INGENIEURSWETENSCHAPPEN

FACULTY OF INGENIEURSWETENSCHAPPEN
Applied Computer Science

Multi-Armed Bandits and Stochastic Reward Game

Computational Game and Theory

Yanfang GUO

Enroll number: 0535945

Prof T. Lenaerts (ULB) and A. Nowé (VUB):

December 2017

1 N-Armed Bandit

The last student number of mine is 5, so the reward distribution is shown in the table 1(*Here the action number starts from 0, not 1.*)

Table 1: Reward distributions for section 1.1

Action	μ	σ
action 0	2.1	1.2
action 1	1.1	0.8
action 2	0.7	2
action 3	1.9	0.9

1.1 N-Armed Bandit plots

For each algorithm, run it 500 times and average the rewards and standard derivations. The figure 1 shows the rewards received, and the Figure 2 shows the standard derivations for 500 trials in each step.

From the Figure 1, the **ϵ-greedy algorithm** with $\epsilon = 0.1$ gives the best result, next is ϵ= 0.01, then **softmax** $\tau = 1$. The $\epsilon = 0$ plays slightly better than $\tau = 0.1$. and the random exploration gives the worst result.

From the Figure 2, we could find that the **softmax** with $\tau = 1, \epsilon = 0.1$ give smaller standard derivations.

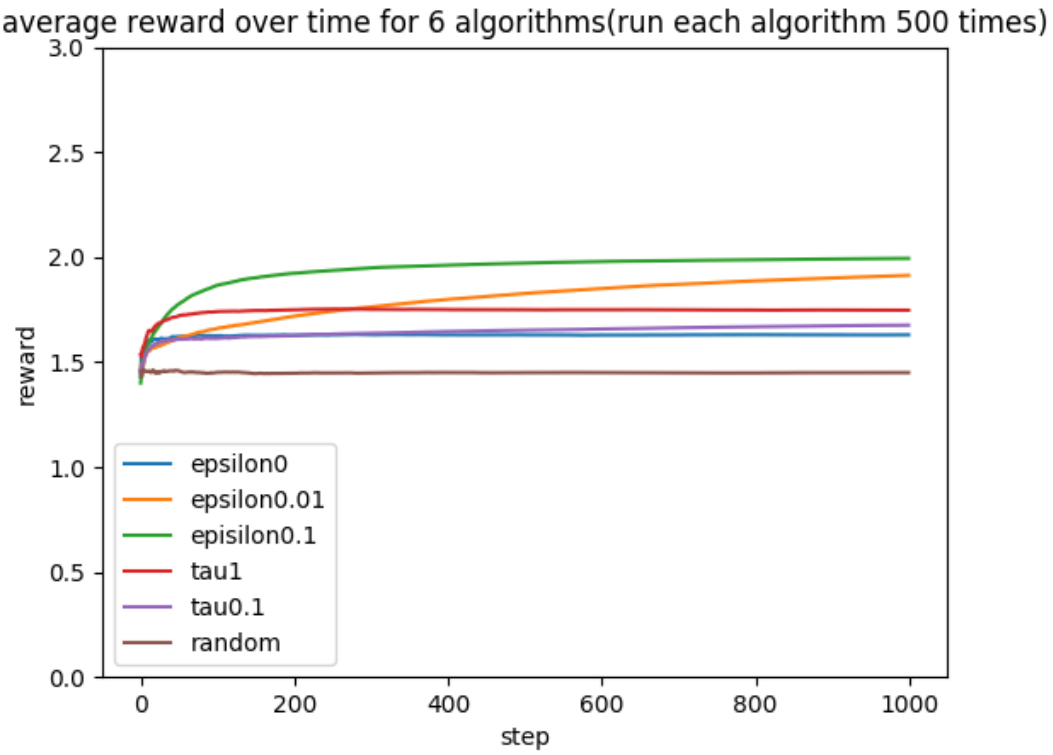

Figure 1: The average reward for 6 algorithms (500 trials)

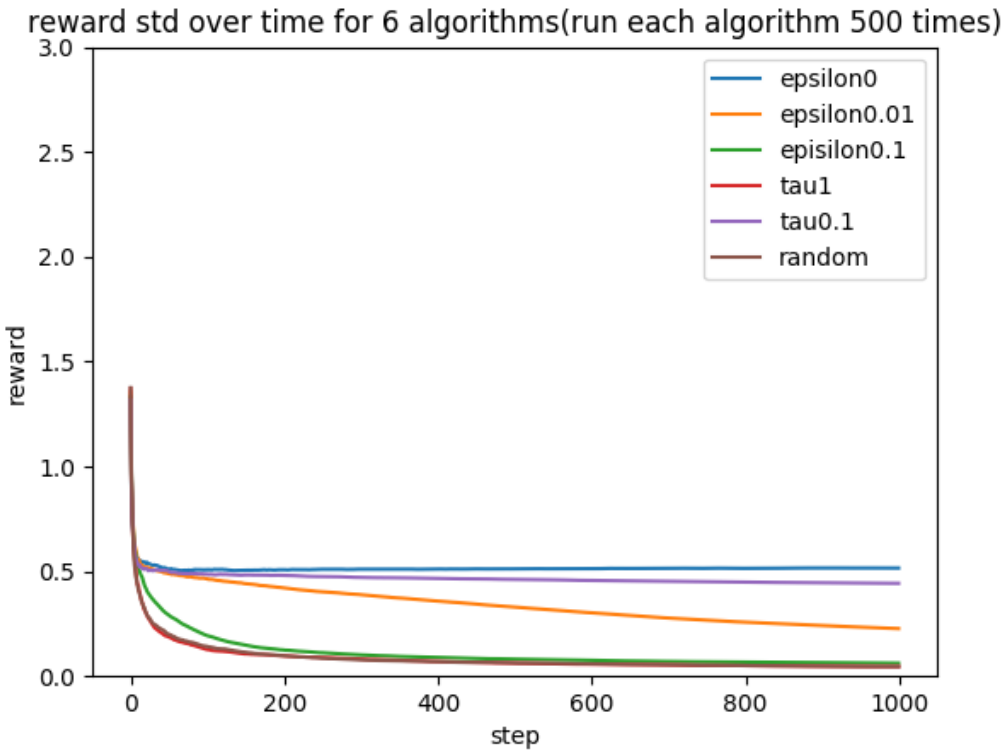

Figure 2: The reward standard derivation for 6 algorithms (500 trials)

Plots for each action: Figure 3 gives the $Q^*(a)$ for all actions and Figure 4 gives the corresponding standard derivation. Figure 5 shows the total action selected times for 6 algorithms running 500 times.

Findings and Conclusions

- From the Figure 3, we could notice that the $\epsilon = 0.1, 0$ and $\tau = 0.1$ did not learn the $Q^*(a)$ well, it still has difference with the mean value of the expected reward.
- From the Figure 5, comparing the subgraph where $\epsilon = 0, 0.01, 0.1$, we could find the larger ϵ leads the larger proportion of choosing action 0.
- From the Figure 5, comparing the subgraph where $\tau = 1, 0.1$, we could find the larger τ leads the larger proportion of choosing action 0.
- The more random explorations, the smaller differences between $Q^*(a)$ and related μ, which means the agents learn the $Q^*(a)$ for each action.

