

DESIGN AND ANALYSIS OF BELLWETHER:  
A FLYING WING FEATURING THE BELL-SHAPED SPANLOAD

By

Scott Weekley

Bachelor of Science in Mechanical and Aerospace Engineering  
Oklahoma State University  
Stillwater, OK  
2019

Submitted to the Faculty of the  
Graduate College of the  
Oklahoma State University  
in partial fulfillment of  
the requirements for  
the Degree of  
Master of Science  
May, 2021

---

DESIGN AND ANALYSIS OF BELLWETHER:  
A FLYING WING FEATURING THE BELL-SHAPED SPANLOAD

Thesis Approved:

Dr. Jamey D. Jacob

---

Thesis Advisor

Dr. Ryan Paul

---

Dr. Andy Arena

---

## ACKNOWLEDGMENTS

Author recognizes aircraft design partner and friend, Cole Kelly, for being an integral part of this effort from its inception. From design to modeling to analysis to concepts in aircraft construction, this research effort would not have been possible without his influence. Thanks to Hill Jordan, Alex Langseth, and Boone Miller for the great work on Leading-Edge Flow Sensing concepts and designs.

Thanks to Dr. Andy Arena for constructive discussions and guidance on the topic of aircraft stability & control.

Author credits Dr. Aaron Alexander for expertise in computational fluid dynamics with STAR-CCM+.

Thank you to Dr. Ryan Paul for joining the team and becoming a massive influence in the understanding of advanced aerodynamics and applications of vortex lattice methods as well as helping to earn the NASA grant for fabrication of the aircraft designed in this thesis.

Author gives thanks to advisor Dr. Jamey Jacob for support financially and intellectually and for providing the tools and housing for the aircraft construction at his research lab, *Excelsior*.

Special thanks to family for the undying support and unconditional love.

---

Acknowledgments reflect the views of the author and are not endorsed by committee members or Oklahoma State University.

---

Name: SCOTT WEEKLEY

Date of Degree: MAY, 2021

Title of Study: DESIGN AND ANALYSIS OF BELLWETHER: A FLYING WING FEATURING THE BELL-SHAPED SPANLOAD

Major Field: MECHANICAL AND AEROSPACE ENGINEERING

Abstract:

This body of work details the design and analysis, as well as discussion of requirements for eventual fabrication and flight testing, of a flying-wing aircraft. The design presented is focused primarily on matching the lift distribution to a bell curve and its effect on aerodynamic performance for this aircraft configuration. Various analysis methods of the software-driven design are justification for the wing shape: panel method, vortex lattice method, and computational fluid dynamics. The key phenomenon examined by the analysis methods, and to be measured from the on-board inertial measurement unit, is proverse yaw. The unique wing twist used to match the desired lift distribution encourages a favorable longitudinal force differential by the aileron-deflected wingtips engulfed in upwash flow to induce thrust at the wingtips. This yawing motion as a consequence of roll promotes a naturally coordinated turn without the demand for rudder authority. The observation of proverse yaw coupled with vorticity scenes, resultant from computational fluid dynamics, synchronizes this work with emerging research suggesting the bell spanload is the apt model for avian flight. Additionally, this study parameterizes lift distribution to investigate, through highly-computational methods, the centralization of lift and the aerodynamic effect with the change in shape of the flow field. The inboard movement of the vortex cores with centralized lift is found to be directly proportional to the proverse yaw control power responsible for coordinated flight without an auxillary yaw device. Level and banked-flight analyses are used to weight the tradeoffs between the creation of favorable yaw and the aerodynamic efficiency of the lift distribution in steady operation. The significance of this validation through computational methods may be used to justify reducing the size of a vertical tail on more traditional aircraft configurations or to consider using a bell-shaped lift distribution on flying-wing aircraft for greater efficiency in banked turns.

## TABLE OF CONTENTS

Chapter	Page
<b>I. INTRODUCTION</b> . . . . .	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Background . . . . .	2
<b>II. LITERATURE REVIEW AND PREVIOUS WORK</b> . . . . .	<b>4</b>
2.1 Literature Review . . . . .	4
2.1.1 Origin and Implementation of Elliptical Lift Distribution . . . . .	4
2.1.2 Discovery and Applications of Bell-Shaped Lift Distribution . . . . .	6
2.1.3 Avian Flight Formation . . . . .	12
2.1.4 Literature Review - In Summary . . . . .	15
<b>III. DESIGN METHODOLOGY</b> . . . . .	<b>18</b>
3.1 Span Loading Design Target . . . . .	20
3.1.1 Span Load Analysis Using Vortex Lattice Methods (VLM) . . . . .	21
3.2 Airfoil Analysis and Selection . . . . .	22
3.3 Aircraft Modeling . . . . .	23
<b>IV. COMPARING VLM AND CFD SOLUTIONS FOR THE BELL- WETHER WINGS</b> . . . . .	<b>25</b>
4.1 Wing-Only Analysis of Bellwether in Vortex Lattice Methods . . . . .	25
4.2 Wing-Only Analysis of Bellwether in STAR-CCM+ . . . . .	26
4.3 Wing-Only Comparison of Analysis Methods . . . . .	34
4.3.1 Mathematical Justification for Curve Scaling . . . . .	36

Chapter	Page
<b>V. CFD ANALYSIS OF BELLWETHER . . . . .</b>	<b>40</b>
5.1 Mesh Development . . . . .	40
5.2 Level, Steady Flight Performance . . . . .	41
5.3 Banked Turn Performance . . . . .	44
<b>VI. AVIAN-INSPIRED AIRCRAFT DESIGN AND PERFORMANCE 48</b>	<b>48</b>
6.1 Longitudinal Stability and Performance . . . . .	48
6.2 Lateral Stability of Bellwether . . . . .	51
6.3 CAD Modeling, Internal Design, and Fabrication Techniques . . . . .	53
<b>VII. FUTURE WORKS . . . . .</b>	<b>56</b>
7.1 Leading-Edge Flow Sensing . . . . .	56
7.1.1 Theory . . . . .	57
7.1.2 Design and Integration . . . . .	57
7.1.3 Wind Tunnel Demonstration System . . . . .	58
<b>VIII. CONCLUSION . . . . .</b>	<b>60</b>
8.1 Summary . . . . .	61
<b>REFERENCES . . . . .</b>	<b>62</b>
<b>APPENDICES . . . . .</b>	<b>64</b>
0.1 Aircraft CAD Drawings and Renderings . . . . .	64
0.2 Data Plots . . . . .	78
0.3 CFD Mesh Scenes . . . . .	86
0.4 Aircraft Mold Design and Fabrication . . . . .	89
0.5 Experimental Models and Images . . . . .	92
0.6 CFD Streamlines and Vorticity Scenes . . . . .	100

Chapter	Page
0.7 Flying Wing Macro . . . . .	114

## LIST OF FIGURES

Figure		Page
1	Elliptical Lift Distribution (a) v. Bell-Shaped Lift Distribution (b) [2] . .	2
2	Change in Regions of Downwash from Elliptical (Left) to Bell-Shaped (Right) Lift Distribution [2] . . . . .	3
3	A finite wing, considered as due to vortices replacing the wing [16] . . . .	5
4	Change in shape of vortex ribbons at great distances behind the wing [16]	5
5	$\mu$ Variation in Relation to Aerodynamic Characteristics Ratios [14] . . .	8
6	Induced drag ratio compared to the circulation ratio [14] . . . . .	8
7	Circulation distribution of various spanloads [14] . . . . .	9
8	Downwash velocity distribution of various spanloads [14] . . . . .	9
9	Common cranes (grus grus) flying in formation over lake Fehér near Sándorfalva, Hungary [14] . . . . .	14
10	Aerodynamic Parameter Percent Change Between $Y/b = 0.148$ and $-0.444$ for the Trailing Wing Compared to the Leading Wing [14] . . . . .	15
11	Optimal Position and Beneficial Range from Avian, ELD, and BSLD re- search [14] . . . . .	16
12	Methodology Flow Chart . . . . .	19
13	Lift Distributions to be Matched . . . . .	21
14	Region of Downwash Observed by XFLR5 Analysis . . . . .	22
15	Polars of Interpolated Airfoils . . . . .	23
16	Interpolated Airfoil Cross Sections . . . . .	23
17	Planform Sections . . . . .	24
18	Trefftz Plane Results from AVL . . . . .	26
19	Flow Domain used for the Wing-Only Analysis of Bellwether . . . . .	27
20	Physics Settings for the Wing-Only Analysis of Bellwether . . . . .	28
21	Mesh Scene of Wing-Only Model . . . . .	29
22	Planform Mesh Scene of Wing-Only Model . . . . .	29
23	Prism Layer Mesh Scene of Wing-Only Model . . . . .	30
24	Bellwether Wing-Only CFD - Coefficients Monitor . . . . .	30
25	Bellwether Wing-Only Unsteady RANS - Coefficients Monitor . . . . .	31
26	Bellwether Wing-Only CFD - Forces Monitor . . . . .	31
27	Bellwether Wing-Only Unsteady RANS - Forces Monitor . . . . .	32
28	Bellwether Wing-Only CFD - Wall $Y^+$ . . . . .	32
29	Mesh Resolution Percent Error . . . . .	33
30	Bellwether Wing-Only CFD - Vorticity Vector Top View . . . . .	33



Figure		Page
31	Bellwether Wing-Only CFD - Wake Vorticity Scalar Scene . . . . .	33
32	Bellwether Wing-Only CFD - Vortex Core Scene . . . . .	34
33	Comparison of Lift Curve Slopes of the Wing-Only Bellwether Model for VLM and CFD Analyses . . . . .	35
34	Constrained Streamlines on Bellwether Wing-Only Planform View . . . .	35
35	Comparison of Vortex Lattice Methods' $c^*CL$ Curves to a Scaled Curve of $dL$ from CFD at a Common Lift Coefficient . . . . .	36
36	Longitudinal Mesh at the Root Airfoil . . . . .	41
37	Horizontal Mesh with Wake Refinement Enabled . . . . .	41
38	Bellwether Wall $Y^+$ . . . . .	42
39	CFD-Matched Lift Distributions . . . . .	42
40	Comparison of Lift-to-Drag Ratios of the Various Wing Designs . . . . .	43
41	Bellwether Vorticity Vector Scene . . . . .	44
42	Elliptical Model Vorticity Vector Scene . . . . .	45
43	Yaw Control Power per Unit Roll from 20 Anti-Symmetric Aileron Deflection	46
44	Yaw Moment Monitor by Iteration of Steady RANS Solver in STAR-CCM+	46
45	Aileron Mesh Scene on Longitudinal Plane . . . . .	47
46	Bellwether - Lift Curve and L/D Developed by Alpha Sweep Analysis in STAR-CCM+ . . . . .	49
47	Propulsor Placement in Line with Center of Gravity ISO . . . . .	50
48	Longitudinal Static Stability from CM v. Alpha Plot for Varying Center of Gravity Locations . . . . .	51
49	Coefficients of Roll ( $C_l$ ) and Yaw ( $C_n$ ) with Respect to Sideslip Angle ( $\beta$ )	52
50	Geometric Twist Angles for Each Spanwise Airfoil . . . . .	53
51	CAD Model of Bellwether Internals . . . . .	54
52	Bellwether - Rear View . . . . .	55
53	Bellwether - Front View . . . . .	55
54	Port Wing Mold Assembly and the Integration of Leading-Edge Flow Sensors	58
55	Sensing Sleeve Mounted in the Endeavor Wind Tunnel . . . . .	59
56	Bellwether Side View - Avian Inspiration . . . . .	64
57	Propulsor Placement in Line with Center of Gravity . . . . .	73
58	Bellwether Planform View . . . . .	73
59	Bellwether1 . . . . .	74
60	Bellwether2 . . . . .	75
61	Bellwether3 . . . . .	76
62	Bellwether4 . . . . .	76
63	Bellwether5 . . . . .	77
64	Bellwether6 . . . . .	77
65	Lift and Drag Monitor of Bellwether - Unsteady RANS results . . . . .	78
66	Residual Plot . . . . .	79

Figure		Page
67	Drag Distribution in Roll . . . . .	79
68	Cosine Model - Drag Distribution in Roll . . . . .	80
69	Cosine Model - Lift Distribution in Roll . . . . .	80
70	Modern Bell - Drag Distribution in Roll (bin direction is backwards on this plot) . . . . .	81
71	Modern Bell - Lift Distribution in Roll (bin direction is backwards on this plot) . . . . .	81
72	Prandtl Bell - Drag Distribution in Roll . . . . .	82
73	Prandtl - Lift Distribution in Roll . . . . .	82
74	Elliptical - Drag Distribution in Roll (bin direction is backwards on this plot) . . . . .	83
75	Elliptical - Lift Distribution in Roll . . . . .	83
76	Combine Alpha Sweep . . . . .	84
77	Distributions at Common CL of All Analysis Methods . . . . .	85
78	Wake Refinement . . . . .	86
79	Aileron Mesh 1 . . . . .	86
80	Aileron Mesh 2 . . . . .	87
81	Aileron Mesh 3 . . . . .	87
82	Aileron Mesh 4 . . . . .	88
83	Fuselage Mold . . . . .	89
84	Inboard Wing Mold . . . . .	89
85	Inboard Wing Mold - Top View . . . . .	89
86	Outboard Wing Mold . . . . .	90
87	Outboard Wing Mold - Top View . . . . .	90
88	Mold 1 . . . . .	90
89	Mold 2 . . . . .	91
90	Mold3 . . . . .	91
91	System of Components for Leading-Edge Flow Sensing Sleeve . . . . .	92
92	Wind Tunnel Setup for 3D-Printed Bellwether Model . . . . .	93
93	Wind Tunnel Flow Visualization . . . . .	94
94	Wind Tunnel 1 . . . . .	95
95	Wind Tunnel 2 . . . . .	95
96	Wind Tunnel 3 . . . . .	96
97	Wind Tunnel 4 . . . . .	96
98	Wind Tunnel 5 . . . . .	97
99	Water Tunnel 1 . . . . .	97
100	Water Tunnel 2 . . . . .	98
101	Water Tunnel 3 . . . . .	98
102	Water Tunnel 4 . . . . .	99
103	Water Tunnel 5 . . . . .	99
104	Bellwether Vorticity 1 . . . . .	100
105	Bellwether Vorticity 2 . . . . .	101
106	Bellwether Vorticity 3 . . . . .	102

Figure		Page
107	Bellwether Vorticity 4 . . . . .	103
108	Bellwether Vorticity 5 . . . . .	103
109	Elliptical Vorticity 1 . . . . .	104
110	Elliptical Vorticity 2 . . . . .	105
111	Elliptical Vorticity 3 . . . . .	106
112	Cosine Vorticity 1 . . . . .	107
113	Cosine Vorticity 2 . . . . .	108
114	Cosine Vorticity 3 . . . . .	109
115	Modern Bell Vorticity 1 . . . . .	110
116	Modern Bell Vorticity 2 . . . . .	111
117	Bellwether Streamlines 1 . . . . .	111
118	Bellwether Streamlines 2 . . . . .	112
119	Bellwether Streamlines 3 . . . . .	112
120	Bellwether Vector Scene . . . . .	113
121	Bellwether Vortex Core . . . . .	113

## CHAPTER I

### INTRODUCTION

#### 1.1 Motivation

The shape of the lift distribution across a finite wing greatly influences aerodynamic performance, in terms of induced drag, and the need for structural rigidity at the wing root. In 1921, Ludwig Prandtl presented the classical result that minimum induced drag is achieved for a finite wing when the lift distribution is elliptical in shape [16]. This understanding has prevailed and modern wing designs are still commonly compared to the elliptical lift distribution to assess efficiency with the common span efficiency factor,  $e$ . In 1933, Prandtl revised his previous work after recognizing that the elliptical distribution may not be the most efficient if a wing was not subject to a span constraint. Rather, a bell-shaped, spanwise loading results in lower induced drag for a given lifting capacity [11] if the designer accepts a higher wingspan. This optimization was achieved in the 1933 paper by applying a bending moment constraint rather than assuming a fixed span, effectively increasing aspect ratio rather than shaping the lift distribution to achieve an  $e$  of 1. A beneficial consequence of the bell spanload, not recognized by Prandtl at the time, is proverse yaw[2] which can obviate the need for a vertical stabilizer to provide a means for turn coordination in flying wing configurations further reducing overall aircraft drag. In fact, the vertical stabilizer may be omitted altogether if the resulting aircraft is immune from flight conditions where asymmetric thrust would be encountered (i.e., single-engine) and has no need to operate off of paved landing surfaces in crosswind conditions where alignment of the aircraft's longitudinal axis to the runway direction is desired prior to touchdown. Albion Bowers[2] recently lead a group of researchers at NASA Armstrong through an experiment to validate the purported efficiencies of a flying-wing glider featuring the 1933 Bell spanload. The researchers observed the favorable proverse yaw in data collected from their flight test experiments, and provided insights to the aerodynamics community synchronizing aerodynamic theory and observations of avian formation flight.

This body of work describes progress towards the author's design, manufacturing, and plans for flight testing of Bellwether: a flying wing platform featuring the 1933 Prandtl Bell lift distribution. Bellwether is intended to be used for atmospheric data collection while demonstrating minimum induced drag. The Bell spanload is particularly well-suited to this application for its induced drag characteristics which will increase platform endurance, while keeping the centroid of the aerodynamic load close to the wing root which serves to limit bending moments that must be passed through the center section. Bellwether has been de-

signed using a combination of medium- and high-fidelity aerodynamic analysis techniques. Vortex Lattice Methods (VLM) were used to design the geometric twist distribution to match the desired span loading and to analyze local section aerodynamics towards selecting appropriate airfoils. Following lofting of the flying wing with the desired twist distribution, in SolidWorks, the geometry was extracted for mesh generation. Numerous simulations of computational fluid dynamics (CFD) have been performed to validate the medium-fidelity analysis techniques, allowing for an interesting comparison to be made between VLM and CFD. This thesis compares the bell-shaped lift distribution (BSLD) to the industry-preferred elliptical distribution (ELD), and it rigorously investigates the performance of other spanloads as a parametric study on lift distribution and its affect on aerodynamic performance. These differences are quantified for flying-wing aircraft configurations. Variance in geometric wing twist are used to produce the desired distributions, and the different aircraft models are evaluated by various computational methods. This work provides insight to the aircraft design process, and it details the methods by which the experimental aircraft design is generated and analyzed.

## 1.2 Background

This design follows in the footsteps of the Preliminary Research Aerodynamic Design to Lower Drag, or Prandtl-D platform developed at NASA Armstrong[2]. A combination of geometric and aerodynamic twist was utilized to ensure the lift distribution per unit span conformed to the bell shape proposed by Prandtl as a more efficient alternative to the elliptical spanload when a span constraint is not present. Past research gives the expectation that the design will realize an up to 11.1% decrease in induced drag while increasing the span by 22.5% for a given structural weight, which this research effort intends to verify through more robust aerodynamic analysis than previous efforts have utilized.

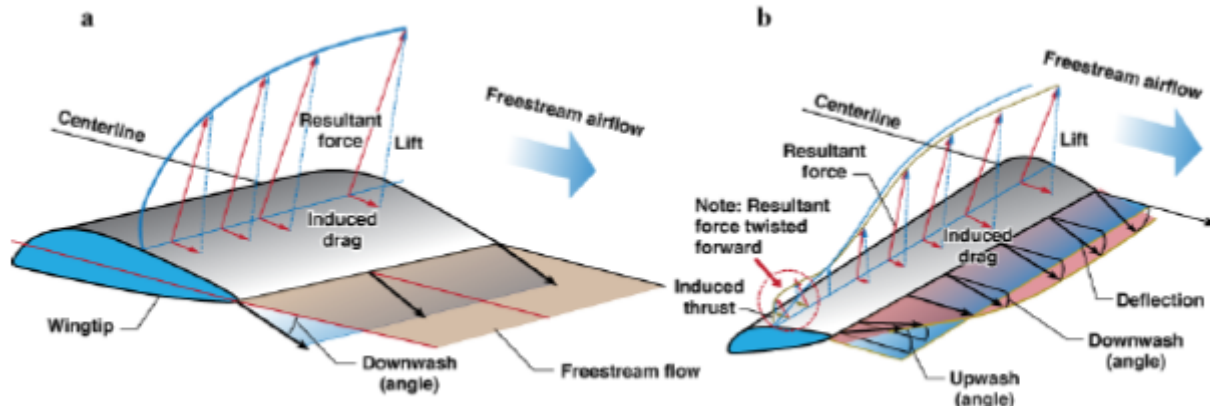


Figure 1: Elliptical Lift Distribution (a) v. Bell-Shaped Lift Distribution (b) [2]

*Figure 1* illustrates how the wing-tip twist to a negative angle of attack in the region of upwash can induce thrust, resulting in the favorable proverse yaw characteristic. Once

ailerons are deflected anti-symmetrically, the lift vector on the high-lift wing will benefit from the upwash, and it can be thought of as tilting forward. Due to the negative angle of attack, the opposite wing tip is producing a negative lift vector, when the aileron is actuated upward, tilting in the direction of the freestream, and thus, it fails to utilize the region of upwash. This resultant force differential on the wingtips shows an ability for proverse yaw in roll.

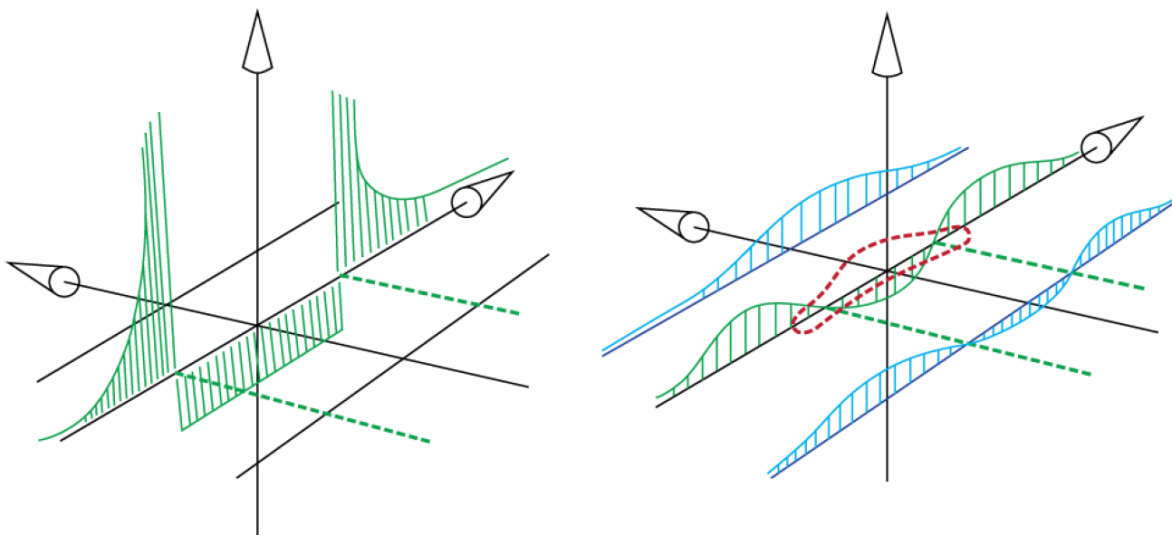


Figure 2: Change in Regions of Downwash from Elliptical (Left) to Bell-Shaped (Right) Lift Distribution [2]

The movement of the upwash region inboard (*Figure 2*) matches what is believed to be exhibited by bird flight. This has been examined by the formation of flight. To achieve an optimal formation, birds overlap their wing tips such that they are capturing the entire region of upwash. The amount of overlap is congruent with the inboard region of upwash produced from the application of Prandtl's revision of the most efficient spanwise lift distribution [2].

The lift-to-drag ratios, flow field, and roll-yaw characteristics are evaluated in this research by computational methods for four different lift distributions (later defined), and they will be used to justify the shape of the finalized wing design for optimal aerodynamic performance. Before outlining the design methodology and procedures, a literature review is provided for a comprehensive understanding of the emerging research on the application of this newly-analyzed lift distribution and its relationship to avian flight observations.

## CHAPTER II

### Literature Review and Previous Work

#### 2.1 Literature Review

This section journeys through a brief history of the industry-standard lift distribution before exploring examples and research in support of an alternative. Discussion of the analysis tools, methods, and results of various research help us understand what is known and how to best contribute to the advancement of this alternative lift distribution. Research examining avian flight formation provides a biomimetic link between bird flight and aircraft design. This is used to suggest that birds benefit from the alternative lift distribution both in formation flight and performing perfectly coordinated turns with favorable roll-yaw characteristics. The collective theories and experiments are used to inform the aircraft design detailed in this thesis.

##### 2.1.1 Origin and Implementation of Elliptical Lift Distribution

For a century, aircraft have been designed using the same theory presented by Ludwig Prandtl in "*Applications of Modern Hydrodynamics to Aeronautics*" in 1921 [16], where he describes the minimum drag solution for a finite wing is to design the wing such that the distribution across the span takes the shape of an ellipse. This major theoretical installment discusses many topics and applications. The portion most pertinent to modern aerodynamics is the discussion on the theory of lift as it pertains to a monoplane. The theory described in the published manuscript relies on the lifting-line approximation which describes a lifting surface as a vortex-inducing line orthogonal to the free stream. This simplification leads to the horseshoe vortex representation (*Figure 3*). The lifting line generates a bound vortex inducing a downward component to freestream velocity. It is assumed that the lift is produced uniformly across this line which means the total circulation is produced at the ends of the lifting line creating the free vortices at the wingtips. As is easily depicted in (*Figure 3*), the flow direction is downward between the vortices at the ends of the lifting line, and upwash occurs outside of the horseshoe. This can also be viewed as a vortex ribbon (*Figure 4*) formed from the curling of the flow at the wingtips in reaction to the pressure differential caused on the wing. The high-pressure fluid flows to the region of low pressure on top of the wing resulting in a curling flow in the wake of a finite lifting surface. The true nature of the wake vortices is to dissipate and gravitate at a distance from the wings, but since the greatest influence on the wing occurs nearest the body, the underpinnings of Prandtl's theory makes calculations reliant upon considering all vortex filaments as running in a straight line opposite the aircraft's direction of motion. The simplification makes the problem linear and solvable.

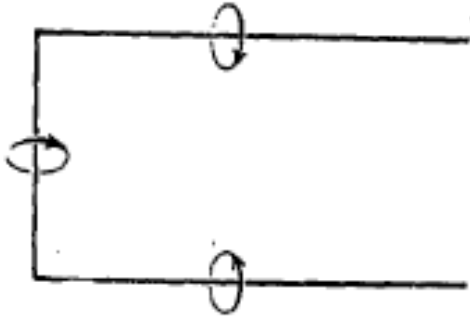


Figure 3: A finite wing, considered as due to vortices replacing the wing [16]

The influence of the vortical structure to the lifting surface increases the effective angle of attack and the induced drag as the ends of the vortex sheet are positioned closer together, which is to say the aspect ratio is inversely proportional to the induced drag (the next section will utilize this idea by the reframing of design constraints to achieve a new analytical solution). Prandtl uses this basis for the derivation of an elliptical shape of circulation distribution across the span having fixed wingspan, total lift, and velocity (derivation in text [16]). Vorticity and its relationship to the flow field and aerodynamic performance is an ongoing theme in this thesis that will draw qualitative and quantitative comparisons to the general overview provided in this section.

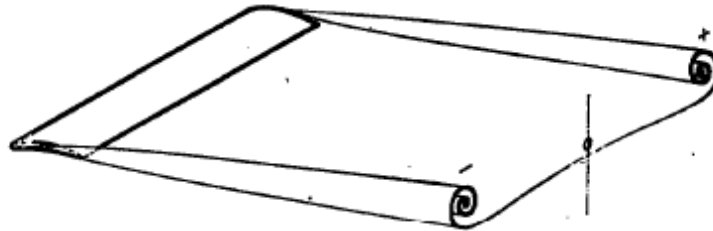


Figure 4: Change in shape of vortex ribbons at great distances behind the wing [16]

Prandtl's early work also applies the lifting-line theory to multiple wing configurations suggesting the elliptical lift distribution for a two-wing system.

Researchers at Pisa University, Italy [9] reference the Prandtl's Best Wing System [8], derived from Ludwig Prandtl's 1924 work "*Induced Drag of Multiplanes*" [17], in their effort to optimize a box wing configuration for minimal induced drag. Inspired by Prandtl's theories of a century prior, the Italian research team aim to solve a modern optimization problem in the drag reduction of large, transport aircraft. The designs of which have been refined over decades to a point of near stagnation. Noting that a 1% reduction in drag for a large aircraft can save 400,000 liters of fuel and 5,000 kilograms of noxious emissions annually [20], improvements in aerodynamic efficiency have great societal benefits. The researchers at Pisa University present models produced by the in-house code MSD (Multiple Shape Design), and analysis through CFD, to introduce an automated solution to optimization of box-wing aircraft configurations. Two-dimensional curves and airfoils, in the form of .Dat files, are used to construct all of the main features of the "PrandtlPlane" by use of the MSD code:



fuselage, wings, holes, fillets, etc. The swept box wing was analyzed in CFD at transonic mach regimes to simulate flight conditions observed by most transport aircraft. The fuselage is comparable to the Airbus A380, sporting a wingspan of 78 meters, so it is noted that the reduction of the span of the PrandtlPlane to 70 meters produces a weight reduction of 30%, though it is not mentioned how this is reconciled with the aft wing of equivalent span. The researchers proceed with this design methodology for a smaller unmanned aircraft which was fabricated as a scaled model for wind-tunnel testing. Though the results indicated longitudinal static stability, the aircraft does not begin to produce lift until around 6 angle of attack, and the trim point is shown graphically to occur at 12. These results are as unconventional as the aircraft configuration, but the aerodynamic efficiency maintains satisfactory. This research does not provide evidence of an aerodynamic advantage to the industry standard in the implementation of the Prandtl’s Best Wing Design for a box-wing aircraft, but it offers a design option to be further explored in the search for reduction of induced drag. 90% of total drag is due to friction and induced drag, which depends heavily on the distribution of lift across the wing span. It is posited in this body of work that, for today’s large transport aircraft, ”no significant induced drag reduction is now possible” due to the decades of aircraft design optimization. However, a new branch of research into alternative lift distributions, spurred by none other than Ludwig Prandtl, may provide evidence to the contrary.

### 2.1.2 Discovery and Applications of Bell-Shaped Lift Distribution

In 1933, Ludwig Prandtl issued a publication titled “*On Wings with the Least Induced Drag*” in the “*Journal of Aviation Engineering and Motorized-Airship Aeronautics*” amending the case that the elliptical distribution is the optimum distribution to minimize induced drag for all cases [11]. He notes that it is the desired solution when the aircraft designer has prescribed gross weight and wingspan as design constraints. Prandtl derives an analytic solution for an alternative curve that initializes gross lift (*Equation 2.1.2*) and moment of inertia of the lift distribution (*Equation 2.1.3*) in the determination of induced drag (*Equation 2.1.1*) to achieve the minimal induced drag solution. This solution resembles a bell curve which gradually reduces the sectional lift produced to zero at the wingtips.

$$D_i = \rho \int_{-s}^s \Gamma w dy \quad (2.1.1)$$

$$L = \rho V \int_{-s}^s \Gamma dy \quad (2.1.2)$$

$$I = Lr^2 = \rho V \int_{-s}^s \Gamma y^2 dy \quad (2.1.3)$$

For the listed equations  $\rho$  is density,  $V_\infty$  is freestream velocity,  $\Gamma$  is circulation,  $w$  is downwash,  $y$  is the non-dimensional spanwise location with 0 being the wing centerline,  $s$  is the semispan, and  $r$  is the radius of gyration for an ELD wing.

*Equation 2.1.2* and *Equation 2.1.3* are transformed into *Equation 2.1.4* and *Equation 2.1.5*, respectively, where the parameterized spanload,  $b$ , is introduced.  $\Gamma_0$  is the circulation

at the center of the wing, and  $\mu$  is a ratio of circulations across the span (*Equation 2.1.6*) for  $\mu = 0$  representing the ELD.

$$L = \frac{\pi}{4}\rho b V_\infty \Gamma_0 \left(1 - \frac{\mu}{4}\right) \quad (2.1.4)$$

$$Lr^2 = \frac{\pi}{64}\rho b^3 V_\infty \Gamma_0 \left(1 - \frac{\mu}{2}\right) \quad (2.1.5)$$

$$\mu = -\frac{\Gamma_2}{\Gamma_0} \quad (2.1.6)$$

$$\Gamma = \left(\Gamma_0 - \Gamma_2 \left(\frac{y}{s}\right)^2\right) \sqrt{1 - \left(\frac{y}{s}\right)^2} \quad (2.1.7)$$

After allowing span to become a parameter, *Equation 2.1.8* is substituted into *Equation 2.1.4* to isolate the circulation at the center of the wing as a function of the gross lift, the freestream velocity, and the circulation ratio across the span (*Equation 2.1.9*).

$$b = 4r \sqrt{\frac{1 - \frac{\mu}{4}}{1 - \frac{\mu}{2}}} \quad (2.1.8)$$

$$\Gamma_0 = \frac{L}{\pi \rho V_\infty r} \sqrt{\frac{1 - \frac{\mu}{4}}{1 - \left(\frac{\mu}{2}\right)^3}} \quad (2.1.9)$$

Transferring these variables to express induced drag yields *Equation 2.1.10*.

$$D_i = \frac{L^2}{8\pi\rho V_\infty^2 r^2} \frac{\left(1 - \frac{\mu}{2}\right)\left(1 - \frac{\mu}{2} + \frac{\mu}{4}\right)}{1 - \left(\frac{\mu}{4}\right)^3} \quad (2.1.10)$$

From the table shown in *Figure 5* it is apparent that the minimum induced drag occurs at  $\mu$  equal to unity, but the function continues to decrease past this local minimum (*Figure 6*). However, the function loses its rational meaning beyond this inflection point because the negative lift at the wingtips would cause a negative bending moment (M). Since the prescribed structural weight is a function of bending moment, and the negative bending moment does not result in a negative structural weight, the equation fails to hold the relationship among these terms. Therefore, the largest reasonable value occurs at a  $\mu$  of 1, which is also the point of lowest induced drag.

*Figure 5* also shows a 22.47% increase in span (1.0000 to 1.2247) from the elliptical distribution at  $\mu = 0$  to the bell-shaped distribution  $\mu = 1$ . The change in shape and increase in span for the variation in circulation ratio is plotted in *Figure 7*.

$\mu = 1$  is the analytical solution for minimal induced drag, but another artifact of relaxing the wingspan constraint is the variation of downwash velocity distribution with change in circulation ratio (*Figure 8*). It can be seen that a negative downwash (upwash) occurs at the wingtips of circulation ratios of 0.5 and greater. For the bell-shaped lift distribution, this location occurs at the 70.4% semispan. The upwash at the wingtip can provoke a forward force, induced thrust. The variation in the downwash velocity, and its transition to upwash, along with the actuation of induced thrust to create a favorable moment on an aircraft have

	$f(\mu)_b$	$f(\mu)_{\Gamma_0}$	$f(\mu)_{D_i}$
$\mu$	$\sqrt{\frac{1 - \frac{\mu}{4}}{1 - \frac{\mu}{2}}}$	$\sqrt{\frac{1 - \frac{\mu}{2}}{\left(1 - \frac{\mu}{4}\right)^3}}$	$\frac{\left(1 - \frac{\mu}{2}\right)\left(1 - \frac{\mu}{2} + \frac{\mu^2}{4}\right)}{\left(1 - \frac{\mu}{4}\right)^3}$
0.00	1.000000	1.000000	1.000000
0.25	1.035098	1.030498	0.945778
0.50	1.080123	1.058080	0.909621
0.75	1.140175	1.079456	0.892126
1.00	1.224745	1.088662	0.888889

Figure 5:  $\mu$  Variation in Relation to Aerodynamic Characteristics Ratios [14]

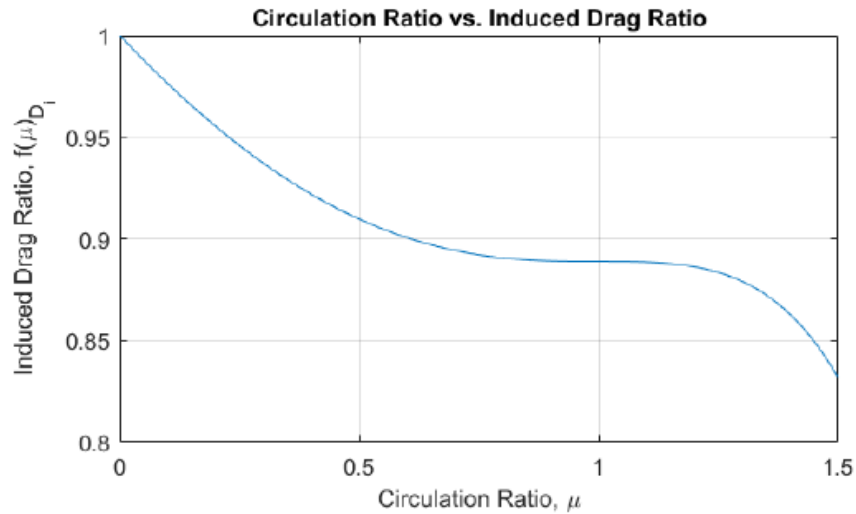


Figure 6: Induced drag ratio compared to the circulation ratio [14]

implications on the aircraft design presented in this thesis and are the central themes to this body of work.

In 1950, Robert T. Jones creates his own solution to the minimal-induced-drag problem in his publication “*The Spanwise Distribution of Lift for Minimum Induced Drag of Wings having a given Lift and a given Bending Moment*” [13] that mirrors the constraints set by Prandtl’s revision: gross lift, freestream velocity, and root bending moment. Like Prandtl, Jones alleviated the limitation on wingspan for a set structural weight (the root bending moment constraint). His expression of minimum induced drag (*Equation 2.1.11* - derivation found in original text [13]) yields a 15% increase in span with a 15% reduction in induced drag in the application of a bell spanload as opposed to the existing ELD.

$$D_i = \frac{L^2}{\pi \frac{\rho}{2} V_\infty^2 (2s_e)^2} \left[ 8 \left( \frac{s_e}{s} \right)^4 - 16 \left( \frac{s_e}{s} \right)^3 + 9 \left( \frac{s_e}{s} \right)^2 \right] \quad (2.1.11)$$

In accordance with the other analytical solutions for minimum induced drag, Jones applies Prandtl’s lifting-line theory. He notes that the results are not restricted to that approxima-

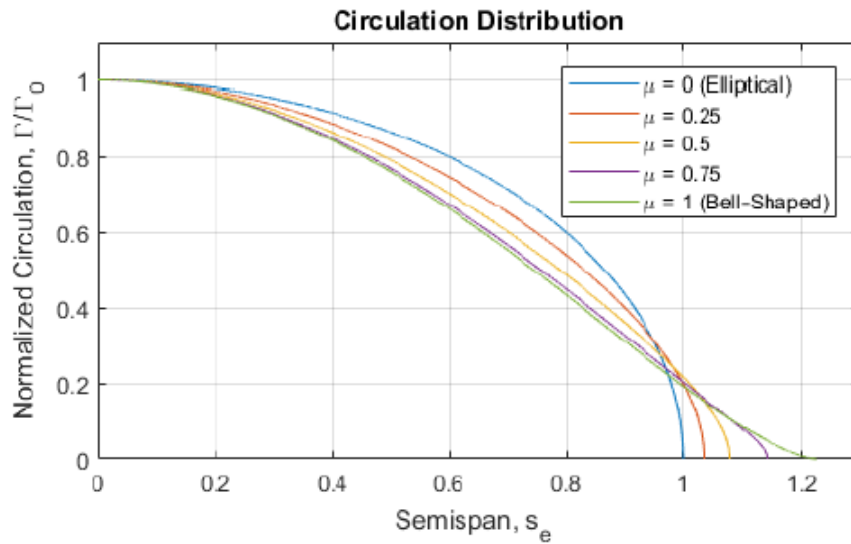


Figure 7: Circulation distribution of various spanloads [14]

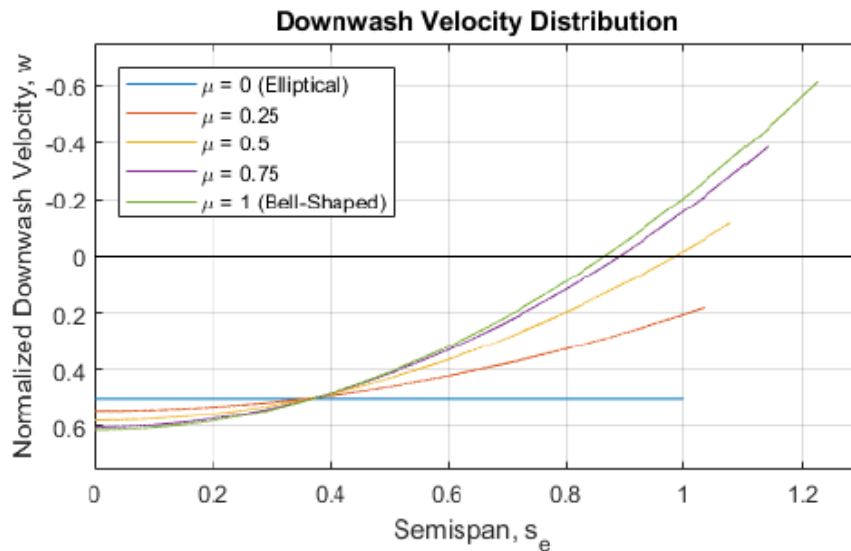


Figure 8: Downwash velocity distribution of various spanloads [14]

tion stating that the induced drag of a lifting surface will be equivalent to that of a lifting line if the spanwise distributions are the same. This is to say that the mathematical solution may be applied to a flying wing, like the planform of the aircraft presented in this thesis, not simply lifting surfaces spanning orthogonal to the freestream flow.

Around the same time Prandtl issued his revision of the minimal drag solution, Reimar Horten independently derived an equivalent solution naming it "bell shaped" in wing loading. Though the methods were different, Horten generated the solution and added to the effort in providing a calculation for the induced drag across the span. This novel finding lead to the discovery of a consequence of the bell-shaped lift distribution not recognized by Prandtl at the time; it produces proverse yaw. This is the act of the aircraft yawing in coordination with a banked turn without the use of an auxillary yaw device. By contrast,

aircraft today experience adverse yaw as a consequence of the high-lifting wing producing greater induced drag as a consequence of lift in a banked turn.

The authors of "*Tailless Aircraft in Theory and Practice*" [15] stand firm against the claims made in favor of the bell-shaped distribution in its chapter entitled "Fables, Misjudgments and Prejudices, Fairy Tales and Myths" positing that proverse yaw is not observed, and additional drag is a consequence of the application of this distribution to a flying wing. Both claims are made in reference to Horten's flying wings which produce adverse yaw in roll and, cited admission from Horten, increased the induced drag by a factor of 4/3 in cruising flight. Emerging research provide considerable counterarguments to the criticisms posed by the authors [15].

In 2016, researcher of NASA Armstrong address this theory in the development and testing of the Preliminary Research Aerodynamic Design to Lower Drag, or Prandtl-D [2]. This aforementioned research simplified the equations of circulation and lift distributions derived by Ludwig Prandtl (*Equation 2.1.12* and *Equation 2.1.13*, respectively) and confirmed a decrease of up to 11.1% in induced drag while increasing the span by 22.5% in the application of the BSLD for a set structural weight enforced by a restriction on root bending moment. These are the equations from which this work will reference to graphically match the computational results through CFD to the BSLD.

$$\Gamma(Y) = \Gamma_0 \left(1 - \left(\frac{2Y}{b}\right)^2\right)^{\frac{3}{2}} \quad (2.1.12)$$

$$L(Y) = L_0 \left(1 - \left(\frac{2Y}{b}\right)^2\right)^{\frac{3}{2}} \quad (2.1.13)$$

Additionally, this research was the first to perform experimental flight tests which observed proverse yaw. The promising demonstration of coordinated flight without a tail provoked recent interest in research regarding the application of this lift distribution to flying wings.

D. F. Hunsaker and W. F. Phillips of Utah State University have done extensive research in the wake of NASA's recent discoveries. One of the recent contributions by these authors, in addition to the translation of Prandtl's 1933 publication [11], produces minimal induced drag solutions from various design constraints [12]. By removing the constraints of gross weight and wingspan, as prescribed by Ludwig Prandtl in 1933, the researchers solve for optimum lift distributions for three example rectangular wings as a result of varying design constraints. The constraints and their resultant optimal distributions follow:

- 1) Prandtl's suggested constraint on gross lift and moment of inertia of gross lift without a constraint on wingspan governed by equation 2.1.14.
- 2) Gross weight, maximum stress, and wing loading of a rectangular wing planform presented in equation 2.1.15.
- 3) An optimal lift distribution resulting from design choices of gross weight, maximum deflection, and wing loading shown in equation 2.1.16.

$$\frac{bL^{\sim}(\theta)}{L} = \frac{4}{\pi}[\sin(\theta) - \frac{1}{3}\sin(3\theta)] \quad ; z = -\frac{b\cos(\theta)}{2} \quad (2.1.14)$$

$$\frac{bL^{\sim}(\theta)}{L} = \frac{4}{\pi}[\sin(\theta) - 0.13564322\sin(3\theta)] \quad ; z = -\frac{b\cos(\theta)}{2} \quad (2.1.15)$$

$$\frac{bL^{\sim}(\theta)}{L} = \frac{4}{\pi}[\sin(\theta) - 0.05971587\sin(3\theta)] \quad ; z = -\frac{b\cos(\theta)}{2} \quad (2.1.16)$$

The equations above, detailed in the research production [12], are obtained assuming a non-structural weight distribution given by 2.1.17 and only differ by the magnitude of a Fourier coefficient. This method holds for rectangular wings of constant chord and thickness. Variations of chord length and thickness along the span require numerical methods that exceed the scope of the research. The research goes further and shows that for any fixed lift and distribution of weight there exists an optimum wingspan for which the condition of minimal induced drag is satisfied. However, the solutions presented for each set of constraints are derived specifically for rectangular wings and cannot be applied when wing sweep is introduced. Though the solutions for the rectangular planform may not contribute to tailless flight, it provides groundwork for aircraft designers concerned with drag reduction, weight reduction, and fuel economy of commercial airliners.

$$W_n^{\sim}(z) = (W - W_r)\frac{L(z)^{\sim}}{L} - W_s^{\sim}(z) \quad (2.1.17)$$

Investigation of non-elliptical lift distribution wings for commercial aircraft application has been carried out by the Aerospace Integration Research Centre at Cranfield University [5] in efforts to reduce the tail size of such aircraft. Authors use early-stage design tools, such as XFLR5 and Athena Vortex Lattice (AVL), to generate Prandtl's 1933 solution to replace the wing of a commercial aircraft. Consistent with the previous research methods, the wingspan is not selected as a design constraint, rather span extension and wing twist are the output of yet another analytical revolution in aerodynamics by the early works of Ludwig Prandtl, Lifting-Line Theory. The baseline model used for analysis is the wing of a small, transport aircraft, SF50. The baseline is compared to two competing models. One applies the analytical solution for the twist to produce minimum drag, while the other uses this technique with the allowance of increased wingspan (Prandtl's prescription). Vertical tailplane size is parametric for the three competing models to gauge aerodynamic performance with reduction in structural weight. The approach inspired by Prandtl shows a 14% increase in lift-to-drag ratio, a 44.34% reduction in vertical tail weight, and up to 17% improvement in range of the aircraft. Flight stability in dutch roll mode was compromised for efficiency metrics in both models deviating from the baseline. Control surface sizing and placement must be further investigated in this study of lift distribution to aptly evaluate its effect on aerodynamic performance and aircraft stability. Despite the promising results, the analysis methods do not apply to swept-wing configurations.

Seeking to contribute computation to the application of a bell-shaped lift distribution to swept wings, professors at the Mangalore Institute of Technology and Engineering in India [22] implement the Prandtl design to a forward-swept flying wing. They believe that an increase in efficiency to this wing configuration will make it superior to a traditional, aft-swept flying wing since it is inherently more maneuverable. Another benefit of this adaptation to the forward-swept wing is that the reduction in load held by the outboard wing sections further reduced the chance of tip stall. The wing modeling begins with a set root and tip airfoil. Tip twist is parameterized at angles of 0, 2, 5, -2, and -5 degrees to modify the lift distribution while maintaining constant aerodynamic twist (constant airfoil cross sections along the wing). Each configuration is analyzed through computational methods with ANSYS. The researchers conclude that the aircraft with the tip twist of -2 relative to the freestream flow exhibits the greatest aerodynamic performance, but the aerodynamic coefficients for the other configurations are not presented. Though this tip twist is assumed to resemble the Prandtl bell curve, the lift distributions of these configurations are not shown to be validated. Additionally, the research does not relate the findings to the performance of a traditional swept wing to justify the use of a forward sweep.

Richter et al [18] studied control surface efficacy in the generation of proverse yaw on straight tapered flying wings applying the bell spanload. The relationship between aircraft geometry and proverse yaw control power is realized by key variants: taper ratio, wing chord, span length, twist distribution, and outboard wing control surface (OWCS) size. Generally, an increase in the size of the OWCS exacerbates the yaw effect on the aircraft, whether proverse or adverse yaw. This effect is tied to the amount of upwash or downwash experienced at the outboard wing section. It is shown that the proverse yaw control power is maximized when the control surface is engulfed in the region of upwash. It follows that the lift distribution greatly affects the yaw ability in roll given that the distribution of lift commands the regions of upwash and downwash. Researchers conclude that the application of the bell spanload, increased wingspan, and greater tip twist suggest straight tapered flying wings may be controllable and naturally coordinate turns. Though this research parameterized geometries particular to straight tapered wings, the findings of the relationship among the lift distribution, regions of upwash/downwash, control surface size and placement, and yaw control power can be used for aircraft design prescriptions of various configurations. The results provide additional support for the emerging research advocating for the use of the BSLD, as opposed to the ELD, for favorable yaw in banked turns of tailless aircraft. The research presented in this thesis will source these results for aileron sizing and placement for the aircraft design, *Bellwether*.

### 2.1.3 Avian Flight Formation

Modern modeling of avian flight is done by assuming birds apply the elliptical lift distribution, since it has been thought to be the most efficient. This explanation fails to explain why birds overlap their wings in flight formations, and it is contrary to reasonable explanations of how birds perform effortlessly coordinated turns without an auxiliary yaw device to counteract the adverse yaw that is common with the ELD. Beyond the observation of proverse yaw in accordance with Horten's theory, Chief Scientist Albion Bowers of the National

Aeronautical and Space Administration (NASA) Armstrong Flight Research Center relate their observations of the bell-shaped lift distribution, and its effect on the flow field, to avian flight formation to suggest that birds dynamically utilize this distribution for increased aerodynamic performance [2]. Dr. Bowers states three main distinctions between avian flight and aircraft designed for elliptical spanloading:

1. The feathers at birds' wingtips are soft, flexible, and narrow with thin structures (ligaments, tendons, muscles, and bones) incapable of supporting load necessary to match an ELD. By contrast, aircraft wings are reinforced structures designed to carry a distributed lifting force out to the wingtips.
2. Birds fly in formation to receive aerodynamic benefits from the wing vortex roll-up of the bird ahead of it in formation. This is observed that birds do this in a v-formation with the wings overlapping. Aircraft also benefit from flying in formation, however, the optimum location for aircraft is to position themselves with wingtips aligned.
3. Despite their sharp-tipped, narrow wings, birds do not experience wingtip stall. The location of stall on a bird's wing is observed to occur about 20% of the semispan from the root. Aircraft of swept wings with sharp tips, primarily fighter aircraft, experience wingtip stall and require mitigation techniques to avoid the detriment to aircraft control in these events.

This is to suggest that the lift generated by bird wings tapers at the wingtips, and thus, the distributed load across the span cannot match that of an ellipse. The lift distribution is the ultimate influencer of the flow field.

The vortex shed line, which is the point at which the region of downwash transitions to upwash [18], is shown in *Figure 2* to be inboard at the 0.704 spanwise location for the BSLD. It is suggested that the wing overlap in the v-formation of bird flight coincides with this vortex shed line. A trailing bird, it is thought, willingly positions itself to fly with enough wing overlap as to span the upwash region for energy savings without much intrusion into the region of downwash. NASA researchers note that this location varies from previous research on different flocks of birds: Portugal [7] citing a vortex core separation of 0.753 for the overlap of northern ibis flying formation, Spedding [10] noting 0.76 as the location for his kestrel research, and NASA's own analytical result of 0.704 semispan location. Though Portugal, Spedding, and others prior have used ELD to model avian flight, all results of inboard vortex cores suggest upwash flow at the wingtips. Referring back to *Figure 1*, this suggests that birds coordinate turns with favorable yaw by the induced thrust at the wingtips that is a natural consequence of the BSLD.

