Imitation Learning with Temporal Logic Constraints

Zining Fan, He Zhu

Problem

Linear Temporal Logic

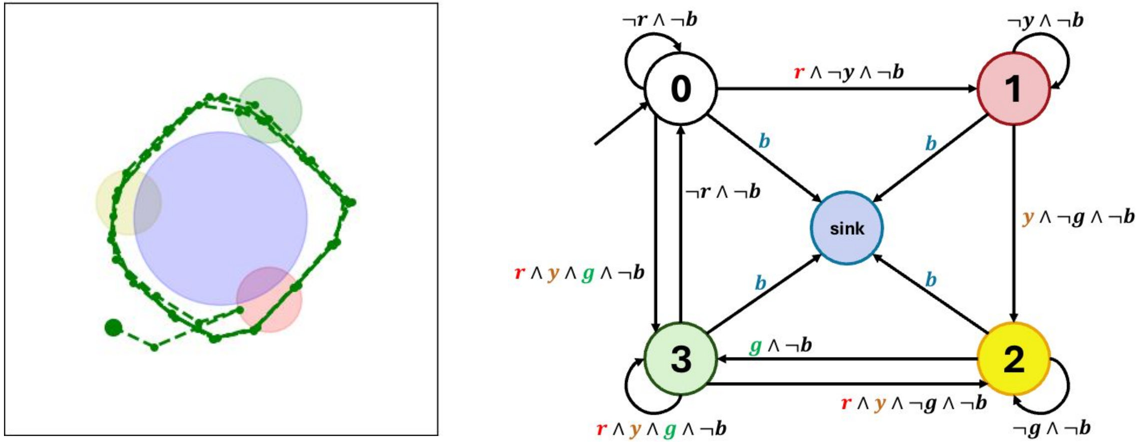
- LTL provides a flexible language for specifying temporally dependent tasks, such as alternating between subgoals while staying safe.

RL under LTL

- A trajectory satisfies an LTL formula if and only if it visits an Limit Deterministic Büchi Automaton (LDBA)-accepting state infinitely often.
- Agents only get **sparse rewards** when reaching accepting states
- **Ineffective exploration** toward such states over infinite horizons

Example

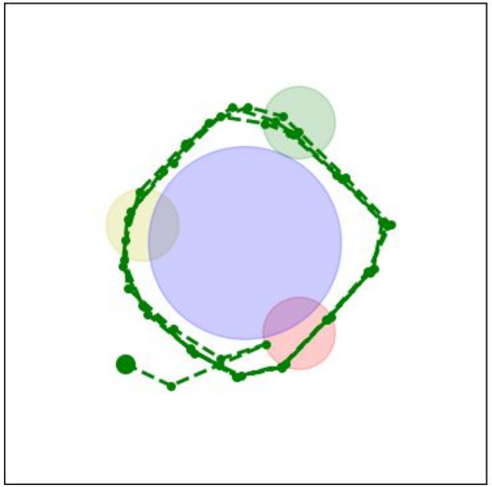
- Oscillating infinitely between the yellow, green, and red zones while avoiding the blue zone
- $\varphi = GF(y \wedge XF(g \wedge XFr)) \wedge G\neg b$



Left:FlatWorld Cycle environment with LTL spec φ.
Right: Limit Deterministic Büchi Automaton (LDBA) for φ accepts paths reaching state 3 infinitely.