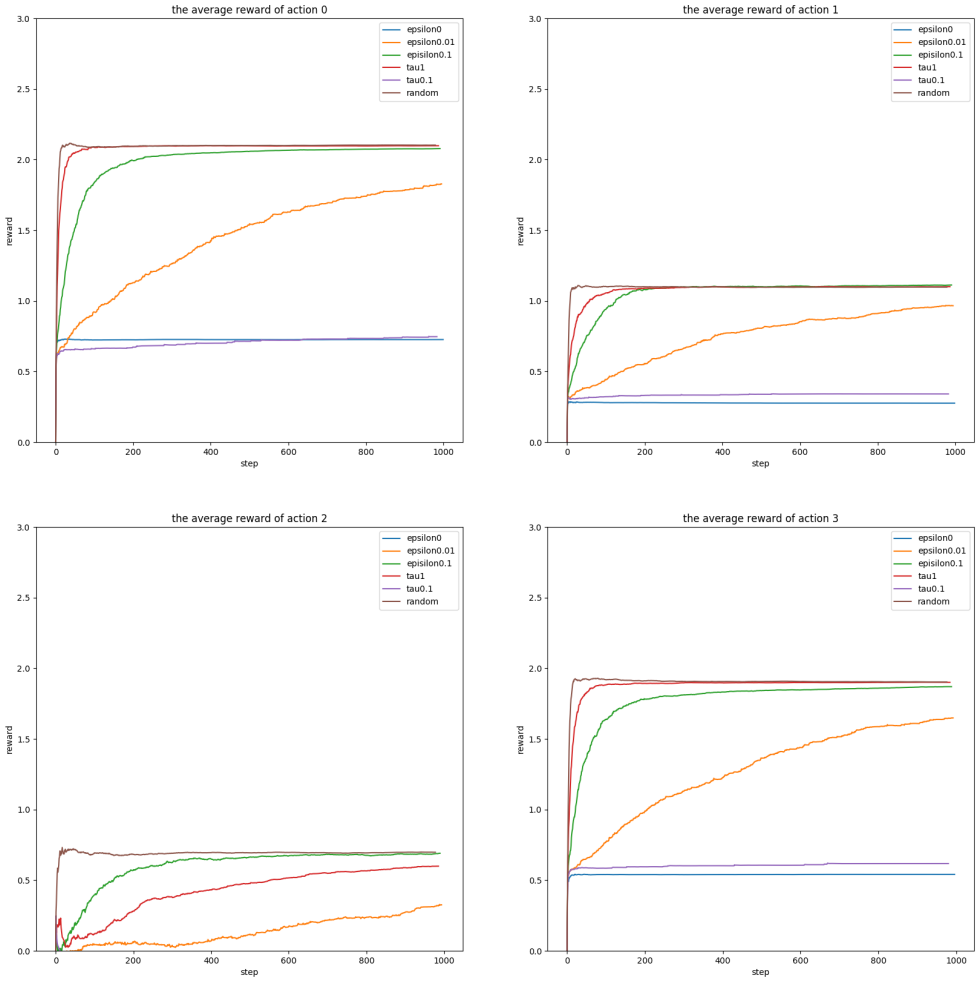

Figure 3: The reward standard derivation for 6 algorithms (500 trials)

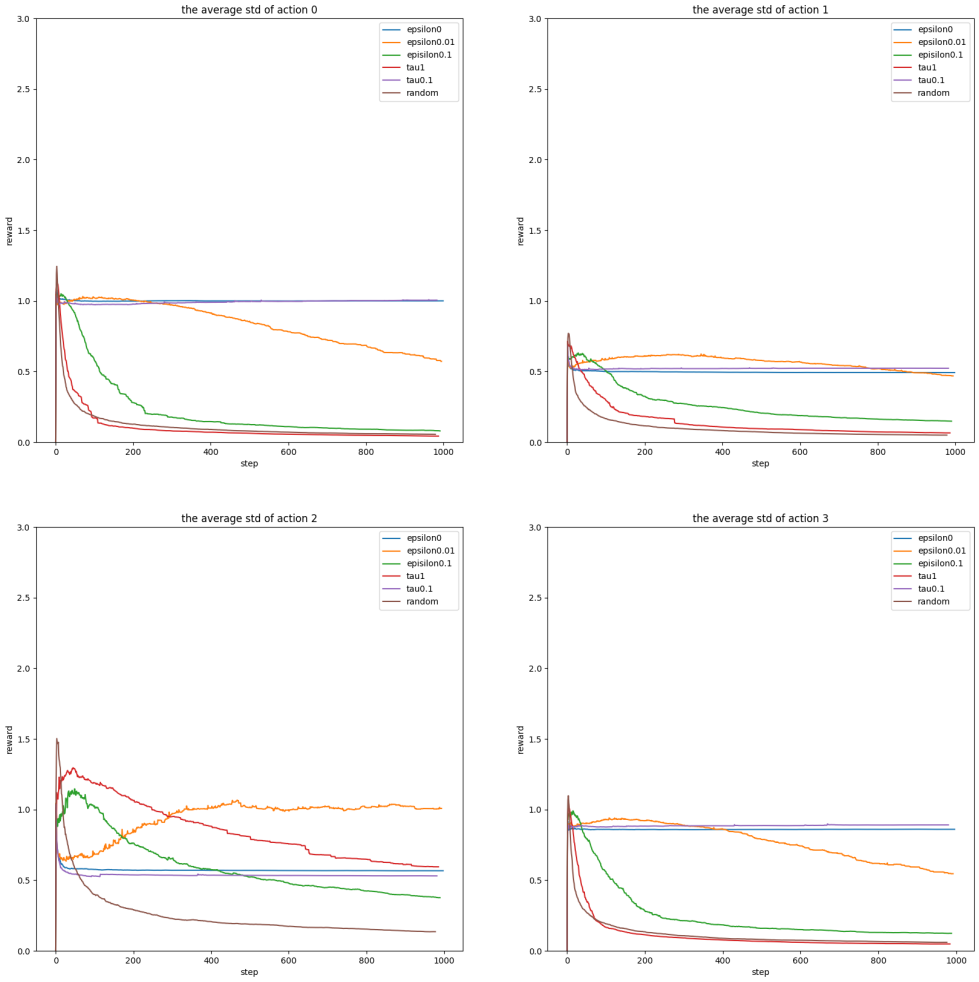

Figure 4: The reward standard derivation for 6 algorithms (500 trials)

The total action selection is 1000*500 = 500000.

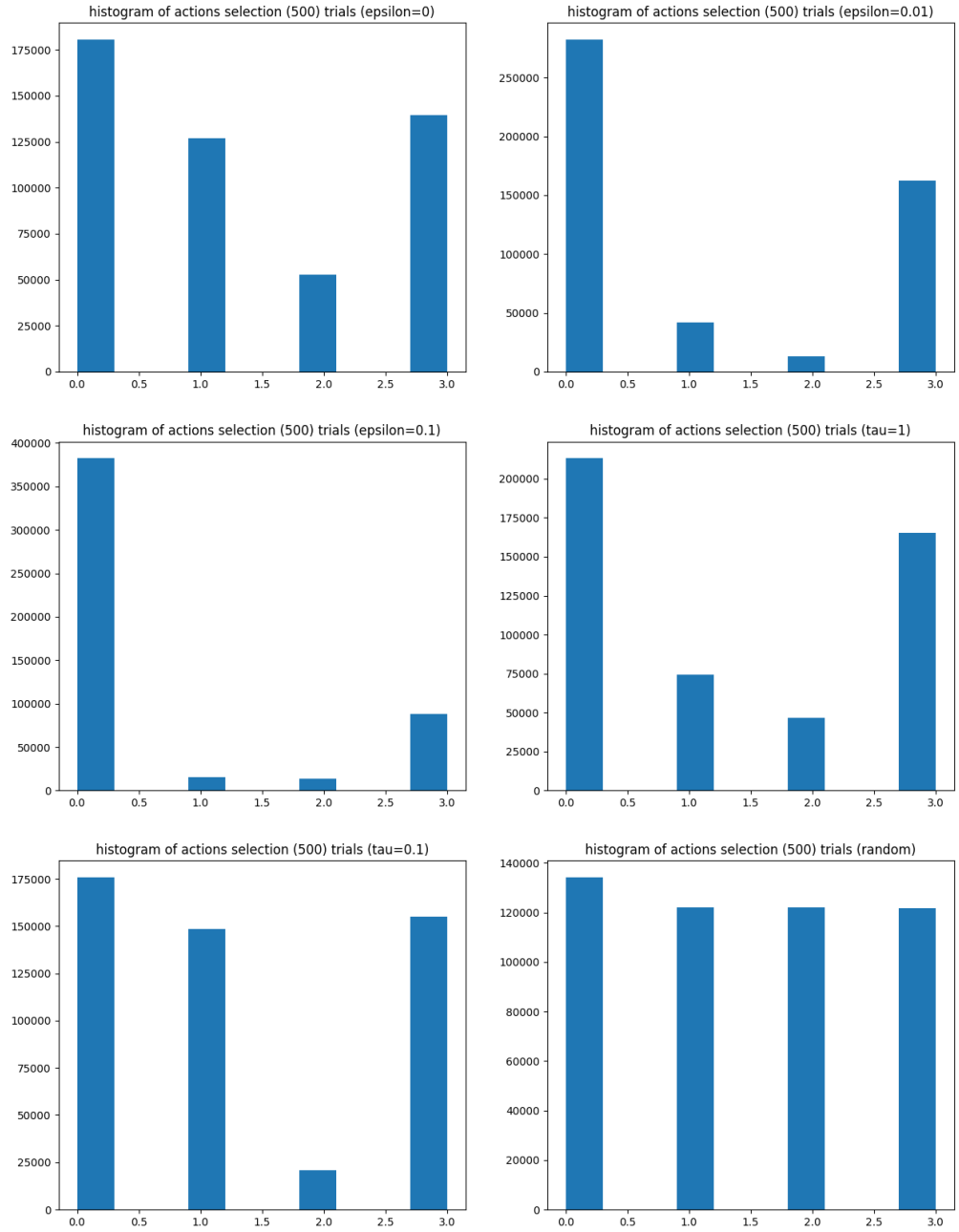

Figure 5: The total action selected times for 6 algorithms (500 trials)

1.2 Doubling the standard derivation

We also explore actions that have higher σ, we double the σ and the reward distributions are show in table 2.

