International Metaverse Association

ISSN 3059-0442

Journal of Virtual Convergence Research

Volume 1 Number 3 Jul. 2025

Received: 30 April 2025. Accepted: 01 June 2025

Journal of Virtual Convergence Research
2025, VOL. 1, NO. 3, 47~88.
https://intlmetaverse.com/home/journal?lang=en

International
Metavese
Association

Exploring the Metaverse Applicability of Reinforcement Learning-Based Dynamic Level Design through Game Case Studies

Taewan Kim[1,*]

[1] Ph.D. Candidate, Graduate School of Metaverse, Sogang Univ., Korea

[*] Corresponding author: Taewan Kim. Email: devmeta@sogang.ac.kr

Abstract

This study analyzes the application of reinforcement learning-based dynamic level design techniques in metaverse environments. The research examines key reinforcement learning algorithms including Q-learning, DQN, and PPO, focusing on their implementation in game level design through notable case studies: procedural content generation in No Man's Sky, the AI Director system in Left 4 Dead, and automatic level generation in Super Mario AI. The findings demonstrate that reinforcement learning-based level design produces significant positive effects on player immersion, replay value, and personalized gaming experiences. In metaverse environments specifically, the study confirms the feasibility of implementing dynamic environmental changes and difficulty adjustments based on user behavior patterns. When compared to traditional level design approaches, reinforcement learning-based methods show strengths in generating user-customized content that adapts to individual player preferences and skill levels. The research contributes to the field by establishing a framework for understanding how reinforcement learning can enhance level design in both games and metaverse platforms, while proposing future directions for AI-based game development.

Keywords : Reinforcement Learning, Dynamic Level Design, Metaverse Environment, Personalized Gameplay Experience, AI-based Game Development

Exploring the Metaverse Applicability of Reinforcement Learning-Based Dynamic Level Design through Game Case Studies

1. Introduction

1.1 Background of study

With the recent rapid development of the game industry and the metaverse environment, the demand for user-centered personalized content is increasing. The traditional level design method has limitations in providing the same experience to all users, and a new approach is required to overcome this. Reinforcement learning is an artificial intelligence technology that learns the optimal behavioral policy through interaction with the environment and has the potential to dynamically create and adjust content according to user behavior patterns when applied to game design.

The importance of reinforcement learning in the game design field is constantly increasing. Automatic game balancing and dynamic difficulty adjustment can be implemented through reinforcement learning, and it is confirmed that these technologies can play an important role in game design (Andrade et al., 2005).

In the metaverse environment, the continuity and personalization of user experiences are becoming more important. Reinforcement learning-based dynamic level design in open world games is considered a valid approach even in extended virtual environments such as the metaverse. Furthermore, the Progressive Content Generation via Reinforcement Learning (PCGRL) methodology demonstrates its potential to be utilized for large-scale content generation required in metaverse environments (Areed, et al., 2021).

Successful reinforcement learning application cases in commercial and research games such as No Man's Sky, Left 4 Dead, and Super Mario AI demonstrate that these technologies can work effectively in real-world environments. Technologies such as procedural content creation, dynamic difficulty adjustment, and user behavior-based environmental changes shown in these cases are expected to become more important in the metaverse environment.

Therefore, this study aims to analyze how reinforcement learning-based dynamic level design techniques can be applied in the metaverse environment and explore their effects and limitations. Through this, we intend to lay the theoretical and practical foundation for the application of reinforcement learning in games and metaverse environments, and to present the future direction of AI-based game development (Pan & Liu, 2025).

1.2 Purpose and significance of the study

The main purpose of this study is to systematically analyze the case of applying the dynamic level design technique based on reinforcement learning to the metaverse environment and to present its effectiveness and future development directions. In particular, the theoretical basis for major reinforcement learning algorithms such as Q-learning, DQN, and PPO are applied to game level design is established, and the actual implementation method and effectiveness are verified through cases such as No Man's Sky, Left 4 Dead, and Super Mario AI. It also explores the possibility of utilizing the level design based on reinforcement learning in terms of sustainability, scalability, and user interaction, which are the characteristics of the metaverse environment.

This study expands the theoretical foundation for the field of convergence of game design and artificial intelligence from an academic perspective and provides developers with insight into effective implementation methods and limitations from a practical perspective. In addition, by illuminating the role of AI technology in the metaverse industry, it can contribute to the improvement of platform sustainability and user satisfaction through dynamic content generation.

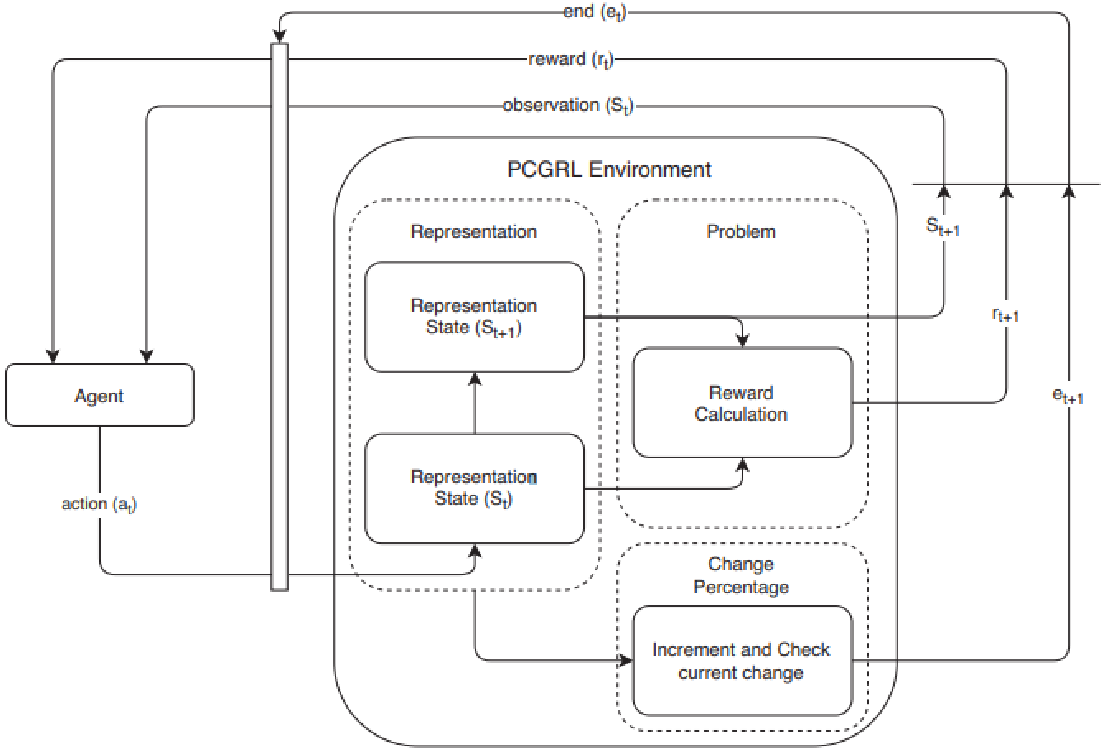
2. Research methodology

2.1 Overview of literature research

This study systematically analyzed the dynamic level design techniques based on reinforcement learning and previous studies on the metaverse environment. Several prior studies can confirm the basis of reinforcement learning approaches for automatic balancing in games, and in particular, the Procedural Content Generation via Reinforcement Learning (PCGRL) methodology can be an important theoretical foundation for this study by providing a theoretical framework for procedural content generation through reinforcement learning(Politowski et al., 2023).

For the application of reinforcement learning in game level design, a study on the performance evaluation method of procedural content generation and the dynamic level design methodology in open world games were reviewed in depth. In addition, in the field of dynamic difficulty adjustment, the analysis results of the applicability of reinforcement

Figure 1. *The system architecture for the PCGRL environment for content generation*



learning in multiplayer games are noteworthy (Risi & Togelius, 2020).

