© July 2026 | IJIRT | Volume 13 Issue 2 | ISSN: 2349-6002

Comparative Study of Curriculum Learning and Behavior Cloning in Reinforcement Learning-Based Multi-Agent Systems

Sachin Shankarrao Damre[1], Dr. Sushil Kulkarni[2]

[1]*M. tech 3rd Semester Student, Department of Computer Science and Information Technology, MBES College of Engineering, Ambajogai, India*

[2]*Head and Guide, Department of Computer Science and Information Technology, MBES College of Engineering, Ambajogai, India*

Abstract—**Reinforcement Learning (RL) has become one of the most influential paradigms in Artificial Intelligence for developing autonomous agents capable of learning optimal behavior through interaction with dynamic environments. Recent advancements in Deep Reinforcement Learning (DRL) have enabled intelligent systems to solve complex decision-making tasks in gaming, robotics, autonomous navigation, and industrial automation. This review paper presents a comprehensive survey of RL, Deep RL, Curriculum Learning, Behavior Cloning, Generative Adversarial Imitation Learning (GAIL), and Unity ML-Agents. Particular emphasis is placed on intelligent multi-agent game environments where cooperation, competition, and coordination significantly increase learning complexity. The paper provides a comparative analysis of Curriculum Learning and Behavior Cloning, identifies current research trends, highlights limitations in existing literature, and discusses future directions. The study reveals that combining curriculum-based training with imitation learning techniques offers promising opportunities for developing scalable and robust intelligent agents in complex gaming environments.**

Index Terms—**Reinforcement Learning, Deep Reinforcement Learning, Curriculum Learning, Behavior Cloning, GAIL, PPO, Unity ML-Agents, Multi-Agent Systems, Game AI, Intelligent Agents.**

I. INTRODUCTION

Artificial Intelligence has revolutionized modern computing systems by enabling machines to perceive, reason, learn, and make decisions. Among the various learning paradigms, Reinforcement Learning has gained substantial attention because it allows agents to learn through interaction rather than relying solely on labeled datasets. In RL, an agent observes the environment, performs actions, receives rewards, and gradually improves its policy to maximize long-term returns.

Game environments have emerged as ideal benchmarks for evaluating intelligent behavior. They offer controlled environments, measurable objectives, and varying levels of complexity. Landmark achievements such as Atari game mastery and Go-playing systems demonstrated the potential of RL and DRL techniques. Modern game environments require agents to handle partial observability, delayed rewards, strategic planning, and collaboration with other agents.

Multi-agent environments further increase complexity by introducing interactions among multiple intelligent entities. Such environments require coordination, competition, communication, and adaptive decision-making. Examples include team-based games, cooperative navigation tasks, and Freeze Tag simulations. Researchers have increasingly adopted Curriculum Learning and Imitation Learning techniques to address the challenges associated with long training times and sparse rewards.

This review paper investigates major RL techniques and provides a publication-oriented comparison between Curriculum Learning and Behavior Cloning within intelligent multi-agent game environments.

II. FUNDAMENTALS OF REINFORCEMENT LEARNING

Reinforcement Learning is a computational framework where an agent learns optimal behavior through repeated interaction with an environment. Unlike supervised learning, RL does not require labeled examples. Instead, learning occurs through rewards and penalties. The RL framework consists of Agent, Environment, State, Action, Reward, Policy, Value Function, and Model. At every time step, the agent observes a state, chooses an action according to its policy, and receives a reward from the environment. The objective is to maximize cumulative discounted reward.

Most RL problems are represented using Markov Decision Processes (MDPs). An MDP is defined by states, actions, transition probabilities, and rewards. The Markov property assumes that future states depend only on the current state and action.

Major RL categories include:
1. Model-Based Reinforcement Learning.
2. Model-Free Reinforcement Learning.
3. Value-Based Methods.
4. Policy-Based Methods.
5. Actor-Critic Methods.

The exploration-exploitation trade-off remains one of the most critical challenges in RL. Exploration enables discovery of new strategies, while exploitation leverages existing knowledge. Balancing both is essential for effective learning.

