Combining Flight Physics and Data to Predict Propeller Speed for Zipline's Hybrid Aircraft

Scott Nguyen*

This paper presents the development of a lightweight onboard algorithm to estimate the propeller speed (RPM) required for steady-level flight of Zipline's P2 hybrid aircraft in fixed-wing mode. The algorithm relies solely on onboard sensor inputs—including pitch angle, body-frame velocity, and estimated wind components—to compute airspeed and angle of attack. Thrust is modeled as a quadratic function of RPM, scaled by a learned efficiency term approximated by a fourth-order polynomial in AoA and airspeed. While the model captures key trends and operates reliably across most flight conditions, some deviations from truth data are observed, particularly in out-of-distribution cases. Nonetheless, the algorithm remains interpretable, computationally efficient, and suitable for onboard use where robustness and simplicity are prioritized over perfect accuracy.

Nomenclature

A	=	Reference wing area, m^2
C_D	=	Drag coefficient
C_L	=	Lift coefficient
$\frac{C_L}{C_D}$	=	Lift-to-drag ratio
f	=	Thrust-to-RPM^2 scaling factor
L	=	Lift force, N
D	=	Drag force, N
T	=	Thrust force, N
v	=	Airspeed magnitude, m/s
v_x, v_y	=	Body-frame velocity components, m/s
w_x, w_y	=	Wind velocity components, m/s
$\vec{v}_{\text{body}}$	=	Body-frame velocity vector, m/s
$\vec{v}_{\text{wind}}$	=	Wind velocity vector, m/s
$\vec{v}_{\text{air}}$	=	Air-relative velocity vector, m/s
ρ	=	Air density, kg/m^3
θ	=	Pitch angle, rad
γ	=	Flight path angle, rad
AoA	=	Angle of attack, rad
RPM	=	Propeller rotation speed, rev/min
x	=	Generic variable (e.g., AoA or airspeed) for polynomial fitting
$\hat{x}$	=	Standardized input variable
%Error	=	Percent RPM prediction error
R^2	=	Coefficient of determination (regression fit quality)

*

I. Introduction

Zipline's P2 platform features a hybrid vehicle, the "Zip," which transitions between quadcopter and fixed-wing flight. In airplane mode, a rear-facing propeller provides forward thrust. One challenge in this regime is determining the correct propeller speed (RPM) to maintain constant airspeed, despite real-world variations in wind, pitch, and angle of attack (AoA).

To address this, we were provided with 2D flight test data, including IMU-derived measurements, estimated wind, propeller speed, and a lookup table for lift-to-drag (L/D) ratios over a range of AoAs. The objective is to design a robust, onboard algorithm that maps observable states—pitch angle, body-frame velocity, and wind estimates—to the RPM required for trimmed, steady-level flight.

This report outlines the development and validation of such an algorithm, prioritizing simplicity, interpretability, and real-time performance over high-fidelity complexity. The final solution is intended for reliable onboard use within Zipline's GNC architecture.

II. Problem Statement

The goal is to develop a lightweight onboard algorithm that predicts the required propeller speed (RPM) to maintain constant airspeed during fixed-wing flight of Zipline's P2 vehicle. The algorithm must rely solely on sensor-accessible inputs—pitch angle, body-frame velocity, and estimated wind—and operate in real time.

It must account for changes in flight conditions, including variations in wind and angle of attack, using a simplified 2D steady-flight model. The predicted RPM should closely match observed values from flight test data, ensuring robustness and reliability for GNC integration.

III. Background

To maintain constant airspeed in fixed-wing flight, the Zipline P2 must satisfy steady, trimmed conditions where net external forces are zero:

$$T = D, \quad L = W \tag{1}$$

Here, T is thrust, D is drag, L is lift, and W is the vehicle's weight.

Lift and drag are modeled as:

$$D = \frac{1}{2}\rho v^2 C_D A, \quad L = \frac{1}{2}\rho v^2 C_L A \tag{2}$$

with ρ as air density, v as airspeed, C_D and C_L as aerodynamic coefficients, and A as reference area.

