

# MAE 332

## Cruise Efficient Short Takeoff Business Jet Sizing Report

### Team 1

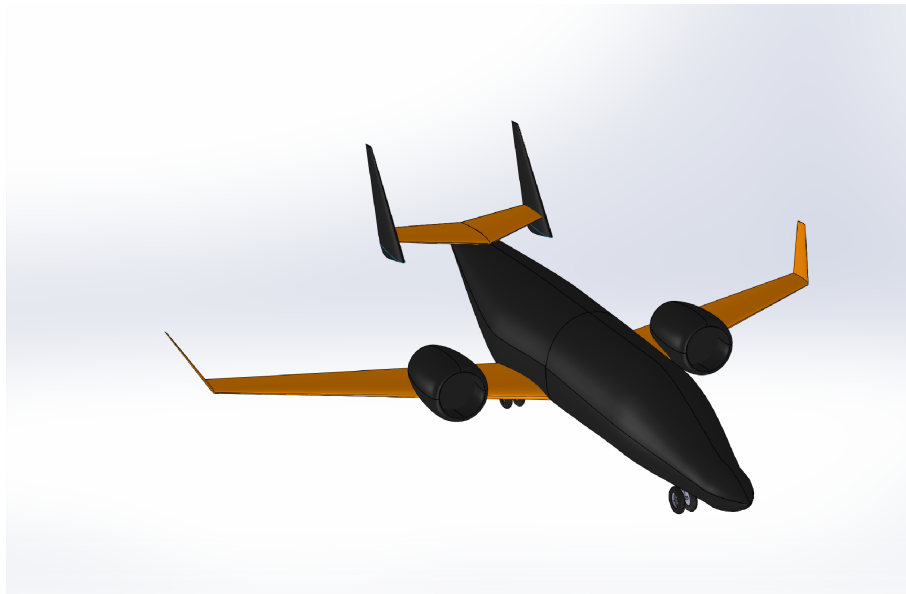


Figure 1: An isometric view of the final design



Princeton University  
Professor L. Martinelli  
Dept. of Mechanical and Aerospace Engineering  
March 1, 2020

# Contents

1	Executive Summary	7
	1.1 Aircraft Mission	7
	1.2 Overview	7
	1.3 General Description and comparison to other Business Jets	7
2	Sizing	9
	2.1 Take-off Weight Estimate	9
	2.2 Constraint Plane	10
	2.3 Wing Sizing	11
	2.4 Wing Dihedral Angle	12
	2.5 Wing Location	12
	2.6 Tail Sizing	12
	2.6.1 H-tail vs Conventional Tail	13
	2.7 Fuselage Length and Diameter	14
3	Configuration	15
4	Propulsion Analysis and Calculations	19
	4.1 Engine Selection	19
	4.2 Williams International FJ44-2A Specifications	19
5	Aerodynamic Driven Iteration	21
	5.1 First Iteration: [Configuration Overview]	21
	5.1.1 Airfoil Selection	21
	5.1.2 Wing Design	23
	5.1.3 Flap and Control Surface Design	24
	5.1.4 Tail Design	26
	5.1.5 Fuselage Design	26
	5.1.6 Upper Surface Blowing	29
	5.1.7 Engine Positioning + Design	29
	5.1.8 Nacelle Design	30
	5.2 Second Iteration: [OpenVSP Configuration]	30
	5.2.1 Wing Position Iteration	30
	5.2.2 Wing Sweep Iteration	31
	5.2.3 Tail Design and Position Iteration	32
	5.3 Third Iteration: [2D CFD Analysis and USB Redesign]	32
	5.3.1 Grid Generation	33
	5.3.2 2D Ansys Simulation	33
	5.3.3 Iteration	33
	5.3.4 Upper Surface Blowing Redesign	34
	5.4 Fourth Iteration: [Full Plane Optimization in SUAVE]	36
	5.4.1 Plane Configuration	36
	5.4.2 Flight Quality Verification	37
	5.4.3 Custom Optimization Framework	40
	5.4.4 Results	41
	5.5 Final Configuration: [Verification With 3D CFD Analysis]	44
	5.5.1 Mesh Generation	44

	5.5.2 3D Ansys Simulation . . . . .	45
	5.5.3 Results . . . . .	45
6	Landing Gear . . . . .	48
	6.1 Arrangement . . . . .	48
	6.2 Tire Sizing . . . . .	50
	6.3 Shock Absorbers . . . . .	52
	6.4 Kinematics . . . . .	53
	6.5 CAD Models . . . . .	54
7	Structural Design, Analysis and Configuration . . . . .	58
	7.1 V-n Diagram / Gust Envelope . . . . .	58
	7.2 Main Wing . . . . .	59
	7.3 Vertical Tail . . . . .	62
	7.4 Horizontal Tail . . . . .	63
	7.5 Fuselage . . . . .	63
	7.6 Volume Constraints . . . . .	65
8	Extreme Thermal Exposure . . . . .	66
	8.1 Thermal Barrier Coating . . . . .	66
	8.2 Titanium Aluminide . . . . .	66
9	Flying Qualities Assessment . . . . .	69
	9.1 Aerodynamic Coefficients . . . . .	69
	9.2 Construction of State Matrices . . . . .	70
	9.2.1 Longitudinal Mode . . . . .	70
	9.2.2 Lateral Directional Mode . . . . .	71
	9.2.3 Eigenvalues . . . . .	71
	9.3 Trim Conditions . . . . .	71
	9.4 Steady State Behaviour in Level Flight . . . . .	73
	9.5 Initial Condition Response . . . . .	73
10	FAR 23 Specifications . . . . .	75
	10.1 FAR 23.1 - Flight Recording . . . . .	75
	10.2 FAR 23.2000 - Controllability and Structure . . . . .	75
	10.3 FAR 23.2005 - Aircraft Category . . . . .	75
	10.4 FAR 23.21 - Mass and Static Properties . . . . .	75
	10.5 FAR 23.211 - Stall Modes . . . . .	75
	10.6 FAR 23.2115 - Take off Performance . . . . .	75
	10.7 FAR 23.2225 - Structural Loading . . . . .	75
	10.8 FAR 23.2155, 23.2305 - Landing Gear and Taxiing . . . . .	76
	10.9 Conclusions and Other Comments . . . . .	76
11	Business Projections and Cost Analysis . . . . .	77
12	Appendix . . . . .	83
	12.1 V-n Diagram Calculations . . . . .	83
	12.2 Titanium Alpha Alloy Properties . . . . .	86
	12.3 SUAVE Optimization Code . . . . .	90
	12.4 MATLAB Code . . . . .	95

12.5 Initial Condition Response Graphs . . . . .	119
12.5.1 Responses using unaltered VSP coefficients . . . . .	119

## List of Tables

1 Table of Requirements . . . . .	7
2 Table of Designed Parameters . . . . .	8
3 Weight Fractions . . . . .	9
4 Design Point From Weight Estimate and Constraint Plane . . . . .	11
5 Design Points for Wing Sizing. . . . .	12
6 Design Points for Wing Sizing. . . . .	13
7 Design Points for Wing Sizing. . . . .	13
8 Table of Designed Parameters . . . . .	17
9 Engine Comparison . . . . .	19
10 FJ44-2A Engine Data . . . . .	20
11 FJ44-2A Engine Data for Suave . . . . .	20
12 Initial Conditions, Optimized Results, Bounds . . . . .	42
13 Optimization Objective . . . . .	43
14 Optimized constraint parameters . . . . .	43
15 Optimization Constraints and Bounds . . . . .	43
16 Landing Gear Arrangement . . . . .	50
17 Wheel Loading and Braking Force . . . . .	51
18 Goodyear Aviation 186F03-2 Tires [1] . . . . .	52
19 Oleo Struck Shock Absorber Size . . . . .	53
20 Aircraft Mass . . . . .	69
21 Aerodynamic Coefficients (taken from VSP) . . . . .	70
22 Trim Conditions . . . . .	72
23 Cost Analysis Inputs . . . . .	78
24 Price Points . . . . .	79

## List of Figures

1 An isometric view of the final design . . . . .	1
2 A 3-view of the final design with dimensions . . . . .	7
3 Constraint plane generated by initial sizing process . . . . .	11
4 Graph showing how drag varies with fineness ratio [2] . . . . .	14
5 NASA & Boeing analyses of powered-lift methods . . . . .	15
6 First configuration iteration . . . . .	16
7 Second configuration iteration . . . . .	17
8 Flight Envelope for the TigerPounce . . . . .	18
9 FJ44-2A Dimensions . . . . .	20
10 Airfoil Choices . . . . .	22
11 Airfoil Choices Continued . . . . .	22
12 Takeoff Characteristics for Each Airfoil . . . . .	22
13 Takeoff Characteristics for Each Airfoil Continued . . . . .	23
14 Flap Design . . . . .	25
15 Flap Design . . . . .	25
16 Sears-Haack Ideal Body Distributions . . . . .	27
17 Area Distribution of Initial Configuration [Black], Sears-Haack Body [Blue] . . . . .	27



18	Area Distribution of Initial Configuration [Black], Sears-Haack Body [Blue]	28
19	Wing Configuration Options . . . . .	31
20	Wing Sweep Options . . . . .	31
21	Empennage Configuration Options . . . . .	32
22	Design Revision 1: 2D Mach and Pressure Countour for Engine + Wing Configuration . . . . .	33
23	Wing-Engine Configurations . . . . .	34
24	Final Configuration of Engine-Wing . . . . .	34
25	Engine Housing Configuration . . . . .	35
26	Engine Housing Configuration . . . . .	35
27	Engine Positioning and Nacelle Redesign . . . . .	36
28	YC-14 Data [Curve Fit for High AoA Reference in SUAVE] . . . . .	37
29	$\frac{L}{D}$ vs. Time, SUAVE . . . . .	37
30	Specific Fuel Consumption vs. Time, SUAVE . . . . .	38
31	Weight vs. Time, SUAVE . . . . .	38
32	Drag vs. Time, SUAVE . . . . .	39
33	Altitude vs. Time, SUAVE . . . . .	39
34	Thrust vs. Time, SUAVE . . . . .	39
35	Quarter Chord Sweep vs. Aspect Ratio Optimization Space . . . . .	41
36	Altitude vs. Aspect Ratio Optimization Space . . . . .	41
37	Altitude vs. Quarter Chord Wing Sweep Optimization Space . . . . .	42
38	Comparison of initial and optimized design points . . . . .	44
39	Simplified Geometry for Meshing Process . . . . .	44
40	3D Mesh . . . . .	45
41	3D Incompressible CFD . . . . .	46
42	$C_L$ vs. $\alpha$ . . . . .	46
43	3D Pressure Coefficient Distribution at Cruise . . . . .	47
44	3D Mach Number Distribution at Cruise . . . . .	47
45	Geometry of Tricycle Landing Gear <b>3</b> . . . . .	48
46	Wheel Loading Geometry <b>3</b> . . . . .	49
47	Geometry of Nose Wheel <b>3</b> . . . . .	50
48	Kinetic Energy Per Brake <b>3</b> . . . . .	51
49	Tire Deflection Under Load <b>3</b> . . . . .	52
50	Landing Gear with Color Code and Kinematic Drawing . . . . .	54
51	Parameters or Landing Gear CAD Models . . . . .	54
52	Side view of Landing Gear in Static Position . . . . .	55
53	Side view of Aircraft with Landing Gear in Static Position . . . . .	55
54	Isometric view of Main Landing Gear in Static Position . . . . .	56
55	Front view of Aircraft with Landing Gear in Static Position . . . . .	56
56	Side view of Aircraft with Landing Gear Retracted . . . . .	57
57	Top view of Aircraft with Landing Gear Retracted . . . . .	57
58	Initial Structure Model . . . . .	58
59	V-n Diagram with Gust Envelope Overlaid . . . . .	59
60	Von Mises Stress on the Main Wing Under Load . . . . .	61
61	Deformation of the Main Wing Under Load . . . . .	61
62	Von Mises Stress on the Vertical Tail . . . . .	62
63	Deformation of the Vertical Tail Under Load . . . . .	62
64	Von Mises Stress on the Horizontal Tail . . . . .	63
65	Deformation of the Horizontal Tail Under Load . . . . .	64
66	Von Mises Stress on the Fuselage Due to Pressure Load . . . . .	65

67	Von Mises Stress on the Fuselage Due to Pressure Load	65
68	Deformation of the Fuselage Under Load	65
69	Representation of cockpit (yellow) and cabin (green) seating, cargo volume	66
70	Thermal Barrier Coating, a thermal protection system commonly applied to turbine blades within the engine housing.	67
71	Thermal Analysis conducted on CREO Parametric 5.0.	67
72	Angle of Attack vs. Time, SUAVE	72
73	A figure showing the iterations to reach converged trim conditions	72
74	3D plot of 60s flight simulation	73
75	Minimum Sell	79
76	TigerPounce Cost Breakdown	80
77	TigerPounce Revenue	81
78	Lateral, x(1)	119
79	Lateral, x(2)	119
80	Lateral, x(3)	120
81	Lateral, x(4)	120
82	Longitudinal, x(1)	121
83	Longitudinal, x(2)	121
84	Longitudinal, x(3)	122
85	Longitudinal, x(4)	122
86	Lateral, x(1) (Altered Coefficients)	123
87	Lateral, x(2) (Altered Coefficients)	123
88	Lateral, x(3) (Altered Coefficients)	124
89	Lateral, x(4) (Altered Coefficients)	124
90	Longitudinal, x(1) (Altered Coefficients)	125
91	Longitudinal, x(2) (Altered Coefficients)	125
92	Longitudinal, x(3) (Altered Coefficients)	126
93	Longitudinal, x(4) (Altered Coefficients)	126

# 1 Executive Summary

## 1.1 Aircraft Mission

This report presents the final design configuration for the "TigerPounce" Business jet. The TigerPounce is designed to exceed leading business jet performance by being able to safely takeoff and land on the shortest commercial runway in the world, which is 1300 [ft]. With this capability, the TigerPounce will have access to over 20,000 more airports than other leading six passenger jets, such as the Honda Jet. [4].

## 1.2 Overview

Table [1] below displays the design constraints for the jet. These are the 'hard' constraints that were used to drive the design process. They were agreed upon either by the novelty of this design, or by direct comparison to other leading business aircraft such as the Honda Jet [5].

Requirement	Value
Number of Passengers	6
Number of Crew	2
Take off distance	1300 [ft]
Landing Distance	1300 [ft]
Range	1500 [nm]
Cruise Mach	0.85
Cruise Alt	40,000 [ft]
Climb Rate @ 10000'	3990 [ $\frac{ft}{min}$ ]

Table 1: Table of Requirements

## 1.3 General Description and comparison to other Business Jets

Below is a 3-view of the final configuration.

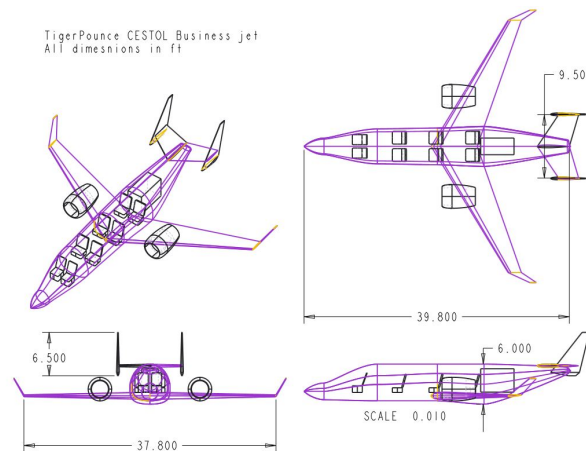


Figure 2: A 3-view of the final design with dimensions

As can be seen in Figure [2], the TigerPounce uses two turbofan engines mounted high on the wing to generate upper surface blowing during take off. This drastically increases the lift produced and allows for a shorter overall take-off length. To protect the wings

from the high exhaust temperatures, titanium-aluminium plates shield the critical areas and help prevent structural failure. The TigerPounce makes use of an H-tail and winglets in the interest of favourable drag characteristics.

Table 2 displays a number of the critical design parameters.

Parameter	Value	Source
Take off Weight	9609 [lb]	Weight Fraction Calculations [3], SUAVE optimization
Fuel Capacity	1536 [lb]	Weight Fraction Calculations [3], SUAVE optimization
Wingspan	38.8 [ft]	Constraint Plane [6]
Wing Area	172 [ft <sup>2</sup> ]	Constraint Plane [6]
Wing AR	9.4	SUAVE optimization
Sweep <sub>¼</sub>	28.6°	SUAVE optimizationm
L/D <sub>max</sub>	16.3	SUAVE computational model
λ	.29	Trapezoidal Assumption $\frac{b}{2} \frac{C_T + C_R}{2} = \frac{S_{ref}}{2}$
Vertical Tail Span	1.98 [m]	$C_{VT} = \frac{L_{VT} S_{VT}}{b S_{ref}} = 0.085$
Vertical Tail AR	4.44	Raymer [3]
Horizontal Tail Span	2.9 [m]	$C_{HT} = \frac{L_{HT} S_{HT}}{c S_{ref}} = .85$
Horizontal Tail AR	2.44	Raymer [3]
Fuselage Diameter	1.67 [m]	2 seats + Aisle
Fuselage Length	12 [m]	Finesse Ratio + Cargo Volume Requirements
Engines	2	T/W constraint (TO Thrust = $8.65 [\frac{kN}{engine}]$ )
SFC	.73 [ $\frac{lb}{hr}$ ]	Mfr Spec [3]

Table 2: Table of Designed Parameters

## 2 Sizing

### 2.1 Take-off Weight Estimate

To calculate the take off weight for the aircraft, weight fractions were estimated and combined for the mission. Assuming that the only change in aircraft weight throughout the flight is due to fuel consumption, weight fractions describe the ratio of aircraft weight at the beginning and at the end of a certain section of the flight mission. Using Raymer’s textbook on aircraft design [3], weight fractions were calculated (or estimated) for the mission and are displayed in Table 3. It should be noted that despite the aircraft’s mission simply being passenger transport, the F.A.A. requires aircraft to have a loiter time included, to account for holding time in airport landing patterns. The assumed loiter time was 30 minutes, and a fuel reserve of 6% was included in addition to this.

Mission Segment	Weight Fraction $\frac{W_{N+1}}{W_N}$
Take off	.97
Climb	.985
Cruise	.8725
Landing	.995
Whole Flight	.8295

Table 3: Weight Fractions

The fuel fraction of the whole flight describes the total weight of fuel burnt as a fraction of the aircraft’s weight, and can be written as:

$$W_{f/0} = \frac{W_{final}}{W_{TakeOff}} = \frac{W_{empty} + W_{payload}}{W_{empty} + W_{payload} + W_{fuel}} \tag{1}$$

This fraction can be used to solve the following non-linear parametric equations. These equations account for the structural weight of the aircraft. With advanced composite structures the wings and tails could be made 20% lighter, and the fuselage 10% lighter. Since Raymer’s *Aircraft Design* did not include a structural weight factor for the business jet class of aircraft, it the empty weight ratio was approximated by an average between that of a jet trainer and a jet transport; ultimately the empty weight ratio was within 5% of peer aircraft such as the Cessna Citation, suggesting that such estimation was reasonable, especially considering the use of composites– which can lighten the overall structural weight by as much as 5%.

$$W_{TO}(1 - s) - W_{fuel} - W_{payload} = 0 \tag{2}$$

$$S = 1.25(W_{TakeOff})^{-0.85} \tag{3}$$

A payload weight was estimated given the design requirements, accounting for the 6 passengers, 2 crew, and their respective baggage. It was assumed that each person weighed 200 lbs, and had 100 lbs of baggage each. After solving these equations using the method outlined in Raymer’s textbook [3], the take off weight was estimated to be 9733 [lbs], or 4415 [kg]. The full process and code for this calculation can be found in Section 12.

## 2.2 Constraint Plane

After finding a rough estimate for take off weight, a constraint plane was constructed. A constraint plane plots thrust loading against wing loading, or  $\frac{Thrust [lb]}{Weight [lb]}$  against  $\frac{Weight [lb]}{Wing Area [ft^2]}$ , for a given aircraft. For this design, four regimes were considered, represented by four different lines on the plane. These regimes were take off, cruise, landing and climb. Based on the intersection of these lines, a 'design point' can be found. Through continued iteration and estimation adjustment, the 'design point' can be improved until it represents something physical for the aircraft design. The cruise regime was plotted by making simplifications and assumptions on the 'master equation' presented below:

$$\frac{T}{W} = \frac{\beta}{\tilde{\alpha}} \left( \frac{q_{inf} S}{\beta W} \left( k_1 \left( \frac{n\beta W}{q_{inf} S} \right)^2 + k_2 \left( \frac{n\beta W}{q_{inf} S} \right) + C_{D_0} + C_{D_r} \right) + \frac{P_s}{V} \right) \quad (4)$$

Since no change in elevation occurs during cruise, it was assumed that  $P_s = \frac{dh}{dt} = 0$ ; however, for the climb specification – 3990 ft/min – the term was included in the equation to produce a different curve. It was also assumed that  $k_2 = C_{D_r} = 0$ . Some values such as  $q_{inf}$  were calculated using given design requirements. All other values, such as  $\tilde{\alpha} = .375$  (cruise) and  $.7388$  (climb) were estimated using Raymer's design text [3]. While the previous equations determine the 'cruise-efficient' nature of the CESTOL characteristics of the plane, the short takeoff and landing requirements separate calculations notably different from those of a CTOL plane. Foremost is the amplification of  $CL_{max}$  achieved by passive thrust vectoring via overblown flaps; during the takeoff and landing approaches, the thrust vector of the engines will be directed downwards to provide direct lift at the expense of axial thrust. This allows for an effective  $CL_{max}$  greater than that achievable via aerodynamics alone. To account for this, the fractional difference between the  $CL_{max}$  necessary for takeoff and  $CL_{max_{aero}} = 3.0$  was determined and calculated as a fraction of the takeoff weight  $W_{TO}$ :

$$\left( \frac{T}{W_{TO}} \right)_V = \frac{CL_{max} - 3}{CL_{max}} \quad (5)$$

Once summed with the axial thrust-to-weight ratio  $\left( \frac{T}{W_{TO}} \right)_{axial}$  necessary for takeoff, the  $\frac{T}{W_{TO}}$  for the takeoff constraint was plotted as the magnitude

$$\frac{T}{W_{TO}} = \sqrt{\left( \frac{T}{W_{TO}} \right)_{axial}^2 + \left( \frac{T}{W_{TO}} \right)_V^2} \quad (6)$$

where  $\left( \frac{T}{W_{TO}} \right)_{axial}$  was numerically solved for via force-balance equation for takeoff distance  $S_{TO}$

$$S_{TO} = \frac{1}{2B} \ln \frac{A}{A - BV_{TO}^2} \quad (7)$$

where

$$A = g \left( \frac{T_o}{W} - \mu \right) \quad (8) \quad B = \frac{g}{W} \left( \frac{1}{2} (\rho S (C_{D_g} - C_{L_g}) - a) \right) \quad (9)$$

where  $a$  is a constant relating velocity to thrust losses and  $C_{D_g}$  &  $C_{L_g}$  are the grounded drag and lift coefficients, respectively [7]. Likewise, the wing loading necessary for landing was not calculated via landing speed but by the distance equation [3]

$$S_{TO} = 400m; S_{TO} = 5 * \frac{W}{S} * \frac{1}{CL_{max}} + 137 \tag{10}$$

An import design decision made during this process was the selection of  $CL_{max}$  available for takeoff and landing; various powered lift techniques can generate an effective  $CL_{max}$  of up to 7 [8], but such a choice could not be made arbitrarily. Per eqn.10, the maximum allowable wing loading increases in proportion to  $CL_{max}$ ; however, as more thrust is used for lift augmentation rather than propulsion the necessary thrust/weight ratio also increases. Ultimately a  $CL_{max}$  of 4.3 was chosen as to approach the boundary of as many constraint curves as possible and thus maximize the efficiency of the design. This design point and its place in the final constraint plane is presented in Figure 3. The final converged design point is shown in Table 4.

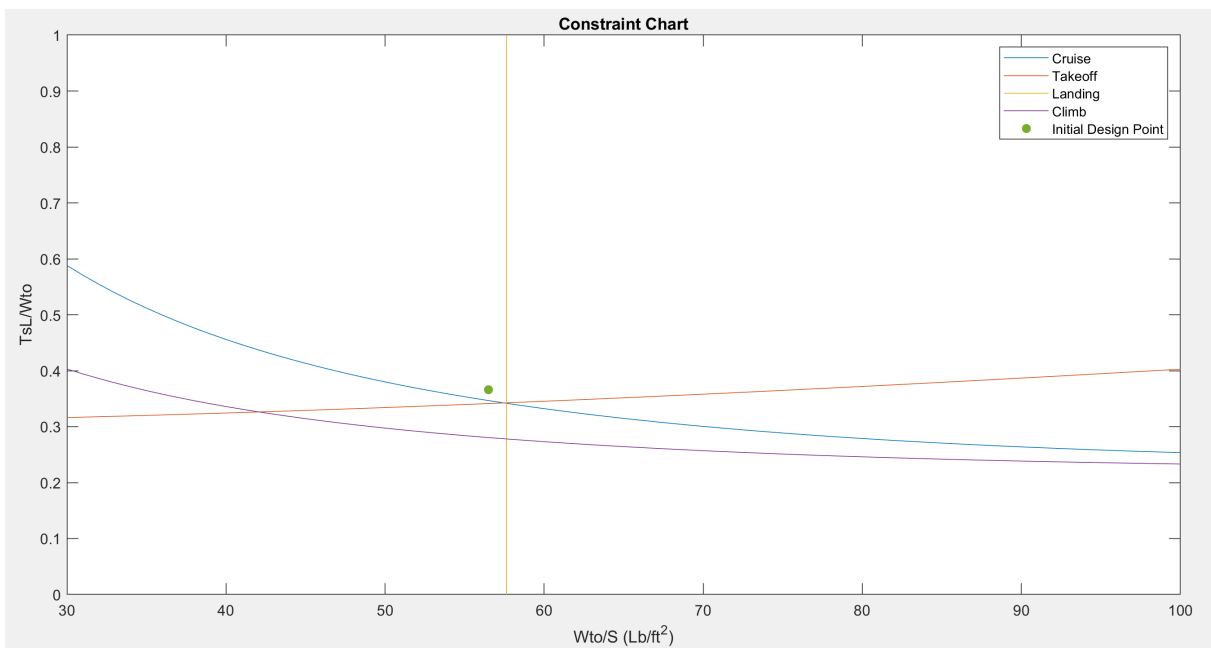


Figure 3: Constraint plane generated by initial sizing process

Design Parameter	Value
Take off Weight	4415 [kg]
W/S	276 [kg/m²]
T/W	.366

Table 4: Design Point From Weight Estimate and Constraint Plane

### 2.3 Wing Sizing

With a maximum take off weight and a design point on the constraint plane, the aircraft itself could now be dimensioned. This began with wing sizing, which required solving a sequence of simple algebraic relations relating wingspan, aspect ratio, wing area, and the constraint plane design point. These calculations are detailed in Section 12. Quarter-chord sweep angle was determined by reference to empirical trends relating wing sweep and cruise mach number [3]; for a cruise regime at  $M = .85$ , a quarter-chord sweep of 25 ° was selected. The thickness/chord ratio was similarly determined by empirical trends, settling upon an average  $fracTC$  of 15%. The taper ratio is determined by the given

wing area and span, when the assumption that the wing shape is trapezoidal is made. The ratio of  $C_T/C_R$  can be solved for explicitly. All of these values are shown in Table 5. It should be noted that in these calculations the value for wingspan does not account for the diameter of the fuselage.

Parameter	Value
Span	11.82 [m]
Wing Area	15.98 [m <sup>2</sup> ]
Quarter Chord Sweep	25°
$\lambda$	0.286
$\frac{T}{C}$	15%-

Table 5: Design Points for Wing Sizing.

## 2.4 Wing Dihedral Angle

A non zero wing dihedral is desirable, since it tends to create stability in flight by rolling the aircraft back to level when it is perturbed. However, backward wing sweep creates an effective dihedral angle since the aft-sweep has the same effect on stability. According to Raymer [3], this effect can be quantified by:

$$\text{Rolling Moment} \propto -\sin(2(\text{sweep})) \quad (11)$$

However, in the context of a STOL craft, a neutral wing with a dihedral of 0° is more suitable. This provides maximum lift in order to take off in as short a distance as possible, and the stability can be managed by a feedback control system.

## 2.5 Wing Location

The location of the wing along the length of the fuselage depends on the center of mass of the aircraft. The wing's center of pressure, which can be approximated by the wing's 1/4 chord location, should be placed in concordance with the aircraft's center of mass. This reduces the complexity of aircraft balance, and minimizes unnecessary stress on the air frame. Whilst the precise center of mass location cannot be determined until the full configuration (including fuel tank and landing gear location) has been designed, it was assumed the center of mass is slightly behind center of the fuselage. The wing was placed according to this assumption.

A low wing configuration was chosen for several reasons. Firstly, space in the cabin is a premium for business jets, and so a high or mid wing would subtract from space to move around. A low wing would also be optimal structurally speaking, since the wingbox could penetrate through the entire body of the wing beneath where the passengers are. A low wing configuration also allows higher placement of the engines, keeping the engine intakes clear of debris on less well built/maintained runways (the sort of runway this aircraft is designed to comfortably operate on).

## 2.6 Tail Sizing

The horizontal and vertical tails were sized according to the following tail volume coefficient equations:

$$c_{VT} = \frac{L_{VT}S_{VT}}{b_W S_W}$$

$$c_{HT} = \frac{L_{HT}S_T}{C_{S_W}}$$



where  $L_{HT,VT}$  is the lever arm from the quarter chord of the wing to the respective empennage component. However, due to the endplate effect – which can increase lift by up to 46% [9], the necessary horizontal tail area was adjusted accordingly. The tail sizing parameters are as follows:

### Vertical Stabilizer

Parameter	Value
Height	1.98 [m]
Wing Area	3.26 [m <sup>2</sup> ]
Leading Edge Sweep	37.5°
$\lambda$	0.25

Table 6: Design Points for Wing Sizing.

### Horizontal Stabilizer

Parameter	Value
Span	2.90 [m]
Wing Area	3.48 [m <sup>2</sup> ]
Leading Edge Sweep	30°
$\lambda$	.625

Table 7: Design Points for Wing Sizing.

#### 2.6.1 H-tail vs Conventional Tail

The concept of an H-tail was reached through attempts to maximize the control effect of the elevators and horizontal stabilizers, a necessity given the short takeoff and landing requirements as well as the short lever arm of the tail relative to the main wing quarter chord. Not only does the endplate effect of the H-tail allow for greater force from a smaller horizontal stabilizer, but vortex lattice evaluations showed a slight increase in  $\frac{L}{D_{max}}$  as well, an unexpected but not unwelcome change. The tails were sized as to maintain the sum area of the vertical stabilizers to satisfy the the vertical tail coefficient equation as shown above.

## 2.7 Fuselage Length and Diameter

In order to size the Fuselage of the aircraft, the seating layout, cargo volume and required fuel volume were taken into account. The aircraft is required to carry 6 passengers, and it was assumed the jet would have rows of 2 seats, similar to other business jets of this sort. It was decided that the fuselage shape and diameter would be designed in a similar fashion to that of the Honda Jet [5]. This led to a fuselage average diameter of 1.67 m. The fuselage length was chosen to be 12 m, in order to account for the payload requirements.

A quantity that is often used to verify and calculate fuselage length is the *fineness ratio*, which is defined as the ratio of fuselage length to diameter. This concept is discussed by Roskam’s work on aircraft design [2], suggesting the ideal fineness ratio for aircraft is between 6 and 8. It is discussed how this is un-achievable for airliners due to their capacity requirements, and oftentimes a penalty in fuselage drag is taken to accommodate more payload.

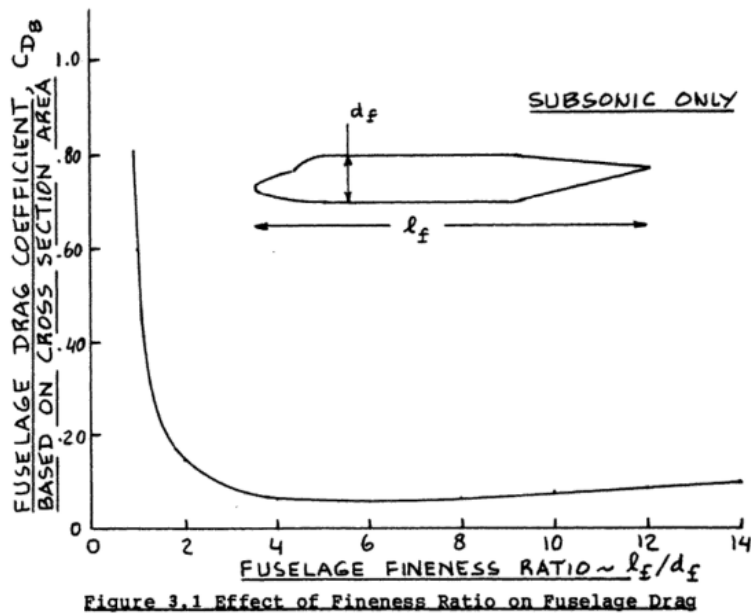


Figure 4: Graph showing how drag varies with fineness ratio [2]

The proposed fuselage has a fineness ratio of 7.2, which is in the ideal region presented by Roskam [2]. This is a good indicator that the width to length of the fuselage has been designed in the correct region.

### 3 Configuration

The primary challenge of the TigerPounce’s mission is to optimize aerodynamic efficiency at cruise conditions while still generating enough lift to takeoff and land within a 1000m runway; these parameters are often at odds with each other as designs optimized for low speed lift – such as unswept, high aspect-ratio wings – would be structurally and aerodynamically unstable during transonic cruise conditions. As such, STOL designs functioning purely by aerodynamic effects were immediately ruled out. Furthermore, the constraint plane generated during the sizing process indicated that landing on a short runway would require very low wing loading – less than  $161 \frac{kg}{m^2}$ . Research conducted by Boeing in the 1960s when developing the YC-14 confirmed these conclusions and suggested the implementation of powered-lift methods such as those shown in figure 5 8 were analyzed to find the optimal design solution.

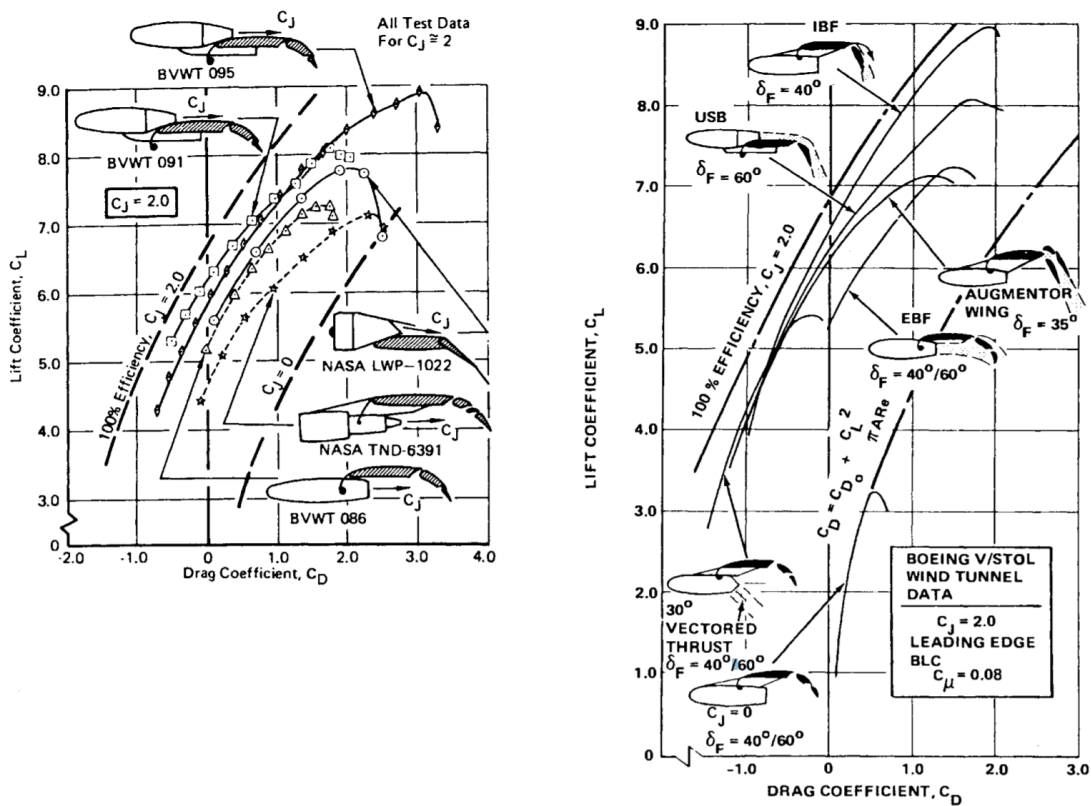


Fig. 14 Powered lift polar comparison. Both NASA and Boeing data agreed that the USB approach produced the best result. Although the augmentor wing and the internally blown flap (IBF) showed good performance, it was impracticable to duct through the wing the large quantities of air required.

Figure 5: NASA & Boeing analyses of powered-lift methods

Several possible configurations were assessed during this phase of design, most notably: directly vectored thrust, externally blown flaps, upper surface blowing, and internally blown flaps. The analyses were as follows:

- Thrust vectoring: Although this was prominent in our minds due to its highly visible application in V/STOL military aircraft, issues such as engine-out stability, technological complexity and aerodynamic efficiency – as shown in Fig. 5 – quickly eliminated thrust vectoring from consideration.
- Externally blown flaps: Externally blown flaps offered simplicity in design as well as a strong lift to drag ratio; however, the engines had to be mounted below the

wing, which would likely necessitate a high-wing configuration for sufficient ground clearance – not necessarily a negative but an important consideration nonetheless

- Upper surface blowing: Uses the Coanda effect to indirectly vector thrust blown over the upper surface of the wing and flaps; this option offers very high lift with minimal mechanical complexity and mounts the engine in front/on top of the wing, allowing for a configuration similar to many business jets with the exception of the engine mounting location.
- Internally blown flaps: Internally blown flaps offer superior lift to upper surface blowing and externally blown flaps, but the ducting system is mechanically complex and the setup would likely require a high wing for similar reasons to externally blown flaps.

With the exception of thrust vectoring, the three other powered-lift methods remained in consideration for implementation. Since each had sufficient lift-augmentation capabilities, selection of which method would be used was pushed back until aerodynamic configuration studies could be conducted as a means of additional differentiation.

Aerodynamic configuration began with a blend of STOL and traditional business jet characteristics– a low, aft mounted wing with thickness (15%) and aspect ratio derived from empirical models [3], winglets, and a cruciform tail sized using volume coefficient methods but a straight wing for optimal takeoff performance– as shown in figure 6.

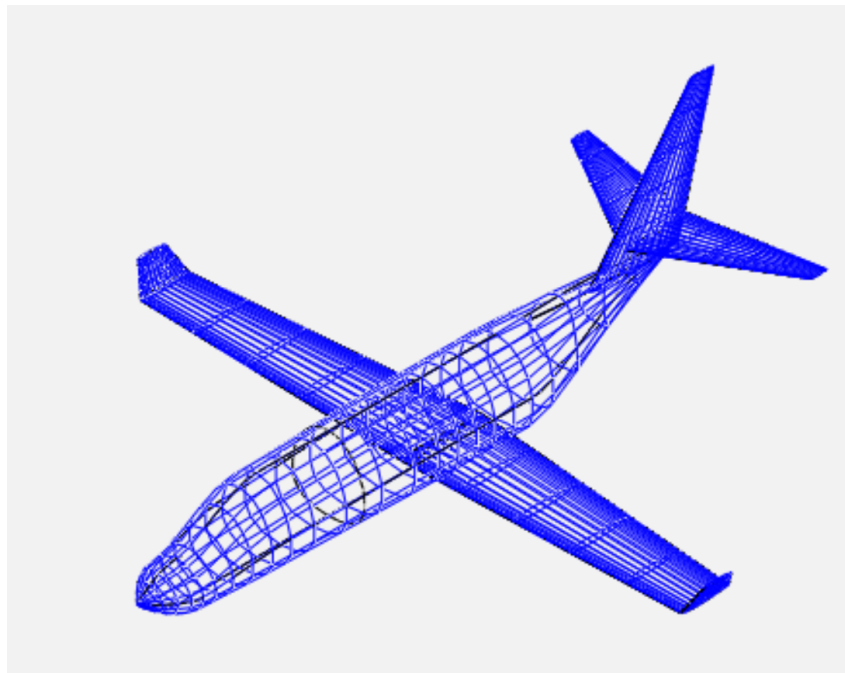


Figure 6: First configuration iteration

However, once modeled in OpenVSP, it became immediately clear that the short lever arm of the empennage required an unusually large – and correspondingly high-drag – set of stabilizers, and the straight wing would generate unacceptably high wave drag during transonic cruise. To remedy this, an H-tail was implemented and the wing was swept back in accordance with empirical trends, resulting in the design shown in figure 7 below.

From this starting point, aerodynamic modeling was conducted using vortex-lattice modeling in OpenVSP. This process was used not only to evaluate and feed back into the initial sizing Matlab script but to quickly provide first-order insight the various possible

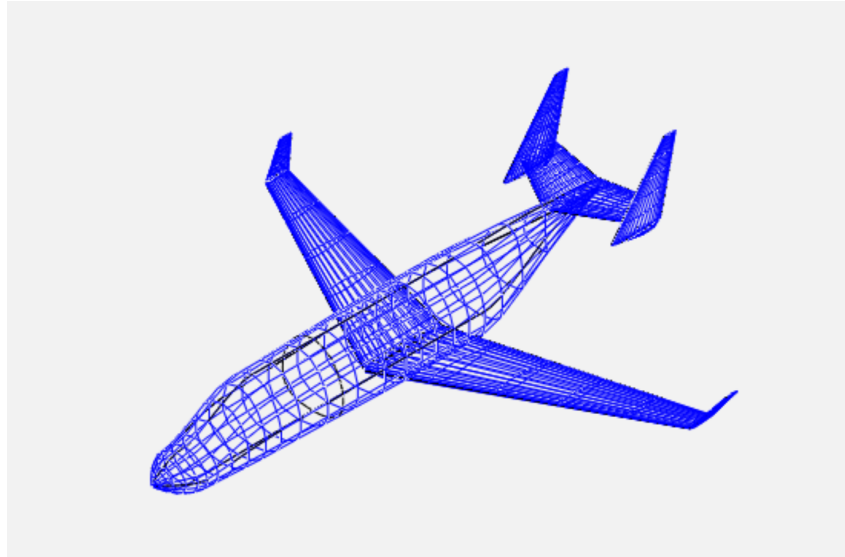


Figure 7: Second configuration iteration

configurations for the plane. Aerodynamically significant aspects such as tail configuration, wing sweep, and wing placement were varied across the design space in order to find the most efficient design. Each independent variable was altered one at a time from the H-tail, swept-wing design previously described—this both provided a control case for the study and also isolated the impact of each variable. The tested parameter space is described in Table 8 below:

Parameter	Values
Wing Sweep ( $^{\circ}$ )	0, 10, 20, 30
Wing Position – z	Low, Mid, High
Wing Position – x (m from nose)	3.048, 4.572, 6.096
Empennage Configuration	Conventional, Cruciform, High

Table 8: Table of Designed Parameters

It must be noted that the high wing design was not tested as a direct alteration from the reference configuration—interference effects from the high-mounted wing and engines on the empennage necessitate the implementation of a T-tail, so the high-wing variation was modeled as such. Furthermore, the mid-wing configuration, while typically the most aerodynamically efficient mounting, was not tested since the resulting wing box would pass directly through the passenger cabin, directly interfering with the mission specifications.

In-depth aerodynamic analysis of these designs will be discussed in *Section 8: Aerodynamics*, but the most important conclusions are presented here:

- Aerodynamic efficiency is independent of  $\lambda_1$  – all levels had an L/D of 15.7. However, given that the Tigerpounce will cruise at  $M=.85$ , in the transonic regime, greater sweep is desirable due to the increased critical Mach number and corresponding decrease in wave drag. Furthermore, any  $CL_{max}$  advantages that wings of lower sweep have during takeoff and landing are rendered irrelevant by the powered-lift methods to be implemented during those phases of flight.
- Aerodynamic efficiency is also independent of the fore/aft placement of the wing;

this gave confirmation that the wing placement could be adjusted to optimize static margin without causing aerodynamic or sizing issues.

