Reflect-to-Explore: A Self-Adaptive Exploration Strategy for Deep Reinforcement Learning

Chenyu Zhou
Shanghai Pinghe School

Abstract

Cognitive science research shows that humans rely on two key cognitive regulation mechanisms during learning: rapid behavioral adjustment based on immediate feedback, and strategy optimization through reflection. In order to achieve similar adaptability in reinforcement learning, this study proposes a confidence-based two-layer reflection mechanism learning framework. The framework consists of a fast adaptation layer and a reflection layer: the fast adaptation layer adjusts the exploration rate and learning rate by monitoring the confidence changes between adjacent episodes to quickly adapt to performance fluctuations. When the reflection layer detects a recent trend of persistent inefficiency or performance degradation, it adjusts the exploration rate and learning rate in combination with the experience replay mechanism to optimize the strategy and improve long-term adaptability. We designed a multi-dimensional confidence evaluation system that combines path efficiency, reward stability, and success rate to provide a more comprehensive decision-making evaluation standard for intelligent agents. In a control experiment in a dynamic maze environment, this method increased the task completion rate from 37.9% to 50.3%, while reducing the average number of steps by 7.8%, showing stronger environmental adaptability and providing new ideas for adaptive decision-making of intelligent agents in dynamic environments.

*Keywords:*reinforcement learning; cognitive regulation; reflection mechanism; dynamic environment adaptation; confidence evaluation

1 Introduction

Humans have an innate learning adaptability, and can continuously adjust strategies based on task feedback to adapt to environmental changes and optimize learning outcomes (Fleur et al., 2021). For example, a student who is good at studying may initially use his usual study methods when facing a new course. However, when his test scores are lower than expected, he will reflect on his study methods and try new study methods or optimize practice methods based on these feedbacks (Ochilova, 2021). If the adjustment brings obvious progress, he is more inclined to continue to implement the new strategy; if the effect is not good, it will prompt him to further reflect and adjust, so as to gradually find a more suitable study method for himself, so as to improve his future performance.

The core of this adaptive learning ability lies in the dynamic evaluation and reflection mechanism. During the learning process, we not only rely on past experience, but also continuously monitor the current environment and our own performance, and make adjustments based on feedback when the strategy does not meet expectations (Ochilova, 2021). However, existing reinforcement learning methods still have significant limitations when dealing with dynamic environments. Although algorithms such as Deep Q-Network (DQN) and Proximal Policy Optimization (PPO) perform well in static tasks, they still have the problem of insufficient adaptability when facing dynamic environments. Although meta-reinforcement learning (Meta-RL) extracts general knowledge through multi-task training to accelerate adaptation to new tasks, its high computational overhead and reliance on a large amount of training data limit its practical application scope (Radulescu et al., 2021). To address the above challenges, this study proposes a two-layer reflective reinforcement learning framework based on confidence evaluation.

The main contributions of this study include:

(1)A two-layer reflective learning framework based on confidence evaluation is proposed. Through the synergy of the fast adaptation layer and the reflective layer, the framework adjusts the strategy based on different confidence trigger conditions to improve the adaptability of the agent in a dynamic environment.

(2)A multi-dimensional confidence evaluation method is designed. Through the comprehensive analysis of path efficiency, reward stability and success rate, the real-time quantitative evaluation of the learning performance of the agent is realized, providing a reliable evaluation benchmark for adaptive strategy optimization.

(3)The effectiveness of this method is verified through systematic experiments in static and dynamic maze environments. The experimental results show that the mechanism is significantly superior to the benchmark method in terms of task completion rate and path optimization.

2 Related Research

This section first reviews the theory of human cognitive regulation, and then analyzes existing reinforcement learning methods and their limitations, as well as the characteristics of meta-learning methods.

2.1 Human cognition and learning mechanism

Flavell's metacognitive theory shows that learners optimize the learning process through cognitive monitoring and cognitive regulation (Flavell, 1979). Nelson and Narens (1990) further proposed a two-layer cognitive framework, in which the object layer is responsible for the execution of specific tasks, while the meta layer is used to monitor and regulate and continuously evaluate the performance of the object layer. Zimmerman's self-regulated learning research reveals the key role of reflection mechanisms in the learning process. Excellent learners can establish an effective cycle between the three stages of planning, execution and reflection (Cleary & Zimmerman, 2012). These cognitive theories provide an important theoretical basis for designing intelligent systems with adaptive capabilities.

2.2 Reinforcement learning methods and their limitations

In reinforcement learning, the balance between exploration and exploitation is crucial to the learning effect of the agent. Traditional methods such as ε-greedy and Softmax are effective in static environments, but they have limitations such as fixed exploration rates that are difficult to adapt to dynamic changes and are prone to falling into suboptimal strategies in high-dimensional environments (Foucault & Meyniel, 2024; Sutton & Barto, 2018). To address these issues, researchers have developed improvements such as confidence-based dynamic exploration strategies and priority experience replay techniques, showing that the introduction of dynamic adjustment mechanisms can help enhance the performance of reinforcement learning in complex environments (Yu & Dayan, 2005; Collins & Frank, 2014; Lake et al., 2017).

At the same time, deep reinforcement learning methods have been used to improve the adaptability of intelligent agents in dynamic environments. DQN improves its ability to handle complex state spaces by combining deep neural networks and experience replay mechanisms, but its fixed experience replay strategy does not work well in dynamic environments (Mnih et al., 2015; Radulescu et al., 2021). Although the PPO algorithm improves learning stability, its conservative adjustment mechanism also reduces the response speed to environmental changes (Wang et al., 2018).

2.3 Application and Challenges of Meta-Learning in Dynamic Environments

Meta-Learning solves the adaptability challenges faced by reinforcement learning in dynamic environments by ‘learning how to learn’. Its core idea is to enable agents to acquire general learning strategies and the ability to quickly adapt to new tasks through training across

multiple tasks, similar to how humans quickly learn new things through past experience (Chalvidal et al., 2022). Among them, Model-Agnostic Meta-Learning (MAML) optimizes model initialization parameters to enable agents to quickly adapt to new tasks after a small number of gradient updates, but requires a lot of computing resources and focuses on knowledge transfer rather than continuous optimization of a single task (Gupta et al., 2018; Tao et al., 2024). RL^2 methods focus on optimizing exploration strategies, but problems such as strategy forgetting or adaptation lag will occur when the environment changes rapidly (Norman & Clune, 2024). These limitations indicate that existing meta-learning methods still need to be improved in dealing with continuous adaptation problems in dynamic environments.

