

An imitation-enhanced actor-critic approach for maze navigation^aNurgül Kalaycı^{*1}, Mehmet Dinçer Erbaş²¹Department of Software Engineering, Faculty of Engineering, Iğdır University, Iğdır, Türkiye²Department of Computer Engineering, Faculty of Engineering, Bolu Abant İzzet Baysal University, Bolu, Türkiye

Cite this article as: Kalaycı, N., & Erbaş, M. D. (2026). An imitation-enhanced actor-critic approach for maze navigation. *J Comp Electr Electron Eng Sci*, 4(2), 64-71. <https://doi.org/10.51271/JCEES-0038>

Received: 27.01.2026

Accepted: 01.04.2026

Published: 13.10.2026

ABSTRACT

Aims: This paper proposes the imitation-enhanced actor-critic (IEAC) method that integrates imitation learning (IL) into the actor-critic (AC) algorithm and evaluates its effect on the learning speed of agents in both obstacle and obstacle-free mazes. The study aims to evaluate and compare the learning efficiency of IEAC against both standard AC and tabular Q-learning algorithms in diverse maze environments.

Methods: The proposed approach combines AC with a probabilistic observational imitation model that operates without direct internal access or inter-agent experience sharing. IEAC's performance was benchmarked against AC, tabular Q-learning, and behavioral cloning (BC) through comprehensive simulations in 20x20 and 50x50 maze configurations.

Results: Simulation results indicate that IEAC significantly accelerates learning and achieves higher stability compared to the standard AC and Q-learning baselines. Specifically, IEAC agents reached optimal paths with substantially fewer environmental steps. Unlike the BC baseline, which failed in obstacle-heavy settings, IEAC successfully balanced guided imitation with independent exploration to overcome local optima.

Conclusion: The findings demonstrate that augmenting the AC framework with stochastic imitation enhances convergence speed and robustness, particularly as environmental complexity increases. The proposed IEAC approach offers a scalable and efficient solution for autonomous navigation tasks where rapid learning from observation is required.

Keywords: Social learning, reinforcement learning, multi-agent systems

^aThis study was presented as a conference paper at the International Conference on ICE 2025 (ICE2025), held in Ankara, Türkiye, on November 6, 2025, and its abstract was published in the conference proceedings.

INTRODUCTION

Social learning is an innate process observed in both animals and humans. Due to the limitations and inefficiencies of individual trial-and-error learning, social learning strategies, in which individuals learn from observing or interacting with others, have evolved as advantageous mechanisms (Henrich, 2016). Imitation, one of the main forms of social learning, has attracted attention in both biological and artificial systems. In autonomous agents, imitation learning (IL) has been achieved through experience sharing (Bredeche & Fontbonne, 2022; Christianos et al., 2021; Uwano et al., 2018), learning from demonstrations provided by a leader or expert (Argall et al., 2009; Block et al., 2024; Guenter & Billard, 2007; Mathewson & Pilarski, 2017; Rao et al., 2007), or imitation through observation (Bentivegna et al., 2005; Erbas et al., 2014), which closely reflects the imitation process in humans by reproducing superior observed behaviours. The imitation strategy adopted in this study relies on observational learning, where the agent reproduces observed behaviours without direct internal access to the leader's parameters.

Reinforcement learning (RL) is a fundamental paradigm in machine learning that enables agents to learn through trial and error by updating their policies according to feedback

from rewards. Although RL enables agents to discover solutions autonomously, the exploration process often results in slow convergence, especially in complex environments. Since IL relies on community-based rather than individual exploration, integrating IL with reward-based RL algorithms—especially in multi-agent systems—offers potential benefits. Indeed, this idea has been explored extensively in the literature (Christianos et al., 2021; Codevilla et al., 2018; Erbas et al., 2014; Guenter & Billard, 2007; Guo et al., 2019; Ndousse et al., 2021; Nguyen, 2024; Sallab et al., 2017; Uwano et al., 2018; Yang et al., 2024).

The maze solving problem is one of the most widely used benchmarks for testing learning algorithms, with applications in robotics and autonomous navigation (Rao et al., 2007; Erbas et al., 2014; Uwano et al., 2018; Ndousse et al., 2021; Sharma et al., 2023). Using mazes with obstacles increases the complexity of navigation, making the task more representative of real-world conditions and allowing a clearer evaluation of algorithmic performance and generalization.

This study proposes the imitation enhanced actor-critic (IEAC) model, which combines IL with the actor-critic (AC) algorithm to improve learning efficiency. The aim of this

*Corresponding Author: Nurgül Kalaycı, nurgul.kalayci@igdir.edu.tr



study is to simulate and compare the learning performance of agents navigating from a starting point to a goal in both obstacle and obstacle-free maze environments. The primary objective is to evaluate the performance disparities between the standard AC and the proposed IEAC models, thereby demonstrating the efficacy of the IEAC approach across diverse maze configurations.

Rather than proposing a fundamentally new learning paradigm, this study investigates a simple imitation mechanism integrated into the classical AC framework. Unlike complex hybrid IL+RL models that rely on continuous experience sharing or deep neural networks, the proposed approach isolates the impact of storing the leader's path in a discrete memory and allowing the follower agent to utilize it through a probabilistic mode-switching mechanism. The primary objective is to experimentally analyze the isolated effect of this stochastic imitation on learning speed and stability in maze navigation tasks.

The following section introduces the hybrid learning method and the problem definition. Then, the experimental results are presented, followed by a concluding discussion.

METHODS

Ethics

Ethical approval was not required for this study as it does not involve human participants, animal subjects, or identifiable personal data. All procedures were carried out in accordance with the ethical rules and principles.

Study Design

This study examines the goal-reaching performance of two agents in the same maze environment: one is trained with the classical AC algorithm and the other with the proposed IEAC model. This experimental setup allows a direct comparison of learning speed and stability. Before introducing the experimental environment, some definitions are useful for clarifying the agents' learning mechanisms.