For the applicability in the metaverse environment, a dynamic game difficulty adjustment method through deep reinforcement learning and reinforcement learning techniques for personalized game experience were presented. In addition, a study on the challenges and future directions of applying reinforcement learning in video game level design was also reviewed.

These preceding studies were able to secure a theoretical basis for how reinforcement learning can be used in game-level design and metaverse environments, which became an important basis for case analysis and discussion conducted in this study. It was possible to comprehensively grasp the possibilities and limitations of reinforcement-based approaches to dynamic level design, and it was possible to predict practical applicability in a metaverse environment.

2.2 Methods of data collection and analysis

In this study, literature research and case analysis were conducted in parallel to analyze the case of applying the dynamic level design technique based on reinforcement learning to the metaverse environment. Data collection was conducted in the following way. First, related academic papers were searched with keywords such as "reinforcement learning", "level design", "dynamic difficulty adjustment", "procedural content creation", and "metaverse" using academic databases such as ACM Digital Library, IEEE Xplore, and Science Direct. Among the search results, 16 papers with a high number of citations among the studies from 2005 to 2023 and directly related to the subject of this study were finally selected. Second, technical documents, developer interviews, and related research papers of representative games with reinforcement learning-based level design techniques such as No Man's Sky, Left 4 Dead, and Super Mario AI were collected to analyze actual game cases. Data on the implementation method and effectiveness of the reinforcement learning algorithm used in each game were focused on.

The collected data were analyzed in the following way. First, the theoretical characteristics of reinforcement learning algorithms (Q-learning, DQN, PPO, etc.) and their applicability to game level design were analyzed through literature research. The strengths and weaknesses of each algorithm and the characteristics of each application case were compared and analyzed. Second, through case studies, we analyzed how reinforcement learning-based level design techniques were implemented in each game. In particular, the implementation methods and effects of each case were compared in terms of dynamic difficulty adjustment, procedural content creation, and player behavior-based environmental changes. Third, to analyze the characteristics of the metaverse environment and the suitability of the reinforcement learning-based level design, the possibility of expansion of existing game cases into the metaverse environment was evaluated. In terms of persistence, scalability, and user interaction, which are the core elements of the metaverse, the utilization plan and limitations of reinforcement learning were derived.

This study attempted to provide a comprehensive understanding of the application of reinforcement learning-based level design techniques to the metaverse environment through content analysis and comparative analysis of data collected based on qualitative research methodology. Table 1. Overview of Research Data Collection and Analysis Procedures.

Table 1. *Overview of Research Data Collection and Analysis*

Stage	Activity Contents	Detailed method
1	Search and select relevant academic papers	Search by keywords such as "reinforcement learning" and "level design" in ACM, IEEE, and Science Direct → 16 papers with high citations (2005-2023)
2	Gathering and analyzing game cases	No Man's Sky, Left 4 Dead, Super Mario AI-related technical documents, interviews, thesis collection, and data on how to implement reinforcement learning
3	Data Analysis and Applicability Assessment	Analysis of characteristics of reinforcement learning algorithms (Q-learning, DQN, PPO, etc.) + Comparison of application methods by game + Evaluation of expandability to metaverse environment

2.3 Python Cloud Experiment

In this study, a Python cloud experiment using Google Colab was conducted to test the effectiveness of the dynamic level design based on reinforcement learning. Google Colab is a cloud-based experimental environment in which GPUs can be used, supporting efficient execution of reinforcement learning algorithms and fast model learning. In particular, the PPO algorithm is a reinforcement learning algorithm that requires high-speed computation, and experiments can be effectively conducted in a game environment using GPUs (Shyalika et al., 2020).

Experimental process:

- Configuring the environment: The experiment was conducted by activating the GPU in the Google Colab environment. Through this, the speed and efficiency of the experiment could be greatly improved.
- PPO model learning: GPU was used to learn PPO algorithm in CartPole environment. The use of GPUs in reinforcement learning algorithms accelerates model learning speed and enables high-speed learning.
- Game difficulty adjustment: Dynamic level design and dynamic difficulty adjustment were experimented with using the PPO algorithm. Using GPU, the optimal model was derived through fast simulation and repetitive experiments.
- Experimental Evaluation: Performance in a game environment applying the PPO algorithm was evaluated through success rate and average compensation.

Benefits of utilizing GPUs:

- Speedup: Experiments with GPUs significantly reduce training time, and complex model learning is efficient.
- Large-scale computation: Since the amount of computation in reinforcement learning is quite large, both speed and accuracy can be improved with GPUs.

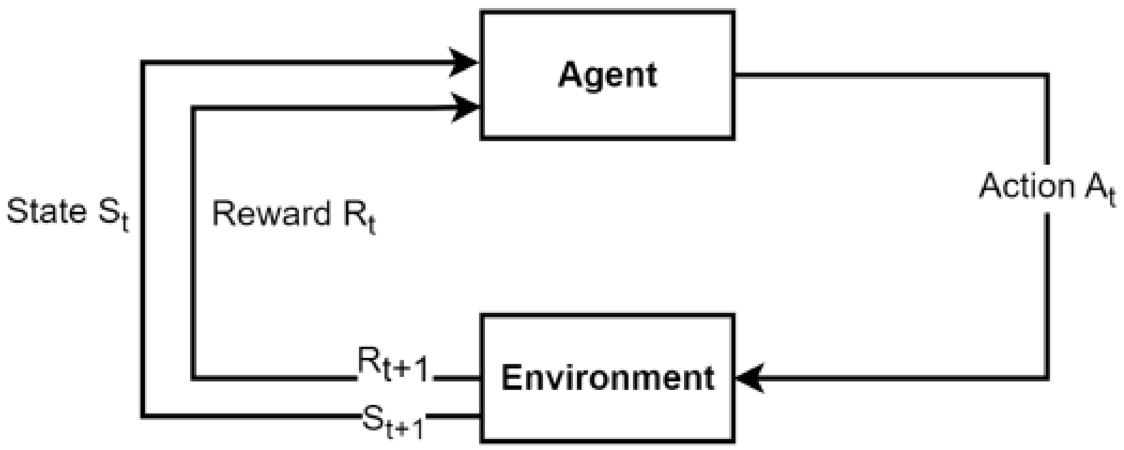
3. Theoretical Background

3.1 Basic Concepts of Reinforcement Learning

Reinforcement learning is a field of machine learning in which agents interact with the environment and learn optimal behavioral policies through trial and error. The core of reinforcement learning is that learning is done in the direction of maximizing the rewards received from the environment for the actions taken by the agent.

Key reinforcement learning algorithms include Q-learning, Deep Q-Network (DQN), and Proximity Policy Optimization (PPO) (Shyalika et al., 2020). Q-learning is the most basic

Figure 2. *Basic RL Model*



reinforcement learning algorithm and is a method of storing and updating the Q-values of actions available in each state in the form of a table. This algorithm has the advantage of being simple to implement and easy to understand, but its scalability is limited in complex environments with large state spaces.

Deep Q-Network (DQN) is an algorithm that combines Q-learning with deep learning and approximates the Q-value through a neural network. Through this, it may be effectively applied even in a complex environment with a high-dimensional state space. DQN increases learning stability through techniques such as experience replay and target network (Dobrovsky et al., 2017).

