Design of an Experimental Jet to Evaluate Alternative Control Methods to Maintain Yaw Authority on Tailless Aircraft and Reduce Radar Cross Section

A report presented to
Department of Aerospace Engineering
San José State University

In fulfillment of the Graduate Program requirements for
Master of Science in Aerospace Engineering

by

Tripp Selvig

May 2024

approved by

Dr. Gonzalo Mendoza
Faculty Advisor

ABSTRACT

Initial Aircraft Design and Comparative Tailless Analysis

Tripp Selvig

The purpose of this report is to study the effects that alternative control surface can have on an aircraft in an attempt to remove vertical geometry, namely vertical stabilizers and rudders, during operation from an existing platform. Additionally, a study will be conducted using the MATLAB plugin P.O. Facets to further evaluate the benefits regarding the reduction of the aircraft's radar cross section compared against the original variant as well as any other additional benefits such as overall drag reduction.

For AE 295A, the first section of this report, Chapters 1-11, serves to detail the steps involved in creating the base aircraft and the baseline viability of the shape regarding weight and balance, and stability analysis. For AE 295B, Chapters 12-14 establish the model for the tailless model of the aircraft, describe and create the model and MATLAB analysis of the stability of each aircraft and provide a comparison of the results, and finally evaluate the radar cross section of each aircraft to compare what the change in geometry has to the overall radar visibility.

ACKNOWLEDGMENTS

I would like to thank many teachers within the Aerospace Engineering Department at San José State University, with a particular emphasis on Professor Gonzalo Mendoza for taking me under his wing as this projects Graduate Advisor, Professor Sean Montgomery for providing a great deal of assistance with baseline design calculations and decisions, and Professor Long Lu for his assistance in understanding stability analysis and helping foster my immense interest in the heart of this project, aircraft stability and control systems. Lastly, I would like to thank Kevin Li, a student in the Mechanical Engineering Department at San Jose State University creating the CAD model of the drag rudder and spoilers displayed in Section 9.

Table of Contents

List of Figures

List of Tables

Symbols

Symbol	Definition	Units (SI)
C_D	Drag Coefficient	---------
C_L	Lift Coefficient	---------
b	Wingspan	ft (m)
D	Drag Force	lb_f (N)
L	Lift Force	lb_f (N)
S	Wing Area	ft^2 (m^2)
V	Velocity	ft/s (m/s)
W	Weight	lbs (N)
$\bar{x}$	Distance from wing apex to the leading edge of the mean aerodynamic chord (MAC)	ft
Greek Symbols		
α	Angle of attack	deg or rad
ρ	Air density	slugs/ft^3 (kg/m^3)
Subscripts		
$( )_D$	Drag	---------
$( )_e$	Elevator area	---------
$( )_f$	Force	---------
$( )_h$	Horizontal Stabilizer area	---------
$( )_L$	Lift	---------
$( )_{NP}$	Neutral Point	ft
$( )_r$	Rudder area	---------
$( )_v$	Vertical Stabilizer area	---------
Acronyms		
AoA	Angle of Attack	---------
AR	Aspect Ratio (b^2/S)	---------
MOI	Moment of Inertia	ft*lbs
RCS	Radar Cross Section	---------

1. Mission Specification

1.1 Motivation

The aerospace industry has an aggressive history of innovation through the motivation of warfare and political conflict which has led to amazing feats of engineering in progressively shorter amounts of time. From the advent of the airplane in 1902, to the space race between nations which resulted in humans landing on the Moon in 1969, we have seen an impressive evolution of the industry and technology driving it in a comparably impressive amount of time. As a result of this growth and development, militaries around the world have begun relying on new developments in aviation technology in an attempt to outpace the technology of today. One such area of interest is the advent of tailless aircraft and the alternative control surfaces and systems they employ to safely accomplish tailless flight. Tailless aircraft are the current focus for next generation fighters, intelligence, strike, and reconnaissance (ISR) aircraft, and bombers as they bring one massive advantage, a largely reduced radar cross section (RCS), which is largely accomplished by removing the vertical geometry of traditional tails. As a result, the focus of this project will be designing a testbed aircraft involving additional multiple alternative control methods and a traditional vertical stabilizer as a foundation, followed by testing with the removal of the tail to evaluate yaw stability and control authority and compare the RCS of each aircraft.

1.2 Literature Review

There have been many attempts at flying wings and alternative control surfaces since the first jet fighters were introduced at the end of World War II ranging from the Horten 229 to the modern Northrop Grumman B-2 Spirit. Aerospace engineering has come a long way to ensure the latter is able to fly for 30 hours with mid-air refuels [1], and the first part of the literature reviewed for this project dives into the history of alternative control testing, followed by an examination of potential methods moving forward. Lastly the evaluation of RCS improvements or alternatives will be included in the final section of this review.

NASA has had a history of testing the limits of flight through its X-Plane program, and alternative control surfaces and programs are no exception. Starting in April of 1987, NASA began testing modified wing geometry on an active Navy aircraft, the F-18, to see the potential benefits of thrust vectoring and strakes when entering traditionally unsafe high alpha maneuvers and flight. Thus, the F-18 High Alpha Research Vehicle (HARV) project was born. The project was split into three phases to establish the purpose of testing, implement thrust vectoring, and finally employ hinged strakes. Stage one was successfully conducted in 1987 at an angle of attack of 55 degrees. This test showed the effects of airflow over the lifting surfaces of the unmodified aircraft and served as a baseline for further testing. The flow is visualized with the help of a smoke generator and yarn tufts attached to the skin of the aircraft in Figure 1.1 below [2]. Additional air pressure sensors were placed over the fuselage and wings at various locations to evaluate the difference more accurately in pressure and lift at high angles of attack.

Figure 1.1 - F-18 HARV flow visualization during phase one testing. [2]

Phase two testing began in 1991 and employed thrust vectoring nozzles over the exhaust to aid in controllability in high alpha flight envelopes. Importantly, the thrust vectoring nozzles were able to add controllability in high alpha maneuvers where traditional control surfaces began to fail, specifically the control surfaces on the tail (rudders and stabilators). After 193 test flights in phase two, including successfully controllable testing up to 70 degrees angle of attack, hinged strakes were added to the sides of the aircraft's nose to improve yaw authority in the further high alpha flight envelope testing of phase three. The strakes in phase three proved to be successful control surfaces above 35 degrees angle of attack, further adding to new data in high alpha flight envelope maneuvering. The F-18 HARV program ended in September 1996 after 385 successful flights [2].

Running parallel to the F-18 HARV project was the F-15 Advanced Control Technology for Integrated Vehicles (ACTIVE). The F-15 ACTIVE was another NASA X-Plane program based off an existing platform, the F-15B, that was modified with canards and thrust vectoring exhaust nozzles capable of deflecting up to 20 degrees in a 360-degree arc [3]. While the HARV project focused on flight in previously unsafe high alpha envelopes, the ACTIVE program put a larger focus on thrust vectoring and overall maneuverability for the first phase of testing between 1993 and 1999 until the introduction of testing the Intelligent Flight Control System (IFCS) programs from 1999-2008.

The IFCS program is of particular interest to this project as it studied the extreme limits of stability and control using closed loop computer systems [3] that managed the thrust vectoring nozzles to achieve flight in previously unsafe envelopes. To accomplish this, the IFCS integrated a dual-channel computer for each nozzle, set up in a closed loop system to manage the input, output, and fault detection between the pilot, flight control computer, and nozzle receiver.

Following the successful testing in the early phases of both the F-18 HARV and F-15 ACTIVE tests, NASA moved towards the design of a completely tailless fighter project. The X-36 Tailless Fighter Agility Research Aircraft was designed at 28% scale as a remotely operated testbed for evaluating the potential maneuverability and survivability of tailless aircraft design for future aircraft, particularly fighter aircraft. The X-36 was built by Boeing's Phantom Works facility and featured canards, split aileron flaps, and a thrust vectoring exhaust nozzle in place of

a traditional tail [4]. Without a tail, the X-36 was inevitably going to be a statically unstable aircraft, due to the coupling of roll and yaw forces and lack of rudder. To account for this, the X-36 also employed a single-channel digital fly-by-wire control system with active control systems to attempt to stabilize the aircraft, which proved successful in the subsequent flight testing. Over the six months of testing between May 17, 1997, and November 12, 1997, the X-36 completed 31 research flights that ranged from testing maximum angle of attack during flight, successful to 40 degrees, cruise at an altitude of 20,200 feet, and stable flight at a maximum speed of 234 miles per hour [4].

The above aircraft have all proven potential alternative control surfaces and methods to add control authority in multiple flight axes or nontraditional envelopes, and today research continues to push the envelope even further. In one example, three students from Tufts University in Medford Massachusetts analyzed four different lift distributions to evaluate the agility, which they define as the airfoils ability to normalize the induced yaw axis control derivative. The comparison starts with a recently published NASA article (2021) that evaluated a Prandtl 1933 airfoil to show the proverse nature regarding aileron input and positive yaw authority [5]. The group then compares the lift distribution of the Prandtl 1920, Prandtl 1933, Jones 1950, and Klein and Viswanathan 1975 to further study the proverse nature of these airfoils. Figure 1.2 below, provided by the journal [5], shows the effects of the lift distributions along and how in the case of the Klein and Viswanathan 1975, it can be increased past the root value along the spanwise axis between the root and approximately 0.4 span.

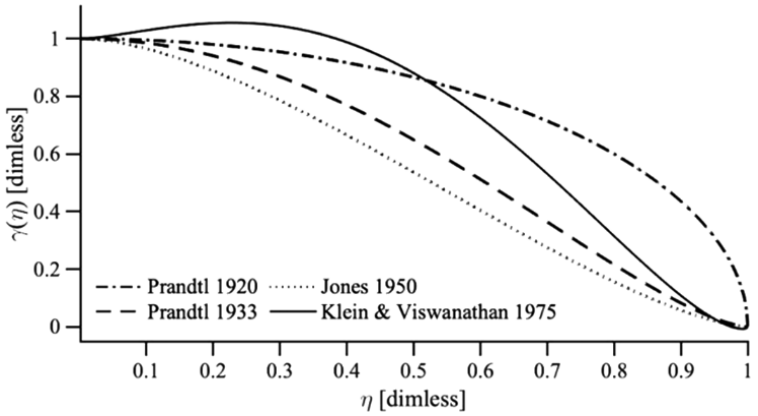


Figure 1.2 – “Lift distributions along the spanwise axis normalized to unity at the centerline” [5].

Further study of airfoils that exhibit this benefit could provide additional control and stability in operation without the need for additive control surfaces, potentially allowing for even further enhanced control when the new devices, such as split ailerons, are implemented.

Rather than designing a new wing to be used on an entirely new aircraft, researchers made up of a mixture of Ph. D students and faculty of Manchester University as well as an engineer from BAE Systems AIR have published a 2020 paper in the AIAA Journal proposing the addition of a few actuation control system to an existing platform to provide a new control system that would assist in reducing the induced yaw-forces exhibited when the aircraft rolls. The proposed systems were implemented on the University of Manchester’s MAGMA aircraft which was developed “as a low-risk demonstrator for non-conventional flight controls” [6]. The modelled aircraft operates with vertical stabilizing fins with no active control systems to add a layer of stability during unconventional testing. The team aimed to control the aircraft in all axes using conventional

conformal trailing edges in place of a traditional vertical stabilizer. To accomplish this, the team employed additional control systems to apply lateral drag asymmetrically along the lateral axis during roll to account for the coupled yaw moment. The control system was defined by entering control shapes which are "defined as the control surface deflections which yield a lift distribution corresponding to an aerodynamic mode shape" [6]. After testing, the team found the maximum yaw moment was constrained to the maximum deflection of the control surfaces, the spoilers (90 degrees) and the trailing edge control surfaces (10 degrees) [6]. These results show promise in continued application of asymmetrical drag control to replace traditional yaw-control in absence of a vertical control surface.

Building on their previous 2020 study, the same group from Manchester University continued modification of control surfaces on the same aircraft to further push controllability of the yaw-axis under induced yaw forces during a roll without a vertical stabilizer. By altering the trailing edge control surfaces on the wing and the spoilers located aft of the quarter chord line the team proposed that operation up to a 45-degree bank angle the new control surfaces and system can maintain a manageable sideslip angle between -2 degrees (proverse) and 5 degrees (adverse). With the new design the aircraft was reportedly able to maintain safe and level flight. For aerodynamic modeling of the control system, the team used a range of spanwise and chordwise load distributions for potential sideslip angles the aircraft may be exhibited to at zero angle of attack flight and compared it to the actual sideslip state of the aircraft in flight. By doing so, they were able to create a control input that interpolates an intermediate spanwise and chordwise load at each point on the wing for the computer to account for in control surface actuation [7]. Having this lower order assumption allowed for a more efficient control system to be implemented, saving on overall computing power and required electronics to accompany the system. This ensured minimal overall changes to the base aircraft when implementing the new control system. Both pitch and roll rate were evaluated at each section and accounted for by changing the local angle of attack of the appropriate area. This method of slightly altering an existing aircraft proves at a basic level that it may be possible to use similar aspects in the second iteration of the proposed design, thus warranting further study.

Rather than studying provers effects of airfoil choice or modification of current aircraft control surfaces, a duo from the Air Force Research Laboratory on Wright-Patterson Air Force Base in Ohio proposed the application of foldable wingtips to enhance yaw control on a studied tailless platform. The proposed design involves a baseline aircraft, the Supersonic Generic Air Vehicle Research Model (SUGAR), designed as a tailless supersonic testbed. The aircraft incorporates trailing edge control surfaces as well as control surfaces on the wingtips that include the ability to rotate the wingtip up to +/- 90 degrees from the base in line with the wing [8]. The trailing edge control surfaces are categorized as inboard and outboard, with the inboard designed to apply a greater drag force while applying a smaller roll moment. All control devices are designed as split flaps to allow airflow manipulation on both the high- and low-pressure surfaces of the wing, a successful technique that is seen in practice on the Northrop Grumman B-2 Spirit. The aircraft performed three testing configurations. The first involved keeping the wingtips inline using only split flap deflection (zero wingtip deflection), the second involved both wingtips deflected to +90 degrees from the wing and split flap deflection during flight, and the final test replicated the second with the wingtips being deflected to -90 degrees instead. While this testing is an interesting approach, it adds vertical geometry to the aircraft which is being avoided in this project.

The sizing of the initial aircraft is crucial to ensure the tailless variant will maintain a similar level of controllability and stability during evaluation and to accomplish this the textbook *Aircraft Design: A Conceptual Approach* by Danial P. Raymer is used in the early stages of design. The book outlines design concepts and follows up with real-life examples and the equations used to create an aircraft preliminary design using a mixture of mission design requirements and similar aircraft to create a base aircraft to build upon and alter. Following the generic size requirements, in this case for what Raymer states as a "Military Jet Trainer" allows for a reasonably accurate first pass at design without committing unnecessary time. An example is provided below in Fig 1.3 Raymer's empty weight fraction table [9].

Table 1.1 – Image of Raymer's empty weight fraction vs max weight table [9].

Table 3.1 Empty Weight Fraction vs W_0

$W_e/W_0 = AW_0^C K_{vs}$	A	{A-metric}	C
Sailplane—unpowered	0.86	{0.83}	−0.05
Sailplane—powered	0.91	{0.88}	−0.05
Homebuilt—metal/wood	1.19	{1.11}	−0.09
Homebuilt—composite	1.15	{1.07}	−0.09
General aviation—single engine	2.36	{2.05}	−0.18
General aviation—twin engine	1.51	{1.4}	−0.10
Agricultural aircraft	0.74	{0.72}	−0.03
Twin turboprop	0.96	{0.92}	−0.05
Flying boat	1.09	{1.05}	−0.05
Jet trainer	1.59	{1.47}	−0.10
Jet fighter	2.34	{2.11}	−0.13
Military cargo/bomber	0.93	{0.88}	−0.07
Jet transport	1.02	{0.97}	−0.06
UAV—Tac Recce & UCAV	1.67	{1.53}	−0.16
UAV—high altitude	2.75	{2.48}	−0.18
UAV—small	0.97	{0.86}	−0.06

K_{vs} = variable sweep constant = 1.04 if variable sweep = 1.00 if fixed sweep

Like the Raymer book, *Airplane Design* by Jan Roskam is used for preliminary design and sizing. Roskam uses similar methods with data from real world aircraft and serves as a guide to create preliminary sizing estimates for the aircraft desired without needing to create and alter physical or CAD geometry models, saving time in the early phases and giving a strong foundation to design the geometry going forward. Using this alongside the Raymer design book serves as a relative check to confirm the preliminary values of the proposed aircraft are realistic to move forward with. Figure 1.4 below shows Roskam's weight fraction table to demonstrate the comparison with Raymer's in Figure 1.3 [10].

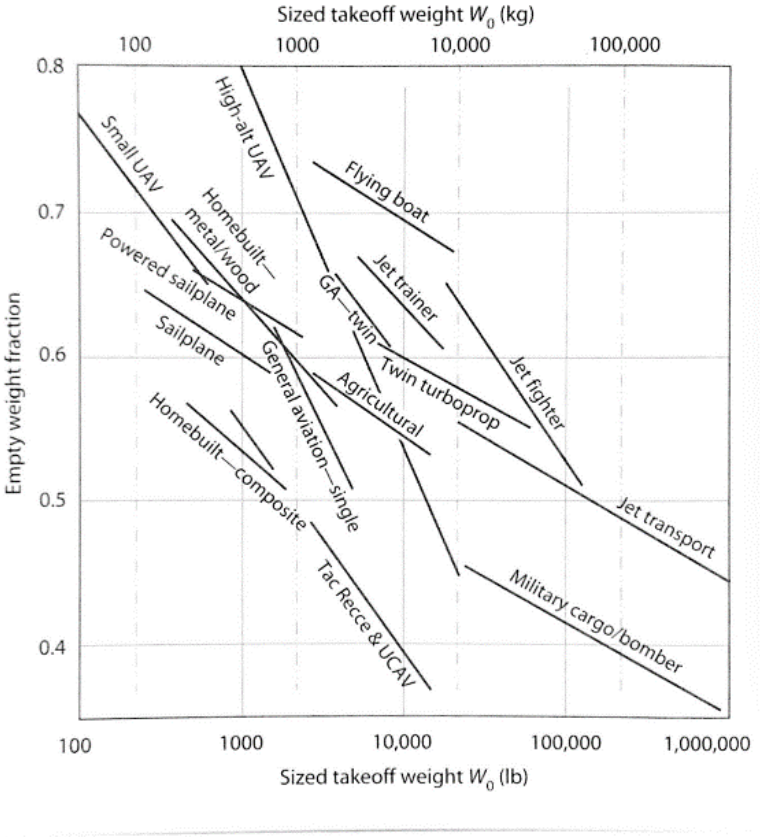


Fig. 3.1 Empty-weight fraction trends.

Figure 1.4 – Roskam's empty weight fraction [10].

Given the nature of the proposed aircraft, advanced control systems will likely have to be applied to ensure stable flight characteristics of the tailless model. Roskam's *Airplane Flight Dynamics and Automatic Flight Controls* provides an abundance of information regarding sizing of aircraft and potential flight control applications for a desired mission or configuration. The text goes into detail about interactions of stability and control derivatives and modern state-space systems, including how to derive them analytically and apply the state-space controls in both traditional and modern control theory [11]. This book will prove critical in checking computational values and their validity.

Reduction of RCS is one of the fundamental goals of this project, and the understanding of what are effective methods and why they work is important to making design decisions and compromises to accomplish this goal. Stadmore noted in a 1979 AIAA journal entry that reduction of RCS would be crucial going forward [12], and this assumption was correct. Since the advent of the F-117 Nighthawk to the B-2 Spirit, engineers have attempted to lower RCS to the size of small animals such as birds to allow slower, high payload aircraft to operate safely while entering and leaving enemy combatant ranges. The paper mentions that the two primary methods in evading radar are electronic countermeasures (ECM) and radar cross section reduction. ECM effectively returns aggressive radiative waves to the radar source when the aircraft sensors detect incoming radar waves, scrambling the return signal. The issue with this method is that the radar station is

then aware of an issue giving them an opportunity to react. Reduction of RCS on the other hand allows the enemy to see the aircraft, but at a size that looks as small as insects or birds. Through this method the enemy radar station has to decide whether to react to a radar detection signal that may very well be completely benign, effectively hiding in plain sight. While there are newer methods today such as certain paints and materials that absorb radar, the paper goes into detail about changing geometry and how it can analytically reduce RCS [12].

1.3 Project Proposal

With the above research in mind, this project is designed as a proof-of-concept evaluation for potential alternative control surfaces and methods to effectively remove the need for a vertical stabilizer on ISR/fighter sized aircraft in hopes of lowering the radar cross section. The design concept/geometry will be like the style of ISR aircraft such as the Textron Scorpion [1 3], the Grumman A-5 Intruder [1 4], and the Martin B-57 Canberra [1 5], all pictured below this section. To accomplish this, the project will be split into two parts. The first part of the project is aimed at designing a base aircraft with a vertical stabilizer and three different alternative control surfaces being drag rudders, thrust vectoring nozzles, and potentially hidden internal rudders that utilize ram air to act as additive thrust vectoring to the sides of the fuselage in the event the prior two options prove insufficient. The second part of the project will be studying the removal of the vertical stabilizer and how it affects the RCS and S&C analysis compared to the traditional design.

Figure 1.6 – Textron Scorpion ISR jet [13]

Figure 1.6 – Grumman A-6 Intruder [14]

Figure 1.5 – Martin B-57 Canberra from the 556th Recon Squadron [15]

1.4 Methodology

Testing on the aircraft will be simulated using Vortex Lattice Method analysis through open-source programs (such as XFLR5) to prove the effectiveness of the added surfaces and systems in additive yaw authority and comparing the forces in the yaw axis to the traditional tail and rudder force. Once the aircraft has proven ample yaw authority without use of the rudder, a second iteration will be created by removing the vertical stabilizer. The comparison of stability and control analysis and radar cross section analysis between the two versions is the primary focus of the project. A meaningful reduction in RCS while maintaining comparable stability and control analysis, within a reasonable margin, through the addition of the proposed control surfaces and devices on the second iteration will be considered a successful project.

2. Preliminary Sizing Estimates

2.1 Mission Description

The proposed aircraft is designed as a testing aircraft, like the NASA X-Plane projects that have occurred since the 1950's. The aircraft serves as a proof of concept for the implantation of alternative control methods to an existing airframe, the initial aircraft, to prove effective stability and control when removing the vertical stabilizer, and to further prove that removing the vertical geometry reduces the RCS by useful levels. As such the aircraft will be designed to takeoff, climb to a cruise altitude of approximately 30,000 feet, perform control testing and for most of the flight duration, and cruise back to the runway before beginning final approach and landing. Both aircraft models will be evaluated using the POFacets MATLAB plugin to determine total RCS at different angles. As this is a proof of concept, the aircraft will not be physically produced for this paper, however defining the mission profile, pictured below in Figure 2.1, helps to appropriately size the aircraft during the design phase.

2.2 Mission Profile

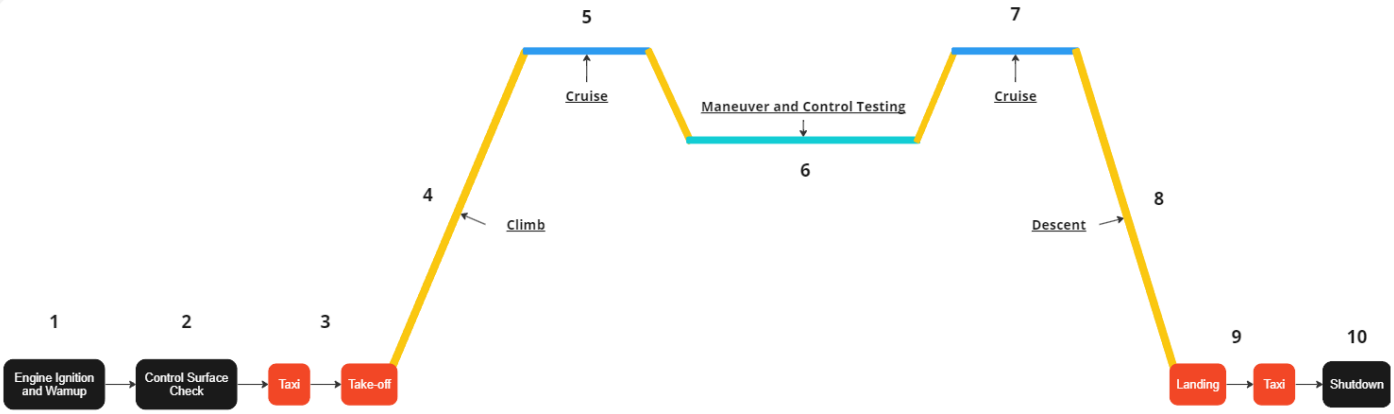


Figure 2.1 – Mission profile diagram

2.3 Preliminary Mission Sizing

Table 2.1 - Mission requirements

Mission Requirement	Value (SI)
Payload	1 Pilot (180 lbs)
Crew Requirements	1 Pilot, Strike Cockpit (180 lbs)
Maximum Takeoff Weight	14,500 lbs
Range	1000 mi
Cruise Velocity	350 kts
Maximum Velocity	600 kts
Clean Stall Velocity	120 kts
Landing Stall Velocity	110 kts
Service Ceiling	50,000 ft
Maneuvering Requirements	+5g/-3g, Tailless Yaw-Stability
Safety Certification Requirements	Military

Climb Rate	7,000 ft/min
Takeoff Distance	2,500 ft
Landing Distance	3,500 ft

2.4 Preliminary Mission Sizing Estimates

Using the Raymer method [9] and MATLAB to efficiently run calculations, the preliminary sizing for the aircraft requiring a maximum takeoff weight of 7500 lbs is provided in Table 2.2 below.

Table 2.2 – Preliminary mission sizing estimates

Maximum Takeoff Weight (W_0)	14,500 lbs
Wing Loading	38.067 lbs/ft^2
$C_{L\ Max}$ (NACA M6 Airfoil Re = 500,000)	1.2577
$C_{L\ Max}$ (Raymer Method)	0.9433
Wing Sweep	35 deg
Wing Area (Raymer Method)	150 ft^2

2.5 Preliminary Drag Polar Evaluation

Drag polar estimates are done with initial value estimates using equations from the Raymer [9] and Roskam [10] books and produced in MATLAB.

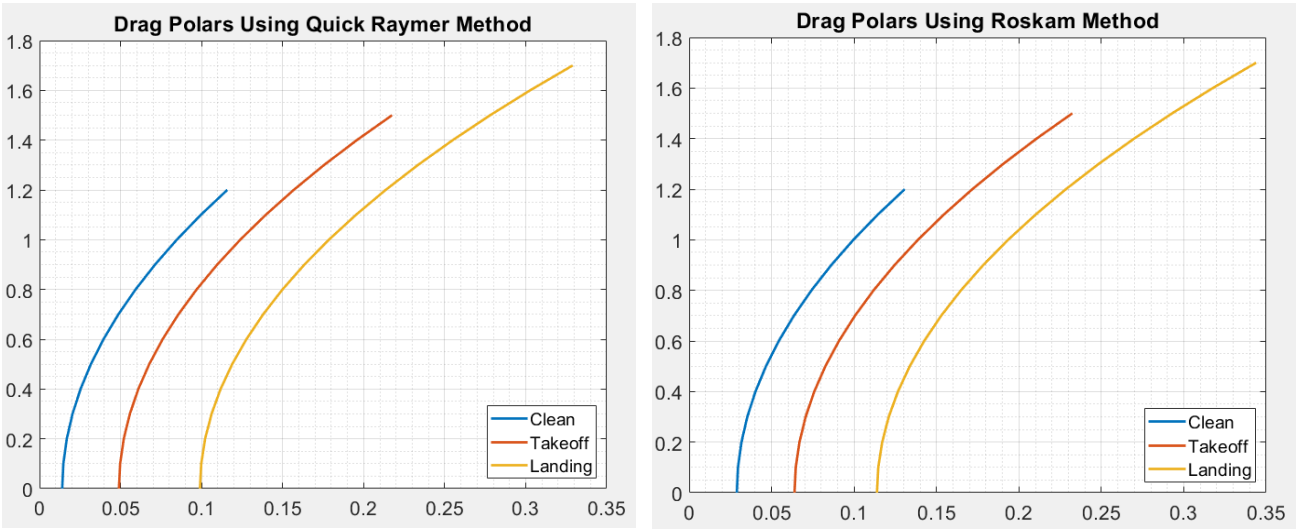


Figure 2.2 – Preliminary drag polar estimates using Raymer and Roskam methods.

Table 2.3 – Raymer and Roskam estimate values

	CD0 Raymer	CD0 Roskam	AR	e	K	L/D max Raymer	L/D max Roskam
Takeoff	.049	.0639	5	.85	.0749	8.3	7.2
Landing	.0990	.1139	5	.8	.0796	5.6	5.3
Clean (subsonic)	.0140	.0289	5	.9	.0707	15.8	11.0

Table 2.4 – MATLAB induced drag results

Aspect Ratio (unitless)	5
Oswald Efficiency Clean(e)	0.90
Oswald Efficiency Takeoff (e)	0.85
Oswald Efficiency Landing (e)	0.80
Induced Drag Coefficient Clean (k)	0.0707
Induced Drag Coefficient Takeoff (k)	0.0749
Induced Drag Coefficient Landing (k)	0.0796

3. Propulsion

3.1 Propulsion Requirements

Proper sizing for propulsion in aircraft is vital during the development phase. Implementing a powerplant that produces more power than necessary for the design may lower efficiency or unnecessarily increase weight, and insufficient power may restrict performance or prevent operation all together. Starting off with an evaluation of a similarly sized aircraft regarding weight and initial wing area, the Northrop T-38 Talon trainer jet, gives a strong foundation for the base propulsion choice. The early variants of the T-38 used two General Electric J85-5A turbojet engines capable of 2,050 lb_f each, with a maximum thrust value of 2,900 lb_f with the use of the engine's afterburner. Current variants use an upgraded J85 engine that is enhanced to the J85-GE-21 which is capable of outputting 3,600 lb_f and 5,000 lb_f with the use of afterburner [16]. Since the required design intends to utilize thrust vectoring as a primary method of adding yaw authority, the higher thrust J85-5R variant proves to be more effective in application.

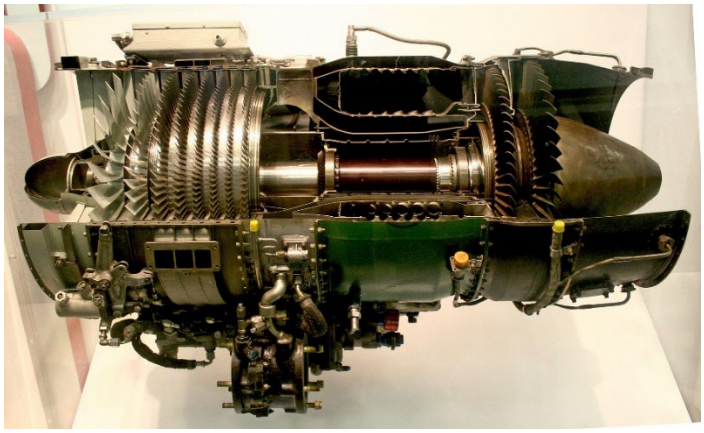


Figure 3.1 – General Electric J85-GE-17A [16]

3.2 Engine Specifications

The General Electric J85-GE-21 is a dual stage turbine with annular combustors and a single-spool 9 stage compressor with the following sizing and performance characteristics displayed below in Table 3.1.

Table 3.1 – J85-GE-21 sizing and performance characteristics

Characteristics	**Specification**
Length	112.5 in
Diameter	20.8 in
Dry Weight	684 lbs
Maximum Thrust	3,600 lb_f dry / 5,000 lb_f afterburner
Pressure Ratio	8.3:1
Air Mass Flow	53 lb/s
Turbine Inlet Temperature	1,790 F
Specific Fuel Consumption	1.24 lb/(lb_f*h) dry / 2.13 lb/(lb_f*h) afterburner
Thrust-to-Weight Ratio	5.25 dry / 7.3 afterburner

3.3 Differential Thrust and Thrust Vectoring

While a single more powerful engine would be sufficient to properly power the plane to the desired flight speeds and simplify operation, employing dual engines comes with the benefit of being able to employ differential thrust. Differential thrust is the use of varying the thrust output of each engine independently during flight. Since the engines are mounted equidistant from each other off the longitudinal axis, by modulating the thrust of the engines differently a yaw force can be induced on the aircraft. The chosen engines are also capable of producing substantially more thrust with afterburner, further amplifying the potential effect of any yaw forces.

Additionally, the aircraft is designed with the intention of applying 3-ring thrust vectoring nozzles on each engine to allow for additional yaw authority during operation without a rudder. In a now declassified Defense Technical Information Center notice titled "Thrust Vectoring Nozzle for Modern Military Aircraft", Daniel Ikaza illustrates the effectiveness of thrust vectoring, and a multitude of different nozzle designs to allow implementation on modern day aircraft. Even with some of the simplest designs, such as hinged reaction struts, the nozzles can facilitate 20 degrees of thrust vectoring, or deflection, in one axis during modelled operation [17]. With a three-ring system, pictured below in figure 3.2 a layout utilizing four actuators, thrust can be deflected in both the pitch and yaw axis, and any combination of the two, with some designs allowing up to 35 degrees of deflection.

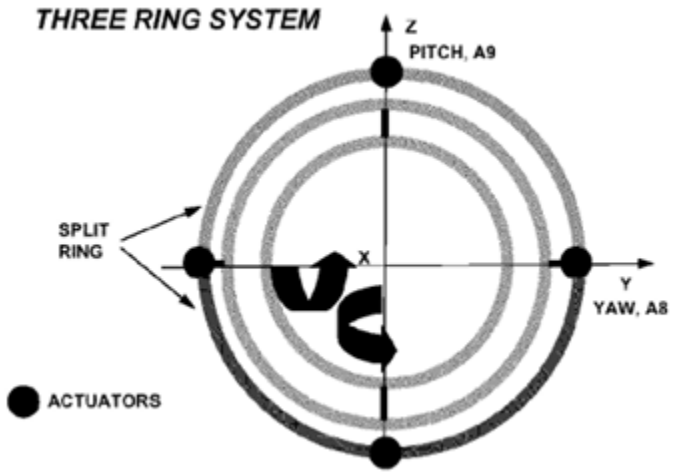


Figure 3.2 – Three-ring thrust vectoring nozzle diagram [17]

For operational consistency, the proposed aircraft is currently limited to a more realistic 20 degrees deflection and the results of a combined thrust vectoring and differential thrust in the positive yaw axis are below in Tables 3.2 and 3.3. The evaluation shows a combination of single engine vectoring with and without afterburner, as well as dual engine vectoring in the same configuration. Importantly, the evaluation is done using thrust force (lb_f) alone, not considering the full moment force for raw clarity, as moment can be altered for more favorable results during fuselage design. The MATLAB code used to quickly derive the results in the following tables is shown in Appendix A.

Table 3.2 – Dual engine differential thrust and single engine thrust vectoring

	Standard Thrust, Right Vector		Standard Thrust, Right Vector (AB)		Left Afterburner Right Standard Vectored	
Deflection (deg)	Forward Thrust (lbf)	Yaw Thrust (lbf)	Forward Thrust (lbf)	Yaw Thrust (lbf)	Forward Thrust (lbf)	Yaw Thrust (lbf)
0	7200	0	8600	0	8600	0
1	7199.45	62.83	8599.24	87.26	8599.45	62.83
2	7197.81	125.64	8596.95	174.50	8597.81	125.64
3	7195.07	188.41	8593.15	261.68	8595.07	188.41
4	7191.23	251.12	8587.82	348.78	8591.23	251.12
5	7186.30	313.76	8580.97	435.78	8586.30	313.76
6	7180.28	376.30	8572.61	522.64	8580.28	376.30
7	7173.17	438.73	8562.73	609.35	8573.17	438.73
8	7164.97	501.02	8551.34	695.87	8564.97	501.02
9	7155.68	563.16	8538.44	782.17	8555.68	563.16
10	7145.31	625.13	8524.04	868.24	8545.31	625.13
11	7133.86	686.91	8508.14	954.04	8533.86	686.91
12	7121.33	748.48	8490.74	1039.56	8521.33	748.48
13	7107.73	809.82	8471.85	1124.76	8507.73	809.82
14	7093.06	870.92	8451.48	1209.61	8493.06	870.92
15	7077.33	931.75	8429.63	1294.10	8477.33	931.75
16	7060.54	992.29	8406.31	1378.19	8460.54	992.29
17	7042.70	1052.54	8381.52	1461.86	8442.70	1052.54
18	7023.80	1112.46	8355.28	1545.08	8423.80	1112.46
19	7003.87	1172.05	8327.59	1627.84	8403.87	1172.05
20	6982.89	1231.27	8298.46	1710.10	8382.89	1231.27

Table 3.3 – Dual engine thrust vectoring

	Both Afterburner, Right Vectored		Standard Thrust, Both Vectored		Both Afterburner Both Vectored	
Deflection (deg)	Forward Thrust (lbf)	Yaw Thrust (lbf)	Forward Thrust (lbf)	Yaw Thrust (lbf)	Forward Thrust (lbf)	Yaw Thrust (lbf)
0	10000	0	7200	0	10000	0
1	9999.24	87.26	7198.90	125.66	9998.48	174.52
2	9996.95	174.50	7195.61	251.28	9993.91	348.99
3	9993.15	261.68	7190.13	376.82	9986.30	523.36
4	9987.82	348.78	7182.46	502.25	9975.64	697.56
5	9980.97	435.78	7172.60	627.52	9961.95	871.56
6	9972.61	522.64	7160.56	752.60	9945.22	1045.28
7	9962.73	609.35	7146.33	877.46	9925.46	1218.69
8	9951.34	695.87	7129.93	1002.05	9902.68	1391.73
9	9938.44	782.17	7111.36	1126.33	9876.88	1564.34
10	9924.04	868.24	7090.62	1250.27	9848.08	1736.48
11	9908.14	954.04	7067.72	1373.82	9816.27	1908.09
12	9890.74	1039.56	7042.66	1496.96	9781.48	2079.12
13	9871.85	1124.76	7015.46	1619.65	9743.70	2249.51
14	9851.48	1209.61	6986.13	1741.84	9702.96	2419.22
15	9829.63	1294.10	6954.67	1863.50	9659.26	2588.19
16	9806.31	1378.19	6921.08	1984.59	9612.62	2756.37
17	9781.52	1461.86	6885.39	2105.08	9563.05	2923.72
18	9755.28	1545.08	6847.61	2224.92	9510.57	3090.17
19	9727.59	1627.84	6807.73	2344.09	9455.19	3255.68
20	9698.46	1710.10	6765.79	2462.55	9396.93	3420.20

As the results demonstrate, a considerable amount of force is capable of being produced by deflecting up to 20 degrees, proving effectiveness as a potential method of controlling the adverse effects of roll-yaw coupling during operation.

4. Airfoil Analysis

4.1 Airfoil Requirements

Airfoil selection is a crucial process when determining aircraft performance and wing design. Different airfoils provide different benefits that can more appropriately fit the mission requirements of the proposed aircraft and some aircraft have used multiple airfoils at different spans on the wing to optimize performance even further. As seen in Chapter 1 of this report, studies have shown that certain airfoils can provide benefits to one of the largest hurdles of this proposed design, roll-yaw coupling, by providing a proverse effect on coupling during roll. Similarly, a quick study into reflexed airfoils shows that by adjusting the trailing edge of the airfoil, the moment coefficient can be altered to provide a more desirable stability criteria for the wing alone.

4.2 Proposed Airfoils

Airfoil analysis is conducted with a combination of XFOIL and XFLR5 using airfoil data provided by airfoiltools.com. As a result, an analysis of NACA airfoils is conducted as the data is readily available. The NACA M-Series airfoils, particularly the M-6 and M-12, feature a reflexed chord line to provide additional stability when the vertical control surfaces are not employed. Below shows an XFLR5 analysis of the NACA M-6 Airfoil at 65% thickness, 85% thickness, standard, and the NACA M-12 Airfoil at full size. All evaluations are done with a 39.37 ft chord, the current estimated root size, and at Reynold's Number values ranging from 500,000 to 9,000,000 to properly cover expected operational range of the aircraft.