Due to limited propeller performance data, thrust is approximated using a quadratic relationship:

$$T = f \cdot \mathrm{RPM}^2 \tag{3}$$

The efficiency term f is statistically fit from flight data as a function of airspeed and angle of attack.

All required inputs—pitch angle, body-frame velocities, and estimated wind—are accessible onboard. From these, airspeed and AoA are derived to define the aerodynamic state. The algorithm then estimates the RPM required to maintain force balance across different flight conditions.

IV. Methods

This section outlines the development of an onboard RPM estimation algorithm, designed for real-time use with sensor-accessible inputs and implemented in Python.

A. State Variable Selection

The algorithm uses five onboard-measurable inputs:

- Pitch angle θ
- Body-frame velocities v_x, v_y
- Estimated wind components w_x, w_y

From these, we compute:

$$\vec{v}_{\text{air}} = \begin{bmatrix} v_x \\ v_y \end{bmatrix} - \begin{bmatrix} w_x \\ w_y \end{bmatrix}, \quad v = \|\vec{v}_{\text{air}}\|, \quad \gamma = \arctan 2(v_y - w_y, v_x - w_x), \quad \text{AoA} = \theta - \gamma \tag{4}$$

B. Trimmed Flight and Drag Modeling

Assuming steady, level flight:

$$T = D, \qquad L = W \tag{5}$$

Drag is modeled as:

$$D = \frac{1}{2}\rho v^2 C_D A \tag{6}$$

with:

- $\rho = 1.225\,\mathrm{kg/m^3}$ (air density)
- $C_D = 0.13$ (drag coefficient)
- $A = 0.625\,\mathrm{m^2}$ (reference area)

C. Thrust-RPM Relationship

Thrust is approximated as a quadratic function of RPM:

$$T = f \cdot \text{RPM}^2 \quad \Rightarrow \quad \text{RPM} = \sqrt{D/f} \tag{7}$$

The thrust coefficient f varies with AoA and airspeed.

D. Fitting the Thrust Coefficient Surface

To capture the dependence of f on aerodynamic state, we fit a 4th-order bivariate polynomial surface using least-squares regression. AoA and airspeed are standardized before fitting to ensure numerical stability. This statistical model captures nonlinear effects while remaining computationally efficient for onboard use.

V. Results

A. Preliminary Analysis

1. State Variable Distributions

Figure 1 shows the distribution of six key variables: body-frame velocities (v_x, v_y), wind components (w_x, w_y), pitch angle, and angle of attack (AoA). Most are unimodal and bounded, with pitch centered near 0°, consistent with trimmed flight. AoA is bimodal, indicating transitions between flight regimes. Wind shows the most variability.

