Yash Vinesh Gadhia
180100130
IIT Bombay

The Learning to Fly Challenge

Abstract

Controlling high-dimensional autonomous systems is a challenging task. One such system is that of Unmanned Aerial Vehicles (UAVs). In this particular challenge, we aim to solve the heading control task for aircrafts. I propose a method that aims to solve this task using a reinforcement learning approach, more specifically using the PPO algorithm. It also involves designing a reward function to specifically target the various termination conditions of the task, to enable the agent to learn better.

Introduction

Autonomous control of aircraft is a challenging high-dimensional continuous control problem with applications in Unmanned Aerial Vehicles (UAVs), autopilot systems and flight simulators. A moving aircraft has a large state space comprising translation and rotation in three dimensions, and a continuous action space comprising inputs to control surfaces and the aircraft's propulsion. Conventional autopilot systems are commonly implemented through nested proportional-integral-derivative (PID) controllers, which require parameter tuning and are vulnerable to control instability in perturbed flight conditions. This problem domain is well suited for reinforcement learning which enables agents to learn from interactions with an environment. However, previous works applying reinforcement learning to aircraft control have been limited to methods simplifying the problem by discretizing the action space and reducing the number of dimensions in the state space. Instead, here the attempt is to achieve full six degrees of aircraft control with continuous actions and move towards a new generation of self-flying machines that operate safely and robustly.

Methodology

Given that the flight conditions are provided as a gym environment, I tried to train various RL agents using the default state features and reward functions through stablebaselines3. Upon experimentation, I observed that TD3 and DDPG were performing poorly while PPO showed

some promising results. But on printing the delta heading and altitude, I observed them growing unboundedly. I thus decided to modify the reward function to enable better training.

Looking at the existing reward function which is the geometric mean of various error terms, we see that it would be very difficult for the network to differentiate between the different error terms and optimize the ones that were performing poorly. Furthermore, not all the error terms are equally important as far as the termination conditions are concerned. I have hence written wrappers allowing for modified reward functions.

I first set the reward function to penalize for any heading deviation beyond 6deg (as 10deg is the cut-off). Less than 6deg gets a maximum reward of 360, while any further deviation obtained a quadratic negative reward. This provided a decent increase in the reward. Observing the heading deviations during the runs, it was obvious that they were well within 10deg.

Next up was a constraint on the roll. While this is not part of the termination condition, it would be required. I tried two versions - a linear constraint and an exponential one. Both of these provide a maximum reward of 360 (to keep it in the same range as the heading constraint) while providing a minimum of -400. This is to make sure extreme roll conditions are observed by the reward function and not hidden by good heading control. We observe that the exponential roll term led to the highest reward.

Experimenting with other terms did not affect the results substantially, and hence I have not included those results here. Note that I observed altitude control was performed automatically, as is observed in the comments in the code of Gym-JSBSim.

Results

Model	Episode Length	Reward
Default reward	1198	55.26
Heading	598	59.88
Heading + Roll Linear	1799	76.84
Heading + Roll Exp	823	88.40

Benefits of the chosen methodology

PPO algorithm has been shown to work well for continuous action spaces in the past and has also been used for several control system tasks. This along with carefully designing the reward function to specifically target the various termination conditions seems to be the best way forward to solve the heading control task.

Conclusion

Thus we have obtained a model that runs well on the heading control task, and only terminates due to not meeting the acceleration bounds. And this is exactly how I plan to extend the reward function in the future. Experiments with an L2 norm of acceleration did not achieve any improvement in results. I will hence model the acceleration reward term on other functions like exponential or piecewise.

References

1. Antonin Raffin, Ashley Hill, Adam Gleave, Anssi Kanervisto, Maximilian Ernestus, and Noah Dormann. Stable-baselines3: Reliable reinforcement learning implementations. Journal of Machine Learning Research, 22(268):1–8, 2021

2. John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. CoRR, abs/1707.06347, 2017