4.3 Individual Airfoil Analysis

The following analysis of each airfoil is done with the use of XFLR5 importing airfoil data from airfoiltools.com. Each airfoil has been refined to have 125 points for consistency and accuracy. Evaluation is completed at 500,000 Re, and then from 1,000,000 Re to 9,000,000 Re in intervals of 1,000,000 Re for each analysis.
NACA M-6 65% XFLR5 Analysis from -10 to +20 degrees angle of attack:

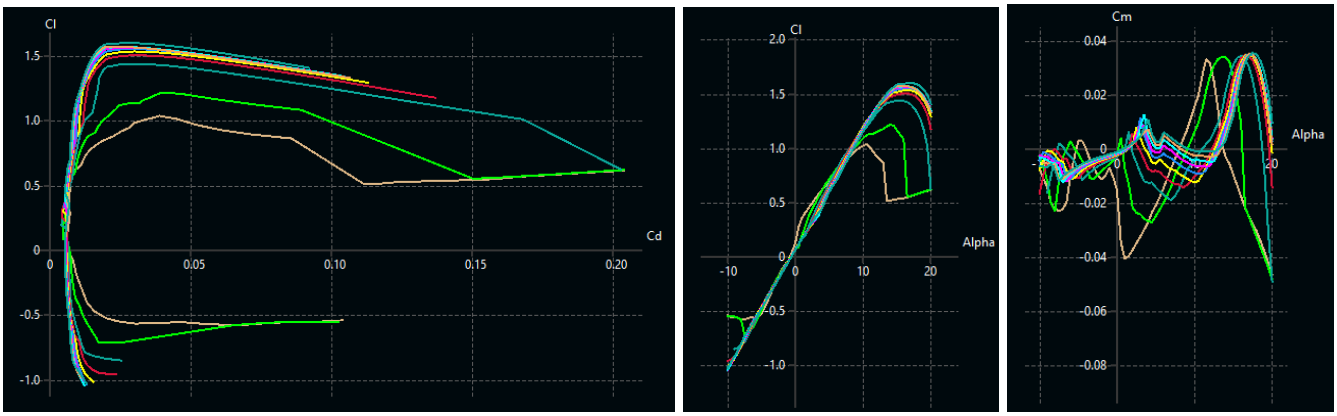


Figure 4.1 – M6 65% C_l/C_d Graph (left), C_l/α Graph (middle), and C_m/α graph (right)

NACA M-6 85% XFLR5 Analysis from -10 to +20 degrees angle of attack:

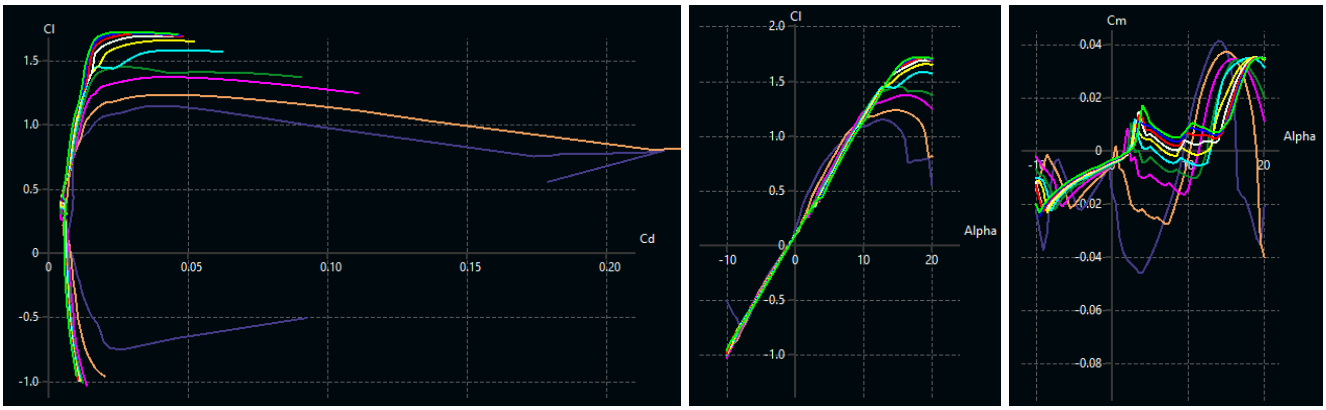


Figure 4.2 – M6 85% C_l/C_d Graph (left), C_l/α Graph (middle), and C_m/α graph (right)

NACA M-6 XFLR5 Analysis from -10 to +20 degrees angle of attack:

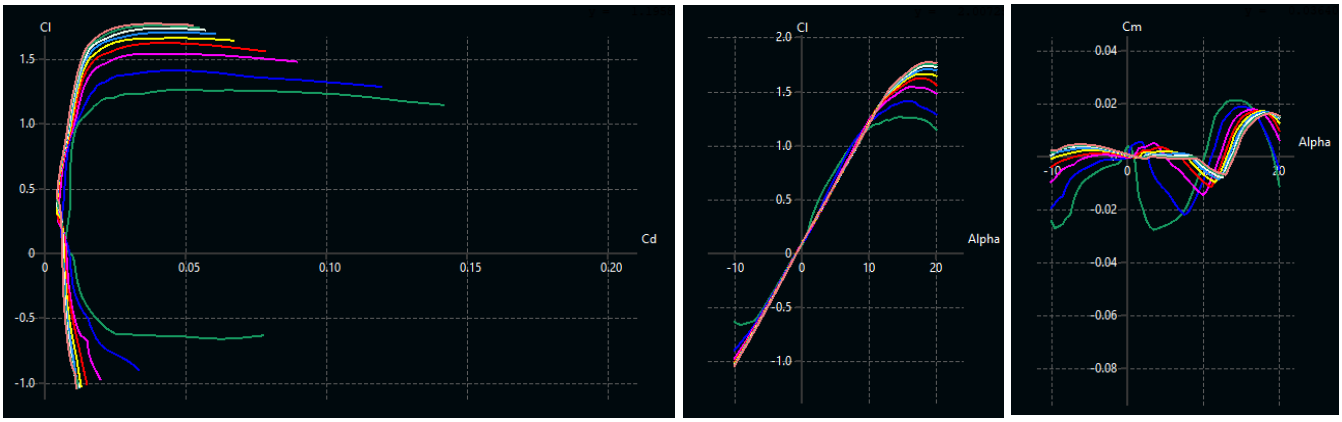


Figure 4.3 – M6 C_l/C_d Graph (left), C_l/α Graph (middle), and C_m/α graph (right)

NACA M-6 XFLR5 Analysis from -10 to +20 degrees angle of attack:

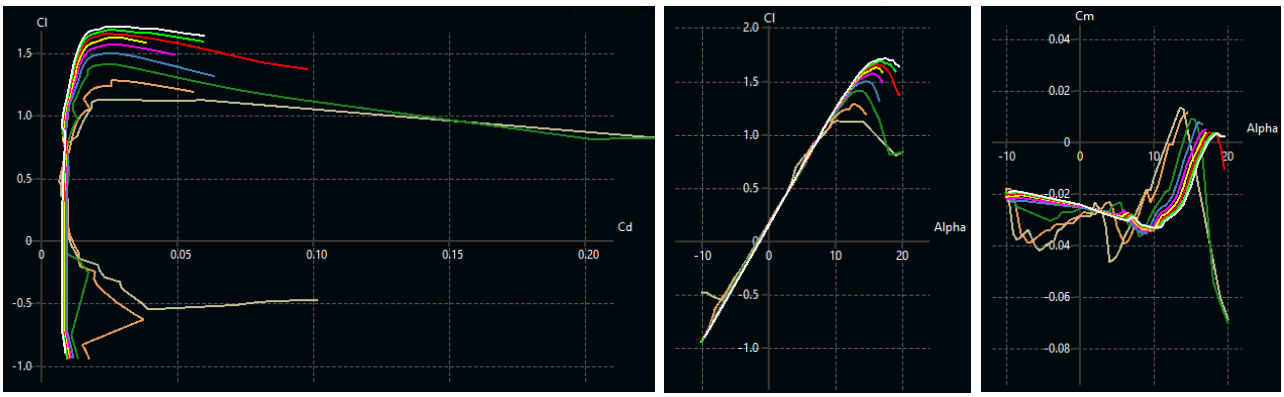


Figure 4.4 – M12 C_l/C_d Graph (left), C_l/α Graph (middle), and C_m/α graph (right)

4.4 Comparative Airfoil Analysis

With the individual results at a large range defined, it is important to also compare the potential airfoils against each other. Doing so allows for a surface level analysis for which airfoil could be utilized the most effectively at each end of the range of operational Reynold's numbers. The analysis below is conducted first at Re = 1,500,000 followed by results at Re = 8,500,000. Importantly, the comparative analysis is completed at a range from -20 degrees angle of attack up to a more drastic 30 degrees angle of attack for a full range potential operational values (and potentially beyond).

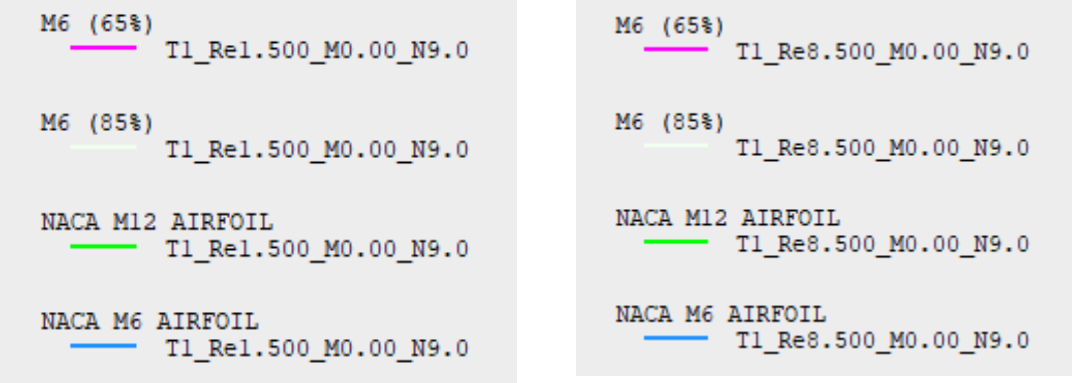


Figure 4.5 – Legend for the comparative XFLR5 analysis

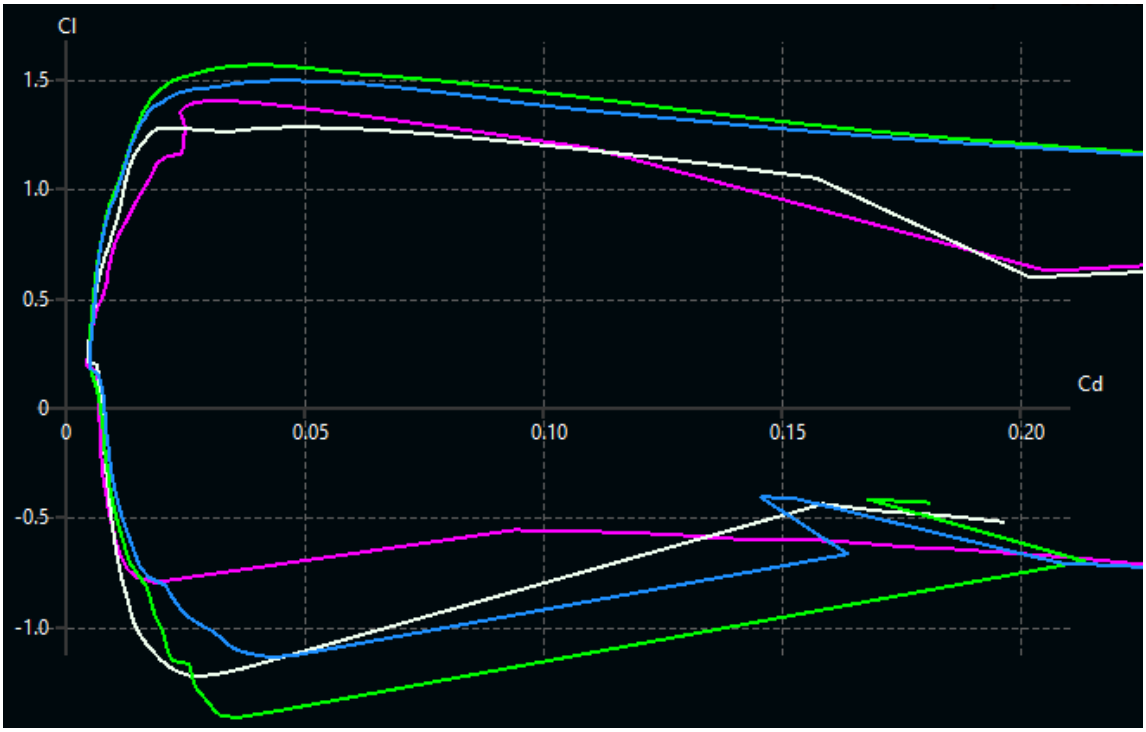


Figure 4.6 – C_l/C_d comparison at Re = 1.5E6

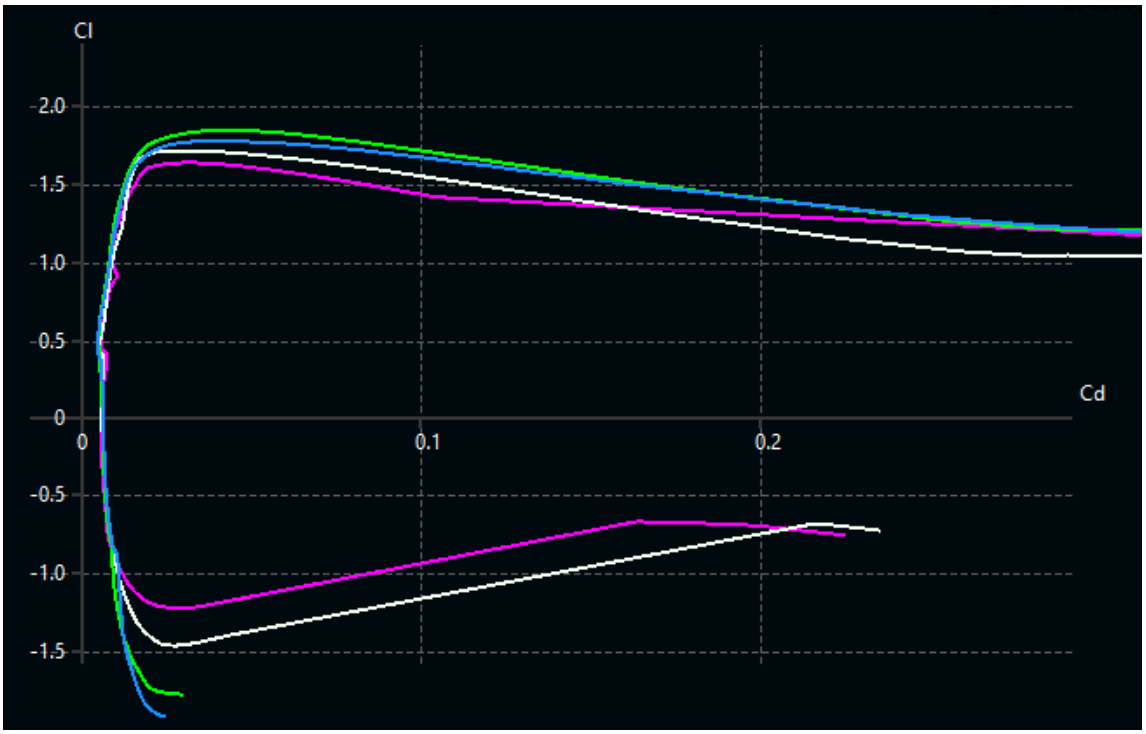


Figure 4.7 – C_l/C_d comparison at Re = 8.5E6

In the coefficients of lift vs drag comparison, the results of the airfoils have a much larger range results at the lower Re value compared to the higher Re value. As expected with thinner airfoils at lower speeds, the behavior is also more erratic. Interestingly, at the lower Re value it can be observed that the 65% thickness M6 airfoil begins to perform more favorably at positive C_l values compared to the 85% thickness M6, however it performs drastically worse at negative C_l values. Similarly, the 85% M6 performs more favorably at negative C_l values compared to the full size M6 up until what appears to be stall conditions.

At the higher Re evaluation, results are more consistent over the entire range evaluated. Looking to the 65% M6, lower performance is seen at high C_l and low C_d values, however as C_d increases, the 65% airfoil maintains a more consistent C_l value and eventually becomes more efficient. This is also seen to a smaller extent at the lower Re value, meaning it could prove more useful in the event the aircraft operates under higher drag conditions. Below continues the analysis with a comparison of C_l vs. α.

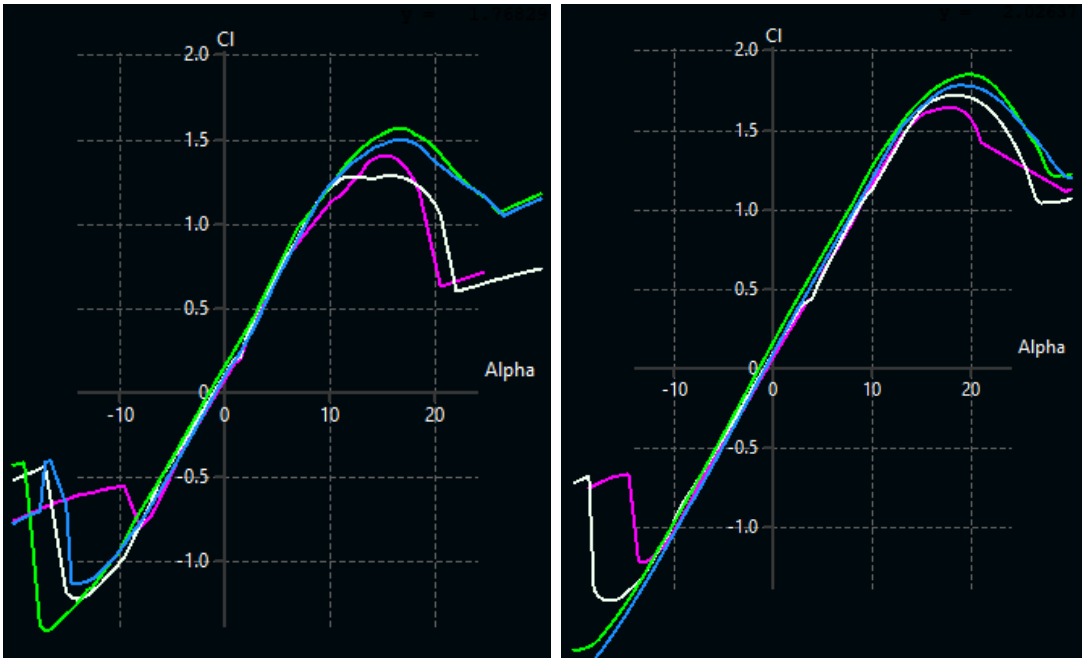


Figure 4.8 – C_l/α at Re = 1.5E6 (left) and Re = 8.5E6 (right)

The results for both high and low Re operating conditions remain consistent with expectations. The airfoils all perform similarly, with the M12 maintaining slightly higher C_l values during operation, which is also seen in the C_l vs C_d comparison. While all the airfoils exhibit similar stall conditions, apart from the 65% at negative angles of attack entering stall conditions much quicker at lower Re, the M12 and full size M6 do not enter stall under the analyzed negative angles of attack at the higher Re value, something that could be useful for a strike aircraft during aggressive maneuvers. Aside from these conditions, all airfoils remain similar enough to be considered for application. The final comparison below demonstrates C_m vs α.

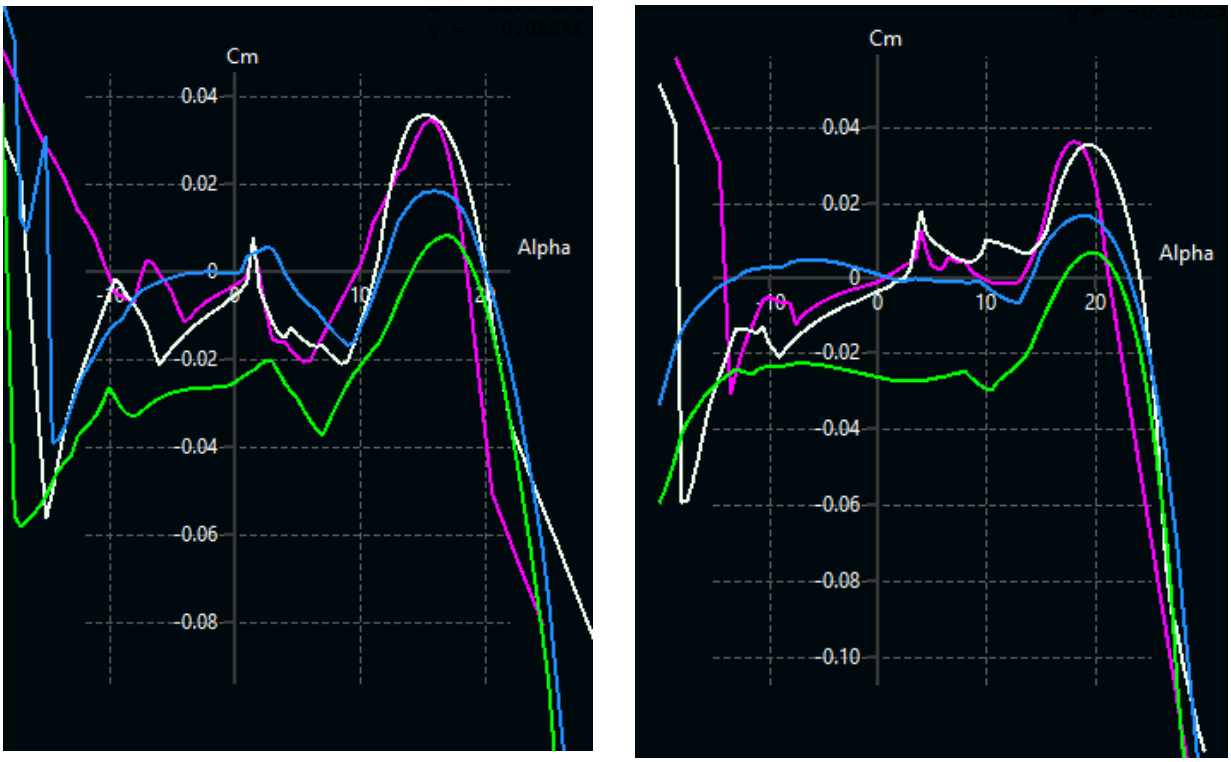


Figure 4.9 – C_m/α at Re = 1.5E6 (left) and Re = 8.5E6 (right)

Unlike the C_l results, the pitching moment coefficients are drastically different between the two evaluations. Strangely, the full size M6 (blue) exhibits erratic behavior, bouncing around constantly between being stable and unstable around both the 65% and 85% variants. The M12 (green) is consistently the most stable at both Re values, and the 65% (magenta) and 85% (white) compare very similarly over the majority of α ranges during both tests. While the M12 may look like the best option due to the instability of the other three at higher Re and less stable nature at lower Re, this is not necessarily the case. The reflexed nature of the M series airfoils creates this effect, and further increasing the amount of trailing edge reflex in the chord allows for the stability, or negative stability, to be increased or decreased. This is desirable in the case of a flying wing or tailless aircraft as it assists statically in addressing the issue of roll-yaw coupling and lack of control with a rudder. As a result, the M6 airfoils will likely be the best option going forward.

4.5 Airfoil Discussion and Selection

While the airfoils expectedly performed similarly, the 65% M6, or NACA M665, was chosen moving forward with the design. The aircraft possesses a large wing area compared to other ISR aircraft, that is more in line with US fourth-generation fighters and has a long root chord to match. The NACA M665 still exhibits the desired characteristics of the base M6, including the necessary reflex, but at only 65% thickness. When expanded to the desired root chord length this allows for a much thinner wing overall, improving drag characteristics.

5. Matching Graph

5.1 Matching Graph

To determine whether the theorized aircraft meets the general performance requirements for powered flight, a matching graph was created using Raymer's [9] methods and equations. For jet powered aircraft, the designed dot (red) must be above the lines shown in the graph to confirm that the thrust to weight ratio is adequate for flight. The matching graph shown below in Figure 5.1 was done using calculations for the proposed GE-J85 engines without the use of afterburner, ensuring the base engine power provides enough thrust for operation. Code used for the matching graph is provided in Appendix B.

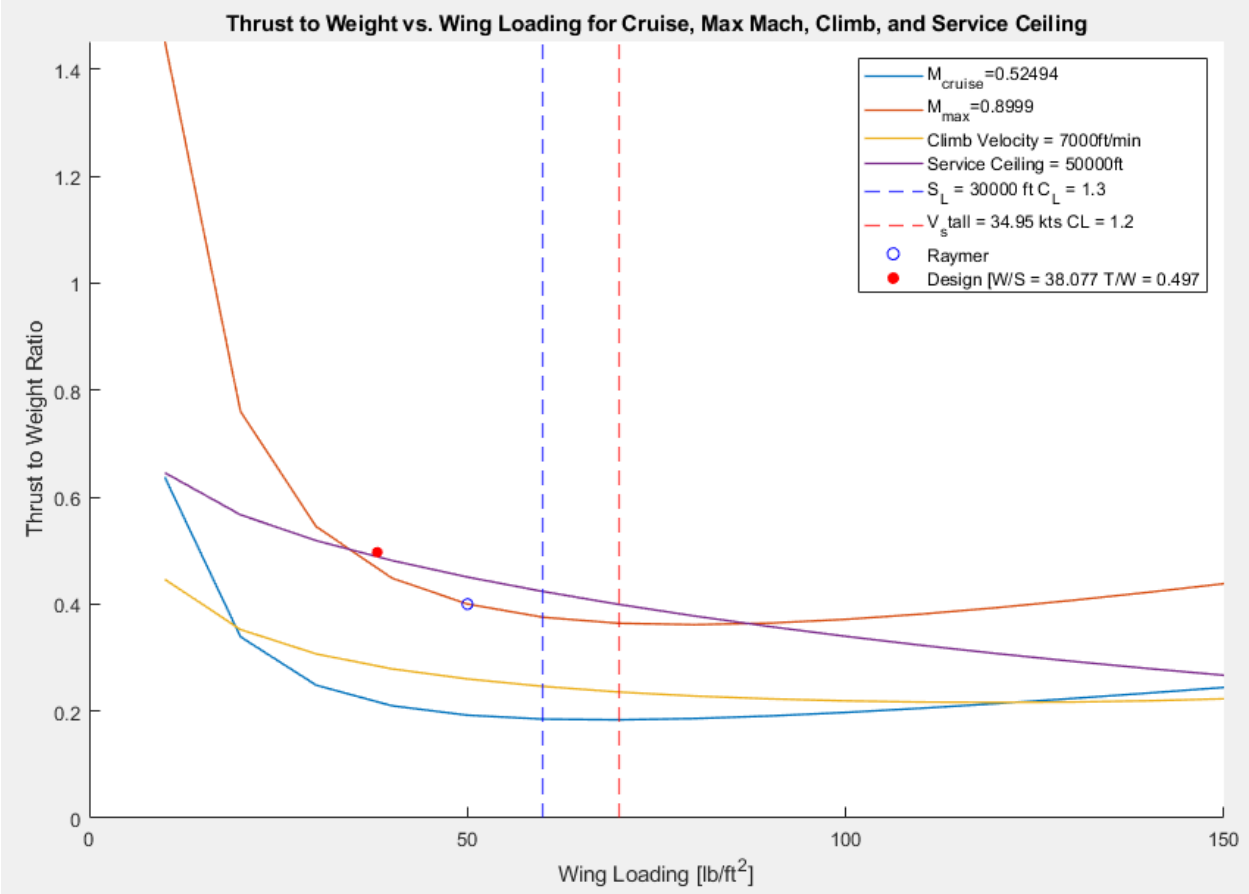


Figure 5.1 – Aircraft matching graph

6. Aircraft Configuration

6.1 Overall Configuration

The base test aircraft is designed similarly to many fourth-generation military ISR aircraft with a very large wing area more akin to fighter style aircraft. Given the desired flight configuration for the second model lacks a vertical stabilizer, additional stability from the larger wing area is shown to be worth the tradeoff of the increase in both weight and induced drag. Figure 6.1 below shows a multi-view layout of the aircraft geometry.

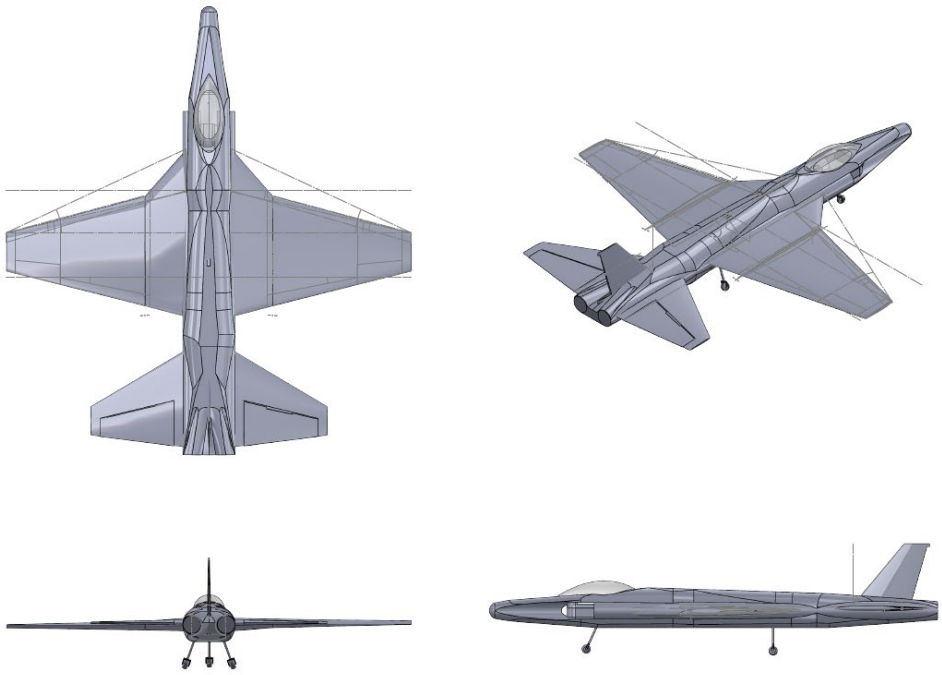


Figure 6.1 – Multi-view aircraft geometry

6.2 Fuselage Configuration

Consistent with fourth-generation strike and fighter aircraft of the past, the fuselage is a standard military style configuration with mid mounted wings that blends into the empennage of the aircraft. This configuration allows the aircraft to make extreme maneuvers during flight that other designs may not allow for. By blending the empennage into the fuselage, the design ensures a strong connection for lifting and control surfaces without the any centralized stress point at the connection.

6.3 Empennage Configuration

As previously stated, the empennage is joined into the fuselage making up the aft section of the aircraft. Both the vertical and horizontal stabilizer share the GOE 444 airfoil due to its thin and symmetrical nature saving weight and overall induced drag. Keeping in line with modern day strike and intelligence aircraft, the aircraft does not employ the now fighter standardized

stabilators. Instead opting for split elevons, unique to this design, to assist in yaw-moment and doubling as an additional airbrake if necessary. The comparative aircraft is intended to operate without the use of a vertical stabilizer, resulting in the base aircraft utilizing a relatively small rudder.

6.4 Wing Configuration

The aircraft utilizes a mid-wing configuration like the F-16 Viper. Below is a brief list of wing features:

- 25-degree initial wing sweep leading edge strake
- 15-degree wing leading edge weep to wingtip
- Taper Ratio = 0.3
- Wing area = 380.81 ft^2
- AR = 4.06
- Split Ailerons
- Split Flaps

Due to the low relative wing loading, leading edge control devices were deemed unnecessary for the design after XFLR5 VLM analysis for calculated takeoff and landing speeds.

6.5 Propulsion Configuration

Two General Electric J85-GE-21 engines are located at the rear of the empennage in a pusher configuration. The engines are offset of the fuselage centerline by 20.1 inches at the center point of each engine. To assist in additional yaw control, each engine is equipped with a 3-D thrust vectoring nozzle capable of 20 degrees of thrust deflection in the aircraft XZ-plane as well as differential thrust in the event of single engine failure or if additional control is called for during operation.

6.6 Landing Gear Configuration

The aircraft employs a traditional tricycle gear style landing gear seen in most fourth and fifth generation single seat military aircraft today. Due to the lack of wing tanks, the main rear gear retracts into the wing near the connection at the fuselage, and the forward gear retracts backwards into the aircraft near the pilot. The landing gear is of little consideration for the purpose of this project and is simply accounted for during weight and balance alone.

6.7 Configuration Selection Discussion

Aside from the relatively large wing area, the aircraft shows a similar configuration to other ISR aircraft that have been tested for decades and are still in operation. The purpose of this test is to propose alternative control methods for existing platforms to replace vertical geometry in the hopes of reducing drag and RCS while maintain relative stability and control to the tailed counterparts through added control systems, therefore it is important to stay in line with the general configuration philosophy of other similar aircraft.

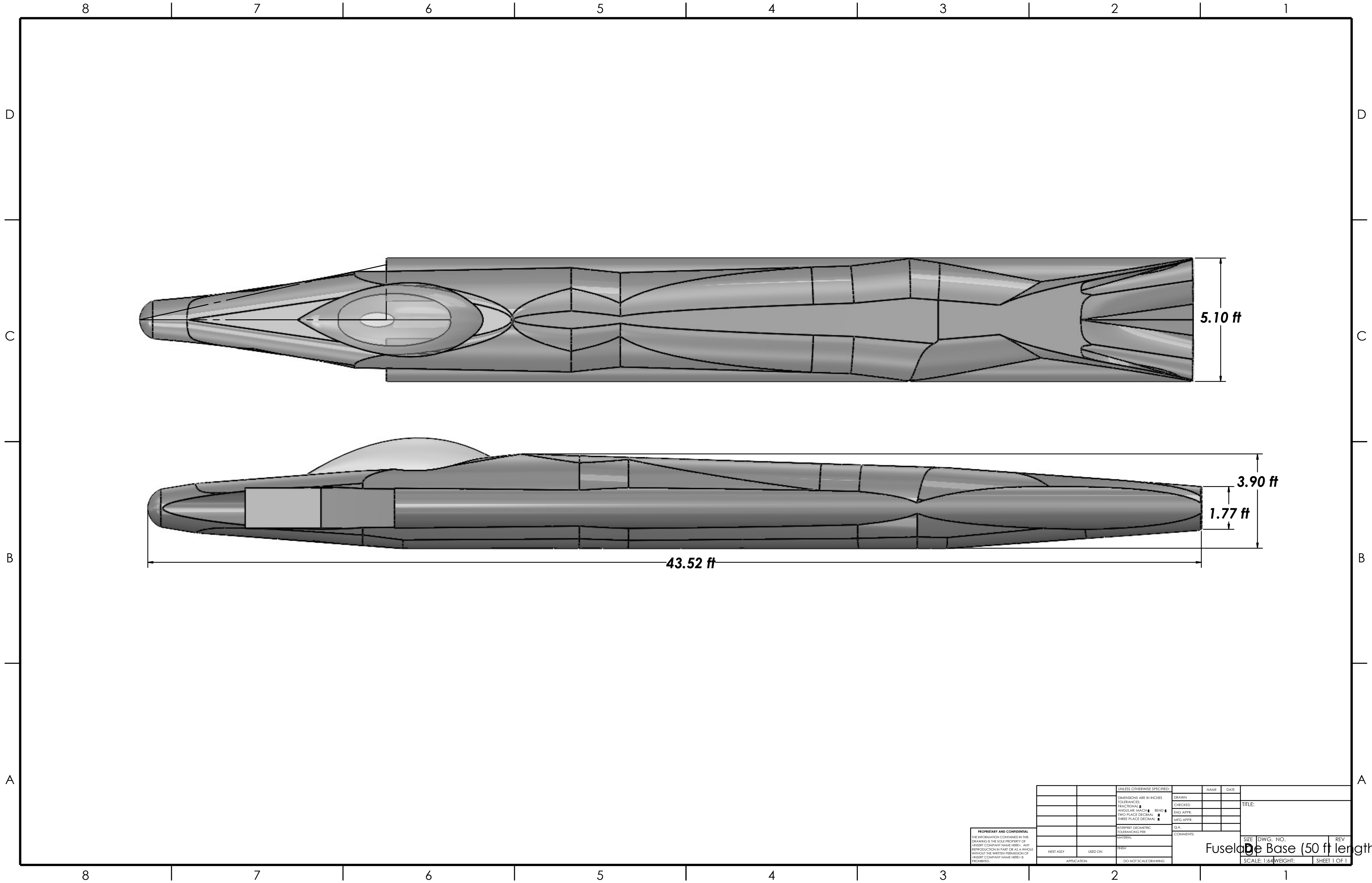
7. Fuselage Design

7.1 Fuselage Sizing

The fuselage is designed to be integrated with the wing, like modern day fighters. This is done to eliminate additional geometry that may further increase the RCS of the aircraft. The desired length found from previous analysis is in the range of 40 - 45 feet, similar in size to an F-16 or T-38. Due to the testing nature of the aircraft, a smaller integrated fuel tank is fitted internally allowing for just over 500 miles worth of fuel to be loaded to power the two General Electric J85 engines discussed in the propulsion section. Simple cylinders have been added to the fuselage model that are true to size of the J85's and the proper weight has been added for initial center of mass simulation. Additional weight will be added to the front of the fuselage once the landing gear and intakes have been properly designed and integrated.

7.2 Fuselage Drawing

The following page contains a CAD drawing of the fuselage with location and size of the internal fuel tank and engines.


8
7
6
5
4
3
2
1
D
C
B
A
5.10 ft
3.90 ft
1.77 ft
43.52 ft
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN INCHES
TOLERANCES:
FRACTIONAL ±
ANGULAR: MACH ± BEND ±
TWO PLACE DECIMAL ±
THREE PLACE DECIMAL ±
INTERPRET GEOMETRIC TOLERANCING PER:
MATERIAL
FINISH
NEXT ASSY
USED ON
APPLICATION
DO NOT SCALE DRAWING
NAME
DATE
DRAWN
CHECKED
ENG APPR.
MFG APPR.
Q.A.
COMMENTS:
TITLE:
SIZE
D
DWG. NO.
REV
Fuselage Base (50 ft length
SCALE: 1:64
WEIGHT:
SHEET 1 OF 1

8. Empennage Design

8.1 Empennage Overview

The empennage is designed to blend into the fuselage like most fourth and fifth generation fighter aircraft. By blending the fuselage and empennage into a single body, higher structural rigidity is achieved, facilitating the extreme maneuverability called for by fighters. Additionally, by having a blended configuration, the design eliminates a potential point of force coupling that may happen with tail boom designs due to the connection point between boom and fuselage.

Both the vertical and horizontal stabilizer utilize the GOE 444 airfoil, a thin symmetrical airfoil with low C_L and C_D characteristics. Since neither of these surfaces are the primary lift surface, the lower C_D is a more valuable tradeoff. Analysis of the GOE 444 airfoil is completed in section 8.2 below.

8.2 Empennage Airfoil Analysis

As stated, the GOE 444 is a thin symmetrical airfoil with modest C_L and C_D characteristics. Figure 8.1 below shows the imported airfoil in XFLR5, and the subsequent diagrams show the analysis at multiple Re values and angles of attack ranging from -15 degrees to 25 degrees.

Figure 8.1 – Imported GOE 444 airfoil

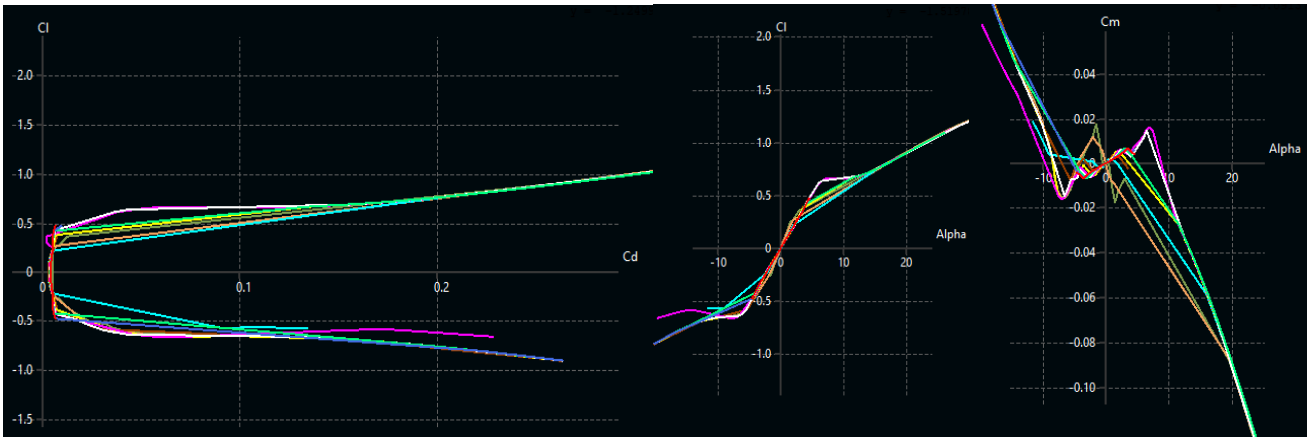


Figure 8.2 – GOE 444 C_l/C_d Graph (left), C_l/α Graph (middle), and C_m/α Graph (right)

8.3 Empennage Sizing

Both test aircraft will feature the same horizontal stabilizer, sizing featured below in Table 8.1, while only the base aircraft will feature the vertical stabilizer, featured in Table 8.2

Table 8.1 – Horizontal stabilizer parameters

Parameter	Value
Wing area (S)	108 ft
Wingspan (b)	18 ft
Aspect ratio (AR)	3
Taper ratio	0.3
MAC	6.58 ft
Span location MAC	3.69 ft
Root chord	9.23 ft
Tip chord	2.77 ft
Wing c/4 sweep angle	30 degrees

Table 8.2 – Vertical stabilizer parameters

Parameter	Value
Wing area (S)	60.26 ft
Aspect ratio (AR)	2
Taper ratio	0.3
MAC	4.49 ft
Span location MAC	3.08 ft
Root chord	7.5 ft
Tip chord	2.5 ft
Wing c/4 sweep angle	39.81 degrees
Moment arm (L_{VT})	14 ft

8.4 Empennage Control Surfaces

Like the control surfaces employed on the wing, the horizontal stabilizer utilizes split elevators in favor of traditional stabilators seen on modern day fighters. While this does limit more extreme maneuverability, it allows for an additional source of yaw control to assist in stability and control during roll maneuvers, and doubles as an additional airbrake. Tables 8.3 and 8.4 display control surface sizing for the horizontal and vertical respectively, and Figure 8.3 features a CAD drawing of the horizontal stabilizer and split elevator.

Table 8.3 – Elevator sizing parameters

Parameter	Value
S_e/S_h	0.469
Rudder spanwise fractions	0.1 to 0.9
Rudder chordwise fractions	0.5 inner, 0.4 outer

Table 8.4 – Rudder sizing parameters

Parameter	Value
S_r/S_v	0.093
Rudder spanwise fractions	0.9
Rudder chordwise fractions	0.25 lower, 0.1 upper

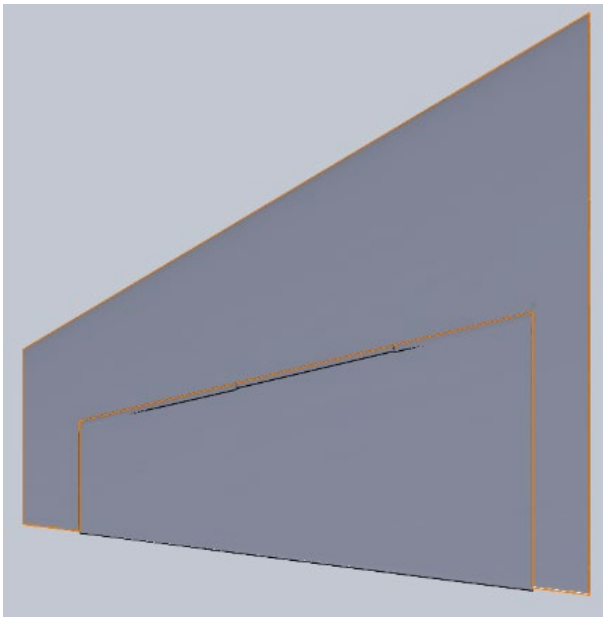


Figure 8.3 – Horizontal stabilizer with split elevator

9. Wing Design

9.1 Main Wing Design

The NACA M665 airfoil was chosen for wing design as discussed above due to the reflexed nature of the NACA M series airfoils. Having a reflexed chord line adds a negative pitching moment to the trailing edge of the wing which has been shown to have beneficial flight attributes when applied to flying wing configurations for aircraft [5], resulting in a proverse yaw effect during aileron roll maneuvers. While the NACA M6 did show to have better flight characteristics regarding C_l and C_d, the size of the wing would make the M6 far too heavy for the desired design requirements, so the tradeoff of slightly worse flight characteristics in favor of a thinner wing was chosen.