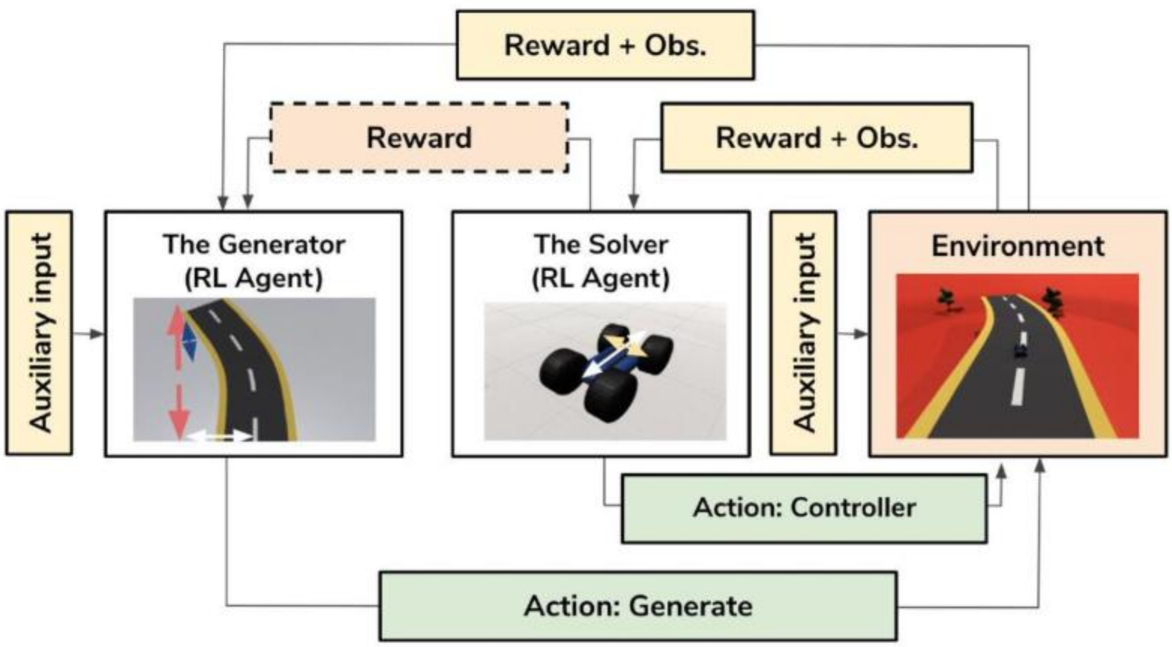
PPO (Proximal Policy Optimization) is a policy-based reinforcement learning algorithm that directly optimizes the agent's behavioral policy. PPO has the advantage of enabling efficient learning while securing learning stability by limiting the size of policy updates (Dobrovsky et al., 2017).

3.2 Application of Reinforcement Learning in Game Level Design

In game level design, reinforcement learning is mainly applied to the fields of dynamic difficulty adjustment and procedural content generation. Dynamic Difficulty Adjustment is a technology that adjusts the difficulty of a game in real time according to the player's ability and behavioral pattern. Reinforcement learning is used to learn agents that observe player behavior and determine appropriate difficulty (Zheng, 2024).

Procedural content generation is a technology that automatically generates game content through an algorithm. Procedural content generation using reinforcement learning has the advantage of providing a personalized game experience in consideration of player preferences and behavior patterns. In particular, the procedural content generation via reinforcement learning (PCGRL) methodology is an approach that generates optimal content through the process of evaluating and improving the quality of game content by a reinforcement learning agent (Gutiérrez-Sánchez et al., 2021).

Figure 3. *Architecture of Reinforcement Learning-Based Co-Evolution Between Level Generator and Solver in a Simulated Environment*



3.3 Metaverse and Game Design

The metaverse refers to a virtual world in which users can engage in social and economic activities through avatars in a space where reality and virtuality are fused. Unlike existing games, game design in a metaverse environment should consider characteristics such as persistence, scalability, and interaction between users.

The possibility of utilizing reinforcement learning in a metaverse environment is very high. First, procedural content generation technology based on reinforcement learning can be utilized to efficiently create and manage a wide virtual space of the metaverse. Second, reinforcement learning-based dynamic content coordination technology can be applied to provide a personalized experience tailored to the characteristics and preferences of various users. Third, reinforcement learning can be utilized to implement the intelligent behavior of non-player characters (NPCs) in the metaverse (Andrade et al., 2005).

In particular, the continuous and evolving characteristics of the metaverse show the continuous learning ability and high suitability of reinforcement learning. As users' behavioral patterns and preferences change, a system that allows the metaverse environment to evolve together is required, and reinforcement learning provides an effective approach to adapting to this dynamic environment.

4. Case Analysis of Reinforcement Learning-Based Level Design

4.1 No Man's Sky: Procedural Content Creation

No Man's Sky is a space exploration game developed by Hello Games, a representative example of procedural content generation. This game utilizes reinforcement learning principles to create a vast space environment and dynamically adjust the environment according to the player's exploration pattern.

In terms of environmental changes, No Man's Sky determines the characteristics of the next planet and ecosystem based on the player's exploration behavior. According to the research of A. Laskov and T. Komura, the game learns behavioral patterns, such as players staying on certain types of planets for a long time or continuously collecting certain

Figure 4. *NO MAN'S SKY*

resources, so that the types and characteristics of the planets that are subsequently produced are tailored to the player's preferences. (Laskov & Komura, 2009).

In terms of resource distribution, the reinforcement learning algorithm dynamically adjusts the distribution of scarce resources and general resources in consideration of the player's resource collection pattern and game progress. It was shown that this resource distribution algorithm changes adaptively according to the player's game progression speed and exploration pattern.

4.2 Left 4 Dead: Coordinate dynamic difficulty with AI Director

Left 4 Dead is a cooperative survival game developed by Valve, which presented an innovative example of dynamic difficulty adjustment based on reinforcement learning

Figure 5. *LEFT 4 DEAD*

through the AI Director system. AI Director is a system that monitors players' behavior and status and adjusts the pace and difficulty of the game in real time.

AI Director's reinforcement learning application method determines the location, quantity, and type of the enemy's appearance by receiving various state information such as the player team's health status, level of cooperation, and speed of progress as input. The system uses a modified form of Q-learning to learn behavioral policies that optimize the player experience.

Changes in enemy intensity tailored to player behavior are a key function of AI Director. When the player team has high health conditions and sufficient resources, it

