AI Planning Annotation in Reinforcement Learning: Options and Beyond

Junkyu Lee, Michael Katz, Don Joven Agravante, Miao Liu,
Tim Klinger, Murray Campbell, Shirin Sohrabi, Gerald Tesauro

IBM research AI

August 5 10:40 AM EST.

PRL Workshop at ICAPS 2021 (August 5-6)
Bridging the Gap Between AI Planning and Reinforcement Learning

Overview

- Introduction
- Background
- Planning annotated Reinforcement Learning (PaRL) Task
- Solving PaRL Task
- Conclusion and Future Work

Introduction - Motivation

AI Planning

- Efficient for finding goal directed sequence of actions
- Scalable through compact symbolic encoding of state transition system
- Plans are natively explainable
- Difficult to obtain symbolic models in general
- Cannot handle perception-oriented high dimensional inputs, e.g. images

Reinforcement Learning

- Easily handle low level control signals, high dimensional inputs
- No need to provide symbolic models
- High cost of maintaining large amount of data
- Most RL methods are suffering from sample inefficiency
- Learned policy is not easy to understand

Integrated Method

- Tasks are handled by AI Planner
- Low level perceptions are handled by RL
- Improve Scalability and Sample Efficiency
- Obtain Easier to understand plans or policy

Background – RL and Options Framework

Markov Decision Process $\mathcal{M} = \langle S, A, P, R, \gamma \rangle$

S : states $\{s_1, s_2, \ldots, s_N\}$

A : actions $\{a_1, a_2, \ldots, a_m\}$

P : probability functions $\{p(s'|s,a) | s, s' \in S, a \in A\}$

R : reward functions $\{r(s,a) | s \in S, a \in A\}$

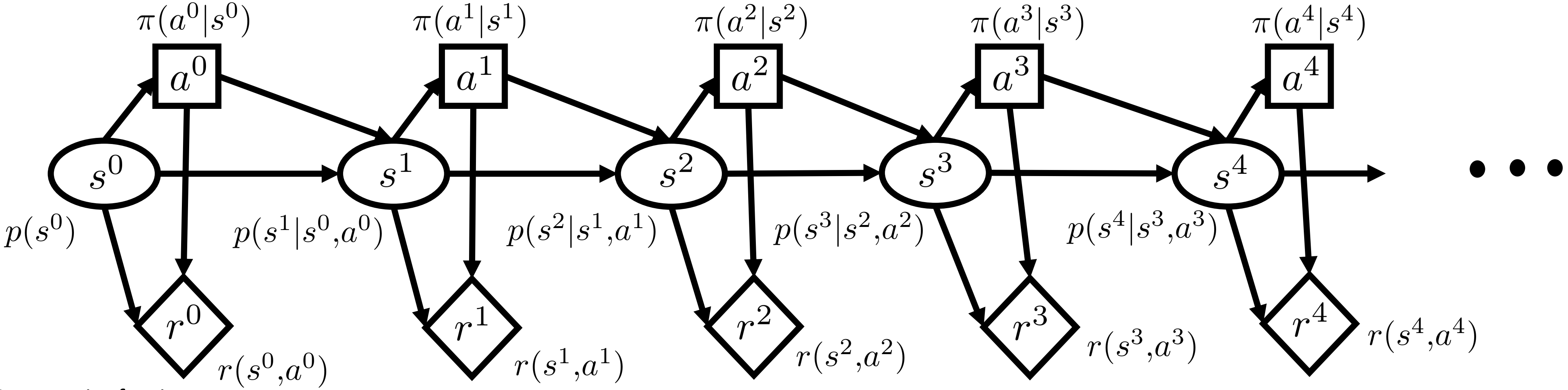
γ : Discounting factor $\gamma \in (0, 1]$

stationary stochastic policy $\pi(a|s) : S \times A \to [0,1]$

$$\text{MEU} = \max_\pi \lim_{k\to\infty} \mathbb{E}_\pi[\textstyle\sum_{t=0}^{t=k} \gamma^t r^t]$$

$$V^\pi(s) = \textstyle\sum_a \pi(a|s)[r(s,a) + \gamma \sum_{s' \in S} p(s'|s,a) V^\pi(s)]$$

$$V^*(s) = \max_\pi V^\pi(s)$$



Super script for time step

Background – RL and Options Framework

Options Framework $\langle \mathcal{M}, O \rangle$ [Sutton, Precup, and Singh 1999]