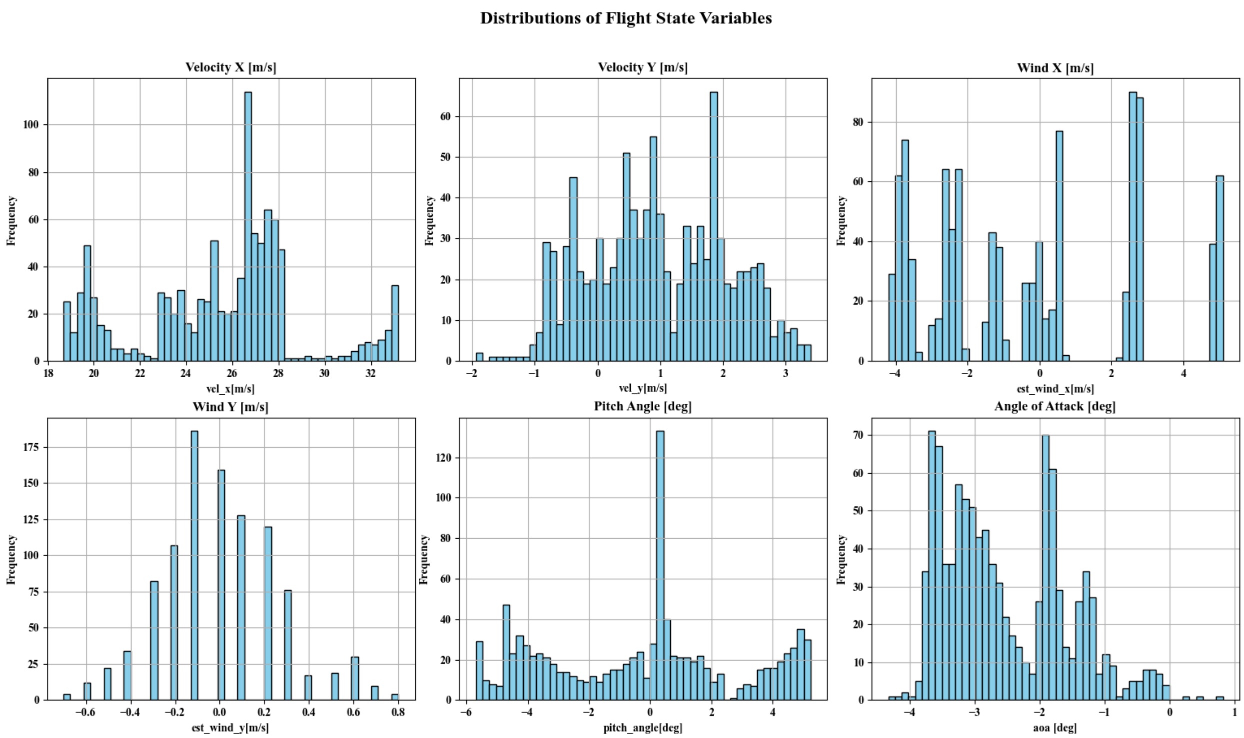

Fig. 1 Distributions of key flight state variables.

2. Lift and Drag Characterization

A 10$^{\text{th}}$-order polynomial fit to the C_L/C_D vs. AoA curve enables estimation of lift coefficients assuming $C_D = 0.13$:

$$C_L = C_D \cdot \left(\frac{C_L}{C_D}\right)$$

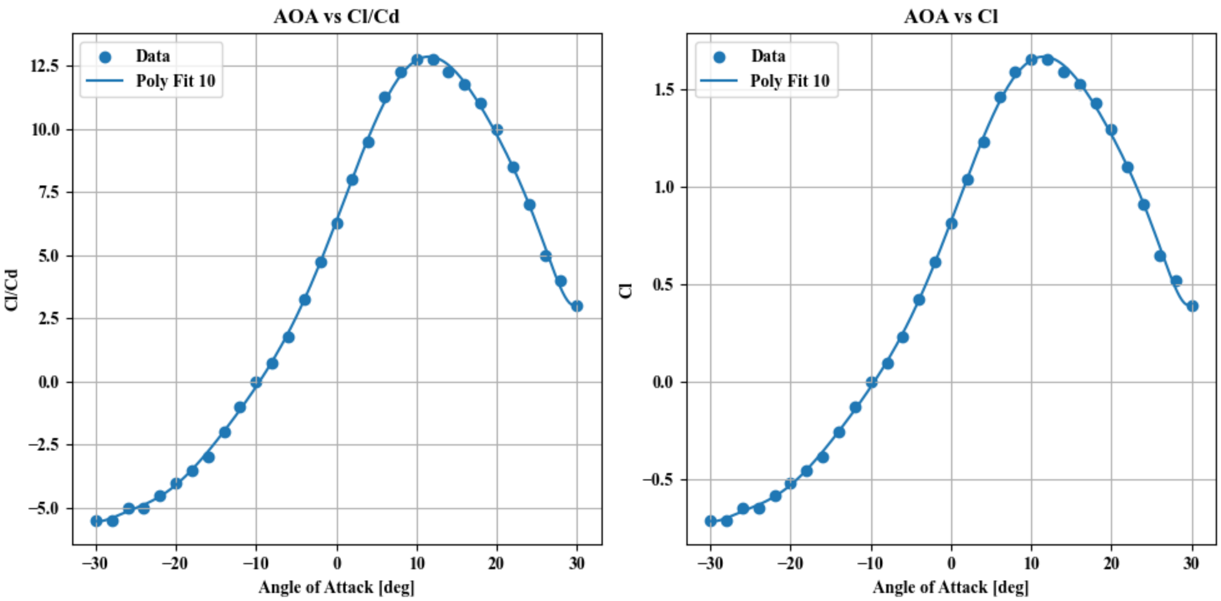

Fig. 2 Polynomial fit for C_L/C_D and C_L vs. AoA.