Kyle Lukacovic provides a charitable contribution to the examination of wingtip overlap of bird and aircraft flight formation and its effect on aerodynamic performance in "*A Parametric Study of Formation Flight of a Wing Based on Prandtl's Bell-Shaped Lift Distribution*" [14]. This study adds a computational analysis to two models of the NASA BSLD wing, Prandtl-D, in flight formation parameterizing wing spacing and comparing the overall efficiency of the two-wing system. The substantial outcomes of this study are (1) the BSLD





Figure 9: Common cranes (*grus grus*) flying in formation over lake Fehér near Sándorfalva, Hungary [14]

shows potential for significant increase in performance of formation flight and (2) the ELD is unfit to model bird flight, which seems to relate much more closely to the BSLD on the basis of wing overlap in flight formation. The study models the Prandtl-D using the planform, wing geometry, and airfoil data provided in the NASA publication [2]. The trailing aircraft is placed at different spanwise locations in the wake of the leading Prandtl-D model. The trailing model is set at a distance of one wingspan behind the lead for all simulations. The wing spacing is denoted as  $Y/b$ . For  $Y/b$  equal to zero, the port wingtip of one wing is aligned with the starboard wing of another. Negative values of  $Y/b$  indicate a wing overlap, whereas positive values correspond to the length of spanwise gap between the wings of the two models. *Figure 10* tabulates the results of a collection of CFD simulations ranging the  $Y/b$  parameter from 0.148 to -0.444. This table presents the aerodynamic benefits received by the trailing aircraft as a percentage of the experience of the leading model.

A Trefftz plane analysis was used to determine the reduction of induced drag for the two-wing system. This is a far-field analysis that identifies the forces in the wake of a lifting body by capturing velocity vectors in the wake. The perturbation velocities are related to the aerodynamic parameters on the Trefftz plane. These analysis planes have been set up to determine the aerodynamic benefits received by the trailing aircraft and quantify the efficiency of the flight formation for various wing spacing. CFD shows maximum trailing-wing benefits of a  $L/D$  ratio of 28.5% and a reduced system induced drag of 33.9%. It is noted that this is not a direct comparison to the efficiency savings of formation flight for ELD systems, but "literature research of studies of aircraft in differing configurations reported comparable induced drag reduction of 20-30%" [14]. This is to say that the ideal wing overlap in a BSLD system outweighs the aerodynamic benefits of the wing-overlap

optimization for an ELD system in flight formation.

<b>Trailing Wing Compared to the Leading Wing</b>				
<b>Wing Spacing, Y/b</b>	<b>Lift Percent Change, %</b>	<b>Total Drag Percent Change, %</b>	<b>Lift-to-Drag Ratio Percent Change, %</b>	<b>System Induced Drag Percent Change, %</b>
0.148	2.195	-6.489	7.855	-14.939
0.074	2.734	-6.805	10.234	-16.840
0.037	3.101	-7.785	11.805	-18.746
0.000	3.576	-9.040	13.870	-20.656
-0.037	4.147	-10.645	16.555	-23.155
-0.074	4.789	-12.517	19.783	-25.907
-0.111	5.430	-14.511	23.326	-28.851
-0.148	5.797	-16.083	26.072	-31.123
-0.185	5.891	-17.090	27.718	-32.704
-0.222	5.795	-17.652	28.473	-33.696
-0.259	5.547	-17.690	28.231	-33.930
-0.296	5.176	-17.264	27.123	-33.504
-0.333	4.697	-16.387	25.216	-32.316
-0.370	4.082	-14.978	22.418	-30.396
-0.444	2.396	-11.037	15.100	-24.522

Figure 10: Aerodynamic Parameter Percent Change Between  $Y/b = 0.148$  and  $-0.444$  for the Trailing Wing Compared to the Leading Wing [14]

The ELD design aircraft wings has an average optimal spanwise position at  $Y/b = -0.128$  where the vortex is at the wingtip location ( $Y/b = 0$ ). It is hypothesized by the author that the reason for the trailing wingtip to span beyond the vortex shed line is that it exposes more of the higher loaded wing area to the beneficial upwash of the vortex while avoiding much of the ramifications of the downwash. *Figure 11* gives the optimal wing overlap of different systems and for different aerodynamic objectives for the bell-shaped lift distribution. Additionally, the beneficial range of operation of the trailing wing is tabulated for each metric. "BSLD wide range of beneficial positioning of  $0.407b$  (41% of the full wingspan) or greater" is a substantial claim resultant from this research that relates to Dr. Bowers and the juxtaposition of the ELD and BSLD flow fields in *Figure 2*. The average range for trailing wings of ELD is around  $0.250$  which is significantly less than the range allowed for trailing wings of BSLD to observe aerodynamic benefit. The difference is ascribed to the wider vortex created by the bell spanload as opposed to the narrow vortex core generated by elliptical wings at the wingtips resulting in the sharp, distinct discontinuity between regions of upwash and downwash.

#### 2.1.4 Literature Review - In Summary

For over a century, from the guidance of Ludwig Prandtl, aircraft wings have been designed with the idea that lift distributed across the span should take the shape of an ellipse for the most efficient aerodynamic result. By Ludwig Prandtl's own revision, the discovery of the

	Optimal Position, Y/b (Y/b <sub>e</sub> )	Beneficial Range, b (b <sub>e</sub> )
Avian Flight (BSLD – Lift)	-0.185	0.735
ELD - System Induced Drag	-0.128	0.250
BSLD - Lift	-0.185	0.407
BSLD - Total Drag	-0.259	0.444
BSLD - Lift-to-Drag Ratio	-0.222	0.407
BSLD - System Induced Drag	-0.259	0.481

Figure 11: Optimal Position and Beneficial Range from Avian, ELD, and BSLD research [14]

bell-shaped lift distribution has sparked research that suggest that this spanload is the most efficient method of generating lift for a fixed amount of lift and a maximum root bending moment (or structural weight). Most notably, Dr. Bowers at NASA Armstrong is the first to measure proverse yaw by experimental flight data. The data materialized Horten’s theory which recognized the phenomenon of induced thrust as a consequence of inboard upwash to encourage favorable yaw without need for a vertical tail. NASA researchers correlate the characteristics of the BSLD with bird flight which has been previously understood with ELD analysis models. These correlations, advanced by Kyle Lukacovic of California Polytechnic State University [14], provide three explanations to bird flight being modeled by BSLD that have failed to be explained by ELD:

1. Birds can effortlessly and precisely coordinate turns without the use of an auxiliary yaw device, which would be necessary to counteract the adverse yaw ELD wings produce in roll consequential to the region of downwash engulfing the wingspan. This suggests that the flow field generated by bird flight contains a vortex shed line that lies within the wingspan (i.e. upwash is created within the span).
2. Birds have thin and light feathers, ligaments, tendons, and bones at their wingtips not suitable to carry the load prescribed by an ELD. Furthermore, unlike thin-tipped aircraft, birds do not experience tip stall; they stall close to the 20% span location. This can be explained with the application of the BSLD to suggest that the load across the span tapers to zero at the wingtip.
3. Birds overlap their wings when flying in formation to optimize energy savings. The beneficial location can vary, but the ELD demands a much narrower range of wingtip locations for the trailing bird to receive benefits. The spanwise area for formations of ELD wings is hardly overlapping wingtips. Birds much more closely match the wing overlap that proves to be most beneficial for wings designed with a BSLD.

This is pertinent because, in part, the aircraft design presented in this thesis is a biomimetic solution in understanding how birds can maneuver with no observable yaw authority.

This work turns to the aircraft design process that seeks to leverage this understanding of inboard upwash and its assistance to proverse yaw.

## CHAPTER III

### Design Methodology

Overall, the design lift distribution is intended to match the Prandtl Bell shape. Prandtl [11], and subsequent researchers of the bell spanload, noted that optimization of this problem would require the abolition of the constraints placed on aircraft wingspan; however, this design relies on placing that constraint. Although the theoretical underpinnings of this lift distribution dictate no inherent wingspan, practical limits apply on the high-side to keep vehicle weight below the maximum allowed for operations under the Federal Aviation Administration's Part 107 rules. Thus, design iterations were performed using historically validated mass property models from Oklahoma State's long history with development and construction of fixed-wing unmanned aircraft systems. It was found that the design could be reasonably guaranteed to fall below the 55 lb maximum (categorically remaining a small, unmanned aircraft (SUAS)) while allowing for carriage of the desired atmospheric sensing payloads if the span was limited to 25 feet. Additionally, this design point is congruent with lab space availability and our current construction capability. These constraints allow for the Bellwether aircraft to benchmark the Prandtl-D 3C from NASA Armstrong to define a planform. Thus, this aircraft, *Bellwether*, is prescribed a 25 ft. wingspan (not including the width of the fuselage), a 25 sweep, and 2.5 dihedral angles that have been demonstrated previously. All analysis methods will use a flow of thermodynamic values reflective of standard atmospheric conditions at 1000 ft. asl. to simulate the flight conditions expected for Stillwater, OK.

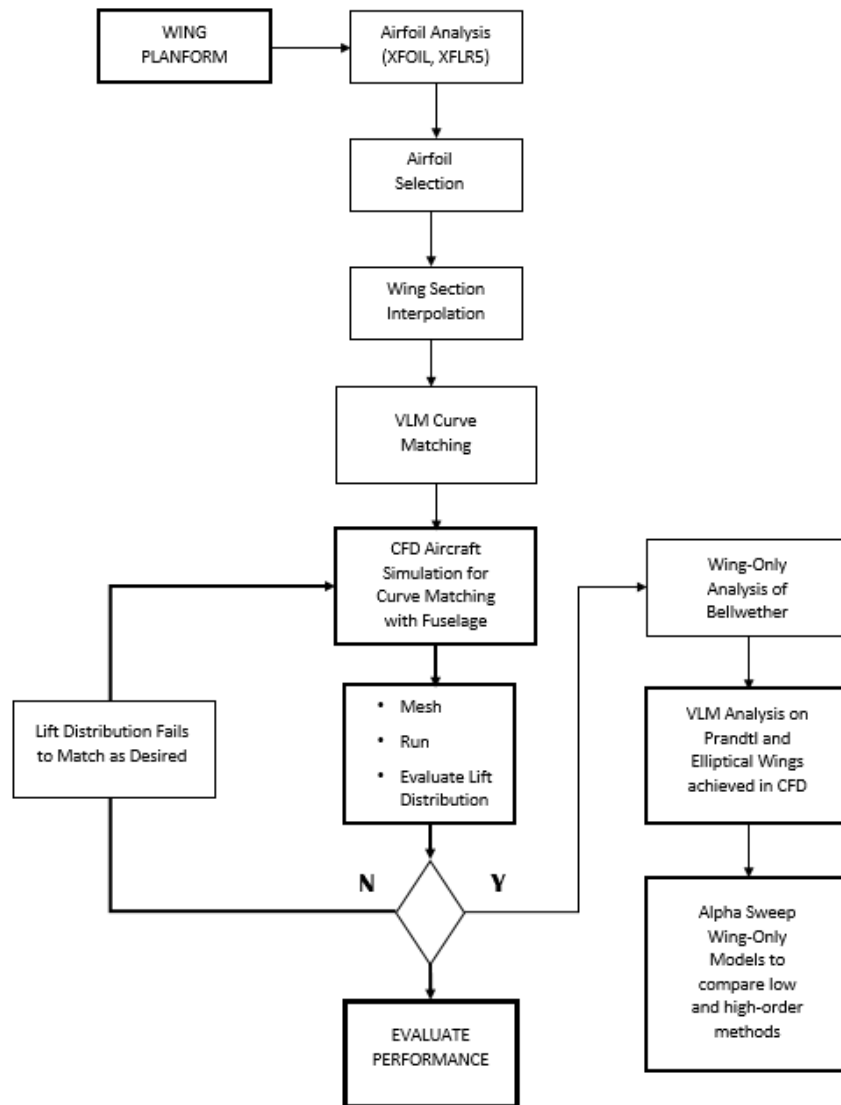


Figure 12: Methodology Flow Chart

With the known planform and atmospheric flight conditions previously stated, the design methodology steps through the chronology beginning with airfoil analysis by low-fidelity, panel methods. XFOIL, developed by Mark Drela [6], and XFLR5 are used to design airfoil sections on the basis of Reynold’s number and desired sectional lift to match the desired spanload. Once the root and tip airfoils are selected, they are geometrically linearly lofted together adding consistent aerodynamic twist along the wing. The root and tip airfoils are interpolated to create the airfoils at each foot of the span. All airfoils are saved as ‘.dat’ files to import coordinates to and from different software packages. These spanwise airfoils are used as control points individually rotated to introduce geometric twist along the wing. This geometric twist is varied such that its addition to the set aerodynamic twist will yield lift distributions desired by this comparative study. Vortex Lattice Methods are used to estimate the sectional rotations required for each lift distribution evaluated. Four separate

lift distributions, and in turn four separate wing designs, are generated for comparison. CAD modeling applies these geometric wing twists, suggested by VLM, to a model with a fuselage designed to make each lift distribution graphically continuous at the root. CFD evaluates each flying wing’s adherence to the distribution it seeks to match. Conventional wing design does not account for the effect of the fuselage to the lift distribution, but the design of this flying wing incorporates a lifting-body fuselage to ensure a continuous curve validated by high-fidelity, CFD analysis. Changes to wing twist, and observation of its adherence to its desired lift distribution, iterate until convergence on the desired curve passes visual inspection. The design of Bellwether concludes when the aircraft has matched the curve described by Prandtl’s revision of the minimal induced drag solution. After finalization of the aircraft design, the wings are analyzed in absence of the fuselage to add comparison of aerodynamic performance among the various analysis methods. Since VLM are not reliable in resolving shear and pressure forces influenced by bluff fuselages, the comparison of XFLR5, AVL, and CFD are done as wing-only analyses.

The methodology flow chart *Figure 12* guides the reader through the process leading to the aircraft design and analysis of Bellwether, though the presentation of this work is not precisely adherent to the temporal flow chart. The analysis of the wing-only geometry takes precedence in providing correlations among the methods of aerodynamic analysis. This is done to encourage future aircraft designers to choose rapid results of VLM for highly-iterative design changes. So, the analysis of the final wing design absent the fuselage is presented first in this thesis comparing the low- and high-fidelity analysis tools before the presentation of computational fluid dynamic processes and results of the four different aircraft designs matching different lift distributions. This design process chose to accept computational expense of iterating in CFD to ensure the final aircraft design, including the fuselage, matches the promising spanload. The computational burden is placed on Oklahoma State University’s High-Powered Computing Center (HPCC), Pete.

### 3.1 Span Loading Design Target

Given a planform, the objective of the flying wing design becomes to match the target Prandtl Bell lift distribution by appropriate specification of geometric and aerodynamic twist across the span. The Prandtl Bell distribution is shown by the blue line in Figure 13. This line is generated from Equation 3.1.3, and is simply a scaled version of the circulation distribution achieved from Prandtl’s 1933 analysis. Also included in the figure are curves representative of the modern bell distribution (orange, Equation 3.1.1), the cosine distribution (gray, Equation 3.1.2), and the aforementioned elliptical distribution (green, Equation 3.1.4). These curves are included for reference, as they have been extensively used in past aerodynamic analyses.

$$L' = \frac{1}{\sigma\sqrt{2\pi}} \exp -\frac{1}{2}\left(\frac{z - \mu^2}{\sigma}\right) \quad \textit{ModernBell} \quad (3.1.1)$$

$$L' = A \cos C \frac{y_0}{B} + A \quad \textit{Cosine} \quad (3.1.2)$$

$$L' = L_{y=0}(1 - y^2)^{\frac{3}{2}} \quad \textit{PrandtlBell} \quad (3.1.3)$$

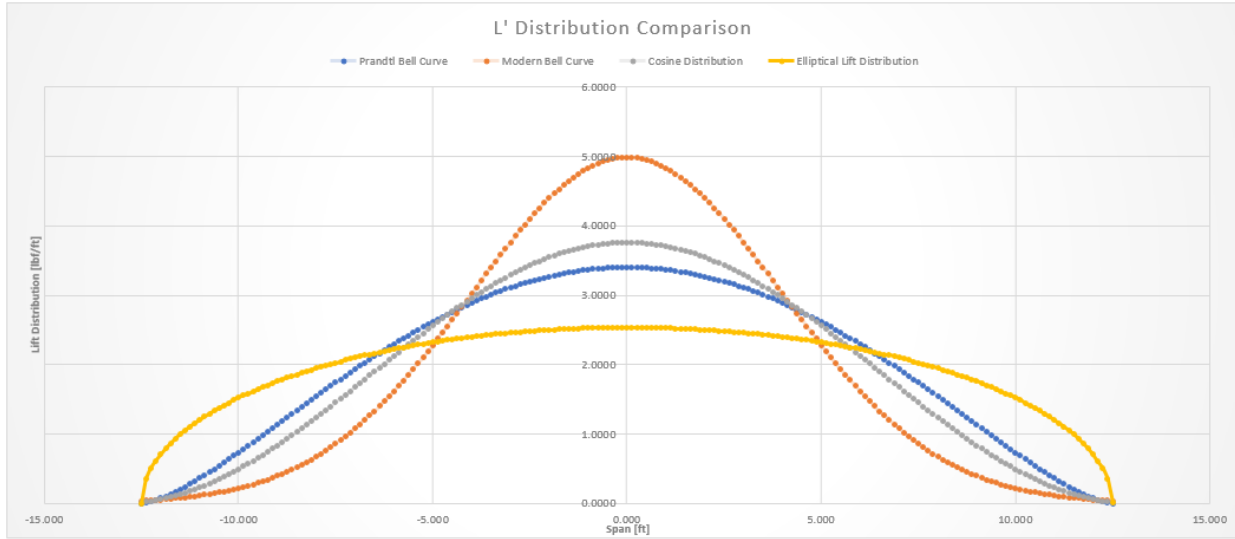


Figure 13: Lift Distributions to be Matched

$$L' = \frac{x^2}{a^2} + \frac{y^2}{b^2} \quad \textit{Ellipse} \quad (3.1.4)$$

### 3.1.1 Span Load Analysis Using Vortex Lattice Methods (VLM)

Vortex Lattice Method, VLM, is a medium-fidelity software method for evaluating the aerodynamics of 3D wings subject to the limitations of potential flow. This numerical method models aircraft lifting surfaces as a thin sheet of discrete vortices. Each vortex ring induces velocity at all points in the flowfield domain. A system of equations linear is generated which enforce zero normal flow boundary condition at each vortex ring's control point based on the amount of bound circulation retained in each vortex ring. The system of equations is readily solvable using standard matrix inversion techniques from linear algebra. Knowing the circulation bound by each vortex ring allows for rapid calculation of the lift and induced drag. The insight brought by the tools leveraging this method allows for changes and quick results as a result of these modifications, without the designer being burdened by the more expensive calculations required by more robust methods, such as CFD.

The rapid, lower-fidelity analyses are performed using XFLR5 and Athena Vortex Lattice (AVL). The flying wing is created and analyzed directly within the software package XFLR5. At the spanwise locations indicated in *Figure 16*, the interpolated airfoils are angled to modify the lift distribution until the target distribution presented in Section 3.1 is achieved.



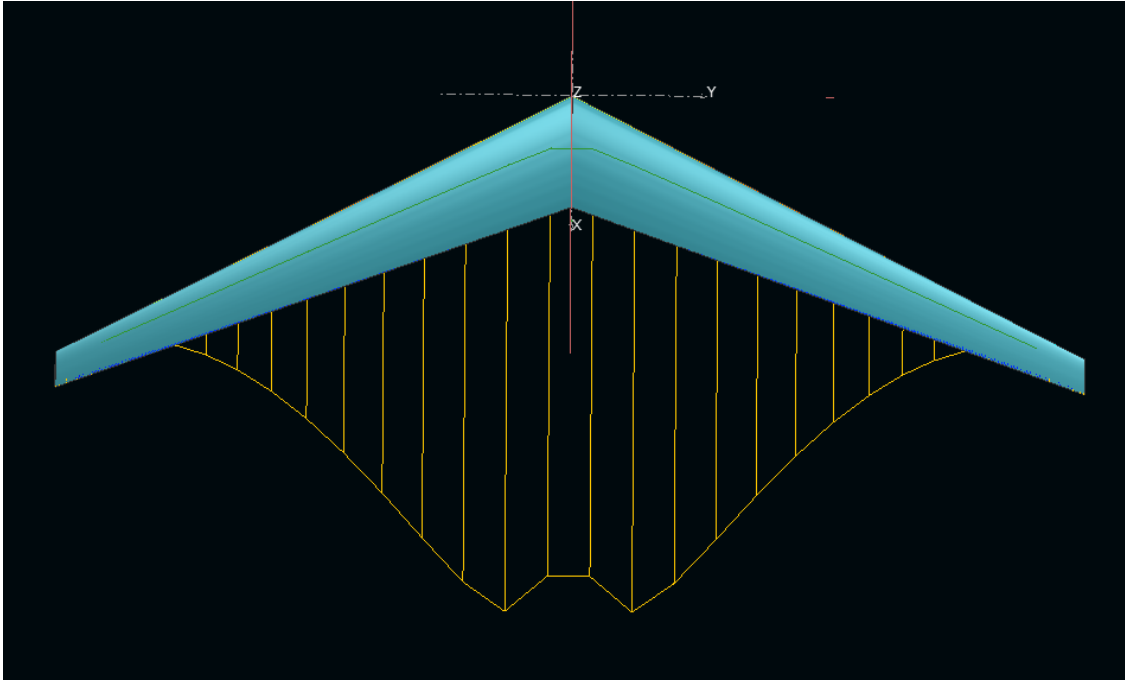


Figure 14: Region of Downwash Observed by XFLR5 Analysis

### 3.2 Airfoil Analysis and Selection

Using low-Reynold's number airfoils from the UIUC [21] database, a batch analysis gave guidance for the selection of a root airfoil. The analysis ranged Reynold's numbers of 200,000 to 700,000 by increments of 100,000 in order to capture all likely conditions experienced by all wing sections in flight. *Figure 15* displays the performance of the selected foils across this operating range with the root airfoil shown in bright green, the tip airfoil in blue, and the interpolated airfoils between the two.

Since the root airfoil is to be geometrically linearly lofted to the tip, good performance at a large range of angle of attack,  $\alpha$ , values is desired. It proved beneficial to add camber to a NACA 4412 airfoil. This change shows greater consistency in performance of the lift-to-drag ratio for alphas ranging from 8 to 0. The airfoil selected as the root, *BellwetherOne*, was crafted by a 50% interpolation of the NACA 4412 airfoil with the Wortmann FX74.CL5.140.

Unique to this design, the selection of the tip airfoil is not significantly dependent upon Reynold's number or lift-to-drag performance; it is designed to taper the lift distribution to zero at the tips to match the bell spanload. This wing selects a symmetric airfoil at the tip to employ enough aerodynamic twist to make it easier to match the center-loaded distributions without needing excessive geometric twist. The tip airfoil is positioned at a slightly negative angle of attack to (1) taper the lift at the outboard wing section and (2) effectuate the region of upwash for induced thrust. The anti-symmetric aileron deflection will engage the utility of this design in producing proverse yaw.

The semispan airfoils are created at specified sections to build control points from which angles of attack may be independently changed to match the desired lift distribution. The creation of these airfoils are simply interpolated in XFLR5 between the root and tip airfoils

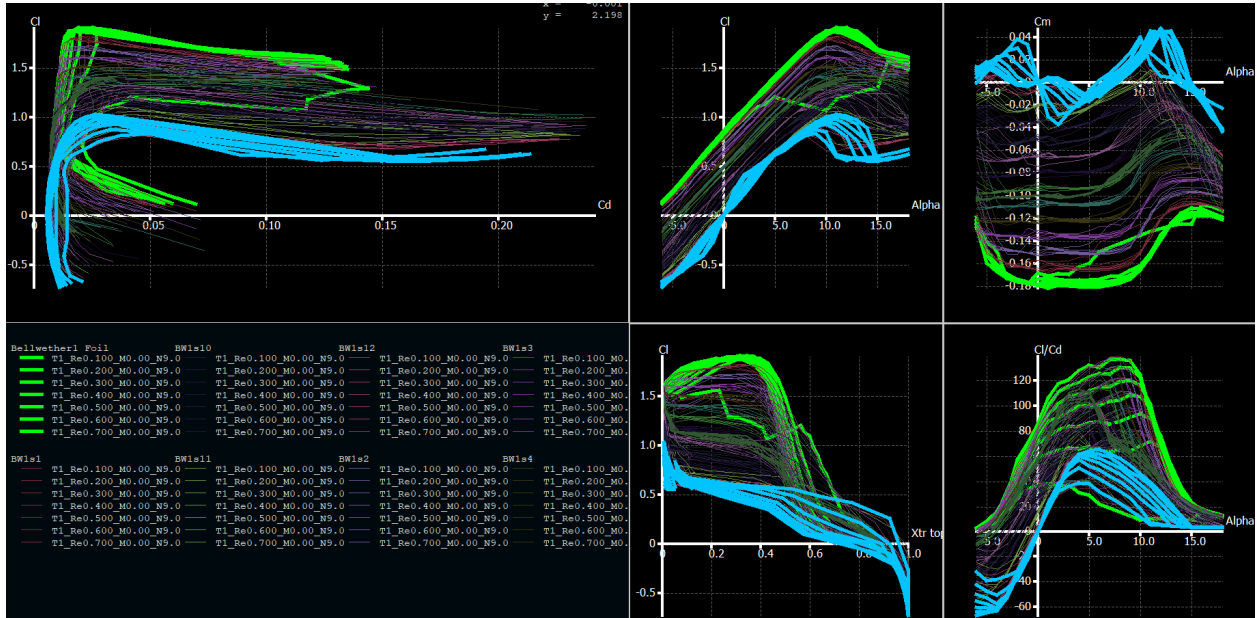


Figure 15: Polars of Interpolated Airfoils

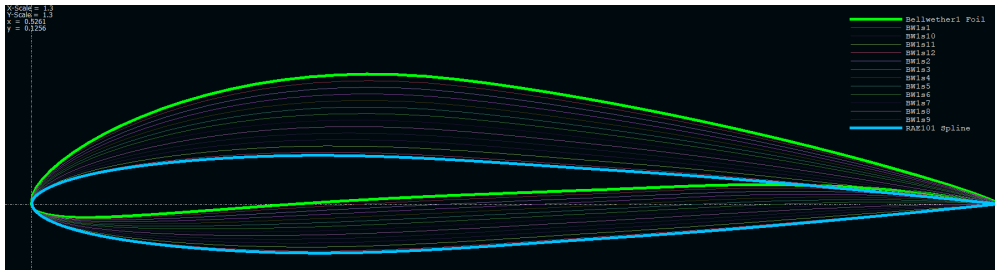


Figure 16: Interpolated Airfoil Cross Sections

to match a continuous lofting between the two. *Figure 16* shows the change in shape of the airfoil sections and the shape of each airfoil, from root (green) to tip (blue), whose location along the span is shown in *Figure 17*.

### 3.3 Aircraft Modeling

With the use of SolidWorks, a baseline flying wing is made for the purpose of preliminary analysis. The wing-section control points are placed at angles of attack determined by the particular airfoil cross section's performance as evaluated by XFLR5. With help of *Figure 15*, the section's angle is graphically determined by the product of the chord and coefficient of lift that section is required to produce to adhere to the desired lift distribution. This is used to initialize the wing geometry prior to entering the loop of high-computational analysis for lift distribution curve matching. This is done for four different models each achieving one of the desired spanloads: Modern Bell, Cosine, Prandtl Bell, and Elliptical. *Equations 3.1.1, 3.1.2, 3.1.3, 3.1.4* define these curves, respectively, and *Figure 13* provides the graphical representation of the curves to be achieved from the aircraft models.

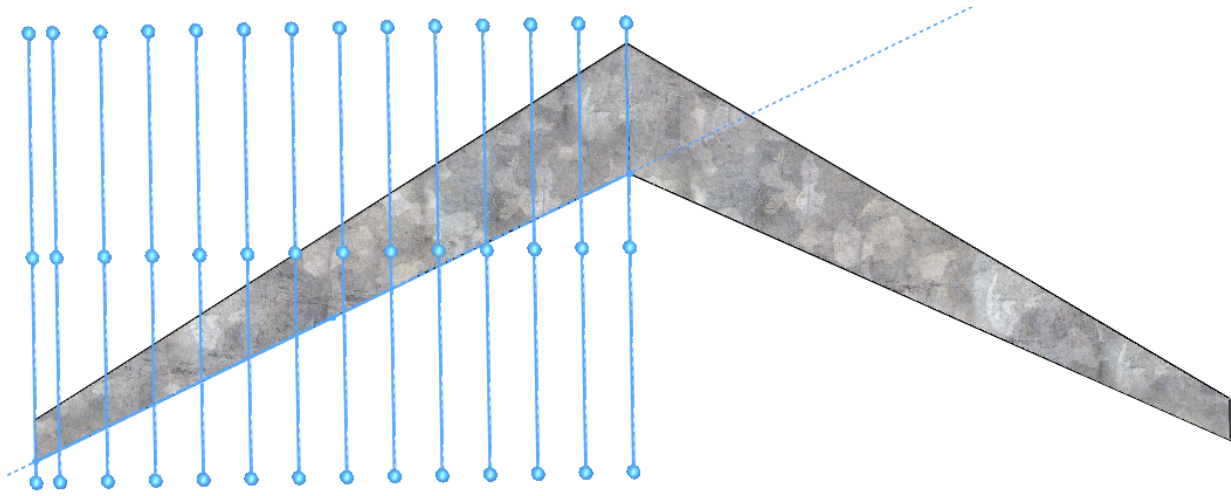


Figure 17: Planform Sections

Modeling a wing of the elliptical distribution presents design considerations in deviating from the centrally concentrated spanloads. The first attempt aimed to use geometric twist to achieve the lift distribution while keeping the same wing sections used for the other curves. However, this requires outboard wing sections to experience local angles of attack in excess of 9 which is not a feature that would be seen if the design was meant for that distribution. The solution is to decrease aerodynamic twist by changing the airfoil cross sections such that less camber is needed at the root and more at the tip. Since the root airfoil was generated by adding camber to a NACA 4412 airfoil, that airfoil has been chosen as the constant wing section of the ELD model. Congruent with the processes of the other distributions, the curve determined the lift required for each semispan section of the wing. The coefficient of lift is determined from the chord length and lift required and the flight condition. The initial guess at each section's angle of attack is generated from the lift-curve slope for each airfoil in XFLR5. CFD iterations are used to modify geometric twist of the airfoil sections for closer adherence to the elliptical curve.

This study adds Cosine and Modern Bell curves to offer a wider selection of designs from which to compare lift-to-drag ratios and other aerodynamic performance metrics. The Cosine curve closely resembles the Prandtl curve with slightly more central loading. The addition of this spanloading allows for a closer look at how an incremental deviation to the theoretical minimal induced drag will perform using high-fidelity aerodynamic tools. Introducing the Modern Bell spanloading presents a highly-centralized lift distribution which can be expected to shift the vortex cores farther inboard resulting in a larger region of upwash within the span. Presenting various flow fields may contribute to the understanding of vortex core location and proverse yaw control power.

Though the aileron sizing should be specific to each flow field to maximize yaw authority, each wing design has been modeled with ailerons sized in accordance with the suggestion by Richter et al [18] for favorable yaw moment in the flow field resulting from Prandtl's lift distribution. This guidance results in aileron placement ranging from the 80% span to the wingtip. The control surface width extends from the trailing edge to the quarter chord tapering with the wing geometry.

## CHAPTER IV

### Comparing VLM and CFD Solutions for the Bellwether Wings

The analysis from the lower-order methods, XFLR5 and AVL, are compared to computational fluid dynamics results from STAR-CCM+ using the same wing geometry without the influence of a fuselage. In particular, the coefficient of lift at pre-stall angles of attack are compared among these analysis methods as well as the shape of the lift distributions presented as  $c^*CL$  across the wingspan in vortex lattice methods.

#### 4.1 Wing-Only Analysis of Bellwether in Vortex Lattice Methods

XFLR5 was used to model the wing for the VLMs. Beginning with the root airfoil, *BellwetherOne*, the interpolated airfoils shown in *Figure 16* are placed one foot apart along the span as shown in *Figure 17* (with the exception of the last interpolated airfoil to the tip airfoil, *RAE 101*, whose distance is half a foot). In the wing editor window, the x and y panels used for vortex lattice analysis are selected to use cosine spacing for each airfoil section to provide a higher concentration of panels on the leading and trailing edge of the wing as well as the root and the tip. All analysis methods found in XFLR5 were compared at a 0 degree angle of attack in a fixed-speed analysis at a freestream velocity of 35 feet per second. Analysis methods VLM1 and VLM2 each produced coefficient of lift values of 0.762 regardless of the addition of viscosity to the analysis settings. The Ring Vortex analysis setting produced a coefficient of lift of 0.788, and lifting-line theory (LLT) output a coefficient of 0.830. LLT is not generally used for swept wings, so that analysis method will be disregarded for this aircraft configuration. The analysis setting used for the comparison of coefficients of lift across angles of attack ranging from -4 to 10 degrees is VLM1.

The wing geometry from XFLR5 was directly imported into AVL for analysis. Solving at a 0 angle of attack yields a coefficient of lift of 0.85 in the inviscid analysis. The Trefftz plane results (*Figure 18* - downwash marked in blue), as with XFLR5 (*Figure 14*), indicate the transition from upwash to downwash begins within the span at the same approximate location.

The discrepancies in the coefficients will be discussed with the results from CFD and the evaluation of performance from all methods across the prescribed range of angles of attack. The curve matching of lift distributions across the span will be performed at a constant coefficient of lift for all analysis methods. Before introducing these results, the methodology and design settings in STAR-CCM+ must be outlined.

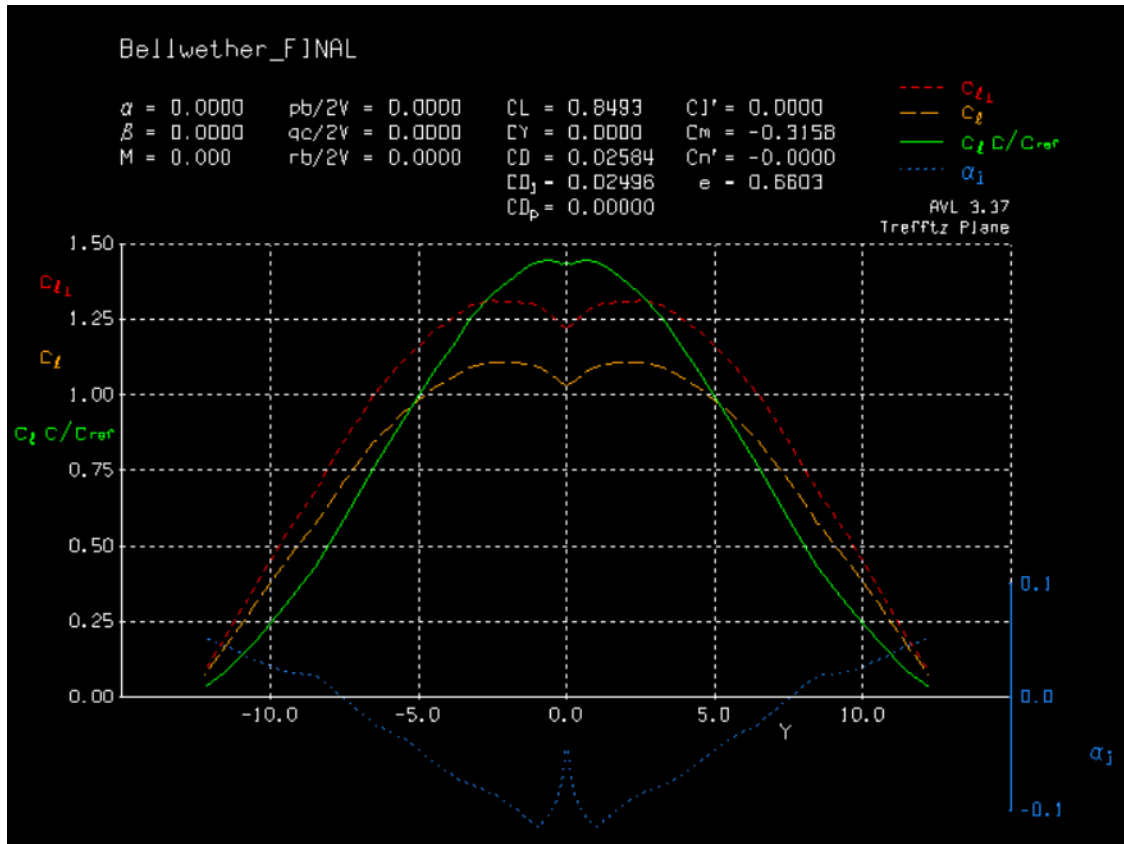


Figure 18: Trefftz Plane Results from AVL

## 4.2 Wing-Only Analysis of Bellwether in STAR-CCM+

The CAD model of Bellwether is first imported as a parasolid into STAR-CCM+ in the creation of a surface mesh. A geometric block part of (25 x 30 x 50) cubic meters is used as the flow domain from which the aircraft is extracted through a Boolean Subtract feature. The wing surfaces are split by patch for additional surface control and command of prism layer design.

The wing-extracted domain is added as a region with a region for each part and a boundary for each part surface. The inlet of the domain is set as a velocity inlet with a magnitude of 35 feet per second. The aft surface of the block domain is set as a pressure outlet at atmospheric pressure. The walls of the domain are set as a freestream with a mach number of 0.031 in correspondence with the inlet velocity magnitude and air at 1000 feet above sea level to simulate atmospheric conditions in Stillwater, OK. A physics continuum is added with the settings listed in *Figure 20*. Completion of the physics continuum settings prompts reports and plots to be made to track desired performance parameters. Reports for coefficients of lift, drag, and moment, as well as lift and drag forces, are monitored and plotted on the same graph tracking their values per iteration of RANS flow solving. To output distributions of lift and drag across the wingspan an accumulated force table is added as a tool. These tables are referenced as the data series used in plotting the distributions, particularly used for the

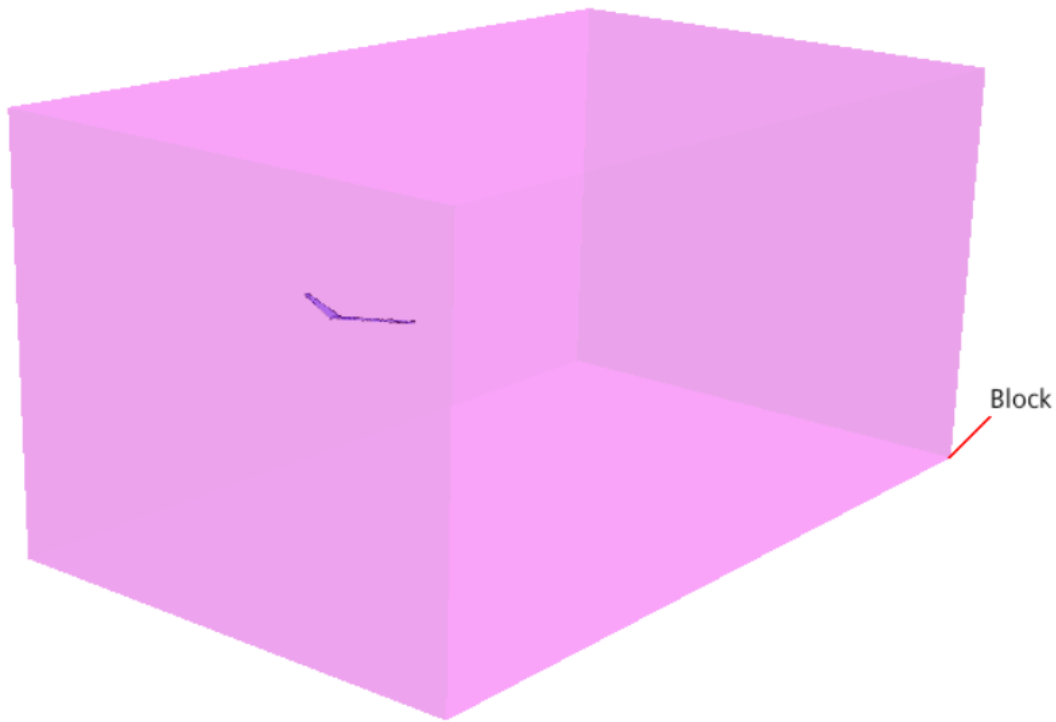


Figure 19: Flow Domain used for the Wing-Only Analysis of Bellwether

comparison of lift distribution to lower order methods. The data is collected as the average force experienced within spanwise bins. The bin number is set to 20, which is to say, there are 20 bins of equal width from wingtip to wingtip within each an accumulated force is the output. This allows for an aggregate force distribution across the span. All main physics reports, monitors, plots, and scenes (with the exception of the vorticity scenes which will be explained later) can be duplicated by using the attached macro *FlyingWingMacro.java* found in Appendix 0.7. Most of the settings are universal and can be transferred to any flying wing re-imported into the domain, though some modifications may be required. Guidance for most of these required changes can be found in the comments of the macro code. Once the simulation has been constructed to report all of the physical properties desired, setup proceeds with mesh settings and design.

Mesh design begins with the creation of a 3D-Automated mesh added to the flow domain. A polyhedral mesher is selected with surface remesher enabled. Prism layer mesher is added to accurately capture boundary layer by constructing thin, prismatic cells along boundary surfaces. Surface controls are added for control of these prism layer numbers and thickness for each surface independently. Default controls such as base size, target surface size, minimum surface size, surface growth rate, and prism layer values are modified through an iterative process to achieve a suitable mesh by inspection as a baseline. Leading and trailing edges of the wing are examined to ensure the cell sizes are fine enough to produce smooth surfaces in the mesh scenes as an indication that the model will be represented well when the flow physics are introduced. Imperfections on thin or or highly-curved surfaces may be the

<b>STAR-CCM+ Physics Continuum Settings</b>
All y+ Wall Treatment
Cell Quality Remediation
Coupled Energy
Coupled Flow
Gas - Air
Gradients
Ideal Gas
K-Omega Turbulence
Reynolds-Averaged Navier Stokes
Solution Interpolation
SST (Menter) K-Omega
Steady
Three Dimensional
Turbulent
Wall Distance

Figure 20: Physics Settings for the Wing-Only Analysis of Bellwether

result of cells adjacent to the surface too large to capture the gradient of the surface. This may result in a curved surface appearing jagged or lacking in resolution. Alternatively, this problem may occur if the parasolid lacks resolution. The tessalation density determines how closely the parasolid matches the CAD part. Density increases allow for a smaller minimum surface size and higher quality mesh on the part surfaces but come at the cost of increased cell count, and thus, greater computational effort and file size. Importation of the parasolid is chosen with a 'very fine' tessalation density after a 'medium' density proved insufficient for the trailing edge of the flying wing. Examination of prism layers, mesh size, and surface boundaries (*Figures 21, 22, and 23*) initiate a baseline mesh to be refined with guidance from scalar scenes, reports, and residual plots. The mesh is validated by a mesh independence study. This study involves changing all mesh values to be relative to the base size then changing the base size by 20 percent. If the physics values reported are closely similar to the values solved for by the original mesh, then the mesh is said to be largely independent of the cell size, meaning an increase in the cell count and computational effort does not provide much aid in producing a more accurate result.

After iterations of meshing to create a visibly suitable mesh design, the physics are set to run to 1000 iteration. In unpacking the physics results, two main metrics are evaluated in the determination of a quality mesh design: convergence and wall y+. Checking convergence of monitors and plots indicate whether the solver has achieved what it determines to be a consistent answer to the physical parameters it is tracking. Wall y+ is a scalar measure of velocity gradients near surface boundaries, essentially, indicating at which areas on the surface of the part are losing information in adjacent cells close to the surface. This value

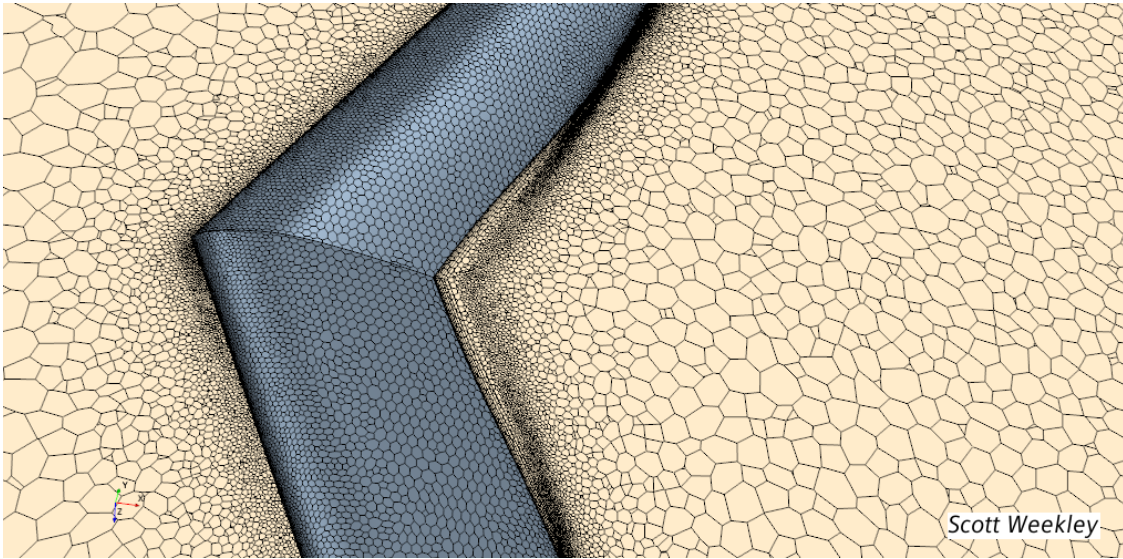


Figure 21: Mesh Scene of Wing-Only Model

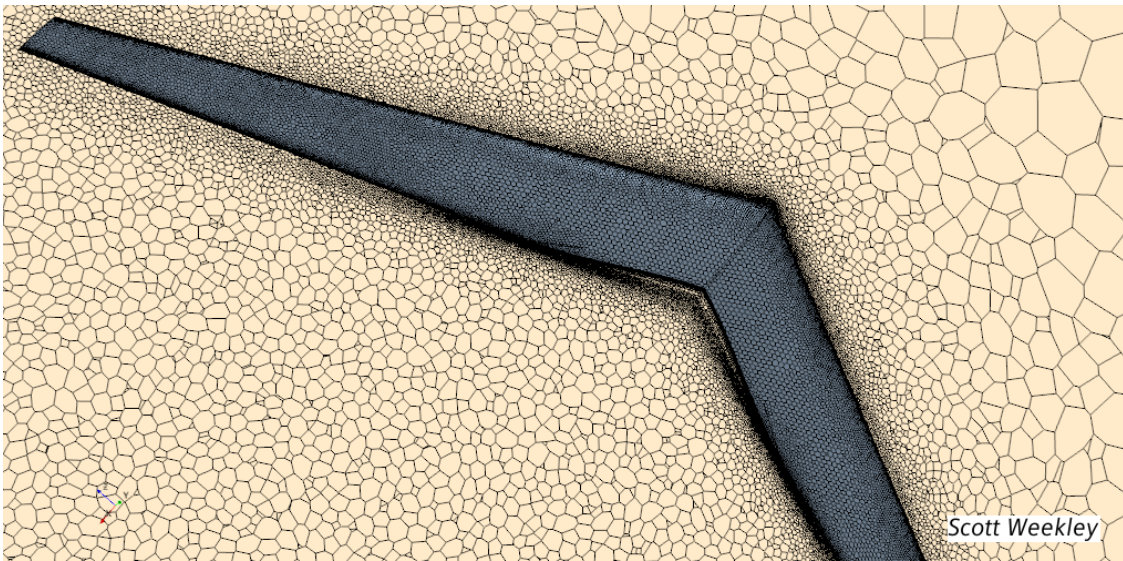


Figure 22: Planform Mesh Scene of Wing-Only Model

is desired to be close to unity. Coefficients and forces are monitored at each iteration shown in *Figure 24* and *Figure 26*, respectively. The values appear to converge as quickly as 100 iterations. Wall  $y^+$  scalar values shown in *Figure 28* show values higher than optimal on the wing, particularly on the leading edge. Due to the quick convergence of the physical parameters, despite the sub-optimal wall  $y^+$  scenes, the CFD analysis proceeds with a mesh independence study to validate the mesh design.

The base size of the mesh is increased by 20%, and the physics is run again to 1000 iterations. The comparison of the force coefficients is presented in *Figure 29*, and they indicate this mesh design to be independent of cell size and accurately reporting the flow physics. Additionally, the unsteady RANS physics solver ran on the same mesh to provide



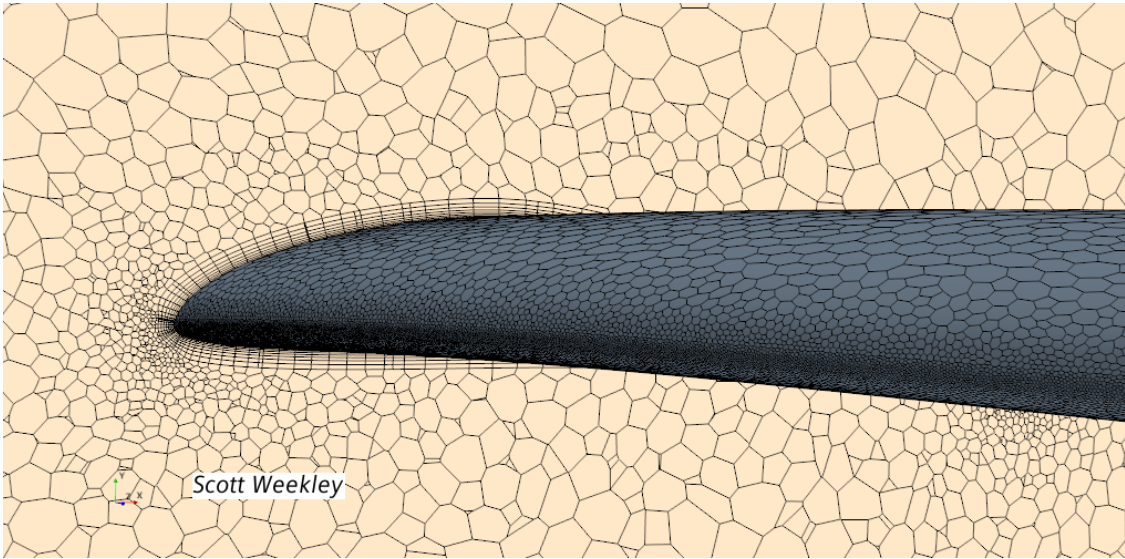


Figure 23: Prism Layer Mesh Scene of Wing-Only Model

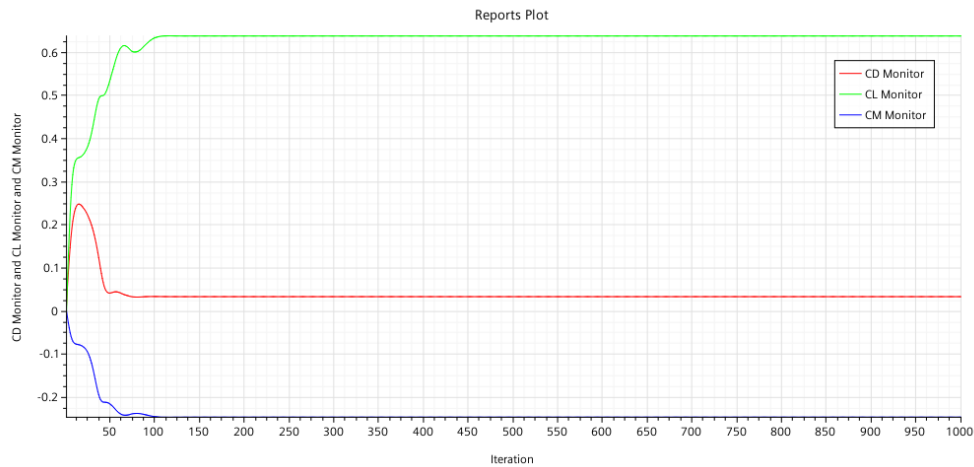


Figure 24: Bellwether Wing-Only CFD - Coefficients Monitor

greater validation of the mesh and physics reports. The physics ran 5000 steps reporting coefficients of lift and drag of 0.6363 and 0.03254 with a percent error of 0.1418% and 0.1346%, respectively, compared to the steady RANS physics solution.

Visualizing vorticity in the wake of the aircraft is a strong qualitative tool for understanding the flow field and the changes caused by differences in lift distribution. This becomes particularly pertinent in *Chapter V*. Here, it will be described how each of these scenes are generated.

