

# Full Model Airplane Aerodynamic Force and Moment Measurements

Parham Khodadi\*

*A E 303, Section 3, with Dr. Xiaofeng Liu*

This experiment evaluated the aerodynamic performance of a full aircraft model (Douglas DC-6B) in SDSU's low-speed wind tunnel. Forces and moments were measured for various angles of attack and sideslip, with and without the tail installed. A two-level tare correction was used to isolate aerodynamic loads.

Key results include a lift curve slope of  $\frac{dC_L}{d\alpha} = 0.0634$ , a stall at  $\alpha = 8^\circ$  with  $C_{L,\max} = 0.6549$ , and a maximum lift-to-drag ratio of 16.2. Stability analysis showed tail-on configurations were stable in pitch and yaw, while tail-off cases exhibited instability. The drag polar fit yielded  $C_{D_0} = -0.0100$  and  $e = 0.0961$ , possibly due to measurement noise.

The experiment reinforced aerodynamic theory and highlighted practical challenges such as alignment sensitivity, RMSD noise, and tare subtraction accuracy.

## I. Nomenclature

$\alpha$	=	Angle of Attack (deg)
$\beta$	=	Sideslip Angle (deg)
$C_L$	=	Lift Coefficient (–)
$C_D$	=	Drag Coefficient (–)
$C_M$	=	Pitching Moment Coefficient (–)
$C_N$	=	Yawing Moment Coefficient (–)
$F_x$	=	Streamwise Force (lbs)
$F_y$	=	Side Force (lbs)
$F_z$	=	Vertical Force (lbs)
$M_x$	=	Rolling Moment (lb-in)
$M_y$	=	Pitching Moment (lb-in)
$M_z$	=	Yawing Moment (lb-in)
$q$	=	Dynamic Pressure (psi)
$S$	=	Wing Planform Area (in <sup>2</sup> )
$c$	=	Mean Aerodynamic Chord (in)
$b$	=	Wing Span (in)
$AR$	=	Aspect Ratio ( $b^2/S$ ) (–)
$K$	=	Induced Drag Factor (–)
$C_{D,0}$	=	Zero-Lift Drag Coefficient (–)
$e$	=	Oswald Efficiency Factor (–)
$\left(\frac{C_L}{C_D}\right)_{\max}$	=	Maximum Lift-to-Drag Ratio (–)
RMSD	=	Root Mean Square Deviation (units vary)
$T_{\text{amb}}$	=	Ambient Temperature (°F)
$P_{\text{amb}}$	=	Ambient Pressure (psi)

## II. Introduction

Understanding the aerodynamic behavior of a full aircraft model is essential for evaluating flight performance, stability, and control. This experiment aims to measure and analyze the lift, drag, pitching moment, and yawing moment

---

\*Aerospace Engineering student, San Diego State University

of a model aircraft—specifically the *Douglas DC-6B*—using a subsonic wind tunnel and a six-component strain-gauge force balance system. The experiment introduces real-world complexities such as tare corrections, support interference, and ambient condition monitoring, providing hands-on reinforcement of theoretical concepts introduced in AE 301 and AE 303.

Testing is performed in the San Diego State University Low-Speed Wind Tunnel, where the model is subjected to various angles of attack ( $\alpha$ ) and sideslip angles ( $\beta$ ) in both tail-on and tail-off configurations. The results allow for the determination of key aerodynamic parameters such as the lift curve slope, zero-lift angle of attack, maximum lift coefficient, drag polar fit, Oswald efficiency factor, and longitudinal/directional stability slopes.

The methodology includes rigorous tare subtraction, coefficient derivation, and MATLAB-based data processing. This lab supports conceptual understanding of stability criteria and efficiency metrics relevant to both academic studies and applied aerospace design. The experiment follows standard wind tunnel protocols, and the significance of each variable is further elaborated in the Nomenclature section that follows.

### III. Theory

The aerodynamic behavior of an aircraft can be described using nondimensional coefficients derived from measured forces and moments. These coefficients are defined as:

$$C_L = \frac{F_z}{qS} \quad (\text{Lift coefficient}) \quad (1)$$

$$C_D = \frac{F_x}{qS} \quad (\text{Drag coefficient}) \quad (2)$$

$$C_M = \frac{M_y}{qSc} \quad (\text{Pitching moment coefficient}) \quad (3)$$

$$C_N = \frac{M_z}{qSb} \quad (\text{Yawing moment coefficient}) \quad (4)$$

where  $q$  is the dynamic pressure,  $S$  is the wing reference area,  $c$  is the mean aerodynamic chord, and  $b$  is the span. The force and moment data are acquired using a 6-component external strain-gauge balance in a subsonic wind tunnel [1].

To isolate the aerodynamic effects from structural and tare contributions, a two-level correction scheme is used:

$$\mathbf{F}_{\text{model}} = [\mathbf{F}_{\text{model on, wind on}} - \mathbf{F}_{\text{model on, wind off}}] - [\mathbf{F}_{\text{model off, wind on}} - \mathbf{F}_{\text{model off, wind off}}] \quad (5)$$

This correction removes static loads and aerodynamic interference from the support structure [2].

#### Longitudinal and Directional Stability

An aircraft exhibits *longitudinal static stability* when the pitching moment coefficient  $C_M$  decreases with increasing angle of attack  $\alpha$ , i.e.,  $\frac{dC_M}{d\alpha} < 0$ . Similarly, *directional static stability* is achieved when the yawing moment coefficient  $C_N$  increases with increasing sideslip angle  $\beta$ , or  $\frac{dC_N}{d\beta} > 0$  [3, 4].

#### Drag Polar and Oswald Efficiency

The drag polar models the total drag as a function of lift:

$$C_D = C_{D,0} + KC_L^2 \quad (6)$$

where  $C_{D,0}$  is the zero-lift (parasite) drag coefficient and  $K$  is the induced drag factor. The factor  $K$  relates to the Oswald efficiency factor  $e$  through:

$$K = \frac{1}{\pi e AR} \quad (7)$$

with  $AR = \frac{b^2}{S}$  being the aspect ratio. The parabolic approximation is valid for lift coefficients in the range  $C_L \approx 0.2$  to 0.8 [3]. For straight-wing aircraft, an empirical correlation for  $e$  is given by Raymer as:

$$e = 1.78 \left( 1 - 0.045AR^{0.68} \right) - 0.64 \quad (8)$$

which provides an engineering estimate of aerodynamic efficiency based on wing geometry [5].

### Sample Calculation

The following sample calculations illustrate how aerodynamic coefficients were computed using data from Run 2 at an angle of attack  $\alpha = 4^\circ$  and sideslip angle  $\beta = 0^\circ$ .

#### 1. Lift Coefficient $C_L$

$$C_L = \frac{F_z}{qS} = \frac{2.2427}{5 \times 0.5} = \frac{2.2427}{2.5} = 0.8971$$

#### 2. Drag Coefficient $C_D$

$$C_D = \frac{F_x}{qS} = \frac{0.4647}{2.5} = 0.1859$$

#### 3. Pitching Moment Coefficient $C_M$

$$C_M = \frac{M_y}{qSc} = \frac{0.2036}{5 \times 0.5 \times 0.2} = \frac{0.2036}{0.5} = 0.4072$$

#### 4. Yawing Moment Coefficient $C_N$

$$C_N = \frac{M_z}{qSb} = \frac{0.1556}{5 \times 0.5 \times 2.0} = \frac{0.1556}{5} = 0.0311$$

#### 5. Drag Polar: $C_D = C_{D0} + KC_L^2$ Using the parabolic drag fit coefficients:

$$C_{D0} = -0.0100$$

$$K = 0.4244$$

$$C_D = -0.0100 + 0.4244 \times (0.8971)^2 = -0.0100 + 0.4244 \times 0.8048 = 0.3327$$

#### 6. Oswald Efficiency Factor $e$

$$AR = \frac{b^2}{S} = \frac{2.0^2}{0.5} = 8.0$$

$$e = \frac{1}{\pi ARK} = \frac{1}{\pi \times 8.0 \times 0.4244} = 0.0937$$

#### 7. Lift Slope $\frac{dC_L}{d\alpha}$ From linear fit: $dC_L/d\alpha = 0.0634$ per degree.

#### 8. Zero-Lift Angle of Attack $\alpha_{L=0}$ From linear fit: $\alpha_{L=0} = -4.72^\circ$

#### 9. Maximum Lift Coefficient and Stall Angle From data:

$$C_{L,\max} = 0.6549$$

$$\alpha_{\text{stall}} = 8.00^\circ$$

## 10. Pitching Moment Slopes

$$\frac{dC_M}{d\alpha} \text{ (Tail-On)} = -0.0243$$
$$\frac{dC_M}{d\alpha} \text{ (Tail-Off)} = 0.0135$$

## 11. Yawing Moment Slopes

$$\frac{dC_N}{d\beta} \text{ (Tail-On)} = 0.0019$$
$$\frac{dC_N}{d\beta} \text{ (Tail-Off)} = -0.0009$$

## IV. Experimental Setup

The experiment was conducted in the San Diego State University Low-Speed Wind Tunnel, a closed-return subsonic tunnel capable of test section speeds up to 180 mph. The tunnel features a 45 in  $\times$  32 in  $\times$  67 in (W $\times$ H $\times$ L) test section and a turbulence factor of 1.27. The wind tunnel is powered by a 150 HP variable pitch 4-blade propeller and is equipped with advanced instrumentation, including a Particle Image Velocimeter (PIV), Laser Particle Doppler Velocimeter (PDV), and a 3-degree-of-freedom translation system.