Table 1 10th-Order Polynomial Coefficients for C_L/C_D vs. AoA

Term	x^0	x^1	x^2	x^3	x^4	x^5	x^6	x^7	x^8	x^9	x^{10}
Coeff	6.334e+00	8.368e-01	7.595e-03	-2.275e-03	-1.086e-04	4.097e-06	2.691e-07	-4.476e-09	-2.993e-10	1.979e-12	1.249e-13

3. Initial RPM Modeling from Lift and Drag

We first attempted to model RPM as a function of lift and drag. While some structure was visible (Fig. 3), the relationship lacked the robustness needed for onboard prediction.

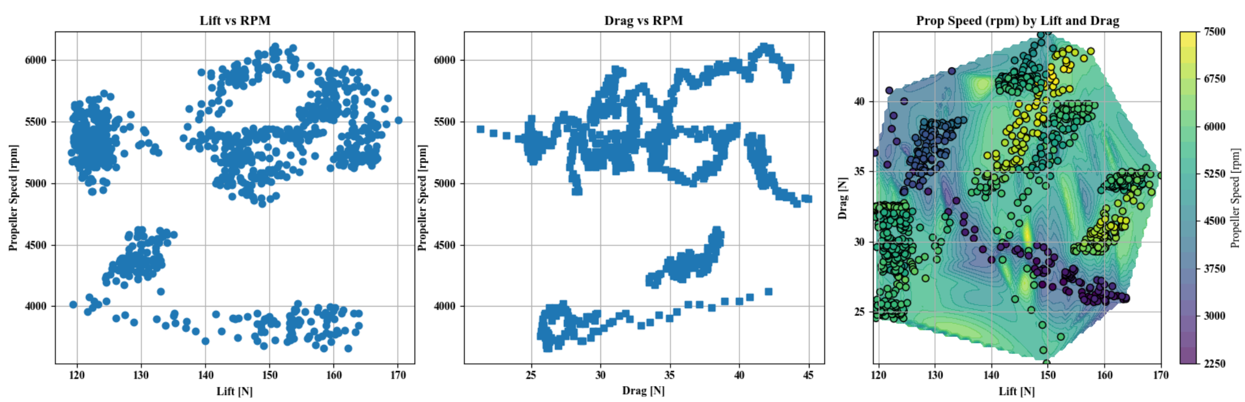

Fig. 3 Contour plot of RPM vs. lift and drag.

4. AoA and Airspeed-Based Modeling

Switching to AoA and airspeed revealed a much smoother, more structured mapping (Fig. 4), motivating a polynomial regression approach.