O : options $\{o_1, o_2, \ldots, o_{|O|}\}$

$o = \langle I_o, \pi_o, \beta(o) \rangle$ I_o : Option Initiation set

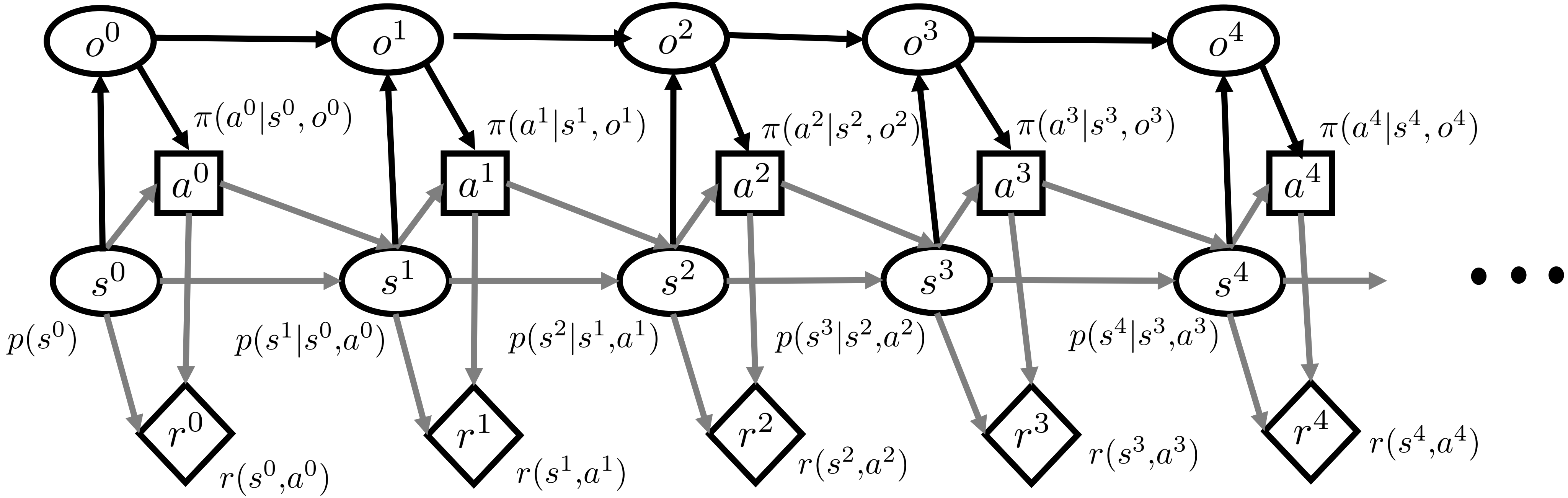
π_o : Intra option policy function

$\beta(o)$: Option termination set

Option level policy $\mu(o'|s,o) : S \times O \times O \to [0,1]$

Intra Markovian option policy $\{\pi_o(a|s,o) | o \in O\}$

Option level MDP is semi-MDP since duration of option execution steps is random variable