- Conventional and cruciform tails are aerodynamically inferior to T and H tails in this application; the short lever arm from the wing’s center of pressure requires stabilizer surfaces prohibitively large without the endplate effects offered by T and H configurations.
- The high wing T-tail and low-wing H-tail possess similar aerodynamic performance, but the low-mounted wing offers mission-specific advantages, most notably an increase in effective tail upsweep—very desirable in STOL situations.

Given these results of the configuration study, upper surface blowing emerged as the best choice for implementation on the TigerPounce, primarily for its ability to be incorporated as part of a low-wing design. This configuration evaluation process in the end confirmed the fitness of a conventional business jet configuration for the CESTOL mission once modified to accommodate upper surface blowing engines and provided a strong starting point for optimization through more rigorous conceptual methods such as SUAVE. Notable alterations from the previously shown iteration include:

- Fuselage lofting modification in order to better approximate the Sears-Haack body for efficient transonic flight
- The wing was moved forward .283 m to create a desired static margin of 8%.
- Engine placement was formalized for upper surface blowing such that the exhaust jet is clear of the empennage.

In addition, a flight envelope for the configuration is shown below

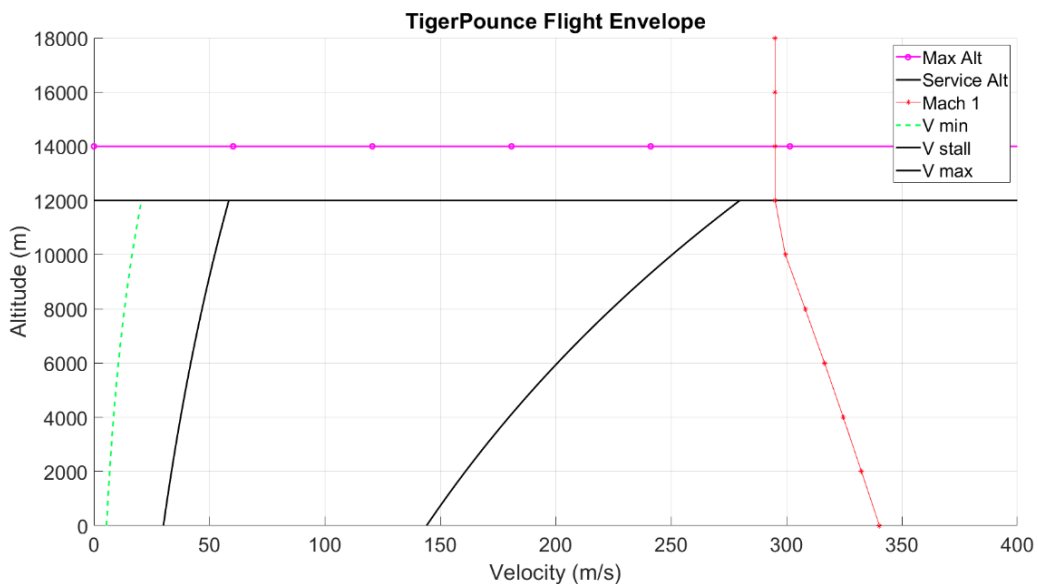


Figure 8: Flight Envelope for the TigerPounce

## 4 Propulsion Analysis and Calculations

### 4.1 Engine Selection

When selecting an engine, three primary criteria were kept in mind: 1) the engine must currently exist on the market. In order to make this design feasible, we do not wish to rely on any sort of technology that may be developed in the future. Therefore, the power plant for this aircraft must be an engine that is currently available on the market. 2) The engine must come from a reliable company. It is desirable to have an engine that is highly reliable, so the engine must come from a reputable manufacturer. 3) it must provide enough thrust to meet the required T/W ratio for all phases of flight, so the plane can fly.

Table 9 below contains a comparison of multiple engine models that fit within the criteria stated above. The three companies considered are Williams International, General Electric, and Honeywell. General Electric and Honeywell are widely known in the aircraft industry, however, a majority of their efforts are directed toward larger aircraft than the one being designed in this project. Williams International specializes in small gas turbines. They have multiple engine models ranging from 1000 to 3600 lbs, which will be more than sufficient for this design. Many well known companies in the aircraft industry - such as Cessna, Cirrus, and Beechcraft - have used Williams International engines to power their light business jets. Out of the three companies considered, Williams International is the primary contender.

As noted in Table 4 above, the required thrust to weight ratio for the design is 0.41 and the takeoff weight is 9733 lb. Multiplying these two numbers yields a required thrust of 3990 lb. With two engines, each engine must produce at least around 2000 lb of thrust to meet the requirement. In order to improve the safety of the design, it is necessary to have more than 3990 lb of thrust. With safety and optimization of weight and cost in mind, the best engine for this design is the Williams International FJ44-2A. With 2300 lb of thrust, a dry weight of 525 lb, and a SFC of 0.46 lb/lbf\*hr, the FJ44-2A will provide plenty of thrust to meet the design requirements while also providing a safety factor of 1.15.

Manufacturer	Model	Takeoff Thrust (lbf)	Dry Weight (lb)	SFC
Williams International	FJ44-1	2100	460	
Williams International	FJ44-2A	2300	525	0.46
Williams International	FJ44-3	3000	530	
General Electric	HF120	2050	466	<0.7
General Electric	CJ610-1	2850	396	0.97
General Electric	CJ610-2B	2400	400	0.97
General Electric	CJ610-4	2850	410	0.97
General Electric	CJ610-5	2950	421	0.97
Honeywell	TFE731	3500	734	0.5

Table 9: Engine Comparison

### 4.2 Williams International FJ44-2A Specifications

The FJ44-2A is a twin spool turbopfan with a single-stage fan and three-stage axial compressor direct driven by a two-stage turbine, a single-stage centrifugal compressor driven



by a single-stage turbine, an annual combustor, a full length bypass duct and an exhaust mixer. Table 10 below contains various data regarding the FJ44-2A, and Table 11 contains all of the values required to run a Suave analysis for the engine. Figure 9 depicts the geometry of the engine.

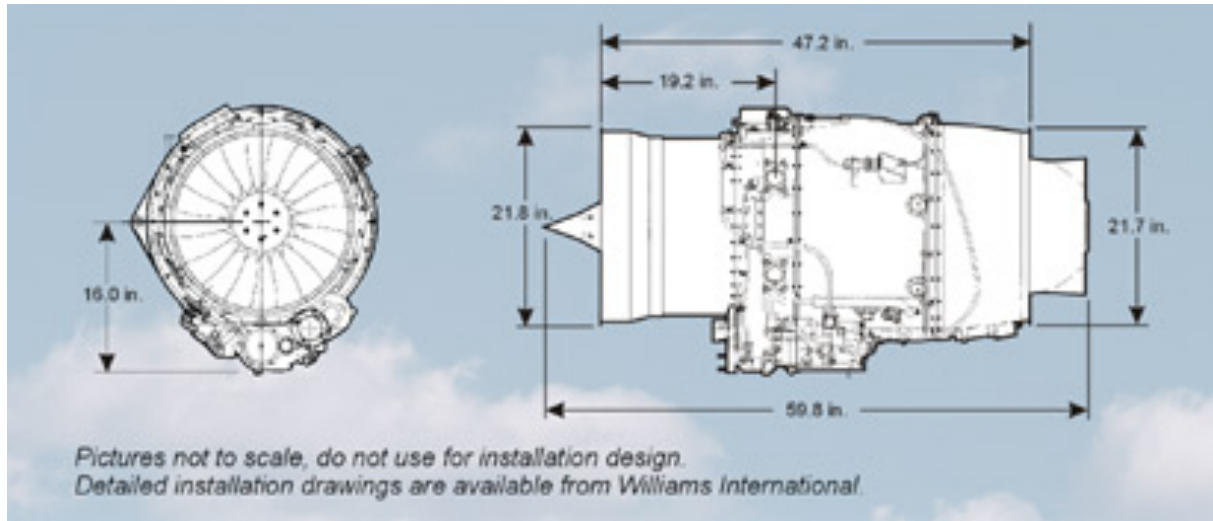


Figure 9: FJ44-2A Dimensions

Length Overall (in)	59.8	Between flanges (in)	47.3
Height Overall (in)	29.6	Forward flange OD (in)	21.8
Aft flange OD (in)	21.7	Dry Weight (lb)	520
Maximum thrust (lb)	2300	Max. Inter-turbine Temp. (F)	1508
Low pressure rotor speed (RPM)	18150	High pressure rotor speed (RPM)	40700

Table 10: FJ44-2A Engine Data

Bypass ratio	2.162	Number of engines	2
Engine length (in)	42.7	Nacelle diameter (in)	30
Inlet nozzle polytropic efficiency	0.98	Inlet nozzle pressure ratio	0.98
Nozzle polytropic efficiency	0.95	Nozzle pressure ratio	0.99
Nozzle polytropic efficiency	0.95	Nozzle pressure ratio	0.99
Compressor polytropic efficiency	0.91	Compressor pressure ratio	1.24
Compressor polytropic efficiency	0.91	Compressor pressure ratio	6.43
Fan polytropic efficiency	0.93	Fan pressure ratio	1.74
Combustor efficiency	0.99	Turbine inlet temperature (F)	1481
Combustor pressure ratio	0.95	Turbine mechanical efficiency	0.99
Turbine polytropic efficiency	0.93	Turbine mechanical efficiency	0.99
Turbine polytropic efficiency	0.93	Total thrust (lb)	4600
Altitude (ft)	40000	Mach number	0.85
ISA deviation	0		

Table 11: FJ44-2A Engine Data for Suave



## 5 Aerodynamic Driven Iteration

The following section highlights the design iteration process, primarily with respect to the aerodynamic design consideration inputs that played a role in the final design as well as optimization of the plane. In general however, it is important to note that the process of aerodynamic design and optimization was conducted in tandem with input and design considerations from many other disciplines, most notably a structural point of view. In any airplane design, weight plays an integral role in development, and thus structural considerations were an integral component of the aerodynamic iteration process. Overall, the aerodynamic design process flow consisted of the following

1. **Initial Design Sizing:** initial design sizing using takeoff weight estimation methods in addition to for example volume coefficient methods for vertical and horizontal tail sizing.
2. **Rough Refinement in OpenVSP:** After obtaining the initial configuration, rough refinement was conducted in OpenVSP for major aerodynamic iteration such as tail type selection, wing placement and engine placement.
3. **2D CFD:** 2D CFD was performed using Ansys2D after obtaining an initial configuration primarily in order to refine the upper surface blowing technique that was utilized in order to achieve the high lift characteristics for takeoff.
4. **Full-Plane Refinement in SUAVE:** After obtaining and rough model of the plane in OpenVSP, plane characteristics were modelled in SUAVE to both a.) corroborate expected flight quality properties of the initial sizing, and b.) to be able to achieve multi objective optimization for the whole plane.
5. **3D CFD:** Full-plane 3D CFD was performed in order to verify whole plane characteristics, especially takeoff lift coefficient, and aerodynamic properties at cruise.

### 5.1 First Iteration: [Configuration Overview]

Overall, the aerodynamics of the airplane were driven by the design requirements. Primarily, aerodynamics were designed to achieve the high lift characteristics needed for short take off, as well as provide good flight performance and drag qualities at cruise so as to be able to meet the mission range requirements these parameters are often at odds with each other as designs optimized for low speed lift – such as unswept, high aspect-ratio wings – would be structurally and aerodynamically unstable during transonic cruise conditions. As such, STOL designs functioning purely by aerodynamic effects were immediately ruled out. Rather, a variety of the powered-lift method were analyzed to find the optimal design solution. Initial aerodynamic considerations were a key driver in overall airplane design from the beginning of the design process, and initial aerodynamic work was based on the general output of configuration sizing efforts.

#### 5.1.1 Airfoil Selection

Initial airfoil choice is based primarily upon guidance found in Raymer. As a baseline, a supercritical wing will be chosen. Supercritical wings are flatter on top resulting in minimized angular change of flow with respect to upper surface air. The angular change to the air that is normally part of lift generation is decreased and this lift reduction is compensated for by deeper curved lower surfaces accompanied by a reflex curve at the

trailing edge. This results in a much weaker standing shock wave towards the rear of the upper wing surface and a corresponding increase in critical mach number. These shock affects are increasingly . Thus, the following airfoil types will be considered for the design based on the preceding rationale:

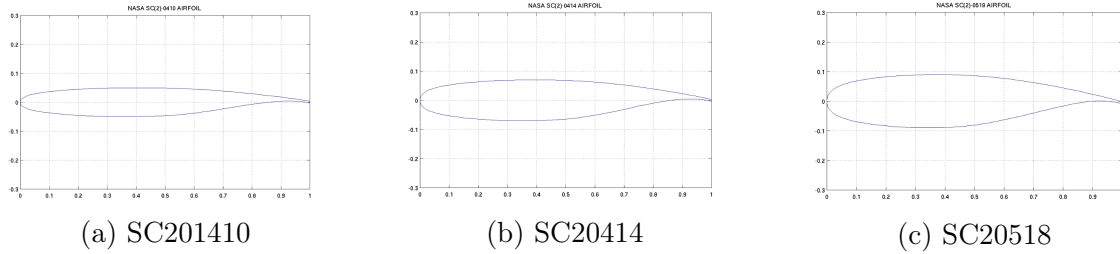


Figure 10: Airfoil Choices

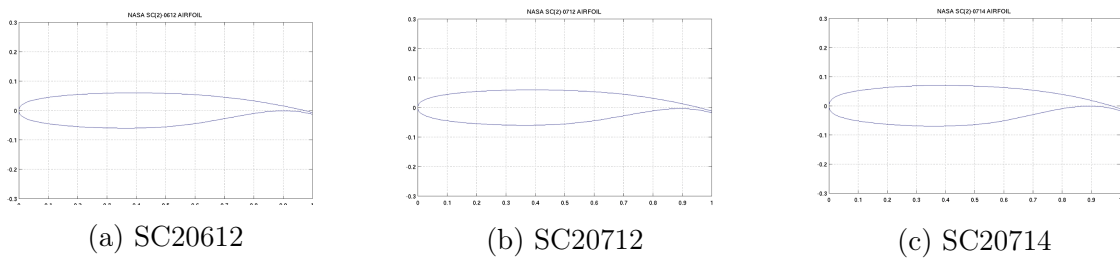


Figure 11: Airfoil Choices Continued

The airfoils along with NACA 0012 (for a baseline symmetric reference) will be tested using VSP-AERO for the plane configuration, and a final choice will be made based primarily on lift and drag characteristics. For high angle of attack takeoff requirements, verification will be conducted using 2D viscid simulations in Ansys. OpenVSP uses inviscid vortex lattice method theory to compute lift, and thus cannot compute inviscid effects or flow separation effects at high angles of attack, or effects due to thickness.

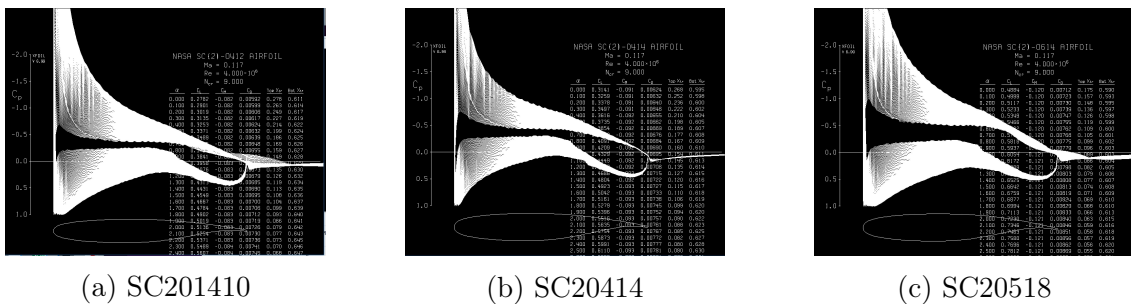


Figure 12: Takeoff Characteristics for Each Airfoil

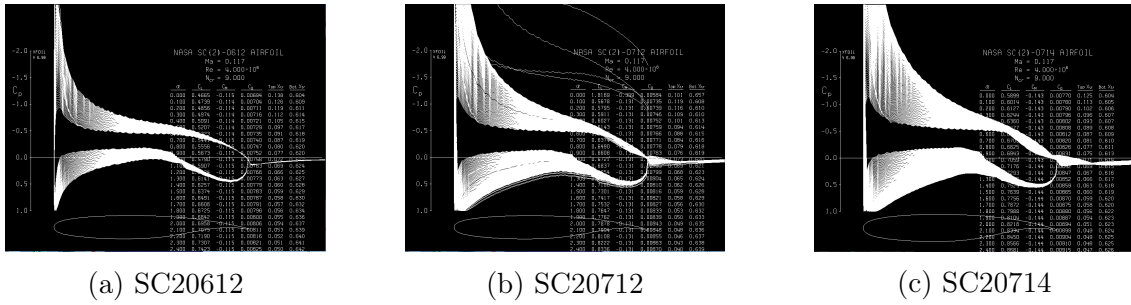


Figure 13: Takeoff Characteristics for Each Airfoil Continued

The above figures show the different coefficients of pressure on the respective airfoil at takeoff conditions (Mach 0.117 at sea level with a Reynolds Number of 4 million), using a program called *xfoil*. To see how the airfoils perform at different angles of attack during takeoff, a range from 0 to 15 degrees was set for the angle of attack. Based on the tests, the NASA SC20714 performed the best. This analysis was based on the absolute values of the pressure coefficients at the 0-degree point and 15-degree points. Also, airfoils, like the SC20612, SC20712, and SC20414, experienced incalculably high pressure before the airfoil reached a 12-degree angle of attack. The other airfoils showed promisingly low pressure coefficients at the 0-degree point, but as the angle of attack increased, the airfoils' efficiencies dropped. It is important to note that *xfoil* is not trustworthy once flow becomes transonic because *xfoil* does not reliably account for compressibility. Therefore, *xfoil* was not used for anything more than for takeoff analysis.

### 5.1.2 Wing Design

Wing design is of highest importance in the design of the aircraft. Wing design has a large effect on the performance of the aircraft of all ranges of the flight envelope. For the initial design of the wing, the following characteristics will be taken into account

1. **Thickness to Chord Ratio:** A relatively thick airfoil will be chosen due to favorable stall characteristics. At this stage in the design process, the thickness to chord ratio is essentially the most important wing design choice, as it plays a big role in determining drag and lift characteristics. In general, the thinner, the better it is from a drag perspective. However, a thinner wing for a given load is heavier. Thus, a balance will be constructed based on iteration in the VSP Aero design environment, with values of thickness to chord starting around 0.1.
2. **Sweep:** In order to increase divergent mach number within reasonable limits, wing sweep can be increased. This is an important factor in the design of such a transonic airplane, and an initial value will be set in the range of 25-30 degrees for quarter chord sweep. Final design decision will be made based on iteration in the VSP analysis environment and SUAVE optimization.
3. **Fuel Volume:** It is important to keep in mind that most fuel is generally contained within the wings, and thus the wings are sized to include requisite fuel in order to satisfy the mission design requirements (which is based on mission length, as well as  $\frac{L}{D}$  convergence, and specific fuel consumption of the engines).
4. **Position Aft of Nose:** Position aft of nose was based largely driven by stability. Wing positioning was achieved by estimating the center of pressure of the wing

(quarter chord), and using correlation for static margin to place the wing the a given relative to the center of mass of the plane (computed using VSP Aero). However, these details are not or primary concern at this stage in the design process.

5. **Height Relative to Fuselage:** The position of the wing relative to the fuselage (high wing, low wing) will be evaluated based on drag properties as well as structural considerations on interference effects with the tail.
6. **Dihedral angle:** Dihedral angle was chosen primarily by design advice given in Raymer. A non zero wing dihedral is desirable, since it tends to create stability in flight by rolling the aircraft back to level when it is perturbed. However, backward wing sweep creates an effective dihedral angle since the aft-sweep has the same effect on stability. According to Raymer [3], this effect can be quantified by

$$\text{Rolling Moment} \propto -\sin(2(\text{sweep})) \quad (12)$$

However, in the context of a STOL craft, a neutral wing with a dihedral of  $0^\circ$  is more suitable. This provides maximum lift in order to take off in as short a distance as possible, and the stability can be managed by a feedback control system.

7. **Aspect Ratio:** Aspect ratio is given based on the design assumptions made for calculation of  $\frac{L}{D}$ . Starting value is chosen at 9, and will be iterated using the VSP Aero computational analysis software. Although aspect ratio was primarily dictated by primary sizing convergence, aspect ratio will be an important design consideration in further iteration of the plane. High aspect ratio wings are affected less by tip vortices, and thus high aspect ratio wings do not experience as much lift loss and drag increase due to tip effects as much as low aspect ratio wings. However, it is also noted that higher aspect ratio wings are less robust to stalling, and that lower aspect ratio wings have more favorable stall delay characteristics. These tradeoffs are extremely important in light of the STOL requirements.

Note that variable wing sweep was not considered in the initial wing design. However, variable wing sweep of the trailing edge was considered upon further design iteration in order to complement landing gear requirements.

### 5.1.3 Flap and Control Surface Design

Flap and control surfaces were driven primarily by requirements on stability, but also by takeoff and landing high lift requirements. However, it was important to note that the flaps and slats are a major source of drag, which was considered upon further iteration. Specifically, the following aerodynamic considerations were taken into account when designing control surfaces and high lift devices:

1. **Flaps:** Flaps are a “high lift / high drag” devices—not only do they improve the lifting ability of the wing at slower speeds by changing the camber, or curvature of the wing, but when extended fully they also create more drag. This means an aircraft can descend (or lose altitude) faster, without gaining airspeed in the process. Thus, the flaps will be an integral component of the high lift design that is performed.

For the initial design, four types of flaps were considered: plain, slotted, split. Split flaps were ruled out because they do not change the maximum lift coefficient, but rather are used to increase drag upon descent. Due to the high lifting requirements, split flaps were ruled out. Plain flaps were also initially considered in order to lower

the aft portion of the wing, increasing its camber, which in turn causes the wing to produce more lift. Although plain flaps are generally simpler and lighter than slotted flaps, slotted flaps were chosen for the initial design configuration due to their high lift properties. The slotted will be designed to physically increase the overall surface area of the wing, as well as provide additional energy to the air on the upper surface of the wing, ensuring that as the airspeed decreases, the air still has sufficient momentum to reach the rear of the wing (and thus prevent are crucial in preventing boundary layer separation).

2. **Slats:** Similar to flaps, spoilers were considered on the leading edge of the main wing in order to changing the camber, or curvature of the wing, to improve lifting ability at slower speeds. However, slats were not included in the first design iteration, as they add unnecessary complexity and weight if they are not needed. That is, slats are only added if the deemed necessary to meet the STOL design requirement.
3. **Spoilers:** Similar to slats, spoilers were not considered in the first design iteration due to complexity and weight penalty cost.

Overall, the initial location and size of control surfaces were designed for stability. A picture is included below

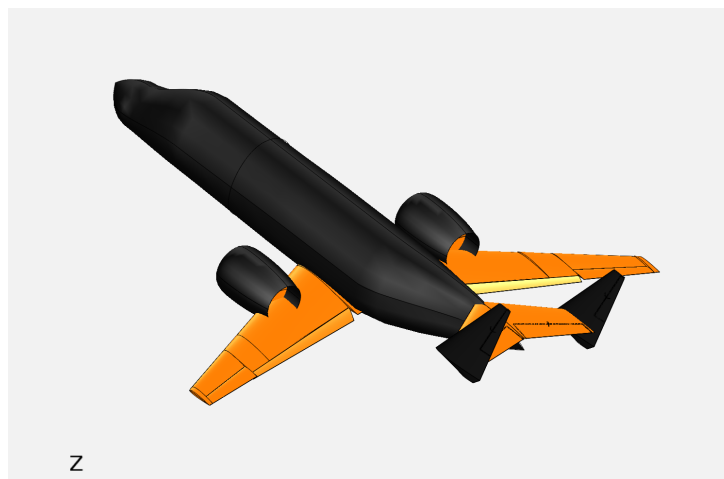


Figure 14: Flap Design

Baseline estimates were provided based on advice given in Raymer, as well as peer-based comparison to similar planes such as the Honda Jet. The span of the flaps is approximately 58 percent of the wingspan, and the chord is approximately 16 percent of the wing chord. Flap configuration from multiple views is shown below for reference

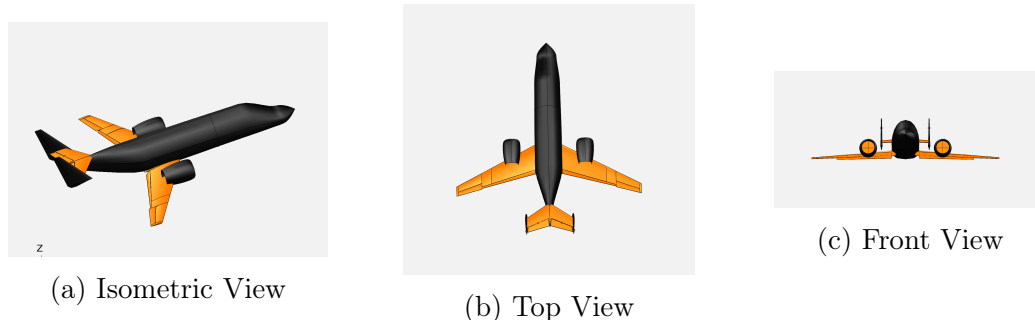


Figure 15: Flap Design

#### 5.1.4 Tail Design

Initial design sizing for the horizontal tail was given based on the horizontal tail volume coefficient (which was based on historical correlation). This is not discussed here to avoid repetition. Aside from horizontal tail area however, the configuration of the tail provides key aerodynamic and stability qualities that are crucial to the overall success of the plane in meeting the given design requirements. Most importantly, aerodynamic decisions about the tail of the aircraft were conducted around trim, stability and control properties (due to plane symmetry however, trim was only a primary concern for the horizontal tail).

Overall, the tail design was largely driven by the need for stability at high angles of attack condition. Based on design advice given in Raymer, a high tail, H-tail design was initially chosen to avoid turbulent flow conditions (upwash effects, downwash effects, etc) at high angles of attack. An H-tail was initially chosen because of its favorable end plate effects (which effectively increase aspect ratio) that would allow for the design of a smaller tail and ultimately save on weight penalty costs.

Overall, note that initial tail positioning is achieved according to guidance in Raymer based on historical correlation regarding the ratio of tail arm to mean chord. All final decisions regarding tail positioning, sizing and geometric properties will be made in a data driven way that optimizes for lowest takeoff weight of the plane while still providing stability and absolutely meeting design requirements. One important parameter to note is that aspect ratio is relatively low with respect to wing aspect ratio (9), one of the reasons being to delay tail stall.

#### 5.1.5 Fuselage Design

Fuselage design was driven primarily based on structural, cargo, and passenger requirements. For brevity, and the sake of confining this section to primarily aerodynamic related decisions these will not be discussed here.

From an aerodynamics perspective, fuselage design input was given primarily based on area ruling, which is important for aircraft such as the TigerPounce cruising at transonic speeds. At transonic speeds, local shock wave formation across the body of the aircraft causes wave drag, which is the main constituent of the drag on the aircraft at cruising conditions (which comprise the majority of the mission segment of the airplane). Thus, decreasing overall cruising wave drag was a primary concern in the development and refinement of the fuselage. In order to minimize wave drag, an approach was taken to minimize the derivative of the cross sectional area over the length of the aircraft. In addition to smoothing the cross sectional area change across the length of the fuselage, effort was taken (within reasonable structural and design limits) to emulate the Sears-Haack body which provides an ideal cross sectional area distribution. By the Whitcomb Area rule, the value of wave drag of a body that follows an equivalent area distribution along the length of the plane to that of the Sears-Haack body will have wave drag roughly equivalent to a Sears-Haack body scaled by the same maximum radius. The Sears-Haack body is defined by the equation

$$r(x) = R_{max} [x - x^2]^{\frac{3}{4}} \quad (13)$$

where  $x \in \{0, 1\}$  is the non-dimensionalized position along the length of the plane,  $r$  is the radius, and  $R_{max}$  is the maximum radius. In addition, the Sears-Haack body cross

sectional area distribution can be found as

$$A(x) = \pi R_{max}^2 [4(x - x^2)]^{\frac{3}{2}} \tag{14}$$

A plot of both the ideal Sears-Haack area distribution as well as radius distribution is found below

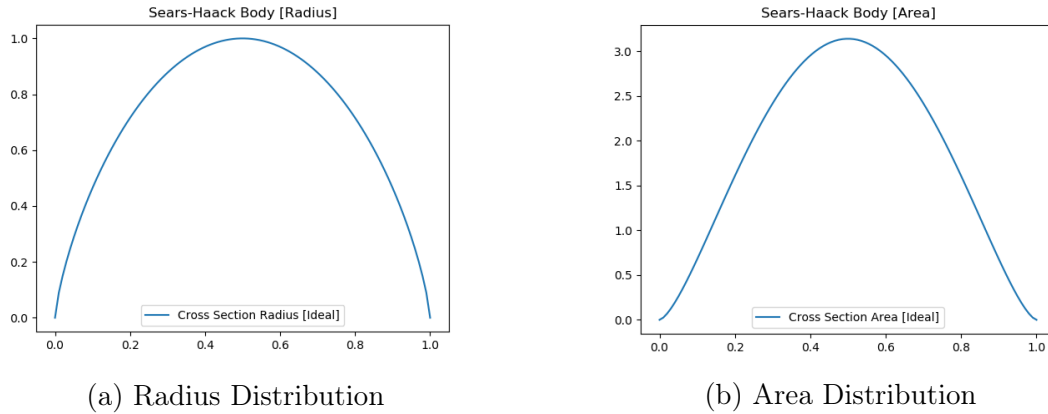


Figure 16: Sears-Haack Ideal Body Distributions

Based on initial sizing, the cross sectional area of the plane was generated in comparison to the ideal Sears-Haack body in OpenVSP as follows

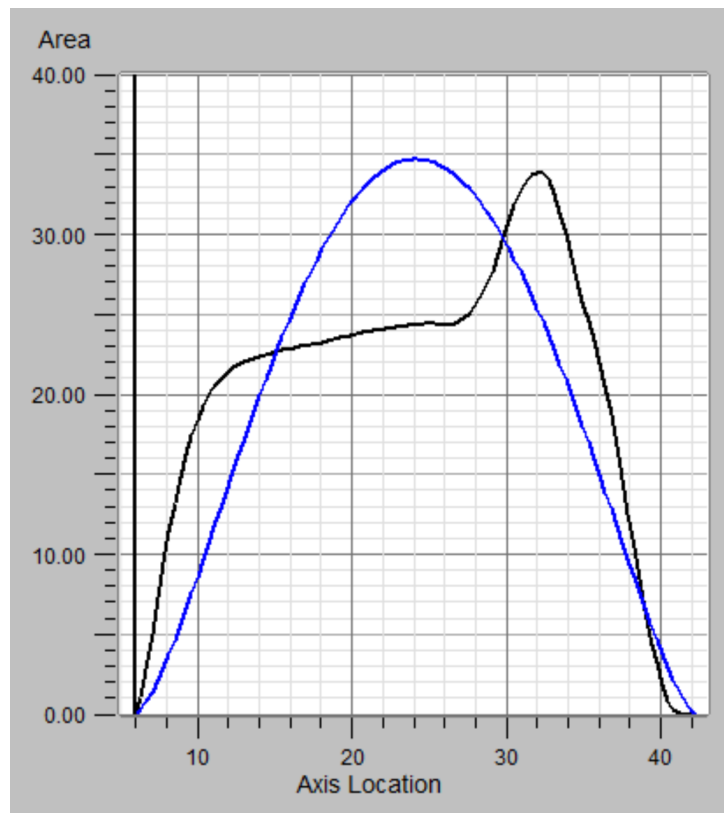


Figure 17: Area Distribution of Initial Configuration [Black], Sears-Haack Body [Blue]

It can be seen that the area distribution follows the Sears-Haack body quite poorly in the mid section from around 15-30 feet from the nose. The main differential with respect to the Sears-Haack body comes from the fact that the fuselage increases in area fairly

smoothly from the tip of the plane to the back of the cockpit, but then essentially stabilizes as the fuselage smooths out. The next major increase in body area around 25 feet from the tip of the plane occurs around the start of the main wing section.

Although rather little can be done to converge the area towards the ideal Sears-Haack body distribution, minor adjustments were performed to the fuselage using the OpenVSP Python API so that the area would converge. Wave drag optimization was performed by specifying the  $L_2$  norm integral across the shape, and using an off the shelf gradient based optimization framework to tweak the fuselage in the direction of largest descent (within bounds) with respect to the difference between the Sears-Haack ideal body, and the TigerPounce body. The number of section slices can be set using the API. However, it must be noted that OpenVSP wave drag calculations are updated using an area rule method to find the volume wave drag for a given Mach number. The number of slices and rotation about the center are specified, and then at each slice an area is found and an area distribution is fit along each rotation. The fit is created by Eminton-Lord wave drag evaluation. This means that increasing the number of slices increases the number of points in the distribution used for calculating wave drag, which then increases drag instead of making the calculation more precise. This posed a problem, since it was noticed that increasing the number of points kept increasing the drag for the same shape instead of leading to convergence. This problem was only found after lots of trial and error and reading!

Note that all adjustments were performed in accordance with fuel volume, passenger and cargo requirements and thus did not violate any design requirements. A graph of area distribution corresponding to the updated fuselage design is shown below

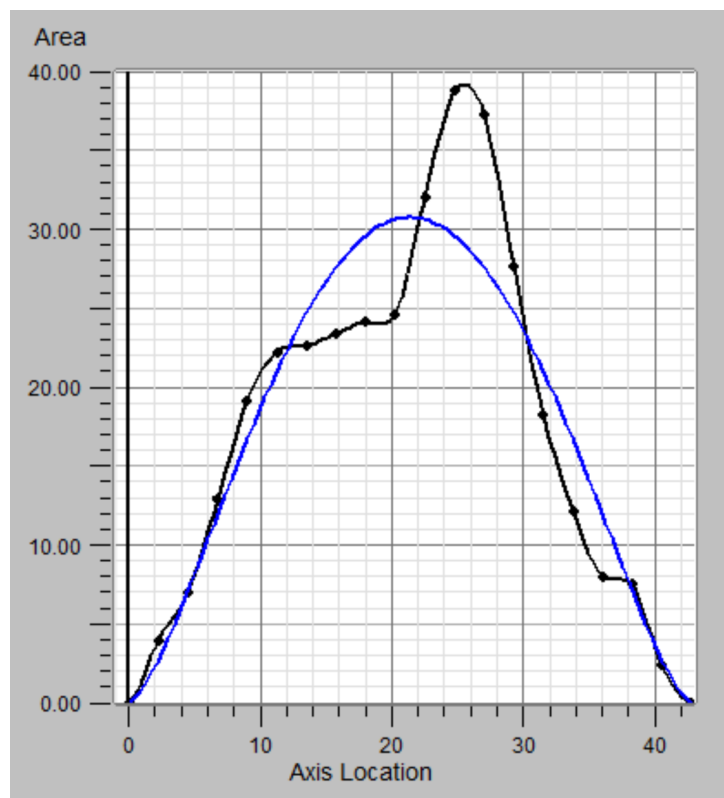


Figure 18: Area Distribution of Initial Configuration [Black], Sears-Haack Body [Blue]

Finally, it is important to note that the differential between the ideal body and Tiger-



Pounce body will largely change as a function of wing design and nacelle design. In addition, fuselage shape optimization alone is a small part of the full-plane, multi parameter optimization that will be conducted using the SUAVE computational environment at a later stage in the design process. Thus, the initial body re-design is subject to change.

### 5.1.6 Upper Surface Blowing

One of the most challenging aerodynamic considerations in the development of the Tiger-Pounce had to do with the aerodynamic design surrounding the STOL takeoff requirement. The STOL requirement dictated that some sort of high lifting device/concept was necessary, and upper surface blowing was chosen as the preferred method since it is generally more efficient than externally blown flaps according to data obtained from YC-14 design.

The design goals of upper surface blowing from an aerodynamics perspective are as follows:

1. **Facilitate STOL Design Requirement:** The primary objective of upper surface blowing is to facilitate the STOL design requirement.
2. **Minimize Thrust Impediment:** As a complementary goal to facilitating the STOL design requirement, the goal of upper surface blowing design is to minimize thrust losses from the engine.

### 5.1.7 Engine Positioning + Design

Initial thrust sizing requirement estimates indicated that a 2 engine design would be sufficient to satisfy the design requirements. Thus, aerodynamic design was conducted around a 2 engine design. This was beneficial as opposed to a 4 engine design because it would allow for weight savings as well as drag based savings that would come from having 2 additional engines and nacelles.

Overall, engine positioning was driven largely by the following aerodynamic considerations:

1. **Distance From Fuselage:** From an aerodynamic perspective, care was taken to place the engines away from the fuselage to prevent fuselage interaction with the turbulent exhaust stream of the engines. However, it is noted that placing engines farther away from the fuselage necessarily incurs a weight penalty, as the structure required to support the engines is greater for any cantilevered mass placed farther from fulcrum.
2. **Vertical Distance From Center of Mass:** Consideration was given to place the engines vertically with respect to the center of mass as close as possible. Doing so allowed for the reduction of thrust based moment that would occur if thrust generation was placed away from the center of mass of the plane.
3. **Tail Interference Effects:** Special consideration was given to place the engines outside of any direct upstream position with respect to the tail. This was primarily motivated by stability issues that would be incurred during cruise and especially during takeoff if the tail was in the turbulent exhaust stream of the jet engines.

### 5.1.8 Nacelle Design

Initial nacelle design was designed around 2 key principles:

1. **Facilitating Upper Surface Blowing:** The primary concern of the nacelles was to provide an appropriate structure that allowed for the STOL requirement to be realized. To that effect, the nacelles was initially designed around minimizing interference with upper surface blowing. To that end, it was decided that the nacelle support structure should be underneath the wing rather than on top of the wing, where the primary lift capabilities from upper surface blowing were generated. Initially, the nacelles were designed to cause high pressure effects on the underside of the engine, in order to aid lift generation.
2. **Drag Minimization:** In addition, the nacelles were designed to minimize drag, and were streamlined to the minimum structural requirement necessary to support the engine with safety margins.

Lastly, it is noted that although not a purely aerodynamic concern, nacelles were designed to minimize structural weight.

## 5.2 Second Iteration: [OpenVSP Configuration]

Given the initial configuration of the plane based off of the  $\frac{T_0}{W}$  and  $\frac{W}{S}$  constraint plane analysis, as well as the aforementioned design criteria, an initial model of the plane was generated in OpenVSP. A figure of the initial model of the plane is shown below Given the initial plane configuration, and rough aerodynamic configuration described above, the model was tested under various configurations to provide a "rough" layer of optimization of the configuration using OpenVSP. OpenVSP allowed for the determination of aircraft as an infinitely thin sheet of discrete vortices to compute lift and induced drag. However it is important to note that VLM model fidelity not include the influence of the thickness or viscosity, and thus it will be necessary to utilize a higher fidelity model for the final configuration evaluation. Overall, the VLM solver in OpenVSP allowed for quick iteration upon the following parameters

1. **Wing Position:** Wing position, from high to low was varied.
2. **Wing Sweep:** Wing sweep was varied in accordance with design advice found in Raymer as a baseline.
3. **Tail Design and Position:** Tail design (normal tail, H-tail, etc) and position (high, medium, low, etc.) was iterated for a variety of different wing configurations.

At this stage the best design based on that which minimized weight criteria under the given design requirements. The details of the iteration in OpenVSP are provided below.

### 5.2.1 Wing Position Iteration

Overall, three wing positions were considered: high, medium and low. The aerodynamic benefits of the high wing were that for the STOL requirements, the high wing position would allow room for the wing flaps that were originally designed for high lift generation. In addition, added benefits of the high would be that high wings generally decreases the "floating" tendency of lighter airplanes upon landing approach due to ground effects. Further, high wings would add a physical safety margin for unpaved runways where low

wings could potentially be caught in some type of grass or debris present on the runway.

With the aforementioned considerations in mind, final design choice was given after iteration in OpenVSP, as shown below.

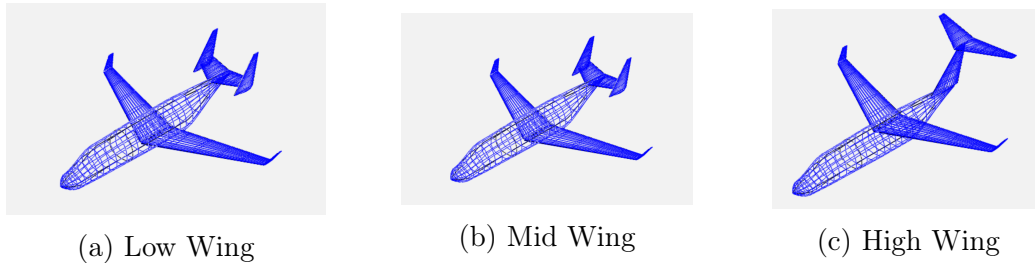


Figure 19: Wing Configuration Options

A comparison of the designs is included below:

Design Number	$\frac{C_L}{C_{D\ cruise}}, M=.85$	$S_{ref}$	$W_{TO}$
Low Wing	~15.8	15.9 m <sup>2</sup>	4415 kg
Mid Wing	~18.1	15.2 m <sup>2</sup>	4196 kg
High Wing	~15.4	16.17 m <sup>2</sup>	4461 kg

While the mid-wing mounting has superior aerodynamic qualities, the mission requirements preclude the usage of a mid wing; the wing box otherwise would pass directly through the plane’s cabin space. Since both low and high wing designs have comparable efficiency, both options remained in consideration from a purely aerodynamic perspective. However, as discussed in the Configuration section, a low wing was decided upon for the increase in effective tail upsweep that it provides.

### 5.2.2 Wing Sweep Iteration

Overall, various wing sweeps were considered based on nominal design advice given in Raymer. Overall, the goal of wing sweep iteration was to reduce the adverse effects of transonic flow and local shock formation that occurs in the transonic region. In addition, an added benefit of wing sweep is that sweeping wings improves stability and has a natural dihedral effect.

With the aforementioned considerations in mind, final design choice was given after iteration in OpenVSP, as shown below.

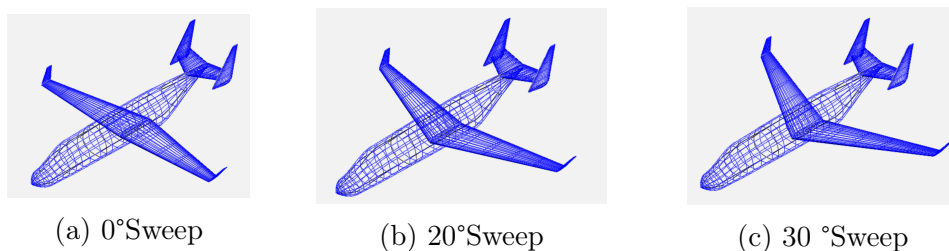


Figure 20: Wing Sweep Options

A comparison of the designs is included below:

Design Number	$\frac{C_L}{C_{D\ cruise}}, M=.85$	$S_{ref}$	$W_{TO}$
0 °	~15.8	15.9 m <sup>2</sup>	4415 kg
20 °	~15.7	16 m <sup>2</sup>	4426 kg
30 °	~15.5	16.14 m <sup>2</sup>	4450 kg

Although the unswept wing has the highest aerodynamic efficiency (albeit only marginally), the wave drag generated by such a wing would be intolerable without an impractically low thickness-to-chord ratio. Overall, the aerodynamic efficiency is not impacted by wing sweep and thus the matter can be left as-is (25 °) until SUAVE optimization.

### 5.2.3 Tail Design and Position Iteration

Overall, various tail designs were considered based on nominal design advice given in Raymer. Overall, the goal of tail design iteration was to meet the design goals based on short take of and landing requirements, and as a secondary result minimize drag.

With the aforementioned considerations in mind, final design choice was given after iteration in OpenVSP, as shown below.