With the NACA M665 airfoil chosen, overall wing design could be completed and analyzed. The design features a wingspan of 39.56 ft, wing area of 380.81 ft^2, leading to an aspect ratio of 4.06. The wing incorporates a variable sweep angle over the leading edge, starting with a sharper 25-degree sweep from the root and decreasing to a 15-degree sweep for most of the chord. The 25-degree sweep is designed to act as a leading-edge strake to add additional vortex lift during high angle of attack maneuvers. The strake also doubles as a weight reduction method by making the overall wing geometry smaller, as the initial wing had a constant 15-degree sweep from root to tip. The leading-edge strake follows the NACA M665 airfoil design and is not altered or thinned. While thinning of the leading edge has been proven and may improve flight performance, altering an airfoil is outside of the scope of the current project. Table 9.2 displays the wing geometry and sizing.

Table 9.1 – Wing geometry and sizing

Parameter	Value
Reference Wing Area (S)	380.81 ft
Aspect Ratio (AR)	4.10
Wingspan (b)	39.56 ft
Mean Aerodynamic Chord (c)	8.77 ft
Aerodynamic Center Longitudinal Location (x)	24.55 ft
Aerodynamic Center Lateral Location (y)	0 ft
Taper Ratio (λ)	0.3
Leading Edge Sweep Angle (Λ)	25 deg (strake), 15 deg
Quarter Chord Sweep Angle (Λ)	25 degrees
Trailing Edge Sweep Angle (Λ)	14.6 deg
Dihedral Angle (Γ_W)	0 deg
Incidence Angle (i_W)	1.5 deg
Geometric Twist Angle	-4 deg

9.2 Main Wing Control Surfaces

To assist in flight control during the second configuration (no vertical stabilizer), the wing incorporates both split flaps and split ailerons. The split flaps are designed to function like a split aileron rather than a traditional split flap, with the top surface also being capable of deflection upwards if additional yaw force is needed. The split flap is 10% of the chord from the trailing edge

and begins at 20% of the span from the fuselage, ending at 70% span where the split aileron begins. The aileron is slightly larger regarding chord length at 20% of the chord, running from 70% span to 95% span of the wing. Sizing and positioning of the aileron is designed to optimize the potential yaw control when operated in an "open" configuration, where each side of the aileron opens away from the other. This operation will create additional drag on the wingtip in turn inducing a yaw moment to assist in control with yaw-roll coupling. No leading-edge control surfaces were added to the design as analysis has proven proper lift requirements were met between the large wing area and trailing edge control surfaces, shown below in section 9.3. Figure 9.1 demonstrates the approximate sizing and locations of the control surfaces on the wing.

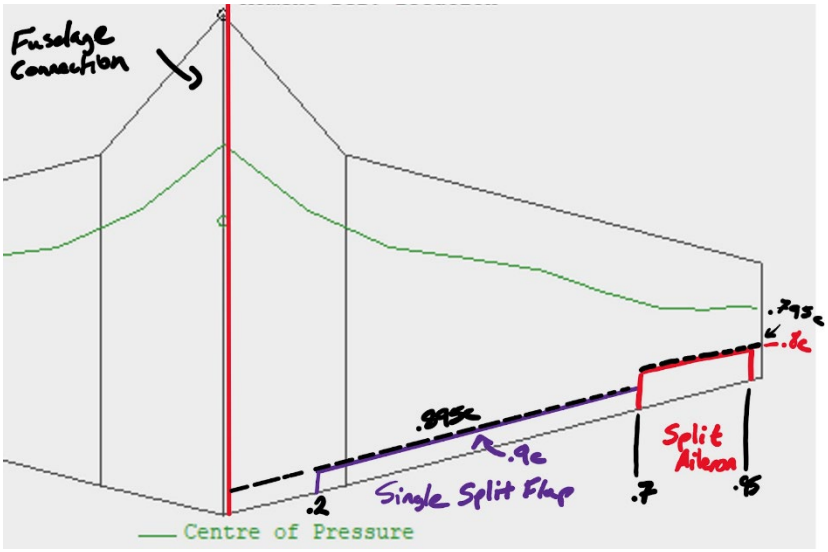


Figure 9.1 – Sketch of control surfaces on XFLR5 wing design

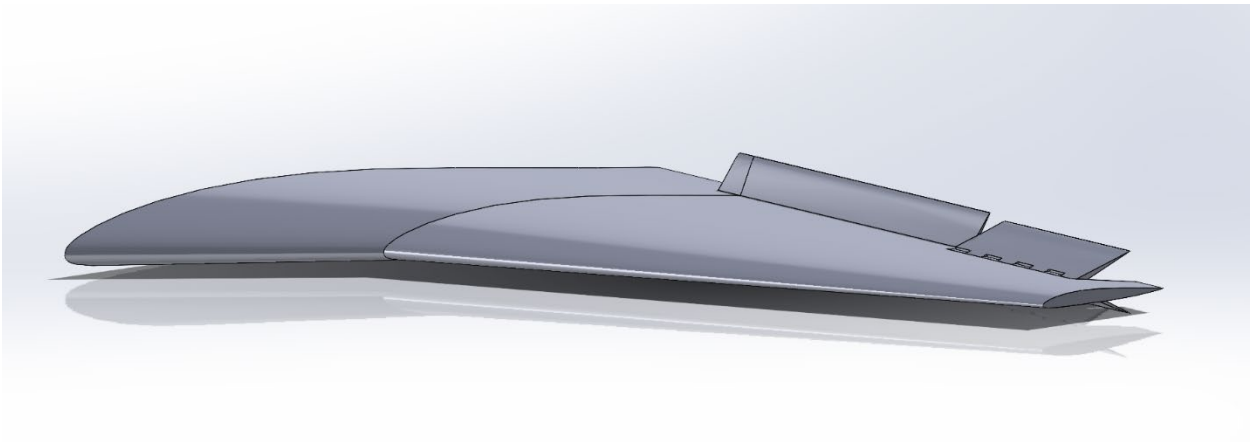


Figure 9.2 – CAD side view of wing with control surfaces

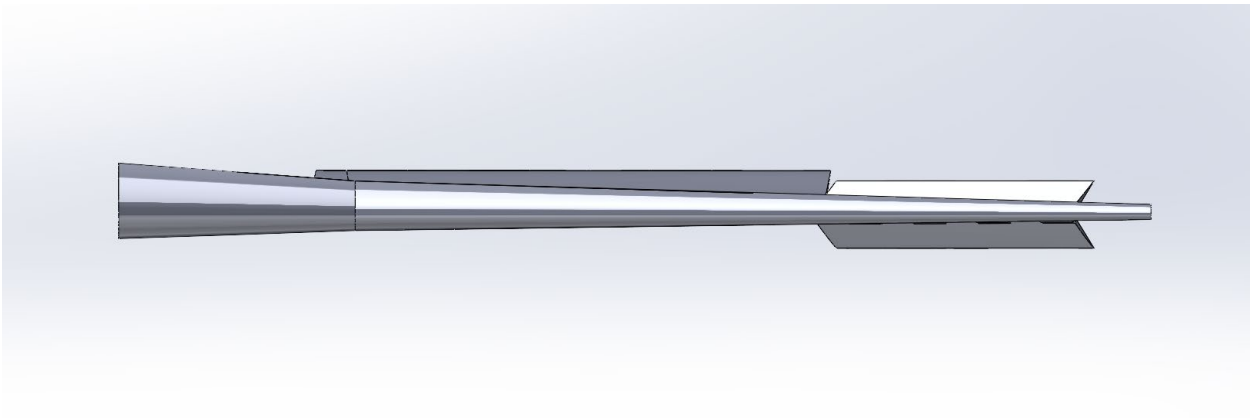


Figure 9.3– CAD front view of wing with control surfaces

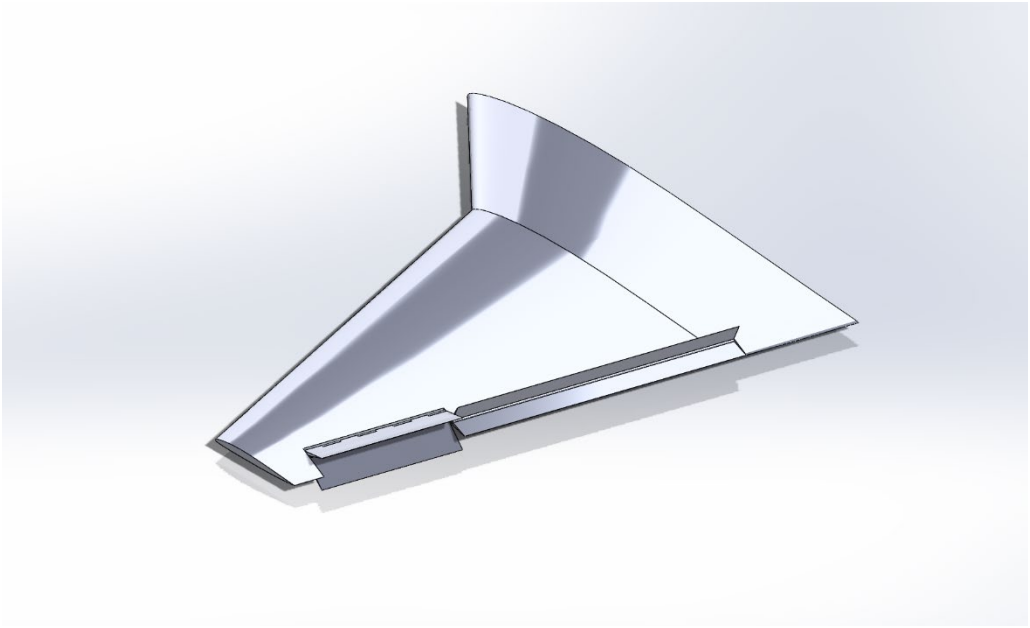


Figure 9.4 – CAD rear view of wing with control surfaces

9.3 Main Wing Analysis

With the wing geometry imported into XFLR5, an analysis was completed to identify the effectiveness of the design. Table 9.2 shows the MATLAB calculated wing requirements, and Table 9.3 below shows the results of the analysis, and the following figures display the analysis itself. For comparison, a calculated increase in Cl_{Max} during takeoff and landing flap deflection is added to Table 9.2 using equation 12.21 from Raymer's *Aircraft Design: A Conceptual Approach* [9], where $\Lambda_{H.L}$ is the flap deflection angle.

$$\Delta \mathbf{Cl}_{\boldsymbol{Max}} = \mathbf{0.9}\Delta \mathbf{Cl}_{\boldsymbol{Max}} * \left(\frac{\boldsymbol{S_{Flapped}}}{\boldsymbol{S_{Reference}}}\right) * \boldsymbol{cos}\boldsymbol{\Lambda_{H.L.}} \tag{9.1}$$

Table 9.2 – MATLAB calculated Cl_{Max} requirements

Required Takeoff Cl_{Max}	1.55
Required Landing Cl_{Max}	1.75
Calculated Takeoff Cl_{Max} (Raymer)	1.82
Calculated Landing Cl_{Max} (Raymer)	2.12

Table 9.3 – XFLR5 analysis results

Wing c/4 sweep angle	15.971 degrees
Flap hinge line sweep angle	11.013 degrees
Clean V_{Stall}	34.945 kt (64.718 km/h)
c_{root}	16.228 ft (4.946 m)
c_{tip}	3.690 ft (1.125 m)
Root airfoil Re number at V_{Stall}	6.0003×10^6 (5.928e6 from XFLR5)
Tip airfoil Reynolds number at V_{Stall}	1.3645×10^6 (1.348e6 from XFLR5)
Root airfoil name	NACA M665 (M6 65% Thickness)
Tip airfoil name	NACA M665 (M6 65% Thickness)
Root airfoil Cl_{Max}	1.560
Tip airfoil Cl_{Max}	1.352
Clean Cl_{Max}	1.489
Takeoff flap deflection angle	15 degrees
Landing flap deflection angle	40 degrees
3-D Takeoff flap lift increase	0.114
3-D Landing flap lift increase	0.258
Takeoff Cl_{Max}	1.603
Landing Cl_{Max}	1.748

Due to the nature of XFLR5 analysis, it is reasonable to expect small uncertainties in the results. While the landing Cl_{Max} falls slightly short of the calculated requirement (0.012 less), between the analysis results and the hand calculations, 1.75 and 2.12 respectively, the wing design should prove more than effective for further analysis.

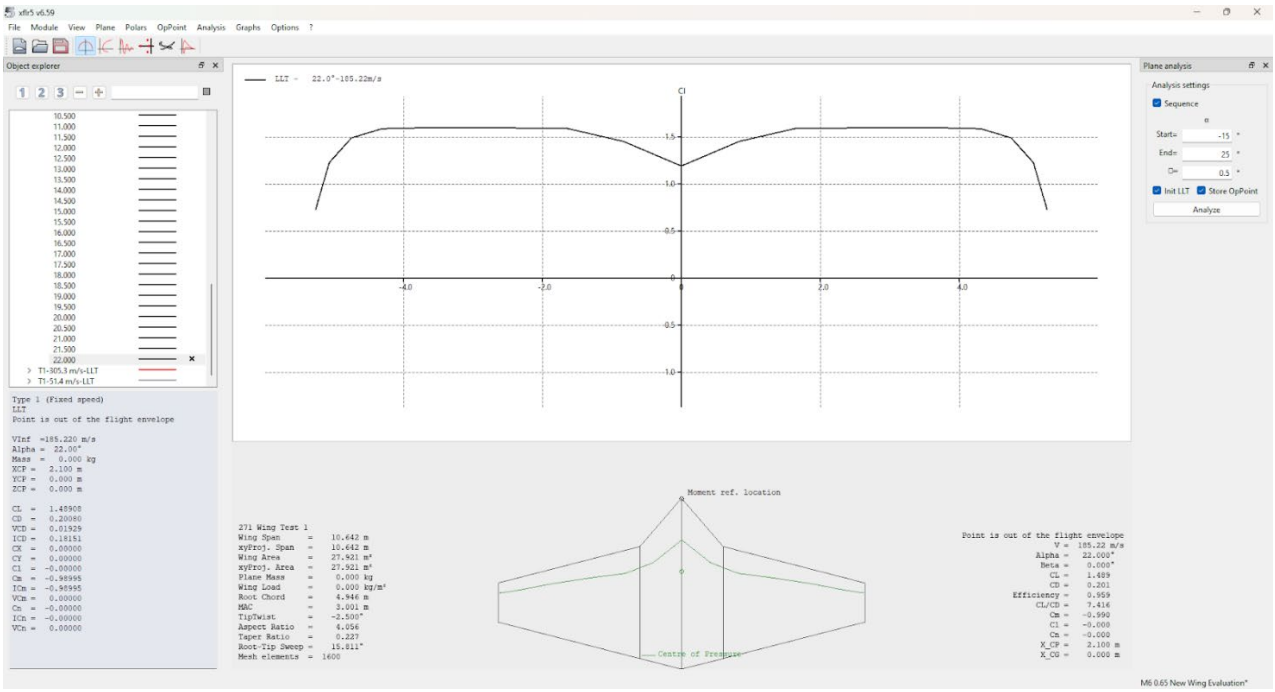


Figure 9.5 – XFLR flight coefficient analysis

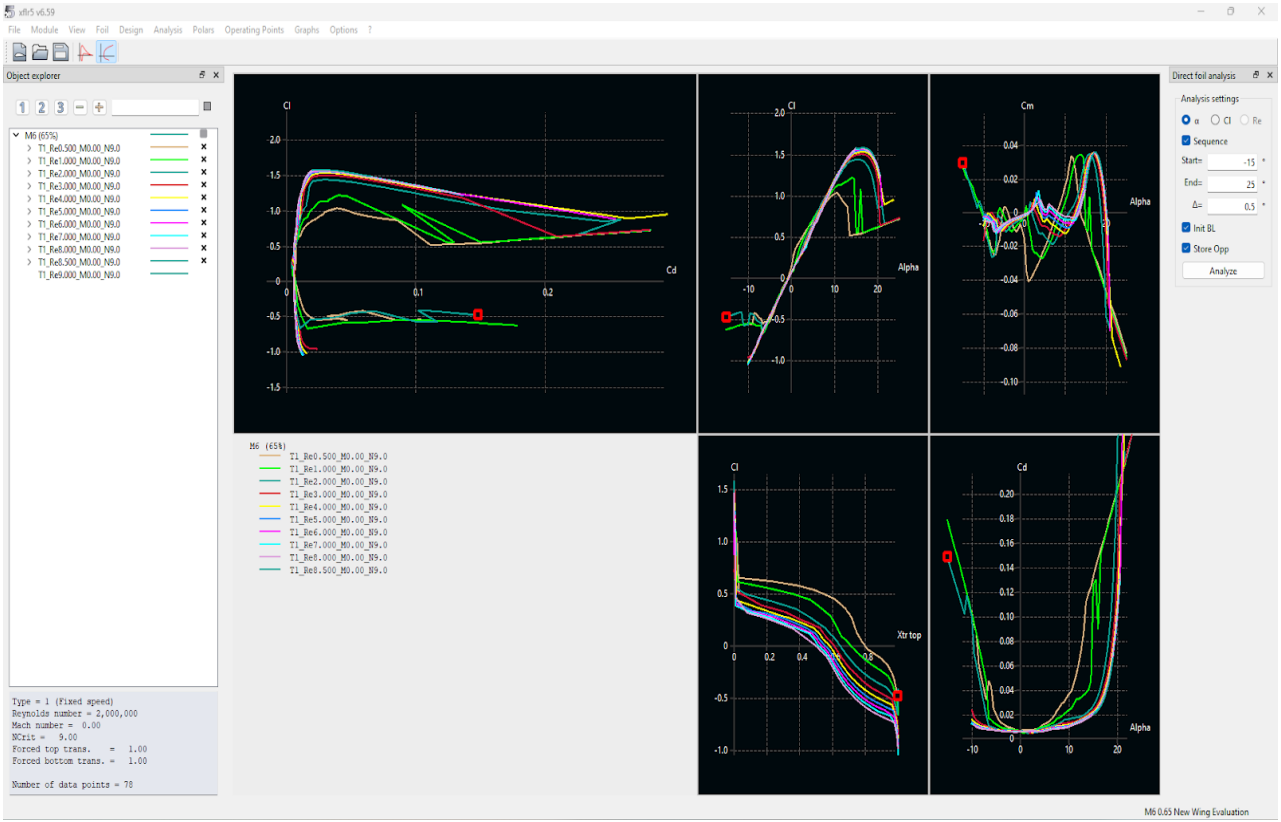


Figure 9.6 – XFLR5 wing Cl_{Max} analysis

10. Weight and Balance

10.1 Weight and Balance Overview

Before getting to the first major analysis of the project, stability and control, weight and balance of the current model must be assessed. To accomplish a relatively accurate weight and balance assessment, a mix of CAD modeling in SolidWorks and weight assumptions using Daniel Raymer's weight estimation table, Table 15.2 [5]. The combination of both methods gives an estimation that is accurate enough for further analysis. Since the aircraft is not going to be produced physically, components can and have been applied as point loads in the CAD model as well.

10.2 Weight and Balance Assumptions

As stated above, weight assumptions for the design were calculated using the table provided by Raymer below in Figure 10.1 [5]. While these assumptions may not be usable for a construction aircraft, they will suffice to give a close enough approximation for the simulated tests going forward regarding stability and controls. The assumptions are compared against initial and adjusted values to ensure the weight used in the matching graph above in Section 5 is properly updated and feasible for the proposed design.

Table 15.2 Approximate Empty Weight Buildup

	Fighters		Transport & Bomber		General aviation			
	lb/ft^2	kg/m^2	lb/ft^2	kg/m^2	lb/ft^2	kg/m^2	Multiplier	Approximate location
Wing	9	44	10	49	2.5	12	$S_{\text{exposed planform}}$	40% MAC
Horizontal tail	4	20	5.5	27	2	10	$S_{\text{exposed planform}}$	40% MAC
Vertical tail	5.3	26	5.5	27	2	10	$S_{\text{exposed planform}}$	40% MAC
Fuselage	4.8	23	5	24	1.4	7	$S_{\text{wetted area}}$	40–50% length
	Weight ratio		Weight ratio		Weight ratio			
Landing gear*	0.033		0.043		0.057		TOGW	centroid
Landing gear—Navy	0.045		—		—		TOGW	centroid
Installed engine	1.3		1.3		1.4		Engine weight	centroid
"All-else empty"	0.17		0.17		0.1		TOGW	40–50% length

*15% to nose gear, 85% to main gear; reduce gear weight by 0.014 W_0 if fixed gear.

Figure 10.1 – Raymer approximate component aircraft weights [5]

10.3 Initial Aircraft Weight Balance Calculations

To assist in efficiently calculating the center of mass of the aircraft through different iterations, an excel calculator was used. The calculator itself is based off Daniel Raymer's RDSwin software [9] and was created by Professor Sean Montgomery of San Jose State University. All values were edited to properly reflect the measurements of the aircraft CAD and XFLR5 models and estimated values from Figure 10.1. Table 10.1 below displays the results of the calculator.

Table 10.1 – Weight and balance calculator results

Component	Weight (lbs)	X – Location (ft)	Z – Location (ft)	X – MOI (ft*lbs)	Z – MOI (ft*lbs)
Wings	2162.4	19.7	6.9	42527.3	15007.1
Horizontal Stabilizer	212.9	35.0	7.2	7450.7	1526.3
Vertical Stabilizer	120.5	37.5	8.8	9508.1	2236.3
Fuselage	1,419.8	18.3	6.9	26030.0	9768.4
Main Landing Gear	530.0	26.3	0.0	13911.8	0.0
Nose Landing Gear	93.5	9.2	0.0	857.3	0.0
Engine Weight	1,368.0	27.6	6.9	37734.0	9411.8
Engine Components Installation	547.2	27.6	6.9	15093.6	3764.7
Misc. Equipment	1,450.0	18.3	6.9	26583.3	10005.0
Pilot	180	15.3	7.3	2760.0	1314.0
Fuel	5478.9	23	6.9	126015.1	37804.5
Empty Weight Allowance (10%)	919.6	21.8	6.4	17969.6	5172.0
Empty Weight CG	10,115.3	**21.8**	**6.4**	197,665.8	56,891.6
Takeoff Gross Weight CG	14,500	**22**	**6.6**	326,440.8	96010.1

With the center of mass established, the stability analysis in Section 11 was able to be conducted and analyzed both mathematically and within XFLR5.

11. Initial Stability Analysis

11.1 Stability Analysis Overview

The results of the stability analysis for the initial aircraft are listed below in the following subsections. It is important to ensure that the test aircraft functions safely for the pilot and serves as an effective testbed and comparison aircraft for further study and analysis. Section 11.2 covers an initial stability analysis regarding the neutral point along the longitudinal axis (X_{NP}) and CGs calculated in Section 10, and Section 11.3 reviews an XFLR5 Vortex Latice Method 2 analysis of both longitudinal and lateral stability modes.

11.2 Initial Aircraft Stability Analysis

Figure 11.1 below shows the XFLR5 wing and stabilizer geometry model as well as the X_{NP} after the successfully completed flight analysis from a range of -30 degrees AoA to 20 degrees AoA. The test was conducted at a range from calculated Stall Speed (35 ktas) up to the calculated maximum cruise speed (593.5 ktas). Figure 11.2 shows the CAD model with accompanying relevant geometry locations (inches) for the stability analysis. Lastly, Table 11.1 displays the results of the initial stability analysis.

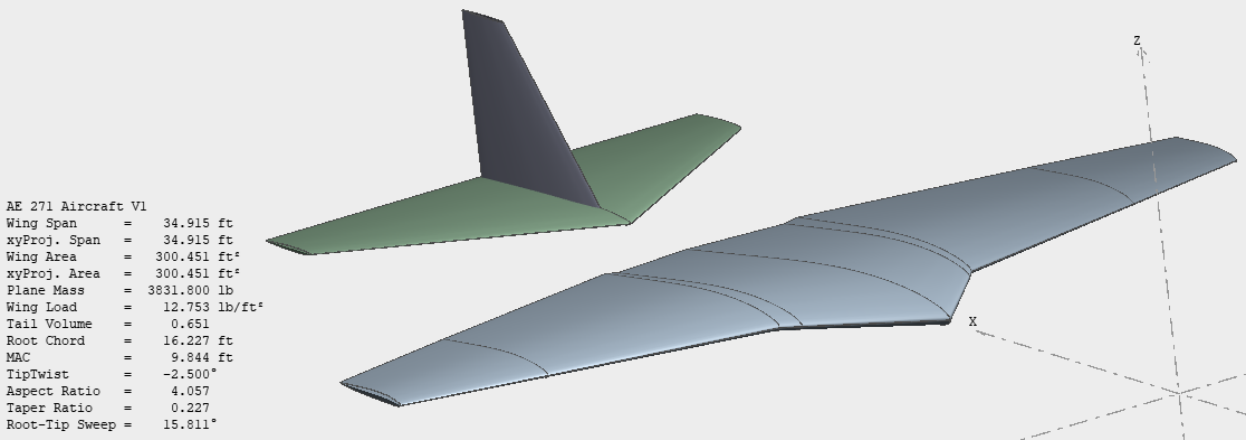


Figure 11.1 – XFLR5 aircraft model for initial stability analysis

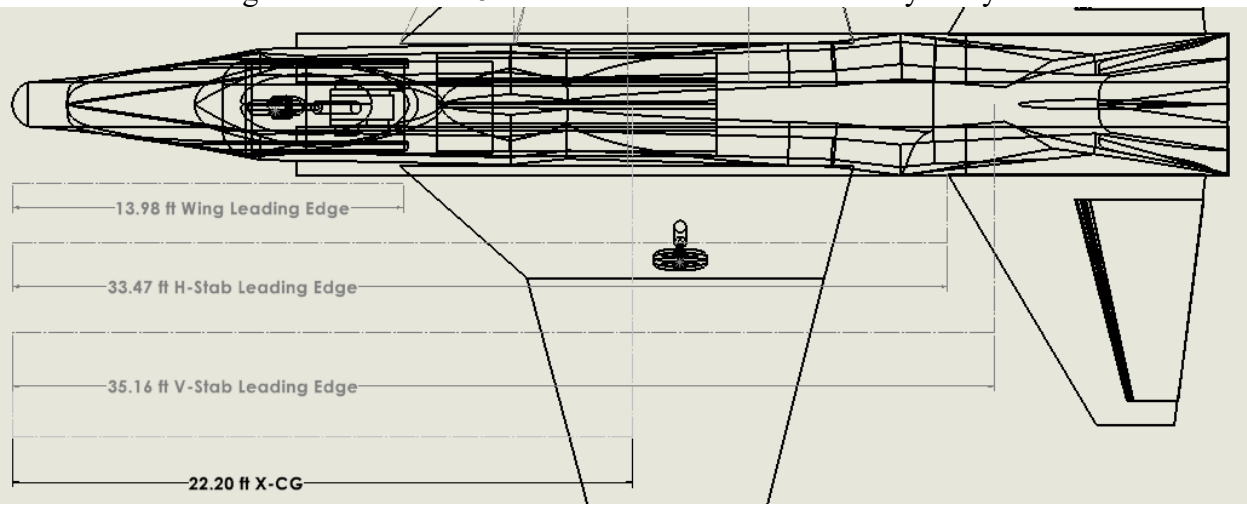


Figure 11.2 – CAD model of aircraft with relevant stability measurements

Table 11.1 – Stability analysis results

Parameter	Value
X_{NP} (neutral point location) from XFLR5	24.553 ft
X_{CG_Fore}	22.2 ft
X_{CG_Aft}	22.5 ft
$\bar{c}$	9.844 ft (23.825 ft from x = 0 Datum)
Static $Margin_{Fore} = (X_{NP}-X_{CG\_Fore})/\bar{c} * 100\%$	23.90 %
Static $Margin_{Aft} = (X_{NP}-X_{CG\_AFt})/\bar{c} * 100\%$	20.85 %
X_0 distance from origin (reference point) to aircraft nose	0 ft
X_1 distance from aircraft nose to wing apex	13.981 ft
$\bar{x}$ distance from wing apex to the leading edge of the mean aerodynamic chord (MAC)	9.844 ft
X_{NP_MAC} distance from MAC leading edge to neutral point	-0.728 ft
$X_0 + X_1 + \bar{x} + X_{NP\_MAC}$ (Should match X_{NP})	24.553 ft
$X_{CG\_MAC\_Fore}$ distance from MAC leading edge to forward CG	1.625 ft
$X_{CG\_MAC\_Aft}$	1.925 ft
NP_MAC = XNP_MAC/$\bar{c}$ * 100%	-3.056 %

With a positive static margin between 20.85 % and 23.90 %, the aircraft is shown to be statically stable in the longitudinal axis.

11.3 Modal Stability Analysis

With stability and controllability being one of the primary concerns of the study, modal stability analysis is crucial for both variations of the aircraft. To accomplish this, a simplified model fuselage was constructed and attached to the wing model used for analysis in the previous section. The model, shown below in Figure 11.3, was subjected to an inviscid analysis at standard sea level conditions for both longitudinal and lateral stability with inertial parameters calculated by hand using the following equations provided in the Raymer textbook [9]. Final inertial parameters are displayed below in Table 11.2.

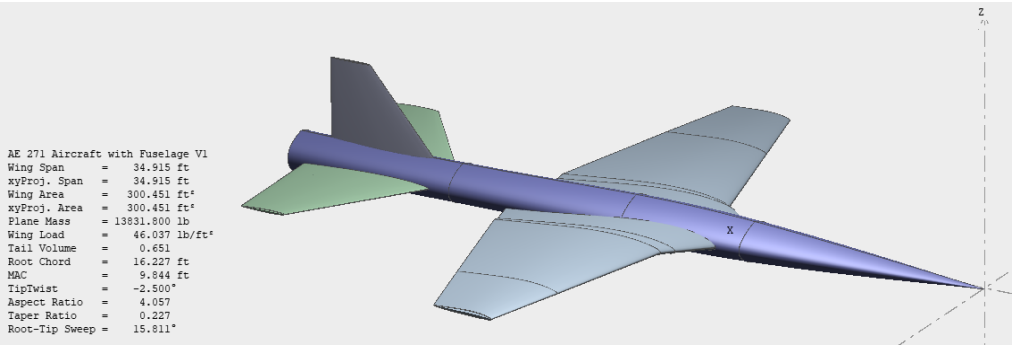


Figure 11.3 – XFLR5 model for aircraft modal stability analysis

$$Roll\ Moment\ of\ Inertia: I_{xx} = \frac{b^2 M R_x^2}{4} = \frac{b^2 W R_x^2}{4g} \tag{11.1}$$

$$Pitch\ Moment\ of\ Inertia: I_{yy} = \frac{L^2 M R_y^2}{4} = \frac{L^2 W R_y^2}{4g} \tag{11.2}$$

$$Yaw\ Moment\ of\ Inertia: I_{zz} = \left(\frac{b+L}{2}\right)^2 \frac{M R_z^2}{4} = \left(\frac{b+L}{2}\right)^2 \frac{W R_z^2}{4g} \tag{11.3}$$

Table 11.2 – Inertial parameters for stability analysis

ıertia paramete	Mean value	ɔain(unit/ctrl)	Unit
Mass	14500.000	0.000	lb
CoG_x	22.000	0.000	ft
CoG_z	1.200	0.000	ft
Ixx	9328.787	0.000	lbm.ft²
Iyy	25464.591	0.000	lbm.ft²
Izz	52568.240	0.000	lbm.ft²
Ixz	0.000	0.000	lbm.ft²

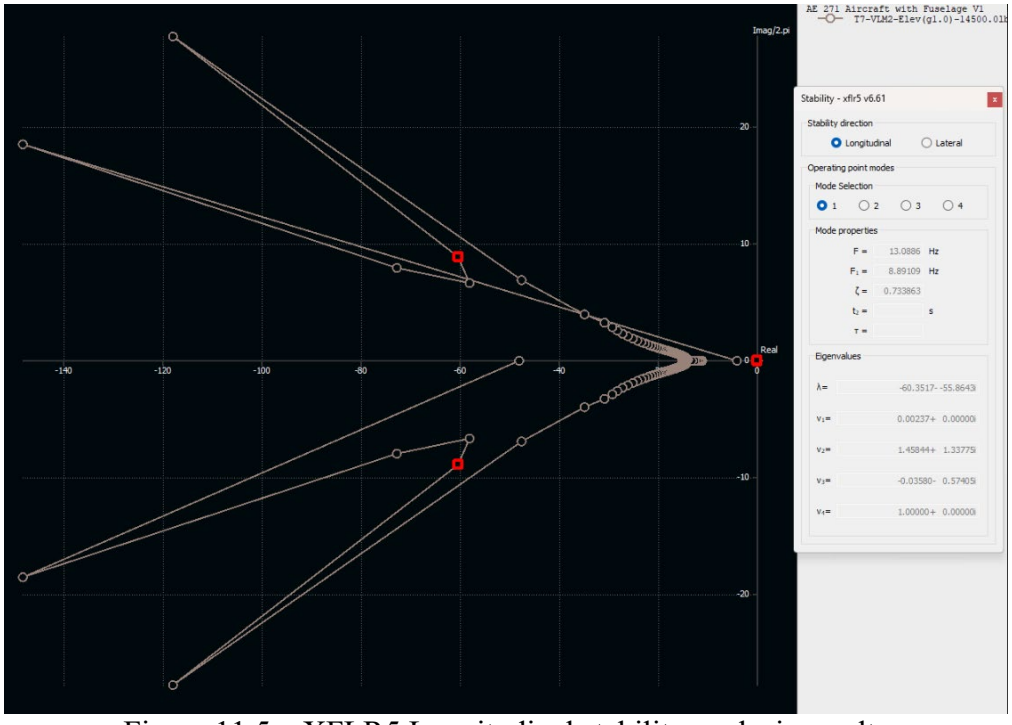


Figure 11.5 – XFLR5 Longitudinal stability analysis results

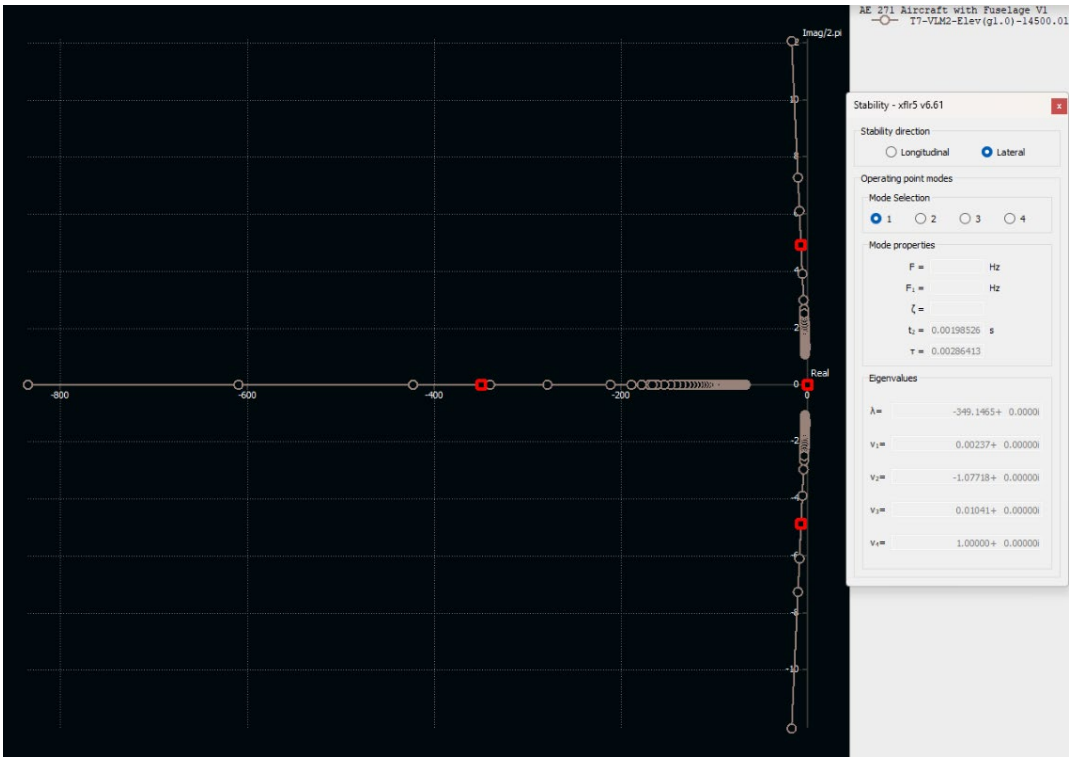


Figure 11.6 – XFLR5 lateral stability analysis results

After the simulation completed, eigenvalues were produced for each tested angle within the range of -5 degrees to 5 degrees angle of attack. Displayed below in Table 11.3, the aircraft shows full stability in the longitudinal mode, with an extremely stable Short-Period Mode and a marginally stable Phugoid Mode. Lateral modes also show high levels of stability regarding the Roll and Dutch-Roll Mode, with the Spiral Mode remaining unstable which is to be expected due to the aircrafts lack of dihedral in the wing.

Table 11.3 – XFLR5 modal stability results

___Longitudinal modes____	Short-Period Mode		Phugoid Mode	
Eigenvalue:	-60.35+ -55.86i	-60.35+ 55.86i	-0.0006183+ -0.02603i	-0.0006183+ 0.02603i
u/u0:	0.002373+ 0i	0.002373+ 0i	0.002373+ 0i	0.002373+ 0i
w/u0:	1.458+ 1.338i	1.458+ -1.338i	-1.82e-05+-8.033e-09i	-1.82e-05+8.033e-09i
q/(2.u0/MAC):	-0.333+ -5.34i	-0.333+ 5.34i	2.289e-06+1.528e-08i	2.289e-06+-1.528e-08i
theta(rad):	1+ 0i	1+ 0i	1+ 0i	1+ 0i

___Lateral modes____	Roll Mode	Dutch-Roll Mode		Spiral Mode
Eigenvalue:	-349.1+ 0i	-6.86+ -30.67i	-6.86+ 30.67i	6.625e-05+ 0i
v/u0 :	0.002373+ 0i	0.002373+ 0i	0.002373+ 0i	0.002373+ 0i
p/(2.u0/Span):	-1.077+ 0i	-5.147e-05+2.135e-05i	-5.147e-05+-2.135e-05i	6.214e-06+ 0i
r/(2.u0/Span):	0.01041+ 0i	0.0002013+0.0009205i	0.0002013+-0.0009205i	0.002183+ 0i
phi(rad):	1+ 0i	1+ 0i	1+ 0i	1+ 0i

11.4 Stability Analysis Discussion

Initial stability analysis and calculations show the aircraft to be statically stable during flight. This shows reflects the initial design goal of an ISR aircraft capable of stable loiter during flight. While the aircraft could be reassessed in the event a more dynamic flight model is required, for the purposes of this study it is not necessary to take such action. The results serve as a valid baseline for the comparison to come with the rudderless variant in the following chapters, as well as providing the eigenvalues necessary to begin designing control systems for both aircraft. While the poles presented in Section 11.3 could be used during creation and analysis of the MATLAB Simulink control program, the following section regarding the tailless variant created an issue with XFLR5 and therefore, all values will be calculated analytically for both aircraft. This analysis does, however, show the aircraft is stable in the current configuration and is therefore a valid platform for upcoming comparative testing.

12. Tailless Variant Weight and Balance

12.1 Tailless Weight and Balance Overview

With analysis of the base aircraft complete, it is time to move on to the tailless variant. This section will cover the weight and balance calculations of the tailless variant, the change to longitudinal stability of the variant, and feature a comparison in the conclusion to validate and explain any unexpected changes between the two aircraft.

12.2 Tailless Weight and Balance Calculations

After modifying the tail geometry by removing the vertical stabilizer of the aircraft to create the tailless variant, the analysis of the estimated weight and balance was conducted using the same calculator as the tailed variant in Chapter 10.

Table 12.1 – Weight and balance calculator results for tailless variant

Component	Weight (lbs)	X – Location (ft)	Z – Location (ft)	X – MOI (ft*lbs)	Z – MOI (ft*lbs)
Wings	2162.4	19.7	6.9	42527.3	15007.1
Horizontal Stabilizer	212.9	35.0	7.2	7450.7	1526.3
Fuselage	1419.8	18.3	6.9	26030.0	9768.4
Main Landing Gear	501.1	26.3	0.0	13154.8	0.0
Nose Landing Gear	88.4	9.2	0.0	810.7	0.0
Engine Weight	1368.0	27.6	6.9	37734.0	9411.8
Engine Components Installation	547.2	27.6	6.9	15093.6	3764.7
Misc. Equipment	1371.1	18.3	6.9	25136.8	9460.6
Pilot	180	15.3	7.3	2760.0	1314.0
Fuel	5092.9	23	6.9	117137.4	35141.2
Empty Weight Allowance (10%)	767.1	21.9	6.4	16793.8	4893.9
Empty Weight CG	8438.1	**21.9**	**6.4**	184,731.7	53,832.2
Takeoff Gross Weight CG	13711	**22.2**	**6.6**	304,629.1	90288.0

By removing the vertical stabilizer, a minor shift in the Z- moments was to be expected and is represented in the above table. Notably, the aircraft can operate with less fuel due to the reduction of weight, further decreasing the overall weight in addition to the removal of geometry.