Prediction problems in RL can be addressed using different estimation approaches, most notably Monte Carlo (MC) methods and temporal difference (TD) learning. While MC methods rely on complete episodes to update value estimates, TD learning performs incremental updates based on bootstrapped predictions. Among these, TD methods are particularly effective because they update predictions based on subsequent estimates. A specific and widely used form of TD learning is TD (0), which performs single-step updates based on the current reward and the estimated value of the next state. The core of this approach is the TD-error denoted by δ_t , whose value at time t is defined as:

$$\delta_t = R_{t+1} + \gamma V(S_{t+1}) - V(S_t), \quad (1.1)$$

where R_{t+1} is the reward received after transitioning from state S_t to S_{t+1} , γ is the discount factor determining the importance of future rewards, $V(S_{t+1})$ is the estimated value of the next state, and $V(S_t)$ is the current estimate. This error updates $V(S_t)$ toward a more accurate value (Sutton & Barto, 2014). The actor and critic update their policy and state-value function according to:

$$V(S_t) \leftarrow V(S_t) + \alpha_c \times \delta_t \quad \text{and} \quad \pi(A|S_t) \leftarrow \pi(A|S_t) + \alpha_a \times \delta_t, \quad (1.2)$$

where $\alpha_c=0.1$ and $\alpha_a=0.2$ are the learning rates for the critic and actor, respectively. The discount factor is $\gamma=0.8$ and the exploration followed an ϵ -greedy policy with $\epsilon=0.1$, for limited random exploration.

The experimental environment consists of two both 20x20 and 50x50 mazes: one without obstacles (empty maze, **Figure 1 and 2** for 20x20 and 50x50 dimensions, respectively.) and the other containing obstacles at specific locations (obstacle maze, see **Figure 3 and 4** for 20x20 and 50x50 dimensions, respectively). The agents, without any direct information exchange or leader-follower interaction, attempt to reach the goal at the bottom-right corner starting from the top-left corner. As shown in **Figure 1**, the agents can move in eight different directions. The shortest paths leading to the goal in the empty and obstacle mazes are illustrated in **Figure 2 and 4**, respectively. At each step, agents receive a reward of +300 upon reaching the goal, a penalty of -10 for hitting a wall or crossing the maze boundary, and a step cost of -0.5 for each movement. If an agent fails to reach the goal within 50 steps, it returns to the starting position and begins a new trial. Each trial includes 90,000 steps, and the agents update their policies over a total of 100 trials.