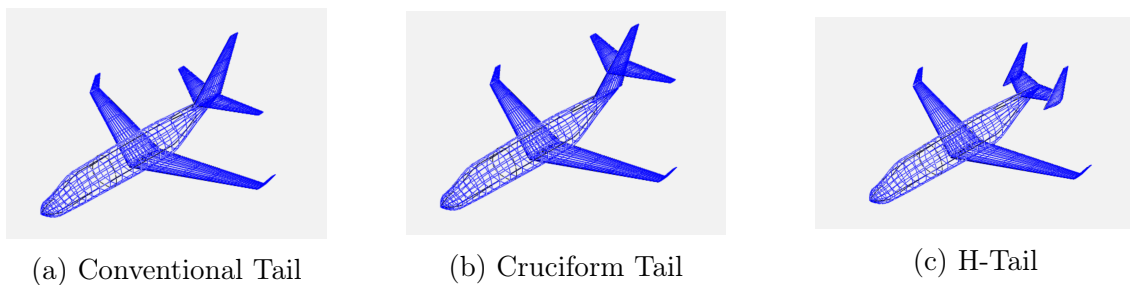


Figure 21: Empennage Configuration Options

Design Number	$\frac{C_L}{C_{D\ cruise}}, M=.85$	$S_{ref}$	$W_{TO}$
Conventional Tail	~ 15	16.35 m <sup>2</sup>	4511 kg
Cruciform Tail	~15.2	16.26 m <sup>2</sup>	4486 kg
H-Tail	~15.8	15.9 m <sup>2</sup>	4415 kg

The H-Tail – as well as the T-tail configuration, which is not shown – both have a reduced total surface area due to the endplate effect; this reduction in area corresponds to a reduction in drag as shown by the increased aerodynamic efficiency, validating not only both tail configurations but also both high and low wing configurations, since the high wing design requires a T-tail. However, as discussed previously the low-wing H-tail design was chosen as a result of mission-driven considerations.

## 5.3 Third Iteration: [2D CFD Analysis and USB Redesign]

Given the iteration in OpenVSP, 2D CFD analysis was performed primarily to refine the effects of upper surface blowing and nacelle-wing interaction and design. The main areas

of concern were the position of the wing relative to the outlet exhaust of the engine, and the design of the engine outlet in order to facilitate upper surface blowing without impeding thrust generation. In addition, consideration was taken into account to reduce shock-waves, and minimize drag where possible. In order to iterate quickly at this point in the design stage, 2D cross sections were taken of the engine and wing configuration at the center of the engine. Although this may not provide an accurate reflection of the 3D blowing effects, doing so allowed us to rapidly iterate and fix major design flaws (especially with respect to shock wave generation).

### 5.3.1 Grid Generation

Grid generation was provided using PointWise A structured mesh was used along the airfoil based on  $Y^+$  calculations at cruising condition. Far field conditions were set, and a momentum boundary condition was set along the engine based on the reported manufacturer thrust.

### 5.3.2 2D Ansys Simulation

After the 2D mesh was generated, viscous simulations were conducted using Ansys. The results of the first iteration are shown below as a means of reference.

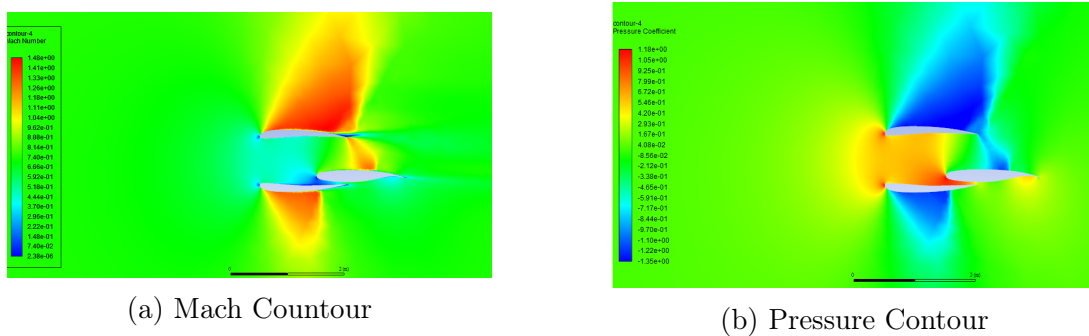


Figure 22: Design Revision 1: 2D Mach and Pressure Countour for Engine + Wing Configuration

The usefulness of the 2D simulations become visibly apparent upon inspection of the above figures. Originally, a small gap was designed in between the bottom of the engine and the bottom of the wing with the goal being to achieve a high pressure area underneath the wing, and a very high velocity flow on top of the wing that would result in maximum lift generation for the given configuration. However 2D simulations show that in fact, the opposite occurred. The flow became choked in the gap and reached sonic conditions, ultimately leading to unnecessary wave drag. In addition, it was noted that engine cross section design needed to be re-designed in order to prevent large shock generation. The most straightforward way to do so was by changing the airfoil shape of the engine cross section.

### 5.3.3 Iteration

Further iteration was performed given design feedback made apparent in preliminary simulations on the following configurations

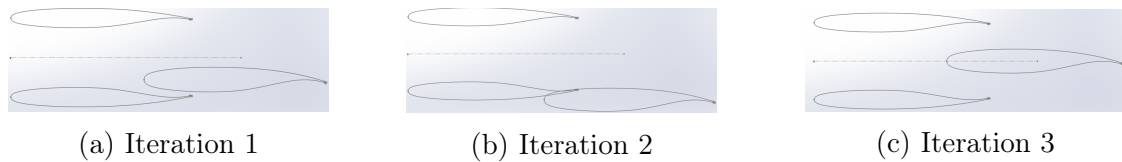


Figure 23: Wing-Engine Configurations

Major changes that were made from the initial configuration to the final configuration include the following

1. **Engine Housing Cross Section Redesign:** In order to eliminate the large shock waves present in the transonic regime, the engine housing was redesigned with a supersonic airfoil.
2. **Gap Elimination:** In order to eliminate local shock waves, the gap between the wing and the bottom of the engine cover were eliminated.
3. **Wing Position:** Wing position was moved farther back relative to the engine in order to provide more surface area for the upper surface blowing effects to cover. However, a trade-off occurred between wing positioning aft of the engine, and the weight penalty that occurred as a result of necessary support structures that would have to be included the more the engine was cantilevered off of the main wing.

A 2D cross section of the final configuration is shown below.



Figure 24: Final Configuration of Engine-Wing

### 5.3.4 Upper Surface Blowing Redesign

With the configuration of the wing nominally set, a full redesign of the engine and upper surface blowing was conducted. First, the engine was modelled accurately based on manufacturers drawings and dimensions. In addition, the engine shell was redesigned to incorporate a supercritical airfoil cross section to minimize shock waves that arose in 2D simulations. A drawing of the engine is shown below

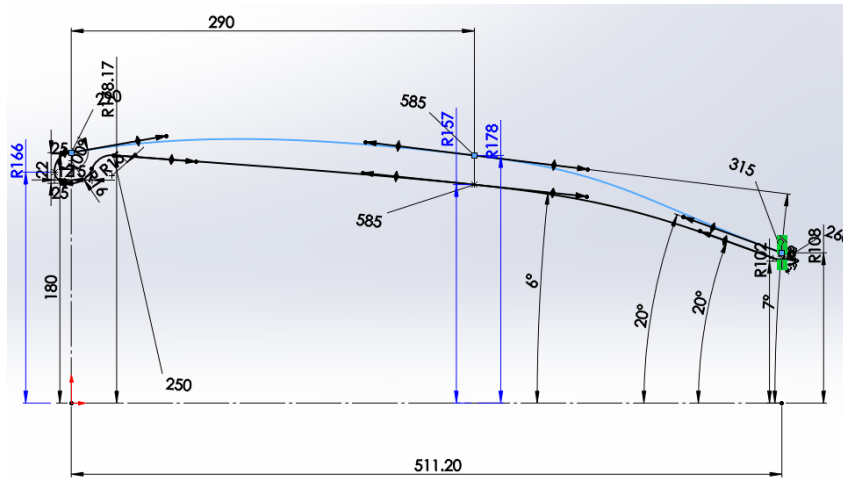


Figure 25: Engine Housing Configuration

With the new engine housing designed, the nacelle was redesigned to support the new wing-engine configuration. A picture of the nacelle design is shown below

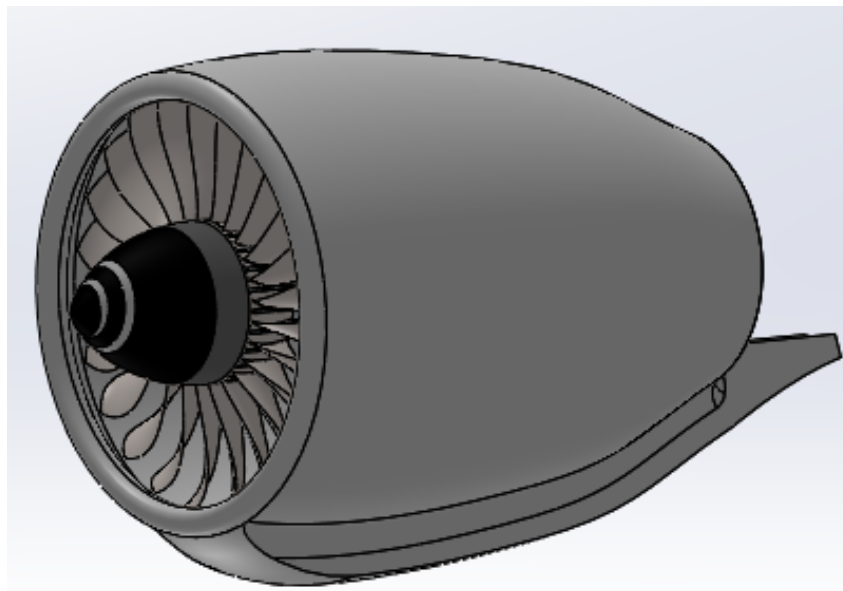


Figure 26: Engine Housing Configuration

Before mounting the engine on the plane, care was taken to ensure that the engine internals would not interfere with the wing, and would reflect the final 2D iteration. Engines on the current configuration of the plane are shown below

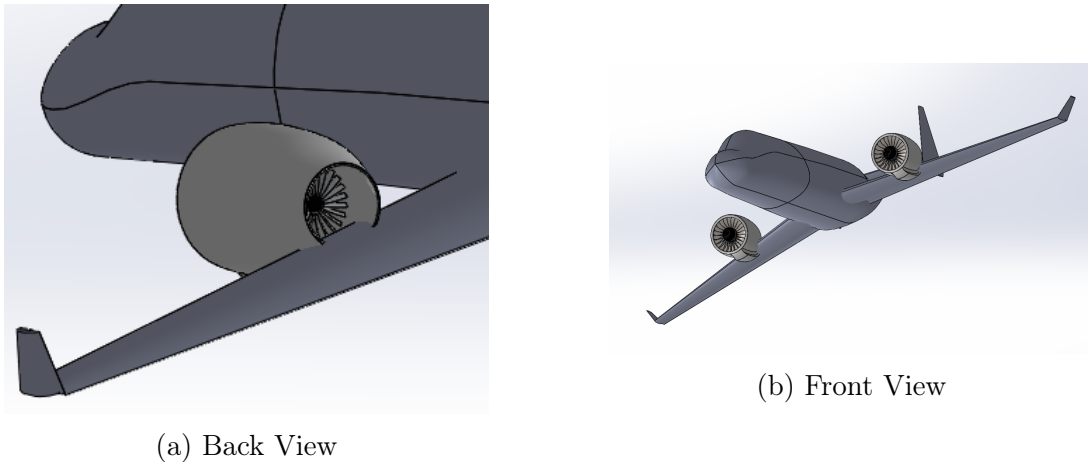


Figure 27: Engine Positioning and Nacelle Redesign

## 5.4 Fourth Iteration: [Full Plane Optimization in SUAVE]

Full plane optimization was conducted in SUAVE. Optimization was conducted Talk about optimization of wing sweep, aspect ratio, etc. Talk about lift coefficient curve fitting with the YC-14, tail configuration, custom multi processing pipeline.

### 5.4.1 Plane Configuration

Custom plane configuration was carried out in the SUAVE Python environment. This included adding all relevant aerodynamic and structural properties to accurately reflect the current design iteration of the plane. However, it is important to note that the the vertical components of the H-tail had to be added in as two separate wings symmetric about the vertical tail. Doing so allowed for more accurate drag calculations (by increasing wetted area).

In addition, a major part of ensuring the accuracy of the SUAVE computational environment was setting up the energy network. The energy network was set up to accurately reflect the two engines that were on the plane. A baseline energy network was provided given manufacturers data, as well as various thermodynamic calculations from NASA.

Finally one of the major customizations that was provided on top of the baseline SUAVE computational layer was modelling the high lift capabilities of the TigerPounce. As a baseline, the high angle of attack lift generation during takeoff and landing was curve fitted from the YC-14 data shown below (scaled by the ratio of wing areas).



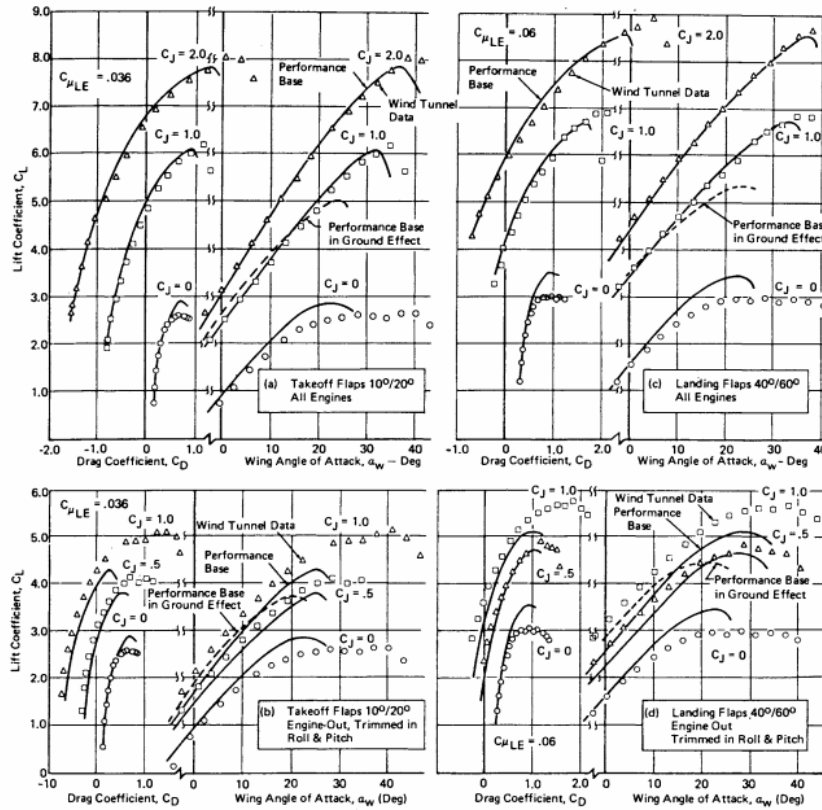


Figure 28: YC-14 Data [Curve Fit for High AoA Reference in SUAVE]

### 5.4.2 Flight Quality Verification

Before conducting full plane optimization, it was necessary to first verify that the plane was modeled correctly in SUAVE at this point in the design phase. This was achieved by first designating a characteristic mission segment based on the design requirements. Subsequent simulation was run, and the results are plotted below.

An extremely important assumption in calculating range as well as geometric properties of the design is the ratio of lift to drag at cruise. The following graph corroborates the approximate lift to drag ratio of cruise of 15.5.

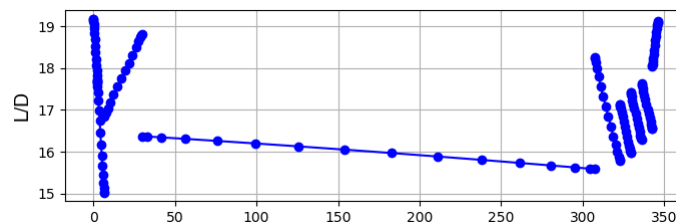


Figure 29:  $\frac{L}{D}$  vs. Time, SUAVE

In addition, to verifying  $\frac{L}{D}$ , thrust specific fuel consumption as well as weight as a function of time along the mission segment were also computed and plotted using the SUAVE computing environment. Specific fuel consumption of approximately  $0.74 \frac{lb}{lb \cdot hour}$  is corroborated. A thrust model was computed based on available engine data, and a plot of

thrust specific fuel consumption is plotted below. The graph shows fuel consumption of around 0.74 for the length of mission.

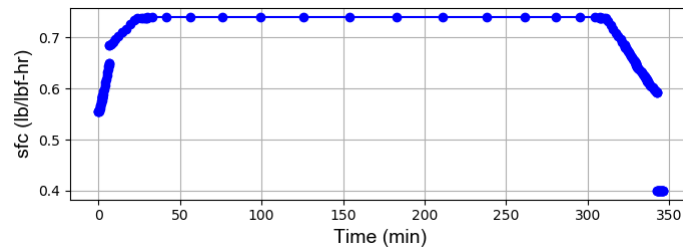


Figure 30: Specific Fuel Consumption vs. Time, SUAVE

The graph of weight as a function of time is also included below. Note that it is highly sensitive to the fuel weight burn (which accounts for discrepancies between final weight of the plane after the mission in the SUAVE environment versus computed final weight using weight fraction estimation methods). However, total fuel burn was within a 10 percent margin of what was calculated based on initial sizing.

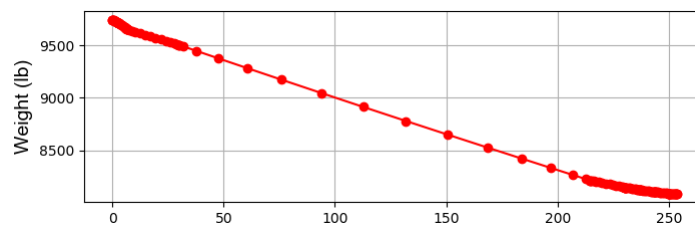


Figure 31: Weight vs. Time, SUAVE

A graph of drag versus time. It can be noted that  $C_{D_0}$  is approximately equal to the value of  $C_{D_0}$  found using VSP Aero. This further corroborates drag assumptions and drag dependent geometric design properties that were relevant to range calculation.

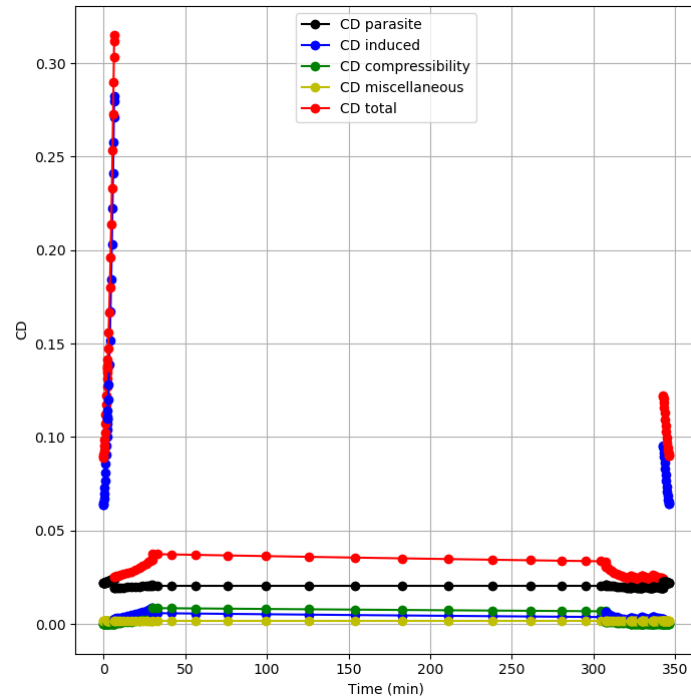


Figure 32: Drag vs. Time, SUAVE

Further, a graph of cruising altitude was produced using the SUAVE computational environment to corroborate that the design achieves the cruising altitude design requirement of 10,500 feet, as well as show general mission climb and descent patterns.

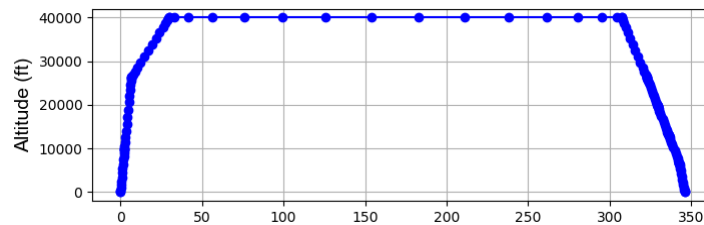


Figure 33: Altitude vs. Time, SUAVE

It is also useful to plot thrust for the duration of the mission in order to visualize altitude effects on thrust. Static sea level required thrust of 1800 lbf per engine is achieved, and it can be noted that the thrust curve drops off similar to manufacturer data, suggesting that the model is accurate for the most part.

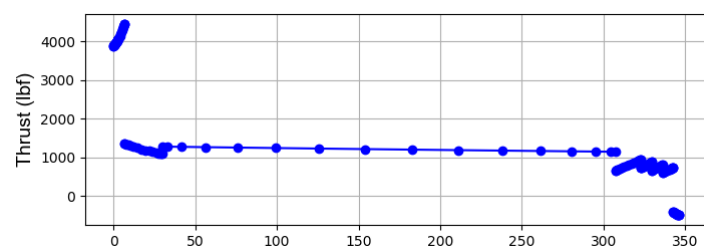


Figure 34: Thrust vs. Time, SUAVE

### 5.4.3 Custom Optimization Framework

In general, SUAVE supports multi-parameter constrained optimization. SUAVE is extremely flexible in allowing the user to minimize some objective under a set of given constraints by iteratively modifying input variables. In general, the SUAVE optimization framework can be described mathematically as

$$\begin{aligned}
 & \text{minimize} \\
 & \quad f(x) \\
 & \text{subject to} \\
 & \quad g_j(x) = 0, j \in \{1, \dots, l\} \\
 & \quad h_k(x) \leq 0, k \in \{1, \dots, m\} \\
 & \quad lb_i \leq x_i \leq ub_i(x) = 0, i \in \{1, \dots, n\} \\
 & \quad x \in \mathbb{R}
 \end{aligned}$$

where  $\mathbf{x}$  is a vector containing  $n$  design variables,  $x_i$ , and each variable is bounded by lower and upper bounds  $lb_i$  and  $ub_i$ . There are  $l$  equality constraints  $g(x)$  and  $m$  inequality constraints  $h(x)$ .

For the purposes of our multi objective optimization, an algorithm was written on top of the SUAVE optimization framework to parallel process the feature design optimization space and deliver optimization results in parallel. The algorithm is detailed below

---

#### Algorithm 1 Parallel Optimization

---

```

1: procedure PARALLEL( $i, o, c, a$ ) ▷ inputs, objectives, constraints, aliases
2:   for input  $\in i$  do
3:      $\text{seg}(i) \rightarrow i_1 \dots i_k$  ▷ segment  $n$  inputs into  $k$  segments
4:   end for
5:    $\text{cartesian product} () \rightarrow i_1 \times i_2 \dots \times i_n$  ▷ compute cartesian product to
   descretize design space into  $n$  by  $k$  elements
6:    $\text{process}()$  for each partition,  $n_1 k_1 \dots n_n k_n$  ▷ spawn process,  $p_i$  for each
   n-dimensional segment
7:   for  $p_i \in \text{processes}$  do
8:

```

$$\begin{aligned}
 & \text{minimize} \\
 & \quad f(x) \\
 & \text{subject to} \\
 & \quad g_j(x) = 0, j \in \{1, \dots, l\} \\
 & \quad h_j(x) = 0, k \in \{1, \dots, m\} \\
 & \quad lb_i \leq x_i \leq ub_i(x) = 0, i \in \{1, \dots, n\} \\
 & \quad x \in \mathbb{R}
 \end{aligned}$$

```

9:   end for
10:  compute  $\min \{f(x)_1 \dots f(x)_n\}$  ▷ find minimum objective
11:  return  $(param_1 \dots param_n)_{min}$  ▷ return suspension corresponding to minimum in
   optimal camber-displacement error design space
12: end procedure

```

---

Note that the algorithm computes the global minimum for the design space under the condition that a global minimum is not found on a partition boundary. This is deemed acceptable, as the probability of such occurring is extremely low. The algorithm was run on an 80 core linux server, and allowed for 10x improvement in run time. This allowed for fast iteration and was crucial to the design process.

### 5.4.4 Results

Optimization results are shown below. Important features of the parameter space are shown in the contour plots below. First, a plot of quarter chord wing sweep versus aspect ratio is plotted

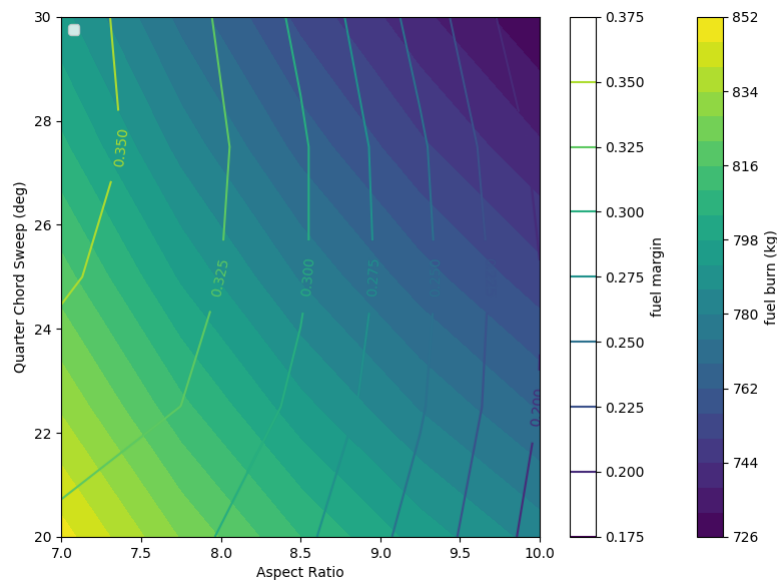


Figure 35: Quarter Chord Sweep vs. Aspect Ratio Optimization Space

Next, a plot of altitude vs. aspect ratio is shown below

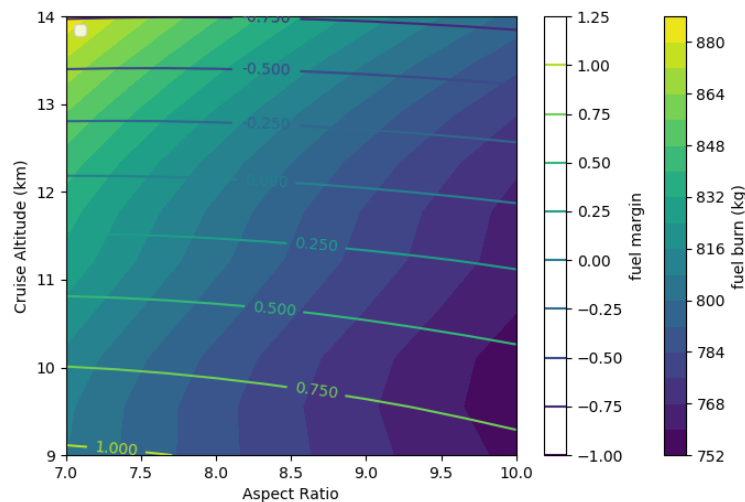


Figure 36: Altitude vs. Aspect Ratio Optimization Space

Finally, a plot of altitude vs. quarter chord wing sweep is shown

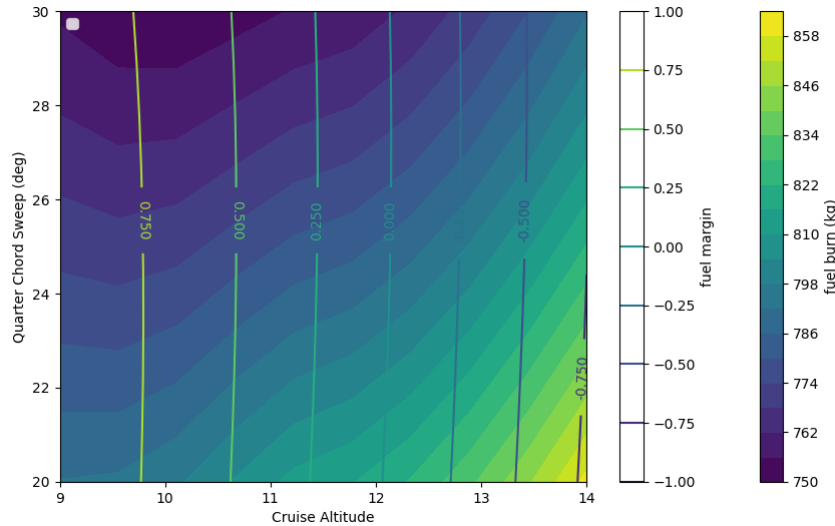


Figure 37: Altitude vs. Quarter Chord Wing Sweep Optimization Space

Thus, based on comparison to historical correlation values found in Raymer as well as values obtained from initial sizing and manufacturers data in the case of the engine, the SUAVE modelling was deemed sufficiently reasonable to proceed with optimization efforts.

For the optimization, a set of constraints, objective function, and variables in the optimization was setup in the SUAVE environment. For comparison, a table of initial and final results of the optimization is included in the table below

Variable	Initial	Optimized	Lower	Upper
Wing AR [-]	8.5	9.4	7	10
Quarter Chord Sweep [°]	25	28.6	20	30
Tip Twist [°]	0	-4	-10	-10
Cruise Altitude [km]	12.19	11.8	9	14

Table 12: Initial Conditions, Optimized Results, Bounds

From these results, we notice a variety of things. First, wing aspect ratio was increased. This was most likely a result of tip vortex generation reduction that resulted in less total drag and overall minimized the objective function of fuel burn. Starting from an initially feasible design, the optimizer tries to obtain a more efficient aircraft with lower fuel burn while meeting the constraints including the gate constraint and the takeoff and landing field length constraints. This results in an increase in the main wing aspect ratio in order to reduce the induced drag.

Next, quarter chord sweep was increased by about 10 %. This may have been a result of the optimizer increasing divergent mach number as a result of greater wing sweep, which affects a major component of the mission segment since the plane was set to cruise in the transonic regime.

In addition, wing twist was decreased from the initial value of 0 to a final value of -4 °. Although this may decrease washout effect and ensure wing tip stall delay under high angles of attack, doing so may also reduce potential for lift generation to meet the short takeoff field length requirement. It is noted that the optimizer is sensitive to flap settings,

which would have to be further experimented with in order to refine the optimization effort. For this, little intuition is provided as a basis to evaluate the optimizer performance.

For the optimization, the objective function was set to minimize fuel burn for the given mission segment. This was deemed a reasonable objective since minimizing fuel burn would essentially minimize weight and drag on the plane, maximize range for a given amount of fuel and in other words ultimately reduce the cost of flight.

Objective	Initial Reference	Optimized
Fuel Burn [kg]	750	694

Table 13: Optimization Objective

The aerodynamic optimization process ultimately saved 56 kg of fuel; since wing area and thrust were fixed as constants during said optimization, the new wing loading, thrust-to-weight ratio, and corresponding constraint plane could be generated as presented below by using the pre-optimization wing area and thrust:

Parameter	Initial	Optimized
Wing Loading	276	272
$\frac{T_o}{W}$	.366	.371

Table 14: Optimized constraint parameters

The constraints used in the optimization are also included for reference. Constraints were based on mission design requirements, to ensure that the plane met the proposed design requirements. Special constraints had to be given in based on the initial sizing estimates and construction of the  $\frac{T}{W}$  vs.  $\frac{W}{S}$  diagram. Note that due to the way SUAVE is set up, the optimization implicitly meets the mission range requirement of 1500 nm.

Constraint	Bound
Design Range Fuel Margin [-]	$\geq 1E^{-6}$
$\frac{T}{W}$ [-]	Constraint Plane Analysis Curve
$\frac{W}{S} \frac{lb}{ft^2}$	Constraint Plane Analysis Curve
Short Takeoff Field Length [ft]	$\leq 1300$
Short Landing Field Length [ft]	$\leq 1300$
Maximum Throttle [-]	$\leq 1$

Table 15: Optimization Constraints and Bounds

Lastly, a graph showing the current version of the iteration on the  $\frac{T}{W}$  vs.  $\frac{W}{S}$  design space is shown below.

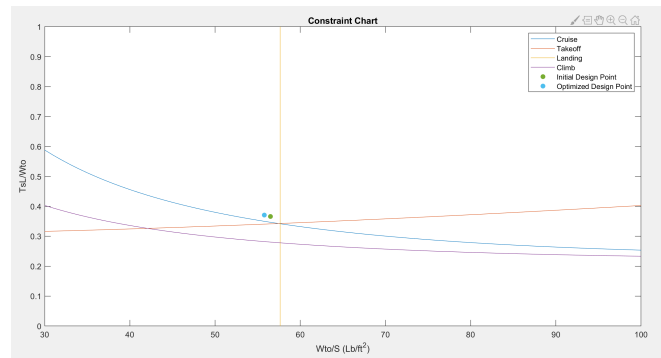


Figure 38: Comparison of initial and optimized design points

## 5.5 Final Configuration: [Verification With 3D CFD Analysis]

For the final iteration of aerodynamic design, a full plane simulation was conducted using 3D CFD Ansys software. Moving to a higher fidelity model for the final airplane configuration was deemed a crucial step in verifying the design properties as well as checking that the design met given design requirements where possible. The details of the 3D simulation are outline below.

### 5.5.1 Mesh Generation

Before standard 3D mesh generation was conducted in PointWise, the geometry of the plane was simplified. This included smoothing of various surfaces along the nacelle in order facilitate mesh construction. A picture showing the simplified geometry is shown below.

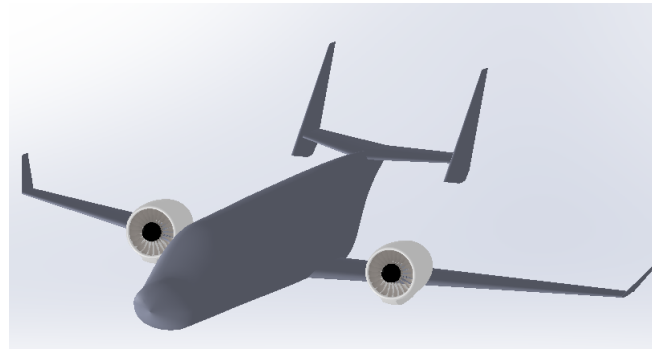


Figure 39: Simplified Geometry for Meshing Process

However note that for final construction, the turbine blades were deleted, and the interior of the engine was simplified to allow for a clean momentum boundary condition to be applied. A figure of the mesh is included below



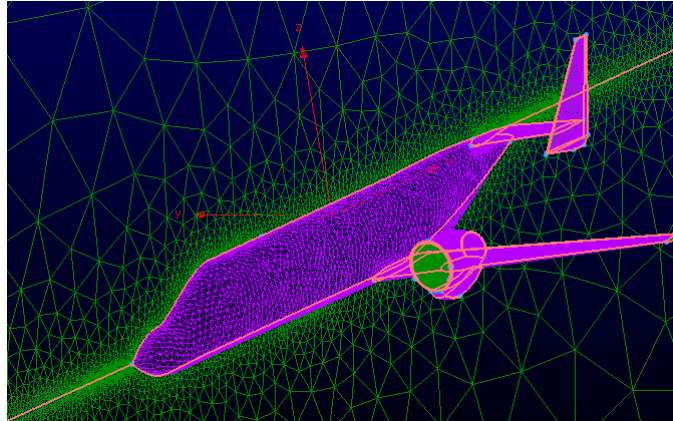


Figure 40: 3D Mesh

### 5.5.2 3D Ansys Simulation

After constructing the mesh, it was exported to Ansys for simulation. Full RANS simulations were run for both takeoff configuration and cruise condition ( $Re = 5\text{million}$ ) at various angles of attack. In Ansys, an overview of major parameter choices is set below:

1. **SA Turbulence Model:** The Spalart–Allmaras was chosen for the turbulence model since flow is assumed to be steady attached flow.
2. **2nd Order Discretization for Flow and Turbulence Variables:** 2nd order discretization was chosen for flow and turbulence variables to obtain higher accuracy. The model still converged under these conditions.
3. **Roe Flux Difference Splitting Scheme:** Roe Flux Difference Splitting Scheme generally is reported to work well for subsonic and transonic regimes and was thus chosen accordingly.

### 5.5.3 Results

Overall, good convergence was obtained with the mesh exported from PointWise, and CFD was run without major difficulties. A brief overview of major goals for 3D CFD as well as results are included below:

1. **Upper Surface Blowing Verification:** One of the major forces driving 3D CFD production was to verify upper surface blowing, and thus verify that the design requirements were met by the given design. Upper surface blowing was modeled by adding a momentum source term from the engine into the model. A graph of the vector velocity field and skin shear stress distribution showing the presence of the upper surface blowing effects are included below. Note that due to the no slip condition, the velocity on the skin is zero, and thus shear stress is shown as a proxy to show the effects of upper surface blowing.

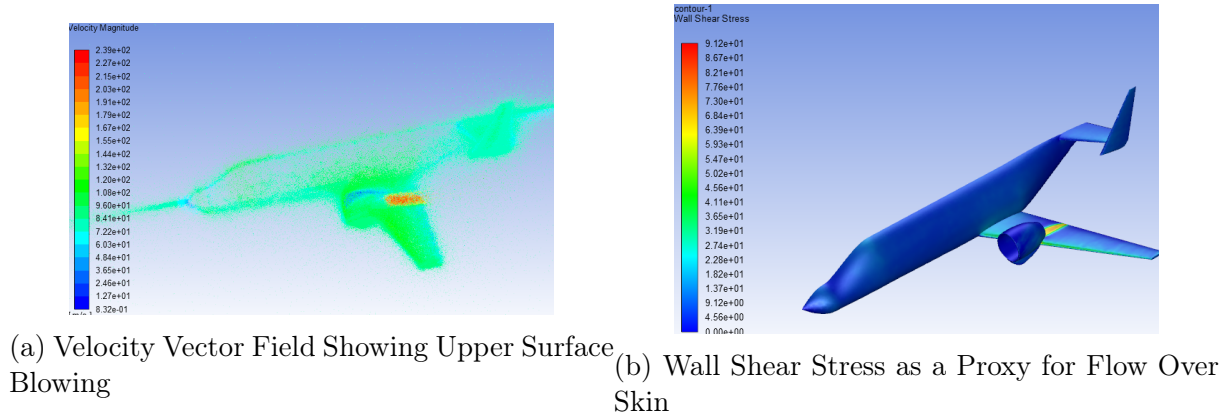


Figure 41: 3D Incompressible CFD

In order to verify upper surface blowing, lift coefficient was simulated for takeoff conditions. A graph of  $C_{L\alpha}$  is shown below

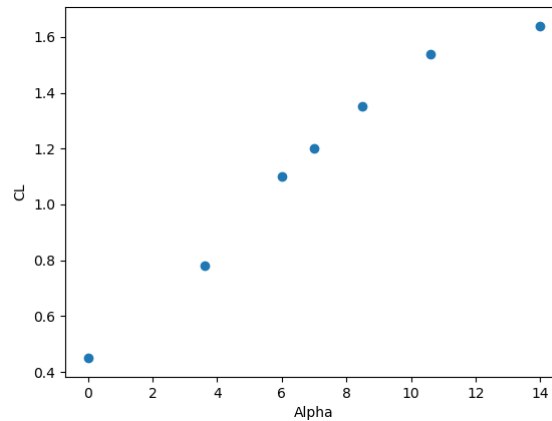


Figure 42:  $C_L$  vs.  $\alpha$

However, note that flap deflection is not included. The  $C_{L\alpha}$  curve obtained, for no flap deflection, matches approximately with the data from the YC-14. Thus, it is reasonable to assume that with flap deflection, the design could achieve lift coefficients similar to the YC-14 of at least 4.3 (and thus meet the STOL design requirement).

- 2. **Cruising Compressibility Effects and Shock Formation:** A major goal of 3D-CFD production was to verify the design at cruise conditions. This included the analysis of compressibility effects and shock formation. CFD was run with both a momentum condition and temperature condition applied to the engine exhaust outlet in order to simulate the effects of upper surface blowing. A graph of both pressure distribution and mach number at cruise conditions ( $M = 0.8$ ) is shown below

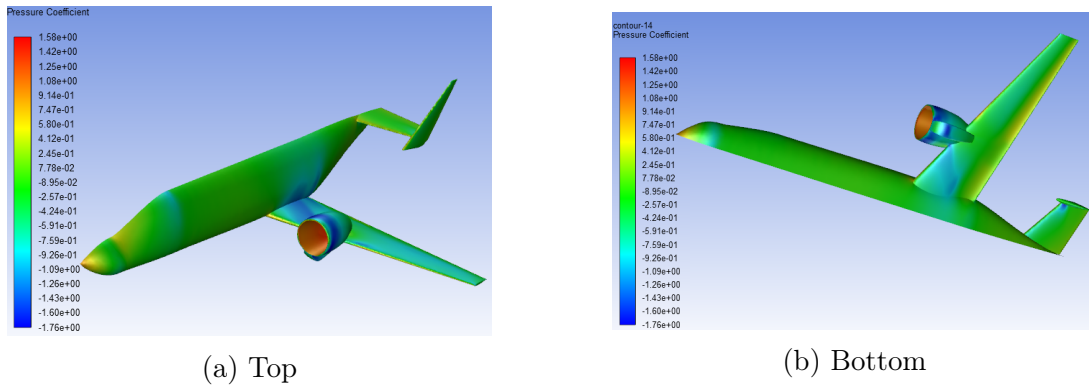


Figure 43: 3D Pressure Coefficient Distribution at Cruise

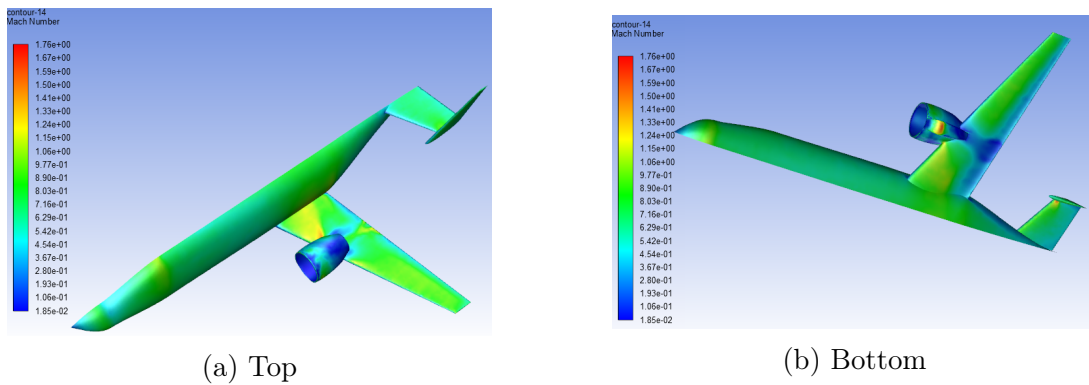


Figure 44: 3D Mach Number Distribution at Cruise

It is observed that nacelle interference with the wing is quite strong, leading to high wave drag production. In addition, the underside of the nacelle leads to strong shockwave production. Future iterations should focus on further iteration of nacelle design, as well as nacelle positioning to minimize wing interaction.

Note that drag results from 3D CFD were excluded from analysis for this report. The mesh was not resolved enough in order to verify accurate drag results, and thus reporting them would result in little value.

## 6 Landing Gear

Raymer lays out a simple method to determine the size of the landing and tires for an aircraft in his book "Aircraft Design: A Conceptual Approach." The process is broken down into four main steps: choosing a landing gear arrangement, sizing the tires, sizing the shock absorbers, and, if needed, developing the kinematics of retracting the landing gear. Each of these is briefly laid out below. For details see Chapter 11 in Raymer's book. All of the information and equation in this section come from Raymer unless otherwise noted. All of the calculations were done in MATLAB using the script called *LandingGear.m* which is visible in a Listing in the Appendix.

### 6.1 Arrangement

There are six major groups of landing gear, most business jets including ours have a tricycle arrangement. This means that there is one piece of gear in front of the center of gravity and two main sets of landing gear behind the center of gravity. The advantages of such an arrangement include the ability to land at large crab angles, good ground visibility from the cockpit, and a flat cabin floor for passengers and cargo.

