AE 598 Reinforcement Learning Homework - 1

John S George *
University of Illinois at Urbana-Champaign, Champaign, IL, 61820

The report contains the required plots organized into sections 1 and 2 for the grid world and pendulum problem. Each section is divided into subsections for each algorithm with the corresponding plots.

Nomenclature

Q = Action value

V = State value

π = Policy function

s = State

a = Action

γ = Discount factor

G = Returns

α = Learning rate

I. Grid World

This section contains plots obtained from Policy iteration, Value iteration, SARSA, and Q-learning algorithm for grid world problem. Variables common to all algorithms are:

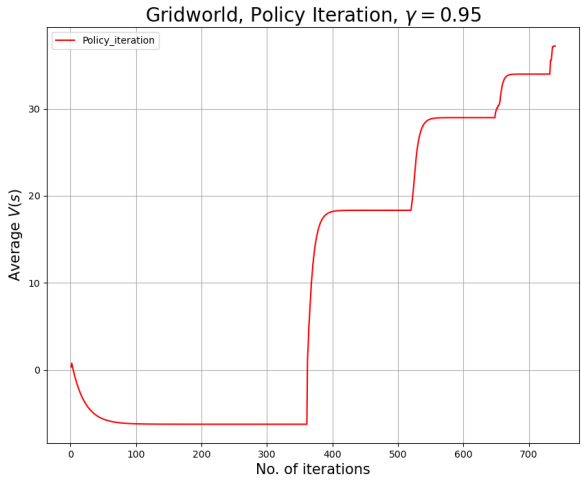
- Discount factor, $\gamma = 0.95$
- Number of actions = 4
- Number of states = 25

A. Policy Iteration

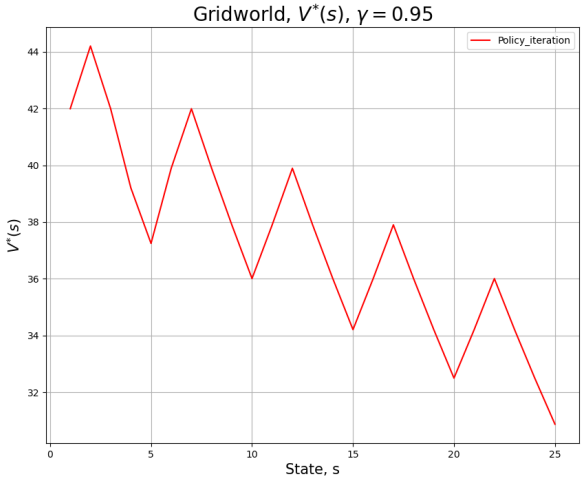
- Maximum iterations = 4000
- Error tolerance = 10^{-8}

Fig.1(a) shows the mean of the value function versus the number of value iterations, Fig.1(b) shows the optimal state value for each state, and Fig.1(c) shows the optimal action at each state. Some states has multiple optimum actions, however, Fig.1(c) shows one case.

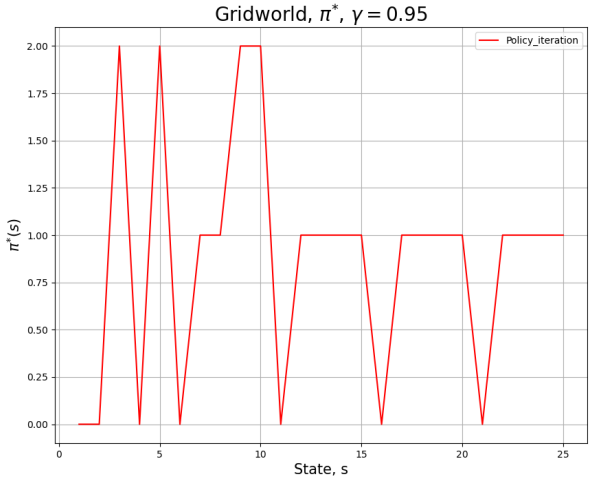
*Graduate Student, Department of Aerospace Engineering, 104 S Wright St, Urbana, IL, 61801



(a) Average state value



(b) Optimal state value at each state



(c) Optimum policy at each state

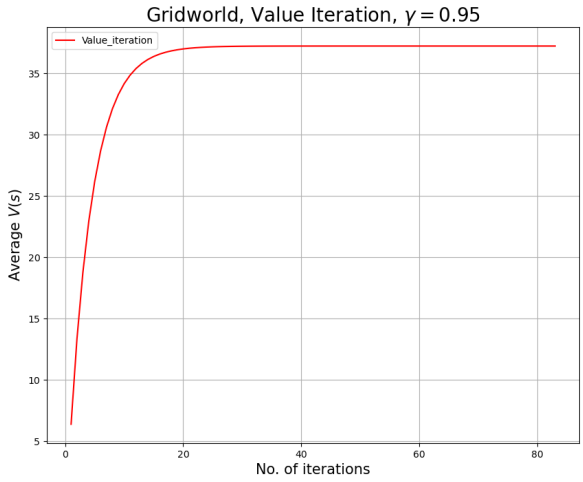
Fig. 1 Policy iteration on grid world problem.

B. Value Iteration

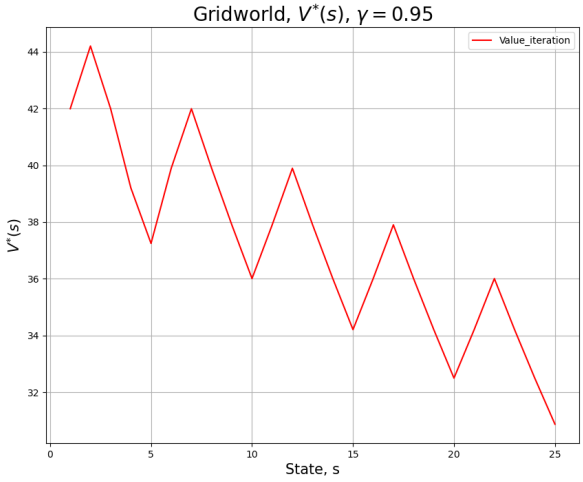
- Maximum iterations = 4000
- Error tolerance = 10^{-8}

Fig.2(a) shows the mean of the value function versus the number of value iterations, Fig.2(b) shows the optimal state value for each state, and Fig.2(c) shows the optimal action at each state. Some states has multiple optimum actions, however, Fig.2(c) shows only one case.

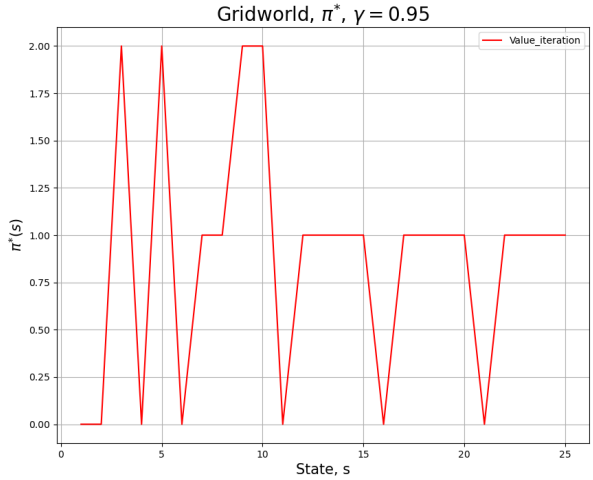
Value iteration converges to optimum state value faster than policy iteration.



