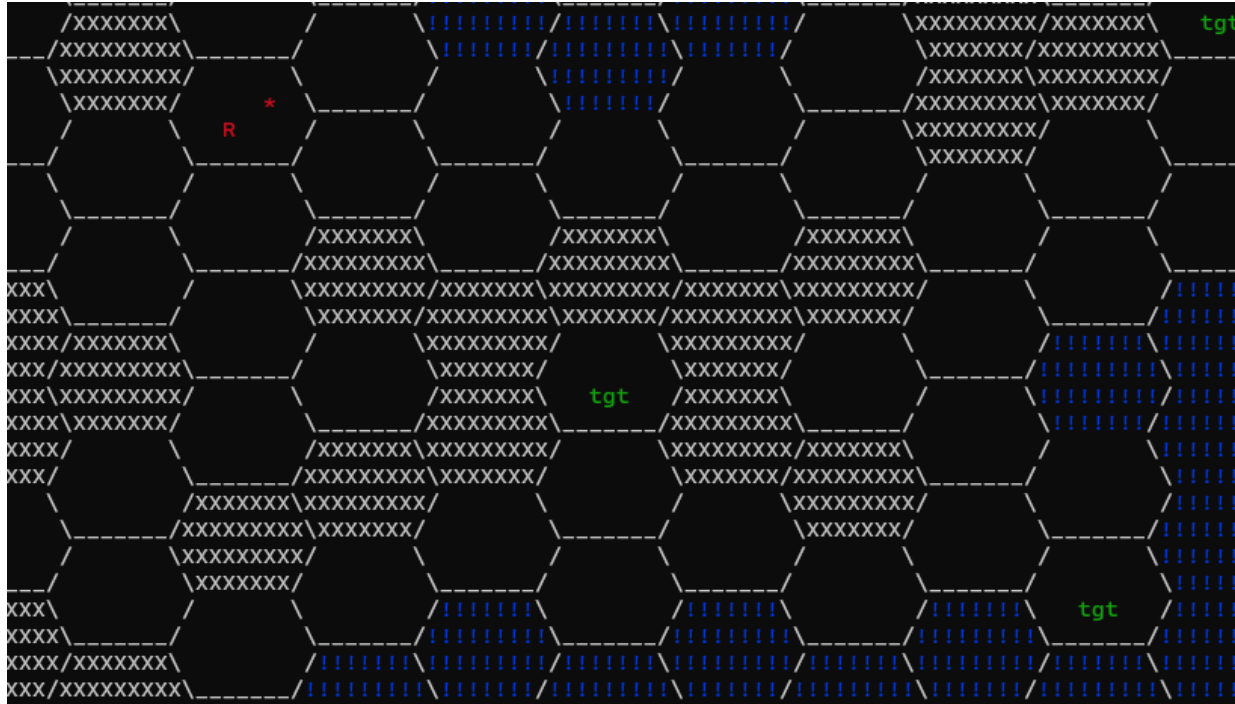


HexBot Robot Environment (A3 Update)

COMP3702 Artificial Intelligence 2022



You have been tasked with developing a **reinforcement learning** algorithm for automatically controlling HexBot, a multi-purpose robot which operates in a hexagonal environment. **For this version of the task, the HexBot must navigate to one of the designated targets while incurring the minimum penalty score possible. Widgets do not need to be considered.** To aid you in this task, we have provided a simulator and visualisation for the HexBot robot environment which you will interface with to develop your solution.

For A3, the HexGrid environment has non-deterministic action outcomes with unknown probabilities which are randomised for each testcase. The cost of each available action and the penalties for collision with obstacles and hazards are also unknown and randomised for each testcase. Updates to this document are shown in magenta text.

Hexagonal Grid

The environment is represented by a hexagonal grid. Each cell of the hex grid is indexed by (row, column) coordinates. The hex grid is indexed top to bottom, left to right (i.e. the top left corner has coordinates (0, 0) and the bottom right corner has coordinates (n_rows-1, n_cols-1)). Even numbered columns (starting from zero) are in the top half of the row, odd numbered columns are in the bottom half of the row. e.g.

```

row 0, col 0           row 0, col 2           ...
           row 0, col 1           row 0, col 3
row 1, col 0           row 1, col 2           ...
           row 1, col 1           row 1, col 3
...           ...           ...           ...

```

Two cells in the hex grid are considered adjacent if they share an edge. For each non-border cell, there are 6 adjacent cells.

Robot

The HexBot robot occupies a single cell in the hex grid. In the visualisation, the robot is represented by the cell marked with the character 'R'. The side of the cell marked with '*' represents the front of the robot. The state of the robot is defined by its (row, column) coordinates and its orientation (i.e. the direction its front side is pointing towards).

The robot has 4 available nominal actions:

- Forward → move to the adjacent cell in the direction of the front of the robot (keeping the same orientation)
- Reverse → move to the adjacent cell in the opposite direction to the front of the robot (keeping the same orientation)
- Spin Left → rotate left (relative to the robot's front, i.e. counterclockwise) by 60 degrees (staying in the same cell)
- Spin Right → rotate right (i.e. clockwise) by 60 degrees (staying in the same cell)

Each time the robot selects an action, there is a fixed probability (set randomly based on the seed of each testcase) for the robot to 'drift' by 60 degrees in a clockwise or counterclockwise direction (separate probabilities for each drift direction) before the selected nominal action is performed. The probability of drift occurring depends on which nominal action is selected, with some actions more likely to result in drift. Drifting CW and CCW are mutually exclusive events.

Additionally, there is a fixed probability (also set randomly based on the seed of each testcase) for the robot to 'double move', i.e. perform the nominal selected action twice. The probability of a double move occurring depends on which action is selected. Double movement may occur simultaneously with drift (CW or CCW).

The reward received after each action is the minimum/most negative out of the rewards received for the nominal action and any additional (drift/double move) actions.

Obstacles

Some cells in the hex grid are obstacles. In the visualisation, these cells are filled with the character 'X'. Any action which causes the robot or any part of a Widget to enter an obstacle cell results in collision, causing the agent to receive a negative obstacle collision penalty as reward. This reward replaces the movement cost which the agent would have otherwise incurred. The outside boundary of the hex grid behaves in the same way as an obstacle.

Additionally, the environment now contains an additional obstacle type, called 'hazards'. Hazards behave in the same way as obstacles, but when collision occurs, a different (larger) penalty is received as the reward. As a result, avoiding collisions with hazards has greater importance than avoiding collisions with obstacles. Hazards are represented by '!!!' in the visualisation.

Targets

The hex grid contains a number of 'target' cells. In the visualisation, these cells are marked with 'tgt'. For a HexBot environment to be considered solved, one of the target cells must be occupied by the HexBot. Environments may contain multiple targets.