International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Volume: 04 Issue: 05 | May – 2025 DOI: 10.55041/ISJEM03388
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

# FLAPPY BIRD AI USING REINFORCEMENT LEARNING

**Mrs.G.Monika[1], E.Manusha[2], G.Sathwika[3], T.Indhu[4]**

Assoc. Professor of CSE(AI&ML) of ACE Engineering College[1] Students of Department CSE(AI&ML) of ACE Engineering College[2,3,4]

---------------------------------------------------------------------***---------------------------------------------------------------------

**Abstract -** Reinforcement learning plays a crucial role in solving problems where multiple solutions exist. Flappy Bird AI applies deep reinforcement learning to train an agent that learns to play Flappy Bird without human intervention. Despite having no prior knowledge of the bird or pipes, the AI analyzes game states and scores to develop an optimal strategy. A Convolutional Neural Network (CNN) processes visual input, while Q-learning with Deep Q-Networks (DQN) helps the AI make smart decisions. By balancing exploration and exploitation, superhuman performance will be attained in navigating obstacles. Enhancements like experience replay and target networks will improve learning efficiency. Flappy Bird AI will demonstrate the potential of reinforcement learning in autonomous gaming and intelligent decision-making systems.

**Key Words:** Flappy Bird AI, Reinforcement Learning, Deep-Q-Network, convolutional Neural Network, Q-Learning, Autonomous Gaming AI,Game AI.

## 1. INTRODUCTION

This study introduces an AI agent of Flappy Bird that has been built with reinforcement learning (RL) and a Deep Q-Network (DQN). The agent independently plays the game environment, acquiring the best actions through a reward mechanism. In contrast to earlier approaches like Genetic Algorithms and fixed Deep Neural Networks, our RL model provides real-time adaptability and constant improvement. To support increased engagement, we incorporated procedural level generation to allow dynamic gameplay that adapts with the skill of the player. This marriage of RL and adaptive design illustrates a promising way to develop more intelligent and responsive gaming experiences.

## 2. BACKGROUND OF THE PROJECT

Flappy Bird is a 2D side-scroller with high difficulty and straightforward tap controls, and the player must have accurate timing to not collide with pipes. Its clean look and repetitive mechanics make Flappy Bird well-suited for AI research, particularly real-time decision-making. Reinforcement Learning (RL) algorithms such as Deep Q-Networks (DQN), Genetic Algorithms (GA), and Neuroevolution have successfully trained agents based on game feedback. Our project draws on these and combines RL with dynamic level generation and adaptive difficulty. Instead of merely maximizing AI performance, the system dynamically adjusts challenges according to player skill. This provides a more personalized, immersive experience. The adaptive gameplay adapts in real time, maintaining difficulty balance. By integrating RL and procedural generation, the game becomes more intelligent and responsive.

## 3.LITERATURE REVIEW

### Daniel Shiffman [1]:

In The Nature of Code, Shiffman explores the use of evolutionary algorithms and natural selection principles in programming intelligent systems. His Neuroevolution-based implementation in the Flappy Bird game, using the p5.js library, demonstrates how genetic algorithms can evolve neural networks over generations. Although effective in simulating learning behaviour, the approach suffers from slower convergence and lacks adaptability during gameplay.

### Kevin Chen [2]:

This project investigates Deep Reinforcement Learning in Flappy Bird using Q-learning as part of Stanford's CS229 course. The research demonstrates how an agent can learn to navigate obstacles through trial-and-error interactions with the environment. Chen's use of experience replay and Q-value updates underlines the potential of Deep Q-Networks (DQN) for learning optimal policies in dynamic environments like Flappy Bird.

**André Brandão, Pedro Pires, and Petia Georgieva [3]:**

The authors compare Reinforcement Learning (RL) and Neuro evolution techniques in the Flappy Bird game. They conclude that RL, particularly Deep Q-Learning, enables faster convergence and better adaptability. The paper highlights how RL agents outperform evolutionary models in maintaining consistent performance across varying game conditions.