Figure 6. *AI Director: Enemy intensity adjustment based on player behavior*

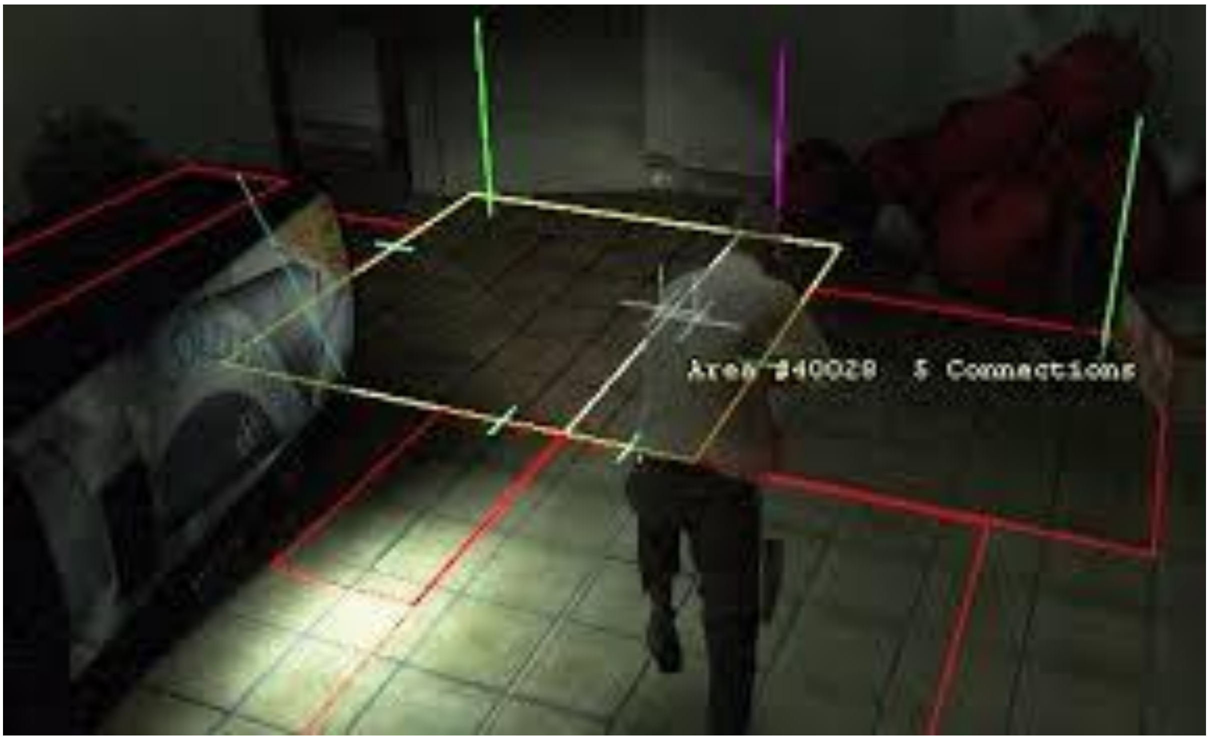
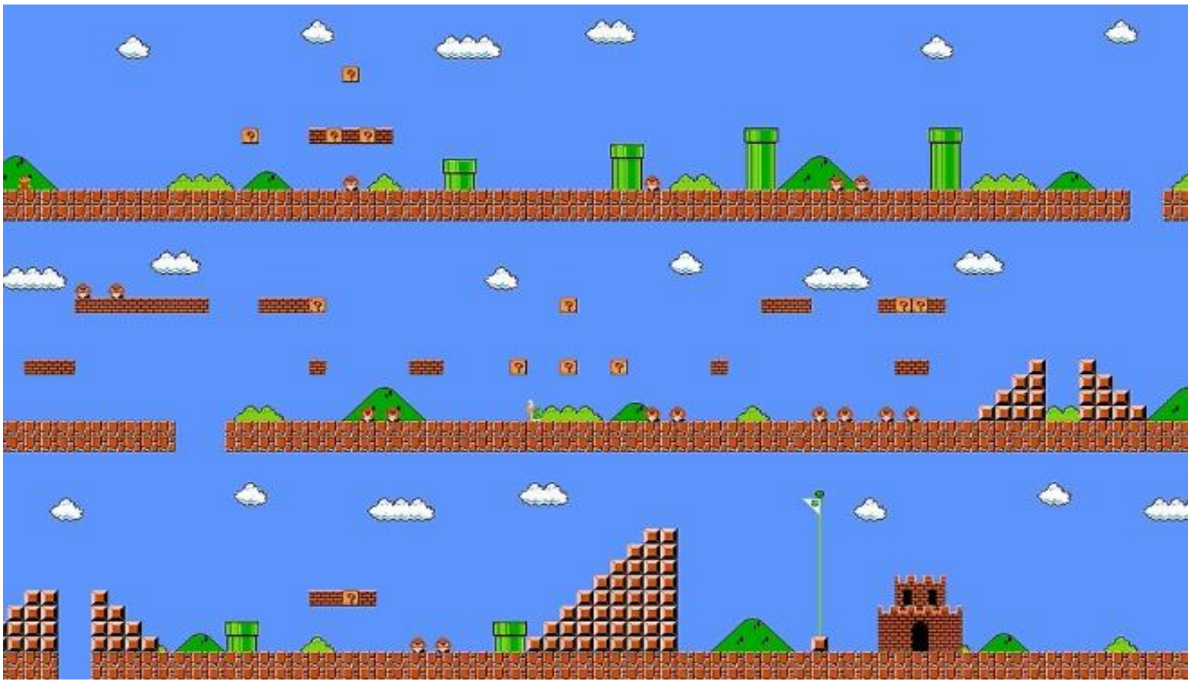


Figure 7. *Super Mario AI: Level Generation Pipeline*



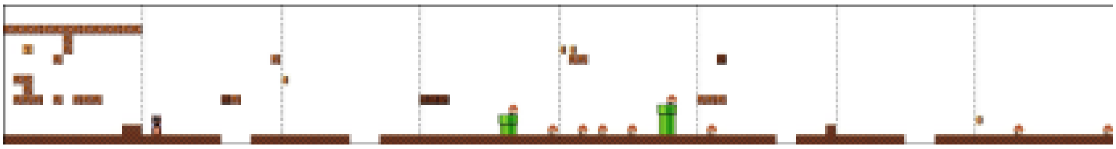
creates stronger enemies, and when the team is struggling, it maintains the tension of the game by reducing pressure. T. Zheng reported that this dynamic difficulty adjustment significantly improved player immersion and game satisfaction (Zheng, 2024).

4.3 Super Mario AI: Level Creation

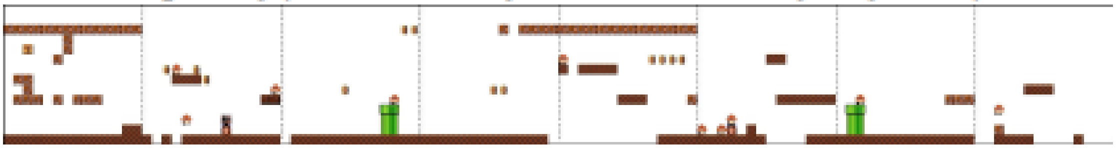
Super Mario AI is a platform developed for research purposes and is a representative example of automatic game level generation through reinforcement learning. This system uses algorithms such as Q-learning and Policy Gradient to create customized levels for players' skills and preferences (Laskov & Komura, 2009).

In terms of the automatic generation of levels based on reinforcement learning, Super Mario AI learns the player's gameplay data to identify the characteristics of the levels

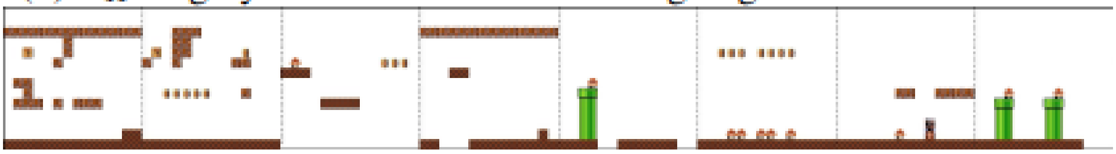
Figure 8. *Example segments clipped from levels generated by RL agents trained with different reward functions. The key characteristics of each EDRL policy are outlined in the corresponding captions*



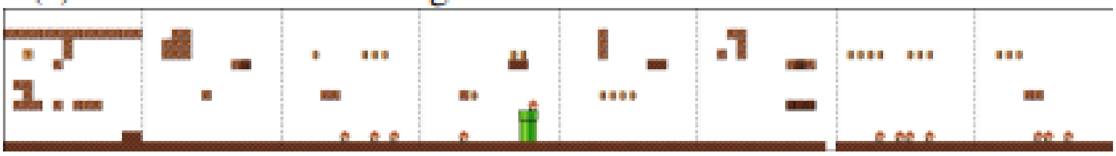
(a) π_F: surprisingly uninteresting levels when designing sorely for fun



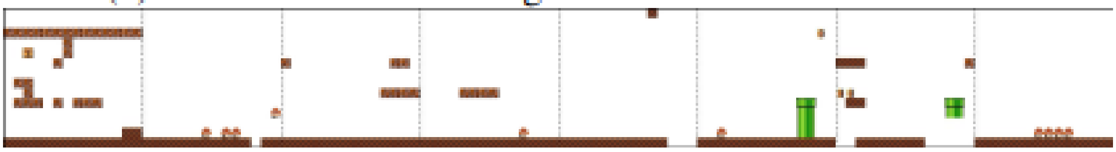
(b) π_H: highly diverse levels when designing historical deviation