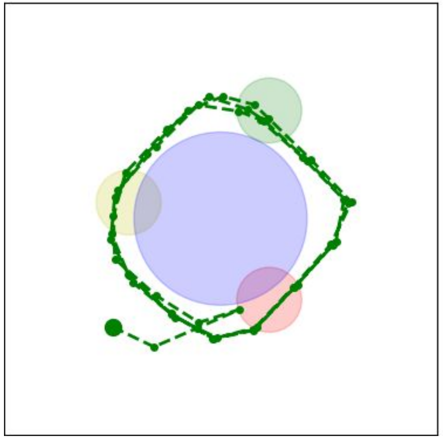
Goal

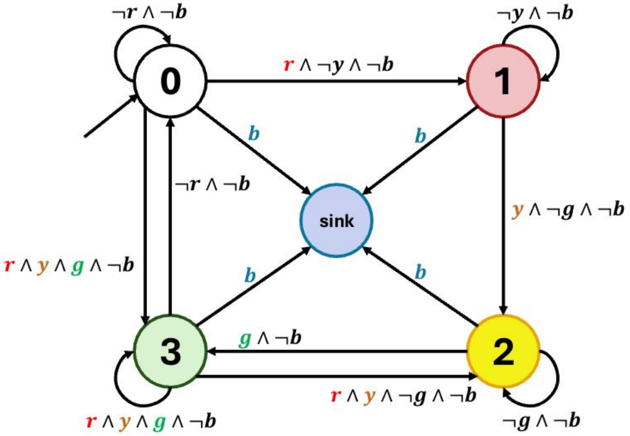
- Use suboptimal demonstrations visiting accepting states once or twice
- Leverage them to mitigate reward sparsity in LTL-based policy optimization

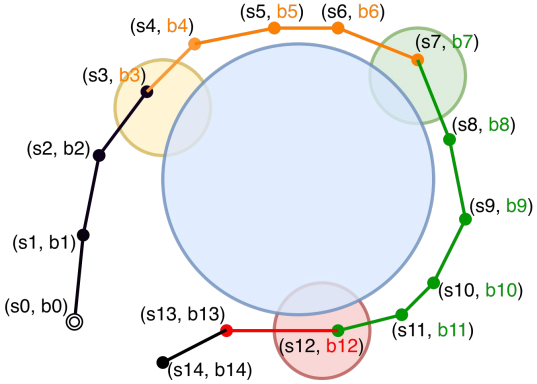


Product MDP

- Product MDP merges the environment’s states s with the automaton state b from an LTL formula.







Temporal Logic Imitation Learning

$$\pi^* = \arg\max_{\pi_\phi \in \Pi} \left(P(\pi_\phi \models \varphi), J_\pi(\phi)\right)$$

probabilistic satisfaction of an LTL formula φ

imitation learning objective Jπ

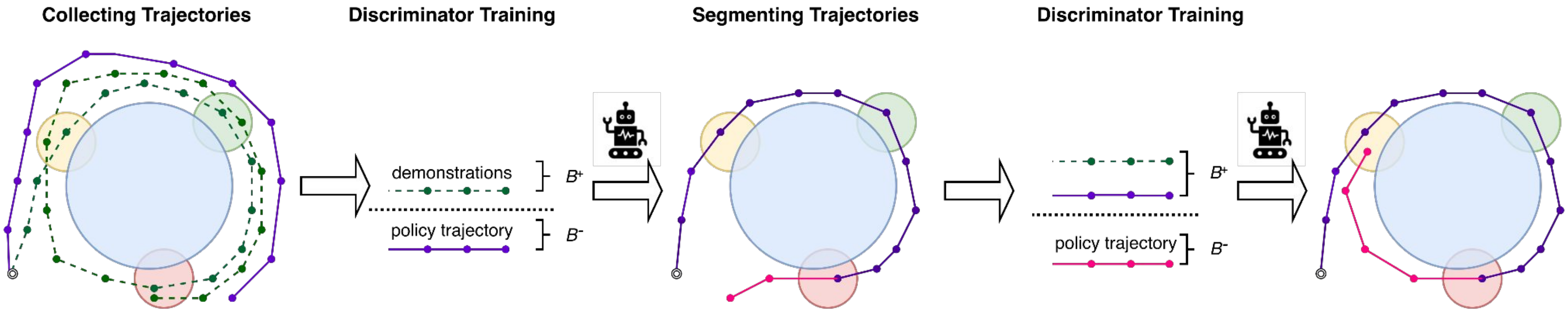
$$J_\pi(\phi) = \mathbb{E}_{\tau \sim \pi_\phi}\left[\sum_{t=0}^{\infty} \gamma^t R((s,b))\right]$$

$$R((s,b)) = \tanh(f_\psi(s))$$

a GAIL-style discriminator fψ

Segmented Imitation

Segment policy trajectories at accepting states; train each segment to reach the next accepting state.