*Figure 30* is a top view of a wake plane made from a derived part in STAR-CCM+. The vorticity vector length corresponds to the magnitude. The largest vector is shown in red. The vector lengths are modified in the properties window until the maximum vorticity vector interacts with the wing for ease in identifying the location with the STAR-CCM+ measurement tool. This is the central location about which the flow is turning (vortex core), and it

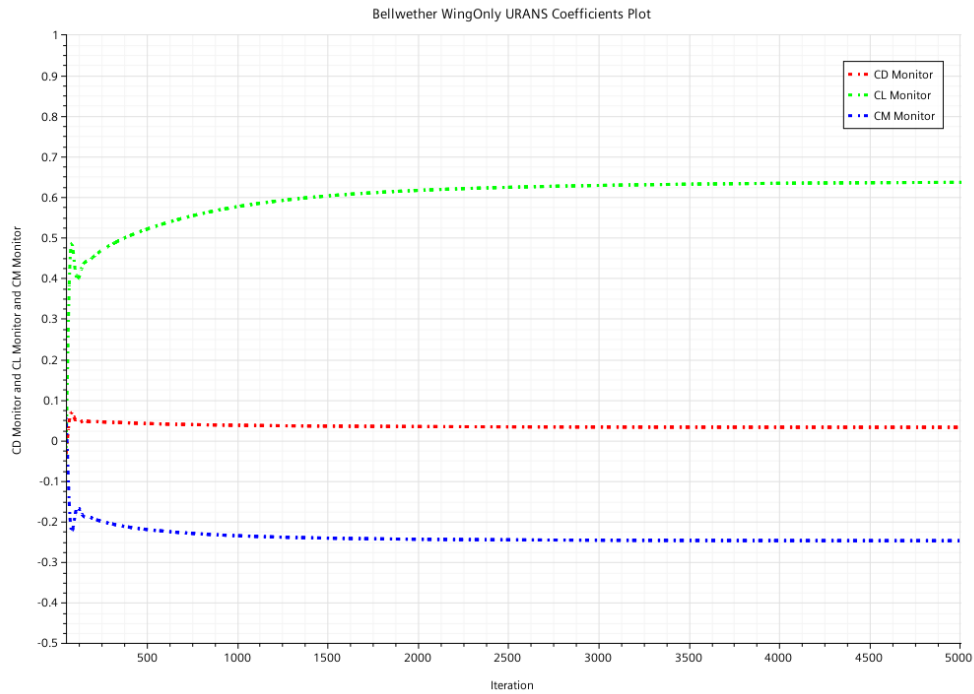


Figure 25: Bellwether Wing-Only Unsteady RANS - Coefficients Monitor

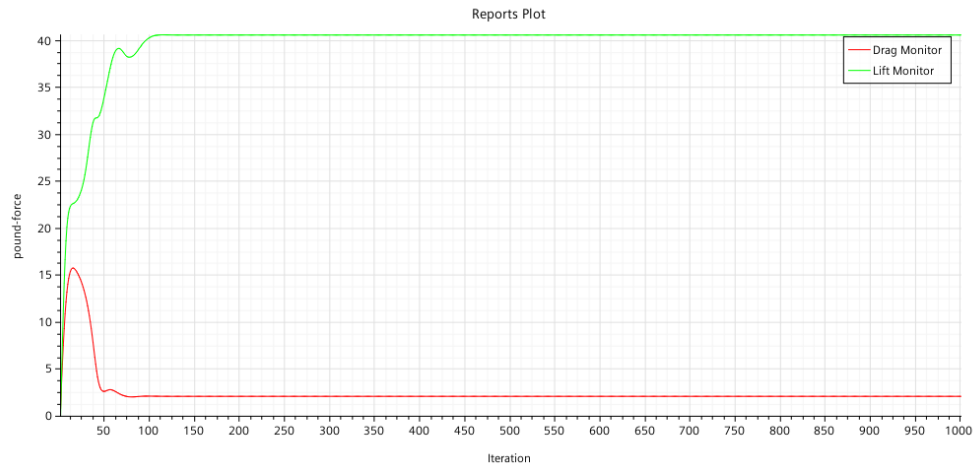


Figure 26: Bellwether Wing-Only CFD - Forces Monitor

is normal to the rotation of the flow (i.e. the rotation is spanwise with each vortex rotating towards the aircraft centerline). The difference in vector directions from port to starboard vortices indicate opposite directionality. The flow outboard of the vortex center is where the region of upwash occurs. The identification of the core location provides a qualitative understanding of the influence on the change in flow field to aerodynamic performance when this work turns to the comparison of the various lift distributions.

Another way to create this view in STAR-CCM+ is to add the wake plane to a scalar scene where vorticity about the longitudinal axis is the scalar quantity (*Figure 31*). The

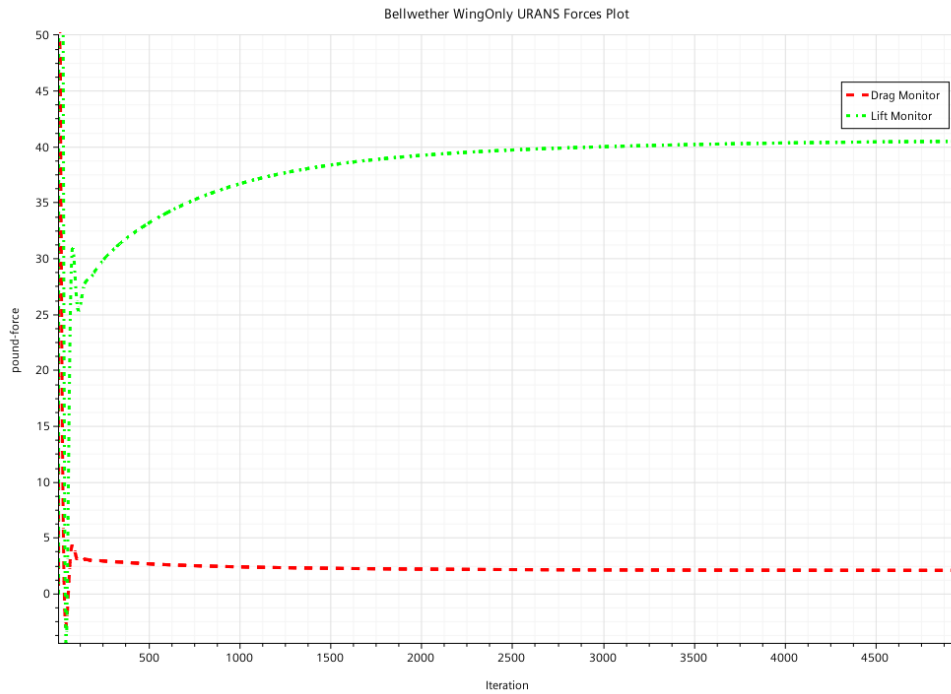


Figure 27: Bellwether Wing-Only Unsteady RANS - Forces Monitor

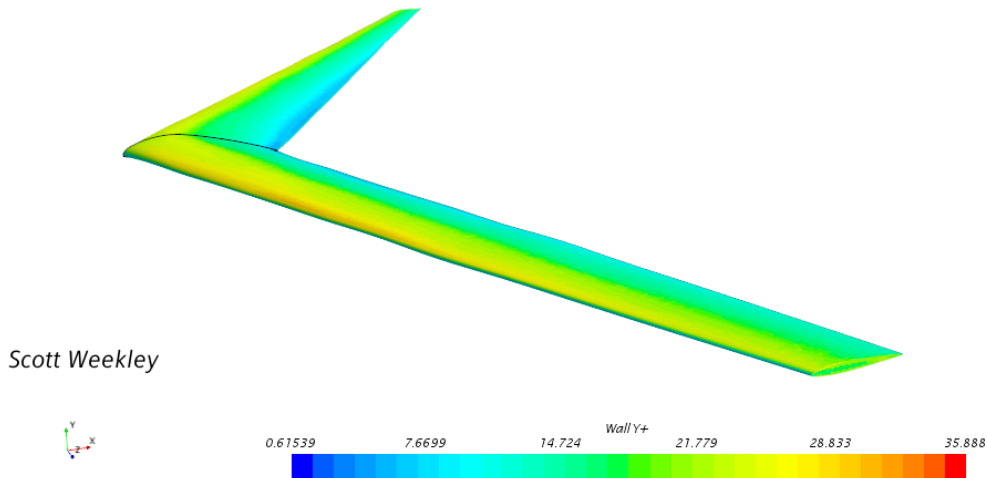


Figure 28: Bellwether Wing-Only CFD - Wall Y+

scalar values are single values stored in each cell lying on the plane. The wake plane is the same derived part used in *Figure 30*. Upper and lower bounds may be set in the properties window to control the color gradient to better illustrate the change in vortex intensity around the core.

Alternatively, the vortex cores may be generated directly in a displayer with moderate computational expense (*Figure 32*). This is accomplished by creating a vortex core as a

Base Size (ft.)	CL	CD	Cell Count
1.25	0.637225	0.032498	4,548,463
1.5	0.641063	0.032832	3,378,926
<b>% ERROR</b>	<b>0.59866</b>	<b>1.01859</b>	

Figure 29: Mesh Resolution Percent Error

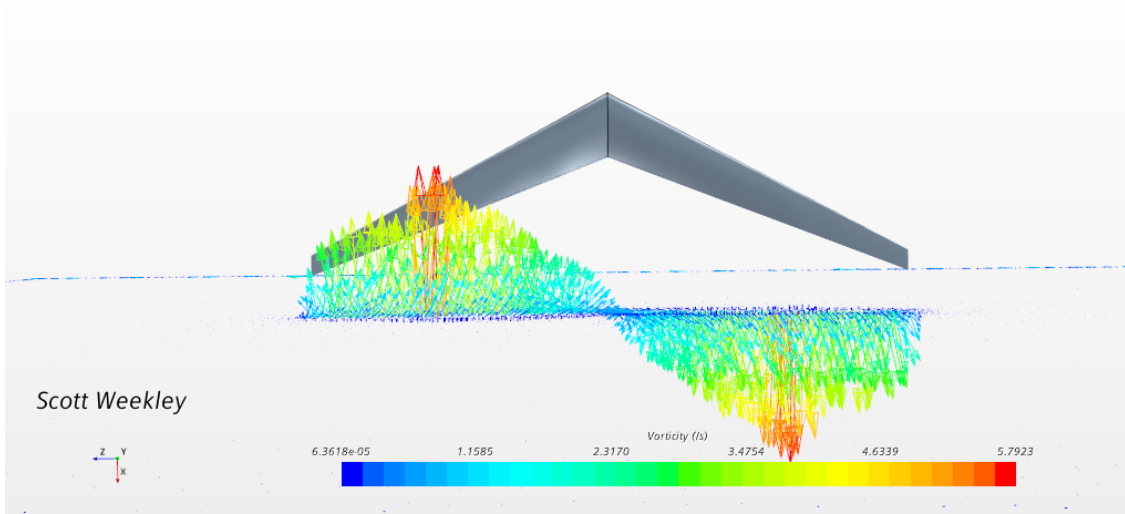


Figure 30: Bellwether Wing-Only CFD - Vorticity Vector Top View

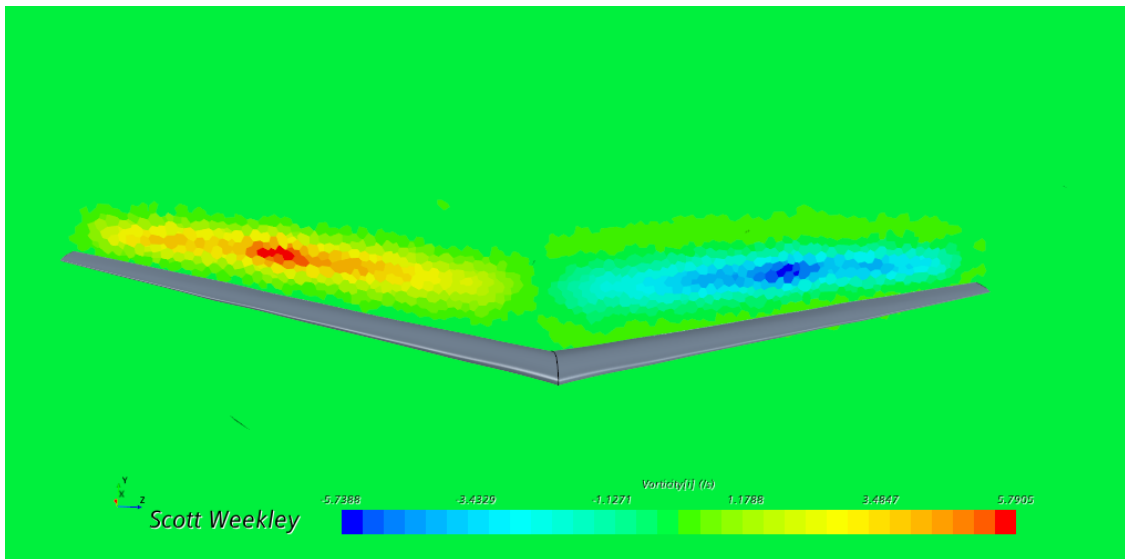


Figure 31: Bellwether Wing-Only CFD - Wake Vorticity Scalar Scene

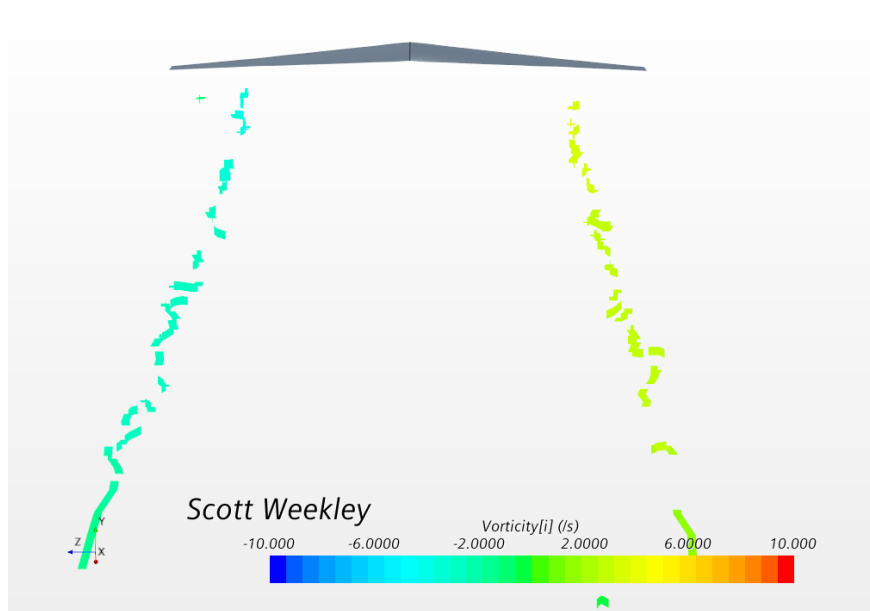


Figure 32: Bellwether Wing-Only CFD - Vortex Core Scene

derived part, choosing the entire flow domain as the part added to the vortex core, and viewing that part in a scalar scene. The scalar field of the scene must be set as the vorticity about the axis of interest.

The qualitative tools aid in connecting this research with the work done by Ludwig Prandtl [11] and Dr. Bowers of NASA [2] in *Chapter V*, but the wing-only analysis progresses with the quantitative results to compare VLM to CFD.

### 4.3 Wing-Only Comparison of Analysis Methods

The alpha sweep in the vortex lattice methods range from -4 to 10. The VLM1 analysis in XFLR5 solved for performance at 2 increments. AVL results were produced in 1 steps. The range of angles of attack in CFD are from -2 to 6 in increments of 2 to reduce the computational burden. *Figure 33* illustrates the linear shift in CL values among the analysis methods, though the slopes are similar. AVL, XFRL5, and STAR-CCM+ generate coefficient of lift values at the zero angle of attack of 0.85, 0.762, and 0.64, respectively. The constrained streamlines shown in *Figure 121* show the 3-dimensional "leakage" of flow captured in the high-fidelity analysis performed by STAR-CCM+. The flow runoff reduces the lift produced at the sections with the most cross flow. This spanwise flow is not observed by the lower-fidelity methods which helps explain the over estimation of lift coefficient by the vortex lattice analyses at the root sections showing the most cross flow. In the comparison of the shape of the lift distributions, the wings used in the VLMs were analyzed at a constant CL of 0.64 to match the value produced in STAR-CCM+ to maintain the imposition of fixed lift and freestream velocity constraints. Geometric rotation of the wing is used to achieve a common CL among the low-order analysis methods. This adjustment requires analysis of the wing in AVL at an angle of attack of -2.2 and an angle of attack of -1.3 in XFRL5. The

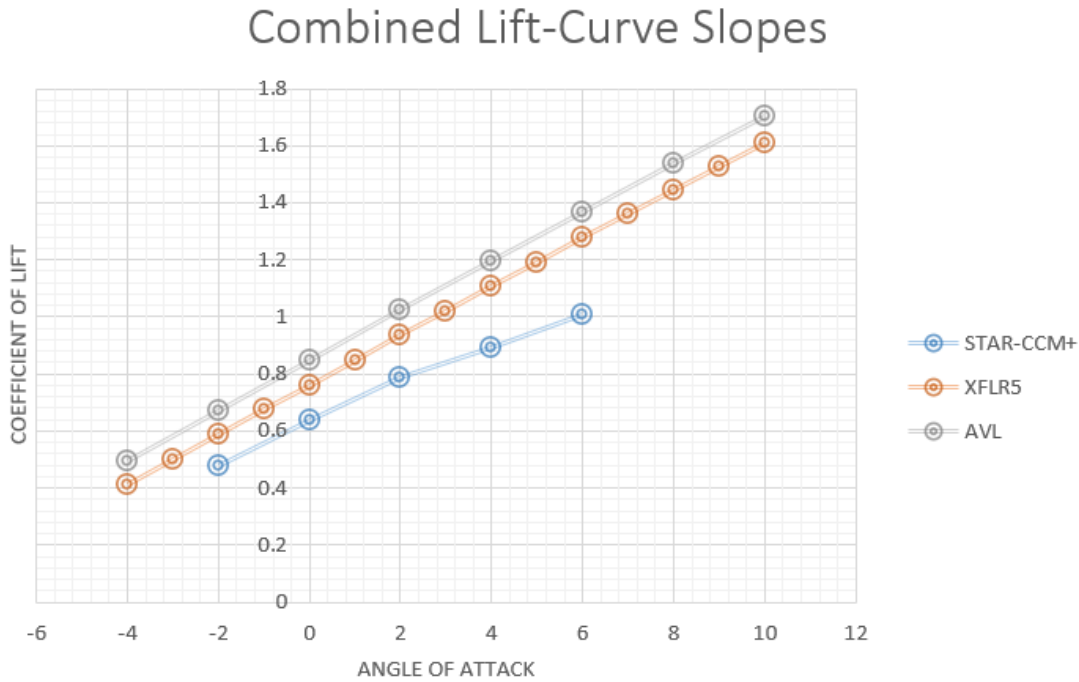


Figure 33: Comparison of Lift Curve Slopes of the Wing-Only Bellwether Model for VLM and CFD Analyses

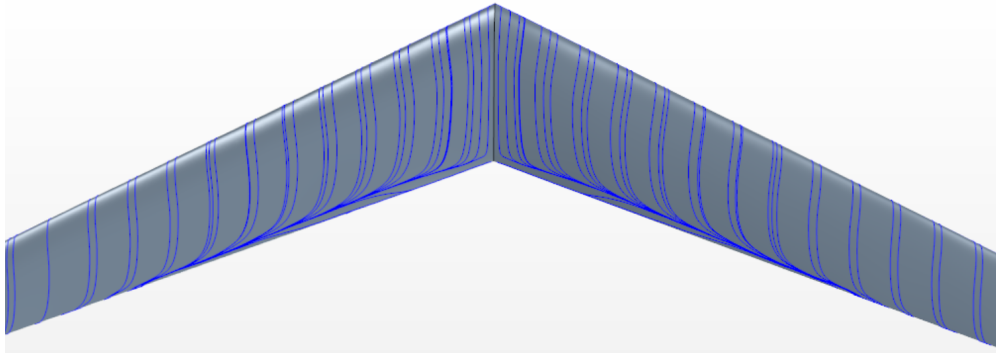


Figure 34: Constrained Streamlines on Bellwether Wing-Only Planform View

resultant distributions of chord multiplied by the coefficient of lift at each section along the wing,  $c*CL$ , is plotted in *Figure 35* against the shape of the differential lift,  $dL$ , distribution output by STAR-CCM+ and the Prandtl curve that represents the desired distribution. Recall that this wing was designed with the lifting-body fuselage in STAR-CCM+ to match the Prandtl curve, and *Figure 35* overlays the wing-only distributions to show commonality of the lower-order methods to CFD to inform the aircraft designer of the efficacy of VLM for rapid results in capturing the flow field. Since  $L'$ ,  $dL$ ,  $c*CL$ , and circulation are scalable distributions of one another, the  $dL$  distribution, shown by STAR-CCM+ results, is scaled by a factor of 0.62 to illustrate its shape in contrast with the vortex lattice results.

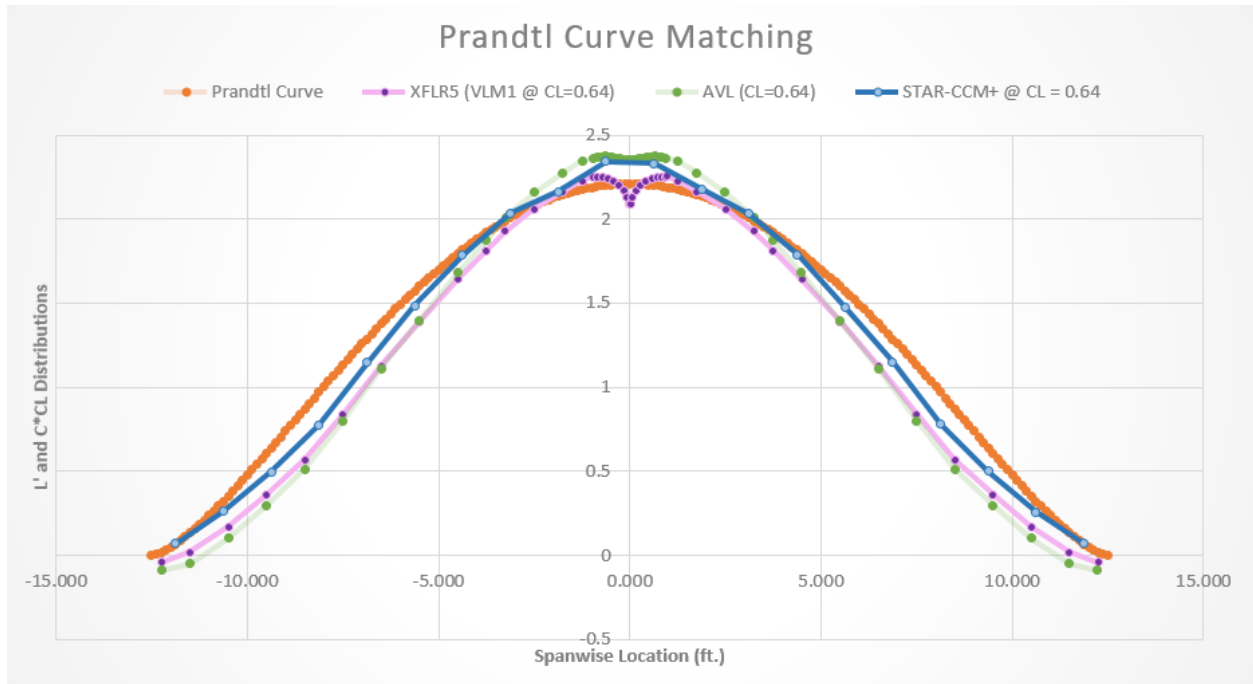


Figure 35: Comparison of Vortex Lattice Methods’  $c^*CL$  Curves to a Scaled Curve of  $dL$  from CFD at a Common Lift Coefficient

### 4.3.1 Mathematical Justification for Curve Scaling

A challenge faced in this research was found when trying to compare the medium-fidelity aerodynamic solutions to the high-order CFD computations. The CFD computations output physics parameters describing the flow at each cell in the computational grid. Overall forces and moments acting on surfaces that the grid has been generated over can be output in post-processing. The solution of the full Navier-Stokes equations at each point in the computational grid means that the overall forces and moments acting on the outer-mold line of the geometry include components from the pressure distribution and skin friction from streamline scuffing in and near the boundary layer. In addition to outputting the total forces and moments at a specified point in the computational domain, the Star-CCM software allows the user to define equidistant chordwise strips on the surface geometry and to isolate force components acting along these strips.

The vortex lattice method discretizes the geometry by placing potential flow singularities, vortex-ring elements in common implementations, over the mean camber line of the lifting surfaces. For the purposes of explanation, the discussion below is considering a simple VLM with only one horseshoe vortex in the chordwise direction. A linear system of equations is developed to enforce the boundary condition of zero-normal flow through the control point of each vortex ring. The boundary condition equation, in the absence of body angular rates, at one vortex ring control point may be expressed as  $(\vec{V} + \sum_{i=1}^n \vec{g}_{ki}\Gamma_i)\hat{n}_k = 0$ , where  $\vec{V}$  is the freestream velocity,  $\vec{g}_{ki}$  is the velocity induced by vortex ring  $i$  at control point  $k$ ,  $\Gamma_i$  is the strength of the vortex ring  $i$ , and  $\hat{n}_k$  is the panel normal vector for control point  $k$ . Writing this equation for each control point in the geometry results in a matrix expression

in the form represented below:

$$[AIC][\Gamma] = [w_i] \quad (4.3.1)$$

In this equation, the aerodynamic influence coefficient matrix ( $AIC$ ) is an  $n \times n$  matrix containing the terms  $\vec{g}_{ki}$  for a unit vortex strength,  $w_i$  is an  $n \times 1$  vector containing the term  $\vec{V}_{n_k}$  (the normal flow component) for each control point based on the current inflow condition and geometric twist at the local strip of the wing. For a given geometry and inflow condition, the  $n \times 1$  circulation strength vector  $\Gamma$  can be solved for by inverting the  $AIC$  matrix and pre-multiplying the vector of normal flow on the right-hand side of the above equation, which is formed by simply considering the freestream flow contribution at each of the panel control points.

Knowledge of the circulation distribution across the span allows for computation of several aerodynamics quantities of interest. Firstly, applying the Kutta-Joukowski theorem gives the spanwise lift distribution:

$$L'(y) = \frac{dL(y)}{dy} = \rho V_\infty \Gamma(y) \quad (4.3.2)$$

Note that the  $dL$  and  $L'$  distributions is the same shape as the  $\Gamma$  distribution since for a given flow solution, the density and inflow velocity are constant.

The incremental lift on segment  $dy$  of span can then be written as:

$$dL(y) = \rho V_\infty \Gamma(y) dy \quad (4.3.3)$$

From the definition of 2-D lift coefficient in airfoil theory, the lift per unit span can also be written as:

$$L'(y) = \frac{1}{2} \rho V_\infty^2 c(y) C_l(y) \quad (4.3.4)$$

Therefore, local lift coefficient at spanwise location  $y$  can be written as:

$$C_l(y) = \frac{2\Gamma(y)}{c(y)V_\infty} = \frac{2}{c(y)V_\infty} \Gamma(y) \quad (4.3.5)$$



For a given wing geometry and inflow condition, it is easily seen that the  $c(y)C_l(y)$  distribution is simply a scaled version of the  $\Gamma$  distribution. Thus, many readily available outputs from a Vortex Lattice Method, or similar, are seen to be the same shape, and simply scaled versions of each other.

$$\Gamma(y) \xrightarrow{\text{Same Shape}} \frac{dL(y)}{dy} = \rho V_\infty \Gamma(y) \xrightarrow{\text{Same Shape}} c(y)C_l(y) = \frac{2}{V_\infty} \Gamma(y) \quad (4.3.6)$$

Induced drag is quite simple to predict in a VLM after the  $\Gamma$  distribution is solved. Traditional VLM formulations calculate the induced drag in the Trefftz plane, theoretically located infinitely far behind the lift system using the equation below:

$$D_i = \frac{\rho}{2} \sum_{i=1}^n \Gamma_i V_{n_i,i} ds_i \quad (4.3.7)$$

In the expression,  $D_i$  is the total induced drag,  $\rho$  is the density of the freestream flow,  $\Gamma_i$  is the bound circulation at spanwise wing station  $i$ ,  $V_{n_i,i}$  is the velocity normal to the wake in the Trefftz plane at spanwise station  $i$ , and  $ds_i$  is the incremental distance along the span at section  $i$ . As the wake normal velocity in the Trefftz plane is simply the sum of all the velocity increments from each bound and trailing vortex in the lift system, the total induced drag is simply a function of the resultant  $L'(y)$  distribution.

The lift distribution shape is useful in an aerodynamics optimization problem as it fully determines the induced drag that will result from producing the required amount of lift with a given spanload. The lift distribution is also useful for determination of wing structural loading and the overall wing-root bending moment. Because the spanload distribution is designed using a combination of geometric and aerodynamic twist, it is critically important to remember that  $L'(y)$ ,  $\Gamma(y)$ ,  $dL$ , and  $c(y)C_l(y)$  distributions will all be the same shape and simply scaled versions of one-another but that the spanwise distribution of local  $C_l$  will result in a different shape because the chord is not constant along the span.

## Induced Drag Comparison from VLM and CFD

The induced drag is the entirety of the drag force predicted by the VLM implementation as no physics to include any contributions from skin friction terms are not included in the potential flow formulation. Conversely, the CFD drag prediction contains both the lift induced drag and contributions from viscous effects. There is no simple way to isolate only the lift induced drag portion of the CFD solution (without the computational expense of a Trefftz plane as a derived part [14]) or to reliably incorporate  $C_{Dp}$  type drag contributors in a VLM

formulation.

Therefore, the knowledge that induced drag is entirely due to the shape of the resultant spanwise lift distribution is applied to compare the VLM and CFD solutions. From the VLM, readily available outputs include the spanwise chord distribution and local section  $C_{ls}$ . These can be multiplied together and plotted as a representation of  $\Gamma$  or  $dL$ . In the CFD post-processing, the lift over a strip of the wing can be output and co-plotted with the VLM  $c*Cl$  distribution. The resultant curves are not expected to be of the same magnitude, but should be the same shape. Therefore, the plots are made by applying a linear scale factor to one of the curves to make the peak of the curves overlay each other. Then, the shape of the curves from the two methodologies can be directly compared. If the shape is the same, not only is the resultant lift distribution from each method equivalent, the induced drag will be the same as well.

## CHAPTER V

### CFD Analysis of Bellwether

The previous section validates the use of vortex lattice methods to develop rapid results for the aircraft designer. This section turns to how CFD is leveraged to evaluate aerodynamic performance of the BSLD and how it compares to its elliptical wing counterpart. The analyses performed onward in this body of work will be done using high-fidelity tools on aircraft designs with the fuselage included to best capture performance. Beginning with CFD validation techniques used in the previous section, this section progresses through high-fidelity computation for the comparison of lift-to-drag ratios of the four different wing models at level, steady flight and roll-yaw coupling in an aileron-deflected banked turn. Interpretation of vorticity vector scenes describe upwash regions in the aircraft wake providing discussion of the relationship of inboard upwash to proverse yaw.

#### 5.1 Mesh Development

After importation of the aircraft parasolid models, the surfaces are split by patch for additional control of mesh settings. Select surfaces are broken up by sections: wing leading edge, wing trailing edge, central patches of both wings, wingtip leading edges, wingtip trailing edges, wingtip mid-section, fuselage nose, fuselage body, and fuselage tail, which are selected to be displayed in the planform view of the scene shown in *Figure 41*. This offers specific control of these areas to directly command scalar values of interest, such as wall  $y^+$ , turbulent kinetic energy, and convective courant number. These values can be used to provide guidance in the iterative mesh design process to achieve convergence of physical flow properties. Prism layer design and wake refinement are two vital parameters in capturing the flow physics. *Figure 36* and *Figure 37* visually depict the outcome of these designs. It is pertinent to note, for purposes of computational expense, that iterations of physics runs are used to refine the prism layer design and base cell size to create a mesh that begins to reduce residuals across iterative steps, and limit wall  $y^+$  and turbulent kinetic energy values, before wake refinement is enabled. This is done to capture the surfaces of the model through iterative mesh/run steps before adding wake refinement which is likely to introduce millions more cells into the domain increasing computational cost. Once the mesh design is completed, and wake refinement enabled, analysis proceeds to check mesh independence by evaluating the current mesh with a mesh of 20% reduced base size with all mesh elements enabled as "relative to base."

The physics for the aircraft models without aileron deflection are run on a mesh of 23

million cells. The computational burden is placed on the High-Powered Computing Center (HPCC) at Oklahoma State University.

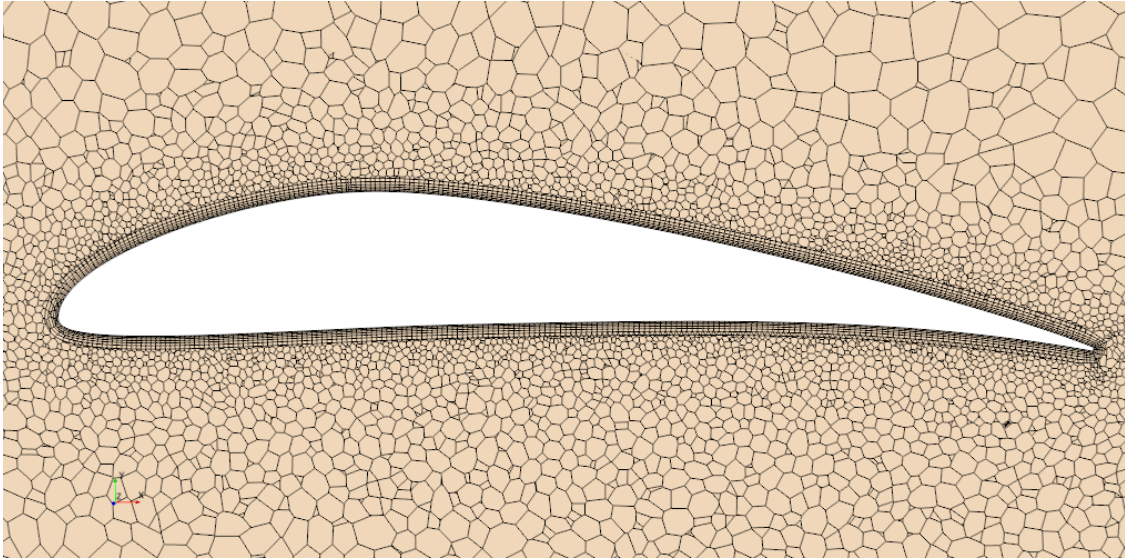


Figure 36: Longitudinal Mesh at the Root Airfoil

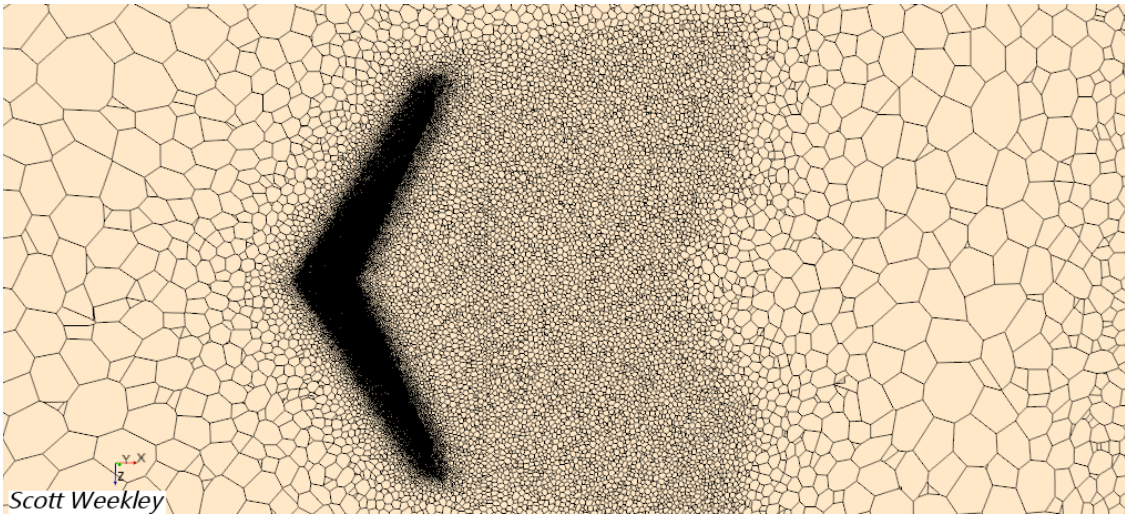


Figure 37: Horizontal Mesh with Wake Refinement Enabled

## 5.2 Level, Steady Flight Performance

Each model is first analyzed at the 0 degree angle of attack at a freestream velocity of 35 fps. The performance was originally recorded by the unsteady Reynold's-Averaged Navier Stokes (U-RANS) solver for each model for 1 physical second of 5000 iterative steps. The results, albeit small and bounded, were oscillatory. A steady RANS model was introduced to provide a level, convergent result within the bounded U-RANS results for clarity of presentation.

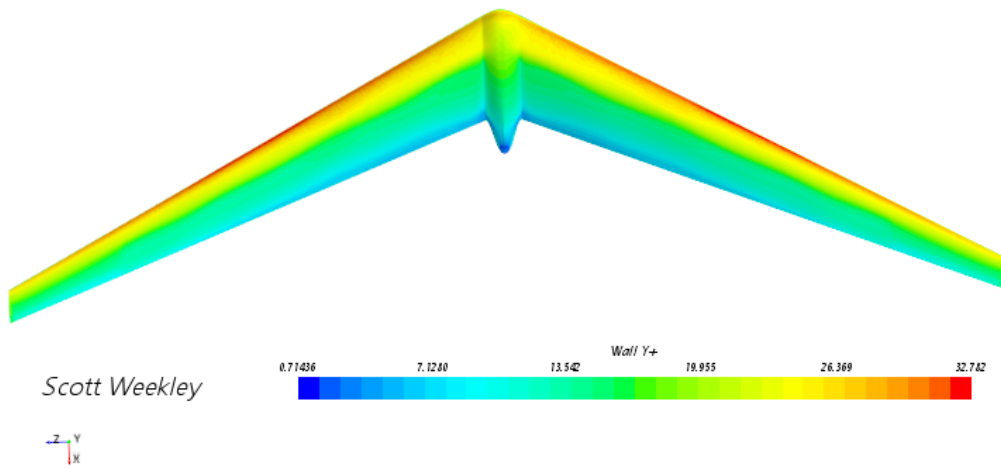


Figure 38: Bellwether Wall Y+

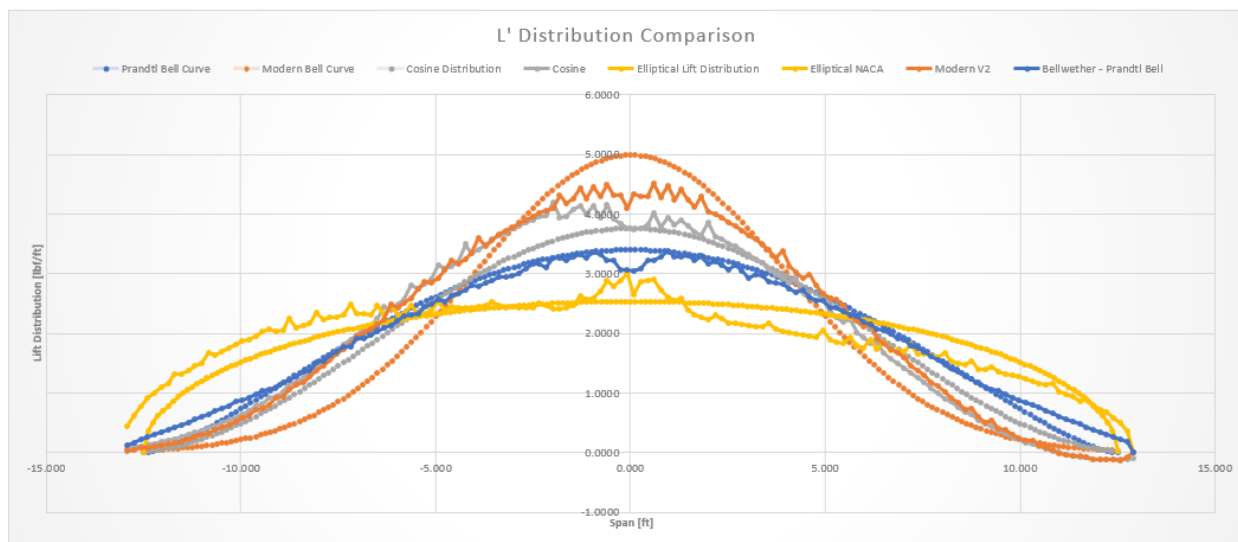


Figure 39: CFD-Matched Lift Distributions

The lift distributions,  $dL$ , across the span are plotted against the desired curves in *Figure 39* to confirm each wing is producing lift congruent with how it is designed. Next, the aerodynamic performance of each is measured in the ratio of lift to drag produced at the common flight condition. To save on computational expense, the Modern Bell and Cosine curves have not been iterated extensively to match their curves. Instead, they have been designed with enough distinction to provide significant differences to the flow fields produced by the Prandtl and ELD wings. The Bellwether design, applying the bell spanload outlined by Ludwig Prandtl, shows greatest efficiency in level flight (*Figure 40*). The ELD wing is found to be operating at a much lower coefficient of lift (0.323) than the other distributions, so this comparison of the ELD to the other wings on this basis is ill-founded. The ELD design failed to adhere to the fixed-lift design constraint placed upon the other distributions. This can be corrected by running an alpha sweep and comparing the performance of the

ELD at an incidence angle where the CL matches that of the other distributions. However, this method is likely to introduce greater parasitic drag, so a  $L/D$  comparison would not be representative of the efficiency of lift generation. A comparison could be made in setting up a Trefftz plane in STAR-CCM+ as a grid of planar probes from which the velocity vectors recorded are used to calculate induced drag from the flow field. This may be a suitable method of comparing induced drag computationally, and the truest comparison of aerodynamic efficiency in level flight, but it cannot entirely capture the aerodynamic performance without also accounting for the parasite drag by measuring the total drag and calculating induced drag ( $C_D = C_{D0} + C_{Di}$ ). No clear comparison in level flight is made to ELD, but it is determined that increasing the centralization of lift deviating from the bell spanload results in decreased aerodynamic efficiency for this aircraft configuration in this flow regime.

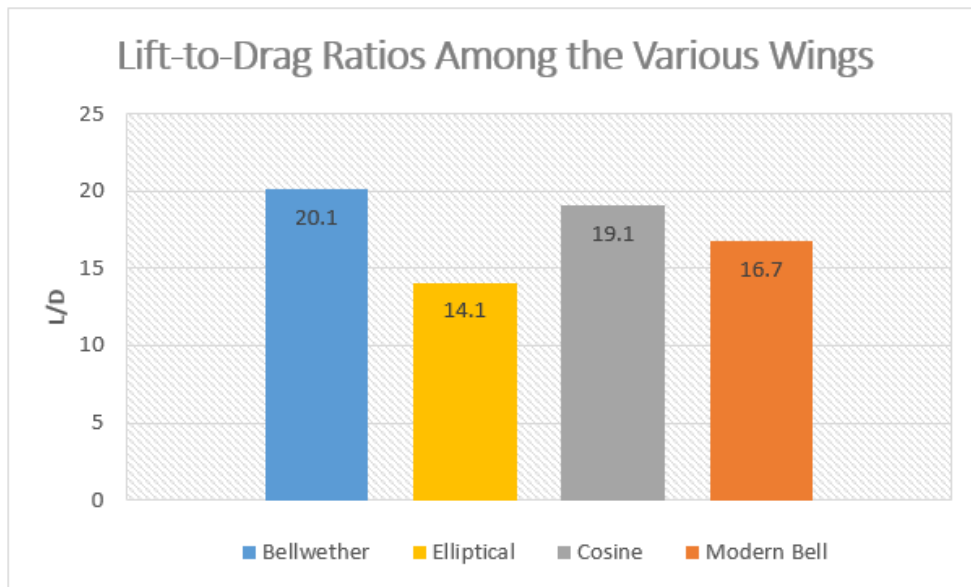


Figure 40: Comparison of Lift-to-Drag Ratios of the Various Wing Designs

For a physical explanation of how the change in lift distribution affects the roll-yaw coupling response of the aircraft, we revert to the discussion of the regions of upwash and downwash discovered by Dr. Bowers and the research team at NASA responsible for the design and testing of the Prandtl-D flying wing [2]. Referencing *Figure 2* depicts the sharp induction of upwash flow at the wingtips caused by the elliptical lift distribution. The bell spanload offers a smoother transition between regions of upwash and downwash that begin within the wingspan. Depictions of trailing vortices for Bellwether and the ELD model are observed to relate the computational findings to the Prandtl-D research and avian flight correlations [2] [14].

*Figure 41* and *Figure 42* use vorticity vector scenes from STAR-CCM+ to illustrate where these regions occur for Bellwether and the Elliptical model, respectively. ELD (*Figure 42*) places the center of its vortex at the wingtip with a sharp gradient in vorticity vector strength. By contrast, BSLD (*Figure 41*) captures a much smoother change in vorticity

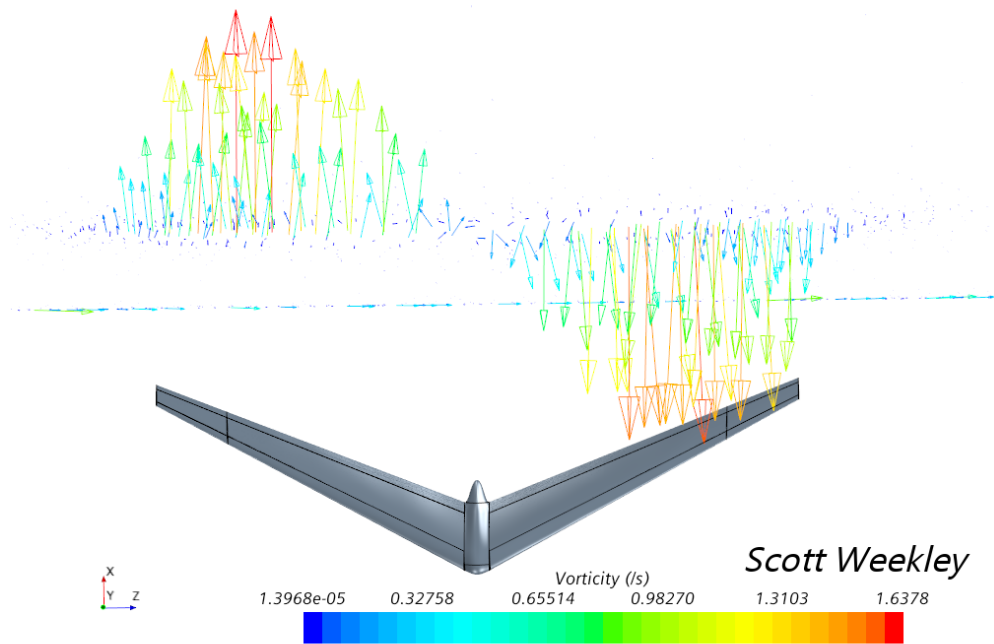


Figure 41: Bellwether Vorticity Vector Scene

magnitude congruent with *Figure 2*. The measurement tool in STAR-CCM+ was used to measure the location of the maximum vorticity vector to the aircraft centerline which spans the region of downwash and indicates where the transition from upwash to downwash occurs (i.e. distance from the root to the vortex core). This distance measures 9.17 feet which is 70.5% of the 13-foot semispan. The measured vortex core location nearly exactly mirrors the 70.4% semispan location found by NASA’s investigation of this lift distribution in 2016 [2].

### 5.3 Banked Turn Performance

In order to simulate a banked turn, the aircraft models are revised with the addition of ailerons from the 80% span position to the RAE 101 airfoil section at the wingtips. This design choice was advised by the range of optimal placement between 76% and 82% found by Richter et al [18] for maximum proverse yaw control. The inboard and outboard surfaces of the ailerons extend to the quarter chord, and they are linearly lofted together as a solid. They are designed such that the angle of deflections can be easily modified for a downward deflection on the port side aileron and an upward deflection on the starboard aileron. This banked turn simulation deflects the ailerons 20 anti-symmetrically.

Like the level-flight analysis, the banked-turn CFD records yaw moment by the implicit U-RANS solver for each model tested. The results presented are from the subsequent analyses using the steady RANS solver to output a clear value within the oscillatory bounds of the U-RANS result. This is done primarily for validation of the physics continuum and clear

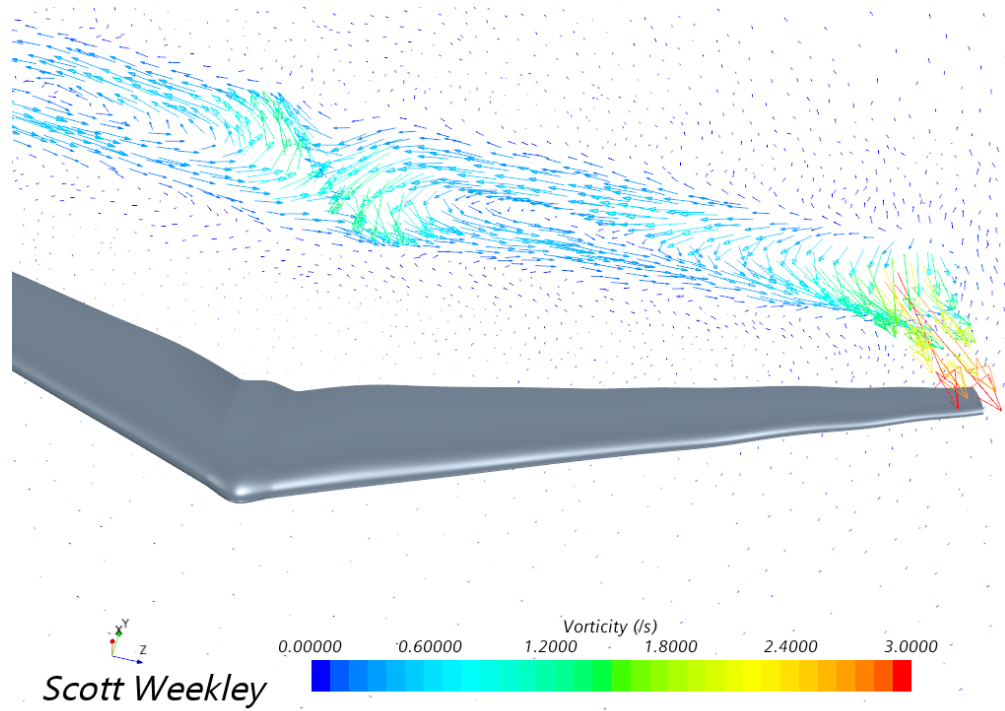


Figure 42: Elliptical Model Vorticity Vector Scene

presentation of results. The creation of proverse yaw (*Figure 43*) is evident for the wing designs in which the ailerons are engulfed in the upwash. The distributions centralizing lift the most experience the greatest effect of proverse yaw. Bellwether, utilizing the spanloading described by Prandtl, exhibits the greatest aerodynamic efficiency in level flight (*Figure 40*) and is the closest to performing a perfectly coordinated turn in roll (*Figure 43*). Deviations from this spanloading shows decreases in overall flight performance. As discussed in the level flight section, the  $L/D$  comparison does not apply to the elliptical wing. The ELD is reported to be the most efficient solution for a fixed wingspan. Since the analyses of this work does not alleviate wingspan constraint for a fixed structural weight (or root bending moment), previous research would lead to the belief that this design should perform best when the ailerons are not engaged. As expected, the ELD operates with clear adverse yaw effects in roll. *Figure 44* tracks the convergent yaw values for each physics iteration. The first 100 iterations of the steady RANS results have been clipped to remove extraneous values from the plot.

