REINFORCEMENT LEARNING FOR THE TRACKING OF UNMANNED AERIAL VEHICLES

Members:
Sun Zizhuo (Hwa Chong Institution)
Jaden Tjeng
(NUS High School of Mathematics and Science)

Mentor:
Jerry S/O Tamilchelvamani
(Defence Science and Technology Agency)

Introduction

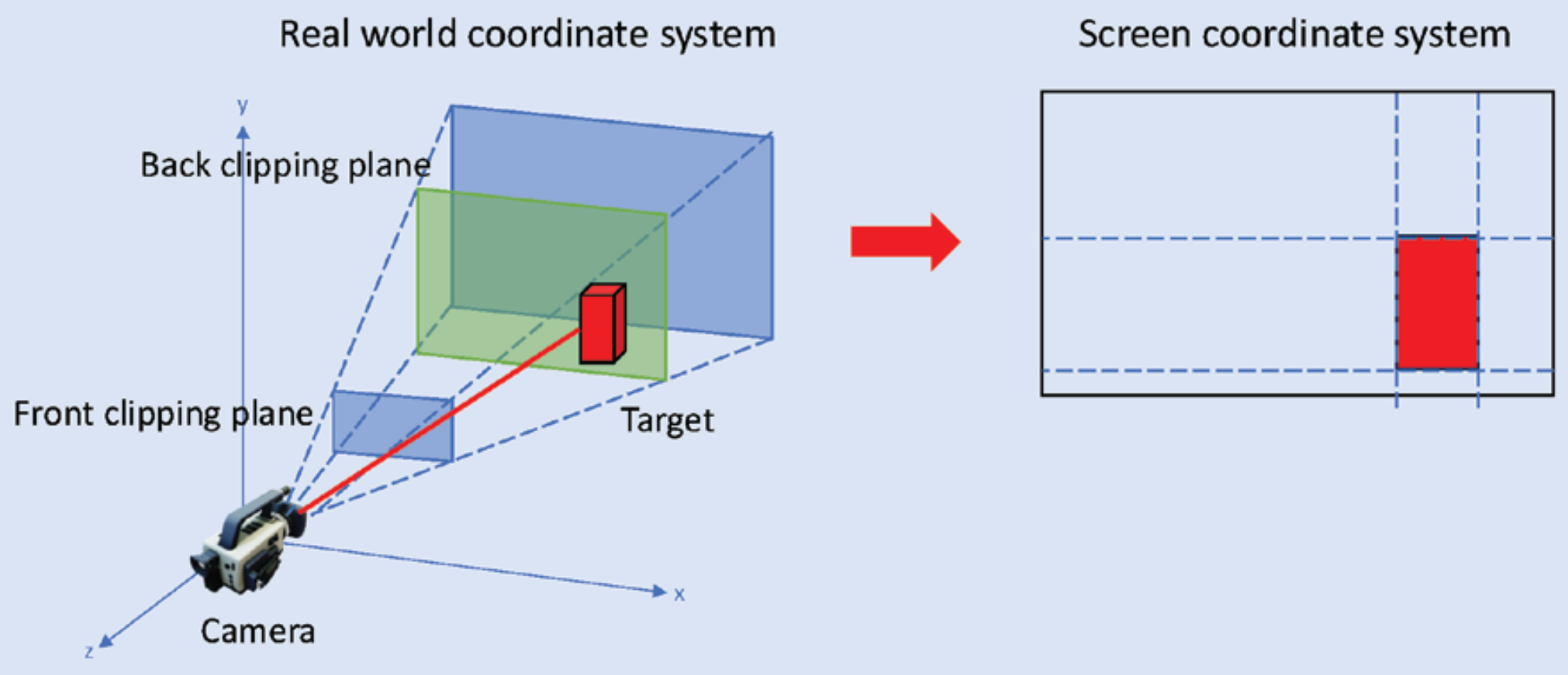
The increasing prevalence of Unmanned Aerial Vehicles (UAVs) raises concerns about unauthorised operations, privacy violations, and threats to airspace security due to its misuse. Hence, the ability to effectively track, monitor and take down such drones is crucial for mitigating these risks.

Traditionally, Proportional-Integral-Derivative (PID) controllers are used for this purpose, but controllers trained with Proximal Policy Optimisation (PPO)-based reinforcement learning also show promising potential. This study compares the performance of a PID controller and a PPO controller in tracking drone targets through simulations conducted in Unity.