12.3 Tailless Longitudinal Stability Analysis

Longitudinal stability analysis was completed using the neutral point data from the modified XFLR5 model and the calculations from the same analysis for the tailed variant in Chapter 11. It is included in this section for further comparison of the changes caused by the removal of tail geometry and is seen below in Table 12.2.

Table 12.2 – Longitudinal stability values

Parameter	Value
X_{NP} (neutral point location) from XFLR5	26.480 ft
X_{CG_Fore}	21.8 ft
X_{CG_Aft}	22.2 ft
$\bar{c}$	9.844 ft (23.825 ft from x = 0 Datum)
Static $Margin_{Fore} = (X_{NP}-X_{CG\_Fore})/\bar{c} * 100\%$	19.64 %
Static $Margin_{Aft} = (X_{NP}-X_{CG\_AFt})/\bar{c} * 100\%$	17.96 %
X_0 distance from origin (reference point) to aircraft nose	0 ft
X_1 distance from aircraft nose to wing apex	13.981 ft
$\bar{x}$ distance from wing apex to the leading edge of the mean aerodynamic chord (MAC)	9.844 ft
X_{NP_MAC} distance from MAC leading edge to neutral point	-2.655 ft
$X_0 + X_1 + \bar{x} + X_{NP\_MAC}$ (Should match X_{NP})	26.480 ft
$X_{CG\_MAC\_Fore}$ distance from MAC leading edge to forward CG	2.025 ft
$X_{CG\_MAC\_Aft}$	1.625 ft
$NP\_MAC = XNP\_MAC/\bar{c} * 100\%$	11.144 %

Against previous expectations, the aircraft became more stable in the longitudinal axis when the additional weight was removed from the tail. This is due to a favorable shift in the aircraft's center of gravity from the requirement for differing amounts of fuel and a lighter landing gear using Raymer's assumptions [5], and a massively shifted neutral point location which resulted in the fore and aft static margins increasing by 76.4% and 108.4% respectively.

12.4 Tailless Weight and Balance Discussion

Removing the tail geometry has led to interesting results. First, by lowering the overall weight of the tail of the aircraft, components such as the main landing gear and nose gear could be lightened slightly, balancing out the expected shift in center of gravity resulting in minor shift towards the nose of 0.4 ft and 0.3 ft for the fore and aft cg's respectively. Accompanying some minor shifts to the Z-moments was a large aft shift of the aircraft's X-neutral point of 1.93 ft. This, in combination with the forward shift of both cg's, resulted in a massive increase to the longitudinal static stability of the aircraft, which was against initial expectations of the aircraft becoming more unstable overall. While this could impact the aircraft by resulting in less favorable results when lateral stability and Dutch-Roll are evaluated due to the lack of static stability contributed by a

traditional tail, it also opens the opportunity to add an additional 800 lbs worth of fuel or equipment during operation. As a result, both aircraft will be evaluated at using the longitudinal stability values of the base aircraft, since real life application would undoubtedly make use of the change in weight and optimize for near identical flight models in this regard.

With the removal of the tail geometry and associated weight calculations complete, it is time to move to evaluating the stability modes of the aircraft. Due to limitations encountered using XFLR5 when removing the vertical stabilizer, calculations have been shifted to MATLAB in the following section.

13. Stability and Control Analysis

13.1 Stability and Control Analysis Overview

Stability and control design is arguably the largest factor in aircraft design as it is tightly coupled with the flight dynamics of the aircraft in question. Changes in aircraft shape influence the natural stability, or instability, requiring additional control systems to be designed during creation. These control systems can range from simple movement of control surfaces to allowing for extreme maneuverability and control over conventionally unstable aircraft. In this section, the stability and control (S&C) derivatives will first be calculated within MATLAB for both variants of the aircraft, followed by an analysis and comparison for both aircraft in the longitudinal axis, as well as the lateral-directional axis. The base aircraft will be in standard configuration to serve as a foundation for comparison while the tailless variant will utilize both the designed drag rudders, in place of the normal rudder, and lateral thrust vectoring to attempt to make up for the lack of vertical control surfaces.

13.2 Stability and Control Analysis Limitations and Assumptions

As previously discussed in Section 12.4, XFLR5 was unable to produce stability characteristics for the tailless variant, and as a result, all stability and control derivatives were calculated analytically through MATLAB for both variants of the aircraft to ensure a proper baseline for comparison. This brings up some complications, as mentioned by the United States Air Force in the July 1982 *Background Information and User Guide for MIL-F-8785C, Military Specification – Flying Qualities of Piloted Airplanes*, in reference to the coupling forces involved with the lateral-directional derivatives and subsequently, the calculations of such, says “As yet, no one has found a truly simplifying principle for lateral-directional flying qualities requirements” [19, p.96].

With that said, lacking CFD simulation results will likely result in a large degree of error for the true accuracy of the analytically derived stability and control derivatives. However, since the main goal of this exercise is to compare a tailless variant to the traditional aircraft designed, so long as both sets of derivatives come from the same source and are calculated the same way, the simulation can be used to compare the outcomes.

Regarding assumptions, atmospheric conditions were taken from engineeringtoolbox.com [20] and the simulation was completed with the following conditions:

- Cruise Altitude = 30,000 ft
- Cruise Speed = 350 kts = 590.73 ft/s
- Angle of Attack (α) = 0 degrees
- Sideslip Angle (β) = 10 degrees
- Air Density (σ) = $8.91*10^{-4}$ slugs/ft^3
- Dynamic Viscosity (u_∞) = $3.107*10^{-7}$ slugs/(ft*s)
- Gravity (g) = 32.17 ft/s^2

All calculations can be seen at the end of this chapter in the provided MATLAB code.

13.3 Stability and Control Longitudinal Approximation Analysis

Longitudinal analysis of both aircraft monitors the dynamics of the model regarding the pitch and angle of attack during operation. Since all derivatives relate to forward thrust velocity, elevator deflection (δe), and AoA (α), both models are expected to perform similarly. This analysis is conducted for the sake of full dynamic analysis of both aircraft to ensure a comprehensive comparison between the models. Table 13.1 displays the base aircraft longitudinal S&C derivatives, while Table 13.2 shows the S&C derivatives of a Cessna T-37 [21], to ensure the derived constants are within a reasonable range.

Table 13.1 – Aircraft longitudinal stability and control derivatives

Longitudinal Derivates	**Value**
U_1	590.73 (ft/s)
X_u	-1.0410 (1/s)
X_α	32.1700 (ft/s^2)
$X_{\delta e}$	0.0000 (ft/s^2)
Z_u	-0.1089 (1/s)
Z_α	-445.0057 (ft/s^2)
$Z_{\delta e}$	-16.7579 (ft/s^2)
M_u	-0.3392 (1/s^2)
M_α	-23.7063 (1/s^2)
$M_{\dot{a}}$	-11.9032 (1/s)
M_q	-0.9996 (1/s)
$M_{\delta e}$	-20.3232 (1/s^2)

Table 13.2 – Cessna T-37 longitudinal stability and control derivatives [21]

Longitudinal Derivates	**Value**
U_1	516.53 (ft/s)
X_u	-0.0112 (1/s)
X_α	10.9335 (ft/s^2)
$X_{\delta e}$	0.0000 (ft/s^2)
Z_u	-0.1416 (1/s)
Z_α	-442.4657 (ft/s^2)
$Z_{\delta e}$	-42.7090 (ft/s^2)
M_u	0.0000 (1/s^2)
M_α	-19.4229 (1/s^2)
$M_{\dot{a}}$	-1.1566 (1/s)
M_q	-2.4797 (1/s)
$M_{\delta e}$	-31.0767 (1/s^2)

Seeing that most all the derivatives are within one decimal place of the AIAA published T-37 derivatives, the analytically calculated derivatives are considered to be accurate enough for continued evaluation. Section 13.3.1 covers the step response and bode plots for the short period and phugoid approximations of the systems.

13.3.1 Longitudinal System Approximate Analysis Comparison

An initial approximation of the response is a good way to check if the state space system created from the proposed longitudinal derivatives is a valid model to base the control system transfer function off. To show this, the pitch angle response is approximated regarding an elevator deflection, seen as a step response. The short period response approximation, seen as a dashed red line below in Figures 13.1 and 13.2, should track the full system, seen in blue, accurately within a short range of time and fall off as time goes on. Conversely, the long period approximation, or phugoid response, should not track near as accurately and is expected to see a 180-degree phase shift in the initial frequency response. Figure 13.1 displays the base aircraft approximation comparison, while Figure 13.2 shows the tailless variant response.

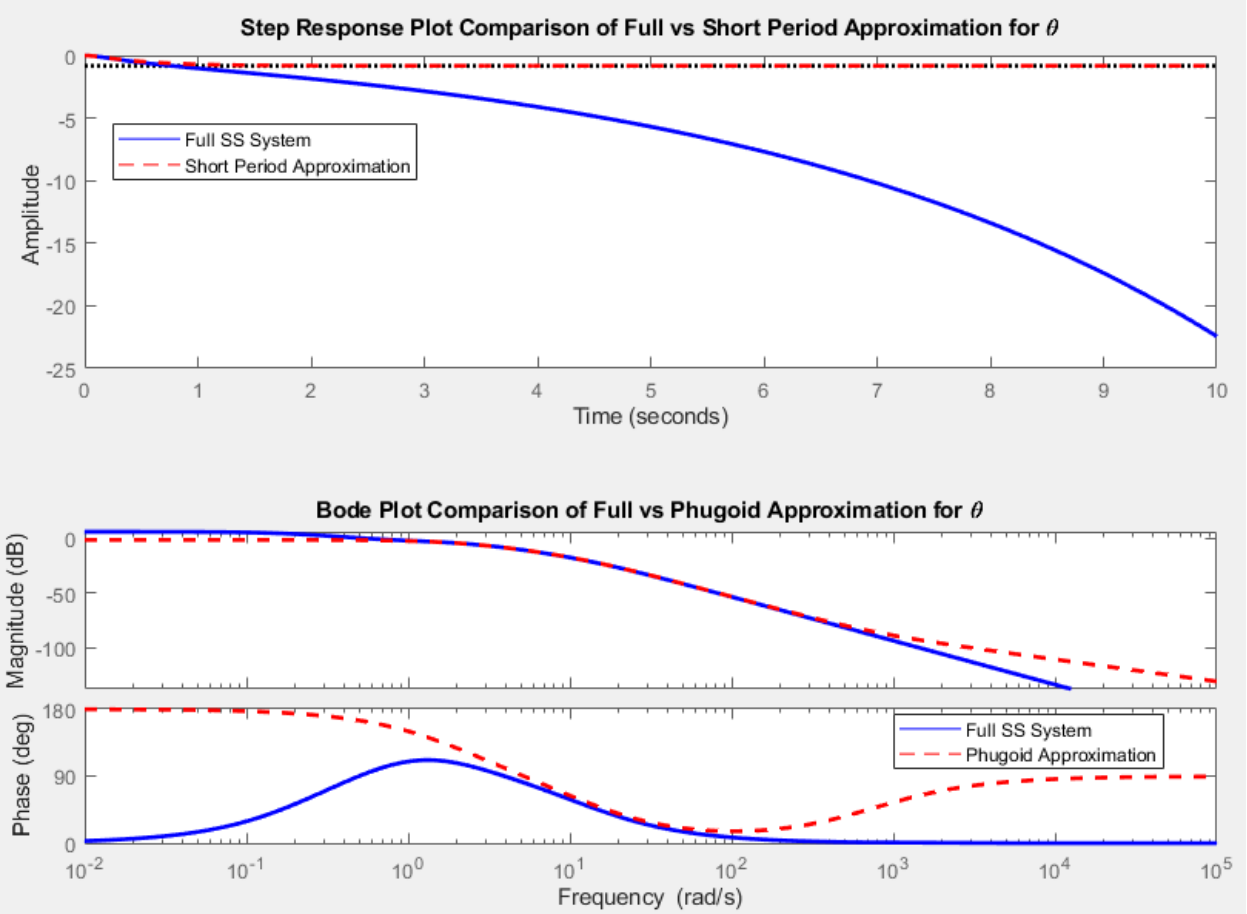


Figure 13.1 – Base aircraft longitudinal approximated response

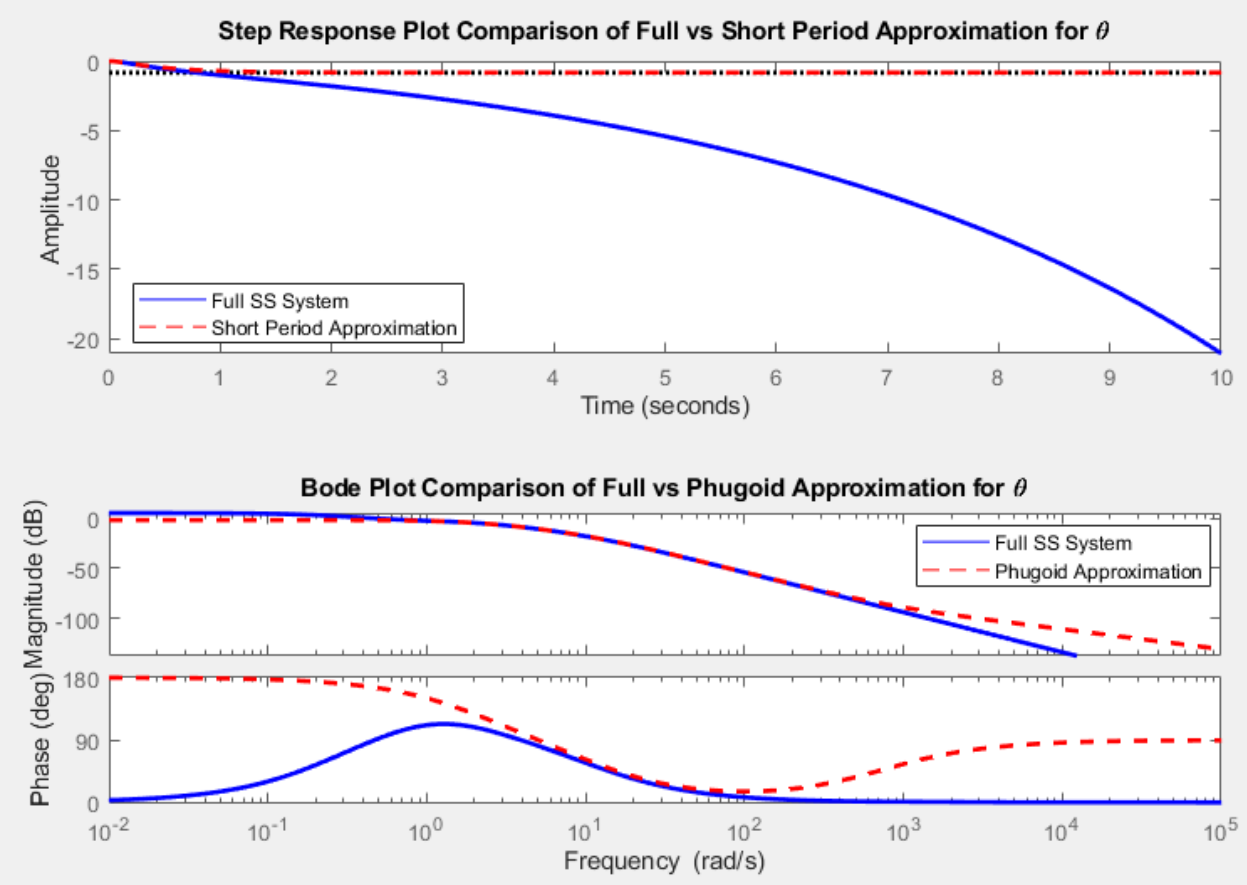


Figure 13.2 – Tailless aircraft longitudinal approximated response

As expected of the two models with identical longitudinal derivatives, the response is the same between the two. The short period approximation quickly diverges from the expected true system response and will require stabilization from the integrated control systems, while the phugoid response tracking closely in magnitude, particularly in the mid frequency steady state, and is phase shifted in the small and high frequency domains. With this approximation complete, a full model can be constructed and tuned for longitudinal stability and control analysis using MATLAB Simulink.

13.4 Stability and Control Lateral-Directional Approximation Analysis

Lateral-Directional stability analysis involves evaluating the response of the aircraft to a change, or perturbation, in the yaw and/or roll axis during flight. It is separated from the longitudinal analysis since the two states, yaw and roll, are coupled in their actions and reactions, meaning that a disturbance in yaw input induces a roll as well. In The three states are coupled together to a degree, however for initial analysis the longitudinal coupling is comparatively small enough to consider it irrelevant. Compared to the longitudinal derivatives, the two aircraft have differing values due to the change in geometry along the roll and yaw axis, shown in Tables 13.3 and 13.4 for the base and tailless variants respectively. Table 13.5 once again displays the Cessna T-37 S&C derivatives as a baseline to ensure the calculated derivatives of the model aircraft are in the realm of expected results.

Table 13.3 – Base aircraft lateral-directional stability and control derivatives

Lateral-Directional Derivates	Value
Theta	5 (deg)
L_p	-0.9869 (1/s)
L_β	-15.3688 (ft/s^2)
L_r	0.5483 (ft/s^2)
$L_{\delta r}$	2.9027 (1/s)
$L_{\delta a}$	5.2248 (ft/s^2)
Y_p	0.0000 (ft/s^2)
Y_β	-10.9274 ($1/s^2$)
Y_r	-0.0227 ($1/s^2$)
$Y_{\delta r}$	10.7211 (1/s)
$Y_{\delta a}$	0.0000 (ft/s^2)
N_p	-0.0029 ($1/s^2$)
N_β	1.5853 ($1/s^2$)
N_r	-0.3216 (1/s)
$N_{\delta r}$	-1.5069 ($1/s^2$)
$N_{\delta a}$	0.0000 ($1/s^2$)

Table 13.4 – Tailless aircraft lateral-directional stability and control derivatives

Lateral-Directional Derivates	Value
Theta	5 (deg)
L_p	-0.9910 (1/s)
L_β	-0.6519 (ft/s^2)
L_r	0.5505 (ft/s^2)
$L_{\delta r}$	2.9146 (1/s)
$L_{\delta a}$	5.2463 (ft/s^2)
Y_p	0.0000 (ft/s^2)
Y_β	-11.5562 ($1/s^2$)
Y_r	-0.0227 ($1/s^2$)
$Y_{\delta r}$	11.5424 (1/s)
$Y_{\delta a}$	0.0000 (ft/s^2)
N_p	-0.0030 ($1/s^2$)
N_β	0.0690 ($1/s^2$)
N_r	0.0000 (1/s)
$N_{\delta r}$	-0.0399 ($1/s^2$)
$N_{\delta a}$	0.0000 ($1/s^2$)

Table 13.5 – T-37 lateral-directional stability and control derivatives [21]

Lateral-Directional Derivates	Value
Theta	5 (deg)
L_p	-1.1693 (1/s)
L_β	-6.7383 (ft/s^2)
L_r	0.2450 (ft/s^2)
$L_{\delta r}$	1.0707 (1/s)
$L_{\delta a}$	12.9199 (ft/s^2)
Y_p	0.0000 (ft/s^2)
Y_β	-29.5547 ($1/s^2$)
Y_r	0.0000 ($1/s^2$)
$Y_{\delta r}$	17.0836 (1/s)
$Y_{\delta a}$	0.0000 (ft/s^2)
N_p	-0.0459 ($1/s^2$)
N_β	5.6418 ($1/s^2$)
N_r	-0.2628 (1/s)
$N_{\delta r}$	-1.8619 ($1/s^2$)
$N_{\delta a}$	-1.2957 ($1/s^2$)

Once again, both the base and tailless models are within an order of magnitude for most of the AIAA derivatives for the Cessna T-37, meaning further evaluation can be completed with relative certainty to the accuracy of the results. Between the two evaluated models, a notable difference can be seen involving the sideslip induced roll moment (L_β), and the yaw damping derivative from vertical geometry (N_r). With the lack of vertical geometry of the tailless model, the control system is going to need to do most of the work to make up for the lack of natural yaw damping, with the added benefit of a much smaller induced roll moment from any sideslip experienced.

Lastly, the base model does not make use of the yaw direction thrust vectoring and is relegated strictly to rudder deflection for yaw control. Since the tailless variant utilizes drag rudders and thrust vectoring in place of the rudder control, the derivative for calculating both $N_{\delta r}$ ($C_{N\delta r}$), and $Y_{\delta r}$ ($C_{Y\delta r}$), are calculated slightly differently. The changes in equations are displayed below in eq. 13.1 - 13.8 [11]. Equations 13.1 and 13.2 show the difference between the coefficients of yawing moment due to rudder deflection and drag rudder extension respectively, while 13.3 and 13.4 displays the displays coefficient of lift of the vertical stabilizer given a rudder deflection or drag rudder deflection. Equation 13.5 shows the base calculation for the coefficient of rudder induced side force Y compared to side forced from both drag rudder deflection and lateral thrust vectoring in 13.6 and 13.7. Finally, eq 13.8 shows the total Y force coefficient induced by the combination of 13.6 and 13.7 used in the calculation of $Y_{\delta r}$ for the tailless variant.

Since the rolling moment contribution from the rudder/drag rudders is considered to be negligible [11], a change in calculations between the two variants was deemed unnecessary and the same base equation was used to calculate the value in each aircraft. As a final note, all angles are calculated in radians to match dimensionless derivative formula requirements.

$$C_{N_{\delta r}} = -\frac{\theta_{\delta r} * q_v * (\frac{S_v}{S})}{S * b} \quad (13.1)$$

$$C_{N_{\delta dragrudder}} = \frac{b_{drag_r} * \bar{c}_{drag_r} * y_{drag_r}}{4 * S * b} \quad (13.2)$$

$$C_{L\alpha_v} = L_{\delta e} * (-\beta + \sigma_{sidewash}) * \theta_{\delta r} \quad (13.3)$$

$$C_{L\alpha_{dragrudder}} = L_{\delta e} * (-\beta + \sigma_{sidewash}) * \theta_{\delta drag_r} \quad (13.4)$$

$$C_{Y_{\delta r}} = C_{L\alpha_v} * \theta_{\delta r} * q_v * (\frac{S_v}{S}) \quad (13.5)$$

$$C_{Y_{\delta dragrudder}} = C_{L\alpha_{dragrudder}} * \theta_{\delta drag_r} * q_v * (\frac{S_{dragrudder}}{S}) \quad (13.6)$$

$$C_{Y_{ThrustVector}} = \frac{\cos\theta_{\delta drag_r} * Thrust * Engine\ Moment\ Arm}{q * S * b} \quad (13.7)$$

$$C_{Y_{\delta r}} = |C_{Y_{\delta dragrudder}}| + C_{Y_{ThrustVector}} \quad (13.8)$$

With the differences in equations in mind and derivatives calculated, the next following subsections cover modal analysis of the two models followed by the short period and phugoid approximation of the Dutch-roll mode before full system Simulink analysis in section 13.5.

13.4.1 MATLAB Modal Analysis of the Base Aircraft and Tailless Variant

The MATLAB Modal Analysis is completed using the state-space matrix of each variant by utilizing the *ss2tf* function to convert the state-space matrix to a transfer function which is then evaluated using the *damp(_)* function. The *damp(_)* function evaluates the poles of the denominator of the transfer function entered and displays them as a single output showing the poles, damping, frequency response (rad/s), and time constant (s).Table 13.5 displays the modal output for the base aircraft, and Table 13.6 shows the modal output for the tailless variant. Any poles that are shown to be negative are displaying positive stability attributes in the associated mode.

Table 13.6 – Modal output for the base aircraft

Pole	Damping	Frequency (rad/TimeUnit)	Time Constant (TimeUnit)
0.00e+00	-1.00e+00	0.00e+00	Inf
-1.06e-01 + 1.28e+00i	8.25e-02	1.29e+00	9.42e+00
-1.06e-01 - 1.28e+00i	8.25e-02	1.29e+00	9.42e+00
-1.08e+00	1.00e+00	1.08e+00	9.26e-01
-3.52e-02	1.00e+00	3.52e-02	2.84e+01

The above figure shows, from top to bottom, that the heading angle is a neutral pole, which is expected, the coupled Dutch-Roll mode is negative, followed by a stable roll mode, and lastly a stable spiral mode. With every mode showing stability, the base aircraft further proves itself as a solid ISR testbed.

Table 13.7 – Modal output for the tailless variant

Pole	Damping	Frequency (rad/TimeUnit)	Time Constant (TimeUnit)
0.00e+00	-1.00e+00	0.00e+00	Inf
-1.00e+00	1.00e+00	1.00e+00	9.99e-01
-8.18e-03 + 2.85e-01i	2.87e-02	2.85e-01	1.22e+02
-8.18e-03 - 2.85e-01i	2.87e-02	2.85e-01	1.22e+02
7.24e-03	-1.00e+00	7.24e-03	-1.38e+02

The tailless variant once again displays a zero-pole heading angle, stable roll mode, a Dutch-Roll mode that shows the poles are essentially zero meaning near instability, and the spiral mode is near zero in the positive axis, showing instability. These results were expected; however, they were initially expected to be considerably more unfavorable meaning that the system may require minimal tuning and will be done automatically through Simulink's auto PID tuning software in the following section.

13.4.2 Lateral-Directional System Approximate Analysis Comparison

After completing modal analysis of the two aircraft, a quick approximation of the Dutch-Roll of each aircraft is shown below. The approximation was carried out the same way as the longitudinal section, however a difference in results is expected given the varying stability characteristics shown in the modal analysis. Figures 13.3 and 13.4 display the Dutch-Roll approximations of the base aircraft and tailless variant respectively.

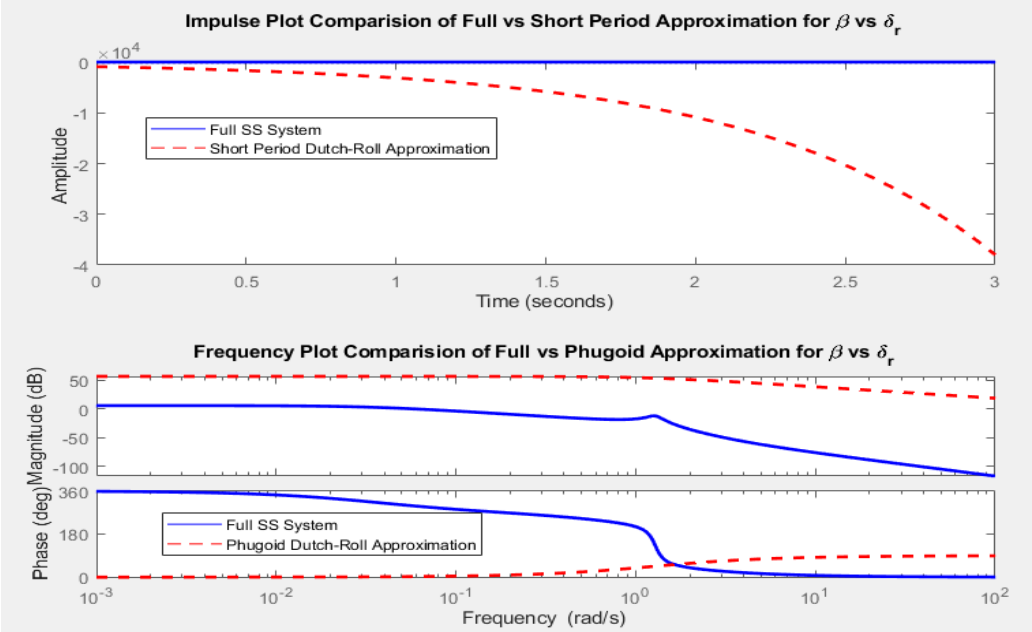


Figure 13.3 - Base aircraft Dutch-Roll approximated response

Like the longitudinal analysis, the short period quickly diverges but to an even greater extent. In the phugoid approximation, a considerable difference in magnitude is shown as well as a full 360-degree phase shift, meaning that it is not a good approximation and, in both instances, active control systems will be required to compensate.

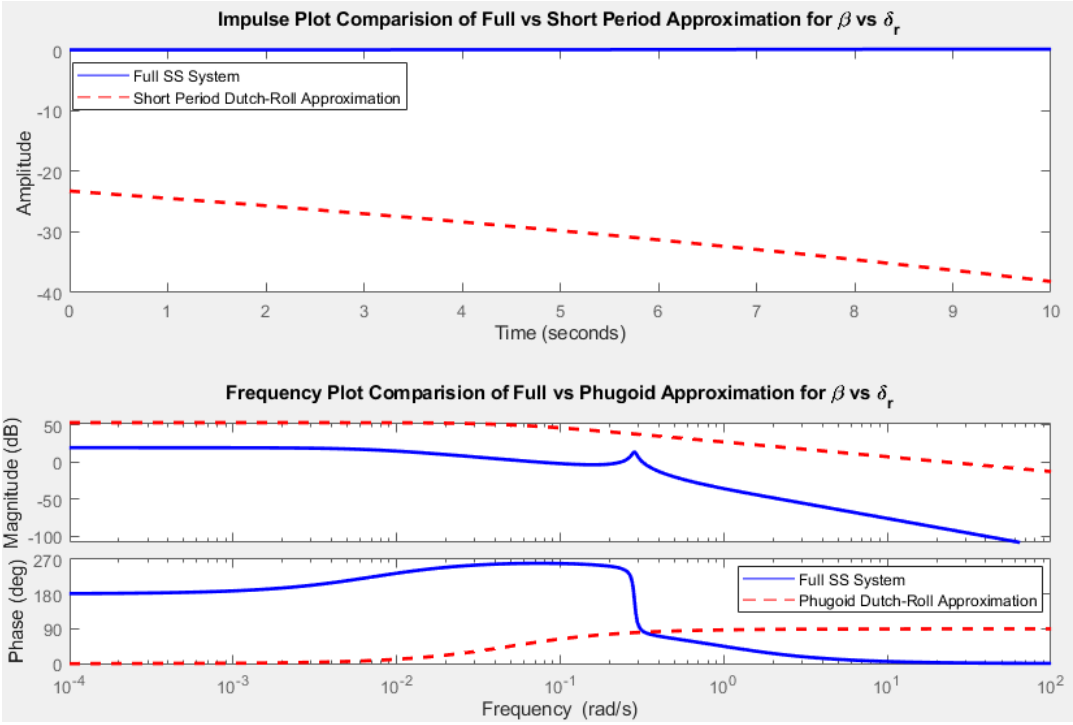


Figure 13.4 - Tailless variant Dutch-Roll approximated response

Once again the short period approximation does not track, and in fact never converges, while a similar lack tracking is shown in the phugoid response. This time however, the response is phase shifted the expected 180-degrees. Since the approximation continues to deviate drastically, active control systems will again be required. The following section, 13.5, displays and full Simulink simulation and features a comparison of the results and PID tuning values required for stability of each aircraft when experiencing a step input. Full MATLAB code is posted at the end of the report in Appendix C.

13.5 MATLAB Simulink Full System Stability Analysis and Comparison

With the models established and the approximation complete, a full system analysis involving pitch, roll, and yaw stability of both aircraft could be performed. All three axes of the aircraft were subjected to a step input of +5-degrees, holding position for two seconds, -10-degrees, holding position for two seconds, and finally returning to neutral with a final +5-degree input. This step input is shown below in Figure 13.5. From there, the system was tuned using the Simulink PID autotune function with the comparison results shown below in section 13.5.1.

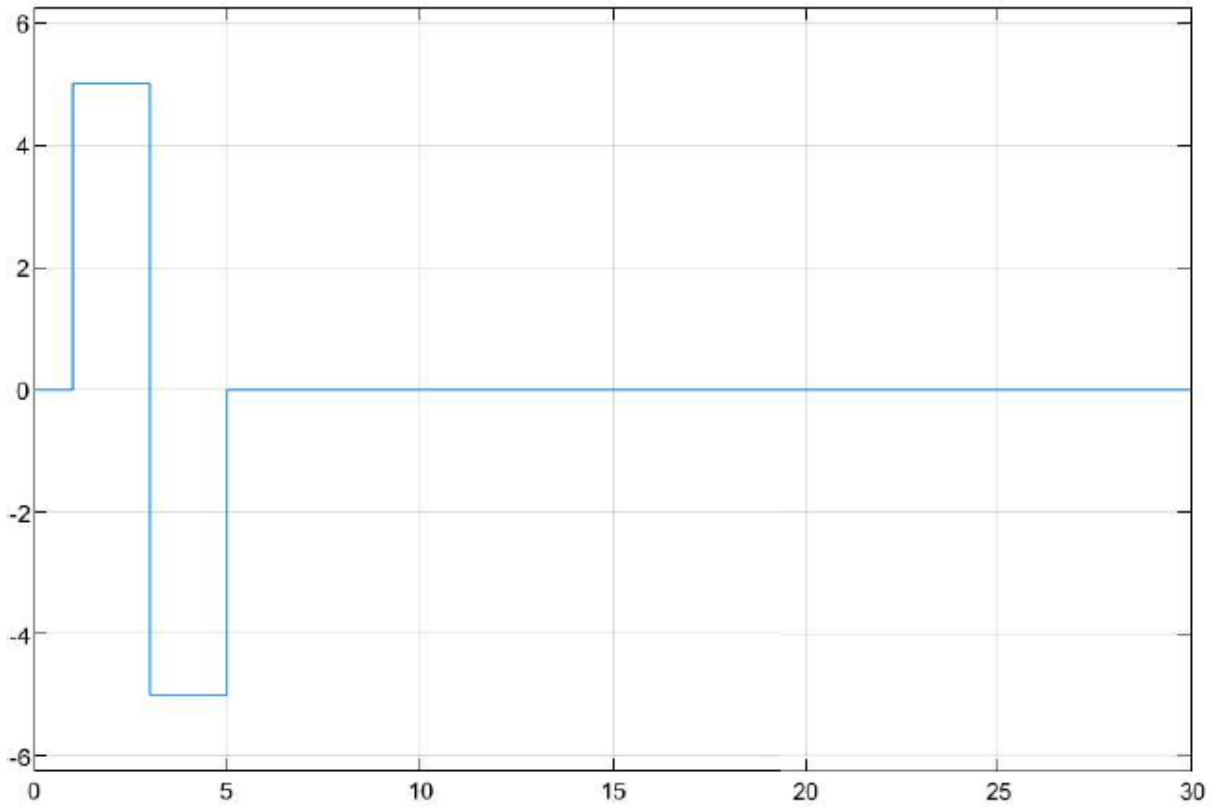


Figure 13.5 – Step input for Simulink simulation

The Simulink control systems used for stabilization are displayed following the MATLAB code used for the control systems in Appendix C.

13.5.1 Dual Model System Comparison

When comparing the two models, it is important to not only display the results, but also the way that they were achieved. In this case, tuning the PID controllers for each axis input based on the associated transfer function was the main method for successfully acquiring similar results. Tables 13.8-13.10 display the PID values for the base aircraft vs. the tailless variant regarding the elevators, ailerons, and rudders/drag rudder and thrust vectoring respectively. Actuator limits were set to 18 degrees for the elevators, and 20 degrees for the ailerons and rudders during simulation based on similar aircraft [21].

Table 13.8 – Elevator PID values for base aircraft (left) and tailless variant (right)

Proportional (P):	2.14	Proportional (P):	2.14
Integral (I):	3.70	Integral (I):	3.70
Derivative (D):	0.26	Derivative (D):	0.26
Filter coefficient (N):	335.11	Filter coefficient (N):	335.11

As expected with the pitch angle, the elevator PID values are identical since the longitudinal stability derivatives remain the same between both aircraft.

Table 13.9 – Aileron PID values for base aircraft (left) and tailless variant (right)

Proportional (P):	434.88	Proportional (P):	434.88
Integral (I):	13.09	Integral (I):	13.09
Derivative (D):	784.31	Derivative (D):	784.31
Filter coefficient (N):	417.31	Filter coefficient (N):	417.31

Interestingly, the aileron PID tuning values for the base aircraft proved very effective in the tailless variant as well, which is shown in the comparison graphs. While other much smaller values worked for the tailless variant as well, none returned a response that was as robust and quick as the base aircraft values.

Table 13.10 – Yaw-direction PID values for base aircraft (left) and tailless variant (right)

Proportional (P):	0.25	Proportional (P):	-.0065
Integral (I):	0.004	Integral (I):	-1.39e-06
Derivative (D):	3.233	Derivative (D):	-0.41
Filter coefficient (N):	0.678	Filter coefficient (N):	0.025

The PID tuning involving the yaw-axis is the first time seeing a difference between the two models. This is expected given the difference in geometry and natural stabilization that accompanies a vertical tail, as well as the introduction of both drag rudders and thrust vectoring as different forms of control surfaces on the tailless variant. The PID tuned system responses from the simulation are shown below in Figures 13.6-13.8.

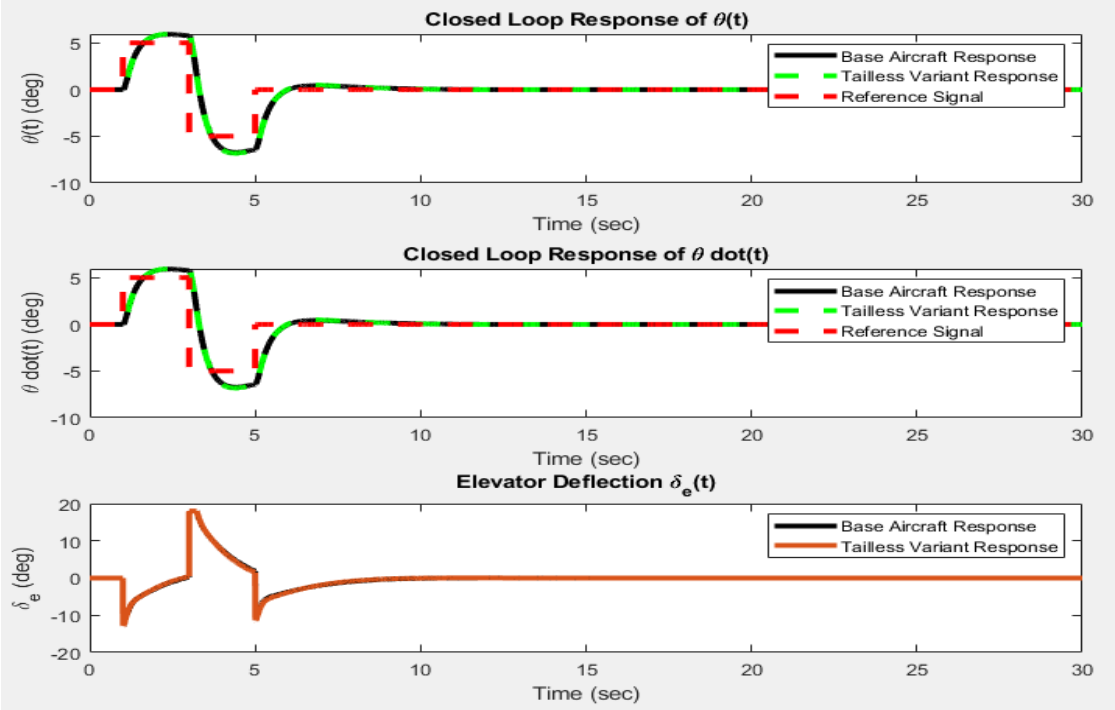


Figure 13.6 – Comparison of base aircraft and tailless model in pitch response

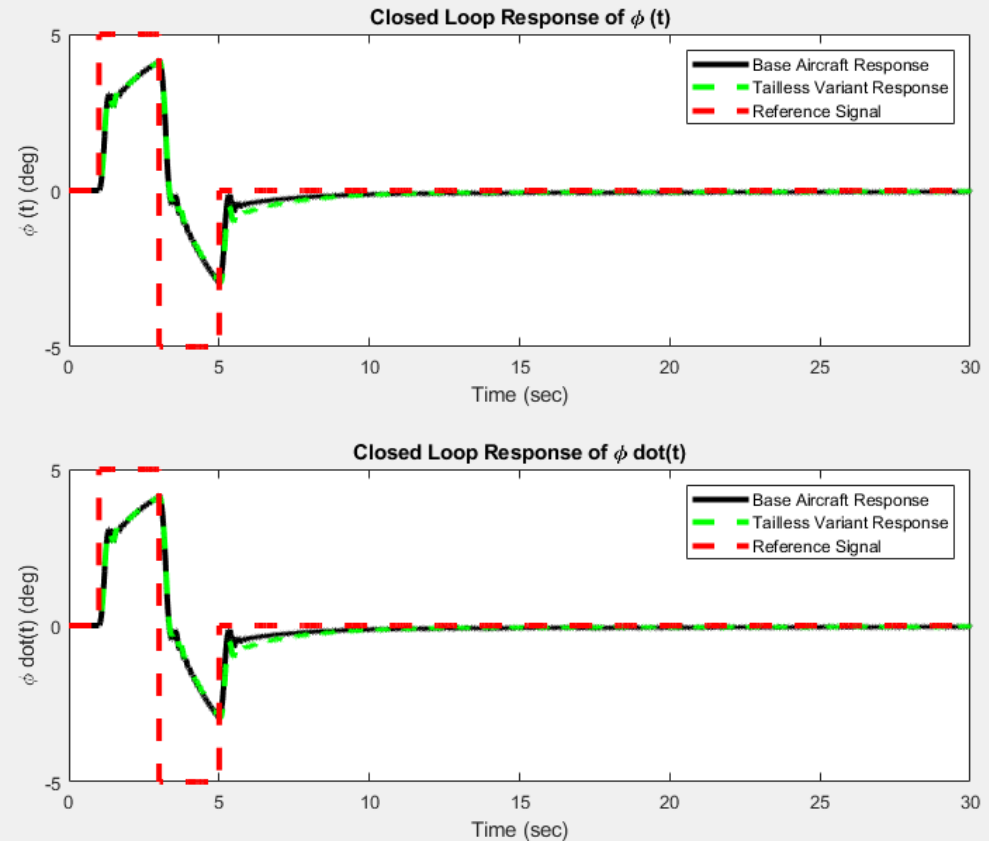


Figure 13.7 – Comparison of base aircraft and tailless model in roll response

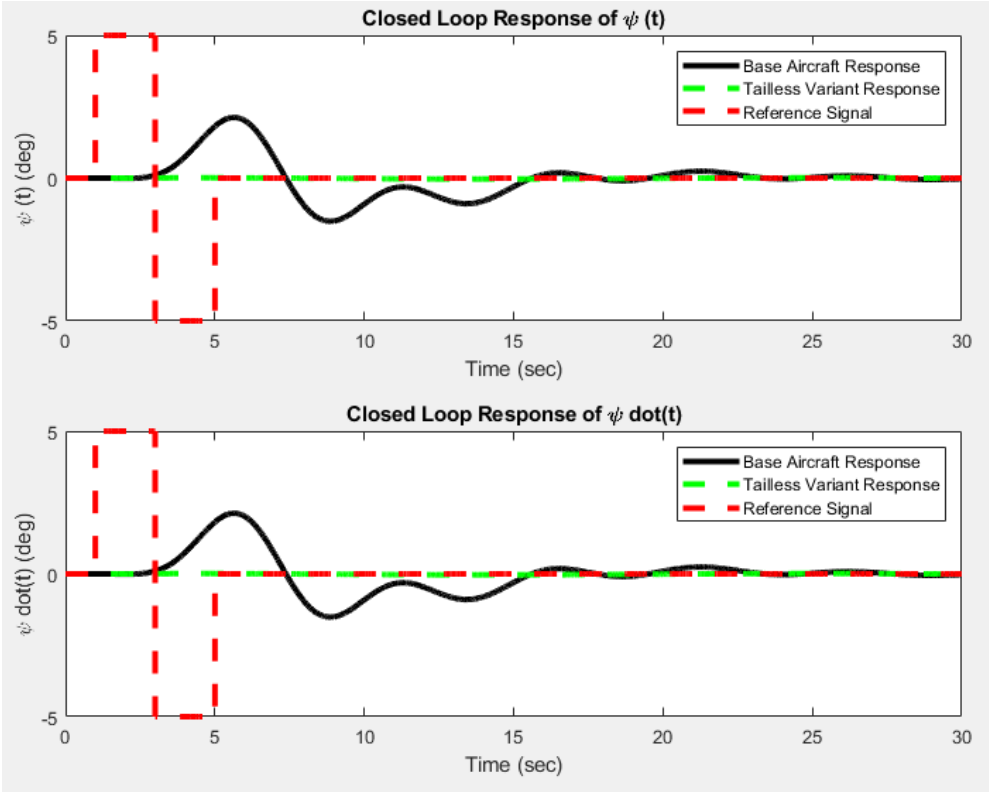


Figure 13.8 – Comparison of base aircraft and tailless model in yaw response

As displayed in Figure 13.6, both models respond identically in pitch response due to the same longitudinal derivatives and control tuning. Using the same PID controller values, the roll response in both aircraft can be tracked similarly, showing validity in these alternative controls in favor of removing the vertical geometry. Due to the nature of the hand calculated S&C derivatives regarding side force impact to the geometry, the tailless variant saw little disturbance regarding heading direction from the sideslip angle. This is because the main calculations involving side force impact from sideslip involve the geometry of the vertical stabilizer. To prove that the heading angle can be appropriately controlled, an additional simulation was run using the ailerons to conduct a "coordinated turn maneuver" [22]. This is done using equation 13.9 below.

$$\tan\theta = \frac{\dot{\Psi}}{g_D}\frac{V_T}{\cos\theta} \tag{13.9}$$

It is noted from this equation that a yaw-rate command, to be used as a heading angle coordinator, cand be measured as a function of roll angle while including a pitch-rate function within a Simulink closed loop control system [22]. This is done by adding an additional loop to the control system to record roll angle and feeding that back into an error block after a yaw angle gain input. Finally, by integrating the roll output signal as a function of gravity and velocity, a heading angle Ψ is produced. Figure 13.9 below shows the transfer function provided in *Aircraft Control and Simulation: Dynamics, Control Systems, and Autonomous Systems* [22].