Examination of an aileron mesh scene (*Figure 45*) encourages discussion on improvements to be made in the modeling and mesh design for the aileron-deflected models. These models have been made such that the port aileron deflects 20 down with attachment along the high-pressure surface of the wing. Conversely, the starboard aileron is deflected 20 up with attachment at the upper wing surface. The deflection angle creates sharp edges of surface discontinuity. CAD features such as this should be avoided for purposes of importation as surface mesh for CFD simulation. Notice the prism layer contraction to the sharp edge causing the thin, prismatic layer to no longer follow the flow direction to precisely capture



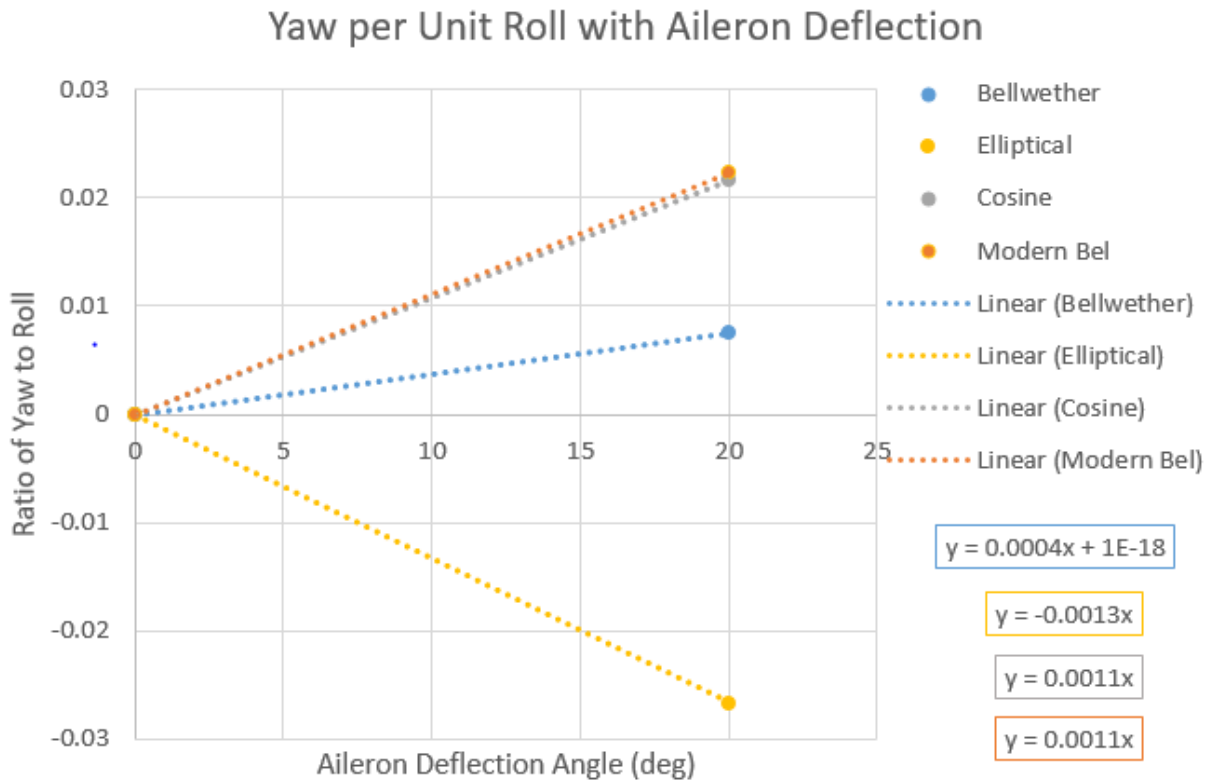


Figure 43: Yaw Control Power per Unit Roll from 20 Anti-Symmetric Aileron Deflection

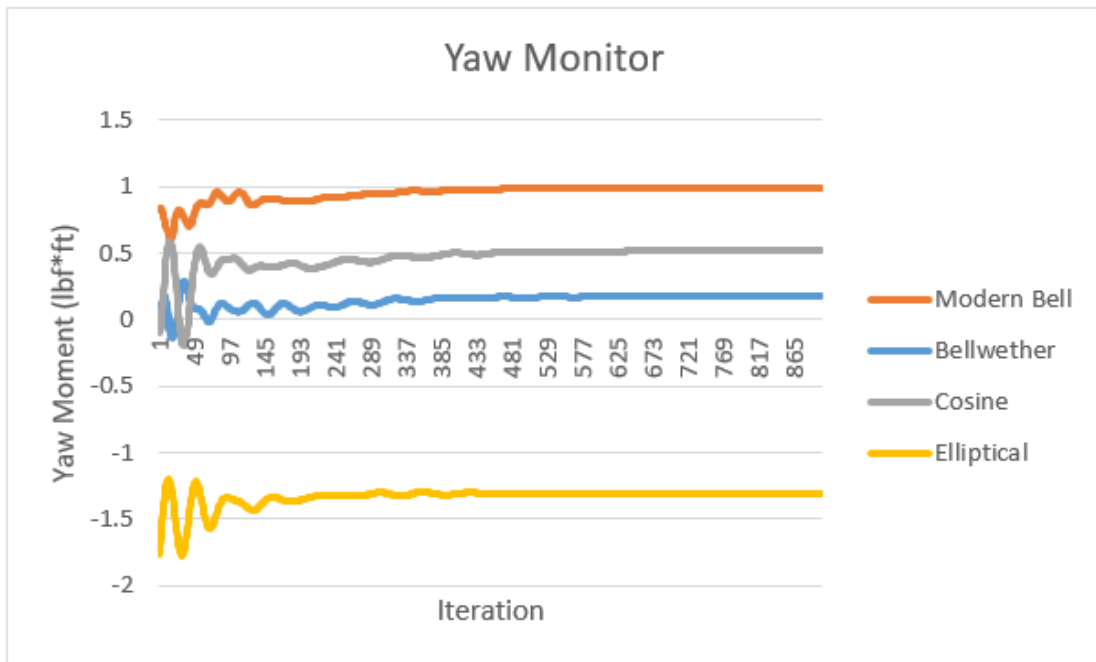


Figure 44: Yaw Moment Monitor by Iteration of Steady RANS Solver in STAR-CCM+

velocity gradients close to the surface boundary. Future CAD modeling could create a filled boundary by creating planar surfaces in the gaps of the ends of each aileron and a surface bounding the upper surfaces. This may also be remedied using a solid loft feature from creating a sketch profile in the corner gaps at the ends of each aileron and using the attachment surface as a guide curve to avoid a potential non-manifold-body error in STAR-CCM+.

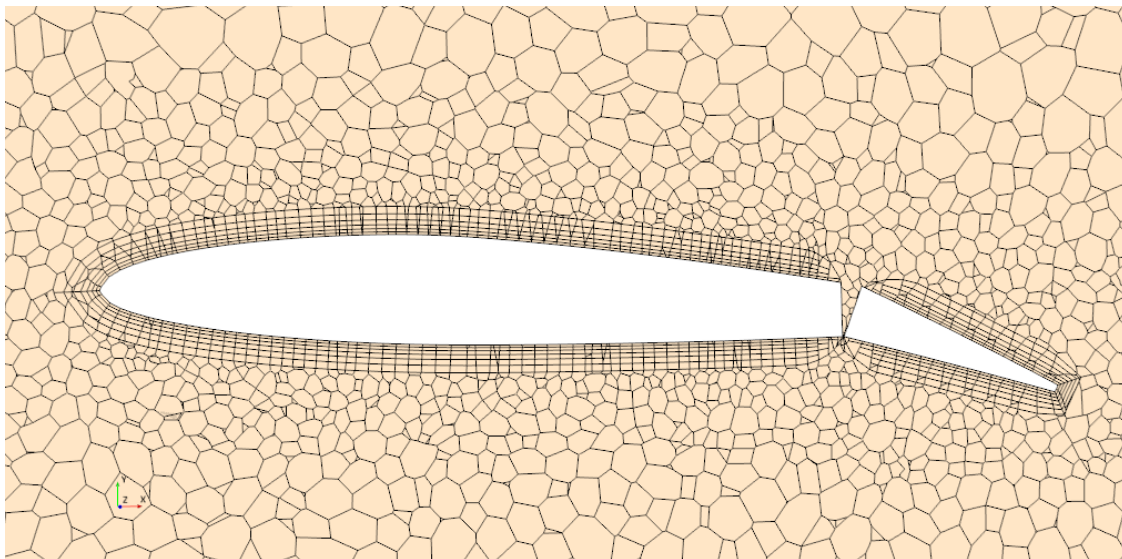


Figure 45: Aileron Mesh Scene on Longitudinal Plane

A smooth, continuous surface allows for greater control of prismatic layer design and, in turn, improved accuracy of velocity and pressure gradients around the body. It can also be seen in *Figure 45* that the trailing edge curvature is not matched causing a sharp, angular feature. Again, the prism layers are forced to turn at the sharp angle instead of rounding smoothly. The aileron surface mesh should be imported at an increased tessalation density to build small surfaces of high curvature with greater accuracy. This must be coupled with refined mesh settings on the control surface such as decreasing minimum surface and target surface sizes.

## CHAPTER VI

### Avian-Inspired Aircraft Design and Performance

The previous sections detail analytical and computational methods to justify the aerodynamic design of Bellwether. The aerodynamic shape has been driven by the application of the BSLD and artistic resemblance to birds with the shaping of the lifting-body fuselage. This section discusses aircraft performance metrics for stability and control from computational results and progresses to CAD modeling of the final aircraft shape, internal structure, and basic construction techniques.

#### 6.1 Longitudinal Stability and Performance

Understanding of the aircraft's performance begins with unpacking the results of an alpha sweep. The results give a graphical understanding of longitudinal performance across incidence angles (*Figure 46*) to illustrate the range for optimal performance and the stall region where the linear, elastic region of lift loses its efficacy. The range of greatest lift-to-drag ratio can be seen between  $-2$  and  $2$  operate at high coefficients of lift ranging from 0.7 to 1 for maximum aerodynamic performance. This is optimal for the aircraft to complete its mission of atmospheric data collection while demonstrating minimum induced drag.

To understand stability of the aircraft the center of gravity must be estimated after modeling the aircraft and all of the internal components. The center of gravity (c.g.) location is approximated by a spreadsheet of point masses in cooperation with the SolidWorks c.g. displayer to model each component and report the location with respect to the model. Density of aircraft composite skin made of fiberglass layers and a divinycell foam core is approximated based on previous aircraft fabricated through Oklahoma State University's long history of building unmanned aircraft. Other known internal components (Pixhawk OrangeCube autopilot, electric motor, RunCam camera, video transmitter, receiver, 12" propeller, servo motors, and batteries) are available in the Unmanned Systems Research Institute (USRI) lab space, *Excelsior*, to be sized and weighed for accurate modeling. The c.g. location is used as the origin for many aircraft stability concerns through STAR-CCM+ including the reports and monitors constructed to track the coefficient of moment about the c.g. Since this aircraft configuration operates without the use of a tail, there is little margin for error in the placement of the c.g. for maintaining longitudinal stability. As not to induce a positive or negative pitching moment that would require additional control, the propulsor placement is designed to run through the center of gravity in line with the freestream, or longitudinal axis. *Figure 47* shows how this design works with a 3D-printed, fin-like support

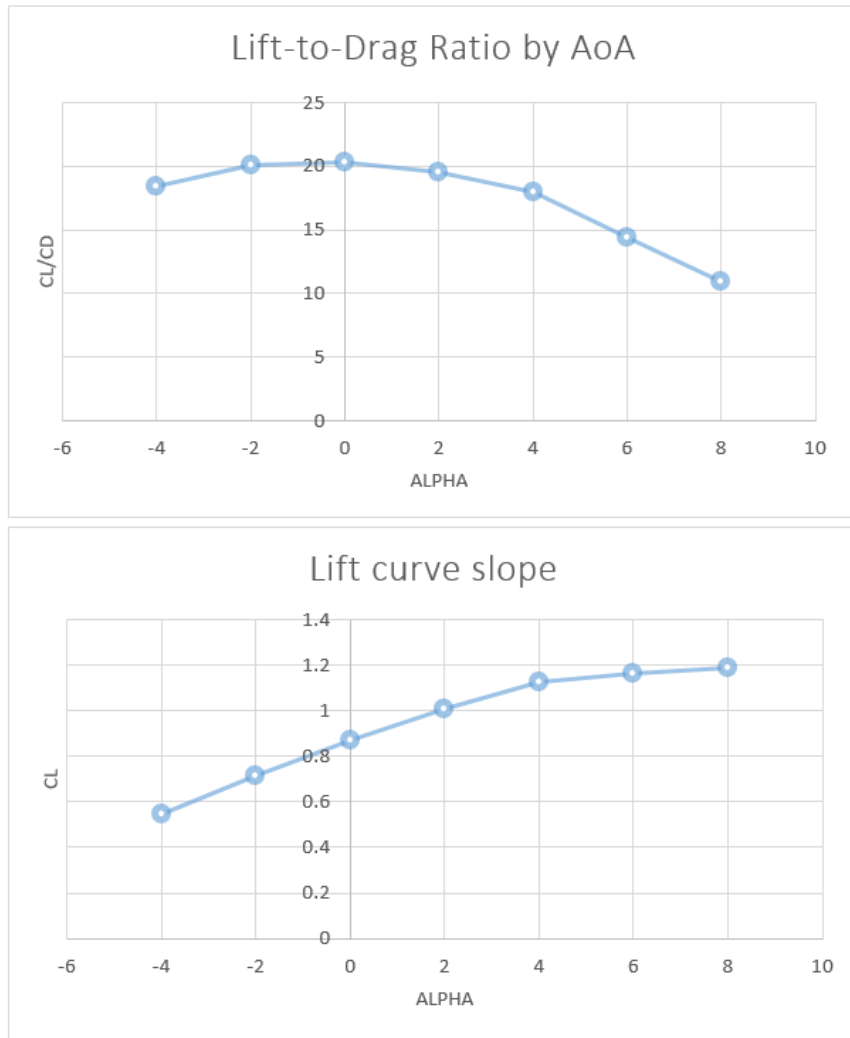


Figure 46: Bellwether - Lift Curve and L/D Developed by Alpha Sweep Analysis in STAR-CCM+

structure for the propulsor shaft. This addition is adhered to the composite fuselage, and the motor shaft slides through the fuselage and the fin to drive the propeller. This design allows for the CFD analysis of pitch stability to be representative of the aircraft behavior without needing to account for the pitching moment induced by the propulsive system. Additionally, the propeller placement was chosen to operate aft of the fuselage as not to influence the freestream flow on the fuselage, since the lift generated by the body was included in the design and analysis to match the Prandtl bell lift distribution.

For longitudinal stability, the pitching moment about the c.g. for various angles of attack is the primary plot used to understand the aircraft's behavior in flight. For a positive moment corresponding to a pitch up, and negative for pitch down, *Figure 48* plots  $C_m v. \alpha$  for a range of c.g. locations along the longitudinal axis. The purpose for ranging the point of interest:

1. Finding the aerodynamic center which, for a flying wing, is equivalent to the neutral point and is the location about which pitching moment is constant and independent

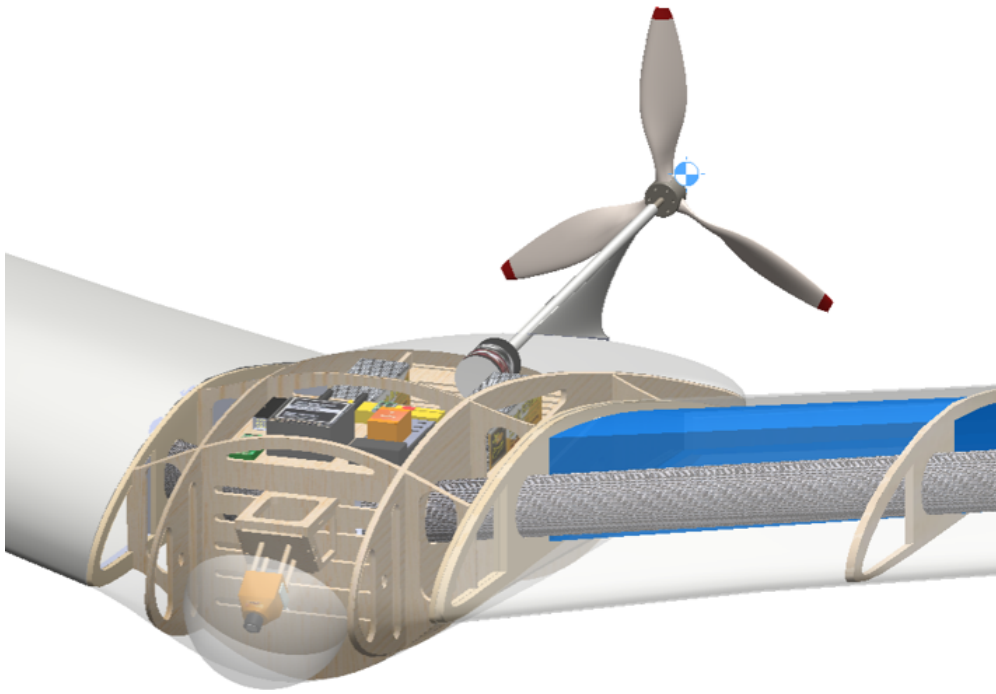


Figure 47: Propulsor Placement in Line with Center of Gravity ISO

of incidence angle.

2. Graphically identifying the c.g. locations to satisfy longitudinal static stability:  $Cm_\alpha < 0$ ,  $Cm_{\alpha=0} \geq 0$ , and it must cross the x-axis to achieve trim.
3. Noting the range of static margin required for longitudinal static stability, where the static margin is equivalent to the distance of the center of gravity from the neutral point divided by the mean aerodynamic chord;
 
$$SM = \frac{(X_{np} - X_{cg})}{MAC} \times 100\%$$

The optimal point can be seen to be a static margin, SM, of 20% to satisfy the enumerate criteria above. At this c.g. location the  $Cm_\alpha$  slope crosses the x-axis at 0 angle of attack, meaning there exists no positive or negative pitch moment at this trim point, and it will seek to return to this operating point when subject to perturbations. Conveniently, the trim point occurs at the angle of attack of greatest efficiency by metric of  $L/D$  as graphed above (*Figure 46*). The computer-calculated center of gravity marked the location at a SM of 26.5%. Bellwether has been designed with a modular propulsion bay that slides out the back hatch of the fuselage to swap or rearrange the propulsive components. Rails are designed within the bay to allow for the batteries powering the electric motor to be positioned at various longitudinal locations to set the c.g. to achieve the desired SM. *Figure 48* provides a top view of the fuselage to show the c.g. relation to the body and the c.g. travel necessary to attain the SM corresponding to the desired performance curve. Each criterion for longitudinal static stability has been met and, thus, concluding analysis for pitch-axis control.

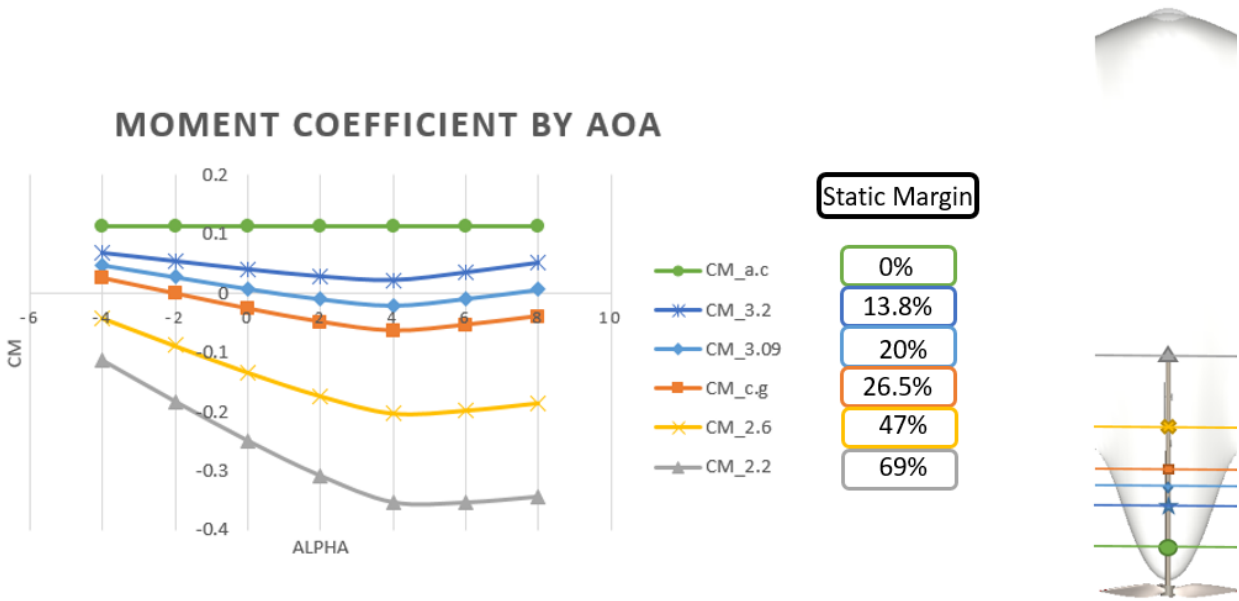


Figure 48: Longitudinal Static Stability from CM v. Alpha Plot for Varying Center of Gravity Locations

## 6.2 Lateral Stability of Bellwether

In order to better understand aircraft performance in a banked turn, the effect of sideslip to yaw and roll must be in account. To observe this effect, CFD analysis is performed on Bellwether for freestream sideslip angles of 2.5, 5, 7.5, and 10. The aircraft is rotated inside the flow domain by an exposed parameter responsible for setting yaw angle in degrees. If unchanged, the lab coordinate system will remain to output forces and moments in accordance with a zero-sideslip condition. This fixed coordinate system is said to be in wind frame, since it maintains an axis along the direction of the freestream regardless of aircraft position. However, the desired outcomes of forces and moments are considered with respect to the body of the aircraft, so the reports and monitors must be made to reflect its changes in position. An adaptable coordinate system is not made available in *Flying-WingMacro.java*, but the change can be made with relative ease. A new coordinate system can be developed in the CAD window of STAR-CCM+ relating the points and/or planes constructing the Cartesian coordinates to the aircraft body so the reference system changes with body rotation and translation. A less adaptable method, but equally effective, is to define a new local coordinate system under the "Tools" tab of the main page. It can be defined as Cartesian, cylindrical, or spherical. For this application, it is easiest to create a Cartesian system that sets an axis as a rotated vector from the original Cartesian coordinates.

Alternatively, the resultant forces and moments may be passed through a rotation matrix to correct for the change of frame as was done here. A Direction Cosine Matrix (DCM) changes the resultant vectors of roll, pitch, and yaw moments from wind frame to body frame (*Equation 6.2.1* and *Equation 6.2.2*) [23].

$$\begin{bmatrix} C_l \\ C_m \\ C_n \end{bmatrix}_{body} = DCM_{bw} \begin{bmatrix} C_l \\ C_m \\ C_n \end{bmatrix}_{wind} \quad (6.2.1)$$

The coefficients of these moments are transformed for each  $\beta$  value of 2.5, 5, 7.5, and 10 for a constant angle of attack,  $\alpha = 0$ .

$$DCM_{bw} = \begin{bmatrix} \cos(\alpha)\cos(\beta) & \sin(\beta) & \sin(\alpha)\cos(\beta) \\ -\cos(\alpha)\sin(\beta) & \cos(\beta) & -\sin(\alpha)\sin(\beta) \\ -\sin(\alpha) & 0 & \cos(\alpha) \end{bmatrix}^{-1} \quad (6.2.2)$$

Roll and yaw coefficients ( $C_l$  and  $C_n$ , respectively) are plotted from the beta sweep (*Figure 49*). The analysis indicates that a negative roll moment is created for positive sideslip with a slight yaw moment into the freestream suggesting that static directional, or weathercock, stability is inherent. The trendlines of the figure indicate the roll and yaw stability coefficients to be  $C_{l\beta} = -0.0086$  and  $C_{n\beta} = 0.0008$ , respectively.

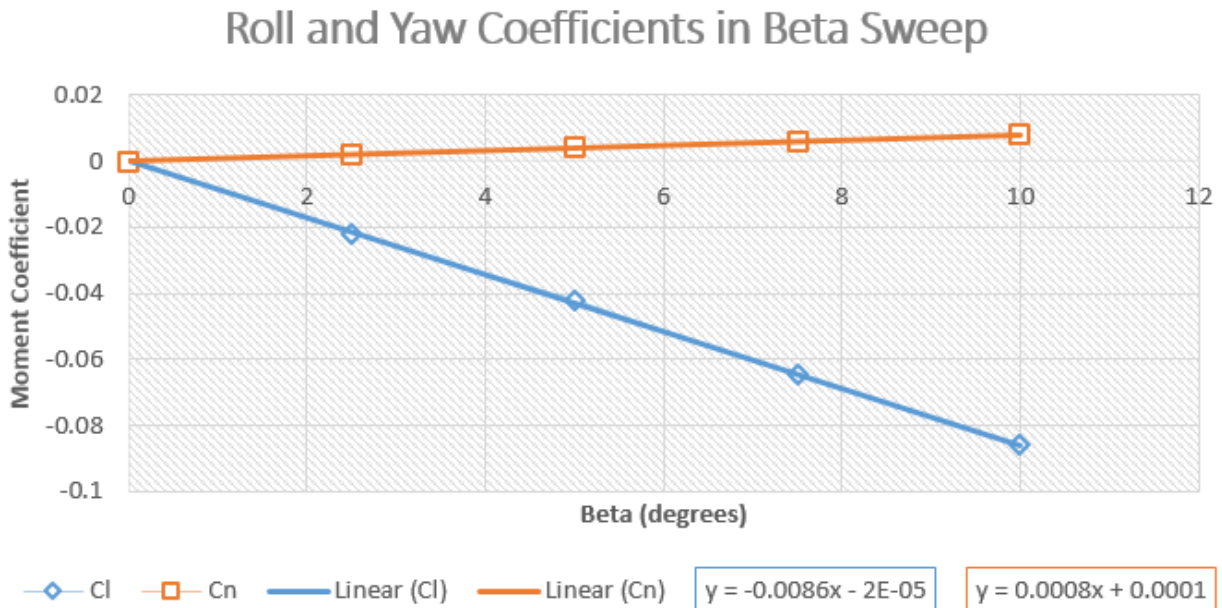


Figure 49: Coefficients of Roll ( $C_l$ ) and Yaw ( $C_n$ ) with Respect to Sideslip Angle ( $\beta$ )

Perfect coordination would assume zero sideslip, but that may not be the case in flight. Relating these results back to the banked turn performance offers a more comprehensive understanding of the turn coordination. The coefficient of roll for the 20 aileron deflection,  $C_{l\delta=20}$ , is approximately 0.25 from CFD results. From *Figure 49*, this positive roll moment is only slightly counteracted in sideslip, so it is acting only marginally as a wing leveler. If the linear relationship holds for greater angles of sideslip, it would take a  $\beta$  of 30 to nullify the roll moment caused by 20 aileron deflection. The yaw moments created are much less pronounced. The yaw coefficient with respect to sideslip (per degree) is 0.0008. The proverse yaw created by aileron deflection is roughly  $C_{n\delta=20} = 0.0018$ , which is to say the effect of sideslip is additive to proverse yaw. The Modern Bell and Cosine wing models operate with

yaw control is excess of  $C_{n\delta=20} = 0.005$ . By contrast, the Elliptical wing creates a coefficient of nearly  $C_{n\delta=20} = -0.007$  that appears to be negated at 8.75 of sideslip according to the linear trend reported by STAR-CCM+. A tradeoff made by seeking more centralized lift distributions for increase yaw control power may not be worth sacrificing  $L/D$  performance in level flight, since the directional stability will add favorable yaw in sideslip.

### 6.3 CAD Modeling, Internal Design, and Fabrication Techniques

The modeling begins with the importation of .dat files as x,y,z coordinates into SolidWorks to create splines of specific airfoil geometry. From these splines, sketch blocks are created so that they may be inserted as different sizes at different locations and angles of twist. The airfoils are positioned from *BellwetherOne* at the root to *RAE 101* at the tip of the wing in accordance with *Figure 50*.

<b>BellwetherOne</b>	<b>BW8</b>
<b>6.16°</b>	<b>3.68°</b>
<b>BW2</b>	<b>BW9</b>
<b>7.14°</b>	<b>1.2°</b>
<b>BW3</b>	<b>BW10</b>
<b>6.3°</b>	<b>1.04°</b>
<b>BW4</b>	<b>BW11</b>
<b>6.24°</b>	<b>0.2°</b>
<b>BW5</b>	<b>BW12</b>
<b>5.87°</b>	<b>-0.8°</b>
<b>BW6</b>	<b>BW13</b>
<b>5.08°</b>	<b>-1.8°</b>
<b>BW7</b>	<b>RAE 101</b>
<b>4.1°</b>	<b>-2°</b>

Figure 50: Geometric Twist Angles for Each Spanwise Airfoil

The semispan wing sections are lofted together and connected to the fuselage cross section at the centerline through the use of guide curves to obtain the desired biomimetic shape. The fuselage shape was made iteratively through high-fidelity analysis to ensure it produces the lift required to promote the smooth, continuous bell curve. The aerodynamic shape is hollowed out for the creation of the structural components. The ribs are created at the semispan locations of the sketch blocks created from the .dat file splines and lofted to an adjacent intersection curve made from the interaction between the wing and a reference plane distanced the rib thickness from the airfoil sketch block. A lofted feature is chosen over an extrusion to adhere to the sweep and dihedral along the span. The fuselage ribs and bulkheads are made using the same technique. Lightning holes are laser cut in the Aeroply ribs and bulkheads to reduce the weight of the structural components while remaining to support shear flow. The fuselage internal structure harbors specific bays: camera system, avionics, propulsion, and storage. Each compartment is accessible through magnetic hatches composed from the fiberglass skin.



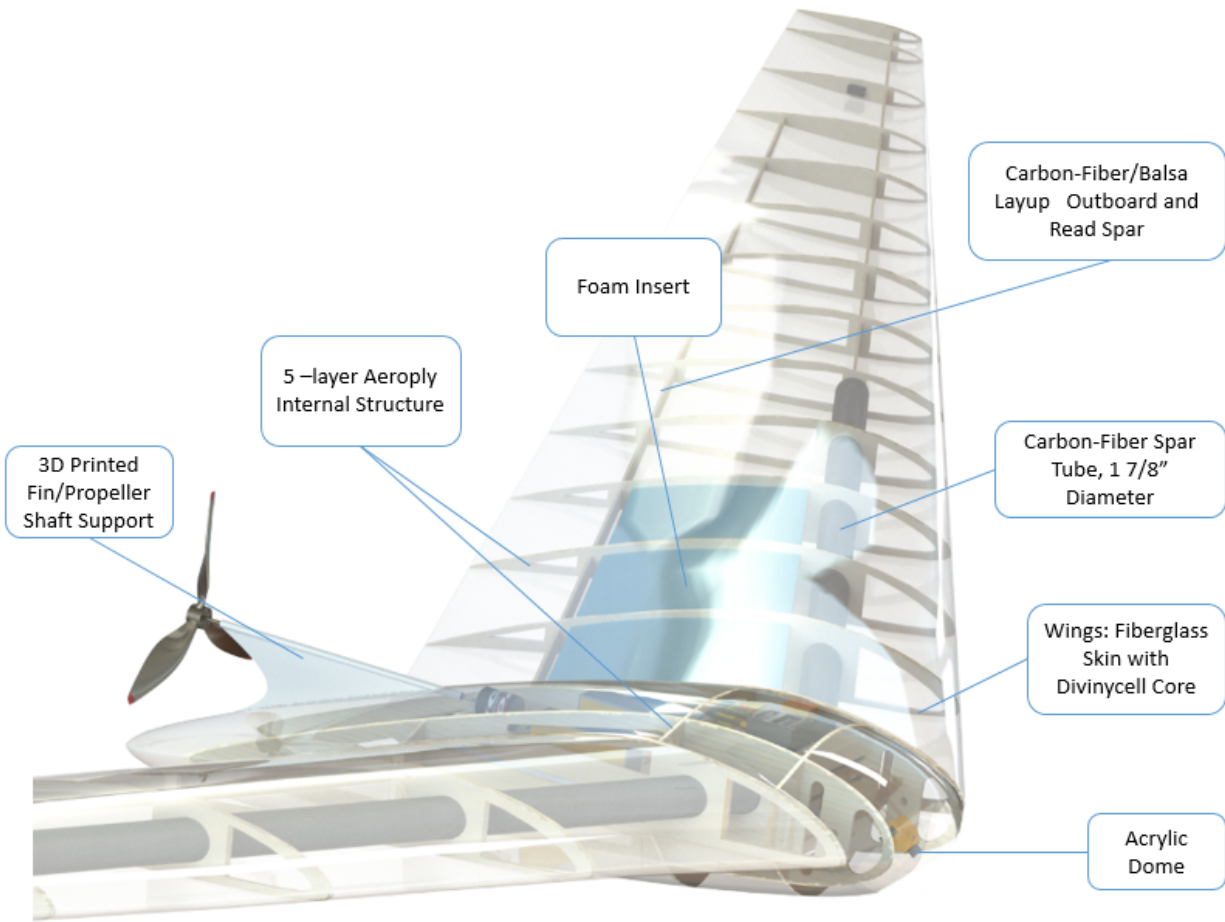


Figure 51: CAD Model of Bellwether Internals

The wing bending moment is supported by a 6' carbon fiber spar positioned at the quarter chord and a balsa wood spar positioned at the 3/4 chord. The balsa spar is adhered inside the wing by a carbon-fiber infused epoxy in the composite layup process. For additional support at the root, 3 blocks of dense foam are fit between the first 3 wing sections. The dense, supporting foam is cut using a computer numerical control (CNC) router which is the same technique used to shape the aircraft molds. The mold designs and images of CNC operation can be seen in Appendix 0.4. Due to the unique wing twist the aircraft molds require a non-linear parting line for the halves of the structure (Appendix 0.4). This results in a piecewise linear surface as the interface between top and bottom molds of an aircraft part. Each half of a wing mold is made as a solid, lofted body in CAD from which the wing shape is extracted with a lofted cut feature using the wing profiles. This aircraft fabrication process has begun, and its creation and experimental conclusions will ensue as future works.

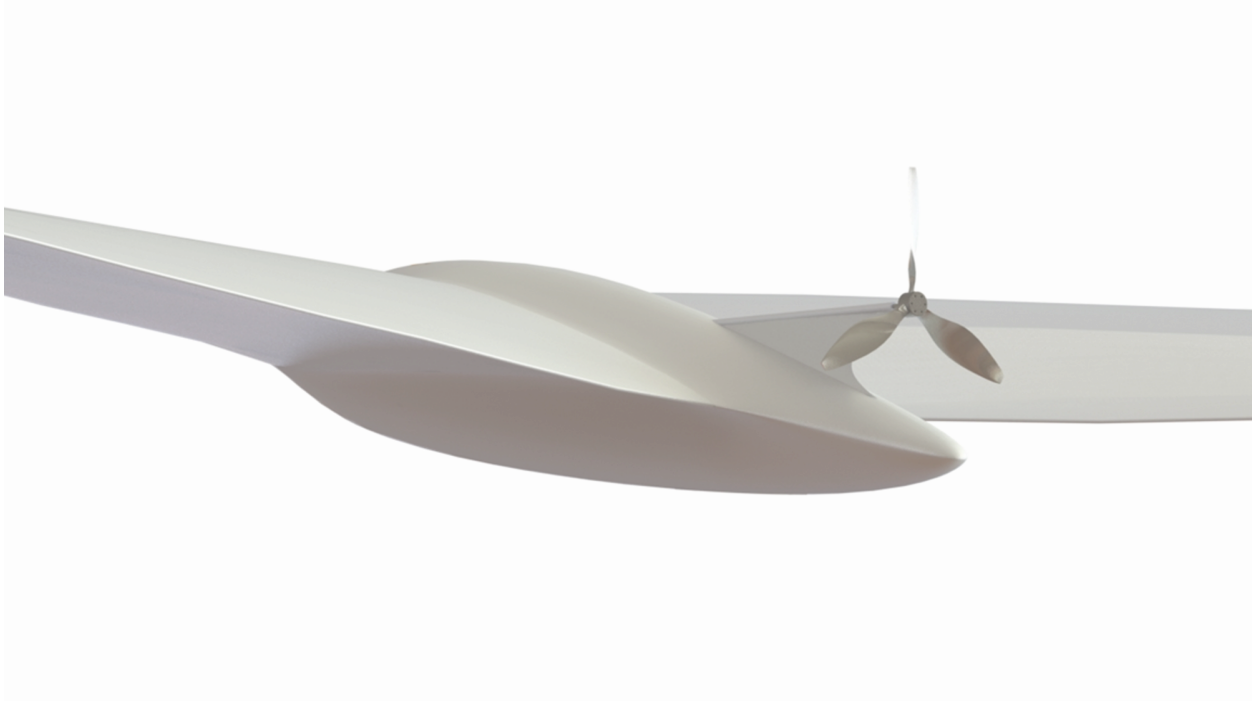


Figure 52: Bellwether - Rear View

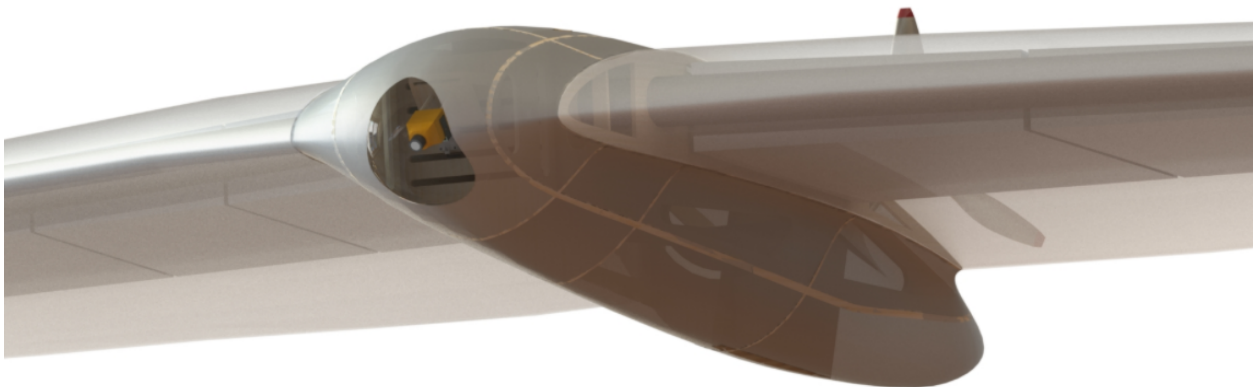


Figure 53: Bellwether - Front View

## CHAPTER VII

### Future Works

The ambitions for this research effort extend to seeking experimental validation through numerous methods. Some methods have been tried, but no conclusive data have been obtained to present in this document. Images of wind tunnel and laser-optic water tunnel testing can be found in Appendix 0.5. The pinnacle of comparing aircraft performance from computation to experimentation will come from the full-size fabrication of the 26 ft. wingspan aircraft. The roll-yaw coupling is to be parsed from the inertial measurement unit (IMU) to retrieve data that may be suitable for comparison to the aileron-deflected model in CFD. A pitot probe will provide airspeed to couple with the IMU to determine if the aircraft is flying at the expected cruise speed for the lift required (operating CL expected for level flight). In addition to the standard sensors used on unmanned aircraft systems, a network of sensors are to be integrated in the wing of the aircraft as a unique, theoretical method of determining the distribution of lift experimentally.

#### 7.1 Leading-Edge Flow Sensing

To revert to a common theme, we can see many examples of flow sensing found in the natural world such as with avian flight. Birds have neural networks of mechanoreceptors responsible for sensing dynamic, atmospheric conditions in flight. In the case of the peregrine falcon, vibration magnitude is the sensory stimulus used for sensing incidence angle when engaged in a rapid dive maneuver [4]. This research aims to mimic this streamlined, built-in sensing by gathering pressure data along the wing for the purpose of outputting incidence angle and obtaining sectional lift coefficient,  $C_l$ . Flush air data sensing (FADS) are an effective method of obtaining this information, but there are drawback to their implementation. Such booms require intrusive and rigid mechanical ties to the body of the aircraft. These external designs are susceptible to damage for a belly-landing aircraft such as Bellwether. Seeking an internal and well-integrated method of flow sensing leads to the pursuit of an alternative.

A fundamental aspect of this research is evaluating the effect of lift distribution to performance of a flying wing. Currently, experimental data is not present in confirming the computational results. An experimental flow sensing project has begun to validate the lift distribution from flight testing. 5-hole probes are used to measure stagnation points on the leading edge of the aircraft. This section discusses the theory, design, and integration of the flow-sensing system into a wing of Bellwether.

### 7.1.1 Theory

Discrete surface pressures measured at 5 ports on the leading edge of a wing are passed through an algorithm to calculate air-data parameters as detailed by researchers at North Carolina State University [19]. The real-time measurement of aerodynamic parameters in flight enhances aircraft capabilities while preventing dangerous stall situations. This is particularly useful for this flying wing which operates at a very high coefficient of lift and notices stall effects at a 7 angle of attack.

The measurement of the stagnation point ( $P_0$ ), which is the point of maximum pressure, can be used to calculate the freestream velocity ( $U$ ) and any incidence angle.

$$P_0 = P + \frac{1}{2}\rho U^2 \quad (7.1.1)$$

$$U = \sqrt{\frac{2(P_0 - P)}{\rho}} \quad (7.1.2)$$

A parabola is used for the estimation of the shape of the leading edge of the wing section. The pressure distribution measured from the 5 ports are used as inputs to an algorithm solving the exact potential-flow solution using a nonlinear regression. The angle of attack and lift coefficient are found using a look-up table which is found to be accurate for subsonic, pre-stall flow regimes [19].

The shape of an airfoil is given by the expression of the nose radius ( $r$ ) of the airfoil and the initial slope of the camber line ( $\lambda$ ) from the reduction of a high-order polynomial derivation (complete derivation and theory cited [19]) resulting in *Equation 7.1.3*.

$$y = \pm(2rx)^{\frac{1}{2}} + \lambda x \quad (7.1.3)$$

The algorithm is made with the ability to predict the lift coefficient and angle of attack on a wing section, and it can be extended to compute the aerodynamic parameters for sections along the wing. Due to the unique aerodynamic and geometric twist of this wing, and added redundancy, this design proceeds with multiple processing points on the semispan.

### 7.1.2 Design and Integration

This pressure-sensing experiment involves the integration of pressure sensors within the composite shell. 3D-printed sleeves harboring the pressure sensors are to be placed at specified locations along the span. Each sleeve is made by lofting two intersection curves. The intersection curves are made from the creation of planes parallel to the freestream and their interaction with the wing section. Extruded cuts are made so that the sleeve only occupies the leading edge and does not extend down the chord. The part is shelled to create internal volume for the pressure sensors. 5 extruded circular cuts are made at the leading edge in precise locations: the point on the leading edge of zero curvature, 2% chord location on top and bottom, and 5% chord location top and bottom. The holes punches are made orthogonal to the surface.

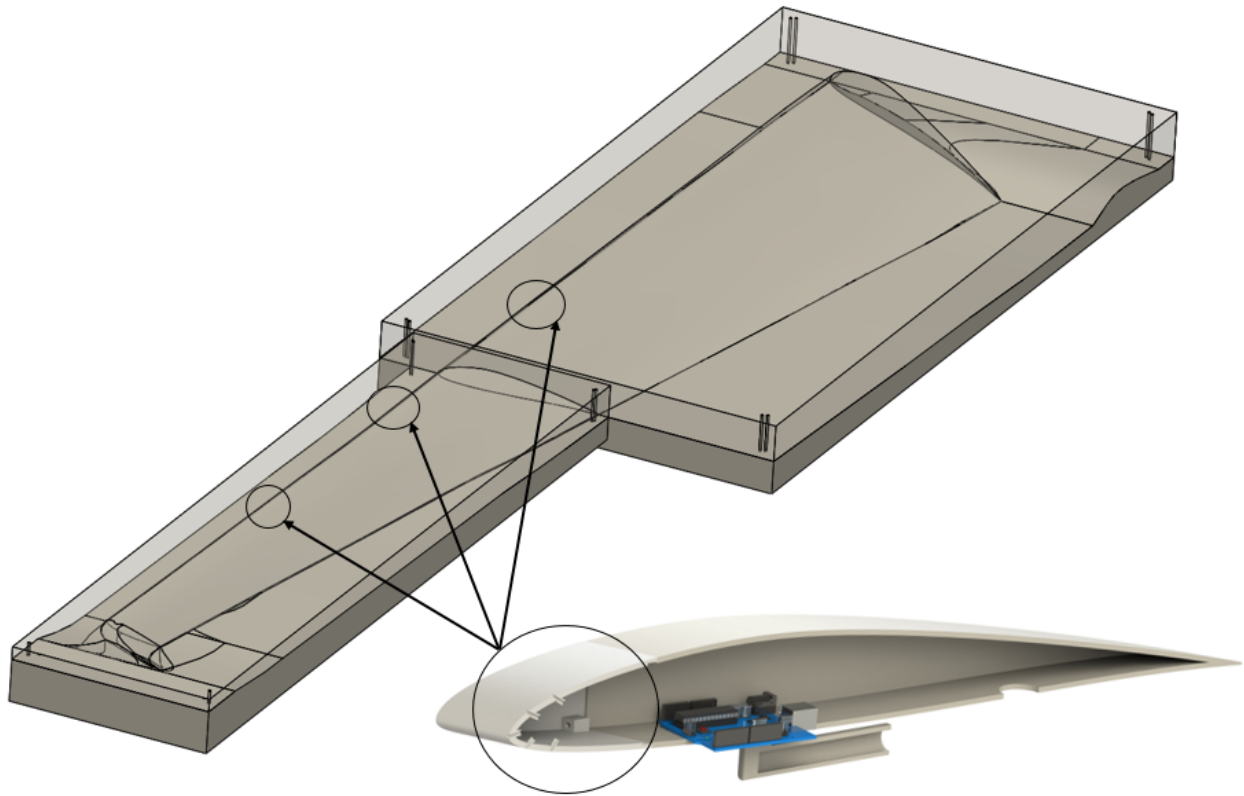


Figure 54: Port Wing Mold Assembly and the Integration of Leading-Edge Flow Sensors

3 locations are selected at the presumed area of highest gradient in the lift distribution to experimentally determine its adherence to the Prandtl bell curve. Each 3D-printed sleeve must then be designed separately to follow the sectional mold line of the wing. Due to symmetry, the leading-edge inserts will only be placed on the port wing (*Figure 54*). The starboard wing will harbor ballast mirroring the masses and locations of the sensing system.

The sleeves must be applied in such a way in order for the pressure holes to receive freestream air, but the design also favors internal integration. The composite layup process will accommodate this design by cutting out the skin and core material at the leading edge locations for the sensing sleeves prior to the application of epoxy. After the creation of the composite within the mold, the sleeves may be epoxied and the Arduino wired internally to create a neural sensing network within the wing that runs to the power source in the avionics bay.

### 7.1.3 Wind Tunnel Demonstration System

A leading-edge sleeve designed for the 7.5 ft. semispan location is printed out of ABS plastic (*Figure 55*) to used as a demonstration system for wind-tunnel testing. The model adds a sting attachment for the model to be place on the force balance in the wind tunnel. Thin extrusion features are made on the inside of the sleeve to connect rubber tubes from the holes

on the leading edge to their respective pressure sensors. The pressure sensors are wired to an Arduino computer board responsible for applying the algorithm and storing the in-flight data to an SD card. The first sleeve (*Figure 55*) has been tested for basic readings of pressure data containing only 1 pressure sensor for the initial test. This model is representative of what will be included in the wing, but a second model (like the sleeve shown in *Figure 54*) is made to more accurately simulate the in-flight reading since the model will take the shape of the entire wing section housing the sensing mechanism.

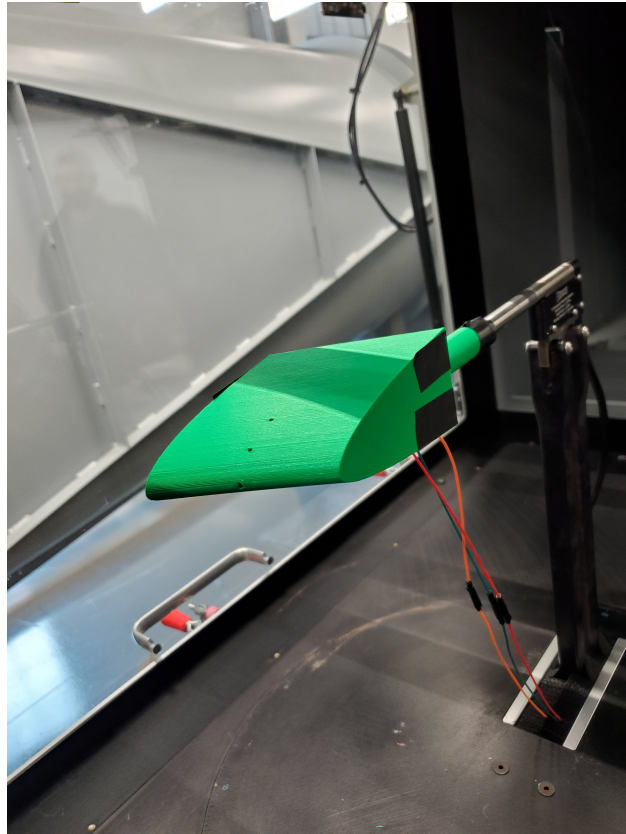


Figure 55: Sensing Sleeve Mounted in the Endeavor Wind Tunnel

The algorithm has yet to be applied for the determination of aerodynamic properties.

## CHAPTER VIII

### Conclusion

This thesis set out to design an unmanned aircraft using a bell-shaped lift distribution and investigate, through highly-computational methods, the centralization of lift and the aerodynamic effect on the change of shape of the flow field. Vortex Lattice Methods are performed to provide the aircraft designer with confidence in the use of rapid tools for aerodynamic calculation. VLM is compared to CFD to illustrate its viability for initial-stage calculations. The author suggests the use of the lower-order methods for highly-iterative design processes of subsonic lifting bodies.

The primary theory tested is the centralization of lift and the effect on aerodynamic performance with the change in the flow field. Four separate wing models have been made, each designed with a different distribution of lift by the command of aerodynamic and geometric twist. Observation of the vortical structures and core locations in the flow field provide an intuition for the relationship of the centralization of lift to roll-yaw coupling. Further parameterizations would provide greater clarity to this relationship:

1. More aileron deflection angles to be tested
2. Changes in aileron area and inboard reach with relation to the vortex core location
3. Alpha sweep for ELD to compare to BSLD at common CL
4. Swap wingspan constraint for bending moment constraint and compare BSLD to ELD

It is stated that the change in shape of the lift distribution from elliptical to bell decreases the drag by about 11% with a 22.5% increase in span for the same structural weight [2]. As discussed in the early sections, aspect ratio is inversely proportional to induced drag. However, this is not to say that the BSLD is more efficient than ELD in level flight. Greater parasite drag is expected from the increase in span, and since  $C_D = C_{D0} + C_{Di}$  the overall drag could remain the same (or even increase) dependent upon the ratio of increased parasite drag to the induced drag reduction. No evidence in this study, or in the referenced research, is provided to confidently suggest that any aircraft configuration will benefit in level flight from deviating from the ELD.

The elliptical wing in this study operates at a much lower coefficient of lift (0.323) than the other distributions. Therefore, it cannot be accurately compared in terms of aerodynamic performance since it did not meet the required lift constraint. The CL can be matched through an alpha sweep analysis of the wing and extracting the performance metrics at the CL corresponding to a particular point on the lift-curve slope. This may fix the issue of the lift constraint, but adding geometric rotation to the wing to alter CL introduces

more parasite drag. A better way to do this comparison could be to set up a Trefftz plane in STAR-CCM+ as a grid of planar probes from which the velocity vectors recorded are used to calculate induced drag from the flow field. Though this may be the best method of comparing induced drag computationally, no clear process using software tools has been identified by the author to provide evidentiary results in favor of the BSLD to ELD in level flight.

Distributions of centralized lift have a clear advantage in performing a banked turn without need for an auxillary yaw device. The change in the flow field for the creation on inboard upwash allow for the presence of proverse yaw for anti-symmetric aileron deflection. The inboard travel of the vortex cores is proportional to proverse yaw control power. When the aileron is engulfed in upwash, induced thrust will occur so long as the deflected aileron does not produce enough downwash to counteract the upwash from the inboard vortex core. The behavior of favorable yaw into a banked turn and the location and size of the vortex core are used as evidence of its application to bird flight. From coordinated turns to the variable wing spacing in avian flight formation, it is suggested that birds use the much wider inboard vortex created by the BSLD to set their wing overlap to enhance aerodynamic efficiency in the wake of the leading bird.

It could potentially be reasoned that the incorporation of this spanload can reduce the tail size of more traditional aircraft configurations. The application of the BSLD may prove to be beneficial to standard aircraft if the reduction in structural weight is of greater importance than any losses occurring in level flight, though true comparisons of the ELD to BSLD in level flight have not been performed. This body of work is not conclusive in the advocacy for the application of a bell spanload for optimal aerodynamic efficiency. However, these findings are congruent with the suggestions of emerging research positing that this spanload is more apt for modeling avian flight.

## 8.1 Summary

1. VLM overestimates the lift coefficient compared to CFD for a swept wing designed by the BSLD, but the distribution across the span closely matches the high-fidelity lift distribution.
2. CFD results suggest BSLD as most fit for performing coordinating turns from the examination of roll-yaw coupling behavior.
3. CFD shows favorable L/D for BSLD in level flight for a fixed wingspan constraint compared to the other distributions operating at the same CL (cosine and modern bell curves).
4. Vorticity scenes produced in STAR-CCM+ agree with Ludwig Prandtl's revised work and NASA's Prandtl-D project locating the vortex core produced by the BSLD to occur at the 70.4-70.5% semispan location.
5. Congruent with emerging research, most notably the work from Kyle Lukacovic [14] in the wake of the Prandtl-D [2], the activation of the induced thrust at the wingtips and shape of the flow field suggest avian flight is best modeled by the bell spanload.



## REFERENCES

- [1] John D. Anderson, *Fundamentals of aerodynamics*, sixth ed., McGraw-Hill Education, New York, 2017.
- [2] Albion H. Bowers, Oscar J. Murillo, Robert Jensen, Brian Eslinger, and Christian Gelzer, *On wings of the minimum induced drag: Spanload implications for aircraft and birds*, NASA scientific and technical information (STI) program (March 2016).
- [3] Steven A. Brandt, *Introduction to aeronautics*, third ed., American Institute of Astronautics and Aeronautics, Inc., Reston, Virginia, 2015.
- [4] C. Brücker, D. Schlegel, and M. Triep, *Feather vibration as a stimulus for sensing incipient separation in falcon diving flight*, *Natural Resources* **7**, 411-422. (2016).
- [5] Martin Carrizalez, Estela Bragado, and Mudassir Lone, *Non-elliptic lift distribution wings to decrease vertical tailplane size in commercial aircraft*, American Institute of Astronautics and Aeronautics (2019).
- [6] Mark Drela, *Xfoil: An analysis and design system for low reynolds number airfoils*, MIT Dept. of Aeronautics and Astronautics (1986).
- [7] Portugal SJ et al, *Upwash exploitation and downwash avoidance by flap phasing in ibis formation flight*, **DOI: 10.1038/nature12939**. (2014).
- [8] A. Frediani and G. Montanari, *Prandtl's best wing system (eds) advanced design problems in aerospace engineering*, Plenum Press (New York).
- [9] Aldo Frediani, Matteo Gasperini, Guido Saporito, and Andrea Rimondi, *Development of a prandtlplane aircraft configuration*, Department of Aerospace Engineering.
- [10] Spedding GR, *The wake of a kestrel (falco tinnunculus) in gliding flight*, *J. Exp. Biol.* **127** (1987).
- [11] D. F. Hunsaker and W. F. Phillips, *Ludwig prandtl's 1933 paper concerning wings for minimum induced drag, translation and commentary*, American Institute of Astronautics and Aeronautics **AIAA SciTech 2020 Forum** (January 6-10, 2020).
- [12] D. F. Hunsaker, W. F. Phillips, and J. J. Joo, *Designing wings with fixed twist for minimum induced drag*, American Institute of Astronautics and Aeronautics (January 2017).

