# Discrete Muti-addicted State Q-agents Making Decisions

**Kaiwen Bian**
University of California, San Diego
Salk Institute of Biological Study
kbian@ucsd.edu

## 1 Introduction

It have long been proposed that reinforcement learning algorithms and the neural mechanism of human decision making is highly alike (Niv (2009)) and many previous studies (Mollick and Kober , 2020) have used these algorithms as a computational tool to reason about human behaviors. In this study, we extend from the work of Redish in 2004, an TD learning algorithm that have been modified to model the effect of drug addiction through dopamine surges, to try to model the effect of a monotonic decreasing dopamine surge function on an Q-agent's ability to make decision under a discrete chain multi-addiction states setting. Specifically, we built different search strategies into the Q-agent (some resembles more to human reasoning logic than others) and created a discrete chain environment to model decision making under addiction and also when there exist higher natural rewards existing in the environment.

## 2 Methods

Traditionally in the study of addiction through the perspective of reinforcement learning algorithms, one approach is to study value-iteration (Redish (2004)), specifically Temporal Difference Reinforcement Learning (TDRL) because of its resemblance to dopamine's functionality in the human decision making circuit. In the human reinforcing circuitry, the striatum area of basal ganglia incorporates environmental state (sensory motor information) from the cortex with dopamine reward prediction error signal from the ventral tagmental area to adjusts the weights on action selection back to the cortex, influencing movements, decision making, and further reward processing. This is very alike in how TDRL update its understanding of the values of the environment.

$$\delta = R(s_i) + \gamma V^{\pi}(s_{i+1}) - V^{\pi}(s_i) \tag{1}$$

$$V^{\pi}(s_i) \leftarrow V^{\pi}(s_i) + \eta\delta \tag{2}$$

$s_i$ is current state, $V^{\pi}(s_i)$ and $V^{\pi}(s_{i+1})$ is the value of the current and next state under the current policy $V^{\pi}$, $R(s_i)$ is the reward feedback of the current state, and $\gamma$ and $\eta$ would be the discounting and learning rate accordingly. At each time stamp, an reward prediction error (RPE) is produced by taking the discounted difference of current understanding of the trajectory (value from $s_i$ and onward) and true reward plus the future understanding of the trajectory (real value plus the value from $s_{i+1}$ and onward). Such RPE have been thought to serve the role that dopamine serves in the human brain during decision making, to not serve as a pleasure signal, but rather a internal signal indicative of the discrepancy between expectations and observations. In this study, we extend the algorithm beyond TDRL to Q-learning, an agent that is actually capable of making decision and move around in the environment.

### 2.1 Addicted Q-learning Agent

In normal Q-learning, it deals with the value of the action $Q(s_i, a_i)$ directly instead of the value of a state like in TDRL for a mathematical reason (Sutton and Barto , 2018).

$$\delta = R(s_i) + \gamma \max_{a \in A_{i+1}} Q(s_{i+1}, a_{i+1}) - Q(s_i, a_i) \tag{3}$$

$$Q(s_i, a_i) \leftarrow Q(s_i, a_i) + \alpha\delta \tag{4}$$

The structure of the update preserves, the only difference being that we are using the $max$ operator to compare the current understanding of the trajectory against the best trajectory that we can think of so far (real reward $R(s_i)$ plus the best Q-value from next state $\max_{a \in A_{i+1}} Q(s_{i+1}, a_{i+1})$) to guarantee convergence. Furthermore, we modified the error signal $\delta$ to incorporate the effect of drugs (abundant dopamine surge) by taking the $max$ operator again on the error of error plus surge and just surge (Redish (2004)).

$$\delta_{addicted} = max(\delta + D(t), D(t)) \qquad (5)$$

The $D(t)$ dopamine surge term is a monotonic decreasing function that can be changed for different purposes. All experiments in this study uses a exponential decrease function of $D(t) = D(t_0)d^t$ where $t$ is the number of trails and $d$ is the decay constant. see the full update expression in Appendix equation 1.

### 2.2 Discrete Chain and Rewards

Environment is an major component of reinforcement learning as the structure and reward feedback for the environment dictates what the agent can learn. In this study we have adapted similar design as Redish 2004 with a discrete chain-like environment. Particularly, we design the action space of the agent to be only $a \in$ (forward, backward, stay). Such environmental setting also made it easier fro us to compare the action of the agent with an pure stochastic modeling as the environment is essentially a discrete Markov Chain. For all experiments conducted in this study, we have used a chain length of 10. Since we are interested in the effect of multi-addiction state's effect on the agent's action selection, we have built in two drug rewards at $s_0$ and state $s_9$. We have also built in a natural reward at $s_5$ that has doubling the reward of the drug state to model whether the Q-learning agent is able to gain such understanding of the environment under the effect of addiction, this feature would be used for some of the studies of this paper, which would be stated in the according section. For details, see Appendix figure 1.

