Cairo University

Faculty of Engineering

Computer Engineering Department

CMPS458 Reinforcement Learning Report Assignment 1

Team Name/Number: 2
First member Name: Chahd Ihab ElZalaky
Second Member name: Amira Ibrahim Ahmed

Supervisor: Ayman AboElhassan

October 22, 2025

Deliverables

Repo link: https://github.com/amiraelgarf/policy-iteration-assignment

Video record link: https://github.com/amiraelgarf/policy-iteration-assignment/issues/1

Discussion

0.1 Experiments

The experiement starts by creating the grid environment, with random placements for the goal, mines and agent. Then, the agent is trained on this environment using policy iteration to generate the optimal policy that directs the agent to the goal. To test this policy, a new environment is created with the same placements for the goals and mines, but with a different position for the agent chosen randomly, so now the agent uses the learnt optimal policy to be able to reach the goal and avoid the mines. The random placements for the goals, mines and agents are chosen using a seed that is changed at each sample experiment to obtain a new grid environment. After training , the agent has the optimal policy so it can use it during testing to reach the goal. The agent during testing eventually reaches the goal following the optimal policy but it may sometimes deviate due to the stochasticity of the environment, but it is able to go back on track to the right trajectory to the goal. The only case where it may fall into a mine is if the mine is right next to the goal and the agent intended to move to the goal, but due to the stochasticity, it ended up falling in a mine.
Another experiment we conducted is testing the optimal policy on a completely new environment with new placements for the goal, mines and agent.The agent ended up falling in the mine during testing because the policy he was trained on was for a different goal placement (hence a different goal state) therefore the agent was using an invalid policy. It reached the goal once but that's because the policy the agent trained on happened to coincide with the with the testing goal (happened by mere coincidence).
In conclusion, the agent's behavior in this case is random.

0.2 Question Answers

1. **What is the state-space size of the 5x5 Grid Maze problem?**
   25 states, one for each cell of the grid.

2. **How to optimize the policy iteration for the Grid Maze problem?**
   It could be optimized by using a threshold to stop iterating early instead of waiting until full convergence.

3. **How many iterations did it take to converge on a stable policy for the 5x5 maze?**
   It depends on the seed used to set the positions of the goals, mines, and the agent. It also depends on the discount factor used in the value function calculation. In our experiments, it took approximately 5 iterations on average.

4. **Explain, with an example, how policy iteration behaves with multiple goal cells.**
   During policy evaluation, the cells close to the goals will have high values, hence the policy's actions will point toward them. During policy improvement, the agent

chooses the action that leads toward the highest expected value, which means it will learn a policy that directs it toward the closest or most easily reachable goal cell, considering rewards, mine positions, and stochastic movement.

Example: Imagine goals at (0,4) and (4,4) in a 5x5 grid, and mines at (1,1) and (3,3). From square (2,2), policy evaluation calculates values based on paths to both goals. During improvement, the agent at (2,2) chooses actions leading to the goal at (0,4) because it yields a higher expected reward due to being further from the mines.

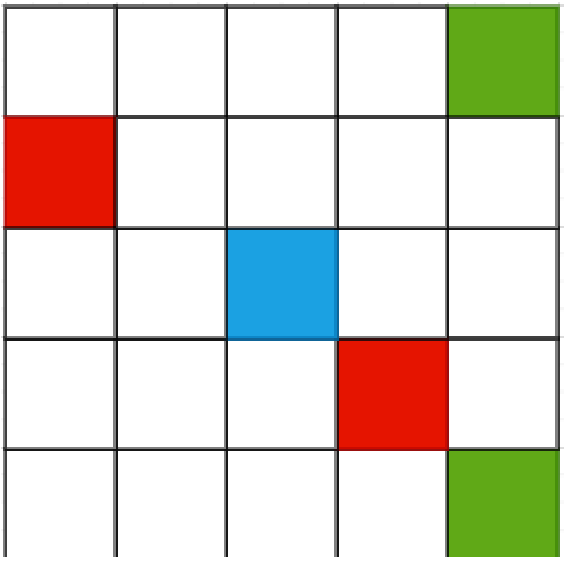

Figure 1: Grid example with multiple goals

5. Can policy iteration work on a 10x10 maze? Explain why.

Technically yes, because the state space now consists of 100 states, which is still feasible for the learning algorithm. However, it will be more computationally expensive and will take longer to converge.

6. Can policy iteration work on a continuous-space maze? Explain why.

No, because a continuous-space maze implies an infinite number of states, which is incompatible with policy iteration. Policy iteration relies on having a finite, discrete number of states to perform value updates and policy improvements.

7. Can policy iteration work with moving bad cells (like Pac-Man moving ghosts)? Explain why.

No, because policy iteration requires a static environment in which the transition probabilities $P(s'|s,a)$ remain constant over time. The evaluation phase assesses the current policy based on fixed goal and mine positions. If the “ghosts” (mines) move constantly, the environment becomes non-stationary, preventing the agent from properly evaluating and improving its policy.