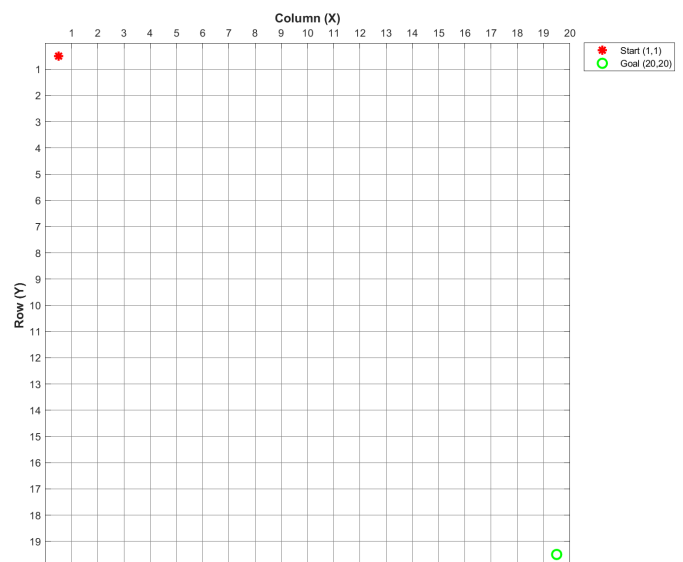


Figure 1. The 20x20 maze without obstacles

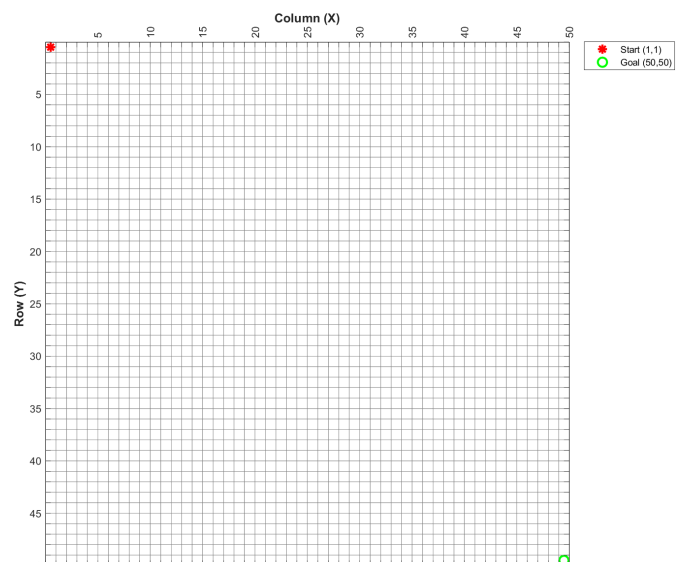


Figure 2. The 50x50 maze without obstacle

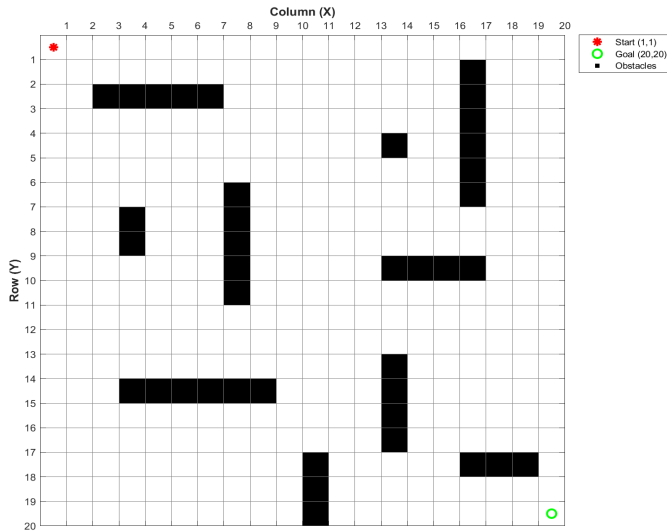


Figure 3. The 20x20 maze with obstacles

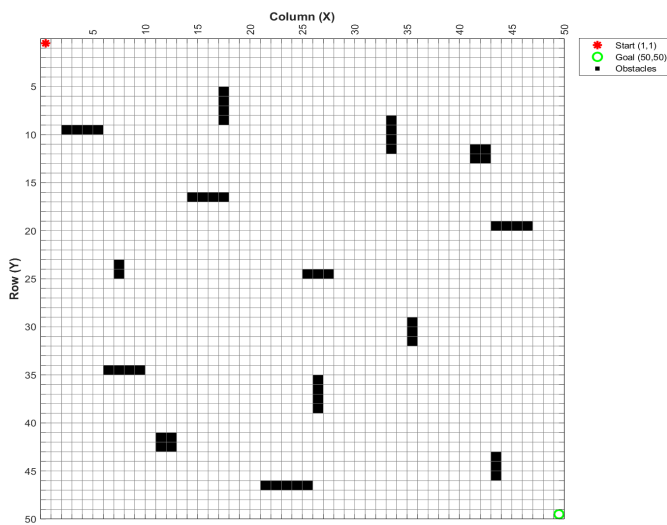


Figure 4. The 50x50 maze with obstacles

In the AC algorithm, the actor is responsible for action selection, while the critic estimates state values. That is, the actor updates its policy based on the temporal-difference (TD) error, while the critic adjusts the state-value function using the same error value.

In this study, the tabular AC method is utilized for the training of the agents, where state and action spaces are maintained in discrete matrices. To balance the exploration-exploitation trade-off, an ϵ -greedy ($\epsilon=0.1$) approach is adopted. The agent does not use a probabilistic distribution like Softmax during the decision-making phase; instead, it directly selects the action with the highest preference value during exploitation. Policy parameters are updated directly using the TD error (δ), without relying on function approximation or deep neural networks. The single-step AC algorithm developed by Sutton and Barto (2014) is provided below as a baseline:

Pseudocode of AC Algorithm

Input:

- State-action preference matrix: $P(s,a)$
- State-value matrix: $V(s)$

Parameters:

- Learning rates : $\alpha_a, \alpha_c > 0$
- Discount factor: $\gamma \in (0,1)$

Loop each experiment:

- Initialize S
 - Loop for each step: (if S is not terminal)
 - Select action A using ϵ -greedy based on $P(S, \cdot)$
 - Take action A , observe R and next state S'
 - $\delta = R + \gamma V(S') - V(S)$, (When S_{t+1} is terminal $V(S_{t+1}, w) = 0$)
 - $V(S) \leftarrow V(S) + \alpha_c \delta$
 - $P(S, A) \leftarrow P(S, A) + \alpha_a \delta$
 - $S \leftarrow S'$

In this study, the first agent is trained using the standard AC algorithm and records its best-performing path. The path coordinates generated by this agent are stored in a static memory matrix, designated as Path, which serves as the reference for the imitation process. Within each run, a path is considered successful and stored in memory only if the agent reaches the goal and the path length is shorter than the currently recorded path.

In the IEAC model, the second agent operates in two modes: a normal mode and an imitation (copy) mode. In normal mode, the agent learns through its own AC policy, whereas in imitation mode, it observes and occasionally imitates the successful actions of the first agent. The agent switches between modes when it collides with a wall, moves outside the maze, or exceeds 30 consecutive steps in the same mode. This mechanism prevents continuous imitation and ensures a more dynamic learning process.

The initial mode is selected probabilistically based on a predefined imitation probability, p ($p \in \{0.1, 0.3, 0.5\}$). Before each trial begins, the probability of imitation determines whether the agent starts in imitation or normal mode. During learning, agents compute the TD-error based on their actions to update both the state-value function and the policy. The detailed pseudocode for this procedure, incorporating the probabilistic initialization and update rules, is given in the following algorithm.

When the follower agent (agent 2) enters the copy mode, an alignment step is executed if its current position (S_t) is not on the recorded reference Path. The agent does not teleport to the beginning of the path or start copying blindly; instead, it calculates the Euclidean distance between its current position and all nodes on the Path to identify the closest node ($S_{closest}$). The next action is then determined by $A_t = \text{sign}(S_{closest} - S_t)$, ensuring the agent takes the shortest step toward the reference path. This step is not an independent heuristic but a geometric alignment necessary for meaningful imitation. Furthermore, to address how different distance metrics affect the learning curves, Euclidean and Manhattan distance metrics were compared (Figure 5 and 6). The analysis reveals that while the distance metric does not alter the ability to reach the optimal solution, using the Euclidean metric provides slightly faster initial learning across all imitation probabilities.

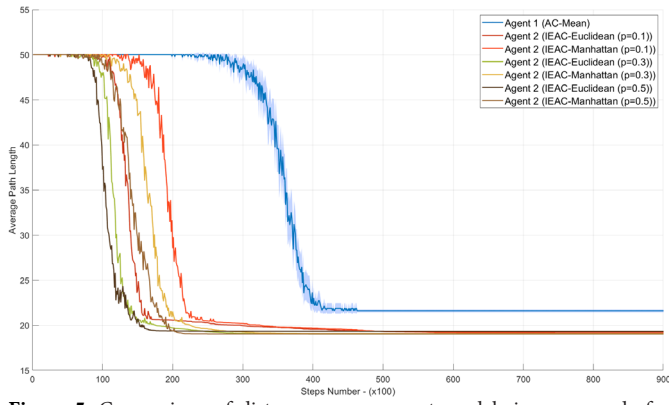


Figure 5. Comparison of distance measurement models in copy mode for 20x20 empty maze

AC: Actor-critic, IEAC: Imitation enhanced actor-critic

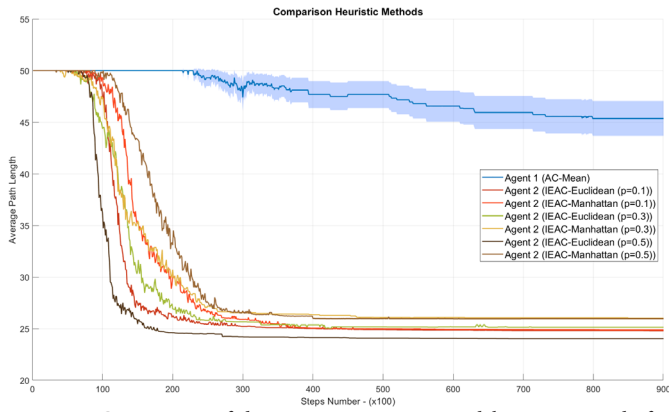


Figure 6. Comparison of distance measurement models in copy mode for 20x20 obstacle maze

AC: Actor-critic, IEAC: Imitation enhanced actor-critic

Pseudocode of Imitation Enhanced Actor-Critic (IEAC) Algorithm

Input:

- **State-action preference matrix:** $P(s, a)$ and State-value matrix: $V(s)$

Parameters:

- **Learning rate:** $\alpha_a, \alpha_c > 0$ (for actor and critic respectively)
- **Discount factor:** $\gamma \in (0, 1)$
- **Imitation probability:** $p \in \{0.1, 0.3, 0.5\}$

Memory:

Path: Optimal path matrix recorded by the Leader agent (stores state-action pairs).

