2022UG000009
Amar Singh
RL_Assignment_1

Design, Analyse, and Simulate a Maze World as an MDP

Amar Singh.

RL Assignment

1. Formalize the Grid Maze as MDP

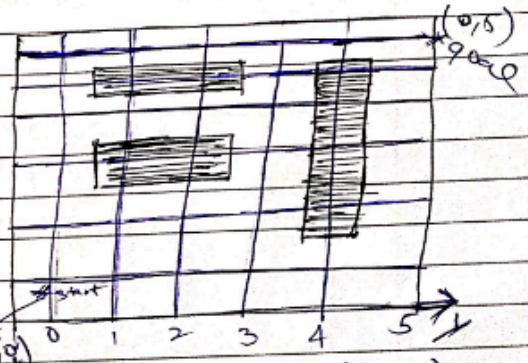
a) State Space: We use a 2D Grid matrix with integer coordinate & zero based indexing with (0,0). A state is a cell $S = (r, c)$ (row, column) where $r, c \in \{0, \dots N-1\}$ and cell is not a wall. The goal cell is terminal. (We are assuming $N = 6$).

Action space = $A = \{UP, Down, Left, Right\} = \{U, D, L, R\}$ All action available in every non-terminal State.

Transition function $p(s'/s,a)$

$U = (-1, 0)$ $L = (0, -1)$
$D = (+1, 0)$ $R = (0, +1)$

From state $s = (r, c)$ & action a the next cell is $(r', c') = (r, c) + \Delta a$



- If (r', c') is outside the grid or is a wall the agent stays in place $s' = s$, otherwise $s' = (r', c')$
- If s' is the goal, the episode terminates.

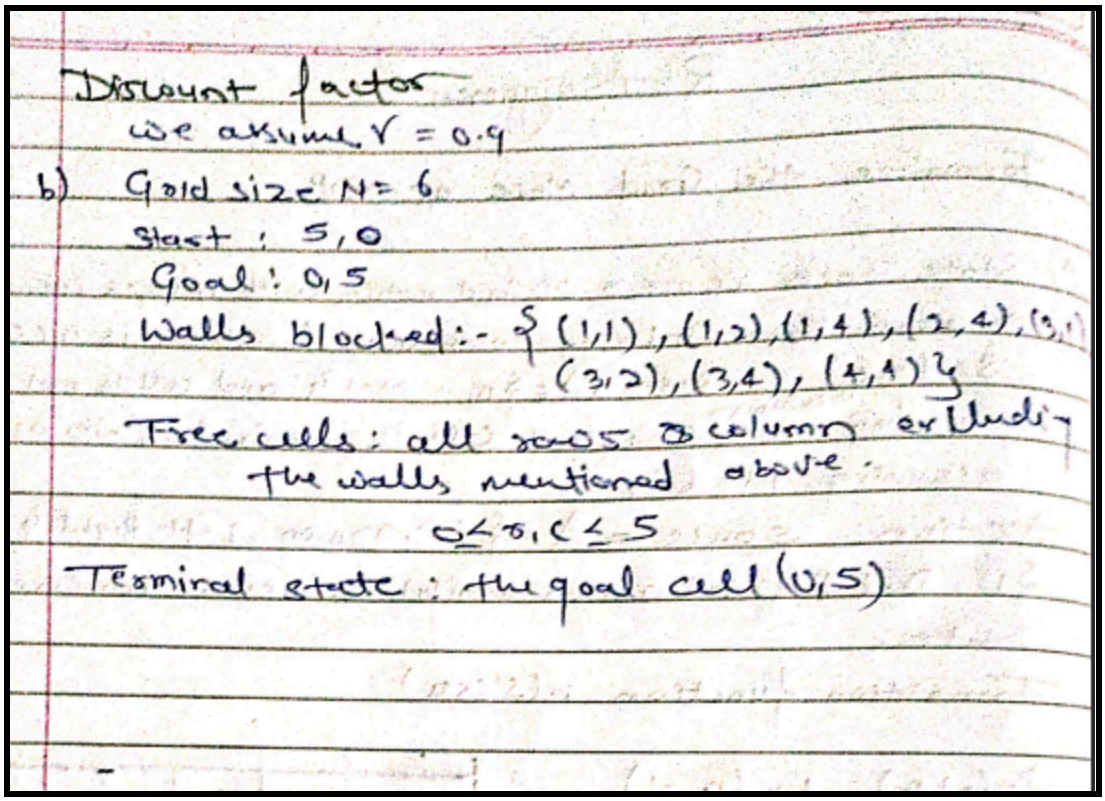
for any non terminal s,

$$P(s'/s,a) = \begin{cases} 1, & \text{if } s' = (s,a) \\ 0, & \text{otherwise.} \end{cases}$$

Reward function.