Table 2: Reward distributions for section 1.1

Action	μ	σ
action 0	2.1	2.4
action 1	1.1	1.6
action 2	0.7	4
action 3	1.9	1.8

1.3 Plots for this sections

Run each algorithms for 500 times, and then average the results per step.

Figure 6 shows the average rewards received for these 6 algorithms. Compare that with Figure 1, we can found that the **greedy search** $\epsilon = 0.1$ gives better result, then $\epsilon = 0.01$, comes after **softmax** with $\tau = 1$, while the **softmax** plays slightly better than that of $\epsilon = 0$, random selection still gives the worst performance. This presents the same tendency with that of Figure 1.

Compare the Figure 2 and 7, the larger reward standard derivation distributions gives similar overall standard derivations.

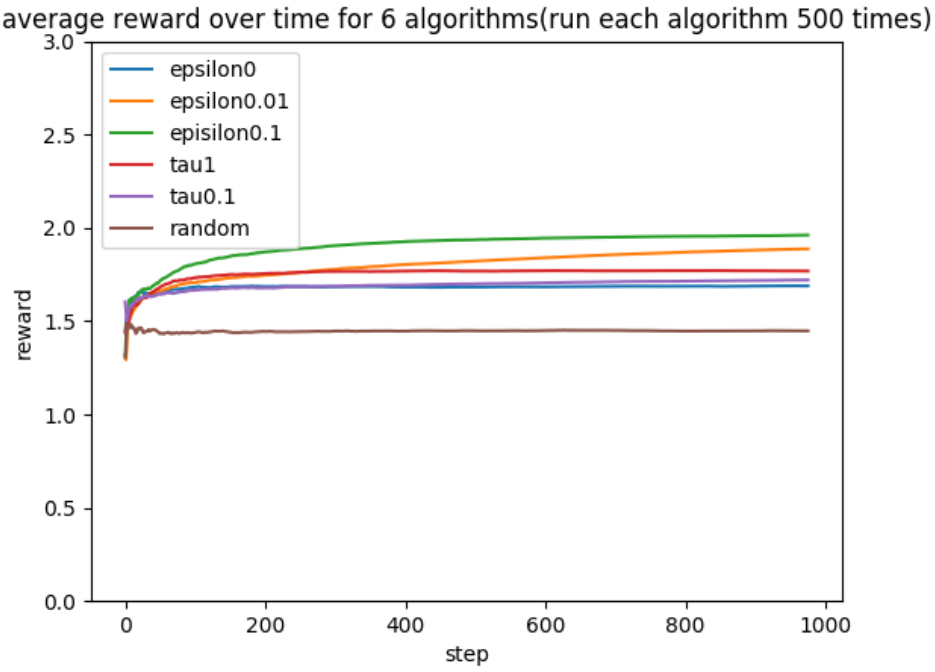

Figure 6: The average reward for 6 algorithms (500 trials)

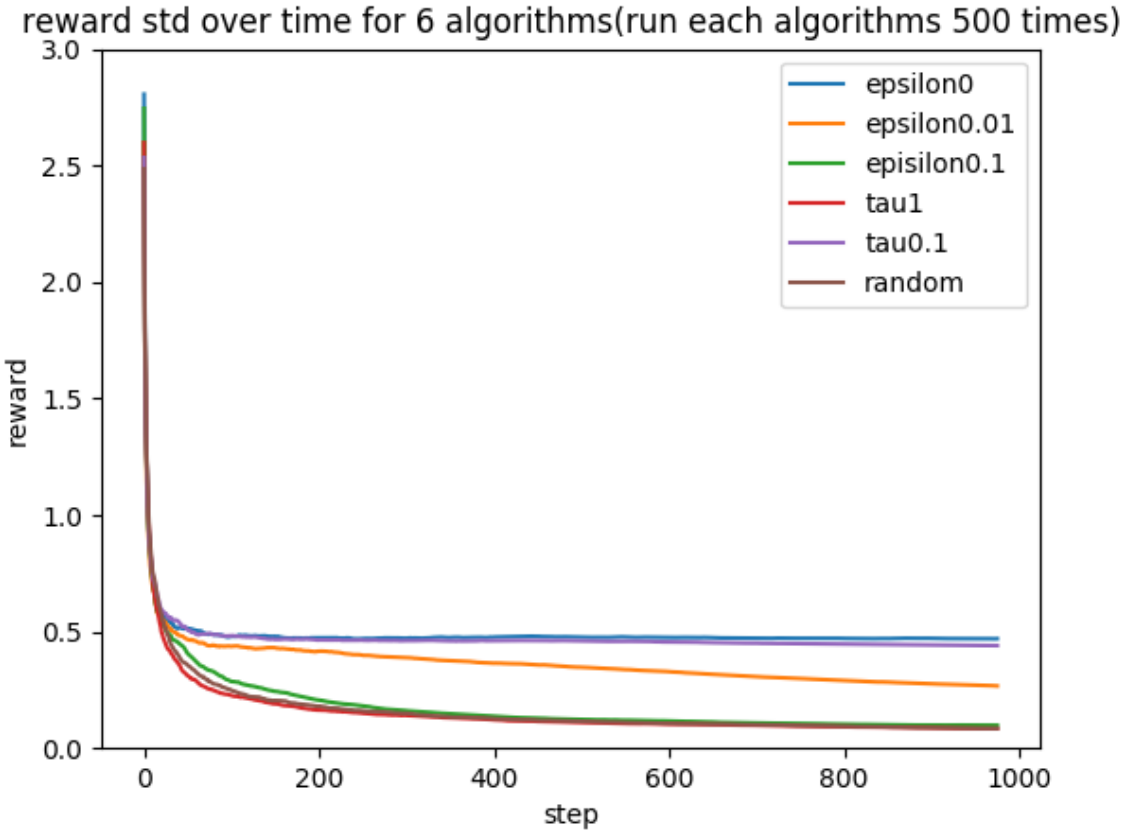

Figure 7: The reward standard derivation for 6 algorithms (500 trials)

Figure 8 gives the $Q(a)$ for all actions and Figure 9 gives the correspond standard derivation. Figure 10 shows the total action selected times for 6 algorithms running 500 times.

Findings and Conclusions

- Compare the histogram for σ and 2σ distributions, we could find the 2σ's reward distribution results more random explorations.
- Compare the Figure 4 and Figure 9, we could find higher σ reward distributions give similar standard derivations in reward receiver per actions.