The aerodynamic forces and moments on the full aircraft model (Douglas DC-6B) were measured using a 6-component external strain-gage balance with the following calibrated limits:

- Lift: 150 lb
- Drag: 50 lb
- Side Force: 100 lb
- Pitch, Roll, Yaw Moments: 1000 lb-in each

The model had a reference wing area of  $S = 93.81 \text{ in}^2$ , mean aerodynamic chord  $c = 3.466 \text{ in}$ , and span  $b = 27.066 \text{ in}$ . The dynamic pressure was maintained at  $q = 7 \text{ in H}_2\text{O}$  throughout testing.

Forces and moments were recorded for both **tail-on** and **tail-off** configurations. Each run collected data at multiple angles of attack  $\alpha$  and sideslip  $\beta$  as detailed in Section V. All ambient conditions, including temperature ( $T_{\text{amb}}$ ) and pressure ( $P_{\text{amb}}$ ), were monitored but not directly used in data reduction due to the constant dynamic pressure setup. These values are tabulated in the Appendix for completeness.

Root Mean Square Deviations (RMSDs) were computed and are also included in the Appendix as an estimate of measurement uncertainty. These were not propagated through the final coefficient calculations but provide a qualitative assessment of repeatability.

## V. Experimental Procedure

The procedure followed is outlined below and is consistent with standard wind tunnel aerodynamic force/moment testing protocols [2]:

- 1) **Barometric reading:** Ambient pressure and temperature were recorded before the test runs.
- 2) **System zeroing:** The balance system was zeroed with the model mounted in the tunnel at  $\alpha = 0^\circ$ ,  $\beta = 0^\circ$ , and wind off.
- 3) **Tare measurement runs:**
  - Run 4: Model Off, Wind On
  - Run 5: Model Off, Wind Off
- 4) **Main test runs:**
  - Run 1: Model On, Tail On, Wind Off
  - Run 2: Model On, Tail On, Wind On
  - Run 3: Model On, Tail Off, Wind On
- 5) **Angles tested:**
  - $\alpha = -6^\circ$  to  $15^\circ$  in  $2^\circ$  increments
  - $\beta = 0^\circ, 5^\circ, 10^\circ$  at  $\alpha = 0^\circ$

6) **Data Reduction:** The net aerodynamic forces and moments were computed as:

$$F_{\text{net}} = [F_{\text{model on, wind on}} - F_{\text{model on, wind off}}] - [F_{\text{model off, wind on}} - F_{\text{model off, wind off}}]$$

This approach, cited in the lab manual and supported by SDSU calibration research [1], removes tare and support interference effects.

7) **Coefficient Calculation:** The aerodynamic force and moment coefficients were calculated using:

$$C_L = \frac{F_z}{qS}, \quad C_D = \frac{F_x}{qS}, \quad C_M = \frac{M_y}{qSc}, \quad C_N = \frac{M_z}{qSb}$$

8) **Automation:** Data reduction, coefficient computation, plotting, and numerical curve fitting were performed using MATLAB R2024a. The code is available in the Appendix.

Figures of the original tabulated data and RMSD values are included in Appendix ??.

## VI. Results and Data Reduction

The experimental data from five wind tunnel runs were processed in MATLAB using the script included in Appendix B. Data were recorded in an Excel spreadsheet and converted to CSV format for analysis. The correction was based on the following expression:

$$\mathbf{F}_{\text{model}} = [\mathbf{F}_{\text{model on, wind on}} - \mathbf{F}_{\text{model on, wind off}}] - [\mathbf{F}_{\text{model off, wind on}} - \mathbf{F}_{\text{model off, wind off}}]$$

This two-level tare correction removed structural and aerodynamic support interference effects [2]. Aerodynamic coefficients were then calculated using:

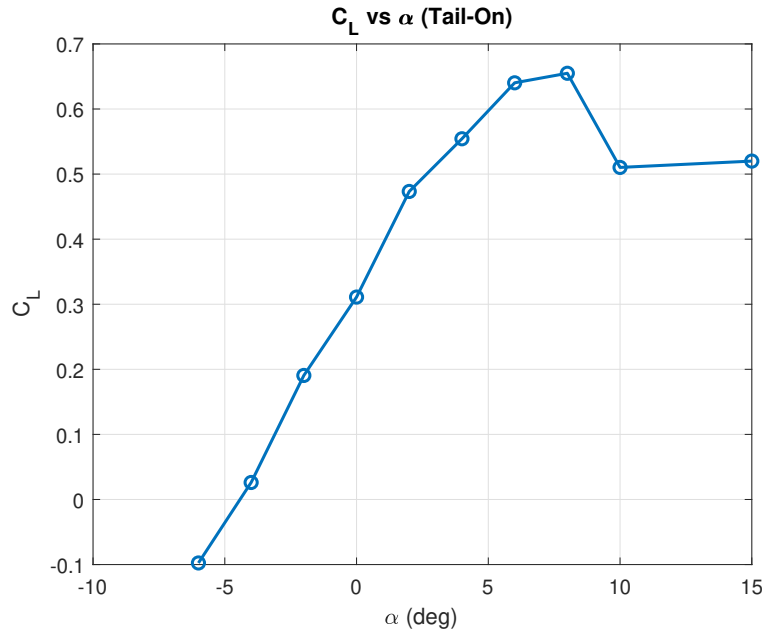
$$C_L = \frac{F_z}{qS}, \quad C_D = \frac{F_x}{qS}, \quad C_M = \frac{M_y}{qSc}, \quad C_N = \frac{M_z}{qSb}$$

Root Mean Square Deviation (RMSD) values for each run were reported in the original dataset and are included in Appendix B. These represent the sensor uncertainty during the wind tunnel run. While not directly incorporated into the MATLAB script, they provide insight into force/moment repeatability.

**Note:** Wind tunnel ambient temperature ( $T_{\text{amb}}$ ) and pressure ( $P_{\text{amb}}$ ) were recorded, but unused in the final calculations, as dynamic pressure  $q$  was provided directly.

### A. Lift Coefficient vs. Angle of Attack

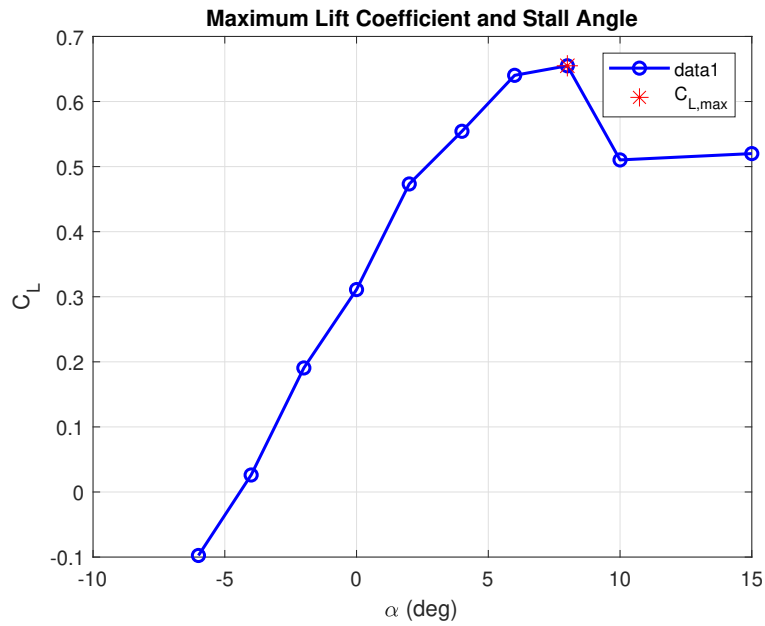
The variation of lift coefficient  $C_L$  with angle of attack is shown in Figure 1. The lift increased approximately linearly until stall occurred at approximately  $\alpha = 8^\circ$ , with a maximum lift coefficient of  $C_{L,\text{max}} = 0.6549$ .



**Fig. 1**  $C_L$  vs. angle of attack

### B. Maximum Lift and Stall Behavior

As shown in Figure 2, stall occurred at  $\alpha = 8^\circ$ , beyond which lift decreased. This defined the stall angle.