(a) Average state value



(b) Optimal state value at each state



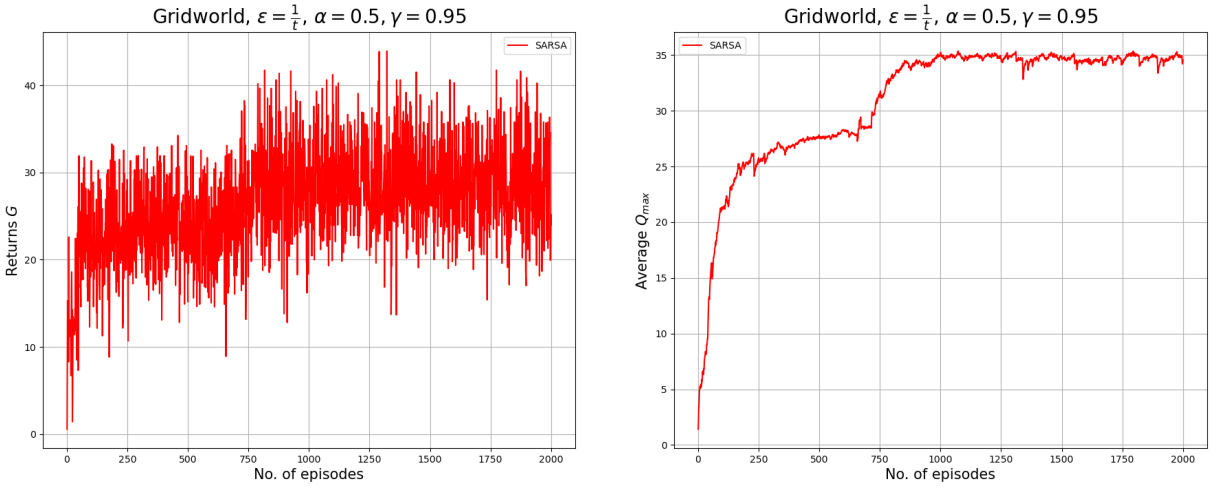
(c) Optimum policy at each state

Fig. 2 Value iteration on grid world problem.

C. SARSA

- Maximum number of episodes = 2000
- Unless specified $\alpha = 0.5$, and $\epsilon = \frac{1}{t}$, where t is the step number, i.e., ϵ-greedy policy is set to prioritize greedy actions towards the end of an episode.

Fig.3(a) shows the returns gained in each episode and Fig.3(b) shows the average of the maximum action value in each episode. Fig.4(a) shows the optimal state value for each state obtained from TD(0) algorithm, and Fig.4(b) shows the optimal action at each state. Some states has multiple optimum actions, however, Fig.4(b) shows only one case. Fig.4(c) is a sample trajectory generated by following the optimal trajectory obtained from SARSA.

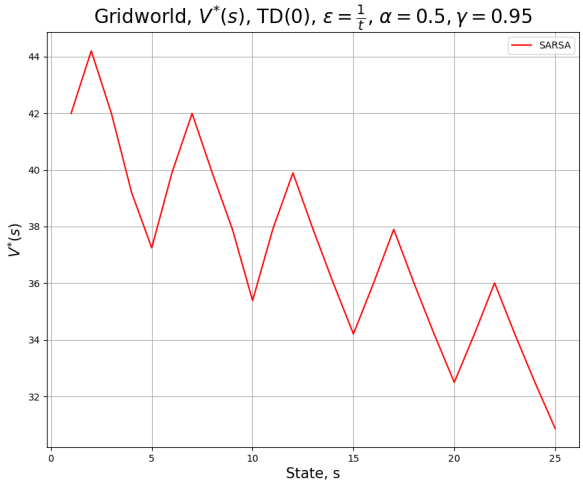


(a) Returns over episodes **(b) Average Q_{max} over episodes**

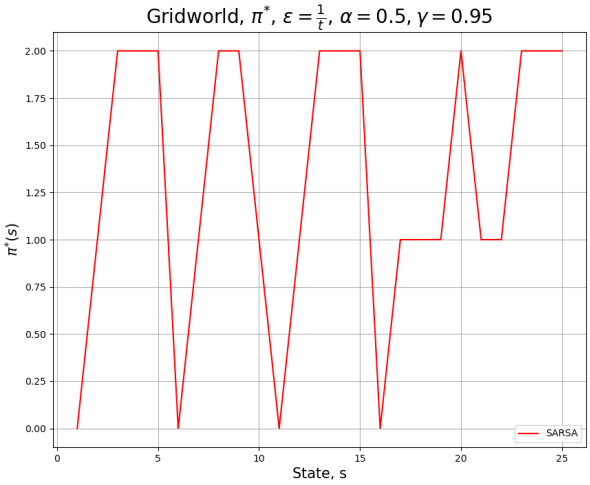
Fig. 3 Learning curves for SARSA

Fig.5(a) shows the learning curves for varying ϵ, and a fixed learning rate, $\alpha = 0.5$. Maximum action value was obtained for a decaying ϵ as it prioritizes exploration at the beginning of an episode and exploitation towards the end. It can also be observed that for smaller ϵ the returns and action value is larger, as smaller ϵ facilitates greedier actions. It can also be observed that the average Q_{max} obtained by SARSA is slightly lower than the those obtained from Policy iteration, Value iteration and Q-learning. This is because SARSA converges to the optimal Q-value function in the space of ϵ-greedy policy only (assuming each state action pair is visited infinite times).

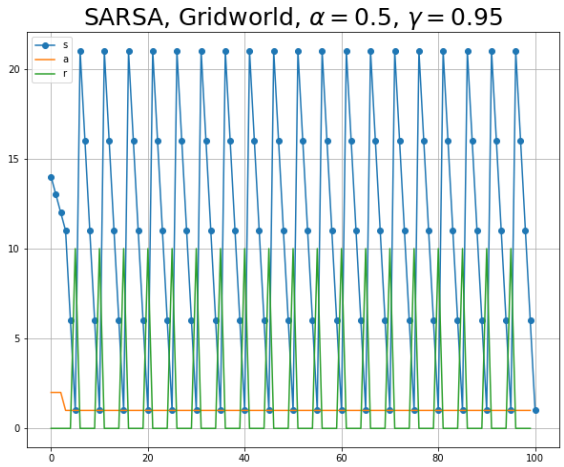
Fig.5(b) shows the learning curves for varying α and a decaying ϵ. It can be observed that extreme values of α often lead to slower convergence. For larger values of α, the updates are large and hence the slower learning rate, while for smaller values of , the updates are small leading to slower convergence.