Mathematically, the return is defined as the cumulative discounted reward, where future rewards are discounted using a factor gamma. The optimal policy maximizes expected returns over time.

III. DEEP REINFORCEMENT LEARNING TECHNIQUES

Deep Reinforcement Learning combines deep neural networks with RL algorithms to handle high-dimensional state spaces.

A. Deep Q-Network (DQN): DQN uses neural networks to approximate Q-values. Experience replay and target networks improve training stability. DQN achieved human-level performance across numerous Atari games.

B. Double Deep Q-Network (DDQN): DDQN addresses the overestimation bias present in DQN by separating action selection and action evaluation. This leads to improved stability and more accurate value estimation.

C. Actor-Critic Methods: Actor-Critic architectures employ two models. The Actor learns the policy while the Critic estimates state values. This approach improves convergence and learning efficiency.

D. Proximal Policy Optimization (PPO): PPO is among the most widely used DRL algorithms because it offers stable training and strong performance. PPO constrains policy updates using a clipping mechanism, preventing excessively large updates.

E. Soft Actor-Critic (SAC): SAC maximizes both reward and entropy, encouraging exploration. It performs exceptionally well in continuous control environments.

Advantages of DRL include automatic feature extraction, scalability, and adaptability. Limitations include computational expense, sample inefficiency, reward engineering challenges, and hyperparameter sensitivity.

Table 1: Comparison of RL Algorithms

Algorithm	Type	Advantages	Limitations
DQN	Value-Based	Simple, Effective	Overestimation Bias
DDQN	Value-Based	Stable Learning	Higher Complexity
PPO	Policy-Based	Robust Performance	More Computation
Actor-Critic	Hybrid	Fast Convergence	Sensitive Tuning

IV. CURRICULUM LEARNING

Curriculum Learning is inspired by the human learning process in which simpler concepts are mastered before advancing to more complex tasks. In RL, curriculum learning structures the training process by gradually increasing task difficulty. The curriculum may be designed manually or automatically. Manual curricula rely on domain expertise, while automated approaches dynamically adjust difficulty based on agent performance.

Benefits include:
- Faster convergence.
- Better sample efficiency.
- Improved generalization.
- Enhanced learning stability.
- Reduced exploration burden.

Curriculum Learning has been successfully applied in robotics, autonomous driving, navigation tasks, and game-based environments. In Freeze Tag scenarios, agents can initially learn movement, then pursuit strategies, followed by cooperative behaviors.
Despite its advantages, curriculum design remains challenging. Selecting appropriate difficulty progression significantly influences performance outcomes.

V. BEHAVIOR CLONING AND GAIL

Behavior Cloning is an imitation learning technique where agents learn directly from expert demonstrations. The problem is formulated as supervised learning, where states are inputs and expert actions serve as labels. Behavior Cloning reduces exploration requirements and accelerates early-stage learning. It is particularly effective when expert data is available. However, the method often suffers from compounding errors because agents may encounter states not represented in training demonstrations. Generative Adversarial Imitation Learning (GAIL) extends Behavior Cloning through adversarial training. A discriminator attempts to distinguish expert behavior from agent-generated behavior, while the policy learns to imitate experts.
Advantages:
- Faster training.
- Lower exploration costs.
- Practical deployment.

Limitations:
- Dependence on demonstration quality.
- Limited adaptability.
- Reduced robustness in unseen scenarios.

Recent studies combine Behavior Cloning with RL fine-tuning to achieve improved performance and generalization.

VI. UNITY ML-AGENTS FRAMEWORK

Unity ML-Agents is a powerful open-source toolkit developed by Unity Technologies. It enables the creation of intelligent agents within highly customizable game environments.

The framework supports:
- Reinforcement Learning.
- Imitation Learning.
- Curriculum Learning.
- Self-Play.
- Multi-Agent Training.

Unity ML-Agents integrates with Python-based training pipelines and supports PPO, SAC, and custom algorithms. Researchers widely utilize the framework because of its rich visualization capabilities and ease of environment design. In multi-agent game scenarios, ML-Agents allows developers to define rewards, observations, and agent interactions while monitoring training performance. This flexibility makes it a preferred platform for academic research and industrial prototyping.