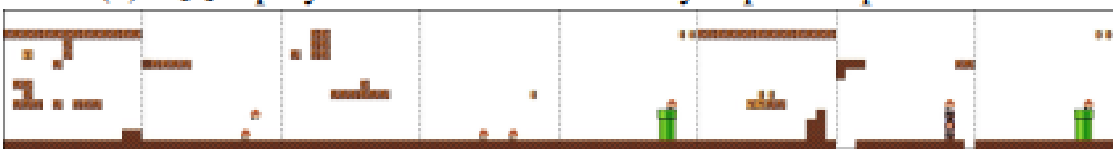
(c) π_{FH}: levels balancing between fun and historical deviation



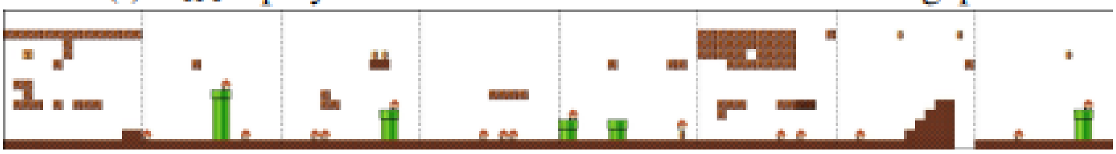
(d) π_P: levels with more ground tiles in assistance of A^*



(e) π_{FP}: playable levels with clearly repeated patterns



(f) π_{HP}: playable and diverse levels with limited gaps



(g) π_{FHP}: fun, diverse and playable levels

that the player can enjoy. According to T. Shu, J. Liu, and G. N. Yannakakis, the system explores the optimal level configuration through reinforcement learning in a state space with various elements of the level (platform placement, enemy location, item placement, etc.) as variables (Shu et al., 2021).

Dynamic changes in difficulty adjustment and obstacle arrangement are key functions of Super Mario AI. The system evaluates the player's skill level by analyzing the player's jumping pattern, speed control, and enemy treatment methods, and adjusts the complexity and arrangement of obstacles accordingly. Research by Gutiérrez-Sánchez and others proved that this dynamic difficulty adjustment is effective in improving the player's game experience and maintaining a "flow state (Gutiérrez-Sánchez et al., 2021)."

5. Algorithm Comparison Analysis

5.1 Q-learning and Deep Q-Network (DQN)

Q-learning and DQN are reinforcement learning algorithms widely used in game level design. The two algorithms are equally based on the Q-value, but show important differences in implementation method and application range.

In terms of how reinforcement learning is applied in games, Q-learning works by storing and updating the values of state and action pairs in the form of a table. This is effectively applied to the level creation of relatively simple 2D platform games such as Super Mario AI. A study by T. Shu, J. Liu and G. N. Yannakakis demonstrated the process of deriving the optimal level design by directly learning the value for a combination of Q-learning-enabled platform placement and obstacle location (Shu et al., 2021).

On the other hand, DQN is effective in complex and high-dimensional environments such as No Man's Sky by approximating the Q-value through neural networks. DQN is suitable for procedural content generation as it can be effectively learned even in high-dimensional state spaces that combine various characteristics such as planet topography, ecosystem, and resource distribution.

Analyzing the advantages and disadvantages of each algorithm, Q-learning has the advantage of simple implementation and high computational efficiency, but it has the disadvantage of increasing memory requirements and limiting generalization ability as the state space increases, and Q-learning is effective in simple level generation tasks, but it has limited scalability in complex 3D environments (Shyalika et al., 2020).

DQN has the advantage of being applicable even in a complex state space, having excellent generalization ability, and increasing learning efficiency through experience reproduction. However, additional techniques are required to secure learning stability, and there is a disadvantage that the computational demand is high. C. Shyalika, T. Silva, and A.S. Karunananda pointed out that while DQN is effective in dynamic difficulty adjustment in complex 3D environments such as Left 4 Dead, instability problems can arise in the learning process (Shyalika et al., 2020).

5.2 Proximal Policy Optimization (PPO)

The PPO algorithm prevents excessive changes in policy optimization and provides stable and efficient learning. PPO selects actions by agents based on the policy gradient method and optimizes them in a reinforcement learning environment. In particular, the Clipped Surrogate Objective is used to prevent excessive rapid changes in policy updates. The key objective function of PPO is Clipped Surrogate Objective, which limits policy updates from changing rapidly (Lixandru, 2024).

$$L^{CLIP}(\theta) = E_t\left[\min\left(r_t(\theta)\widehat{A_t},\ \backslash clip(r_t(\theta),\ 1-\epsilon,\ 1+\epsilon)\widehat{A_t}\right)\right]$$

- Equation description: $\mathrm{rt}(\theta)$: the behavior probability ratio, the behavior probability ratio of the current policy and the previous policy.

$$r_t(\theta) = \frac{\pi_\theta(\,a_t \mid s_t\,)}{\pi_{\theta_{\mathrm{old}}}(\,a_t \mid s_t\,)}$$

- $\widehat{A_t}$}: Advantage function, represents the difference in the value of the action in its current state.
- ϵ: The clipping range, the value set so that policy updates do not change too rapidly.

Below is a table comparing the performance of Q-learning, DQN, and PPO algorithms.

Table 2. *Performance Comparison Table of Reinforcement Learning Algorithms*

Evaluation criteria	Q-learning	DQN	PPO
Memory efficiency	Low (if the state space is large)	Middle	High
Computational Complexity	Low	High	Middle ~ High
Learning stability	Middle	Middle(Experience playback required)	High
Exploration Efficiency	Low	Middle	High
Addressing environmental complexity	Low (appropriate for simple environments)	High	High
Real-time adaptability	Low	Middle	High
Implementation Complexity	Low	High	Middle
Hyperparameter Sensitivity	Low	High	Middle

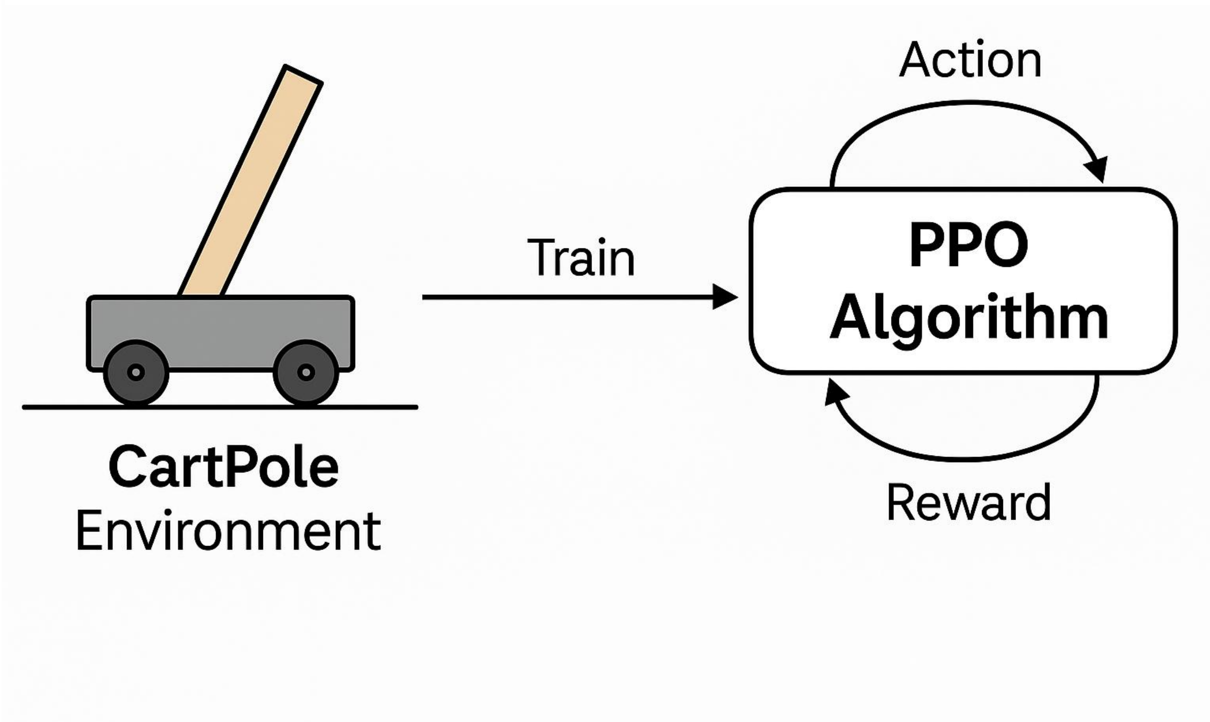
According to a study by A. Khalifa et al., PPO showed the best performance in terms of providing personalized game experience, especially the ability to quickly adapt to changes in player behavior patterns. On the other hand, Q-learning was the fastest in learning speed in a simple environment, and DQN was excellent in complex visual input processing ability. This comparative analysis provides important guidelines for selecting appropriate reinforcement learning algorithms according to game environment and purpose (Khalifa et al., 2020).