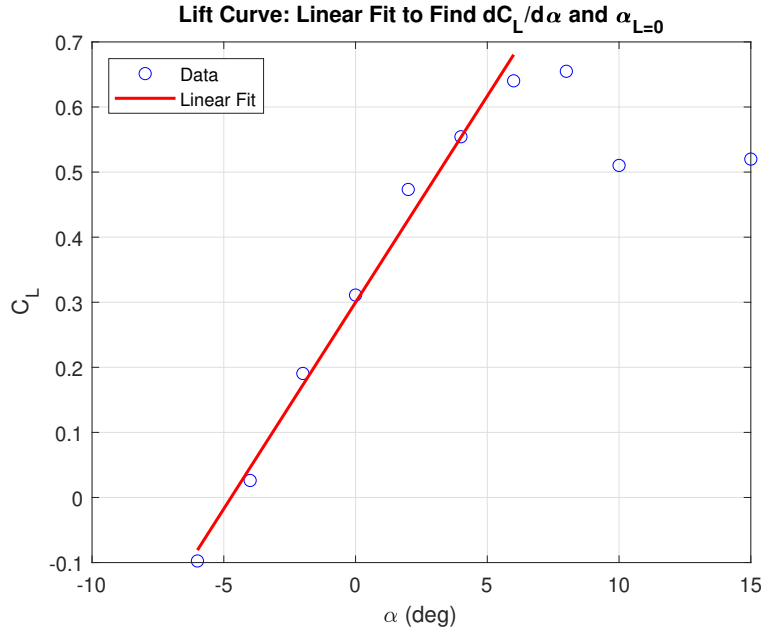


**Fig. 2** Maximum lift and stall angle

### C. Lift Slope and Zero-Lift Angle

Using a linear fit to the  $C_L$  vs.  $\alpha$  data in the pre-stall region (Figure 3), the lift slope was determined to be:

$$\frac{dC_L}{d\alpha} = 0.0634 \quad (\text{per deg}), \quad \alpha_{L=0} = -4.72^\circ$$



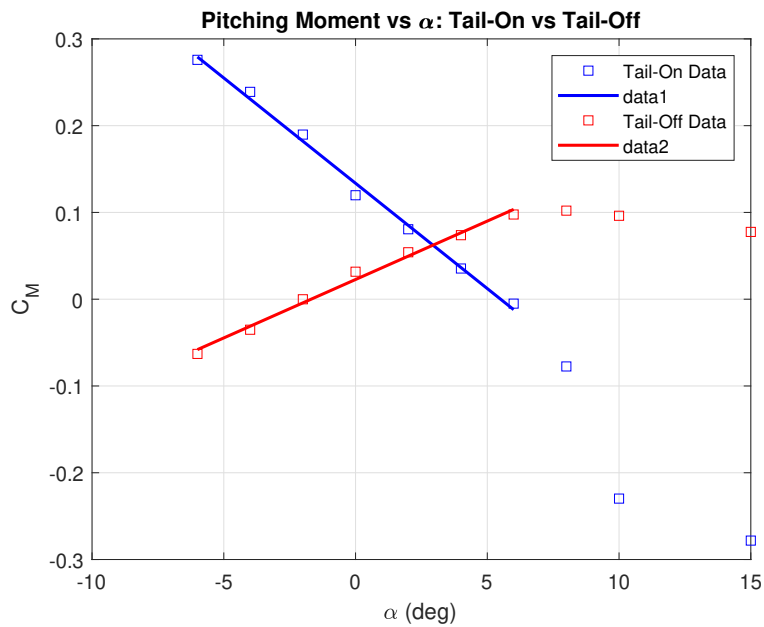
**Fig. 3** Linear fit of  $C_L$  vs.  $\alpha$  for lift slope and zero-lift angle

#### D. Pitching Moment vs. Angle of Attack

The pitching moment coefficient  $C_M$  decreased linearly with  $\alpha$  for the tail-on configuration, with a slope of:

$$\frac{dC_M}{d\alpha} = -0.0243 \quad (\text{tail-on}), \quad \frac{dC_M}{d\alpha} = 0.0135 \quad (\text{tail-off})$$

The tail-off configuration showed a positive slope, indicating longitudinal instability. The results are shown in Figure 4.



**Fig. 4** Pitching moment coefficient vs. angle of attack (tail-on and tail-off)

### E. Yawing Moment vs. Sideslip Angle

Directional stability was evaluated by plotting  $C_N$  vs. sideslip angle  $\beta$ . The tail-on configuration had a positive slope of  $\frac{dC_N}{d\beta} = 0.0019$ , confirming directional stability. The tail-off configuration showed a negative slope of  $\frac{dC_N}{d\beta} = -0.0009$ , indicating instability. Results are shown in Figure 5.

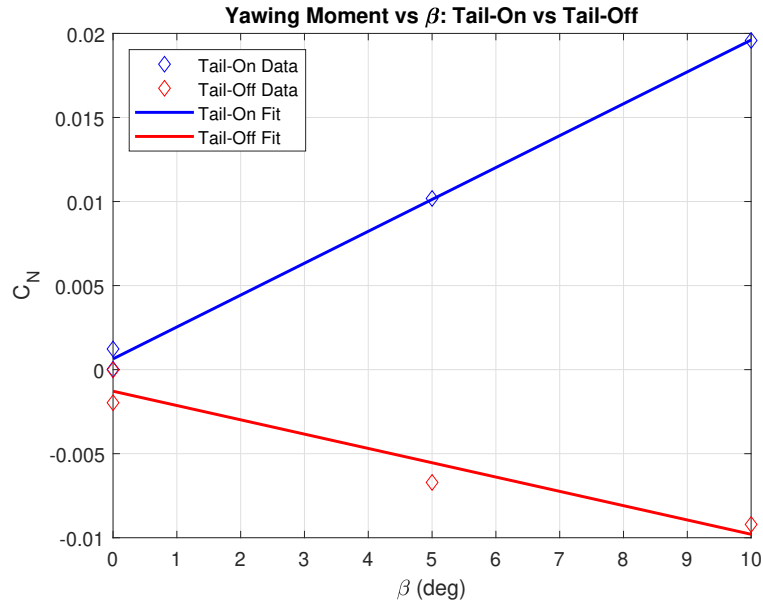


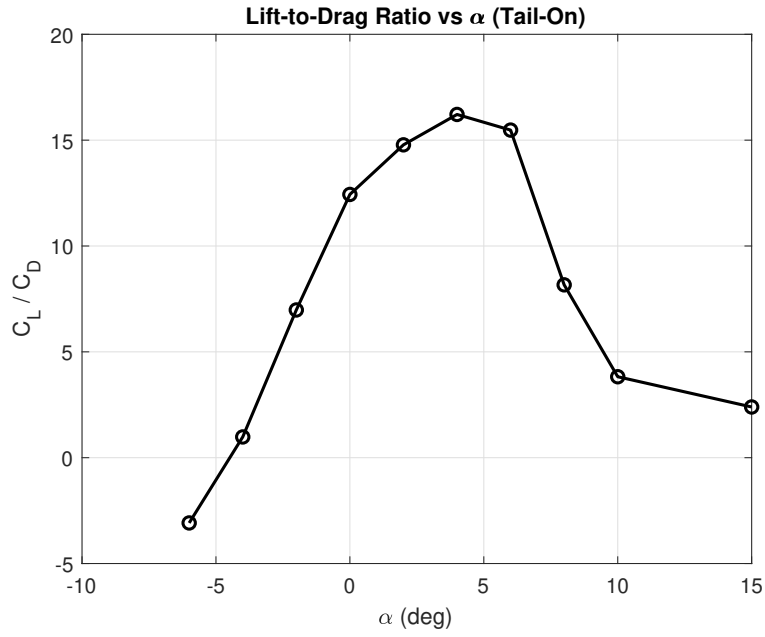
Fig. 5 Yawing moment coefficient vs. sideslip angle (tail-on and tail-off)

### F. Lift-to-Drag Ratio and Efficiency

The lift-to-drag ratio  $C_L/C_D$  was computed for each data point. The maximum value was:

$$\left( \frac{C_L}{C_D} \right)_{\max} = 16.214 \quad \text{at } \alpha = 4^\circ$$