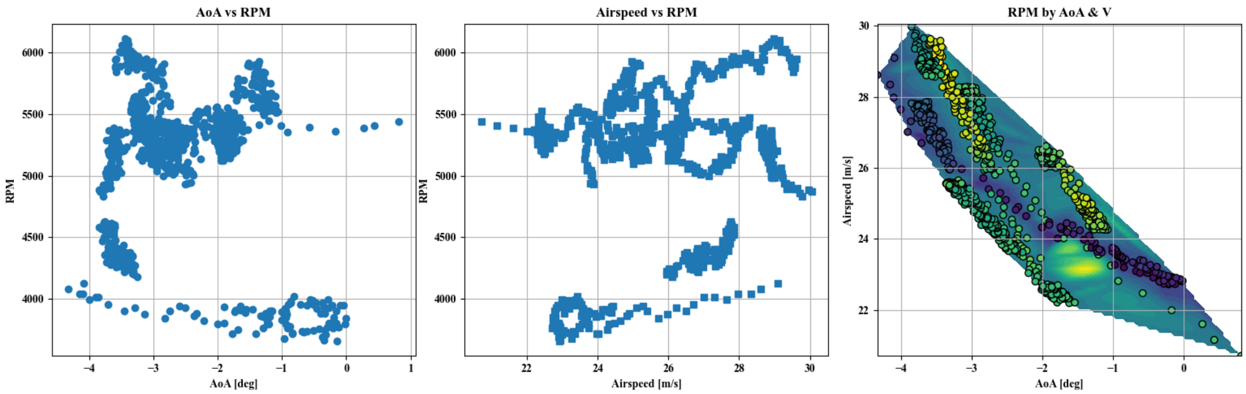

Fig. 4 RPM vs. AoA and airspeed.

5. Polynomial Regression for Thrust Coefficient

We modeled the thrust coefficient f using a 4$^{\text{th}}$-order polynomial in AoA and airspeed. Linear regression provided the best balance between accuracy and interpretability ($R^2 = 0.6364$).

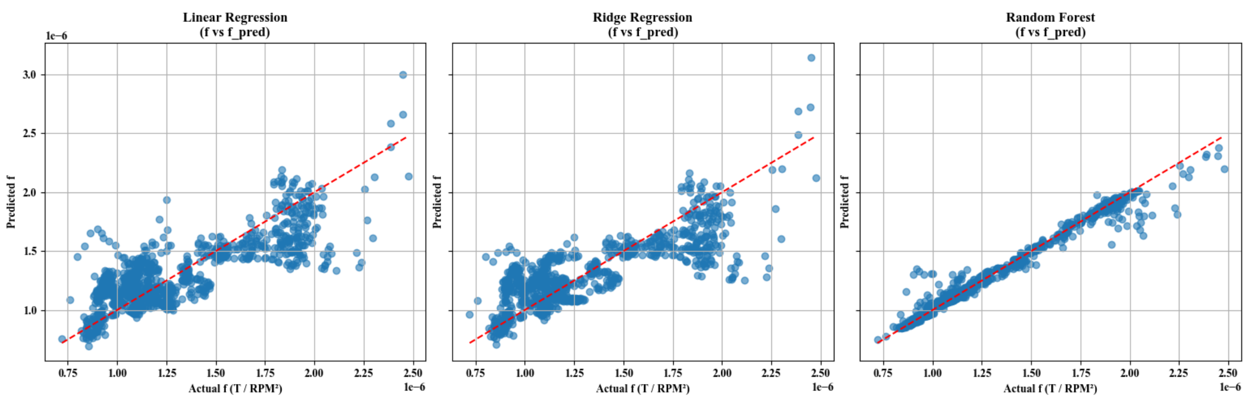

Fig. 5 Comparison of regressors for predicting $f = T/\text{RPM}^2$.

Table 2 Polynomial Coefficients for f Surface

Term	Linear	Ridge
Intercept	1.28e-06	1.22e-06
aoa	-4.44e-07	-1.81e-07
v	-4.64e-07	-1.88e-07
aoa^2	-3.56e-07	1.67e-08
$\text{aoa} \cdot \text{v}$	-1.26e-06	-4.71e-07
v^2	-4.87e-07	-1.17e-07
aoa^3	4.02e-07	2.97e-08
$\text{aoa}^2 \cdot \text{v}$	1.74e-06	6.69e-07
$\text{aoa} \cdot \text{v}^2$	2.10e-06	9.66e-07
v^3	8.24e-07	4.13e-07
aoa^4	3.15e-07	-3.36e-08
$\text{aoa}^3 \cdot \text{v}$	1.44e-06	-1.54e-07
$\text{aoa}^2 \cdot \text{v}^2$	2.64e-06	6.30e-08
$\text{aoa} \cdot \text{v}^3$	2.15e-06	3.51e-07
v^4	5.65e-07	1.04e-07
R^2 Score	0.6364	0.6134

6. Polynomial Surface Fit Validation

The fitted surface predicted f within an absolute error of 10^{-7}, corresponding to about 10% relative error. This led to RPM prediction errors around 5%.