3 Methods

3.1 Algorithm Framework

This study proposes a reinforcement learning algorithm based on a two-layer reflection mechanism with confidence evaluation. The algorithm framework includes four core modules (as shown in Figure 1): learning module, confidence evaluation module, fast adaptation module and reflection module.

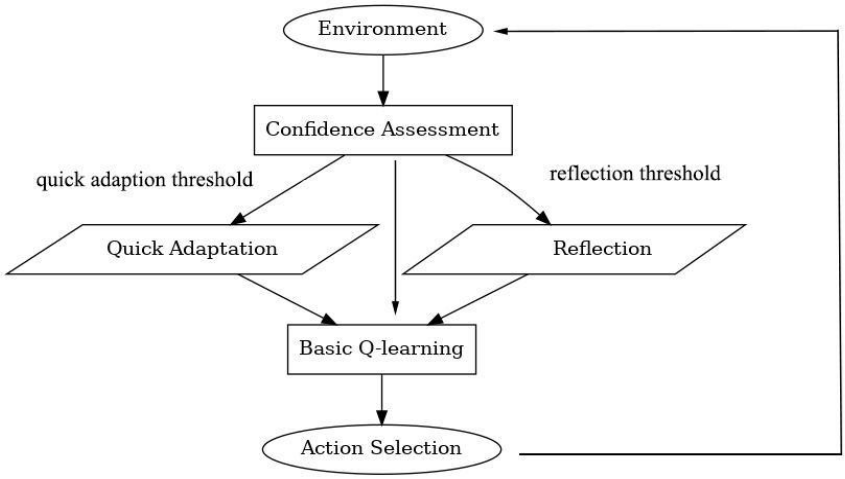


Figure 1: Dual-level Reflection Framework for Reinforcement Learning with Confidence-based Adaptation

(1) Learning module

The basic learning module uses the Q-learning algorithm to establish the interaction between the agent and the environment and is responsible for the basic decision-making process.

(2) Confidence evaluation module

The confidence evaluation module monitors the performance of the agent in real time and dynamically calculates the confidence to trigger the fast adaptation layer or reflection layer.

(3) Fast adaptation module

The fast adaptation module is triggered by monitoring whether the confidence difference between adjacent episodes exceeds the threshold. After triggering, according to the comparison result between the confidence and the preset critical value, different degrees of exploration rate and learning rate parameter adjustment are adopted to achieve immediate response to environmental changes.

(4) Reflection module

The reflection module is triggered when it detects that the recent confidence level is continuously below the threshold with a small fluctuation range, or shows a significant downward trend. After being triggered, the module dynamically adjusts the exploration rate and learning rate parameters based on the relationship between the path efficiency and the critical value, and combines the optimization experience replay strategy to improve the long-term learning effect of the intelligent agent.

3.2 Confidence Evaluation System

This study designed a multi-dimensional confidence evaluation system to comprehensively analyze the behavior performance of the agent and conduct real-time evaluation. The confidence can be calculated as:

$$\text{confidence} = 0.3 \times path\_efficiency + 0.3 \times reward\_stability + 0.4 \times success\_rate \quad (1)$$

Wherein, the definitions of each indicator are as follows:

(1) Path efficiency

Path efficiency measures the degree of path optimization when the agent performs a task. The calculation formula is as follows:

$$path\_efficiency = \min(L_{optimal} / L_{actual}, 1.0) \quad (2)$$

Where $L_{optimal}$ represents the shortest path length, and L_{actual} represents the actual path length. When the ratio is closer to 1, it indicates that the path selected by the agent is better.

(2) Reward stability

Reward stability is used to measure the consistency of the agent's decision-making. The calculation formula is as follows:

$$reward\_stability = \frac{1}{1 + \sigma(\mathrm{Rt})} \quad (3)$$

Where $\sigma(\mathrm{Rt})$ is the standard deviation of the reward for the most recent t steps. The smaller the value, the lower the reward fluctuation and the more stable the agent's behavior.

(3) Success rate

The success rate represents the proportion of successful completion of the agent in the most recent n tasks. The calculation formula is as follows:

$$success\_rate = \frac{\mathrm{N}_{success}}{n} \tag{4}$$

This indicator uses a sliding window mechanism to continuously update the completion status of the last n tasks.

The weights (0.3, 0.3, 0.4) in the confidence calculation are determined based on preliminary experimental tuning to balance the contribution of path efficiency, reward stability, and success rate to confidence.

3.3 Algorithm Implementation

This section introduces the specific implementation of three agents: the basic ε-greedy strategy agent, the confidence agent, and the double-layer reflection agent. The experimental part will verify the effectiveness of the confidence mechanism and the double-layer reflection mechanism in turn through two sets of progressive comparative experiments.

3.3.1 Basic Agent Implementation

In Experiment 1, we compared the baseline ε-greedy strategy agent and the improved confidence agent based on it.

(1) ε-greedy strategy agent

The basic agent uses the standard Q-learning algorithm combined with the ε-greedy strategy for action selection. The Q value is updated using:

$$\mathrm{Q}(s,a) \leftarrow \mathrm{Q}(s,a) + \alpha\left[r + \gamma \max_{a'} \mathrm{Q}(s',a') - \mathrm{Q}(s,a)\right] \tag{5}$$

The exploration rate ε uses a multiplicative decay strategy:

$$\varepsilon_{\mathrm{t}} = \max(\varepsilon_{\min}, \varepsilon_{t-1} \times decay) \tag{6}$$

Where ε_0 =1.0 is the initial exploration rate, ε_{min} =0.3 is the minimum exploration rate, and decay is the decay rate parameter (0.999). Other parameter configurations are as follows:

The learning rate α is set to 0.05

The discount factor γ is set to 0.9

(2) Confidence agent

Based on the ε-greedy strategy, the confidence agent introduces a confidence-driven exploration rate adjustment mechanism. When the confidence is lower than 0.3, the exploration rate is set to the maximum value (1.0); when the confidence is higher than 0.7, the exploration rate is reduced to 0.3; when the confidence is between 0.3 and 0.7, the exploration rate is adjusted using inverse linear interpolation. The specific calculation formula is as follows:

$$\varepsilon = \begin{cases} 1.0, \text{confidence} \le 0.3 \\ 0.3, \text{confidence} \ge 0.7 \\ 0.3 + (1.0 - 0.3) \times (1 - \text{confidence}), 0.3 < \text{confidence} < 0.7 \end{cases} \quad (7)$$

3.3.2 Agent with two-layer reflection mechanism

In order to improve the adaptability of the agent in a dynamic environment, this study further expands on the confidence agent and introduces a two-layer reflection mechanism, including a fast adaptation layer and a reflection layer. After multiple sets of experimental verification, (0.25, 0.45) was selected as the final threshold configuration. The detailed analysis of the threshold parameters will be discussed in Section4.2.3.