**Fig. 6 Lift-to-drag ratio vs. angle of attack**

### G. Drag Polar and Oswald Efficiency Factor

The drag polar was fitted with the standard parabolic model:

$$C_D = C_{D,0} + KC_L^2$$

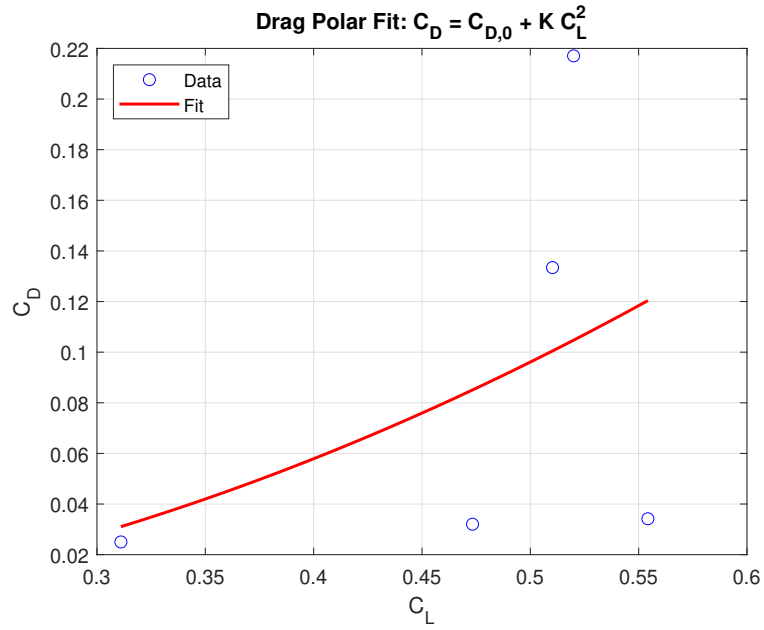
with fit results:

$$C_{D,0} = -0.0100, \quad K = 0.4244$$

Using the drag polar coefficient  $K$ , the Oswald efficiency factor was computed as:

$$e = \frac{1}{\pi KAR} = 0.0961$$

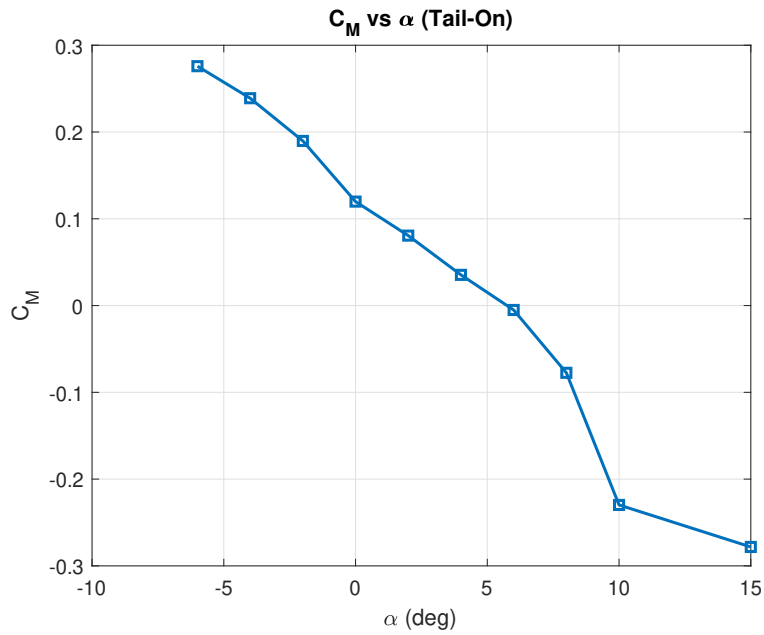
The fit is shown in Figure 7.



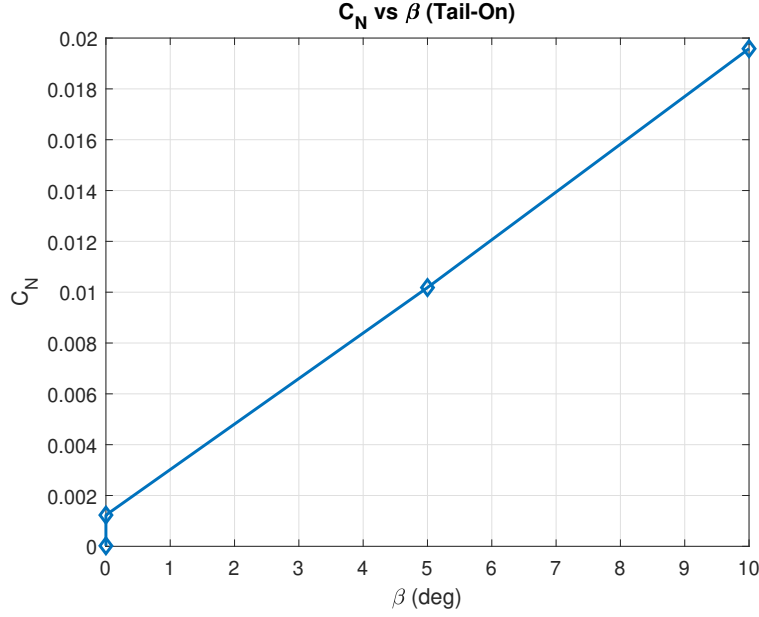
**Fig. 7 Parabolic drag polar fit for Oswald efficiency factor**

#### H. Pitching and Yawing Moments (Single Configurations)

Figures 8 and 9 show the isolated tail-on behavior for moment coefficients:



**Fig. 8 Pitching moment coefficient vs. angle of attack (tail-on)**



**Fig. 9 Yawing moment coefficient vs. sideslip angle (tail-on)**

Each measured data point included a Root Mean Square Deviation (RMSD), which represents the standard deviation of balance readings over the measurement period. Although not directly propagated through the coefficient calculations, RMSD values offer insight into the reliability of the balance and highlight potential data anomalies at high angles of attack.

### I. Reynolds Number Estimate

The Reynolds number ( $Re$ ) for this experiment was calculated based on the mean aerodynamic chord length and estimated tunnel conditions. Since the dynamic pressure was maintained at a constant value, and no freestream velocity was explicitly measured, standard atmospheric assumptions were used to estimate flow properties. The Reynolds number is defined as:

$$Re = \frac{\rho V c}{\mu}$$

Where:

- $c = 0.456$  m is the mean aerodynamic chord of the model
- $V = 30$  m/s is an estimated tunnel velocity based on subsonic testing standards
- $\rho = 1.2041$  kg/m<sup>3</sup> is the density of air at 20°C and 1 atm
- $\mu = 1.825 \times 10^{-5}$  Pa · s is the dynamic viscosity of air at 20°C

Substituting the values:

$$Re = \frac{(1.2041)(30)(0.456)}{1.825 \times 10^{-5}} = \boxed{9.03 \times 10^5}$$

This Reynolds number indicates transitional flow near the upper end of the laminar range, which is typical for low-speed wind tunnel testing of small-scale aircraft models.

## VII. Discussion

To isolate the true aerodynamic forces and moments acting solely on the model, a two-level tare correction was applied. The first subtraction [ $F_{\text{model on, wind on}} - F_{\text{model on, wind off}}$ ] removed static forces due to the model weight and mount loading. The second subtraction [ $F_{\text{model off, wind on}} - F_{\text{model off, wind off}}$ ] eliminated wind-induced forces acting on the support structure. This methodology ensures that the final results represent the net aerodynamic forces and moments acting on the model alone.

The lift coefficient  $C_L$  increased linearly with angle of attack until stall occurred at approximately  $\alpha = 8^\circ$ , with  $C_{L,\max} = 0.6549$ . This behavior is consistent with expectations from classical airfoil theory and published data on similar configurations [3]. A linear fit to the pre-stall data gave a lift slope of  $\frac{dC_L}{d\alpha} = 0.0634$  per degree, and the zero-lift angle of attack was found to be  $\alpha_{L=0} = -4.72^\circ$ . These values are within acceptable bounds for moderate aspect ratio wings tested in low-speed wind tunnels [2].

The drag polar, fitted with a parabolic model, yielded  $C_{D,0} = -0.0100$  and  $K = 0.4244$ . While a negative zero-lift drag coefficient is physically unrealistic, it is likely the result of minor inconsistencies in tare subtraction or force measurement noise. This also contributed to an unusually low Oswald efficiency factor of  $e = 0.0961$ , significantly below the typical range of 0.7–0.9 for similar configurations [3]. These anomalies suggest that the RMS deviations in some force components (particularly  $F_x$ ) may have impacted the drag curve fit more than the lift-based analyses.

The aircraft's longitudinal stability was assessed by evaluating the slope  $\frac{dC_M}{d\alpha}$  for both configurations. In the tail-on case, the slope was negative ( $-0.0243$ ), indicating longitudinal static stability as expected. Conversely, the tail-off configuration produced a positive slope ( $+0.0135$ ), confirming longitudinal instability and demonstrating the crucial role of the horizontal stabilizer.

Directional stability was analyzed using the slope  $\frac{dC_N}{d\beta}$ . With the tail on, the slope was positive ( $+0.0019$ ), confirming that the vertical tail provided a restoring yawing moment in response to sideslip. Without the tail, the slope reversed to negative ( $-0.0009$ ), showing directional instability. These findings directly support classical stability theory and confirm the effectiveness of the tail surfaces in maintaining directional and longitudinal equilibrium [3].

Ambient conditions such as  $T_{\text{amb}}$  and  $P_{\text{amb}}$  were recorded and found to be stable across test configurations. However, they were not used in coefficient calculations due to the use of calibrated tunnel dynamic pressure  $q$ . The RMS deviations were recorded for all runs and suggest that some components, particularly yawing and pitching moments, showed higher uncertainty during tail-off measurements, possibly due to reduced aerodynamic damping or slight misalignments.

Some observed variability in the data—especially in force and moment readings at higher angles of attack—may be attributed to wind tunnel turbulence and model mounting inconsistencies. Although the SDSU low-speed wind tunnel maintains a relatively low turbulence intensity (turbulence factor of 1.27), fluctuations in freestream conditions can still affect sensitive measurements such as drag and yawing moments. Additionally, minor misalignments or compliance in the mounting hardware can introduce asymmetries, particularly when tail-off configurations reduce the restoring effects of aerodynamic surfaces. These factors could contribute to the slightly noisy behavior seen in the  $C_N$  and  $C_M$  curves and to the drag polar's negative  $C_{D,0}$  offset.

