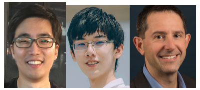
Learning to Correct Mistakes: Backjumping in Long-Horizon Task and Motion Planning



Yoonchang Sung[1]*, Zizhao Wang[1]*, and Peter Stone[1,2]

*Equal contribution

[1]The University of Texas at Austin [2]Sony AI

Challenges in long-horizon planning:

- Long-horizon search in TAMP problems is intractable due to a large depth and branching factor
- Early actions may make future actions infeasible, leading to many backtracking steps

Research question:

- **How can we identify a culprit variable to improve planning efficiency?**

Two learning models with two sampling methods:

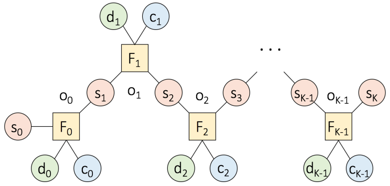
- Imitation learning: directly predicting a culprit variable by leveraging access to the true culprit in training data
- Plan feasibility prediction: counterfactual approach for binary prediction
- Sampling methods: batch sampling and forgetting

Contributions:

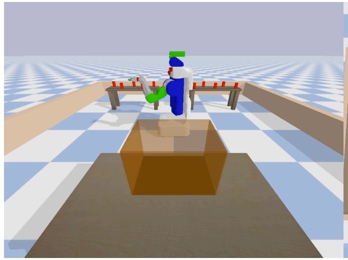
- We exploit long-horizon dependency in TAMP and propose to learn a backjumping policy for planning efficiency
- Our models empirically outperform baselines (e.g., backtracking) in two representative domains

Dirty laundry:

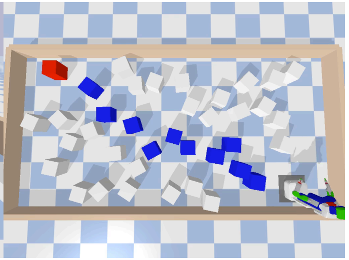
- Generalization is tested in similar tasks only: what is good representation?
- Several assumptions (perfect action and observation), but other work exists addressing them
- A really long horizon of hundred of actions has yet to be evaluated



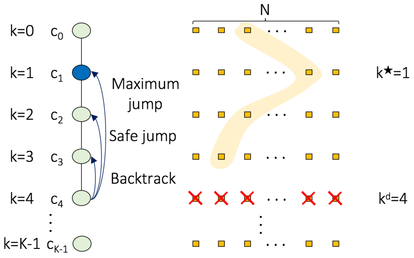
[TAMP as constraint satisfaction]



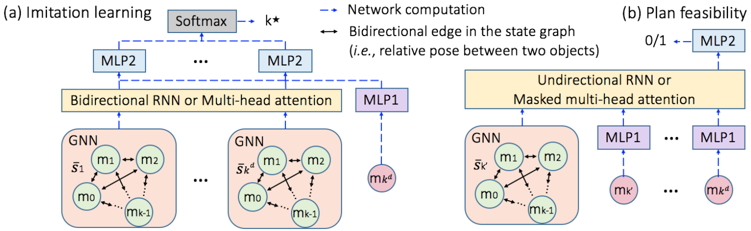
[Pick-and-place task]



[NAMO task]



[Search tree example]



[Proposed learning models]

Task	Backtracking	IL RNN	IL Attn	PF RNN	PF Attn
Packing	4414 ± 879	**2464 ± 464**	2638 ± 602	**2205 ± 313**	**2062 ± 297***
NAMO	(21 ± 10) ×10^4	**543 ± 187**	**425 ± 153***	**529 ± 188**	2614.7 ± 709.5
Packing (11)	12098 ± 2518	**5350 ± 1094***	**7044 ± 1481**	**6142 ± 767**	**7109 ± 809**
Packing (12)	34719 ± 6514	**15139 ± 3080***	**16339 ± 3971**	22377 ± 3244	31824 ± 3925
Packing (**BS**)	13541 ± 4205	**4464 ± 1160**	7073 ± 2040	**4556 ± 749**	**4311 ± 690***

[Main result: the number of nodes visited in the search tree obtained by solving 100 problems]