Supplemental Material
RePReL : Integrating Relational Planning and Reinforcement Learning for Effective Abstraction

Harsha Kokel,[1] **Arjun Manoharan,**[2] **Sriraam Natarajan,**[1]
Balaraman Ravindran,[2] **Prasad Tadepalli** [3]

[1]The University of Texas at Dallas, [2]Robert Bosch Centre for Data Science and Artificial Intelligence at Indian Institute of Technology Madras, [3]Oregon State University
hkokel@utdallas.edu, arjunman@cse.iitm.ac.in, Sriraam.Natarajan@utdallas.edu,
ravi@cse.iitm.ac.in, tadepall@eecs.oregonstate.edu

1 Appendix

1.1 Environments

This section provides further detail for each of the environment used for empirical evaluations. The code should be available at https://starling.utdallas.edu/papers/RePReL.

Craft World Figure 1 presents the map of the **Craft World** used in the empirical evaluations. It is a 11×11 grid with 8 points of interest shown in the figure. The environment provides a reward of -1 for every step and 100 for goal state. We use terminal reward $t_R = 100$ for each subtask.

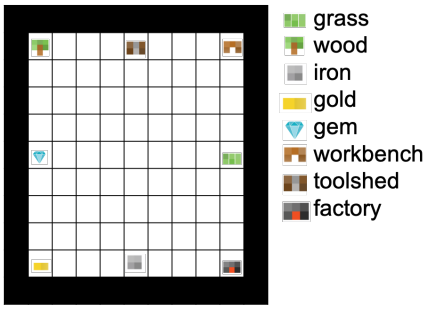


Figure 1: Map of the **Craft World** indicating eight locations: grass, wood, iron, gold, gem, workbench, toolshed, and factory. Black cells in the grid represent walls.

Office World The map of the **Office World** is reproduced from Illanes et al. (2020) in Fig. 2 for convenience. Plants are indicated by '*' in the map, these locations are inaccessible to the agent. Walls in the map are indicated by bold black line. Any step towards wall or inaccessible locations will keep the agent in the same location. Accessible locations include two coffee-rooms, one office desk, and one mail-room. Coffee-room locations are indicated by blue coffee mugs, mail-room is indicated by green envelope, and office desk is highlighted with a hand. Locations A, B, C, and D are also accessible locations, they marked in the map. This environment has reward of -1 for every step, 100 for goal state, and -10 for attempt of going to inaccessible locations. Terminal reward $t_R = 100$.

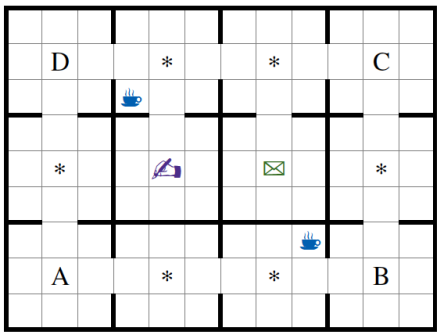


Figure 2: Map of the **Office World** from Illanes et al. (2020)

Extended Taxi World The 8×8 grid of the extended taxi world domain is shown in Fig. 3. Each of the three passenger can have R, G, B, or Y as their pickup and drop location. Hence, there are 36 possible combinations of passenger pickup and drop locations. Task is to drop passenger in sequence. Like office world, every step has reward -1, goal state has reward 100, and step in invalid direction has reward -10. We used 100 for terminal reward.

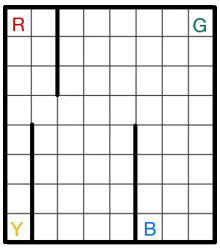
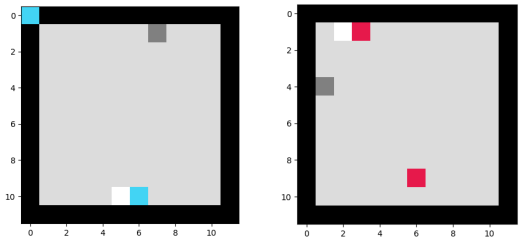


Figure 3: Grid of the **Extended Taxi World** indicating R, G, B, and Y locations. Bold black lines indicate walls.