$r(s, a, s')$

- Step cost (valid move to a non-goal cell) = -1
- Bump into wall / of grid (invalid move) = -2
- Enter the goal cell = +10



Discuss how wall placement and reward values affect agent behaviour.

Wall blocks direct paths

When the agent is trying to move in a straight path but if the wall is there it will not be able to move straight which will force the agent to take more steps and because of that there will be less reward points as each step costs -1 .

In a maze , the greedy policy wants to take up and right , but if the wall is there it will lead to time waste as it will keep on circling.

For example if it starts from (5,0) the greedy wants to go straight up. But if it encounters the wall like (3,1), it has to adjust otherwise it will not be able to reach the goal.

How reward values affect behaviour

Step Penalty (-1) : makes the agent take a shorter path. If it's 0 it doesnt mind wandering.

Bump Penalty(-2) : Punishes the wrong moves , it teaches the agent not to keep walking to walls.

Goal Reward : (+10) : Strongly motivates the agent to reach the goal , if it's not their agent might just wander because moving only loses reward points.

Based on simulation results, what are the strengths and weaknesses of random vs. greedy policies?

Random Policy

Strengths :

It explores everywhere , so it doesn't get stuck in one spot.

Sometimes it accidentally finds a path around tricky wall placement that a greedy agent might avoid.

Weakness

Very inefficient - takes a lot of steps

Only success is around 60 percent.

Rewards are very low as most of the time wastes steps and bumps into the wall most of the time in negative.

Greedy Policy

Strengths

Always efficient - reached exactly in 10 steps.

Success rate is 100%

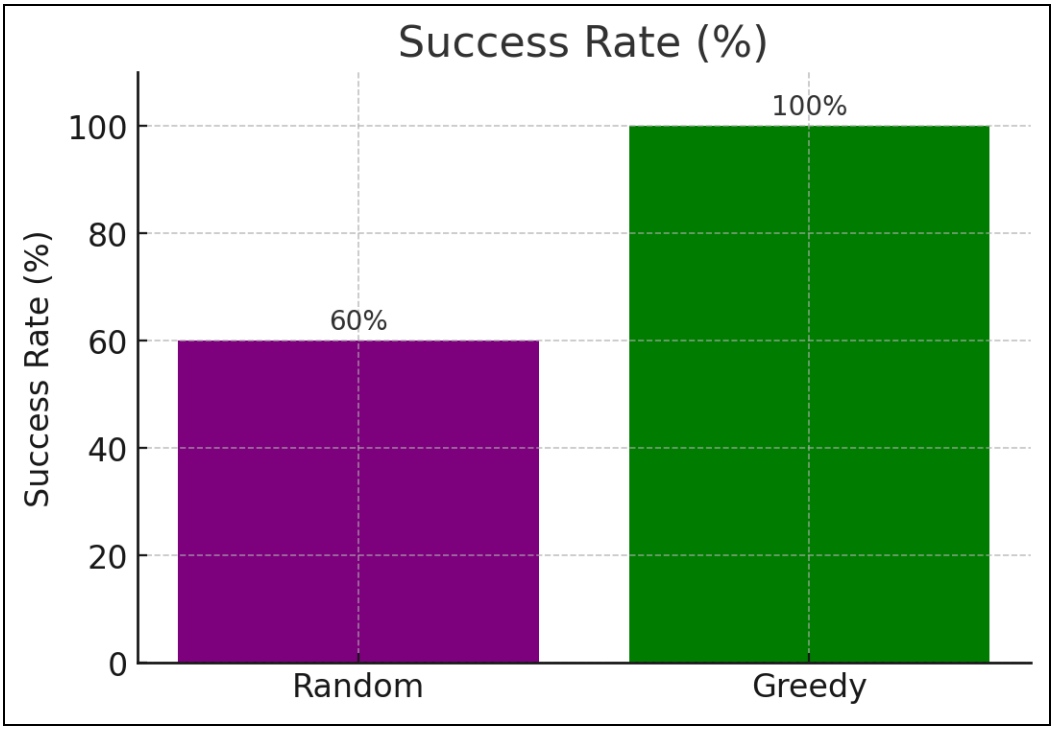
Rewards are positive as it reaches the goal and avoids penalties.