Materials and Methods

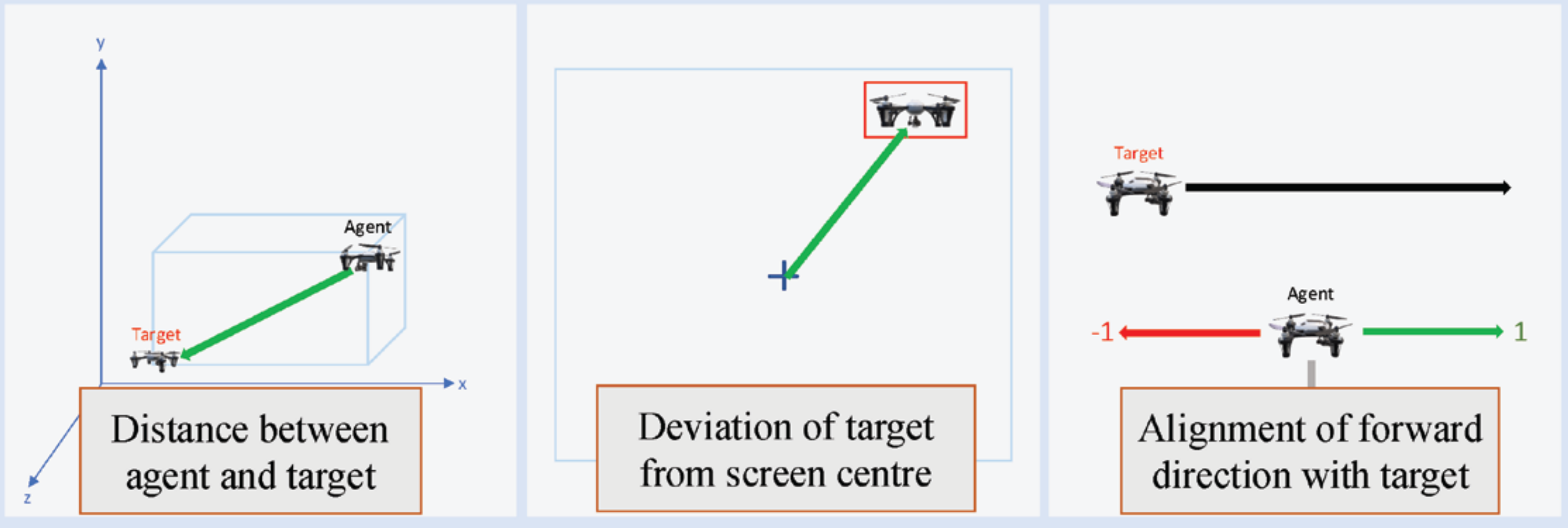
Homogenous Projection

To simulate real-life inputs as closely as possible, it is not realistic to pass in the absolute 3D coordinates of the target. As such, 2D screen coordinates *(x', y')* as well as screen-space width and height *w',h'* had to be obtained from a 3D point, *(x, y, z)* to be passed in as inputs



Proximal Policy Optimisation

Reinforcement learning was conducted using Unity's ML-Agents with Proximal Policy Optimisation (PPO). The agent was given the inputs of position and forward direction of its own entity and screen-space coordinates of the target and obstacles. Through creating an appropriate reward function, the agent will converge to an optimal policy to achieve the task of tracking the target.