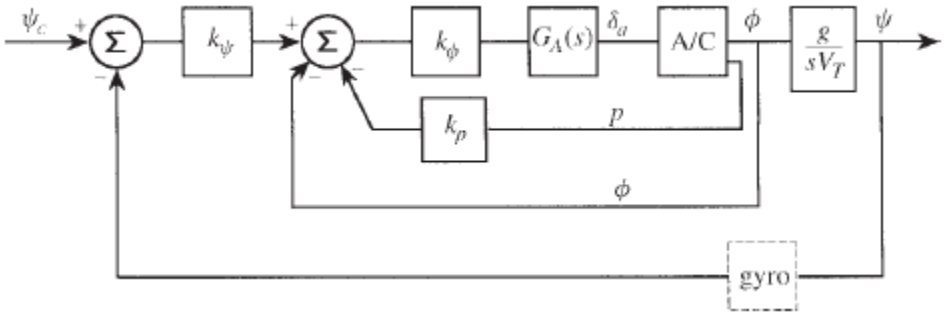


Figure 13.9 – Coordinated turn maneuver autopilot example [22]

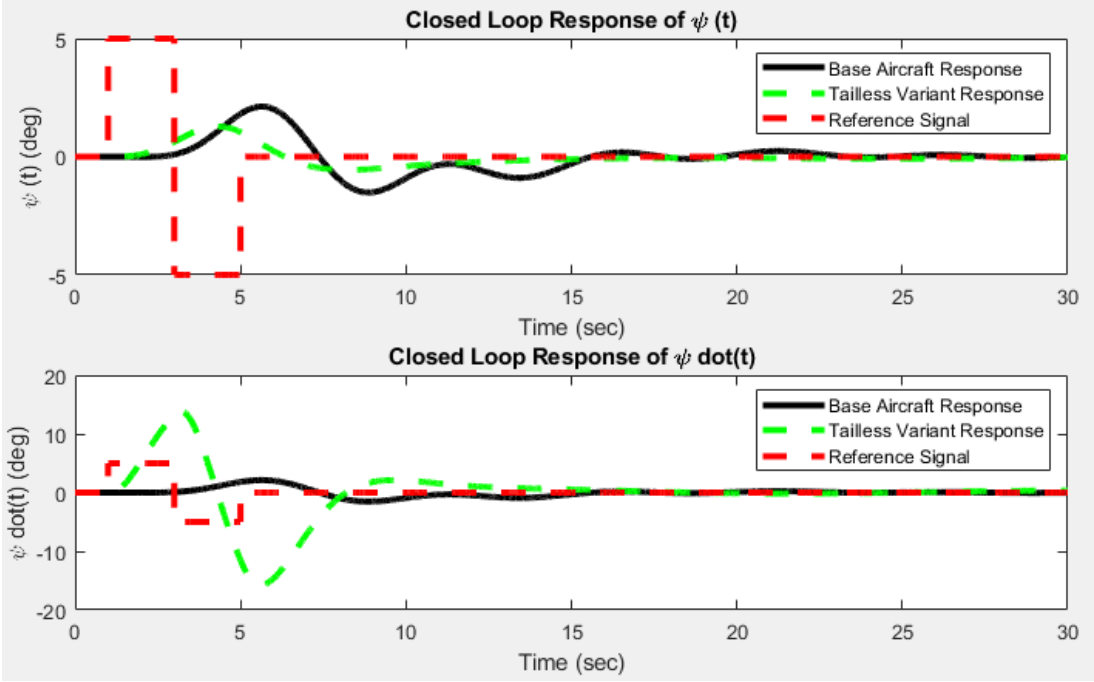


Figure 13.10 – Comparison of base aircraft and tailless model in induced yaw response

Figure 13.11 below displays the comparison of PID input values for the final simulation comparison.

Table 13.11 – Heading direction PID values for base aircraft (left) and tailless variant (right)

Proportional (P):	0.25	Proportional (P):	0.04
Integral (I):	0.004	Integral (I):	0.0014
Derivative (D):	3.233	Derivative (D):	0.16
Filter coefficient (N):	0.678	Filter coefficient (N):	0.77

When compared to the values for the first yaw stability PID comparison in Figure 13.10, the tailless values are much closer to the tailed variant, with the exception being the derivative off by a factor of 20. This final analysis concludes the study of the S&C comparison.

13.6 Stability and Control Analysis Overview

After careful analysis of both aircraft, there is clear potential in removing vertical geometry and replacing the rudder with alternative controls such as drag rudders and lateral thrust vectoring. In real life application this of course comes with the downside of additional actuators in the wings, and the addition of 3D thrust vectoring nozzles, which bring additional complications regarding upkeep costs and overall maintenance, however these downsides may be overshadowed by the additionally benefits of both drag and radar cross section reduction. RCS evaluation and comparison is conducted in the following section to complete the overall analysis and validity of the benefits of removing vertical geometry.

14. Radar Cross Section Evaluation

14.1 Radar Cross Section Evaluation Overview

As technology and warfare advance, new methods of detection and avoidance are constantly evolving to give any side an upper hand when approaching a conflict. In the modern era, reduction of detection is vital to ensure effective operation of a given mission in enemy territory, and one of the most important aspects of this for aircraft is lowering the radar cross section (RCS). This can be done in several ways such as radar absorbing coatings, changes in construction material, and the focus of this study, removing or altering geometry of the platform.

To evaluate the RCS of the aircraft accurately and affordably in this report, the MATLAB plugin POFacets, short for Photo-Optical Facets, created by Dr. David Jenn of the Naval Post Graduate School in Monterey California, is used after importing the CAD model into MATLAB in .stl format. Once the model is imported, monostatic (single source and receiver) and bistatic (different source and receiver) radar evaluations can be run on each version of the aircraft and compared to analyze the practical benefits of modifying the geometry. The standard coordinate system used for the RCS evaluation is pictured below in Figure 14.1.

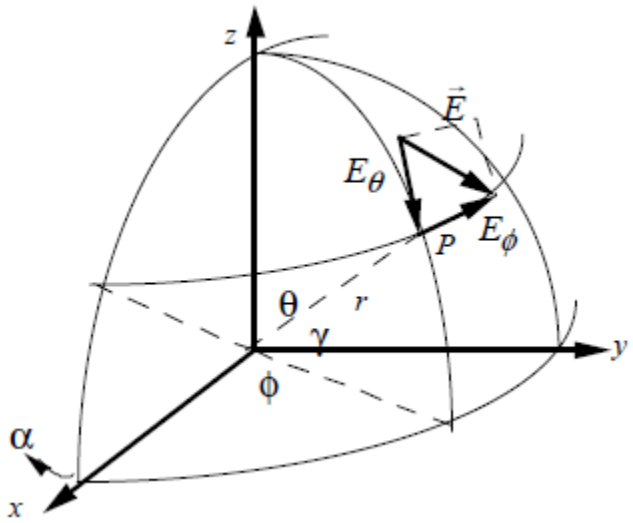


Figure 14.1 – POFacets coordinate system [26]

14.2 Radar Cross Section Evaluations Theory and Simulation

RCS analysis has been the focus of study for aircraft since the post World War 2 era and involves a few components relating the power of the radar, power returned, and geometry of the aircraft at its core. POFacets utilizes the radar equation to "describe the performance of a radar for a given set of operation, environmental, and target parameters" [26]. To effectively measure the RCS of the aircraft in multiple situations where it may be detected, evaluation is broken up into two different types of radar testing; monostatic, where the radar receiver and transmitter are located on the same platform or location, and bistatic radar in which the transmitter and receiver are in different positions. Monostatic radar evaluation, pictured below in Figure 14.2, will simulate an

enemy aircraft detecting the proposed model and bistatic radar evaluation, Figure 14.3, will simulate the aircraft being detected from a ground array [26].

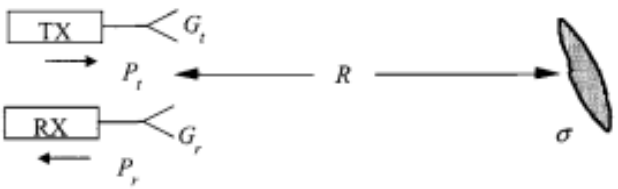


Figure 14.2 – Monostatic radar example [18]

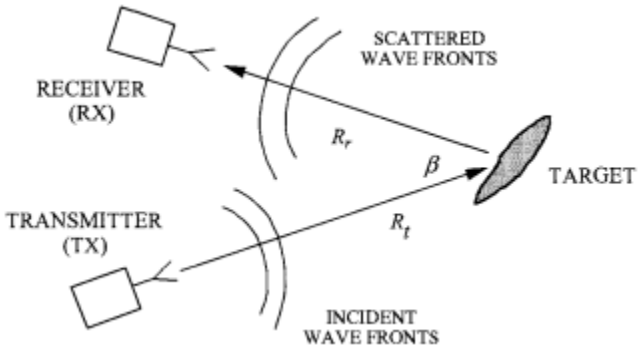


Figure 14.3 – Bistatic radar example [25]

The base equations involved in RCS evaluation involve the relation of RCS (σ), described in Equations 14.1, and displayed 14.2 involving range to target from radar station (R), incident electric field amplitude (E_i), and scattered electric field amplitude (E_s), and the radar range equation (RRE).

$$\boldsymbol{\sigma = \frac{Power\ reflected\ to\ receiver/Unit\ solid\ angle}{Incident\ power\ density/4p}} \tag{14.1}$$

$$\boldsymbol{\sigma = \lim_{R\to\infty} 4\pi R^2 \frac{|\overrightarrow{E_s}|^2}{|\overrightarrow{E_i}|^2}} \tag{14.2}$$

The RRE begins with a comparison of the power transmitted to the target, the power scattered back to the receiver (incident power density), antenna gain, and range of the target from the receiver and is shown below when related to RCS in Equation 14.3.

$$\boldsymbol{W_s = \frac{P_t G_t \sigma}{(4\pi R^2)}} \tag{14.3}$$

When further related to the area of reflection (A_{er}), a function of area multiplied by a reflection efficiency (e), gain relationship related to A_{er} and the receiver antenna, wavelength λ, potential signal loss (L), and lastly processing gain of the receiver (G_p) gives equation 14.4 for the modern form of the RRE [26].

$$\boldsymbol{P_r = \frac{P_t G_t G_r \sigma \lambda^2 L G_p}{(4\pi)^3 R^4}} \tag{14.3}$$

This equation is fundamental in basic evaluation of the RCS used in the POFacets plugin and uses the assumption that $0 \leq L \leq 1$ and $G_p \geq 1$ depending on radar type and initial conditions. Lastly, both aircraft are tested assuming standard aircraft aluminum construction with no additional coatings to assist in RCS reduction, as the study is interested in the RCS delta between the two aircraft assuming identical material construction. With these definitions in place, the simulation could be set up for comparison.

14.3 Radar Cross Section Evaluation Results

14.3.1 Base Example of Monostatic Evaluation

To set a baseline to identify how different shapes respond to monostatic radar evaluation, a sphere with radius 20 ft and cube consisting of 40 ft long sides, both to approximately match the wingspan of the aircraft, were tested at a frequency of 10 GHz varying angle theta 360 degrees around each shape in one-degree increments.

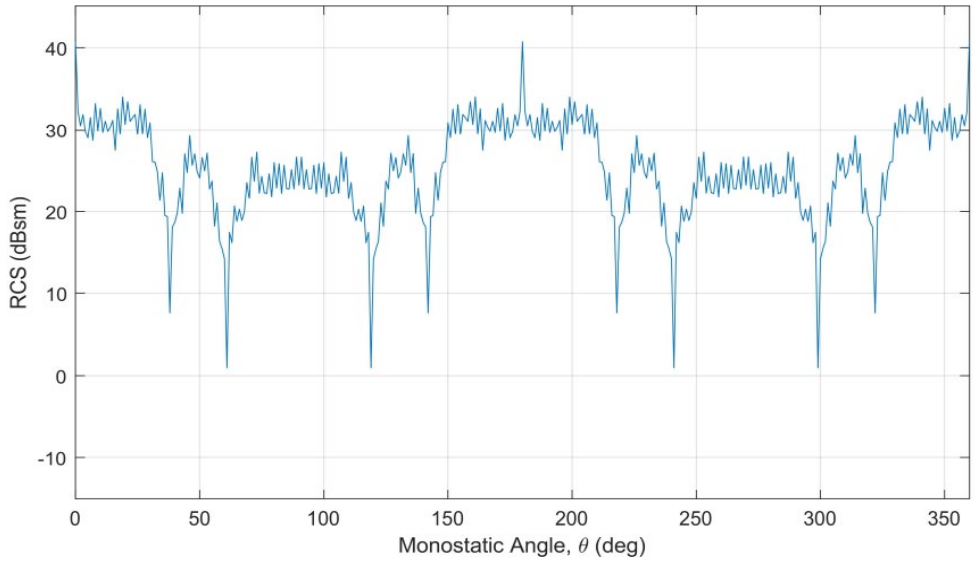


Figure 14.4 – Sphere linear monostatic RCS diagram

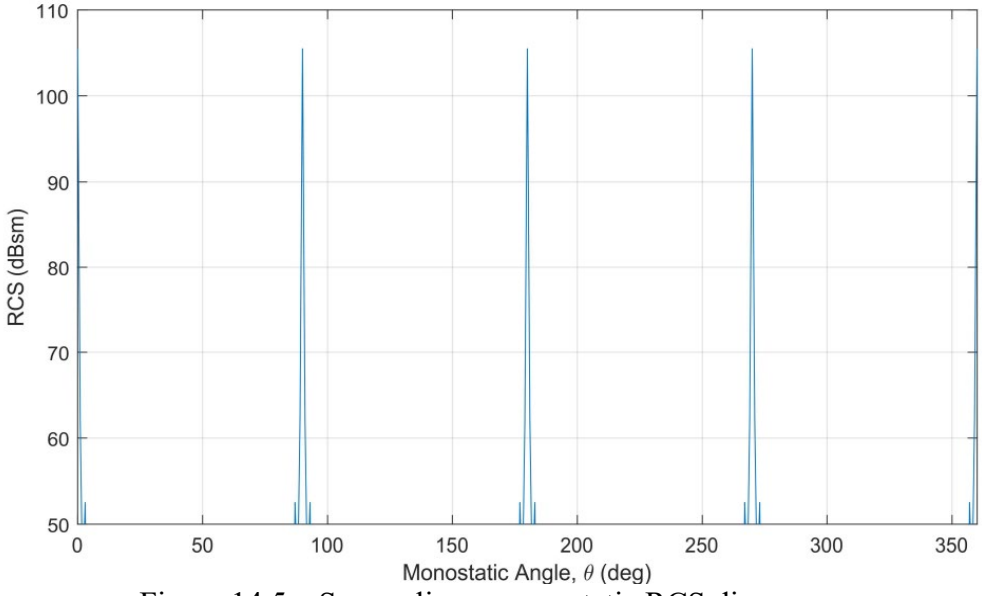


Figure 14.5 – Square linear monostatic RCS diagram

Looking at the two figures, the sphere emits a constant and mostly consistent radar response with small dips and peaks as the radar interacts with each facet, which is expected since the model is composed of many small flat plates. Conversely, the square displays extremely sharp returns when each face is seen head on, and near zero at any other angle as the radar is deflected away from the original source. The two show the importance of geometry when in a scenario where the signal is sent and received by the same source, such as an incoming aircraft. Figure 14.6 below shows the results in a polar plot configuration, followed by Figure 14.7 displaying the associated RCS on the geometry of each model.

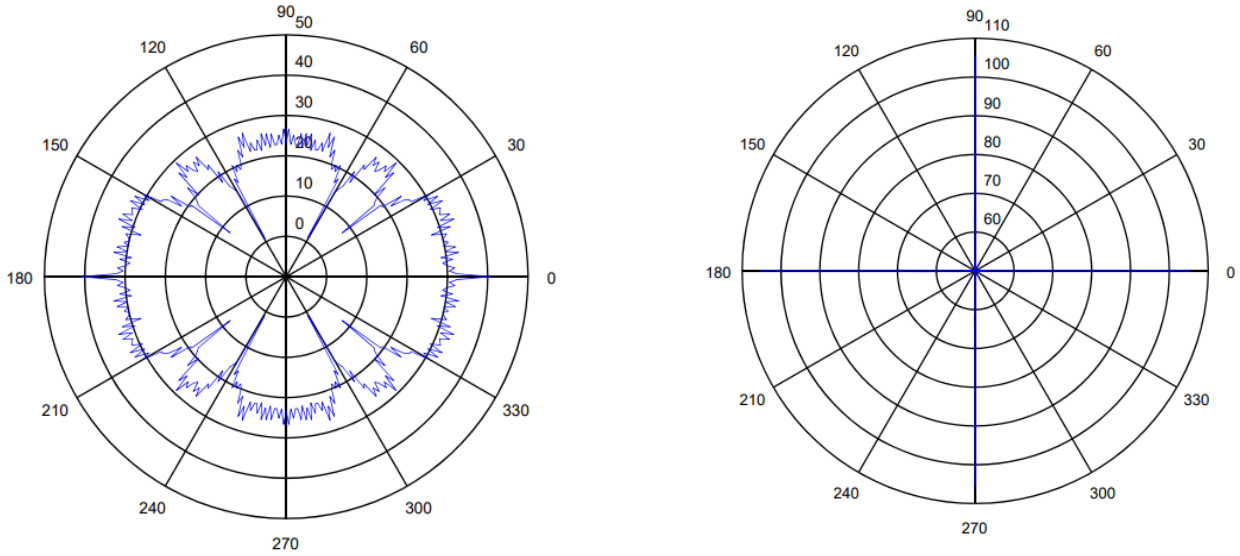


Figure 14.6 – Polar RCS plots of the sphere (left) and square (right)

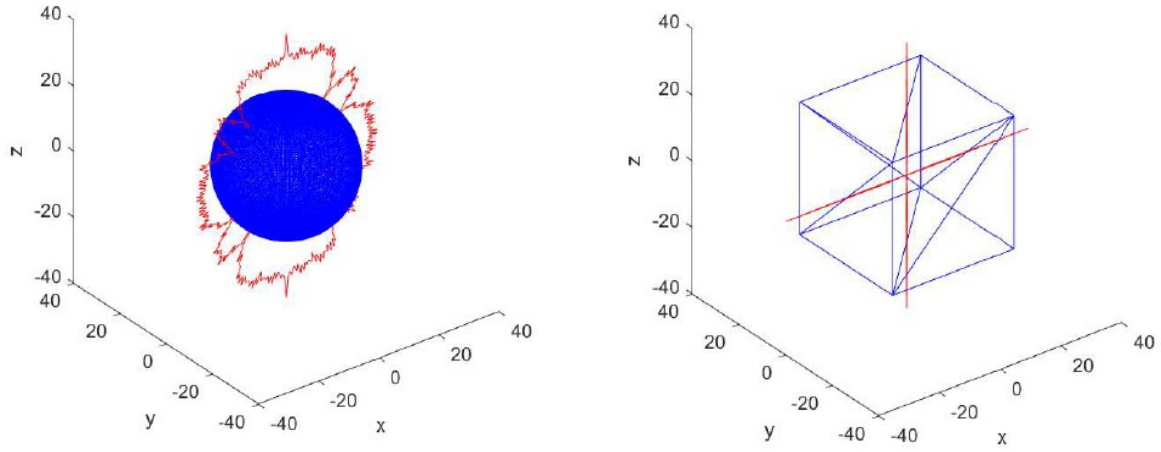


Figure 14.7 – 3D RCS plots of the sphere (left) and square (right)

14.3.2 Aircraft Monostatic Evaluation

Monostatic evaluation was conducted at a frequency of 10 GHz varying angle theta 360 degrees around the entire aircraft in one-degree increments.

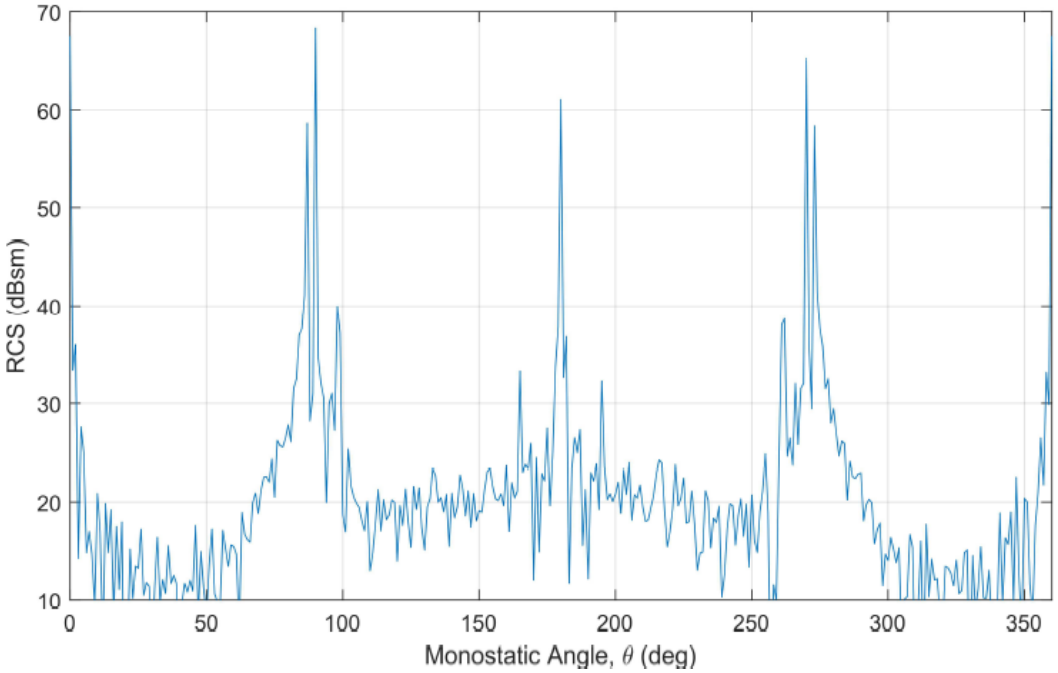


Figure 14.8 – Tailed variant linear RCS diagram

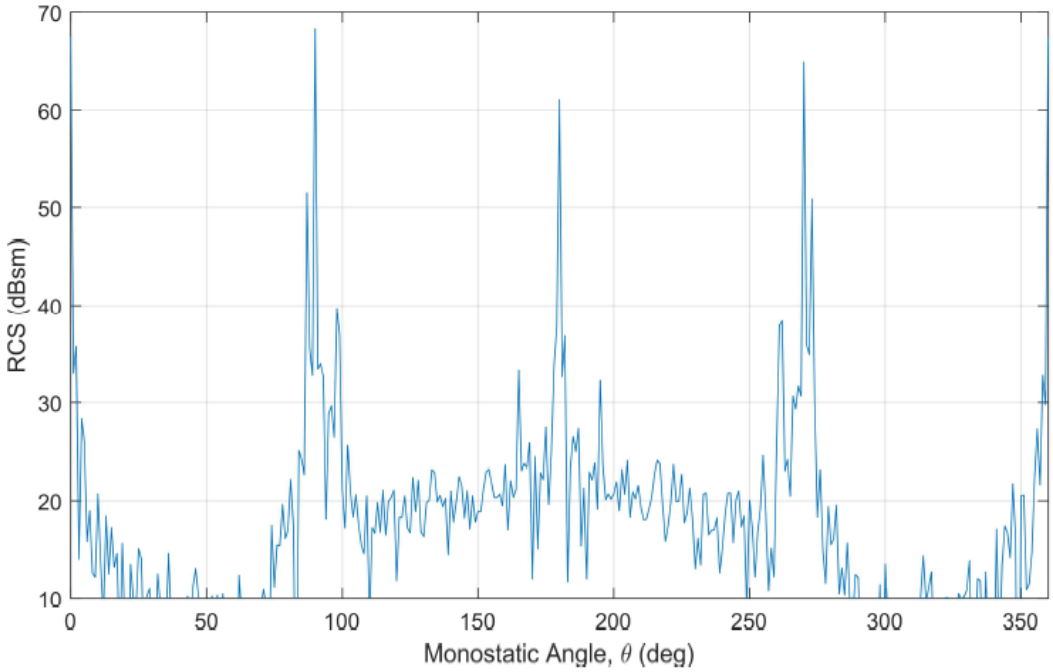


Figure 14.9 – Tailless variant linear RCS diagram

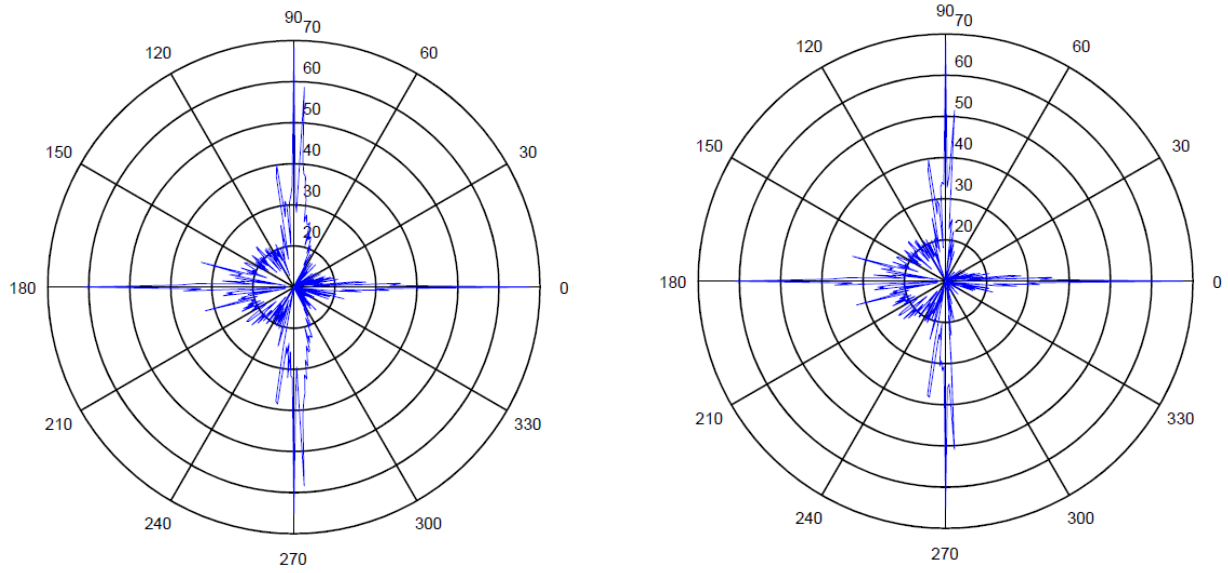


Figure 14.10 – Polar RCS plot comparison of tailed variant (left) and tailless variant (right)

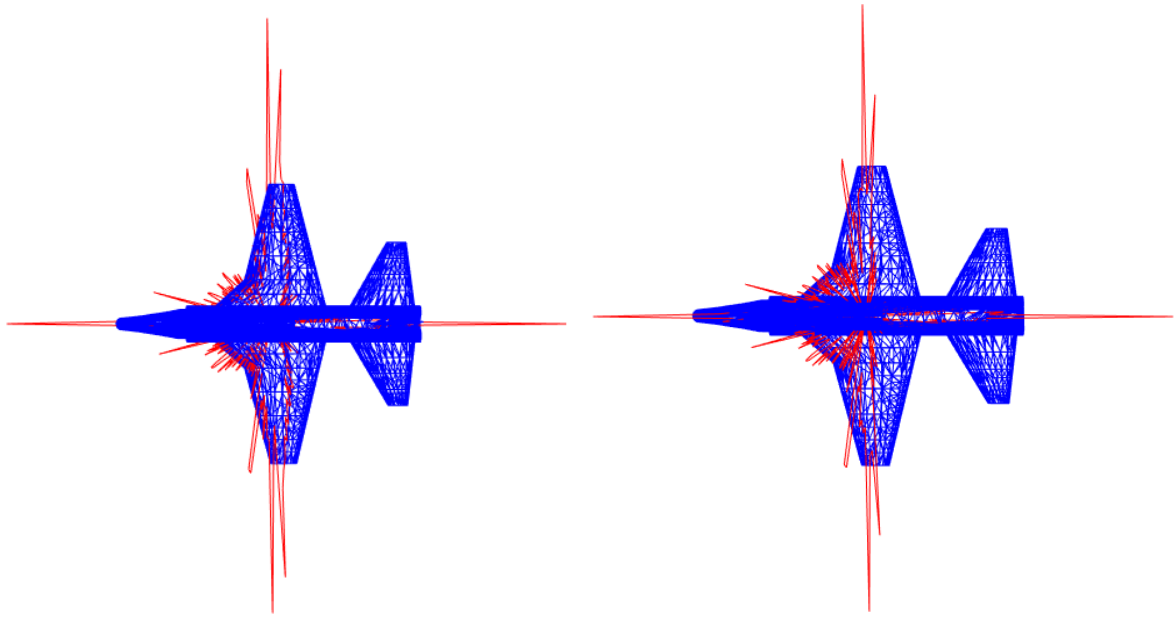


Figure 14.11 – RCS displayed over tested model for tailed variant (left) and tailless variant (right)

A noticeable reduction in RCS return signal can be identified when removing the tail geometry of the aircraft during monostatic evaluation, particularly when the aircraft is viewed from the side at an angle between the orthogonal and tail. *Note: There is minor asymmetry near the nose of the aircraft resulting in slightly different RCS response on the longitudinal axis.*

14.3.3 Base Example of Bistatic Evaluation

Once again, a baseline example of the bistatic RCS evaluation is shown on the sphere and cube alike. The example is conducted at the same frequency of 10 GHz. Due to the symmetry of the test geometry, the ground plane is considered irrelevant for baseline testing.

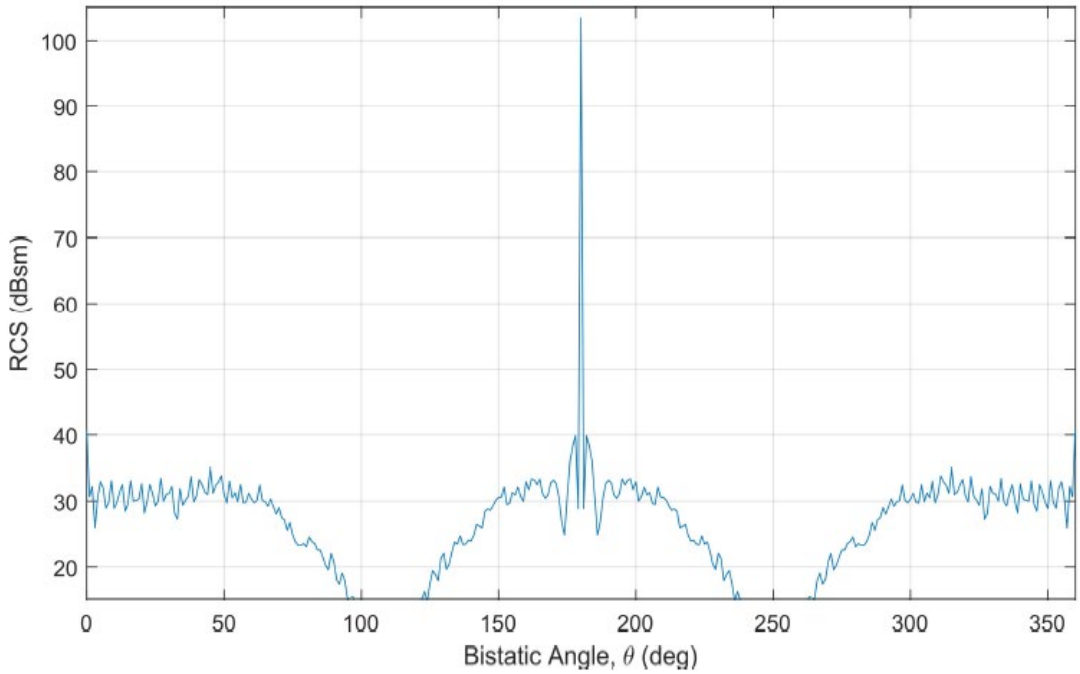


Figure 14.12 – Sphere linear bistatic RCS diagram

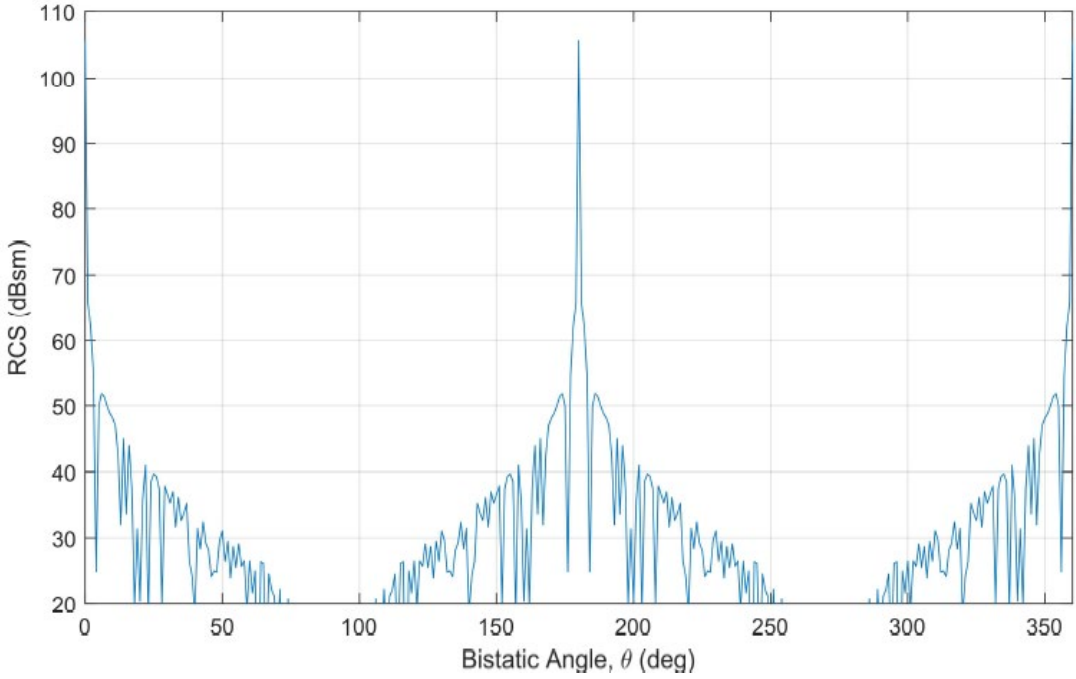


Figure 14.13 – Square linear bistatic RCS diagram

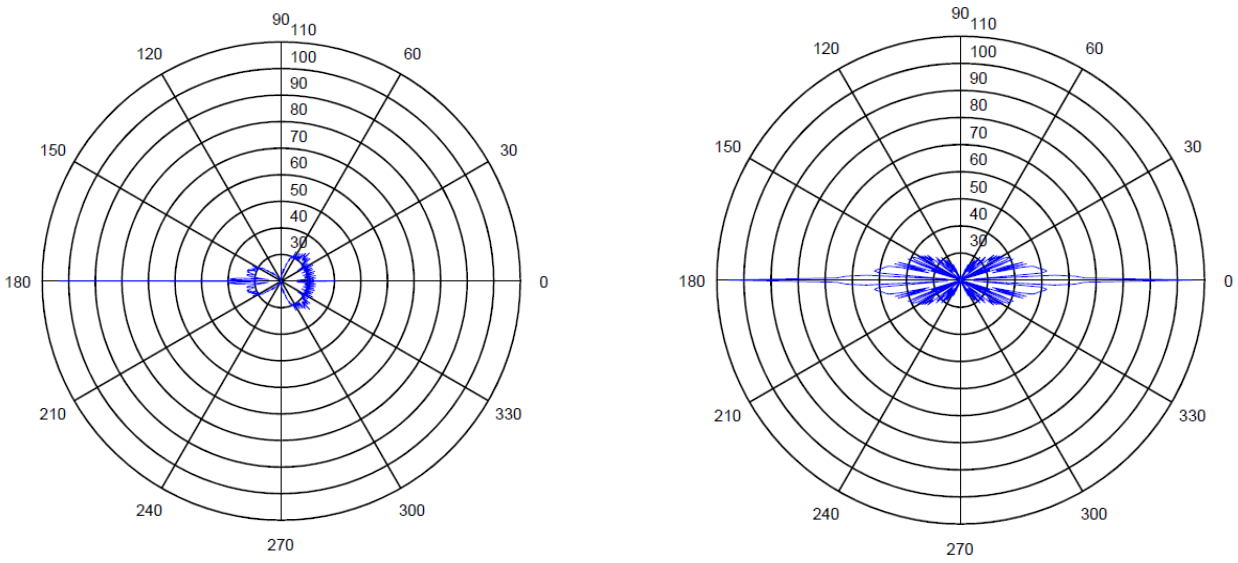


Figure 14.14 - Polar bistatic RCS plots of the sphere (left) and square (right)

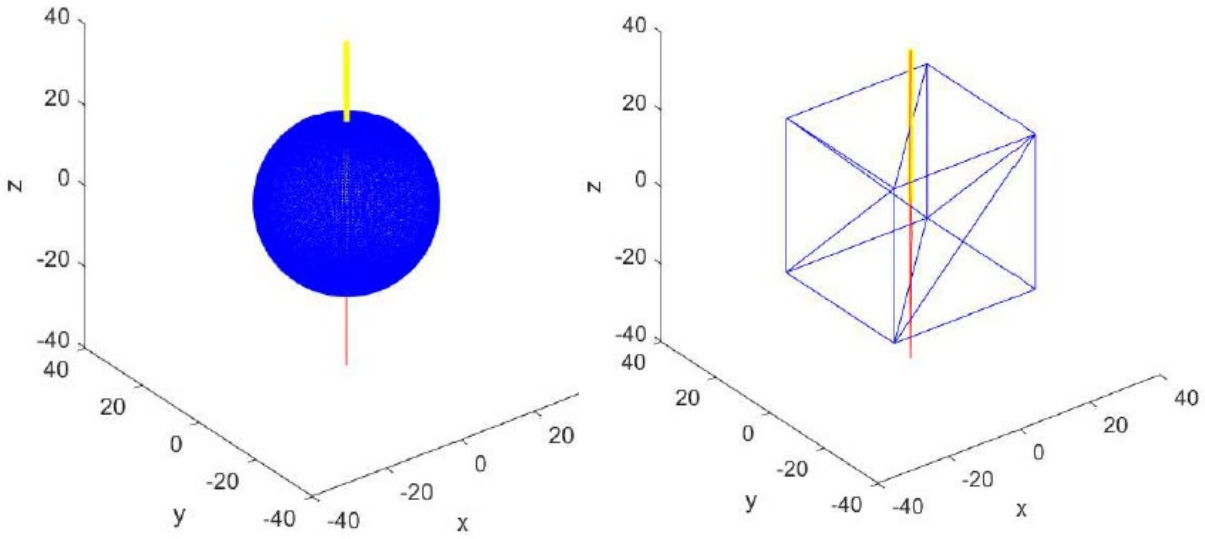


Figure 14.15 – 3D bistatic RCS plots of the sphere (left) and square (right)

The bistatic analysis baseline for both objects shows a stark contrast in results from the monostatic analysis, especially regarding the sphere. This is because rather than a single source both emitting and receiving the signal, one source sends the signal while another source receives it. When the radar is deflected at an angle off the cube in this case, the receiver can capture some parts of the signal, rather than the monostatic case where it is completely deflected away. With these baseline comparisons displayed, it is time to move to the aircraft analysis.