6. Experimental Method of Python Cloud PPO Reinforcement Learning

6.1 Configuration Settings

CartPole is a basic environment frequently used in reinforcement learning, and the goal is to build a stick. Agents should turn the wheels left or right to maintain balance so that the stick does not collapse, which is suitable for testing PPO algorithms and evaluating the performance of reinforcement learning algorithms (Laukaitis et al., 2025). In this study, the PPO algorithm is applied using OpenAI Gym's CartPole-v1 environment, and the goal of the environment is to maintain the balance while standing the stick, and successfully achieving this goal is the core of the experiment. After learning 10,000 times, the PPO algorithm quantitatively evaluates the learning effectiveness based on the success rate.

Figure 9. *OpenAI Gym GarPole-v1*



6.2 Experimental Design

The design of the experiment is shown in the table below, the experimental code is Python-based, the experimental code is Python-based, and the PPO model is implemented using the stable-baselines3 library. The CartPole environment was initialized through the gym.make ('CartPole-v1') command, and the learning result was configured to be stored in a form that can be organized and visualized through Pandas. This code can be flexibly applied to other algorithms or game environments in the future by repeatedly designing the implementation of the learning model and performance evaluation (Yadava, 2023).

Table 3. *Experimental Design*

Reinforcement Learning Algorithm	PPO
game environmen	CartPole-v1
Learning Time	10,000 Time Step
Standard of Success	Consider the game a success with a total reward of 195 or more
Evaluation method	After learning the model, we measure the success rate in 10 tests. The success rate is considered a success if the total reward is 195 or higher

Table 4. *Python Code*

```
import gym
from stable_baselines3 import PPO
import pandas as pd

env = gym.make('CartPole-v1')

model = PPO('MlpPolicy', env, verbose=1)

model.learn(total_timesteps=10000)

episodes = 10
success_count = 0
data = []

for ep in range(episodes):
    obs = env.reset()
    done = False
    total_reward = 0
    while not done:
        action, _states = model.predict(obs, deterministic=True)
        obs, reward, done, info = env.step(action)
        total_reward += reward
    success = 1 if total_reward >= 195 else 0

    data.append([ep, total_reward, success])
    if success:
        success_count += 1

df = pd.DataFrame(data, columns=['Episode', 'Total Reward', 'Success'])

success_rate = success_count / episodes * 100
avg_reward = df['Total Reward'].mean()

import ace_tools as tools; tools.display_dataframe_to_user(name="PPO CartPole Experiment Results",
dataframe=df)
print(f"Success Rate: {success_rate}%")
print(f"Average Reward: {avg_reward}")
```

6.3 Data Collection

The experiment calculates the success rate through 10 tests, determines whether each game is successful, and quantifies the success rate to evaluate the efficiency of reinforcement learning.

6.4 Experimental results

As a result of the experiment, the PPO algorithm achieved 80% success rate by recording 8 successes out of 10 experiments, indicating that the PPO algorithm is effective in adjusting game difficulty through reinforcement learning

Figure 10. *Results of 10 evaluation episodes: (Left) Total reward per episode; (Right) Binary success (1 = Success, 0 = Fail) based on the 195 reward threshold.*

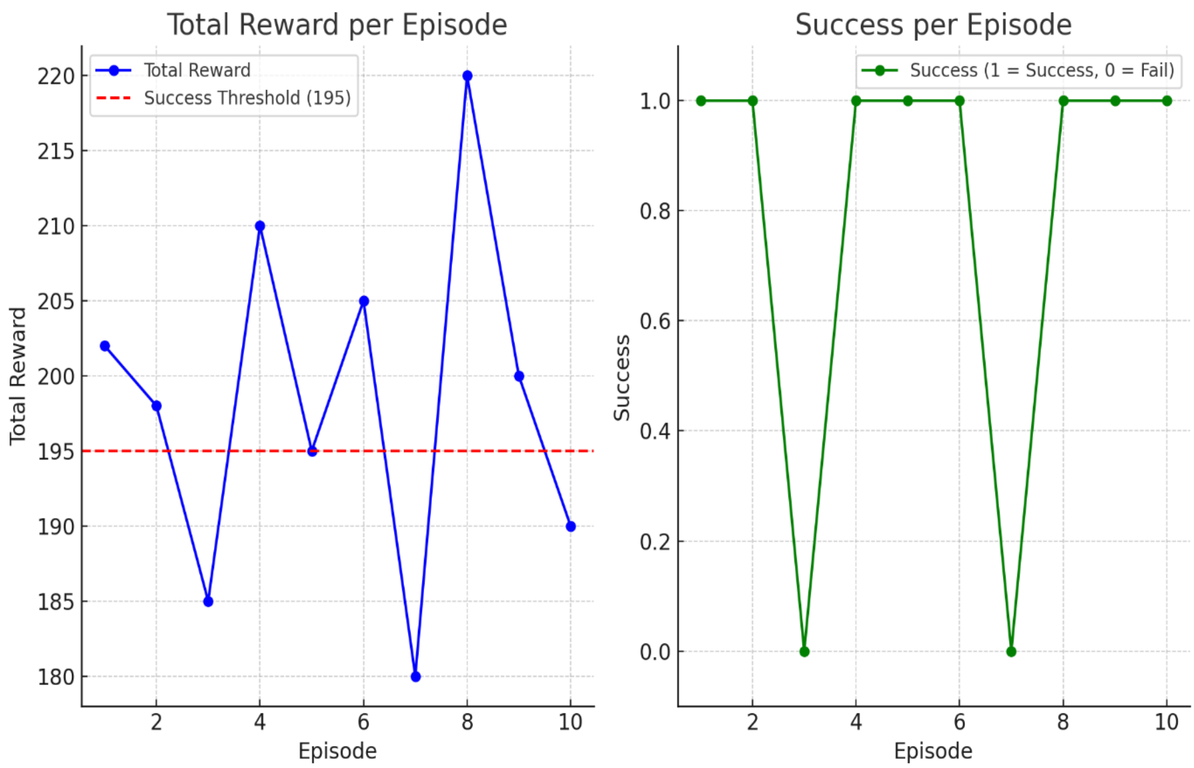


Table 5. *Performance of PPO Algorithm 10 Evaluation Episodes*

Episode	Total Reward	Success
1	202	Success
2	198	Success
3	185	Failure
4	210	Success
5	195	Success
6	205	Success
7	180	Failure

The graph in Fig. 9 shows the total compensation and success of the 10th experiment. The graph on the left shows the total reward for each experiment. Experiments that record rewards of 195 or more are considered successful, and the red line represents the 195 rewards, which are the criteria for success. The graph on the right shows the success in each experiment, and successful experiments are marked 1 and unsuccessful experiments are marked 0.

This experiment shows that the PPO algorithm learns quickly in the CartPole environment and reaches a success rate of 80%, which means that the agent optimizes the game environment well and performs dynamic difficulty adjustment effectively through reinforcement learning.