Distance between agent and target

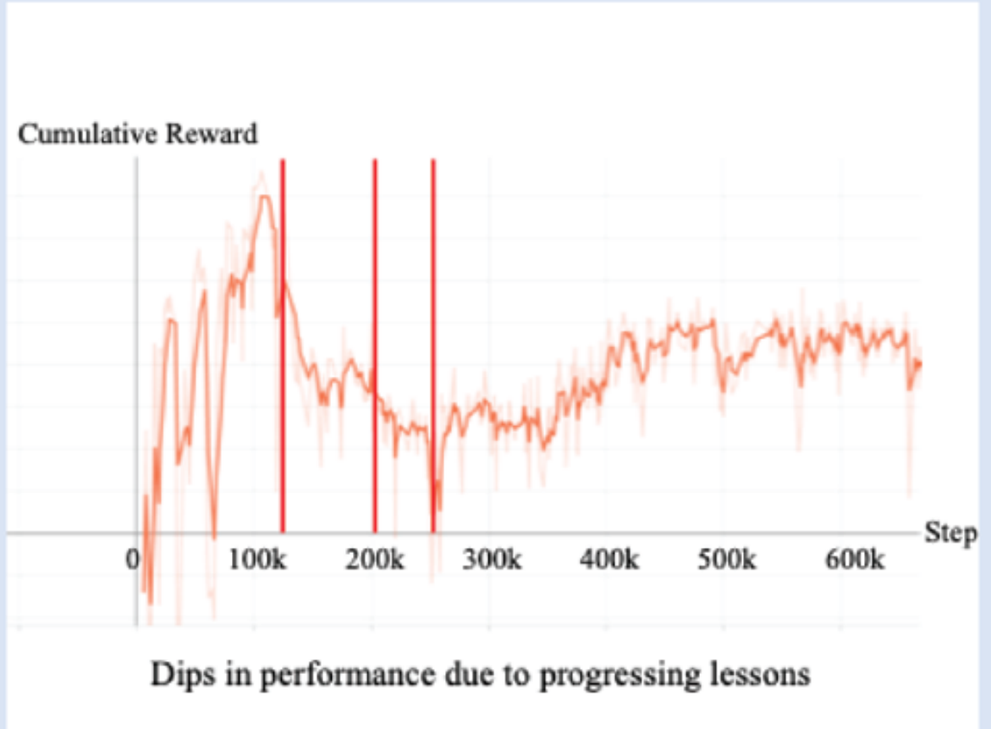
Deviation of target from screen centre

Alignment of forward direction with target

An episode was terminated whenever the agent was more than 60 metres away from the target or could not view the target for more than 5 seconds.

Curriculum Learning

The curriculum that was devised gradually incremented the target's speed and maximum speed variation. The agent was also subjected to increasingly difficult target movements.



Dips in performance due to progressing lessons

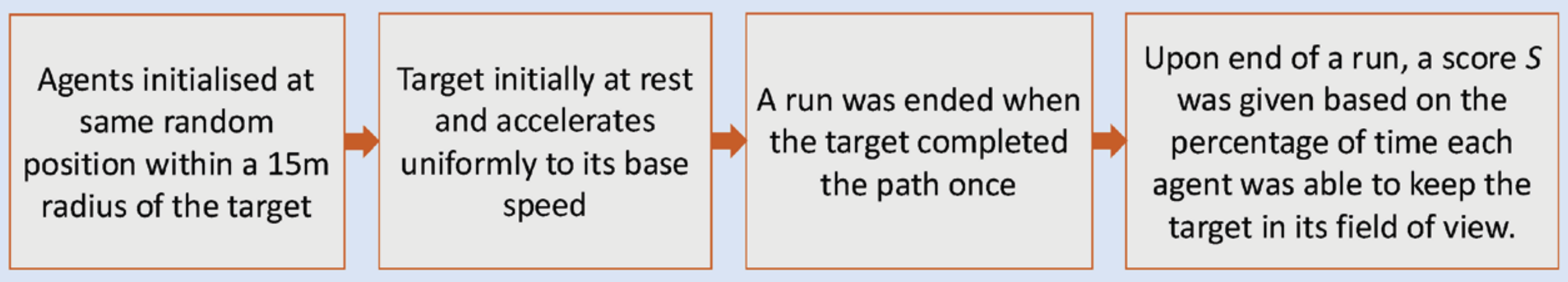
Dips in cumulative reward occurred whenever training parameters were adjusted during curriculum learning. Temporary performance drops, indicated by dips in cumulative reward, reflects the agent's adaptation to increasing task complexity.