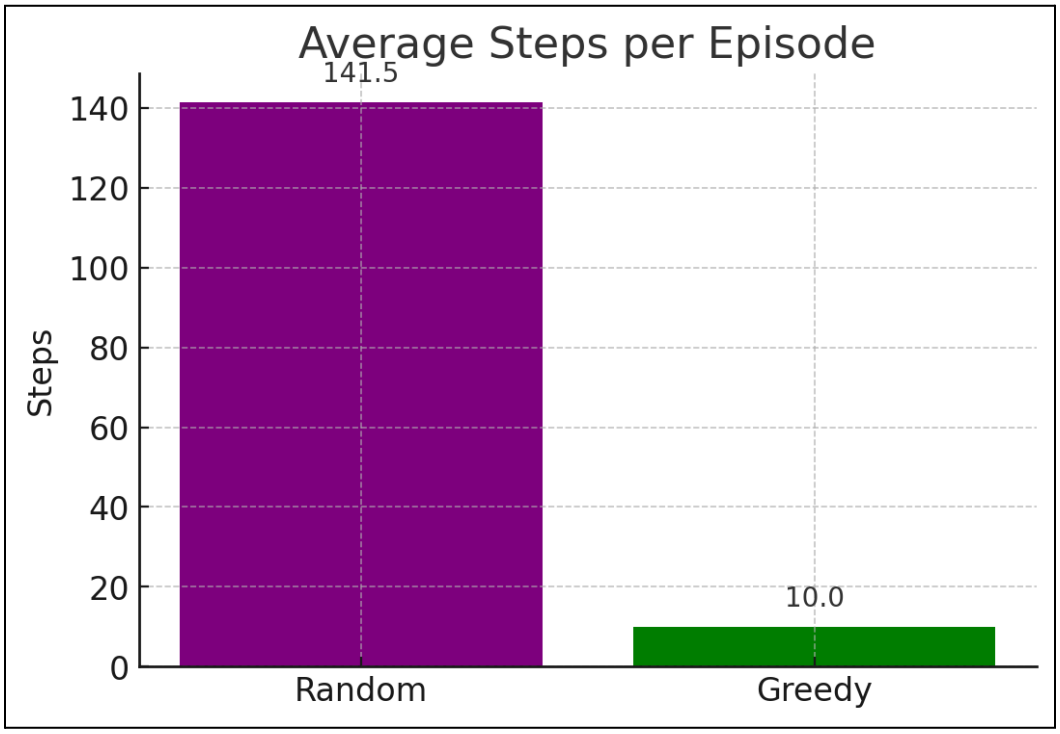
Weakness

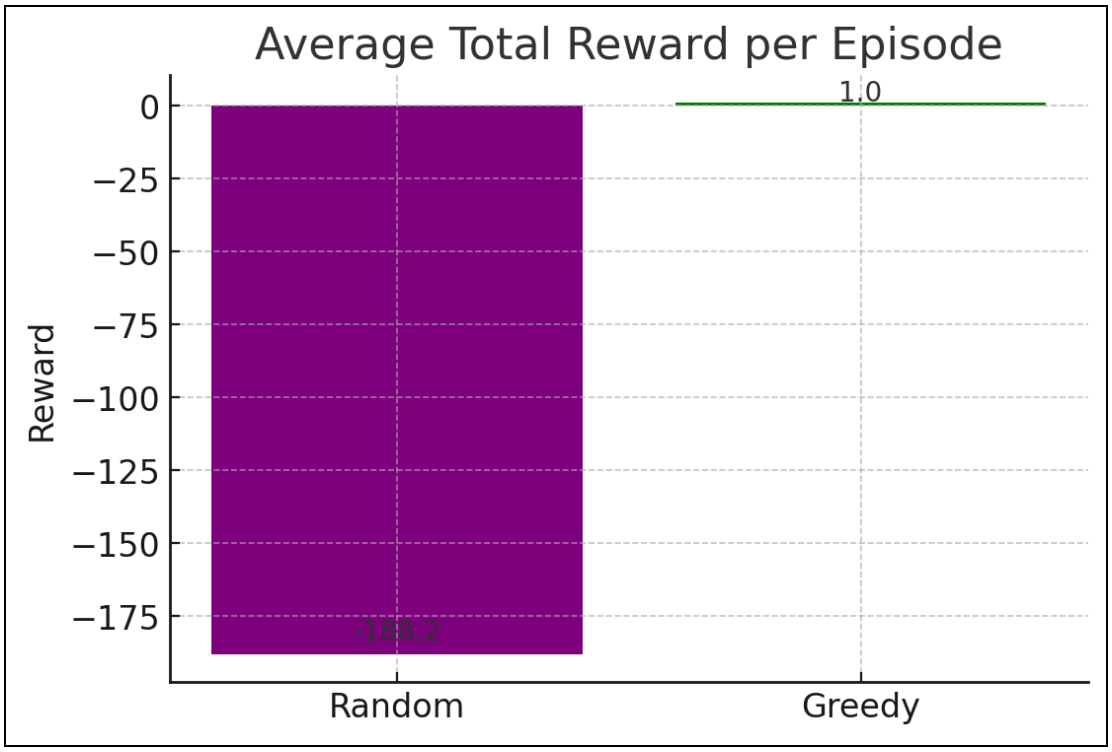
It only looks the direct path towards the goal

If a wall blocks that direct path , it may loop or fail unless we add some exploration.

Policy	Success Rate	Steps (min/max/avg)	Reward (min/max/avg)
