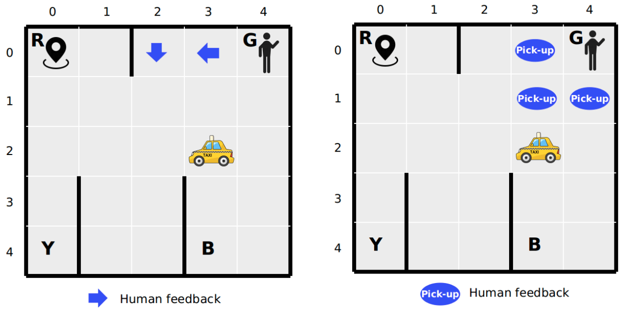


Taxi Domain

- Task: navigate to the passenger, pick up the passenger, then navigate to the destination and drop off the passenger.

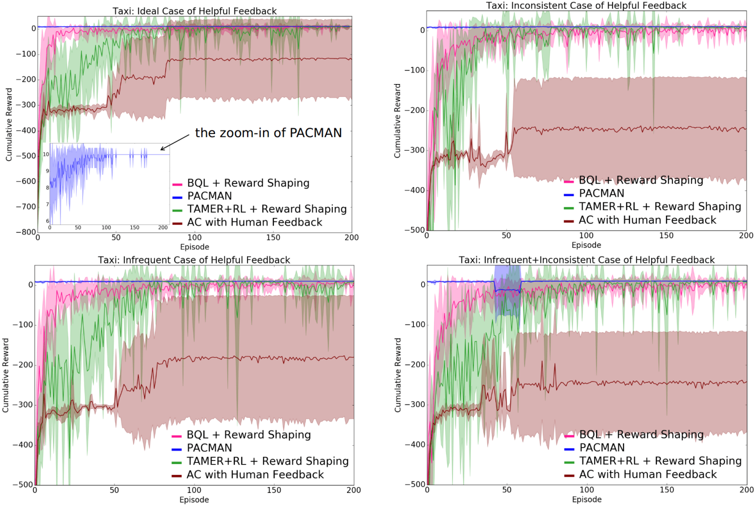


Scenarios on Taxi

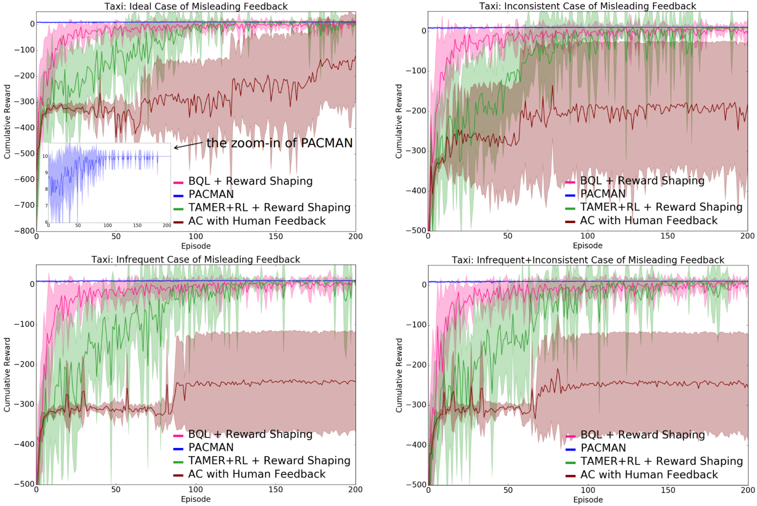


- **Helpful feedback**: consider a passenger may suggest a path that would guide the taxi to detour and avoid the slow traffic during the rush hour.
- **Misleading feedback**: consider a passenger who is not familiar enough with the area and may inaccurately inform the taxi of his location before approaching the passenger.

Results about Helpful Feedback



Results about Misleading Feedback



Conclusion and Future Work

- A unified framework that simultaneously considers prior knowledge, learning from environmental reward and human feedback, which enables "human-centered planning and learning".
  - A significant jump-start at the early stage, which accelerates the learning process.
  - Robustness.
- Future Work.
  - More difficult tasks with high-dimensional sensory input.
  - Autonomous driving or mobile service robots.