ROB 537 Homework 3: Reinforcement Learning

Nathan Butler | butlnath@oregonstate.edu

# Overview

The objective of this assignment is to train an agent to explore a 5x10 gridworld with the objective of finding a door. The environment includes:

* a door initialized at coordinates (9,1) with a reward of 20;
* a solid wall (gray) the agent cannot move across;
* a reward of -1 for every time the agent is in a state other than the red door state.

The agent starts at a random location and has five actions (move in four directions or stay in place), the state of the system is the location of the agent (x,y), and an episode is 20 time steps.

# Academic Statement

Attach your code as an appendix and write a statement at the beginning of the assignment asserting that all work is your own work (including code (minus the code we provide)). Adding to this, cite any outside information you reference while working on this assignment and the people you work with.

# SARSA Performance

1 - Implement SARSA to solve this problem. How did the algorithms perform? Include learning curves and plots of the learned value tables.

SARSA FOR 200 EPOCHS

['right', 'right', 'up', 'down', 'down']

['right', 'right', 'down', 'up', 'down']

['down', 'right', 'up', 'up', 'down']

['right', 'left', 'right', 'down', 'right']

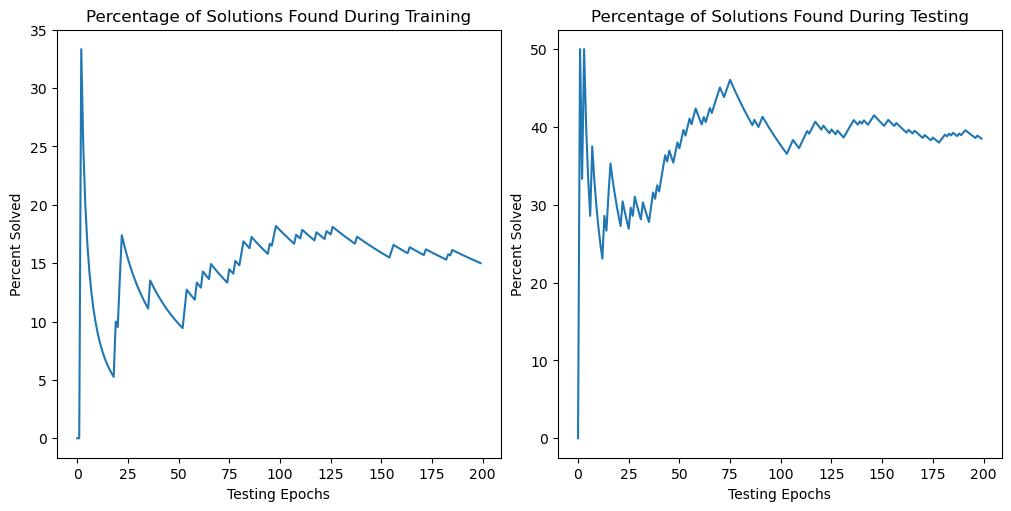
['down', 'down', 'right', 'left', 'left']

['left', 'down', 'up', 'right', 'left']

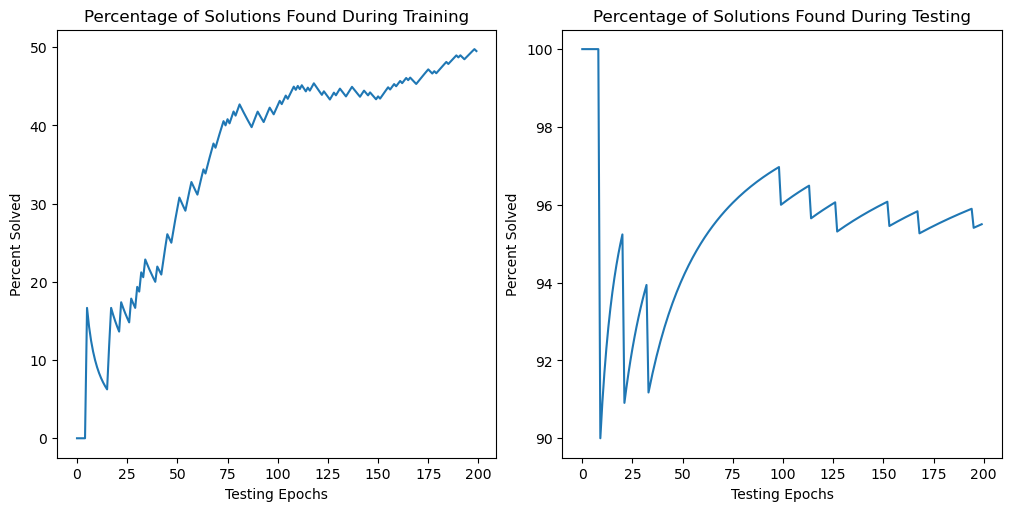
['down', 'right', 'right', 'down', 'down']