(1) Fast adaptation layer

The fast adaptation layer aims to improve the ability of the agent to cope with short-term performance degradation and achieves immediate response by monitoring the confidence changes between adjacent episodes. When the confidence drop between two adjacent episodes exceeds 0.25, the fast adaptation mechanism is triggered. After being triggered, the system calculates the average confidence of the last three time steps and adjusts the parameters accordingly: if the average confidence is lower than 0.3, the exploration rate (1.05 times the current value, upper limit 0.9) and learning rate (1.02 times the current value, upper limit 0.3) are greatly increased to accelerate the policy update; if the average confidence is not lower than 0.3, the exploration rate (1.03 times the current value, upper limit 0.8) and learning rate (1.01 times the current value, upper limit 0.25) are moderately increased to fine-tune the current policy.

(2) Reflection layer

The reflection layer is mainly used to evaluate the learning trend of the agent in the process of continuous interaction, and triggers policy adjustment by detecting the change of confidence in the recent window. There are two triggering conditions: one is that when the average confidence of the last 8 state-action interactions is less than 0.45 and the confidence variance is less than 0.1, it indicates that the agent is in a stable inefficient state; the other is that when the confidence trend (the mean difference between the second half and the first half of the window) is less than -0.15 and the current trajectory length exceeds 10, it indicates that the performance is continuously declining. After the reflection is triggered, the system selects an adjustment strategy based on the path efficiency. If the path efficiency is less than 0.1, an aggressive adjustment is made to increase the exploration rate and learning rate to 1.08 times (upper limit 0.95) and 1.03 times (upper limit 0.4) of the current value respectively; if the path efficiency is not less than 0.1, a moderate adjustment is made to increase the exploration rate and learning rate to 1.03 times (upper limit 0.8) and 1.01 times (upper limit 0.25) of the current value respectively. In addition, the deep reflection layer also

introduces a priority-based experience replay mechanism. This mechanism maintains an experience pool with a capacity of 2000 and uses the Softmax sampling strategy to select replay samples:

$$P(i) = \frac{\exp(r_i)}{\sum_j \exp(r_j)} \tag{8}$$

Where $P(i)$ represents the probability of experience sample i being selected, and r_i is the reward value of the experience sample.

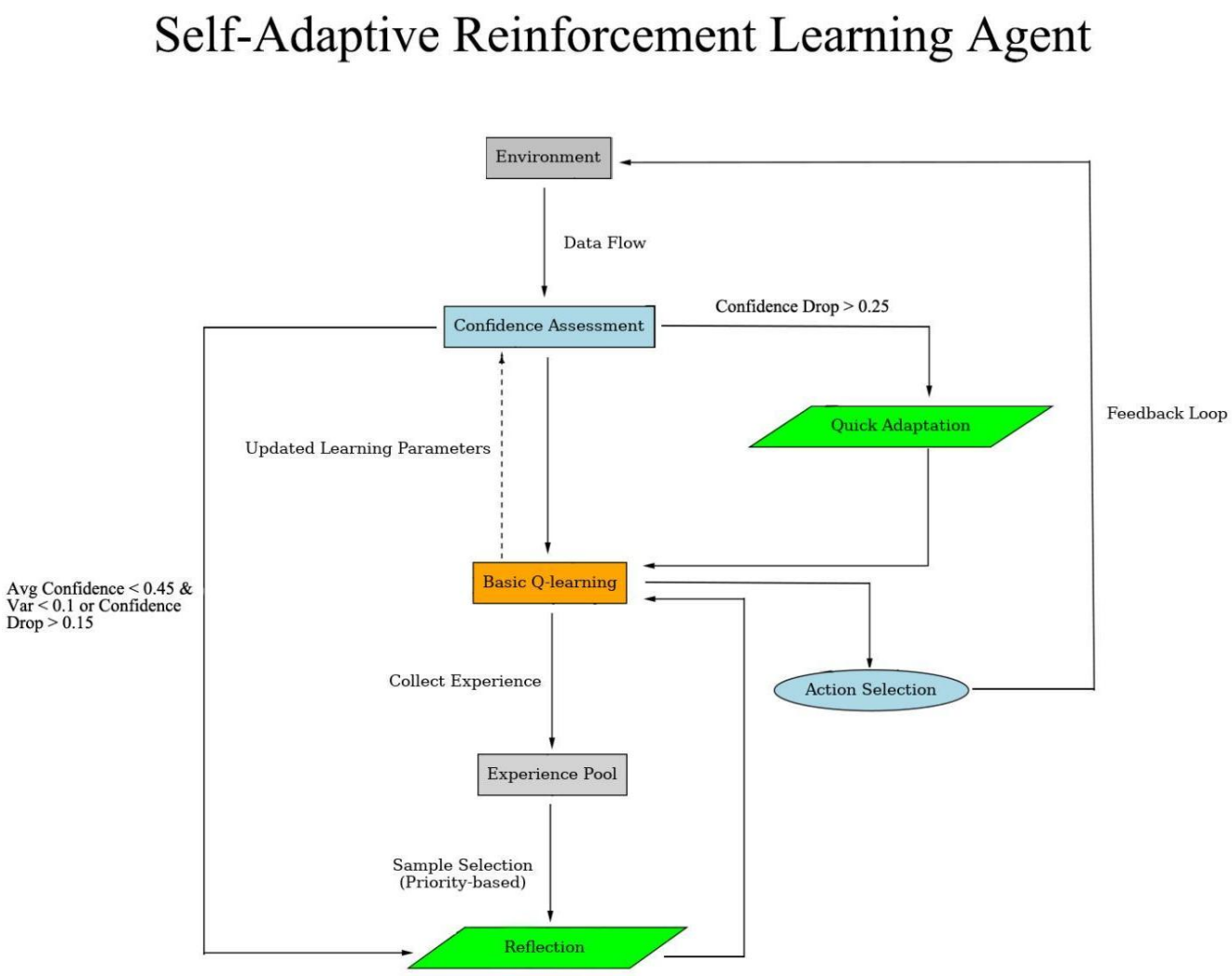


Figure 2 shows the overall framework of the two-layer reflection mechanism proposed in this paper, including three core modules of confidence assessment, rapid adaptation and reflection and their interactive relationships.