(a) Optimal state value

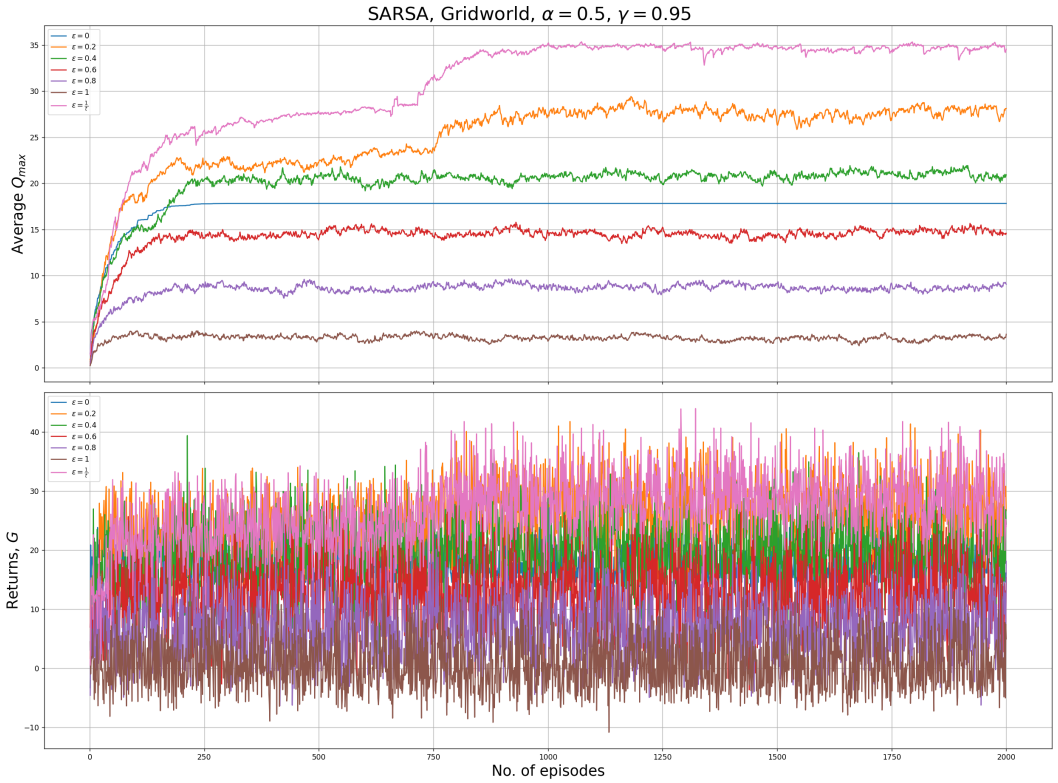


(b) Optimal policy

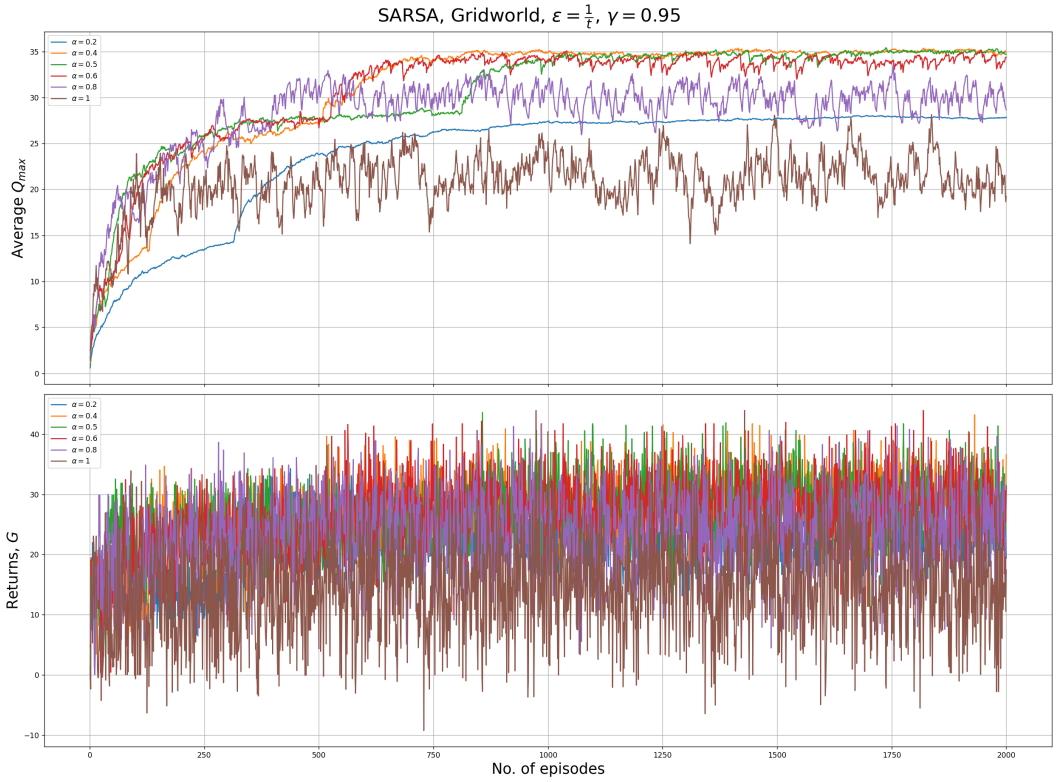


(c) Sample trajectory

Fig. 4 SARSA



(a) Learning curves for varying ϵ



(b) Learning curves for varying α

Fig. 5 Learning curves for varying α and ϵ for SARSA

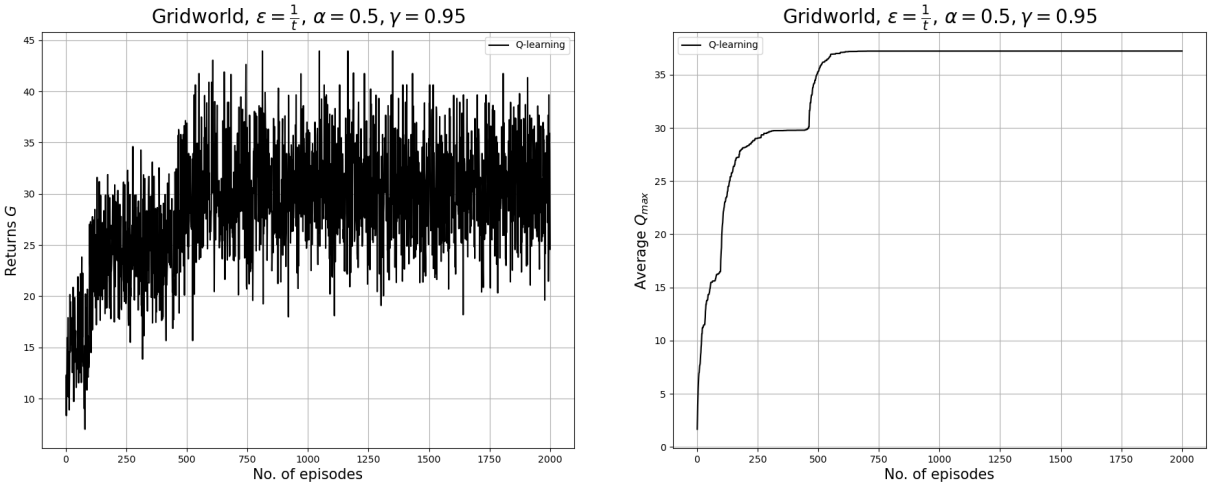
D. Q-learning

- Maximum number of episodes = 2000
- Unless specified $\alpha = 0.5$ and $\epsilon = \frac{1}{t}$, where t is the step number, i.e., ϵ-greedy policy is set to prioritize greedy actions towards the end of an episode.