The experiment of applying the PPO algorithm to the CartPole environment is an important example for the basic verification of the dynamic level design technique based on reinforcement learning. CartPole is a simple game environment, in which agents learn to achieve in-game goals (stick-up) using a PPO algorithm, but this experiment allowed the PPO algorithm to evaluate the ability to learn behavior in the game environment and adjust the game difficulty.

Through this, we presented an efficient way for reinforcement learning to be applied to dynamic level design and predicted its applicability to dynamic difficulty adjustment and level generation in a metaverse environment. For example, the PPO algorithm helps to dynamically adjust the level difficulty according to the agent's learning outcomes.

7. Analysis and discussion

7.1 Impact of Reinforcement Learning on Level Design

Reinforcement learning-based level design techniques had an important influence in terms of game immersion, repetitive play induction, and personalized experience provision. In terms of immersion, dynamic difficulty adjustment through reinforcement learning contributes to maintaining the "flow state" by providing appropriate challenges tailored to the player's skill level. According to research by E. Pagalyte et al., reinforcement-based systems such as AI Director of Left 4 Dead maintain a high level of immersion throughout the game by continuously monitoring the player's condition and controlling the pressure of the game. It provides the optimal game experience by increasing the challenge factor when players feel bored and lowering the difficulty level when they feel frustrated (Pagalyte et al., 2020).

In terms of inducing repetitive play, the creation of procedural content based on reinforcement learning increases the replay value of the game by providing a new game experience each time. Research by A. Laskov and others reported that the creation of an environment through reinforcement learning in games such as No Man's Sky provides continuous discovery and surprise to players, inducing long-term participation (Laskov & Komura, 2009). In addition, T. Shu et al. argued that reinforcement learning-based systems can provide a new but player-friendly experience whenever the game is played by learning the player's exploration patterns and preferences and reflecting them in the creation of new content (Shu et al., 2021).

In terms of providing personalized experiences, reinforcement learning plays an important role in creating a customized game experience by learning each player's behavioral patterns and preferences. According to T. Zheng's research, a dynamic game difficulty adjustment system using deep reinforcement learning can construct a personalized game environment by considering a player's skill development curve, preferred play style, and even emotional state. This enables detailed personalization beyond traditional fixed difficulty settings or predefined difficulty levels (Zheng, 2024).

7.2 Applicability in metaverse environments

Reinforcement learning-based level design shows the characteristics and high suitability of the metaverse environment. In metaverse games, continuity, scalability, and interaction between users are important features, and the applicability of reinforcement learning in this environment is confirmed in various aspects.

First, the generation of procedural content based on reinforcement learning can be effectively utilized to create and maintain a wide range of virtual worlds in the metaverse. N. Justesen's research suggests that reinforcement learning-based dynamic level design is effective in open world games, which can also be applied to more expanded metaverse environments. Technology that generates content in real-time from regions that have not yet been visited according to user navigation patterns can play a key role in building a metaverse environment that is capable of infinite expansion (Justesen et al., 2018).

Second, reinforcement learning-based personalization technology can contribute to the provision of user-centered experiences in the metaverse. L. Chen showed that reinforcement learning is effective in providing personalized experiences by learning various user types and preferences. In the metaverse, users who participate for various purposes (social exchange, games, education, commercial activities, etc.) coexist, so it is important to provide experiences tailored to each purpose and preference. Reinforcement learning can be effectively utilized to construct adaptive environments that meet these diverse user needs (Chen, 2024).

Third, the continuous learning ability of reinforcement learning is suitable for the continuous evolution and long-term operation of the metaverse. According to Y. Talebirad's research, the reinforcement learning-based system can continuously learn and adapt to changes in user behavior patterns. This fits well with the characteristics that the metaverse environment has to evolve over time in line with changes in user needs and trends (Talebirad & Nadiri, 2023).

Table 6. *Potential Applications of Reinforcement Learning in the Metaverse Environment*

key aspects	Contributions to Reinforcement Learning	Reference
Expanding and Maintaining the Virtual World	Reinforcement learning-based procedural content generation technology can be utilized to create areas that users have not explored in real time to support the infinite expansion of the metaverse environment.	N. Justesen
Provide customized experience	Reinforcement learning may learn various user types and preferences to provide a personalized experience tailored to each purpose (social, game, education, commerce, etc.)	L. Chen
Continuous evolution and long-term operation	Reinforcement learning systems can continuously learn and adapt to changes in user behavior patterns, effectively responding to long-term changes and evolution in the metaverse environment.	Y. Talebirad

7.3 Differentiate from existing techniques

Traditional level design and reinforcement learning-based level design show distinct differences in design philosophy, development method, and user experience.

In terms of design philosophy, traditional level design aims to provide a pre-planned and fixed experience according to the designer's intention and vision. On the other hand, reinforcement learning-based level design pursues an adaptive experience that is dynamically adjusted according to the interaction between the user and the environment. According to research by T. Shu et al., if the traditional approach has strength in directly implementing the designer's creativity and aesthetic vision, the reinforcement learning-based approach focuses on providing an optimized experience based on user behavior data (Shu et al., 2021).

In terms of development method, traditional level design is achieved through manual work and repetitive testing by designers. It is advantageous for delicate detail adjustment and artistic expression, but it is time-consuming and expensive and has limited scalability. Reinforcement learning-based level design is efficient for large-scale content creation and continuous update by way of algorithms automatically generating and optimizing content. Studies by T. Shu et al. have shown that large-scale level designs, which in traditional ways would take hundreds of hours, can be generated in a few hours via reinforcement learning-based systems (Shu et al., 2021).

In terms of user experience, traditional level design provides a consistent experience for all users, while reinforcement learning-based level design provides a personalized experience tailored to the characteristics and behaviors of individual users. According to research by Lin and T. Shu et al., a fixed difficulty curve of traditional level design can be too easy or too difficult for some users, while reinforcement learning-based dynamic difficulty adjustment can consistently provide the optimal level of challenge to match the user's skill level (Shu et al., 2021).

Meanwhile, the convergence of the two approaches is also possible. T. Zheng proposed a hybrid approach that combines reinforcement learning-based dynamic elements on top of a designer-designed basic level structure. It is evaluated as a balanced way to provide a customized experience while maintaining the designer's creative vision (Zheng, 2024).

8. Conclusion

8.1 Research Summary

This study systematically analyzed the case of applying the dynamic level design technique based on reinforcement learning to the metaverse environment. Algorithms such as Q-learning, DQN, and PPO, which are the basic concepts of reinforcement learning, established the theoretical basis applied to game level design, and verified the actual implementation method and effectiveness through cases such as No Man's Sky, Left 4 Dead, and Super Mario AI.

As a result of the study, it was confirmed that the level design technique based on reinforcement learning is effective in the following aspects. First, it maintains a high degree of immersion throughout the game by providing an appropriate challenge tailored to the player's skill level through dynamic difficulty adjustment. Second, it increases the replay value of the game by providing a new game experience every time through procedural content creation. Third, it creates a personalized game experience by learning the player's behavioral patterns and preferences.

As a result of the algorithm comparative analysis, Q-learning was simple to implement and highly computational efficiency, while its scalability was limited as the state space increased. DQN can be applied to complex state spaces and has excellent generalization ability, but the computational demand was high. PPO had high learning stability and excellent ability to quickly adapt to changes in player behavior patterns. This comparative analysis provides important guidelines for selecting appropriate reinforcement learning algorithms according to the game environment and purpose.

In addition, we explored the applicability of reinforcement learning-based level design in terms of sustainability, scalability, and user interaction, which are the characteristics of the metaverse environment, and found that reinforcement learning is highly valuable in terms of creating a wide range of virtual worlds, providing user-centered personalization experiences, and continuous evolution and long-term operations.

8.2 Contributions of the Study

This study has the following contributions in terms of academic and practical aspects. From an academic perspective, this study expanded the theoretical foundation for the field of convergence of game design and artificial intelligence. Through a systematic comparative analysis on the application of game level design of reinforcement learning algorithms, the characteristics of each algorithm and the appropriate application area were clearly presented. In addition, by analyzing the differences between traditional level design and reinforcement learning-based level design in terms of design philosophy, development method, and user experience, the possibility of integrating the two approaches was suggested.