For each experiment do:

Agent 1 training (leader):

- Learn for t time steps using AC algorithm (see above pseudocode.)
- If (target reached) & (current path is shortest) then
 - $Path \leftarrow \{(s_0, a_0), (s_1, a_1), \dots\}$

Agent 2 training (observer):

- Initialize: S_0 at $t = 0$ in start position)
- While S is not terminal (target) do
 - Select mode (copy or normal) based on probability p .

- If (mode=copy) then
 - **Check:** Is $S \in Path$?
 - If ($S \in Path$) then
 - $A \leftarrow Path(S)$ (select recorded action)
 - else
 - $A \leftarrow ClosestState(S)$ (select appropriate action to approach the path)
- else
 - Sample action $A_t \sim P(s, a)$ (standard action selection)
- Execute action A_t , observe reward R_{t+1} and new state S_{t+1} .
- If ($R_{t+1} = R_{collision}$) or (mode_step_count > max_step) then
 - Switch mode (normal $\Rightarrow$ copy or copy $\Rightarrow$ normal)
- $\delta_t = R_{t+1} + \gamma V(S_{t+1}) - V(S_t)$ (If S_{t+1} is terminal, $V(S_{t+1}) = 0$)
- $V(S_t) \leftarrow V(S_t) + \alpha_c \delta_t$
- $P(S_t, A_t) \leftarrow P(S_t, A_t) + \alpha_a \delta_t$
- $S_t \leftarrow S_{t+1}$ and $t \leftarrow t + 1$

The mode_step_count variable defined in the IEAC algorithm tracks the number of consecutive steps the follower agent spends in the selected mode. This variable establishes an upper limit (max_step) to prevent the agent from getting stuck in an initial suboptimal mode or blindly copying the entire path without integrating its own exploration. To determine the optimal upper limit, a sensitivity analysis was conducted evaluating thresholds of 15, 30, and 45 steps (Table). The analysis reveals that setting the limit to 15 steps results in rapid convergence (12.100 steps) but yields a suboptimal final path length (25.90) with higher variance. Conversely, a limit of 45 steps guarantees a 100% success rate with very low variance (1.36) but drastically slows down convergence (43,700 steps) and restricts the agent's exploration, resulting in a longer path (27.00). Therefore, max_step=30 was selected as the optimal threshold, providing the best balance: achieving the shortest final path length (25.44) and rapid convergence (17,800 steps) while maintaining a high success rate (98%).

Table. Sensitivity analysis for mode switching threshold

Max_step	Convergence step	Final path length	SD (stability)	Success rate (%)
15	12.100	25.90	4.33	97%
30	17.800	25.44	4.27	98%
45	43.700	27.00	1.36	100%

SD: Standard deviation, Max: Maximum

The overall working principle of the model is presented visually in Figure 7. The following section presents the comparative results of the AC, IEAC, behavioral cloning (BC) and Q-learning.

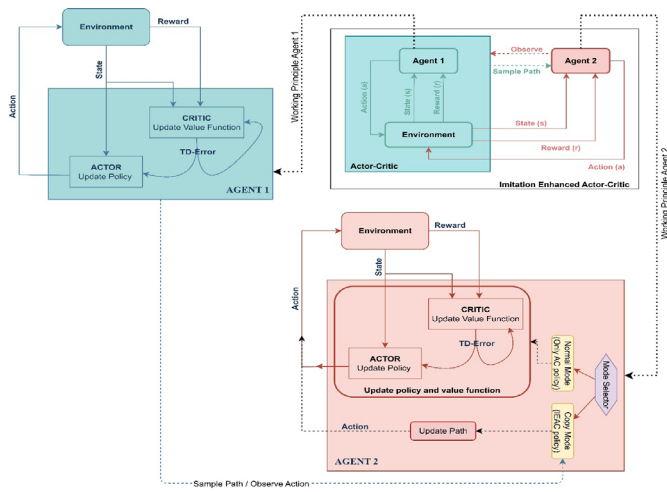


Figure 7. Diagram of working principle for IEAC method

IEAC: Imitation enhanced actor-critic

RESULTS

To ensure the statistical reliability of the results, each experiment was conducted over $n=100$ independent runs with different random initial conditions. Instead of an episode-based evaluation, a continuous step-based evaluation was utilized. The x-axis in the graphs represents every 100 steps taken by the agents. Every 100 steps, the shortest path from the start to the goal is calculated and recorded using the agent's current policy matrix. The shaded areas in the graphs denote the standard deviation (SD) and the 95% confidence interval (CI). In the empty maze (Figure 8), the standard AC algorithm started learning in approximately 30,000 steps and converged to the optimum path length around 40,000 steps. The agent using the IEAC method, however, started learning much earlier (at approximately 10,000 steps for $p=0.5$) and converged to the shortest path length in roughly 18,000 steps. As illustrated in Figure 8, the IEAC agent reaches the optimal path approximately 22,000 steps earlier than the standard AC agent. Furthermore, to evaluate the efficacy of the IEAC method from a broader perspective, comparative analyses were performed alongside tabular Q-learning and BC. As observed in Figure 8 and 9, although the Q-Learning method provides an improvement over standard AC, it fails to achieve the early convergence provided by the IEAC method. The BC method, which relies solely on imitation, completely fails in complex obstacle environments due to its inability to conduct independent exploration.