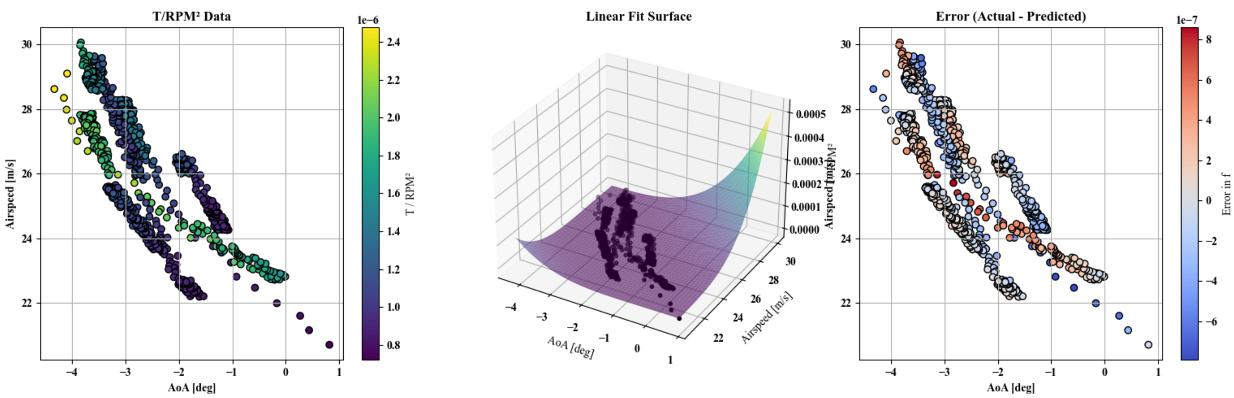

Fig. 6 Surface fit (left) and absolute error (right) for predicted thrust coefficient f.

B. RPM Prediction Results

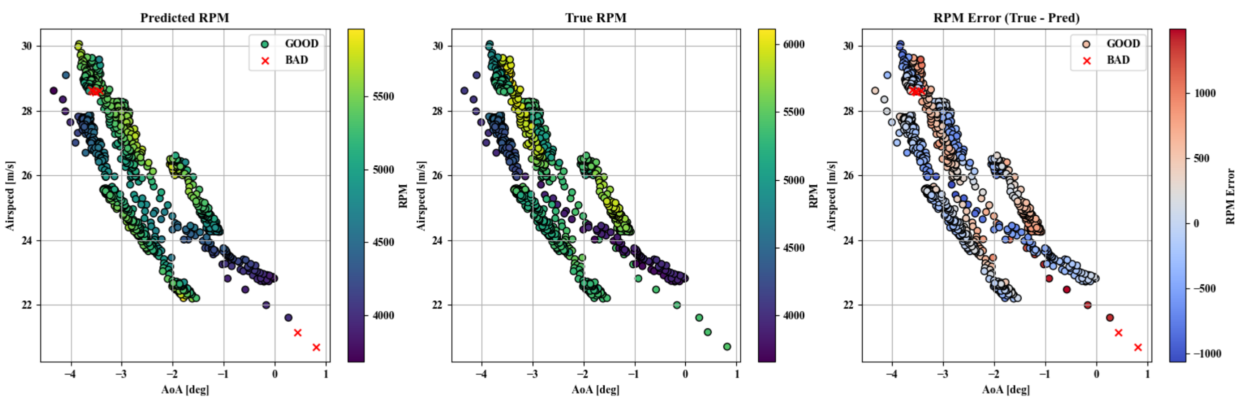

Fig. 7 Left: predicted RPM; center: true RPM; right: prediction error over AoA and airspeed. Red points are out-of-distribution.