### 2.3 Various Search Strategies

The Q-learning process using the modified error essentially points a direction of the agent to follow already, by taking the maximization step at every single time stamp, we can formulate the most naive way of doing selection: select the action with the maximized Q-value at every step, this is also known as the Greedy algorithm. However, since the setting of the environment doesn't just have a single reward state, it may be too biased to always choose the best Q-value as the better ones may just have not been discovered yet. An classic approach to this problem would be the Epsilon Greedy policy, where at every step, a random value is generated between 0 and 1 informally and if it is beneath certain $\epsilon$ threshold, then the agent selects a random action to explore the environment. For our study we choose $\epsilon = 0.1$, meaning that there would be 10 percent of the time where the agent is exploring the environment. Since the setup of our environment is quite simple, we believe that such low threshold would be sufficient. Lastly, we have also built in the Boltzmann Exploration strategy, where it select actions according to a probability distribution estimated by the Q-values and actions with higher estimated values are chosen more frequently, but also giving a chance for the lower valued actions to be chosen. The probability distribution comes from taking the exponential and then normalizing the Q-value of one action against all other actions at this state (Sutton and Barto , 2018).

$$P(a|s) = \frac{\exp(Q(s,a)/\tau)}{\sum_{b\in\mathcal{A}} \exp(Q(s,b)/\tau)} \qquad (6)$$

This strategy have been deemed to be a more human-like strategy because it is developing not only the Q understanding of an action but also understanding of the relationship between action through a probabilistic paradigm, forming a much robust interpretation of the space. Empirical testing results from our study have also shown that this strategy does perform the best under addicted conditions.

### 2.4 Procedure

We have conducted multiple experiments to evaluate the performance of the Q-agent under an addicted state. We have established some base parameters that we use for all our experiments. For particular experiments, some parameter may be changed and it would be stated in the according sections.

We first examined (1) under addicted state and addicted state with the presents of doubling natural reward, how well does Epsilon Greedy and Boltzmann Exploration strategy perform and delivers the results through a heat map representation of Q-value for different state across different trials, attempting to demonstrate the learning process. We then (2) turned addiction effect off (no dopamine surge addition to the error) and examines the performance of the two strategy through the comparison of average RPE. At last, we (3) built a discrete stochastic model (random walk) and compare the expected visit time of the agent using the two strategies with such random model.

| Parameter | Value |
|---|---|
| alpha | 0.1 |
| gamma | 0.9 |
| epsilon | 0.1 |
| num_trials | 100 |
| num_states | 10 |
| num_actions | 3 |
| initial_dopamine_surge | 1 |
| dopamine_decay_rate | 0.99 |
| reward_states | [0, 9] |
| drug_reward | 1 |
| addicted | True |
| exploration_strategy | epsilon_greedy |
| if_natural | False |
| natural_reward_states | [5] |
| natural_reward_boost | 2 |

Table 1: Base Configuration Settings

## 3 Results

We have trained two agents with two different search strategies (Epsilon Greedy and Boltzmann Exploration, in here we would refer them as $A_e$ and $A_b$) under two conditions (addicted and addicted with doubling reward presented) to demonstrate the effect of addiction on the agent.

### 3.1 Heat Map Representation of Learning

From using a heat map representation of learning process, the result have indicated that (1), under pure addiction condition, both agents find both of the drug states and exhibits similar learning effect highlighted previously by Redish 2004 where, as trials increases, the nature of the algorithm propagates the reward from the drug states back to previous states. Shown in both Forward Move graphs propagating back from $s_9$ and in the Backward Move graphs propagating from $s_0$ forward. Moreover, (2), under the setting where doubling value reward is presented in the environment at $s_5$, $A_e$ seems to fail to acknowledge such higher reward and still choose to go back to $s_0$ or $s_9$ for where the lower drug rewards are at. On the other hand, $A_b$ seems to built a more robust understanding of the environment and constantly choose to go forward from $s_4$ to $s_5$, from $s_6$ back to $s_5$ or constantly staying at $s_5$. It is also worth pointing out that (3), throughout the experimentation, $A_e$'s performance is not steady and does not always come to similar action selection while $A_b$ is much more robust with its action selections. For details of the heat map, see Appendix figure 2.