**Pratik Manoj Desai and Rhugaved Rajendra Narmade [4]:**

This study presents a Flappy Bird-playing agent trained using NeuroEvolution of Augmenting Topologies (NEAT). The authors show that dynamic neural network structures evolved through NEAT can learn complex patterns. However, the model lacks the ability to fine-tune strategies based on continuous feedback, unlike reinforcement learning techniques.

**Yash Mishra, Vijay Kumawat, and Selvakumar Kamalanathan [5]:**

The research evaluates the performance of a neural network–based Flappy Bird agent trained via Genetic Algorithms. Although the agent learned to survive longer, the paper notes inefficiencies in training speed and adaptability. The findings suggest that reinforcement learning could overcome such limitations by utilizing real-time reward feedback.

**Kachapuram BasavaRaju, V Kakulapati, and Vinay Manikant [6]:**

This paper explores Flappy Bird automation using TensorFlow. The authors focus on building an end-to-end machine learning pipeline to train agents in a simulated environment. Although not centered on reinforcement learning, the study lays a foundation for incorporating frameworks like TensorFlow.js with DQN models for browser-based gameplay.

**TensorFlow.js Team [7]:**

This documentation is critical for developers aiming to train and deploy reinforcement learning models in the browser. It enables integration of DQN models for real-time gameplay by supporting GPU acceleration and efficient model deployment in JavaScript environments.

**Tom M. Mitchell [8], Stephan Marsland [9], and Ethem Alpaydin [10]:**

These foundational texts in machine learning offer detailed explanations of supervised, unsupervised, and reinforcement learning models. They support the theoretical grounding of this study and inform model design decisions, such as policy learning, Q-value updates, and reward maximization.

International Scientific Journal of Engineering and Management (ISJEM) ISSN: 2583-6129
Volume: 04 Issue: 05 | May – 2025 DOI: 10.55041/ISJEM03388
An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata

## 4.Comparsion Table

| S. No | Title | Author's | Methodology Used | Findings from the Reference Paper |
|---|---|---|---|---|
| 1 | Learning Flappy Bird with Neuro-evolution | Daniel Shiffman. | Genetic Algorithms, Neuro evolution | Demonstrated evolution of agents using neural networks; slow convergence and no real-time adaptability. |
| 2 | Reinforcement Learning Flappy Bird | Kevin Chen | Q-Learning, experience replay | Achieved stable gameplay with Deep Q-Learning |
| 3 | RL Neuro evolution for Flappy Bird | Andre Brandao, Pedro Pires and Petia Georgieva | Deep Q-Learning, NEAT Comparison. | RL provided better adaptability and convergence than neuro evolution techniques. |
| 4 | Flappy Bird using NEAT | Pratik Desal and Rhugaved Narmade | NEAT (Neuro Evolution of Augmenting Topologies) | Evolved neural networks with dynamic structure; less efficient compared to RL for real-time adaptation. |
| 5 | Neural Approach for Game AI | Yash Mishra, Vijay Kumawat, Selvakumar Kamalanathan | Neural Networks, Genetic Algorithms | Neural agent learned obstacle navigation, but training was slow and unstable in changing environments. |
| 6 | Tensor-Flow based Flappy Bird AI | Tensor-Flow based Flappy Bird AI | ML Pipeline using TensorFlow | Built a trainable agent in TensorFlow; laid groundnut for RL integration in browser-based apps. |
| 7 | Machine Learning (Textbook) | Tom M.Mitchell | Reinforcement Learning concepts | Provided foundational knowledge of RL policies, Q-values and Markov Decision Processes. |