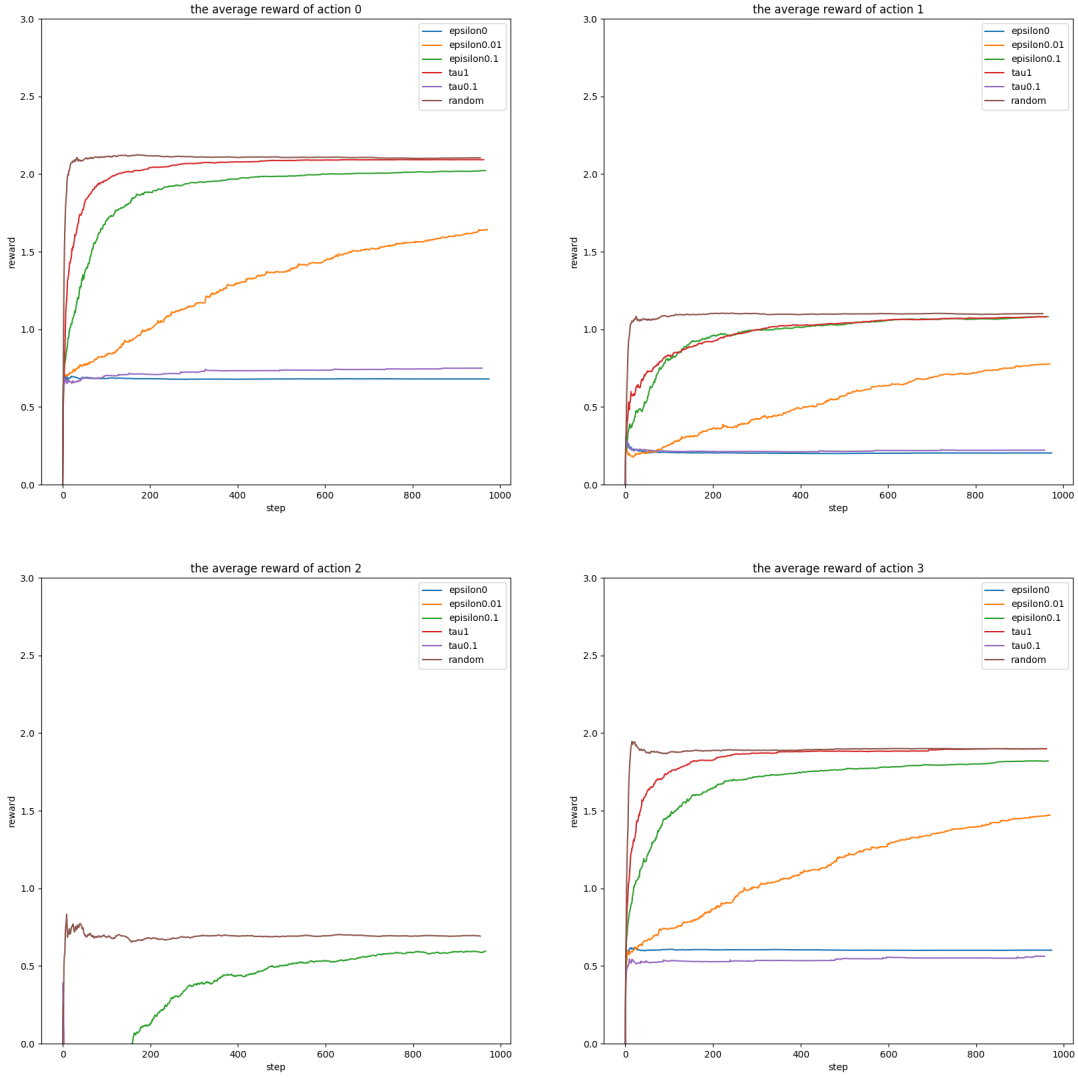

Figure 8: The reward standard derivation for 6 algorithms (500 trials)

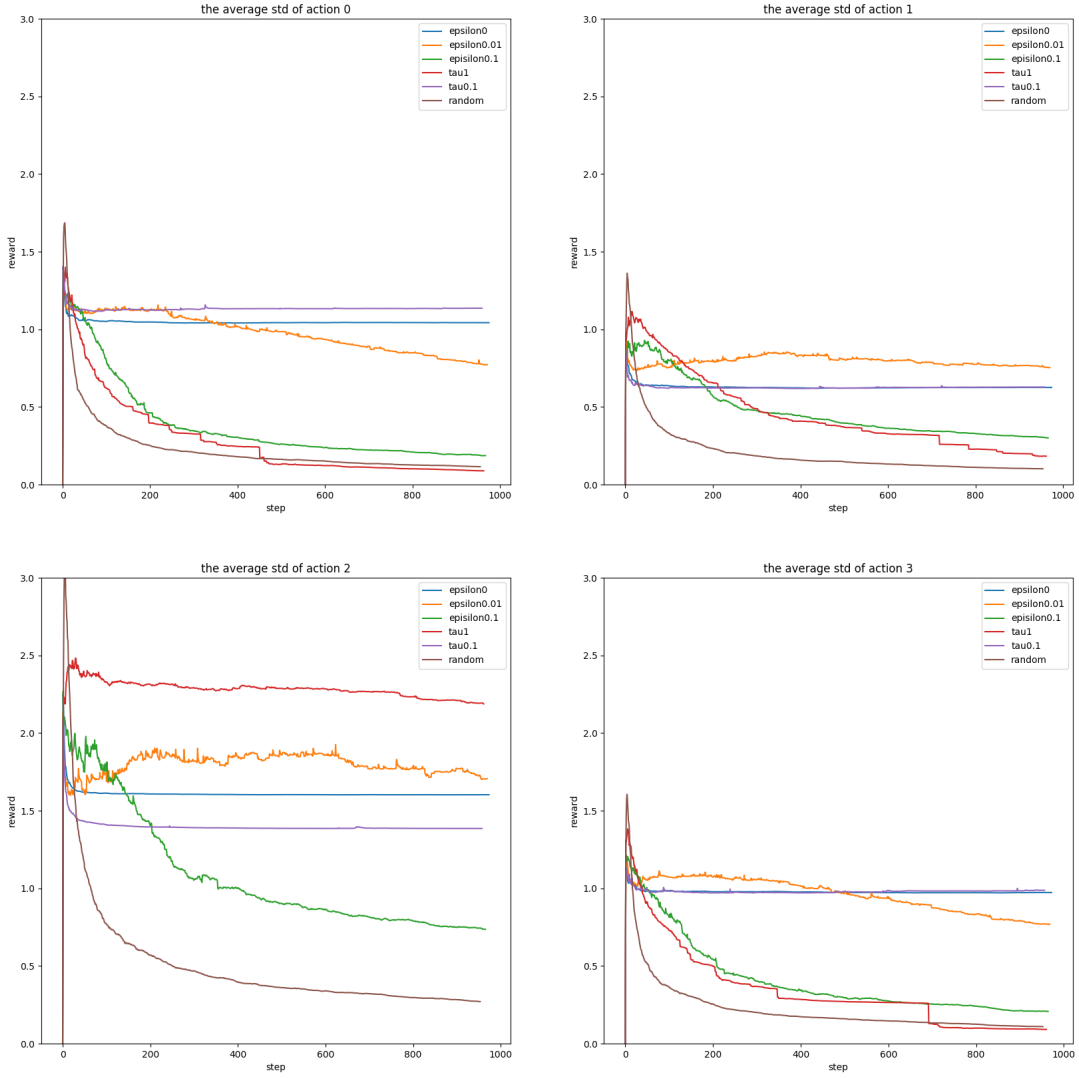

Figure 9: The reward standard derivation for 6 algorithms (500 trials)