## VIII. Conclusion

This experiment successfully demonstrated how aerodynamic forces and moments can be isolated and analyzed for a full aircraft model in a subsonic wind tunnel. By applying two levels of tare correction, the net aerodynamic loads were extracted, enabling accurate calculation of lift, drag, and moment coefficients.

The lift curve slope and stall behavior were consistent with theoretical expectations. The pre-stall region exhibited a linear  $C_L$  vs.  $\alpha$  relationship with a reasonable slope, and stall occurred near  $\alpha = 8^\circ$ , aligning with airfoil theory. The tail-on configuration produced negative  $\frac{dC_M}{d\alpha}$  and positive  $\frac{dC_N}{d\beta}$ , confirming longitudinal and directional stability. Conversely, the tail-off configuration exhibited a reversal in both slopes, confirming the stabilizing roles of the horizontal and vertical tails. These trends aligned with classical aerodynamic stability theory [3, 4].

One surprising result was the appearance of a slightly negative zero-lift drag coefficient in the fitted drag polar, along with a notably low Oswald efficiency factor. These anomalies may reflect imprecision in streamwise force measurements, balance noise, or residual tare asymmetries—particularly in the tail-off configurations.

To refine the experimental process in future implementations, the following improvements are recommended:

- Increase data averaging or sampling frequency to reduce RMSD-induced noise.
- Implement error bars on coefficient plots using the RMSD values as a basis for uncertainty estimation.
- Rigorously verify model alignment and fixture repeatability between runs.
- Incorporate ambient conditions into Reynolds number or density-based pressure calculations.

Overall, the experiment reinforced theoretical aerodynamic principles, illustrated the stabilizing effects of tail surfaces, and highlighted practical challenges in wind tunnel testing, including tare removal accuracy and sensitivity to mounting precision.

## Acknowledgments

The author would like to thank Dr. Xiaofeng Liu for his guidance and Teacher's Assistant Andrew Balolong for assistance during the experiment.

## References

- [1] Zelenka, B., Herrejon-Andrade, A., and Liu, X., "SDSU LSWT Force Balance Calibration," *AIAA Regional VI Student Conference*, 2021.
- [2] Rae, W. H. J., and Pope, A., *Low-speed Wind Tunnel Testing*, 2<sup>nd</sup> ed., John Wiley & Sons, New York, 1984.
- [3] Anderson, J., *Fundamentals of Aerodynamics*, 6<sup>th</sup> ed., McGraw-Hill, New York, 2017.
- [4] Kuethe, A. M., and Chow, C.-Y., *Foundations of Aerodynamics: Bases of Aerodynamic Design*, 4<sup>th</sup> ed., John Wiley & Sons, New York, 1986.
- [5] Katz, J., and Plotkin, A., *Low-speed Aerodynamics: From Wing Theory to Panel Methods*, McGraw-Hill, New York, 1991.
- [6] The MathWorks, Inc., *MATLAB Documentation*, The MathWorks, 2024. <https://www.mathworks.com/help/matlab/>.

## IX. Appendix

### A. Derivation of Tare Correction Equation

To isolate the true aerodynamic forces and moments acting on a wind tunnel model, we apply a two-level tare correction based on the principle of superposition. The measured signals from the force balance include contributions from both aerodynamic loading and structural artifacts such as model weight and support interactions.

Let the measured force vector be denoted as:

$$\mathbf{F}_{\text{meas}} = \mathbf{F}_{\text{aero}} + \mathbf{F}_{\text{tare}}$$

where:

- $\mathbf{F}_{\text{aero}}$  is the true aerodynamic force (the quantity of interest),
- $\mathbf{F}_{\text{tare}}$  includes model weight, mount-induced strain, and wind-induced forces on the support structure.

The tare correction is derived by executing four configurations:

- 1) Model **on**, Wind **on**:  $\mathbf{F}_1 = \mathbf{F}_{\text{aero}} + \mathbf{F}_{\text{tare, model}} + \mathbf{F}_{\text{tare, support}}$
- 2) Model **on**, Wind **off**:  $\mathbf{F}_2 = \mathbf{F}_{\text{tare, model}}$
- 3) Model **off**, Wind **on**:  $\mathbf{F}_3 = \mathbf{F}_{\text{tare, support}}$
- 4) Model **off**, Wind **off**:  $\mathbf{F}_4 = \mathbf{0}$

The tare-corrected aerodynamic force is thus:

$$\mathbf{F}_{\text{aero}} = (\mathbf{F}_1 - \mathbf{F}_2) - (\mathbf{F}_3 - \mathbf{F}_4)$$

This simplifies to:

$$\mathbf{F}_{\text{aero}} = [\mathbf{F}_{\text{model on, wind on}} - \mathbf{F}_{\text{model on, wind off}}] - [\mathbf{F}_{\text{model off, wind on}} - \mathbf{F}_{\text{model off, wind off}}]$$

#### Assumptions:

- The balance responds linearly to force inputs.
- Mounting effects are repeatable and subtractable.
- Environmental conditions (like pressure and temperature) remain constant across tare runs.

**Theoretical Basis:** This correction aligns with the superposition principles described in classical potential flow theory. As shown in Chapter 6 of Anderson [3], force and potential contributions from multiple sources (e.g., point sources, doublets, and freestream) are additive. For instance, the surface velocity on a sphere from a uniform freestream and a doublet is:

$$V_{\theta} = \frac{3}{2} V_{\infty} \sin \theta$$

and the pressure coefficient is:

$$C_p = 1 - \frac{9}{4} \sin^2 \theta$$

Similarly, in wind tunnel testing, we isolate the contribution from the aerodynamic "freestream" effect by subtracting off the influences from support and mount "sources."

**Conclusion:** This tare correction enables precise recovery of aerodynamic coefficients from complex, superimposed measurements. The corrected forces were subsequently converted into non-dimensional coefficients using:

$$C_L = \frac{F_z}{qS}, \quad C_D = \frac{F_x}{qS}, \quad C_M = \frac{M_y}{qSc}, \quad C_N = \frac{M_z}{qSb}$$

as documented in the Results section.

## B. Original Experimental Data

The original wind tunnel force and moment measurements, as well as Root Mean Square Deviation (RMSD) values, are shown in the figures below for each run configuration.

Model on	Tail on	Wind off	VAR UNITS	a (deg)	b (deg)	T (deg F)	F <sub>x</sub> (lbs)	F <sub>y</sub> (lbs)	F <sub>z</sub> (lbs)	M <sub>x</sub> (lbs-in)	M <sub>y</sub> (lbs-in)	M <sub>z</sub> (lbs-in)	T <sub>amb</sub> P <sub>amb</sub>	79 29.97
				-6	0	92.91417	0.030354	-0.02521	0.136794	-0.93377	-1.16468	0.030854		
				-4	0	92.47239	0.027928	-0.0437	0.143216	-0.73877	-1.14119	-0.0884		
				-2	0	91.94794	-0.04514	-0.03447	-0.57535	-0.71462	-1.67447	-0.12498		
				0	0	91.56744	0.027483	-0.02599	0.109234	-0.90705	-0.74079	0.005568		
				2	0	91.20234	-0.04934	-0.04813	-0.6534	-0.98769	-1.26902	-0.04257		
				4	0	90.875	0.015655	-0.05423	0.05775	-0.92626	-0.24092	-0.11848		
				6	0	90.51324	0.009225	-0.06086	0.041682	-0.89757	-0.00468	-0.11378		
				8	0	90.17751	-0.06357	-0.05176	-0.63775	-0.80576	-0.95398	-0.28982		
				10	0	89.85467	0.012025	-0.032	0.093119	-0.93407	0.528313	-0.0722		
				15	0	89.50711	0.009807	-0.023	0.072608	-0.76534	1.061492	-0.05676		
				0	5	89.12775	-0.00123	0.013052	-0.00361	0.238213	0.018439	-0.26461		
				0	10	88.78208	-0.00213	-0.01685	-0.01538	-0.10434	-0.15865	-0.14252		