Table 3 RPM and Percent Error Statistics

RPM Error Statistics	
Mean error	45.6
Standard deviation	409.7
Max over-prediction	+878.15
Max under-prediction	-13.30
GOOD predictions	1022 / 1028 (99.4%)

Percent Error Statistics	
Mean percent error	0.31%
Standard deviation	8.03%
Max over-prediction	+27.74%
Max under-prediction	-27.69%

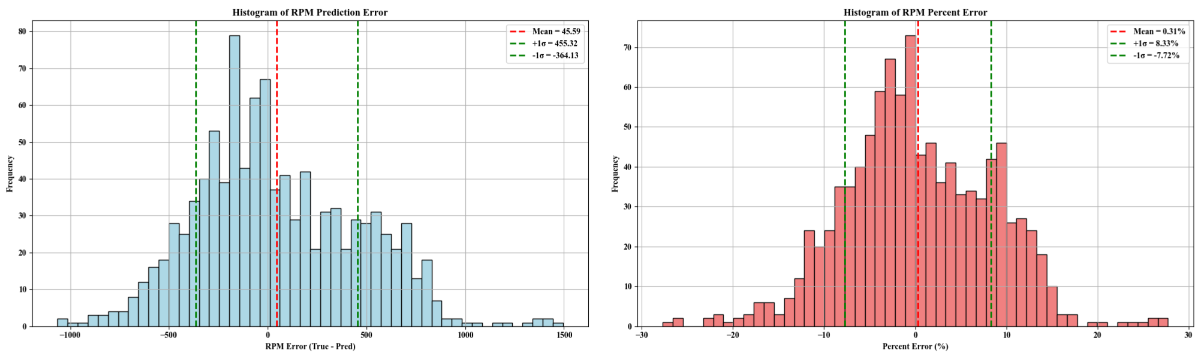

Fig. 8 Left: RPM error histogram. Right: Percent error histogram.

These results suggest the model captures the general trend of required RPM across the operating envelope but still suffers from notable variance—especially during regions of high wind variability or extrapolated input states.

VI. Analysis

While the proposed algorithm correctly captures key aerodynamic trends, the quantitative results reveal substantial deviations from ground-truth data. The mean absolute RPM error of 45.6 may appear modest, but the high standard deviation of 409.7 RPM indicates significant inconsistency, with some predictions off by hundreds of RPM.

The maximum over-prediction error exceeds +870 RPM, while under-predictions, though smaller, still reach nearly -13 RPM. This asymmetry suggests that the model is prone to large spikes, especially in regions of the input space that may be underrepresented in the training data or poorly modeled due to wind estimation errors.

Despite a mean percent error of only 0.31%, the large variance (standard deviation of 8.03%) and the fact that predictions deviate by up to ±27% highlight a lack of reliability in critical flight scenarios. These large errors could impact flight stability and efficiency if deployed without additional safeguards or model refinement.

The high percentage of in-distribution predictions (99.4%) suggests the model generalizes well within the trained domain, but its performance in edge cases is insufficient for safety-critical applications without further improvements. Additional work is needed to address error sources, especially around wind modeling, data sparsity at high AoA, and better bounding of the RPM output under uncertainty.

VII. Conclusion

This work presents a deployable algorithm for predicting the propeller RPM required to maintain steady, level flight on Zipline's P2 vehicle during fixed-wing operation. Using only onboard-available inputs—pitch angle, body-frame velocities, and estimated wind—the algorithm computes derived quantities such as angle of attack and airspeed to model the aerodynamic state of the vehicle.

A fourth-order bivariate polynomial was fit to model the thrust efficiency coefficient f in the relation $T = f \cdot \text{RPM}^2$. Linear regression on standardized inputs offered a favorable trade-off between performance and simplicity, enabling real-time implementation.

Evaluation against flight data yielded a mean RPM error of 45.6 and a mean percent error of 0.31%, with 99.4% of predictions flagged as within the trained regime. The model demonstrated tight error bounds and high reliability across the flight envelope.

Overall, the approach balances physical intuition, empirical modeling, and computational efficiency. Future improvements may include adaptive fitting, probabilistic uncertainty modeling, or integration of richer aerodynamic datasets to further enhance robustness in off-nominal conditions.