Random	60% (12/20)	34 / 200 / 141.5	-287 / -37 / -188.2
Greedy	100% (20/20)	10 / 10 / 10	1 / 1 / 1





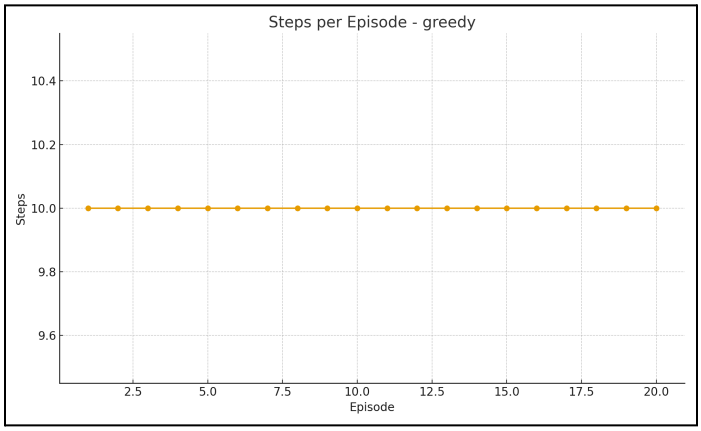


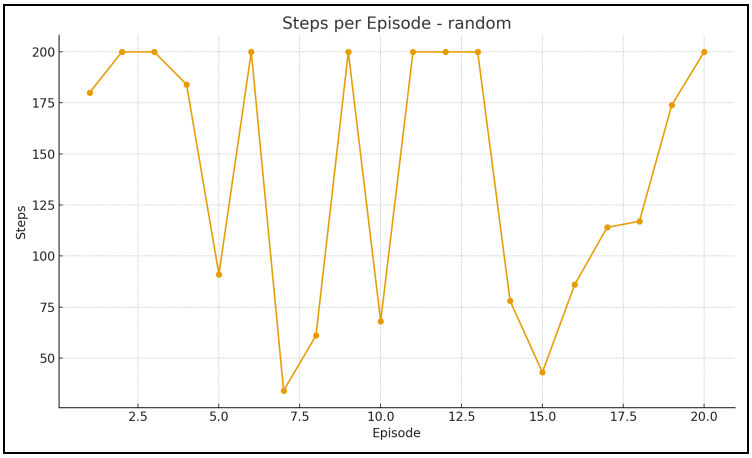
Suggest simple ways to further improve navigation in this environment.