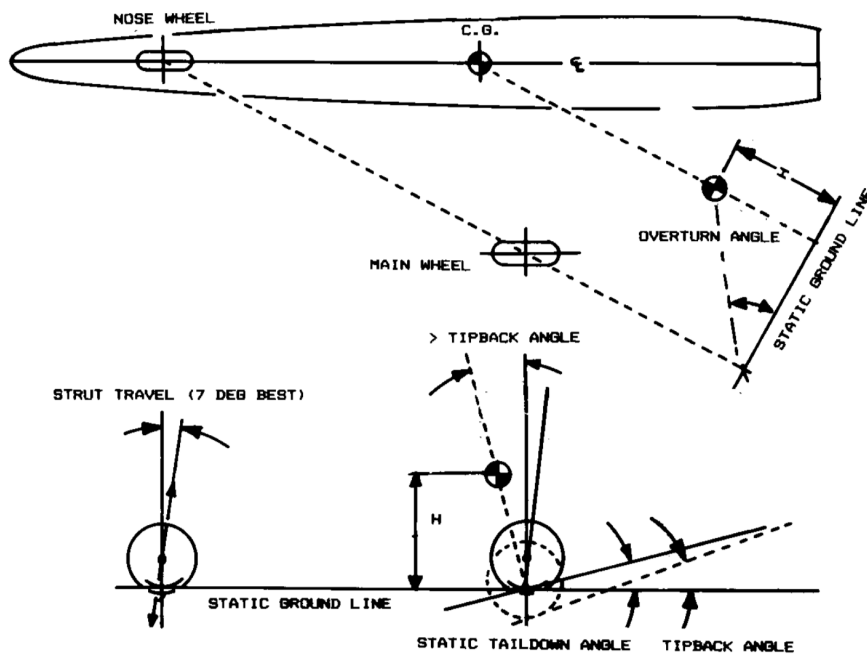


Figure 45: Geometry of Tricycle Landing Gear [3]

Determining the location of tricycle landing gear is more complicated than most of the other systems are there are a lot of things to take into consideration. Figures 45 and 46 show most of the important characteristics when determining the location of the wheels in relation to the aircraft's center of gravity and tail position. The process for determining the location of the wheels for TigerPounce went as follows:

1. choose a distance between left and right main gears and the height of the center of gravity above the ground
2. Pick a tip back angle

3. Put wheel into OpenVSP and check static tail down angle (angle from bottom of wheel to lowest point of tail)
4. If the tip back angle is roughly  $1^\circ$  larger than the static tail down angle then move on, else return to step 1
5. Pick an overturn angle (should be less than  $63^\circ$ )
6. Calculate position of nose gear
7. Check that the nose gear has between 8% and 16% of the static load at all c.g. locations, if so done, if not return to step 5

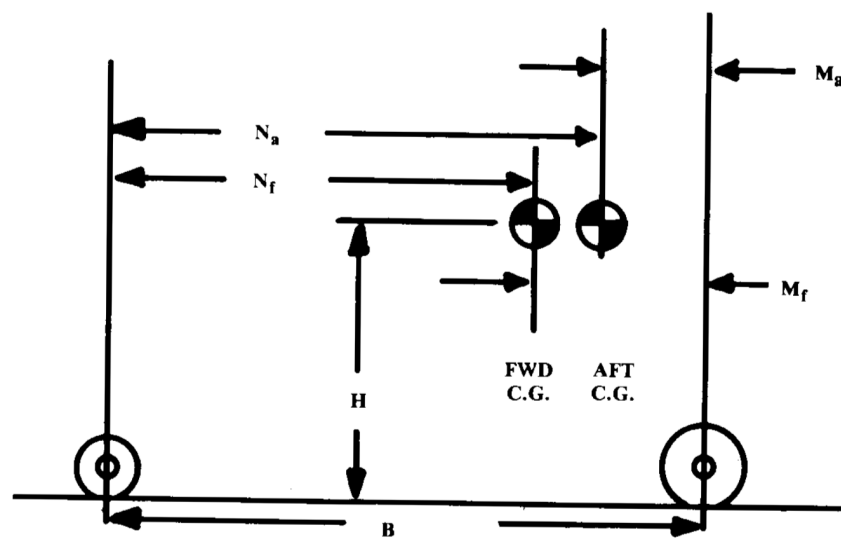


Figure 46: Wheel Loading Geometry<sup>[3]</sup>

The ratio of loading for the gear (step 7) and the exact load felt by the gear can be determined by the ratio of distances between the center of gravity and the landing gear. As seen in Figure 46, the maximum load on the main gear happens when the center of gravity is at its aft-most location while the maximum static load on the nose gear occurs with the fore-most c.g. location. The static load is calculated to be the MTOW of the design plus an additional 25% in order to account for future design changes. The maximum load that will be felt by the nose gear is actually the static load plus a dynamic load from the aircraft wanting to rotate forwards when breaking. Table 16 shows the final values for the arrangement of TigerPounce's landing gear, refer to Figures 45 and 46 for variable name meaning.

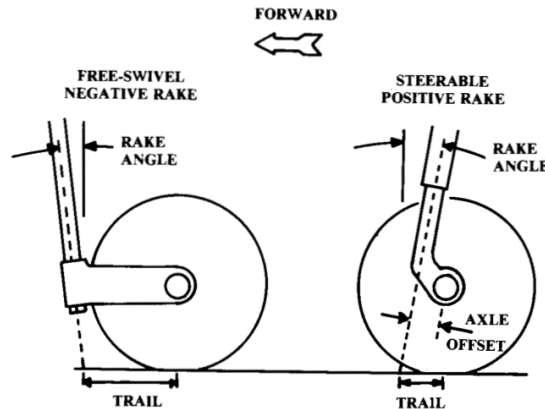


Figure 47: Geometry of Nose Wheel [3]

Parameter	Final Value
MTOW	9733 lbs
Dist. b/w Main Gear	4.4 ft
H	4.5 ft
Tip Back Angle	19°
Static Taildown Angle	18.2°
Overturn Angle	41.8°
B	18.8907 ft
Rake Angle	15°
Trail	1.79 in
c.g. Fore-most (from nose)	23.924 ft
c.g. Aft-most (from nose)	25.354 ft
$\Delta_{c.g.}$	1.43 ft
$M_a$	1.5495 ft
$N_a$	17.3412 ft
$M_f$	2.9795 ft
$N_f$	15.9122 ft
Max Load Main	11 168 lbs
Max Static Load Nose	1919 lbs
Max Load Nose	2820 lbs

Table 16: Landing Gear Arrangement

The rake angle and trail are shown in Figure 47. Our aircraft will allow the pilots to have control over the nosewheel using the rudder pedals. This is called a "steerable" nose wheel – as opposed to a swivel wheel that just turns freely – and this requires a positive rake angle. For small aircraft this is often close to 15° whereas it would be 7° for a FAR 25 aircraft. Similarly, the trail needs to be smaller than the radius of the wheel and on the suggestion of Raymer is set to be 20% of the radius.

## 6.2 Tire Sizing

In order to determine the size of the tires one must first pick how many to put on each strut. Our system utilizes two wheels per strut, allowing both smaller wheel sizes as less weight is carried by each wheel and helping to keep control of the aircraft in case one wheel loses tire pressure. This is in keeping with the FAR 23 requirements for landing

gear. With this in mind it is possible to determine the maximum load each tire will feel, Table 17 shows these values for the TigerPounce. It also includes the maximum braking energy needed based on the stall speed of the fully loaded aircraft. This is because the wheel must be large enough to house brakes big enough to dissipate the kinetic energy of braking.

Parameter	Final Value
Main Wheel Max Load	2792 lbs
Nose Wheel Max (Static) Load	959 lbs
Main Wheel Max (Total) Load	1086 lbs
$V_{stall}$	167 $\frac{ft}{s}$
Kinetic Energy Braking	$4.22 \times 10^6$ ft lbs

Table 17: Wheel Loading and Braking Force

The loadings in the above Table will help to narrow down the wheel size, but equally as important is having the diameter of the hub large enough for brakes. Comparing the braking energy per wheel of our aircraft with the plot in Figure 48, from Raymer, it shows that our braking is on par with the expected value for small jets and that we need a hub size of somewhere between 8 and 10 inches.

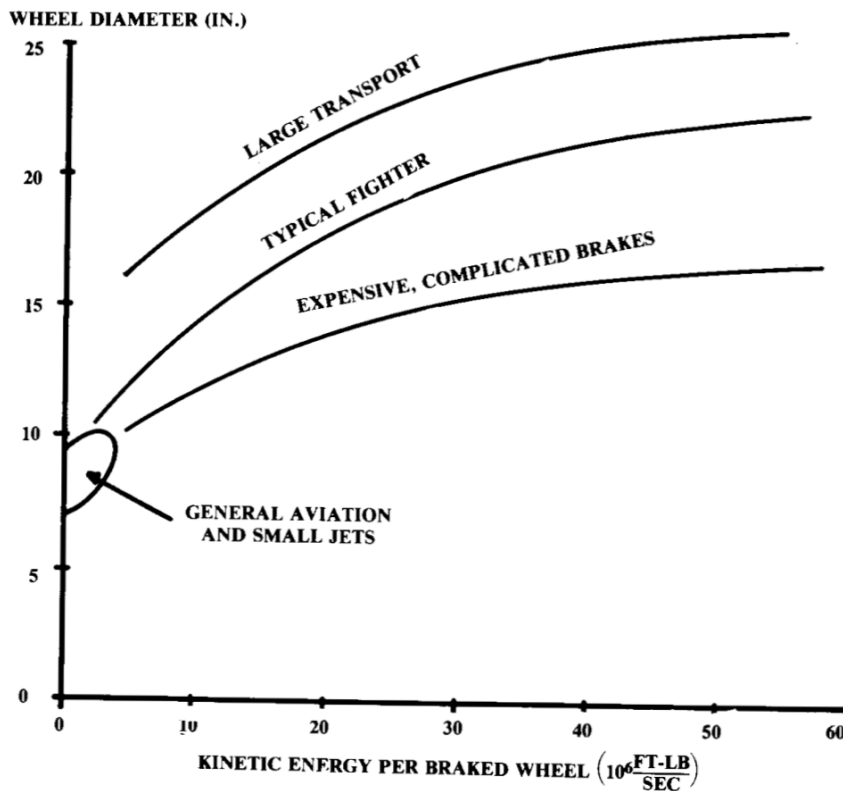


Figure 48: Kinetic Energy Per Brake [3]

With all of these parameters determined, the next step is to consult a landing gear manufacturer’s catalogue to find the appropriate wheels. We decided to use Type VII wheels as they are what most classical jets use and are able to hold up to the high landing speeds of a CESTOL aircraft. Looking at a catalogue from Goodyear Aviation, we chose a set of 18x4.4 wheels specifically with part number 186F03-2. [1] The specifications of these tires are in Table 18.

Parameter	Value
Tire Ply	10
Speed Rating	210 MPH
Rated Load	3550 lbs
Max Brake Load	5320 lbs
Max Inflation	180 psi
Inflated Diameter	17.9 in
Width	4.45 in
Rim Diameter	10 in
Rim Width	3.5 in

Table 18: Goodyear Aviation 186F03-2 Tires [1]

Due to a limited number of sizes we had to go with slightly larger wheels to make sure the tires could take enough load, but it means there is extra room for growth if needed. Further, we used the same wheels for the entire aircraft, but if they lead to too much drag on takeoff it would be possible to downsize the nose wheels. We also wanted to make sure that we had a hub diameter on the larger end allowing the brakes would be on the larger end which would help dissipate the energy required by STOL aircraft more easily.

Once the tires have been determined it is possible to figure out what level they will be pressurized to and what their rolling radius will be under a static load. Figure 49 does a good job of showing how under load the tire will deform. Using the weight of the aircraft one can determine how much area of the tire will be on the ground ( $A_p$ ) and therefore what the rolling radius ( $R_r$ ) is for a given tire. In our case the rolling radius of the tires was 7.9 inches for both the nose and main gear as they use the same tire.

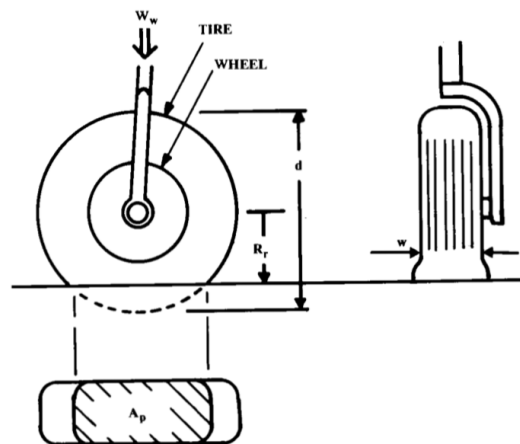


Figure 49: Tire Deflection Under Load [3]

### 6.3 Shock Absorbers

Sizing the shock absorbers is quite simple once a particular type of shock absorber has been chosen. There are a lot of different options when it comes to landing gear, but most jet aircraft, including the TigerPounce, use Oleo (short of oleopneumatic) shock-struts. We utilize the simplest kind of oleo which is called an "oleo strut" as the shock absorber itself acts as the strut for the landing gear. This helps save space but leads to more difficult and slower maintenance.



Calculating the size of the oleos is done by first determining the stroke length. This is done by determining how much energy needs to be absorbed by the oleo and then using an efficiency factor to estimate the length. For calculations a efficiency factor of .65 was used (the worst case for simple oleos to ensure the length is long enough) and a tire efficiency of 47% – the deflection of the tire on landing also helps absorb energy. The most important part of the oleo design for this aircraft, or any STOL aircraft, is to account for the extra vertical velocity that a STOL airplane carries during landing. A vertical velocity of  $15 \frac{ft}{s}$  is used instead of the usual  $10 \frac{ft}{s}$ . And, based on FAR 23 requirements, the calculations are done for an aircraft landing at MTOW. This may seem odd but it means that the aircraft can land right after takeoff if some sort of emergency arises – the requirement is being able to land at 80% of MTOW, but being a business jet we wanted to make the safety of our customers paramount. Once the stroke length is known it is easy to calculate the total length of the oleo system, including the extra chamber that carries the pressure, and from there compute the diameter. The stroke length has to take into account the fact that the tire will be traveling at an angle since the struts are not aligned vertically. Table 19 shows the final size of our nose and main oleo strut shock absorbers.

Parameter	Final Value
Main Oleo Length	2.9540 ft
Main Oleo Diameter	2.9891 in
Main Oleo Stroke Length	1.1172 ft
Nose Oleo Length	2.8916 ft
Nose Oleo Diameter	2.1240 in
Nose Oleo Stroke Length	1.1172 ft

Table 19: Oleo Struck Shock Absorber Size

## 6.4 Kinematics

The last part of designing the landing gear was to design the system that retracts the gear on takeoff and re-extends it on landing. This was done using simple bar linkages and didn't involve any calculations. Instead, all the work was done in CAD with drawings being made to show where the linkages would rotate. The retraction system for both the nose and main gear can be seen clearly in Figure 50. The nose gear utilizes a single hydraulic piston (purple) to pull the landing gear up, while the main gear uses a hydraulic system (purple) with a system of linkages (yellow) in order to take up less space than a just a hydraulic piston would. In both cases the gear retracts forwards so that in the event of a hydraulics issue the gear would likely be pulled out to a locked down position by the flow of air along the belly of the aircraft. The downlock and uplock are both internal split collect locks that are often found in hydraulic systems. The integration of flaps for the landing gear is left to future iterations of the TigerPounce as it moves towards a later stage of design.



(a) Nose Gear

(b) Main Gear

Figure 50: Landing Gear with Color Code and Kinematic Drawing

Figure 50 shows all of the key pieces of the landing gear. Apart from the ones mentioned above, the green portions are the oleo – with the thinner part being designed to retract into the longer outer part – the red parts are the torque links that keep the lower half of the gear attached when the aircraft is not on the ground, and the black and gray parts of the wheel and tire. This figure shows the gear fully extended as it would be just before or after takeoff.

### 6.5 CAD Models

This section shows what the landing gear looks like both as a stand alone subsystem and connected to the aircraft. The models are done parametrically so that it will be easy to modify for future design iterations. Figure 51 shows all of the parameters and equations that are used in creating the models.

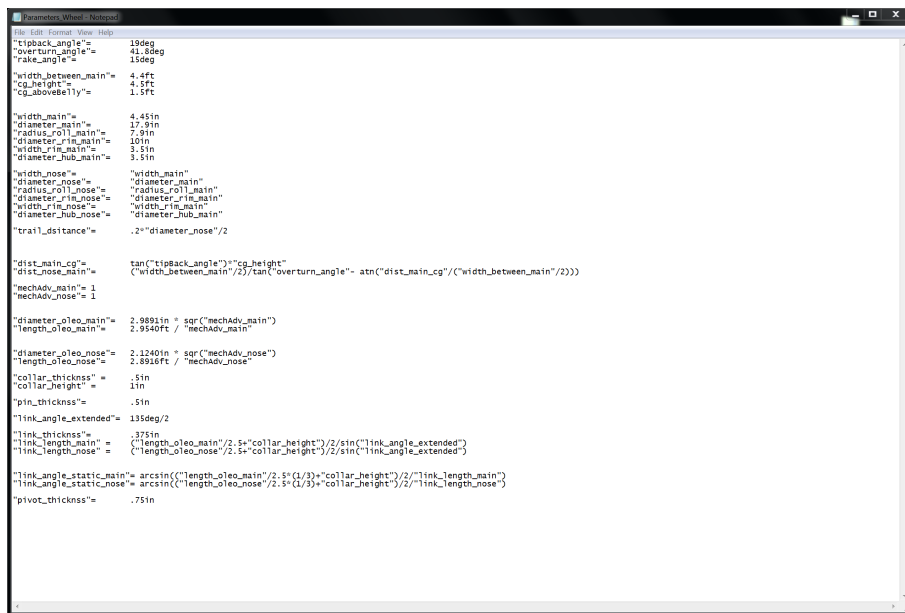


Figure 51: Parameters of Landing Gear CAD Models

Figures 52-55 show what the landing gear looks like in a static configuration both on its own and mounted to the aircraft. Both the oleos and the wheels are in their deformed position, although the oleos are only 2/3 retracted as they have to absorb more energy on landing than they do while static.



Figure 52: Side view of Landing Gear in Static Position

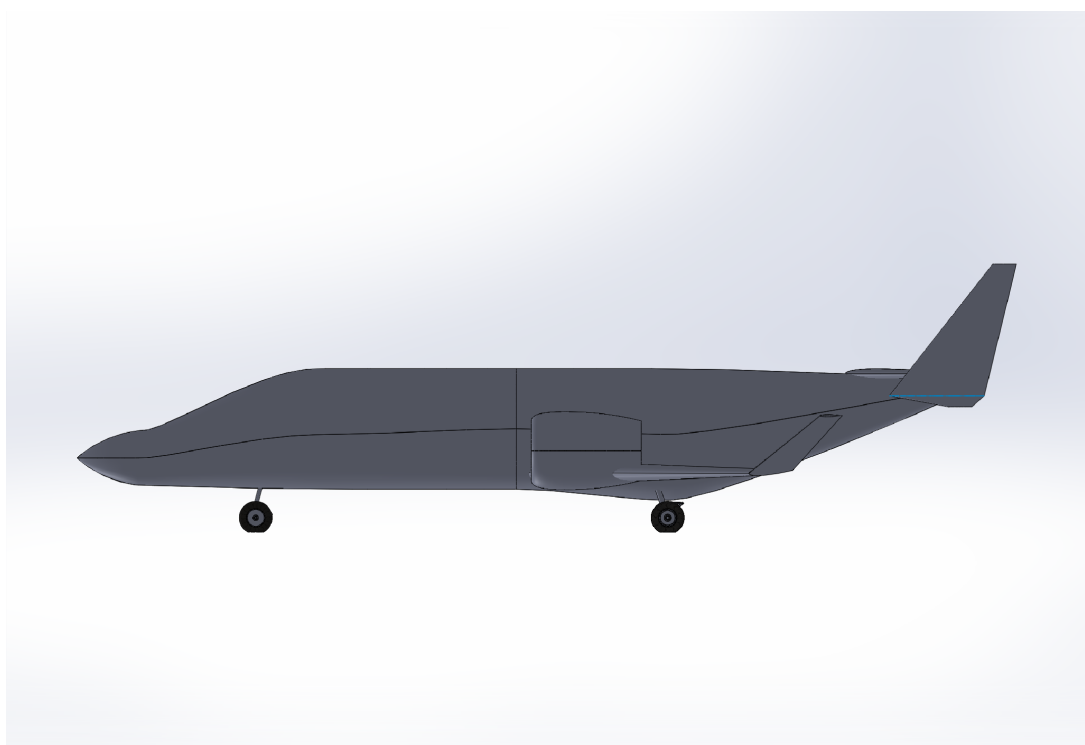


Figure 53: Side view of Aircraft with Landing Gear in Static Position

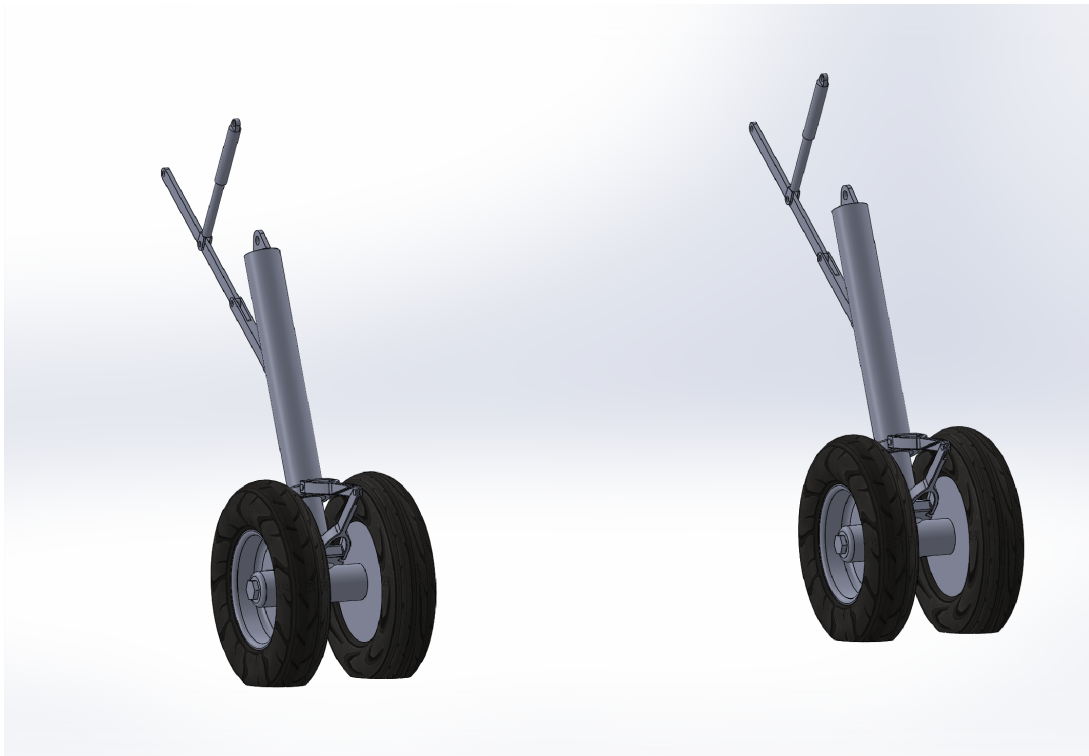


Figure 54: Isometric view of Main Landing Gear in Static Position

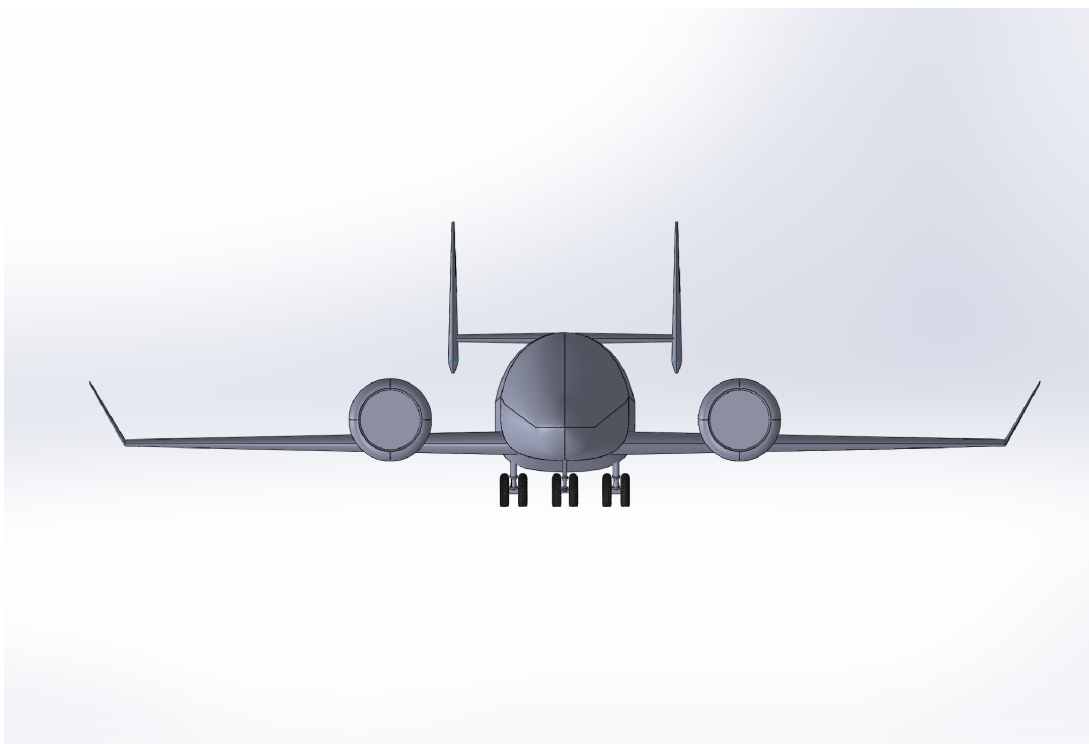


Figure 55: Front view of Aircraft with Landing Gear in Static Position

Figures [56](#) and [57](#) show where the gear is located upon retraction. The gear is designed to be fully located within the fuselage in order to allow for maximum possible fuel volume in the wings and to avoid hurting the lifting ability of the wings on takeoff and landing. This is especially important for the TigerPounce since it is a STOL aircraft where every little bit of lift on takeoff is important.

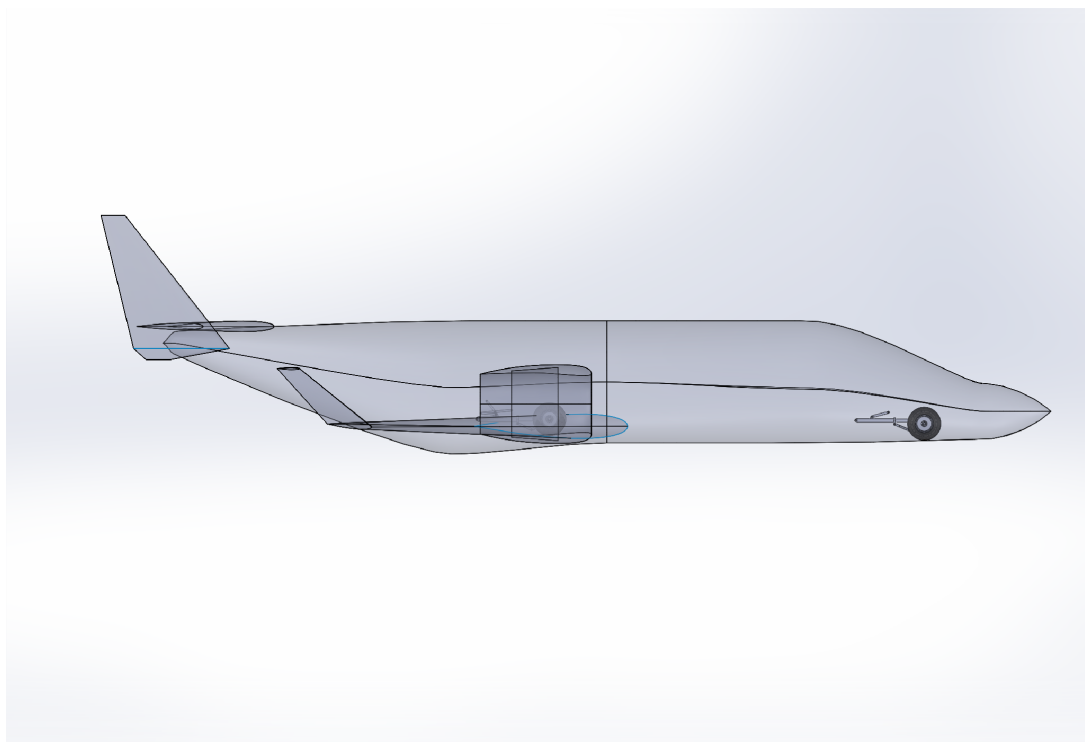


Figure 56: Side view of Aircraft with Landing Gear Retracted

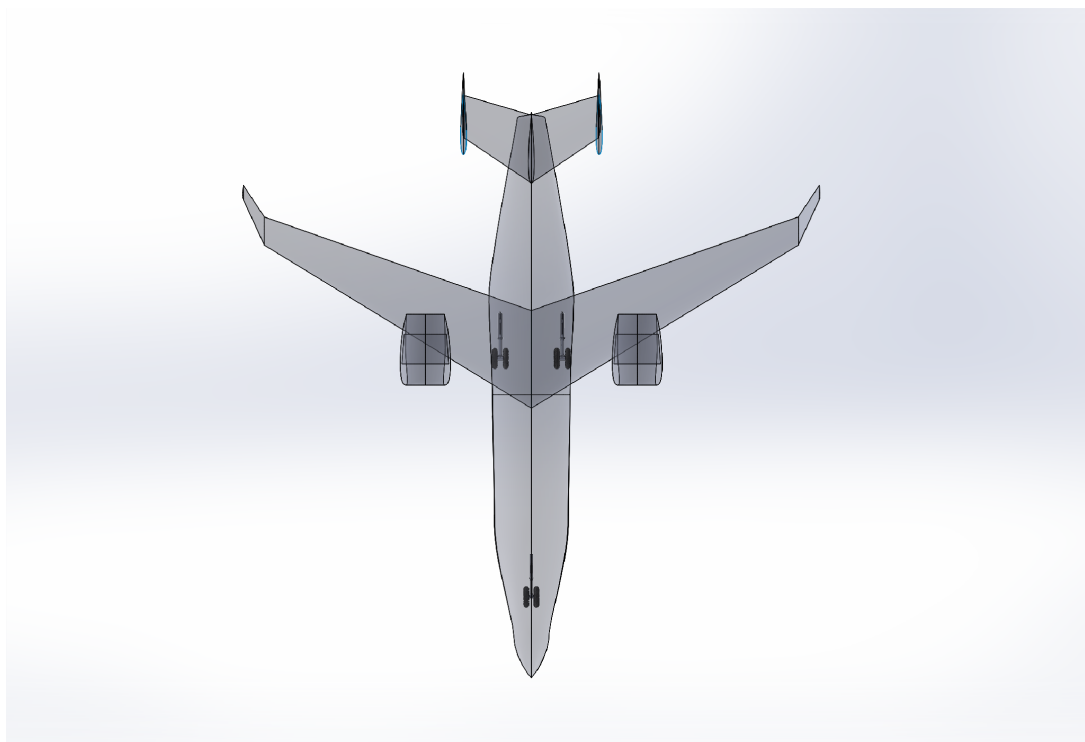


Figure 57: Top view of Aircraft with Landing Gear Retracted

## 7 Structural Design, Analysis and Configuration

Structural design began as soon as the configuration of the aircraft was chosen. The first step was to take the configuration that openVSP generated, and create a solid model using an aluminum structure in Creo. This model can be seen in [58](#).

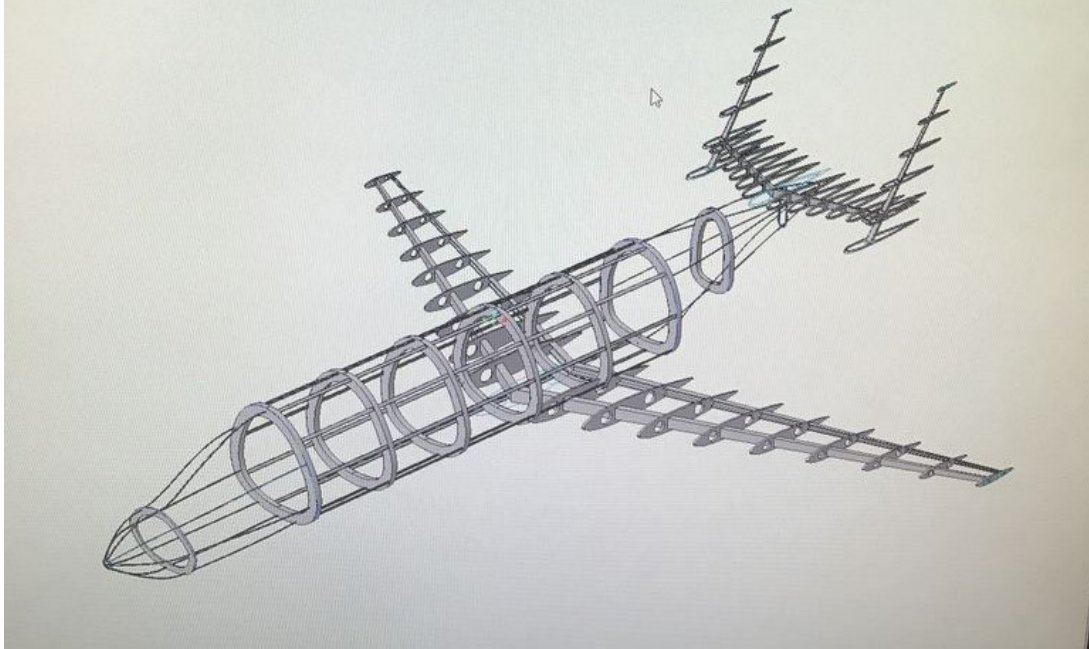


Figure 58: Initial Structure Model

This initial structure was created as a scaling of existing structures. It was further optimized using finite element analysis to create the lightest structure possible that can withstand the loads induced by this aircraft's operation. The structural analysis was broken into four sections to streamline the analysis: main wing, vertical tail, horizontal tail, and fuselage.

### 7.1 V-n Diagram / Gust Envelope

To determine the maximum loads experienced by the aircraft, a V-n diagram with an overlaid gust envelope needed to be developed. There were various calculations that went into developing the diagram show in [59](#), and they can be found in the Appendix [\(12\)](#). The process was centered around two principle assumptions: the maximum and minimum load factors. Note that the minimum load factor was not explored in much detail for the purposes of TigerPounce since its mission renders it irrelevant. The main components of the V-n diagram are the aforementioned load factors, the V-speeds, stall limit curves, and gust limits. They are tabulated below:

V-n Diagram Main Components				
Load Limits	Factor	V-speeds ( $\frac{ft}{s}$ )	Stall Limit Curves	Gust Limits
$n_+ = +2.5$		$V_C = 822.55$	$n_+ = (0.00009031)V^2$	$\Delta n_C = +1.08$
$n_- = -1$		$V_{corner} = 166.38$	$n_- = -(0.00005544)V^2$	$\Delta n_D = -0.65$
		$V_D = 945.93$		
		$V_{NE} = 851.33$		
		$V_S = 105.23$		

And below is the V-n diagram:

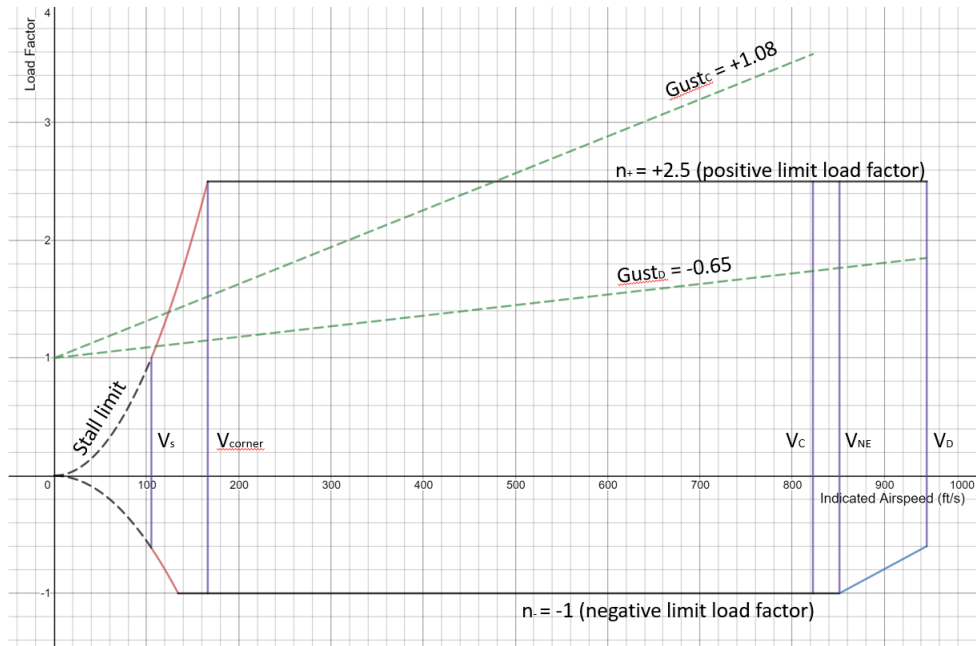


Figure 59: V-n Diagram with Gust Envelope Overlaid

As we can see from this diagram, the maximum positive load factor under gust is  $n_{max} = 3.58$ . This was the critical design principle used to ensure that the aircraft structure could withstand the necessary loading. In the below sections, the successive structural analysis can be found.

## 7.2 Main Wing

The most important features of the main wing are the ribs, the spars, and the skin. To design an optimal wing, serious consideration was put into finding the ideal number of ribs, the ideal skin thickness, and the ideal spar shape. Determining the number of ribs and thickness of the skin is a dual optimization problem with respect to the weight of the aircraft and the structural integrity of the wing: a high number of ribs prevents buckling but incurs a weight penalty, whereas a low number of ribs keeps weight in check but induces buckling more readily. The first principle of finding a solution was minimizing weight. This meant (ideally) arriving at the smallest number of ribs and smallest possible skin thickness that still satisfied the load requirement and didn't buckle under the stress. Various CREO iterations were worked through to complete this task, and it was found that the ideal number of ribs was 9, the ideal skin thickness was 0.040", and the ideal



spar shape was a C-channel. The general structure of the main wing can be seen in the below pictures.

Once the initial structure of the wing was done, the next step was to determine the loads on the wing to setup the simulation. In structural analysis, you normally design for the worst case scenario to ensure the structure can withstand all phases of flight. The loads on the wing were determined to be the weight of the engine, the required lifting force, and the weight of the fuel tanks. The weight of the structure itself also contributes to the load but it was assumed to be negligible in comparison to the other two loads and was not analyzed.

To determine the load due to lift the V-N diagram in figure 59 was used. It was determined that the max load factor is 3.58. The wing was assumed to have an elliptical load distribution to simplify the analysis but provide a better estimate than a triangular distribution. The load profile equation for an elliptical distribution is given by:

$$q(x) = K\sqrt{L^2 - x^2} \quad (15)$$

where K is shape factor, L is the length of the wing and x is the position along the wing [10]. The total lift is given by integrating equation 15. To solve for the shape factor K, the total lift is written in terms of the load factor (n) and divided by 2 because there are two sides of the wing. Integrating the load profile equation across the wing and setting it equal to load factor equation gives

$$K = \frac{2W_{TON}}{L^2\pi}$$

By plugging K back into the lift profile equation the load as a function of position is determined.

$$q(x) = \frac{2W_{TON}\sqrt{L^2 - x^2}}{L^2\pi} \quad (16)$$

To make the simulation run as fast as possible the aircraft skin was suppressed and the load was applied directly to the front and rear spars. In reality, this load would be transferred from the skin, to the ribs, to the spars. The ribs and skin are analyzed in a separate analysis. The last thing to determine was how much of the load should be placed on the front spar and how much on the rear spar. To do this the load was assumed to act at the quarter chord (an estimation of the aerodynamic center). The spars were then assumed to be supports with the ribs acting as a beam between the two. A static analysis was done to determine what proportion of the load went to each support. From this analysis it was easy to see that much more load would be on the front spar than the rear spar in its current configuration. The model was then changed so that the front spar was moved as far forward of the quarter chord and the rear spar moved forward as well. After relocation the front spar was located 9.5 inches forward of the quarter chord and the rear spar 22.5 inches behind. This equates to the front spar bearing 57% of the load and the rear spar 43%.

Next the load from the engine was determined. For our analysis we used a point load to represent the engine. Each engine weighs 525 lbs and is located 7.5 feet or 90 inches from the center line. The engine weight was modeled as a point load and placed at the correct location for the simulation.

Last the fuel tank load needed to be included in the analysis. During the initial design of the aircraft it was decided that the aircraft requires 1660 lbs of fuel. The density of jet fuel is 6.71 lbs/gal which means the fuel tanks need to have a volume of 260 gallons. This includes a little extra volume for trapped fuel. Therefore a 130 gallon fuel tank is needed on both sides of the wing. After a volume analysis it was determined the tank



could fit in the first 40% span of the wing. Because the FEA was beginning to slow down we decided not to solid model the fuel tank. Instead its weight was evenly distributed between the front and rear spars in the first 40% of the span. Had the tank been solid modeled this is what the load would have transferred to.

Once the loads had been determined the final step to setting up the FEA was material assignment. The structure of the wing is to be made from Aluminum 7075. This material has a density of  $.102 \frac{lb}{ft^3}$ , and a tensile yield strength of 73000 psi [11]. A safety factor of 5% was used so stresses higher than 69000 psi were not exceeded when doing the FEA. The first simulation showed that the maximum stress was 49000 psi on the two spars close to the center line as was to be expected. The max stress was below the safety margin of 69000 psi so the size of the spars were decreased a little bit at a time with the simulation repeated until the wing was as light as possible.

The displacement of the wing under load also needs to be considered. To establish a target for displacement a report on wing deflection of the Boeing 787 was used [12]. The report stated that the wing deflected 25 feet vertically. Using the 787 wingspan of 197 feet and the wingspan of our aircraft the deflection was scaled using a proportion. The calculation said that an equivalent deflection for our aircraft is 28". Therefore, while optimizing the spar sizing a deflection of 28" was not exceeded.

After optimizing the wing through many iterations images of the final FEA are shown below. Max stress was 57000 psi, much lower than the 69000 psi safety margin we did not want to exceed. The limiting factor in this design was actually the deformation. Max deformation is 27.2" and is what prevented further shrinking of the spars. If the deflection is later allowed to be more than 28" then the spars could be further decreased to take on about 10000 psi more. Using the mass properties feature in Creo and the density of 7075 Aluminum the weight of the wing including spars, ribs, and skin was determined to be 1228 pounds.