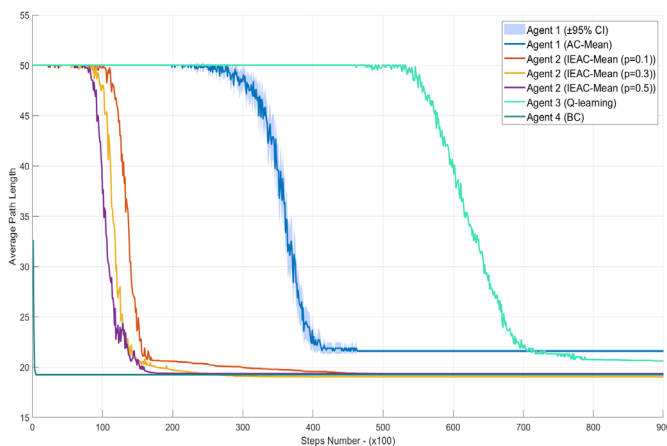


Figure 8. Comparison learning curves of AC, BC, Q-learning, and IEAC with different probabilities in the 20×20 empty maze

AC: Actor-critic, BC: Behavioral cloning, IEAC: Imitation enhanced actor-critic

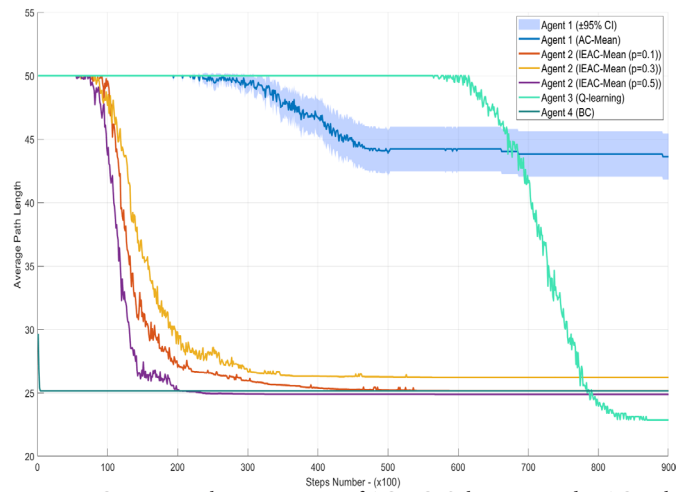


Figure 9. Comparison learning curves of AC, BC, Q-learning, and IEAC with different probabilities in the 20×20 obstacle maze

AC: Actor-critic, BC: Behavioral cloning, IEAC: Imitation enhanced actor-critic

For all probability values ($p \in \{0.1, 0.3, 0.5\}$), the IEAC method reached the optimum solution much faster than the AC method. Moreover, the minimal differences among imitation trials with different probabilities indicate that, in the obstacle-free environment, the imitation process provides a strong learning acceleration regardless of probability. In this environment, the learning curve of the AC agent exhibits a narrow 95% CI due to its relatively high learning rate, whereas the learning curves of the IEAC agents mostly remain outside this interval.

When examining the SD graph for the probability value $p=0.3$ in Figure 10, it is observed that the variance of the IEAC method narrows much earlier than that of the AC method for all probability values. This finding proves that social learning support not only increases speed but also reduces the instability (variance) of the agent in the exploration process, thus offering a more stable learning process. It should be noted that SD graphs are presented only for $p=0.3$ as a representative example, as similar variance patterns were observed across all probability values tested.

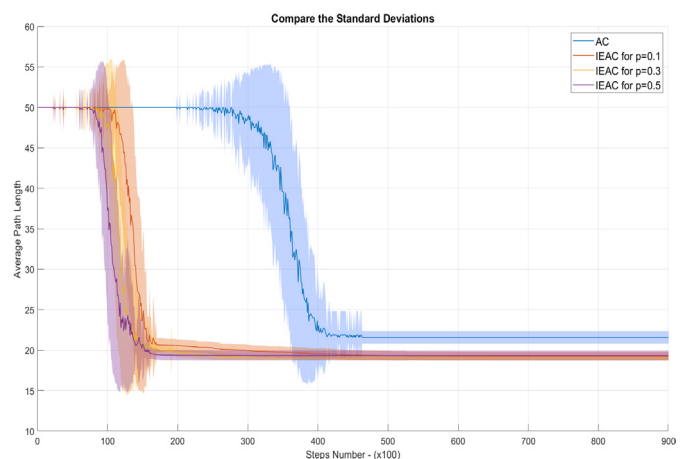


Figure 10. Standard deviation areas for all possibilities of IEAC in the 20×20 empty maze

IEAC: Imitation enhanced actor-critic

It is essential to present the mode imitation statistics, which quantify the extent to which the IEAC-trained agent imitates the leader over time. These results are critical as they demonstrate that the observer agent (agent 2) does not merely replicate the leader's actions exactly; rather, it utilizes

the imitation mechanism strategically to enhance its own learning process. The analysis illustrating the duration the agent remains in “copy mode” over time, for all probability values considered in this study, is provided in Figure 11. In this figure, the copy mode statistics for the learner agent were calculated by taking the average number of steps the agent remained in copy mode (i.e., imitated) over a rolling window of 100 steps.

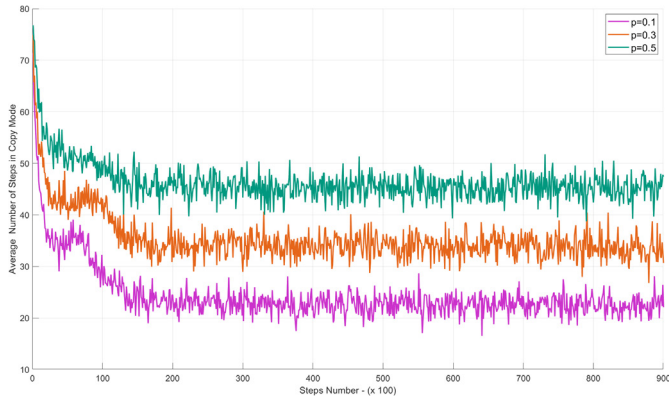


Figure 11. Copy mode statistics for all possibilities of IEAC in the 20x20 empty maze

IEAC: Imitation enhanced actor-critic

A close examination of the graph reveals a consistency between the imitation probability values and the duration spent in copy mode. Furthermore, it is observed that the imitation rate is initially high at the onset of training but exhibits a gradual downward trend as time progresses. This indicates that the learner agent, rather than merely replicating the leader exactly, utilizes the imitated actions to refine its own learning policy and reduces its reliance on imitation as its policy becomes optimized. However, it should be noted that the imitation rates never converge to zero. This persistence is inherent to the structure of the algorithm, which stochastically compels the agent to enter imitation mode at certain intervals based on the predefined probability values.