- [13] Robert T. Jones, *The spanwise distribution of lift for minimum induced drag of wings having a given lift and a given bending moment*, Ames Aeronautical Laborator (December 1950.).
- [14] Kyle Lukacovic, *A parametric study of formation flight of a wing based on prandtl's bell-shaped lift distribution*, Master's Thesis **California Polytechnic State University** (2020).
- [15] Karl Nickel and Michael Wohlfahrt, *Tailless aircraft in theory and practice*, American Institute of Aeronautics and Astronautics, Portland, OR, September 1, 1994.
- [16] Ludwig Prandtl, *Applications of modern hydrodynamics to aeronautics*, NACA Report **No 116** (1921).
- [17] Ludwig Prandtl, *Induced drag on multiplanes*, NACA TN 182 (1924).
- [18] Jonathhan Richter, Kevin Hainline, and Ramesh K. Agarwal, *Examination of proverse yaw in bell-shaped spanload aircraft*, Mechanical Engineering and Material Science Independent Study (2019).
- [19] Aditya Saini and Ashok Gopalathnam, *Leading-edge flow sensing for aerodynamic parameter estimation*, AIAA Journal **Vol. 56, No. 12** (December 2018).
- [20] W. W. Schneider, *The importance of aerodynamics in the development of commercially successful transport aircraft*, Aerodynamic Drag Reduction Technology. Proceeding of CEAS/DRAGNET European Drag Reduction Conference (June 19-21, 2000).
- [21] M. S. Selig, *Uiuc airfoil data site, department of aerospace engineering.*, (January 2007).
- [22] R. Srinath and D. S. Sahana, *Aerodynamic analysis of forward swept wing using prandtl-d wing concept*, Mangalore Institute of Technology and Engineering.
- [23] B. L. Stevens and F. L. Lewis, *Aircraft control and simulation*, John Wiley Sons, Hoboken, NJ, 1992.

listings verbatim

## APPENDICES

### 0.1 Aircraft CAD Drawings and Renderings

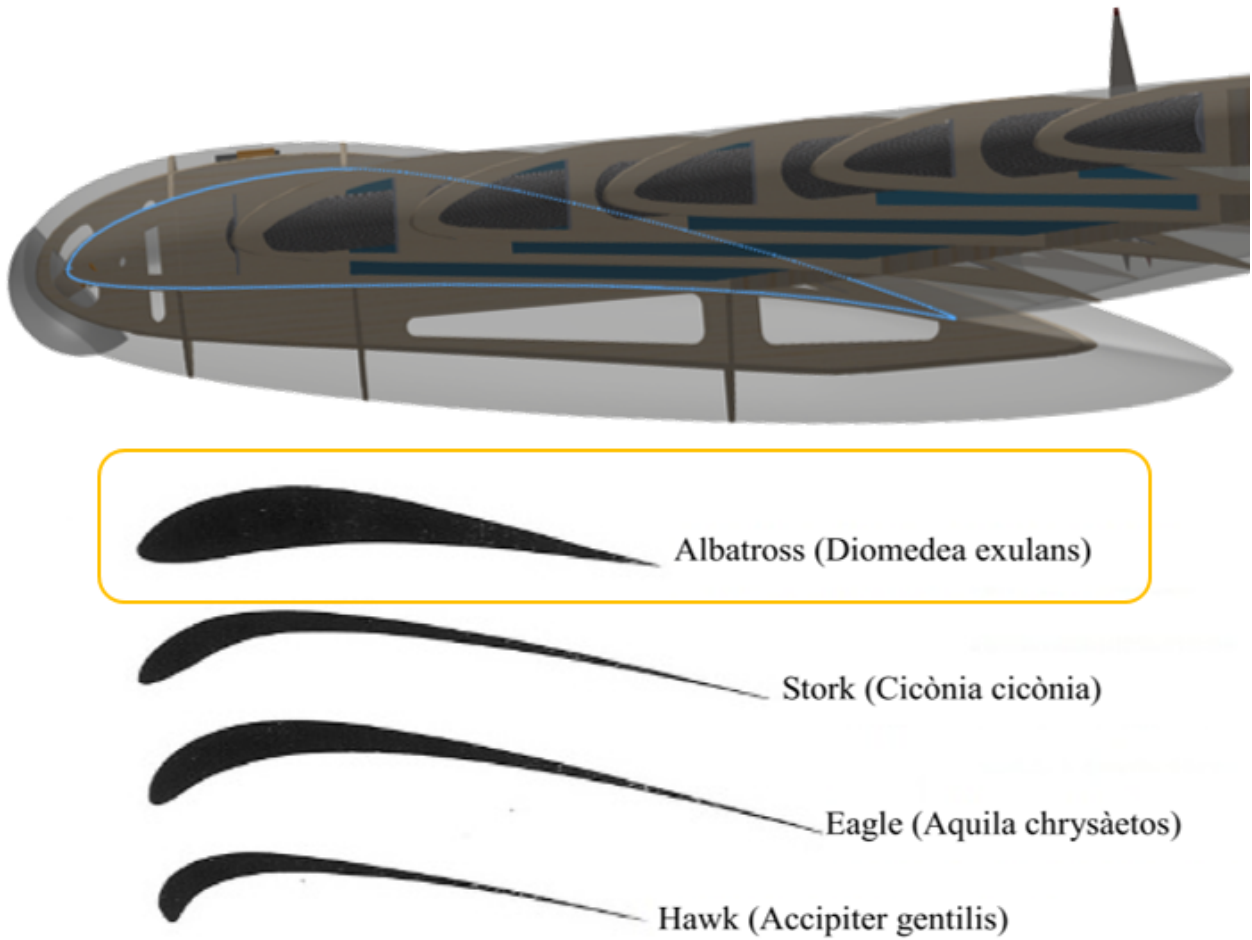


Figure 56: Bellwether Side View - Avian Inspiration

2

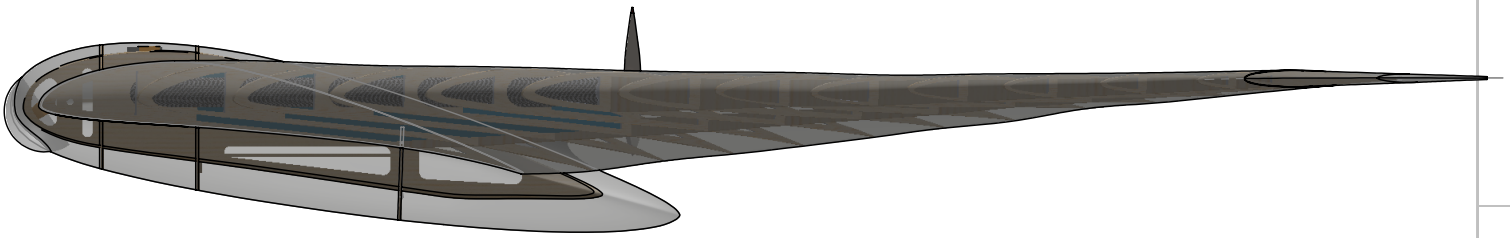
1

# *Bellwether*

## Avian-Inspired Flying Wing

B

B



A

A

**PROPRIETARY AND CONFIDENTIAL**  
 THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF <INSERT COMPANY NAME HERE>. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF <INSERT COMPANY NAME HERE> IS PROHIBITED.

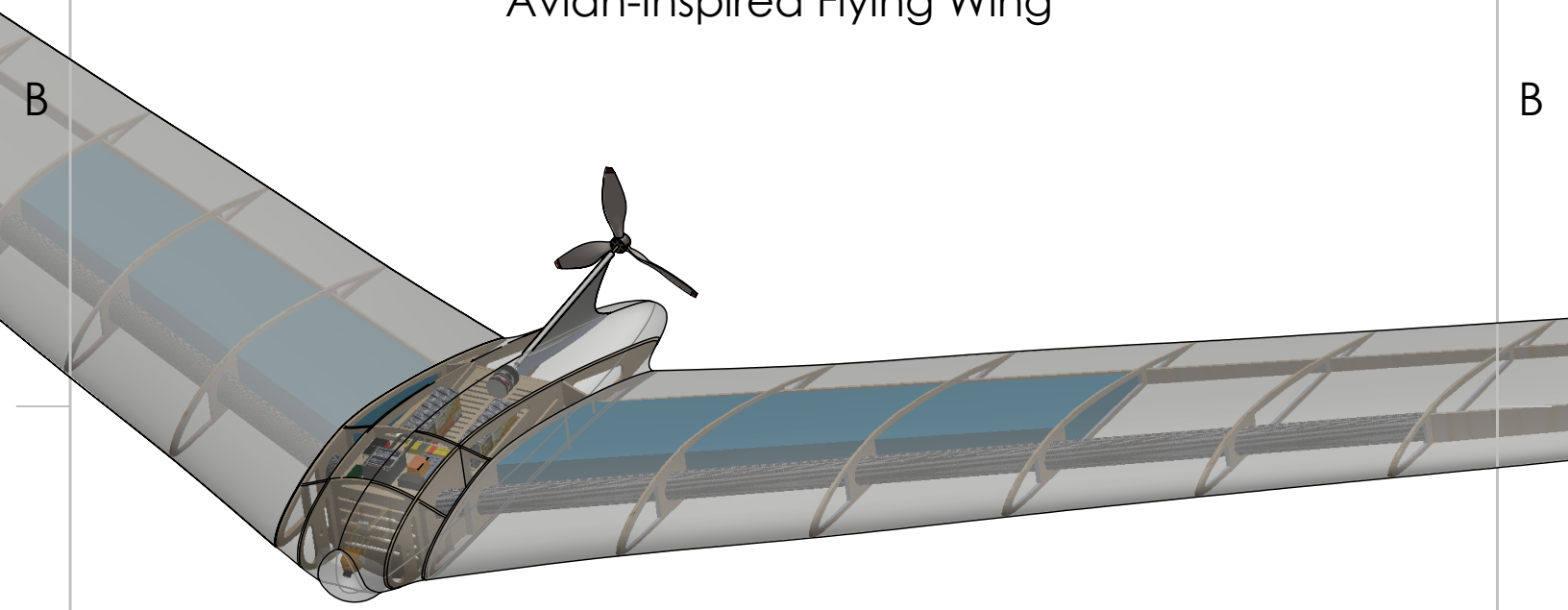
		UNLESS OTHERWISE SPECIFIED:		NAME	DATE	<p style="font-size: 1.2em; margin: 0;">Scott Weekley</p>							
		DIMENSIONS ARE IN INCHES				<p style="margin: 0;">TITLE:</p> <p style="font-size: 1.5em; margin: 0;">Bellwether</p>							
		TOLERANCES:											
		FRACTIONAL ±											
		ANGULAR: MACH ± BEND ±											
		TWO PLACE DECIMAL ±				<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 10%;">SIZE</td> <td style="width: 50%;">DWG. NO.</td> <td style="width: 40%;">REV</td> </tr> <tr> <td style="font-size: 1.5em; text-align: center;"><b>A</b></td> <td style="text-align: center; font-size: 1.2em;">Avian Section</td> <td></td> </tr> </table>		SIZE	DWG. NO.	REV	<b>A</b>	Avian Section	
SIZE	DWG. NO.	REV											
<b>A</b>	Avian Section												
		THREE PLACE DECIMAL ±				<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%;">SCALE: 1:96</td> <td style="width: 50%;">WEIGHT:</td> </tr> <tr> <td colspan="2" style="text-align: right;">SHEET 1 OF 1</td> </tr> </table>		SCALE: 1:96	WEIGHT:	SHEET 1 OF 1			
SCALE: 1:96	WEIGHT:												
SHEET 1 OF 1													
		INTERPRET GEOMETRIC TOLERANCING PER:											
		MATERIAL											
		FINISH											
	NEXT ASSY	USED ON											
	APPLICATION												
						DO NOT SCALE DRAWING							

2

1

# *Bellwether*

## Avian-Inspired Flying Wing



B

B

A

A

**PROPRIETARY AND CONFIDENTIAL**  
 THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF <INSERT COMPANY NAME HERE>. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF <INSERT COMPANY NAME HERE> IS PROHIBITED.

		UNLESS OTHERWISE SPECIFIED:		NAME	DATE	<p style="font-size: 1.2em; margin: 0;">Scott Weekley</p> <p style="font-size: 1.5em; margin: 0;">TITLE: Bellwether</p>																									
		DIMENSIONS ARE IN INCHES				<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 10%;">DRAWN</td> <td style="width: 10%;"></td> <td style="width: 10%;"></td> <td style="width: 10%;"></td> </tr> <tr> <td>CHECKED</td> <td></td> <td></td> <td></td> </tr> <tr> <td>ENG APPR.</td> <td></td> <td></td> <td></td> </tr> <tr> <td>MFG APPR.</td> <td></td> <td></td> <td></td> </tr> <tr> <td colspan="4">Q.A.</td> </tr> <tr> <td colspan="4">COMMENTS:</td> </tr> </table>		DRAWN				CHECKED				ENG APPR.				MFG APPR.				Q.A.				COMMENTS:			
DRAWN																															
CHECKED																															
ENG APPR.																															
MFG APPR.																															
Q.A.																															
COMMENTS:																															
		TOLERANCES:																													
		FRACTIONAL ±																													
		ANGULAR: MACH ± BEND ±																													
		TWO PLACE DECIMAL ±																													
		THREE PLACE DECIMAL ±																													
		INTERPRET GEOMETRIC TOLERANCING PER:																													
		MATERIAL																													
		FINISH																													
NEXT ASSY	USED ON					SIZE	DWG. NO.	REV																							
						<b>A</b>	Avian Down																								
APPLICATION		DO NOT SCALE DRAWING				SCALE: 1:96 WEIGHT:		SHEET 1 OF 1																							

2

1

2

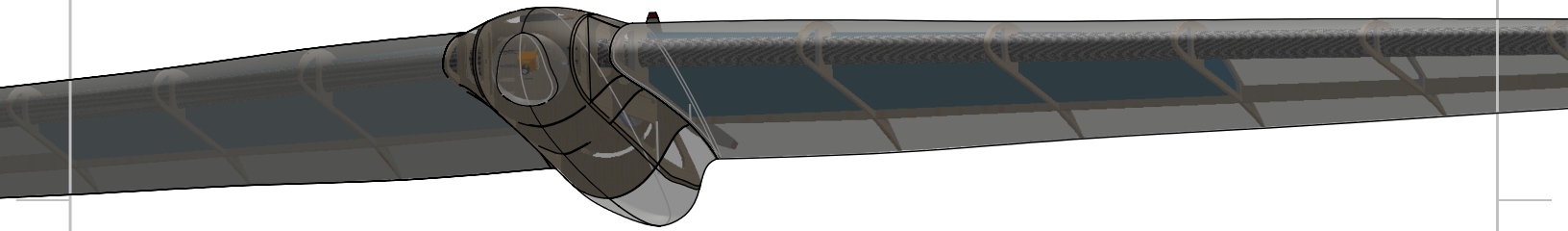
1

# *Bellwether*

## Avian-Inspired Flying Wing

B

B



A

A

**PROPRIETARY AND CONFIDENTIAL**  
 THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF <INSERT COMPANY NAME HERE>. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF <INSERT COMPANY NAME HERE> IS PROHIBITED.

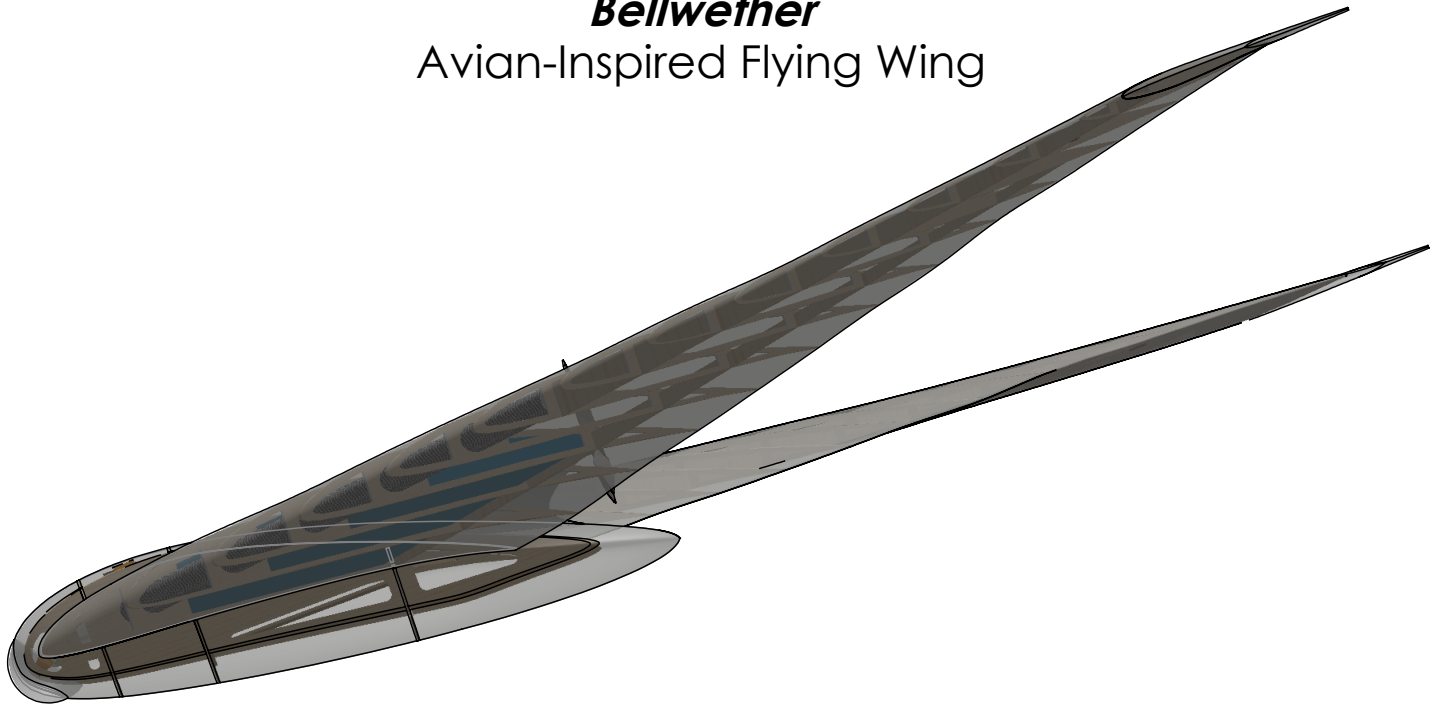
		UNLESS OTHERWISE SPECIFIED:		NAME	DATE	Scott Weekley TITLE: <h2 style="margin: 0;">Bellwether</h2>	
		DIMENSIONS ARE IN INCHES	DRAWN			SIZE <b>A</b> DWG. NO. Avian Front REV SCALE: 1:96 WEIGHT: SHEET 1 OF 1	
		TOLERANCES:	CHECKED				
		FRACTIONAL ±	ENG APPR.				
		ANGULAR: MACH ± BEND ±	MFG APPR.				
		TWO PLACE DECIMAL ±	Q.A.				
		THREE PLACE DECIMAL ±	COMMENTS:				
		INTERPRET GEOMETRIC TOLERANCING PER:					
		MATERIAL					
		FINISH					
NEXT ASSY	USED ON						
APPLICATION		DO NOT SCALE DRAWING					

1

2

1

# Bellwether Avian-Inspired Flying Wing



B

B

A

A

**PROPRIETARY AND CONFIDENTIAL**  
THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF <INSERT COMPANY NAME HERE>. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF <INSERT COMPANY NAME HERE> IS PROHIBITED.

		UNLESS OTHERWISE SPECIFIED:		NAME	DATE
		DIMENSIONS ARE IN INCHES	DRAWN		
		TOLERANCES:	CHECKED		
		FRACTIONAL ±	ENG APPR.		
		ANGULAR: MACH ± BEND ±	MFG APPR.		
		TWO PLACE DECIMAL ±	Q.A.		
		THREE PLACE DECIMAL ±	COMMENTS:		
		INTERPRET GEOMETRIC TOLERANCING PER:			
		MATERIAL			
		FINISH			
NEXT ASSY	USED ON				
APPLICATION		DO NOT SCALE DRAWING			

Scott Weekley		
TITLE:		
Bellwether		
SIZE	DWG. NO.	REV
<b>A</b>	Avian Swoop	
SCALE: 1:96	WEIGHT:	SHEET 1 OF 1

2

1

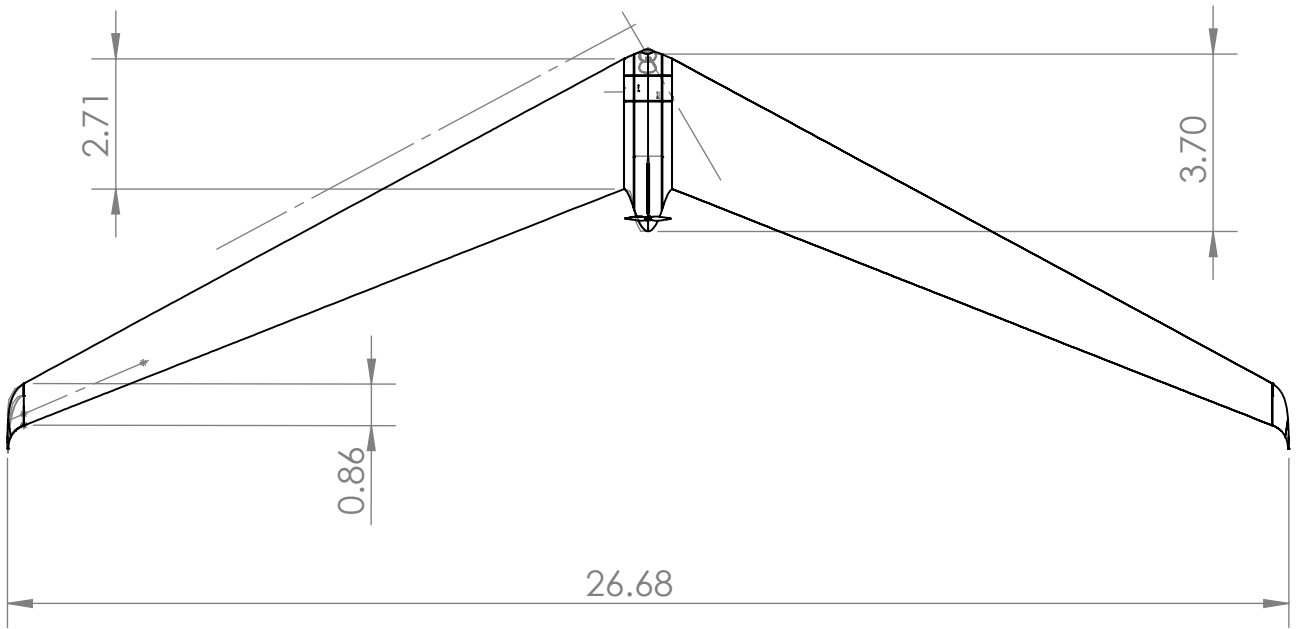
2

1

# Bellwether Planform Measurements

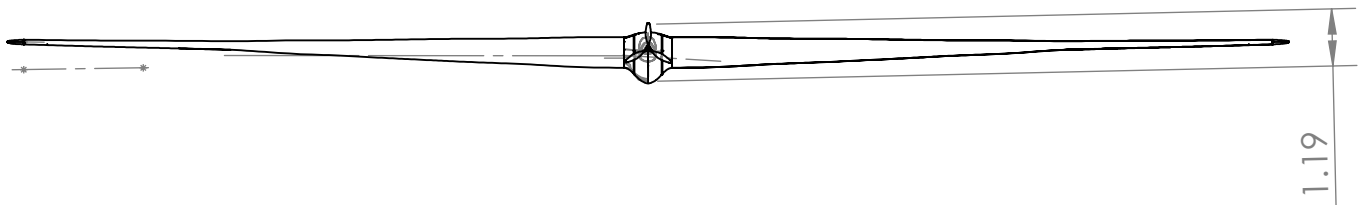
B

B



A

A



**PROPRIETARY AND CONFIDENTIAL**  
 THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF <COMPANY NAME>. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF <COMPANY NAME> IS PROHIBITED.

		DIMENSIONS ARE IN INCHES TOLERANCES: FRACTIONAL ± ANGULAR: MACH ± BEND ± TWO PLACE DECIMAL ± THREE PLACE DECIMAL ±		NAME	DATE
				DRAWN	
				CHECKED	
				ENG APPR.	
				MFG APPR.	
				Q.A.	
				COMMENTS:	
NEXT ASSY	USED ON	MATERIAL			
		FINISH			
APPLICATION		DO NOT SCALE DRAWING			

Scott Weekley

Bellwether

SIZE	DWG. NO	REV.
<b>A</b>	Bellwether_Planform	
SCALE:1:96	WEIGHT:	SHEET 1 OF 1

2

1



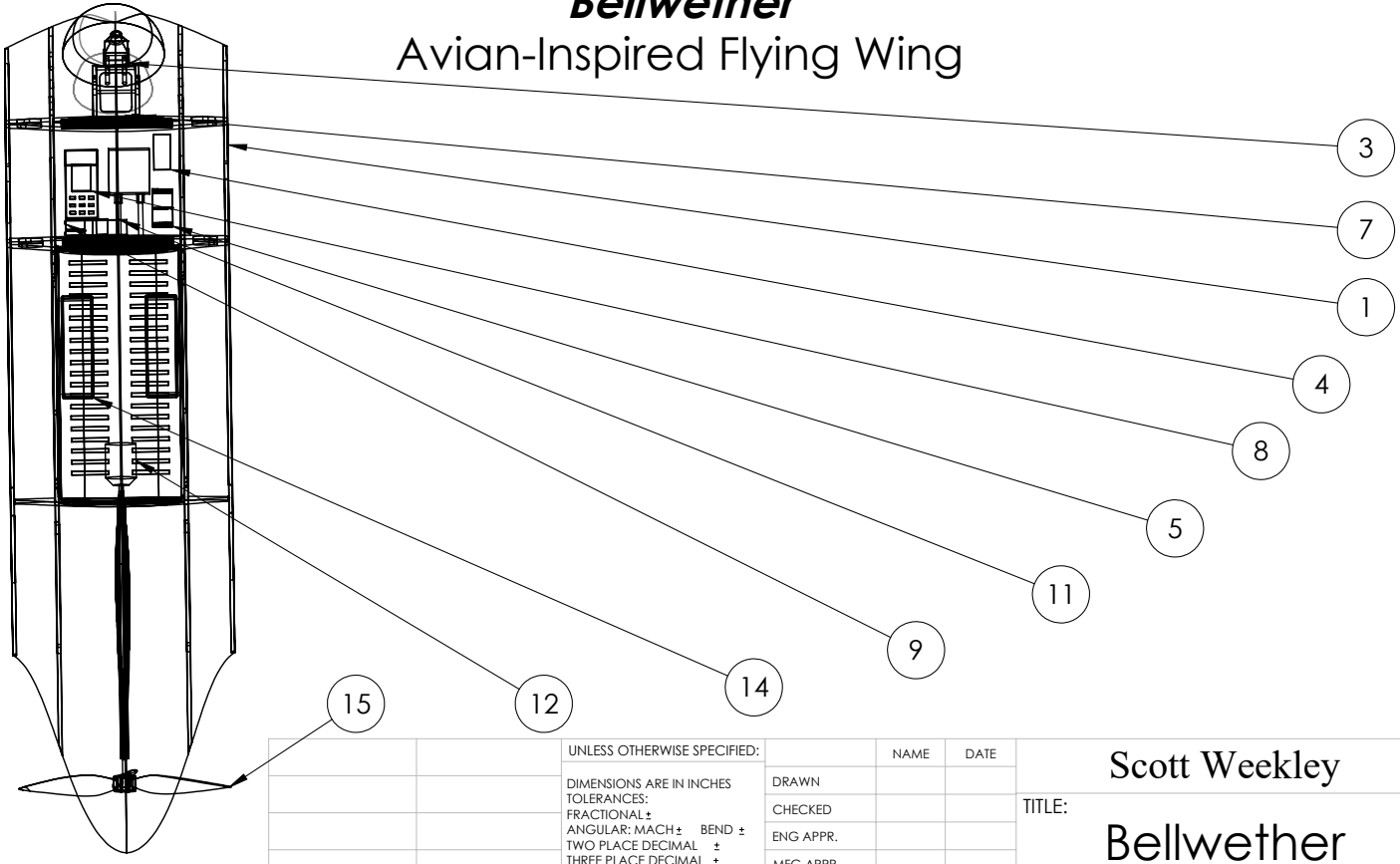
2

1

# Bellwether Avian-Inspired Flying Wing

B

B



A

A

**PROPRIETARY AND CONFIDENTIAL**  
 THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF <INSERT COMPANY NAME HERE>. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF <INSERT COMPANY NAME HERE> IS PROHIBITED.

		UNLESS OTHERWISE SPECIFIED:		NAME	DATE
		DIMENSIONS ARE IN INCHES	DRAWN		
		TOLERANCES:	CHECKED		
		FRACTIONAL: ±	ENG APPR.		
		ANGULAR: MACH ± BEND ±	MFG APPR.		
		TWO PLACE DECIMAL ±	Q.A.		
		THREE PLACE DECIMAL ±	COMMENTS:		
		INTERPRET GEOMETRIC TOLERANCING PER:			
		MATERIAL			
		FINISH			
NEXT ASSY	USED ON				
APPLICATION		DO NOT SCALE DRAWING			

Scott Weekley		
TITLE:		
Bellwether Fuselage Components		
SIZE	DWG. NO.	REV
<b>A</b>		
SCALE: 1:96 WEIGHT:		SHEET 1 OF 2

2

1

2

1

ITEM NO.	PART NUMBER	DESCRIPTION	QTY.
1	BellwetherV4.5 Fuselage w bulkheads		1
2	Bellwether V9 Solid Wings w Ribs LH		1
3	Camera_5.8GHz_RunC am_Swift2		1
4	Electric Motor Speed Controller (ESC)		1
5	FREEZE_Aircraft Controls Reciever Jeti		1
6	FREEZE_Servo_2.16		2
7	FREEZE_Video Transmitter		1
8	OrangeCube_Autopilo t		1
9	Power Sensor		1
10	TelemetryDiskAntenna		1
11	TelemetryRadio_RFD90 0x		1
12	Turnigy G32 770kV		1
13	5.25ft carbon fiber spar no mass		2
14	LiPo_4500mAh_22.2V		2
15	Propeller		1

B

B

A

A

<p><b>PROPRIETARY AND CONFIDENTIAL</b> THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF &lt;INSERT COMPANY NAME HERE&gt;. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF &lt;INSERT COMPANY NAME HERE&gt; IS PROHIBITED.</p>			UNLESS OTHERWISE SPECIFIED:		NAME	DATE	<p>Scott Weekley</p> <p>TITLE: <b>Bill of Materials</b></p>	
			DIMENSIONS ARE IN INCHES TOLERANCES: FRACTIONAL ± ANGULAR: MACH ± BEND ± TWO PLACE DECIMAL ± THREE PLACE DECIMAL ±	DRAWN				
			INTERPRET GEOMETRIC TOLERANCING PER:	CHECKED				
			MATERIAL	ENG APPR.				
			FINISH	MFG APPR.				
	NEXT ASSY	USED ON	COMMENTS:				Q.A.	
	APPLICATION		DO NOT SCALE DRAWING				REV	
							SIZE: <b>A</b>	
							DWG. NO.	
							SCALE: 1:96 WEIGHT:	
							SHEET 2 OF 2	

2

1

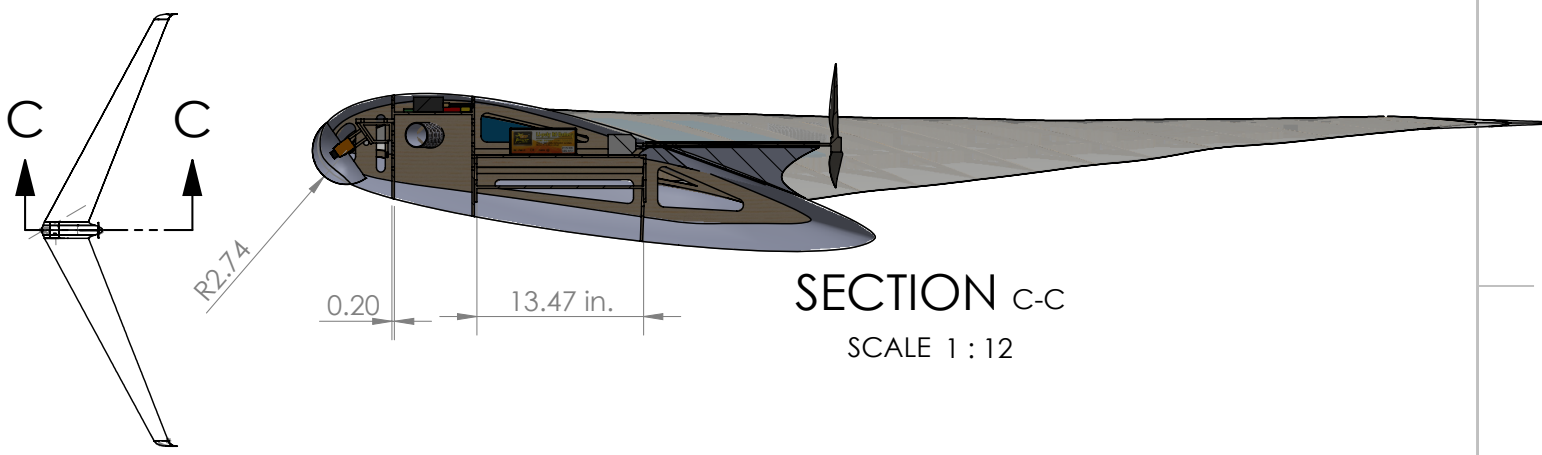
2

1

# Bellwether Avian-Inspired Flying Wing

B

B



SECTION C-C  
SCALE 1 : 12

A

A

**PROPRIETARY AND CONFIDENTIAL**  
 THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF <INSERT COMPANY NAME HERE>. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF <INSERT COMPANY NAME HERE> IS PROHIBITED.

		UNLESS OTHERWISE SPECIFIED:		NAME	DATE	Scott Weekley	
		DIMENSIONS ARE IN INCHES	DRAWN			TITLE:	
		TOLERANCES:	CHECKED			Bellwether	
		FRACTIONAL ±	ENG APPR.			SIZE	DWG. NO.
		ANGULAR: MACH ± BEND ±	MFG APPR.			<b>A</b>	Center Cut
		TWO PLACE DECIMAL ±	Q.A.			SCALE: 1:96	WEIGHT:
		THREE PLACE DECIMAL ±	COMMENTS:				SHEET 1 OF 1
		INTERPRET GEOMETRIC TOLERANCING PER:					
		MATERIAL					
		FINISH					
	NEXT ASSY	USED ON					
	APPLICATION	DO NOT SCALE DRAWING					

2

1

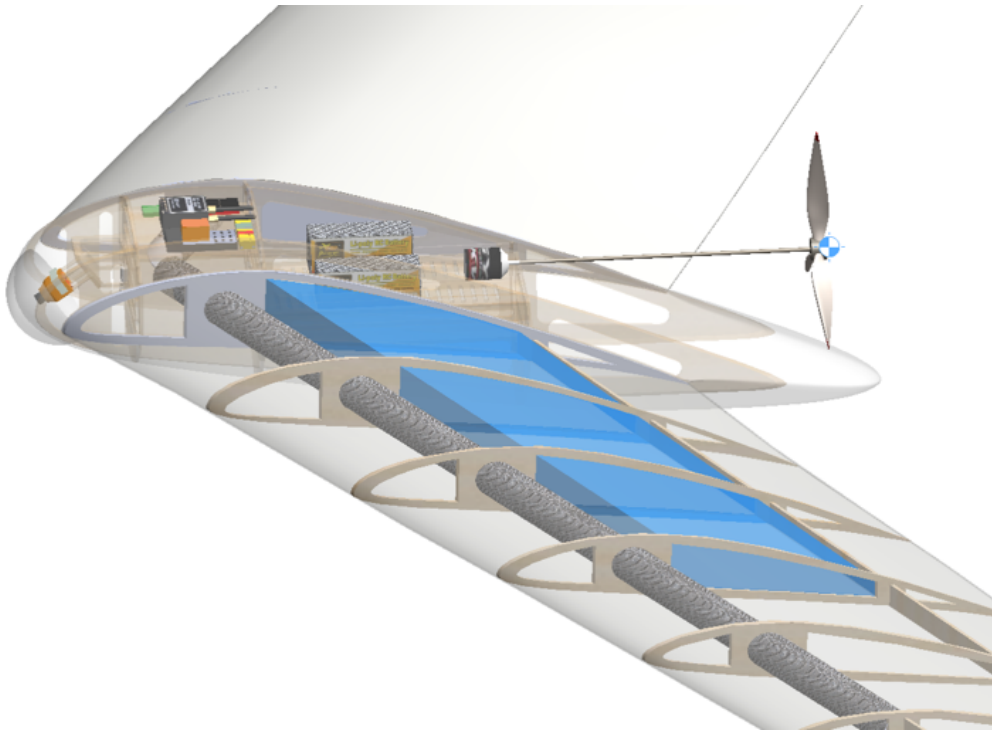


Figure 57: Propulsor Placement in Line with Center of Gravity

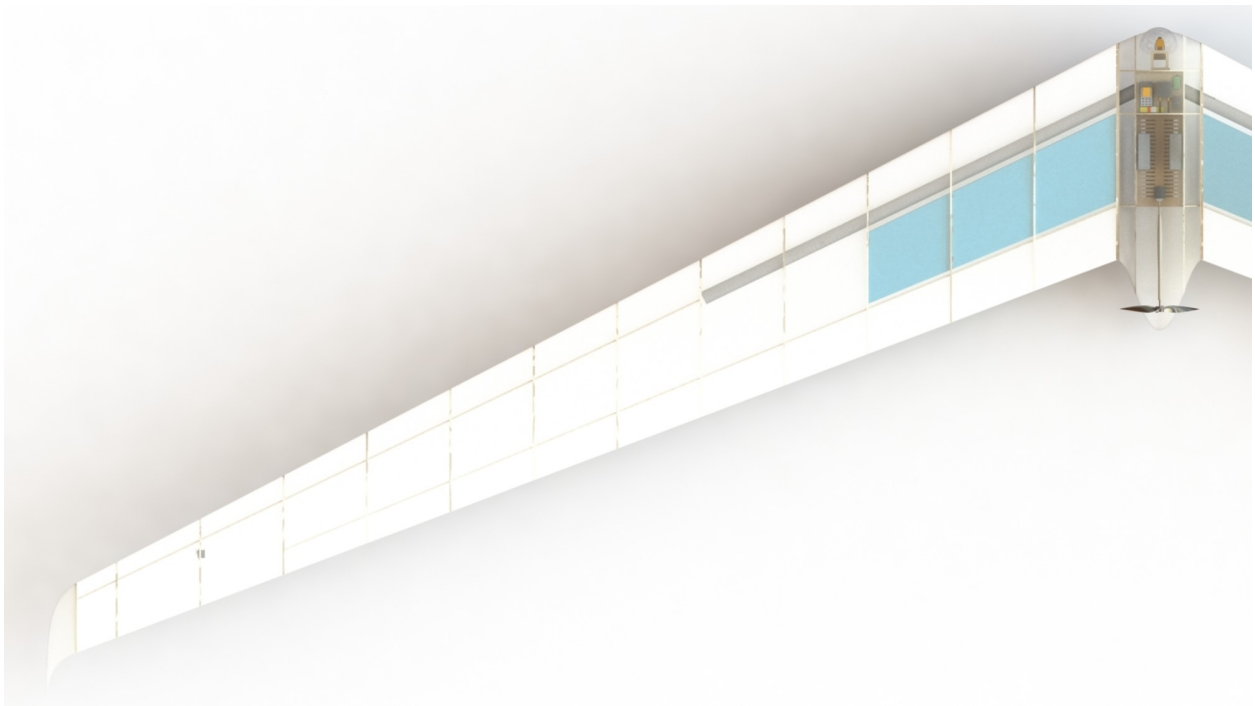


Figure 58: Bellwether Planform View

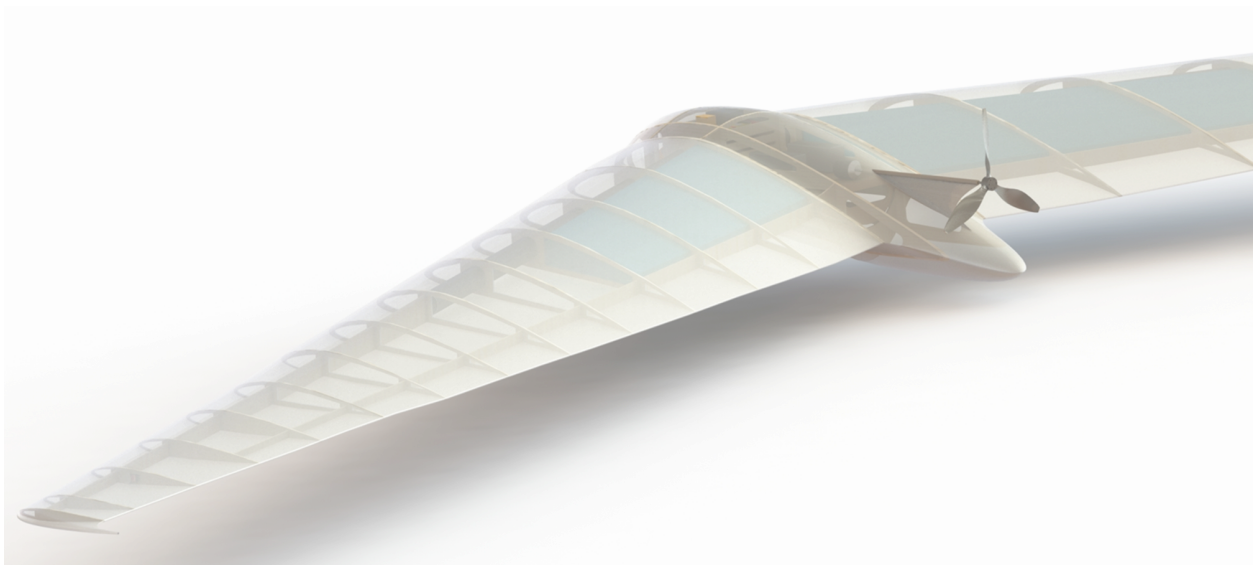


Figure 59: Bellwether1



Figure 60: Bellwether2

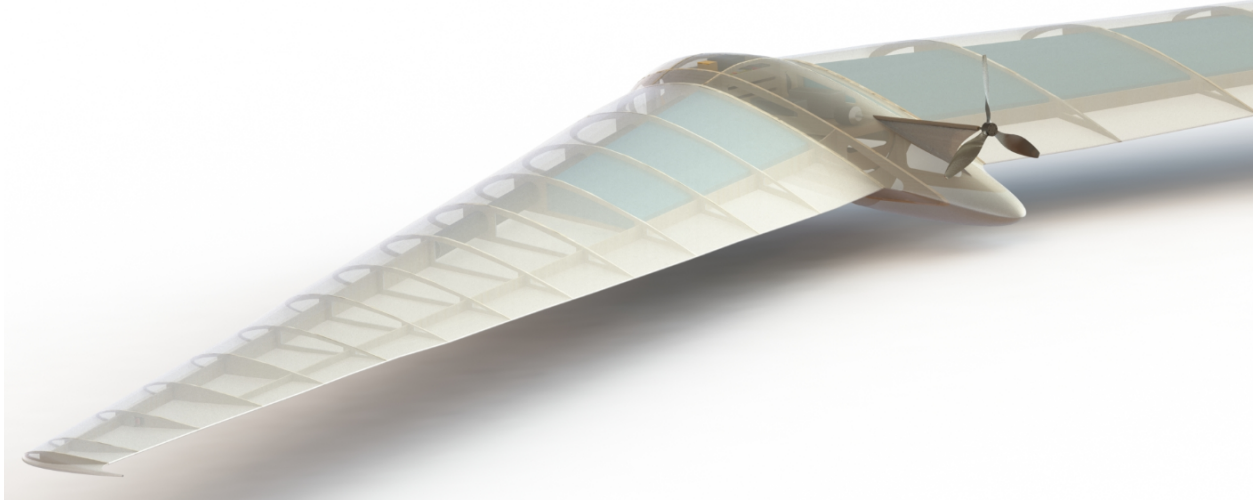


Figure 61: Bellwether3

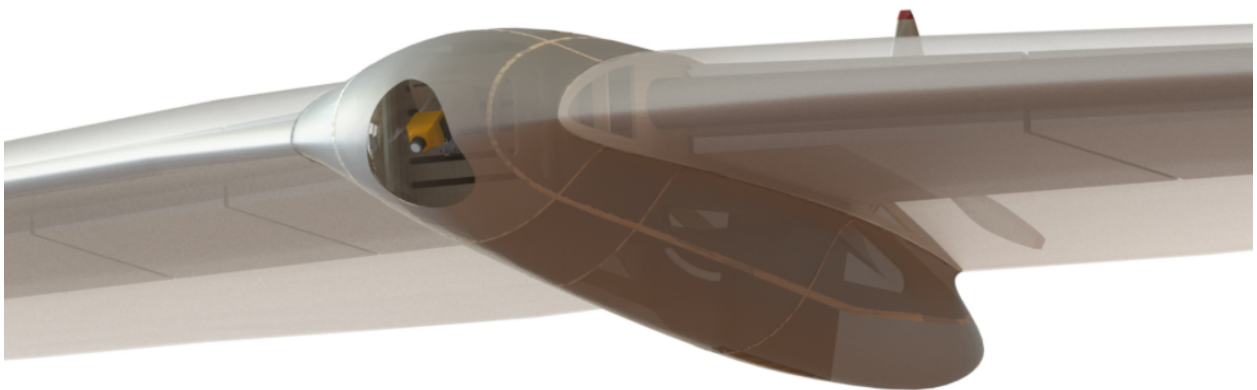


Figure 62: Bellwether4

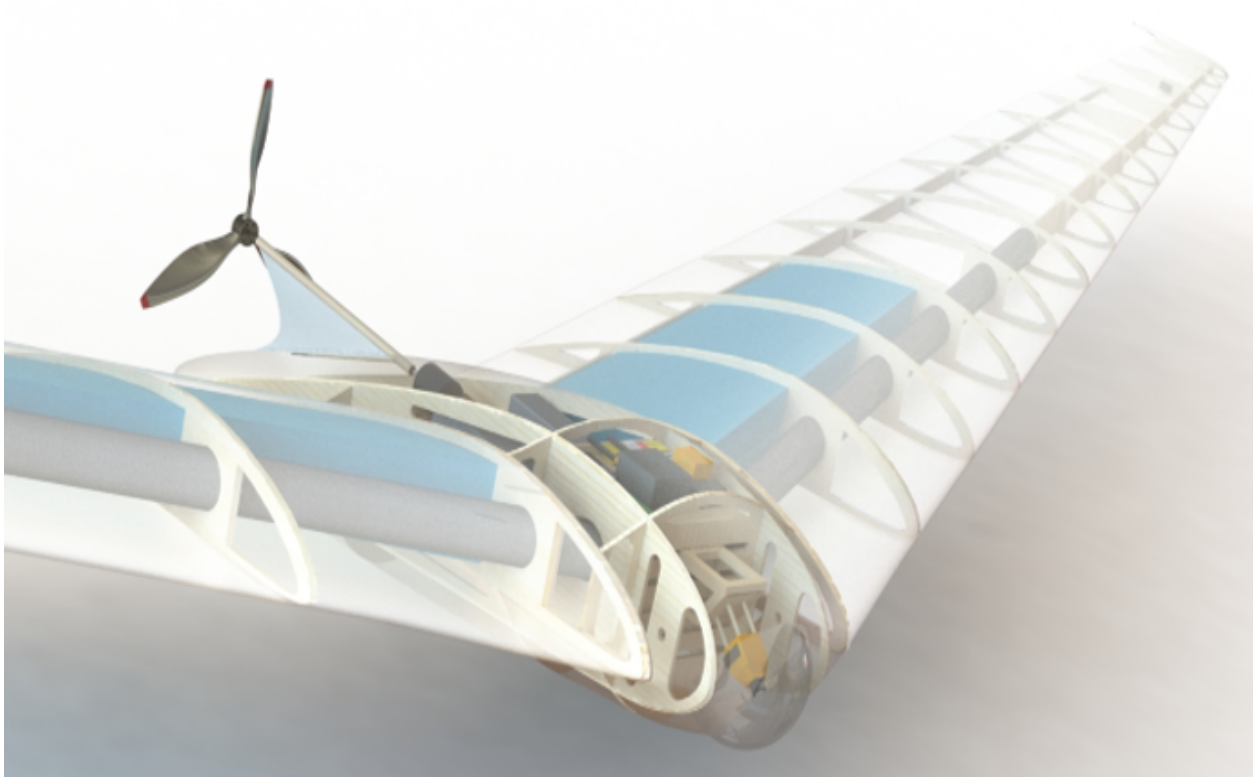


Figure 63: Bellwether5

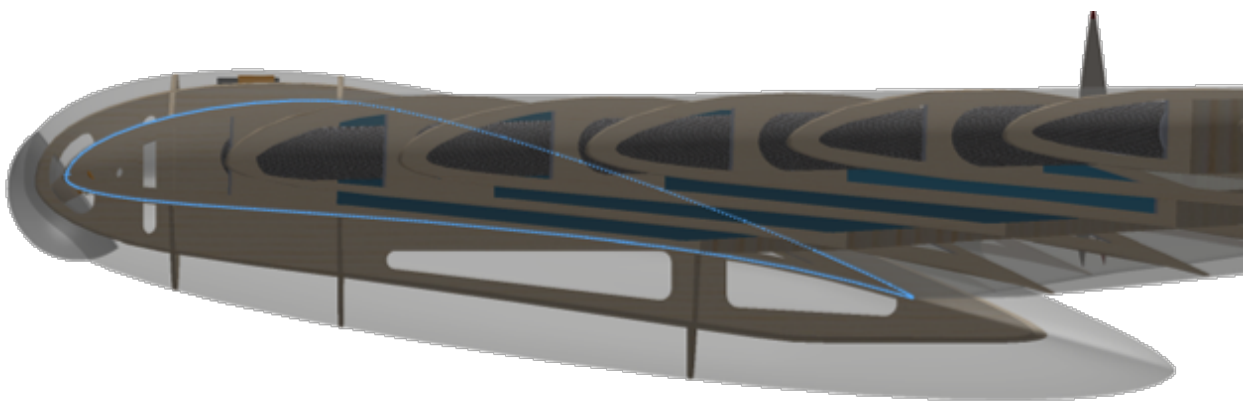


Figure 64: Bellwether6



## 0.2 Data Plots

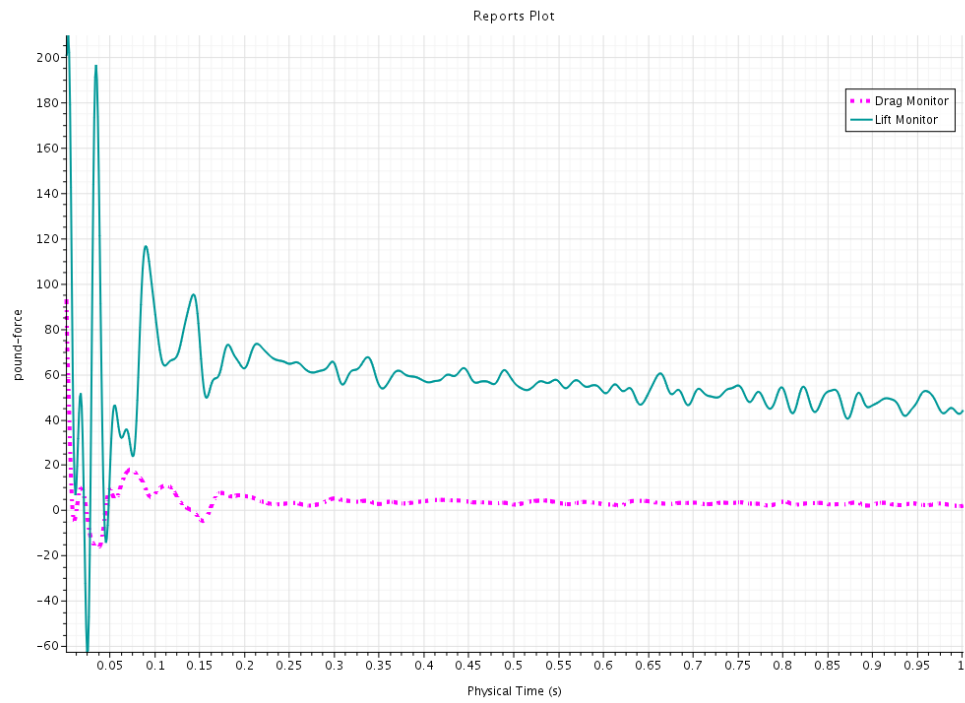


Figure 65: Lift and Drag Monitor of Bellwether - Unsteady RANS results

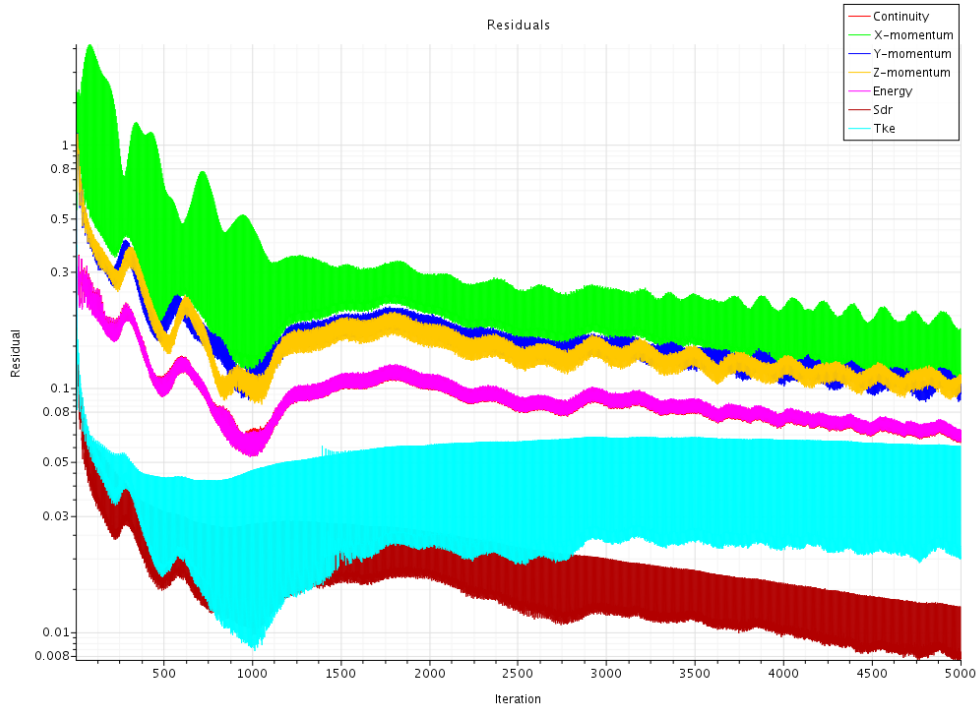


Figure 66: Residual Plot

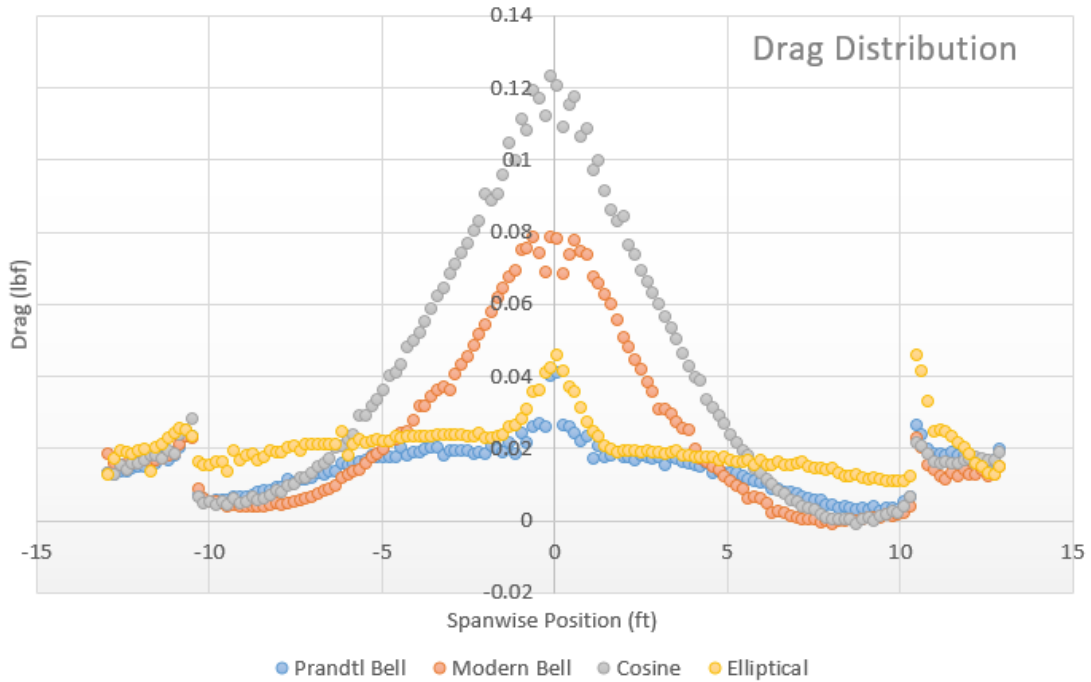


Figure 67: Drag Distribution in Roll

Relevant citations: Prandtl21 AdvancedUAV Bowers WingSections UASDesign [1] [3] XFLR5Low [18] [12] [21]

