Part 1

0.1 Report the policy, value function and number of iterations needed by value iteration when using a tolerance of 0.01 and starting from a value function set to 0 for all states.

```
Values:
[ 60.62388836  66.03486523  71.80422632  77.09196339  59.81429704
  65.18237783  77.83066489  84.14118981  58.09361039   7.98780239
  84.86704922  91.78159355  69.49584217  76.80962081  91.78159355
 100.           0.        ]
Number of iterations:
20
Policy:
[3 3 3 1 3 3 3 1 1 3 3 1 3 3 3 0 0]
```

Policy 4 points, value function 4 points and number of iterations 2 points.

0.2 Report the policy, value function and number of iterations needed by policy iteration to find an optimal policy when starting from the policy that chooses action 0 in all states.

```
Values:
[ 60.63256172  66.03897428  71.8062328   77.09295576  59.81945165
  65.18457679  77.83151901  84.14149059  58.0955782    7.98862928
  84.86730581  91.78165089  69.4968138   76.80991653  91.78165089
 100.           0.        ]
Number of iterations:
5
Policy:
[3 3 3 1 3 3 3 1 1 3 3 1 3 3 3 0 0]
```

Policy 4 points, value function 4 points and number of iterations 2 points.

0.3 Report the number of iterations needed by modified policy iteration to converge when varying the number of iterations in partial policy evaluation from 1 to 10. Use a tolerance of 0.01, start with the policy that chooses action 0 in all states and start with the value function that assigns 0 to all states. Discuss the impact of the number of iterations in partial policy evaluation on the results and relate the results to value iteration and policy iteration.

Number of iterations in partial policy evaluation	Number of iterations until tolerance criterion is satisfied
1	20
2	12
3	9
4	8
5	7
6	7
7	7
8	6
9	6
10	6

5 points for reporting the results.

Discussion

- Increasing the number of steps in partial policy evaluation decreases the number of iterations (number of policy improvement steps), but this also increase the time per iteration. 5 points.
- When the number of steps in partial policy evaluation is small, this approximates value iteration and when it is large, this approximates policy iteration. 5 points.