NEURAL INFORMATION PROCESSING SYSTEMS

Segmented Imitation

- Q Update

$$J_Q(\theta) = \mathbb{E}_{((s,b),a,(s',b'))\sim B}\left[\frac{1}{2}\left(Q_\theta((s,b),a) - \hat{y}\right)^2\right]$$

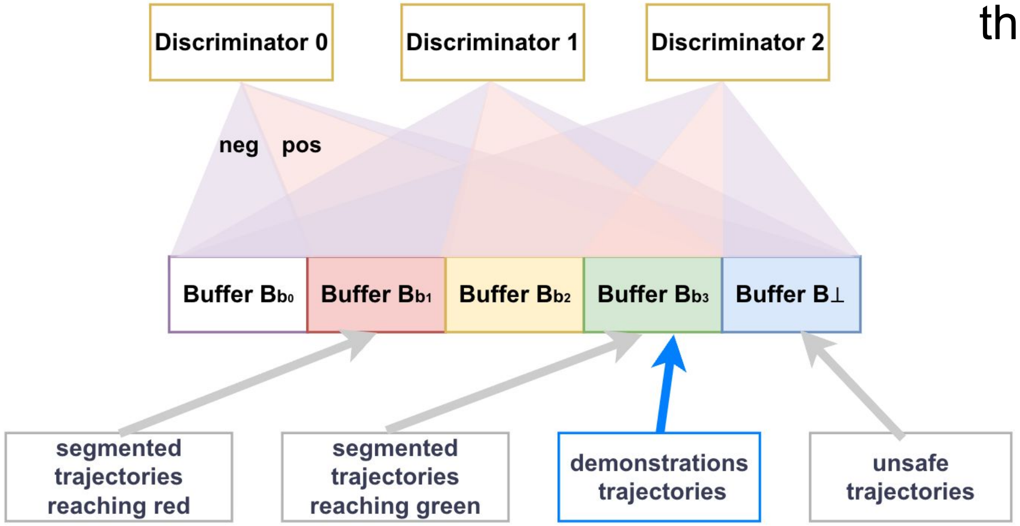
$$\hat{y} = \begin{cases} 1/(1-\gamma), & b \in \mathcal{B}^\star \\ R((s,b)) + \gamma \mathbb{E}_{a'\sim\pi_\phi(\cdot|(s',b'))}\left[Q_{\mathtt{targ}}\right], & b \notin \mathcal{B}^\star \end{cases}$$

- Policy Update

$$J_\pi(\phi) = \mathbb{E}_{(s,b)\sim B}\left[\mathbb{E}_{a\sim\pi_\phi(\cdot|(s,b))}\left[\alpha \log \pi_\phi(a|(s,b)) - Q_\theta((s,b),a)\right]\right]$$

Multi-Stage Discriminator Learning

Stage-specific reward shaping: One GAIL-style discriminator is assigned to each automaton state, distinguishing trajectories that progress beyond the current state from those that fail to advance beyond it.