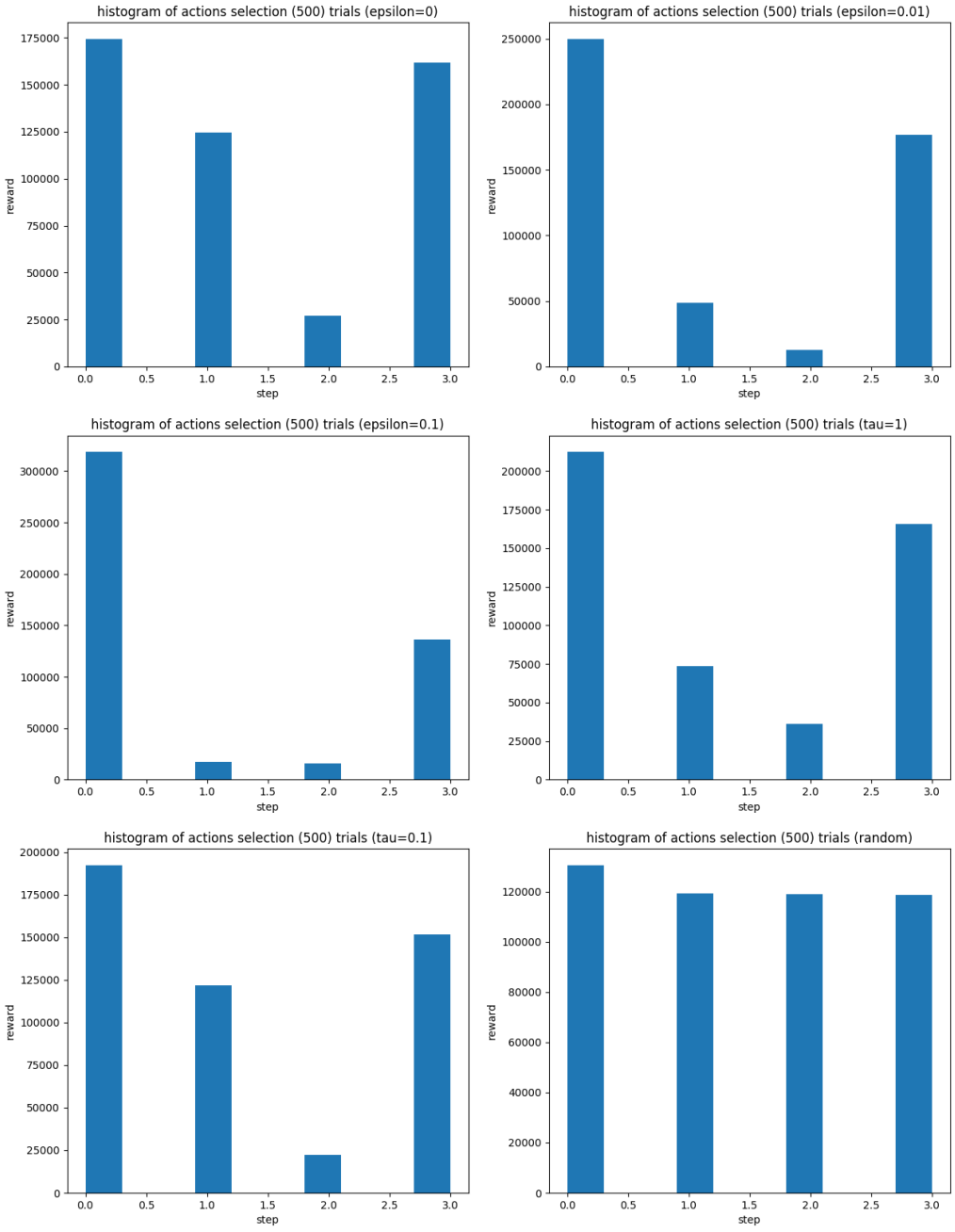

Figure 10: The total action selected times for 6 algorithms (500 trials)

1.4 Plots for Time Varying Parameters

In this subsection, we explore the time varying parameters performance. We choose the $\epsilon = \frac{1}{\sqrt{t}}$ and $\tau = 4 * \frac{1000-t}{1000}$, We compare these 2 algorithms performance with $epsilon = 0.01, 0.1.\tau = 1, 0.1$

Findings and Conclusions

- At first,the time varying parameters gives lower average reward because larger random exploration probabilities, and with the time increasing, the ϵ becomes smaller and the τ becomes smaller which means less random exploration, the average rewards converge.
- From the experiment results, the time varying ϵ gives better result than fixed parameters. The reward is sample average, for **softmax** with time varying τ does not give better result in this experiment.

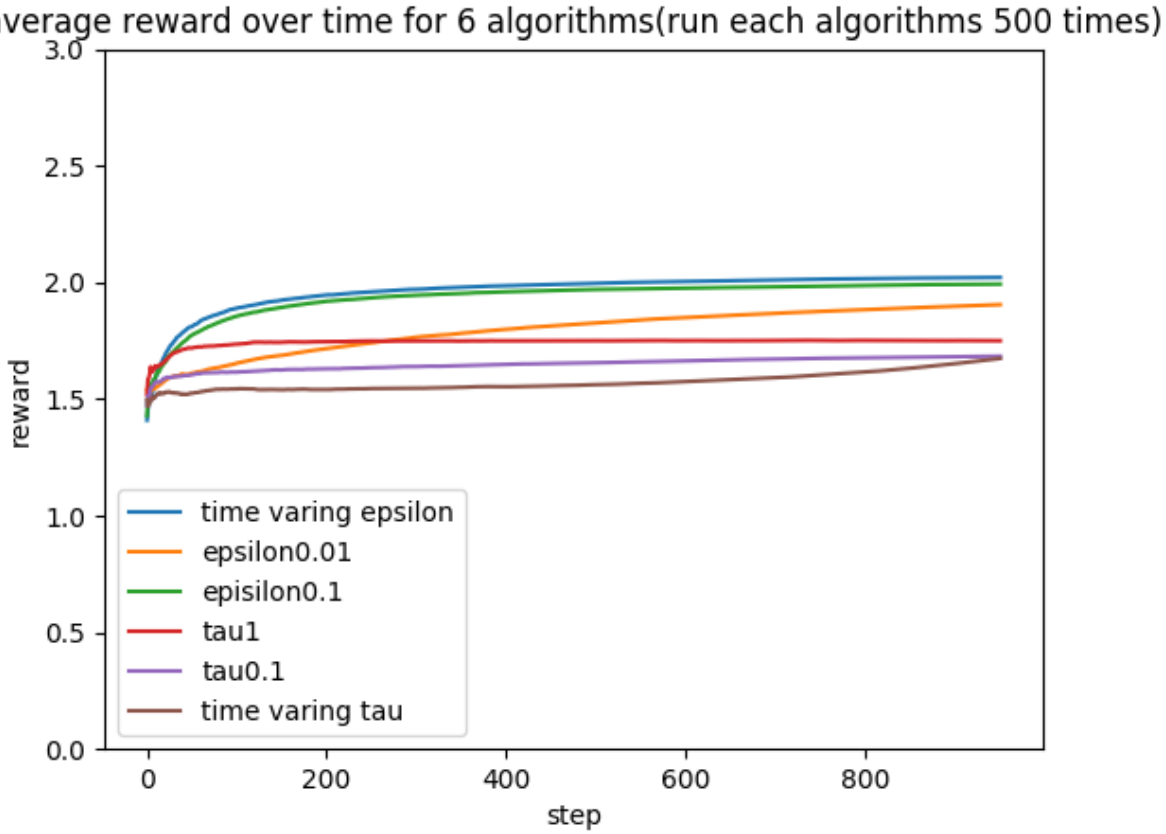

Figure 11: The average reward for time varying ϵ,τ and compare them with other 4 algorithms (500 trials)

