BORDEAUX INP Enseirb-Matmeca

PDDL spaceship adventure

EI9IS315 Knowledge representation

S9 - Year 2024-2025

Author: Mohamed Kherraz

February 2, 2025

Contents

0.1 Introduction 2
0.2 Problem Formulation 2
0.3 Results 3
0.4 Conclusion 3

0.1 Introduction

In the field of symbolic AI automated planning is a well known problem. Therefore we will test the state of art knowledge representation method to formularize the problem of a spaceship trying to craft a jet booster by collecting parts and transporting them to the central space station while managing its fuel. PDDL (Planning Domain Definition Language) is the adequate tool for solving such optimisation problem thanks to its modular nature and simplicity of defining actions, constraints, and goals. In this report we will test the performance of PDDL on the spaceship optimization problem.

0.2 Problem Formulation

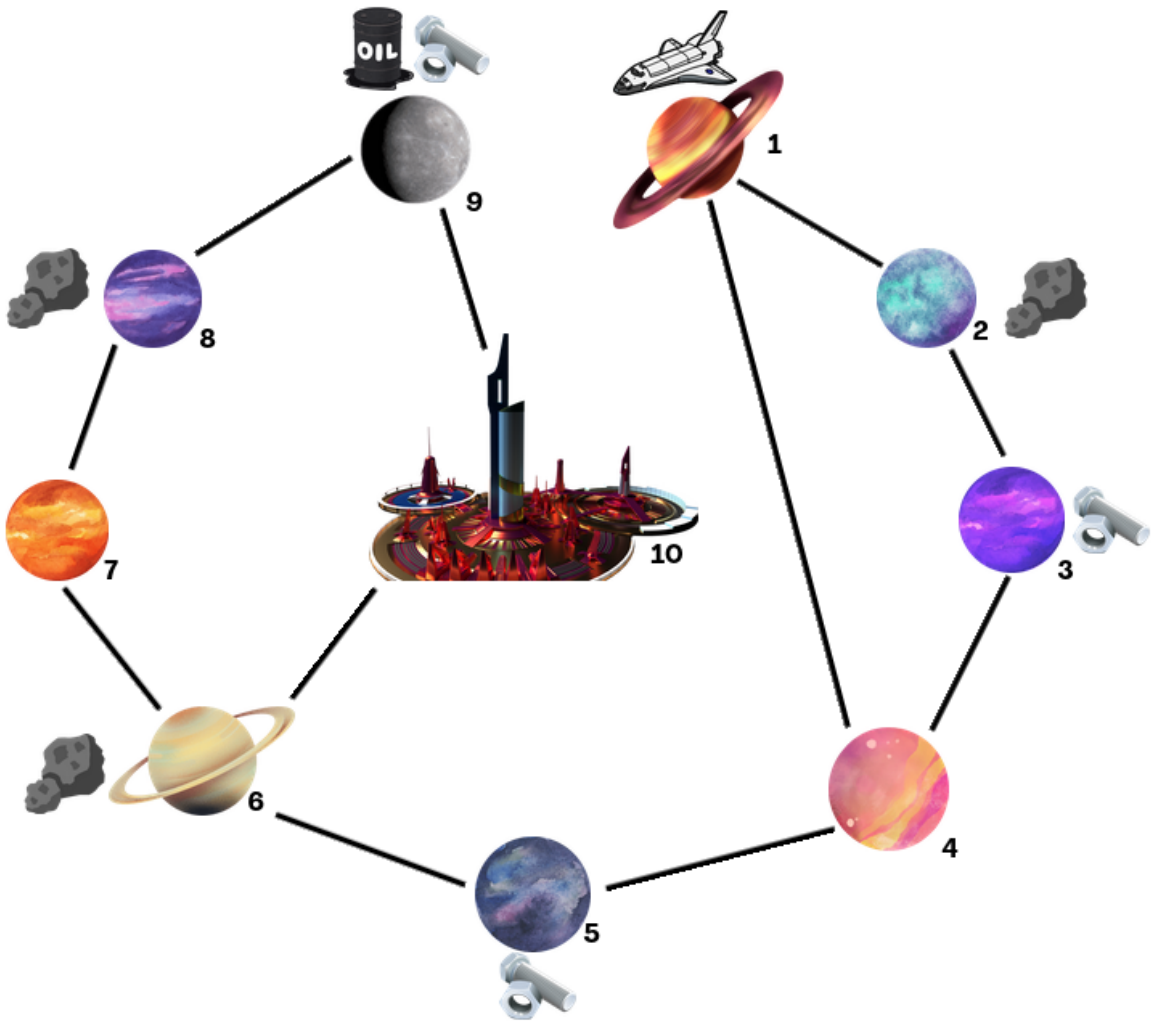

Figure 1: Representation of the example universe.