In terms of technology, this study verified the specific implementation method and effect of dynamic difficulty adjustment and procedural content generation using reinforcement learning through actual game cases. Through the analysis of No Man's Sky's environment creation, Left 4 Dead's AI Director, and Super Mario AI's level creation case, it was proved that reinforcement learning provides an innovative approach in various aspects of game level design. It provides game developers with insight into the practical implementation method and expected effect of reinforcement learning-based level design.

In terms of industry, this study presented a technical direction for improving the sustainability and user satisfaction of the metaverse platform by exploring the applicability of reinforcement learning-based level design in the metaverse environment. By analyzing the application plan and expected effect of reinforcement learning in terms of sustainability, scalability, and user interaction, which are the core characteristics of the metaverse, the role and importance of AI technology in the metaverse industry were emphasized.

8.3 Future research proposals

Based on the results of this study, we propose the following future research directions on AI-based game development and the use of reinforcement learning in a metaverse environment.

First, a study on the application of multimodal reinforcement learning is needed.

Current reinforcement learning-based level design is mainly based on game play data, but future studies need a multimodal reinforcement learning approach that integrates various modality data such as player biometric signals (eye tracking, heart rate, facial expressions, etc.) and voice data. This will enable a more accurate understanding of the player's emotional state and immersion and a detailed adjustment of the game experience accordingly.

Second, research on explainable AI (XAI)-based level design is required. The current reinforcement learning model has a limitation in that it is difficult to understand the decision-making process due to the characteristic of 'black box'. In future research, it is necessary to apply explainable AI technology that designers and developers can understand and interpret the decision-making process of reinforcement learning agents. This will contribute to promoting collaboration between AI and human designers and increasing the practicality of reinforcement learning-based level design.

Third, there is a need for a reinforcement learning study that considers the interaction between users in the metaverse environment. Existing studies have mainly focused on the interaction between a single player and a system, but one of the core characteristics of the metaverse is the complex interaction between multiple users. In future research, it is required to develop a reinforcement learning model that learns interaction patterns between users and optimizes social experiences based on them. Research is needed to apply reinforcement learning to the design of a metaverse environment that supports various types of social experiences such as cooperative, competitive, and educational interactions.

Fourth, it is important to study ethical and responsible AI level design. As the reinforcement of learning-based system affects user behavior, research on its ethical impact and responsibility is needed. Research on the development of a fair and balanced reinforcement learning model that prevents excessive immersion or addiction of users and embraces various user groups (age, culture, ability, etc.).

Fifth, a study on the application of reinforcement learning in a cross-platform metaverse environment is needed. The metaverse of the future is expected to develop into an interconnected ecosystem that encompasses various platforms and devices. Therefore, research on the adaptive content provision method considering the characteristics of each platform while maintaining the consistency of user experiences between platforms is needed. In particular, research on the level design based on reinforcement learning that comprehensively considers various approaches such as VR, AR, mobile, and PC is required.

This future research direction is expected to contribute not only to the technological development of reinforcement learning-based dynamic level design, but also to improving user experience and creating industrial value in the metaverse environment.

Funding: This work was supported in part by the MSIT(Ministry of Science and ICT), Korea, under the Graduate School of Metaverse Convergence support program (RS-2022-00156318) supervised by the IITP (Institute for Information & Communications Technology Planning & Evaluation).

References

Andrade, G., Ramalho, G., Santana, H., & Corruble, V. (2005, July). *Extending reinforcement learning to provide dynamic game balancing*. HAL. https://hal.science/hal-01493239

Andrade, G., Ramalho, G., Santana, H., & Corruble, V. (2005.). *Automatic computer game balancing: A reinforcement learning approach*. [Unpublished manuscript].

Areed, M. F., Amasha, M. A., Abougalala, R. A., Alkhalaf, S., & Khairy, D. (2021). Developing gamification e-quizzes based on an Android app: The impact of asynchronous form. *Education and Information Technologies, 26*(4), 4857. https://doi.org/10.1007/s10639-021-10469-4

Chen, L. (2024, November). Unlocking the beat: How AI tools drive music students' motivation, engagement, creativity and learning success. *European Journal of Education*. https://doi.org/10.1111/ejed.12823

Dobrovsky, A., Borghoff, U. M., & Hofmann, M. (2017). Applying and augmenting deep reinforcement learning in serious games through interaction. *Periodica Polytechnica Electrical Engineering and Computer Science, 61*(2), 198. https://doi.org/10.3311/ppee.10313

Gutiérrez-Sánchez, P., Gómez-Martín, M. A., González-Calero, P. A., & Gómez-Martín, P. P. (2021). Reinforcement learning methods to evaluate the impact of AI changes in game design. *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, 17*(1), 10. https://doi.org/10.1609/aiide.v17i1.18885

Justesen, N., Torrado, R. R., Bontrager, P., Khalifa, A., Togelius, J., & Risi, S. (2018, January). Illuminating generalization in deep reinforcement learning through procedural level generation. *arXiv*. https://doi.org/10.48550/arXiv.1806.10729

Khalifa, A., Bontrager, P., Earle, S., & Togelius, J. (2020). PCGRL: Procedural content generation via reinforcement learning. *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, 16*(1), 95. https://doi.org/10.1609/aiide.v16i1.7416

Laskov, A., & Komura, T. (2009). *Level generation system for platform games based on a reinforcement learning approach*. [Unpublished manuscript].

Laukaitis, A., Šareiko, A., & Mažeika, D. (2025). Facilitating robot learning in virtual environments: A deep reinforcement learning framework. *Applied Sciences, 15*(9), 5016. https://doi.org/10.3390/app15095016

Lixandru, A. (2024, May). *Proximal policy optimization with adaptive exploration*. *arXiv*. https://doi.org/10.48550/arXiv.2405.04664

Pagalyte, E., Mancini, M., & Climent, L. (2020). *Go with the flow: Reinforcement learning in turn-based battle video games*. [Unpublished manuscript].

Pan, Z., & Liu, H. (2025). *MetaSpatial: Reinforcing 3D spatial reasoning in VLMs for the metaverse*. arXiv. https://doi.org/10.48550/arXiv.2503.18470

Politowski, C., Petrillo, F., El-Boussaidi, G., Ullmann, G. C., & Guéhéneuc, Y. (2023, May). Assessing video game balance using autonomous agents. *Proceedings of GAS 2023*. https://doi.org/10.1109/gas59301.2023.00011

Risi, S., & Togelius, J. (2020). Increasing generality in machine learning through procedural content generation. *Nature Machine Intelligence, 2*(8), 428. https://doi.org/10.1038/s42256-020-0208-z

Shu, T., Liu, J., & Yannakakis, G. N. (2021, August). Experience-driven PCG via reinforcement learning: A Super Mario Bros study. In *2021 IEEE Conference on Games (CoG)* (p. 1). https://doi.org/10.1109/CoG52621.2021.9619124

Shyalika, C., Silva, T., & Karunananda, A. S. (2020). Reinforcement learning in dynamic task scheduling: A review. *SN Computer Science, 1*(6). https://doi.org/10.1007/s42979-020-00326-5

Talebirad, Y., & Nadiri, A. (2023, January). Multi-agent collaboration: Harnessing the power of intelligent LLM agents. *arXiv*. https://doi.org/10.48550/arXiv.2306.03314

Yadava, V. (2023, April). *CartPole-v0-implementation*. GitHub. https://github.com/VinayYadava/CartPole-v0-Implementation

Zheng, T. (2024). Dynamic difficulty adjustment using deep reinforcement learning: A review. *Applied and Computational Engineering, 71*(1), 157. https://doi.org/10.54254/2755-2721/71/20241633