Background – RL and Options Framework

Semi-MDP

$o_1 = \arg\max_{o \in O} \bar{\mu}(o|s^0)$

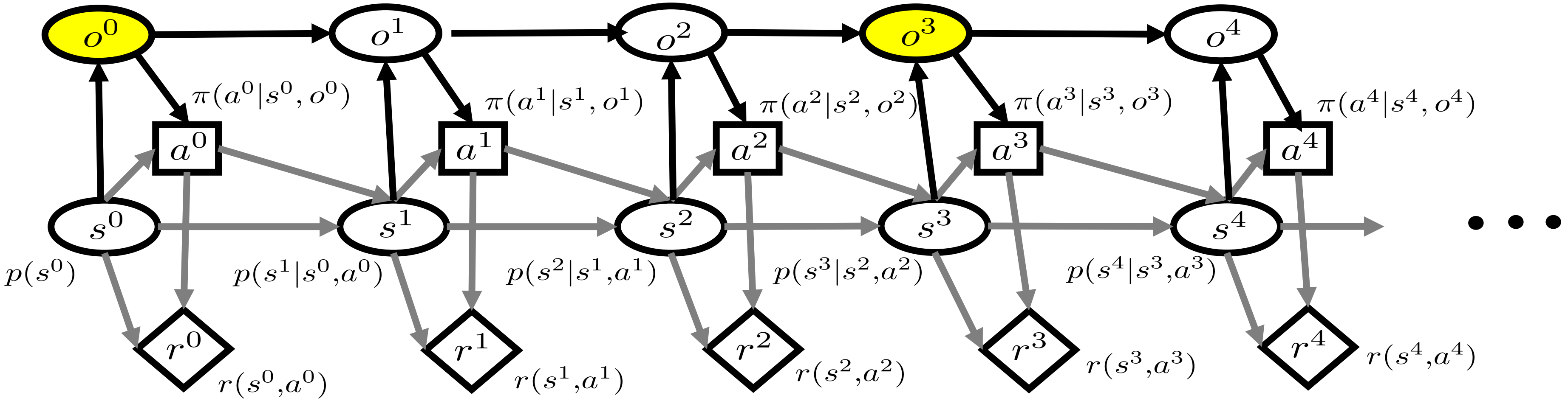
$s^0 \in I_{o_1}$ $\quad$ $s^3 \in \beta_{o_1}$

$\bar{r}(s^0, o_1) = \sum_{t=0}^{t=2} \gamma^t r(s^t, a^t)$

$o_5 = \arg\max_{o \in O} \bar{\mu}(o|o_1, s^3)$

$s^3 \in I_{o_5}$ $\quad$ $s^4 \in \beta_{o_5}$

$\bar{r}(s^3, o_5) = \sum_{t=3}^{t=4} \gamma^{t-3} r(s^t, a^t)$



Value function for SMDP over options: $\bar{V}^{\bar{\mu}}(s) = \sum_{o \in O} \bar{\mu}(o|s)[\bar{r}(s,o) + \gamma \sum_{s' \in S} \bar{p}(s'|s,o)\bar{V}^{\bar{\mu}}(s')]$

Sum of the probability over state transitions over option: $\bar{p}(s'|s,o) = \sum_{j=0}^{\infty} \gamma^j p(s' = s^{t+j}|s = s^t)$

Optimal value over SMDP over options: $\bar{V}^*(s) = \arg\max_{\bar{\mu}} \bar{V}^{\bar{\mu}}(s)$

Background – AI Planning Task

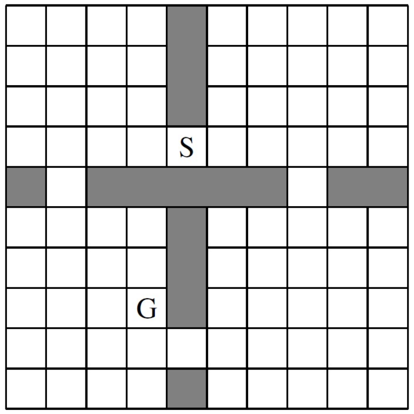
AI Planning Task $\mathcal{T} = \langle V', O', S'_G \rangle$

- V' : variables $\{V_0, V_1, \ldots, V_{|V'|}\}$
- O' : operators $\{O_1, O_2, \ldots, O_{|O'|}\}$
- S'_G : Goal states $S'_G \subseteq S'$
- S' : Planning states

  $$\{(V_0 = v_0, V_1 = v_1, \ldots, V_{|V'|} = v_{|V'|}) | V_i \in V'\}$$

```
(:action move-in-room
    :parameters (?from - location ?to - location ?r - room)
    :precondition (and
        (IN ?from ?r)
        (IN ?to ?r)
        (CONNECTED ?from ?to)
        (in-room ?r)
        (at ?from)
    )
    :effect (and
        (not (at ?from))
        (at ?to)
    )
)

(:action move-out-room
    :parameters (?from - location ?to - location ?r - room ?s - room)
    :precondition (and
        (IN ?from ?r)
        (IN ?to ?s)
        (CONNECTED ?from ?to)
        (CONNECTED-ROOMS ?r ?s)
        (at ?from)
        (in-room ?r)
        (not (at ?to))
        (not (in-room ?s))
    )
    :effect (and
        (not (at ?from))
        (at ?to)
        (not (in-room ?r) )
        (in-room ?s)
    )
)
```