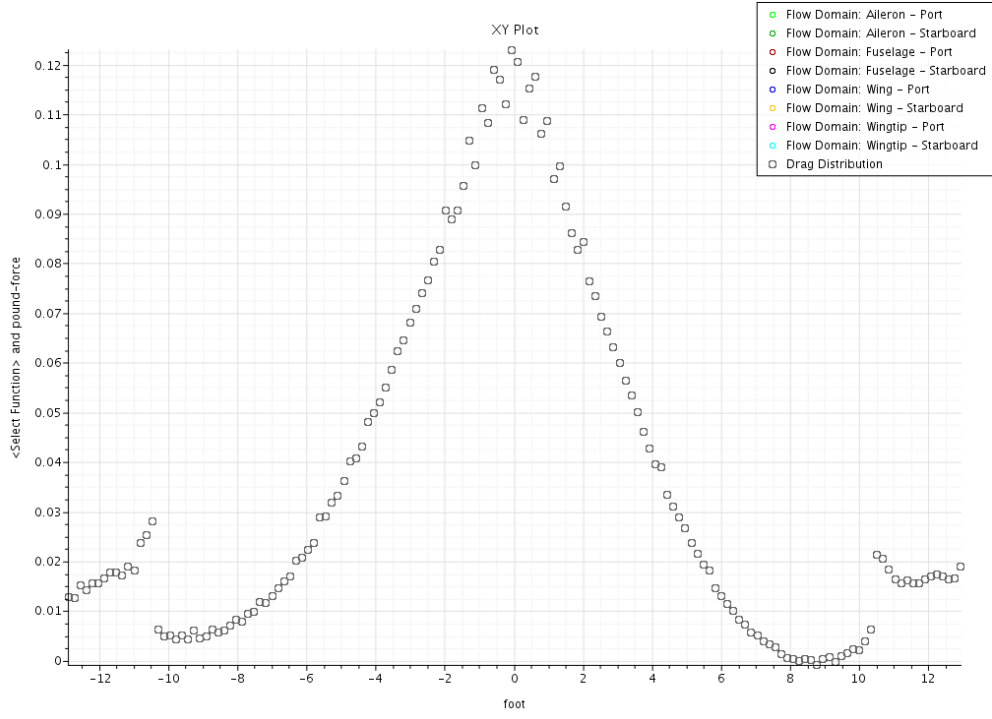


Figure 68: Cosine Model - Drag Distribution in Roll

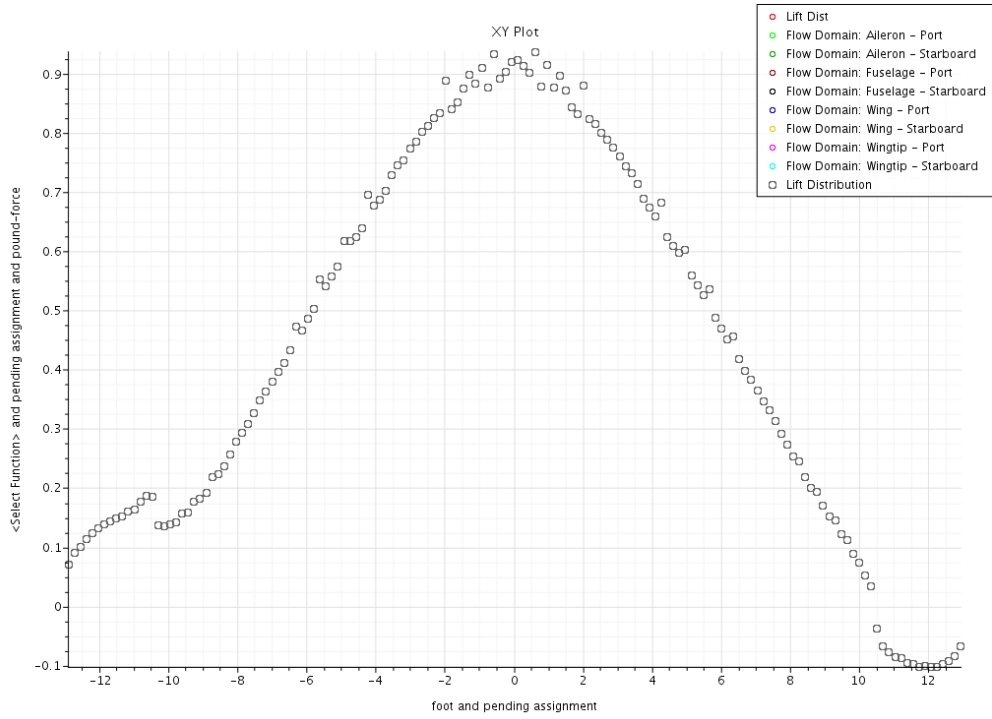


Figure 69: Cosine Model - Lift Distribution in Roll

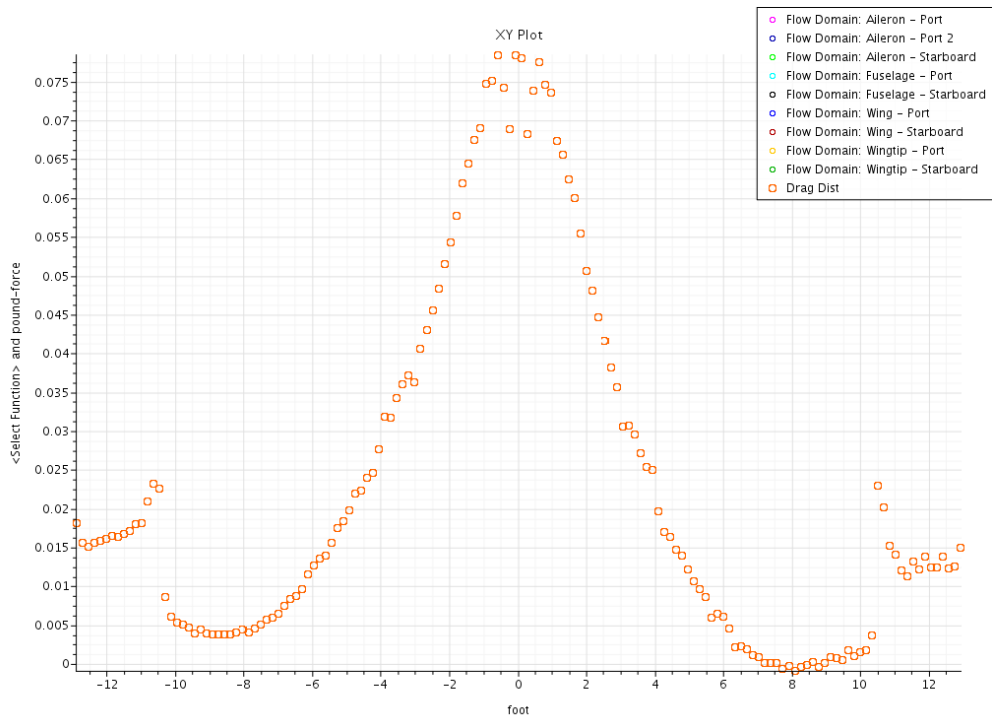


Figure 70: Modern Bell - Drag Distribution in Roll (bin direction is backwards on this plot)

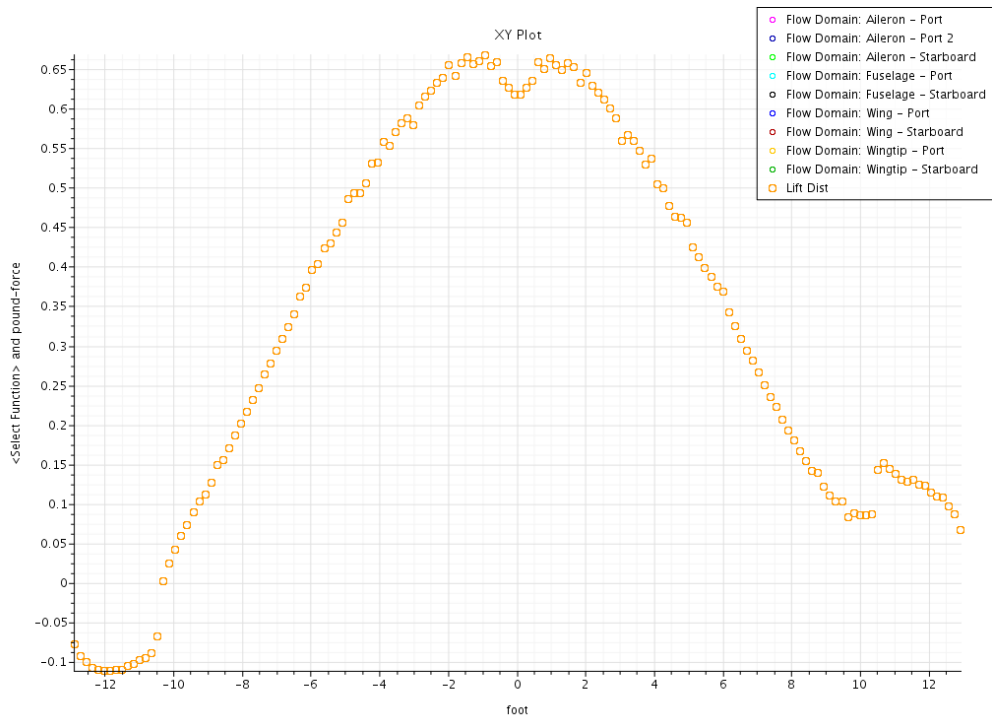


Figure 71: Modern Bell - Lift Distribution in Roll (bin direction is backwards on this plot)

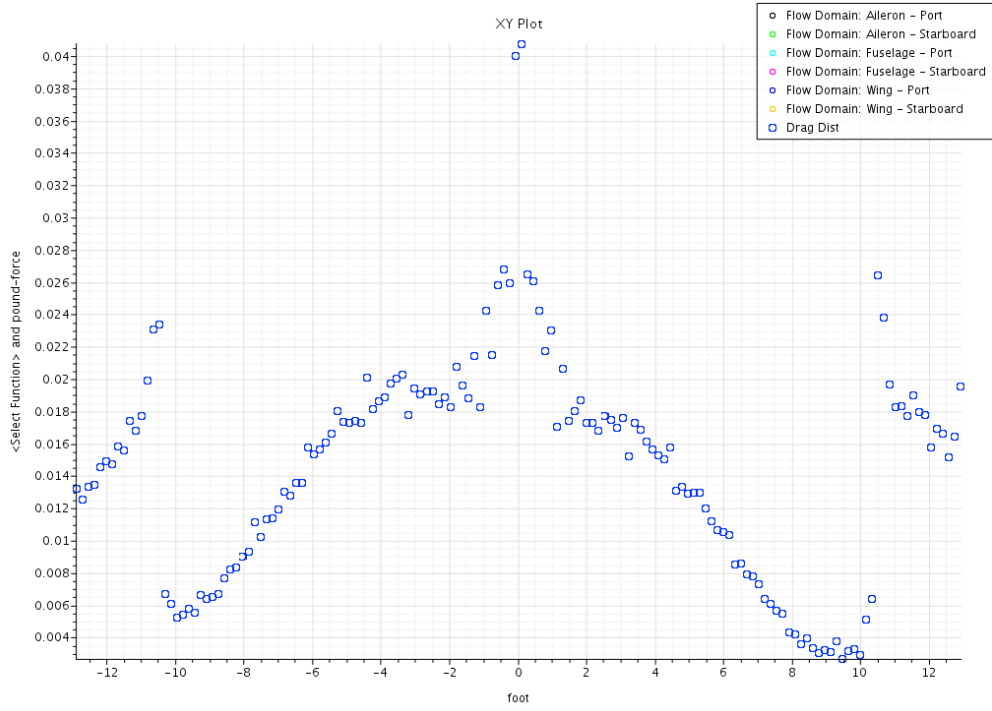


Figure 72: Prandtl Bell - Drag Distribution in Roll

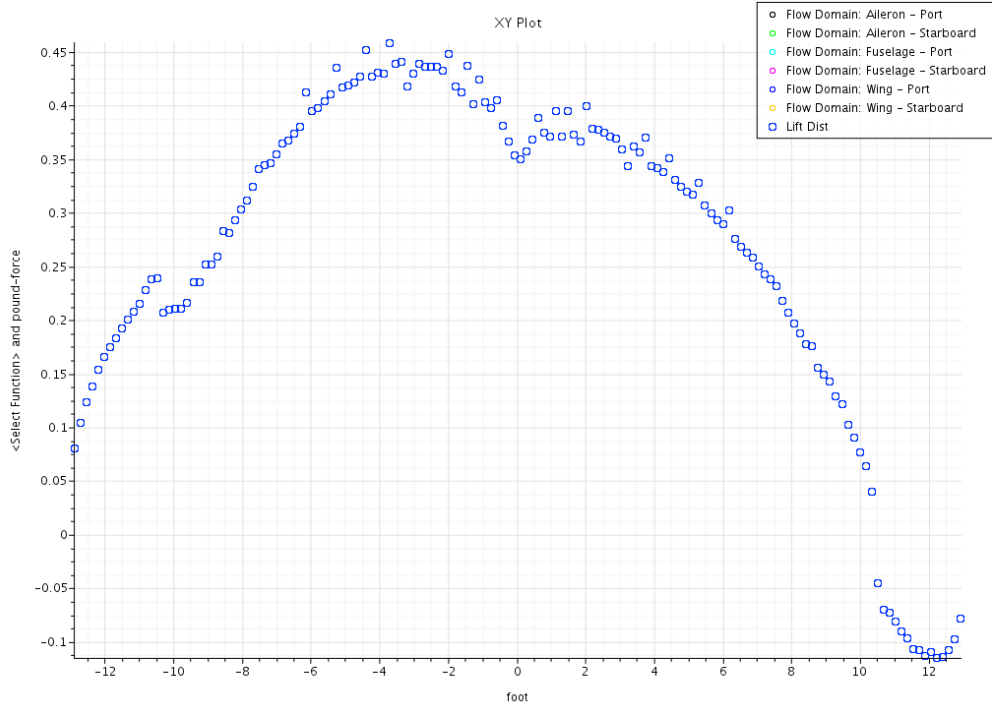


Figure 73: Prandtl - Lift Distribution in Roll

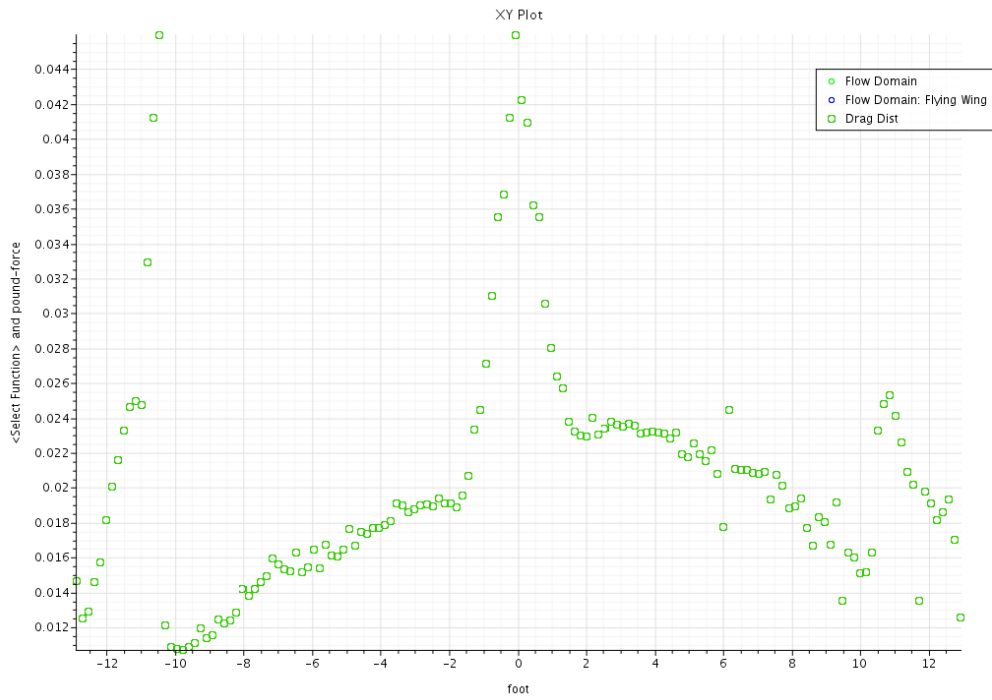


Figure 74: Elliptical - Drag Distribution in Roll (bin direction is backwards on this plot)