14.3.4 Aircraft Bistatic Evaluation

Like the baseline example, bistatic RCS evaluation was conducted at a frequency of 10 GHz by angling angle theta from -89 degrees to +89 degrees due to limitations on radar and receiver maximum location placement. Additionally, the aircraft is offset by 45 degrees along the Y-Axis to represent banking during a simulated reconnaissance mission flight. Linear plot results demonstrate the same test at an additional incident angle phi of 45 degrees, represented as the dashed line, which are omitted from the polar and 3D plots for readability. Controller gain is increased by 50% to more easily show differences that were otherwise zeroed out by the plot.

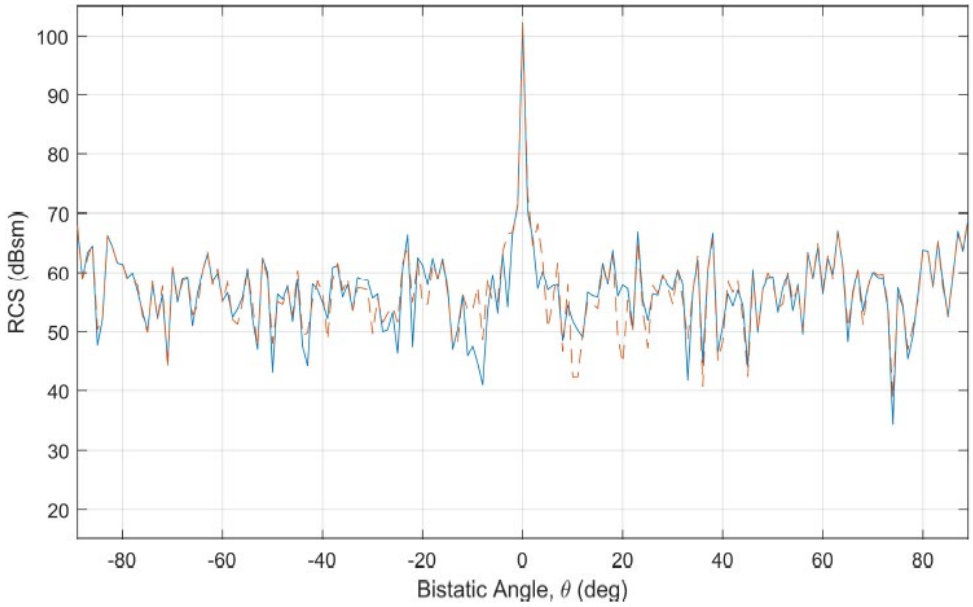


Figure 14.16 – Tailed variant linear bistatic RCS plot

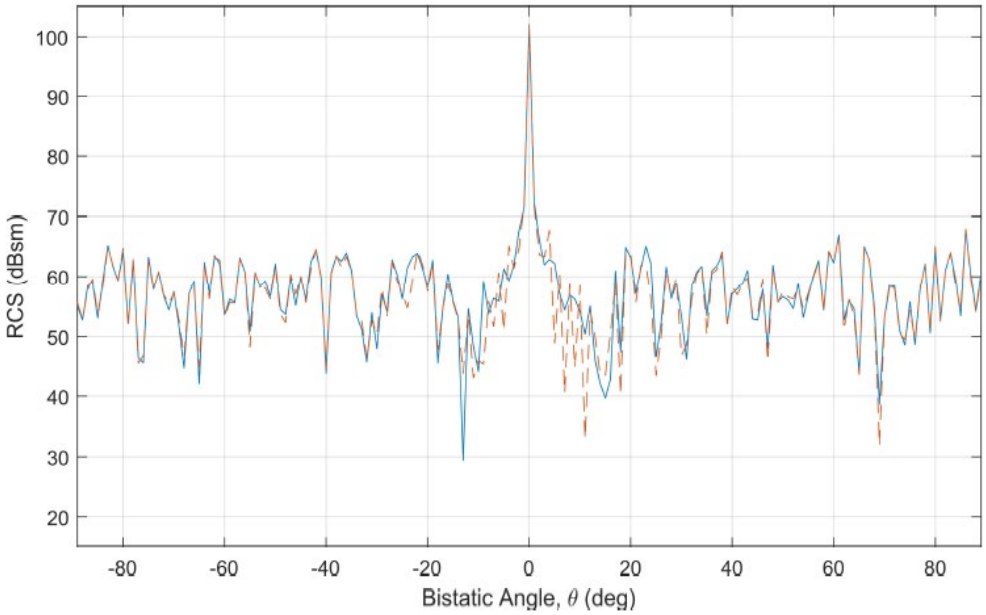


Figure 14.17 – Tailless variant linear bistatic RCS plot

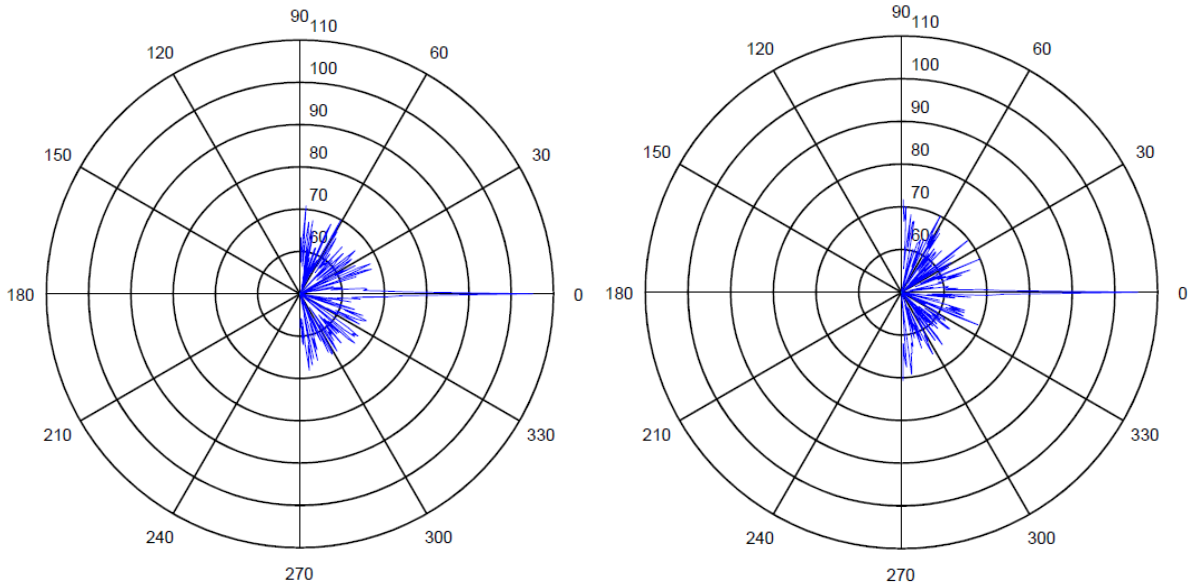


Figure 14.18 – Bistatic RCS polar plots for tailed variant (left) and tailless variant (right)

Figure 14.19 – 3D diagram for bistatic RCS of tailed variant (left) and tailless variant (right)

Like the monostatic results, a reduction is shown in the linear response of the tailless variant compared to the stock aircraft, however on a smaller scale. In particular, the aircraft has larger drops in detectability between 0 and 20 degrees, depending on incident angle, but is otherwise nearly comparable to the stock variant.

14.4 Radar Cross Section Evaluation Discussion

As expected from previous studies, the aircraft exhibited a smaller RCS when removing the tail geometry. The change in monostatic detection could prove useful enough for real world application when considering the nature of an ISR aircraft and the inherent weakness of the intended design against adversary fighter aircraft. While the tailless variant performs reconnaissance, the rear 180 degrees of the aircraft are shown to be considerably less detectable when compared to the tailed variant, potentially allowing the aircraft to remain unseen from a distance where the base could be detected and engaged upon. Bistatic RCS displayed less of a benefit between the two, however this could largely be adjusted based on flight profile during mission or avoided with potential information on locations of radar arrays. Overall, even with the minor benefit to bistatic RCS, the monostatic results show a strong enough argument for the removal of geometry in favor of control systems.

15. Review and Considerations

The overall study proved as a great learning opportunity to expand upon understanding of aircraft design, stability, and the affiliated side effects of geometry on applied control systems and

radar cross section. Starting from the initial sizing estimations of the aircraft, constant changes were made to accommodate different design ideas, realistic weight assumptions for components, and potential impacts to stability performance through each of the geometry design steps. The constant changes helped nurture a stronger understanding of aircraft design and how small changes in weight, center of gravity, and neutral point, can drastically affect the flight dynamics and stability characteristics of an aircraft during the design process.

Adding on to the knowledge gained regarding weight and balance and flight dynamics of the aircraft, the lack of access to CFD gave an opportunity to really dive into how each stability and control derivative is created, and how the environment around an aircraft truly impacts each part of the flight envelope. As stated in Chapter 13, this brings along an unfortunate level of inaccuracy for the actual computational results, however since this was not designed to be a production aircraft, the tradeoff of gaining an understanding of the calculations is comparatively invaluable.

For future versions and testing, an actual CFD simulation would be used to compare the calculated results, as well as allow for more accurate use of MATLAB Simulink to create a realistic model of the control systems. Additionally, going a step further and creating a small-scale physical model could lend itself to creating an opportunity of actual applied controls using the Simulink model created. Seeing the connection between simulated controls and practical applications of the models would be a great opportunity to see how everything learned in the project is applied to the real world.

After reviewing the results of the S&C analysis the tailless variant was able to track the tailed variant accurately in roll and was able to maintain a heading predetermined heading angle even with the pilot entered step response of the system. All that was required was a small change in the control outputs to account for the drag rudders and introduction of thrust vectoring. There was a slight overall reduction in RCS around the entire aircraft, especially regarding the monostatic response, meaning that the removal of geometry did, as expected, create a better “stealth” profile than the base variant. While drag rudders and the thrust vectoring nozzles would likely create additional maintenance complications, they do prove that, in the instance of this design, they can be implemented without a problematic change in geometry. Testing this concept on an existing platform, given provided S&C derivatives and geometry, to further confirm or disprove the results of this project, is another facet that could prove extremely interesting and even valuable to the aerospace field today.

References

[1] "Northrop B-2A Spirit," *Fact sheets: Northrop B-2A Spirit: Northrop B-2A Spirit*, 28 February 2008. https://web.archive.org/web/20080228171308/http://www.nationalmuseum.af.mil:80/factsheets/factsheet.asp?id=2757.

[2] Dunbar, B., "NASA dryden fact sheet - F-18 high angle-of-attack (alpha) research vehicle," *NASA*, 8 October 2015. https://www.nasa.gov/centers/armstrong/news/FactSheets/FS-002-DFRC.html.

[3] Dunbar, B., "F-15 active (Advanced Control Technology for Integrated Vehicles) NASA # 837," *NASA*, 10 May 2017. https://www.nasa.gov/centers/dryden/history/pastprojects/Active/index.html.

[4] Dunbar, B., "NASA Dryden Fact Sheets - X-36 tailless fighter Agility Research Aircraft," *NASA*, 12 August 2015. https://www.nasa.gov/centers/armstrong/news/FactSheets/FS-065-DFRC.html.

[5] Kigotho, O. N., Bodylski, J., and Rife, J. H., "Proverse Yaw Agility of Lift Distributions," *Journal of Aircraft*, vol. 59, Aug. 2022, pp. 1390–1393.

[6] Shearwood, T. R., Nabawy, M. R., Crowther, W. J., and Warsop, C., "Yaw control of maneuvering tailless aircraft using induced drag – A control allocation method based on aerodynamic mode shapes," *AIAA aviation forum*, 2020.

[7] Shearwood, T. R., Nabawy, M. R., Crowther, W., and Warsop, C., "A control allocation method to reduce roll-yaw coupling on tailless aircraft," *AIAA Scitech forum*, 2021.

[8] Rossetti, J. S., and Beblo, R., "The design of foldable wingtips on a tailless aircraft for improved yaw control," *AIAA Scitech forum*, 2022.

[9] Raymer, D. P., *Aircraft design: A conceptual approach*, Reston, VA: American Institute of Aeronautics and Astronautics, 2021.

[10] Roskam, J., *Airplane Design*, Ottawa, Kan.: Roskam Aviation and Engineering Corp., 1989.

[11] Roskam, J., *Airplane Flight Dynamics and automatic flight controls*, Lawrence, Kan.: DARcorporation, 2018.

[12] Stadmore, H. A., "Radar Cross Section Fundamentals for the aircraft designer," *Aircraft Systems and Technology Meeting*, 1979.

[13] Riemen, N., "Scorpion Jet makes FIDAE International Air and Space Debut," *ScorpionJet,* 29 March 2016. https://scorpion.txtav.com/en/logbook/scorpionjet-logbook-entry?id=50EE6B46B44D4044811954E89D0C6006.

[14] “Grumman A-6 Intruder,” *Wikipedia* https://en.wikipedia.org/wiki/Grumman_A-6_Intruder.

[15] Mikesh, R. C., “Martin B-57 Canberra,” *Wikipedia*, USAF, 1995 https://en.wikipedia.org/wiki/Martin_B-57_Canberra.

[16] Acharya, S., “General Electric J85-GE-17A,” *Wikipedia* https://en.wikipedia.org/wiki/General_Electric_J85#/media/File:J85_ge_17a_turbojet_engine.jpg.

[17] Ikaza, D., and Industria De Turbo Propulsores Madrid (Spain), “Thrust vectoring nozzle for modern military aircraft,” *DTIC* Available: https://apps.dtic.mil/sti/citations/ADP011109.

[18] Moorhouse, D. J., and Woodcock, R. J., Wright-Patterson Air Force Base, OH: Air Force Flight Dynamics Laboratory, 1982.

[19] Langley Aeronautical Laboratory, and Stone, R. W., Langley Field, Virginia: NACA, 1952.

[20] Engineeringtoolbox, E., “U.S. standard atmosphere vs. altitude,” *Engineering ToolBox* https://www.engineeringtoolbox.com/standard-atmosphere-d_604.html.

[21] “Appendix H: Stability characteristics of selected aircraft,” *Introduction to Aircraft Flight Mechanics*, Jan. 2003, pp. 613–622.

[22] Stevens, B. L., Lewis, F. L., and Johnson, E. N., *Aircraft Control and Simulation: Dynamics, controls design, and Autonomous Systems*, Hoboken, New Jersey: John Wiley & Sons, 2016.

[23] Lutze, F. H., “AOE 3134 Complete Vehicle Lift and Pitch-Moment Equations,” *AOE 3134* https://archive.aoe.vt.edu/lutze/AOE3134/completevehicle.pdf.

[24] Lutze, F. H., “AOE 3134 Yaw and Roll Moment Equations and Estimation,” *AOE 3134* https://archive.aoe.vt.edu/lutze/AOE3134/YawMoment.pdf.

[25] Williams, J. E., Vukelich, S. R., and Finck, R. D., Greene County , Ohio: AFWAL, 1978.

[26] Jenn, D. C., and Schetz, J. A., *Radar and Laser Cross Section Engineering*, Reston, VA: American Institute of Aeronautics and Astronautics, Inc., 2019.

[27] Chatzigeorgiadis, F., Jenn, D. C., and Shleher, D. C., “Development of Code for A Physical Optics Radar Cross Section Prediction and Analysis Application,” thesis, Naval Postgraduate School, 2004.

Appendix A. MATLAB Thrust Vectoring Analysis Code

Contents

Thrust Vectoring and Differential Thrust Directional Analysis

The purpose of this program is to create a visual (graph) of the effects of propulsion force with a twin engine jet aircraft fitted with a differential thrust control system as well as 3-D thrust vectoring nozzles capable of up to 20 degree deflection in the x-y axis. By: Tripp Selvig

```
clear all, clc, format shortG

MaxThrust_Each = 3600;       % Maximum Engine Thrust of a Single Engine (lbf)
MaxThrustAB_Each = 5000;     % Maximum Engine Thrust of a Single Engine with Afterburner (lbf)

Deflection_Pos= 0:1:20;         % Angle of Thrust Vectoring Deflection in the X-axis (yaw) (degrees)
Deflection_Neg = -20:1:0;       % Angle of Thrust Vectoring Deflection in the Y-axis (pitch) (degrees)

RightDeflection = 270 + Deflection_Pos;  % Angle of Thrust Vectoring Deflection in the X-axis (yaw) (degrees)
LeftDeflection = 270 + Deflection_Neg;   % Angle of Thrust Vectoring Deflection in the X-axis (yaw) (degrees)
```

Right Engine Thrust Vectoring

First example will be no afterburner on either engine

```
ThrustLeftStaticInitial = ones(21,1);
ThrustRightInitial = ones(21,2);
ThrustLeftStatic = ThrustLeftStaticInitial .* MaxThrust_Each;
ThrustRightStatic = ThrustRightInitial .* MaxThrust_Each;
for i = length(ThrustRightInitial)
    j = 1;
    while j <= i
    ThrustRight(j,1) = ThrustRightStatic(j,1) .* sind(RightDeflection(j));
    ThrustRight(j,2) = ThrustRightStatic(j,2) .* cosd(RightDeflection(j));
    TotalThrust_RightVector(j,1) = ThrustRight(j,1) - ThrustLeftStatic(j);
    TotalThrust_RightVector(j,2) = ThrustRight(j,2);
    j = j + 1;
    end
end

% Second Example Involves Left Engine Standard, Right Engine Afterburner Engaged
ThrustRightInitial = ones(21,2);
ThrustRightStaticAB = ThrustRightInitial .* MaxThrustAB_Each;
for i = length(ThrustRightInitial)
    j = 1;
    while j <= i
    ThrustRightAB(j,1) = ThrustRightStaticAB(j,1) .* sind(RightDeflection(j));
    ThrustRightAB(j,2) = ThrustRightStaticAB(j,2) .* cosd(RightDeflection(j));
    TotalThrust_RightVectorAB(j,1) = ThrustRightAB(j,1) - ThrustLeftStatic(j);
    TotalThrust_RightVectorAB(j,2) = ThrustRightAB(j,2);
    j = j + 1;
    end
end
```

```
% Third Example Involves Left Engine Afterburner, Right Engine Standard Thrust
ThrustLeftStaticAB = ThrustLeftStaticInitial .* MaxThrustAB_Each;
for i = length(ThrustRightInitial)
    j = 1;
    while j <= i
    ThrustRight(j,1) = ThrustRightStatic(j,1) .* sind(RightDeflection(j));
    ThrustRight(j,2) = ThrustRightStatic(j,2) .* cosd(RightDeflection(j));
    TotalThrustAB_RightVector(j,1) = ThrustRight(j,1) - ThrustLeftStaticAB(j);
    TotalThrustAB_RightVector(j,2) = ThrustRight(j,2);
    j = j + 1;
    end
end

% Final Example Involves Both Engines Using Afterburner
for i = length(ThrustRightInitial)
    j = 1;
    while j <= i
    TotalThrustAB_RightVectorAB(j,1) = ThrustRightAB(j,1) - ThrustLeftStaticAB(j);
    TotalThrustAB_RightVectorAB(j,2) = ThrustRightAB(j,2);
    j = j + 1;
    end
end
```

Both Engines Thrust Vectoring Positive X Direction

Both engines Thrust Vectoring Positive X Direction No Afterburner

```
ThrustLeftInitial = ones(21,2);
ThrustLeftStaticInitial = ThrustLeftInitial .* MaxThrust_Each;
for i = length(ThrustRightInitial)
    j = 1;
    while j <= i
    BothThrustRight(j,1) = ThrustRightStatic(j,1) .* sind(RightDeflection(j));
    BothThrustRight(j,2) = ThrustRightStatic(j,2) .* cosd(RightDeflection(j));
    BothThrustLeft(j,1) = ThrustLeftStaticInitial(j,1) .* sind(RightDeflection(j));
    BothThrustLeft(j,2) = ThrustLeftStaticInitial(j,2) .* cosd(RightDeflection(j));
    MaxTotalThrust_BothStandard(j,1) = BothThrustLeft(j,1) + BothThrustRight(j,1);
    MaxTotalThrust_BothStandard(j,2) = BothThrustLeft(j,2) + BothThrustRight(j,2);
    j = j + 1;
    end
end

% Both engines Thrust Vectoring Positive X Direction WITH Afterburner
ThrustLeftStaticInitialAB = ThrustLeftInitial .* MaxThrustAB_Each;
for i = length(ThrustRightInitial)
    j = 1;
    while j <= i
    BothThrustRightAB(j,1) = ThrustRightStaticAB(j,1) .* sind(RightDeflection(j));
    BothThrustRightAB(j,2) = ThrustRightStaticAB(j,2) .* cosd(RightDeflection(j));
    BothThrustLeftAB(j,1) = ThrustLeftStaticInitialAB(j,1) .* sind(RightDeflection(j));
    BothThrustLeftAB(j,2) = ThrustLeftStaticInitialAB(j,2) .* cosd(RightDeflection(j));
    MaxTotalThrust_BothAB(j,1) = BothThrustLeftAB(j,1) + BothThrustRightAB(j,1);
    MaxTotalThrust_BothAB(j,2) = BothThrustLeftAB(j,2) + BothThrustRightAB(j,2);
    j = j + 1;
    end
end
```

Appendix B. MATLAB Matching Graph Code

Contents

Matching Graph Proposed Design vs. Raymer Expected Method

Design for the proposed project aircraft using values based on previous aircraft designs. The proposed values are compared to the values given in Raymer's "Aircraft Design: A Conceptual Approach" Additional reference values are provided by Roskam's "Airplane Design"

```
clear all, clc, format shortG    % shortG limits values to 5 sig figs
```

Raymer Tables

Table 5.1 Thrust to Weight Ratio Aircraft Type Typical T/W Jet Trainer 0.4 Jet fighter (dogfighter) 0.9 Jet fighter (other) 0.6

```
% Table 5.3 Thrust/Max Takeoff Weight vs Max Mach Number (MaxMach)
% Thrust/Max Takeoff Weight (T/W_0) = alpha*MaxMach^C
% Aircraft Type               alpha     C
% Jet trainer                 0.488     0.728
% Jet fighter (dogfighter)    0.648     0.594
% Jet fighter (other)         0.514     0.141

% Table 5.5 Wing Loading, Typical Takeoff W/S
% Historical Trends         lb/ft^2
% Jet trainer                 50
% Jet fighter                 70
```

Input Values

User Inputs

```
W_0 = 14500;               % Max Takeoff Weight (lbs)
W_S = 0:10:150;            % Range of W/S for plot (lbs/ft^2)
W_S_Project = 38.077;      % Project based W/S (lb/ft^2)
T_W_Project = 0.497;       % Estimated T/W Requirement

% Raymer Input Values
W_S_Raymer = 50;           % Table 5.5 Value (lb/ft^2)
T_W_Raymer = 0.40;         % Table 5.1 Value

% Speed and Rate Requirements
Cruise_W_W0 = 0.9;         % Cruise Weight/Takeoff Weight
Loiter_W_W0 = 0.85;        % Loiter Weight/Takeoff Weight
Climb_W_W0 = 1;            % Climb Weight/Takeoff Weight
Service_W_W0 = 0.9;        % Service Ceiling Weight/Takeoff Weight

CruiseMach = 0.52494;                     % Cruise Velocity (M)
MaxMach = 0.89990;                        % Max Velocity (M)
ClimbRate = 7000;                         % Max Climb Rate (ft/min)
ClimbAltitude_0 = 0;                      % Starting Climb Altitude (ft)
CruiseAltitude = 30000;                   % Desired Cruise Altitude (ft)
ClimbRateCeiling = 500;                   % Climb Rate at Service Ceiling (Raymer Value) (ft/min)
ClimbRateCeiling_FPS = ClimbRateCeiling/60; % Climb Rate at Service Ceiling (Raymer Value) (ft/sec)
CeilingAltitude = 50000;                  % Desired Service Ceiling Altitude (ft)

% Aircraft Aerodynamic Parameters
C_D0 = 0.015;              % Subsonic Parasitic Drag Coefficient
C_D_CruiseWave = 0.003;    % Estimation from Roskam Figure 3.32 based on F-107
```

```
C_D_MaxWave = 0.004;        % Estimation from Roskam Figure 3.32 based on F-107
AR = 5;                     % Desired Aspect Ratio
e = 0.9;                    % Calculated Clean Oswald Efficiency Factor
K = 1/(pi*AR*e);            % Induced Drag Factor

T_W_0 = 0.488*MaxMach^0.728;    % Estimated T/W based on Max Mach Number (Raymer)
```

Atmospheric Conditions at Cruise Altitude

Using the MATLAB Aerospace Blockset and atmosisa function combined with local weather data for the expected operational area allows proper calculation of air density, local speed of sound, dynamic pressure, and cruise speed at any designated altitude. Temperature and Pressure calculated using turbojet_afterburner function

```
CruiseAltitudeMetric = CruiseAltitude*0.3048;           % Convert Cruise Altitude to Metric for Atmosisa
[~, SoS_CruiseMetric, ~, rho_CruiseMetric] = atmosisa(CruiseAltitudeMetric);
SoS_CruiseImp = SoS_CruiseMetric*3.28084;               % Cruise Altitude Speed of Sound Imperial Units (ft/s)
rho_CruiseImp = rho_CruiseMetric*0.00194032;            % Cruise Altitude Air Density Imperial Units (slugs/ft^3)
Velocity_Cruise = CruiseMach*SoS_CruiseImp*0.592484;    % Velocity at Cruise Altitude (knots)
Velocity_Cruise_FPS = CruiseMach*SoS_CruiseImp;         % Velocity at Cruise Altitude for Dynamic Pressure Calculation (ft/s)
q_Cruise = 0.5*rho_CruiseImp*(Velocity_Cruise_FPS^2);   % Dynamic Pressure during Cruise Altitude/Speed (lb/ft^2)
% End of atmosisa function

WS_Cruise = W_S * Cruise_W_W0;   % Adjusting Initial W/S Range for Cruise Weight (lb/ft^2)
```

T/W Required Calculation for Cruise

Based on Estimated Drag Polar

```
TW_Cruise = (C_D0 + C_D_CruiseWave) * q_Cruise./WS_Cruise + WS_Cruise/(q_Cruise*pi*AR*e);  % Roskam Equation 3.60
ThrustCruise_vs_TotalThrust = turbojet_afterburner(1,CruiseMach,CruiseAltitude);          % Call Engine Power Function (turbojet_afterburner)
TW_Cruise_Takeoff = TW_Cruise * Cruise_W_W0 / ThrustCruise_vs_TotalThrust;  % Convert Cruise T/W to Takeoff T/W (Raymer Eq. 5.4)
WS_Cruise_Ideal = q_Cruise * sqrt(pi*AR*e*(C_D0 + C_D_CruiseWave)/3);       % Ideal Cruise W/S for Best Range (Raymer Eq. 5.14) (lb/ft^2)
C_L_Cruise = 2 * WS_Cruise / (rho_CruiseImp * Velocity_Cruise_FPS.^2);     % Cruise CL Calculation
C_D_Cruise = C_D0 + K *C_L_Cruise.^2;                                       % Cruise CD Calculation
Lift_Drag_Cruise = C_L_Cruise./C_D_Cruise;                                  % Cruise L/D Calculation
```

Atmospheric Conditions at Max Altitude

Once again calling atmosisa for calculations at expected operational service ceiling. Uses equations from Raymer (Eq. 5.4), Roskam (Eq. 3.60) and Gudmundsson (Eq. 3-10).

```
CeilingAltitudeMetric = CeilingAltitude*0.3048;
[~, SoS_CeilingMetric, ~, rho_CeilingMetric] = atmosisa(CeilingAltitudeMetric);
SoS_CeilingImp = SoS_CeilingMetric*3.28084;                     % Ceiling Altitude Speed of Sound Imperial Units (ft/s)
rho_CeilingImp = rho_CeilingMetric*0.00194032;                  % Ceiling Altitude Air Density Imperial Units (slugs/ft^3)
Max_Velocity_Ceiling = MaxMach*SoS_CeilingImp*0.592484;         % Velocity at Ceiling Altitude (knots)
Max_Velocity_Ceiling_FPS = MaxMach*SoS_CeilingImp;              % Velocity at Ceiling Altitude for Dynamic Pressure Calculation (ft/s)
q_Ceiling = 0.5*rho_CeilingImp*(Max_Velocity_Ceiling_FPS^2);    % Dynamic Pressure during Ceiling Altitude/Speed (lb/ft^2)
% End of atmosisa function

WS_Ceiling = W_S * Service_W_W0;   % Adjusting W/S Values at Service Ceiling and Max Speed
```

T/W Required Calculation for Max Speed

```
TW_Max = (C_D0 + C_D_MaxWave)*q_Ceiling./WS_Ceiling + WS_Ceiling/(q_Ceiling*pi*AR*e);     % Roskam Equation 3.60
ThrustMax_vs_TotalThrust = turbojet_afterburner(1,MaxMach,CeilingAltitude);   % Call Engine Power Function (turbojet_afterburner)
TW_Max_Takeoff = TW_Max * Service_W_W0 / ThrustMax_vs_TotalThrust;           % Convert Max T/W to Takeoff T/W (Raymer Eq. 5.4)
WS_Max_Ideal = q_Ceiling * sqrt(pi*AR*e*(C_D0 + C_D_MaxWave)/3);             % Ideal Max W/S for Best Range (Raymer Eq. 5.14) (lb/ft^2)
C_L_Ceiling = 2 * WS_Cruise / (rho_CruiseImp * Max_Velocity_Ceiling_FPS.^2); % Ceiling CL Calculation
C_D_Ceiling = C_D0 + K *C_L_Cruise.^2;                                        % Ceiling CD Calculation
Max_Lift_Drag_Ceiling = C_L_Ceiling./C_D_Ceiling;
```

Climb Rate Calculations

Calculate expected climb rate values to ensure proper sizing and engine choice Utilizes equations from Raymer (Eq. 5.3, 5.27, 5.31) and Gudmundsson (Eq. 3.2 - 3.5)

```
% Atmospheric Conditions during Climb
CeilingAltitudeMetric = ClimbAltitude_0 * 0.3048;
[~, SoS_ClimbMetric, ~, rho_ClimbMetric] = atmosisa(CeilingAltitudeMetric);
SoS_ClimbImp = SoS_ClimbMetric*3.28084;             % Climb Altitude Speed of Sound Imperial Units (ft/s)
rho_ClimbImp = rho_ClimbMetric*0.00194032;          % Climb Altitude Air Density Imperial Units (slugs/ft^3)
```

```
% End of atmosisa function
% Velocities at each interval during climb (500 ft) calculated below

WS_Climb = W_S * Climb_W_W0;    % Adjusting W/S Values during Climb
ClimbRate_FPS = ClimbRate/60;   % Climb Rate (ft/s)

% Preallocate Arrays for Climb Rate Calculations
Velocity_Climb = zeros(1,length(WS_Climb));
Velocity_Climb_FPS = zeros(1,length(WS_Climb));
C_L_Climb = zeros(1,length(WS_Climb));
C_D_Climb = zeros(1,length(WS_Climb));
Lift_Drag_Climb = zeros(1,length(WS_Climb));
TW_Climb = zeros(1,length(WS_Climb));
Mach_Climb = zeros(1,length(WS_Climb));
ThrustClimb_vs_TotalThrust = zeros(1,length(WS_Climb));
TW_Climb_vs_TotalThrust = zeros(1,length(WS_Climb));
ClimbGradient = zeros(1,length(WS_Climb));

% Calculate Thrust/Weight for each W/S value during climb
for i = 1:length(WS_Ceiling)
    % Initialize Takeoff/Weigh values for iteration, results are considered
    % converged when the ratio barely changes
    TW_Initial = 0;
    TW_Updated = 0.1;
    while abs(TW_Initial-TW_Updated) > 0.001
        TW_Updated = TW_Initial;
        Velocity_Climb_FPS(i) = sqrt(WS_Climb(i)/(3*rho_ClimbImp*C_D0)*(TW_Updated + sqrt(TW_Updated^2 + 12*C_D0*K)));
        Velocity_Climb(i) = Velocity_Climb_FPS(i)*0.592484;
        ClimbGradient(i) = ClimbRate_FPS/Velocity_Climb_FPS(i);
        C_L_Climb(i) = 2*WS_Ceiling(i)/(rho_CeilingImp*Velocity_Climb_FPS(i)^2);
        C_D_Climb(i) = C_D0 + K*C_L_Climb(i)^2;
        Lift_Drag_Climb(i) = C_L_Climb(i)/C_D_Ceiling(i);
        TW_Climb(i) = 1/Lift_Drag_Climb(i) + ClimbGradient(i);
        TW_Initial = TW_Climb(i);
    end
    Mach_Climb(i) = Velocity_Climb_FPS(i)/SoS_ClimbImp;
    ThrustClimb_vs_TotalThrust(i) = turbojet_afterburner(1,Mach_Climb(i),ClimbAltitude_0);
    TW_Climb_vs_TotalThrust(i) = TW_Climb(i)*Climb_W_W0/ThrustClimb_vs_TotalThrust(i);
end
```

Service Ceiling Calculations

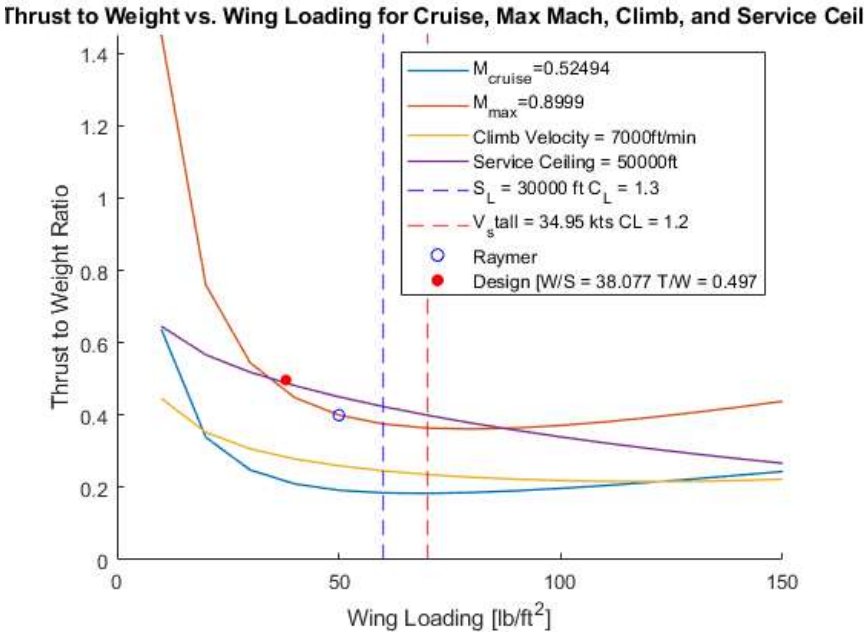
Using Atmospheric conditions calculated in prior section (Atmospheric Conditions at Max Altitude)

```
% Preallocate Arrays for Service Ceiling calculations
Velocity_Ceiling = zeros(1,length(WS_Ceiling));
Velocity_Ceiling_FPS = zeros(1,length(WS_Ceiling));
C_L_Ceiling = zeros(1,length(WS_Ceiling));
C_D_Ceiling = zeros(1,length(WS_Ceiling));
Lift_Drag_Ceiling = zeros(1,length(WS_Ceiling));
TW_Ceiling = zeros(1,length(WS_Ceiling));
Mach_Ceiling = zeros(1,length(WS_Ceiling));
ThrustCeiling_vs_TotalThrust = zeros(1,length(WS_Ceiling));
TW_Ceiling_vs_TotalThrust = zeros(1,length(WS_Ceiling));
CeilingGradient = zeros(1,length(WS_Ceiling));

% Calculate Thrust/Weight for each W/S value during ceiling operations
for i = 1:length(WS_Ceiling)
    % Initialize Takeoff/Weigh values for iteration, results are considered
    % converged when the ratio barely changes
    TW_Initial = 0;
    TW_Updated = 0.1;
    while abs(TW_Initial-TW_Updated) > 0.001
        TW_Updated = TW_Initial;
        Velocity_Ceiling_FPS(i) = sqrt(WS_Ceiling(i)/(3*rho_CeilingImp*C_D0)*(TW_Updated + sqrt(TW_Updated^2 + 12*C_D0*K)));
        Velocity_Ceiling(i) = Velocity_Ceiling_FPS(i)*0.592484;
        CeilingGradient(i) = ClimbRateCeiling_FPS/Velocity_Ceiling_FPS(i);
        C_L_Ceiling(i) = 2*WS_Ceiling(i)/(rho_CeilingImp*Velocity_Ceiling_FPS(i)^2);
        C_D_Ceiling(i) = C_D0 + K*C_L_Ceiling(i)^2;
        Lift_Drag_Ceiling(i) = C_L_Ceiling(i)/C_D_Ceiling(i);
        TW_Ceiling(i) = 1/Lift_Drag_Ceiling(i) + CeilingGradient(i);
        TW_Initial = TW_Ceiling(i);
    end
    Mach_Ceiling(i) = Velocity_Ceiling_FPS(i)/SoS_ClimbImp;
```

```
    ThrustCeiling_vs_TotalThrust(i) = turbojet_afterburner(1,Mach_Ceiling(i),CeilingAltitude);
    TW_Ceiling_vs_TotalThrust(i) = TW_Ceiling(i)*Service_W_W0/ThrustCeiling_vs_TotalThrust(i);
end
ThrustJet = W_0*T_W_Project;
```

Plot Results in Matching Graph

```
figure
hold on
xlim([0 inf])
ylim([0 inf])
plot(W_S,TW_Cruise_Takeoff,'LineWidth', 1.0)
plot(W_S,TW_Max_Takeoff,'LineWidth', 1.0)
plot(W_S,TW_Climb_vs_TotalThrust,'LineWidth', 1.0)
plot(W_S,TW_Ceiling_vs_TotalThrust,'LineWidth', 1.0)
xline(60,'--b','LineWidth',1.0);
xline(70,'--r','LineWidth',1.0);
scatter(W_S_Raymer,T_W_Raymer,30,'blue')
scatter(W_S_Project,T_W_Project,30,'red','filled')
title('Thrust to Weight vs. Wing Loading for Cruise, Max Mach, Climb, and Service Ceiling')
xlabel('Wing Loading [lb/ft^2]')
ylabel('Thrust to Weight Ratio')
legend(['M_c_r_u_i_s_e=',num2str(CruiseMach)],...
       ['M_m_a_x=',num2str(MaxMach)],...
       ['Climb Velocity = ',num2str(ClimbRate),'ft/min'],...
       ['Service Ceiling = ',num2str(CeilingAltitude),'ft'],...
       'S_L = 30000 ft C_L = 1.3',...
       'V_stall = 34.95 kts CL = 1.2',...
       'Raymer',...
       'Design [W/S = 38.077 T/W = 0.497',...
       'Location','northeast')
```



Published with MATLAB® R2024a

Contents


```
function ThrustJet = turbojet_afterburner(Thrust_SL, Mach, Altitude_Imp)
```

```
% Turbojet engine maximum thrust with afterburner at altitude
% Assumes standard atmospheric temperatures and pressures
% Equations and assumptions provided by Gudmundsson Chapter 14.3.3
```

Equations (Gamma = 1.2 at chosen Mach Values due to Compressibilitiy)

Temperature Ratio Below (Eq. 14-66) Temperature/Total Temperature * (1 + ((Gamma - 1)/ 2)) **Mach^2) Simplified for Program: Temperature/Total Temperatuure * (1 + 0.2*M^2) Pressure Ratio Below (Eq. 14-67) Pressure/Total Pressure * (1 + ((Gamma - 1)/ 2)) *Mach^2)^(Gamma/(Gamma - 1) Simplified for Program: Temperature/Total Temperatuure * (1 + 0.2*M^2)^3.5 Maximum Thrust with Afterburner Equations Equation 14-71: Thrust = Thrust_SL * Pressure Ratio * (1 - 0.3**(Temperature Ratio - 1) - 0.1*sqrt(Mach)) Equation 14-72: Thrust = Thrust_SL * Pressure Ratio * (1 - 0.3*(Temperature Ratio - 1) - 0.1*sqrt(Mach) - 1.5*(Temperature Ratio - Thrust Ratio)/Temperature Ratio)