Planning Annotated RL (PaRL)

Planning Annotated RL Task (PaRL) $\langle \mathcal{M}, \mathcal{T}, L \rangle$

$\mathcal{M}$: MDP $\quad$ $\mathcal{T}$: AI Planning Task $\quad$ L : State mapping function $L : S \rightarrow S'$

Given MDP $\mathcal{M} = \langle S, A, P, R, \gamma \rangle$

Option $o = \langle I_o, \pi_o, \beta(o) \rangle$

Initiation set	$I_o : \mathcal{S} \rightarrow \{T, F\}$
Intra-option policy	$\pi_o : S \times A \rightarrow [0, 1]$
Termination set	$\beta_o : S \rightarrow [0, 1]$

Define an option for each operator $Op \in O'$

$$I_{Op} = \{s \in S | \text{precondition}(Op) \subseteq L(s)\}$$

$$\beta_{Op} = \begin{cases} T & \text{if prevail } (Op) \cup \text{ effect } (Op) \subseteq L(s) \\ F & \text{o.w.} \end{cases}$$

AI Planning Task operator

(Move from r5 to c-r5-r3)

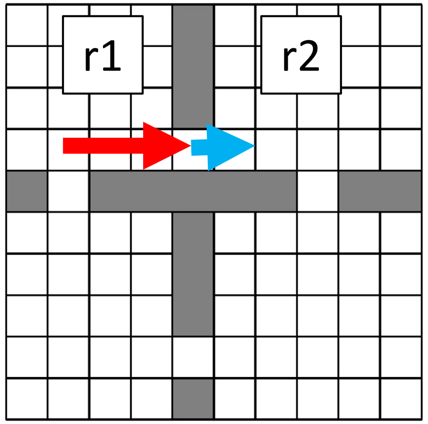
(precondition): in-room(r5)

(effect): in-room(c-r5-r3)

Option for MDP task

$$I_O = \{s \in \mathcal{S} | \textit{in-room(r5:room)} \subseteq L(s)\}$$
$$\beta_O = \{s \in \mathcal{S} | \textit{in-room(c-r5-r3:room)} \subseteq L(s)\}$$

Solving PaRL



$o_1 :=$ (Move from r1 to c-r1-r2)

$s^0 \in I_{o_1}$ $\quad$ $s^3 \in \beta_{o_1}$

$\bar{r}(s^0, o_1) = \sum_{t=0}^{t=3} \gamma^t r(s^t, a^t)$

$o_5 :=$ (Move from c-r1-r2 to r2)

$s^3 \in I_{o_5}$ $\quad$ $s^4 \in \beta_{o_5}$

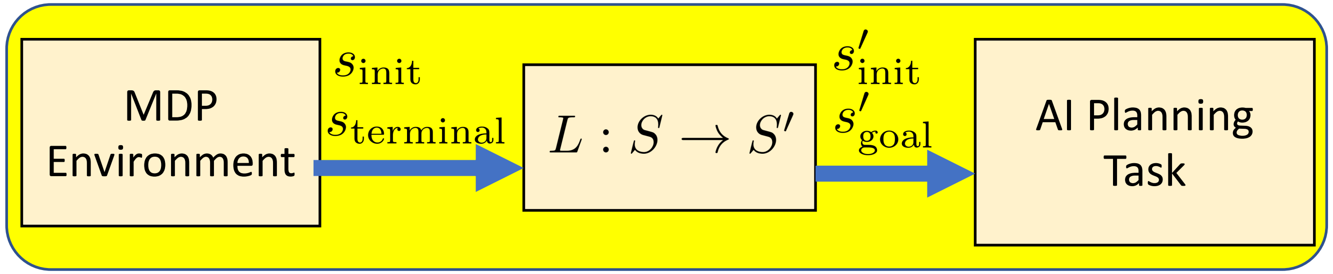
$\bar{r}(s^3, o_5) = \sum_{t=3}^{t=4} \gamma^{t-3} r(s^t, a^t)$