Fig.6(a) shows the returns gained in each episode and Fig.6(b) shows the average of the maximum action value in each episode. Fig.7(a) shows the optimal state value for each state obtained from TD(0) algorithm, and Fig.7(b) shows the optimal action at each state. Some states have multiple optimum actions, however, Fig.7(b) shows only one case. Fig.7(c) is a sample trajectory generated by following the optimal trajectory obtained from SARSA.

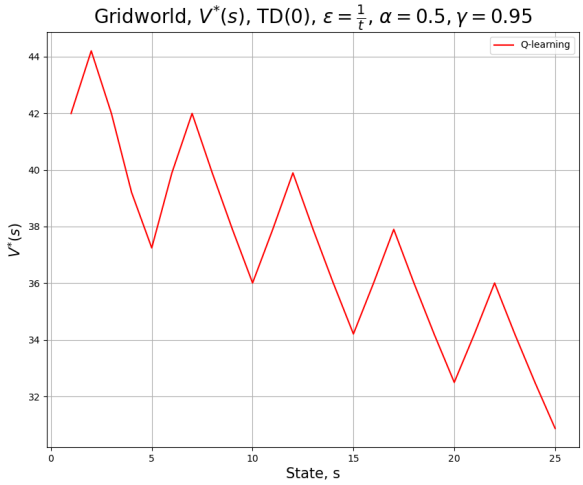
Fig.8(a) shows the learning curves for varying ϵ, and a fixed learning rate, $\alpha = 0.5$. Faster convergence can be observed for larger values of ϵ. This is because Q-learning algorithm is inherently greedy in nature and will converge to optimal Q under the optimal policy (under certain assumptions), so a larger value of ϵ will help in exploration and will augment the inherent greedy nature of the algorithm. The returns G, are also larger for smaller ϵ and decaying ϵ. The mean value of Q_{max} obtained from Q-learning algorithm is comparable to that of Value iteration and Policy iteration as all these algorithms converge to the optimal action-value function under the optimal policy.

Fig.8(b) shows the learning curves for varying α for a decaying ϵ. Better convergence is observed for moderate values of α.