Part 2

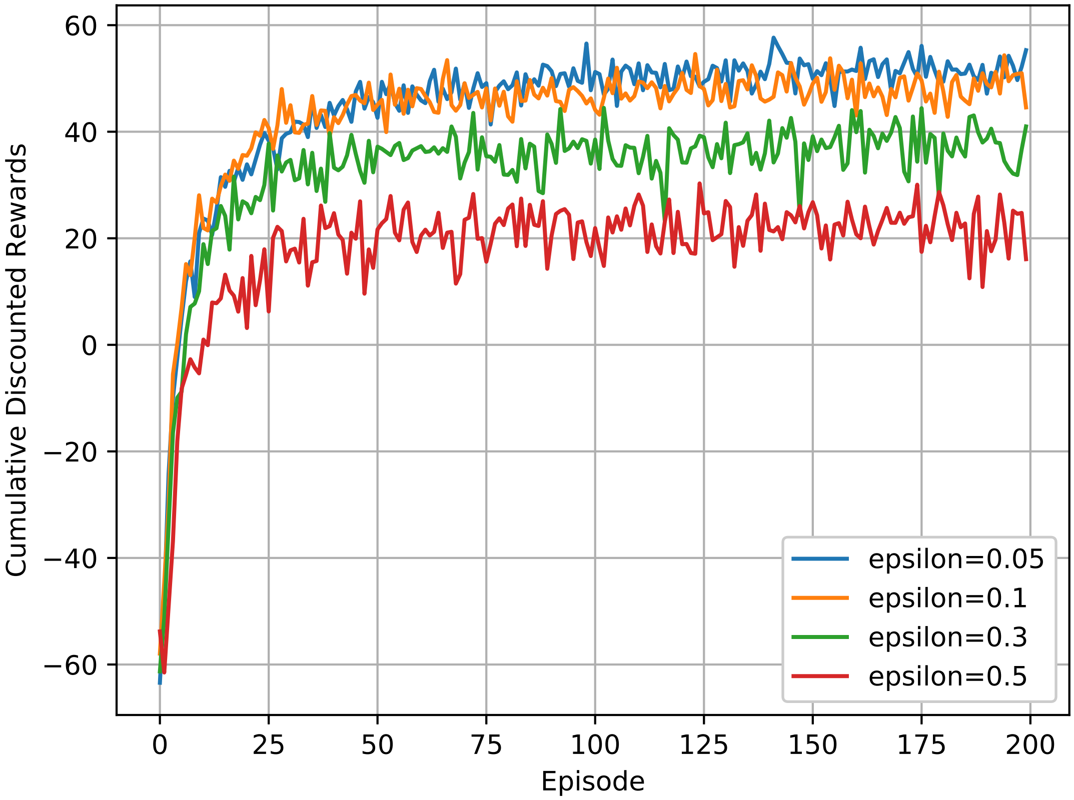

Figure 1: Cumulative discounted rewards vs. episode for various values of ϵ. 15 points.

Discussion

- When epsilon is smaller, the agent exploits more often and therefore earns higher cumulative rewards. 10 points.
- When epsilon is larger, the agent explores more and therefore the cells that are away from the optimal path will be visited more often. When epsilon is smaller, the agent will follow the optimal path more often and therefore will have better Q-value estimates along the optimal path but worse Q-value estimates in the cells that are away from the optimal path. 10 points.
- Note: since epsilon is constant, all states will be visited infinitely often and therefore the Q-values will converge to the optimal Q-function in the long run (regardless of epsilon).

Part 3

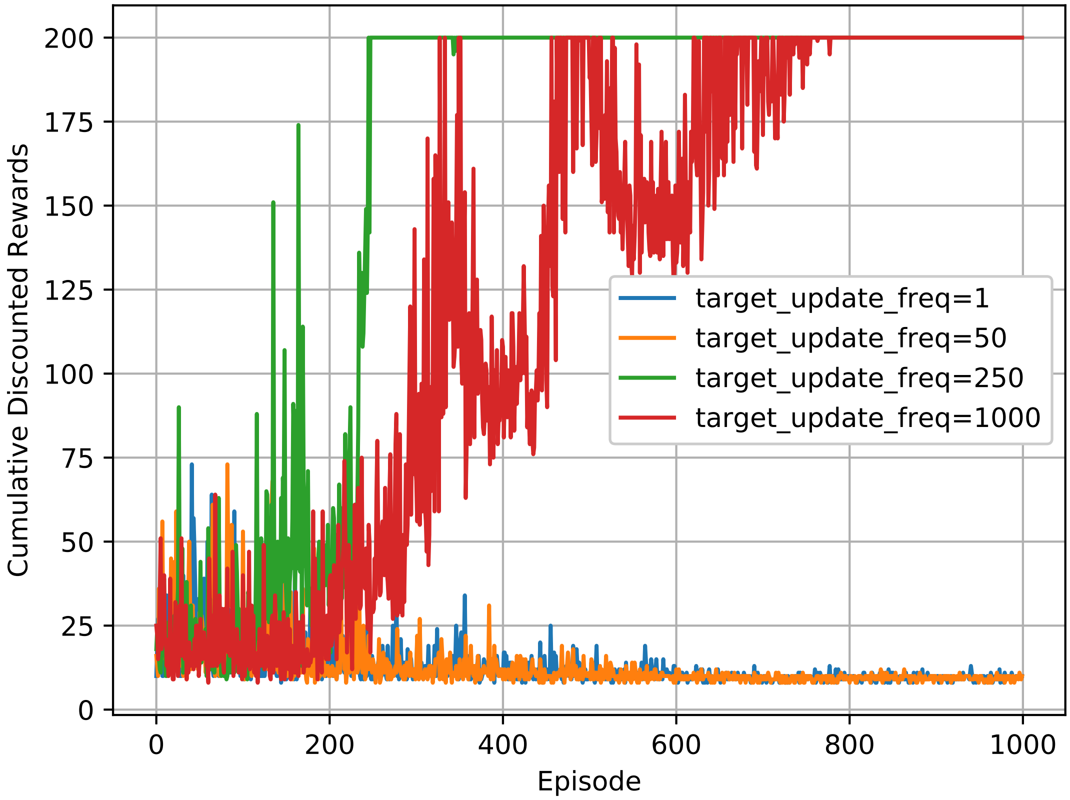

Figure 2: Cumulative discounted rewards vs. episode for various target network update frequencies. 5 points.