- PaRL provides "side information" to the RL agent
  - Can be viewed as a model-based hierarchical RL approach
  - AI planner generates high-level plans at the level of options
    - Offline Planning: option sequence is generated before learning intra option policy functions
    - Online Planning: option sequence is generated while learning policy functions

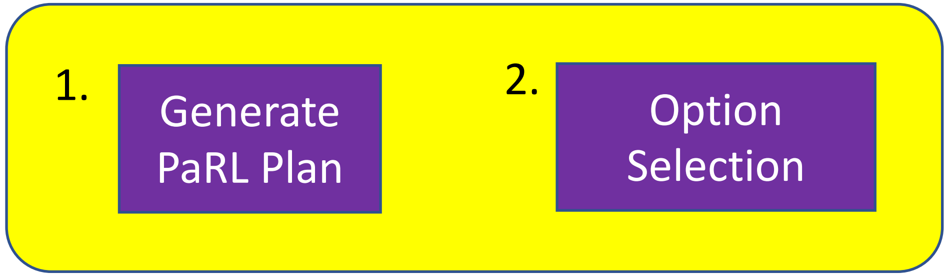
Solving PaRL – Offline Options Training+ SMDP Learning

- Problem: there are many options available to train/use.
- We want to use only “useful” options for solving a problem with a fixed initial state and terminal state.

Planning Annotated RL Task (PaRL)



