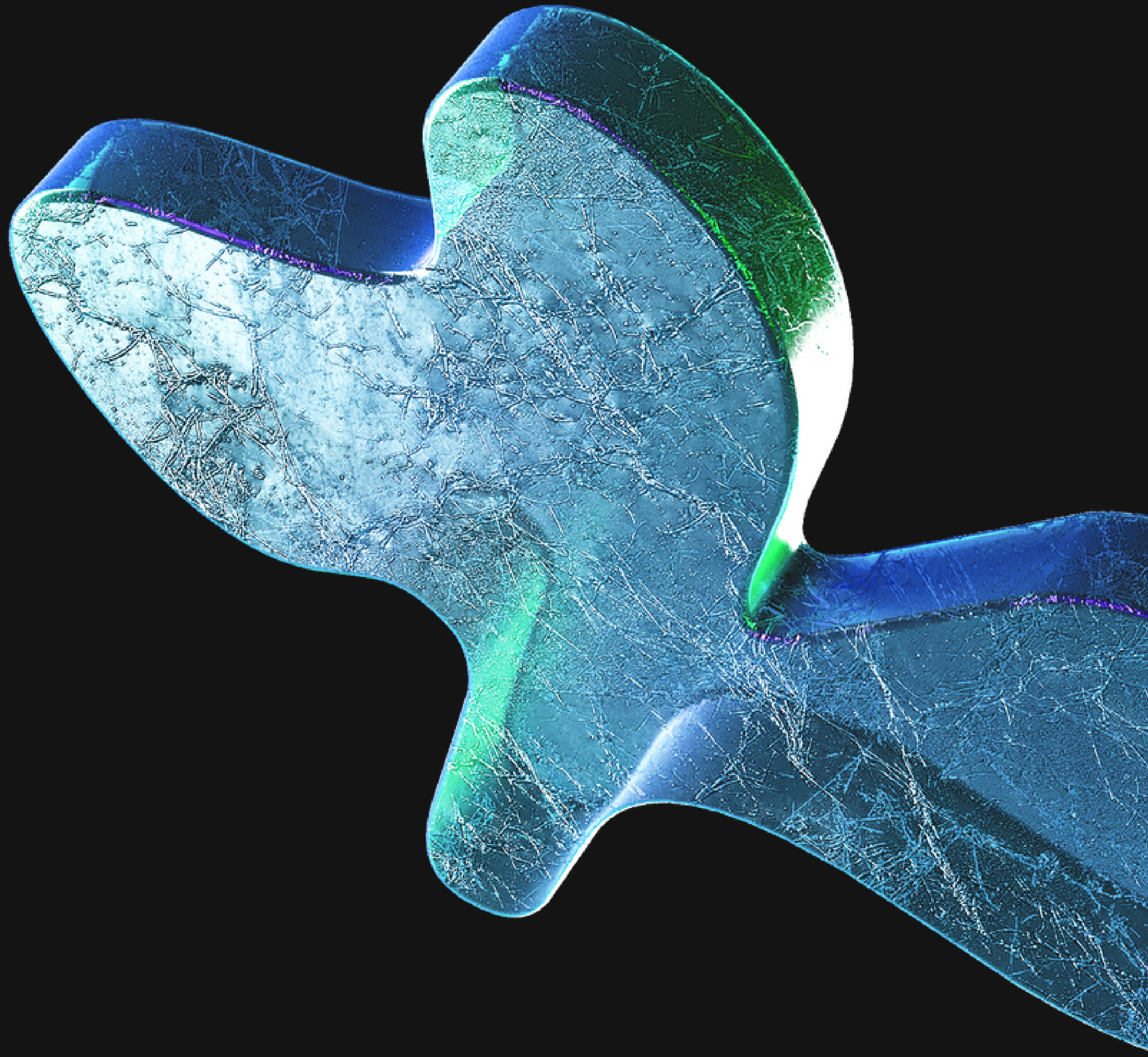
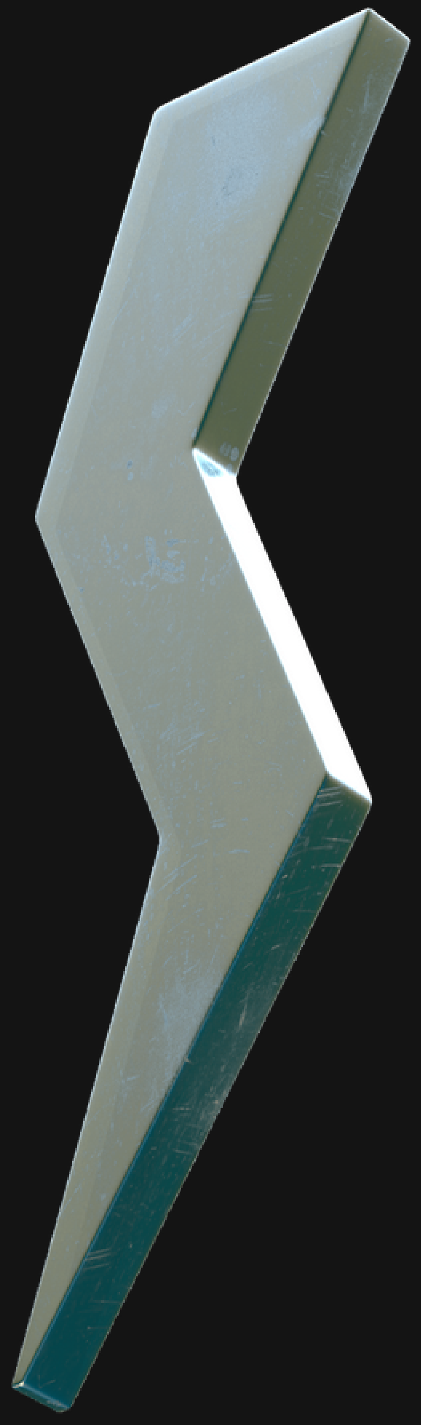