### 3.2 Average RPE Comparison

RPE have been studied a lot as it indicates an key component of learning: what is unexpected and what direction should the update be going towards. Inherently, comparing the RPE for different strategies under addicted and non0addicted state becomes a interesting question. From our result, we have shown that (1) while Greedy strategy exploits maximally and build the most steady representation of the environment, $A_b$ overall-wise perform much steadier and build its representation of the environment much better than $A_e$. It is also note worthy that, though the Greedy strategy builds a stable representation, such representation ma be incorrect due to the lack of exploration when the environment setting gets much more complicated and reward states gets much more sparse. In addition, we have shown that (2) for all strategies, the RPE has a concave shape where the error gradually raises in the beginning and drops gradually as the agent explores the environment more. At last, the comparison between addicted and non-addicted agent have shown that (3) the non-addicted agents are generally more nosier in RPE comparing to the addicted agents, signifying the effect of the dopamine surge really grabbing all the "attention" of the agents. For details of the RPE curve, see Appendix figure 3.

### 3.3 Expected Visits Comparison

Since the setting of the environment allows the agent to make actions, it would be inappropriate to set a baseline random model just by using plain probability since the location of the previous state dictates where the next state could be at. Hence, we have established a simple random walk model with probability of forward and backward jump both being $p(s_i) = 0.5$ to establish a baseline of how the agent would perform under a purely stochastic condition. Furthermore, it is a usual step in discrete stochastic processes reasoning to discuss about the expected or average number of each state to reason about some global characteristic of the chain (environment) or the agent. This part of the study is condcuted under the setting where a doubling natural reward is presented. The random walk was performed over 100 trials with each trail stepping the environment 1000 environmental step. From the random walk expected visits graph, it can be seen that under a pure stochastic movement through the environment, (1) the states on the edge ($s_0$ and $s_9$) are actually less likely to be visited compare to those in the middle, illustrating the power of rewarded drug states pulling the agent towards the edge. The expected visit graph of the two Q-learning agents are generated using trained Q-value tables with the base setting in Table 1 then re-simulated the actions 10000 trials with max of 100 actions per trails. The max action here is a designed choice because (2) the agent would simply be stuck at addiction states going back and forward or choosing the staying action constantly to reach to the drug states, never terminating the trail. Again, similar results are shown here with the results demonstrated in the heat map result where (3) $A_b$ finds much more robust representation of the environment and finds the true high natural reward while the $A_e$ gets stuck in one of the addiction state. Appendix figure 4.

## 4 Discussion

The results from this study delivers an new perspective of looking at the addictive decision making through computational tools. Though no neural network or more modern deep learning methodology is used in this study, we still can reason with the innate mathematical properties and characteristics behind these algorithms and try to find the aspects that resembles human behavior. Again, we have demonstrated in this study that, (1) rewards tends to propagate backward or forward to different states from the drug state, which is consistent with Redish's finding using single drug state (see see Heat Map Representation of Learning). (2) Boltzmann Exploration strategy tends to find much more robust representation of the true environment in both addicted and addicted plus natural reward conditions when comparing to Epsilon Greedy strategy, high lightening the importance of probabilistic reasoning in decision making (see Heat Map Representation of Learning, Average RPE Comparison, and Expected Visits Comparison). We have (3) also demonstrated the power of addiction (dopamine surge), trapping the agent and misleading the agent to build a robust interpretation of environment that gets all the attention on the drug states (see Average RPE Comparison).

However, this study only points out the computation and theoretical aspect of using such strategy to study decision making, no behavioral or neuronal counter parts study was conducted to match the behavior of the algorithm in real organism. More studies need to be conducted on the biological experimentation side to proof the validity of the ideas delivered by the algorithms in this study or to find new addictive behaviors from observations mentioned in this study. In addition, this study only touches on some of the most fundamental search strategies in a discrete reinforcement learning setting. More studies can be conducted to explore and incorporate more algorithms, perhaps even moving to continuous policy optimization realms involving Actor-critic algorithms (Sutton and Barto , 2018) and more.

## References

Jessica A. Mollick and Hedy Kober. 2020. Computational Models of Drug Use and Addiction: A Review. *J Abnorm Psychol.*, 129(6):544–555. doi: 10.1037/abn0000503. PMCID: PMC7416739. PMID: 32757599.

A. David Redish. 2004. Addiction as a computational process gone awry. *Science*, 306(5703):1944–1947.

Yael Niv. 2009. Reinforcement learning in the brain. *Journal of Mathematical Psychology*, 53(3):139–154.

Richard S. Sutton and Andrew G. Barto. 2018. *Reinforcement Learning: An Introduction.* MIT Press.

