

Reinforcement Learning for Game Environment

REINFORCE
policy

Team GAIL RL-2025

Environment Overview

The **Environment** simulates a simplified obstacle-navigation task inspired by the Chrome dinosaur game. The agent *must jump over obstacles to maximize total reward while progressing forward*.

Key Features:

- **Agent:** Moves at a fixed speed along the x-axis.
- **Obstacles:** Appear at varying distances.
- **Actions:** jump over.
- **Termination:** Collision or goal completion.

Score: 0

A: Jump



Markov Decision Process (MDP)

State Space

A **4-dimensional** observation vector:

1. Normalized player height
2. Jump state (binary)
3. Normalized distance to next obstacle
4. Normalized distance to second obstacle

Action Space

Discrete choices:

- **0**: No jump
- **1**: Jump

Reward Function

- **+1** per step survived
- **+50** for passing an obstacle
- **-50** for collision
- **+100** for reaching the goal
- **-2** per jump (penalizing unnecessary jumps)

Terminal Conditions

- Collision with an obstacle
- Goal reached

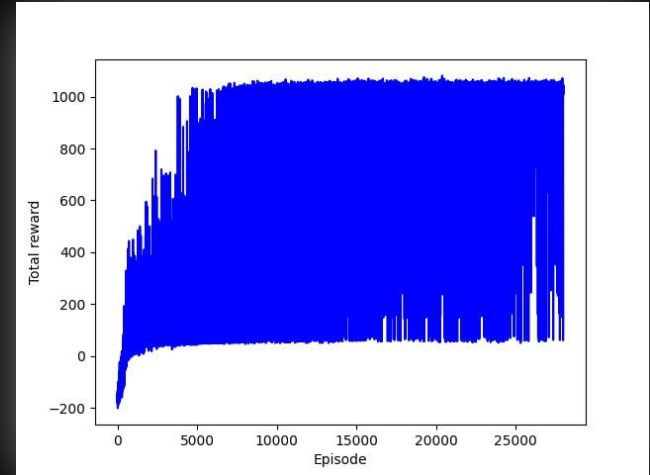
Training Performance

Observations:

- Reward fluctuations due to **random exploration**.
- Reaching near optimal policy after ~10,000 episodes

Hyperparameters Used

- **Discount Factor (Gamma, γ):** 0.99
- **Policy Net:** MLP with single hidden layer of 128 dim
- **Optimizer:** Adam with default parameters (learning rate = 0.0003)
- **Episodes:** 28,000 training episodes



Agent in action



01 ChatGPT

Game play logic code