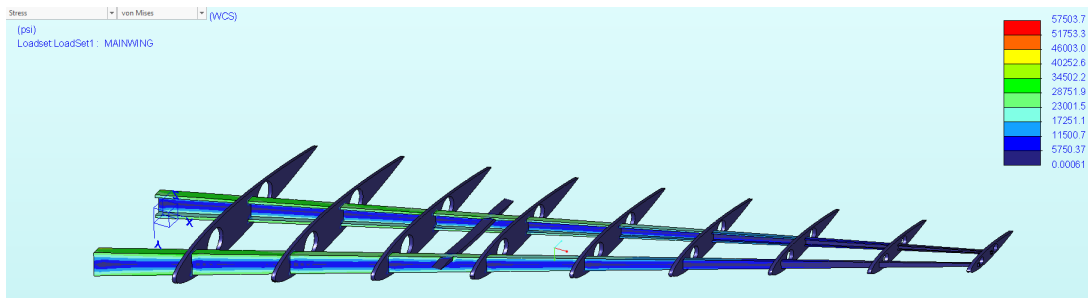


Figure 60: Von Mises Stress on the Main Wing Under Load

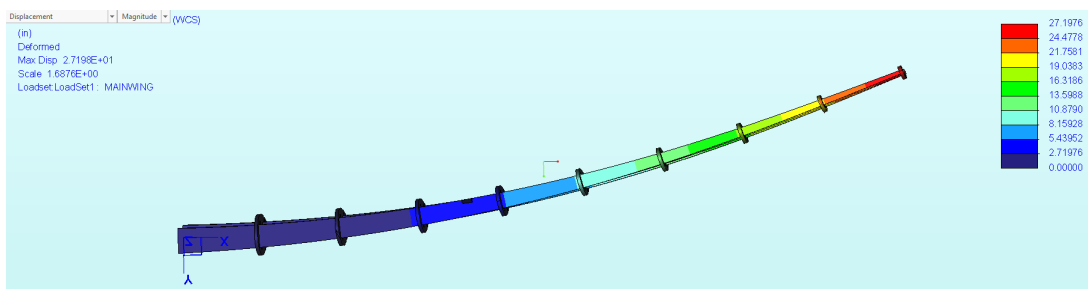


Figure 61: Deformation of the Main Wing Under Load

### 7.3 Vertical Tail

The design of the vertical tail is similar to that of the main wing but with only a single spar. The vertical tail is under minimal load during cruise but loads spike when the rudder is deflected. There is no load from fuel or engines on the vertical tail so the analysis from the main wing was repeated but with only a single load. The load on the vertical tail was again assumed to have an elliptical distribution. Instead of using the V-N diagram to determine the lifting force on the vertical tail, a paper on vertical tail loads was consulted [13]. They stated that the total load for a small transport on the vertical tail is 3,410 lbs. Our business jet is smaller than a the aircraft referenced so we believe this to be a conservative estimate. Our business jet also has two vertical tails due to the H-tail design and therefore in the simulation 1705 lbs was elliptically distributed along the spar of the vertical tail. Aluminum 7075 was assigned as the material, and the simulation ran many times slightly decreasing the size of the spar each time. The cross section at the end of the spar was decreased the most in order to eliminate material where there was little load. Once again deflection under load was the limiting factor and not yield stress. The optimization was stopped with a deformation of 13" and a max stress of 60800 psi. The FEA from the final simulation is shown below. The final structure weight of the pair of vertical tails including spar, ribs, and skin is 176.57 lbs.

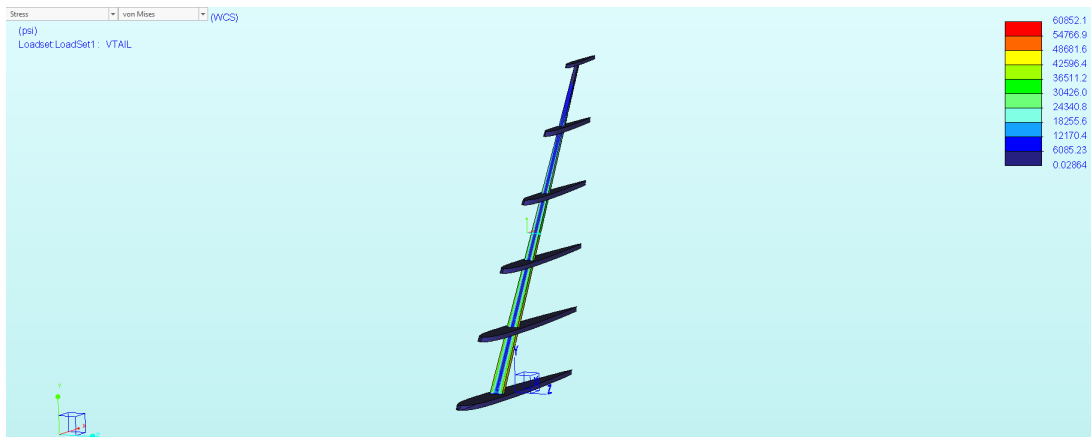


Figure 62: Von Mises Stress on the Vertical Tail



Figure 63: Deformation of the Vertical Tail Under Load

## 7.4 Horizontal Tail

Analysis of the horizontal tail is similar to that of the main wing and vertical tail. Like the vertical tail, there are no loads due to fuel or engines on the horizontal tail. However, due to the H-tail design of the aircraft the horizontal tail must support the two vertical tails. To include this, a point load was added at the end of the horizontal tail equal to the weight of the vertical tail.

The only other load on the horizontal tail is that of the aerodynamic load. During cruise there is aerodynamic load on the tail as lift is generated. However, for structural analysis we are only interested in the max aerodynamic load. This occurs when the elevator is deflected to pitch the aircraft up or down. The largest load the horizontal tail will have to provide is enough force to pitch the aircraft down when the wing is exhibiting its largest load. During the analysis of the main wing we determined max lift was equal to

$$Lift = W_{TON} = 34844lbs$$

The wings center of pressure is 8% of the chord forward of the aircraft's CG. This was determined during the configuration design to achieve the desired static margin. 8% chord equates to a moment arm of 7.2 inches. The moment arm from the CG to the horizontal tails center of pressure (approximated as the quarter chord) is 183.75". By balancing the moment contributions the load on the horizontal tail is determined to be 1,365 lbs. This load seemed to be low so a safety factor of 2 was used and 1,365 lbs was applied to each side of the horizontal tail. The same material properties used on the main wing and vertical tail were used on the horizontal tail.

The final FEA for the horizontal tail is shown below. Although the stress is well below the allowed limit the optimization was stopped because of deformation. We wanted the deformation to remain very low because the vertical tails mount on the ends of the horizontal tail if the HT is largely deformed the VT may become misaligned leading to high drag and instability. Therefore the max stress on the HT is 32000 psi with a deflection of 5". The final weight of the horizontal tail including spar, ribs, and skin is 129.8 pounds.

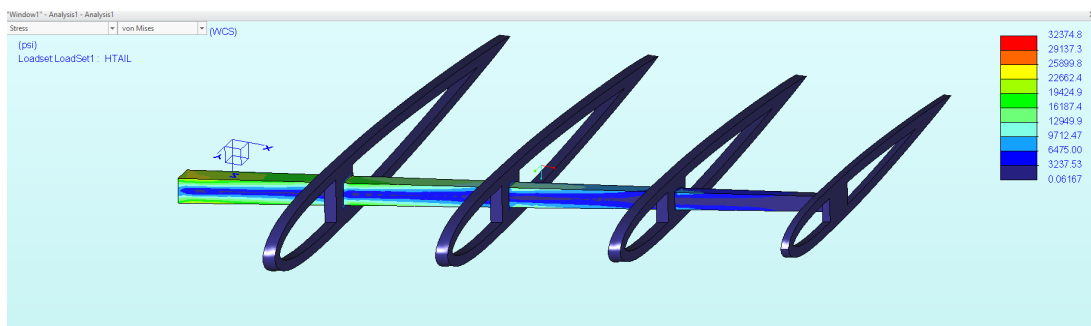


Figure 64: Von Mises Stress on the Horizontal Tail

## 7.5 Fuselage

The analysis of the fuselage was much different than any of the previous analysis. To begin, a solid fuselage model was made using the traditional frame and stringer configuration. The cross sections of the frames are made of an I beam while the stringers use a C shape. Loads on the fuselage are made up of three different kinds[14]. The first load is aerodynamic load including load coming from the wing and tails. The second is the

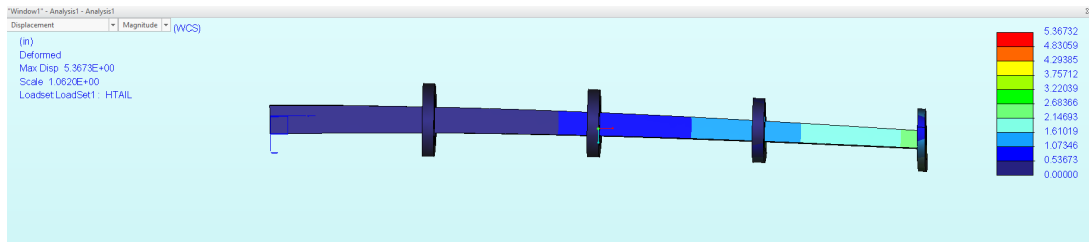


Figure 65: Deformation of the Horizontal Tail Under Load

force of gravity or the weight of the structure itself. This was ignored for previous analysis but must be taken into consideration for the fuselage. The final load to be considered is the pressure induced load. The cabin must be pressurized to enable flight at high altitudes and this pressure induces a load on the skin of the fuselage.

First the aerodynamic loads were added to the simulation this includes 3,141 lbs from the vertical tail, 2,730 lbs for the horizontal tail and 34,844 lbs from the main wing. These loads were applied to the fuselage where each of the components mount to it. The next step was to add loads due to gravity. This includes 1660 lbs of fuel, 525 lbs per engine for two engines, the weight of the main wing and tails, and the weight of the fuselage itself. The final load to be applied is the load due to pressure differences resulting from a pressurized cabin. The cabin is pressurized to 72 kPa, or the pressure at 8,000 feet which is standard for most aircraft. The external pressure during cruise is made up of 13 kPa dynamic pressure and 20 kPa atmospheric pressure. This results in a pressure difference of 39 kPa which was applied using the pressure tool in Creo.

After the simulation was set up in Creo we ran into many problems. The simulation software could not generate a proper mesh because of several flaws in the geometry. The flaws in the geometry are likely because the model was created using a lot of surfaces and splines to fit the complex geometry that had been imported from openVSP. Attempts to fix the geometry was unsuccessful and a new plan had to be devised. We decided to create a new fuselage model similar to that of our fuselage but with less complex geometry. It was decided to only analyze the constant section of the fuselage and to model it as a perfect circle with a diameter as close as possible to that of the original egg shaped fuselage. Frames were solid modeled but stringers and skin were simulated using Creo's idealization feature. In an additional step to speed up the simulation, it was determined to run two separate simulations. The first simulation was done by only applying the load due to pressure differences. This load primarily determines skin thickness and stringer spacing and number. The first simulation verified that a skin of .040" was accurate and the panels were not too large. In other words, we did not need to increase the number of stringers. Next, a second simulation was run with the skin suppressed and applying all of the aerodynamic and gravity loads to the frame of the fuselage. Both simulations can be seen below. The stresses on the fuselage are well below their limit and mainly occur where the stringers meet the frames. This could be further reduced by designing a structurally efficient joint. The deformation of the fuselage is also very small only .5". This was a desired characteristic as excessive fuselage bending creates dangerous and unpredictable flight conditions. Once the structure was found to be sufficient and optimized, the frames and stringers in the original fuselage model was made to match. The mass properties could then be found for the entire fuselage using Creo's mass property calculator. The final weight of the fuselage including frames, stringers, and skin is 1,158 pounds.

By adding the weight of the fuselage, wing, and tails, the total weight of the structure is approximately 2,693 pounds. This appears to be a reasonable weight for the structure at this point in the design. The take off weight was originally calculated as 9,733 lbs.

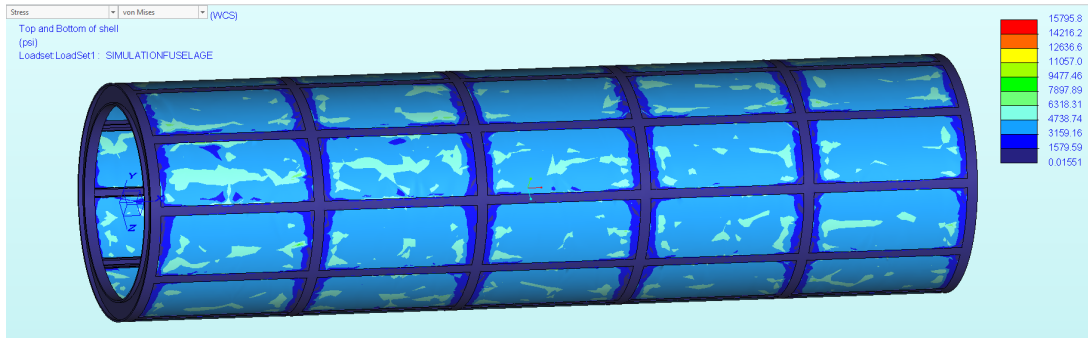


Figure 66: Von Mises Stress on the Fuselage Due to Pressure Load

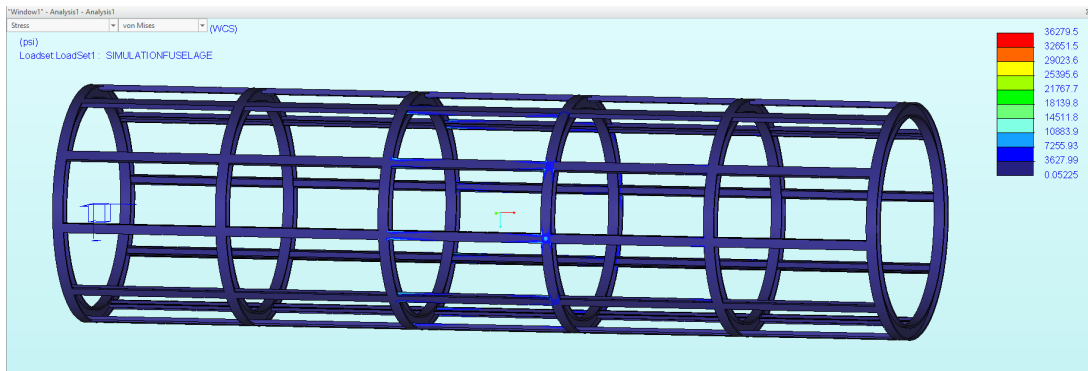


Figure 67: Von Mises Stress on the Fuselage Due to Pressure Load

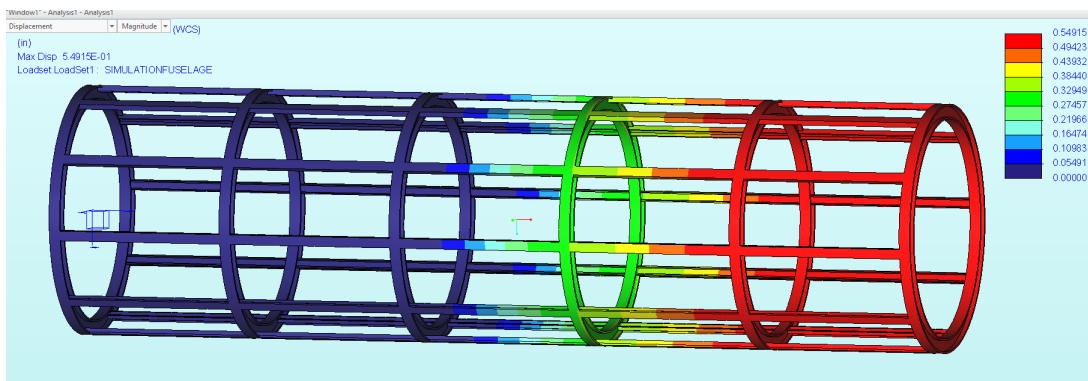


Figure 68: Deformation of the Fuselage Under Load

Subtracting 1,660 lbs of fuel and 2,400 lbs of payload leaves 5,673 lbs. After the 2,693 lbs existing structure is removed 2,980 lbs remain. This remaining weight should be enough for everything else needed to fly such as electronics, floor structure, wiring, fluids, landing gear, doors, and any other structure outside of what has already been designed.

## 7.6 Volume Constraints

Passenger and cargo space specifications were established by competitor design standards [5]. Having established a cross section diameter of 5.5' to be suitable for two seats and one aisle via peer aircraft analysis, the length of the cabin was determined by the need to seat 6 people. The seating configuration is shown in figure [69], confirming the suitability of the arrangement. Furthermore, assuming a cargo density of  $10 \frac{lb}{ft^3}$ ,  $80ft^3$  are necessary to hold the cargo of pilots and passengers. This is easily held in an aft cargo hold of the plane in red in figure [69].

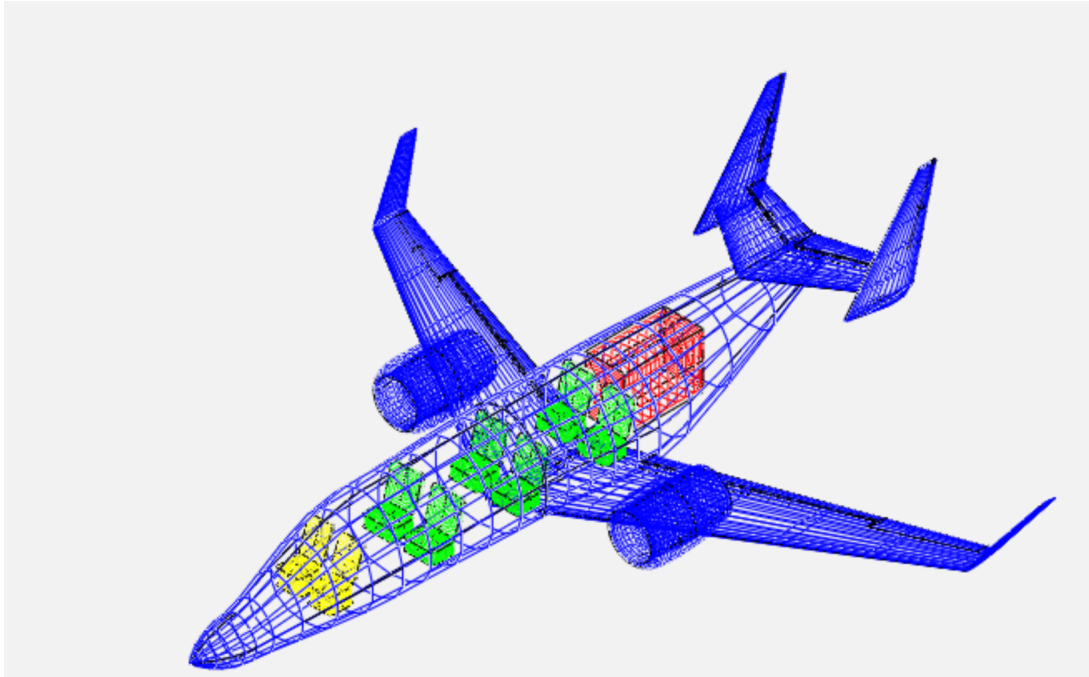


Figure 69: Representation of cockpit (yellow) and cabin (green) seating, cargo volume

## 8 Extreme Thermal Exposure

Although the upper surface blowing provides added lift and allows for a shorter take-off distance, this comes at a cost. Due to the mounting of the motors on top and in front of the wings, the upper surface of the wings is also exposed to the extreme temperature generated by the engine in the form of exhaust. After some calculations it was determined that the engine's exhaust temperatures 787K Kelvin and flows at a rate of 1,334.5 m/s. This is well beyond the thermal limitations of aluminum, the main material selected to form the body of the aircraft. Two solutions were considered to solving this issue, described in the following subsections.

### 8.1 Thermal Barrier Coating

The first solution considered was the use of a ceramic thermal barrier coating similar to those applied to turbine blades within the engines. This is composed of a ceramic coating (which is extremely efficient as a thermal insulator), a bond coating, and then the material selected to provide the strength required for the structural stability of the aircraft. A stream of cooling air is then directed under the thermal barrier coating and used as a heat sink.

After extensive research, it was determined that there was a lack of literature available and a lack of expertise required to apply this system to the aircraft wings. The effects a system like this would have on the structural support of the aircraft was inconclusive. An additional flaw of this solution was the added weight that this would provide, detrimental to the aircraft's performance.

### 8.2 Titanium Aluminide

The second solution to the thermal issue was replacing the aluminum with another metal with a higher thermal resistance and maximum service temperature above that of the engine exhaust. CES EduPack was used to explore metals and alloys with the desired



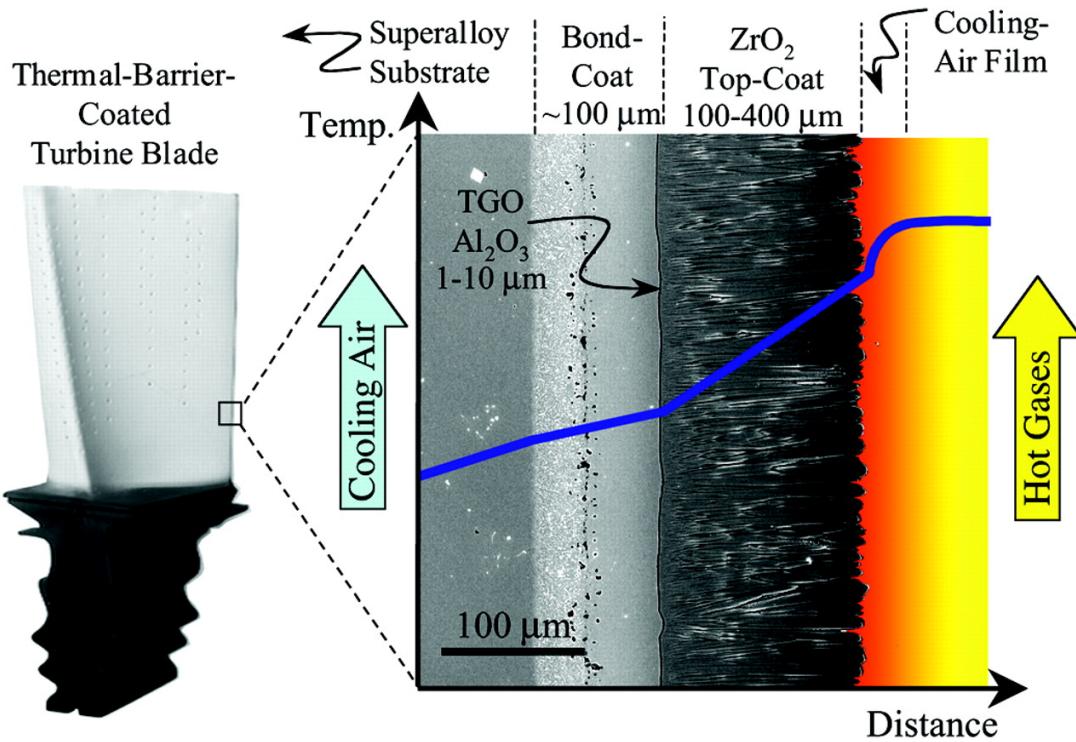


Figure 70: Thermal Barrier Coating, a thermal protection system commonly applied to turbine blades within the engine housing.

qualities, keeping weight in mind as a parameter. The best material found to replace aluminum was Titanium alpha alloy (Ti-24Al-11Nb). A data sheet outlining the properties of the alloy is included in the appendix as a reference. With a density of  $0.165 \text{ lb/in}^3$  this alloy was the lightest material that met all of the parameters to withstand the exhaust heat. Given that the price of this alloy can range from 26.7-27.1 USD per pound, thermal analysis was conducted on CREO and CFD to fully understand the temperature gradient throughout the wings. This allowed us to optimize the design to include as little of the alloy as needed.

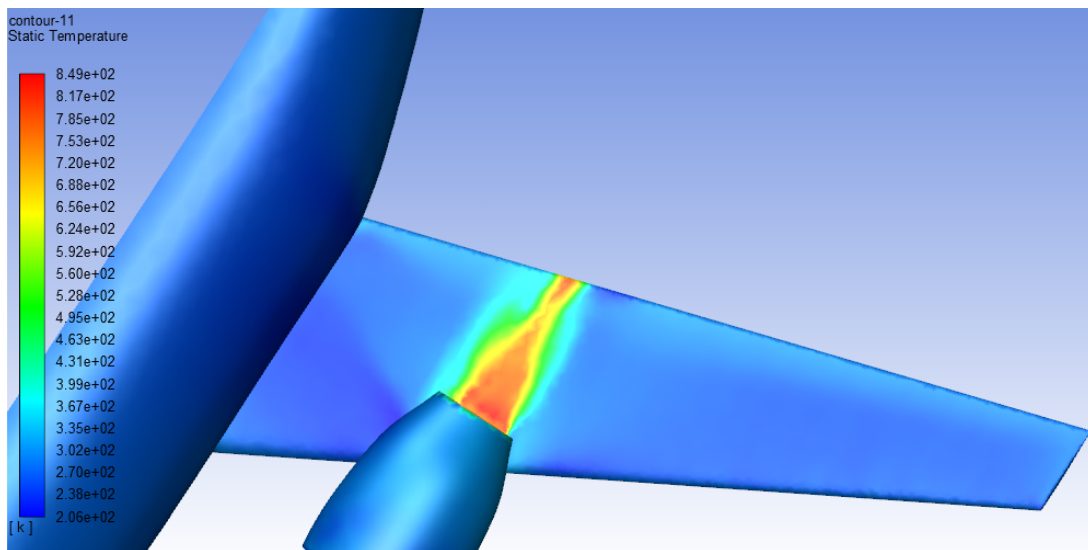


Figure 71: Thermal Analysis conducted on CREO Parametric 5.0.

The wing experiences the highest temperature directly behind the engine. The thermal analysis results are shown above. Here, the temperature gradient is represented by a color

scale. Red being the hottest, directly behind the engine, and blue being the coolest, the ambient temperature at cruising altitude.

As can be seen in the image above, only the section directly behind the engine experiences extreme heat. With some optimization it was found that each wing would require 24.3 square feet of the upper surface to be titanium aluminide to provide the proper thermal support, giving us a lot of clearance for error. This translated over to a total of 875 cubic inches of titanium aluminide needed to provide the appropriate thermal protection. Which is about 144.375 lbs of the alloy, only increasing the overall weight of the aircraft by 58.6 lbs given that aluminum would have accounted for the other weight but will now be replaced by titanium aluminide.



## 9 Flying Qualities Assessment

This section contains a concise flying qualities analysis of the proposed jet, investigating both the longitudinal and lateral dimensional modes. All of the control response simulations were run using software written by Professor R. Stengel [15], detailed in the appendix. Mass and inertial values were found using CREO parametric and the aerodynamic coefficients were calculated using Open VSP [16]. This section analyses three key flight regimes:

- (i) Takeoff, full fuel. (Mach 0.18)
- (ii) Cruise, low fuel. (Mach 0.85)
- (iii) Cruise, full fuel. (Mach 0.85)

The goal of this analysis is to show that the aircraft is stable in all of these flight regimes, and that the aircraft satisfies the FAR-23.2 requirement of being controllable without "exceptional skill of strength from the pilot" [17].

*NB: For the purposes keeping this section from being overly verbose, a fair amount of information and figures have been put (and labeled) in the appendix for reference.*

### 9.1 Aerodynamic Coefficients

In order to run simulations, the aerodynamic coefficients for the aircraft first needed to be obtained. A model of the final configuration was created in Open VSP, with mass properties assigned based on the design constraints, initial sizing, engine choice, and structural analysis outlined above. Interior seating and cargo space was modeled within the Open VSP environment, under the assumption that the average crew/passenger weighed approximately 200 lbs and each crew/passenger was allotted 100 lbs of cargo at a density of 10 lbs/ft<sup>3</sup>. Point masses were used to model both the chosen Williams International FJ44 engines and the fuel mass in each wing (for analyses of takeoff and cruise at full fuel). The aircraft's lifting surfaces (e.g. its wing, fuselage, and tail components) were modeled as shells, with area densities (kg/m<sup>2</sup>) assigned to them such that their total combined weight matched the structural weight calculated in the structural configuration and sizing. Modeling the structural weight as a shell ensures a better estimate for moments of inertia than if all the aircraft's components had been modeled as point masses. The Open VSP model was then exported to CREO Parametric, where mass analyses could be run to verify the values calculated within Open VSP. The mass and body axis inertial properties were calculated and are presented below, with units of Newton Meters. It should be noted that the body axis inertia values are calculated for full/maximum aircraft takeoff weight.

Weight	Value (lbs)
Empty Weight	8133 [lb] = 3690 [kg]
Maximum Take off weight	9733 [lb] = 4410 [kg]

Table 20: Aircraft Mass

$$\mathbb{I} = \begin{bmatrix} I_{xx} & 0 & I_{xz} \\ 0 & I_{yy} & 0 \\ I_{zx} & 0 & I_{zz} \end{bmatrix} = \begin{bmatrix} 1.62 \times 10^5 & 0 & -4.59 \times 10^4 \\ 0 & 2.73 \times 10^5 & 0 \\ -4.59 \times 10^4 & 0 & 4.22 \times 10^5 \end{bmatrix} \quad (17)$$

Inertial values are only quoted to three significant figures, since these values vary depending passenger and payload distribution, as well as the the specifics of final construction. It was decided that to quote these figures to any higher resolution would be egregious and unnecessary for the purposes of this report.

In the final aircraft configuration, the wings are very close to the center of mass (hence, the low static margin). Because the aircraft's fuel is stored in the wings, any change in fuel mass during flight has a small effect on the static margin. In other words, because the fuel mass is in-line with the center of gravity, the aircraft's center of gravity will not shift significantly. The overall lighter weight will affect the body axis moments of inertia, as well as the cruise trim conditions, which has been accounted for. For the purposes of calculation, the  $I_{xx}$  and  $I_{zz}$  values were scaled down by the same fraction the fuel decreases. Whilst this is an approximation, this can be justified since for three dimensional objects the moment of inertia is directly proportional to mass of the object. Since the fuel can be approximated as being distributed along the axis in line with the wings (Y axis), the  $I_{xx}$  (relating to roll) and  $I_{zz}$  (relating to yaw) are what will be most significantly effected by fuel usage, and therefore were scaled accordingly.

The next table details the calculated aerodynamic coefficients for the aircraft. These were used in simulation, and were calculated using VSP software. All other coefficients were calculated using these coefficients, and can be found detailed in the Appendix.

$CL_0$	0.0027	$CM_0$	-0.16	$CY_0$	0	$CL_0$	0	$CN_0$	0
$CL_\alpha$	7.37	$CM_q$	-82.27	$CY_b$	-0.74	$CL_b$	-0.036	$CN_b$	0.004
$CL_q$	31.12	$CM_V$	0.013	$CY_p$	-0.2	$CL_p$	-0.692	$CN_p$	0.077
$CL_{dE}$	0.69	$cm_{dE}$	-3.18	$Cy_r$	0.638	$Cl_r$	0.08	$Cn_r$	-0.28
$CD_0$	0.026	$CY_{dA}$	-0.04	$Cl_{dA}$	-0.39	$Cn_{dA}$	0		
SM	0.08	$CY_{dR}$	0	$CL_{dR}$	0	$Cn_{dR}$	-0.2		

Table 21: Aerodynamic Coefficients (taken from VSP)

## 9.2 Construction of State Matrices

Using the above calculated aerodynamic and stability coefficients, reduced-order state space models for the aircraft's longitudinal and lateral-directional flight dynamics can be constructed [18]. Following the linear time-invariant model:

$$\dot{\mathbf{x}} = F\mathbf{x} + B\mathbf{u} \quad (18)$$

where  $\mathbf{x}$  is a state vector containing four aircraft flight states. Construction of the state matrices is as follows:

### 9.2.1 Longitudinal Mode

For  $\mathbf{x} = [u \ w \ q \ \alpha]^T$  and  $\mathbf{u} = [\delta E \ \delta T]^T$  where  $u$  is the body axis velocity,  $w$  is the body normal velocity,  $q$  is the pitch rate,  $\alpha$  is the angle of attack,  $\delta E$  is the elevator deflection, and  $\delta T$  is the throttle:

$$F = \begin{bmatrix} -0.0103 & -9.81 & -0.0141 & -13.425 \\ 0.0003 & 0 & 0.0012 & 1.188 \\ 0 & 0 & -0.0136 & -0.599 \\ -0.0003 & 0 & 0.999 & -1.188 \end{bmatrix} \quad (19)$$

$$G = \begin{bmatrix} -0.199 & 3.978 \\ 27.807 & 0 \\ -3.227 & 0 \\ -27.807 & 0 \end{bmatrix} \quad (20)$$

### 9.2.2 Lateral Directional Mode

For  $\mathbf{x} = [\beta \ p \ r \ \phi]^T$  and  $\mathbf{u} = [\delta A \ \delta R]^T$  where  $\beta$  is the sideslip angle,  $p$  is the roll rate,  $r$  is the yaw rate,  $\phi$  is the roll angle,  $\delta A$  is the aileron deflection, and  $\delta R$  is the rudder deflection:

$$F = \begin{bmatrix} -0.285 & 2.25 & 1.405 & 0 \\ -0.896 & -0.119 & -0.0282 & 0.039 \\ 1.499 & -1.031 & -9.611 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (21)$$

$$G = \begin{bmatrix} 0.595 & -1.0772 \\ -1.610 & 0 \\ -5.472 & 0.305 \\ 0 & 0 \end{bmatrix} \quad (22)$$

### 9.2.3 Eigenvalues

$$\lambda_{LON} = \begin{bmatrix} -0.6038 + 0.5022i \\ -0.6038 - 0.5022i \\ -0.0022 + 0.0543i \\ -0.0022 - 0.0543i \end{bmatrix} \quad (23)$$

$$\lambda_{LAT} = \begin{bmatrix} -9.818 \\ -0.1022 + 1.362i \\ -0.1022 - 1.362i \\ 0.0066 \end{bmatrix} \quad (24)$$

## 9.3 Trim Conditions

For each of the considered regimes, the 'trimmed' flight conditions are presented in Table [22](#). The two cruise regimes were analyzed using Professor R. Stengel's aforementioned software [15](#), and the take off condition was analyzed using first principles analysis. Given the geometry of the wings and mass of the aircraft, a fairly rudimentary calculation was made to determine takeoff speed, angle of attack, and required thrust. The values for takeoff were then confirmed through SUAVE flight simulation discussed in Section [5](#). Since the takeoff regime includes a range of angles of attack, the stabilator angle was not calculated, however in real flight the stabilator would change as necessary to maintain stable flight. The values quotes for takeoff conditions relate to the beginning ( $t=0$ ) of the flight, for a fully fueled aircraft. The angle of attack is expected to increase throughout the climb regime, and to account for variance in altitude the thrust is also not held constant. This variance in angle of attack is displayed below, and the variance in thrust is shown in Figure [34](#).

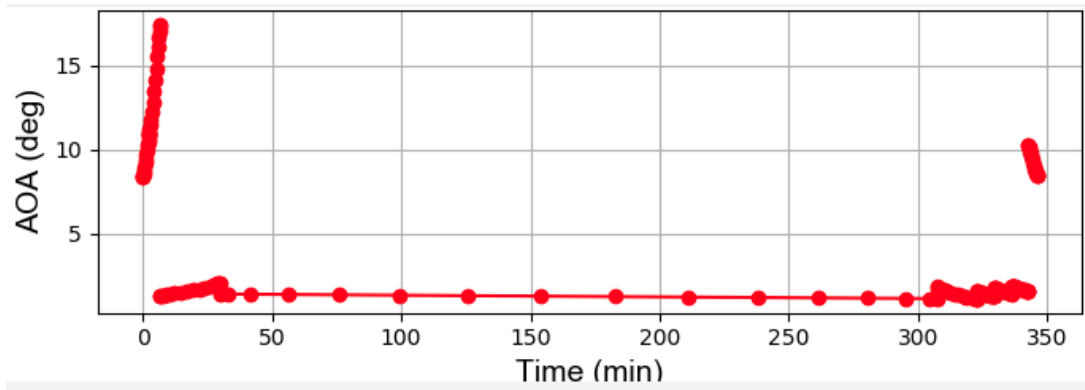


Figure 72: Angle of Attack vs. Time, SUAVE

Trim Variable	(i) Takeoff	(ii) Empty Cruise	(iii) Full Cruise
$V \left(\frac{m}{s}\right)$	61.78	250.68	250.63
$\alpha$ (deg)	8.59	1.85	2.17
Stabilator (deg)	–	-3.22	-3.29
Throttle (%)	86.96	37.32	39.42

Table 22: Trim Conditions

It should be noted that the angle of attack for cruise decreases as fuel is consumed, and to maintain stability the stabilator angle decreases too. This phenomenon occurs since as fuel is used, the aircraft becomes lighter. The lighter aircraft requires less lift to maintain steady flight, and therefore can trim to a lower angle of attack. This lower angle of attack comes with improved drag characteristics, which explains why the velocity of flight increases slightly, and the required throttle usage decreases. All of these are expected trends, which is promising for early analysis. These expected trends are also confirmed by the results from the SUAVE environment.

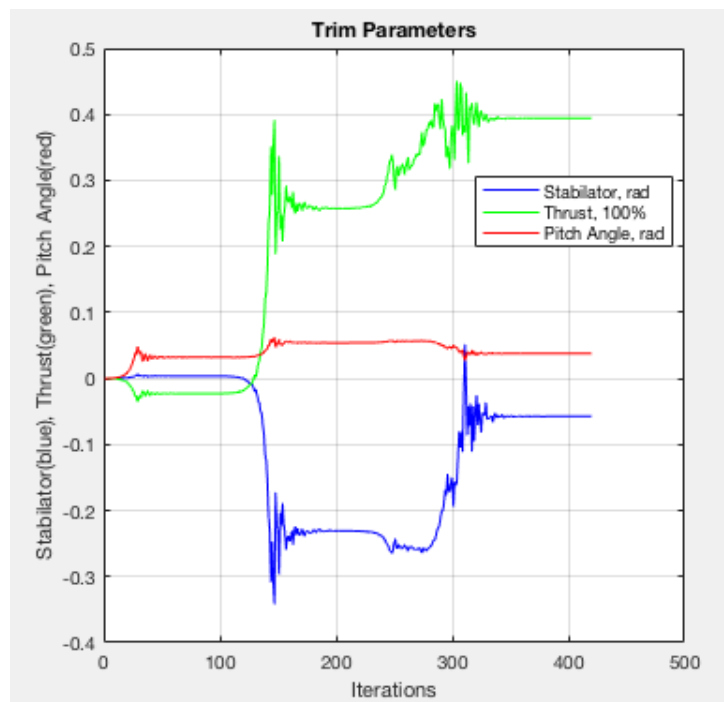


Figure 73: A figure showing the iterations to reach converged trim conditions

## 9.4 Steady State Behaviour in Level Flight

With the calculated trim conditions, respective inertial matrices, and aerodynamic coefficients, full simulations of flight were conducted to ensure the aircraft's stability in cruise. Given zero initial condition perturbations, cruise at Mach 0.85 at an altitude of 12,000 [m] was simulated over a 60s period. A graphical result of the simulated flight path is displayed in Figure 74.

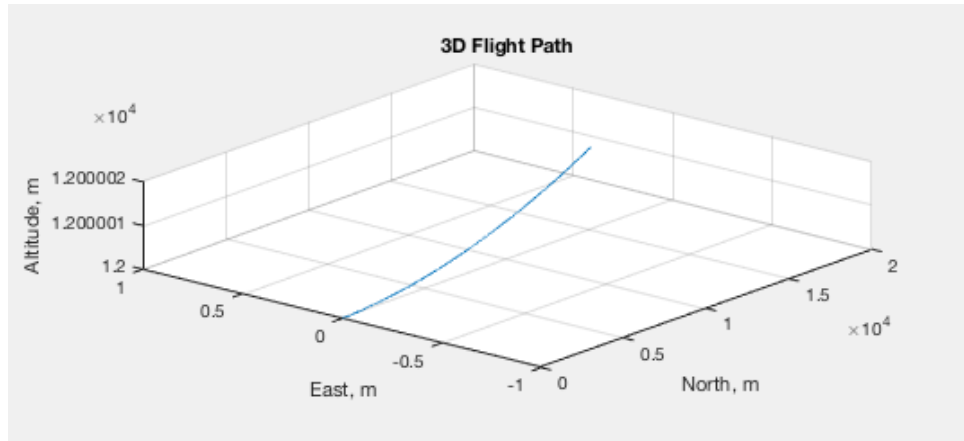


Figure 74: 3D plot of 60s flight simulation

Over the 60 seconds the simulation shows the aircraft is capable of flying in perfectly a straight line, with increase in altitude of 1.5cm. This increase is insignificant in the context of commercial aircraft cruise, and is likely the result of simulation limitation. In real flight this slight increase in altitude would easily be mitigated with higher resolution control and trim systems. For the cruise regime, the aircraft is stable in flight. The code used to create this simulation can be found in the appendix, Section 12.4.

## 9.5 Initial Condition Response

With the steady behaviour of the aircraft confirmed, the next step was to investigate it's response to various impulse initial conditions. The aircraft should respond stably to most initial condition perturbations. In other words, for some initial perturbation, the aircraft should return back to a nominal flight condition without external input. This test acts as another good metric for aircraft's overall stability. Both the lateral directional and longitudinal modes were tested to confirm FAR-23 compliance. These simulations were run using the same software used to analyze the steady state responses. For the case of simplicity all of these responses the aircraft were modeled at cruising altitude and mach number with full fuel capacity.

There were problems getting OpenVSP to properly compile and combine the masses and inertia of all the different aircraft flight components, which led to a simulated static margin within OpenVSP that was dramatically different from the static margin of 0.08 that we had designed around and confirmed through mass modelling in Creo. By recalculating longitudinal aerodynamic coefficients that were affected by this erroneous static margin calculation (notably, lift coefficients with respect to pitch rate) using alternative methods [18], the simulations of the aircraft's initial condition perturbation responses exhibited the originally expected behavior (namely, phugoid and short period oscillation). Furthermore, OpenVSP uses yaw angle  $\phi$  as a lateral-directional state, rather than the sideslip angle  $\beta$  as used in the above model. A correction was applied within the simulation code to account for this discrepancy.

Simulations of these initial condition responses for both the longitudinal and lateral directional modes are included in the Appendix. Figure 78 through Figure 85 showcases the simulated responses for the model using aerodynamic coefficients directly outputted from OpenVSP, while Figure 86 through Figure 93 showcases the simulated response after the above-mentioned correction for static margin. In the longitudinal initial condition responses, the phugoid and damped short-period modes can be clearly seen, as predicted by the frequencies of the eigenvalues. Furthermore, as all four longitudinal eigenvalues are in the LHP, all states in the longitudinal mode are stable. In the lateral-directional initial condition responses, the aircraft exhibits damped oscillatory behavior, as predicted by the negative real and oscillatory eigenvalues. As might be inferred from the one eigenvalue in the RHP, the fourth state – roll angle – is slightly unstable. Our configuration has no dihedral, so this behavior should be expected. A flight controller may be necessary to create stability within this state and satisfy FAR-23 requirements.

## 10 FAR 23 Specifications

Throughout the design process it was important to keep in mind that the final configuration must be concordant with FAR-23 (Federal Aviation Regulations) [17]. The following sections detail the critical FAR-23 regulations, and provide justification to show that they have been met.

### 10.1 FAR 23.1 - Flight Recording

This regulation states that the aircraft must carry flight data and cockpit voice recorders. Although this is not discussed in the report, it is assumed that the sensors and computers necessary to carry such equipment would not significantly affect the weight/flying qualities of the aircraft, and so it is assumed that this regulation would be satisfied were the aircraft ever to be manufactured.

### 10.2 FAR 23.2000 - Controllability and Structure

This regulation states that the aircraft must be controllable without "exceptional skill or strength from the pilot". The aircraft's flying qualities are discussed in section 9, and to the extent that is necessary it has been demonstrated the aircraft is controllable. Structurally, it is required that the aircraft be able to withstand the forces of emergency landing. The aircraft structural analysis demonstrates the strength of the aircraft under extreme loading conditions, including loading on the landing gear, and that this requirement is satisfied. It should be noted that FAR 23.2150 also discusses controllability.

### 10.3 FAR 23.2005 - Aircraft Category

FAR 23.2005 requires that the category of aircraft be determined. Due to its size and cruising regime, TigerPounce falls under the *High Speed Level 3* classification.

### 10.4 FAR 23.21 - Mass and Static Properties

This regulation requires that the empty weight, center of gravity, body axis moments of inertia and static margin must be known and repeatable, as well as being safe to fly with. The mass and static properties of the aircraft are discussed in the report in sections....