**International Scientific Journal of Engineering and Management (ISJEM)** **ISSN: 2583-6129**
**Volume: 04 Issue: 05 | May – 2025** **DOI: 10.55041/ISJEM03388**
**An International Scholarly || Multidisciplinary || Open Access || Indexing in all major Database & Metadata**

| | | | | |
|---|---|---|---|---|
| 8 | Machine Learning: An algorithm Perspective | Stephan Marsland | Deep Q-Learning | Explained optimization techniques like epsilon-greedy and value iteration relevant to RL agents |
| 9 | Artificial Intelligence and Games | Julian Togelius, Georgios N. Yannakakis | Evolutionary Algorithms, Player Modeling. | Evolutionary Algorithms, Player Modeling. |
| 10 | Reinforcement Learning: Introduction | Richard S. Sutton, Andrew G. Barto | Q-Learning, SARSA, Policy Gradient Methods | Comprehensive guide to RL algorithms, forms the basis for model design in projects like Flappy Bird. |

Table 1: Comparison table

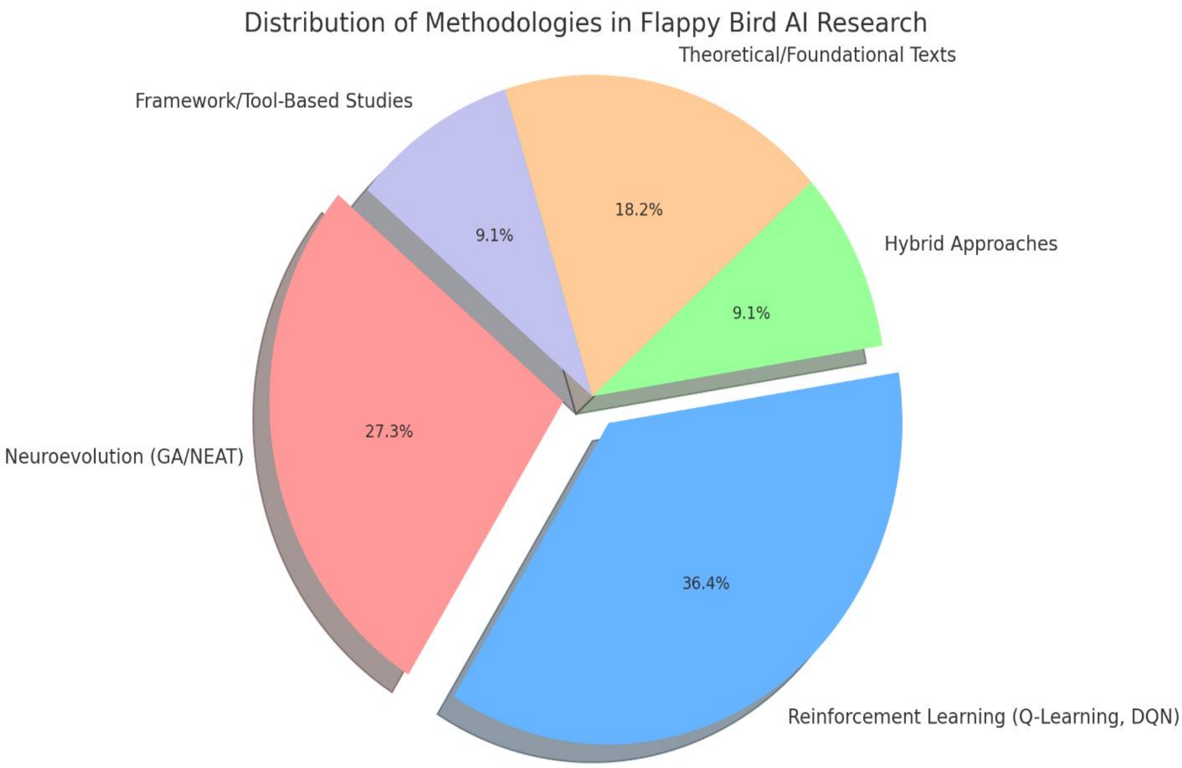


Figure-1: Distribution of Methodologies in Flappy Bird AI Research

## 5.Research Gaps

### Limited Application of Deep Reinforcement Learning Technique

- While traditional Q-learning and NEAT-based methods are widely used, advanced deep RL techniques like Deep Q-Networks (DQN), Double DQN, Dueling DQN, or Policy Gradient Methods are underexplored in the context of Flappy Bird. Most implementations use shallow networks, which limits generalization and learning efficiency.