Reinforcement Drive

The Race For AI

Table of contents



Introduction

Tools and Technologies

Reinforcement Learning - PPO

Why not SAC?

Tuning

Results

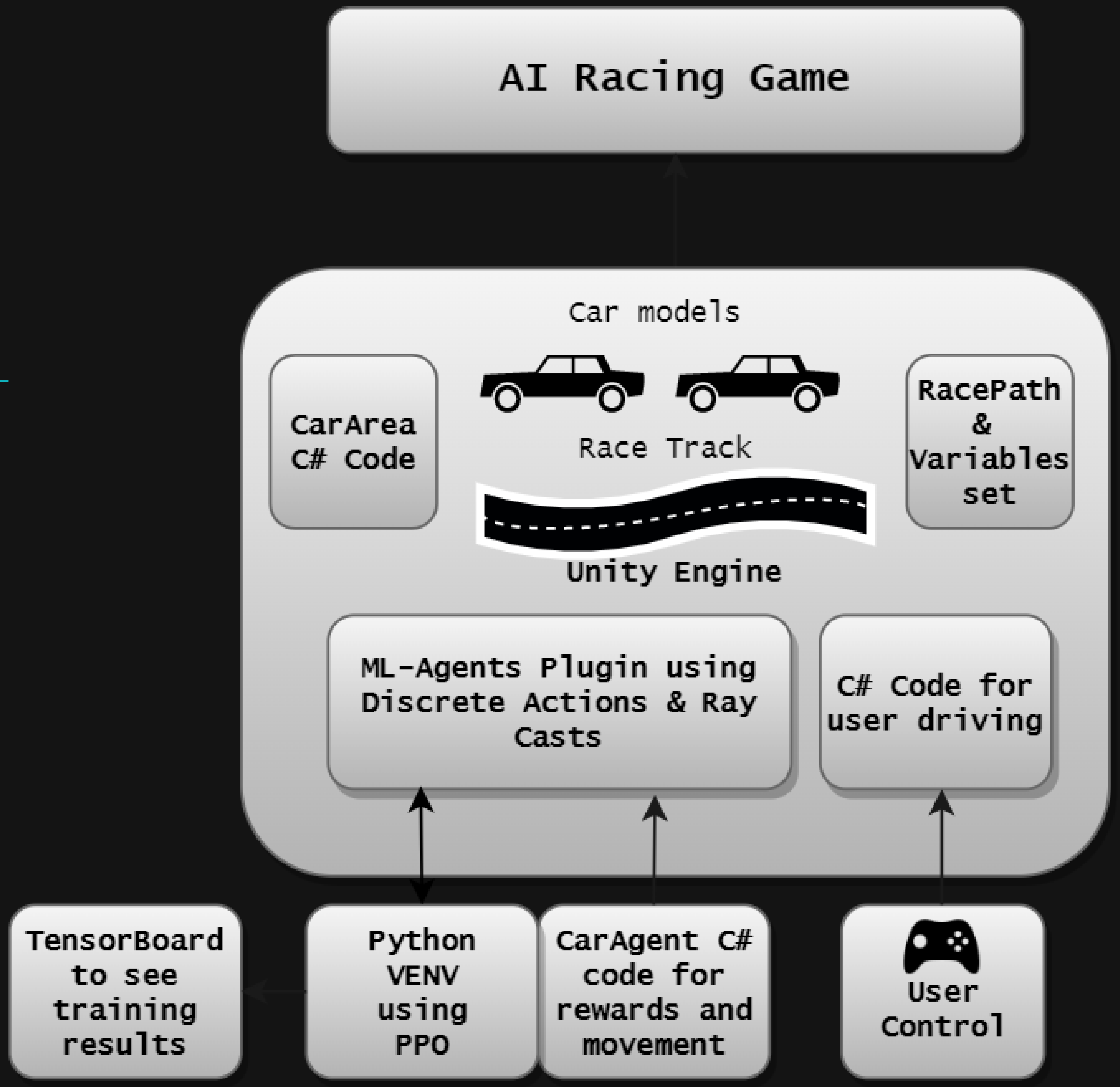
Conclusion

Introduction

- Goal - Create a racing game with AI-controlled opponents using reinforcement learning
- Develop an understanding of AI and reinforcement learning concepts
- Implement key game mechanics - AI behaviour, track creation, car physics

Technologies and Tools

- Unity game engine