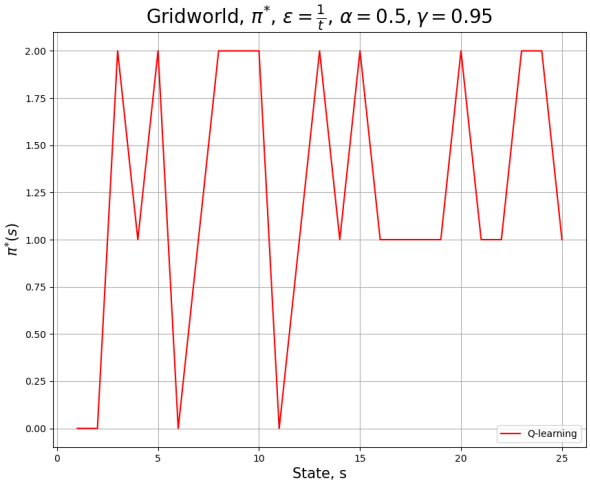


(a) Returns over episodes **(b) Average Q_{max} over episodes**

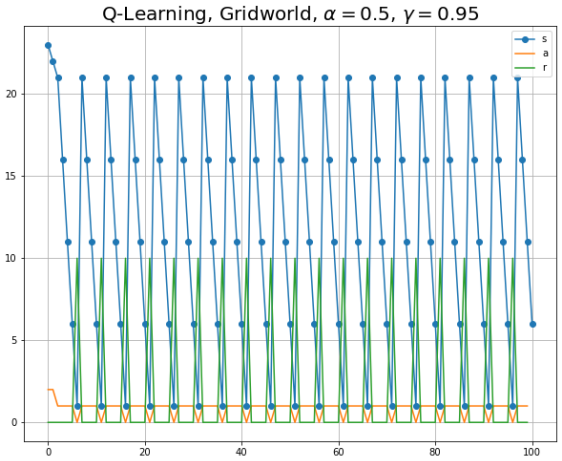
Fig. 6 Learning curves for Q-learning



(a) Optimal state value

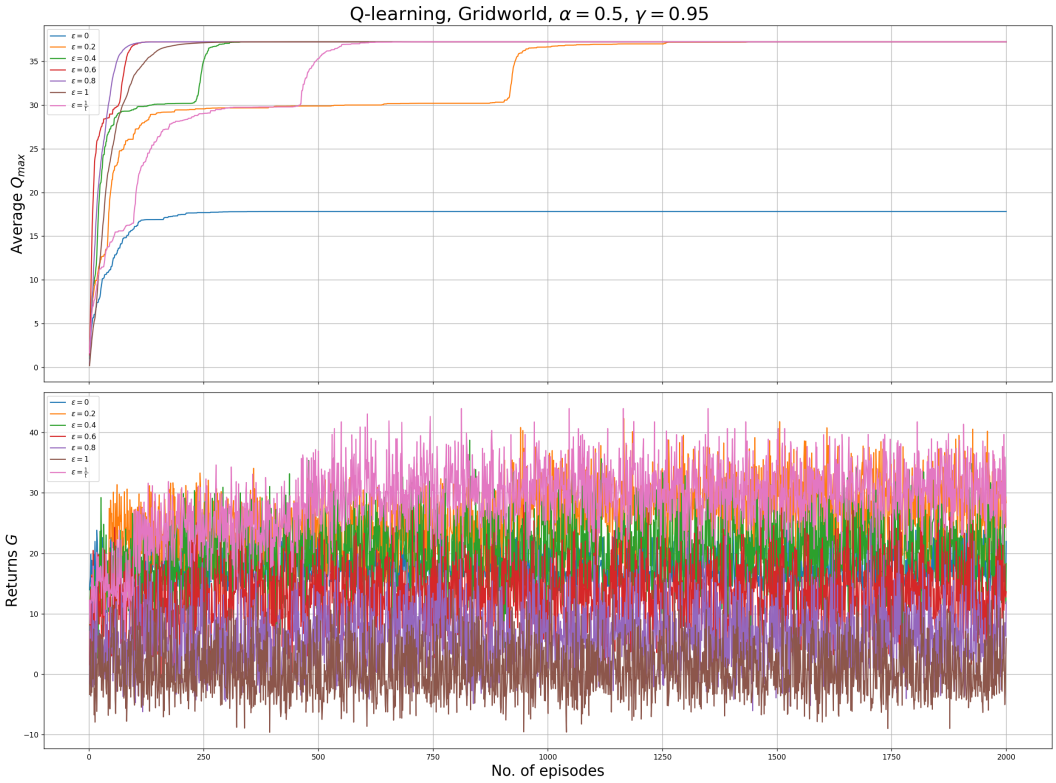


(b) Optimal policy

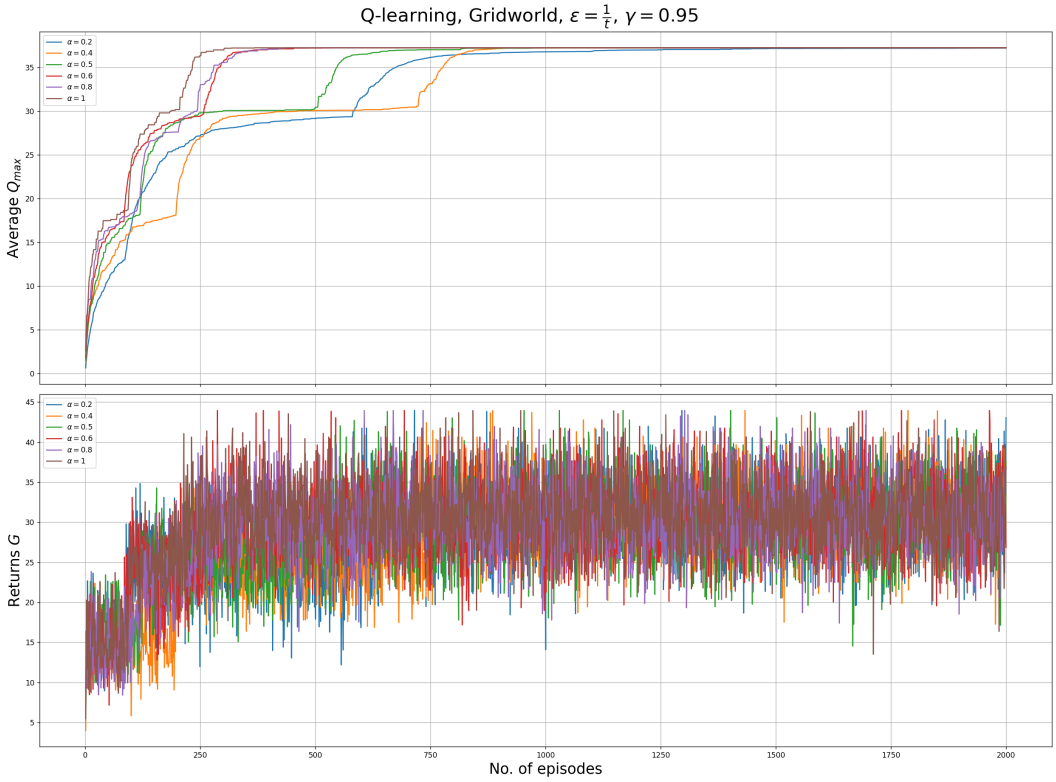


(c) Sample trajectory

Fig. 7 Q-learning



(a) Learning curves for varying ϵ



(b) Learning curves for varying α

Fig. 8 Learning curves for varying α and ϵ for Q-learning

II. Discrete Pendulum

This section contains plots obtained from SARSA, and Q-learning algorithms for discrete pendulum problem. Variables common to all algorithms are:

- Discount factor, $\gamma = 0.95$
- Number of actions = 21
- Number of states = 441 (21 x 21)

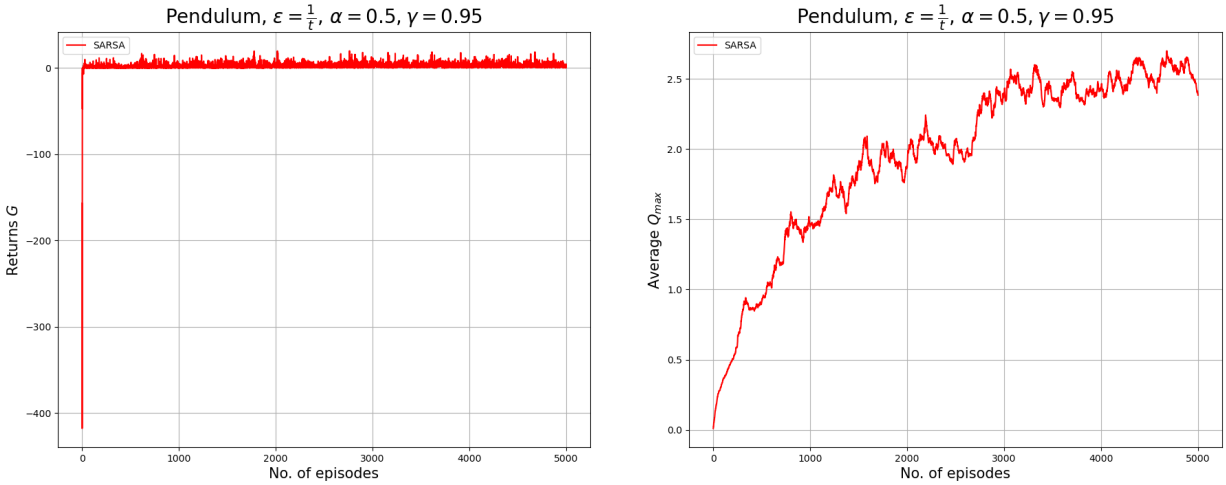
A. SARSA

- Maximum number of episodes = 5000
- Unless specified $\alpha = 0.5$, and $\epsilon = \frac{1}{t}$, where t is the step number, i.e., ϵ-greedy policy is set to prioritize greedy actions towards the end of an episode.

Fig.9(a) shows the returns gained in each episode and Fig.9(b) shows the average of the maximum action value in each episode. Fig.10(a) shows the optimal state value for each state obtained from TD(0) algorithm, and Fig.10(b) shows the optimal action at each state. Fig.10(c) is a sample trajectory generated by following the optimal trajectory obtained from SARSA.

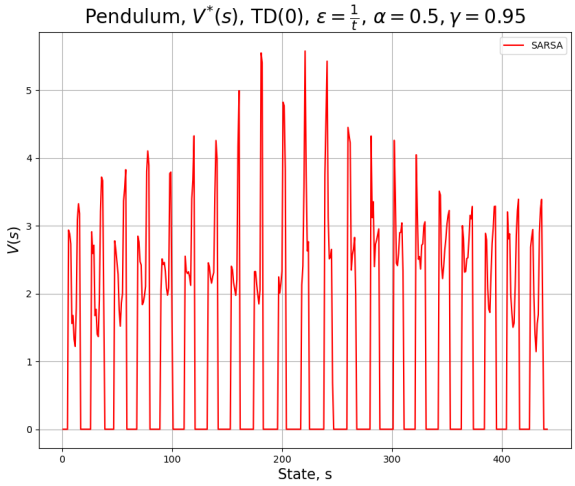
Fig.11(a) shows the learning curves for varying ϵ and a fixed learning rate, $\alpha = 0.5$. Maximum action value was obtained for a decaying ϵ, and for smaller ϵ the action value is larger. The average Q_{max} obtained by SARSA is slightly lower than the those obtained from Q-learning as SARSA converges to the optimal Q-value function in the space of ϵ-greedy policy only (assuming each state action pair is visited infinite times).

Fig.11(b) shows the learning curves for varying α and a decaying ϵ. Moderate values of α provides larger average Q_{max}. The trends observed in SARSA algorithm for pendulum problem is similar to that of grid world problem.