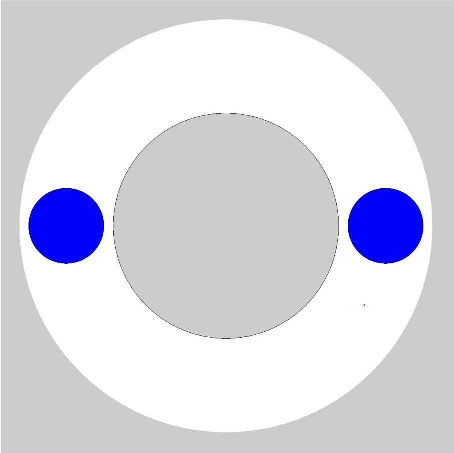
Length of the longest acyclic path in the LDBA from the initial state to b

discriminator for state b

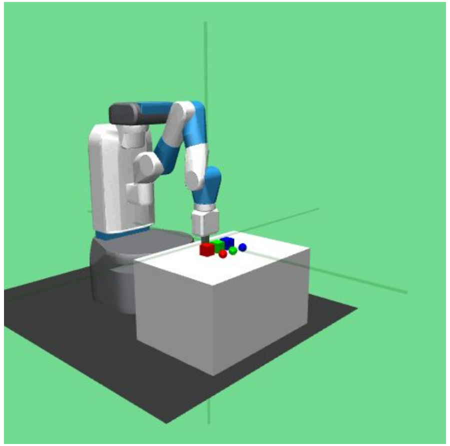
$$R((s,b)) = \frac{\mathrm{SIDX}(b) + \beta \cdot \tanh(f_b(s))}{\mathcal{N}(b)}$$

length of the longest acyclic path in the LDBA from the initial state to any accepting state through b