Begin Function

```
ThrottleRatio = 1.0;
Altitude_Metric = Altitude_Imp*0.3048;  % atmosisa requires metric input
[Temperature_SL, ~, Pressure_SL, ~] = atmosisa(0);  % Local Mach and air density found in main program
[Temperature_Cruise, ~, Pressure_Cruise, ~] = atmosisa(Altitude_Metric);

% Loop below created for multiple configurations of altitude and Mach value(s) selected

% Single Mach Number, Single Altitude Input
if (length(Mach) == 1) && (length(Altitude_Imp) == 1)
    Temp_vs_TotalTemp = Temperature_Cruise/Temperature_SL*(1 + 0.2*Mach^2);
    Pressure_vs_TotalPressure = Pressure_Cruise/Pressure_SL*(1 + 0.2*Mach^2)^3.5;
    if Temp_vs_TotalTemp <= ThrottleRatio
        ThrustJet = Thrust_SL * Pressure_vs_TotalPressure * (1 - 0.3*(Temp_vs_TotalTemp - 1) - 0.1*sqrt(Mach));
    else
        ThrustJet = Thrust_SL * Pressure_vs_TotalPressure * (1 - 0.3*(Temp_vs_TotalTemp - 1) - 0.1*sqrt(Mach) - 1.5*(Temp_vs_TotalTemp - ThrottleRatio)/Temp_vs_Tota
    end

    % Multiple Mach Number, Single Altitude Input
elseif (length(Mach) > 1) && (length(Altitude_Imp) == 1)
    Temp_vs_TotalTemp = zeros(length(Altitude_Imp),length(Mach));
    Pressure_vs_TotalPressure = zeros(length(Altitude_Imp),length(Mach));
    ThrustJet = zeros(length(Altitude_Imp),length(Mach));

    for i = length(Altitude_Imp),length(Mach)
        Temp_vs_TotalTemp(i) = Temperature_Cruise/Temperature_SL*(1 + 0.2*Mach^2);
        Pressure_vs_TotalPressure(i) = Pressure_Cruise/Pressure_SL*(1 + 0.2*Mach^2)^3.5;
        if Temp_vs_TotalTemp(i) <= ThrottleRatio
        ThrustJet(i) = Thrust_SL * Pressure_vs_TotalPressure(i) * (1 - 0.3*(Temp_vs_TotalTemp(i) - 1) - 0.1*sqrt(Mach));
        else
        ThrustJet(i) = Thrust_SL * Pressure_vs_TotalPressure(i) * (1 - 0.3*(Temp_vs_TotalTemp(i) - 1) - 0.1*sqrt(Mach) - 1.5*(Temp_vs_TotalTemp(i) - ThrottleRatio)/
        end
    end

    % Single Mach Number, Multiple Altitudes
elseif (length(Mach) == 1) && (length(Altitude_Imp) > 1)
    Temp_vs_TotalTemp = zeros(length(Altitude_Imp),length(Mach));
    Pressure_vs_TotalPressure = zeros(length(Altitude_Imp),length(Mach));
    ThrustJet = zeros(length(Altitude_Imp),length(Mach));

    for j = length(Mach):length(Altitude_Imp)
        Temp_vs_TotalTemp(j) = Temperature_Cruise(j)/Temperature_SL*(1 + 0.2*Mach^2);
        Pressure_vs_TotalPressure(j) = Pressure_Cruise(j)/Pressure_SL*(1 + 0.2*Mach^2)^3.5;
        if Temp_vs_TotalTemp(j) <= ThrottleRatio
        ThrustJet(j) = Thrust_SL * Pressure_vs_TotalPressure(j) * (1 - 0.3*(Temp_vs_TotalTemp(j) - 1) - 0.1*sqrt(Mach));
        else
        ThrustJet(j) = Thrust_SL * Pressure_vs_TotalPressure(j) * (1 - 0.3*(Temp_vs_TotalTemp(j) - 1) - 0.1*sqrt(Mach) - 1.5*(Temp_vs_TotalTemp(j) - ThrottleRatio)/
        end
    end

    % Multiple Mach Numbers, Multiple Altitudes
elseif (length(Mach) > 1) && (length(Altitude_Imp) > 1)
    Temp_vs_TotalTemp = zeros(length(Altitude_Imp),length(Mach));
    Pressure_vs_TotalPressure = zeros(length(Altitude_Imp),length(Mach));
    ThrustJet = zeros(length(Altitude_Imp),length(Mach));

    for j = 1:length(Altitude_Imp)  % Loop for fulfilling Altitude Array
        for i = 1:length(Mach)  % Loop for fulfilling Mach Array
        Temp_vs_TotalTemp(i,j) = Temperature_Cruise(j)/Temperature_SL*(1 + 0.2*Mach^2);
        Pressure_vs_TotalPressure(i,j) = Pressure_Cruise(j)/Pressure_SL*(1 + 0.2*Mach^2)^3.5;
            if Temp_vs_TotalTemp(i,j) <= ThrottleRatio
            ThrustJet(i,j) = Thrust_SL * Pressure_vs_TotalPressure(i,j) * (1 - 0.3*(Temp_vs_TotalTemp(i,j) - 1) - 0.1*sqrt(Mach));
            else
```

```
            ThrustJet(i,j) = Thrust_SL * Pressure_vs_TotalPressure(i,j) * (1 - 0.3*(Temp_vs_TotalTemp(i,j) - 1) - 0.1*sqrt(Mach) - 1.5*(Temp_vs_TotalTemp(i,j) - Thr
            end
        end
    end
else
    fprintf('\n Error. \n');  % Print error message if no conditions are met
end
```

```
Not enough input arguments.
Error in turbojet_afterburner (line 23)
Altitude_Metric = Altitude_Imp*0.3048;  % atmosisa requires metric input
```

```
end
```

Published with MATLAB® R2024a

Appendix C. MATLAB Control Systems Code

Contents


```
clear all, close all, clc
```

AE 295 Base Aircraft Stability and Controls

In accordance with the requirements for completion of AE295 Design of a Military Class 4 ISR Aircraft for Comparison with a Tailless Variant By: Tripp Selvig

Base Environmental and Aircraft Constants

```
g = 32.17;                          % Gravity [ft/s^2]
h = 30000;                          % Cruise Altitude [ft]
rho = 8.91E-4;                      % Air density at altitude [slugs/ft^3]
u_inf = 3.107E-7;                   % Dynamic Viscosity [slug/(ft*s)]
U1 = 590.73;                        % Thrust Velocity [ft/s] - [350 kts]
q = 0.5*rho*U1^2;                   % Dynamic pressure
L = 43.515;                         % Fuselage length [ft]
b = 39.56;                          % Wingspan [ft]
S = 380.81;                         % Wing area [ft^2]
c = 9.844;                          % Wing Mean Aerodynamic Chord (MAC) [ft]
c_root = 16.23;                     % Wing root chord [ft]
c_tip = 3.68;                       % Wingtip chord [ft]
AR = b^2/S;                         % Aspect Ratio [4.1096 unitless]
S_h = 108;                          % Horizontal Stabilizer wing area [ft^2]
b_h = 18;                           % Horizontal Stabilizer wingspan [ft]
AR_h = b_h^2/S_h;                   % Horizontal Stabilizer Aspect Ratio [3 unitless]
L_h = 18.5;                         % Moment arm of the Horizontal Stabilizer [ft]
L_v = 17;                           % Moment arm of the Vertical Stabilizer [ft]
S_v = 50.72;                        % Vertical Stabilizer wing area [ft^2]
b_v = 5.56;                         % Vertical Stabilizer wingspan [ft]
AR_v = b_v^2/S_v^2;                 % Vertical Stabilizer Aspect Ratio [3 unitless]
S_body_fore = 15.622;               % Frontal surface area of fuselage [ft^2]
S_body_lat = 221.901;               % Lateral surface area of fuselage [ft^2]
b_body = 5.1;                       % Average body diameter at wing root [ft]
S_a = (0.25*b/2)*(0.2*c);           % Drag rudder (aileron) surface area [ft^2]
L_a = 16.3185;                      % Moment arm of drag rudders [ft]
e0 = 1.78*(1-0.045*AR^0.68) - 0.64; % Oswald Efficiency Factor
e = 1/(1+(1/(AR*e0)));              % Span Efficiency Factor
e0_h = 1.78*(1-0.045*AR_h^0.68) - 0.64; % Oswald Efficiency Factor
e_h = 1/(1+(1/(AR_h*e0_h)));        % Span Efficiency Factor
Re_L = (rho*U1*S)/u_inf;            % Reynolds Number
Weight = 14500;                     % Aircraft weight [lb]
Lift = 14500;                       % Lift = Weight of the aircraft durings steady flight [lb]
Thrust = 7200;                      % Engine Thrust from two J85-GE-21 engines [lbf]
Thrust_AB = 10000;                  % Afterburning Thrust [lbf]
Drag = 7200;                        % Aircraft drag during steady flight given by Thrust = Drag [lbf]
Drag_AB = 10000;                    % Aircraft drag during steady flight with afterburners engaged [lbf]
Cd_0 = 0.0113;                      % Clean Cd_0 calculated with Roskam method
Ixx = 837183.33/g;                  % Aircraft Roll moment of inertia (Solidworks MOI) [lb/ft^2]
Iyy = 1111041.29/g;                 % Aircraft Pitch moment of inertia (Solidworks MOI) [lb/ft^2]
Izz = 1926273.53/g;                 % Aircraft Yaw moment of inertia (Solidworks MOI) [lb/ft^2]
Ixz = 0;                            % Aircraft Ixz
V_h = 0.191;                        % Horizontal Stabilizer Volume Coefficient
q_h = 0.5*rho*V_h*(U1^2);           % Dynamic pressure at the horizontal stabilizer
X_cg = 22.35;                       % Aircraft CG [ft]
X_np = 24.553;                      % Aircraft Neutral Point [ft]
AoA_Tail = 0;                       % Horizontal Tail incidence angle [deg]
```

```
X_cg_h = X_cg + L_h;                    % Horizontal Tail CG Location [ft]
X_ac_v = X_cg + L_v;                    % Vertical Stabilizer AC Location [ft]
L_engine = 0.866;                       % Moment arm of each engine [ft]
A_inlet = pi*0.866^2;                   % Engine inlet area [ft^2]
m_dot_inlet = rho*U1*(A_inlet);         % Mass Flow Rate through each engine
V_inlet = m_dot_inlet/(A_inlet*rho);    % Engine Inlet flow velocity [ft/s]
```

Aircraft Flight Regime Assumptions

```
alpha = 0;                      % Steady flight [deg]
alpha_rad = deg2rad(alpha);     % Steady flight [rad]
```

Longitudinal Stability Coefficients

```
a0 = (0.7046326-0.04012974)/6;                                  % Lift slope calculated between 6 and 0 degrees AoA respectively
dele0 = (1.5 - 0)/(14.16);                                      % Change in lift due to elevator deflection
eta_h = q_h/q;                                                  % Dynamic pressure ratio at horizontal stabilizer
CL = Lift/(q*S);                                                % Coefficient of lift
Cd = Drag/(q*S) + CL^2/(pi*e*AR);                               % Coefficient of drag for entire aircraft
Cm_w = CL*(X_np-X_cg)/c;                                        % Pitching Moment Coefficient provided by the Main Wing
Cd_AB = Drag_AB/(q*S);                                          % Coefficient of drag with afterburners engaged
Cdi = (CL^2)/(pi*exp(1)*AR);                                    % Lift induced coefficient of drag
ai = CL/(pi*exp(1)*AR);                                         % Induced AoA
CL_alpha = (a0/(1+(a0/(pi*e*AR))))*(180/pi);                    % Coefficient of lift with respect to alpha (CHECK FOR DEGREES/RADIANS)
CL_alpha_h = (dele0/(1+(dele0/(pi*e_h*AR_h))))*(180/pi);        % Coefficient of lift from the Horizontal Stabilizer with respect to alpha (CHECK FOR DEGREES/RA
CL_dele = (dele0/(1+(dele0*(pi*e*AR_h))))*(180/pi);             % Coefficient of lift with respect to elevator deflection (CHECK FOR DEGREES/RADIANS)
Cm_alpha = CL_alpha*(X_cg - X_np)/c;                            % Pitching Moment Coefficient with respect to change in alpha
Cm_alphadot = -2*q_h*V_h*CL_dele*(X_np - X_cg);                 % Pitching Moment Coefficient with respect to change in alpha rate
Cm_u = 0;                                                       % Change in pitching moment due to change in forward velocity (zero due to steady flight)
CL_deflection = CL/10;                                          % Cl due to control surface deflection
CL_deflection_h = CL_dele/10;                                   % Cl of Horizontal Stabilizer due to control surface deflection
Cm_q = -2*CL_alpha_h*eta_h*V_h*(X_cg_h - X_np);                 % Pitching Moment Coefficient with respect to change in pitch
Cm_dele = -CL_alpha_h*eta_h*V_h*CL_deflection_h*(X_cg_h - X_np);% Pitching Moment Coefficient with respect to change in elevator deflection
Cm = Cm_w - CL_alpha_h*eta_h*(S_h/S)*(X_cg_h-X_cg);             % Pitching Moment Coefficient
```

Development of Forces for Longitudinal Stability Derivatives

```
X = -Cd*q*S;            % Aerodynamic X-force during steady flight
MA = Cm*q*S*c;          % Total Aerodynamic Moment
```

Longitudinal Stability Derivatives

U1 = 590.73; % Thrust Velocity [ft/s] - [350 kts]

```
Xu = (-2*Cd*q*S)/(Weight);                      % Change in X-force caused by change in forward velocity [1/s]
Xalpha = (-Cdi + CL*q*S)/(Weight/g);            % Change in X-force caused by change in angle of attack [ft/s^2]
Xdele = 0;                                      % Change in X-force caused by elevator deflection (assumed to be zero in steady flight) [ft/s^2]
Zu = (-2*CL*q*S)/((Weight/g)*U1);               % Add [1/s]
Zdele = (-CL_alpha-Cd*q*S)/(Weight/g);          % Add [ft/s^2]
Zalpha = (-CL_dele*q*S)/(Weight/g);             % Add [ft/s^2]
Mu = ((Cm_u + 2*Cm)*q*S*c)/(Iyy*U1);            % Add [1/s^2]
Malpha = (q*S*c*Cm_alpha)/Iyy;                  % Add [1/s^2]
Malpha_dot = (q*S*(c^2)*Cm_alphadot)/(2*Iyy*U1); % Add [1/s]
Mq = (q*S*(c^2)*Cm_q)/(2*Iyy*U1);               % Add [1/s]
Mdele = (q*S*c*Cm_dele)/Iyy;                    % Change in Pitching Moment due to elevator defleciton [1/s^2]
```

Lateral Aircraft Constants and Assumptions

NOTE: Due to the many variables and coupling effects witnessed in Lateral-Directional aircraft dynamics and controls, the difficulty of accurately estimating the associated derivatives, without the use of CFD, has been mentioned in many of the referenced source documents and as a result, is likely reflected to an extent in this section of analysis.

```
Ixx_prime = Ixx*cos(alpha).^2 + Izz*sin(alpha).^2 - Ixz*sin(2*alpha);          % Ixx' for Lateral Stability Calculations
Iyy_prime = Iyy;                                                               % Iyy' for Lateral Stability Calculations
Izz_prime = Ixx*sin(alpha).^2 + Izz*cos(alpha).^2 + Ixz*sin(2*alpha);          % Izz' for Lateral Stability Calculations
Ixz_prime = 0.5*(Ixx-Iyy)*sin(2*alpha) + Ixz*cos(2*alpha);                     % Ixz' for Lateral Stability Calculations
beta = 10;                                                                     % Sideslip perturbation assumption [deg] (Roskam Part VII, Chapter 3, Page 94)
beta_rad = deg2rad(beta);                                                      % Sideslip perturbation assumption [rad]
r = 20;                                                                        % Roll perturbation assumption [deg] (Roskam Part VII, Chapter 3, Page 94)
r_rad = deg2rad(r);                                                            % Roll perturbation assumption [rad]
a_inc = sqrt(alpha_rad^2 + beta_rad^2);                                        % Angle of inclination [rad] (USAF Stability and Control DATCOM, 1978)
rho_sidewash = 0;                                                              % Sidewash angle (assumed to be zero for steady flight) [deg]
dela = 10;                                                                     % Aileron deflection angle [deg]
V_v = 0.043;                                                                   % Vertical Tail volume ratio
q_v = 0.5*rho*V_v*(U1^2);                                                      % Dynamic pressure at the horizontal stabilizer
Z_v = -3.08;                                                                   % Distance between Vertical Stabilizer Span MAC and aircraft longitudinal axis
rudder_deflection = 5*(pi/180);                                                % Rudder deflection angle used for lift curve slope [rad/s] (XFLR5 analysis)
dCl_dp = (-90*pi/180)/2.5;                                                     % Required Roll rate performance (From Mil-F-8785C)
dCl_dr = (20*pi/180);                                                          % Required Yaw rate performance (From Mil-F-8785C)
zeta = -(Mq+Malpha_dot)/(2*sqrt(-Iyy*Malpha));                                 % Damping Coefficient
w_n = sqrt(-Malpha/Iyy);                                                       % Natural Frequency [rad/s] - Falls within Level 3 Flight Category (From Mil-F-8785C
```

```
p_ss = (90*pi/180)/1.7;                                                  % Steady-State Roll Rate requirement given by MIL-F-008785A Page 309. [rad/s]
p_hat_ss = (p_ss*b)/(2*U1);                                              % Non-dimensional Steady-State Roll Rate
body_ratio_front = S_body_fore/S;                                        % Frontal body area ratio to wing area
body_ratio_side = S_body_lat/S;                                          % Side body area ratio to wing area
CY_b = (-body_ratio_front*2*beta_rad) - Cd_0*body_ratio_side*beta_rad*a_inc;   % Side force contribution coefficient (USAF Stability and Control DATCOM, eq. 5.2.1.
CY_w = ((-Cd_0*c_root*b_body)/S)*(beta_rad*(beta_rad-a_inc));            % Side force contribution from wing influenced by the body (USAF Stability and Contr
CY_wb = CY_b + CY_w;                                                     % Wing body side force contribution coefficient (USAF Stability and Control DATCOM,
```

Environmental Conditions involving Lateral-Directional Stability

```
Fy_body = q*S_body_lat*(beta/U1)*Cd_0;     % Side Force at airspeed U1 involving sideslip angle [lbf]
Fy_vt = -q*S_v*(beta/U1)*dele0;            % Side Force induced by vertical stabilizer [lbf]
Fy = Fy_body - Fy_vt;                      % Total Side Force
Yv = Fy/(q);                               % Side Force Dimensionless Derivative
up_grad = 0;                               % Upwash Gradient for Nacelles (assumed zero due to zero degree AoA)
```

Lateral-Directional Stability Coefficients

```
eta_v = q_v/q;                                                  % Dynamic Pressure Ratio at Vertical Stabilizer
delr = (1.5-1)/(14.15799-9.19251);                              % Rudder Lift Curve Slope
CL_alpha_v = dele0*(-beta + rho_sidewash) + delr*rudder_deflection;   % Coefficient of lift of Vertical Stabilizer given rudder deflection
Cl_w = Lift/(q*b*(0.5*S));                                      % Coefficient of the Rolling Moment of the Main Wing
l_w = q*S*b*Cl_w;                                               % Main Wing Rolling Moment
Cl_vt = (Thrust*sin(beta_rad))/(q*b_v*(0.5*S_v));               % Coefficient of the Rolling Moment of the Vertical Stabilizer
l_vt = -q_v*S_v*b_v*Cl_vt;                                      % Vertical Stabilizer Rolling Moment
Cn_beta = dele0*q_v*((S_v*L_v)/(S*b));                          % Yaw stiffness derivative
% Cn_beta = (X_ac_v - X_cg)*CL_alpha_v;                         % Not sure this one is correct.
Cn_r = -2*V_v*CL_alpha_v*(l_vt/b);                              % Yaw damping derivative
Cn_p = -2*(((m_dot_inlet^2)*L_engine)/(A_inlet*rho*q*S*c))*(1+up_grad);   % Pitching moment contribution to yaw coefficient due to engine thrust (Roskam Eq. 3.240)
Cn_delr = -delr*q_v*((S_v*L_v)/(S*b));                          % Coefficient of the Yawing Moment regarding Rudder deflection
Cn_dela = 0;                                                    % Coefficient of the Yawing Moment regarding aileron deflection [Negligible]
Cn = Cn_beta*beta + Cn_delr * rudder_deflection;                % Coefficient of the Yawing Moment
Cl = Cl_w + Cl_vt;                                              % Coefficient of the Rolling Moment
n_w = q*S*b*Cn;                                                 % Main Wing Yawing Moment
Cl_beta_v = dele0*q*((S_v*L_v)/(S*b))*(Z_v/L_v);                % Vertical Tail contribution to Side-slip roll moment
Cl_beta = Cl_beta_v;                                            % Coefficient of Rolling Moment regarding Side-Slip
Cl_p = (2*U1/b)*dCl_dp*(pi/180);                                % Roll damping derivative
Cl_r = (2*U1/b)*dCl_dr*(pi/180);                                % Rolling moment due to Yaw Rate
Cl_dp = (2*U1/b)*dCl_dp;
Cl_dr = (2*U1/b)*dCl_dr;
Cl_delr = (p_ss*b*Cl_dr)/(dela*2*U1);
Cl_dela = (p_ss*b*Cl_dp)/(dela*2*U1);
Cy_p = 0;
Cy_beta = CY_wb/beta_rad;                                       % Coefficient of Side force induced by Sideslip (USAF Stability and Control DATCOM, 5.2.1.2-
Cy_r = CL_alpha_v*((2*L_v)/b)*eta_v*(S_v/S);                    % Coefficient of Side force induced by Yaw Rate (Roskam eq. 3.189)
Cy_delr = CL_alpha_v*rudder_deflection*q_v*(S_v/S);             % Coefficient of Side force induced by Rudder Deflection (Roskam eq. 3.79)
Cy_dela = 0;
%Cy = Cy_dela when dela = 0
```

Lateral-Directional Stability Derivatives

```
Theta1 = 5;                             % [deg]
Lp = ((q*S*b^2)/(Ixx_prime*2*U1))*Cl_p;       % [1/s]
Lb = ((q*S*b)/(Ixx_prime))*Cl_beta;           % [1/s^2]
Lr = ((q*S*b^2)/(Ixx_prime*2*U1))*Cl_r;       % [1/s]
Ldelr = ((q*S*b)/(Ixx_prime))*Cl_delr;        % [1/s^2]
Ldela = ((q*S*b)/(Ixx_prime))*-Cl_dela;       % [1/s^2]
Yp = (q*S*b)/(2*(Weight/g)*U1)*Cy_p;          % Should be zero due to no change in pitch attitude [ft/s]
Yb = ((q*S)/(Weight/g))*Cy_beta;              % [ft/s^2]
Yr = (q*S*b)/(2*(Weight/g)*U1)*Cy_r;          % [ft/s]
Ydelr = ((q*S)/(Weight/g))*(-1*Cy_delr);      % [ft/s^2]
Ydela = ((q*S)/(Weight/g))*Cy_dela;           % Negligble affect on side force (Roskam Chapter 3) [ft/s^2]
Np = ((q*S*b^2)/(Izz_prime*2*U1))*Cn_p;       % [1/s]
Nb = ((q*S*b)/Izz_prime)*Cn_beta;             % [1/s^2]
Nr = ((q*S*b^2)/(Izz_prime*2*U1))*Cn_r;       % [1/s]
Ndelr = ((q*S*b)/Izz_prime)*Cn_delr;          % [1/s^2]
Ndela = ((q*S*b)/Izz_prime)*Cn_dela;          % Negligible affect from aileron on Yaw force [1/s^2]
```

Establish State Space Model For Longitudinal Stability Assessment

```
A_long = [Xu                        Xalpha                          0               -g,
          Zu/U1                     Zalpha/U1                       1                0,
          (Mu + Malpha_dot*Zu/U1)   (Malpha + Malpha_dot*Zalpha/U1) (Mq + Malpha_dot)  0,
          0                         0                               1                0];
B_long = [Xdele,
          Zdele/U1,
          (Mdele + Malpha_dot*Zdele/U1),
          0];
C_long_theta= [0, 0, 0, 1];    % This format will output pitch angle Theta
D_long = 0;

% Display A and B Matricies
```

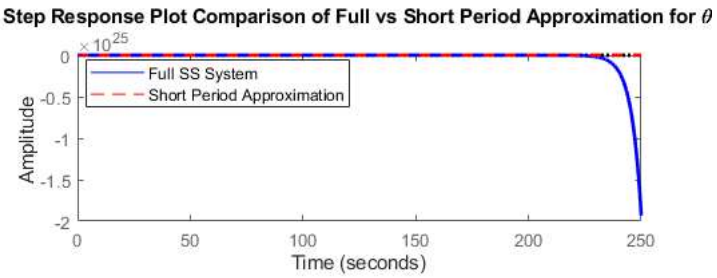
```
disp(A_long);
disp(B_long);

% Define the entire State-Space system for alpha output
longitudinal_ss_alpha = ss(A_long,B_long,C_long_theta,D_long);

% Define Transfer function from State-Space System for Theta output
[TF_de2theta_num,TF_de2theta_denom] = ss2tf(A_long,B_long,C_long_theta,D_long);
```

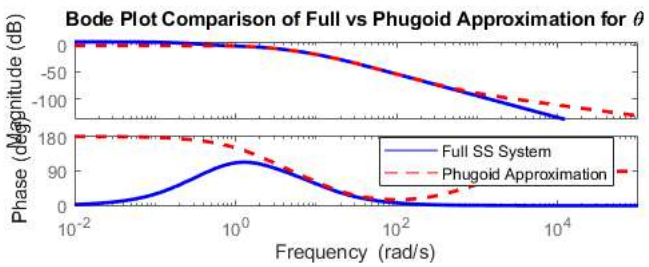
```
      -1.041        32.17            0       -32.17
  -0.00018438     -0.75331            1            0
     -0.33699      -14.739      -12.903            0
            0            0            1            0
            0
    -0.028368
     -19.986
            0
```

Short Period Approximation

Define the state-space model for the Short Period Approximation

```
A_long_sp = [Zalpha/U1                        1,
            (Malpha + Malpha_dot*Zalpha/U1)  (Mq + Malpha_dot)];
B_long_sp = [Zdele/U1,
             Mdele+((Malpha_dot*Zdele)/U1)];
C_long_sp = [1,...
             0];
D_long_sp = 0;

% Define the short period state-space system for alpha output
longitudinalshort_ss_alpha = ss(A_long_sp,B_long_sp,C_long_sp,D_long_sp);

% Plot comparison of the Step responses of the Full vs Short Period
% Approximated systems
figure(1),
subplot(2,1,1);
step(longitudinal_ss_alpha,'b');
hold on
step(longitudinalshort_ss_alpha,'r--')
legend('Full SS System','Short Period Approximation','location','best');
title('Step Response Plot Comparison of Full vs Short Period Approximation for \theta');
set(findall(gcf,'type','line'),'linewidth',2);

% Plot comparison of the Frequency esponses of the Full vs Short Period
% NEEDS WORK
% Approximated systems
figure(1),
subplot(2,1,2);
bode(longitudinal_ss_alpha,'b');
hold on
bode(longitudinalshort_ss_alpha,'r--')
legend('Full SS System','Phugoid Approximation','location','best');
title('Bode Plot Comparison of Full vs Phugoid Approximation for \theta');
set(findall(gcf,'type','line'),'linewidth',2);
```

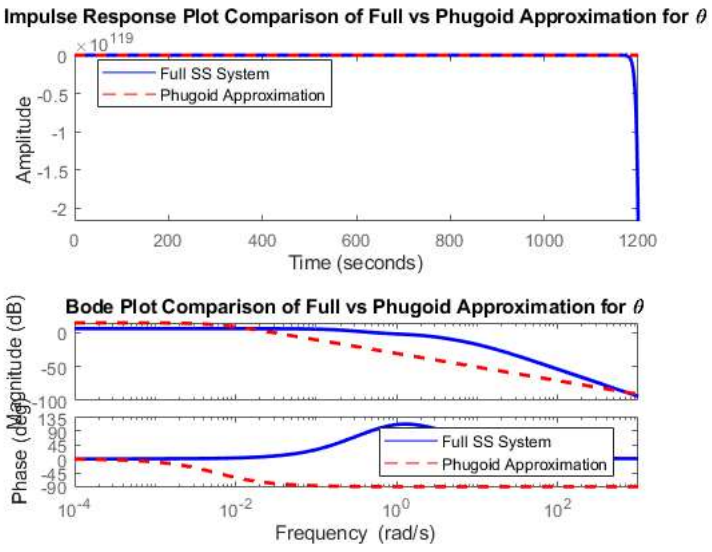




Phugoid Approximation

Define State-Space model for phugoid approximation

```
A_long_phugoid = [Xu     -g,
                  (-Zu/U1) 0];
B_long_phugoid = [Xdele,
                  (-Zdele/U1)];
C_long_phugoid_theta = [0,1];     % For pitch angle output
D_long_phugoid = 0;

% Define C Matrix for full system to evaluate pitch angle theta
C_long_theta = [0, 0, 0, 1];

% Define the full state-space system for theta output
longitudinal_ss_theta = ss(A_long,B_long,C_long_theta,D_long);

% Define the phugoid state-space system for theta output
longitudinalphugoid_ss_theta = ss(A_long_phugoid,B_long_phugoid,C_long_phugoid_theta,D_long_phugoid);

% Plot comparison of the Step esponses of the Full vs Phugoid
% Approximated systems
figure(2),
subplot(2,1,1);
impulse(longitudinal_ss_theta,'b');
hold on
impulse(longitudinalphugoid_ss_theta,'r--')
legend('Full SS System','Phugoid Approximation','location','best');
title('Impulse Response Plot Comparison of Full vs Phugoid Approximation for \theta');
set(findall(gcf,'type','line'),'linewidth',2);

% Plot comparison of the Frequency esponses of the Full vs Phugoid
% Approximated systems
figure(2),
subplot(2,1,2);
bode(longitudinal_ss_theta,'b');
hold on
bode(longitudinalphugoid_ss_theta,'r--')
legend('Full SS System','Phugoid Approximation','location','best');
title('Bode Plot Comparison of Full vs Phugoid Approximation for \theta');
set(findall(gcf,'type','line'),'linewidth',2);
```



Establish Lateral-Direcitonal State-Space Model

A relates to _ (phi), _ (p), sideslip angle (beta), _ (r), and _ (psi)

```
A_lat = [0                  1       0       0           0,
         0                  Lp      Lb      Lr          0,
         (g*cos(Theta1)/U1) (Yp/U1) (Yb/U1) ((Yr/U1)-1) 0,
         0                  Np      Nb      Nr          0,
         0                  0       0       1           0];
% B relates to rudder deflection (delr) and aileron deflection (dela)
B_lat = [0          0,
        Ldelr       Ldela
        (Ydelr/U1) (Ydela/U1)
        Ndelr       Ndela
        0           0];

% Establish C and D matricies for Roll angle and Lat-Dir TF
C_lat_phi = [1 0 0 0 0];
D_lat = [0 0];
```

```
% Lateral-Directional State-Space System
lateral_ss_phi = ss(A_lat,B_lat,C_lat_phi,D_lat);

% Create TF from Lat-Dir SS System
[TF_da2phi_num,TF_da2phi_denom] = ss2tf(A_lat,B_lat(:,2),C_lat_phi,D_lat(:,2));
```

Create Transfer Function for Heading Evaluation

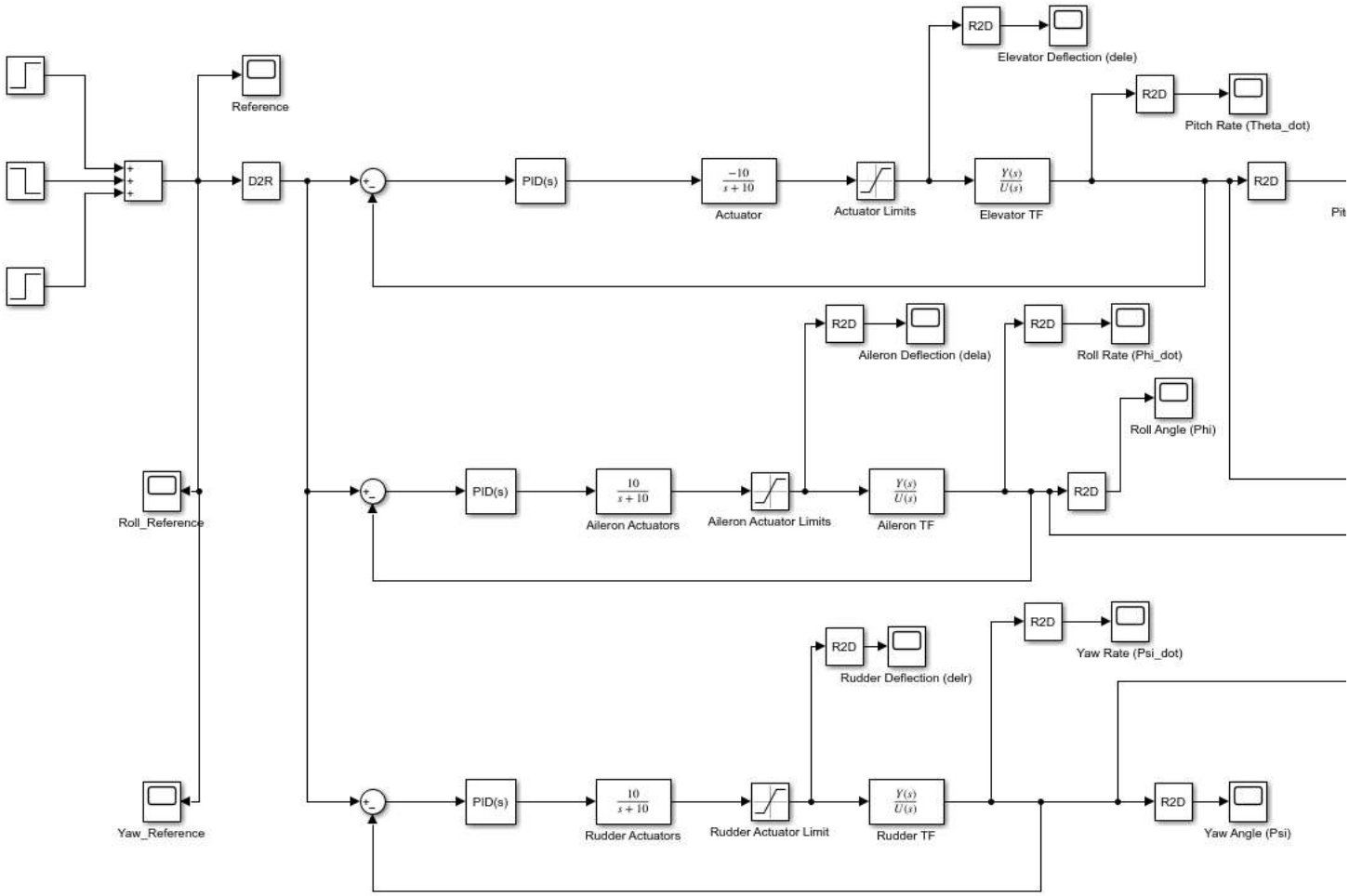
```
C_lat_psi = [0 0 0 0 1];

% Lateral-Directional State-Space System for Psi
lateral_ss_psi = ss(A_lat,B_lat,C_lat_psi,D_lat);

% Create TF from Lat-Dir SS System
[TF_da2psi_num,TF_da2psi_denom] = ss2tf(A_lat,B_lat(:,2),C_lat_psi,D_lat(:,2));
```

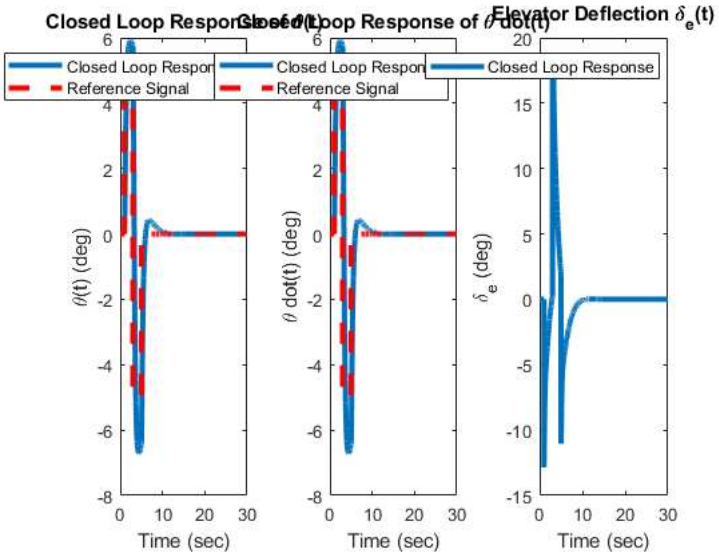
Plot Simulink Pitch Response

Run Simulink system for Open Loop simulation

```
t_f = 30;
open_system("BaseAircraftControl_Simulink.slx")
elevator_eval = sim("BaseAircraftControl_Simulink.slx");
figure(4),          % Theta response
subplot(1,3,1);
plot(elevator_eval.Theta_CL(:,1), elevator_eval.Theta_CL(:,2));
hold on
plot(elevator_eval.Reference(:,1), elevator_eval.Reference(:,2),'r--');
legend('Closed Loop Response', 'Reference Signal');
xlabel('Time (sec)');
ylabel('\theta(t) (deg)');
title('Closed Loop Response of \theta(t)');
set(gca,'fontsize',10);
set(findall(gcf,'type','line'),'linewidth',3);

figure(4),          % Theta_dot response
subplot(1,3,2);
plot(elevator_eval.Theta_dot(:,1), elevator_eval.Theta_dot(:,2));
hold on
plot(elevator_eval.Reference(:,1), elevator_eval.Reference(:,2),'r--');
legend('Closed Loop Response', 'Reference Signal');
xlabel('Time (sec)');
ylabel('\theta dot(t) (deg)');
title('Closed Loop Response of \theta dot(t)');
set(gca,'fontsize',10);
set(findall(gcf,'type','line'),'linewidth',3);

figure(4),          % Elevator Deflection
subplot(1,3,3);
plot(elevator_eval.dele(:,1), elevator_eval.dele(:,2));
legend('Closed Loop Response', 'Reference Signal');
xlabel('Time (sec)');
ylabel('\delta_e (deg)');
title('Elevator Deflection \delta_e(t)');
set(gca,'fontsize',10);
set(findall(gcf,'type','line'),'linewidth',3);
```

```
Warning: Ignoring extra legend entries.
```

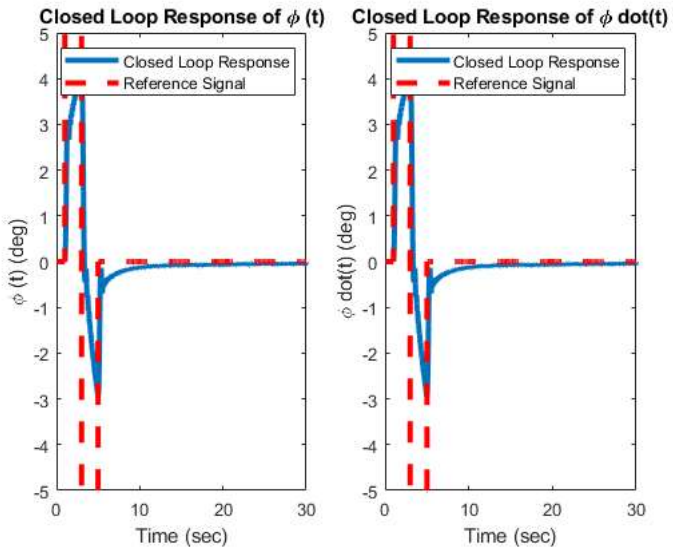




Plot Simulink Roll Response

```
aileron_eval = sim("BaseAircraftControl_Simulink.slx");
figure(5),         % Phi response
subplot(1,2,1);
plot(aileron_eval.Phi(:,1), aileron_eval.Phi(:,2));
hold on
plot(aileron_eval.Reference(:,1), aileron_eval.Reference(:,2),'r--');
legend('Closed Loop Response', 'Reference Signal');
xlabel('Time (sec)');
ylabel('\phi (t) (deg)');
title('Closed Loop Response of \phi (t)');
set(gca,'fontsize',10);
set(findall(gcf,'type','line'),'linewidth',3);
```

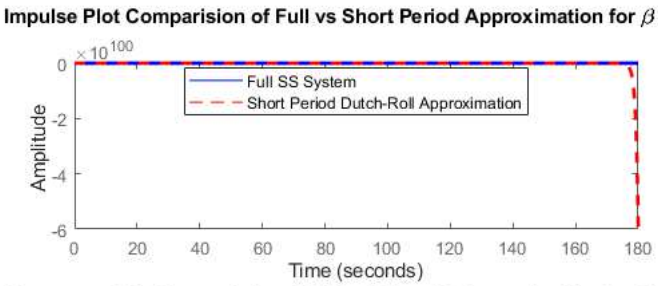
```
figure(5),          % Phi_dot response
subplot(1,2,2);
plot(aileron_eval.Phi_dot(:,1), aileron_eval.Phi_dot(:,2));
hold on
plot(aileron_eval.Reference(:,1), aileron_eval.Reference(:,2),'r--');
legend('Closed Loop Response', 'Reference Signal');
xlabel('Time (sec)');
ylabel('\phi dot(t) (deg)');
title('Closed Loop Response of \phi dot(t)');
set(gca,'fontsize',10);
set(findall(gcf,'type','line'),'linewidth',3);
```

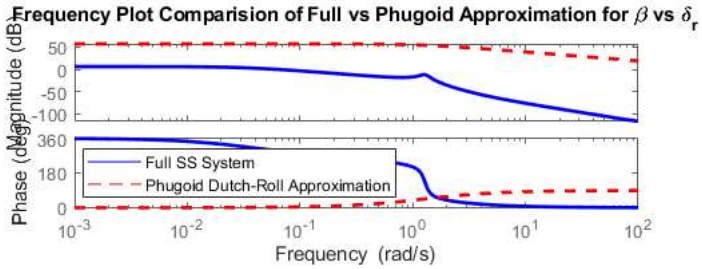


Dutch-Roll Mode Approximation

Output C Matrix for Dutch-Roll Approximation (p)

```
C_lat_p = [0 0 0 1 0];

% Create TF from Lat-Dir SS System
[r_numerator_delr,r_denominator_delr] = ss2tf(A_lat,B_lat,C_lat_p,D_lat,1);
[r_numerator_dela,r_denominator_dela] = ss2tf(A_lat,B_lat,C_lat_p,D_lat,2);
dutchroll_numerator_r = tf(r_numerator_delr,r_denominator_delr);
dutchroll_denominator_delr = tf(r_numerator_dela,r_denominator_dela);

% Roll Approximation Transfer Function
dutchroll_approx_TF = tf([Ndelr*(U1 -Yb) + Nb*Ydelr], [1 -(Nr +(Yb/U1) + (Nb+(1/U1)*(Yb*Nr-Nb*Yr)))]);

% Graph and compare Full SS response vs. Short Period Approximation for
% Dutch-Roll Mode
figure(6),
subplot(2,1,1);
impulse(dutchroll_denominator_delr,'b');
hold on
impulse(dutchroll_approx_TF,'r--');
legend('Full SS System','Short Period Dutch-Roll Approximation','location','best');
title('Impulse Plot Comparision of Full vs Short Period Approximation for \beta vs \delta_r')
set(findall(gcf,'type','line'),'linewidth',2)

figure(6),
subplot(2,1,2);
bode(dutchroll_denominator_delr,'b');
hold on
bode(dutchroll_approx_TF,'r--');
legend('Full SS System','Phugoid Dutch-Roll Approximation','location','best');
title('Frequency Plot Comparision of Full vs Phugoid Approximation for \beta vs \delta_r')
set(findall(gcf,'type','line'),'linewidth',2)
```