The results for the obstacle maze (Figure 9) show that the IEAC agent started learning earlier and converged much faster. The AC agent started learning after approximately 30,000 steps and got stuck in a local optimum around 40,000 steps with an average path length of approximately 44 steps. In contrast, the IEAC agent started learning significantly earlier (after approximately 10,000 steps for $p=0.3$) and reached the optimum solution around 25,000 steps with an average path length of 25 steps. Excessive imitation ($p=0.5$) slightly delays this process as it forces the agent to follow the leader's initially suboptimal path more frequently. Therefore, moderate imitation probabilities ($p=0.3$) provide a more effective balance.

Scalability analysis (50x50 environments): In addition to the 20x20 environments, the proposed models were tested in larger 50x50 maze configurations to evaluate scalability (Figure 12-17). The results in both empty and obstacle-based 50x50 mazes demonstrated consistent patterns: the IEAC method maintained its superiority in convergence speed and stability over the standard AC and Q-learning baselines, proving that the probabilistic imitation mechanism remains highly effective as the state space expands.

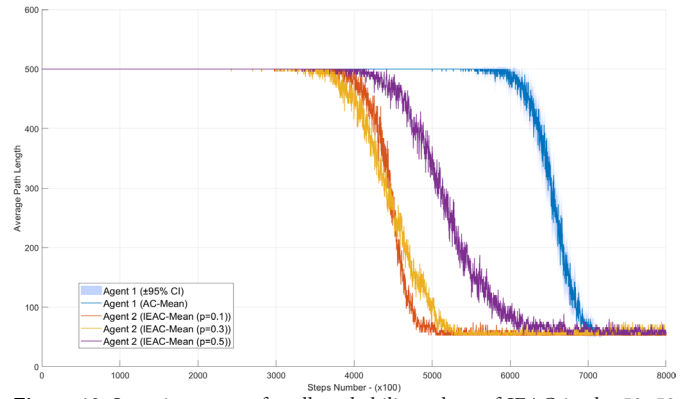


Figure 12. Learning curves for all probability values of IEAC in the 50x50 empty maze

IEAC: Imitation enhanced actor-critic

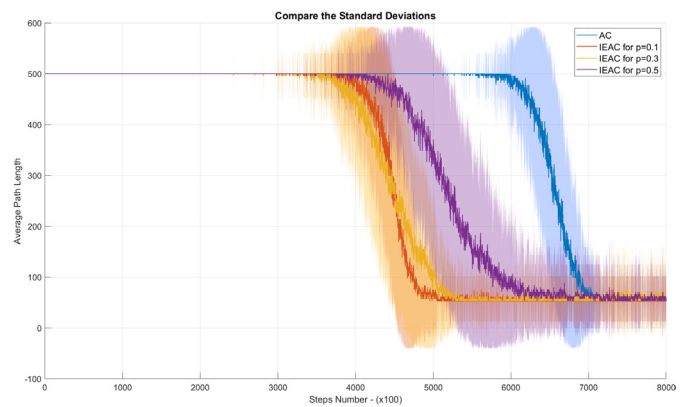


Figure 13. Standard deviation areas for all probability values of IEAC in 50x50 empty maze

IEAC: Imitation enhanced actor-critic

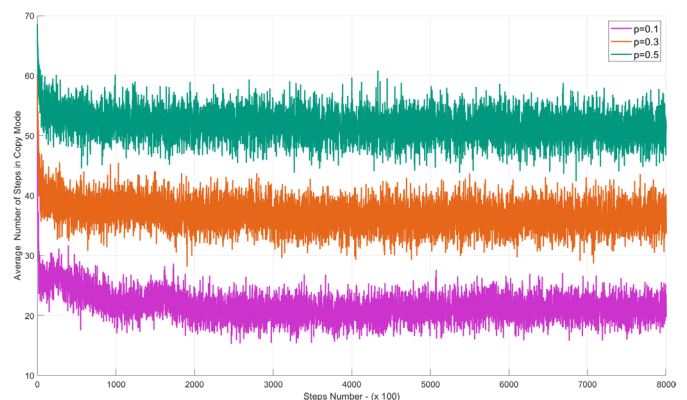


Figure 14. Copy mode statistics for all possibilities of IEAC in the 50x50 empty maze

IEAC: Imitation enhanced actor-critic

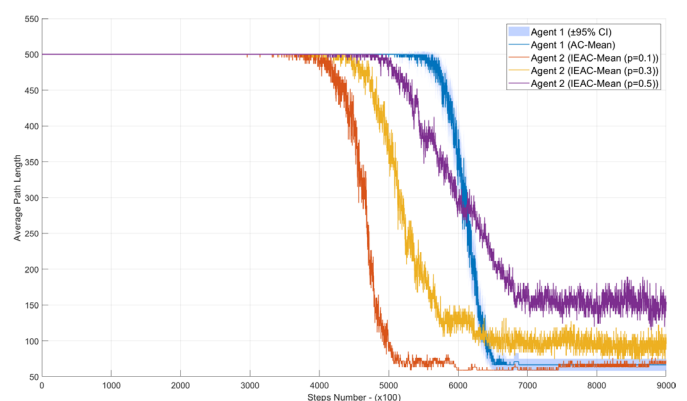


Figure 15. Learning curves for all probability values of IEAC in the 50x50 obstacle maze

IEAC: Imitation enhanced actor-critic

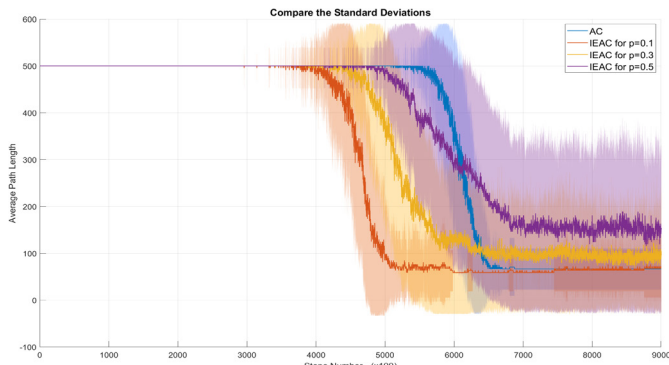


Figure 16. Standard deviation areas for all probability values of IEAC in 50x50 obstacle maze