# Appendix With Figures

## A Environment Chain Setup

The code base for doing simulations for this study is located in this GitHub repository.

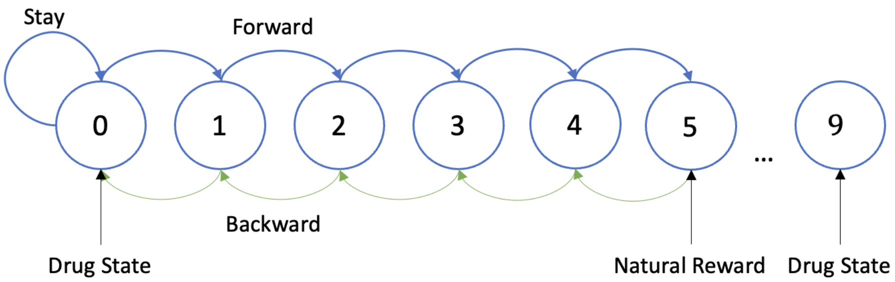

Figure 1: Discrete multi-addiction state chain environment's graphical illustration.

## B Full Update Rule

Full update of Q-value expression of addicted Q-agent:

$$Q(s_i, a_i) \leftarrow Q(s_i, a_i) + \alpha \left( \max_{2} \left( \left( R(s_i) + \gamma \max_{a \in A_{i+1}} Q(s_{i+1}, a_{i+1}) - Q(s_i, a_i) \right) + D(t_0)d^t, D(t_0)d^t \right) \right)$$

The key terms are described in the following list:

- $R(s_i)$: True reward of state $s_i$
- $Q(s_i, a_i)$: Action-value function for taking action $a_i$ in state $s_i$.
- $\alpha$: Learning rate, determining how much new information overrides old information.
- $\gamma$: Discounting factor, determines how important later rewards are
- $R(s_i) + \gamma \max_{a \in A_{i+1}} Q(s_{i+1}, a_{i+1}) - Q(s_i, a_i)$: Temporal difference error term, where $R(s_i)$ is the reward.
- $\max_{a \in A_{i+1}} Q(s_{i+1}, a_{i+1})$: Maximum future value given the current understanding at the next state $s_{i+1}$.
- $D(t_0)d^t$: Dopamine surge monotonic decreasing function.