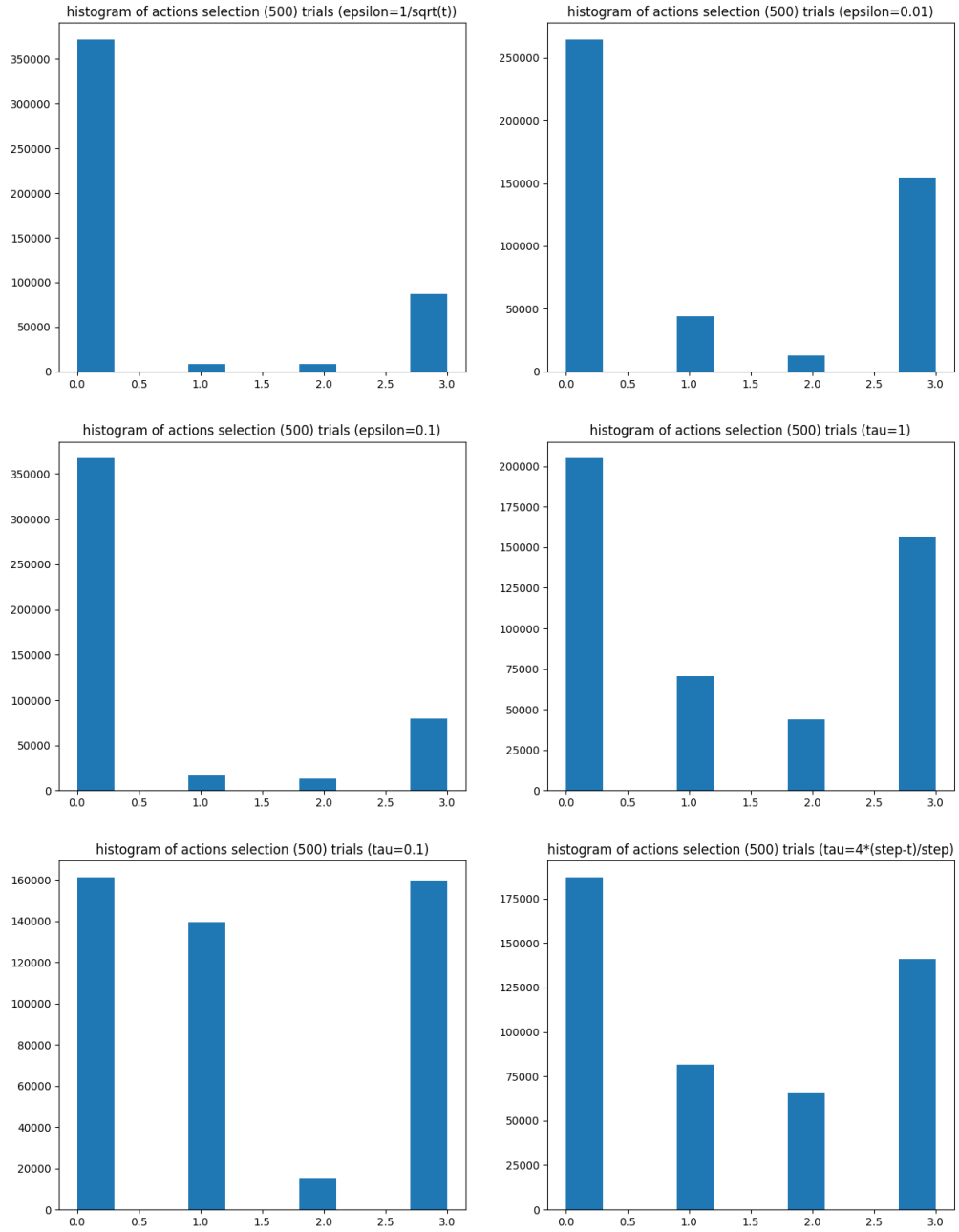

Figure 12: The total action selected times for 6 algorithms (500 trials)

2 Stochastic Reward Game

The section 2 will discuss the reinforcement learning in stochastic reward game. The table 3 shows the reward for joint actions.(*Here the action number starts from 0, not 1*)

Table 3: Stochastic climbing game

	a_0	a_1	a_2
b_0	$\mathcal{N}(11, \sigma_0^2)$	$\mathcal{N}(-30, \sigma^2)$	$\mathcal{N}(0, \sigma^2)$
b_1	$\mathcal{N}(-30, \sigma_0^2)$	$\mathcal{N}(7, \sigma_0^2)$	$\mathcal{N}(6, \sigma_0^2)$
b_2	$\mathcal{N}(0, \sigma_0^2)$	$\mathcal{N}(0, \sigma_0^2)$	$\mathcal{N}(5, \sigma_0^2)$

In this section, two types of Joint-Action learners are implemented. The first type is **simple Boltzmann action selection**, the second type is **optimistic Boltzmann action selection**.

Parameters for these 2 algorithms : Initial temperature 5000 is decayed at rate 0.998^t, $\tau = 5000 * 0.998^t$(I tried several parameter sets and found this parameters is suitable to present the result).At first, the agents will have higher probability of random exploration, then with time increasing, the agents will tend to greedy search.

2.1 Plots for Simple Boltzmann action selection and optimistic Boltzmann action selection

Findings and conclusions

- The Figure 13 shows the reward per episode ($\sigma_0 = \sigma_1 = \sigma = 0.2$), at last, the **optimistic boltzmann** find the best response while the **simple boltzmann** falls into the local minima
- The Figure 14 shows the reward per episode ($\sigma_0 = 4, \sigma_1 = \sigma = 0.1$), at last,the **optimistic boltzmann** find the best response while the **simple boltzmann** falls into the local minima.
- The Figure 15 shows the reward per episode ($\sigma_1 = 4, \sigma_0 = \sigma = 0.1$), at last,the **optimistic boltzmann** and **simple boltzmann** both fall into the local minima.
- From Figure 15 and Figure 17's subplots for these σ-sets, at last the **optimistic boltzmann** falls into the high penalty joint action. A will choose action 0 while B will choose action 1 at last, which is because of the high σ and max reward action selection.

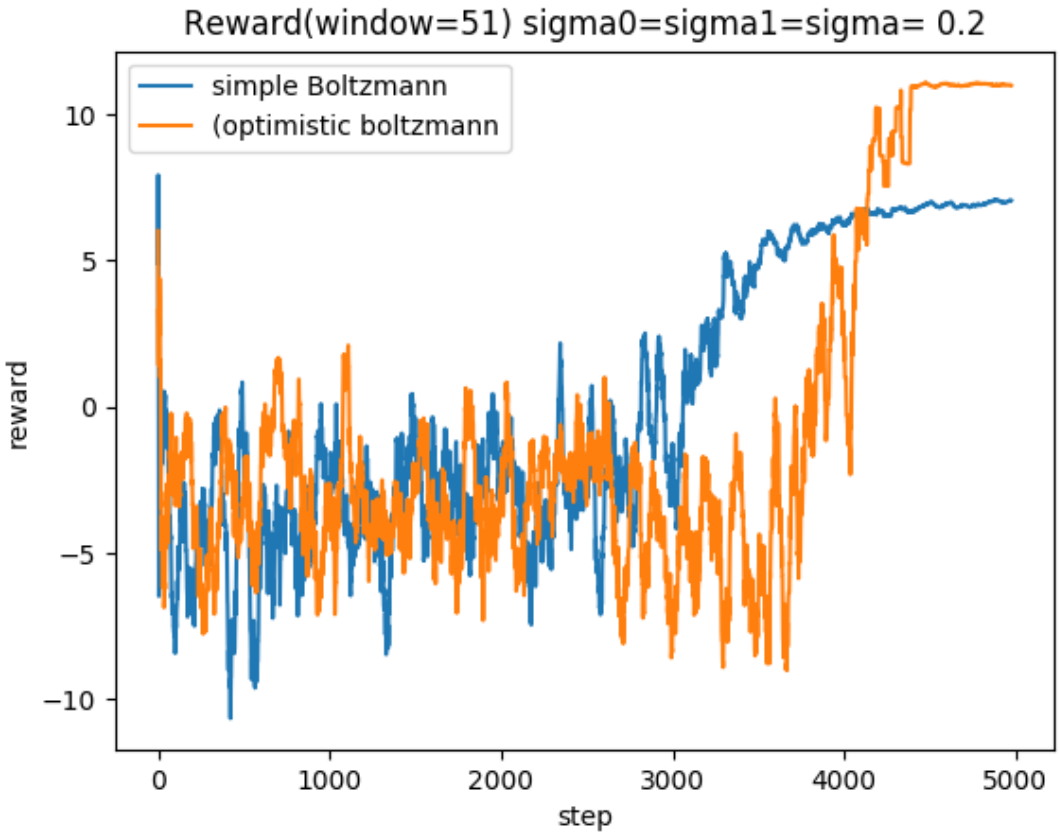

Figure 13: The reward received($\sigma_0 = \sigma_1 = \sigma = 0.2$)

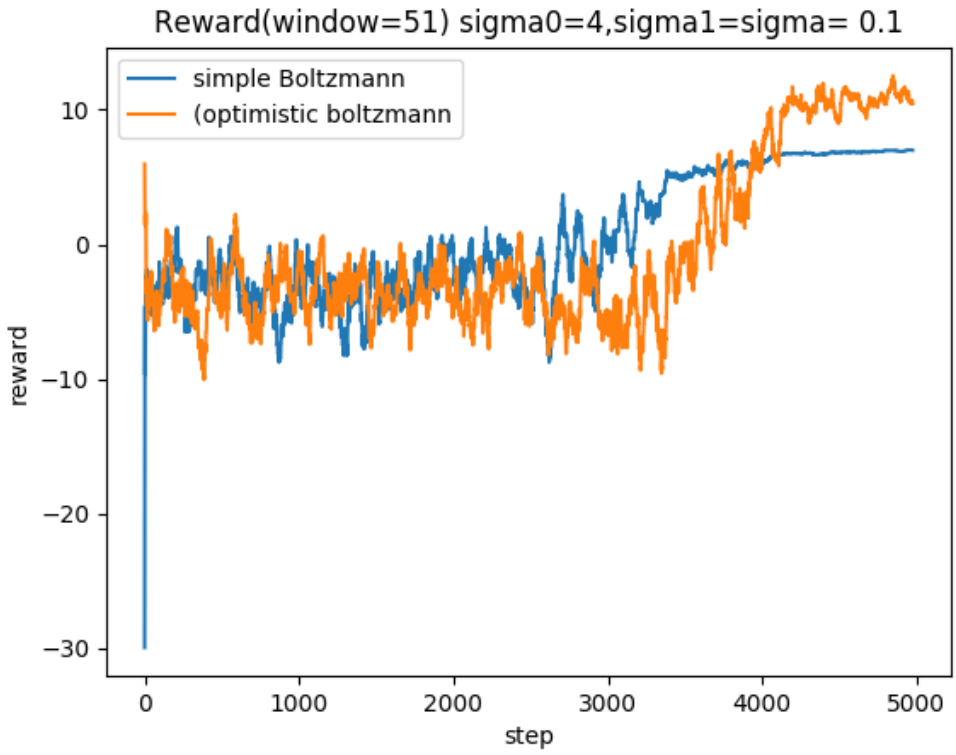

Figure 14: The reward received($\sigma_0 = 4, \sigma_1 = \sigma = 0.1$)