### Insufficient Reward Optimization Strategies

- Existing models often use basic reward functions (e.g., +1 for survival), leading to sparse or delayed rewards that hinder effective learning. Few studies explore reward shaping or dynamic reward scaling, which are crucial for faster and more robust learning.

### Lack of Generalization Across Game Variants

- Many agents are trained on fixed game environments with the same gravity, pipe speed, and gap. There is limited work on training agents that generalize across multiple game settings, dynamic obstacles, or unpredictable conditions, reducing real-world adaptability.

## 6.Proposed Method

This system utilizes a Deep Q-Networks (DQN) to learn to train an AI agent to play Flappy Bird independently using reinforcement learning. DQN applies deep neural networks to estimate Q-values, which allows the agent to deal with high-dimensional state inputs such as bird position, velocity and pipe distance. Methods such as Experience Replay and Target Networks and stability and convergence while training. A fine-grained reward scheme incentivizes safe flight, centering and rewards crashes to counter rare rewards. Epsilon-greedy exploration balances learning with performance. Performance of the agent is assessed using average score, learning curves, latency and FPS. Visualization aids are used to understand Q-value trends and policy behavior. The system is extensible in a modular manner to incorporate Double DQN, Dueling DQN or policy Gradients. It demonstrates a strong framework for real-time intelligent agents. The project emphasizes the potential of reinforcement learning in dynamic game environments.

## 7.Conclusion

This research proves the efficacy of Deep Q-Networks (DQN) in training a learning Flappy Bird agent through reinforcement learning. Through addressing challenges of reward sparsity, instability and poor interpretability, the system guarantees stable, efficient learning. Experience replay target networks and shaped rewards speed up convergence and improves performance. The agent attains high scores while visualization tools offer transparency to its workings. Future research might investigate more complex algorithms such as Double DQN, Dueling DQN and Actor-Critic approaches. Transfer learning might allow generalizability across games. Real-time testing on mobile or web environments might simulate robustness. Adaptive game dynamics provide additional challenge. Incorporating explainable AI might improve decision interpretability and usability in other applications.

## 8.References

[1] Mohammed, S., Singh, N. (2021). Reinforcement Learning and Neuroevolution in Flappy Bird Game. International Journal of Engineering Research & Technology (IJERT).

[2] Zhou, Y., Lin, Y. (2016). Playing Flappy Bird with Deep Reinforcement Learning. Stanford CS231n Project Report.

[3] Yen-Chen Lin (2017). Using Deep Q-Network to Learn How to Play Flappy Bird. GitHub Repository.

[4] Sheth, D., Natarajan, R. (2020). FlapAI Bird: Training an Agent to Play Flappy Bird Using Reinforcement Learning Techniques. arXiv preprint arXiv:2003.09579.

[5] Patil, A., Sharma, A. (2023). Reinforcement Learning for Improving Flappy Bird Game. International Journal of Computer Applications.

[6] Topalovic, S. (2019). Schooling Flappy Bird: A Reinforcement Learning Tutorial using PyTorch. Toptal Engineering Blog.

[7] Kaya, E., Gursoy, M. (2018). Applying Q-Learning to Flappy Bird. Bilkent University EE Department Technical Report.

[8] Li, Z. (2019). Playing Modified Flappy Bird with Deep Reinforcement Learning. Technical Report, zezheng.li.

[9] Jaykar, N., Shinde, P. (2022). Flappy Bird Automation Using Reinforcement Algorithm. EPRA International Journal of Multidisciplinary Research (IJMR).

[10] Deshpande, R., Ahmed, R. (2022). Playing a 2D Game Indefinitely Using NEAT and Reinforcement Learning.