

Practical Sheet 8

Automated Planning

1. Analyze the provided Planning Domain Definition Language (PDDL) implementation of the gripper problem and the respective Python planner script. Identify and describe the predicates, actions, and problem specification. Run the planner script with a different engine (`fast-downward`) and compare the generated plans.

Note: Install the following Python packages for the planner:

```
pip install unified-planning up-pyperplan up-fast-downward
```

2. Modify the gripper problem to handle a more complex scenario:
 - Create a new problem file with three rooms (`roomA`, `roomB`, `roomC`) instead of two. Place six balls initially distributed across the rooms: three balls in `roomA`, two balls in `roomB`, and one ball in `roomC`.
 - Define a goal state where specific balls must be moved to different target rooms. For example, balls from `roomA` should go to `roomB`, balls from `roomB` should go to `roomC`, and the ball from `roomC` should go to `roomA`.
3. Extend the domain to support multiple robots operating in the environment:
 - Replace the predicate (`at-robot ?x`) with (`at-robot ?r ?x`) where `?r` identifies a specific robot and `?x` is its location. Add a predicate (`ROBOT ?r`) to identify robot entities. Moreover, modify all existing actions (`pickup`, `drop`, `move`) to include a robot parameter and update their preconditions and effects accordingly.
 - Create a problem with two robots (`robot1` and `robot2`), each with two grippers and four balls in room A that must be transported to room B.