- C# programming language
- ML-Agents plugin for Unity
- Proximal Policy Optimization (PPO)
- Python Virtual Environment



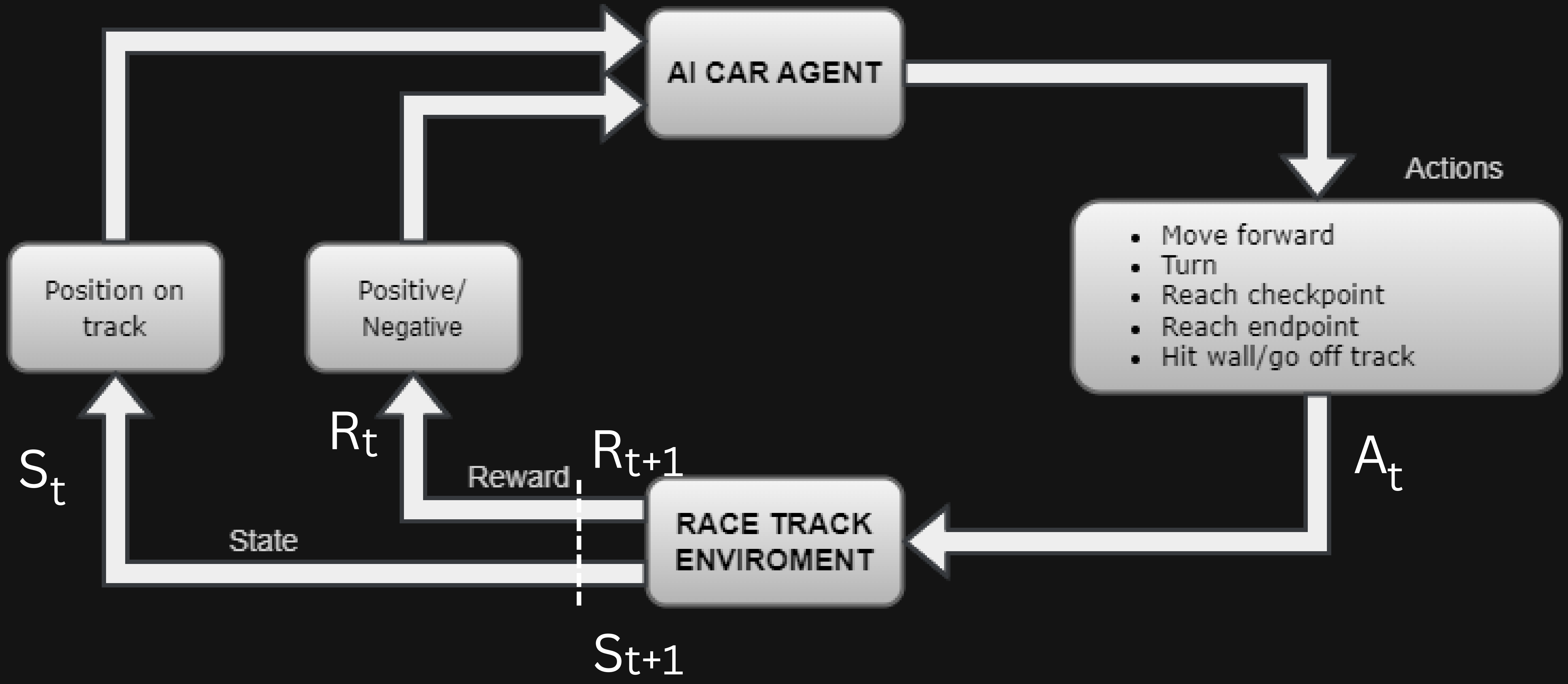
Reinforcement Learning - PPO

- Reinforcement learning: The agent learns by interacting with the environment
- PPO: An efficient, stable, and easy-to-implement algorithm
- On-Policy Algorithm works policy to policy
- Clipped surrogate objective function used to strike a balance between exploration and exploitation

Why not SAC?