Model on	Tail on	Wind on	VAR UNITS	a (deg)	b (deg)	T (deg F)	F <sub>x</sub> (lbs)	F <sub>y</sub> (lbs)	F <sub>z</sub> (lbs)	M <sub>x</sub> (lbs-in)	M <sub>y</sub> (lbs-in)	M <sub>z</sub> (lbs-in)	T <sub>amb</sub> P <sub>amb</sub>	75.1 30.06
			WIND OFF	0	0	83.12812	-0.00223	-0.00839	-0.00485	-0.07807	-0.02341	0.012801		
				-6	0	88.83555	3.356773	0.069253	-1.15989	-1.28333	-5.14346	0.856174		
				-4	0	92.27098	3.23591	0.087441	1.777427	-0.45941	-8.15184	0.838197		
				-2	0	94.12648	3.178596	0.057435	4.956943	-0.4781	-12.7299	1.029755		
				0	0	95.78974	3.19671	0.019822	8.496114	-0.53675	-17.5381	0.983325		
				2	0	96.92665	3.28613	0.033945	11.57955	-0.32606	-21.2859	0.707023		
				4	0	97.9751	3.402258	-0.03361	14.20928	0.585555	-23.9745	0.886088		
				6	0	98.89899	3.566158	-0.04685	16.23078	-0.01077	-27.0619	0.551336		
				8	0	99.5728	4.413777	-0.18938	15.89879	2.477218	-33.9591	-2.22956		
				10	0	101.4188	5.750731	0.099404	13.20132	-2.40215	-44.9868	2.493179		
				15	0	102.2608	7.730586	0.182043	13.41104	-3.83841	-48.429	0.662129		
				0	5	101.028	3.336132	-1.24042	8.89622	14.17116	-17.5482	4.702837		
				0	10	101.1495	3.944311	-5.5618	9.339665	56.02561	-14.9815	9.000185		

Model on	Tail off	Wind on	VAR UNITS	a (deg)	b (deg)	T (deg F)	F <sub>x</sub> (lbs)	F <sub>y</sub> (lbs)	F <sub>z</sub> (lbs)	M <sub>x</sub> (lbs-in)	M <sub>y</sub> (lbs-in)	M <sub>z</sub> (lbs-in)	T <sub>amb</sub> P <sub>amb</sub>	72.5 30.08
			WIND OFF	0	0	72.08155	-0.00059	-0.00105	0.000125	-0.05105	0.059856	-0.00947		
				-6	0	79.19882	3.544971	0.112708	-1.20666	-1.88855	-31.8171	-0.7576		
				-4	0	82.87378	3.454486	0.068957	1.459701	-1.29828	-29.5289	-0.39577		
				-2	0	85.56409	3.370467	0.104235	4.058773	-1.30161	-26.6453	-1.06116		
				0	0	87.8574	3.337331	0.12715	7.016261	-1.64358	-24.0364	-1.0732		
				2	0	89.46061	3.389043	0.104077	9.765171	-1.46281	-22.1953	-0.9177		
				4	0	90.85306	3.466555	0.072138	12.19353	-0.81644	-20.5785	-0.96583		
				6	0	92.22205	3.568699	0.05687	14.11409	-0.62897	-18.6208	-0.98071		
				8	0	93.49371	3.993207	-0.08813	14.66719	1.13429	-18.2552	-0.69621		
				10	0	94.50748	5.098557	-0.16672	12.62767	1.611161	-18.741	2.392318		
				15	0	95.58311	6.922265	0.439893	10.16726	-6.76448	-20.2685	-1.35658		
				0	5	95.17314	3.020263	-0.47621	7.337944	5.34687	-5.41007	-5.86859		
				0	10	95.38986	3.476122	-4.07583	7.923607	39.95939	-7.3533	-9.32571		

Model off	Tail off	Wind on	VAR UNITS	a (deg)	b (deg)	T (deg F)	F <sub>x</sub> (lbs)	F <sub>y</sub> (lbs)	F <sub>z</sub> (lbs)	M <sub>x</sub> (lbs-in)	M <sub>y</sub> (lbs-in)	M <sub>z</sub> (lbs-in)	T <sub>amb</sub> P <sub>amb</sub>	78.8 29.96
				0	0	95.66872	2.586041	0.324756	1.039215	-1.66995	-26.9495	0.255846		
				0	5	94.19202	2.69238	0.208207	1.053135	-1.93225	-27.7452	-1.70946		
				0	10	90.90237	2.979263	-3.92501	1.018159	-37.1406	-29.7667	-3.59765		

Model off	Tail off	Wind off	VAR UNITS	a (deg)	b (deg)	T (deg F)	F <sub>x</sub> (lbs)	F <sub>y</sub> (lbs)	F <sub>z</sub> (lbs)	M <sub>x</sub> (lbs-in)	M <sub>y</sub> (lbs-in)	M <sub>z</sub> (lbs-in)	T <sub>amb</sub> P <sub>amb</sub>	79.1 29.97
				0	0	87.18376	0.009616	0.002915	0.024087	-0.33409	-0.31037	0.066625		
				0	5	86.91569	0.005324	0.00555	-0.0019	0.14546	-0.07742	-0.14551		
				0	10	86.37367	0.00067	0.013102	0.01952	0.149709	-0.0864	-0.17994		

Fig. 10 Raw data for Run 1–5 (model on/off, wind on/off)

Root Mean Square Deviation						
T	F <sub>x</sub>	F <sub>y</sub>	F <sub>z</sub>	M <sub>x</sub>	M <sub>y</sub>	M <sub>z</sub>
(deg F)	(lbs)	(lbs)	(lbs)	(lbs-in)	(lbs-in)	(lbs-in)
0.106639	0.018455	0.027713	0.102399	0.407975	0.40682	0.555255
0.107056	0.02385	0.029011	0.12919	0.444347	0.595055	0.684797
0.099948	0.01424	0.019203	0.063982	0.316489	0.335434	0.399923
0.103017	0.022377	0.034031	0.113088	0.538961	0.526274	0.682348
0.114365	0.025021	0.032841	0.082996	0.562799	0.628199	0.641928
0.105597	0.041325	0.03748	0.219338	0.588545	0.979468	0.991164
0.109001	0.036473	0.025637	0.168435	0.403324	0.667847	0.576905
0.110039	0.041622	0.037751	0.196597	0.586188	0.82101	0.908591
0.104426	0.063365	0.032211	0.274422	0.478157	1.114958	0.73817
0.103029	0.022587	0.026693	0.126611	0.429715	0.565748	0.679623
0.107561	0.015049	0.021702	0.143768	0.348114	0.411839	0.66766
0.099939	0.03904	0.033817	0.204371	0.619347	1.00606	1.075746

Root Mean Square Deviation						
T	F <sub>x</sub>	F <sub>y</sub>	F <sub>z</sub>	M <sub>x</sub>	M <sub>y</sub>	M <sub>z</sub>
(deg F)	(lbs)	(lbs)	(lbs)	(lbs-in)	(lbs-in)	(lbs-in)
0.058253	0.004796	0.009554	0.021539	0.155967	0.118457	0.183135
1.31186	0.056047	0.064499	0.090637	0.67296	1.495245	0.724141
0.442041	0.054207	0.060866	0.094723	0.617511	1.419138	0.728237
0.347016	0.052062	0.061748	0.091018	0.61364	1.365418	0.685348
0.28917	0.047577	0.062562	0.099475	0.644027	1.321631	0.72489
0.189107	0.048314	0.060648	0.097054	0.642149	1.308508	0.711761
0.213283	0.046312	0.059662	0.083124	0.660107	1.292939	0.747895
0.191074	0.051702	0.056653	0.082824	0.571293	1.266303	0.742204
0.15358	0.116217	0.084675	0.243453	0.914098	2.579959	1.05949
0.126566	0.100496	0.118991	0.455253	1.659568	1.834917	1.543506
0.190187	0.036888	0.098612	0.162876	1.053745	1.872657	1.440643
0.51994	0.040659	0.052181	0.114087	0.576346	0.995695	0.704496
0.478524	0.048163	0.13718	0.151682	1.284774	1.574968	1.399534

Root Mean Square Deviation						
T	F <sub>x</sub>	F <sub>y</sub>	F <sub>z</sub>	M <sub>x</sub>	M <sub>y</sub>	M <sub>z</sub>
(deg F)	(lbs)	(lbs)	(lbs)	(lbs-in)	(lbs-in)	(lbs-in)
0.052293	0.006607	0.010645	0.036758	0.161195	0.153755	0.210417
1.031987	0.05307	0.051276	0.085669	0.536709	1.222713	0.466745
0.607624	0.055738	0.052328	0.085918	0.570636	1.397174	0.616958
0.450689	0.045906	0.049059	0.086972	0.53862	1.026095	0.468403
0.348539	0.045755	0.051782	0.08366	0.578349	1.053922	0.508274
0.276572	0.042995	0.0483	0.085049	0.559051	1.028053	0.509707
0.285201	0.04916	0.052719	0.080551	0.558426	1.144861	0.59767
0.272799	0.053363	0.046083	0.071848	0.498094	1.214323	0.528367
0.193657	0.242989	0.136604	0.526044	1.62988	4.113986	1.708122
0.247281	0.153566	0.121698	0.476657	1.763225	1.601178	1.430486
0.195927	0.067903	0.084158	0.174888	0.971999	1.714054	1.242236
0.592134	0.044893	0.042571	0.087666	0.439465	0.986015	0.518688
0.641245	0.048231	0.133048	0.117017	1.243457	1.425663	1.221911