IEAC: Imitation enhanced actor-critic

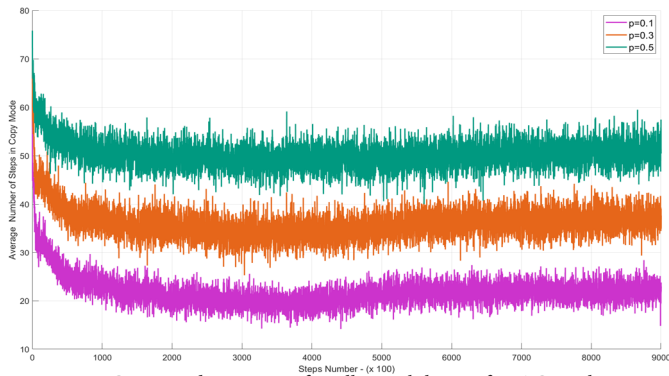


Figure 17. Copy mode statistics for all possibilities of IEAC in the 50x50 obstacle maze

IEAC: Imitation enhanced actor-critic

For each reporting point, the mean and SD were calculated over $n = 100$ independent runs. The 95% CI was calculated as “mean $\pm$ 1.96 $\cdot(\sigma/\sqrt{n})$ ”, where σ represents the sample SD across trials. Additionally, to compare the stability of the learning methods, the SD value was calculated in this study as $\sigma = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n-1}}$.

Furthermore, we also analysed the stability of learning in a fixed obstacle maze environment by considering SDs for a probability value of $p=0.3$. As shown in Figure 18, the IEAC agent initially exhibited a wide range of SDs, and this range gradually narrowed over time. This indicated that imitation-based learning initially increased variance but became more stable as learning progressed. In contrast, the AC agent reflected a less stable learning process, maintaining higher variance over a longer period.

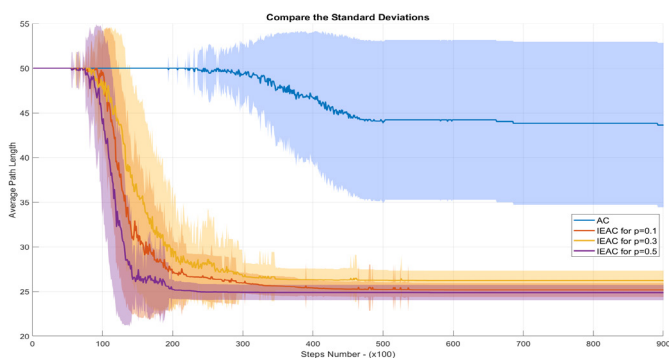


Figure 18. Standard deviation areas for all possibilities of IEAC in the 20x20 obstacle maze

IEAC: Imitation enhanced actor-critic

In summary, in an obstacle-based environment, the wide SD range of the AC agent indicates that the learning curve

oscillates, meaning that training consistency decreases. However, it is clear that the SD range of the IEAC agent is narrower than that of the AC method throughout the entire runtime. This suggests that in a fixed obstacle maze environment, the IEAC method is a more stable method than the AC algorithm.

Finally, following the key findings presented throughout the study, Figure 19 shows the statistics for all probability values, giving the average total number of imitation steps for agent 2 every 100 steps. Here, although the number of imitation steps was initially observed to be around 70 for all probability values, the average number of imitation steps decreased rapidly within the first 10,000 steps of the study (which corresponds to approximately the middle of the second trial), and then the number of imitation steps remained almost constant for each probability value. When Figure 18 and 19 are considered together, it is clear that the results are consistent.

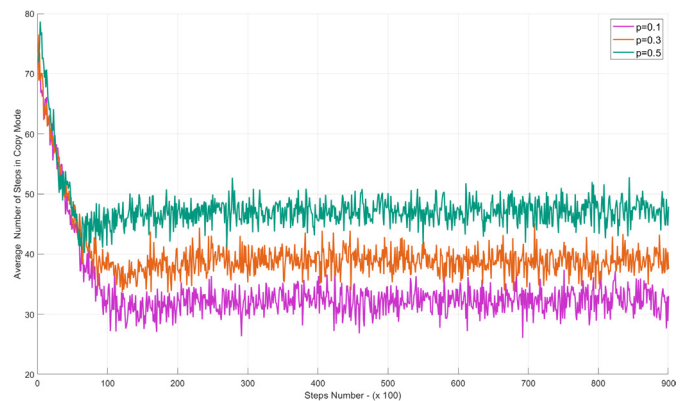


Figure 19. Copy mode statistics for all possibilities of IEAC in the 20x20 obstacle maze

IEAC: Imitation enhanced actor-critic

DISCUSSION

In this study, the IEAC method is based on observing the positive actions of the first agent that led to the goal, without direct information exchange between agents, and then having the second agent imitate these actions under specific conditions. Thus, the learning-by-imitation method, combined with the AC method, is compared with the standard AC algorithm in terms of learning speed, stability, and success in reaching the optimum result.

In an obstacle-free maze environment, although the probabilistic imitation method and the IEAC method are more successful in early learning, it is observed that both AC and IEAC reach the optimum result in the long term. In a fixed obstacle environment, it is clear that the IEAC method provides much better performance than the simple AC method. While the standard AC agent gets stuck because it cannot risk the immediate negative rewards necessary to circumvent the obstacle, the IEAC agent overcomes this problem because it performs the imitation process probabilistically, and comes very close to the optimum solution in reaching the goal. Similar to human behaviour models in the literature, in situations with high exploration costs, the social learning (imitation) mechanism provides a safe learning process by reducing the cost of trial and error. These findings show that in dynamic environments with high uncertainty, choosing an action (through imitation), even if suboptimal, instead of remaining inactive or experiencing indecision, accelerates the learning process.

The literature review revealed that learning through imitation is applied in conjunction with some RL methods. Therefore, it was predicted that positive results could be obtained by combining the AC RL method with the imitation method. To provide insight into whether the proposed IEAC method is useful in learning, the fundamental problem of “finding the shortest path in a maze” was considered. In this regard, experiments were conducted in various 20×20 maze environments, and the learning speed and stability in reaching the optimum result of the method were compared only in a computer environment.

The obstacle maze scenario also highlights a structural limitation of the IEAC model: if the leader agent gets trapped in a local optimum, the follower agent, especially at high imitation probabilities, tends to reproduce this suboptimal strategy. This presents a significant development opportunity for future studies. Instead of a fixed imitation probability (p), developing adaptive mechanisms where p is dynamically adjusted based on TD-error fluctuations, success rates, or stagnation detection would enable the follower to break free from the leader's mistakes and discover its own optimal policy.