- An off-policy algorithm learns from a variety of experiences
- Efficient exploration due to entropy maximization
- More difficult to implement than PPO
- Less computationally efficient than PPO due to replay buffer and dual networks

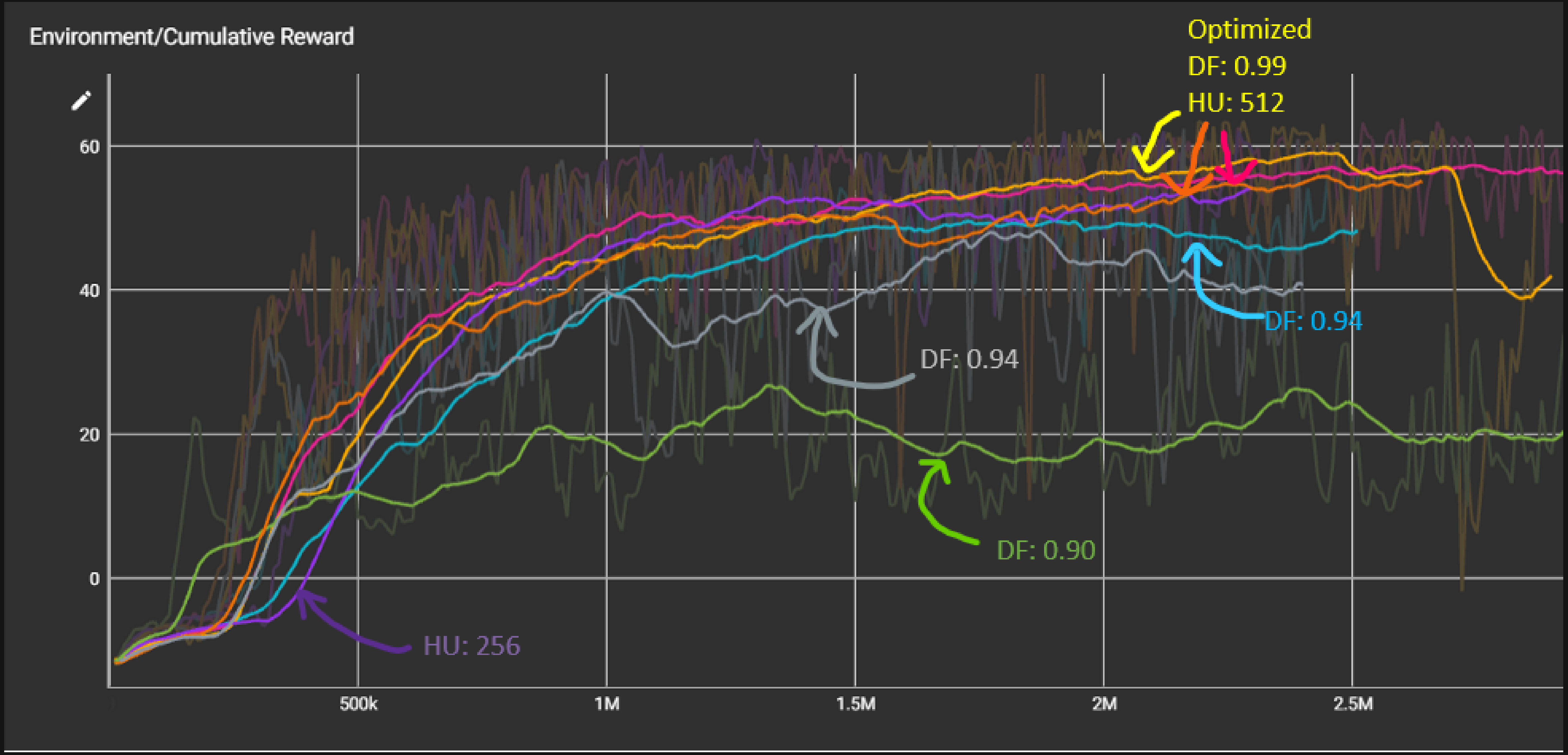
Project Architecture (RL MDP)