Evaluation Criterion

The engagement success of the PID controller and the PPO controller were evaluated based on a score *S,* which indicates the percentage of time each controller could keep the target within its field of view over a period *T*. This is expressed as

$$S = \frac{1}{T}\sum_{t=1}^{T} I_t$$

The following procedure was used for the collection of the evaluation criterion:



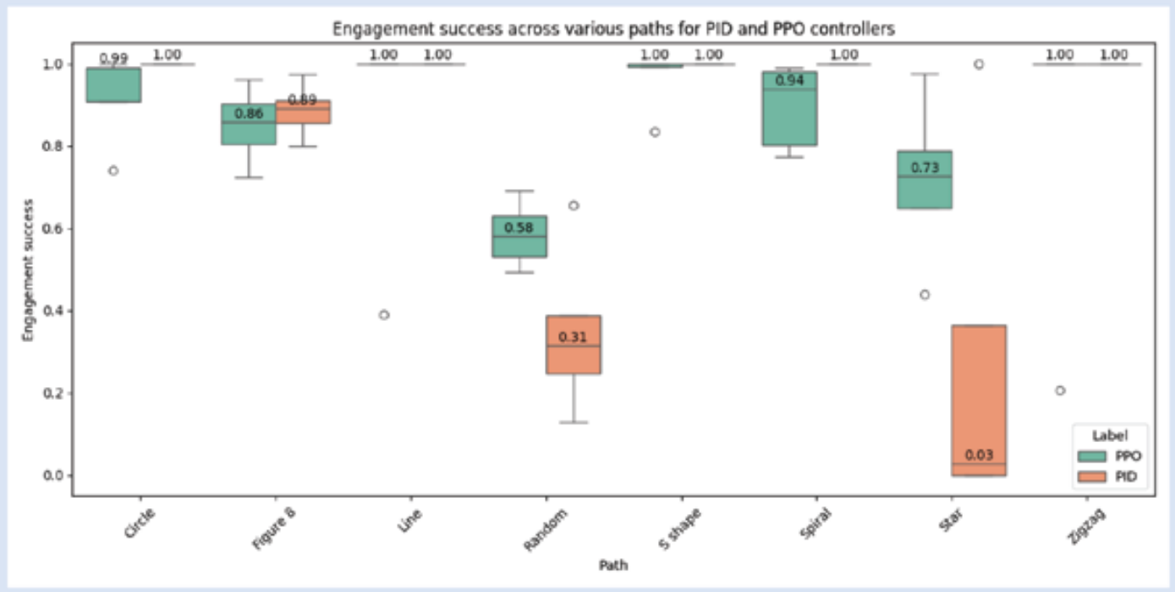
Agents initialised at same random position within a 15m radius of the target

Target initially at rest and accelerates uniformly to its base speed

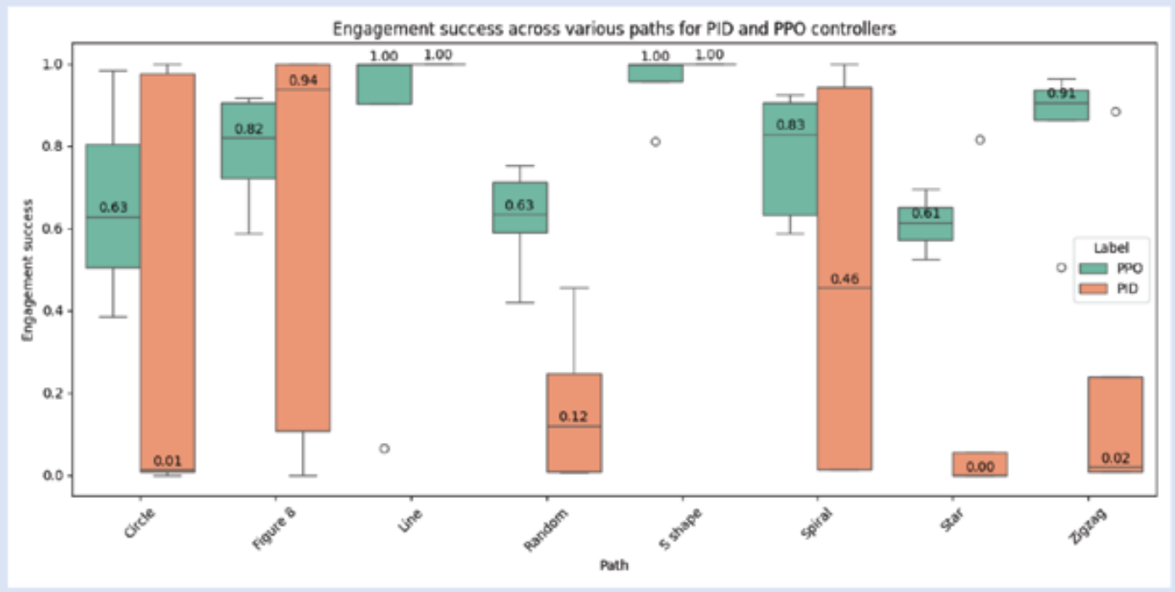
A run was ended when the target completed the path once

Upon end of a run, a score *S* was given based on the percentage of time each agent was able to keep the target in its field of view.