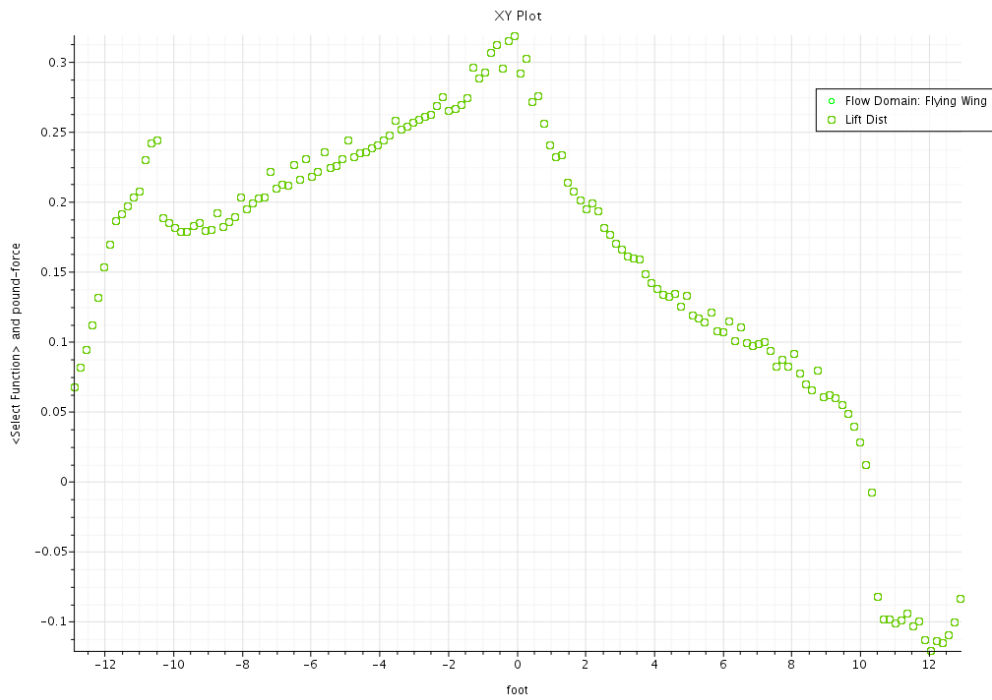


Figure 75: Elliptical - Lift Distribution in Roll

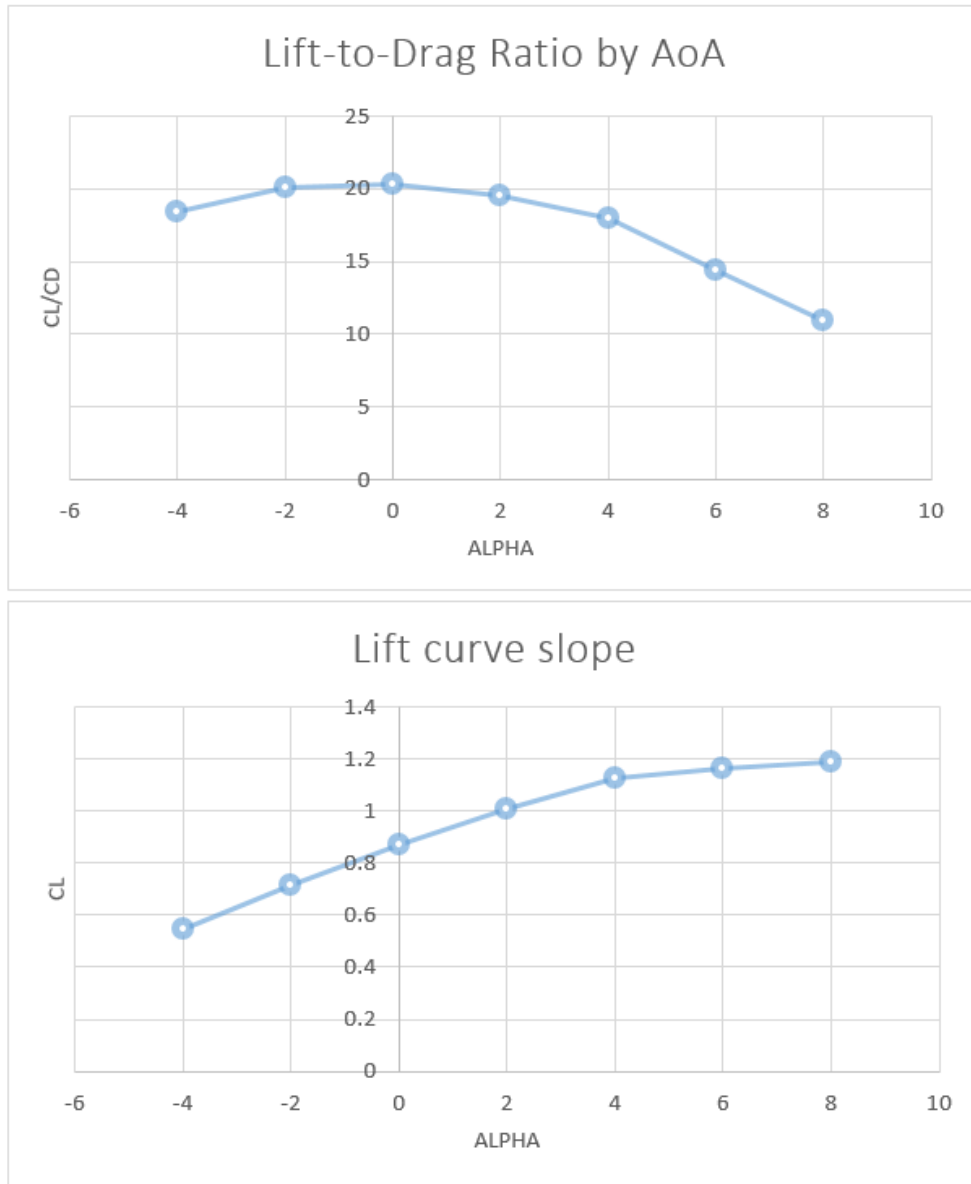


Figure 76: Combine Alpha Sweep

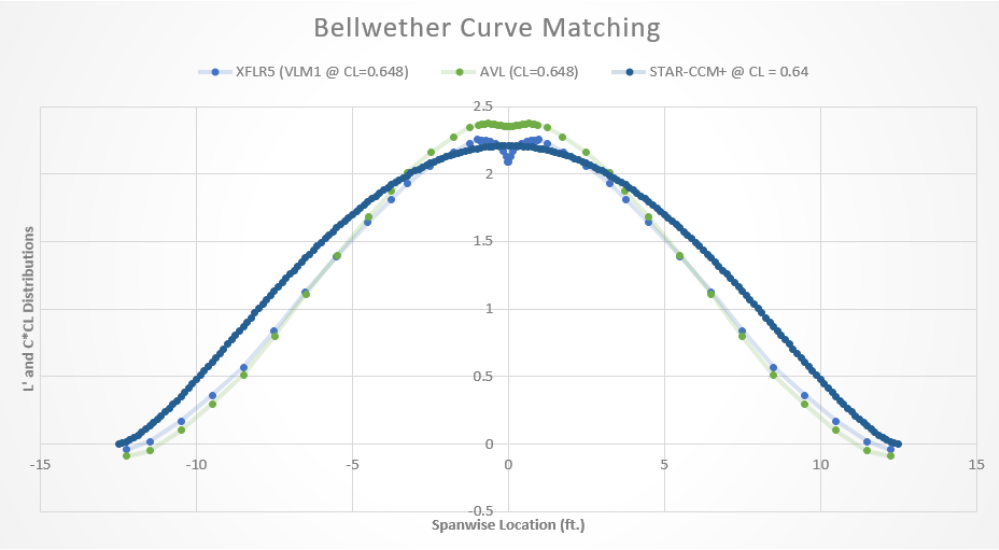


Figure 77: Distributions at Common CL of All Analysis Methods



### 0.3 CFD Mesh Scenes

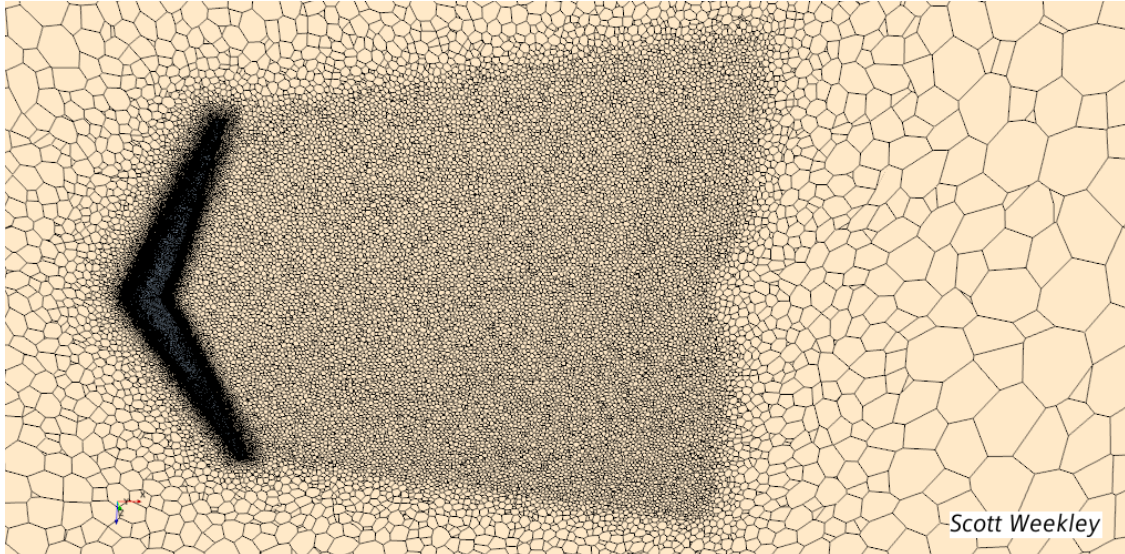


Figure 78: Wake Refinement

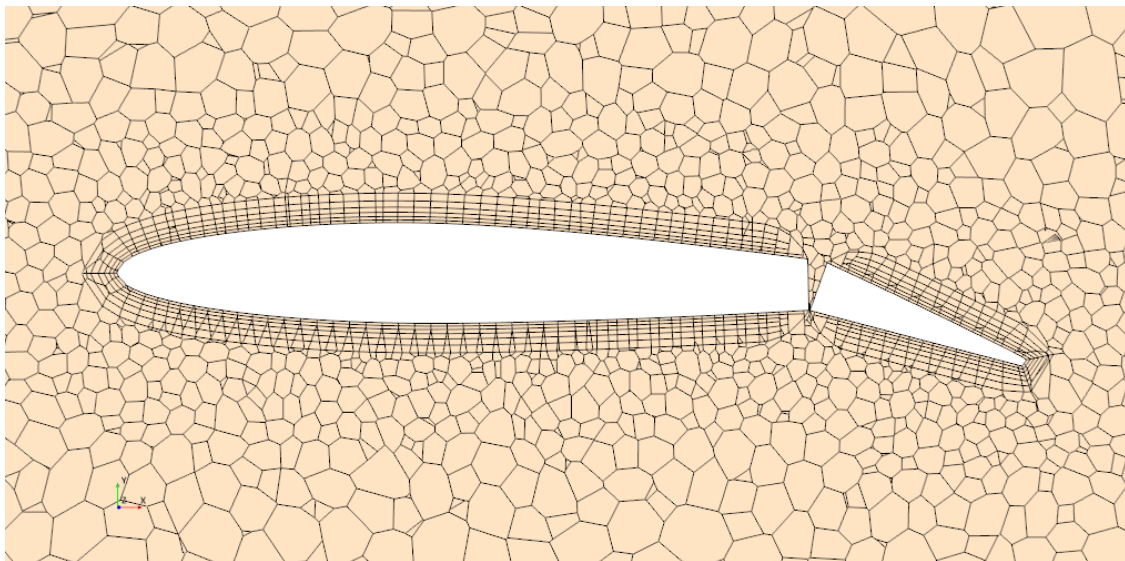


Figure 79: Aileron Mesh 1

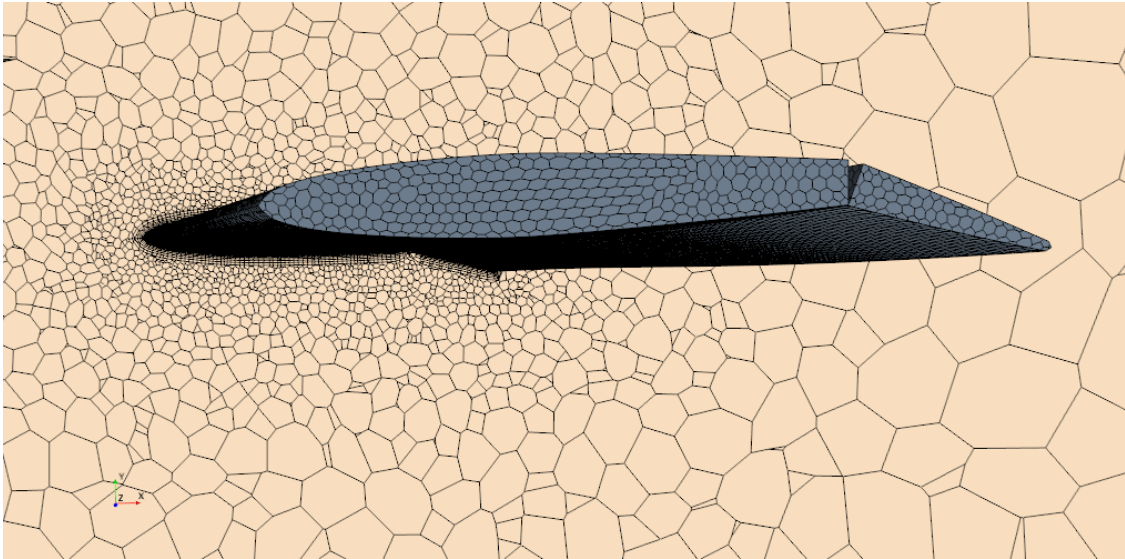


Figure 80: Aileron Mesh 2

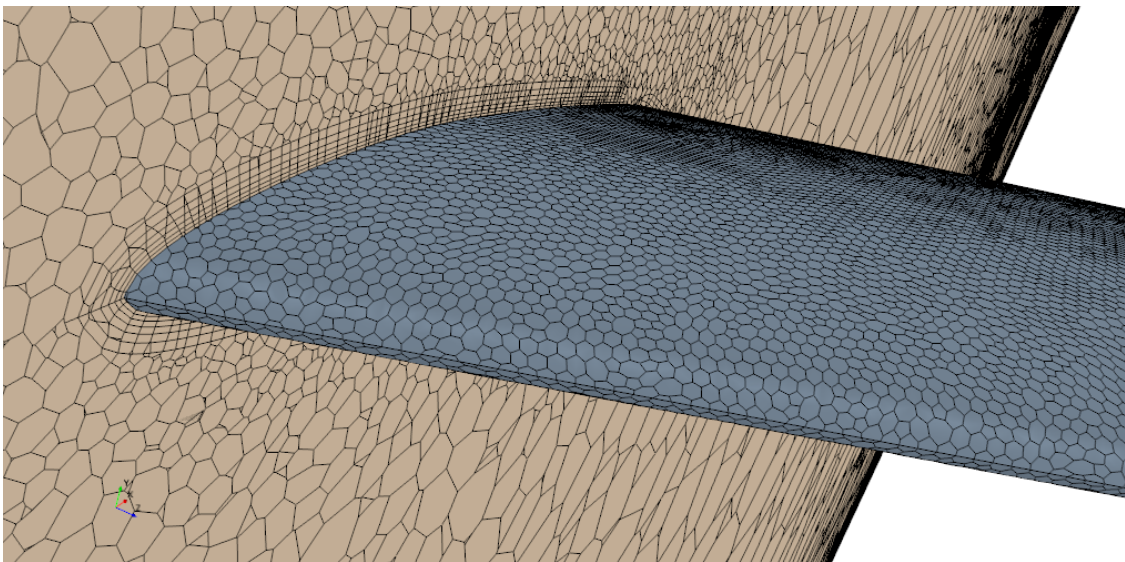


Figure 81: Aileron Mesh 3

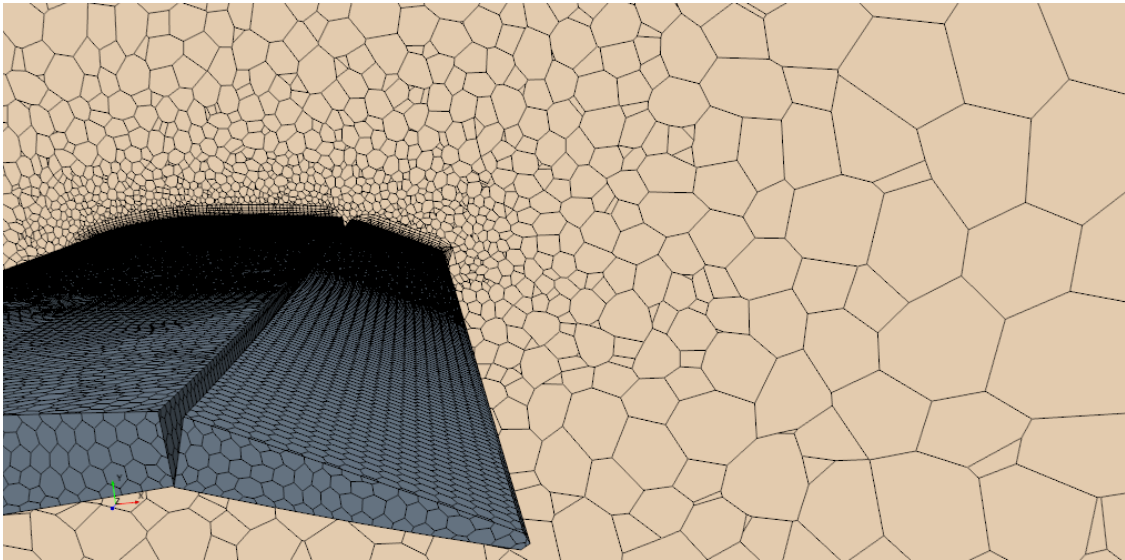


Figure 82: Aileron Mesh 4

## 0.4 Aircraft Mold Design and Fabrication

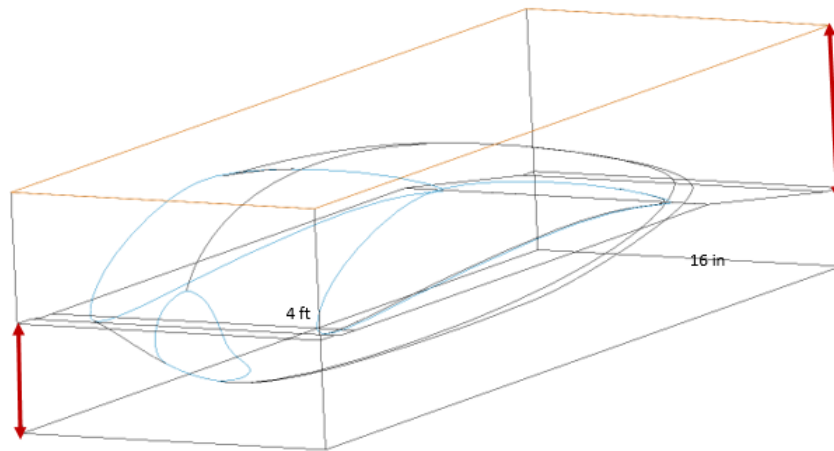


Figure 83: Fuselage Mold

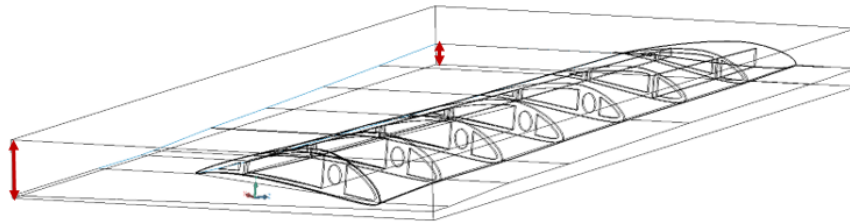


Figure 84: Inboard Wing Mold

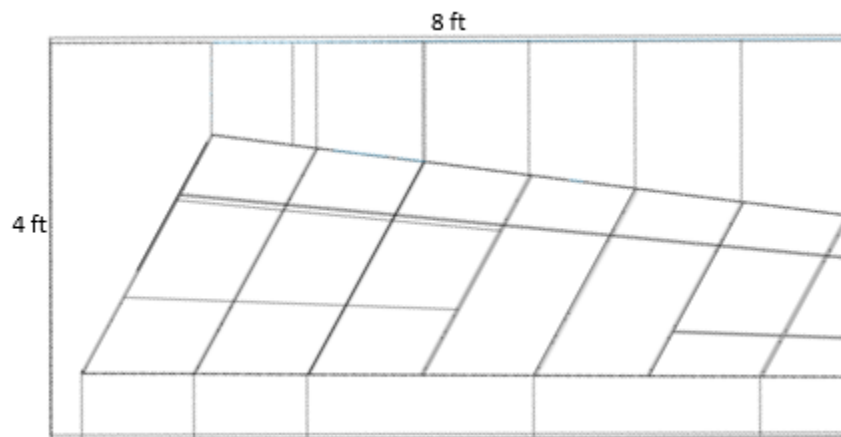


Figure 85: Inboard Wing Mold - Top View

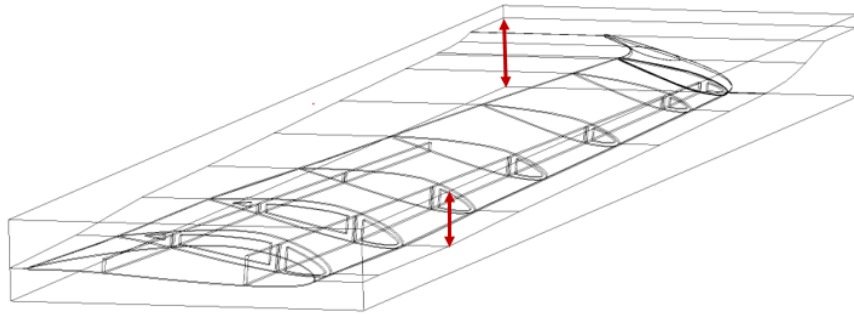


Figure 86: Outboard Wing Mold

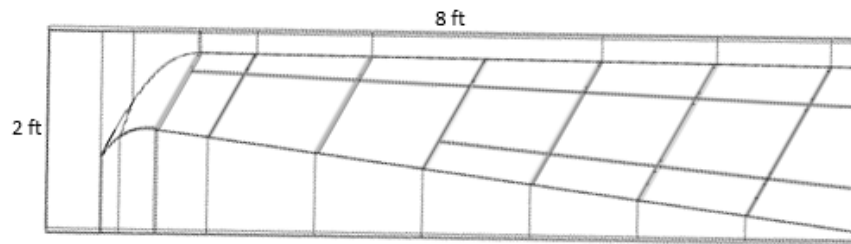


Figure 87: Outboard Wing Mold - Top View



Figure 88: Mold 1



Figure 89: Mold 2



Figure 90: Mold3

## 0.5 Experimental Models and Images

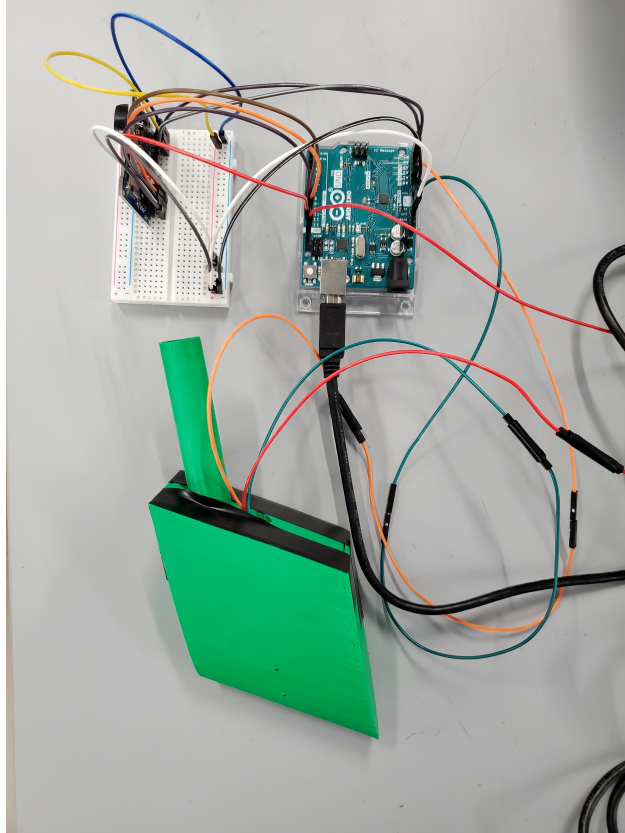


Figure 91: System of Components for Leading-Edge Flow Sensing Sleeve

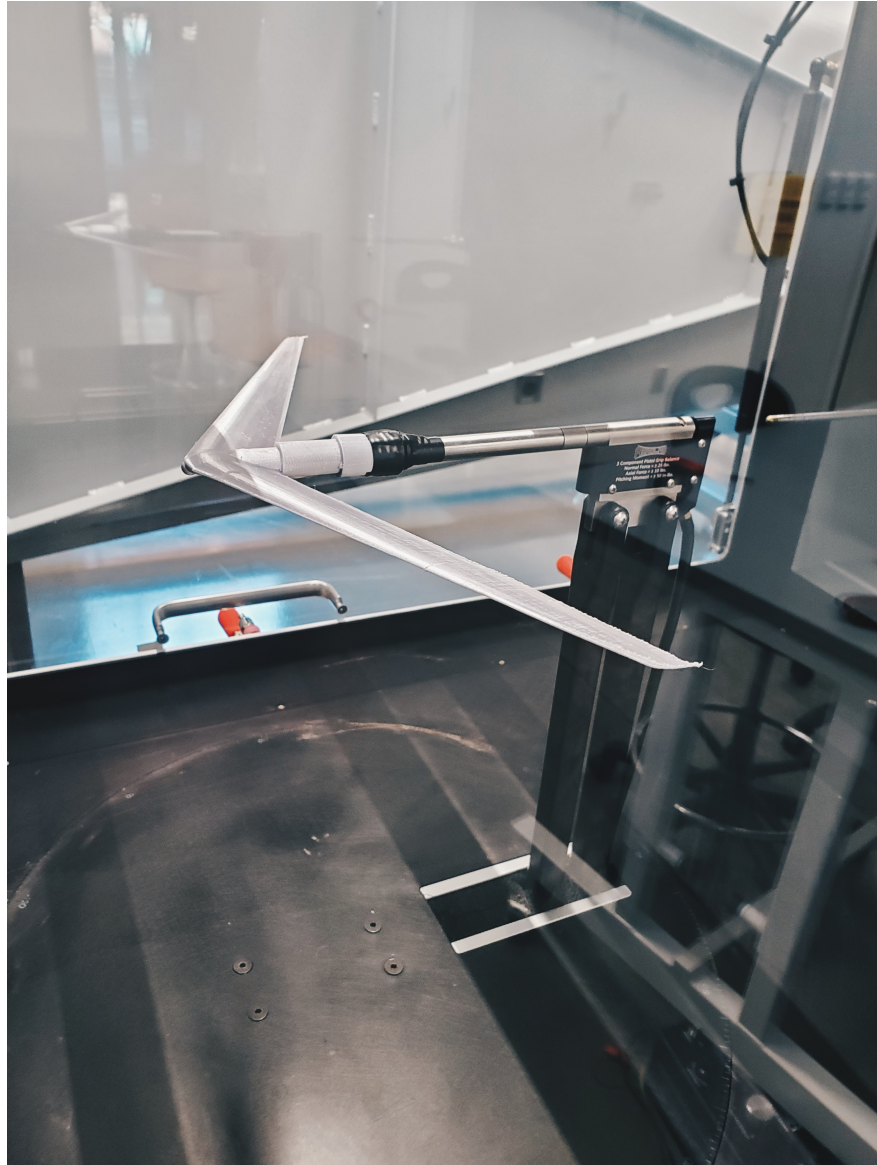


Figure 92: Wind Tunnel Setup for 3D-Printed Bellwether Model



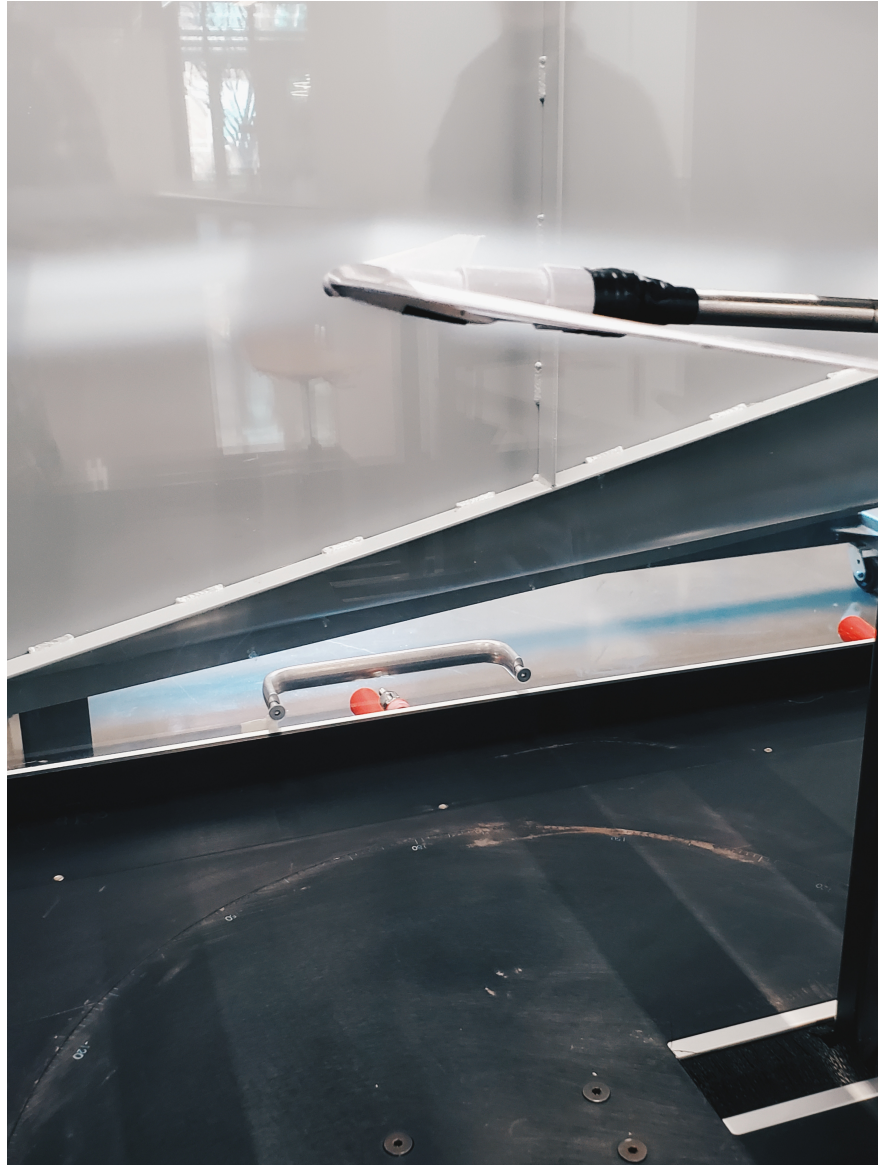


Figure 93: Wind Tunnel Flow Visualization



Figure 94: Wind Tunnel 1

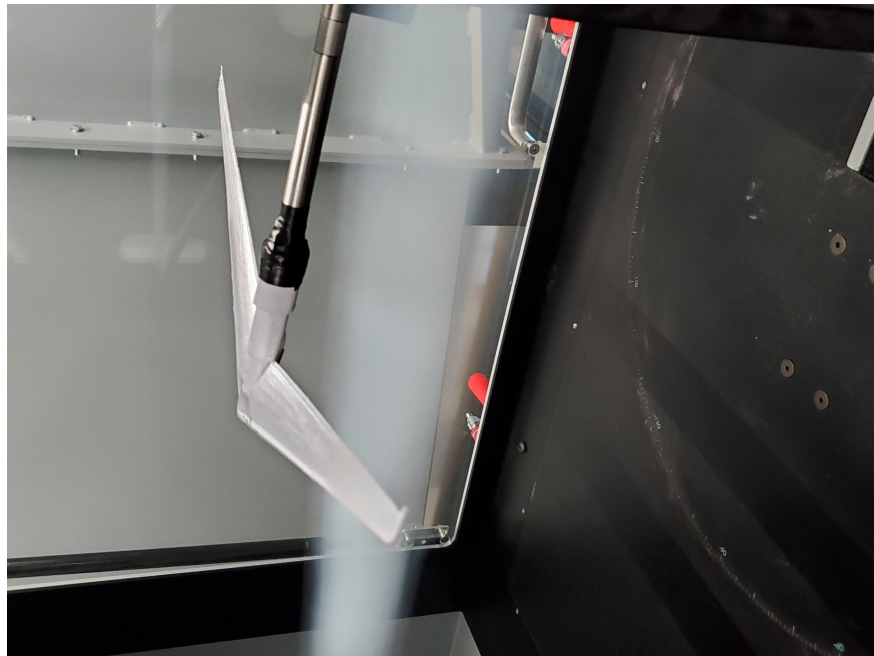


Figure 95: Wind Tunnel 2

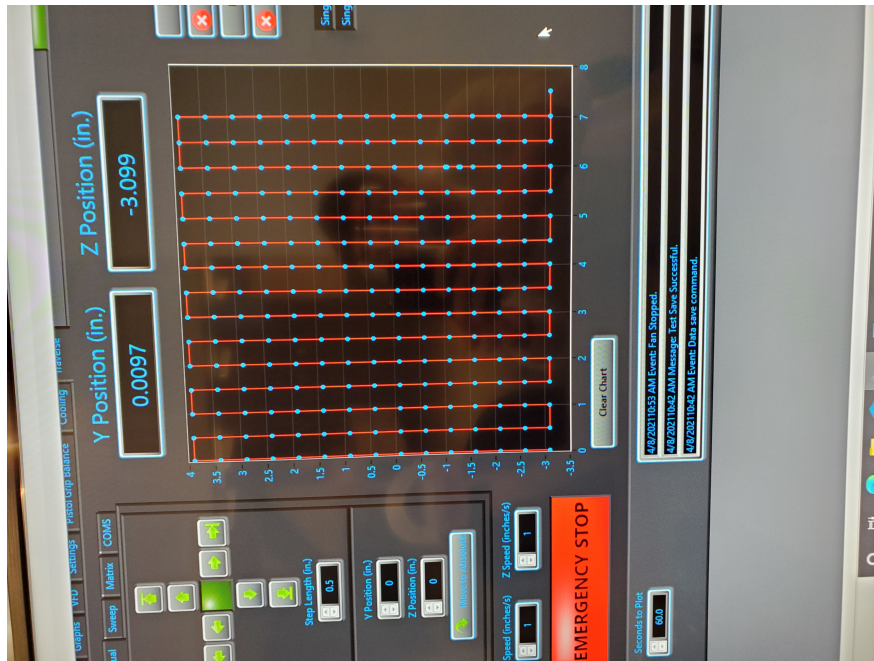


Figure 96: Wind Tunnel 3

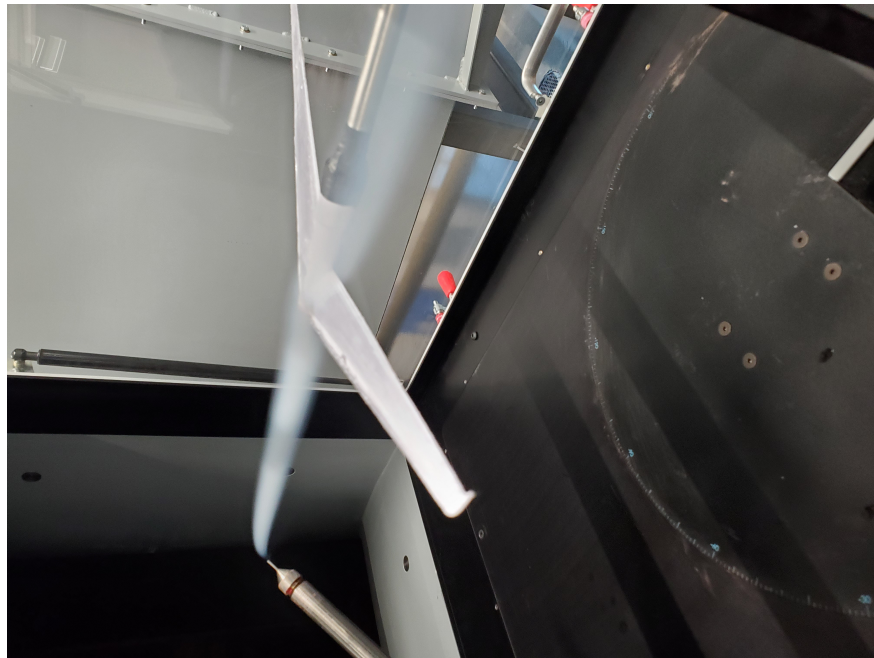


Figure 97: Wind Tunnel 4

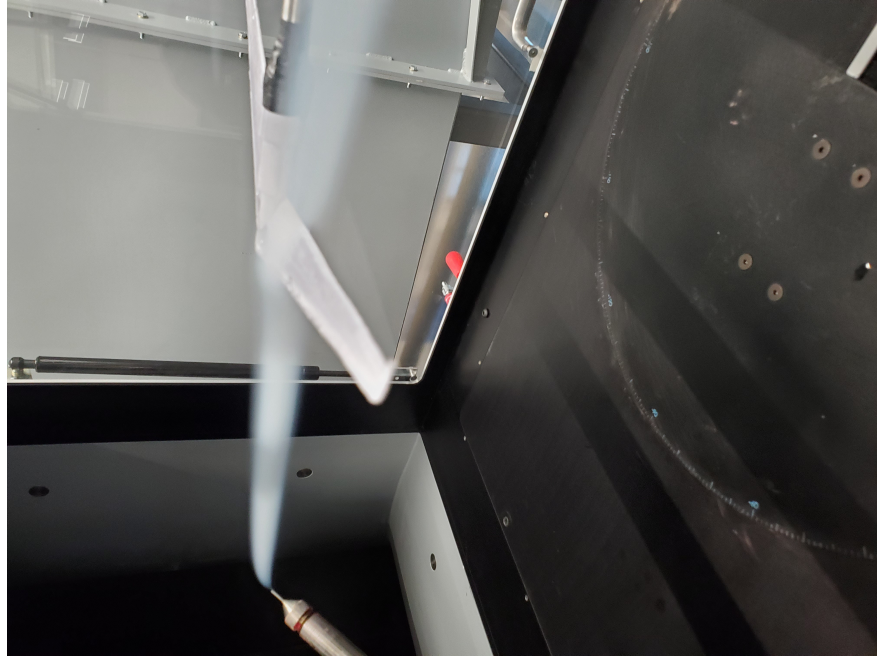


Figure 98: Wind Tunnel 5

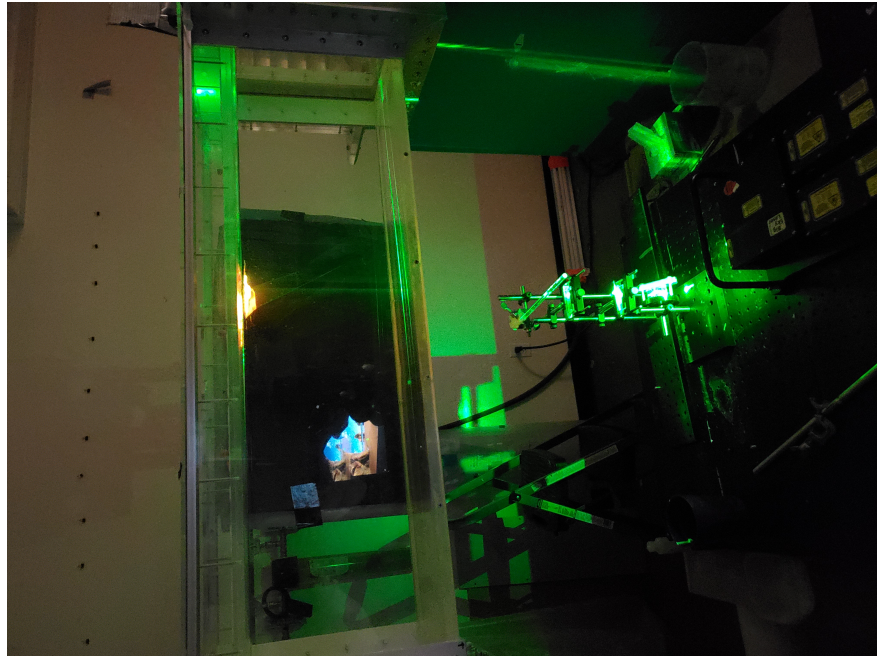


Figure 99: Water Tunnel 1



Figure 100: Water Tunnel 2



Figure 101: Water Tunnel 3



Figure 102: Water Tunnel 4

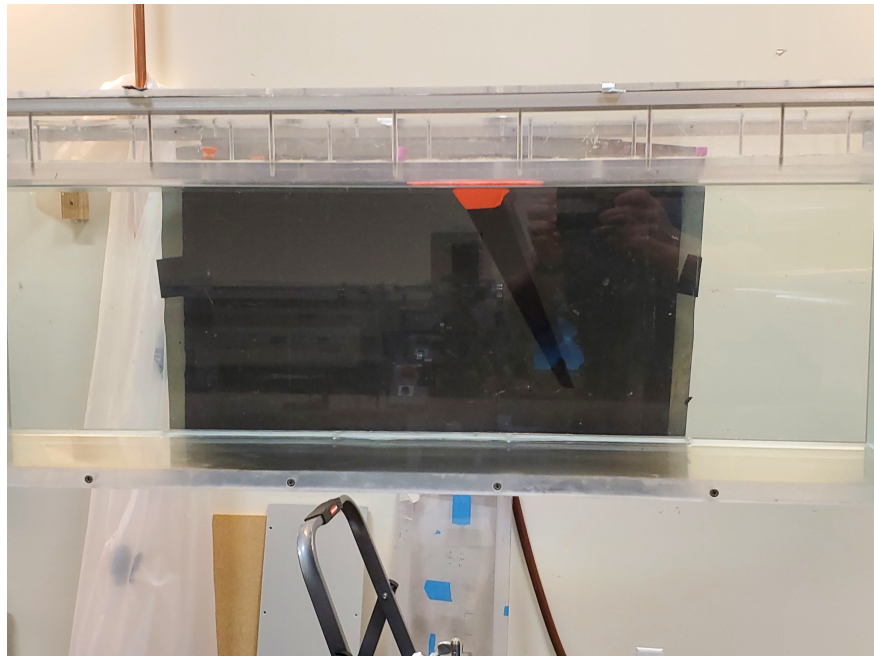


Figure 103: Water Tunnel 5

## 0.6 CFD Streamlines and Vorticity Scenes

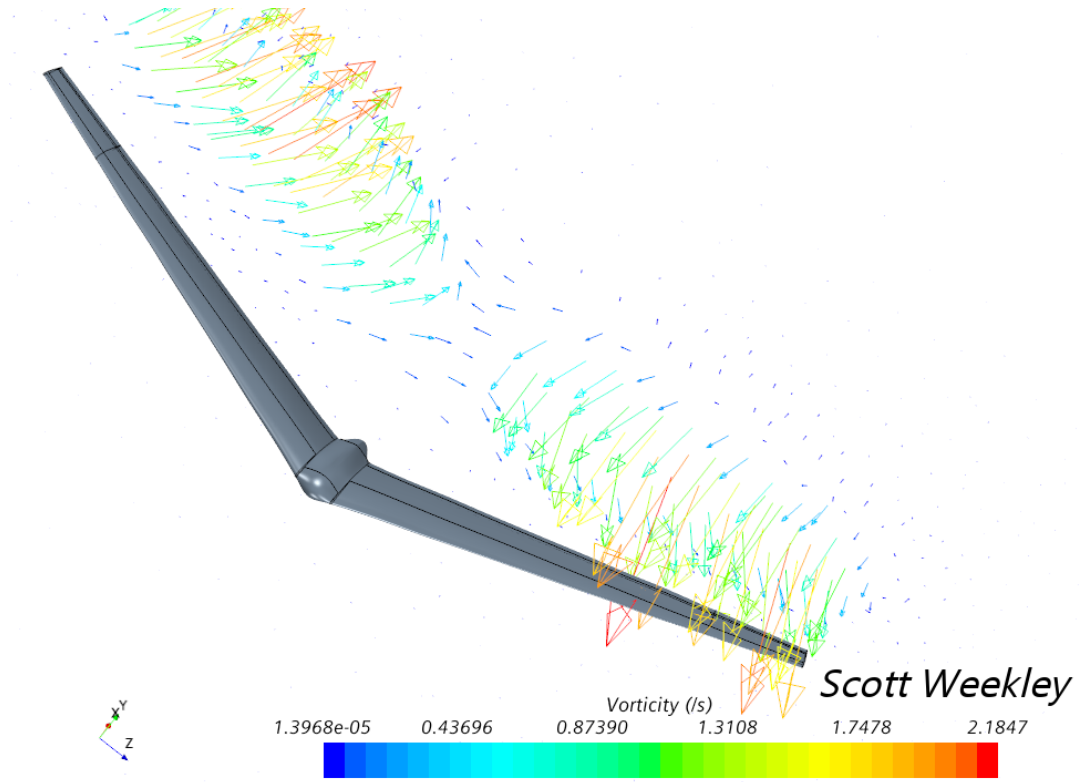


Figure 104: Bellwether Vorticity 1

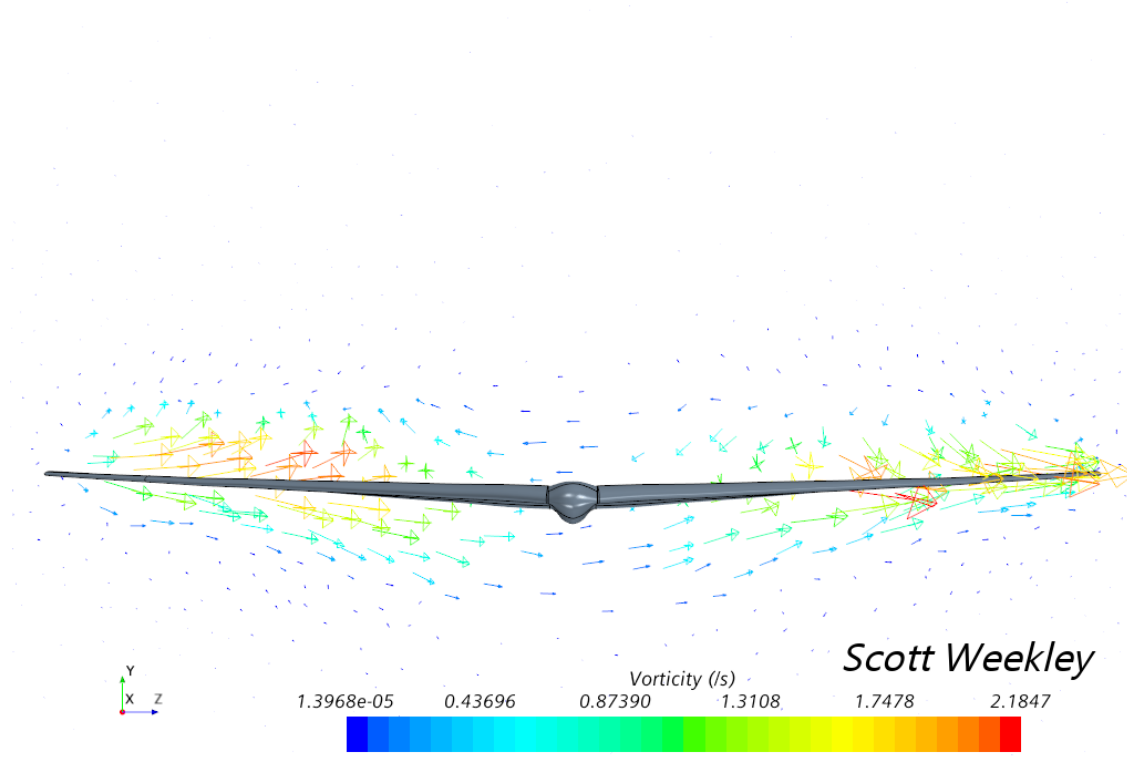


Figure 105: Bellwether Vorticity 2



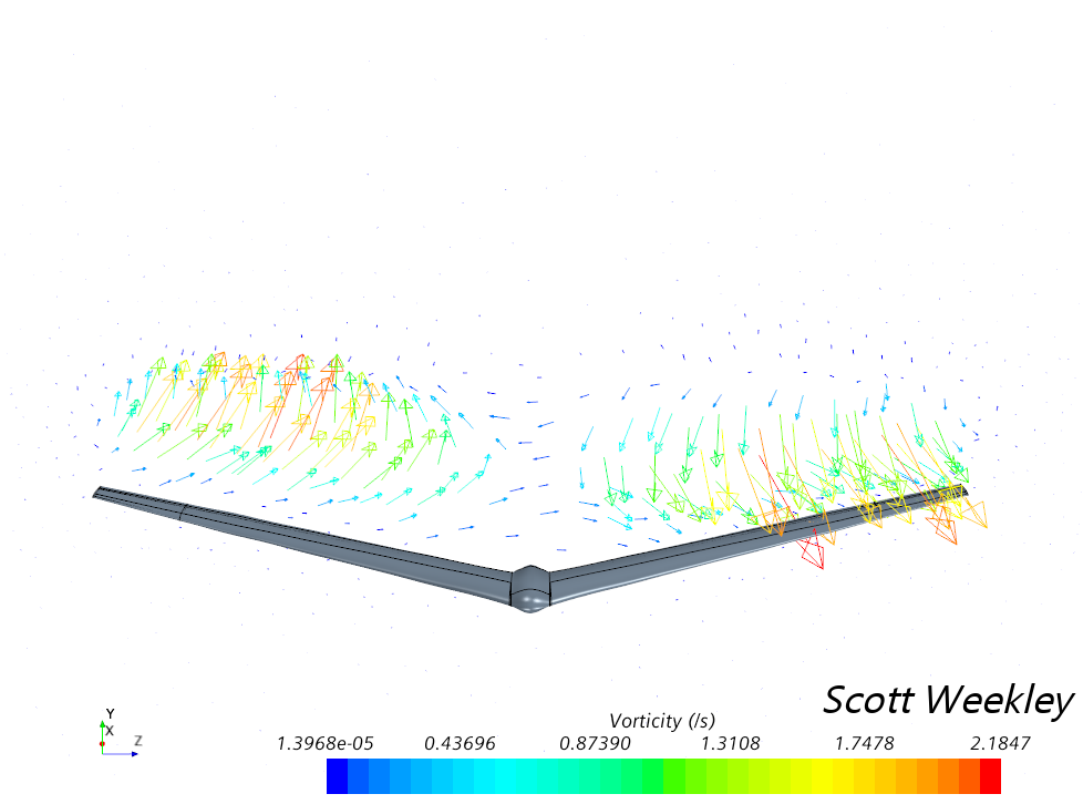


Figure 106: Bellwether Vorticity 3

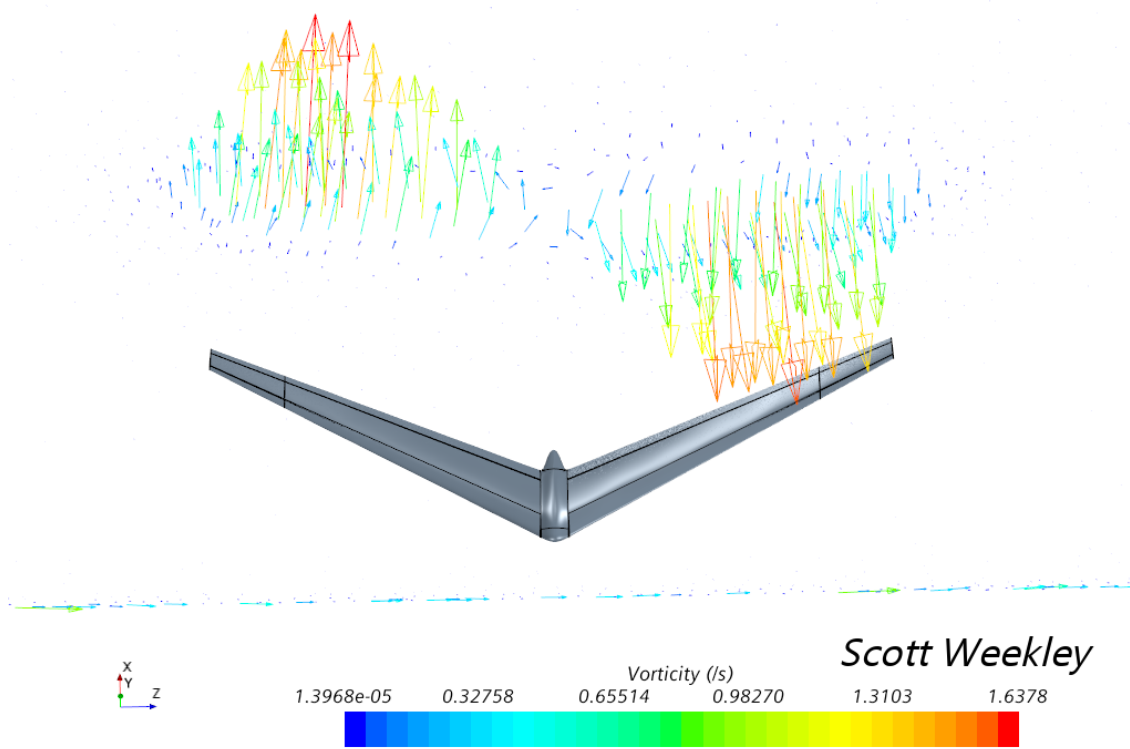


Figure 107: Bellwether Vorticity 4

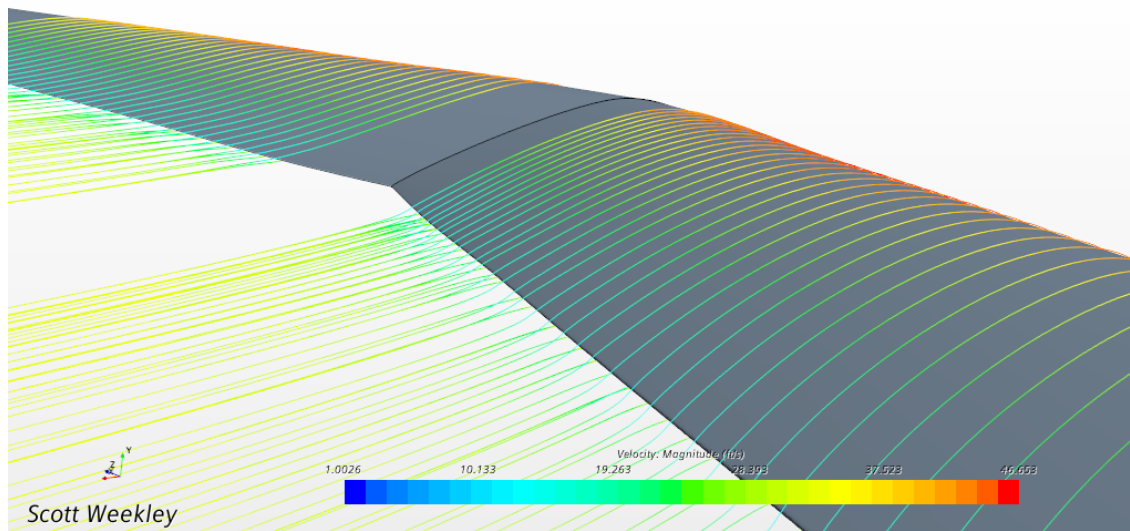


Figure 108: Bellwether Vorticity 5

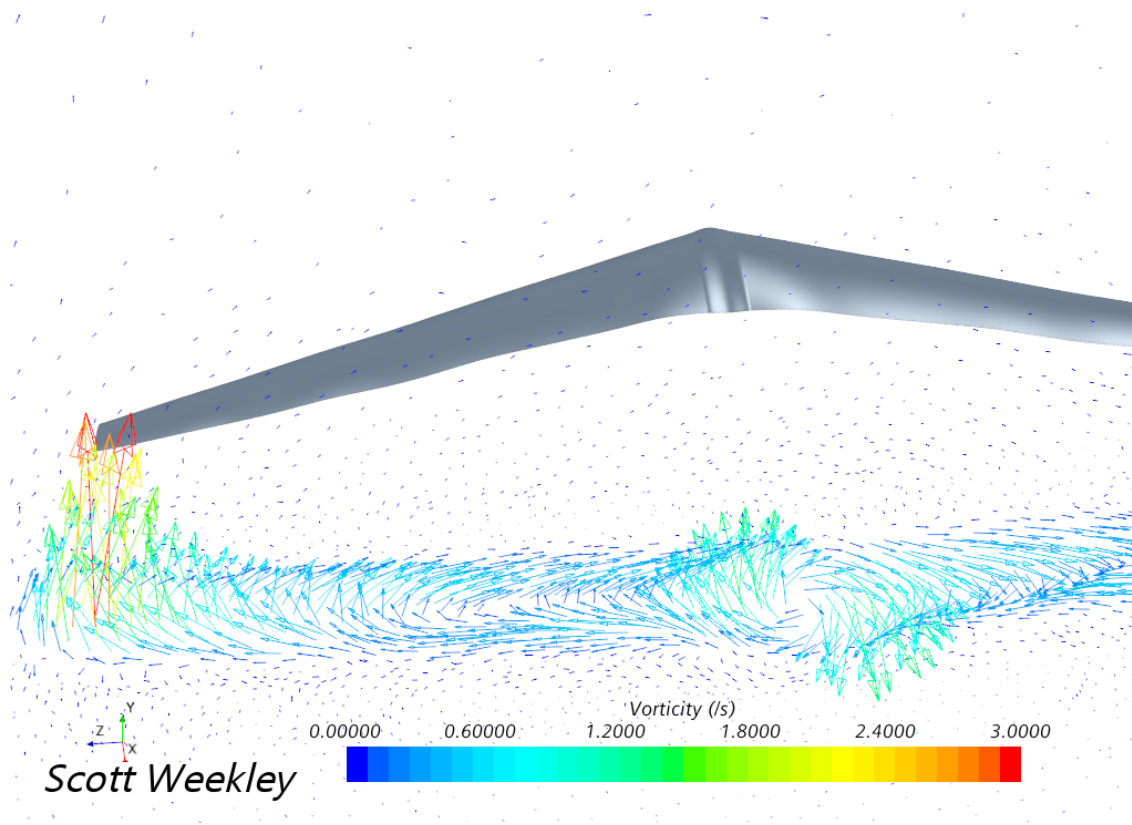


Figure 109: Elliptical Vorticity 1

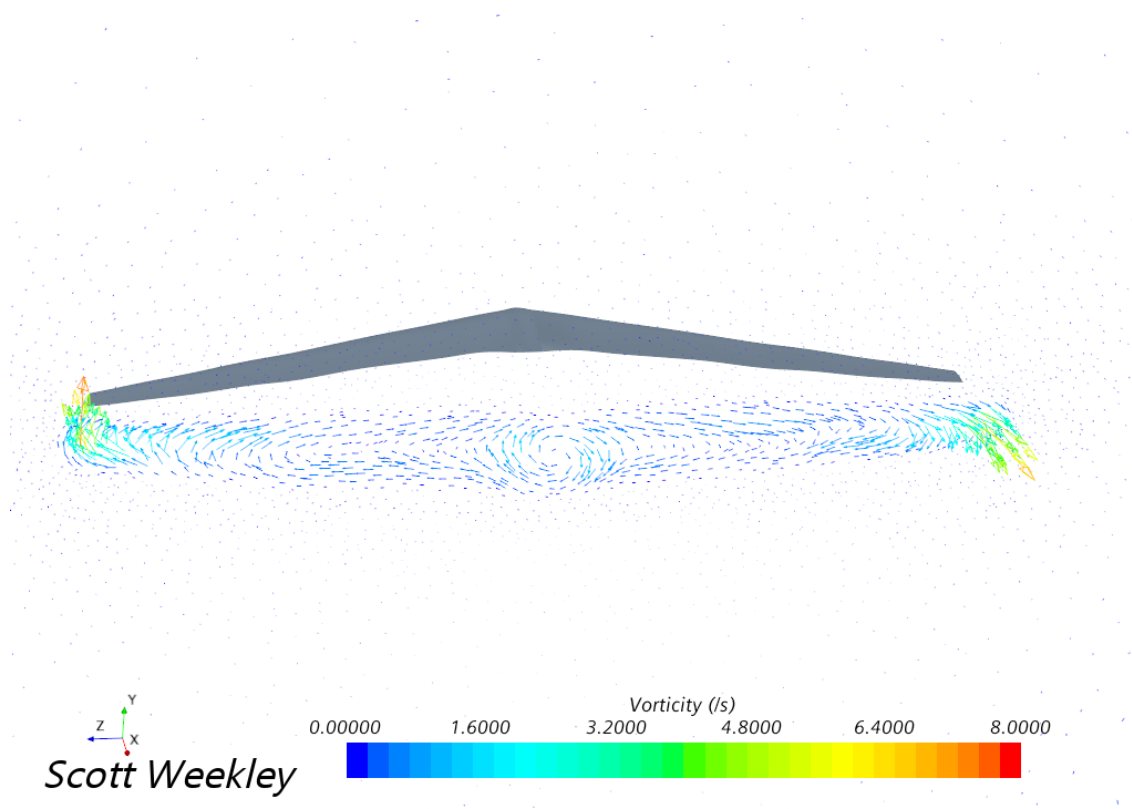


Figure 110: Elliptical Vorticity 2

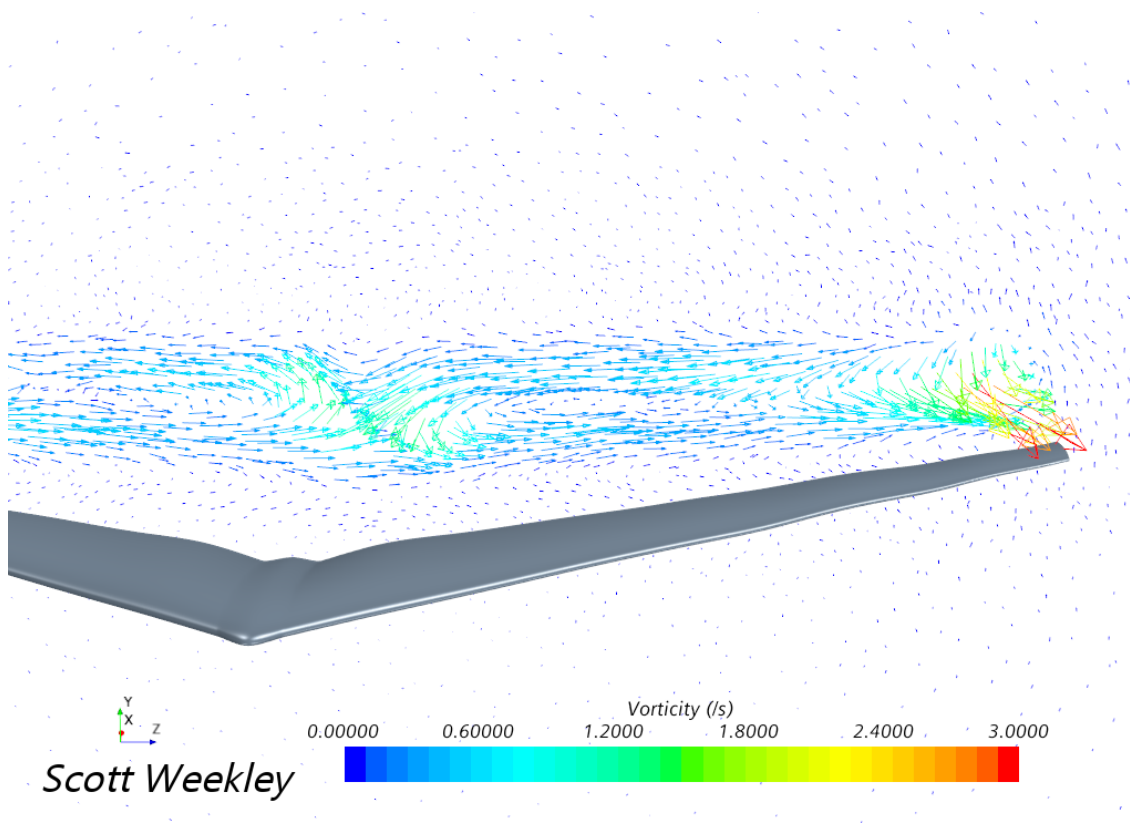


Figure 111: Elliptical Vorticity 3

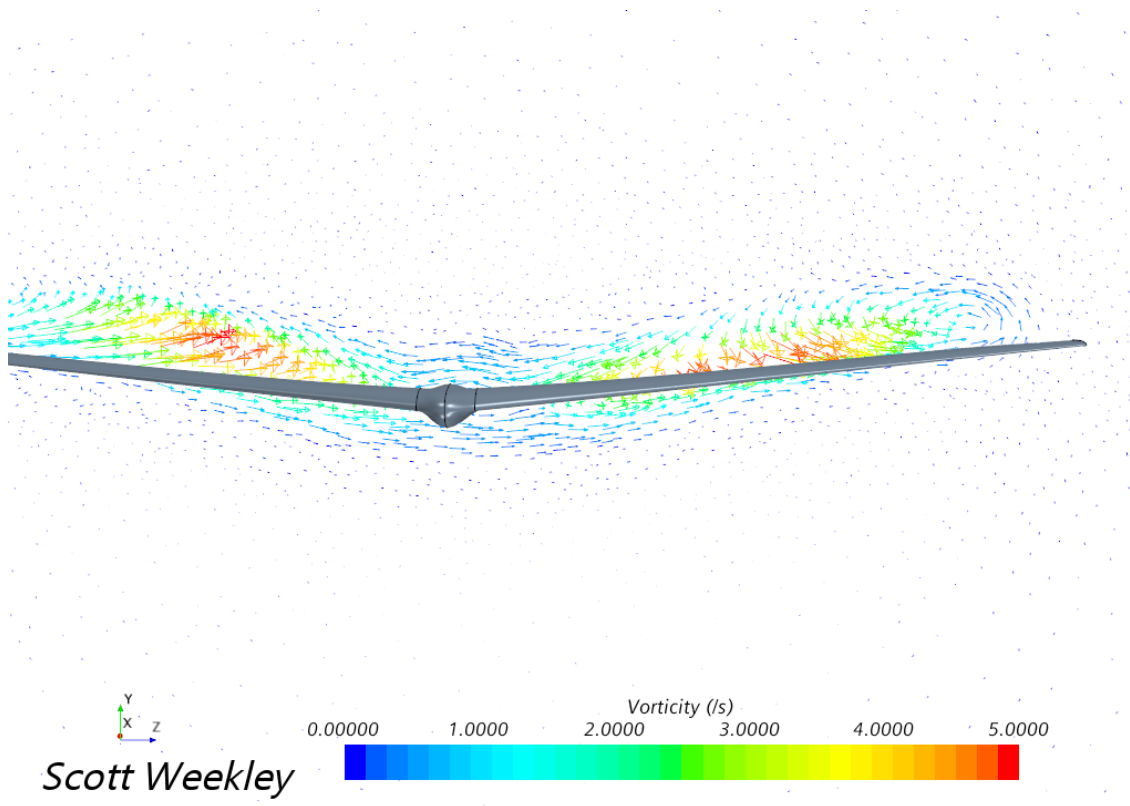


Figure 112: Cosine Vorticity 1

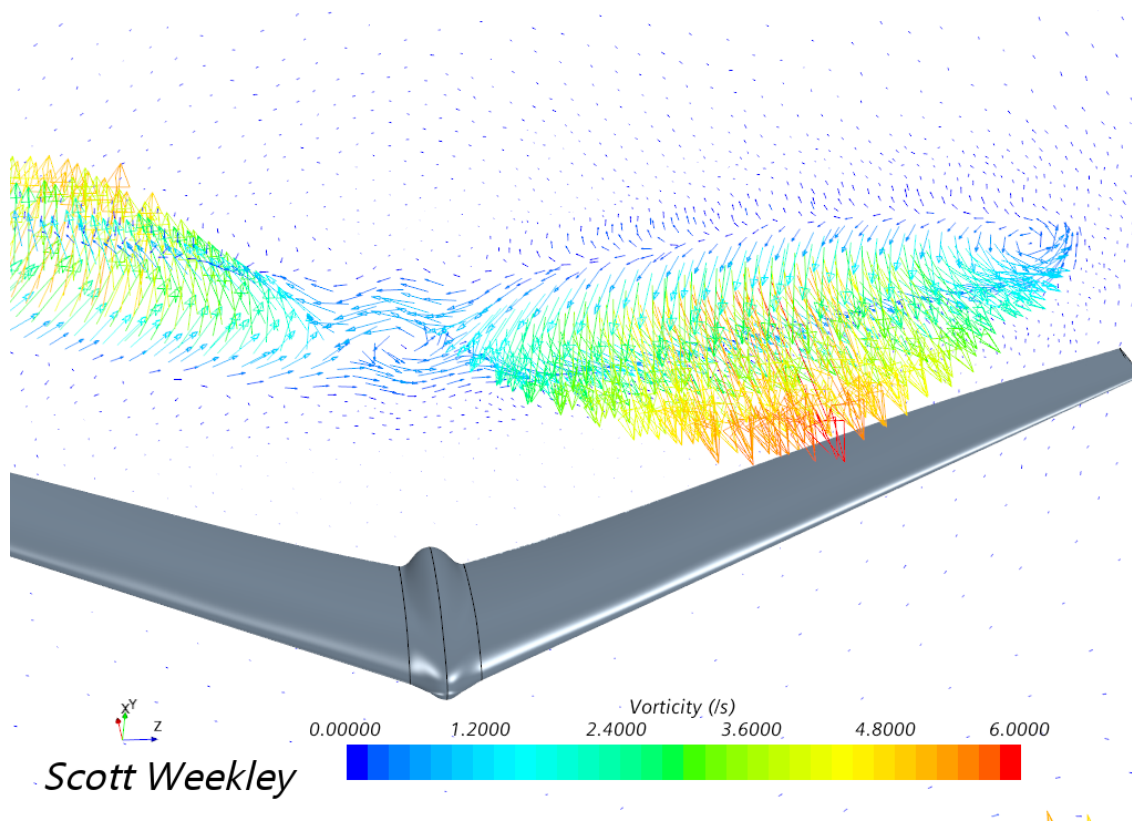


Figure 113: Cosine Vorticity 2

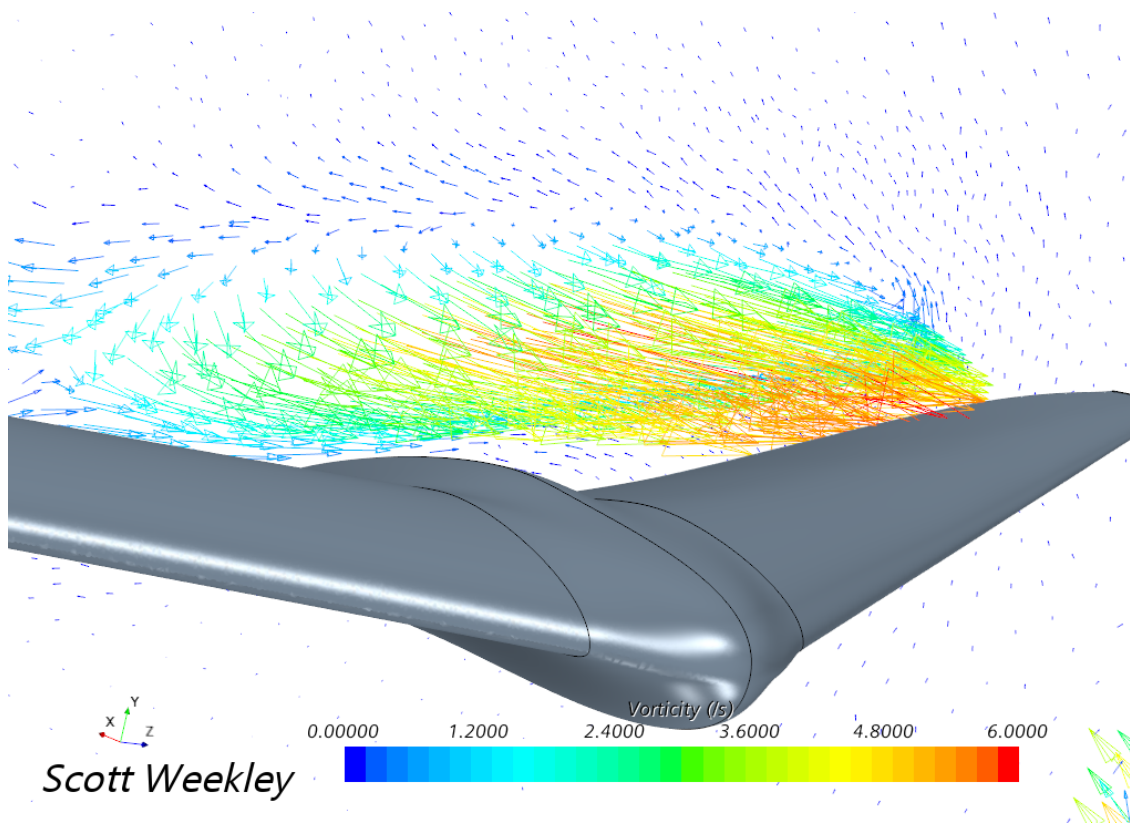


Figure 114: Cosine Vorticity 3



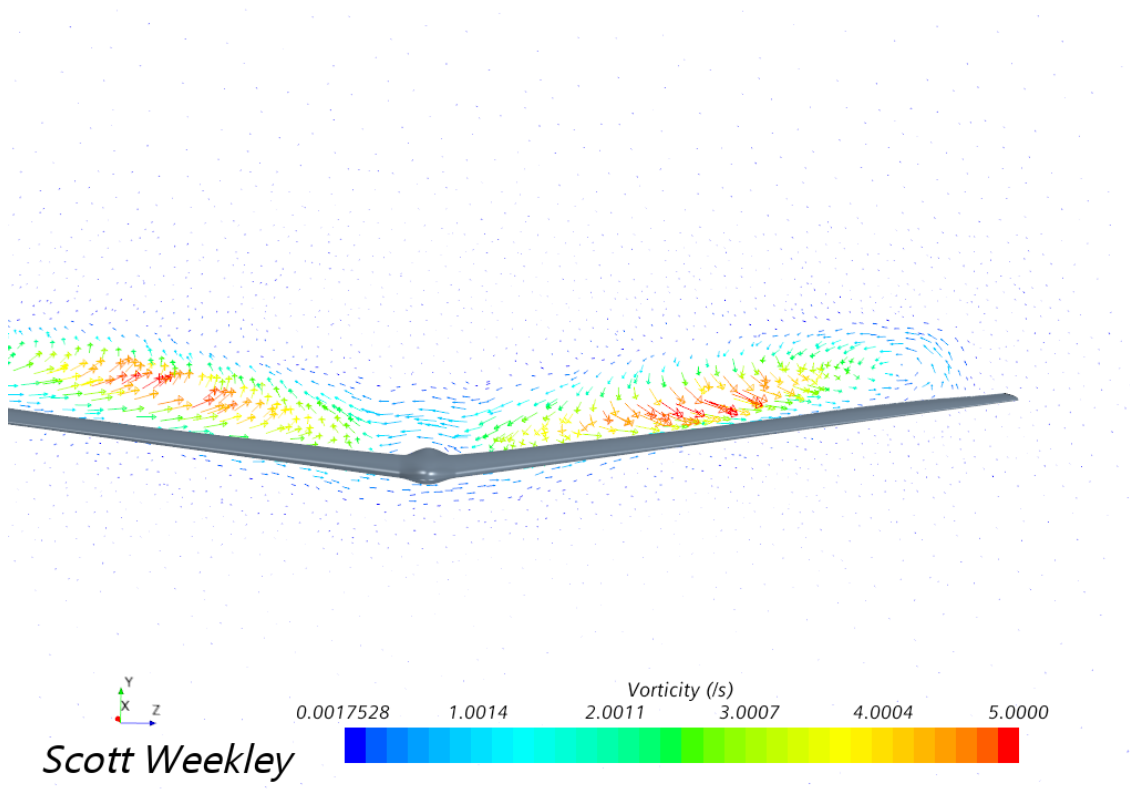


Figure 115: Modern Bell Vorticity 1

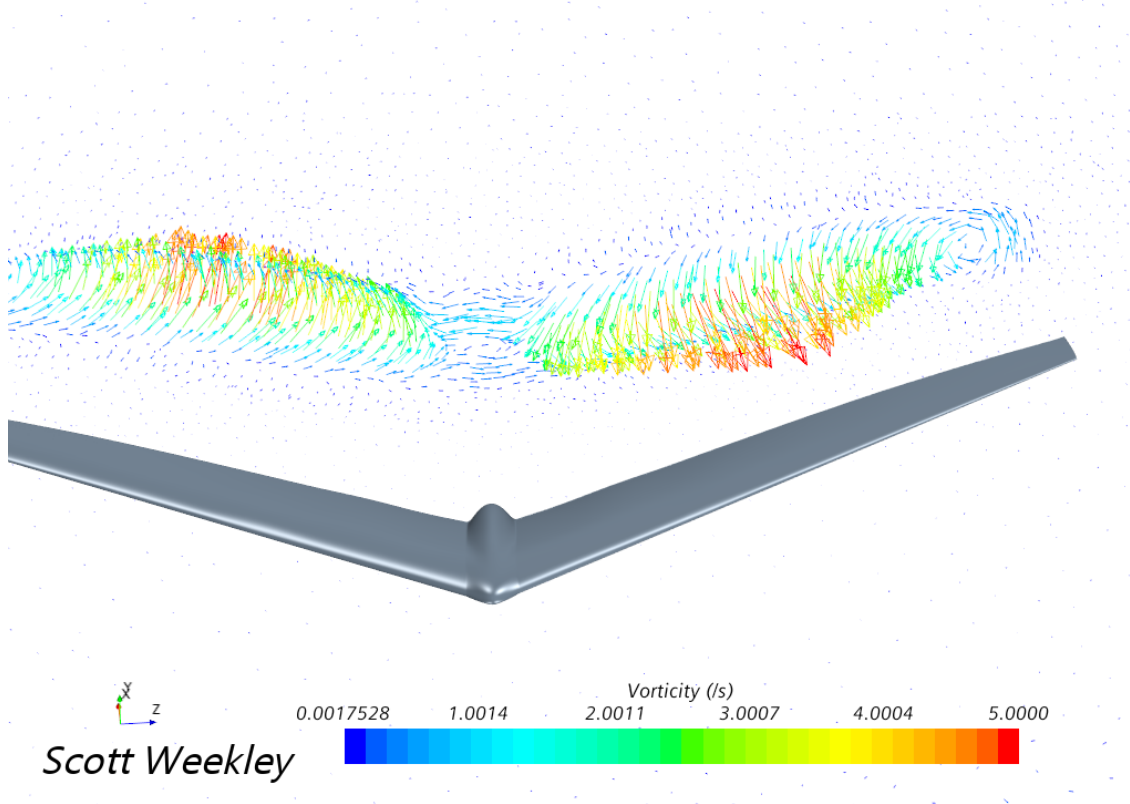


Figure 116: Modern Bell Vorticity 2

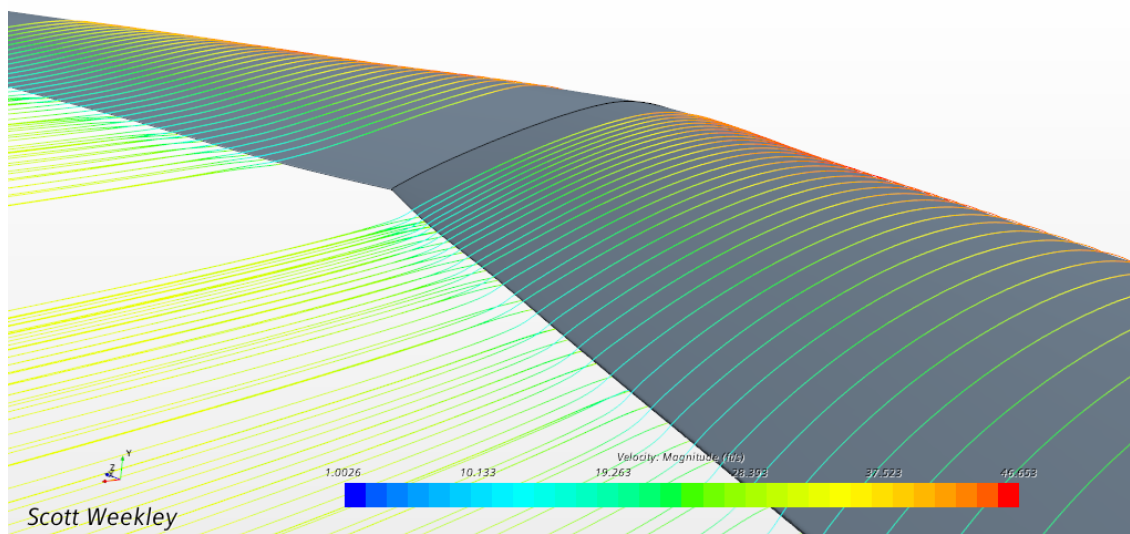


Figure 117: Bellwether Streamlines 1

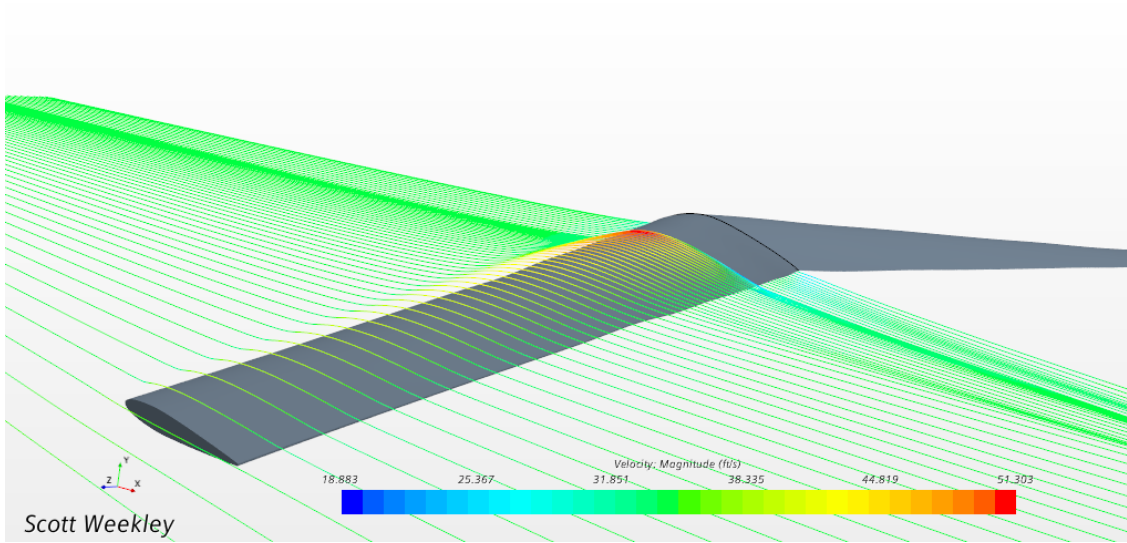


Figure 118: Bellwether Streamlines 2

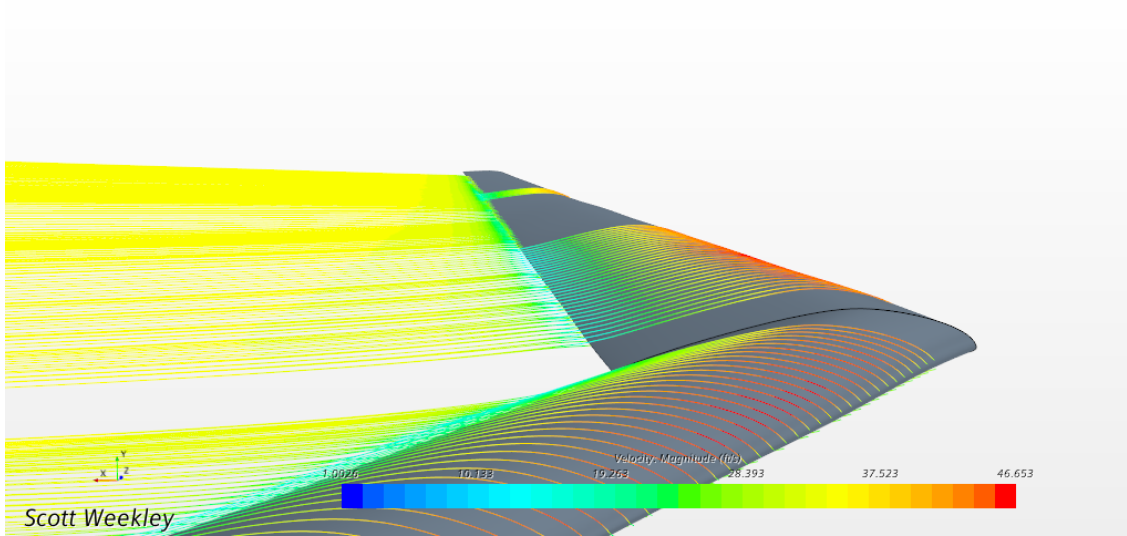


Figure 119: Bellwether Streamlines 3

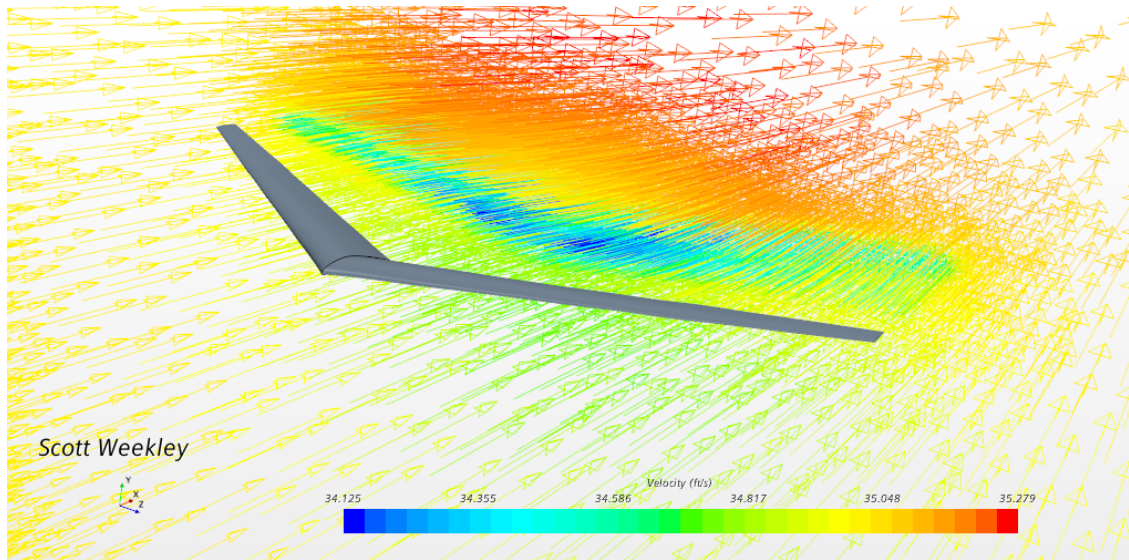


Figure 120: Bellwether Vector Scene

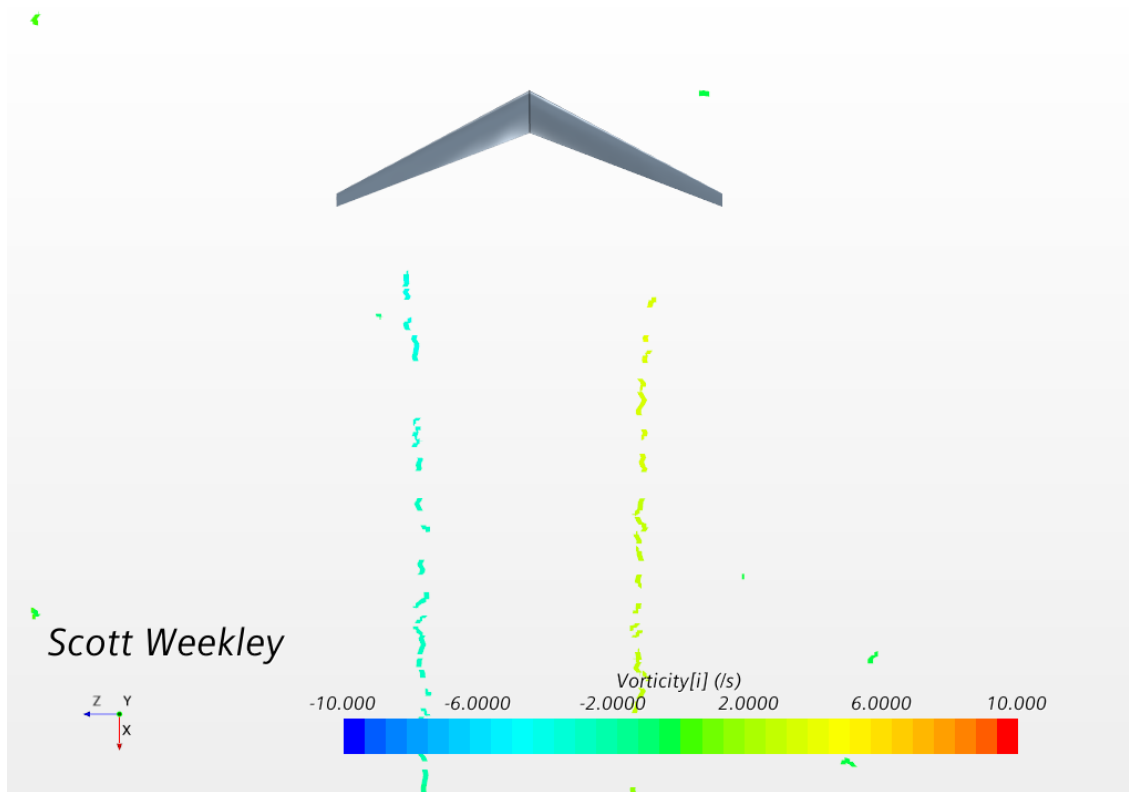


Figure 121: Bellwether Vortex Core

## 0.7 Flying Wing Macro

```

1 // Scott Weekley
2 // 6 August 2020
3 // Setup of Mesh, Physics, Reports, Plots, and Scenes for Aircraft
4 // Used for Design and Analysis of Flying-Wing, Bellwether
5
6
7 //----- Expected Steps Prior to Running
8 //User is to have imported the aircraft as a surface mesh, and
9 //named the surfaces "Flying Wing"
10 //Create a Block part with dimensions of the desired flow domain
11 //NOTE: It may be desired to separate parts of the aircraft for
12 //analysis of specific regions // (this is done in the
13 //analysis of Bellwether). This is accomplished by selecting the
14 // aircraft part surface, then right click to select "Split
15 //by Patch." This will require all // of the dependencies to
16 //reflect this change (all reports, scenes, etc.)
17 //Perform a Boolean-Subtract between the Block part and the Flying
18 //Wing with the Block as the // target part
19 //Rename the Subtract "Flow Domain"
20 //In the Flow Domain surfaces, right click on the Block Surface and
21 // "Split by Patch"
22 //Select the surface from which the flow will enter the domain and
23 // name it "Inlet"
24 //Name the opposite surface "Outlet"
25 //Close
26 //Rename the remaining Block Surface "Freestream"
27 //Assign the Flow Domain to Region and select "Create a Region for
28 // Each Part" and "Create a // Boundary for Each Part Surface"
29 //
30 //-----
31
32
33 //----- Important Modifications for your Design
34 //-----
35 //Depending on the size of your aircraft and domain, the mesh
36 //default controls should change:
37 //
38 //          Base Size
39 //          Target Surface Size
40 //          Maximum Cell Size
41 //The mesh custom controls "Flying Wing" should also change to meet
42 //the needs of your aircraft
43 //
44 //          Target Surface Size
45 //          Number of Prism Layers
46 //          Prism Layer Total Thickness
47 //          Growth Rate
48 //          Wake Refinement Options
49 //NOTE: If the aircraft surfaces are split by patch, all surfaces
50 //need to be added to this // surface control.
51 //In Regions, the surface "Freestream" should include a Mach number
52 //that agrees with the physics // conditions set for the
53 //flight speed
54 //Derived Part, "Longitudinal Tip", should be moved along the Z-
55 //axis to coincide with wing tip