### 10.5 FAR 23.211 - Stall Modes

The aircraft's stall speeds for both cruise and zero thrust must be determined. **this has not been done**

### 10.6 FAR 23.2115 - Take off Performance

The aircraft's "aborted critical take off speed" needs to be calculated. **Not currently done**

### 10.7 FAR 23.2225 - Structural Loading

The aircraft must not fail structurally when exposed to the forces of normal (and abnormal) operation. The structural integrity of the aircraft is discussed in section

## 10.8 FAR 23.2155, 23.2305 - Landing Gear and Taxiing

These requirements discuss that the aircraft should be able to safely taxi around airports, and also safely be able to take off and land. The aircraft's ground stability is discussed in the section... The aircraft's stability during flight maneuvers is discussed in section

## 10.9 Conclusions and Other Comments

After a general and overall analysis, the TigerPounce appears to satisfy all the key FAR-23 requirements, and is therefore 'legal' to fly. It should be noted, however, that this list is significantly condensed compared to the full list of FAR-23 specifications. Be that as it may, it was decided that not every requirement is appropriate to consider at this stage in the design process. For instance, FAR-23.2515 requires that the internal electronics of the aircraft be safe from shorting in the case of a lightning strike. Whilst this is clearly important, it, along with all other considered requirements, is beyond the level of this design. If this aircraft were to be commercially produced, a significant amount of human power would be needed to ensure its total and complete safety during operation.



## 11 Business Projections and Cost Analysis

The DAPCA-IV model for estimating the cost of aircraft was created by the RAND corporation in 1987. It uses the weight of the aircraft and a few other simple parameters to determine rough estimates for the cost of developing and manufacturing new aircraft. While it can be used for aircraft generally it was designed for use by the military. For this reason modified versions of the method had been created that are more accurate for other aircraft types. The model used to evaluate the costs of the TigerPounce in this document is called the "Eastlake Model" and it is developed specifically for the analyzing business aircraft. [19] The differences to the DAPCA-IV model are not very big, mostly consisting of changed constants and the ability to account for different levels of certification. [19]

All of the information and equations presented in this section, unless otherwise stated, are taken from a book by Snorri Gudmundsson, titled "General Aviation Aircraft Design - Applied Methods and Procedures". In the model presented by Gudmundsson all of the constants are calculated for the cost of living in 2012, which means that the first step is to determine the inflation between 2012 and 2019 using the Consumer Price Index (CPI). Checking the government data on CPI from the Bureau of Labor Statistics gives [20]:

$$CPI_{2012} = 1.12 \quad (25)$$

The second step, finding a quantity discount factor (QDF), is similarly used across many of the calculations in the model. The DAPCA-IV model doesn't account specifically for any of the engines, avionics, or some of the other parts of an aircraft they are "purchased" after all the calculations for engineering, development, etc. are done using the empty weight of the aircraft. All of these goods will become cheaper as more aircraft are manufactured because an increase in quantity purchased leads to a drop in price and an "experience effectiveness" factor that accounts for technicians becoming familiar with the systems. Gudmundsson recommends using a experience effectiveness ( $F_{EXP}$ ) of .95.  $N$  is used in this equation, and throughout this section, as the number of units produced.

$$QDF = (F_{EXP})^{1.4427 \ln N} \quad (26)$$

The equations needed to carry out the full analysis can be found in Gudmundsson's book, but the process is as follows. Calculate expected number of hours needed to engineer the aircraft and how long it will take to design all the tooling needed for fabricating the aircraft. Calculate the cost of this labor using an hourly rate of \$92 for engineers and \$61 for those working on the tooling (both adjusted for inflation using  $CPI_{2012}$ ). Determine the cost of developing the plane and doing flight tests. These four costs – engineering, tooling, development, and flight tests – are the one time costs that will be done no matter how many aircraft are ultimately shipped out. The cost of manufacturing, quality control, and materials needed to construct a target number of aircraft can then be estimated. These costs scale greatly based on the number of aircraft being manufactured. Lastly, the cost per aircraft for the landing gear, avionics and power plant can be estimated using the QDF factor outlined above.

The estimations for engineering, development, flight tests, tooling, manufacturing, materials, quality control, and power plant costs are all calculated using:  $N$ , the number of aircraft built in the first five years or production;  $W_{airframe}$ , the weight of just the air frame in pounds (no landing gear, engines, seats, payload, etc.);  $V_{max}$ , the maximum cruising speed of an aircraft in knots;  $T$ , pounds of thrust needed in each engine;  $f_{comp}$  the fraction of the aircraft that is composite;  $F_{CERT}$ , a certification factor that accounts for

what level of certification the plane is getting;  $F_{CF}$  and complex flap factor that increases the price for a plane with complex flaps,  $F_{PRESS}$  a pressurization factor that increases the cost for developing a aircraft that has to be pressurized;  $F_{TAPER}$  a factor that increases the cost of a plane with a tapered wing; and  $Q_m$ , which is the monthly rate of aircraft production. The cost of avionics for all three systems is input at \$60,000 as that was the cost of a state of the art avionics system in 2012 for a FAR 23 aircraft, this value is then adjusted by the QDF and CPI values mentioned above.

Aircraft	TigerPounce	HondaJet[5]	Cessna Citation M2[21]
$W_{empty}$	8133 lbs	7283 lbs	6990 lbs
$V_{max}$	487 kts	422 kts	404 kts
$T$	2300 lbs	1997 lbs	1965 lbs
$f_{comp}$	.017	0	0
$F_{CERT}$	FAR 23	FAR 23	FAR 23
$F_{CF}$	Simple	Simple	Simple
$F_{PRESS}$	Pressurized	Pressurized	Pressurized
$F_{TAPER}$	Tapered	Tapered	Tapered

Table 23: Cost Analysis Inputs

In order to ensure that the values found using this model were somewhat accurate the numbers for the TigerPounce were compared with numbers for similar aircraft. Specifically, the HondaJet and Cessna Citation M2, both of which are light business jets that sit in the same market space as the TigerPounce. Table 23 shows the values used in the model for each of these three aircraft. The weight listed in this table is the basic operating weight of the aircraft, we had to use this value instead of the air frame weight as there is no public data for the air frame weight of the competing aircraft. While none of the three aircraft have composite materials, I felt the easiest way to model the titanium aluminide alloy needed to combat the extreme temperatures on the wing due to upper surface blowing (see Section 8) was to model it as a composite. This makes sense because, similarly to composites, there is an extra cost for the material itself as well as special steps that need to be taken in the development, tooling and manufacturing steps of the process. As such, the 1.7% of the aircraft that is made of titanium aluminide is counted as composite materials in the TigerPounce. Due to the fact that different numbers are used in different equations for the various  $F$  factors the textual values determine which option to use when recreating the equations in the Eastlake Model.

Using the equations outlined above it is possible to do some analysis of the price point for a given aircraft. The minimum selling price, visible in Figure 75, is the price at which an aircraft would have to be sold to exactly break even (inflated due to the use of empty weight rather than airfram weight). This number decreases with an increasing number of aircraft built as the fixed costs can be spread out of a larger number of aircraft. Figure 75 shows that the trend lines for all three aircraft are quite close, with TigerPounce being the most expensive to produce at all quantity levels but only by a small fraction. This price takes into account all of the costs mentioned previously as well as a \$50,000 manufacturer’s insurance for each aircraft.

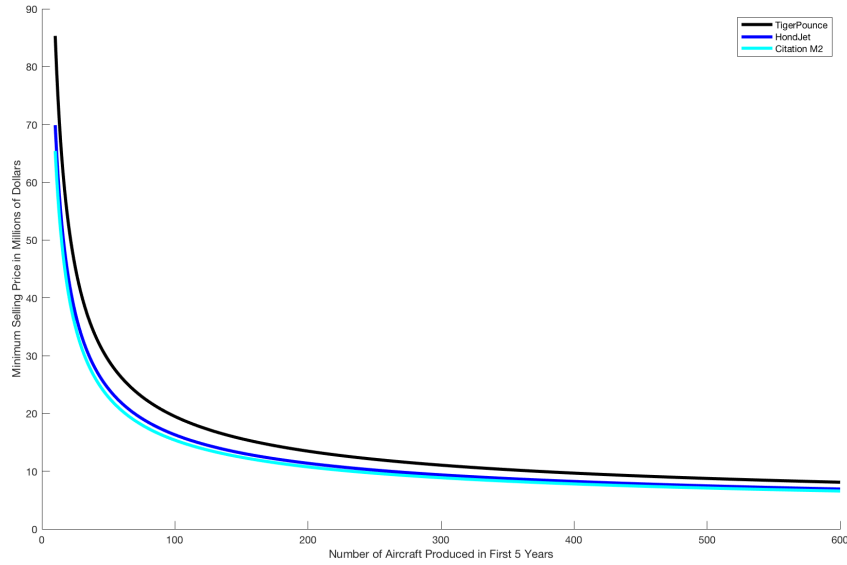


Figure 75: Minimum Sell

While the price needs to be higher than the minimum selling price in order to make a profit, the question becomes how to choose what the price should be. One way is to set the number of aircraft needed to break even,  $N_{BE}$ , to some number and set the price point that way. Looking at the data for business jet sales in 2018 as collected by the General Aviation Manufacturers Association gives a good idea of how many aircraft could be expected to sell in a year. The data shows that 703 business jets were shipped worldwide during 2018. [22] 37 of those were HondaJets, while Cessna managed to sell 34 Citation M2s and anywhere from 21 to 57 copies of five different versions of Citation jets. [22] With this in mind it would seem that setting a break even number of 200-250 aircraft, 5 years of sales, would be a reasonable decision. We have decided to set the price at \$5.4 million dollars which corresponds to a  $N_{BE}$  of roughly 250.

In order to ensure that this price point made sense we wanted to compare it to the previously mentioned aircraft. The results of choosing different break even points for each aircraft are in Table [24] along with the actual price of each. The numbers are greatly inflated here as once again the empty weight has to be used for comparison rather than the air frame rate. But, it clearly shows that while our aircraft is slightly on the pricier side it is within the desired market range.

Aircraft	Price $N_{BE} = 100$	Price $N_{BE} = 200$	Price $N_{BE} = 500$	Actual Price [5, 23]	$N_{BE}$ for Actual
Cessna Citation M2	\$16.1 Million	\$11.0 Million	\$7.1 Million	\$4.2 Million	1800
HondaJet	\$17.0 Million	\$11.5 Million	\$7.4 Million	\$4.6 Million	1700
TigerPounce	\$20.5 Million	\$13.7 Million	\$8.8 Million	\$5.4 Million	1700

Table 24: Price Points

Using the proper air frame weight of the TigerPounce it is possible to look at how the total cost of producing one airplane breaks down at different numbers of aircraft sold, visible in Figure [76]. The lines in red are all of the one time costs and they clearly make up a bulk of the costs until somewhere between 100 and 200 aircraft have been sold. At that point the blue curves, costs that scale with the number of aircraft being sold, start

to dominate the cost of a given aircraft – especially the manufacturing cost. Lastly, the black curves all the way at the bottom are for items that barely change in cost with the number of aircraft sold and therefore become an ever larger percent of the cost, especially the power plant which becomes the second most expensive thing by about 350 aircraft sold.

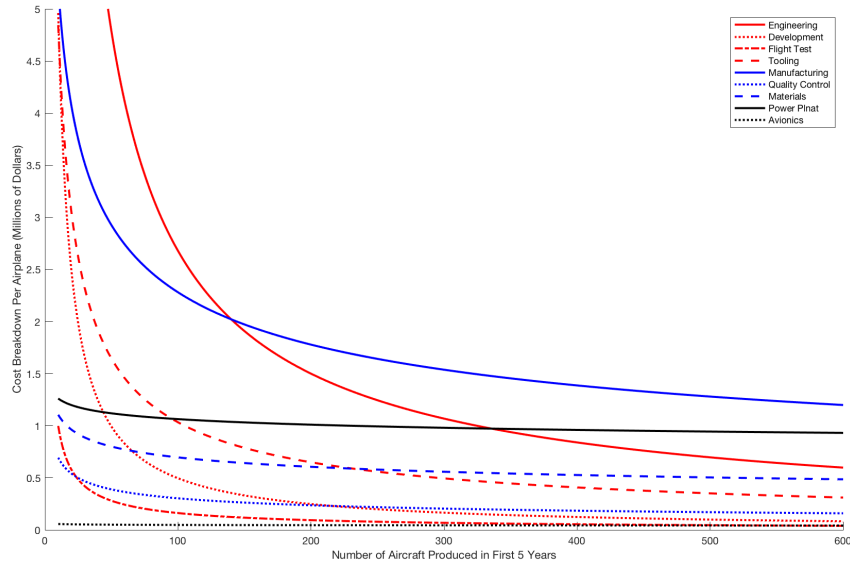


Figure 76: TigerPounce Cost Breakdown

After the total costs have been determined and a price point has been chosen it is useful to create a plot of revenue based on the number of aircraft sold. Figure 77 shows three different lines for potential aircraft prices. The solid blue line represents the chosen price point, where it takes about 250 aircraft to break even and at least 600 aircraft sold to make \$1 billion. It also shows that choosing a steeper price, such as setting  $N_{BE}$  at 100, will lead to the first billion dollars being made after only 200 aircraft are sold. However, this is not a very good choice as it would lead to a price nearly three times larger which would cause the TigerPounce to be noncompetitive in the market.

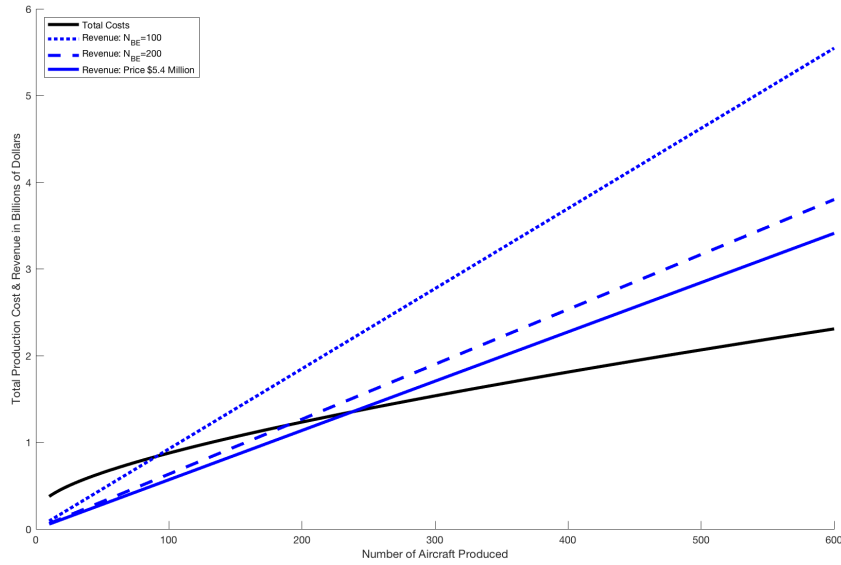


Figure 77: TigerPounce Revenue

Predicting the actual sales of the aircraft is nearly impossible as it depends a lot on the world markets. It would also depend on whether the aircraft is being developed by a known aircraft manufacturer with strong ties to industry and a good marketing scheme, or if it is being developed by a newcomer. In the latter case it is likely that the price would have to be dropped a bit in order to be more enticing to new customers and a good sales team would have to be hired in order to try and compete with the bigger companies.

The MATLAB script used to generate all of the plots in this section can be found in Listing [12.4](#) in the Appendix.

## References

- [1] “Tire specifications.” pdf, 2018.
- [2] J. Roskam, *Airplane design*. DARcorporation, 2002.
- [3] D. Raymer, *Aircraft Design: A Conceptual Approach*. American Institute of Aeronautics and Astronautics, Inc., 2012.
- [4] “<http://ourairports.com/data/>.”
- [5] H. A. Company, “Honda jet brochure.” pdf.
- [6] L. Martinelli, “Mae 322 class slides and office hours,” 2019.
- [7] F. Lutze, “Takeoff and landing.” pdf.
- [8] J. K. Wimpress and C. F. Newberry, “The YC-14 Prototype,” 1998.
- [9] J. H. Jung, M. J. Kim, H. S. Yoon, P. A. Hung, H. H. Chun, and D. W. Park, “Endplate effect on aerodynamic characteristics of three-dimensional wings in close free surface proximity,” *International Journal of Naval Architecture and Ocean Engineering*, vol. 4, no. 4, pp. 477 – 487, 2012.
- [10] “Analytical model of aircraft wing loads – matlab example.”
- [11] “Aluminum 7075-t6; 7075-t651.”
- [12] V. T. Hoang Hoang, *Simplified Design of a Commercial Aircraft Wing Made of Carbon Fiber*. PhD thesis, 2015.
- [13] V. S. Muchchandi, “Design and analysis of a spar beam for the vertical tail of a transport aircraft,” *International Journal of Innovative Research in Science, Engineering and Technology*, vol. 2, Jul 2013.
- [14] I. SEN, *Aircraft Fuselage Design Study*. PhD thesis, 2010.
- [15] R. Stengel, *Flight Dynamics*. Princeton University Press, 2004.
- [16] OpenVSP, “Openvsp wiki.” <http://openvsp.org/wiki/doku.php>.
- [17] F. link, “Electronic code of federal regulations - part 23.”
- [18] R. C. Nelson, *Flight Stability and Automatic Control*. McGraw-Hill Book Company, 1989.
- [19] S. Gudmundsson, *General Aviation Aircraft Design: Applied Methods and Procedures*. Elsevier, 2014.
- [20] “Cpi inflation calculator.”
- [21] “Citation m2.”
- [22] “General aviation aircraft shipment report.” pdf.
- [23] A. Davies, “Cessna citation m2: Buyer’s and investor’s guide.”

## 12 Appendix

### 12.1 V-n Diagram Calculations

$$n_{max} = +2.5$$

$$n_{min} = -1$$

The first major step for constructing the V-n diagram is to determine the V-speeds. These are cruise ( $V_C$ ), corner ( $V_{corner}$ ), dive ( $V_D$ ), never-exceed ( $V_{NE}$ ), and stall ( $V_S$ ). Below are their calculations:

$$V_C = c_{40} * M$$

where  $c_{40}$  is the speed of sound at 40,000 feet and  $M$  is the cruising Mach number of the aircraft. Therefore,

$$V_C = (967.71 \frac{ft}{s})(0.85)$$

$$V_C = 822.55 \frac{ft}{s}$$

The corner velocity is calculated as follows:

$$V_{corner} = \sqrt{\frac{2n_{max}W}{C_{L_{max}}\rho_{SL}S}}$$

where  $n_{max} = +2.5$ ,  $W = 9733 \text{ lb}$ ,  $C_{L_{max}} = 4.3$ ,  $\rho_{SL} = 0.0023769 \frac{slug}{ft^3}$ , and  $S = 172 \text{ ft}^2$ . Therefore,

$$V_{corner} = \sqrt{\frac{2(2.5)(9733)}{(4.3)(0.0023769)(172)}}$$

$$V_{corner} = 166.38 \frac{ft}{s}$$

As for the dive speed, the design limitation here was that the aircraft should not exceed  $M = 1$  so as to avoid supersonic complications to the structure. Therefore, we settled at a 15% increase from the cruise speed:

$$V_D = (1.15)V_C = (1.15)(822.55 \frac{ft}{s})$$

$$V_D = 945.93 \frac{ft}{s}$$

The never-exceed speed is typically 90% of the dive speed. For the purposes of this aircraft, that makes sense as a middle ground between the cruise speed and the dive speed. Therefore,

$$V_{NE} = (0.9)V_D = (0.9)(945.93 \frac{ft}{s})$$

$$V_{NE} = 851.33 \frac{ft}{s}$$

Finally, the stall speed is calculated as follows:

$$V_S = \sqrt{\frac{2W}{\rho_{SL}S C_{L_{max}}}}$$

where  $W = 9733 \text{ lb}$ ,  $\rho_{SL} = 0.0023769 \frac{\text{slug}}{\text{ft}^3}$ ,  $S = 172 \text{ ft}^2$ , and  $C_{L_{max}} = 4.3$ . Therefore,

$$V_S = \sqrt{\frac{2(9733)}{(0.0023769)(172)(4.3)}}$$

$$V_S = 105.23 \frac{\text{ft}}{\text{s}}$$

The stall limit curves were the next critical component of the V-n diagram to be found. They were calculated in the following way:

$$n_+ = \frac{\frac{1}{2}C_{L_{max}}\rho_{SL}S}{W}V^2$$

$$n_+ = \frac{\frac{1}{2}(4.3)(0.0023769)(172)}{9733}V^2$$

$$n_+ = 0.00009031V^2$$

Additionally,

$$n_- = \frac{\frac{1}{2}C_{L_{min}}\rho_{SL}S}{W}V^2$$

$$n_- = \frac{\frac{1}{2}(-2.64)(0.0023769)(172)}{9733}V^2$$

$$n_- = -0.00005544V^2$$

Now that the data necessary for the V-n diagram was complete, the gust envelope for only the positive load factor limits could be found. The first step was to calculate  $C_{L_\alpha}$ :

$$C_{L_\alpha} = \frac{2\pi}{1 + \frac{2}{AR}}$$

where  $AR$  is the aspect ratio. Therefore,

$$C_{L_\alpha} = \frac{2(3.14159)}{1 + \frac{2}{8.5}}$$

$$C_{L_\alpha} = 5.08$$

Now the load increase due to gust at cruise could be calculated in the following way:

$$\Delta n_C = \frac{\rho_C u V_C C_{L_\alpha}}{2\left(\frac{W}{S}\right)}$$

where  $u$  is a typical gust velocity factor that is used to calculate the cruise delta-load as according to FAR compliance. Therefore,

$$\Delta n_C = \frac{(5.87 * 10^{-4})(50)(822.55)(5.08)}{2\left(\frac{9733}{172}\right)}$$

$$\Delta n_C = +1.08$$

As for the delta-load for dive speed, the calculation is quite similar:

$$\Delta n_D = \frac{\rho_D u V_D C_{L_\alpha}}{2\left(\frac{W}{S}\right)}$$



where  $u$  is a typical gust velocity factor that is used to calculate the dive delta-load as according to FAR compliance. Therefore,

$$\Delta n_D = \frac{(5.87 * 10^{-4})(-25)(945.93)(5.08)}{2\binom{9733}{172}}$$

$$\Delta n_D = -0.65$$

As stated earlier, the gust envelope for the positive load factor limits is the only one necessary for calculation (negative load factor limits are rendered irrelevant by the mission). Therefore, the complete V-n diagram with gust envelope for the purposes of this aircraft's mission is shown in Figure [59](#).

## 12.2 Titanium Alpha Alloy Properties



### Titanium, alpha alloy, Ti-24Al-11Nb, alpha - two aluminide (24-11)

Page 1 of 4

#### General information

##### Designation

Ti-24Al-11Nb

##### Typical uses

High temperature applications; Advanced hot structure applications; engine applications.

#### Composition overview

##### Compositional summary

Ti65 / Nb22 / Al14

Material family	Metal (non-ferrous)
Base material	Ti (Titanium)

#### Composition detail (metals, ceramics and glasses)

Al (aluminum)	14	%
Nb (niobium)	21.5	%
Ti (titanium)	64.7	%

#### Price

Price	* 26.7	-	27.1	USD/lb
-------	--------	---	------	--------

#### Physical properties

Density	0.165	lb/in <sup>3</sup>
---------	-------	--------------------

#### Mechanical properties

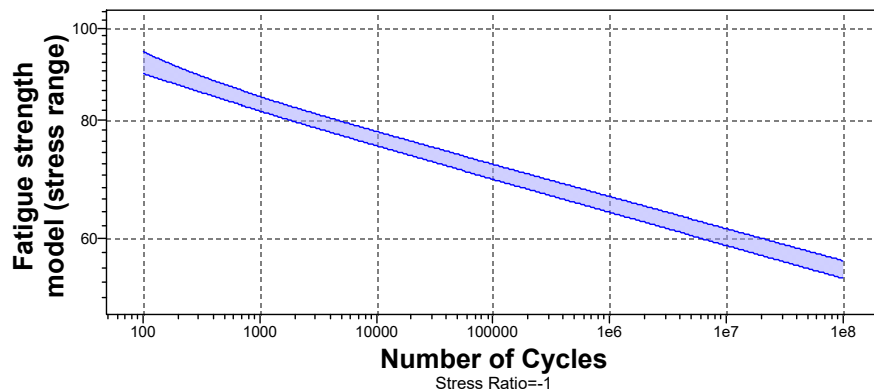
Young's modulus	13.1	-	13.9	10 <sup>6</sup> psi
Yield strength (elastic limit)	84.3	-	85.7	ksi
Tensile strength	104	-	106	ksi
Elongation	1	-	3	% strain
Compressive strength	* 101	-	103	ksi
Flexural modulus	* 13.1	-	13.9	10 <sup>6</sup> psi
Flexural strength (modulus of rupture)	92.8	-	94.3	ksi
Shear modulus	4.64	-	5.22	10 <sup>6</sup> psi
Bulk modulus	14.5	-	18	10 <sup>6</sup> psi
Poisson's ratio	0.35	-	0.37	
Shape factor	16			
Hardness - Vickers	* 100	-	110	HV
Fatigue strength at 10 <sup>7</sup> cycles	* 59.9	-	60.6	ksi
Fatigue strength model (stress range)	* 59.1	-	61.5	ksi

[Parameters:](#) Stress Ratio = -1, Number of Cycles = 1e7cycles

Values marked \* are estimates.  
No warranty is given for the accuracy of this data



Titanium, alpha alloy, Ti-24Al-11Nb, alpha - two aluminide (24-11)



Mechanical loss coefficient (tan delta)	* 5e-4	-	0.001
---	--------	---	-------

**Impact & fracture properties**

Fracture toughness	* 12.7	-	16.4	ksi.in <sup>0.5</sup>
--------------------	--------	---	------	-----------------------

**Thermal properties**

Melting point	* 2.7e3	-	2.82e3	°F
Maximum service temperature	986	-	1.29e3	°F
Minimum service temperature	-459			°F
Thermal conductivity	* 3.47	-	4.62	BTU.ft/hr.ft <sup>2</sup> .°F
Specific heat capacity	* 0.131	-	0.143	BTU/lb.°F
Thermal expansion coefficient	5	-	5.83	µstrain/°F
Latent heat of fusion	155	-	159	BTU/lb

**Electrical properties**

Electrical resistivity	* 126	-	158	µohm.cm
Galvanic potential	* -0.21	-	-0.13	V

**Magnetic properties**

Magnetic type	Non-magnetic
---------------	--------------

**Optical properties**

Transparency	Opaque
--------------	--------

**Processing properties**

Metal casting	Limited use
Metal cold forming	Acceptable

Values marked \* are estimates.  
No warranty is given for the accuracy of this data



**Titanium, alpha alloy, Ti-24Al-11Nb, alpha - two aluminide (24-11)**

Metal hot forming	Acceptable
Metal press forming	Acceptable
Metal deep drawing	Limited use

**Durability**

Water (fresh)	Excellent
Water (salt)	Excellent
Weak acids	Excellent
Strong acids	Acceptable
Weak alkalis	Excellent
Strong alkalis	Acceptable
Organic solvents	Excellent
Oxidation at 500C	Acceptable
UV radiation (sunlight)	Excellent
Galling resistance (adhesive wear)	Limited use

Notes  
High tendency to gall can be overcome by anodizing.

Flammability	Non-flammable
--------------	---------------

**Primary production energy, CO2 and water**

Embodied energy, primary production	* 3.3e5	-	3.63e5	BTU/lb
CO2 footprint, primary production	* 47.3	-	52.2	lb/lb
Water usage	* 7.17e3	-	7.92e3	in^3/lb

**Processing energy, CO2 footprint & water**

Casting energy	* 5.37e3	-	5.94e3	BTU/lb
Casting CO2	* 0.937	-	1.04	lb/lb
Casting water	* 655	-	982	in^3/lb
Rough rolling, forging energy	* 4.31e3	-	4.76e3	BTU/lb
Rough rolling, forging CO2	* 0.752	-	0.831	lb/lb
Rough rolling, forging water	* 161	-	242	in^3/lb
Extrusion, foil rolling energy	* 8.49e3	-	9.39e3	BTU/lb
Extrusion, foil rolling CO2	* 1.48	-	1.64	lb/lb
Extrusion, foil rolling water	* 277	-	415	in^3/lb
Wire drawing energy	* 3.15e4	-	3.48e4	BTU/lb
Wire drawing CO2	* 5.5	-	6.08	lb/lb
Wire drawing water	* 765	-	1.15e3	in^3/lb
Metal powder forming energy	* 1.84e4	-	2.03e4	BTU/lb
Metal powder forming CO2	* 3.43	-	3.79	lb/lb
Metal powder forming water	* 1.29e3	-	1.94e3	in^3/lb
Vaporization energy	* 6.26e6	-	6.92e6	BTU/lb

Values marked \* are estimates.  
No warranty is given for the accuracy of this data



**Titanium, alpha alloy, Ti-24Al-11Nb, alpha - two aluminide (24-11)**

Vaporization CO2	* 1.09e3	- 1.21e3	lb/lb
Vaporization water	* 1.68e5	- 2.52e5	in^3/lb
Coarse machining energy (per unit wt removed)	* 832	- 920	BTU/lb
Coarse machining CO2 (per unit wt removed)	* 0.145	- 0.16	lb/lb
Fine machining energy (per unit wt removed)	* 6.48e3	- 7.16e3	BTU/lb
Fine machining CO2 (per unit wt removed)	* 1.13	- 1.25	lb/lb
Grinding energy (per unit wt removed)	* 1.28e4	- 1.41e4	BTU/lb
Grinding CO2 (per unit wt removed)	* 2.23	- 2.46	lb/lb
Non-conventional machining energy (per unit wt removed)	* 6.26e4	- 6.92e4	BTU/lb
Non-conventional machining CO2 (per unit wt removed)	* 10.9	- 12.1	lb/lb

**Recycling and end of life**

Recycle	✓		
Embodied energy, recycling	* 4.01e4	- 4.43e4	BTU/lb
CO2 footprint, recycling	* 7.32	- 8.09	lb/lb
Recycle fraction in current supply	21.8	- 24.1	%
Downcycle	✓		
Combust for energy recovery	✗		
Landfill	✓		
Biodegrade	✗		

**Notes**

**Other notes**

Intermetallic compound based on Ti3Al. Limited ductility below about 530K, but can be superplastically formed and diffusion bonded.

**Keywords**

ALPHA-TWO ALUMINIDE (24/11), Titanium Metals Corp. (USA);; alpha - two aluminide (24-11);

**Links**

- ProcessUniverse
- Producers
- Reference
- Shape

Values marked \* are estimates.  
No warranty is given for the accuracy of this data

## 12.3 SUAVE Optimization Code

---

```

1 ----- DESCRIPTION -----#
2 This program provides the optimization layer
3 interface in SUAVE.
4 -----#
5 ----- AUTHOR -----#
6 @ Xerxes Libsch 5/14/2019
7 -----#
8 -----#
9 Imports
10 -----#
11
12 port SUAVE
13 om SUAVE.Core import Units, Data
14 port numpy as np
15 port Vehicles_custom
16 port Analyses_custom
17 port Missions_custom
18 port Procedure_custom
19 port Plot_Mission_custom
20 port matplotlib.pyplot as plt
21 om SUAVE.Optimization import Nexus, carpet_plot
22 port SUAVE.Optimization.Package_Setups.scipy_setup as scipy_setup
23
24 port multiprocessing
25 om functools import partial
26 port os
27 om itertools import product
28 port numpy as np
29
30 -----#
31 Run the whole thing
32 -----#
33 f main(inputs, objective, constraints, aliases):
34     problem = setup(inputs, objective, constraints, aliases)
35
36     ## Base Input Values
37     output = problem.objective()
38
39     ## Uncomment to view contours of the design space
40     variable_sweep(problem)
41
42     ## Uncomment for the first optimization
43     output = scipy_setup.SciPy_Solve(problem, solver='SLSQP')
44     print(output)
45
46     print('fuel burn = ', problem.summary.base_mission_fuelburn)
47     print('fuel margin = ', problem.all_constraints())
48

```

---

```
49 Plot_Mission.plot_mission(problem)
50
51 return
52
53 -----
54 Inputs, Objective, & Constraints
55 -----
56
57 f setup(inputs, objective, constraints, aliases):
58
59     nexus = Nexus()
60     problem = Data()
61     nexus.optimization_problem = problem
62
63     # -----
64     # Inputs
65     # -----
66     problem.inputs = inputs
67
68     # -----
69     # Objective
70     # -----
71
72     # throw an error if the user isn't specific about wildcards
73     problem.objective = objective
74
75     # -----
76     # Constraints
77     # -----
78     problem.constraints = constraints
79
80     # -----
81     # Aliases
82     # -----
83
84     problem.aliases = aliases
85
86     # -----
87     # Vehicles
88     # -----
89     nexus.vehicle_configurations = Vehicles_custom.setup()
90
91     # -----
92     # Analyses
93     # -----
94     nexus.analyses = Analyses_custom.setup(nexus.vehicle_configurations)
95
96     # -----
97     # Missions
98     # -----
```

```

99  nexus.missions = Missions_custom.setup(nexus.analyses)
100
101  # -----
102  # Procedure
103  # -----
104  nexus.procedure = Procedure_custom.setup()
105
106  # -----
107  # Summary
108  # -----
109  nexus.summary = Data()
110  nexus.total_number_of_iterations = 0
111  return nexus
112
113  f parallel(inputs, objective, constraints, aliases):
114  # segment the input space
115  inputs_grid = segment(inputs, 2)
116  # find the cartesian product
117  inputs_grid = cartesian_prod(*inputs_grid)
118
119  results = []
120  processes = []
121  for seg, id in zip(inputs_grid, range(len(inputs_grid))):
122      processes.append(multiprocessing.Process(target=main,
123          args=(np.asarray(seg), objective, constraints, aliases), name='{}'.format(i
124  for p in processes:
125      p.start()
126      p.join()
127  for p in processes:
128      print(p, 'NAME: {}'.format(p.name))
129      results.append(p)
130  return processes
131
132  f segment(inputs, n):
133  inputs_grid = [[0] * n for i in range(len(inputs))]
134  for i in range(n):
135      for x, j in zip(inputs, range(len(inputs))):
136          inputs_grid[j][i] = [x[0], x[1], (np.linspace(x[2][0], x[2][1], n+1)[i],
137              np.linspace(x[2][0], x[2][1], n+1)[i+1]), x[3], x[4]]
138  return inputs_grid
139
140  f cartesian_prod(*args):
141  return list(product(*args))
142
143  f post_process(results):
144  print(min(results))
145
146  f variable_sweep(problem):
147  number_of_points = 5
148  outputs = carpet_plot(problem, number_of_points, 0, 0) #run carpet plot, suppr

```



```

149 inputs      = outputs.inputs
150 objective    = outputs.objective
151 constraints  = outputs.constraint_val
152 plt.figure(0)
153 CS = plt.contourf(inputs[0,:],inputs[1:], objective, 20, linewidths=2)
154 cbar = plt.colorbar(CS)
155
156 cbar.ax.set_ylabel('fuel burn (kg)')
157 CS_const = plt.contour(inputs[0,:],inputs[1:], constraints[0,:,:])
158 plt.clabel(CS_const, inline=1, fontsize=10)
159 cbar = plt.colorbar(CS_const)
160 cbar.ax.set_ylabel('fuel margin')
161
162 plt.xlabel('Wing Aspect Ratio')
163 plt.ylabel('Wing Sweep (deg)')
164
165 plt.legend(loc='upper left')
166 plt.show(block=True)
167
168 return
169
170 __name__ == '__main__':
171     # [ tag , initial, (lb,ub)
172     , scaling , units ]
173     inputs = np.array([
174         ['wing_aspect_ratio' , 8.5 , ( 7 , 10 ) ,
175         10. , Units['']],
176         ['wing_sweep_qc' , 25. , ( 20 , 30 ) ,
177         100. , Units.deg],
178         ['cruise_altitude' , 12.19 , ( 9 , 14. ) ,
179         10. , Units.km],
180         ['wing_twist_root' , 0. , ( -5. , 5. ) ,
181         10. , Units.deg],
182     ])
183
184     # [ tag, scaling, units ]
185     objective = np.array([
186         [ 'fuel_burn', 1000, Units.kg ]
187     ])
188
189     # [ tag, sense, edge, scaling, units ]
190     constraints = np.array([
191         [ 'design_range_fuel_margin' , '>', 0. , 1E-1, Units.less], #fuel margin defined
192     ])
193
194     # [ 'alias' , ['data.path1.name', 'data.path2.name'] ]
195     aliases = [
196         [ 'wing_sweep_qc' ,
197         'vehicle_configurations.*.wings.main_wing.sweeps.quarter_chord' ],
198         [ 'wing_aspect_ratio' ,

```

```
199     'vehicle_configurations.*.wings.main_wing.aspect_ratio'      ],
200     [ 'cruise_altitude'                                           ,
201     'missions.base.segments.climb_3.altitude_end'                 ],
202     [ 'fuel_burn'                                                  ,
203     'summary.base_mission_fuelburn'                               ],
204     [ 'design_range_fuel_margin'                                     ,
205     'summary.max_zero_fuel_margin'                               ],
206     [ 'wing_twist_root'                                           ,
207     'vehicle_configurations.*.wings.main_wing.twists.root'      ],
208 ]
209
210 parallel(inputs, objective, constraints, aliases)
```

---

## 12.4 MATLAB Code

Professor R. Stengel's *FLIGHT.m*, used in the control response analysis section.

### FLIGHT.m

```

1 %   FLIGHT.m — Generic 6-DOF Trim, Linear Model, and Flight Path ...
   Simulation
2 %   Euler Angle/Quaternion Option for Rotation (Direction Cosine) Matrix
3 %   MAE 331 Homework 7 Edition — Input Spec Altered
4
5 %   December 12, 2018
6 %   =====
7 %   Copyright 2006–2018 by ROBERT F. STENGEL. All rights reserved.
8
9   clear
10  global GEAR CONHIS SPOIL u x V uInc tuHis deluHis TrimHist SMI ...
   MODEL RUNNING
11
12  disp('**=====**')
13  disp('** 6-DOF FLIGHT Simulation **')
14  disp('**=====**')
15  disp(' ')
16  disp(['Date and Time are ', num2str(datestr(now))]);
17  disp('=====')
18
19
20 %   This is the SCRIPT FILE. It contains the Main Program, which:
21 %       Defines initial conditions
22 %       Contains aerodynamic data tables (if required)
23 %       Calculates longitudinal trim condition
24 %       Calculates stability-and-control matrices for linear model
25 %       Simulates flight path using nonlinear equations of motion
26
27 %   Functions used by FLIGHT:
28 %       AeroModelAlpha.m   High-Alpha, Low-Mach aerodynamic ...
   coefficients of the aircraft,
29 %                               thrust model, and geometric and inertial ...
   properties
30 %       AeroModelMach.m   Low-Alpha, High-Mach aerodynamic ...
   coefficients of the aircraft,
31 %                               thrust model, and geometric and inertial ...
   properties
32 %       AeroModelUser.m   User-defined aerodynamic coefficients of ...
   the aircraft,
33 %                               thrust model, and geometric and inertial ...
   properties
34 %       Atmos.m           Air density, sound speed
35 %       DCM.m             Direction-cosine (Rotation) matrix from ...
   Euler Angles
36 %       RMQ.m             Direction-cosine (Rotation) matrix from ...
   Quaternions
37 %       EoM.m             Equations of motion for integration ...
   (Euler Angles)
38 %       EoMQ.m            Equations of motion for integration ...
   (Quaternions)
39 %       LinModel.m        Equations of motion for defining linear-model
   (F & G) matrices via central differences
40 %
41 %       TrimCost.m        Cost function for trim solution

```

```

42 %      WindField.m      Wind velocity components
43
44 %      DEFINITION OF THE STATE VECTOR
45 %      With Euler Angle DCM option (QUAT = 0):
46 %      x(1)      =      Body-axis x inertial velocity, ub, m/s
47 %      x(2)      =      Body-axis y inertial velocity, vb, m/s
48 %      x(3)      =      Body-axis z inertial velocity, wb, m/s
49 %      x(4)      =      North position of center of mass WRT Earth, ...
      xe, m
50 %      x(5)      =      East position of center of mass WRT Earth, ...
      ye, m
51 %      x(6)      =      Negative of c.m. altitude WRT Earth, ze = -h, m
52 %      x(7)      =      Body-axis roll rate, pr, rad/s
53 %      x(8)      =      Body-axis pitch rate, qr, rad/s
54 %      x(9)      =      Body-axis yaw rate, rr, rad/s
55 %      x(10)     =      Roll angle of body WRT Earth, phir, rad
56 %      x(11)     =      Pitch angle of body WRT Earth, thetar, rad
57 %      x(12)     =      Yaw angle of body WRT Earth, psir, rad
58 %      With Quaternion DCM option (QUAT = 1):
59 %      x(1)      =      Body-axis x inertial velocity, ub, m/s
60 %      x(2)      =      Body-axis y inertial velocity, vb, m/s
61 %      x(3)      =      Body-axis z inertial velocity, wb, m/s
62 %      x(4)      =      North position of center of mass WRT Earth, ...
      xe, m
63 %      x(5)      =      East position of center of mass WRT Earth, ...
      ye, m
64 %      x(6)      =      Negative of c.m. altitude WRT Earth, ze = -h, m
65 %      x(7)      =      Body-axis roll rate, pr, rad/s
66 %      x(8)      =      Body-axis pitch rate, qr, rad/s
67 %      x(9)      =      Body-axis yaw rate, rr, rad/s
68 %      x(10)     =      q1, x Component of quaternion
69 %      x(11)     =      q2, y Component of quaternion
70 %      x(12)     =      q3, z Component of quaternion
71 %      x(13)     =      q4, cos(Euler) Component of quaternion
72
73 %      DEFINITION OF THE CONTROL VECTOR
74 %      u(1)      =      Elevator, dEr, rad, positive: trailing edge down
75 %      u(2)      =      Aileron, dAr, rad, positive: left trailing ...
      edge down
76 %      u(3)      =      Rudder, dRr, rad, positive: trailing edge left
77 %      u(4)      =      Throttle, dT, %
78 %      u(5)      =      Asymmetric Spoiler, dASr, rad
79 %      u(6)      =      Flap, dFr, rad
80 %      u(7)      =      Stabilator, dSr, rad
81
82 %      ...
=====
83 %      USER INPUTS
84 %      =====
85 %      — Flags define which analyses or IC/Inputs will be engaged
86 %      =====
87 %      FLIGHT Flags (1 = ON, 0 = OFF)
88
89      MODEL      =      2;      % Aerodynamic model selection
90                                % 0: Incompressible flow, high angle of attack
91                                % 1: Compressible flow, low angle of attack
92                                % 2: User-Defined model
93      QUAT       =      0;      % 0: Rotation Matrix (DCM) from Euler Angles
94                                % 1: Rotation Matrix (DCM) from Quaternion
95      TRIM       =      1;      % Trim flag (= 1 to calculate trim @ I.C.)