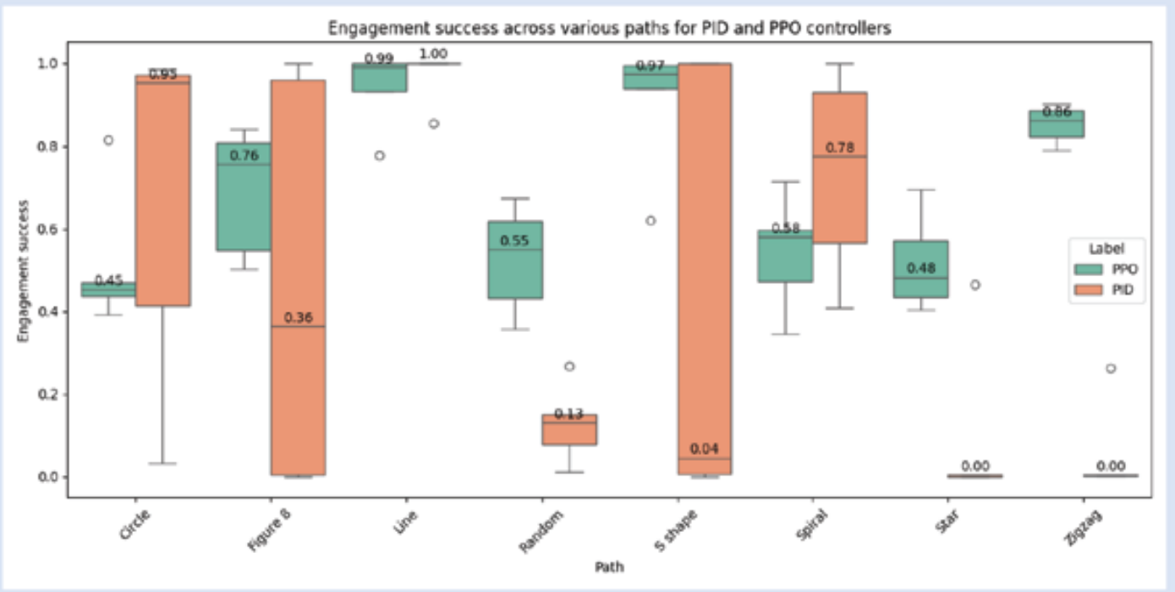
Results



50m x 50m bounding area



100m x 100m bounding area



200m x 200m bounding area

Observation 1:
The results from the performance evaluation revealed that the PPO controller outperformed the PID controller in most cases especially in the Star, Random and Zigzag paths.

Observation 2:
However, there were some cases where the PID controller exhibited better engagement success than the PPO controller such as the Figure 8 path on the 50m x 50m and 100m x 100m bounding area.

Observation 3:
The PID controller had less reliable performance for larger bounding areas while the PPO controller was relatively reliable regardless of bounding area. This was evident from the significantly larger interquartile ranges for engagement success of the PID controller across most paths for the 100m x 100m and 200m x 200m bounding areas while the interquartile range was similar regardless of bounding area for the PPO controller.

Findings

The observed trends could be attributed to the fundamental differences between the two controllers. Through many iterations of training, the PPO controller could anticipate and react to the sharp changes in direction of the target, which the deterministic PID controller was unable to do.

Moreover, the greater reliability of the PPO controller observed from the results could be due to different levels of sensitivity to the initial conditions of the agents. During the experiment runs, it was observed that the PID controller was more sensitive to its starting position. If it was initialised closer to the target, it was more likely to lose track of the target as it was unable to quickly adjust to the target's movements.

Conclusion

This study compares the performance of a PID controller and a PPO controller in tracking drone targets through simulations conducted in Unity. The results revealed that the PPO controller was generally better than the PID controller in maintaining the target within its field of view and tracking it, especially in unpredictable conditions and sharp turns. The PPO controller also exhibited greater reliability, achieving less variability in engagement success over multiple simulation runs.