 © Zipline 2024

Context:

Zipline has unveiled its P2 system, one of the components of which is the “Zip” - a vehicle which can fly like a quadcopter and an airplane. One of the special features is the rear propeller which can provide forward thrust when in airplane-mode.

One of the challenges with flying like this is knowing what rotation speed GNC should command of this propeller in order to maintain the desired Zip flight speed, given that the dynamics of the system are always changing with real-world effects like varying angles of attack and wind.

Attached is some test flight data with IMU derived measurements, the measured propeller speed, and a lift/drag table for the overall vehicle at various angles of attack. The data is for a simplified 2d vehicle model, which is adequate for this task. Assume we are flying as a standard fixed-wing aircraft rather than VTOL.

Task:

From this, show what onboard algorithm you would use to determine what propeller speed is required to maintain a constant airspeed for a given state (such as angle of attack, speed, or whatever you decide is required). Show how well this predictor compares to the flight data.

State any assumptions.

Deliverables:

- An onboard algorithm which takes vehicle state as input and outputs desired propeller speed.
- Plots to show the accuracy (or otherwise) of the developed algorithm.
- All python or Julia code written to process and generate all tables / values used in the onboard algorithm, and additionally to generate any comparison / validation code.

Limitations:

- You must use **python** or **julia**.
- You may not use LLMs to directly generate code other than trivial parts (plotting, data loading)..
- You may use common libraries, matplotlib, scipy, etc.

Attached:

Attached should be a csv-like file, it contains:

- A lift-to-drag table across a range of angles of attack. This represents the estimated lift force relative to drag force, for a given AoA.
- Multiple snippets of flight of a vehicle with 5Hz time series data.

These definitions apply:
- drag = 0.5*rho*v^2*Cd*A (applied in vehicle frame)
- lift = drag * lookup_coeff (applied in vehicle frame)
- Pitch angle is above the horizon

Hints:

- Act as if the algorithm you devise is going to be flying on an aircraft: consider robustness and reliability when designing and coding.
- We will be reviewing your submission in the same way we review actual work; documentation and/or comments are suggested.
- People who replace their brains with LLMs typically do poorly.
- Be prepared to discuss your approach, problems you faced along the way, etc. at your interview.
- This task has been deliberately left as ambiguous. For us reviewing your submission, seeing your thoughts and understanding is as important as getting a 'correct' result:
    - If you try multiple approaches, tell us about them.
    - Don't spend more time than necessary developing a super high fidelity model - instead tell us known shortcomings / future steps.
    - If you are missing information (e.g. density), make a sensible assumption and tell us about it.
    - Have fun!