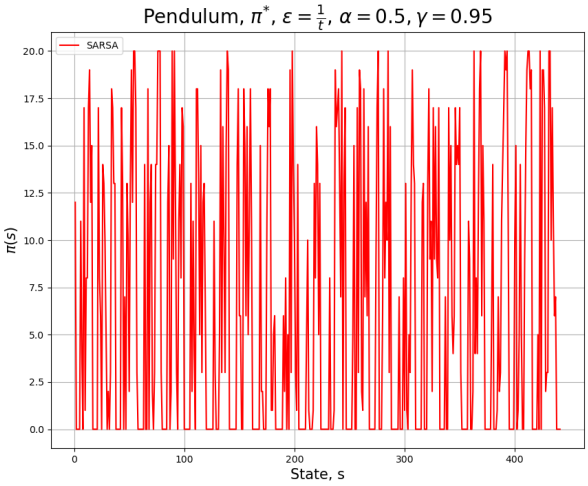


(a) Returns over episodes **(b) Average Q_{max} over episodes**

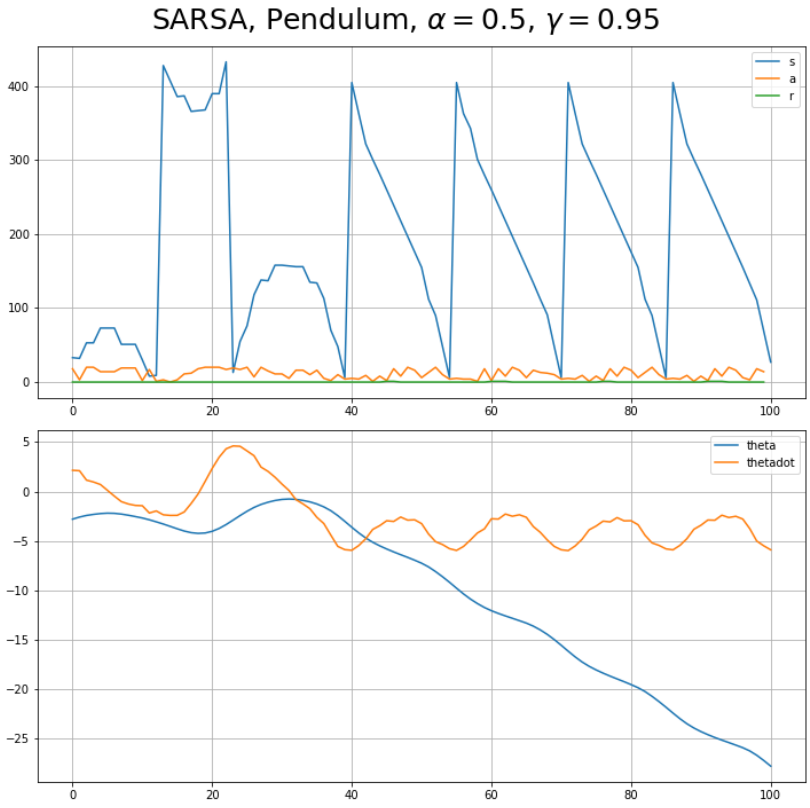
Fig. 9 Learning curves for SARSA



(a) Optimal state value

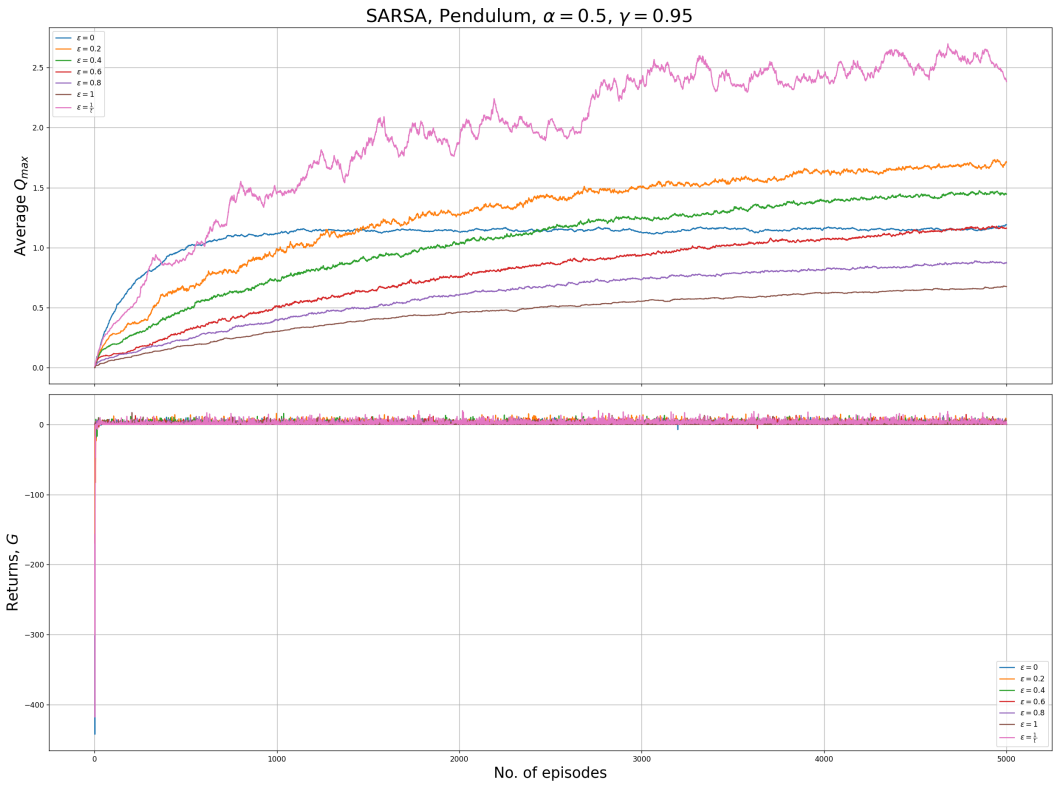


(b) Optimal policy

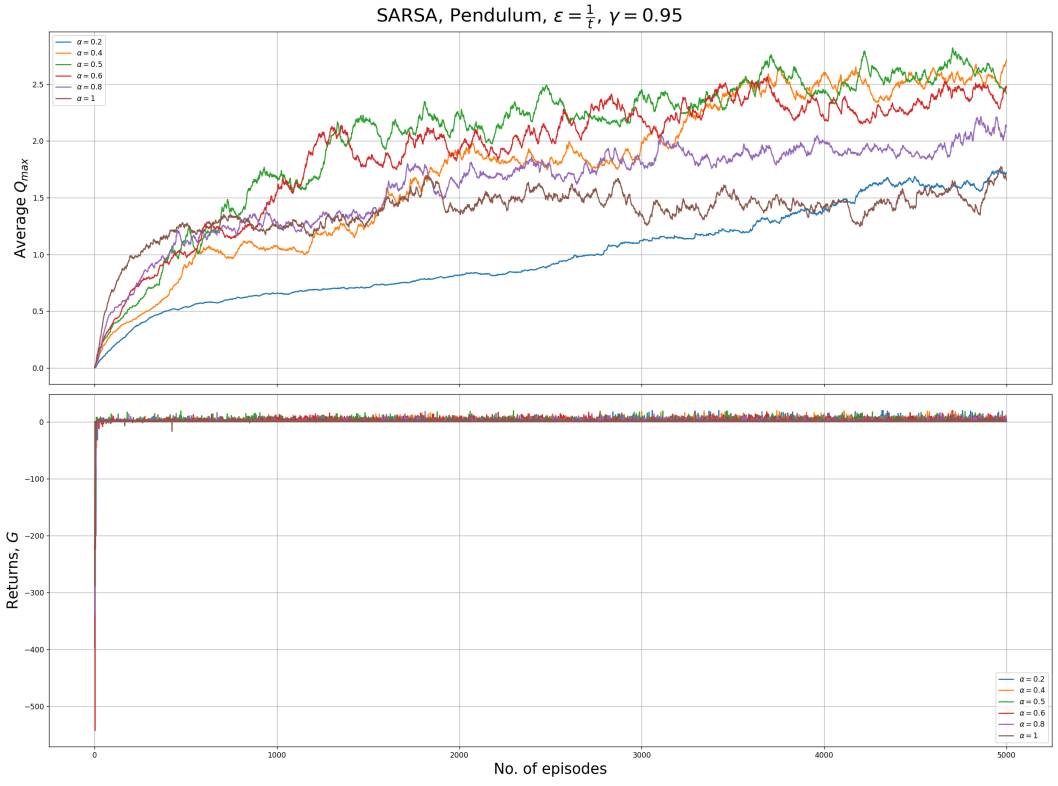


(c) Sample trajectory

Fig. 10 SARSA



(a) Learning curves for varying ϵ



(b) Learning curves for varying α

Fig. 11 Learning curves for varying α and ϵ for SARSA

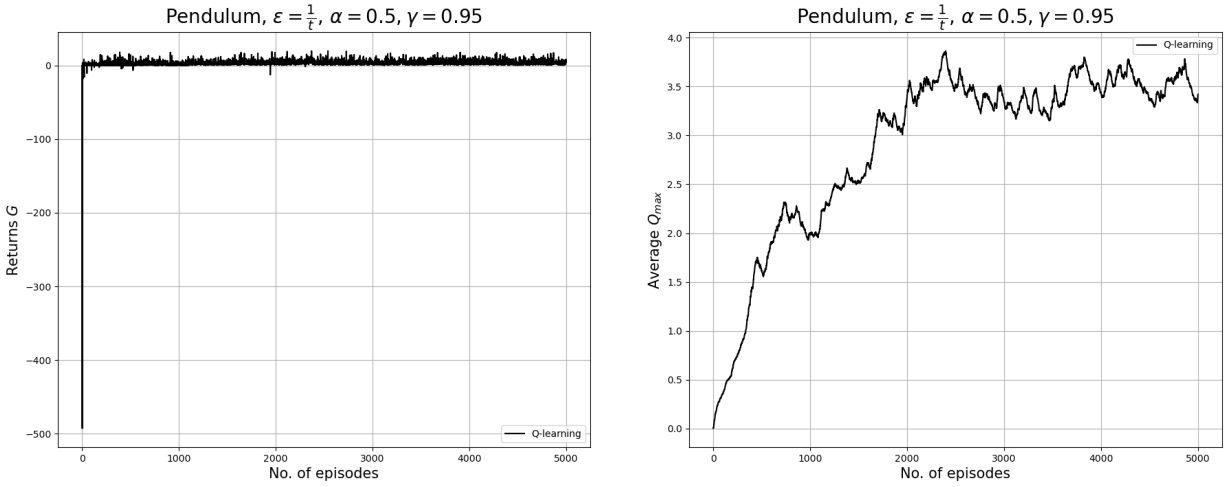
B. Q-learning

- Maximum number of episodes = 5000
- Unless specified $\alpha = 0.5$ and $\epsilon = \frac{1}{t}$, where t is the step number, i.e., ϵ-greedy policy is set to prioritize greedy actions towards the end of an episode.