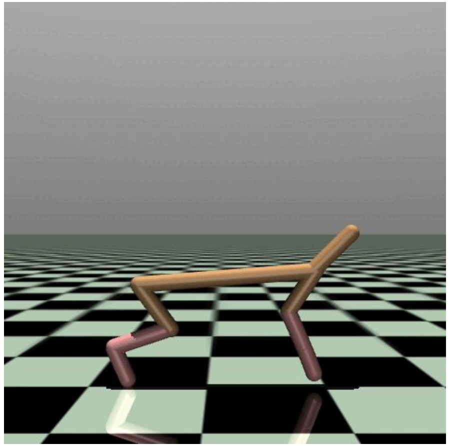
Experiments



$GF(b0 \land XF(b1)) \land G\neg crash$



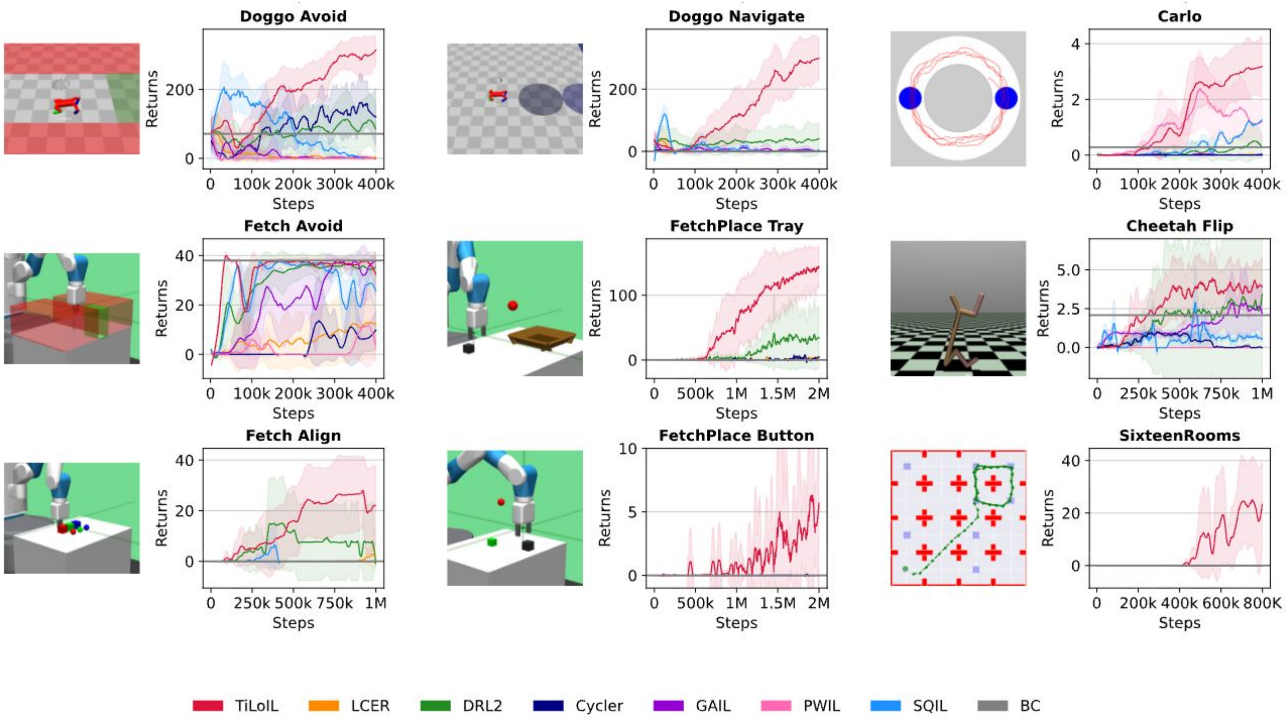
$F(a \land XF(b \land XFc))$



$GF(b \land XFd)$

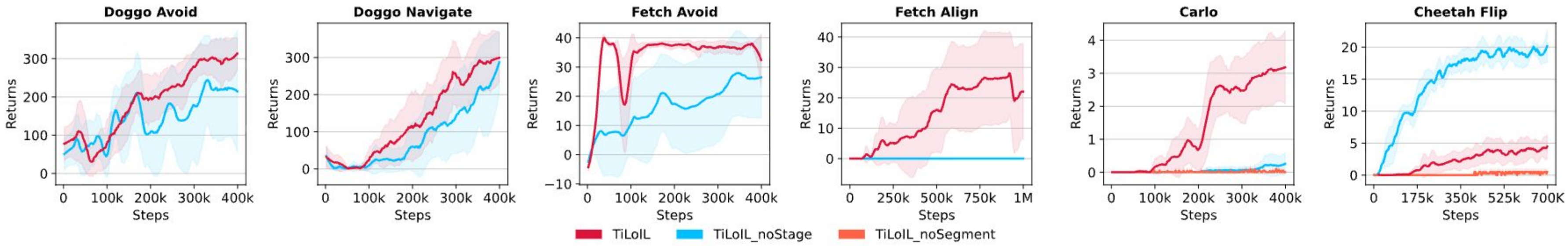
Results

Does TiLoIL help the learning process for LTL-constrained tasks?

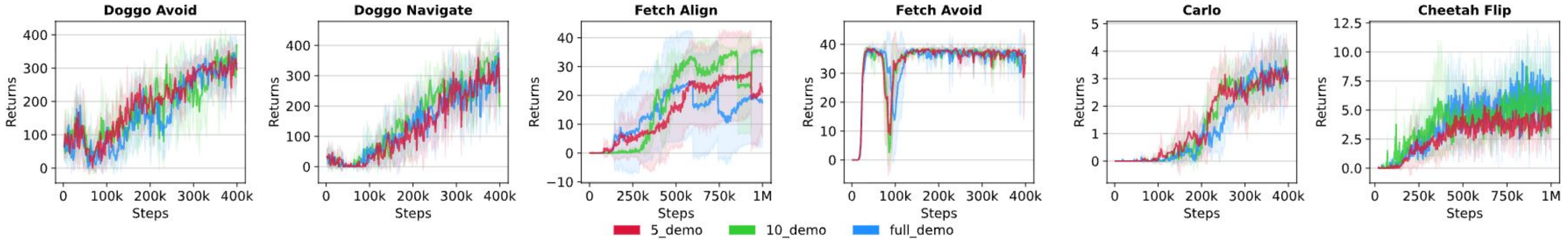


Results

Are segmented imitation and multistage discriminator learning in TiLoIL necessary?



Do we require lots of demonstrations?



Thank you for listening!