

Graph Traversal & Evaluation – Chansey

This document demonstrates a graph traversal and tries to answer 'How well has a task been completed if it fails?'. It has 2 questions.

Q1. A task requesting Chansey (currently at Room 0.01) to come to Room 2.01 is uploaded to the network Task Queue. Does Chansey reach that location?

A. Chansey currently is at Room 0.01. That marks its current position. A task is uploaded to the network and is placed on the Task Queue. The relevant state variables are as follows:

```
"taskQ": {  
    ...  
    ...  
    "@id": "TaskQueue",  
    "tasks": ["<task1>"]  
    ...  
    ...  
}
```

The above portion of the graph is found in `hospital.json`.

As part of Chansey's monitoring actions, the following steps are executed:

- **Check-Task-Queue** which checks if there are tasks in the queue. Since this is now satisfied, **Current-Task** is now updated.
- **Check-Pending-Task** which checks if the robot has accepted a task now finds that there is a pending Current-Task. It sets the appropriate variables and tries to evaluate the goal.

2.01 is on level 2, 10 units from the elevator. The relevant state variables are now set as follows:

```
"deliveryLocation": {  
    ...  
    ...  
    ...  
    "@id": "Delivery_Location",  
    "@value": [  
        ...  
        ...  
        {  
            "@id": "Steps",  
            "heightDifference": 2,  
            "toElevator": 10,  
            "fromElevator": 12,  
            "intoElevator": 2,  
            "sameFloorDistance": 0
```

```

    }
  ]
}
```

This portion of the property graph can be found in [Chansey.json](#).

Starting from the goal to choose the action plan, we move to the action plan selector. The condition here is falsified, as `sameFloorDistance` is 0. As the condition has not been satisfied, the action plan chosen is `Chansey action plan`. The sequence of actions executed are as follows:

- `Move(distance = 10)` which satisfies the preconditions (`distance > 0`) set by the meta model - `Motion`.
- `RequestElevator` which satisfies the preconditions set by the meta model `Elevator Communication`.

Note: 10 units of motion puts it at the `accessPosition` since the distance from the elevator to the room is 10.

- `Move(distance = 2)` which satisfies the preconditions (`distance > 0`) set by the meta model - `Motion`.
- `Move Elevator(floors = 2)` which satisfies the preconditions (robot is inside the elevator) set by the meta model - `Elevator Communication` and with this Chansey reaches the 2nd Floor.
- `Rotate(angle = 180)` which turns the robot onto the outgoing corridor.
- `Move(distance = 12)` satisfies the required preconditions (`distance > 0`) and puts the robot in front of `Room 2.01`.

Since the action plan has been completely executed, on checking the goal we find that the desired result has been achieved.

Q2. Assume the task fails at a certain step. How is the performance of the robot evaluated?

A. Say (with no loss in generality) the task fails at `Rotate(angle=180)`. The robot now evaluates the action that should be carried out on failure. The robot sends the status of the task and its plan to the corresponding evaluator. The actions in the plan with each of their respective scores (metamodel) are as follows:

- `Move(distance = 10)` - 8
- `RequestElevator` - 4
- `Move(distance = 2)` - 8
- `Move Elevator(floors = 2)` - 4
- `Rotate(angle = 180)` - 8
- `Move(distance = 12)` - 8

The total possible score is **40**. The score achieved by the robot is **24**. Thereby, a conclusion based on the relative weightage given to each of the actions, it can be said that the robot has completed **60%** of the task.