Relational Box World Figure 4 represents the three tasks evaluated in the **Relational Box World**. The gem is represented with white color in the grid, walls with black, and player with gray. Pair of colored cell represent a box, the cell on right is a lock and cell on left is either a key or gem. The owned key is represented on the top right corner of the grid. Locks and keys are sampled from 18 different colors. Location of the boxes are also sampled in the beginning of each episode. Task 1 has a lock containing the gem, the player is initialized with the key to open the lock. In Task 2 player has to first collect the free key and then open the lock to collect the gem. Task 3 requires the player to collect the key and open two locks in sequence to reach the gem. The environment provides reward -0.1 for every step, -0.2 for invalid step, and 1 for collecting key or gem. We use terminal reward $t_R = 1$.



(a) Task 1 (b) Task 2

(c) Task 3

Figure 4: Tasks in the **Relational Box World**.

1.2 Propositional Abstraction

The planner in RePReL works with the relational representation of the domain, however the reinforcement learner operates at a propositional level. The advantage of RePReL framework is enabled by the D-FOCI statements that provides abstraction in the propositional state representation. D-FOCI statements can be viewed as a relational version of the Dynamic Bayesian Networks (DBN). In factored MDP represented using a DBN the relevant state literals are identified by iteratively collecting all the literals that influence the relevant literals starting from the reward variable. In RePReL this iterative process of collecting state literals additionally involve the step of grounding.

Given the set of D-FOCI statements for the domain $\mathcal{F}$, current state s, and the grounded operator o_g for the taxi-domain example, the process of obtaining the abstract state is shown in Table 1. The relevant state literals $\hat{s}$ is then presented as flat feature vector to the Reinforcement Learner.

Given:
a. D-FOCI statements $\mathcal{F}$
{taxi-at(L1), move(Dir)} $\xrightarrow{+1}$ taxi-at(L2)
{taxi-at(L1), move(Dir)} $\longrightarrow R$
pickup(P): {taxi-at(L1), at(P, L), in-taxi(P)} $\xrightarrow{+1}$ in-taxi(P)
pickup(P): in-taxi(P) $\longrightarrow R_o$
drop(P): {taxi-at(L1), in-taxi(P), dest(P,L), at-dest(P)} $\xrightarrow{+1}$ at-dest(P)
drop(P): at-dest(P) $\longrightarrow R_o$

b. state s
{ at(p1,r), taxi-at(l3), dest(p1,d1),,
¬at-dest(p1) ¬in-taxi(p1), at(p2,b), ¬at-dest(p2),
¬in-taxi(p2)}

c. grounded operator o_g
o : pickup(P),
θ : {P/p1, L/r}

Depth 1 unrolling:
1. Ground applicable D-FOCI statements that have reward on RHS
pickup(p1): in-taxi(p1) $\longrightarrow R_o$
{taxi-at(l3), move(d)} $\longrightarrow R$
$\theta \leftarrow$ {P/p1,L/r,L1/l3,Dir/d}
2. Collect LHS in relevant state literals
$\hat{s} \leftarrow$ {in-taxi(p1), taxi-at(l3), move(d)}

Depth 2 unrolling:
1. Ground applicable D-FOCI statements that have a relevant literal ($\hat{s}$) on RHS
pickup(p1): {taxi-at(l3), at(p1, r), in-taxi(p1)} $\xrightarrow{+1}$ in-taxi(P)
{taxi-at(l3), move(d)} $\xrightarrow{+1}$ taxi-at(l3)
$\theta \leftarrow$ {P/p1,L/r,L1/l3,Dir/d}
2. Collect LHS in relevant state literals
$\hat{s} \leftarrow$ {in-taxi(p1), taxi-at(l3), move(d), at(p1, r)}

Table 1: Illustrative example of 2-depth unrolling of the D-FOCI statements in taxi-domain.