```

```

38 //
39 -----
40
41
42 // STAR-CCM+ macro: FlyingWingMacro.java
43 // Written by STAR-CCM+ 14.04.013
44 package macro;
45
46 import java.util.*;
47
48 import star.common.*;
49 import star.base.neo.*;
50 import star.material.*;
51 import star.base.report.*;
52 import star.coupledflow.*;
53 import star.prismmesher.*;
54 import star.flow.*;
55 import star.vis.*;
56 import star.dualmesher.*;
57 import star.metrics.*;
58 import star.meshing.*;
59
60 public class FlyingWingMacro extends StarMacro {
61
62     public void execute() {
63         execute0();
64         execute1();
65     }
66
67     private void execute0() {
68
69         Simulation simulation_0 =
70             getActiveSimulation();
71
72         MeshPart meshPart_0 =
73             ((MeshPart) simulation_0.get(SimulationPartManager.class).
74             getPart("Flow Domain"));
75
76         AutoMeshOperation autoMeshOperation_0 =
77             simulation_0.get(MeshOperationManager.class).
78             createAutoMeshOperation(new StringVector(new String[] {"star.
79             resurfacer.ResurfacerAutoMesher", "star.dualmesher.
80             DualAutoMesher", "star.prismmesher.PrismAutoMesher"}), new
81             NeoObjectVector(new Object[] {meshPart_0}));
82
83         autoMeshOperation_0.setLinkOutputPartName(false);
84
85         simulation_0.getUnitsManager().getSystemOption().setSelected(
86             UnitsManagerSystemOption.Type.SYSTEM_USCS);
87
88         autoMeshOperation_0.getDefaultValues().get(BaseSize.class).
89             setValue(0.7);
90
91         PartsTargetSurfaceSize partsTargetSurfaceSize_0 =
92             autoMeshOperation_0.getDefaultValues().get(

```

```

86     PartsTargetSurfaceSize.class);
87     partsTargetSurfaceSize_0.getRelativeSizeScalar().setValue(2.0);
88
89     partsTargetSurfaceSize_0.getRelativeSizeScalar().setValue
90     (100.0);
91
92     PartsMinimumSurfaceSize partsMinimumSurfaceSize_0 =
93     autoMeshOperation_0.getDefaultValues().get(
94     PartsMinimumSurfaceSize.class);
95
96     partsMinimumSurfaceSize_0.getRelativeSizeScalar().setValue(2.0)
97     ;
98
99     DualAutoMesher dualAutoMesher_0 =
100     ((DualAutoMesher) autoMeshOperation_0.getMeshers().getObject(
101     "Polyhedral Mesher"));
102
103     dualAutoMesher_0.setEnableGrowthRate(true);
104
105     PrismAutoMesher prismAutoMesher_0 =
106     ((PrismAutoMesher) autoMeshOperation_0.getMeshers().getObject
107     ("Prism Layer Mesher"));
108
109     prismAutoMesher_0.getPrismStretchingOption().setSelected(
110     PrismStretchingOption.Type.WALL_THICKNESS);
111
112     MaximumCellSize maximumCellSize_0 =
113     autoMeshOperation_0.getDefaultValues().get(MaximumCellSize.
114     class);
115
116     maximumCellSize_0.getRelativeSizeScalar().setValue(1000.0);
117
118     SurfaceCustomMeshControl surfaceCustomMeshControl_0 =
119     autoMeshOperation_0.getCustomMeshControls().
120     createSurfaceControl();
121
122     autoMeshOperation_0.getMesherParallelModeOption().setSelected(
123     MesherParallelModeOption.Type.PARALLEL);
124
125     surfaceCustomMeshControl_0.getGeometryObjects().setQuery(null);
126
127     PartSurface partSurface_2 =
128     ((PartSurface) meshPart_0.getPartSurfaceManager().
129     getPartSurface("Flying Wing"));
130
131     surfaceCustomMeshControl_0.getGeometryObjects().setObjects(
132     partSurface_2);
133
134     surfaceCustomMeshControl_0.setPresentationName("Flying Wing");
135
136     surfaceCustomMeshControl_0.getCustomConditions().get(
137     PartsTargetSurfaceSizeOption.class).setSelected(
138     PartsTargetSurfaceSizeOption.Type.CUSTOM);
139
140     PartsCustomizePrismMesh partsCustomizePrismMesh_0 =
141     surfaceCustomMeshControl_0.getCustomConditions().get(

```



```

PartsCustomizePrismMesh.class);
129
130 partsCustomizePrismMesh_0.getCustomPrismOptions().setSelected(
PartsCustomPrismsOption.Type.CUSTOMIZE);
131
132 PartsRemesherTetPolyWakeRefinementOption
partsRemesherTetPolyWakeRefinementOption_0 =
133     surfaceCustomMeshControl_0.getCustomConditions().get(
PartsRemesherTetPolyWakeRefinementOption.class);
134
135 partsRemesherTetPolyWakeRefinementOption_0.
setPartsWakeRefinementOption(true);
136
137 PartsTargetSurfaceSize partsTargetSurfaceSize_1 =
138     surfaceCustomMeshControl_0.getCustomValues().get(
PartsTargetSurfaceSize.class);
139
140 partsTargetSurfaceSize_1.getRelativeSizeScalar().setValue(2.0);
141
142 PartsCustomizePrismMeshControls
partsCustomizePrismMeshControls_0 =
143     partsCustomizePrismMesh_0.getCustomPrismControls();
144
145 partsCustomizePrismMeshControls_0.setCustomizeNumLayers(true);
146
147 partsCustomizePrismMeshControls_0.setCustomizeTotalThickness(
true);
148
149 NumPrismLayers numPrismLayers_0 =
150     surfaceCustomMeshControl_0.getCustomValues().get(
CustomPrismValuesManager.class).get(NumPrismLayers.class);
151
152 IntegerValue integerValue_0 =
153     numPrismLayers_0.getNumLayersValue();
154
155 integerValue_0.getQuantity().setValue(6.0);
156
157 PrismThickness prismThickness_0 =
158     surfaceCustomMeshControl_0.getCustomValues().get(
CustomPrismValuesManager.class).get(PrismThickness.class);
159
160 prismThickness_0.getRelativeSizeScalar().setValue(5.0);
161
162 surfaceCustomMeshControl_0.getCustomValues().get(
PartsWakeRefinementValuesManager.class).getDistance().setValue
(20.0);
163
164 Units units_1 =
165     ((Units) simulation_0.getUnitsManager().getObject("deg"));
166
167 surfaceCustomMeshControl_0.getCustomValues().get(
PartsWakeRefinementValuesManager.class).getSpreadAngle().
setUnits(units_1);
168
169 surfaceCustomMeshControl_0.getCustomValues().get(
PartsWakeRefinementValuesManager.class).getSpreadAngle().
setValue(572.9577951308231);

```

```

170
171     surfaceCustomMeshControl_0.getCustomValues().get(
        PartsWakeRefinementValuesManager.class).getSpreadAngle().
        setValue(10.0);
172
173     PartsWakeRefinementSize partsWakeRefinementSize_0 =
174         surfaceCustomMeshControl_0.getCustomValues().get(
        PartsWakeRefinementValuesManager.class).getIsotropicSize();
175
176     partsWakeRefinementSize_0.getRelativeSizeScalar().setValue
        (50.0);
177
178     PartsRemesherTetPolyWakeRefinementSet
        partsRemesherTetPolyWakeRefinementSet_0 =
179         surfaceCustomMeshControl_0.getCustomValues().get(
        PartsWakeRefinementValuesManager.class).get(
        PartsRemesherTetPolyWakeRefinementSet.class);
180
181     partsRemesherTetPolyWakeRefinementSet_0.getGrowthRateQuantity()
        .setValue(1.5);
182
183     SurfaceCustomMeshControl surfaceCustomMeshControl_1 =
184         autoMeshOperation_0.getCustomMeshControls().
        createSurfaceControl();
185
186     surfaceCustomMeshControl_1.getGeometryObjects().setQuery(null);
187
188     PartSurface partSurface_1 =
189         ((PartSurface) meshPart_0.getPartSurfaceManager().
        getPartSurface("Freestream"));
190
191     PartSurface partSurface_3 =
192         ((PartSurface) meshPart_0.getPartSurfaceManager().
        getPartSurface("Inlet"));
193
194     PartSurface partSurface_4 =
195         ((PartSurface) meshPart_0.getPartSurfaceManager().
        getPartSurface("Outlet"));
196
197     surfaceCustomMeshControl_1.getGeometryObjects().setObjects(
        partSurface_1, partSurface_3, partSurface_4);
198
199     PartsCustomizePrismMesh partsCustomizePrismMesh_1 =
200         surfaceCustomMeshControl_1.getCustomConditions().get(
        PartsCustomizePrismMesh.class);
201
202     partsCustomizePrismMesh_1.getCustomPrismOptions().setSelected(
        PartsCustomPrismsOption.Type.DISABLE);
203
204     surfaceCustomMeshControl_1.setPresentationName("No Prism Layers
        ");
205
206     PhysicsContinuum physicsContinuum_0 =
207         simulation_0.getContinuumManager().createContinuum(
        PhysicsContinuum.class);
208
209     physicsContinuum_0.enable(ThreeDimensionalModel.class);

```

```

210
211 physicsContinuum_0.enable(ExplicitUnsteadyModel.class);
212
213 physicsContinuum_0.enable(SingleComponentGasModel.class);
214
215 physicsContinuum_0.enable(CoupledFlowModel.class);
216
217 physicsContinuum_0.enable(CoupledEnergyModel.class);
218
219 physicsContinuum_0.enable(IdealGasModel.class);
220
221 physicsContinuum_0.enable(LaminarModel.class);
222
223 LaminarModel laminarModel_0 =
224     physicsContinuum_0.getModelManager().getModel(LaminarModel.
.class);
225
226 physicsContinuum_0.disableModel(laminarModel_0);
227
228 IdealGasModel idealGasModel_0 =
229     physicsContinuum_0.getModelManager().getModel(IdealGasModel.
.class);
230
231 physicsContinuum_0.disableModel(idealGasModel_0);
232
233 CoupledEnergyModel coupledEnergyModel_0 =
234     physicsContinuum_0.getModelManager().getModel(
CoupledEnergyModel.class);
235
236 physicsContinuum_0.disableModel(coupledEnergyModel_0);
237
238 CoupledFlowModel coupledFlowModel_0 =
239     physicsContinuum_0.getModelManager().getModel(
CoupledFlowModel.class);
240
241 physicsContinuum_0.disableModel(coupledFlowModel_0);
242
243 physicsContinuum_0.enable(CellQualityRemediationModel.class);
244
245 physicsContinuum_0.enable(CoupledFlowModel.class);
246
247 physicsContinuum_0.enable(CoupledEnergyModel.class);
248
249 physicsContinuum_0.enable(IdealGasModel.class);
250
251 physicsContinuum_0.enable(LaminarModel.class);
252
253 VelocityProfile velocityProfile_0 =
254     physicsContinuum_0.getInitialConditions().get(VelocityProfile
.class);
255
256 velocityProfile_0.getMethod(ConstantVectorProfileMethod.class).
getQuantity().setComponents(35.0, 0.0, 0.0);
257
258 Region region_0 =
259     simulation_0.getRegionManager().getRegion("Flow Domain");
260

```

```

261 Boundary boundary_0 =
262     region_0.getBoundaryManager().getBoundary("Freestream");
263
264 FreeStreamBoundary freeStreamBoundary_0 =
265     ((FreeStreamBoundary) simulation_0.get(ConditionTypeManager.
class).get(FreeStreamBoundary.class));
266
267 boundary_0.setBoundaryType(freeStreamBoundary_0);
268
269 MachNumberProfile machNumberProfile_0 =
270     boundary_0.getValues().get(MachNumberProfile.class);
271
272 machNumberProfile_0.getMethod(ConstantScalarProfileMethod.class
).getQuantity().setValue(0.031);
273
274 Boundary boundary_1 =
275     region_0.getBoundaryManager().getBoundary("Inlet");
276
277 InletBoundary inletBoundary_0 =
278     ((InletBoundary) simulation_0.get(ConditionTypeManager.class)
.get(InletBoundary.class));
279
280 boundary_1.setBoundaryType(inletBoundary_0);
281
282 VelocityMagnitudeProfile velocityMagnitudeProfile_0 =
283     boundary_1.getValues().get(VelocityMagnitudeProfile.class);
284
285 velocityMagnitudeProfile_0.getMethod(
ConstantScalarProfileMethod.class).getQuantity().setValue(35.0)
;
286
287 Boundary boundary_2 =
288     region_0.getBoundaryManager().getBoundary("Outlet");
289
290 PressureBoundary pressureBoundary_0 =
291     ((PressureBoundary) simulation_0.get(ConditionTypeManager.
class).get(PressureBoundary.class));
292
293 boundary_2.setBoundaryType(pressureBoundary_0);
294
295 Units units_2 =
296     simulation_0.getUnitsManager().getPreferredUnits(new
IntVector(new int[] {0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0}));
297
298 Scene scene_1 =
299     simulation_0.getSceneManager().getScene("Geometry Scene 2");
300
301 scene_1.setTransparencyOverrideMode(SceneTransparencyOverride.
MAKE_SCENE_TRANSPARENT);
302
303 PartDisplayer partDisplayer_4 =
304     ((PartDisplayer) scene_1.getCreatorDisplayer());
305
306 partDisplayer_4.initialize();
307
308 scene_1.getCreatorGroup().setQuery(null);

```

```

309
310     scene_1.getCreatorGroup().setObjects(region_0);
311
312     scene_1.getCreatorGroup().setQuery(null);
313
314     scene_1.getCreatorGroup().setObjects(region_0);
315
316     PlaneSection planeSection_0 =
317         (PlaneSection) simulation_0.getPartManager().
            createImplicitPart(new NeoObjectVector(new Object[] {}), new
            DoubleVector(new double[] {0.0, 0.0, 1.0}), new DoubleVector(
            new double[] {0.0, 0.0, 0.0}), 0, 1, new DoubleVector(new
            double[] {0.0}));
318
319     LabCoordinateSystem labCoordinateSystem_0 =
320         simulation_0.getCoordinateSystemManager().
            getLabCoordinateSystem();
321
322     planeSection_0.setCoordinateSystem(labCoordinateSystem_0);
323
324     planeSection_0.getInputParts().setQuery(null);
325
326     planeSection_0.getInputParts().setObjects(region_0);
327
328     planeSection_0.getOriginCoordinate().setUnits0(units_2);
329
330     planeSection_0.getOriginCoordinate().setUnits1(units_2);
331
332     planeSection_0.getOriginCoordinate().setUnits2(units_2);
333
334     planeSection_0.getOriginCoordinate().setDefinition("");
335
336     planeSection_0.getOriginCoordinate().setValue(new DoubleVector(
            new double[] {0.0, 0.0, 0.0}));
337
338     planeSection_0.getOriginCoordinate().setCoordinate(units_2,
            units_2, units_2, new DoubleVector(new double[] {0.0, 0.0,
            0.0}));
339
340     planeSection_0.getOriginCoordinate().setCoordinateSystem(
            labCoordinateSystem_0);
341
342     planeSection_0.getOrientationCoordinate().setUnits0(units_2);
343
344     planeSection_0.getOrientationCoordinate().setUnits1(units_2);
345
346     planeSection_0.getOrientationCoordinate().setUnits2(units_2);
347
348     planeSection_0.getOrientationCoordinate().setDefinition("");
349
350     planeSection_0.getOrientationCoordinate().setValue(new
            DoubleVector(new double[] {0.0, 1.0, 0.0}));
351
352     planeSection_0.getOrientationCoordinate().setCoordinate(units_2,
            units_2, units_2, new DoubleVector(new double[] {0.0, 1.0,
            0.0}));
353

```

```

354     planeSection_0.getOrientationCoordinate().setCoordinateSystem(
labCoordinateSystem_0);
355
356     SingleValue singleValue_0 =
357         planeSection_0.getSingleValue();
358
359     singleValue_0.getValueQuantity().setValue(0.0);
360
361     singleValue_0.getValueQuantity().setUnits(units_2);
362
363     RangeMultiValue rangeMultiValue_0 =
364         planeSection_0.getRangeMultiValue();
365
366     rangeMultiValue_0.setNValues(2);
367
368     rangeMultiValue_0.getStartQuantity().setValue(0.0);
369
370     rangeMultiValue_0.getStartQuantity().setUnits(units_2);
371
372     rangeMultiValue_0.getEndQuantity().setValue(1.0);
373
374     rangeMultiValue_0.getEndQuantity().setUnits(units_2);
375
376     DeltaMultiValue deltaMultiValue_0 =
377         planeSection_0.getDeltaMultiValue();
378
379     deltaMultiValue_0.setNValues(2);
380
381     deltaMultiValue_0.getStartQuantity().setValue(0.0);
382
383     deltaMultiValue_0.getStartQuantity().setUnits(units_2);
384
385     deltaMultiValue_0.getDeltaQuantity().setValue(1.0);
386
387     deltaMultiValue_0.getDeltaQuantity().setUnits(units_2);
388
389     MultiValue multiValue_0 =
390         planeSection_0.getArbitraryMultiValue();
391
392     multiValue_0.getValueQuantities().setUnits(units_2);
393
394     multiValue_0.getValueQuantities().setArray(new DoubleVector(new
double[] {0.0}));
395
396     planeSection_0.setValueMode(ValueMode.SINGLE);
397
398     scene_1.setTransparencyOverrideMode(SceneTransparencyOverride.
USE_DISPLAYER_PROPERTY);
399
400     planeSection_0.setPresentationName("Horizontal");
401
402     scene_1.setTransparencyOverrideMode(SceneTransparencyOverride.
MAKE_SCENE_TRANSPARENT);
403
404     scene_1.getCreatorGroup().setQuery(null);
405
406     scene_1.getCreatorGroup().setObjects(region_0);

```

```

407
408     scene_1.getCreatorGroup().setQuery(null);
409
410     scene_1.getCreatorGroup().setObjects(region_0);
411
412     PlaneSection planeSection_1 =
413         (PlaneSection) simulation_0.getPartManager().
createImplicitPart(new NeoObjectVector(new Object[] {}), new
DoubleVector(new double[] {0.0, 0.0, 1.0}), new DoubleVector(
new double[] {0.0, 0.0, 0.0}), 0, 1, new DoubleVector(new
double[] {0.0}));
414
415     planeSection_1.setCoordinateSystem(labCoordinateSystem_0);
416
417     planeSection_1.getInputParts().setQuery(null);
418
419     planeSection_1.getInputParts().setObjects(region_0);
420
421     planeSection_1.getOriginCoordinate().setUnits0(units_2);
422
423     planeSection_1.getOriginCoordinate().setUnits1(units_2);
424
425     planeSection_1.getOriginCoordinate().setUnits2(units_2);
426
427     planeSection_1.getOriginCoordinate().setDefinition("");
428
429     planeSection_1.getOriginCoordinate().setValue(new DoubleVector(
new double[] {0.0, 0.0, 0.0}));
430
431     planeSection_1.getOriginCoordinate().setCoordinate(units_2,
units_2, units_2, new DoubleVector(new double[] {0.0, 0.0,
0.0}));
432
433     planeSection_1.getOriginCoordinate().setCoordinateSystem(
labCoordinateSystem_0);
434
435     planeSection_1.getOrientationCoordinate().setUnits0(units_2);
436
437     planeSection_1.getOrientationCoordinate().setUnits1(units_2);
438
439     planeSection_1.getOrientationCoordinate().setUnits2(units_2);
440
441     planeSection_1.getOrientationCoordinate().setDefinition("");
442
443     planeSection_1.getOrientationCoordinate().setValue(new
DoubleVector(new double[] {0.0, 0.0, 1.0}));
444
445     planeSection_1.getOrientationCoordinate().setCoordinate(units_2
, units_2, units_2, new DoubleVector(new double[] {0.0, 0.0,
1.0}));
446
447     planeSection_1.getOrientationCoordinate().setCoordinateSystem(
labCoordinateSystem_0);
448
449     SingleValue singleValue_1 =
450         planeSection_1.getSingleValue();
451

```

```

452     singleValue_1.getValueQuantity().setValue(0.0);
453
454     singleValue_1.getValueQuantity().setUnits(units_2);
455
456     RangeMultiValue rangeMultiValue_1 =
457         planeSection_1.getRangeMultiValue();
458
459     rangeMultiValue_1.setNValues(2);
460
461     rangeMultiValue_1.getStartQuantity().setValue(0.0);
462
463     rangeMultiValue_1.getStartQuantity().setUnits(units_2);
464
465     rangeMultiValue_1.getEndQuantity().setValue(1.0);
466
467     rangeMultiValue_1.getEndQuantity().setUnits(units_2);
468
469     DeltaMultiValue deltaMultiValue_1 =
470         planeSection_1.getDeltaMultiValue();
471
472     deltaMultiValue_1.setNValues(2);
473
474     deltaMultiValue_1.getStartQuantity().setValue(0.0);
475
476     deltaMultiValue_1.getStartQuantity().setUnits(units_2);
477
478     deltaMultiValue_1.getDeltaQuantity().setValue(1.0);
479
480     deltaMultiValue_1.getDeltaQuantity().setUnits(units_2);
481
482     MultiValue multiValue_1 =
483         planeSection_1.getArbitraryMultiValue();
484
485     multiValue_1.getValueQuantities().setUnits(units_2);
486
487     multiValue_1.getValueQuantities().setArray(new DoubleVector(new
488         double[] {0.0}));
489
490     planeSection_1.setValueMode(ValueMode.SINGLE);
491
492     scene_1.setTransparencyOverrideMode(SceneTransparencyOverride.
493     USE_DISPLAYER_PROPERTY);
494
495     planeSection_1.setPresentationName("Longitudinal - Root");
496
497     scene_1.setTransparencyOverrideMode(SceneTransparencyOverride.
498     MAKE_SCENE_TRANSPARENT);
499
500     scene_1.getCreatorGroup().setQuery(null);
501
502     scene_1.getCreatorGroup().setObjects(region_0);
503
504     scene_1.getCreatorGroup().setQuery(null);
505
506     scene_1.getCreatorGroup().setObjects(region_0);
507
508     PlaneSection planeSection_2 =

```



```

506     (PlaneSection) simulation_0.getPartManager().
        createImplicitPart(new NeoObjectVector(new Object[] {}), new
        DoubleVector(new double[] {0.0, 0.0, 1.0}), new DoubleVector(
        new double[] {0.0, 0.0, 0.0}), 0, 1, new DoubleVector(new
        double[] {0.0}));
507
508     planeSection_2.setCoordinateSystem(labCoordinateSystem_0);
509
510     planeSection_2.getInputParts().setQuery(null);
511
512     planeSection_2.getInputParts().setObjects(region_0);
513
514     planeSection_2.getOriginCoordinate().setUnits0(units_2);
515
516     planeSection_2.getOriginCoordinate().setUnits1(units_2);
517
518     planeSection_2.getOriginCoordinate().setUnits2(units_2);
519
520     planeSection_2.getOriginCoordinate().setDefinition("");
521
522     planeSection_2.getOriginCoordinate().setValue(new DoubleVector(
        new double[] {0.0, 0.0, 12.5}));
523
524     planeSection_2.getOriginCoordinate().setCoordinate(units_2,
        units_2, units_2, new DoubleVector(new double[] {0.0, 0.0,
        12.5}));
525
526     planeSection_2.getOriginCoordinate().setCoordinateSystem(
        labCoordinateSystem_0);
527
528     planeSection_2.getOrientationCoordinate().setUnits0(units_2);
529
530     planeSection_2.getOrientationCoordinate().setUnits1(units_2);
531
532     planeSection_2.getOrientationCoordinate().setUnits2(units_2);
533
534     planeSection_2.getOrientationCoordinate().setDefinition("");
535
536     planeSection_2.getOrientationCoordinate().setValue(new
        DoubleVector(new double[] {0.0, 0.0, 1.0}));
537
538     planeSection_2.getOrientationCoordinate().setCoordinate(units_2
        , units_2, units_2, new DoubleVector(new double[] {0.0, 0.0,
        1.0}));
539
540     planeSection_2.getOrientationCoordinate().setCoordinateSystem(
        labCoordinateSystem_0);
541
542     SingleValue singleValue_2 =
543         planeSection_2.getSingleValue();
544
545     singleValue_2.getValueQuantity().setValue(0.0);
546
547     singleValue_2.getValueQuantity().setUnits(units_2);
548
549     RangeMultiValue rangeMultiValue_2 =
550         planeSection_2.getRangeMultiValue();

```

```

551     rangeMultiValue_2.setNValues(2);
552
553     rangeMultiValue_2.getStartQuantity().setValue(0.0);
554
555     rangeMultiValue_2.getStartQuantity().setUnits(units_2);
556
557     rangeMultiValue_2.getEndQuantity().setValue(1.0);
558
559     rangeMultiValue_2.getEndQuantity().setUnits(units_2);
560
561     DeltaMultiValue deltaMultiValue_2 =
562         planeSection_2.getDeltaMultiValue();
563
564     deltaMultiValue_2.setNValues(2);
565
566     deltaMultiValue_2.getStartQuantity().setValue(0.0);
567
568     deltaMultiValue_2.getStartQuantity().setUnits(units_2);
569
570     deltaMultiValue_2.getDeltaQuantity().setValue(1.0);
571
572     deltaMultiValue_2.getDeltaQuantity().setUnits(units_2);
573
574     MultiValue multiValue_2 =
575         planeSection_2.getArbitraryMultiValue();
576
577     multiValue_2.getValueQuantities().setUnits(units_2);
578
579     multiValue_2.getValueQuantities().setArray(new DoubleVector(new
580         double[] {0.0}));
581
582     planeSection_2.setValueMode(ValueMode.SINGLE);
583
584     scene_1.setTransparencyOverrideMode(SceneTransparencyOverride.
585         USE_DISPLAYER_PROPERTY);
586
587     planeSection_2.setPresentationName("Longitudinal - Tip");
588
589     ForceCoefficientReport forceCoefficientReport_0 =
590         simulation_0.getReportManager().createReport(
591             ForceCoefficientReport.class);
592
593     forceCoefficientReport_0.setPresentationName("Coefficient of
594         Drag");
595
596     forceCoefficientReport_0.getReferenceVelocity().setValue(35.0);
597
598     forceCoefficientReport_0.getParts().setQuery(null);
599
600     Boundary boundary_3 =
601         region_0.getBoundaryManager().getBoundary("Flying Wing");
602
603     forceCoefficientReport_0.getParts().setObjects(boundary_3);
604
605     forceCoefficientReport_0.getReferenceArea().setValue(45.0);
606
607

```

```

604 ForceCoefficientReport forceCoefficientReport_1 =
605     simulation_0.getReportManager().createReport(
ForceCoefficientReport.class);
606
607 forceCoefficientReport_1.setPresentationName("Copy of
Coefficient of Drag");
608
609 forceCoefficientReport_1.copyProperties(
forceCoefficientReport_0);
610
611 forceCoefficientReport_1.setPresentationName("Coefficient of
Lift");
612
613 forceCoefficientReport_1.getDirection().setComponents(1.0, 1.0,
0.0);
614
615 forceCoefficientReport_1.getDirection().setComponents(0.0, 1.0,
0.0);
616
617 ForceReport forceReport_0 =
618     simulation_0.getReportManager().createReport(ForceReport.
class);
619
620 forceReport_0.setPresentationName("Drag");
621
622 forceReport_0.getParts().setQuery(null);
623
624 forceReport_0.getParts().setObjects(boundary_3);
625
626 ForceReport forceReport_1 =
627     simulation_0.getReportManager().createReport(ForceReport.
class);
628
629 forceReport_1.setPresentationName("Copy of Drag");
630
631 forceReport_1.copyProperties(forceReport_0);
632 }
633
634 private void execute1() {
635
636     Simulation simulation_0 =
637         getActiveSimulation();
638
639     ForceReport forceReport_1 =
640         ((ForceReport) simulation_0.getReportManager().getReport("
Copy of Drag"));
641
642     forceReport_1.setPresentationName("Lift");
643
644     forceReport_1.getDirection().setComponents(1.0, 1.0, 0.0);
645
646     forceReport_1.getDirection().setComponents(0.0, 1.0, 0.0);
647
648     MomentReport momentReport_0 =
649         simulation_0.getReportManager().createReport(MomentReport.
class);
650

```

```

651     momentReport_0.setPresentationName("Moment - Roll");
652
653     momentReport_0.getDirection().setComponents(-1.0, 0.0, 0.0);
654
655     momentReport_0.getParts().setQuery(null);
656
657     Region region_0 =
658         simulation_0.getRegionManager().getRegion("Flow Domain");
659
660     Boundary boundary_3 =
661         region_0.getBoundaryManager().getBoundary("Flying Wing");
662
663     momentReport_0.getParts().setObjects(boundary_3);
664
665     MomentReport momentReport_1 =
666         simulation_0.getReportManager().createReport(MomentReport.
667             class);
668
669     momentReport_1.setPresentationName("Copy of Moment - Roll");
670
671     momentReport_1.copyProperties(momentReport_0);
672
673     momentReport_1.setPresentationName("Moment - Pitch");
674
675     momentReport_1.getDirection().setComponents(-1.0, 0.0, -1.0);
676
677     momentReport_1.getDirection().setComponents(0.0, 0.0, -1.0);
678
679     MomentReport momentReport_2 =
680         simulation_0.getReportManager().createReport(MomentReport.
681             class);
682
683     momentReport_2.setPresentationName("Copy of Moment - Roll");
684
685     momentReport_2.copyProperties(momentReport_0);
686
687     momentReport_2.setPresentationName("Moment - Yaw");
688
689     momentReport_2.getDirection().setComponents(-1.0, -1.0, 0.0);
690
691     momentReport_2.getDirection().setComponents(0.0, -1.0, 0.0);
692
693     ForceCoefficientReport forceCoefficientReport_0 =
694         ((ForceCoefficientReport) simulation_0.getReportManager().
695             getReport("Coefficient of Drag"));
696
697     ForceCoefficientReport forceCoefficientReport_1 =
698         ((ForceCoefficientReport) simulation_0.getReportManager().
699             getReport("Coefficient of Lift"));
700
701     simulation_0.getMonitorManager().createMonitorAndPlot(new
702         NeoObjectVector(new Object[] {forceCoefficientReport_0,
703             forceCoefficientReport_1}), true, "Reports Plot");
704
705     ReportMonitor reportMonitor_0 =
706         ((ReportMonitor) simulation_0.getMonitorManager().getMonitor(
707             "Coefficient of Drag Monitor"));

```

```

701
702 ReportMonitor reportMonitor_1 =
703     ((ReportMonitor) simulation_0.getMonitorManager().getMonitor(
    "Coefficient of Lift Monitor"));
704
705 MonitorPlot monitorPlot_0 =
706     simulation_0.getPlotManager().createMonitorPlot(new
    NeoObjectVector(new Object[] {reportMonitor_0, reportMonitor_1
    }), "Reports Plot");
707
708 monitorPlot_0.open();
709
710 Scene scene_1 =
711     simulation_0.getSceneManager().getScene("Geometry Scene 2");
712
713 SceneUpdate sceneUpdate_1 =
714     scene_1.getSceneUpdate();
715
716 HardcopyProperties hardcopyProperties_1 =
717     sceneUpdate_1.getHardcopyProperties();
718
719 hardcopyProperties_1.setCurrentResolutionWidth(1018);
720
721 hardcopyProperties_1.setCurrentResolutionHeight(735);
722
723 PlotUpdate plotUpdate_0 =
724     monitorPlot_0.getPlotUpdate();
725
726 HardcopyProperties hardcopyProperties_2 =
727     plotUpdate_0.getHardcopyProperties();
728
729 hardcopyProperties_2.setCurrentResolutionWidth(1016);
730
731 hardcopyProperties_2.setCurrentResolutionHeight(734);
732
733 monitorPlot_0.setPresentationName("Coefficients");
734
735 ForceReport forceReport_0 =
736     ((ForceReport) simulation_0.getReportManager().getReport("
    Drag"));
737
738 simulation_0.getMonitorManager().createMonitorAndPlot(new
    NeoObjectVector(new Object[] {forceReport_0, forceReport_1}),
    true, "Reports Plot");
739
740 ReportMonitor reportMonitor_2 =
741     ((ReportMonitor) simulation_0.getMonitorManager().getMonitor(
    "Drag Monitor"));
742
743 ReportMonitor reportMonitor_3 =
744     ((ReportMonitor) simulation_0.getMonitorManager().getMonitor(
    "Lift Monitor"));
745
746 MonitorPlot monitorPlot_1 =
747     simulation_0.getPlotManager().createMonitorPlot(new
    NeoObjectVector(new Object[] {reportMonitor_2, reportMonitor_3
    }), "Reports Plot");

```

```

748
749     monitorPlot_1.open();
750
751     hardcopyProperties_2.setCurrentResolutionWidth(1018);
752
753     hardcopyProperties_2.setCurrentResolutionHeight(735);
754
755     PlotUpdate plotUpdate_1 =
756         monitorPlot_1.getPlotUpdate();
757
758     HardcopyProperties hardcopyProperties_3 =
759         plotUpdate_1.getHardcopyProperties();
760
761     hardcopyProperties_3.setCurrentResolutionWidth(1016);
762
763     hardcopyProperties_3.setCurrentResolutionHeight(734);
764
765     monitorPlot_1.setPresentationName("Forces");
766
767     simulation_0.getMonitorManager().createMonitorAndPlot(new
768     NeoObjectVector(new Object[] {momentReport_1, momentReport_0,
769     momentReport_2}), true, "Reports Plot");
770
771     ReportMonitor reportMonitor_4 =
772         ((ReportMonitor) simulation_0.getMonitorManager().getMonitor(
773     "Moment - Pitch Monitor"));
774
775     ReportMonitor reportMonitor_5 =
776         ((ReportMonitor) simulation_0.getMonitorManager().getMonitor(
777     "Moment - Roll Monitor"));
778
779     ReportMonitor reportMonitor_6 =
780         ((ReportMonitor) simulation_0.getMonitorManager().getMonitor(
781     "Moment - Yaw Monitor"));
782
783     MonitorPlot monitorPlot_2 =
784         simulation_0.getPlotManager().createMonitorPlot(new
785     NeoObjectVector(new Object[] {reportMonitor_4, reportMonitor_5,
786     reportMonitor_6}), "Reports Plot");
787
788     monitorPlot_2.open();
789
790     hardcopyProperties_3.setCurrentResolutionWidth(1018);
791
792     hardcopyProperties_3.setCurrentResolutionHeight(735);
793
794     PlotUpdate plotUpdate_2 =
795         monitorPlot_2.getPlotUpdate();
796
797     HardcopyProperties hardcopyProperties_4 =
798         plotUpdate_2.getHardcopyProperties();
799
800     hardcopyProperties_4.setCurrentResolutionWidth(1016);
801
802     hardcopyProperties_4.setCurrentResolutionHeight(734);
803
804     monitorPlot_2.setPresentationName("Moments");

```

```

798
799     AccumulatedForceTable accumulatedForceTable_0 =
800         simulation_0.getTableManager().createTable(
            AccumulatedForceTable.class);
801
802     accumulatedForceTable_0.setPresentationName("Drag Distribution"
        );
803
804     accumulatedForceTable_0.getParts().setQuery(null);
805
806     accumulatedForceTable_0.getParts().setObjects(boundary_3);
807
808     AccumulatedForceHistogram accumulatedForceHistogram_0 =
809         ((AccumulatedForceHistogram) accumulatedForceTable_0.
            getHistogram());
810
811     accumulatedForceHistogram_0.getBinDirection().setComponents
        (1.0, 0.0, -1.0);
812
813     accumulatedForceHistogram_0.getBinDirection().setComponents
        (0.0, 0.0, -1.0);
814
815     accumulatedForceHistogram_0.setNumberOfBin(150);
816
817     AccumulatedForceTable accumulatedForceTable_1 =
818         simulation_0.getTableManager().createTable(
            AccumulatedForceTable.class);
819
820     accumulatedForceTable_1.setPresentationName("Lift Distribution"
        );
821
822     accumulatedForceTable_1.getParts().setQuery(null);
823
824     accumulatedForceTable_1.getParts().setObjects(boundary_3);
825
826     AccumulatedForceHistogram accumulatedForceHistogram_1 =
827         ((AccumulatedForceHistogram) accumulatedForceTable_1.
            getHistogram());
828
829     accumulatedForceHistogram_1.setNumberOfBin(150);
830
831     accumulatedForceHistogram_1.getBinDirection().setComponents
        (1.0, 0.0, -1.0);
832
833     accumulatedForceHistogram_1.getBinDirection().setComponents
        (0.0, 0.0, -1.0);
834
835     accumulatedForceHistogram_1.getForceDirection().setComponents
        (1.0, 1.0, 0.0);
836
837     accumulatedForceHistogram_1.getForceDirection().setComponents
        (0.0, 1.0, 0.0);
838
839     XYPlot xYPlot_0 =
840         simulation_0.getPlotManager().createPlot(XYPlot.class);
841
842     xYPlot_0.open();

```

```

843
844     hardcopyProperties_4.setCurrentResolutionWidth(1018);
845
846     hardcopyProperties_4.setCurrentResolutionHeight(735);
847
848     PlotUpdate plotUpdate_3 =
849         xYPlot_0.getPlotUpdate();
850
851     HardcopyProperties hardcopyProperties_5 =
852         plotUpdate_3.getHardcopyProperties();
853
854     hardcopyProperties_5.setCurrentResolutionWidth(1016);
855
856     hardcopyProperties_5.setCurrentResolutionHeight(734);
857
858     xYPlot_0.setPresentationName("Lift Distribution");
859
860     xYPlot_0.getDataSetManager().addDataProviders(new
NeoObjectVector(new Object[] {accumulatedForceTable_1}));
861
862     xYPlot_0.getParts().setQuery(null);
863
864     xYPlot_0.getParts().setObjects(boundary_3);
865
866     XYPlot xYPlot_1 =
867         simulation_0.getPlotManager().createPlot(XYPlot.class);
868
869     xYPlot_1.open();
870
871     hardcopyProperties_5.setCurrentResolutionWidth(1018);
872
873     hardcopyProperties_5.setCurrentResolutionHeight(735);
874
875     PlotUpdate plotUpdate_4 =
876         xYPlot_1.getPlotUpdate();
877
878     HardcopyProperties hardcopyProperties_6 =
879         plotUpdate_4.getHardcopyProperties();
880
881     hardcopyProperties_6.setCurrentResolutionWidth(1016);
882
883     hardcopyProperties_6.setCurrentResolutionHeight(734);
884
885     xYPlot_1.setPresentationName("Drag Distribution");
886
887     xYPlot_1.getParts().setQuery(null);
888
889     xYPlot_1.getParts().setObjects(boundary_3);
890
891     xYPlot_1.getDataSetManager().addDataProviders(new
NeoObjectVector(new Object[] {accumulatedForceTable_0}));
892
893     simulation_0.getSceneManager().createGeometryScene("Mesh Scene "
, "Outline", "Mesh", 3);
894
895     Scene scene_2 =
896         simulation_0.getSceneManager().getScene("Mesh Scene 1");

```



```

897
898     scene_2.initializeAndWait();
899
900     PartDisplayer partDisplayer_5 =
901         ((PartDisplayer) scene_2.getDisplayerManager().getDisplayer("
Mesh 1"));
902
903     partDisplayer_5.initialize();
904
905     SceneUpdate sceneUpdate_2 =
906         scene_2.getSceneUpdate();
907
908     HardcopyProperties hardcopyProperties_7 =
909         sceneUpdate_2.getHardcopyProperties();
910
911     hardcopyProperties_7.setCurrentResolutionWidth(25);
912
913     hardcopyProperties_7.setCurrentResolutionHeight(25);
914
915     hardcopyProperties_6.setCurrentResolutionWidth(1018);
916
917     hardcopyProperties_6.setCurrentResolutionHeight(735);
918
919     hardcopyProperties_7.setCurrentResolutionWidth(1016);
920
921     hardcopyProperties_7.setCurrentResolutionHeight(734);
922
923     scene_2.resetCamera();
924
925     scene_2.setPresentationName("Mesh - Horizontal");
926
927     partDisplayer_5.getInputParts().setQuery(null);
928
929     PlaneSection planeSection_0 =
930         ((PlaneSection) simulation_0.getPartManager().getObject("
Horizontal"));
931
932     partDisplayer_5.getInputParts().setObjects(planeSection_0);
933
934     simulation_0.getSceneManager().createGeometryScene("Mesh Scene "
, "Outline", "Mesh", 3);
935
936     Scene scene_3 =
937         simulation_0.getSceneManager().getScene("Mesh Scene 1");
938
939     scene_3.initializeAndWait();
940
941     PartDisplayer partDisplayer_6 =
942         ((PartDisplayer) scene_3.getDisplayerManager().getDisplayer("
Mesh 1"));
943
944     partDisplayer_6.initialize();
945
946     SceneUpdate sceneUpdate_3 =
947         scene_3.getSceneUpdate();
948
949     HardcopyProperties hardcopyProperties_8 =

```

```

950     sceneUpdate_3.getHardcopyProperties();
951
952     hardcopyProperties_8.setCurrentResolutionWidth(25);
953
954     hardcopyProperties_8.setCurrentResolutionHeight(25);
955
956     hardcopyProperties_7.setCurrentResolutionWidth(1018);
957
958     hardcopyProperties_7.setCurrentResolutionHeight(735);
959
960     hardcopyProperties_8.setCurrentResolutionWidth(1016);
961
962     hardcopyProperties_8.setCurrentResolutionHeight(734);
963
964     scene_3.resetCamera();
965
966     scene_3.setPresentationName("Mesh - Longitudinal - Root");
967
968     partDisplayer_6.getInputParts().setQuery(null);
969
970     partDisplayer_6.getInputParts().setObjects(boundary_3);
971
972     Scene scene_4 =
973         simulation_0.getSceneManager().createScene("Copy of Mesh -
          Longitudinal - Root");
974
975     scene_4.setPresentationName("Copy of Mesh - Longitudinal - Root
          ");
976
977     scene_4.copyProperties(scene_3);
978
979     scene_4.initializeAndWait();
980
981     PartDisplayer partDisplayer_7 =
982         ((PartDisplayer) scene_4.getDisplayerManager().getDisplayer("
          Mesh 1"));
983
984     partDisplayer_7.initialize();
985
986     scene_4.open();
987
988     SceneUpdate sceneUpdate_4 =
989         scene_4.getSceneUpdate();
990
991     HardcopyProperties hardcopyProperties_9 =
992         sceneUpdate_4.getHardcopyProperties();
993
994     hardcopyProperties_9.setCurrentResolutionWidth(25);
995
996     hardcopyProperties_9.setCurrentResolutionHeight(25);
997
998     hardcopyProperties_8.setCurrentResolutionWidth(1018);
999
1000    hardcopyProperties_8.setCurrentResolutionHeight(735);
1001
1002    hardcopyProperties_9.setCurrentResolutionWidth(1016);
1003

```

```

1004     hardcopyProperties_9.setCurrentResolutionHeight(734);
1005
1006     scene_4.setPresentationName("Mesh - Longitudinal - Tip");
1007
1008     partDisplayer_7.getInputParts().setQuery(null);
1009
1010     PlaneSection planeSection_2 =
1011         ((PlaneSection) simulation_0.getPartManager().getObject("
Longitudinal - Tip"));
1012
1013     partDisplayer_7.getInputParts().setObjects(planeSection_2);
1014
1015     partDisplayer_6.getInputParts().setQuery(null);
1016
1017     PlaneSection planeSection_1 =
1018         ((PlaneSection) simulation_0.getPartManager().getObject("
Longitudinal - Root"));
1019
1020     partDisplayer_6.getInputParts().setObjects(planeSection_1);
1021
1022     simulation_0.getSceneManager().createScalarScene("Scalar Scene"
, "Outline", "Scalar");
1023
1024     Scene scene_5 =
1025         simulation_0.getSceneManager().getScene("Scalar Scene 1");
1026
1027     scene_5.initializeAndWait();
1028
1029     PartDisplayer partDisplayer_8 =
1030         ((PartDisplayer) scene_5.getDisplayerManager().getDisplayer("
Outline 1"));
1031
1032     partDisplayer_8.initialize();
1033
1034     ScalarDisplayer scalarDisplayer_0 =
1035         ((ScalarDisplayer) scene_5.getDisplayerManager().getDisplayer
("Scalar 1"));
1036
1037     scalarDisplayer_0.initialize();
1038
1039     Legend legend_0 =
1040         scalarDisplayer_0.getLegend();
1041
1042     BlueRedLookupTable blueRedLookupTable_0 =
1043         ((BlueRedLookupTable) simulation_0.get(LookupTableManager.
class).getObject("blue-red"));
1044
1045     legend_0.setLookupTable(blueRedLookupTable_0);
1046
1047     SceneUpdate sceneUpdate_5 =
1048         scene_5.getSceneUpdate();
1049
1050     HardcopyProperties hardcopyProperties_10 =
1051         sceneUpdate_5.getHardcopyProperties();
1052
1053     hardcopyProperties_10.setCurrentResolutionWidth(25);
1054

```

```

1055     hardcopyProperties_10.setCurrentResolutionHeight(25);
1056
1057     hardcopyProperties_9.setCurrentResolutionWidth(1018);
1058
1059     hardcopyProperties_9.setCurrentResolutionHeight(735);
1060
1061     hardcopyProperties_10.setCurrentResolutionWidth(1016);
1062
1063     hardcopyProperties_10.setCurrentResolutionHeight(734);
1064
1065     scene_5.resetCamera();
1066
1067     scene_5.setPresentationName("Wall Y+");
1068
1069     scalarDisplayer_0.getInputParts().setQuery(null);
1070
1071     scalarDisplayer_0.getInputParts().setObjects(boundary_3);
1072
1073     Scene scene_0 =
1074         simulation_0.getSceneManager().getScene("Geometry Scene 1");
1075
1076     scene_0.setPresentationName("Flying Wing");
1077
1078     PartDisplayer partDisplayer_0 =
1079         ((PartDisplayer) scene_0.getDisplayerManager().getDisplayer("
Outline 1"));
1080
1081     partDisplayer_0.getInputParts().setQuery(null);
1082
1083     partDisplayer_0.getInputParts().setObjects(boundary_3);
1084 }
1085 }

```

VITA

Scott Weekley

Candidate for the Degree of  
Masters of Science

Thesis: DESIGN AND ANALYSIS OF BELLWETHER: A FLYING WING FEATURING  
THE BELL-SHAPED SPANLOAD

Major Field: Mechanical and Aerospace Engineering

Biographical:

Education:

Completed the requirements for the Master of Science in Mechanical and Aerospace Engineering at Oklahoma State University, Stillwater, OK in 2021.

Completed the requirements for the Bachelor of Science in Mechanical and Aerospace Engineering at Oklahoma State University, Stillwater, OK in 2019.

Professional Membership:

American Institute of Aeronautics and Astronautics