VII. LITERATURE REVIEW

Mnih et al. introduced Deep Q-Networks and demonstrated that neural networks could learn control policies directly from raw pixels. Their work laid the foundation for modern DRL research. Schulman et al. proposed PPO, which quickly became one of the most widely adopted algorithms due to its simplicity and effectiveness. Ho and Ermon introduced GAIL, establishing a new direction for imitation learning using adversarial optimization. Bengio et al. proposed Curriculum Learning and demonstrated that structured task progression improves learning efficiency. Silver et al. demonstrated the effectiveness of deep RL in strategic decision-making through AlphaGo, combining neural networks and search techniques.
Recent research has increasingly focused on multi-agent systems, transfer learning, self-play, and explainable AI. Several studies indicate that curriculum-based approaches improve convergence in sparse reward environments, while imitation learning accelerates initial policy acquisition.
Researchers continue investigating hybrid methodologies that integrate RL, Curriculum

Learning, and Behavior Cloning to achieve improved scalability and robustness.

Table 2: Literature Review Summary

Author	Contribution	Year
Mnih et al.	Deep Q-Network	2013
Mnih et al.	Human-Level Atari Control	2015
Schulman et al.	PPO	2017
Ho & Ermon	GAIL	2016
Bengio et al.	Curriculum Learning	2009
Silver et al.	AlphaGo	2016

VIII. COMPARATIVE ANALYSIS OF CURRICULUM LEARNING AND BEHAVIOR CLONING

Curriculum Learning and Behavior Cloning aim to improve training efficiency but follow fundamentally different approaches. Curriculum Learning gradually increases task difficulty, enabling agents to develop skills progressively. This approach promotes exploration, adaptation, and long-term robustness.

Behavior Cloning relies on expert demonstrations and provides rapid policy initialization. Training times are reduced significantly because agents learn from examples rather than extensive exploration.

In Freeze Tag environments, Curriculum Learning may outperform Behavior Cloning when environments are highly dynamic. Conversely, Behavior Cloning may achieve faster initial performance where high-quality demonstrations exist.

Hybrid approaches combine the strengths of both methods. Agents can first learn from expert demonstrations and subsequently refine policies through reinforcement learning under a curriculum framework. Such approaches often yield superior convergence and generalization.

Table 3: Comparative Analysis

Parameter	Curriculum Learning	Behavior Cloning
Learning Method	Progressive Tasks	Expert Demonstrations
Exploration	High	Low
Training Speed	Moderate	Fast
Generalization	Strong	Moderate
Dependence on Experts	Low	High
Adaptability	High	Limited
Scalability	High	Moderate

IX. RESEARCH GAP ANALYSIS

Although extensive research exists in reinforcement learning and imitation learning, several challenges remain unresolved. First, limited comparative studies evaluate Curriculum Learning and Behavior Cloning within identical multi-agent environments. Second, few investigations assess the impact of these methods on cooperative behaviors, communication strategies, and long-term adaptability. Third, current literature lacks standardized benchmarks for comparing training efficiency and generalization across different game environments. Fourth, most studies focus on algorithmic performance while providing limited insight into explainability and transparency.

Finally, hybrid approaches integrating curriculum learning, imitation learning, and reinforcement learning remain underexplored. Addressing these gaps can contribute significantly to next-generation intelligent agent development.

X. PROPOSED RESEARCH DIRECTION

The proposed research focuses on evaluating Curriculum Learning and Behavior Cloning in a Freeze Tag multi-agent environment developed using Unity ML-Agents.

Objectives include:

1. Measuring training convergence.
2. Comparing reward accumulation.
3. Evaluating cooperation efficiency.
4. Assessing policy robustness.
5. Measuring computational cost.

The study aims to establish empirical evidence regarding the effectiveness of both approaches and identify optimal strategies for intelligent multi-agent learning.

XI. FUTURE SCOPE

Future research directions include self-play learning, transfer learning, federated reinforcement learning, meta-learning, explainable AI, and large-scale multi-agent systems. The emergence of foundation models and generative AI creates opportunities for integrating language understanding with reinforcement learning. Human-in-the-loop systems may further enhance agent adaptability and safety.