Fig.12(a) shows the returns gained in each episode and Fig.12(b) shows the average of the maximum action value in each episode. Fig.13(a) shows the optimal state value for each state obtained from TD(0) algorithm, and Fig.13(b) shows the optimal action at each state. Fig.13(c) is a sample trajectory generated by following the optimal trajectory obtained from Q-learning.

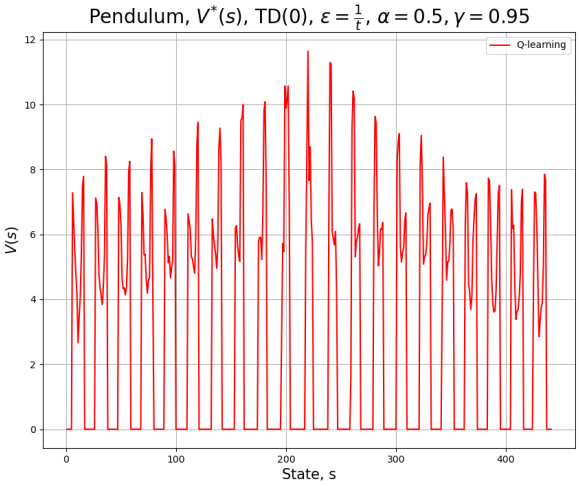
Fig.14(a) shows the learning curves for varying ϵ and a fixed learning rate, $\alpha = 0.5$. Larger average Q_{max} can be observed for larger values of ϵ.

Fig.14(b) shows the learning curves for varying α and a decaying ϵ. Larger average Q_{max} is observed for moderate values of α.