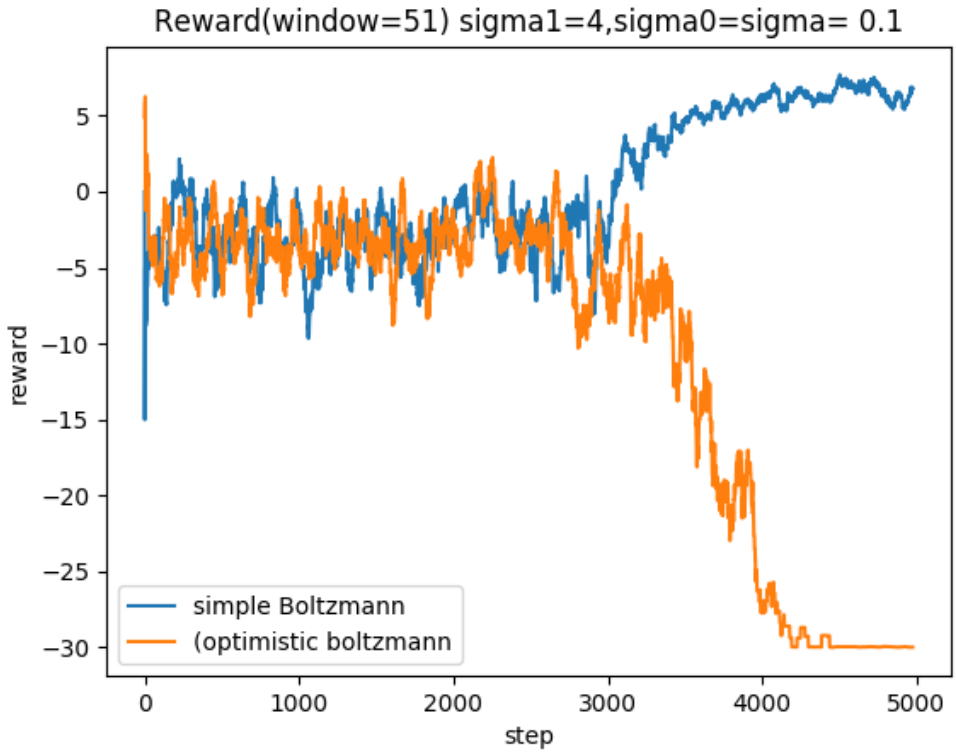

Figure 15: The reward received($\sigma_1 = 4, \sigma_0 = \sigma = 0.1$)

2.2 The probabilities of selecting actions

In order to have a insight of these algorithms and how they find the Nash Equilibrium.

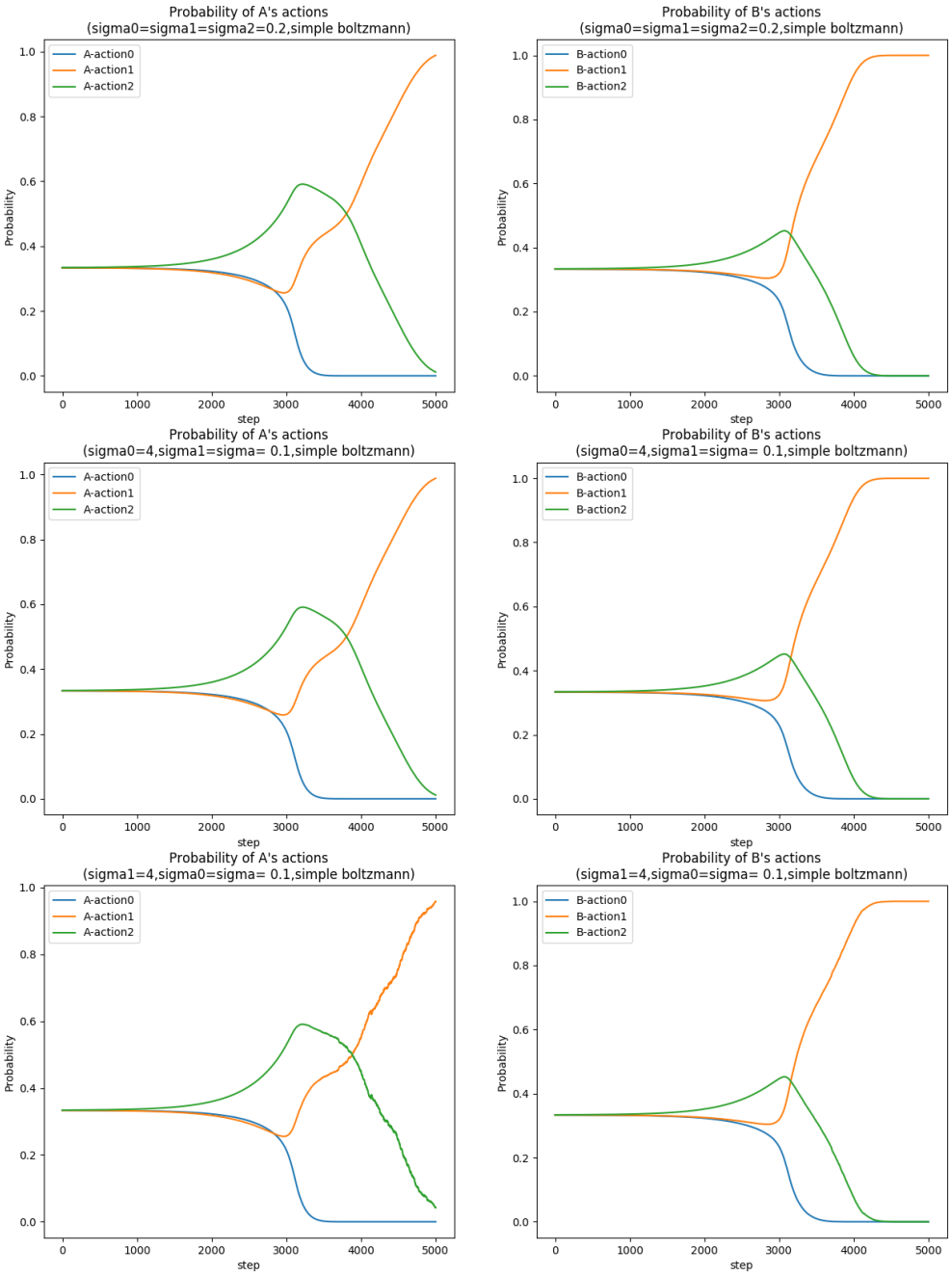

Figure 16: A and B's strategies in climbing game(simple boltzmann)