<u>Option Selection by Offline planning</u>

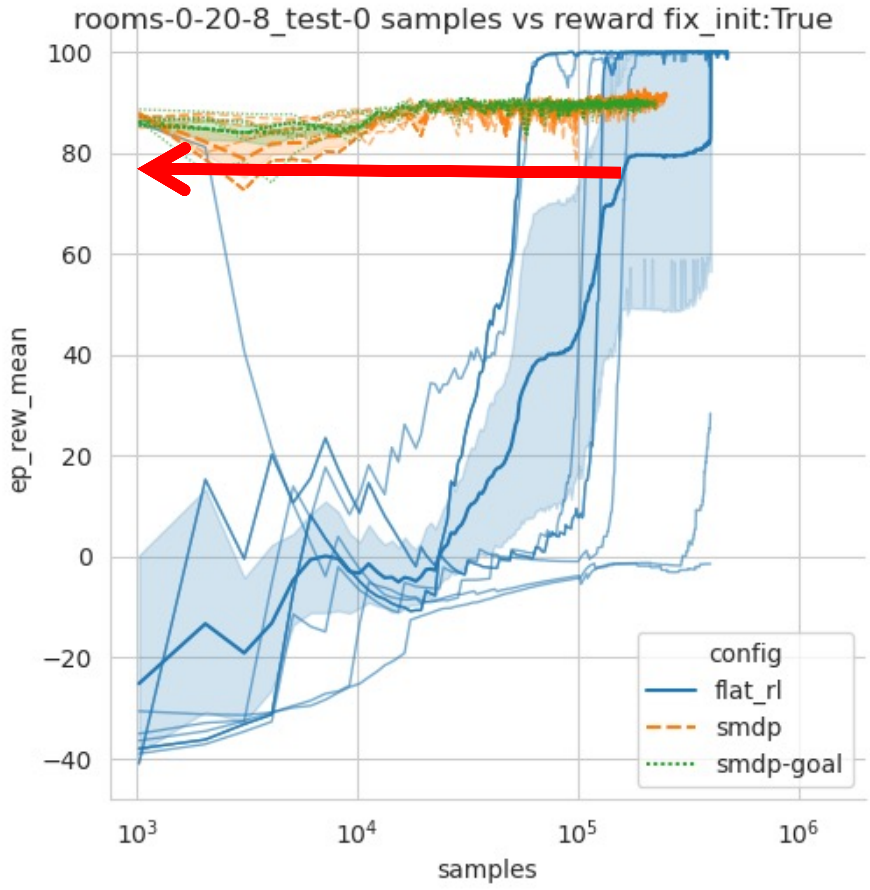


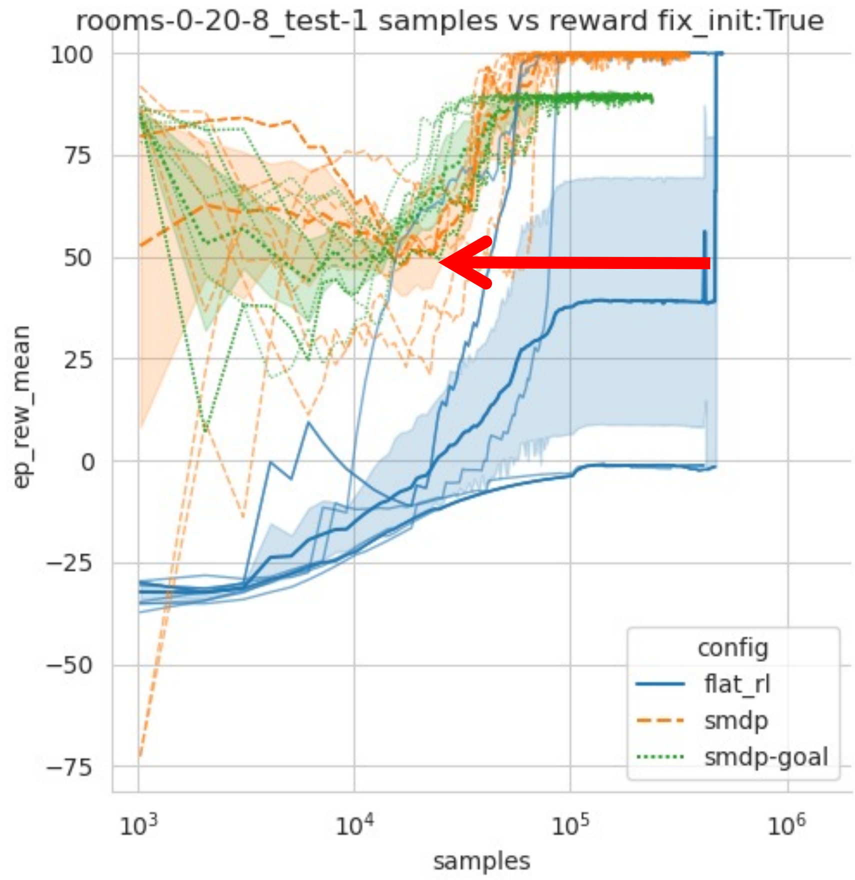
1. AI planner generates plans over options
2. Select options
- Rank options with some score functions such as frequency

- Intra option training: train only selected options by any RL algorithm
- SMDP learning: train option level policy function over the selected options and primitive actions

Solving PaRL – Experiments

- SMDP Learning + Proximal Policy Optimization using pretrained options [Sutton, Precup, and Singh 1999] [Schulman, et. al 2017]
  - Intra-option policy training: PPO with A
  - Option level policy training: PPO with $A \cup O$





Related Works

- Hierarchical RL [Kulkarni, et. al 2016]
  - Define master/slave architecture and master policy generates subgoals for each slave
  - Agent policy at the lower level is similar to the options

- Option Critic [Bacon and Precup 2017]
  - End-to-End approach for training intra option and option level policy functions
  - Learning algorithm: policy gradients derived for option value function

- PEORL/SDRL [Yang, et. al 2018] [Lyu, et. al 2019]
  - Derive a Planning task from BC action language
  - Define 1 option per state transition in planning problem
  - Learning algorithm: R-max learning

- Taskable RL [Illanes, et. al 2020]
  - Derive a planning task from subtasks in RL problem
  - Manually define termination set of options from planning operators
  - Learning algorithm: SMDP-Q learning + Q-learning for intra option training

Conclusion

- Planning annotated RL
  - Annotate an RL task with a planning task and derive hierarchical RL architecture
  - Generate option specifications from planning operators
  - Option level policy learning can utilize AI planning algorithms
- Solving PaRL task
  - Offline approach: utilize AI planner for selecting useful options for RL task
- Future Work
  - Online approach: interleave option selection and intra-option training
  - Learning AI planning task from RL environment