```

```

96     LINEAR = 1;      % Linear model flag (= 1 to calculate and ...
        store F and G)
97     SIMUL  = 1;      % Flight path flag (= 1 for nonlinear simulation)
98     CONHIS = 0;      % Control history ON (= 1) or OFF (= 0)
99     RUNNING = 0;     % internal flag, -
100
101     GEAR   = 0;      % Landing gear DOWN (= 1) or UP (= 0)
102     SPOIL  = 0;      % Symmetric Spoiler DEPLOYED (= 1) or CLOSED ...
        (= 0)
103     dF     = 0;      % Flap setting, deg
104
105     disp(' ')
106     disp(['MODEL = ', num2str(MODEL), '   QUAT = ', num2str(QUAT)])
107     disp(['TRIM = ', num2str(TRIM), '   LINEAR = ', num2str(LINEAR)])
108     disp(['SIMUL = ', num2str(SIMUL)])
109     disp(' ')
110     if MODEL == 0
111         disp(['AeroModel = AeroModelAlpha'])
112     end
113     if MODEL == 1
114         disp(['AeroModel = AeroModelMach'])
115     end
116     if MODEL == 2
117         disp(['AeroModel = AeroModelUser'])
118     end
119
120 % Starting and Final Times for Simulation
121 % =====
122     ti     = 0;      % Initial time for simulation, sec
123     tf     = 60;     % Final time for simulation, sec
124     disp(' ')
125     disp('Simulation Interval')
126     disp(['Initial Time = ', num2str(ti), ' s, Final Time = ...
        ', num2str(tf), ' s'])
127
128 % Initial Altitude (ft), Indicated Airspeed (kt)
129 % =====
130 % These values specify the level-flight trim condition
131 % They also specify initial conditions unless overwritten
132
133 % HW 7 Specific I.C.
134
135     MIC    = 0.85;    % Initial Mach Number
136     hm     = 12000;   % Initial Altitude [m]
137     disp(' ')
138     disp('HW Part 1')
139     disp('=====')
140     disp(['Mach = ', num2str(MIC), ', Altitude, m = ', num2str(hm)])
141
142     [airDens, airPres, temp, soundSpeed] = Atmos(hm);
143     disp(' ')
144     disp(['Air Density      = ', num2str(airDens), ' kg/m^3, Air ...
        Pressure = ', ...
        num2str(airPres), ' N/m^2'])
145     disp(['Air Temperature = ', num2str(temp), ' K,           Sound Speed ...
        = ', ...
        num2str(soundSpeed), ' m/s'])
146
147
148
149     hft    = hm/0.3048; % Initial Altitude, ft
150     VIC    = MIC*soundSpeed; % Initial True Airspeed, m/s

```

```

151 VTASkt = VIC*1.94384; % Initial True Airspeed, kt
152 VKIAS = VTASkt*sqrt(airDens/1.225); % Initial Indicated ...
    Airspeed, kt
153 disp(' ')
154 disp(['hft = ',num2str(hft),' , True Airspeed, m/s = ',num2str(VIC)])
155 disp(['VTASkt = ',num2str(VTASkt),' , VKIAS = ',num2str(VKIAS)])
156
157 % hft = 34450; % Altitude above Sea Level, ft
158 % VKIAS = 244; % Indicated Airspeed, kt
159
160 hm = hft * 0.3048; % Altitude above Sea Level, m
161 VmsIAS = VKIAS * 0.514444; % Indicated Airspeed, m/s
162 disp(' ')
163 disp('Initial Conditions')
164 disp('=====')
165 disp(['Altitude = ',num2str(hft),' ft, = ...
    ',num2str(hm),' m'])
166 disp(['Indicated Airspeed = ',num2str(VKIAS),' kt, = ...
    ',num2str(VmsIAS),' m/s'])
167
168 % US Standard Atmosphere, 1976, Table Lookup for I.C.
169 [airDens,airPres,temp,soundSpeed] = Atmos(hm);
170 disp(' ')
171 disp(['Air Density = ',num2str(airDens),' kg/m^3, Air ...
    Pressure = ', ...
    num2str(airPres),' N/m^2'])
172
173 disp(['Air Temperature = ',num2str(temp),' K, Sound Speed ...
    = ', ...
    num2str(soundSpeed),' m/s'])
174
175
176 % Dynamic Pressure (N/m^2), and True Airspeed (m/s)
177 qBarSL = 0.5*1.225*VmsIAS^2; % Dynamic Pressure at sea level, ...
    N/m^2
178 V = sqrt(2*qBarSL/airDens); % True Airspeed, TAS, m/s
179 TASms = V;
180 disp(' ')
181 disp(['Dynamic Pressure = ',num2str(qBarSL),' N/m^2, True ...
    Airspeed = ', ...
    num2str(V),' m/s'])
182
183
184 D2R = pi/180;
185 R2D = 180/pi;
186
187 % Alphabetical List of Additional Initial Conditions
188 % =====
189 % These values subsume the hft and VKIAS entered above
190
191 alpha = 0; % Angle of attack, deg (relative to air mass)
192 beta = 0; % Sideslip angle, deg (relative to air mass)
193 dA = 0; % Aileron angle, deg
194 dAS = 0; % Asymmetric spoiler angle, deg
195 dE = 0; % Elevator angle, deg
196 dR = 0; % Rudder angle, deg
197 dS = 0; % Stabilator setting, deg
198 dT = 0; % Throttle setting, % / 100
199 hdot = 0; % Altitude rate, m/s
200 p = 0; % Body-axis roll rate, deg/s
201 phi = 0; % Body roll angle wrt earth, deg
202 psi = 0; % Body yaw angle wrt earth, deg
203 q = 0; % Body-axis pitch rate, deg/sec

```

```

204     r      = 0;      % Body-axis yaw rate, deg/s
205     SMI    = 0;      % Static margin increment due to ...
                center-of-mass variation
206                        % from reference, %/100
207     theta  = alpha; % Body pitch angle wrt earth, deg [theta = ...
                alpha if hdot = 0]
208     xe     = 0;      % Initial longitudinal position, m
209     ye     = 0;      % Initial lateral position, m
210     ze     = -hm;    % Initial vertical position, m [h: + up, z: + ...
                down]
211
212 % Initial Conditions Derived from Prior Initial Conditions
213 gamma    = R2D * atan(hdot / sqrt(V^2 - hdot^2));
214                        % Inertial Vertical Flight Path Angle, deg
215     qbar   = 0.5 * airDens * V^2;
216                        % Dynamic Pressure, N/m^2
217     IAS    = sqrt(2 * qbar / 1.225);
218                        % Indicated Air Speed, m/s
219     Mach   = V / soundSpeed;
220                        % Mach Number
221     disp(['Mach number      = ', num2str(Mach), ' ...
222           ', Flight Path Angle = ', num2str(gamma), ' deg'])
223     disp(' ')
224     uInc   = [];
225
226 % AeroModel Selected by MODEL Flag
227 % =====
228     if MODEL == 0
229         disp('<<Low-Mach, High-Alpha Model, AeroModelAlpha>>')
230     end
231     if MODEL == 1
232         disp('<<High-Mach, Low-Alpha Aerodynamic Model, AeroModelMach>>')
233     end
234     if MODEL == 2
235         disp('<<User-Defined Aerodynamic Model, AeroModelUser>>')
236     end
237     disp(' =====')
238
239 % Test Inputs at Initial Condition
240 % =====
241 % Initial Control Perturbation (Test Inputs: rad or percent)
242     delu   = [0;0;0;0;0;0;0;0];
243 % Initial State Perturbation (Test Inputs: m, m/s, rad, or rad/s)
244     delx   = [0;0;0
245               0;0;0
246               0;0;0
247               0;0;0];
248     disp(' ')
249     disp('Initial Perturbations to Trim for Step Response')
250     disp('=====')
251
252     disp('Control Vector')
253     disp('-----')
254     disp(['Elevator   = ', num2str(delx(1)), ' rad, Aileron = ', ...
255           num2str(delx(2)), ' rad, Rudder = ', num2str(delx(3)), ' rad'])
256     disp(['Throttle   = ', num2str(delx(4)), ' x 100%, Asymm Spoiler = ...
257           ', ...
258           num2str(delx(5)), ' rad, Flap = ', num2str(delx(6)), ' rad'])
259     disp(['Stabilator = ', num2str(delx(7)), ' rad'])

```

```

260     disp(' ')
261     disp('State Vector')
262     disp('-----')
263     disp(['u   = ', num2str(delx(1)), ' m/s, v = ', num2str(delx(2)), ' ...
           m/s, w = ', ...
264           num2str(delx(3)), ' m/s'])
265     disp(['x   = ', num2str(delx(4)), ' m, y = ', num2str(delx(5)), ' m, ...
           z = ', ...
266           num2str(delx(6)), ' m'])
267     disp(['p   = ', num2str(delx(7)), ' rad/s, q = ', num2str(delx(8)), ' ...
           rad/s, r = ', ...
268           num2str(delx(9)), ' rad/s'])
269     disp(['Phi = ', num2str(delx(10)), ' rad, Theta = ...
           ', num2str(delx(11)), ' rad, Psi = ', ...
270           num2str(delx(12)), ' rad'])
271
272 %   Control Perturbation History (Test Inputs: rad or 100%)
273 %   =====
274 %   Each control effector represented by a column
275 %   Each row contains control increment  $\Delta u(t)$  at time t:
276     disp(' ')
277     disp('Control Vector Time History Table')
278     disp('=====')
279     disp('Time, sec: t0, t1, t2, ...')
280     tuHis   =   [0 5 10 60]
281
282     disp('Columns: Elements of the Control Vector')
283     disp('Rows:      Value at time, t0, t1, ...')
284     deluHis =   [0 0 0 0 0 0 0
285                 0 0 0 0 0 0 0
286                 0 0 0 0 0 0 0
287                 0 0 0 0 0 0 0]
288
289 %   State Vector and Control Initialization, rad
290     phir   =   phi * D2R;
291     thetar =   theta * D2R;
292     psir   =   psi * D2R;
293
294     windb  =   WindField(-ze, phir, thetar, psir);
295     alphas =   alpha * D2R;
296     betar  =   beta * D2R;
297
298     x      =   [V * cos(alphas) * cos(betar) - windb(1)
299               V * sin(betar) - windb(2)
300               V * sin(alphas) * cos(betar) - windb(3)
301               xe
302               ye
303               ze
304               p * D2R
305               q * D2R
306               r * D2R
307               phir
308               thetar
309               psir];
310
311     u      =   [dE * D2R
312               dA * D2R
313               dR * D2R
314               dT
315               dAS * D2R

```



```

316         dF * D2R
317         dS * D2R];
318
319 % Trim Calculation (for Steady Level Flight at Initial V and h)
320 % =====
321 % Euler Angles used in trim calculation
322 % Trim Parameter Vector (OptParam):
323 %     1 = Stabilator, rad
324 %     2 = Throttle, %
325 %     3 = Pitch Angle, rad
326
327 if TRIM ≥ 1
328     disp(' ')
329     disp('TRIM Stabilator, Thrust, and Pitch Angle')
330     disp('=====')
331     OptParam      =   [];
332     TrimHist      =   [];
333 % Arbitrary starting values (user-selected, e.g., best guess at ...
334 % solution)
335     InitParam     =   [0;0;0]; %[0.0369;0.1892;0.0986];
336 %
337 % Application of 'fminsearch' to minimize Trim Cost
338 % options = optimset('TolFun',1e-10);
339 % [OptParam,J,ExitFlag,Output] = ...
340 %     fminsearch('TrimCost',InitParam,options);
341 %
342 % disp(['Trim Cost = ',num2str(J),' , Exit Flag = ...
343 %     ',num2str(ExitFlag)])
344 % Output
345 % Optimizing Trim Error Cost with respect to dSr, dT, and Theta
346 % TrimHist;
347 % Index= [1:length(TrimHist)];
348 % TrimStabDeg      =   R2D*OptParam(1);
349 % TrimThrusPer     =   100*OptParam(2);
350 % TrimPitchDeg     =   R2D*OptParam(3);
351 % TrimAlphaDeg     =   TrimPitchDeg - gamma;
352 % disp(['Stabilator = ',num2str(TrimStabDeg),' deg, Thrust = ...
353 %     ', ...
354 %     num2str(TrimThrusPer),' x 100%'])
355 % disp(['Pitch Angle = ',num2str(TrimPitchDeg),' deg, Angle of ...
356 %     Attack = ',...
357 %     num2str(TrimAlphaDeg),' deg'])
358 %
359 % Insert trim values in nominal control and state vectors
360 % disp(' ')
361 % disp('Trimmed Initial Control and State Vectors')
362 % disp('=====')
363 % u      =   [u(1)
364 %             u(2)
365 %             u(3)
366 %             OptParam(2)
367 %             u(5)
368 %             u(6)
369 %             OptParam(1)];
370 % format long
371 % x      =   [V * cos(OptParam(3))
372 %             x(2)
373 %             V * sin(OptParam(3))
374 %             x(4)
375 %             x(5)

```

```

371         x(6)
372         x(7)
373         x(8)
374         x(9)
375         x(10)
376         OptParam(3)
377         x(12)];
378     disp('Control Vector')
379     disp('_____')
380     disp(['Elevator   = ', num2str(u(1)), ' rad, Aileron = ...
381           ', num2str(u(2)), '...
382           ' rad, Rudder = ', num2str(u(3)), ' rad'])
383     disp(['Throttle   = ', num2str(u(4)), ' x 100%, Asymm Spoiler = ...
384           ', ...
385           num2str(u(5)), ' rad, Flap = ', num2str(u(6)), ' rad'])
386     disp(['Stabilator = ', num2str(u(7)), ' rad'])
387
388     disp(' ')
389     disp('State Vector')
390     disp('_____')
391     disp(['u   = ', num2str(x(1)), ' m/s, v = ', num2str(x(2)), ' ...
392           m/s, w = ', ...
393           num2str(x(3)), ' m/s'])
394     disp(['x   = ', num2str(x(4)), ' m, y = ', num2str(x(5)), ' m, z ...
395           = ', ...
396           num2str(x(6)), ' m'])
397     disp(['p   = ', num2str(x(7)), ' rad/s, q = ', num2str(x(8)), ' ...
398           rad/s, r = ', ...
399           num2str(x(9)), ' rad/s'])
400     disp(['Phi = ', num2str(x(10)), ' rad, Theta = ...
401           ', num2str(x(11)), ' rad, Psi = ', ...
402           num2str(x(12)), ' rad'])
403     disp(' ')
404     format short
405 end
406
407 figure
408 subplot(1,2,1)
409 plot(Index, TrimHist(1,:), 'b', Index, TrimHist(2,:), 'g', Index, TrimHist(3,:), 'r'),
410     grid, legend('Stabilator', 'Thrust', 'Pitch ...
411                 Angle'), xlabel('Iterations'),
412     ylabel('Stabilator(blue), Thrust(green), Pitch Angle(red)'), ...
413
414 title('Trim Parameters'), legend('Stabilator, rad', 'Thrust, ...
415     100%', ...
416     'Pitch Angle, rad')
417 subplot(1,2,2)
418 semilogy(Index, TrimHist(4,:))
419 xlabel('Iterations'), ylabel('Trim Cost'), grid
420 title('Trim Cost')
421
422 % Stability-and-Control Derivative Calculation
423 % =====
424 if LINEAR >= 1
425     disp(' ')
426     disp('Generate and Save LINEAR MODEL')
427     disp('=====')
428     thresh = ...
429             [.1;.1;.1;.1;.1;.1;.1;.1;.1;.1;.1;.1;.1;.1;.1;.1;.1;.1;.1;.1];
430     xj      = [x;u];

```

```

422     uTemp    =    u;    % 'numjac' modifies 'u'; reset 'u' after the call
423     xdotj    =    LinModel(ti,xj);
424     [dFdX, fac] = numjac('LinModel',ti,xj,xdotj,thresh,[],0);
425     u        =    uTemp;
426     Fmodel   =    dFdX(1:12,1:12)
427     Gmodel   =    dFdX(1:12,13:19)
428     save ('FmodelFile','Fmodel','TASms','hm','TrimAlphaDeg')
429     save ('GmodelFile','Gmodel')
430 end
431
432 % Flight Path Simulation, with Quaternion Option
433 % =====
434 if SIMUL ≥ 1
435     RUNNING = 1;
436     tspan   = [ti tf];
437     xo      = x + delx;
438     u       = u + delu;
439
440     disp('Trimmed Initial Control and State Vectors PLUS Test ...
441           Inputs')
442     disp('=====')
443     disp('Control Vector')
444     disp('-----')
445     disp(['Elevator   = ',num2str(u(1)), ' rad, Aileron = ...
446           ',num2str(u(2)), ' rad, Rudder = ',num2str(u(3)), ' rad'])
447     disp(['Throttle   = ',num2str(u(4)), ' x 100%, Asymm Spoiler = ...
448           ',num2str(u(5)), ' rad, Flap = ',num2str(u(6)), ' rad'])
449     disp(['Stabilator = ',num2str(u(7)), ' rad'])
450
451     disp(' ')
452     disp('State Vector')
453     disp('-----')
454     disp(['u   = ',num2str(xo(1)), ' m/s, v = ',num2str(xo(2)), ' ...
455           m/s, w = ',num2str(xo(3)), ' m/s'])
456     disp(['x   = ',num2str(xo(4)), ' m, y = ',num2str(xo(5)), ' m, ...
457           z = ',num2str(xo(6)), ' m'])
458     disp(['p   = ',num2str(xo(7)), ' rad/s, q = ',num2str(xo(8)), ' ...
459           rad/s, r = ',num2str(xo(9)), ' rad/s'])
460     disp(['Phi = ',num2str(xo(10)), ' rad, Theta = ...
461           ',num2str(xo(11)), ' rad, Psi = ',num2str(xo(12)), ' rad'])
462     disp(' ')
463 % Choice of Euler Angles or Quaternions to define Rotation Matrix
464 % in Equations of Motion for Simulation
465 % ...
466 % =====
467 switch QUAT
468 case 0
469     % Trajectory Calculation using Euler Angles
470     % =====
471     disp('Use Euler Angles to form Rotation Matrix')
472     disp('=====')
473     options = odeset('Events',@event,'RelTol',1e-7,'AbsTol',1e-7);

```

```

474     tic
475     [t,x] = ode45(@EoM,tspan,xo,options);
476     toc
477     kHis = length(t);
478     disp(['# of Time Steps = ',num2str(kHis)])
479     disp(' ')
480
481     case 1
482     %     Trajectory Calculation using Quaternions
483     %     =====
484     disp('Use Quaternion to form Rotation Matrix')
485     disp('=====')
486     Ho = DCM(xo(10), xo(11), xo(12));
487     q4o = 0.5*sqrt(1 + Ho(1,1) + Ho(2,2) + Ho(3,3));
488     q1o = (Ho(2,3) - Ho(3,2)) / (4*q4o);
489     q2o = (Ho(3,1) - Ho(1,3)) / (4*q4o);
490     q3o = (Ho(1,2) - Ho(2,1)) / (4*q4o);
491     xoQ = [xo(1:9); q1o; q2o; q3o; q4o];
492
493     options = odeset('Events',@event,'RelTol',1e-7,'AbsTol',1e-7);
494
495     tic
496     [tQ,xQ] = ode15s(@EoMQ,tspan,xoQ,options);
497     toc
498     kHisQ = length(tQ);
499     disp(['# of Time Steps = ',num2str(kHisQ)])
500     disp(' ')
501     q1s = xQ(:,10);
502     q2s = xQ(:,11);
503     q3s = xQ(:,12);
504     q4s = xQ(:,13);
505     Phi = (atan2(2*(q1s.*q4s + q2s.*q3s), (1 - 2*(q1s.^2 + ...
506         q2s.^2))))*180/pi;
507     PhiR = Phi*pi/180;
508     Theta = (asin(2*(q2s.*q4s - q1s.*q3s)))*180/pi;
509     ThetaR = Theta*pi/180;
510     Psi = (atan2(2*(q3s.*q4s + q1s.*q2s), (1 - 2*(q2s.^2 + ...
511         q3s.^2))))*180/pi;
512     PsiR = Psi*pi/180;
513     qMag = sqrt(q1s.^2 + q2s.^2 + q3s.^2 + q4s.^2);
514
515     t = tQ;
516     x = [];
517     x = [xQ(:,1:9),PhiR(:),ThetaR(:),PsiR(:)];
518     kHis = kHisQ;
519 end
520
521 %     Plot Control History
522 figure
523 subplot(2,2,1)
524 plot(tuHis,R2D*deluHis(:,1) + R2D*delu(1), tuHis, ...
525     R2D*deluHis(:,7) + R2D*delu(7))
526 xlabel('Time, s'), ylabel('Elevator (blue), Stabilator ...
527     (green), deg'), grid
528 title('Pitch Test Inputs'), legend('Elevator, dE', ...
529     'Stabilator, dS')
530 subplot(2,2,2)
531 plot(tuHis,R2D*deluHis(:,2) + R2D*delu(2), tuHis, ...
532     R2D*deluHis(:,3) + R2D*delu(3), tuHis, R2D*deluHis(:,5) + ...
533     R2D*delu(5))

```

```

527     xlabel('Time, s'), ylabel('Aileron (blue), Rudder (green), ...
        Asymmetric Spoiler (red), deg'), grid
528     title('Lateral-Directional Test Inputs'), legend('Aileron, ...
        dA', 'Rudder, dR', 'Asymmetric Spoiler, dAS')
529     subplot(2,2,3)
530     plot(tuHis, deluHis(:,4) + delu(4))
531     xlabel('Time, s'), ylabel('Throttle Setting'), grid
532     title('Throttle Test Inputs')
533     subplot(2,2,4)
534     plot(tuHis,R2D*deluHis(6) + R2D*delu(6))
535     xlabel('Time, s'), ylabel('Flap, deg'), grid
536     title('Flap Test Inputs')
537
538 % Plot State History
539 figure
540 subplot(2,2,1)
541 plot(t,x(:,1))
542 xlabel('Time, s'), ylabel('Axial Velocity (u), m/s'), grid
543 title('Forward Body-Axis Component of Inertial Velocity, u')
544 subplot(2,2,2)
545 plot(t,x(:,2))
546 xlabel('Time, s'), ylabel('Side Velocity (v), m/s'), grid
547 title('Side Body-Axis Component of Inertial Velocity, v')
548 subplot(2,2,3)
549 plot(t,x(:,3))
550 xlabel('Time, s'), ylabel('Normal Velocity (w), m/s'), grid
551 title('Normal Body-Axis Component of Inertial Velocity, z')
552 subplot(2,2,4)
553 plot(t,x(:,1),t,x(:,2),t,x(:,3))
554 xlabel('Time, s'), ylabel('u (blue), v (green), w (red), ...
        m/s'), grid
555 title('Body-Axis Component of Inertial Velocity')
556 legend('Axial velocity, u', 'Side velocity, v', 'Normal ...
        velocity, w')
557
558 figure
559 subplot(3,2,1)
560 plot(t,x(:,4))
561 xlabel('Time, s'), ylabel('North (x), m'), grid
562 title('North Location, x')
563
564 subplot(3,2,2)
565 plot(t,x(:,5))
566 xlabel('Time, s'), ylabel('East (y), m'), grid
567 title('East Location, y')
568
569 subplot(3,2,3)
570 plot(t,-x(:,6))
571 xlabel('Time, s'), ylabel('Altitude (-z), m'), grid
572 title('Altitude, -z')
573
574 subplot(3,2,4)
575 plot((sqrt(x(:,4).*x(:,4) + x(:,5).*x(:,5))),-x(:,6))
576 xlabel('Ground Range, m'), ylabel('Altitude, m'), grid
577 title('Altitude vs. Ground Range')
578
579 subplot(3,2,5)
580 plot(x(:,4),x(:,5))
581 xlabel('North, m'), ylabel('East, m'), grid
582 title('Ground Track, North vs. East')

```

```

583
584     subplot(3,2,6)
585     plot3(x(:,4),x(:,5),-x(:,6))
586     xlabel('North, m'), ylabel('East, m'), zlabel('Altitude, m'), ...
        grid
587     title('3D Flight Path')
588
589     figure
590     subplot(2,2,1)
591     plot(t,x(:,7) * R2D)
592     xlabel('Time, s'), ylabel('Roll Rate (p), deg/s'), grid
593     title('Body-Axis Roll Component of Inertial Rate, p')
594     subplot(2,2,2)
595     plot(t,x(:,8) * R2D)
596     xlabel('Time, s'), ylabel('Pitch Rate (q), deg/s'), grid
597     title('Body-Axis Pitch Component of Inertial Rate, q')
598     subplot(2,2,3)
599     plot(t,x(:,9) * R2D)
600     xlabel('Time, s'), ylabel('Yaw Rate (r), deg/s'), grid
601     title('Body-Axis Yaw Component of Inertial Rate, r')
602     subplot(2,2,4)
603     plot(t,x(:,7) * R2D,t,x(:,8) * R2D,t,x(:,9) * R2D)
604     xlabel('Time, s'), ylabel('p (blue), q (green), r (red), ...
        deg/s'), grid
605     title('Body-Axis Inertial Rate Vector Components')
606     legend('Roll rate, p', 'Pitch rate, q', 'Yaw rate, r')
607
608     figure
609     subplot(2,2,1)
610     plot(t,x(:,10) * R2D)
611     xlabel('Time, s'), ylabel('Roll Angle (phi), deg'), grid
612     title('Earth-Relative Roll Attitude')
613     subplot(2,2,2)
614     plot(t,x(:,11) * R2D)
615     xlabel('Time, s'), ylabel('Pitch Angle (theta), deg'), grid
616     title('Earth-Relative Pitch Attitude')
617     subplot(2,2,3)
618     plot(t,x(:,12) * R2D)
619     xlabel('Time, s'), ylabel('Yaw Angle (psi, deg)'), grid
620     title('Earth-Relative Yaw Attitude')
621     subplot(2,2,4)
622     plot(t,x(:,10) * R2D,t,x(:,11) * R2D,t,x(:,12) * R2D)
623     xlabel('Time, s'), ylabel('phi (blue), theta (green), psi ...
        (red), deg'), grid
624     title('Euler Angles')
625     legend('Roll angle, phi', 'Pitch angle, theta', 'Yaw angle, psi')
626
627     VAirRel      = [];
628     vEarth       = [];
629     AlphaAR     = [];
630     BetaAR      = [];
631     windBody    = [];
632     airDensHis  = [];
633     soundSpeedHis = [];
634     qbarHis     = [];
635     GammaHis    = [];
636     XiHis       = [];
637
638     for i = 1:kHis
639         windb      = ...

```

```

        WindField(-x(i,6),x(i,10),x(i,11),x(i,12));
640     windBody      = [windBody windb];
641     [airDens,airPres,temp,soundSpeed] = Atmos(-x(i,6));
642     airDensHis    = [airDensHis airDens];
643     soundSpeedHis = [soundSpeedHis soundSpeed];
644     end
645
646     vBody          = [x(:,1) x(:,2) x(:,3)]';
647     vBodyAir      = vBody + windBody;
648
649     for i = 1:kHis
650         vE         = DCM(x(i,10),x(i,11),x(i,12))' * ...
        [vBody(1,i);vBody(2,i);vBody(3,i)];
651         VER        = sqrt(vE(1)^2 + vE(2)^2 + vE(3)^2);
652         VAR        = sqrt(vBodyAir(1,i)^2 + vBodyAir(2,i)^2 + ...
        vBodyAir(3,i)^2);
653         VARB       = sqrt(vBodyAir(1,i)^2 + vBodyAir(3,i)^2);
654
655         if vBodyAir(1,i) ≥ 0
656             Alphas = asin(vBodyAir(3,i) / VARB);
657         else
658             Alphas = pi - asin(vBodyAir(3,i) / VARB);
659         end
660
661         AlphaAR    = [AlphaAR Alphas];
662         Betar      = asin(vBodyAir(2,i) / VAR);
663         BetaAR    = [BetaAR Betar];
664         vEarth    = [vEarth vE];
665         Xir       = asin(vEarth(2,i) / sqrt((vEarth(1,i))^2 + ...
        (vEarth(2,i))^2));
666         if vEarth(1,i) ≤ 0 && vEarth(2,i) ≤ 0
667             Xir    = -pi - Xir;
668         end
669         if vEarth(1,i) ≤ 0 && vEarth(2,i) ≥ 0
670             Xir    = pi - Xir;
671         end
672         Gammar    = asin(-vEarth(3,i) / VER);
673         GammaHis  = [GammaHis Gammar];
674         XiHis     = [XiHis Xir];
675         VAirRel   = [VAirRel VAR];
676     end
677
678     MachHis      = VAirRel ./ soundSpeedHis;
679     AlphaDegHis  = R2D * AlphaAR;
680     BetaDegHis   = R2D * BetaAR;
681     qbarHis      = 0.5 * airDensHis .* VAirRel.*VAirRel;
682     GammaDegHis  = R2D * GammaHis;
683     XiDegHis     = R2D * XiHis;
684
685     figure
686     subplot(3,1,1)
687     plot(t, VAirRel')
688     xlabel('Time, s'), ylabel('Air-relative Speed, m/s'), grid
689     title('True AirSpeed, Vair')
690     subplot(3,1,2)
691     plot(t, MachHis')
692     xlabel('Time, s'), ylabel('M'), grid
693     title('Mach Number, M')
694     subplot(3,1,3)
695     plot(t, qbarHis')

```

```

696     xlabel('Time, s'), ylabel('qbar, N/m^2'), grid
697     title('Dynamic Pressure, qbar')
698
699     figure
700     subplot(2,1,1)
701     plot(t, AlphaDegHis', t, BetaDegHis')
702     xlabel('Time, s'), ylabel('Angle of Attack, deg (blue), ...
        Sideslip Angle, deg (green)'), grid
703     title('Aerodynamic Angles'), legend('Angle of Attack, alpha', ...
        'Sideslip Angle, beta')
704     subplot(2,1,2)
705     plot(t, GammaDegHis', t, XiDegHis')
706     xlabel('Time, s'), ylabel('Vertical, deg (blue), Horizontal, ...
        deg (green)'), grid
707     title('Flight Path Angles'), legend('Flight Path Angle, ...
        gamma', 'Heading Angle, psi')
708
709     'End of FLIGHT Simulation'
710     end

```

*AeromodelUser.m* is used to insert and calculate the aerodynamic coefficients of the TigerPounce, and is called by FLIGHT.m.

#### AeroModelUser.m

```

1     function [CD,CL,CY,Cl,Cm,Cn,Thrust] = ...
        AeroModelUser(x,u,Mach,alphan,betar,V)
2     % Tiger Pounce CESTOL Aerodynamic Coefficients, Thrust Model,
3     % and Geometric and Inertial Properties for FLIGHT.m
4     % =====
5     % Copyright 2006–2018 by ROBERT F. STENGEL. All rights reserved.
6     % Called by:
7     %     EoM.m
8     %     EoMQ.m
9     % Values either taken from CAD model, designed, or open VSP
10
11     global m Ixx Iyy Izz Ixz S b cBar
12
13     % Mass, Inertial, and Reference Properties
14     m           = 4410;      %*0.83;   % Take off Weight [kg]
15     Ixx         = 1.6202e5; %*0.83;   % XX inertia [Nm]
16     Iyy         = 2.7286e5; %*0.83;   % YY inertia [Nm]
17     Izz         = 4.2209e5; %*0.83;   % ZZ inertia [Nm]
18     Ixz         = -45900;   %*0.83;   % XZ inertia [Nm]
19     cBar        = 1.56;     % M.A.C (VSP) [m]
20     b           = 12.41;    % Wing Span [m]
21     S           = 18.11;    % Wing Area [m]
22     lHT         = 2.9;     % Length H-Tail [m]
23     lVT         = 1.98;    % Length V-Tail [m]
24     StaticThrust = 24910;  % Static Thrust [N]
25
26     [airDens,airPres,temp,soundSpeed] = Atmos(-x(6));
27     Thrust      = (u(4)*StaticThrust*(airDens/1.225)^0.7)*(1 - ...
        exp((-x(6) - 17000)/2000));
28
29     % Mach Number Effect on All Incompressible Coefficients
30     Prandtl     = 1;      % Prandtl Factor (already taken into account by ...
        VSP)
31
        % so can ignore for these calculations

```



```

32
33
34 % Current Longitudinal Characteristics
35 % =====
36 % Lift Coefficient
37 CLo = 0.0027; % CL_0 (VSP)
38 CLa = 7.37; % (VSP)
39 CLq = 31.12; % (VSP)
40 CLqhat = CLq*2*V/cBar;
41 CLdE = 0.69; % (VSP)
42 CLdS = CLdE;
43
44 % Total Lift Coefficient, w/Mach Correction
45 CL = (CLo + CLa*alphan + CLq*x(8) + CLdS*u(7) + ...
      CLdE*u(1))*Prandtl;
46
47 % Drag Coefficient
48 CDo = 0.026513; % CD_0 (from VSP)
49 Epsilon = 0.093*Prandtl;
50
51 % Total Drag Coefficient, w/Mach Correction
52 CD = CDo*Prandtl + Epsilon*CL^2; % VSP
53
54 % Pitching Moment Coefficient
55 StaticMargin = 0.08; % Static Margin (designed)
56 Cmo = -0.16;
57 Cma = -CLa*StaticMargin;
58 Cmq = -82.27;
59 Cmqhat = Cmq*2*V/cBar;
60 CmV = 0.013;
61 CmdE = -3.18;
62 CmdS = CmdE;
63
64 % Total Pitching Moment Coefficient, w/Mach Correction
65 Cm = (Cmo + Cma*alphan + Cmq*x(8) + CmdS*u(7) + ...
      CmdE*u(1))*Prandtl;
66
67
68 % Current Lateral-Directional Characteristics
69 % =====
70 % Side-Force Coefficient
71 CYo = 0;
72 CYb = -0.744;
73 CYp = -0.2;
74 CYphat = CYp/(b/(2*V));
75 CYr = 0.638;
76 CYrhat = CYr/(b/(2*V));
77 CYdA = -0.04;
78 CYdR = 0;
79
80 % Total Side-Force Coefficient, w/Mach Correction
81 CYo = 0;
82 CY = (CYo + CYb*betar + CYdR*u(3) + CYdA*u(2) + CYp*x(7) ...
      + CYr*x(9))*Prandtl;
83
84
85 % Rolling Moment Coefficient
86 Clo = 0;
87 Clb = -0.036;
88 Clp = -0.692;
89 Clphat = Clp*2*V/b;
90 Clr = 0.08;

```

```

91     Clrhat = Clr/(b/(2*V));
92     CldA = -0.39;
93     CldR = 0;
94
95     % Total Rolling-Moment Coefficient, w/Mach Correction
96     Cl = (Cl0 + Clb*betar + Clp * x(7) + Clr * x(9) ...
97           + CldA*u(2) + CldR*u(3))* Prandtl;
98
99 % Yawing Moment Coefficient
100    Cno = 0;
101    CnBeta = 0.004;
102    Cnp = 0.077;
103    Cnphat = Cnp/(b/(2*V));
104    Cnr = -0.28;
105    Cnrhat = Cnr/(b/(2*V));
106    CndA = 0;
107    CndR = -0.2;
108
109 % Total Yawing-Moment Coefficient, w/Mach Correction
110    Cn = (Cno + CnBeta*betar + Cnp*x(7) + Cnr*x(9) ...
111          + CndA*u(2) + CndR*u(3))*Prandtl;

```

*BizAnalysis.m* is used to generate plots and price points for the TigerPounce as well as the HondaJet and Cessna Citation M2 aircraft. It is based on the Eastlake Model which is modified from DAPCA-IV. [19]

### BizAnalysiz.m

```

1
2
3 CPI = 1.12; % CPI for 2019 vs 2012
4 N = linspace(10,3000,2991); % Number of aircraft 5 years
5 Q_m = N/60; % Monthly production
6 QDF = .95.^(1.4427*log(N)); % Discount for quantity
7 % Assume .95 experience effectiveness
8 f_comp = [0.017;0;0]; % Fraction composite (or special material)
9
10
11 W_airframe = zeros(3,1);
12 V_max = zeros(3,1);
13 T = zeros(3,1);
14
15 W_airframe(1) = 8133; % lbs, weight of airframe
16 V_max(1) = 487; % kts, max cruising true airspeed
17 T = 2300; % lbs, thrust of one engine
18
19 W_airframe(2) = 7283; % lbs, weight of HondaJet
20 V_max(2) = 422; % kts, max cruise HondaJet
21 T(2) = 1997; % lbs, thrust for HondaJet
22
23 W_airframe(3) = 6990; % lbs, weight of Cessna Citation M2
24 V_max(3) = 404; % kts, max cruise Cessna Citation M2
25 T(3) = 1965; % lbs, thrust for Cessna Citation M2
26
27
28 %% Individual Costs
29
30 %Engineering

```

```

31 F_CF = 1; % 1 if flap, 1.03 if complex flap
32 F_COMP = 1 + f_comp; % Composite correction
33 F_PRESS = 1.03; % 1.03 for press., 1 if unpress.
34 F_CERT = 1; % Cost from extra certification (1.15 for FAR25)
35
36 H_ENG = 4.86 * W_airframe.^.777 .* V_max.^.894 * N.^.163 * ...
37 F_CERT * F_CF .* F_COMP * F_PRESS; % Engineering man-hours
38
39
40 R_ENG = 92; % $/hr for engineering work
41
42 C_ENG = 2.0969*CPI * R_ENG * H_ENG; % Engineering Cost
43
44
45 %Tooling
46 F_CF = 1; % 1 if simple flap, 1.02 if complex flap
47 F_COMP = 1 + f_comp; % Composite correction
48 F_PRESS = 1.01; % 1.01 for press., 1 if unpress.
49 F_CERT = 1; % Cost from extra certification (1.05 for FAR25)
50 F_TAPER = 1; % 1 for taper, .95 for constant chord
51
52 H_TOOL = 5.99 * W_airframe.^.777 .* V_max.^.696 * N.^.263 .* ...
53 Q_m.^.066 *...
54 F_CERT * F_CF .* F_COMP * F_PRESS * F_TAPER; % Tooling man-hours
55
56 R_TOOL = 61; % $/hr for tooling work
57
58 C_TOOL = 2.0969*CPI * H_TOOL * R_TOOL; % Tooling cost
59
60 %Manufacturing
61 F_CF = 1; % 1 if flap, 1.01 if complex flap
62 F_COMP = 1 + .25*f_comp; % Composite correction
63 F_CERT = 1; % Cost from extra certification (1.05 for FAR25)
64
65 H_MFG = 7.37 * W_airframe.^.82 .* V_max.^.484 * N.^.641 * ...
66 F_CERT * F_CF .* F_COMP; % Manufacturing man-hours
67
68
69 R_MFG = 53; % $/hr for manufacturing
70
71 C_MFG = 2.0969*CPI * H_MFG * R_MFG; % Manufacturing cost
72
73
74 % Development
75 F_CF = 1; % 1 if flap, 1.01 if complex flap
76 F_COMP = 1 + .5*f_comp; % Composite correction
77 F_PRESS = 1.03; % 1.03 for press., 1 if unpress.
78 F_CERT = 1; % Cost from extra certification (1.10 for FAR25)
79
80 C_DEV = 95.24 * W_airframe.^.63 .* V_max.^1.3 * CPI *...
81 F_CERT * F_CF .* F_COMP * F_PRESS; % Development Costs
82
83
84 %Flight Test
85 F_CERT = 1; % Cost from extra certification (1.5 for FAR25)
86
87 C_FT = 2606.51 * W_airframe.^.325 .* V_max.^.822 * N.^.21 * CPI * F_CERT;
88 % Flight Test Costs
89

```

```

90 %Quality Control
91 F_COMP = 1 + .5*f_comp; % Composite correction
92 F_CERT = 1; % Cost from extra certification (1.5 for FAR25)
93
94 C_QC = .133 * C_MFG * F_CERT .* F_COMP; % Quality Control Costs
95
96
97 %Materials
98 F_CF = 1; % 1 if flap, 1.02 if complex flap
99 F_PRESS = 1.01; % 1.01 for press., 1 if unpress.
100 F_CERT = 1; % Cost from extra certification (1.15 for FAR25)
101
102 C_MAT = 23.066 * W_airframe.^.921 .* V_max.^.621 * N.^.799 * CPI *...
103 F_CERT * F_CF .* F_COMP * F_PRESS; % Development Costs
104
105
106
107 % Landing Gear
108 % Assumed to already have retracting gear, if it doesn't subtract
109 % $7500/aircraft
110
111
112 %Avionics
113 C_AV = [60000;60000;60000] * CPI; % Avionics cost 60000/airplane FAR 23,
114 % 100000/aiplane FAR 25 incl. entertain
115
116
117 %Power Plant
118 N_PP = 2; % Number of engines
119
120 C_PP = 1035.9 * N_PP * T.^.8356 * CPI;
121
122
123
124 %% Cost Build Up
125
126 %Certification
127 C_CERT = C_ENG + C_DEV + C_FT + C_TOOL;
128
129 C_ENG_PER = C_ENG./N;
130 C_DEV_PER = C_DEV./N;
131 C_FT_PER = C_FT./N;
132 C_TOOL_PER = C_TOOL./N;
133
134 C_MFG_PER = C_MFG./N;
135 C_QC_PER = C_QC./N;
136 C_MAT_PER = C_MAT./N;
137
138 C_PP_PER = [C_PP(:,1) .* QDF;C_PP(:,1) .* QDF;C_PP(:,1) .* QDF;];
139 C_AV_PER = C_AV .* QDF;
140
141 C_TOT_PER = C_ENG_PER + C_DEV_PER + C_FT_PER + C_TOOL_PER +...
142 C_MFG_PER + C_QC_PER + C_MAT_PER + C_PP_PER + C_AV_PER;
143
144 C_INS_PER = 50000; % Manufacturing Insurance
145
146 C_VAR = C_MFG_PER + C_QC_PER + C_MAT_PER + C_PP_PER + C_AV_PER + ...
147 C_INS_PER;
148 %% Price Analysis

```

```

149
150 C_MIN_SELL = C_TOT_PER + C_INS_PER;
151
152 p = 3;           % Choose plane: 1 tpounce, 2 honda, 3 cessa
153 N_BE = [100; 200; 500; 1800] - 9;
154
155
156 SELL = zeros(1,length(N_BE));
157 REV = zeros(length(N_BE),length(N));
158 for i = 1:length(N_BE)
159     SELL(i) = C_CERT(p,N_BE(i)) / N_BE(i) + C_VAR(p,N_BE(i));
160     REV(i,:) = SELL(i) * N;
161 end
162
163
164 %% Plots
165
166 n = 591;
167 figure('Position', [400 10 1000 620])
168 ax = gca;
169 set(ax,'DefaultLineLineWidth',3)
170 hold on
171 plot(N(1:n),C_TOT_PER(p,1:n).*N(1:n)/1e9, 'k-')
172 plot(N(1:n),REV(1,1:n)/1e9, 'b:')
173 plot(N(1:n),REV(2,1:n)/1e9, 'b-')
174 plot(N(1:n),REV(3,1:n)/1e9, 'b-')
175 ax.YAxis.Exponent = 9;
176 legend('Total Costs','Revenue: N_{BE}=100', 'Revenue: N_{BE}=200',...
177        'Revenue: N_{BE}=500','location','northwest')
178 ylabel('Total Production Cost & Revenue in Billions of Dollars')
179 xlabel('Number of Aircraft Produced')
180 %
181 figure('Position', [400 10 1000 620])
182 ax = gca;
183 set(ax,'DefaultLineLineWidth',3)
184 hold on
185 plot(N(1:n),C_MIN_SELL(1,1:n)/1e6, 'k-')
186 plot(N(1:n),C_MIN_SELL(2,1:n)/1e6, 'b-')
187 plot(N(1:n),C_MIN_SELL(3,1:n)/1e6, 'c-')
188 legend('TigerPounce','HondJet','Citation M2','location','northeast')
189 ylabel('Minimum Selling Price in Millions of Dollars')
190 xlabel('Number of Aircraft Produced in First 5 Years')
191 %
192 figure('Position', [400 10 1000 620])
193 ax = gca;
194 set(ax,'DefaultLineLineWidth',2)
195 hold on
196 plot(N(1:n),C_ENG_PER(p,1:n)/1e6, 'r-')
197 plot(N(1:n),C_DEV_PER(p,1:n)/1e6, 'r:')
198 plot(N(1:n),C_FT_PER(p,1:n)/1e6, 'r-.')
199 plot(N(1:n),C_TOOL_PER(p,1:n)/1e6, 'r-')
200 plot(N(1:n),C_MFG_PER(p,1:n)/1e6, 'b-')
201 plot(N(1:n),C_QC_PER(p,1:n)/1e6, 'b:')
202 plot(N(1:n),C_MAT_PER(p,1:n)/1e6, 'b-')
203 plot(N(1:n),C_PP_PER(p,1:n)/1e6, 'k-')
204 plot(N(1:n),C_AV_PER(p,1:n)/1e6, 'k:')
205 legend('Engineering','Development','Flight Test','Tooling',...
206        'Manufacturing','Quality Control','Materials','Power ...
207        Plnat','Avionics')
207 ylim([0 8])

```

```

208 ylabel('Cost Breakdown Per Airplane (Millions of Dollars)')
209 xlabel('Number of Aircraft Produced in First 5 Years')

```

*LandingGear.m* is used size the landing gear for this aircraft. It uses equation from Raymer and tires from Goodyear.[\[3, 1\]](#)

### LandingGear.m

```

1 %% Tricycle Gear Arrangement
2
3 g = 32.174;          % ft/s^2, gravitational constant
4
5 MTOW = 9733;        % lb
6 V_stall = 167;      % ft/s
7 cgDiff = 1.43;      % ft, difference between foremost and aftmost cg
8 V_vertical = 15;    % ft/s, for STOL
9
10 mainWidth = 4.4;    % ft, width between main tires
11 H = 4.5;            % ft, height of c.g. above ground
12 tipBack = 19;       % deg, tip back angle
13 overTurn = 41.8;    % deg, overturn angle
14
15 noseWheels = 2;
16 mainWheels = 4;
17
18
19 tipBack_rad = deg2rad(tipBack);    % rad, tip back angle
20 overTurn_rad = deg2rad(overTurn);   % rad, overturn angle
21
22 Ma = tan(tipBack_rad)*H;    % ft, smallest x dist from cg to main
23 B = (mainWidth/2)/tan(overTurn_rad - atan(Ma/(mainWidth/2)));
24                               % ft, x dist between nose and main
25 Na = B - Ma;                % ft, largest x dist from cg to nose
26 Mf = Ma + cgDiff;          % ft, largest x dist from cg to main
27 Nf = Na - cgDiff;          % ft, smallest x dist from cg to nose
28
29 % Check that the nose has between 8 and 16 percent of load
30 Ma/B > .08
31 Mf/B < .16
32
33 % Calculate static loads
34 staticLoad_max_main = 1.25 * MTOW * Na/B;
35 staticLoad_max_nose = 1.25 * MTOW * Mf/B;
36 staticLoad_min_nose = 1.25 * MTOW * Ma/B;
37 % Calculate the max load on the nose wheel
38 dynamic_max_nose = 1.25 * 10*H*MTOW/(g*B);
39
40 % Weight per wheel on nose
41 Ww_nose_static = staticLoad_max_nose/noseWheels;
42 Ww_nose_dynamic = (staticLoad_max_nose + ...
43     dynamic_max_nose)/noseWheels/1.3;
44
45 % Weight per wheel on main
46 Ww_main = staticLoad_max_main/mainWheels;
47
48 % Compare to graph for rim diameter, check with tire
49 KEbrake = MTOW/(2*g)*V_stall^2;
50

```

```

51
52 %% Tire data
53
54 % ...
    https://www.goodyearaviation.com/resources/pdf/tire-specifications-6-2018.pdf
55
56 w_main = 4.45;           % in, max width
57 d_main = 17.9;          % in, max diameter
58 Rr_main = 7.9;           % in, rolling radius
59 P_max_main = 180;        % PSI, tire pressure
60
61 Ap_main = 2.3 * sqrt(w_main*d_main)*(d_main/2-Rr_main);
62 P_main = Ww_main / Ap_main;
63
64
65 w_nose = w_main;
66 d_nose = d_main;
67 Rr_nose = Rr_main;
68 P_max_nose = P_max_main;
69
70 P_nose = Ww_nose_static / Ap_main;
71
72
73 % Only if need drag savings
74 % w_nose = 4.45;           % in, max width
75 % d_nose = 16;            % in, max diameter
76 % Rr_nose = 6.9;          % in, rolling radius
77 % P_max_nose = 55;        % PSI, tire pressure
78 %
79 % Ap_nose = 2.3 * sqrt(w_nose*d_nose)*(d_nose/2-Rr_nose);
80 % P_nose = Ww_nose_static / Ap_nose;
81
82 %% Oleo Sizing
83
84 N_gear = 3;              % Standard Factor
85 eta_oleoFixed = .65;     % Worst case oleo efficiency
86 eta_tire = .47;          % Tire efficiency
87
88 S_main = V_vertical^2/(2*g*eta_oleoFixed*N_gear) - ...
    eta_tire/eta_oleoFixed * (d_main/2-Rr_main) + 1/12;
89                                     % ft, stroke length of shock absorber
90
91
92 mechAdv_main = 1;
93
94 oleoLoad_main = staticLoad_max_main/2; % Load on main oleo
95
96 S_main = S_main/mechAdv_main;
97 oleoLoad_main = oleoLoad_main * mechAdv_main;
98
99 d_oleo_main = .04*sqrt(oleoLoad_main); % in, diameter main oleo
100 l_oleo_main = 2.5*S_main / cos(tipBack_rad); % ft, length main oleo
101
102
103
104 S_nose = V_vertical^2/(2*g*eta_oleoFixed*N_gear) - ...
    eta_tire/eta_oleoFixed * (d_nose/2-Rr_nose) + 1/12;
105                                     % ft, stroke length of shock absorber
106
107
108 mechAdv_nose = 1;
109

```

```

110 trail = .2 * d_nose/2; % in, offset of oleo from hub
111 rake = 15; % deg, angle of nose gear
112
113 oleoLoad_nose = staticLoad_max_nose + dynamic_max_nose;
114
115 S_nose = S_nose/mechAdv_nose; % stroke length nose
116 oleoLoad_nose = oleoLoad_nose * mechAdv_nose; % nose oleo load
117
118 d_oleo_nose = .04*sqrt(oleoLoad_nose); % in, diameter nose oleo
119 l_oleo_nose = 2.5*S_nose / cos(deg2rad(rake)); % ft, lenth nose oleo

```

*RevisedSizing.m* calculates computes the takeoff weight and constraint plane given initial aerodynamic efficiency, thrust, and mission parameter values.

