



WAREHOUSE AUTOMATION

PRESENTED BY –

NEERAJ SAHASRABUDHE

JITESH SONKUSARE



PROBLEM STATEMENT

Automating warehouse operations through the utilization of various Reinforcement Learning (RL) techniques for optimizing a robot's pathfinding process, and subsequently, comparing this performance with conventional algorithms such as A^* to determine the most efficient path

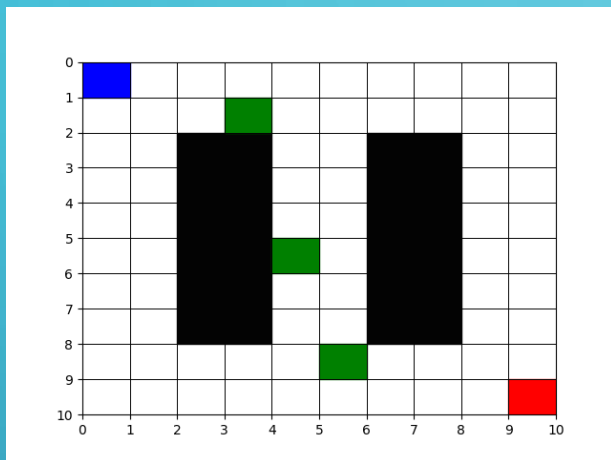
METHODS USED

- Monte Carlo Methods
- Temporal Difference (TD) Learning - Q-Learning
- Deep Q-Learning
- A - Star Search