3.4 Parameter Adaptive Adjustment

3.4.1 Dynamic Adjustment of Exploration Rate

The adjustment of the exploration rate (ε) needs to strike a balance between maintaining the stability of the strategy and maintaining the adaptability. This study designed a multi-factor complex dynamic adjustment formula:

$$\varepsilon = \max(\varepsilon_{\min}, \varepsilon \times decay \times reflcetion\_factor \times success\_factor) \tag{9}$$

$\varepsilon_{\min}$ is the minimum exploration rate (0.01), which ensures that the agent always has a certain probability of exploration to prevent it from being completely trapped in a deterministic strategy. Decay is the basic decay rate (0.997), which makes the exploration rate slowly decrease with the training process, guiding the learning process from exploration-oriented to utilization-oriented.

reflection_factor is determined by the number of reflections, as shown in formula (10):

$$reflection\_factor = 1 + 0.005 \times reflection\_count \tag{10}$$

This factor takes the triggering frequency of the reflection mechanism into account when adjusting the exploration rate. The accumulation of reflections will moderately increase the exploration rate, allowing the agent to increase exploration to find a better strategy when reflection is frequently needed.

success_factor is calculated based on the success rate of recent tasks, as shown in formula (11):

$$success\_factor = 1 - 0.1 \times recent\_success\_rate \tag{11}$$

This factor adjusts the exploration probability according to the success rate of recent tasks, reducing exploration when the success rate is high and increasing exploration when it is low. This adaptive mechanism helps maintain stability when the strategy performs well and increases exploration when it performs poorly.

3.4.2 Dynamic adjustment of learning rate

The adjustment of learning rate (α) adopts an adaptive method based on visit frequency:

$$\alpha = \frac{\alpha_0}{1 + 0.05\sqrt{visit\_count}} \tag{12}$$

where α_0 represents the initial learning rate, and visit_count records the number of visits to the state-action pair. As the number of visits increases, the learning rate gradually decreases to reduce the update frequency of the fully explored area, thereby maintaining the stability of the mastered strategy.

The learning rate and exploration rate adjustment mechanism and the double-layer reflection mechanism cooperate with each other to form a complete parameter adjustment system. Specifically,

when the agent reflects, it will affect the adjustment coefficient of the exploration rate, making the exploration behavior more targeted. At the same time, this learning rate adjustment mechanism based on visit frequency enables the agent to automatically adjust parameters according to experience during the learning process, effectively balancing exploration and utilization.

3.5 Experimental Verification and Evaluation

In order to verify the effectiveness of the proposed method, this section introduces the experimental environment design, evaluation method and specific experimental configuration in detail.

3.5.1 Experimental Environment Configuration

This study designed two experimental environments, static maze and dynamic maze, to comprehensively evaluate the learning efficiency and adaptability of the agent in different scenarios. The static maze is shown in Figure 3. It adopts a fixed 10×10 grid layout, an obstacle density of 20%, a constant target position, and a maximum step limit of 50 steps per round. It is mainly used to test the basic learning ability and exploration efficiency of the agent.

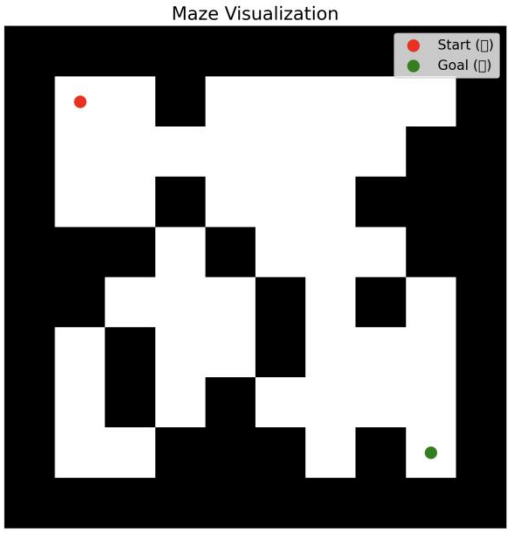


Figure 3: Schematic diagram of static maze environment

The dynamic maze maintains the same grid layout, but the obstacle density is increased to 25%, and a dynamic change mechanism of the environment is introduced. Specifically, the obstacle is randomly adjusted by 1-3 positions every 18 steps to simulate the path change in the real environment; the target position has a 30% probability of changing each time the environment is updated, but it is ensured that it will not overlap with the agent position or fall into the obstacle. The maximum number of steps per episode is set to 200 steps.

3.5.2 Evaluation indicators

The experiment uses the following four key indicators to measure the learning performance of the agent. The average reward (MCR = (1/N)∑Ri) reflects the overall benefit level, where N is the number of episodes and Ri is the total reward of the i-th episode. The task success rate (SR = Ns/N) evaluates the reliability of completing the task, where Ns represents the number of episodes that successfully reach the target. Path efficiency ($PE = \min(1.0, \frac{L_{opt}}{L_{act}})$) measures the degree of decision optimization, where Lopt is the optimal path length and Lact is the actual path length. Reward stability ($RS = \frac{1}{1 + mean(|r_t - r_{t-1}|)}$) is used to evaluate the consistency of the agent's behavior, where Rt represents the total reward of the t-th episode.

3.5.3 Reward mechanism

The reward mechanism includes task completion rewards, obstacle avoidance penalties, and distance-oriented rewards. The agent receives the highest positive reward when it successfully reaches the target location, and receives a negative reward when it collides with an obstacle. At the same time, it receives staged rewards or penalties based on the change in distance from the target point.