### RevisedSizing.m

```

1 %% Variables
2 close all
3 W_person = 200; %lb
4 W_bag = 100; %lb
5 N_people = 8; %#
6 R = 1500; %nm
7 M = .85; %
8 C = .348/3600; % lb/lbf-second
9 T_loiter = 0.5; %hr
10 H_cruise = 40000; %feet
11 V_cargo = N_people*W_bag/10; %ft^3
12 Wpayload = (W_person + W_bag)*N_people;
13 a = 967.70667; % ft/s
14 R_ft = R * (6076 + (1+25/64)/12); % range, ft
15 C = .7/3600; %Estimate based off of raymer table 3.3
16 AR_init = 8.5; %hondaJet
17 LD_init = 15.8; %
18 ARinit = 8.5; % HondaJet
19 CLmax = 4.3; %Raymer 96
20 CLaero = 2.4;
21 mu = .025; %ground friction, see ...
    http://www.dept.aoe.vt.edu/~lutze/AOE3104/takeoff&landing.pdf
22 g = 32.2; %ft/s^2
23 phi = .4778;
24 %% Weight Fractions
25 Wi1 = .97; %Raymer T3.2 warmup
26 Wi2 = .985; %Raymer T3.2 takeoff
27 Wi3 = exp((-R_ft*C)/(M*a*(LD_init))); % -RC/(V(L/D)), Raymer ...
    E3.6 cruise
28 Wi4 = .995; %Raymer T3.2 descent
29 Wx = Wi1*Wi2*Wi3*Wi4;
30 syms x
31 s = 1.25*x^(-.085); %Average jet trainer & transport, ...
    Raymer T3.1
32 Wto = vpsolve(x*(1-s)-1.06*(1-Wx)*x-Wpayload == 0, x);
33 %x*(1-1.25*x^(-.085))-1.06*(1-.8462)*x-2400 == 0
34 %% Engine/Wing Sizing
35 % L = q_infin * CL * S
36 % CL = beta*Wto/(q_infin*S)
37 % Beta = Wi/Wto
38 % CD = K1*CL^2 + CDo + CDr
39 % CDo = .003 Swet/Sref
40 % Fig 3.6 Raymer --> AR_wet = b^2/Swet ~ .8; (Raymer ch.3)

```



```

41 % Swet/Sref = AR/ARwet
42 CDo      = .003*AR_init/.8;
43 beta     = Wil*Wi2;
44 rho_40k  = 5.87*10^-4; % slug/ft^3
45 q_infin  = .5*rho_40k*(a*M)^2;
46 lambda_i = pi/6;
47 e = 4.61*(1-.045*ARinit^.68)*cos(lambda_i)^.15-3.1;
48 K1       = 1/(pi*ARinit*e);
49 alpha    = .375; % raymer 91,92
50 Wtos     = 1:100;
51 TWto     = ...
        (beta/alpha)*(K1*(beta/q_infin).*Wtos+CDo./((beta/q_infin)*Wtos));
52 %% Climb limit (10k')
53 rho_sl   = 0.0023768924; % slug/ft^3
54 dh       = 3990/60; % ft/s, hondajet comp.
55 alpha_c  = (17.56*10^-4)/rho_sl; %density proportional throttle ...
        lapse at 10k'
56 TWto3    = ...
        (Wil*Wi2/alpha_c)*((K1*beta/q_infin).*Wtos+CDo./((beta/q_infin)*Wtos)+66.5/(M*a));
57 %% Takeoff and landing
58 %TOP = (Wtos2)/(CLmax*(Tsl/Wto))
59 CLg     = mu/.08;
60 CDg     = .024+.04*CLg^2;
61 B       = .5*(CDg - mu*CLg)./Wtos;
62 syms TWtoX
63 A = g*(TWtoX*(cos(phi) + sin(phi))-mu);
64 TWto2   = 1:100;
65 % T/W climb
66 for     i = 1:100
67 B       = .5*rho_sl*(CDg - mu*CLg)/Wtos(i);
68 Vstall  = sqrt(Wtos(i)*2/(rho_sl*CLmax));
69 Vto     = Vstall*1.2;
70 TWto2(i) = vpasolve(1312 == (1/(2*B))*log(A/(A-B*Vto^2)), TWtoX);
71 end
72 Vstall = sqrt(Wtos*2/(rho_sl*CLmax));
73 sto    = 1000*4/3;
74 %TWto2 = (20.9*(Wtos)/(CLmax)+69.6*sqrt((Wtos)/CLmax))/sto;
75 syms WS
76 WS_land = vpasolve(1000 == .66*80*(Wil*Wi2*Wi3*WS)/(CLmax)+410, ...
        WS); % raymer E5.11
77 plot(Wtos(30:100), TWto(30:100),Wtos(30:100), ...
        sqrt((TWto2(30:100)).^2+((1-3/CLmax))^2), [WS_land, WS_land], ...
        [0,1], Wtos(30:100), TWto3(30:100))
78 hold on
79 scatter(56.5,.366,'filled');
80 title('Constraint Chart')
81 xlabel('Wto/S (Lb/ft^2)');
82 ylabel('Tsl/Wto');
83 %% Results
84 Wto
85 S = Wto/56.5
86 T = Wto*.366
87 b = sqrt(ARinit*S)
88 % TWopt = T/(Wto-124);
89 % WSopt = (Wto-124)/S;
90 % scatter(WSopt,TWopt,'filled');
91 % legend('Cruise', 'Takeoff', 'Landing', 'Climb', 'Initial Design ...
        Point', 'Optimized Design Point')
92 legend('Cruise', 'Takeoff', 'Landing', 'Climb', 'Initial Design Point')
93 %% Tail Sizing

```

```
94 % Lvt*Svt = .085*42.11*S --> Svt = 45.3
95 % mac = 6 (design v1)
96 % Lht*Sht = .85*6*S
97 % let lht,lvt = 18.5,16.5
98 Af = g*(.366*(cos(phi) + sin(phi))-mu);
99 Bf = .5*rho_sl*(CDg - mu*CLg)/56;
100 Vto = 200.5 %ft/s
```

## 12.5 Initial Condition Response Graphs

### 12.5.1 Responses using unaltered VSP coefficients

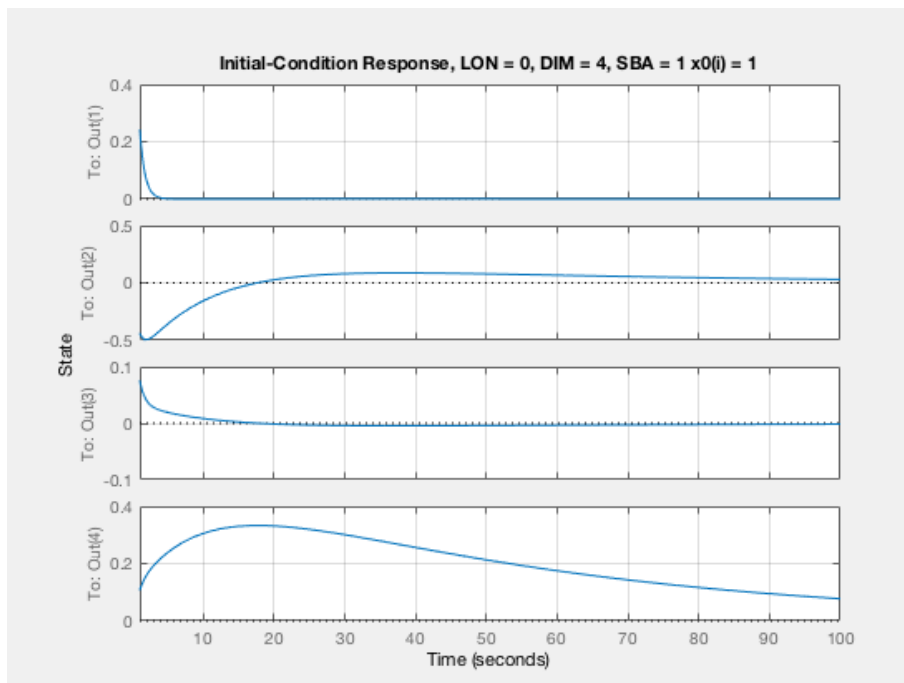


Figure 78: Lateral,  $x(1)$

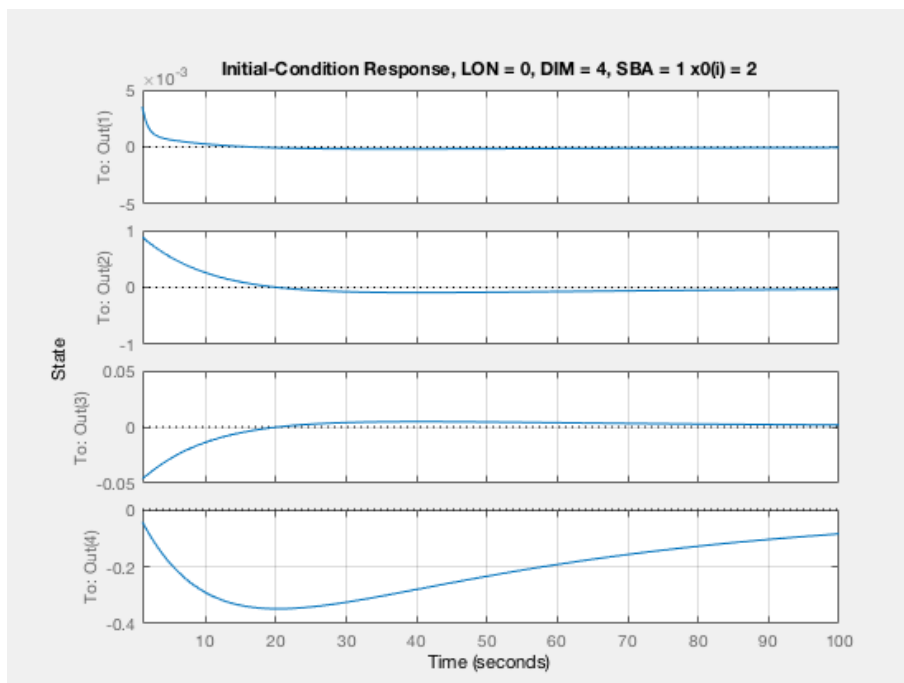


Figure 79: Lateral,  $x(2)$

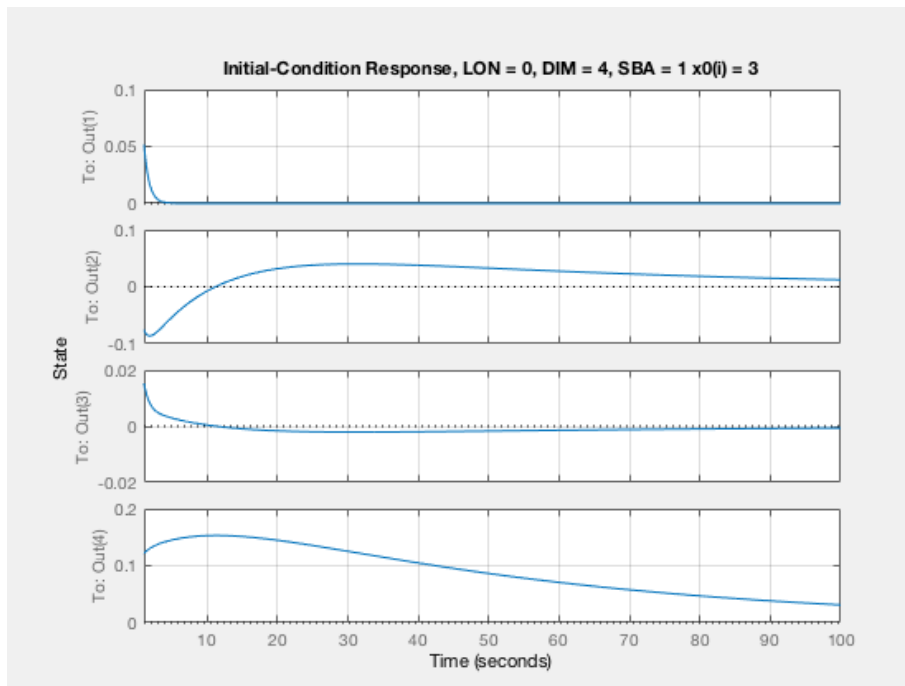


Figure 80: Lateral,  $x(3)$

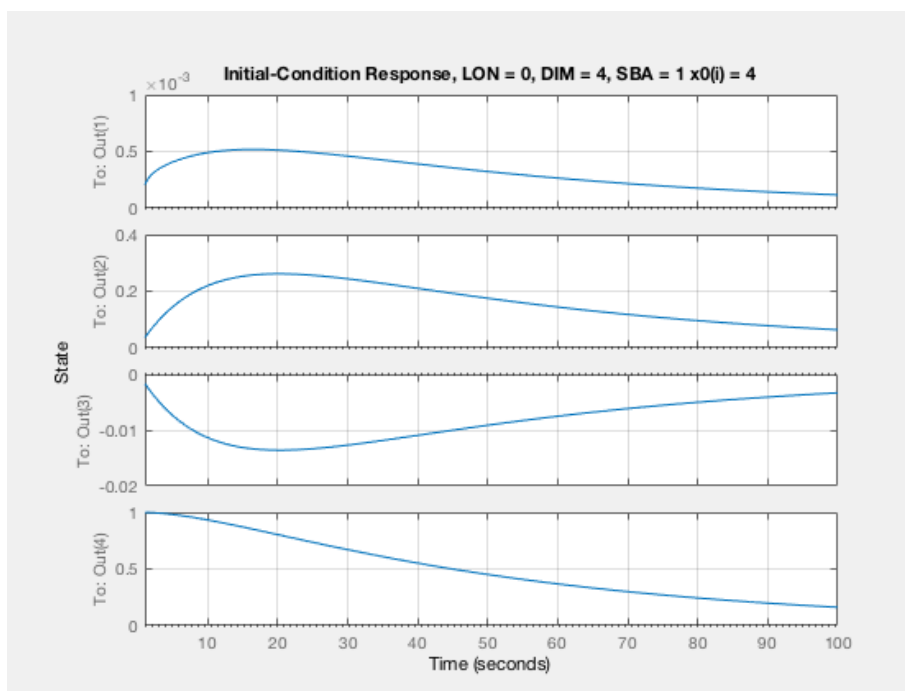


Figure 81: Lateral,  $x(4)$

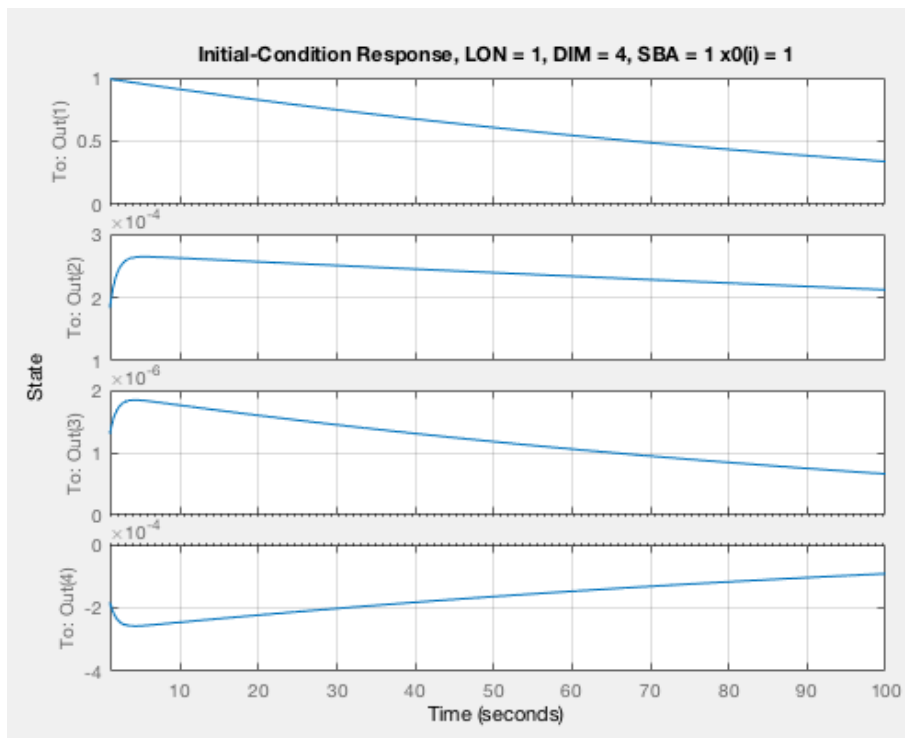


Figure 82: Longitudinal,  $x(1)$

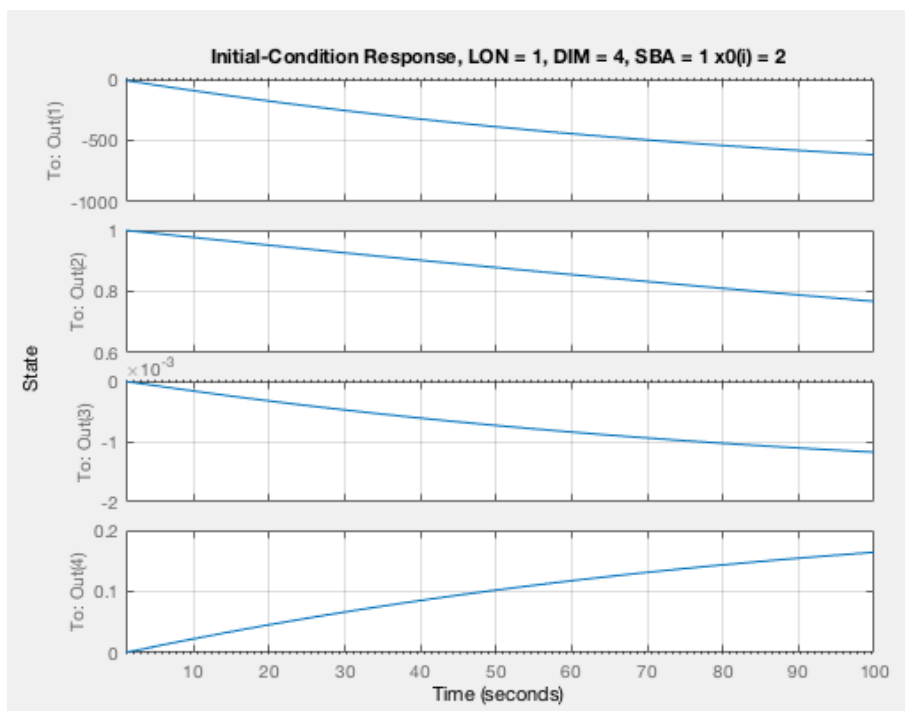


Figure 83: Longitudinal,  $x(2)$

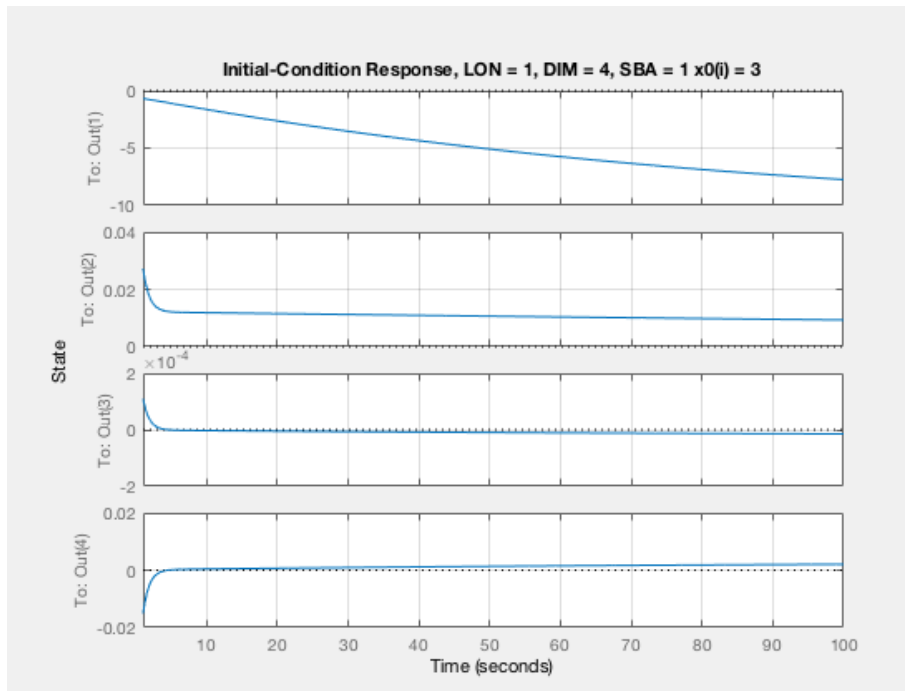


Figure 84: Longitudinal,  $x(3)$

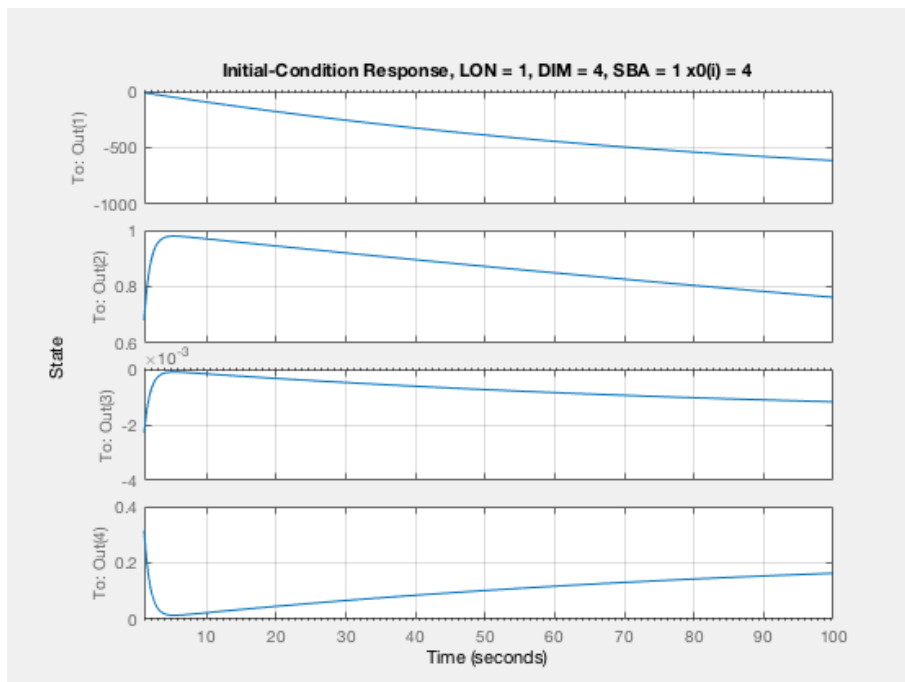


Figure 85: Longitudinal,  $x(4)$

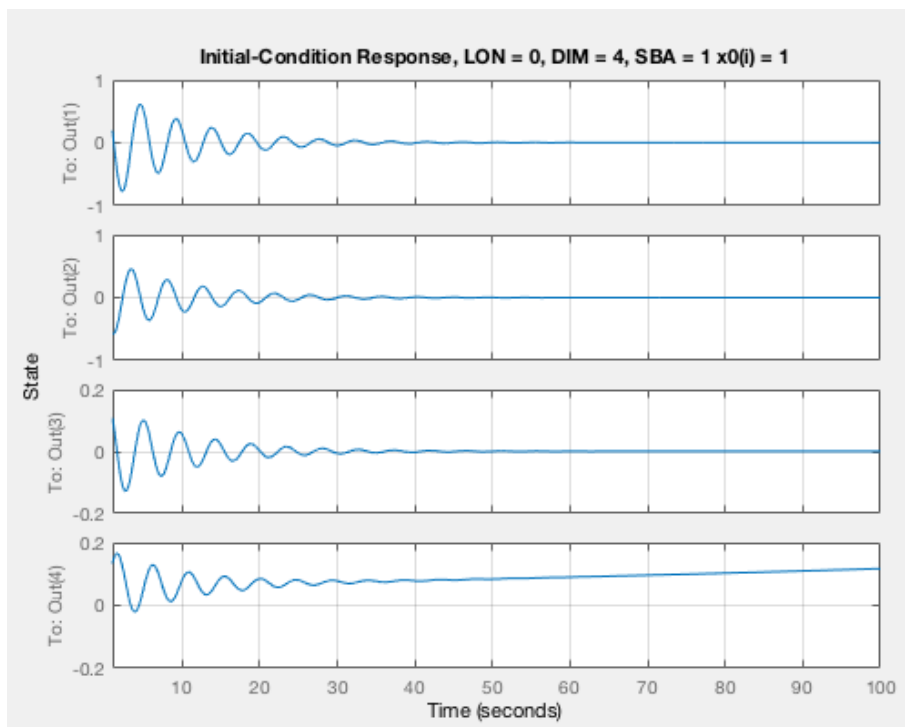


Figure 86: Lateral,  $x(1)$  (Altered Coefficients)

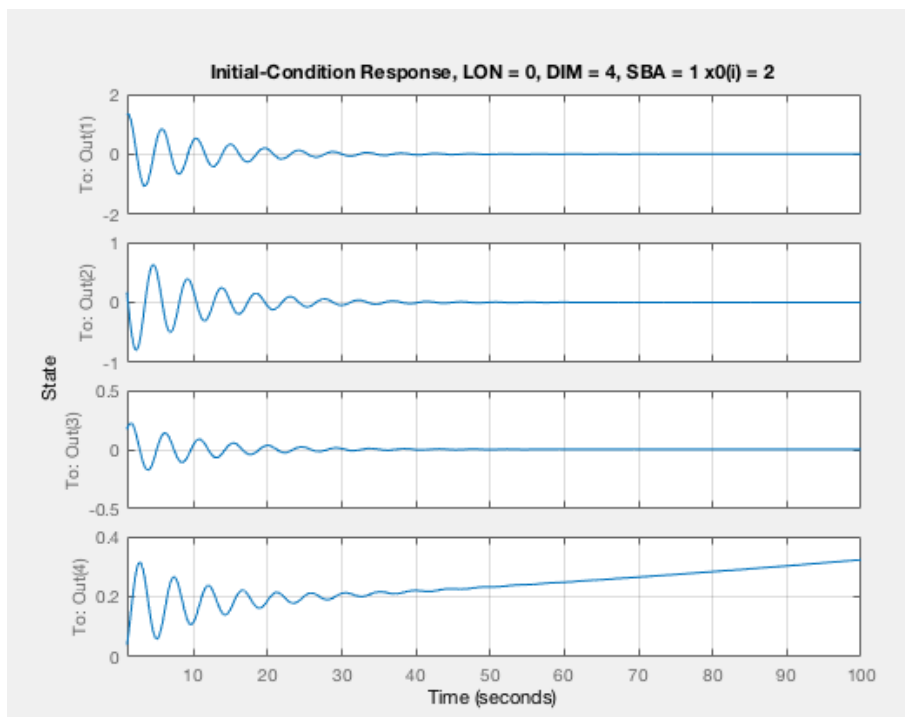


Figure 87: Lateral,  $x(2)$  (Altered Coefficients)

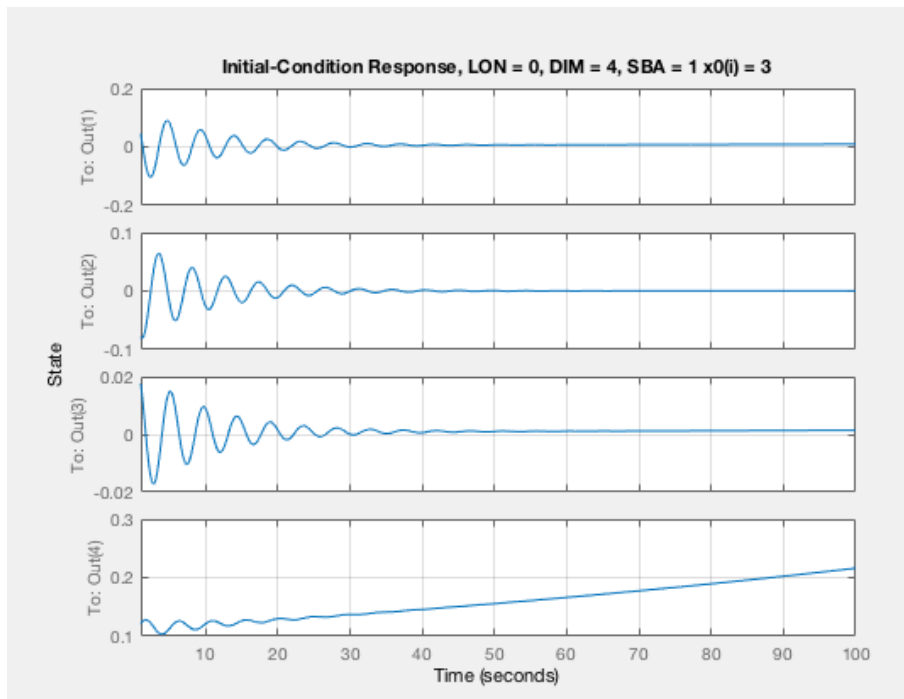


Figure 88: Lateral,  $x(3)$  (Altered Coefficients)

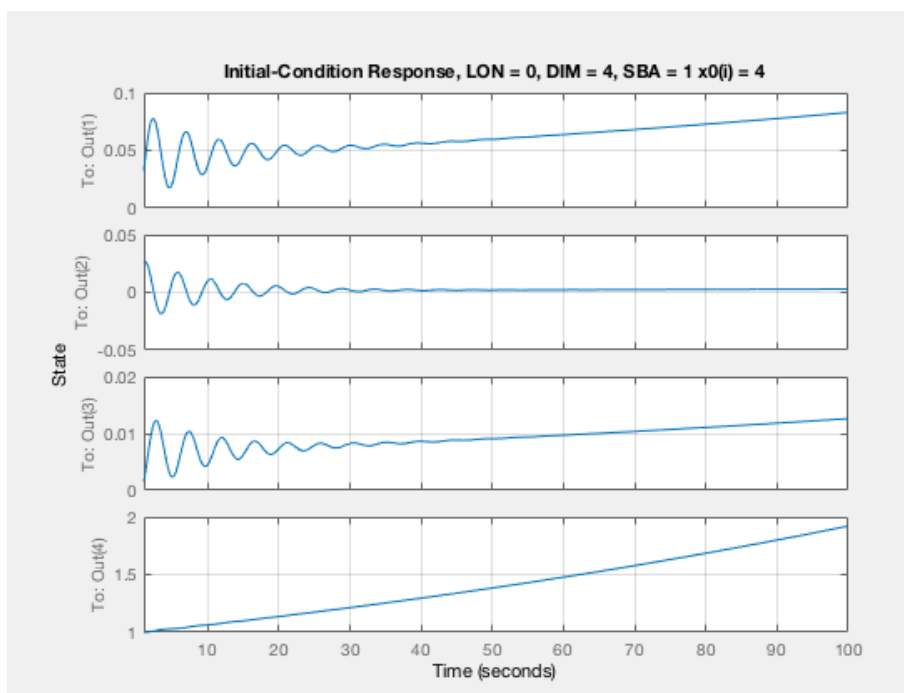


Figure 89: Lateral,  $x(4)$  (Altered Coefficients)



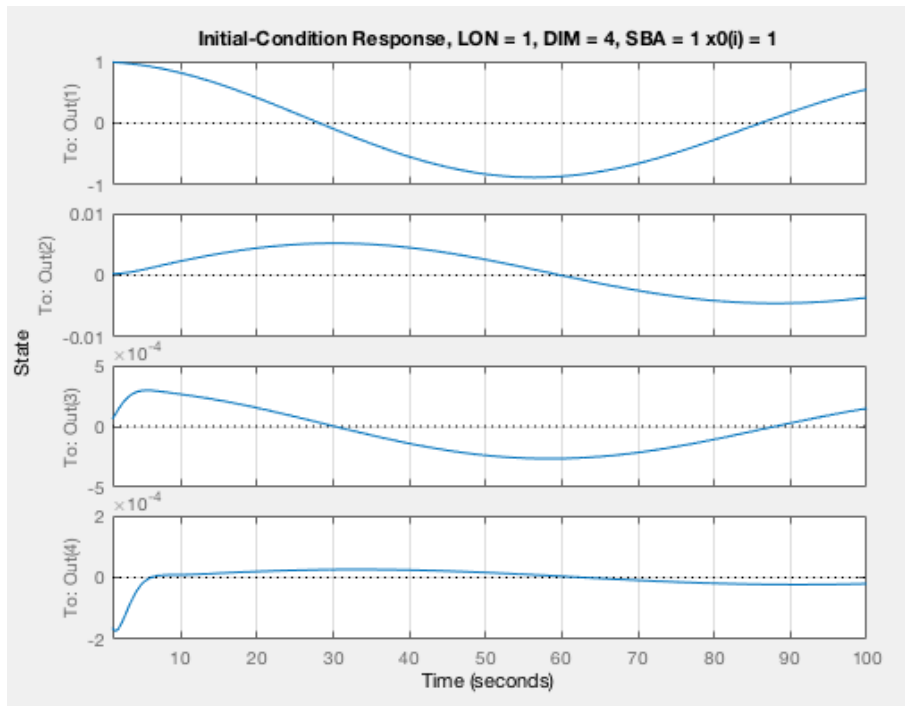


Figure 90: Longitudinal,  $x(1)$  (Altered Coefficients)

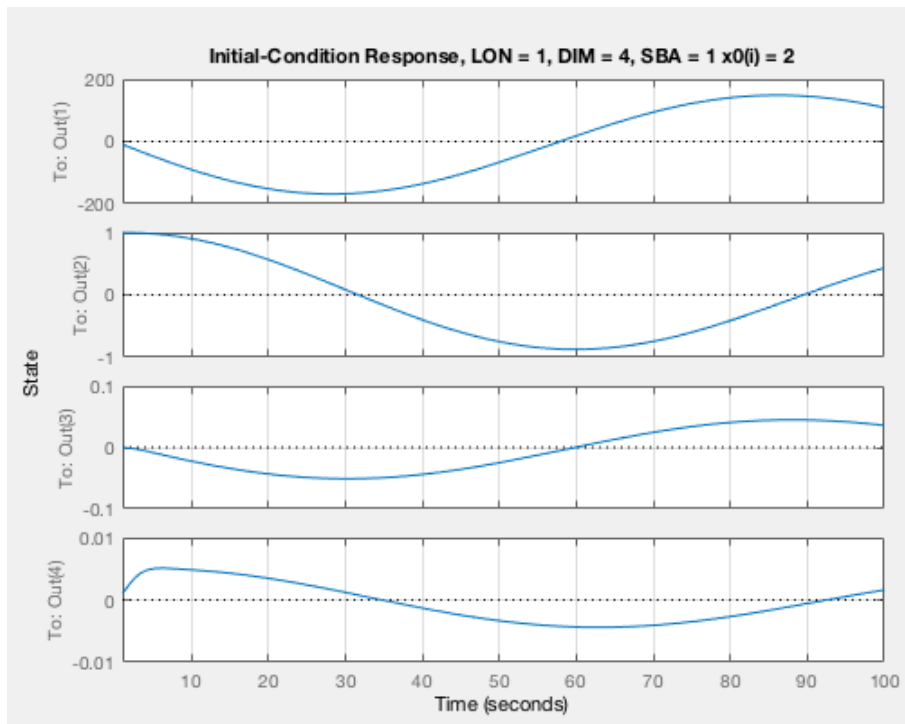


Figure 91: Longitudinal,  $x(2)$  (Altered Coefficients)

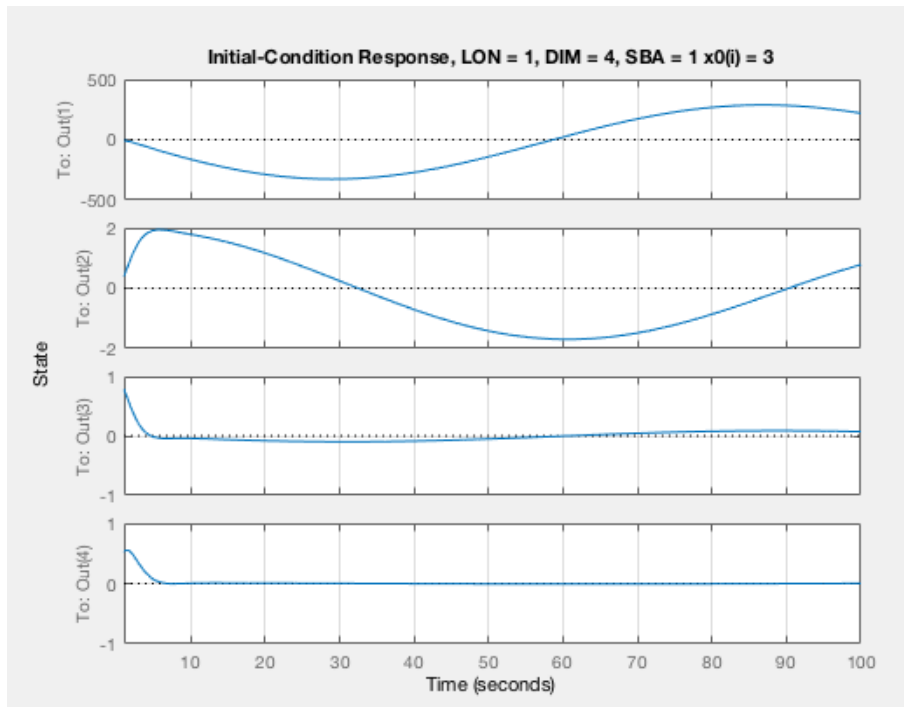


Figure 92: Longitudinal,  $x(3)$  (Altered Coefficients)

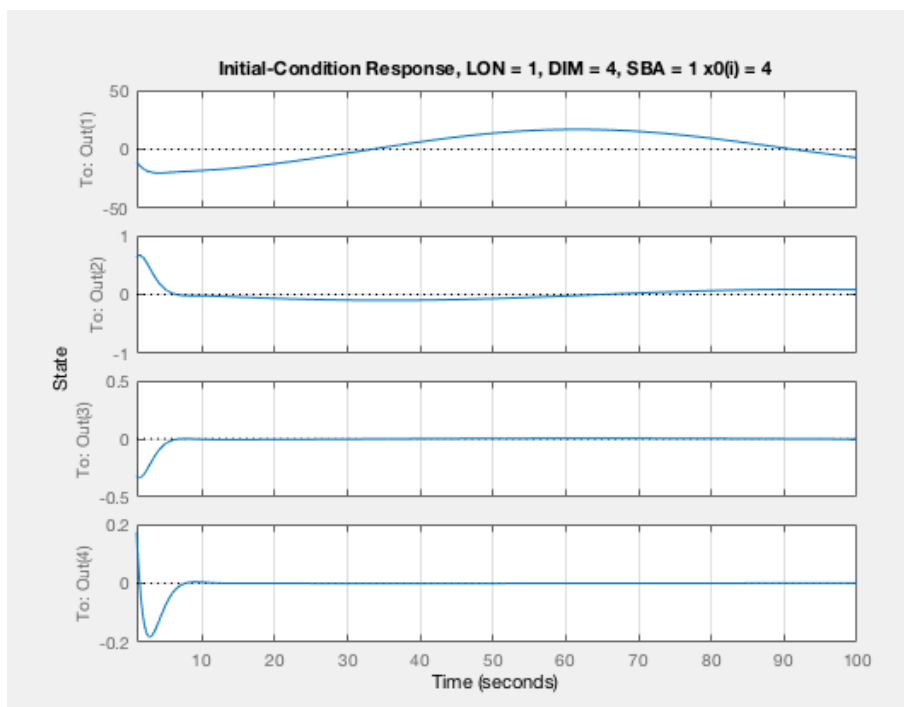


Figure 93: Longitudinal,  $x(4)$  (Altered Coefficients)