

National University of Singapore  
School of Computing

Semester 1, AY2023-24

CS4246/CS5446

AI Planning and Decision Making

---

Due: 23 Oct 2023@09:00

## Assignment 2

### Instructions

- This assignment requires 2 submissions.
  - Submit the PDF file containing your solutions to this assignment on Canvas.
  - You can use this Overleaf project to write your solutions.
  - Submit the ZIP file containing your code to this separate assignment on Canvas.
- You have only one attempt on Canvas.
- You are required to specify your group details in your attempt (group number can be found on Canvas) by updating the `team.tex` file with the relevant details.
- **Total marks: 20; Weightage 10% of final marks**
- On collaboration:
  - The goal of the assignment is to understand and apply the concepts in the class.
  - You may discuss the assignment with other groups via the discussion forum.
  - It is OK for the solution ideas to arise out of such discussions. However, it is considered plagiarism if the solution submitted is highly similar to other submissions or to other sources.
- Citing help and reference
  - At the end of the assignment, clearly cite the sources you referred to when arriving at the answer (Books, External notes, Generative AI tools, etc.,).

### Team members

Group number:

Member 1 details:

- Name:
- NUSNet id:
- Student number:

Member 2 details:

- Name:
- NUSNet id:
- Student number:

# 1 Decision Analysis: EVPI

Singoil refineries, catering to the ASEAN market, like any other petroleum refinery is affected by the demand-supply problem. After analyzing some historical data, the analysts have identified the demand to be three discrete values: 90 million barrels/month, 100 million barrels/month, or 120 million barrels/month. The monthly demand has the following probabilities:

| Demand          | Demand probability |
|-----------------|--------------------|
| 90 mil barrels  | 0.2                |
| 100 mil barrels | 0.5                |
| 120 mil barrels | 0.3                |

Table 1: Oil demand probabilities

The cost of producing oil (per 10 million barrels) is \$70 million. Singoil sells oil at \$90 million per 10 million barrels.

1. (4 marks) Based on the above information, how many barrels of oil should Singoil produce monthly to maximize the profit? (Show your working)
2. (4 marks) Sureoil predictors employ Pauline, [Paul, the Octopus](#)' cousin, to accurately predict the monthly oil demand. How much should Singoil be paying for Pauline's prediction so that the profit is maximized? (Show your working)

## 2 MDP

Moonwalker, the moon rover, is embarking on a mission to the moon to collect “interesting” rock samples in five regions on the moon: (regions  $R1, R2, R3, R4, R5$ ). Moonwalker’s two actions are to drill for rock samples ( $D$ ) or stop to charge the onboard batteries using sunlight ( $C$ ).

1. (4 marks) Formulate Moonwalker’s decision problem as a discrete MDP with the following conditions:
  - (a) Moonwalker starts from  $R1$  and progressively goes to regions with higher index (i.e.,  $R1 \rightarrow R2 \cdots \rightarrow R5$ ). When it reaches  $R5$  it will stay there forever, irrespective of the action.
  - (b) For the other regions if action  $D$  is taken, it will always transition to the next region at the next time step; if action  $C$  is taken, it will always remain in the same region at the next time step.
  - (c) Assume that “interesting” rocks are found only in region  $R3$ , and will give a reward of 256, if action  $D$  is carried out; all other rewards are zero.

Specify the state space and action space using proper set notations; transition function, and reward function as tables for convenience.

2. (2 marks) If the utility of state  $R1$ ,  $U^*(R1) = 4$ , what is the discount factor  $\gamma$  based on the considerations given in the above question?

### 3 Programming component

(6 marks)

In this programming assignment, we will use a modified form of the [elevator domain](#) from the International Planning Competition, 2023.

The Elevator domain models evening rush hours when people from different floors in a building want to go down to the ground floor using elevators. Potential passengers arrive at a floor according to the Poisson process, with specific rates that can be different across different floors. An elevator can move upwards to pick up passengers; once it opens its door to let people in, it can only move down towards the ground floor, though the elevator can stop at intermediate floors to pick up more passengers.

Per each person in an elevator, there is a small penalty in rewards. Additionally, there is a penalty for each person waiting for an elevator. When an elevator delivers a passenger to their destination, a large positive reward is given per person. So, a good policy should be able to minimize the penalties while maximizing the positive rewards by delivering many people so that they can go back home!

#### Handout and task

- Use the `assignment2_handout.zip` file on Canvas to attempt the programming assignment.
- The `README.md` file has additional details about the environment.
- Your task is to complete the `solve_value_iteration` function that implements the Value Iteration algorithm in the `agent/agent.py` file. (It is marked with `FILL_ME`)

#### Submission

- Please write both the team member names and student numbers (Axxxxxxx) at the top of the agent file (as comments) to help us identify your submissions.
- **Zip the agent .py file alone and submit this zip file.**
- Submissions for the programming part will happen on the [AiRENA](#) portal for grading.
- You are **also required** to submit the zip file on Canvas for record-keeping/backup purposes.
- Please refer to the user guide ([AiRENA Guide for CSxx46.pdf](#) file on Canvas) for submission instructions.
  - Please ensure you follow the instructions provided to create your account.
  - **If the account details don't match your information on Canvas, such submissions will not be graded.**
- You need to use the following token to join the course:  
**b68f4d45-a95f-449d-93f7-2e1f0edf154e**

## Grading

- We will evaluate your implementation using three private test cases, and take the average of the score obtained.
- To obtain full 6 marks for this part, you need to attain an average score of **-300** and above.

## Support

- We have prepared a [Google collab file](#) for you to play around and understand the elevator domain.
  - If you have questions about the assignment, please ask on the discussion forums.
-