# C Analysis Figures

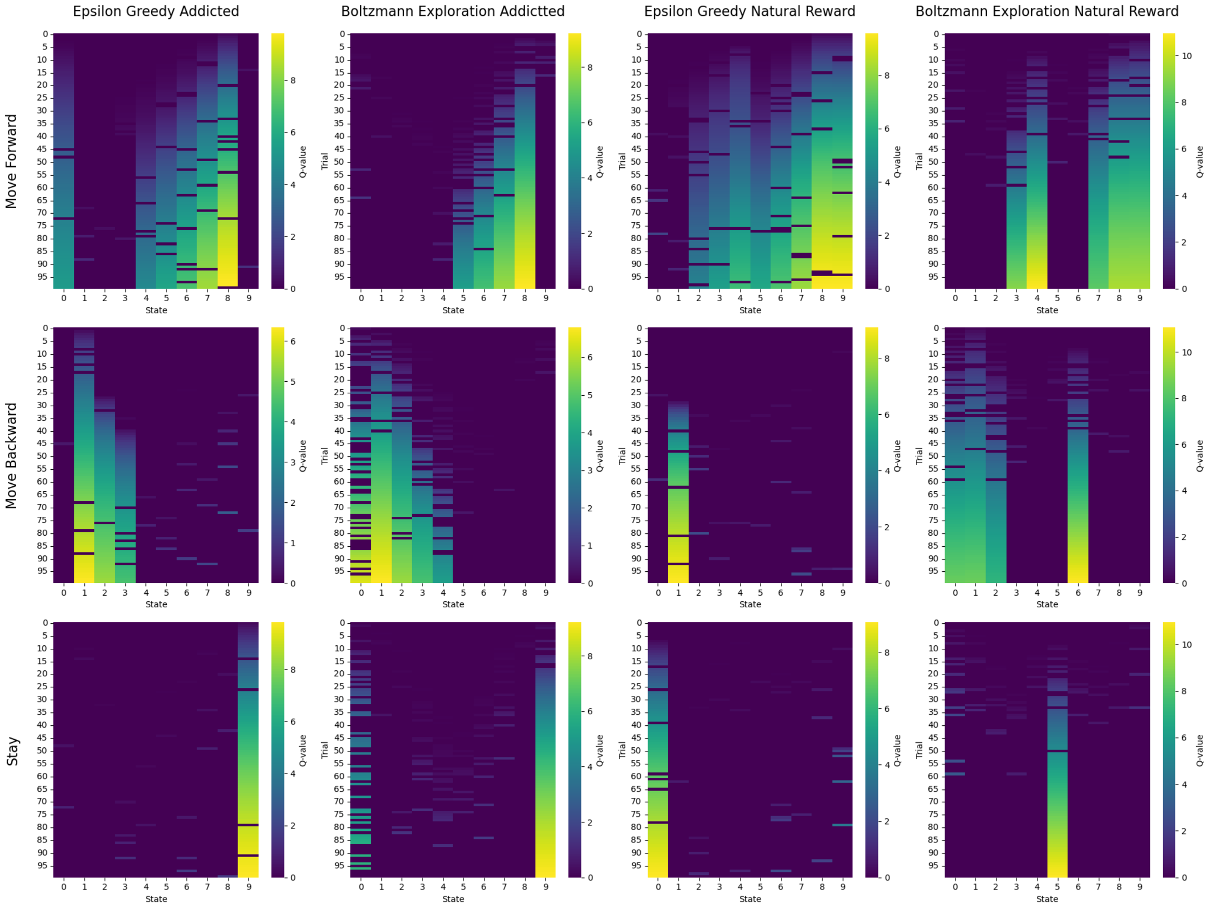

Figure 2: Heat map representation of Q-value in different search strategies and reward states across different trails when natural reward is presented or not.

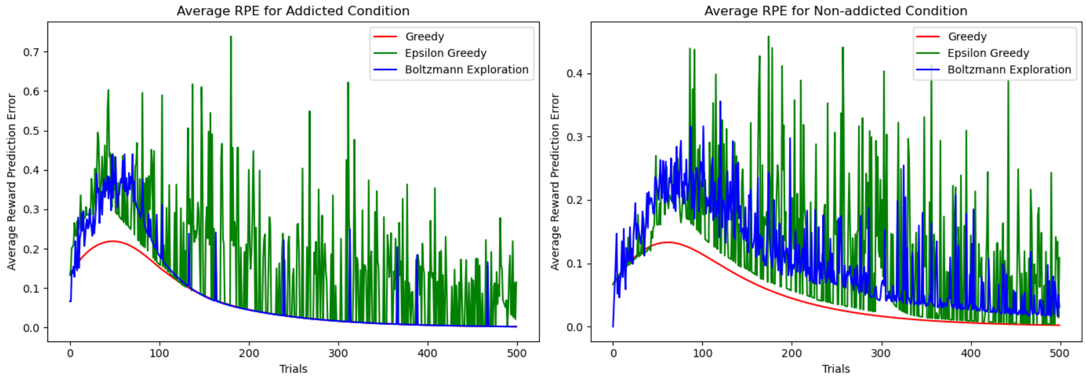

Figure 3: Average RPE for different search strategy Q-agent under an addiction or non-addiction condition (presents of dopamine surge or not).

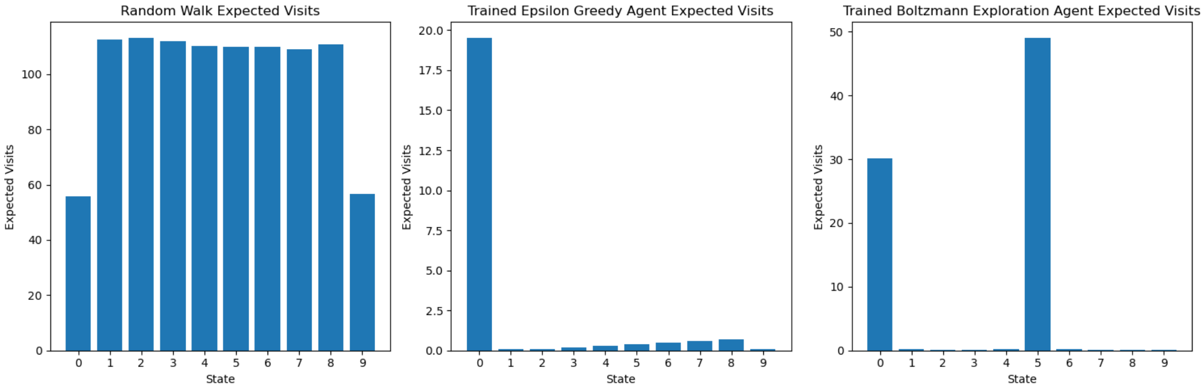

Figure 4: Expected visits of trained Q-agents and random walk stochastic processes when natural reward is presented.