Advanced curriculum generation techniques, automated reward shaping, and imitation learning from multimodal demonstrations represent promising research areas. These developments can support applications in robotics, healthcare, autonomous vehicles, and intelligent virtual environments.

XII. CONCLUSION

Reinforcement Learning has transformed the development of intelligent autonomous systems and continues to drive innovation across multiple domains. Deep Reinforcement Learning techniques such as DQN, DDQN, PPO, and Actor-Critic architectures have demonstrated remarkable success in complex game environments.

Curriculum Learning and Behavior Cloning offer complementary advantages for accelerating training and improving agent performance. Curriculum Learning supports progressive skill acquisition and strong generalization, whereas Behavior Cloning provides rapid learning from expert knowledge. Hybrid approaches integrating both techniques appear particularly promising for future multi-agent systems. This review paper highlights major developments, identifies research gaps, and provides a foundation for future investigations into intelligent game environments using Unity ML-Agents and advanced reinforcement learning methodologies.

REFERENCES

[1] Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., and Riedmiller, M., “Playing Atari with deep reinforcement learning,” arXiv preprint, arXiv:1312.5602, 2013.

[2] Almeida, P., Carvalho, V., and Simões, A., “Reinforcement learning as an approach to train multiplayer first person shooter game agents.”

[3] Dewan, T., Aditya, A., Trivedi, M., and Chen, A., “The use of reinforcement learning in gaming: The Breakout game case study.”

[4] McPartland, M., and Gallagher, M., “Reinforcement learning in first person shooter games,” IEEE Transactions on Computational Intelligence and AI in Games, vol. 3, no. 1, pp. 43–56, 2011.

[5] Unity Technologies, “ML-Agents,” GitHub repository. Available: https://github.com/Unity-Technologies/ml-agents. Accessed: Jun. 20, 2023.

[6] Silver, D., Huang, A., Maddison, C. J., Guez, A., van den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., et al., “Mastering the game of Go with deep neural networks and tree search,” Nature, vol. 529, no. 7587, pp. 484–489, 2016.

[7] Sutton, R. S., and Barto, A. G., Reinforcement Learning: An Introduction, 2nd ed. Cambridge, MA, USA: MIT Press, 2018.

[8] Gagniuc, P. A., Markov Chains: From Theory to Implementation and Experimentation. Hoboken, NJ, USA: John Wiley & Sons, 2017.

[9] Arulkumaran, K., Deisenroth, M. P., Brundage, M., and Bharath, A. A., “Deep reinforcement learning: A brief survey,” IEEE Signal Processing Magazine, vol. 34, no. 6, pp. 26–38, 2017.

[10] Henderson, P., Islam, R., Bachman, P., Pineau, J., Precup, D., and Meger, D., “Deep reinforcement learning that matters,” in Proc. AAAI Conf. Artificial Intelligence, vol. 32, no. 1, 2018.

[11] Szepesvári, C., Algorithms for Reinforcement Learning, Synthesis Lectures on Artificial Intelligence and Machine Learning, vol. 4, no. 1, pp. 1–103, 2010.

[12] Soemers, D. J., Sironi, C. F., Schuster, T., and Winands, M. H., “Enhancements for real-time Monte-Carlo tree search in general video game playing,” in Proc. IEEE Conf. Computational Intelligence and Games (CIG), 2016, pp. 1–8.

[13] Wei, Z., Wang, D., Zhang, M., Tan, A. H., Miao, C., and Zhou, Y., “Autonomous agents in snake game via deep reinforcement learning,” in Proc. IEEE Int. Conf. Agents (ICA), 2018, pp. 20–25.

[14] Baird, L., “Residual algorithms: Reinforcement learning with function approximation,” in Proc. 12th International Conference on Machine Learning (ICML), San Francisco, CA, USA: Morgan Kaufmann, 1995, pp. 30–37.

[15] Schulman, J., Wolski, F., Dhariwal, P., Radford, A., and Klimov, O., “Proximal policy optimization algorithms,” arXiv preprint, arXiv:1707.06347, 2017.