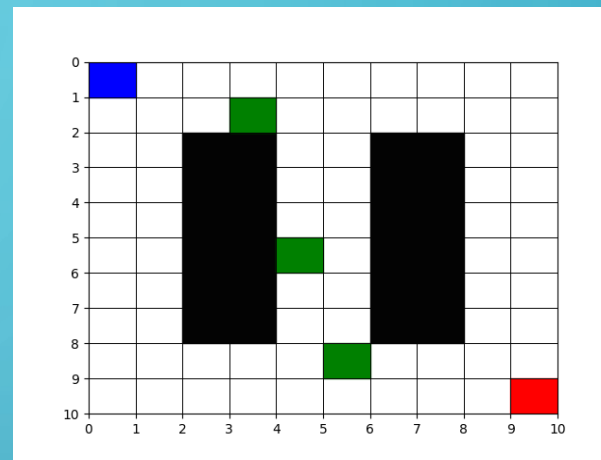


COMPARISON METRICS

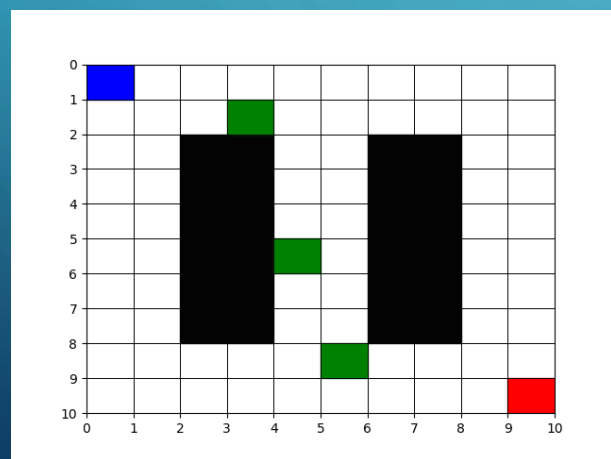
A - Star



Q - Learning

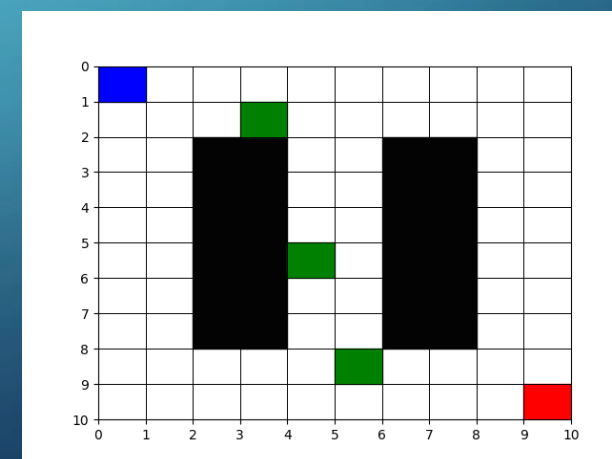


Monte Carlo



**Visualization
of the world**

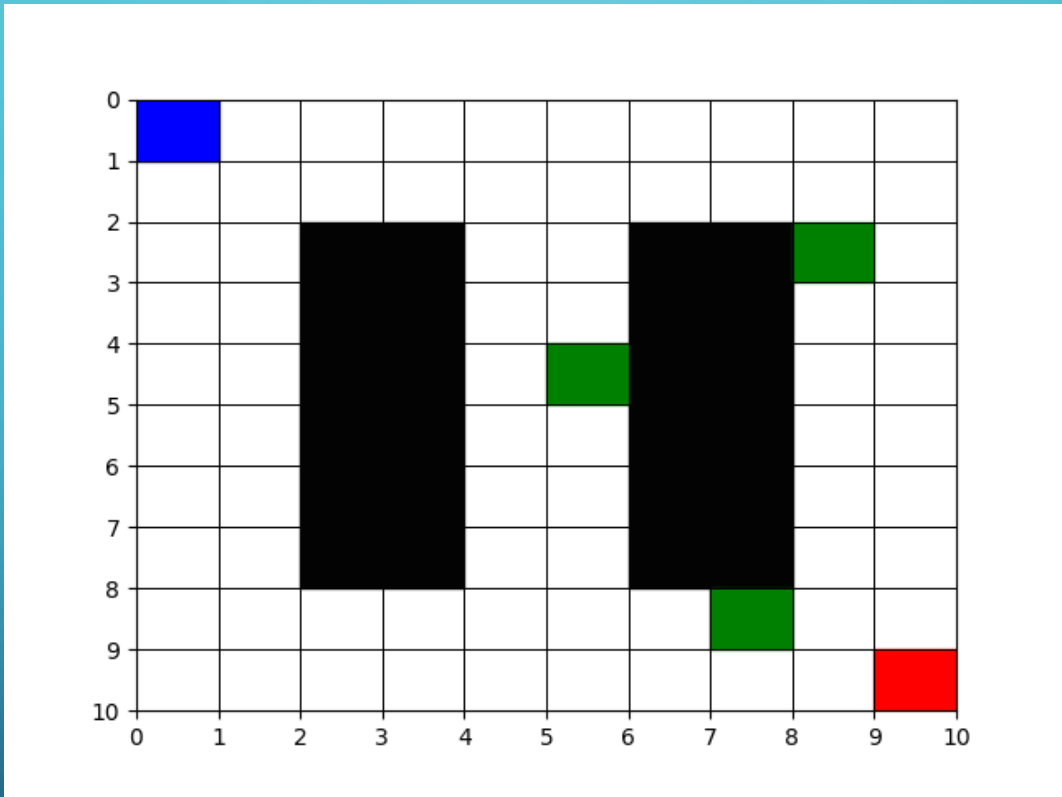
Deep Q - Learning



A 10x10 grid with the following colored cells:

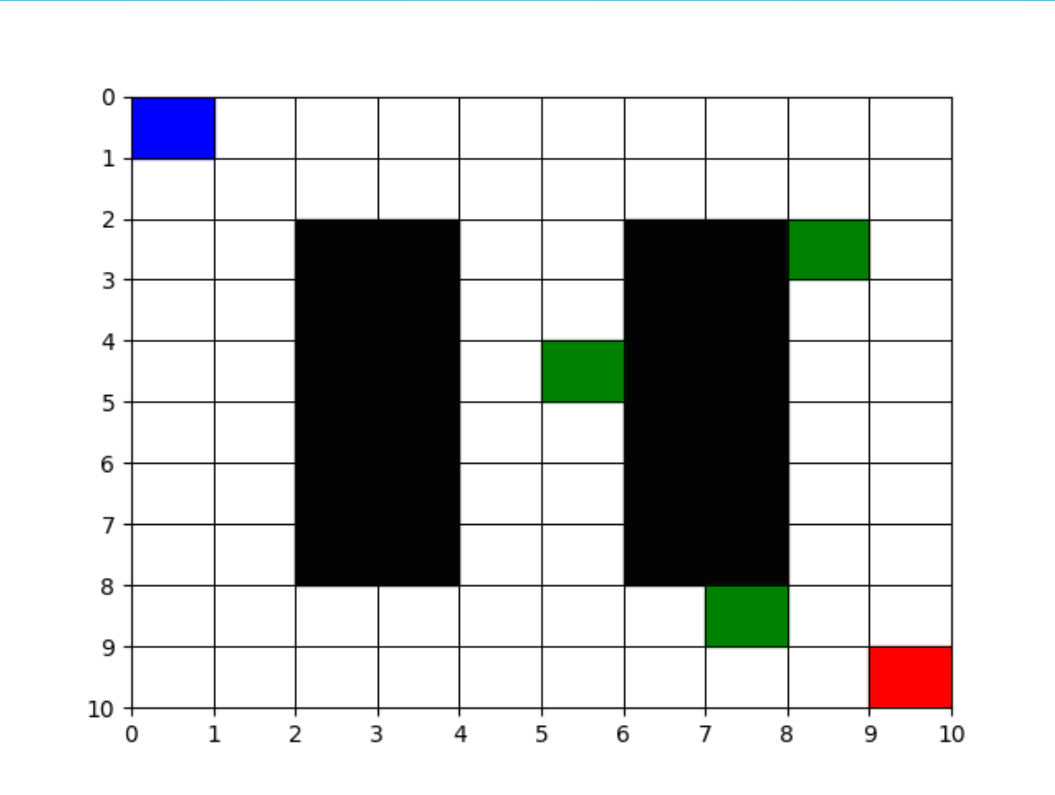
- Blue: (0,0)
- Green: (3,1), (4,5), (5,8)
- Black: (2,2), (3,2), (4,2), (2,3), (3,3), (4,3), (2,4), (3,4), (4,4), (2,5), (3,5), (4,5), (2,6), (3,6), (4,6), (2,7), (3,7), (4,7), (2,8), (3,8), (4,8), (6,2), (7,2), (8,2), (6,3), (7,3), (8,3), (6,4), (7,4), (8,4), (6,5), (7,5), (8,5), (6,6), (7,6), (8,6), (6,7), (7,7), (8,7), (6,8), (7,8), (8,8)
- Red: (9,9)

TESTING FOR VARIED ENVIRONMENTAL CONDITIONS



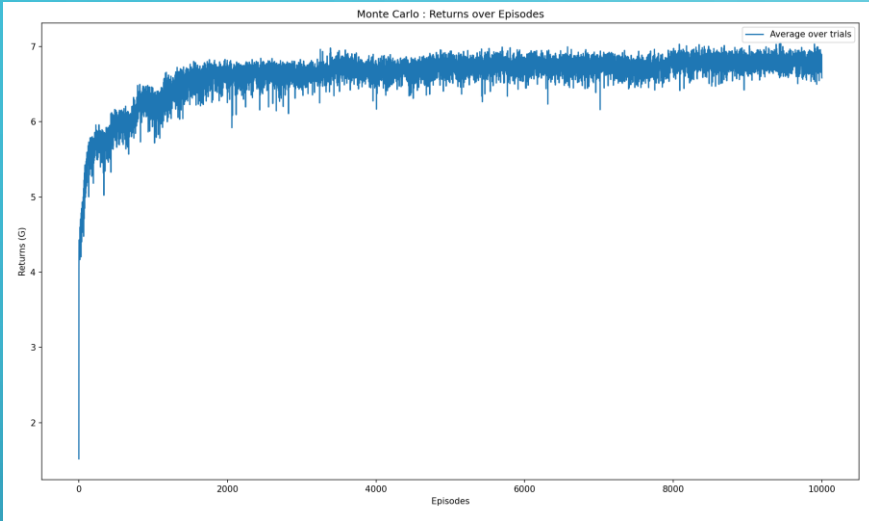
A 10x10 grid representing a testing environment. The grid is indexed from 0 to 10 on both the x and y axes. The cells are colored as follows:

- Blue: (0, 0)
- Black: (2, 2), (3, 2), (4, 2), (2, 3), (3, 3), (4, 3), (2, 4), (3, 4), (4, 4), (2, 5), (3, 5), (4, 5), (2, 6), (3, 6), (4, 6), (2, 7), (3, 7), (4, 7), (2, 8), (3, 8), (4, 8)
- Green: (8, 2), (5, 4), (7, 8)
- Red: (9, 9)

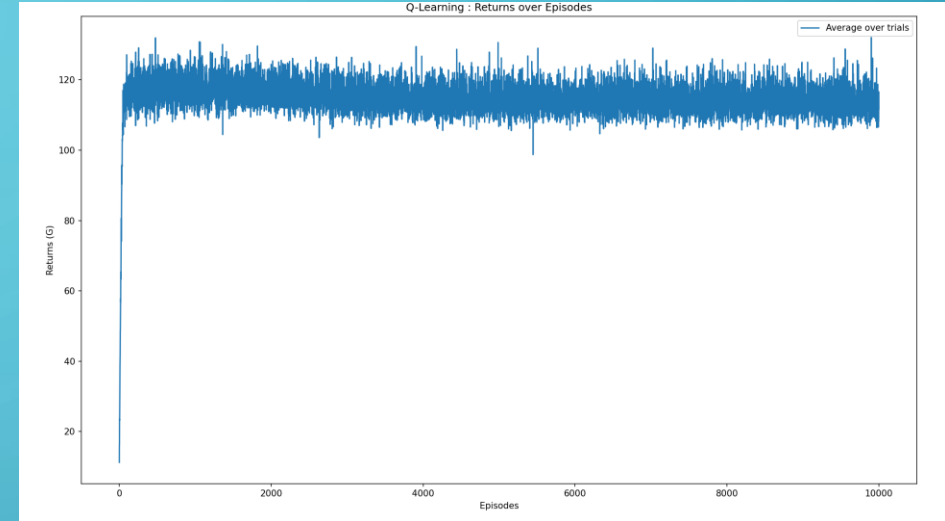


RESULTS

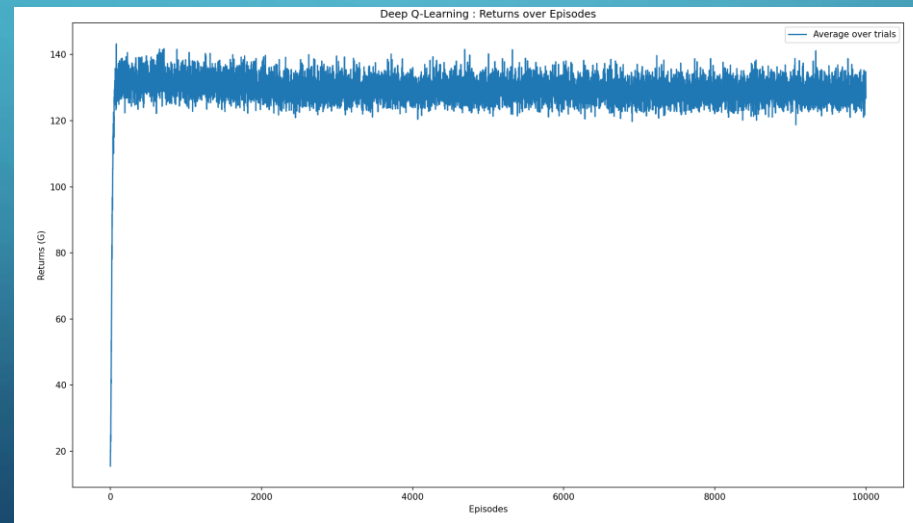
Returns vs Number of Episodes



Monte Carlo



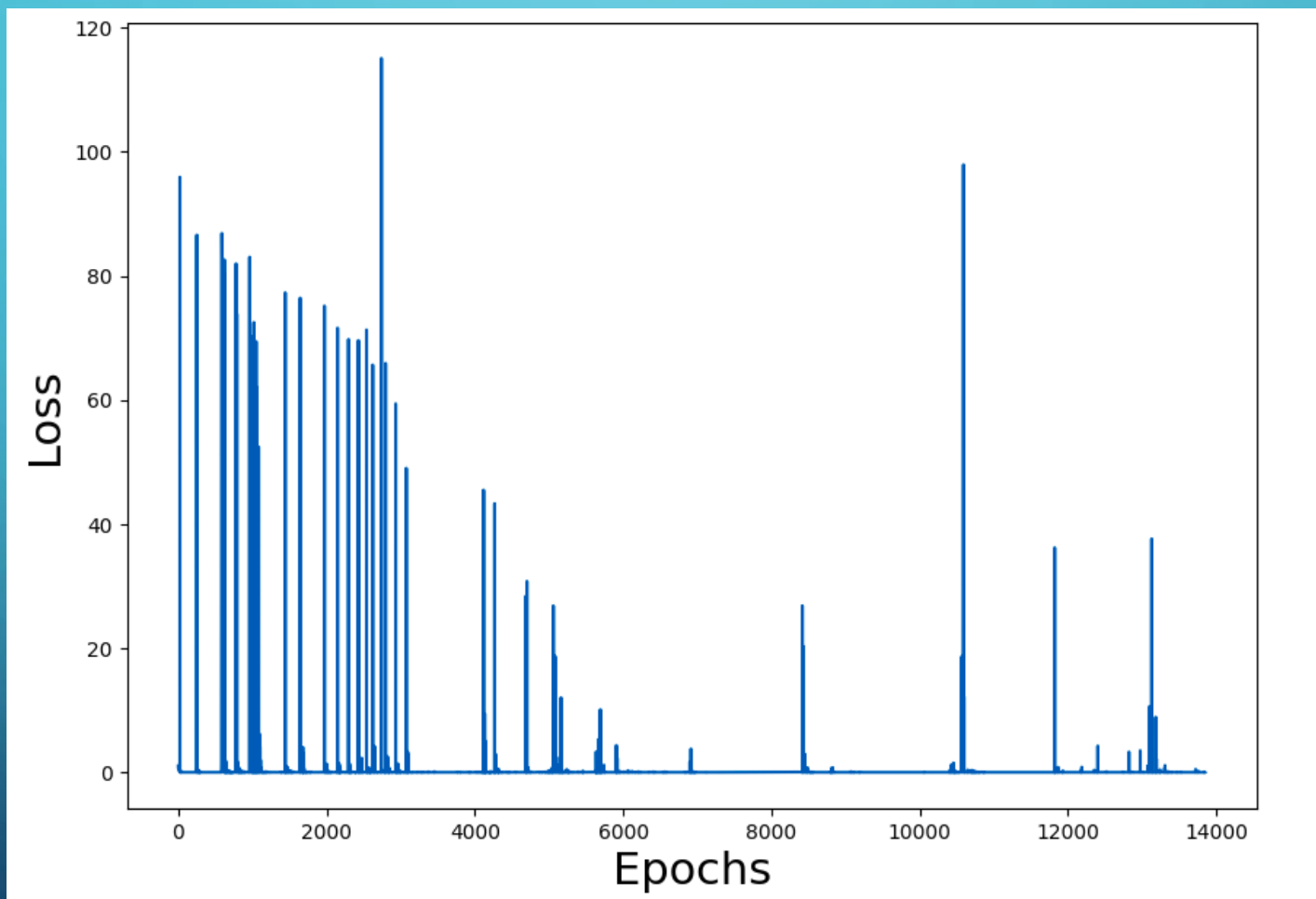
Q-Learning



DQN

RESULTS

Loss vs Epoch for DQN



CONCLUSION

- Reinforcement Learning algorithms can be used to automate robotic warehouse environments
- Optimal policies obtained from RL algorithms provide similar optimal paths as those obtained from A* Search
- As RL algorithms do not require a map of the environment, the implementation in larger environments is simpler and more efficient

FUTURE WORK

- Multi Robot Agent
- Dynamic Obstacles



The background is a blue gradient with decorative white circuit-like lines in the corners. The lines consist of straight segments and small circles, resembling a stylized electronic circuit.

THANK YOU

GitHub Link :

<https://github.com/jitesh3023/Reinforcement-Learning-Final-Project.git>