The spaceship must navigate in the defined universe of 10 planets, pick up parts one at a time, and drop them off at the central space station, as depicted by the figure 1. Once the three parts are collect in the central space station at planet 10 the jet booster can be crafted. However, once a part is picked up, it is extremely heavy that by the time the part is fully loaded on the spaceship, the entire fuel reserve is used. Therefore, the spaceship has to refuel after picking up the part using the fuel stations. If no fuel is

available the spaceship can use collected resources to build a fuel station. This problem can be summarized in the following constraints:

- **Fuel Management**: Refuel at planets with refuel stations.
- **Resource Collection**: Mine resources in order to craft fuel station at any planet.
- **Crafting**: Collect parts to craft the jet booster.
- **Navigation**: Traveling is only allowed between planets linked by wormholes.

In our PDDL action space we used the following actions:

- **travel**: Move the spaceship between connected planets.
- **pick-up**: Collect an item from a planet.
- **drop-off**: Drop an item at a planet.
- **craft-jet-booster**: Craft a jet booster using collected parts.
- **refuel**: Refuel the spaceship at a planet with a fuel station.
- **mine-resources**: Mine resources from a planet.
- **build-fuel-station**: Build a fuel station on a planet using carried resources.

0.3 Results

To test how well our PDDL representation of the problem is we tested different scenarios and tried to find solutions for them.

The first scenario is the one with with the example universe in figure 1, where the PDDL solver always gives the optimal sequence in algorithm ?? in 6.13 ± 0.1 seconds. The PDDL domain and problem definition are only meant to make the problem understandable to the computer so the proof of the optimality and the performance of the solution is highly bound to the solver used.

The second scenario we tested was the one where the resource in planet 2 is removed forcing the spaceship to retrieve the resources from planet 6 or 8. The PDDL solver used the resource in planet 6 for the part in planet 3 and the resource in planet 8 for the part in planet 5 in 1.13 ± 0.7 seconds.

Other extreme case scenarios were tested such as impossible solutions when one more resource is removed which the PDDL solver responded to with a "No plan was found" message prooving that our problem definition does indeed work as entended.

0.4 Conclusion

This experiment allowed us to conclude that PDDL is a good framework which allowed us to solve our resource and fuel management problem and solved it as an example of complex optimization problems. The results showed that PDDL can handle a wide range of scenarios, from resource scarcity to fuel management challenges, while the solver could always come up with optimal or near-optimal plans within a reasonable time.

Algorithm 1: Sequence of action for solving the universe in figure 1

travel spaceship1 planet1 planet2
mine-resources spaceship1 planet2 resource1
travel spaceship1 planet2 planet3
pick-up spaceship1 part1 planet3
build-fuel-station spaceship1 planet3 resource1
refuel spaceship1 planet3
travel spaceship1 planet3 planet4
travel spaceship1 planet4 planet5
travel spaceship1 planet5 planet6
travel spaceship1 planet6 planet10
drop-off spaceship1 part1 planet10
travel spaceship1 planet10 planet6
mine-resources spaceship1 planet6 resource2
travel spaceship1 planet6 planet5
pick-up spaceship1 part2 planet5
build-fuel-station spaceship1 planet5 resource2
refuel spaceship1 planet5
travel spaceship1 planet5 planet6
travel spaceship1 planet6 planet10
drop-off spaceship1 part2 planet10
travel spaceship1 planet10 planet9
pick-up spaceship1 part3 planet9
refuel spaceship1 planet9
travel spaceship1 planet9 planet10
drop-off spaceship1 part3 planet10
craft-jet-booster spaceship1 planet10