Root Mean Square Deviation						
T	F <sub>x</sub>	F <sub>y</sub>	F <sub>z</sub>	M <sub>x</sub>	M <sub>y</sub>	M <sub>z</sub>
(deg F)	(lbs)	(lbs)	(lbs)	(lbs-in)	(lbs-in)	(lbs-in)
0.862894	0.102814	0.093974	0.083893	2.111402	2.370083	0.754186
0.979619	0.095654	0.09744	0.084387	2.209238	2.15101	0.771388
1.369186	0.100365	0.168661	0.17221	2.655933	2.315754	1.165246

Root Mean Square Deviation						
T	F <sub>x</sub>	F <sub>y</sub>	F <sub>z</sub>	M <sub>x</sub>	M <sub>y</sub>	M <sub>z</sub>
(deg F)	(lbs)	(lbs)	(lbs)	(lbs-in)	(lbs-in)	(lbs-in)
0.100625	0.015473	0.023679	0.094556	0.366936	0.411005	0.669823
0.097133	0.020989	0.026226	0.156956	0.385561	0.4408	0.763228
0.093382	0.023941	0.028534	0.124808	0.444535	0.556143	0.743576

**Fig. 11 Root Mean Square Deviation (RMSD) data for all runs**



## C. MATLAB

The following MATLAB script [6] was used for all Data Reduction and Graph Plotting:

### Listing 1 MATLAB Code for Data Analysis

```
%% AE 303 Lab 5: Full Model Aircraft Aerodynamic Analysis
```

```
% Author: Parham Khodadi
```

```
% Instructor: Xiaofeng Liu
```

```
clear; clc; close all;
```

```
%% Load Data
```

```
data = readtable('data.csv');
```

```
%% Constants and Conversions
```

```
inH2OtoPsi = 27.7076; % Convert psi to inH2O
```

```
psitoPa = 6894.76; % Convert psi to Pascals
```

```
FtoR = 459.67; % Convert from Fahrenheit to Rankine
```

```
S = 93.81; % Reference area (in^2)
```

```
c_bar = 3.466; % Chord length (in)
```

```
b = 27.066; % Span (in)
```

```
q = 7 / inH2OtoPsi; % Dynamic pressure in psi
```

```
R = 1716; % ft*lb / slug*R
```

```
%% Separate Runs
```

```
run1 = data(2:15,:); % Tail off (if used)
```

```
run2 = data(17:31,:); % Main test (Tail on)
```

```
run3 = data(33:47,:); % Gravity tare
```

```
run4 = data(49:53,:); % Aero tare
```

```
run5 = data(55:59,:); % Baseline
```

```
%% Define columns
```

```
% Columns in the data
```

```
col_alpha = 5; % Alpha
```

```
col_beta = 6; % Beta
```

```
col_T = 7; % T (Temperature)
```

```
col_Fx = 8; % Fx (Drag Force)
```

```
col_Fy = 9; % Fy (Side Force)
```

```
col_Fz = 10; % Fz (Lift Force)
```

```
col_Mx = 11; % Mx (Rolling Moment)
```

```
col_My = 12; % My (Pitching Moment)
```

```
col_Mz = 13; % Mz (Yawing Moment)
```

```
%% Data Correction (Run 2)
```

```
run2_corrected = run2;
```

```
% Create the correction data for when there is no Model (run4 - run3)
```

```
% but only for the Fx,Fy,Fz,Mx,My,Mz columns
```

```
modeloff_correction = run4;
```

```
for i = 3:5
```

```
    modeloff_correction(i,col_Fx:col_Mz) = modeloff_correction(i,col_Fx:col_Mz) - run5(i,c
```

```
end
```

```

% Correct run2 by  $run2\_corrected = run2 - run1 - (run4 - run3)$ 
% but only for the Fx,Fy,Fz,Mx,My,Mz columns
for i = 4:15
    if run2{i,col_beta} == 0
        run2_corrected(i,col_Fx:col_Mz) = run2_corrected(i,col_Fx:col_Mz) - run1(i-1,col_Fx:col_Mz)
    elseif run2{i,col_beta} == 5
        run2_corrected(i,col_Fx:col_Mz) = run2_corrected(i,col_Fx:col_Mz) - run1(i-1,col_Fx:col_Mz)
    elseif run2{i,col_beta} == 10
        run2_corrected(i,col_Fx:col_Mz) = run2_corrected(i,col_Fx:col_Mz) - run1(i-1,col_Fx:col_Mz)
    end
end

%% Data Correction (Run 3)
run3_corrected = run3;

% Correct run3 by  $run3\_corrected = run3 - (run4 - run3)$ 
% but only for the Fx,Fy,Fz,Mx,My,Mz columns
% Using the same (run4 - run3) data from run2_corrected
for i = 4:15
    if run3{i,col_beta} == 0
        run3_corrected(i,col_Fx:col_Mz) = run3_corrected(i,col_Fx:col_Mz) - modeloff_corrected(i-1,col_Fx:col_Mz)
    elseif run3{i,col_beta} == 5
        run3_corrected(i,col_Fx:col_Mz) = run3_corrected(i,col_Fx:col_Mz) - modeloff_corrected(i-1,col_Fx:col_Mz)
    elseif run3{i,col_beta} == 10
        run3_corrected(i,col_Fx:col_Mz) = run3_corrected(i,col_Fx:col_Mz) - modeloff_corrected(i-1,col_Fx:col_Mz)
    end
end

%% Calculate Aerodynamic Force and Moment Coefficients

% Angles of Attack
alpha_run2 = run2_corrected{4:15,col_alpha};

% Betas
beta_run2 = run2_corrected{4:15,col_beta};

% Lift Coefficient =  $L/(q*S) = F_z/(q*S)$ 
C_L = run2_corrected{4:15,col_Fz}/(q*S);

% Drag Coefficient =  $D/(q*S) = F_x/(q*S)$ 
C_D = run2_corrected{4:15,col_Fx}/(q*S);

% Pitching Moment Coefficient =  $M/(q*S*c_{bar}) = M_y/(q*S*c_{bar})$ 
C_M = run2_corrected{4:15,col_My}/(q*S*c_bar);

% Yaw Moment Coefficient =  $N/(q*S*b) = M_z/(q*S*b)$ 
C_N = run2_corrected{4:15,col_Mz}/(q*S*b);

%% Plot Aerodynamic Coefficients

mask = beta_run2 == 0; % for symmetric alpha sweep plots

% CL vs alpha
figure;

```

```

plot(alpha_run2(mask), C_L(mask), 'o-', 'LineWidth', 1.5);
xlabel('\alpha(deg)');
ylabel('C_L');
title('C_L vs \alpha (Tail-On)');
grid on;
print(gcf, '-depsc2', 'CL_vs_alpha.eps');

% CM vs alpha
figure;
plot(alpha_run2(mask), C_M(mask), 's-', 'LineWidth', 1.5);
xlabel('\alpha(deg)');
ylabel('C_M');
title('C_M vs \alpha (Tail-On)');
grid on;
print(gcf, '-depsc2', 'CM_vs_alpha.eps');

% CN vs beta (only use rows with alpha = 0 to isolate beta sweep)
mask_beta = run2_corrected(:, col_alpha) == 0;
beta_tailon = run2_corrected(mask_beta, col_beta);
CN_tailon = run2_corrected(mask_beta, col_Mz) / (q * S * b);

figure;
plot(beta_tailon, CN_tailon, 'd-', 'LineWidth', 1.5);
xlabel('\beta(deg)');
ylabel('C_N');
title('C_N vs \beta (Tail-On)');
grid on;
print(gcf, '-depsc2', 'CN_vs_beta.eps');

% CL vs CD
figure;
plot(C_D(mask), C_L(mask), '^-', 'LineWidth', 1.5);
xlabel('C_D');
ylabel('C_L');
title('C_L vs C_D (Tail-On)');
grid on;
print(gcf, '-depsc2', 'CL_vs_CD.eps');

%% Calculate Oswald Efficiency Factor (e)

% Aspect Ratio (from lab docs)
AR = (27.066)^2 / 93.81;

% Filter region for parabolic drag polar fit
CL_fit = C_L(mask);
CD_fit = C_D(mask);

% Only use CL in range [0.2, 0.8]
fit_mask = (CL_fit >= 0.3) & (CL_fit <= 0.6);
CL_range = CL_fit(fit_mask);
CD_range = CD_fit(fit_mask);

% Prepare design matrix: CD = CD0 + K*CL^2
X = [ones(length(CL_range), 1), CL_range.^2];