Limitations

The experimental results of this study were simulated in a 2D environment with only two agents. Future research could validate the real-world applicability of the IEAC approach by applying it to physical robots and scaling it to multi-agent systems. Furthermore, the study could be extended to more complex and continuous-field problems: instead of using fixed probability values in the imitation method, more systematic methods that can make and implement their own decisions regarding imitation could be developed, or a method created by combining the AC method using deep neural networks with social learning could shed light on the solution of many problems.

CONCLUSION

As a result, the IEAC method achieves faster and more stable learning than the classical AC approach. Its earlier convergence to the optimal solution, particularly in the obstacle environment, confirms the effectiveness of imitation-based RL in complex settings. The proposed framework is particularly suitable for environments where exploration is costly or risky, making it a promising candidate for real-world autonomous navigation problems.

ETHICAL DECLARATIONS

Ethics Committee Approval

Ethical approval was not required for this study as it does not involve human participants, animal subjects, or identifiable personal data.

Peer Review Process

This manuscript was subject to external peer review.

Conflict of Interest

The authors declare no conflicts of interest related to this study.

Financial Disclosure

The authors received no financial support for the conduct or publication of this research.

Author Contributions

Concept: NK, MDE; Design: NK, MDE; Control: NK, MDE; Resources: NK, MDE; Materials: NK, MDE; Data Collection and/or Processing: NK, MDE; Analysis and/or Interpretation: NK, MDE; Literature Review: NK, MDE; Writing the Article: NK, MDE; Critical Review: NK, MDE.

REFERENCES

- Argall, B. D., Chernova, S., Veloso, M., & Browning, B. (2009). A survey of robot learning from demonstration. *Robotics and Autonomous Systems*, 57(5), 469-483. <https://doi.org/10.1016/j.robot.2008.10.024>
- Bentivegna, D., Cheng, G., & Atkeson, C. (2005). Learning from observation and from practice using behavioral primitives. *Springer Tracts in Advanced Robotics*, 15, 551-560. https://doi.org/10.1007/11008941_59
- Block, A., Foster, D. J., Krishnamurthy, A., Simchowitz, M., & Zhang, C. (2024). Butterfly effects of SGD noise: Error amplification in behavior cloning and autoregression. *Proceedings of the International Conference on Learning Representations (ICLR)*.
- Brederche, N., & Fontbonne, N. (2022). Social learning in swarm robotics. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 377, 1843. <https://doi.org/10.1098/rstb.2020.0309>
- Christianos, F., Schäfer, L., & Albrecht, S. V. (2021). Shared experience actor-critic for multi-agent reinforcement learning. *Advances in Neural Information Processing Systems*, 33, 10707-10717. <http://arxiv.org/abs/2006.07169>
- Codevilla, F., Müller, M., López, A., Koltun, V. & Dosovitskiy, A. (2018). End-to-end driving via conditional imitation learning. In 2018 IEEE International Conference on Robotics and Automation (ICRA). IEEE Press, 1-9. <https://doi.org/10.1109/ICRA.2018.8460487>
- Erbas, M. D., Winfield, A. F., & Bull, L. (2014). Embodied imitation-enhanced reinforcement learning in multi-agent systems. *Adaptive Behavior*, 22(1), 31-50. <https://doi.org/10.1177/1059712313500503>
- Guenther, F., & Billard, A. G. (2007). Using reinforcement learning to adapt an imitation task. *IEEE International Conference on Intelligent Robots and Systems*, 1022-1027. <https://doi.org/10.1109/IROS.2007.4399449>
- Guo, X., Chang, S., Yu, M., Tesauro, G., & Campbell, M. (2019). Hybrid Reinforcement Learning with Expert State Sequences. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01), 3739-3746. <https://doi.org/10.1609/aaai.v33i01.33013739>
- Henrich, J. (2016). *The Secret of Our Success: How Culture is Driving Human Evolution, Domesticating Our Species, and Making Us Smarter*. Princeton University Press. <https://doi.org/10.2307/j.ctvc77f0d>
- Mathewson, K. W., & Pilarski, P. M. (2017). Actor-critic reinforcement learning with simultaneous human control and feedback. *arXiv*. <http://arxiv.org/abs/1703.01274>
- Ndousse, K., Eck, D., Levine, S., & Jaques, N. (2021). Emergent social learning via multi-agent reinforcement learning. *arXiv*. <https://doi.org/10.48550/arXiv.2010.00581>
- Nguyen, S. M. (2024). The intrinsic motivation of reinforcement and imitation learning for sequential tasks. *arXiv*. <https://doi.org/10.48550/arXiv.2412.20573>
- Rao, R. P. N., Shon, A. P., & Meltzoff, A. N. (2007). A Bayesian model of imitation in infants and robots. In C. L. Nehaniv & K. Dautenhahn (Eds), *Imitation and Social Learning in Robots, Humans and Animals* (1st ed., pp. 217-248). Cambridge University Press. <https://doi.org/10.1017/CBO9780511489808.016>
- Sallab, A. El, Abdou, M., Perot, E., & Yogamani, S. (2017). Deep Reinforcement Learning framework for Autonomous Driving. *Proceedings of the IS&T International Symposium on Electronic Imaging Science and Technology*, 70-76. <https://doi.org/10.2352/issn.2470-1173.2017.19.avm-023>
- Sharma, A., Kaur, H., & Prashar, D. (2023). Maze Solver: A Federated Reinforcement Learning Approach. 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), pp.1-5. <https://doi.org/10.1109/ICCCNT56998.2023.10307671>
- Sutton, R. S., & Barto, A. G. (2014). *Reinforcement learning: an introduction* (2nd ed.). MIT Press.
- Uwano, F., Tatebe, N., Tajima, Y., Nakata, M., Kovacs, T., & Takadama, K. (2018). Multi-agent cooperation based on reinforcement learning with internal reward in maze problem. *Journal of Control, Measurement, and System Integration*, 11(4), 321-330. <https://doi.org/10.9746/jcmsi.11.321>
- Yang, X., Gao, J., Wang, P., Long, W., & Fu, C. (2024). Social Learning with Actor-Critic for dynamic grasping of underwater robots via digital twins. *Ocean Engineering*, 306. <https://doi.org/10.1016/j.oceaneng.2024.118070>