3.5.4 Experimental environment configuration

The experiment was repeated 30 times under each environment configuration to ensure the statistical reliability of the experimental results, and all experiments were performed under the same hardware environment and parameter settings.

4 Experimental Results

This section will conduct comparative experiments in static maze environments and dynamic maze environments. In the static maze environment, the performance difference between the ε-greedy strategy agent and the confidence agent is compared, and in the dynamic maze environment, the performance between the confidence agent and the double-layer reflective agent is compared. Through these two sets of experiments, the effectiveness of the proposed method is gradually verified.

4.1 Experiment in static maze environment

4.1.1 Experimental setting

The static maze experiment is based on the fixed maze environment in Section 3.5.1. The focus of the experiment is to evaluate the learning efficiency and strategy optimization ability of the confidence agent in a stable environment.

The experiment compares two agent models:

ε-greedy strategy agent: adopts the ε-greedy strategy, the exploration rate decays from 1.0 to 0.3 according to the exponential law, and there is no confidence adjustment mechanism.

Confidence Agent:Adaptive exploration strategy based on confidence. The exploration rate is dynamically adjusted according to the confidence of comprehensive evaluation of path efficiency, reward stability and success rate (see Section 3.3.1 for details).

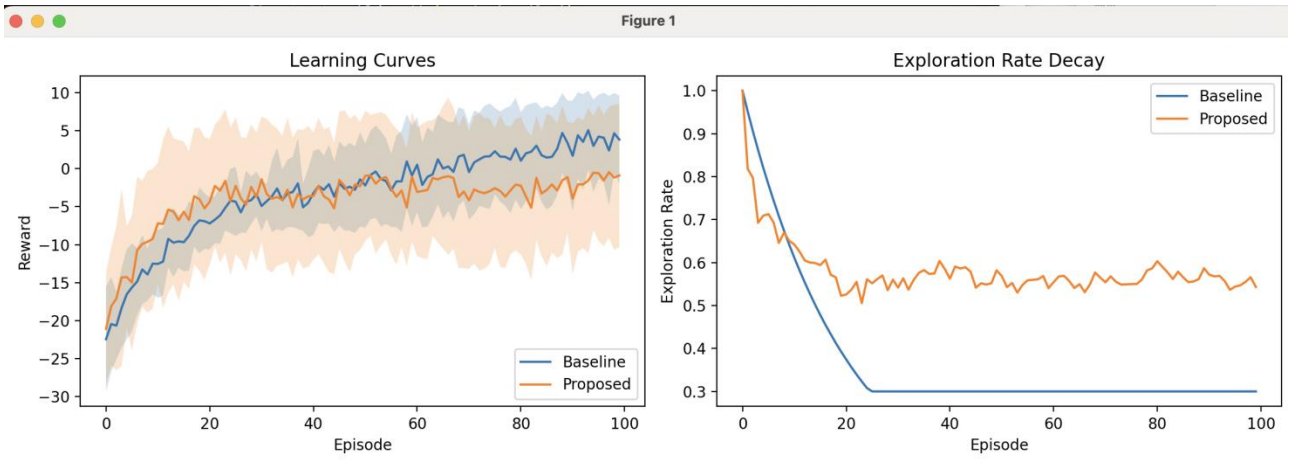
4.1.2 Performance Evaluation

The experiment compares the performance of the ε-greedy strategy agent (Baseline Agent) and the confidence agent (Proposed Agent) in a static maze environment, focusing on the evaluation of three key indicators: Cumulative Reward, Average Steps and Success Rate.

Table 1 Comparison of agent performance in a static maze environment

Model	Average reward	Average number of steps	Success rate
ε-greedy strategy agent	-2.92	44.13	0.32
Confidence agent	-1.69	37.51	0.50

As shown in Table 1, in the static environment, the confidence agent significantly outperforms the ε-greedy agent in both task success rate (50%) and average number of steps (37.51).



(A) (B)

Figure 4:Learning performance comparison between ε-greedy and confidence agents. (A) Learning curves showing the average reward with shaded areas representing variance across multiple runs. (B) Exploration rate decay curves over episodes.

The learning curves (A) demonstrate that the ε-greedy agent exhibits delayed convergence with narrower confidence intervals, indicating a more conservative learning strategy. In contrast,

the reward of the confidence agent increases rapidly and stabilizes after 20 tasks, showing the role of the adaptive exploration strategy in promoting learning efficiency.

The exploration rate decay curves (B) reveal that the confidence-based agent reduces its exploration rate more rapidly during initial learning compared to the ε-greedy agent. While the ε-greedy agent's exploration rate fixates at 0.3 after 20 episodes, the proposed agent maintains a dynamic rate around 0.55 in the later stages, demonstrating sustained strategic adaptation capability.

The experimental results show that the confidence-based adaptive exploration strategy can effectively balance exploration and utilization, and improve the learning efficiency and stability of the agent in a static environment.

4.2 Dynamic maze environment experiment

4.2.1 Experimental setup

The dynamic characteristics and configuration of the environment have been described in detail in Section 3.5.1 of the dynamic maze experiment. The purpose of this experiment is to test the performance improvement of the reflective agent in a dynamic environment and verify its adaptability to dynamic target positions and obstacles.

The experiment compares two agent models:

Confidence agent: an agent model that adopts the confidence-driven exploration strategy described in Section 3.3.1. This model performs better than the ε-greedy strategy in the static maze experiment, so it is used as the control group of this experiment.

Reflection Agent: Based on the confidence agent, the fast adaptation layer and the deep reflection layer (see Section 3.3.2 for details) are integrated to optimize learning and decision-making in a dynamic environment.

4.2.2 Performance evaluation

The experimental results of the dynamic environment show that the reflective agent has significant improvements in multiple performance indicators compared with the confidence agent. The specific results are shown in Table 2 and Figure 5.