Discussion

- Target network stabilizes learning by mitigating divergence. 5 points.
- Using a target network is similar to keeping the next value function fixed in value iteration. 5 points.

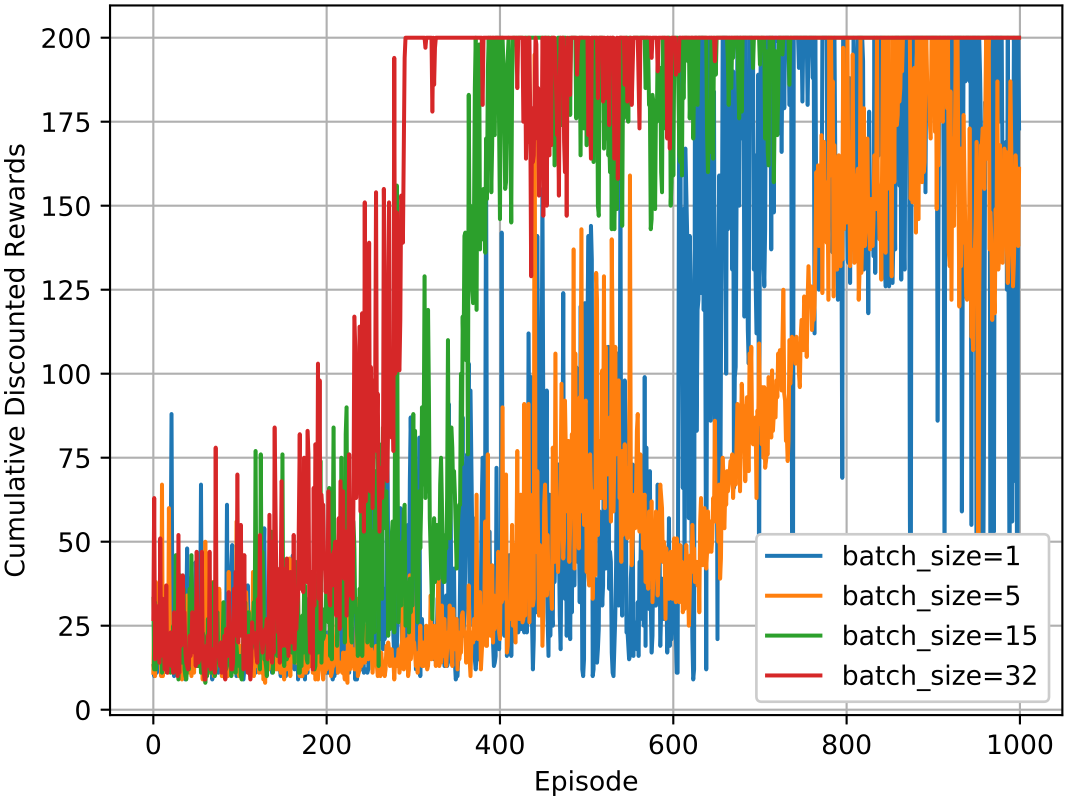

Figure 3: Cumulative discounted rewards vs. episode for various batch sizes. 5 points.

Discussion

- The replay buffer allows multiple steps in the direction of the gradient to be taken and therefore increases convergence by reducing the number of interactions with the environment needed. 5 points.
- When the replay buffer includes all the past experience and we work with batches that include all previous experience, then gradient descent is exact. When the batch does not contain all previous experiences, then gradient descent is stochastic. 5 points.