Nik Szczepanski

Existing autonomous control systems for remote-controlled fixed-wing aircraft fall short when it comes to navigating complex and dynamic environments. Although these systems do very well in open spaces or predictable conditions, their ability to process real-time data, adapt to rapid changes, and avoid obstacles is limited. This restricts their use in challenging terrains like forests, lest they be downed by a wayward gust of wind. These shortcomings not only impede their use in applications such as search-and-rescue missions, wildlife monitoring, and environmental research but also underscore a need for integrating advanced perception and control algorithms into lightweight, fixed-wing platforms.

Fixed-wing craft are much more power efficient then multirotor aircraft, which makes them ideal for long distance flights where it is essential to conserve energy. However, their reduced maneuverability compared to multirotor designs has limited their ability to navigate complex and dynamic environments. To address this limitation, I aim to develop an advanced algorithm capable of real-time obstacle avoidance and dynamic compensation for variables such as wind conditions, altitude changes, and terrain contours. This system will allow fixed-wing aircraft to safely navigate complex environments, thus broadening their potential for use in applications that require finesse and agility.

My plan is to create a model of an airplane in XFLR5, refine it, transfer it to Autodesk Fusion, and use OpenFOAM to acquire my aerodynamic data. In addition to developing the physical model, I will implement a surrogate algorithm to simulate aerodynamics and I will integrate reinforcement learning techniques. Specifically, I plan to use hierarchical reinforcement learning (HRL), with soft actor-critic (SAC) for overall control and twin delayed deep deterministic policy gradient (TD3) for lower-level tasks like engine control and adjustments to the control surfaces. To ensure that the model can balance multiple objectives simultaneously, I will employ multi-objective reinforcement learning (MORL). Furthermore, I will integrate an extended Kalman filter (EKF) SLAM algorithm to enable mapping and localization in unfamiliar environments.

I will then create the environment in Unity, create a surrogate algorithm to simulate aerodynamics and implement HRL, using SAC for overall control and TD3 for lower-level tasks, such as engine control and . I will use MORL so that the model balances multiple objectives, and implement EKF SLAM for mapping. I have modeled and tested my airfoils using XFOIL and created a model of the plane using XFLR5's tools for 3D design. Using XFLR5, I ran some aerodynamic simulations to see how my design performed, and I revised my model based on the data from the simulations. I repeated this process until the design met my performance goals. Since XFLR5 does not have the ability to export meshes natively, so I had to re-model the plane in CAD. I chose Fusion because it is the software I have the most experience in. I exported the airfoils and imported them into Fusion using an extension that can convert the coordinates in an airfoil .DAT file to a spline. I used these profiles to recreate the wings, the vertical and horizontal stabilizers, and their respective control surfaces (flaps, ailerons, elevator, and rudder). Once I completed the model, I created meshes of the components. I am currently teaching myself how to use OpenFOAM so that I can generate the data I need.

Sources:

https://v0xnihili.github.io/xfoil-docs/
https://www.youtube.com/@techwinder/videos
https://apps.autodesk.com/FUSION/en/Detail/Index?id=3044478757760121899&os=Win64&appLang=en
https://arxiv.org/pdf/1801.01290v2
https://arxiv.org/pdf/1802.09477v3
http://ais.informatik.uni-freiburg.de/teaching/ws12/mapping/pdf/slam04-ekf-slam.pdf