Table 2: Dynamic environment performance comparison

Model	Success rate	Average number of steps	Average reward	Path efficiency	Reward stability
Confidence agent	37.9%	154.89	-3.16	0.072	0.876
Reflection Agent	50.3%	142.83	-3.66	0.067	0.845

Table 2 lists the comparative performance indicators of the confidence agent and the reflective agent. In terms of success rate, the reflective agent reached 50.3%, which is 32.7% higher than the confidence agent's 37.9%. It can be further observed from Figure 5 (upper middle) that the success rate of the reflective agent remains higher than that of the confidence agent throughout the training process, and stabilizes at about 0.5 in the later stage of training, while the success rate of the confidence agent stabilizes at about 0.37. This shows that the reflective agent can adapt to environmental changes more effectively and improve the task completion rate.

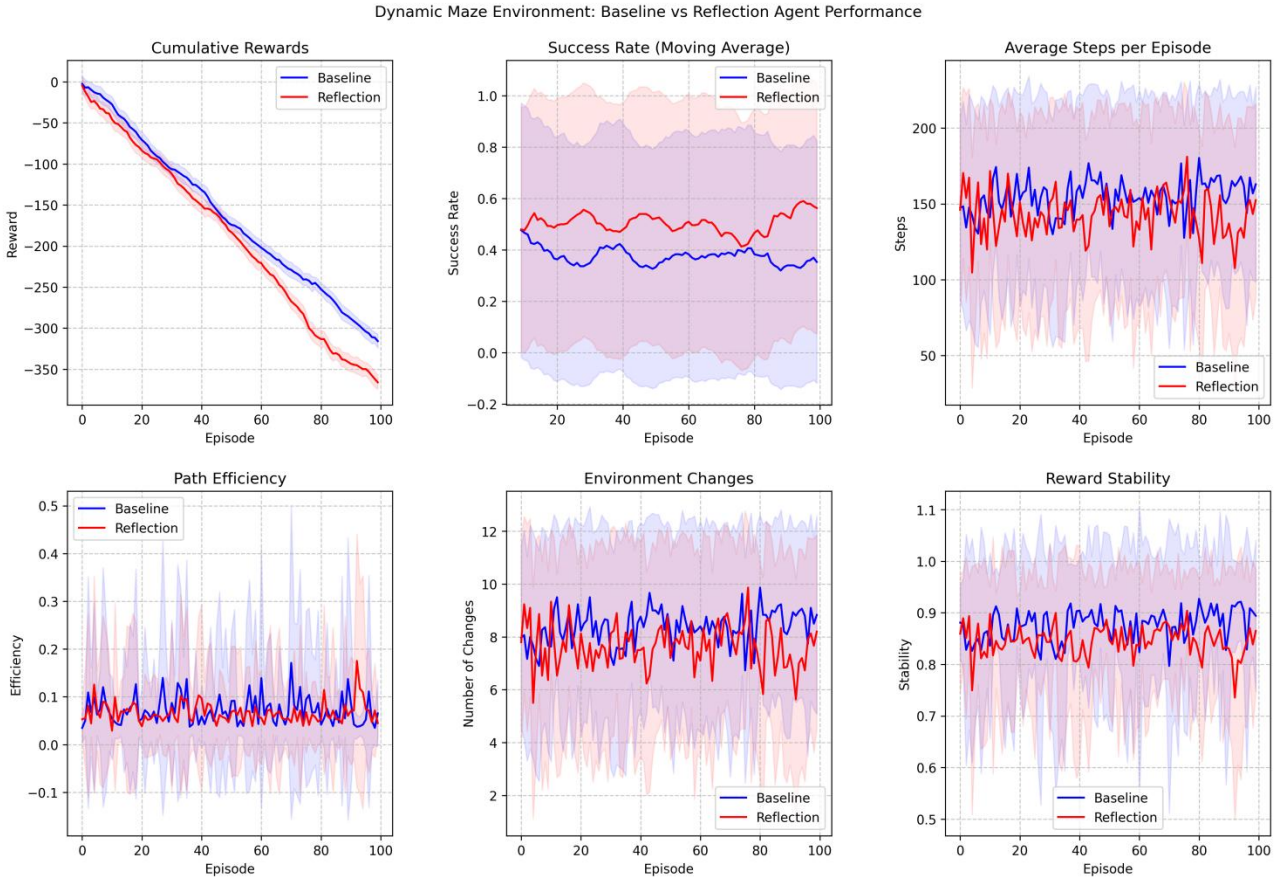


Figure 5 shows the performance comparison of the confidence agent and the reflective agent in a dynamic environment.

In terms of average steps, the average number of steps of the reflective agent is 142.83, which is 7.8% less than that of the confidence agent's 154.89. The average step curve (upper right) shows that the step number of the reflective agent fluctuates greatly, especially in the stage where the environment changes frequently, which may be related to the reflective agent's more exploration in the dynamic environment.

The cumulative reward curve shows that the cumulative reward of the reflective agent decreases faster than that of the confidence agent. Although the overall trend of both is decreasing, the reflective agent has lower cumulative rewards in the dynamic environment, indicating that it focuses more on exploration rather than benefit optimization in the short term.

The path efficiency indicator shows that the path efficiency of the reflective agent is 0.067, slightly lower than the 0.072 of the confidence agent. The path efficiency curve (lower left) shows that the path efficiencies of the two agents are very close, indicating that the path selected by the reflective agent in the dynamic environment may not always be optimal, but it can still complete the task effectively.

The number of environmental changes curve (lower middle) shows that the number of environmental changes experienced by the two models is basically the same, fluctuating between 6 and 9 times.

The reward stability curve (lower right) shows that the reflective agent shows a higher tendency to explore in the dynamic environment, which may lead to slightly larger reward fluctuations in the short term. The confidence agent has a smaller reward fluctuation due to its more conservative strategy. However, this fluctuation does not affect the long-term task completion ability of the reflective agent, and its success rate and average number of steps are still better than the baseline model.

4.2.3 Threshold parameter analysis

In order to verify the effect of the reflection mechanism, this paper studies the impact of thresholds on model performance in a dynamic environment. The three threshold pairs (lower_threshold, upper_threshold) are (0.2, 0.4), (0.25, 0.45) and (0.3, 0.5), and their specific triggering logic has been described in detail in Section 3.3.2. As shown in Figure 6, the experiment evaluates the performance of different configurations from four dimensions: Success Rate, Average Steps, Path Efficiency and Reward Stability.