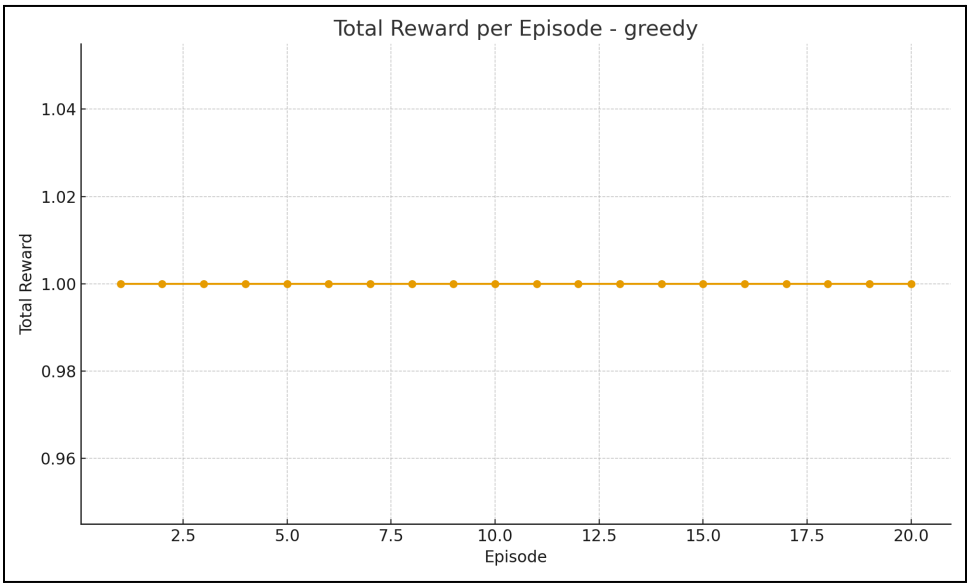
. Allow agents to be greedy most of the time say 90 % but random for sometime say 10 % , it will help to escape the traps near the walls and find more efficient ways.

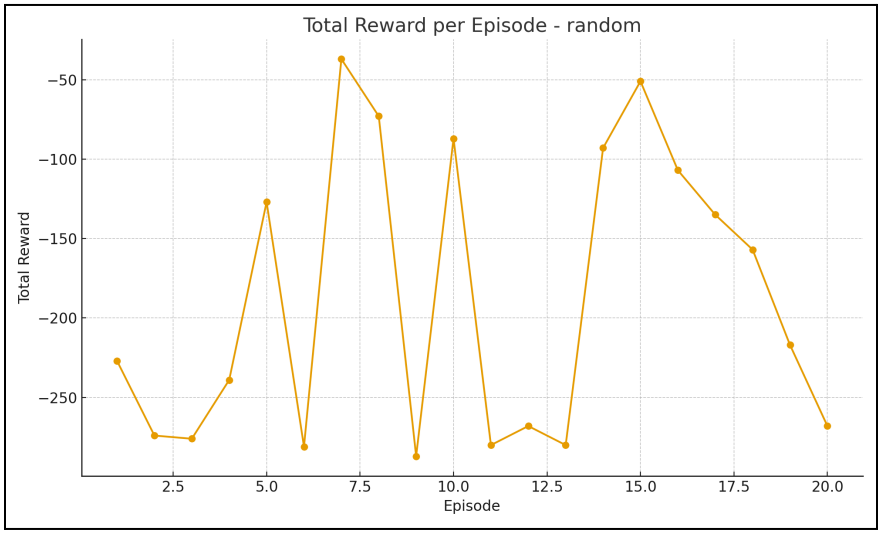
. Making the penalty more harsher say -2 that will push the agent to find shorter and more direct routes.

. We can give some reward points if the agent moves closer to the goal; this will give the agent a sense of progress, not just punishment, until it reaches the final +10.


Steps per Episode - greedy
Steps
Episode
10.4
10.2
10.0
9.8
9.6
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0


Steps per Episode - random
Steps
Episode
200
175
150
125
100
75
50
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0


Total Reward per Episode - greedy
Total Reward
Episode
1.04
1.02
1.00
0.98
0.96
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0


Total Reward per Episode - random
Total Reward
Episode
−50
−100
−150
−200
−250
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0