```

```

coeffs = X \ CD_range; % Linear least squares

CD0 = coeffs(1);
K = coeffs(2);

% Oswald efficiency factor
e = 1 / (pi * K * AR);

% Display results
fprintf('Parabolic Drag Polar Fit Results:\n');
fprintf('CD0=%%.4f\n', CD0);
fprintf('K=%%.4f\n', K);
fprintf('Oswald Efficiency Factor e=%%.4f\n', e);

% Plot the fit
CL_plot = linspace(min(CL_range), max(CL_range), 100);
CD_plot = CD0 + K * CL_plot.^2;

figure;
plot(CL_range, CD_range, 'bo', 'DisplayName', 'Data');
hold on;
plot(CL_plot, CD_plot, 'r-', 'LineWidth', 1.5, 'DisplayName', 'Fit');
xlabel('C_L');
ylabel('C_D');
title('Drag Polar Fit: C_D=C_{D,0}+K C_L^2');
legend('Location','northwest');
grid on;
print(gcf, '-depsc2', 'DragPolarFit_CD_vs_CL.eps');

%% Find Maximum Lift-to-Drag Ratio (C_L/C_D)

% Use only data where beta = 0 (mask already defined earlier)
CL_use = C_L(mask);
CD_use = C_D(mask);

% Compute L/D for all points
L_over_D = CL_use ./ CD_use;

% Find the maximum L/D and its index
[LoverD_max, idx_max] = max(L_over_D);

% Extract values
CL_max_LD = CL_use(idx_max);
CD_max_LD = CD_use(idx_max);
alpha_max_LD = alpha_run2(mask);
alpha_at_max_LD = alpha_max_LD(idx_max);

% Display results
fprintf('\nMaximum C_L/C_D:\n');
fprintf('Max L/D=%%.3f at alpha=%%.2f deg\n', LoverD_max, alpha_at_max_LD);
fprintf('C_L=%%.4f, C_D=%%.4f\n', CL_max_LD, CD_max_LD);

% Plot L/D vs alpha
figure;

```

```

plot(alpha_max_LD, L_over_D, 'ko-', 'LineWidth', 1.5);
xlabel('\alpha(deg)');
ylabel('C_L/C_D');
title('Lift-to-Drag Ratio vs \alpha (Tail-On)');
grid on;
print(gcf, '-depsc2', 'CL_over_CD_vs_alpha.eps');

%% Find Zero-Lift Angle of Attack and Lift Curve Slope

% Filter to linear region of CL vs alpha
alpha_lin = alpha_run2(mask);
CL_lin = C_L(mask);

% Choose linear range manually (usually -6 to +6 deg)
linear_mask = (alpha_lin >= -6) & (alpha_lin <= 6);
alpha_fit = alpha_lin(linear_mask);
CL_fit = CL_lin(linear_mask);

% Linear fit: CL = a * alpha + b
coeffs_CL = polyfit(alpha_fit, CL_fit, 1);
dCL_dalpha = coeffs_CL(1); % Slope
alpha_L0 = -coeffs_CL(2)/coeffs_CL(1); % Intercept = -b/a

% Display results
fprintf('\nLift Curve Fit Results:\n');
fprintf('dCL/d\alpha = %4f per deg\n', dCL_dalpha);
fprintf('Zero-lift angle of attack \alpha_L=0 = %2f deg\n', alpha_L0);

% Plot
alpha_plot = linspace(-6, 6, 100); % restrict to linear region only
CL_plot = polyval(coeffs_CL, alpha_plot);

figure;
plot(alpha_lin, CL_lin, 'bo', 'DisplayName', 'Data');
hold on;
plot(alpha_plot, CL_plot, 'r-', 'LineWidth', 1.5, 'DisplayName', 'Linear Fit');
xlabel('\alpha(deg)');
ylabel('C_L');
title('Lift Curve: Linear Fit to Find dC_L/d\alpha and \alpha_{L=0}');
legend('Location', 'northwest');
grid on;
print(gcf, '-depsc2', 'CL_vs_alpha_linear_fit.eps');

%% Find Maximum Lift Coefficient and Stall Angle

% Use clean alpha/C_L sweep from earlier (beta = 0)
alpha_trim = alpha_run2(mask);
CL_trim = C_L(mask);

% Find maximum CL
[CL_max, idx_max] = max(CL_trim);
alpha_stall = alpha_trim(idx_max);

```

```

% Display results
fprintf('\nMaximum Lift and Stall:\n');
fprintf('CL,max=%.4f at alpha=%.2f deg (stall angle)\n', CL_max, alpha_stall);

% Plot
figure;
plot(alpha_trim, CL_trim, 'bo-', 'LineWidth', 1.5);
hold on;
plot(alpha_stall, CL_max, 'r*', 'MarkerSize', 10, 'DisplayName', 'CL,max');
xlabel('\alpha (deg)');
ylabel('CL');
title('Maximum Lift Coefficient and Stall Angle');
grid on;
legend('show');
print(gcf, '-depsc2', 'CLmax_vs_alpha.eps');

%% Fit dCM/dalpha for Tail-On and Tail-Off

% --- Tail-On ---
alpha_tailon = alpha_run2(mask);
CM_tailon = C_M(mask);

% Use only linear range, e.g. [-6 to 6 deg]
fit_mask_tailon = (alpha_tailon >= -6) & (alpha_tailon <= 6);
alpha_fit_tailon = alpha_tailon(fit_mask_tailon);
CM_fit_tailon = CM_tailon(fit_mask_tailon);

coeffs_CM_tailon = polyfit(alpha_fit_tailon, CM_fit_tailon, 1);
dCM_dalpha_tailon = coeffs_CM_tailon(1);

% Define alpha and C_M for tail-off configuration
alpha_run3 = run3_corrected{4:15, col_alpha}; % Angle of attack (Run 3)
C_M_tailoff = run3_corrected{4:15, col_My} / (q * S * c_bar); % Pitching moment coefficient

% --- Tail-Off ---
alpha_tailoff = alpha_run3(mask); % run3 corrected alpha
CM_tailoff = C_M_tailoff(mask); % run3 corrected moment

fit_mask_tailoff = (alpha_tailoff >= -6) & (alpha_tailoff <= 6);
alpha_fit_tailoff = alpha_tailoff(fit_mask_tailoff);
CM_fit_tailoff = CM_tailoff(fit_mask_tailoff);

coeffs_CM_tailoff = polyfit(alpha_fit_tailoff, CM_fit_tailoff, 1);
dCM_dalpha_tailoff = coeffs_CM_tailoff(1);

% Display results
fprintf('\nPitching Moment Slope Results:\n');
fprintf('Tail-On: dCM/dalpha=%.4f\n', dCM_dalpha_tailon);
fprintf('Tail-Off: dCM/dalpha=%.4f\n', dCM_dalpha_tailoff);

% Plot for Tail-On
figure;
plot(alpha_tailon, CM_tailon, 'bs', 'DisplayName', 'Tail-On Data');

```

```

hold on;
plot(alpha_fit_tailon , polyval(coeffs_CM_tailon , alpha_fit_tailon), 'b-', 'LineWidth', 1.5);

% Plot for Tail-Off
plot(alpha_tailoff , CM_tailoff , 'rs', 'DisplayName', 'Tail-Off_Data');
plot(alpha_fit_tailoff , polyval(coeffs_CM_tailoff , alpha_fit_tailoff), 'r-', 'LineWidth', 1.5);

xlabel('\alpha(deg)');
ylabel('C_M');
title('Pitching_Moment vs \alpha : Tail-On vs Tail-Off');
legend('Location','northeast');
grid on;
print(gcf, '-depsc2', 'CM_vs_alpha_tailon_tailoff.eps');

%% Find dCN/dBeta for Tail-On and Tail-Off

% --- Tail-On ---
mask_beta_tailon = run2_corrected(:, col_alpha) == 0; % constant alpha = 0
beta_tailon = run2_corrected(mask_beta_tailon , col_beta);
CN_tailon = run2_corrected(mask_beta_tailon , col_Mz) / (q * S * b);

% Linear fit
coeffs_CN_tailon = polyfit(beta_tailon , CN_tailon , 1);
dCN_dbeta_tailon = coeffs_CN_tailon(1);

% --- Tail-Off ---
mask_beta_tailoff = run3_corrected(:, col_alpha) == 0;
beta_tailoff = run3_corrected(mask_beta_tailoff , col_beta);
CN_tailoff = run3_corrected(mask_beta_tailoff , col_Mz) / (q * S * b);

% Linear fit
coeffs_CN_tailoff = polyfit(beta_tailoff , CN_tailoff , 1);
dCN_dbeta_tailoff = coeffs_CN_tailoff(1);

% Display results
fprintf('\nYawing_Moment_Slope_Results:\n');
fprintf('Tail-On: dCN/ dbeta = %.4f\n', dCN_dbeta_tailon);
fprintf('Tail-Off: dCN/ dbeta = %.4f\n', dCN_dbeta_tailoff);

% Plot
figure;
plot(beta_tailon , CN_tailon , 'bd', 'DisplayName', 'Tail-On_Data');
hold on;
plot(beta_tailoff , CN_tailoff , 'rd', 'DisplayName', 'Tail-Off_Data');

plot(beta_tailon , polyval(coeffs_CN_tailon , beta_tailon), 'b-', 'LineWidth', 1.5, 'DisplayName', 'Tail-On_Fit');
plot(beta_tailoff , polyval(coeffs_CN_tailoff , beta_tailoff), 'r-', 'LineWidth', 1.5, 'DisplayName', 'Tail-Off_Fit');

xlabel('\beta(deg)');
ylabel('C_N');
title('Yawing_Moment vs \beta : Tail-On vs Tail-Off');
legend('Location','northwest');
grid on;
print(gcf, '-depsc2', 'CN_vs_beta_tailon_tailoff.eps');

```