The experimental results show that all reflective agent configurations have achieved significant improvements compared to the control group (success rate 37.9%, average number of steps 154.89). Under the configuration of (0.25, 0.45), the agent achieves the best comprehensive performance: success rate 50.3%, average number of steps 142.83. The other two configurations also performed well: (0.2, 0.4) achieved a success rate of 48.6% and an average number of steps of 144.06; (0.3, 0.5) achieved a success rate of 50.2% and an average number of steps of 142.62.

The experimental results revealed three key findings:

(1) The medium threshold range (such as (0.25, 0.45)) performed the most balanced in terms of various indicators. Lower thresholds resulted in slightly lower success rates, while higher thresholds maintained higher success rates but relatively lower average rewards.

(2) The reflection mechanism is robust to the threshold parameters, which is reflected in the small performance differences among the three configurations (success rates ranged from 48.6% to 50.3%, and path efficiencies were all between 0.066 and 0.067).

(3) While the reflection mechanism significantly improved the success rate (from 37.9% to approximately 50%), the average reward (-3.66 to -3.88) was lower than that of the control group (-3.16), which may reflect the additional exploration cost required to improve the task completion rate.

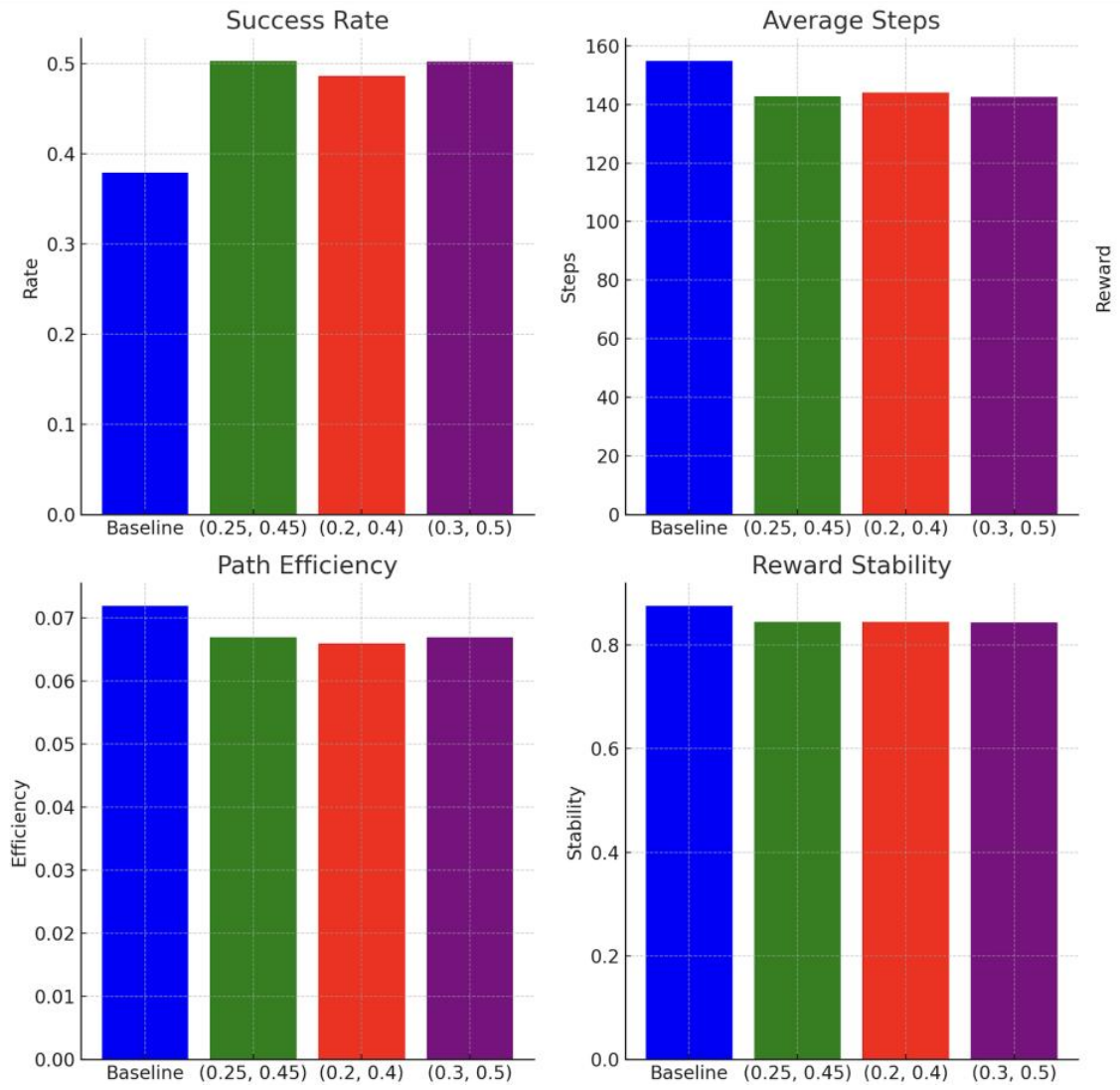


Figure 6:Performance comparison of reflective agents under different threshold configurations

4.3 Results

The experimental results confirm the performance advantages of reflective agents in dynamic environments. The statistical results of 30 independent runs yielded the following key findings:

First, the reflective agent achieved a significant improvement in completion rate, with the success rate increasing from 37.9% of the baseline model to 50.3%. At the same time, the average number of steps decreased from 154.89 to 142.83, indicating that the model's efficiency in path planning has been optimized.

Second, the agent exhibits a trade-off feature between different performance indicators. Although the average reward dropped from -3.16 to -3.66, the path efficiency slightly dropped from 0.072 to 0.067, and the reward stability dropped from 0.876 to 0.845, this slight drop was in exchange for a higher task completion rate.

In addition, the threshold parameter analysis further verified the robustness and adaptability of the reflective mechanism. Under different threshold settings, the parameter combination of (0.25, 0.45) achieves the best balance between success rate and step number optimization while maintaining high reward stability.

5 Discussion

This study simulated two cognitive regulation mechanisms in the human learning process, namely, immediate adaptation and reflective adjustment, through a two-layer reflection mechanism. Experiments show that the fast adaptation layer makes immediate parameter adjustments through short-term confidence evaluation, while the reflective layer triggers adjustments based on performance indicators within a fixed window when it identifies that the agent is trapped in an inefficient decision-making mode. This hierarchical design significantly improves the adaptability of the agent in a dynamic environment. The multi-dimensional confidence evaluation system provides the agent with a more comprehensive decision-making basis.