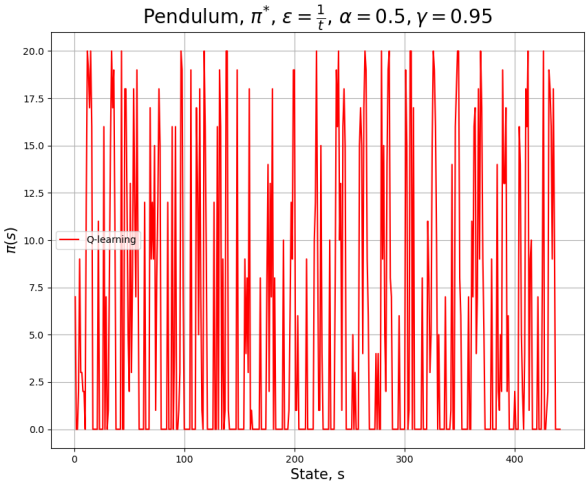


(a) Returns over episodes **(b) Average Q_{max} over episodes**

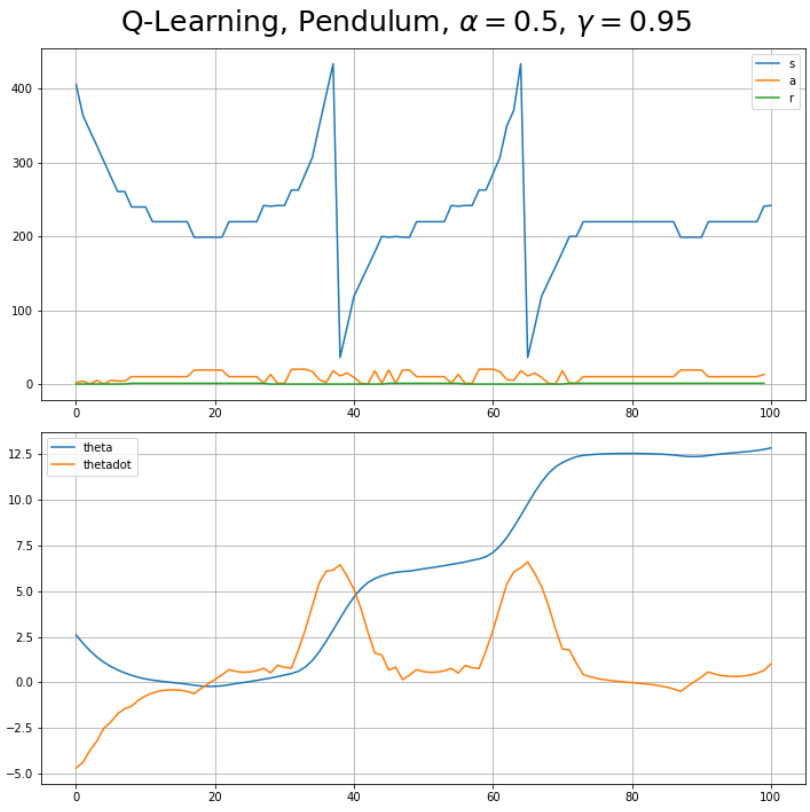
Fig. 12 Learning curves for Q-learning



(a) Optimal state value

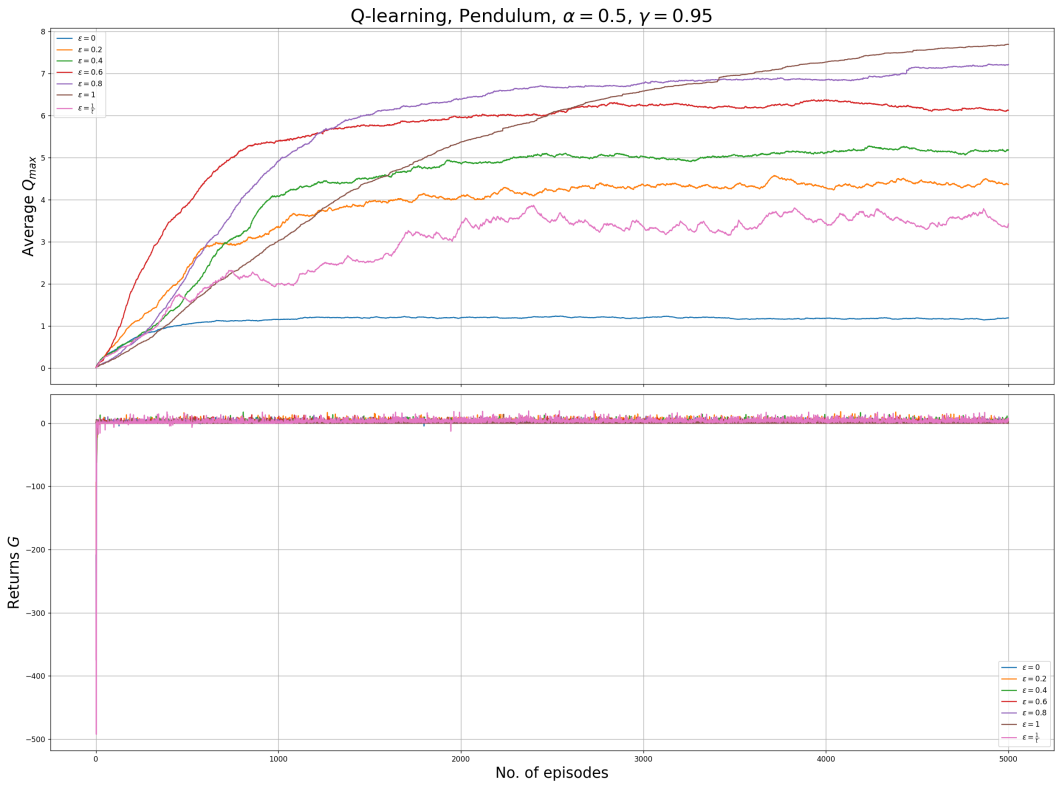


(b) Optimal policy

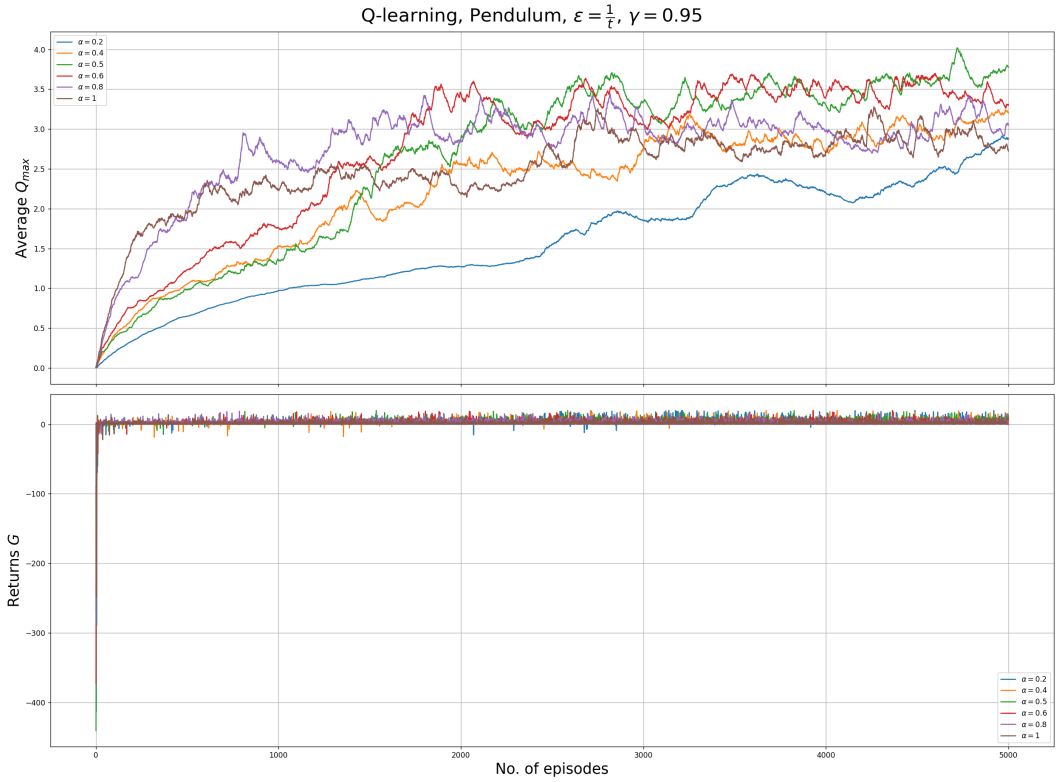


(c) Sample trajectory

Fig. 13 Q-learning



(a) Learning curves for varying ϵ



(b) Learning curves for varying α

Fig. 14 Learning curves for varying α and ϵ for Q-learning