Tuning

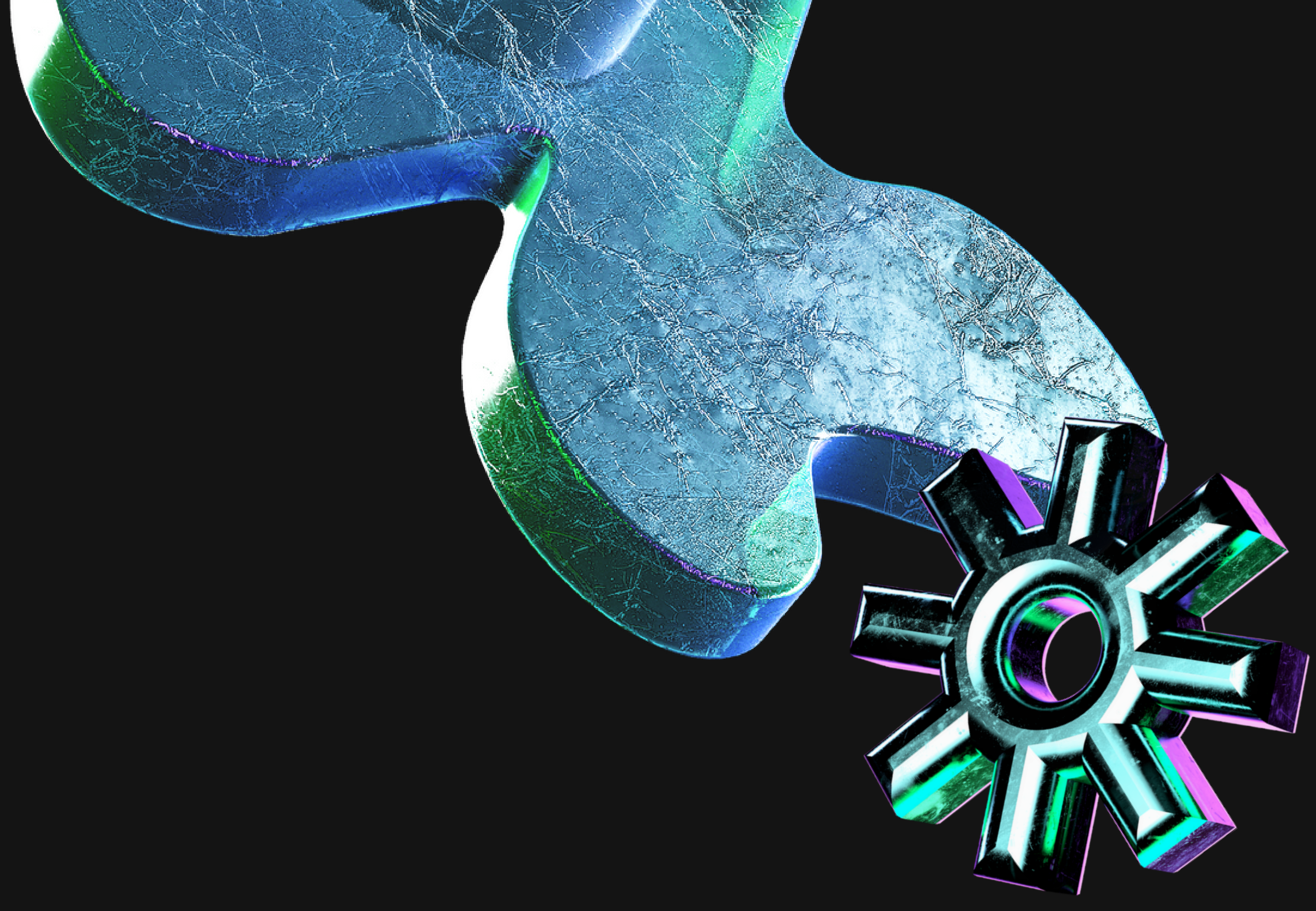
- Hyperparameters - Discount factor, Learning rate, Epsilon
- Reward scaling - Adjusting the magnitude of rewards to improve training stability
- Identifying bottlenecks - Addressing areas where agents struggle or fail
- Monitoring training progress - Tracking cumulative rewards, loss values, and episode lengths

Results


Environment/Cumulative Reward
60
40
20
0
500k
1M
1.5M
2M
2.5M
Optimized
DF: 0.99
HU: 512
DF: 0.94
DF: 0.94
DF: 0.90
HU: 256

Conclusion

- Hyperparameters - Discount factor, Learning rate, Epsilon
- Reward scaling - Adjusting the magnitude of rewards to improve training stability
- Identifying bottlenecks - Addressing areas where agents struggle or fail
- Monitoring training progress - Tracking cumulative rewards, loss values, and episode lengths



Do you have any questions?