However, the current reflection mechanism still has some limitations. Although the effects of different threshold combinations have been compared, how to select the appropriate threshold for a specific environment still needs further research. Secondly, in complex environments, frequent triggering of the reflection layer may increase the computational cost and affect the stability of the strategy. In addition, while the task completion rate has increased, the average reward has slightly decreased, indicating that there is a trade-off between different performance indicators.

This study introduces the reflection mechanism into the field of reinforcement learning. The experimental results verify the effectiveness of the reflection mechanism in a dynamic environment, and its hierarchical design achieves a balance between adaptation and optimization. In the future, it is necessary to optimize the computational efficiency of the reflection trigger mechanism and verify its scalability in more complex environments.

6 Conclusion

This study explored the adaptability of reinforcement learning algorithms based on a two-layer reflection mechanism in dynamic environments. The design of this mechanism draws on two types of cognitive regulation in the human learning process: strategy adjustment based on immediate feedback, and optimization of learning strategies through reflection. Through virtual task experiments, we found that this mechanism can effectively improve the adaptability of intelligent agents to environmental changes. Specifically, the fast adaptation layer adjusts parameters based on

the changes in the confidence of adjacent episodes to cope with rapid performance fluctuations, while the reflection layer triggers the adjustment of strategy parameters when it identifies that the intelligent agent falls into an inefficient decision-making mode by analyzing the confidence index within a fixed window. This hierarchical design enables the intelligent agent to respond to environmental changes more flexibly. In terms of evaluation system design, we combine the indicators of path efficiency, reward stability and success rate to provide a more comprehensive decision-making evaluation standard. Experimental results show that this mechanism effectively improves the task completion rate and reduces the average number of steps in a dynamic environment, but it is also accompanied by certain reward fluctuations. These findings not only provide new ideas for improving the performance of reinforcement learning algorithms in dynamic environments, but also demonstrate the potential of combining cognitive science theory with artificial intelligence algorithm design, laying the foundation for building intelligent systems with stronger environmental adaptability. Future research can conduct in-depth exploration in aspects such as adaptive adjustment of thresholds and reducing reliance on random exploration, so as to further improve this learning method based on human cognitive mechanisms.

References

Chalvidal, M., Serre, T., & VanRullen, R. (2022). Meta-reinforcement learning with self-modifying networks. *arXiv.* https://arxiv.org/abs/2202.02363

Cleary, T., & Zimmerman, B. J. (2012). A cyclical self-regulatory account of student engagement: Theoretical foundations and applications. In S. L. Christenson & W. Reschley (Eds.), *Handbook of research on student engagement* (pp. 237–257). Springer Science. https://doi.org/10.1007/978-1-4614-2018-7_11

Collins, A. G., & Frank, M. J. (2014). Opponent actor learning (OpAL): Modeling interactive effects of striatal dopamine on reinforcement learning and choice incentive. *Psychological Review, 121*(3), 337–366. https://doi.org/10.1037/a0037015

Flavell, J. H. (1979). Metacognition and cognitive monitoring: A new area of cognitive-developmental inquiry. *American Psychologist, 34*(10), 906–911. https://doi.org/10.1037/0003-066X.34.10.906

Fleur, D. S., Bredeweg, B., & van den Bos, W. (2021). Metacognition: Ideas and insights from neuro- and educational sciences. *npj Science of Learning, 6*(1), 13. https://doi.org/10.1038/s41539-021-00089-5

Foucault, C., & Meyniel, F. (2024). Two determinants of dynamic adaptive learning for magnitudes and probabilities. *Open Mind: Discoveries in Cognitive Science, 8,* 615–638. https://doi.org/10.1162/opmi_a_00139

Gupta, A., Mendonca, R., Liu, Y., Abbeel, P., & Levine, S. (2018). Meta-reinforcement learning of structured exploration strategies. *arXiv.* https://arxiv.org/abs/1802.07245

Lake, B. M., Ullman, T. D., Tenenbaum, J. B., & Gershman, S. J. (2017). Building machines that learn and think like people. *Behavioral and Brain Sciences, 40,* e253. https://doi.org/10.1017/S0140525X16001837

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Hassabis, D., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., & Graves, A. (2015). Human-level control through deep reinforcement learning. *Nature, 518*(7540), 529–533. https://doi.org/10.1038/nature14236

Nelson, T. O., & Narens, L. (1990). Metamemory framework. In R. V. Kail & J. W. Hag (Eds.), *Perspectives on the development of memory and cognition* (pp. 3–33). Erlbaum. https://doi.org/10.1016/S0079-7421(08)60053-5

Norman, B., & Clune, J. (2024). First-explore, then exploit: Meta-learning to solve hard exploration-exploitation trade-offs. *arXiv.* https://arxiv.org/abs/2307.02276

Ochilova, V. R. (2021). Metacognition and its history. *Frontline Social Sciences and History Journal, 1*(3), 18–44. https://doi.org/10.37547/social-fsshj-01-03-04

Radulescu, A., Shin, Y. S., & Niv, Y. (2021). Human representation learning. *Annual Review of Neuroscience, 44,* 253–273. https://doi.org/10.1146/annurev-neuro-092920-120559

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction* (2nd ed.). MIT Press. https://web.stanford.edu/class/psych209/Readings/SuttonBartoIPRLBook2ndEd.pdf

Tao, Z., Lin, T. E., Chen, X., Li, H., Wu, Y., Li, Y., Zhou, J., & Zhang, Y. (2024). A survey on self-evolution of large language models. *arXiv.* https://arxiv.org/abs/2404.14387

Wang, J. X., Kurth-Nelson, Z., Kumaran, D., Tirumala, D., Soyer, H., Leibo, J. Z., Botvinick, M., & Vinyals, O. (2018). Prefrontal cortex as a meta-reinforcement learning system. *Nature Neuroscience, 21*(6), 860–868. https://doi.org/10.1038/s41593-018-0147-8

Yu, A. J., & Dayan, P. (2005). Uncertainty, neuromodulation, and attention. *Neuron, 46*(4), 681–692. https://doi.org/10.1016/j.neuron.2005.04.026