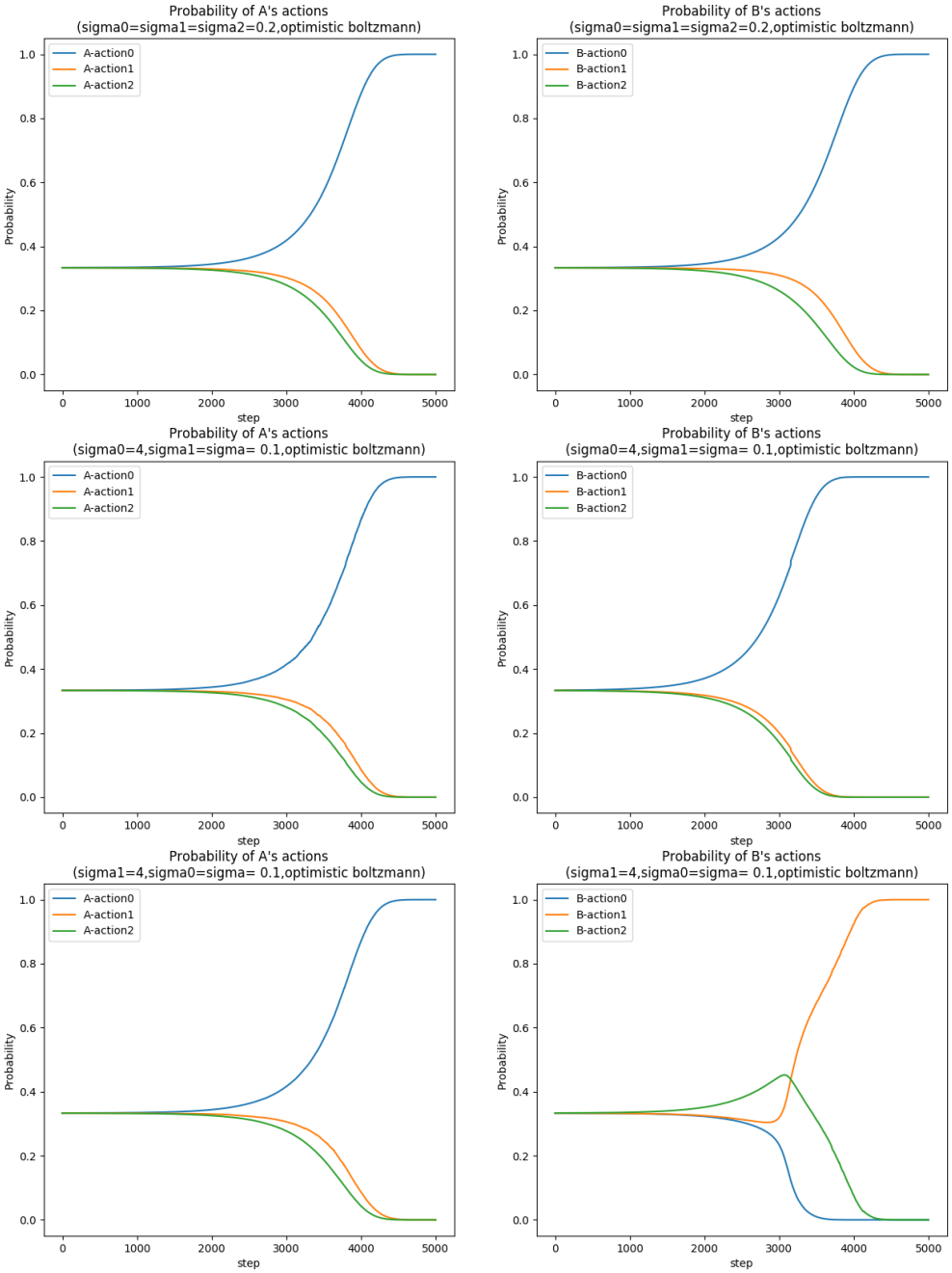

Figure 17: A and B's strategies in climbing game(optimistic boltzmann)

2.3 Discussions

1.How will the learning process change if we make the agents independent learners?

If we make the agents become independent learners, it will fall into the local minima with B always choosing action 2 and B always choosing action 2. I write some code to simulate this($\tau = 5000 * 0.995^t$), and the results are shown in Figure 18 and 19. Finally the reward will be around 5. Because choose action 0 and action 1 will have the probability to have high penalties so A and B will all choose action 2 at last.

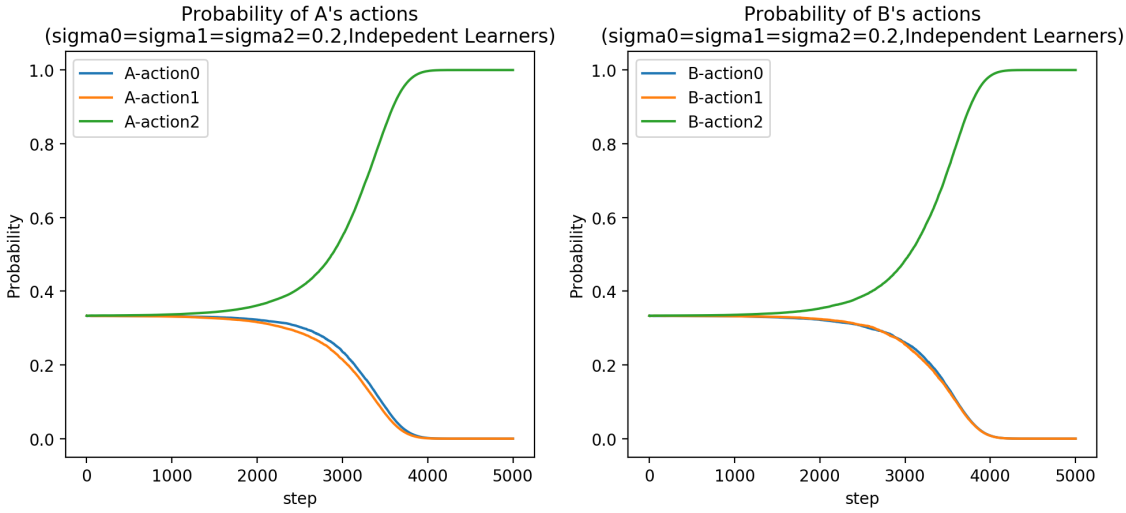

Figure 18: The probabilities of selecting actions for independent learners

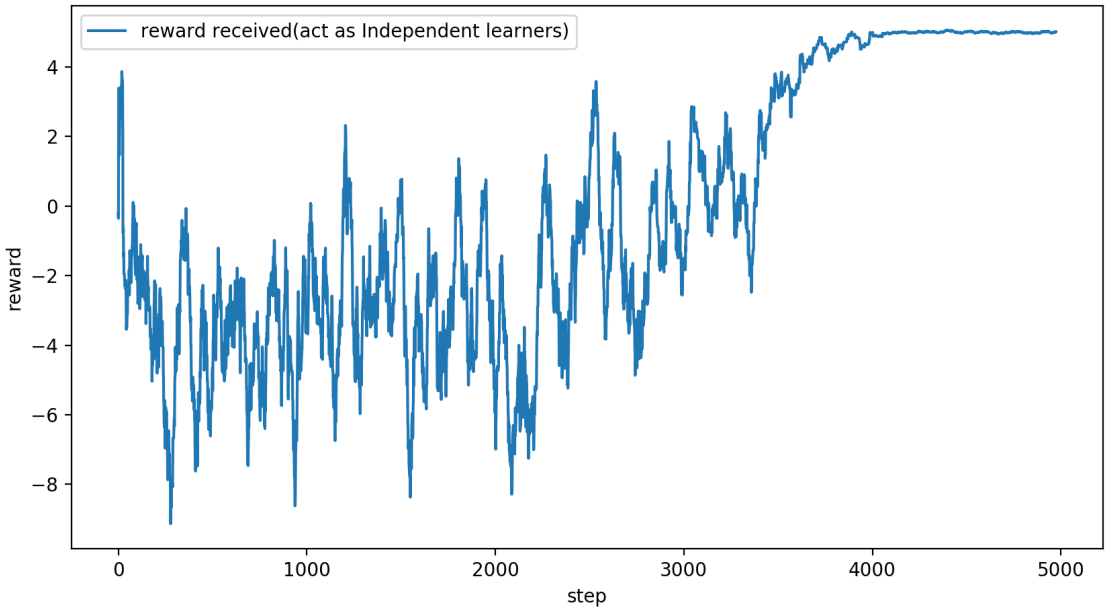

Figure 19: The Reward received for independent learners

2.How will the learning process change if we make the agents always select the action that according to them will yield the highest reward (assuming the other agent plays the best response)?

According to my understanding, always choosing the highest reward is like the **Optimistic Boltzmann**, A and B's best action are action 0 which will yield the best reward 11(if the σ_1 is not too high), more discussion could be found in the previous part.