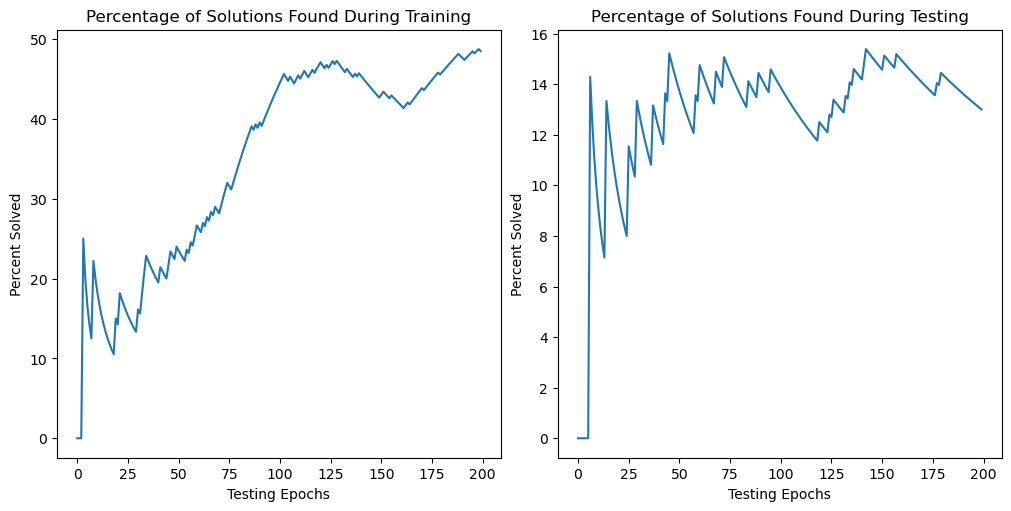
['up', 'up', 'up', 'down', 'down']

['right', 'down', 'down', 'down', 'left']

['down', 'up', 'left', 'left', 'right']

SARSA WITH ALPHA=0.2





Exploitation better for SARSA, exploration better for Q

# Q-Learning Performance

2 - Implement a Q-learning algorithm to solve this problem. How did the algorithms perform? How did solution compare to the SARSA solution? Discuss the implications of your results.

['down', 'down', 'down', 'down', 'left']

['down', 'right', 'down', 'down', 'down']

['right', 'down', 'down', 'down', 'down']

['right', 'down', 'right', 'down', 'down']

['down', 'down', 'right', 'down', 'down']

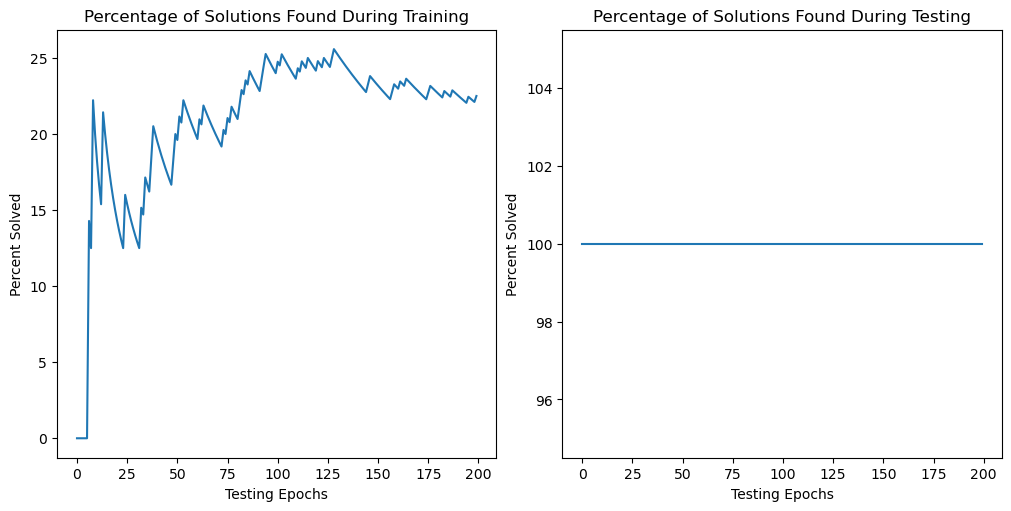
['right', 'right', 'down', 'down', 'down']

['right', 'right', 'right', 'down', 'down']

['up', 'up', 'up', 'down', 'down']

['down', 'down', 'down', 'left', 'left']

['right', 'left', 'left', 'left', 'left']



# Moving Door Evaluation

3 - Now consider the environment where the red door moves randomly by 1 cell every time step. Keep the initial starting location of the door the same as before. Use the EXACT same algorithms from problems 1 and 2 to solve this problem. How does the performance of the agent compare to problems 1 and 2? Does the agent learn a good policy? Describe your results and hypothesize why your agent performs the way it does. Speculate on how you may improve the performance of the agent. Again, plot learning curves and value tables.

# References

[1] Canvas assignment description (maybe)

# Appendix

Link to github code: <INSERT LINK>