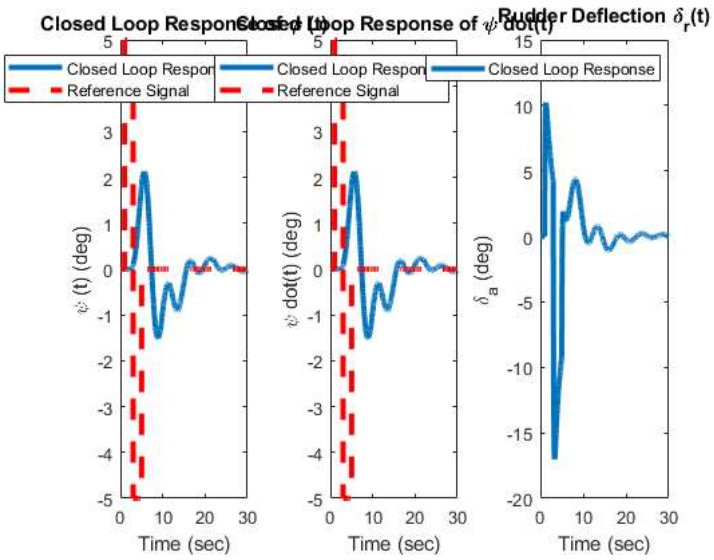




Plot Simulink Yaw Response (Heading Angle)

```
rudder_eval = sim("BaseAircraftControl_Simulink.slx");
figure(7),          % Phi response
subplot(1,3,1);
plot(rudder_eval.Psi(:,1), rudder_eval.Psi(:,2));
hold on
plot(rudder_eval.Reference(:,1), rudder_eval.Reference(:,2),'r--');
legend('Closed Loop Response', 'Reference Signal');
xlabel('Time (sec)');
ylabel('\psi (t) (deg)');
title('Closed Loop Response of \psi (t)');
set(gca,'fontsize',10);
set(findall(gcf,'type','line'),'linewidth',3);

figure(7),          % Phi_dot response
subplot(1,3,2);
plot(rudder_eval.Psi_dot(:,1), rudder_eval.Psi_dot(:,2));
hold on
plot(rudder_eval.Reference(:,1), rudder_eval.Reference(:,2),'r--');
legend('Closed Loop Response', 'Reference Signal');
xlabel('Time (sec)');
ylabel('\psi dot(t) (deg)');
title('Closed Loop Response of \psi dot(t)');
set(gca,'fontsize',10);
set(findall(gcf,'type','line'),'linewidth',3);

figure(7),          % Aileron Deflection
subplot(1,3,3);
plot(rudder_eval.delr(:,1), rudder_eval.delr(:,2));
legend('Closed Loop Response', 'Reference Signal');
xlabel('Time (sec)');
ylabel('\delta_a (deg)');
title('Rudder Deflection \delta_r(t)');
set(gca,'fontsize',10);
set(findall(gcf,'type','line'),'linewidth',3);
```

```
Warning: Ignoring extra legend entries.
```



Published with MATLAB® R2024a

Contents

AE 295 Base Aircraft Stability and Controls

In accordance with the requirements for completion of AE295 Design of a Military Class 4 ISR Aircraft for Comparison with a Tailed Variant By: Tripp Selvig

Base Environmental and Aircraft Constants

```
g = 32.17;                              % Gravity [ft/s^2]
h = 30000;                              % Cruise Altitude [ft]
rho = 8.91E-4;                          % Air density at altitude [slugs/ft^3]
u_inf = 3.107E-7;                       % Dynamic Viscosity [slug/(ft*s)]
U1 = 590.73;                            % Thrust Velocity [ft/s] - [350 kts]
q = 0.5*rho*U1^2;                       % Dynamic pressure
L = 43.515;                             % Fuselage length [ft]
b = 39.56;                              % Wingspan [ft]
S = 380.81;                             % Wing area [ft^2]
c = 9.844;                              % Wing Mean Aerodynamic Chord (MAC) [ft]
c_root = 16.23;                         % Wing root chord [ft]
c_tip = 3.68;                           % Wingtip chord [ft]
AR = b^2/S;                             % Aspect Ratio [4.1096 unitless]
S_h = 108;                              % Horizontal Stabilizer wing area [ft^2]
b_h = 18;                               % Horizontal Stabilizer wingspan [ft]
AR_h = b_h^2/S_h;                       % Horizontal Stabilizer Aspect Ratio [3 unitless]
L_h = 18.5;                             % Moment arm of the Horizontal Stabilizer [ft]
L_v = 0;                                % Moment arm of the Vertical Stabilizer [ft]
S_v = 0;                                % Vertical Stabilizer wing area [ft^2]
b_v = 0;                                % Vertical Stabilizer wingspan [ft]
AR_v = b_v^2/S_v^2;                     % Vertical Stabilizer Aspect Ratio [3 unitless]
b_dragr = (b/2)*0.25;                   % Drag Rudder span [ft]
c_dragr = 0.2*c_tip;                    % Drag Rudder chord [ft]
y_dragr = (b/2)*0.825;                  % Wingspan location of drag rudder MAC [ft]
S_body_fore = 15.622;                   % Frontal surface area of fuselage [ft^2]
S_body_lat = 221.901;                   % Lateral surface area of fuselage [ft^2]
b_body = 5.1;                           % Average body diameter at wing root [ft]
S_a = (0.25*b/2)*(0.2*c);               % Drag rudder (aileron) surface area [ft^2]
L_a = 16.3185;                          % Moment arm of drag rudders [ft]
e0 = 1.78*(1-0.045*AR^0.68) - 0.64;     % Oswald Efficiency Factor
e = 1/(1+(1/(AR*e0)));                  % Span Efficiency Factor
e0_h = 1.78*(1-0.045*AR_h^0.68) - 0.64; % Oswald Efficiency Factor
e_h = 1/(1+(1/(AR_h*e0_h)));            % Span Efficiency Factor
Re_L = (rho*U1*S)/u_inf;                % Reynolds Number
Weight = 13711;                         % Aircraft weight [lb]
Lift = 13711;                           % Lift = Weight of the aircraft durings steady flight [lb]
Thrust = 7200;                          % Engine Thrust from two J85-GE-21 engines [lbf]
Thrust_AB = 10000;                      % Afterburning Thrust [lbf]
Drag = 7200;                            % Aircraft drag during steady flight given by Thrust = Drag [lbf]
Drag_AB = 10000;                        % Aircraft drag during steady flight with afterburners engaged [lbf]
Cd_0 = 0.0113;                          % Clean Cd_0 calculated with Roskam method
Ixx = 2.5917e+04;                       % Aircraft Roll moment of inertia (Solidworks MOI) [lb/ft^2]
Iyy = 3.2336e+04;                       % Aircraft Pitch moment of inertia (Solidworks MOI) [lb/ft^2]
Izz = 5.7784e+04;                       % Aircraft Yaw moment of inertia (Solidworks MOI) [lb/ft^2]
Ixz = 0;                                % Aircraft Ixz
V_h = 0.191;                            % Horizontal Stabilizer Volume Coefficient
q_h = 0.5*rho*V_h*(U1^2);               % Dynamic pressure at the horizontal stabilizer
```

```
X_cg = 22.35;                           % Aircraft CG [ft]
X_np = 24.553;                          % Aircraft Neutral Point [ft]
AoA_Tail = 0;                           % Horizontal Tail incidence angle [deg]
X_cg_h = X_cg + L_h;                    % Horizontal Tail CG Location [ft]
X_ac_v = X_cg + L_v;                    % Vertical Stabilizer AC Location [ft]
L_engine_long = (L-X_cg);               % Longitudinal Moment arm of the engines [ft]
L_engine_lat = 0.866;                   % Lateral Moment arm of each engine [ft]
A_inlet = pi*0.866^2;                   % Engine inlet area [ft^2]
m_dot_inlet = rho*U1*(A_inlet);         % Mass Flow Rate through each engine
V_inlet = m_dot_inlet/(A_inlet*rho);    % Engine Inlet flow velocity [ft/s]
```

Aircraft Flight Regime Assumptions

```
alpha = 0;                  % Steady flight [deg]
alpha_rad = deg2rad(alpha);     % Steady flight [rad]
```

Longitudinal Stability Coefficients

```
a0 = (0.7046326-0.04012974)/6;                                  % Lift slope calculated between 6 and 0 degrees AoA respectively
dele0 = (1.5 - 0)/(14.16);                                      % Change in lift due to elevator deflection
eta_h = q_h/q;                                                  % Dynamic pressure ratio at horizontal stabilizer
CL = Lift/(q*S);                                                % Coefficient of lift
Cd = Drag/(q*S) + CL^2/(pi*e*AR);                               % Coefficient of drag for entire aircraft
Cm_w = CL*(X_np-X_cg)/c;                                        % Pitching Moment Coefficient provided by the Main Wing
Cd_AB = Drag_AB/(q*S);                                          % Coefficient of drag with afterburners engaged
Cdi = (CL^2)/(pi*exp(1)*AR);                                    % Lift induced coefficient of drag
ai = CL/(pi*exp(1)*AR);                                         % Induced AoA
CL_alpha = (a0/(1+(a0/(pi*e*AR))))*(180/pi);                    % Coefficient of lift with respect to alpha (CHECK FOR DEGREES/RADIANS)
CL_alpha_h = (dele0/(1+(dele0/(pi*e_h*AR_h))))*(180/pi);        % Coefficient of lift from the Horizontal Stabilizer with respect to alpha (CHECK FOR DEGREES/RA
CL_dele = (dele0/(1+(dele0*(pi*e*AR_h))))*(180/pi);             % Coefficient of lift with respect to elevator deflection (CHECK FOR DEGREES/RADIANS)
Cm_alpha = CL_alpha*(X_cg - X_np)/c;                            % Pitching Moment Coefficient with respect to change in alpha
Cm_alphadot = -2*q_h*V_h*CL_dele*(X_np - X_cg);                 % Pitching Moment Coefficient with respect to change in alpha rate
Cm_u = 0;                                                       % Change in pitching moment due to change in forward velocity (zero due to steady flight)
CL_deflection = CL/10;                                          % Cl due to control surface deflection
CL_deflection_h = CL_dele/10;                                   % Cl of Horizontal Stabilizer due to control surface deflection
Cm_q = -2*CL_alpha_h*eta_h*V_h*(X_cg_h - X_np);                 % Pitching Moment Coefficient with respect to change in pitch
Cm_dele = -CL_alpha_h*eta_h*V_h*CL_deflection_h*(X_cg_h - X_np); % Pitching Moment Coefficient with respect to change in elevator deflection
Cm = Cm_w - CL_alpha_h*eta_h*(S_h/S)*(X_cg_h-X_cg);             % Pitching Moment Coefficient
```

Development of Forces for Longitudinal Stability Derivatives

```
X = -Cd*q*S;            % Aerodynamic X-force during steady flight
MA = Cm*q*S*c;          % Total Aerodynamic Moment
```

Longitudinal Stability Derivatives

U1 = 590.73; % Thrust Velocity [ft/s] - [350 kts]

```
Xu = (-2*Cd*q*S)/(Weight);                      % Change in X-force caused by change in forward velocity [1/s]
Xalpha = (-Cdi + CL*q*S)/(Weight/g);            % Change in X-force caused by change in angle of attack [ft/s^2]
Xdele = 0;                                      % Change in X-force caused by elevator deflection (assumed to be zero in steady flight) [ft/s^2]
Zu = (-2*CL*q*S)/((Weight/g)*U1);               % Add [1/s]
Zdele = (-CL_alpha-Cd*q*S)/(Weight/g);          % Add [ft/s^2]
Zalpha = (-CL_dele*q*S)/(Weight/g);             % Add [ft/s^2]
Mu = ((Cm_u + 2*Cm)*q*S*c)/(Iyy*U1);            % Add [1/s^2]
Malpha = (q*S*c*Cm_alpha)/Iyy;                  % Add [1/s^2]
Malpha_dot = (q*S*(c^2)*Cm_alphadot)/(2*Iyy*U1); % Add [1/s]
Mq = (q*S*(c^2)*Cm_q)/(2*Iyy*U1);               % Add [1/s]
Mdele = (q*S*c*Cm_dele)/Iyy;                    % Change in Pitching Moment due to elevator defleciton [1/s^2]
```

Lateral Aircraft Constants and Assumptions

NOTE: Due to the many variables and coupling effects witnessed in Lateral-Directional aircraft dynamics and controls, the difficulty of accurately estimating the associated derivatives, without the use of CFD, has been mentioned in many of the referenced source documents and as a result, is likely reflected to an extent in this section of analysis.

```
Ixx_prime = Ixx*cos(alpha).^2 + Izz*sin(alpha).^2 - Ixz*sin(2*alpha);   % Ixx' for Lateral Stability Calculations
Iyy_prime = Iyy;                                                        % Iyy' for Lateral Stability Calculations
Izz_prime = Ixx*sin(alpha).^2 + Izz*cos(alpha).^2 + Ixz*sin(2*alpha);   % Izz' for Lateral Stability Calculations
Ixz_prime = 0.5*(Ixx-Iyy)*sin(2*alpha) + Ixz*cos(2*alpha);              % Ixz' for Lateral Stability Calculations
beta = 10;                                                              % Sideslip perturbation assumption [deg] (Roskam Part VII, Chapter 3, Page 94)
beta_rad = deg2rad(beta);                                               % Sideslip perturbation assumption [rad]
r = 20;                                                                 % Roll perturbation assumption [deg] (Roskam Part VII, Chapter 3, Page 94)
r_rad = deg2rad(r);                                                     % Roll perturbation assumption [rad]
a_inc = sqrt(alpha_rad^2 + beta_rad^2);                                 % Angle of inclination [rad] (USAF Stability and Control DATCOM, 1978)
rho_sidewash = 0;                                                       % Sidewash angle (assumed to be zero for steady flight) [deg]
dela = 10;                                                              % Aileron deflection angle [deg]
V_v = 0.043;                                                            % Vertical Tail volume ratio
q_v = 0.5*rho*V_v*(U1^2);                                               % Dynamic pressure at the horizontal stabilizer
Z_v = -3.08;                                                            % Distance between Vertical Stabilizer Span MAC and aircraft longitudinal axis
drag_rudder_deflection = 15*(pi/180);                                   % Rudder deflection angle used for lift curve slope [rad/s] (XFLR5 analysis)
```

```
dCl_dp = (-90*pi/180)/2.5;                                                  % Required Roll rate performance (From Mil-F-8785C)
dCl_dr = (20*pi/180);                                                       % Required Yaw rate performance (From Mil-F-8785C)
zeta = -(Mq+Malpha_dot)/(2*sqrt(-Iyy*Malpha));                              % Damping Coefficient
w_n = sqrt(-Malpha/Iyy);                                                    % Natural Frequency [rad/s] - Falls within Level 3 Flight Category (From Mil-F-8785C
p_ss = (90*pi/180)/1.7;                                                     % Steady-State Roll Rate requirement given by MIL-F-008785A Page 309. [rad/s]
p_hat_ss = (p_ss*b)/(2*U1);                                                 % Non-dimensional Steady-State Roll Rate
body_ratio_front = S_body_fore/S;                                           % Frontal body area ratio to wing area
body_ratio_side = S_body_lat/S;                                             % Side body area ratio to wing area
CY_b = (-body_ratio_front*2*beta_rad) - Cd_0*body_ratio_side*beta_rad*a_inc;    % Side force contribution coefficient (USAF Stability and Control DATCOM, eq. 5.2.1.
CY_w = ((-Cd_0*c_root*b_body)/S)*(beta_rad*(beta_rad-a_inc));               % Side force contribution from wing influenced by the body (USAF Stability and Contr
CY_wb = CY_b + CY_w;
```

Environmental Conditions involving Lateral-Directional Stability

```
Fy_body = q*S_body_lat*(beta/U1)*Cd_0;     % Side Force at airspeed U1 involving sideslip angle [lbf]
Fy_vt = -q*S_v*(beta/U1)*dele0;            % Side Force induced by vertical stabilizer [lbf]
Fy = Fy_body - Fy_vt;                      % Total Side Force
Yv = Fy/(q);                               % Side Force Dimensionless Derivative
up_grad = 0;                               % Upwash Gradient for Nacelles (assumed zero due to zero degree AoA)
% alpha_v =
```

Lateral-Directional Stability Coefficients

```
eta_v = q_v/q;                                                              % Dynamic Pressure Ratio at Vertical Stabilizer
delr = (1.5-1)/(14.15799-9.19251);                                          % Rudder Lift Curve Slope
dela0 = (2-0.5)/(13.52031-(-1.311193));                                     % Drag Rudder Lift Curve Slope
Thrust_vector = (cos(20)*Thrust_AB*L_engine_long)/(q*S*b);                  % Thrust vector component to Side Force [lbs] (At max deflection of 20 degrees laterally
CL_alpha_v = dele0*(-beta + rho_sidewash) + delr*drag_rudder_deflection;     % Coefficient of lift of Vertical Stabilizer given rudder deflection
Cl_w = Lift/(q*b*(0.5*S));                                                  % Coefficient of the Rolling Moment of the Main Wing
l_w = q*S*b*Cl_w;                                                           % Main Wing Rolling Moment
Cl_vt = 0;                                                                  % Coefficient of the Rolling Moment of the Vertical Stabilizer (Zero due to lack of vert
l_vt = -q_v*S_v*b_v*Cl_vt;                                                  % Vertical Stabilizer Rolling Moment
Cn_beta = dele0*q_v*((S_v*L_v)/(S*b));                                      % Yaw stiffness derivative
% Cn_beta = (X_ac_v - X_cg)*CL_alpha_v;                                     % Not sure this one is correct.
Cn_r = -2*V_v*CL_alpha_v*(l_vt/b);                                          % Yaw damping derivative
Cn_p = -2*(((m_dot_inlet^2)*L_engine_lat)/(A_inlet*rho*q*S*c))*(1+up_grad); % Pitching moment contribution to yaw coefficient due to engine thrust (Roskam Eq. 3.240
Cn_delr = -delr*q_v*((S_v*L_v)/(S*b));                                      % Coefficient of the Yawing Moment regarding Rudder deflection
Cn_del_dragr = (0.25*b_dragr*c_dragr*y_dragr)/(S*b);                        % Coefficient of the Yawing Moment regarding Drag Rudder
Cn_dela = 0;                                                                % Coefficient of the Yawing Moment regarding aileron deflection [Negligible]
Cn = Cn_beta*beta + Cn_del_dragr * drag_rudder_deflection*Fy;               % Coefficient of the Yawing Moment
Cl = Cl_w + Cl_vt;                                                          % Coefficient of the Rolling Moment
n_w = q*S*b*Cn;                                                             % Main Wing Yawing Moment
Cl_beta_v = dele0*q*((S_v*L_v)/(S*b))*(0);                                  % Vertical Tail contribution to Side-slip roll moment (Zero due to lack of vertical tail
Cl_beta = dele0*q*((-Fy)/(S*b));                                            % Coefficient of Rolling Moment regarding Side-Slip
Cl_p = (2*U1/b)*dCl_dp*(pi/180);                                            % Roll damping derivative
Cl_r = (2*U1/b)*dCl_dr*(pi/180);                                            % Rolling moment due to Yaw Rate
Cl_dp = (2*U1/b)*dCl_dp;
Cl_dr = (2*U1/b)*dCl_dr;
Cl_delr = (p_ss*b*Cl_dr)/(dela*2*U1);                                       % Coefficient of Roll Moment due to Rudder Deflection
Cl_dela = (p_ss*b*Cl_dp)/(dela*2*U1);                                       % Coefficient of Roll Moment due to Aileron Deflection
Cy_p = 0;                                                                   % Coefficient of Side force induced by pitch change (negligible for steady flight)
Cy_beta = CY_wb/beta_rad;                                                   % Coefficient of Side force induced by Sideslip (USAF Stability and Control DATCOM, 5.2.
Cy_r = CL_alpha_v*((2*L_v)/b)*eta_v*(S_v/S);                                % Coefficient of Side force induced by Yaw Rate (Roskam eq. 3.189)
Cy_del_tvec = Thrust_vector;                                                % Coefficient of Side force induced by Thrust Vectoring
Cy_del_dragr = CL_alpha_v*drag_rudder_deflection*q_v*(S_a/S);               % Coefficient of Side force induced by Rudder Deflection (Roskam eq. 3.79)
Cy_delr = -Cy_del_dragr + Cy_del_tvec;                                      % Total Coefficient of Side Force induced by Drag Rudder Deflection and Thrust Vectoring
Cy_dela = 0;
%Cy = Cy_dela when dela = 0
```

Lateral-Directional Stability Derivatives - NEED TO ADD CONSIDERATION FOR THRUST VECTORING FOR YAW AUTHORITY

```
Theta1 = 5;                                % [deg]
Lp = ((q*S*b^2)/(Ixx_prime*2*U1))*Cl_p;    % [1/s]
Lb = ((q*S*b)/(Ixx_prime))*Cl_beta;        % [1/s^2]
Lr = ((q*S*b^2)/(Ixx_prime*2*U1))*Cl_r;    % [1/s]
Ldelr = ((q*S*b)/(Ixx_prime))*Cl_delr;     % [1/s^2]
Ldela = ((q*S*b)/(Ixx_prime))*-Cl_dela;    % [1/s^2]
Yp = (q*S*b)/(2*(Weight/g)*U1)*Cy_p;       % Should be zero due to no change in pitch attitude [ft/s]
Yb = ((q*S)/(Weight/g))*Cy_beta;           % [ft/s^2]
Yr = (q*S*b)/(2*(Weight/g)*U1)*Cy_r;       % [ft/s]
Ydelr = ((q*S)/(Weight/g))*(1*Cy_delr);    % [ft/s^2]
Ydela = ((q*S)/(Weight/g))*Cy_dela;        % Negligble affect on side force (Roskam Chapter 3) [ft/s^2]
Np = ((q*S*b^2)/(Izz_prime*2*U1))*Cn_p;    % [1/s]
Nb = ((q*S*b)/Izz_prime)*(Cn);             % [1/s^2]
Nr = ((q*S*b^2)/(Izz_prime*2*U1))*Cn_r;    % [1/s]
Ndelr = ((q*S*b)/Izz_prime)*(-Cn_del_dragr);  % [1/s^2]
Ndela = ((q*S*b)/Izz_prime)*Cn_dela;       % Negligible affect from aileron on Yaw force [1/s^2]
```

Establish State Space Model For Longitudinal Stability Assessment

```
A_long = [Xu                    Xalpha                  0              -g,
          Zu/U1                 Zalpha/U1               1               0,
```

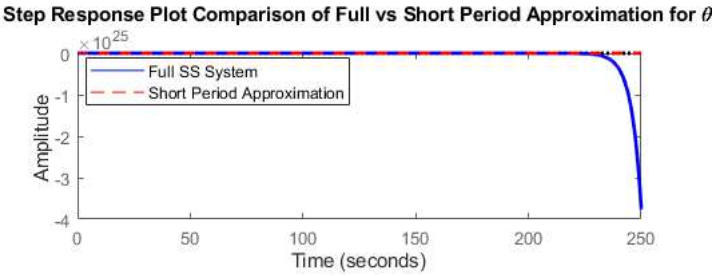
```
          (Mu + Malpha_dot*Zu/U1)   (Malpha + Malpha_dot*Zalpha/U1) (Mq + Malpha_dot)  0,
          0                         0                               1                 0];
B_long = [Xdele,
          Zdele/U1,
          (Mdele + Malpha_dot*Zdele/U1),
          0];
C_long_theta= [0, 0, 0, 1];    % This format will output pitch angle Theta
D_long = 0;

% Display A and B Matricies
disp(A_long);
disp(B_long);

% Define the entire State-Space system for alpha output
longitudinal_ss_alpha = ss(A_long,B_long,C_long_theta,D_long);

% Define Transfer function from State-Space System for Theta output
[TF_de2theta_num,TF_de2theta_denom] = ss2tf(A_long,B_long,C_long_theta,D_long);
```

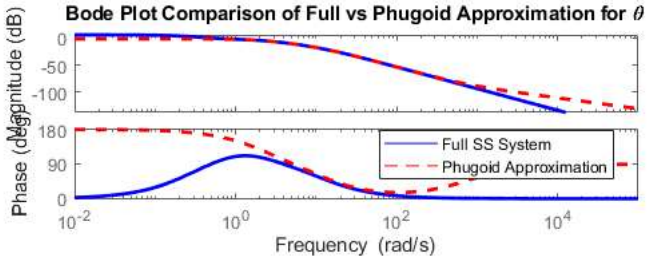
```
     -1.0955        32.17            0       -32.17
  -0.00018438     -0.79666            1            0
     -0.36011      -15.191      -13.781            0
            0            0            1            0
            0
    -0.029855
      -21.327
            0
```

Short Period Approximation

Define the state-space model for the Short Period Approximation

```
A_long_sp = [Zalpha/U1                       1,
            (Malpha + Malpha_dot*Zalpha/U1)  (Mq + Malpha_dot)];
B_long_sp = [Zdele/U1,
             Mdele+((Malpha_dot*Zdele)/U1)];
C_long_sp = [1,...
             0];
D_long_sp = 0;

% Define the short period state-space system for alpha output
longitudinalshort_ss_alpha = ss(A_long_sp,B_long_sp,C_long_sp,D_long_sp);

% Plot comparison of the Step responses of the Full vs Short Period
% Approximated systems
figure(1),
subplot(2,1,1)
step(longitudinal_ss_alpha,'b');
hold on
step(longitudinalshort_ss_alpha,'r--')
legend('Full SS System','Short Period Approximation','location','best');
title('Step Response Plot Comparison of Full vs Short Period Approximation for \theta');
set(findall(gcf,'type','line'),'linewidth',2);

% Plot comparison of the Frequency esponses of the Full vs Short Period
% NEEDS WORK
% Approximated systems
figure(1),
subplot(2,1,2)
bode(longitudinal_ss_alpha,'b');
hold on
bode(longitudinalshort_ss_alpha,'r--')
legend('Full SS System','Phugoid Approximation','location','best');
title('Bode Plot Comparison of Full vs Phugoid Approximation for \theta');
set(findall(gcf,'type','line'),'linewidth',2);
```

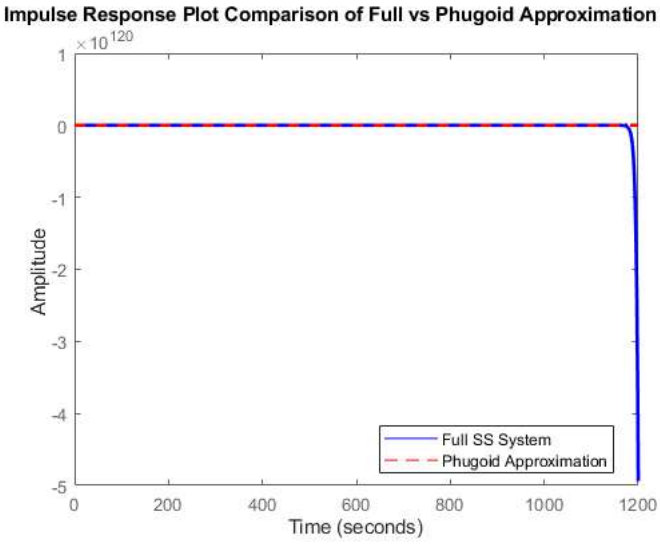


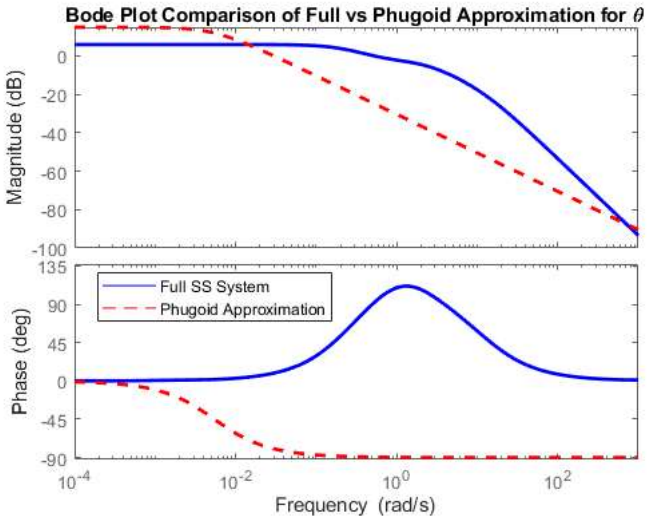


Phugoid Approximation

Define State-Space model for phugoid approximation

```
A_long_phugoid = [Xu     -g,
                  (-Zu/U1) 0];
B_long_phugoid = [Xdele,
                  (-Zdele/U1)];
C_long_phugoid_theta = [0,1];     % For pitch angle output
D_long_phugoid = 0;

% Define C Matrix for full system to evaluate pitch angle theta
C_long_theta = [0, 0, 0, 1];

% Define the full state-space system for theta output
longitudinal_ss_theta = ss(A_long,B_long,C_long_theta,D_long);

% Define the phugoid state-space system for theta output
longitudinalphugoid_ss_theta = ss(A_long_phugoid,B_long_phugoid,C_long_phugoid_theta,D_long_phugoid);

% Plot comparison of the Step esponses of the Full vs Short Period
% Approximated systems
figure,
impulse(longitudinal_ss_theta,'b');
hold on
impulse(longitudinalphugoid_ss_theta,'r--')
legend('Full SS System','Phugoid Approximation','location','best');
title('Impulse Response Plot Comparison of Full vs Phugoid Approximation for \theta');
set(findall(gcf,'type','line'),'linewidth',2);

% Plot comparison of the Frequency esponses of the Full vs Short Period
% Approximated systems
figure,
bode(longitudinal_ss_theta,'b');
hold on
bode(longitudinalphugoid_ss_theta,'r--')
legend('Full SS System','Phugoid Approximation','location','best');
title('Bode Plot Comparison of Full vs Phugoid Approximation for \theta');
set(findall(gcf,'type','line'),'linewidth',2);
```





Establish Lateral-Direcitonal State-Space Model

A relates to _ (phi), _ (p), sideslip angle (beta), _ (r), and _ (psi)

```
A_lat = [0                 1       0       0           0,
         0                 Lp      Lb      Lr          0,
         (g*cos(Theta1)/U1) (Yp/U1) (Yb/U1) ((Yr/U1)-1) 0,
         0                 Np      Nb      Nr          0,
         0                 0       0       1           0];
% B relates to rudder deflection (delr) and aileron deflection (dela)
B_lat = [0          0,
        Ldelr       Ldela
        (Ydelr/U1) (Ydela/U1)
        Ndelr       Ndela
        0           0];

% Establish C and D matricies for Roll angle and Lat-Dir TF
C_lat_phi = [1 0 0 0 0];
D_lat = [0 0];

% Lateral-Directional State-Space System
lateral_ss_phi = ss(A_lat,B_lat,C_lat_phi,D_lat);

% Create TF from Lat-Dir SS System
[TF_da2phi_num,TF_da2phi_denom] = ss2tf(A_lat,B_lat(:,2),C_lat_phi,D_lat(:,2));
```

Create Transfer Function for Heading Evaluation

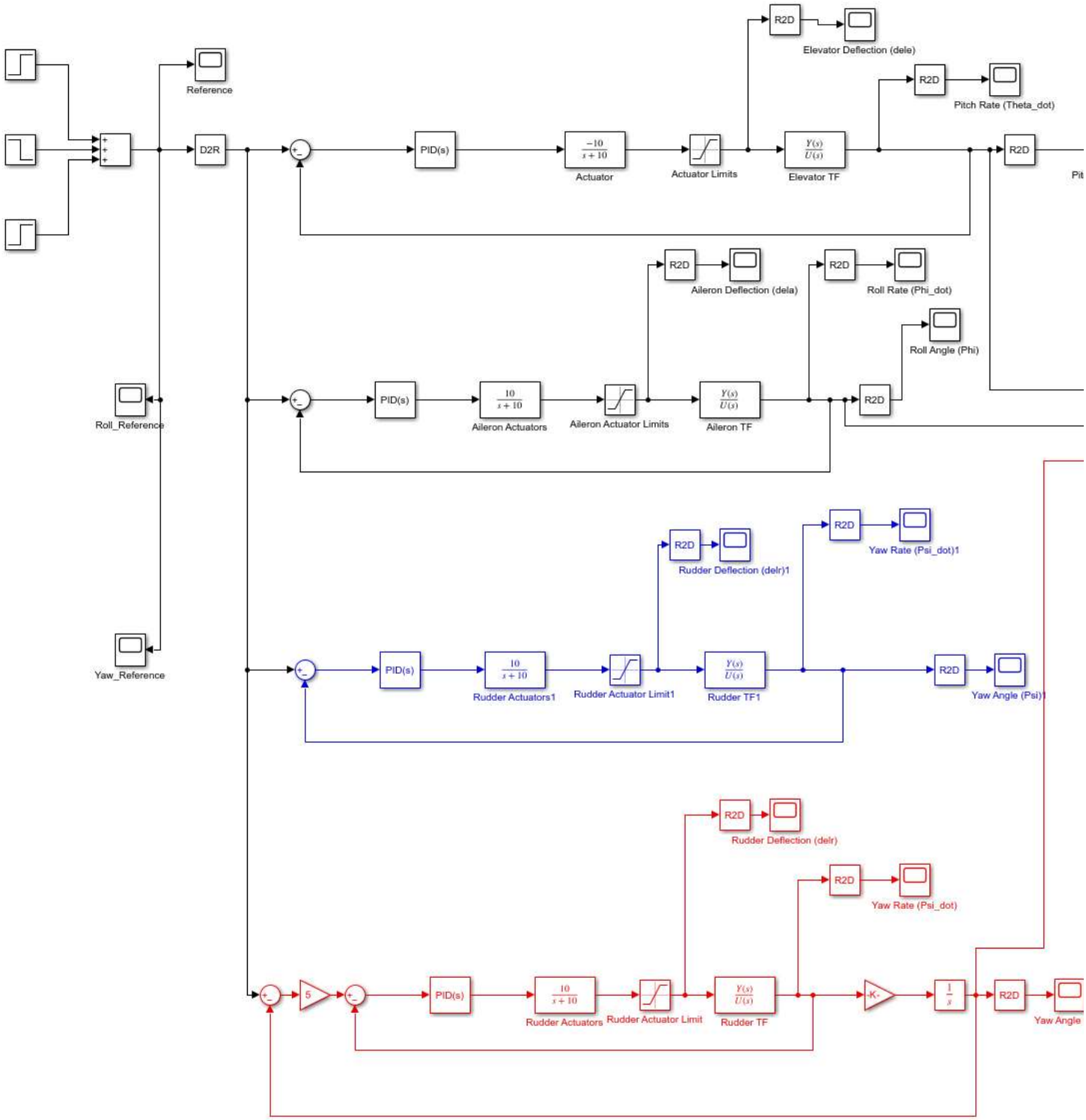
```
C_lat_psi = [0 0 0 0 1];

% Lateral-Directional State-Space System for Psi
lateral_ss_psi = ss(A_lat,B_lat,C_lat_psi,D_lat);
```

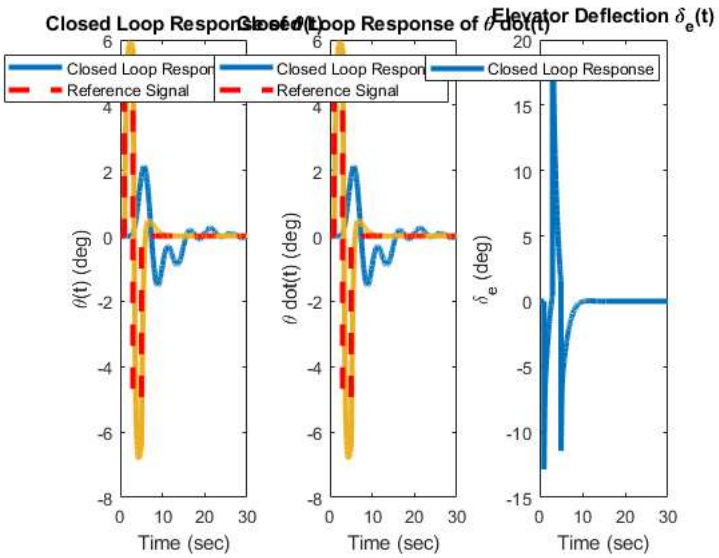
```
% Create TF from Lat-Dir SS System
[TF_da2psi_num,TF_da2psi_denom] = ss2tf(A_lat,B_lat(:,1),C_lat_psi,D_lat(:,2));
```

Open Simulink Simulation and Plot Elevator Response

```
t_f = 30;
open_system("TaillessAircraftControl_Simulink.slx")
T_elevator_eval = sim("TaillessAircraftControl_Simulink.slx");

% Plot the Closed Loop Responses
figure(7),          % Theta response
subplot(1,3,1);
plot(T_elevator_eval.T_Theta_CL(:,1), T_elevator_eval.T_Theta_CL(:,2));
hold on
plot(T_elevator_eval.T_Reference(:,1), T_elevator_eval.T_Reference(:,2),'r--');
legend('Closed Loop Response', 'Reference Signal');
xlabel('Time (sec)');
ylabel('\theta(t) (deg)');
title('Closed Loop Response of \theta(t)');
set(gca,'fontsize',10);
set(findall(gcf,'type','line'),'linewidth',3);

figure(7),          % Theta_dot response
subplot(1,3,2);
plot(T_elevator_eval.T_Theta_dot(:,1), T_elevator_eval.T_Theta_dot(:,2));
hold on
plot(T_elevator_eval.T_Reference(:,1), T_elevator_eval.T_Reference(:,2),'r--');
legend('Closed Loop Response', 'Reference Signal');
xlabel('Time (sec)');
ylabel('\theta dot(t) (deg)');
title('Closed Loop Response of \theta dot(t)');
set(gca,'fontsize',10);
set(findall(gcf,'type','line'),'linewidth',3);

figure(7),          % Elevator Deflection
subplot(1,3,3);
plot(T_elevator_eval.T_dele(:,1), T_elevator_eval.T_dele(:,2));
legend('Closed Loop Response', 'Reference Signal');
xlabel('Time (sec)');
ylabel('\delta_e (deg)');
title('Elevator Deflection \delta_e(t)');
set(gca,'fontsize',10);
set(findall(gcf,'type','line'),'linewidth',3);
```

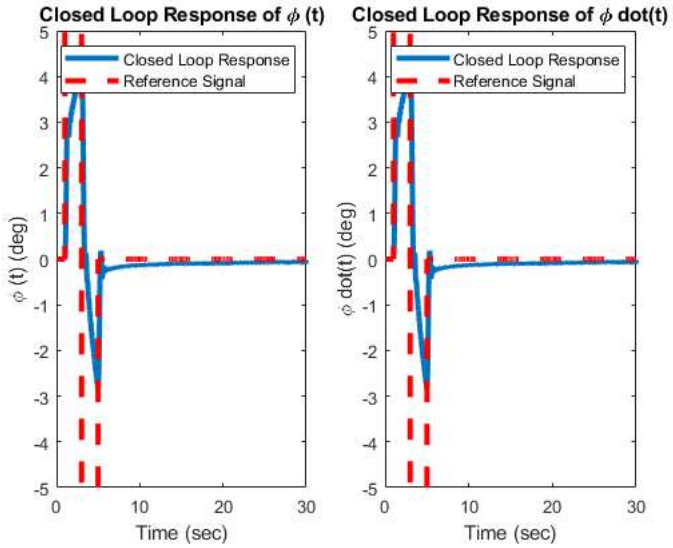
```
Warning: Ignoring extra legend entries.
```


Reference
D2R
PID(s)
$\frac{-10}{s+10}$
Actuator
Actuator Limits
$\frac{Y(s)}{U(s)}$
Elevator TF
R2D
Elevator Deflection (dele)
Pitch Rate (Theta_dot)
Pit
Roll_Reference
Aileron Actuators
$\frac{10}{s+10}$
Aileron Actuator Limits
Aileron TF
Aileron Deflection (dela)
Roll Rate (Phi_dot)
Roll Angle (Phi)
Yaw_Reference
Rudder Actuators1
Rudder Actuator Limit1
Rudder TF1
Rudder Deflection (delr)1
Yaw Rate (Psi_dot)1
Yaw Angle (Psi)1
5
Rudder Actuators
Rudder Actuator Limit
Rudder TF
Rudder Deflection (delr)
Yaw Rate (Psi_dot)
-K-
$\frac{1}{s}$
Yaw Angle



Plot Simulink Roll Response

```
T_aileron_eval = sim("TaillessAircraftControl_Simulink.slx");
figure(8),          % Phi response
subplot(1,2,1)
plot(T_aileron_eval.T_Phi(:,1), T_aileron_eval.T_Phi(:,2));
hold on
plot(T_aileron_eval.T_Reference(:,1), T_aileron_eval.T_Reference(:,2),'r--');
legend('Closed Loop Response', 'Reference Signal');
xlabel('Time (sec)');
ylabel('\phi (t) (deg)');
title('Closed Loop Response of \phi (t)');
set(gca,'fontsize',10);
set(findall(gcf,'type','line'),'linewidth',3);

figure(8),          % Phi_dot response
subplot(1,2,2)
plot(T_aileron_eval.T_Phi_dot(:,1), T_aileron_eval.T_Phi_dot(:,2));
hold on
plot(T_aileron_eval.T_Reference(:,1), T_aileron_eval.T_Reference(:,2),'r--');
legend('Closed Loop Response', 'Reference Signal');
xlabel('Time (sec)');
ylabel('\phi dot(t) (deg)');
title('Closed Loop Response of \phi dot(t)');
set(gca,'fontsize',10);
set(findall(gcf,'type','line'),'linewidth',3);
```

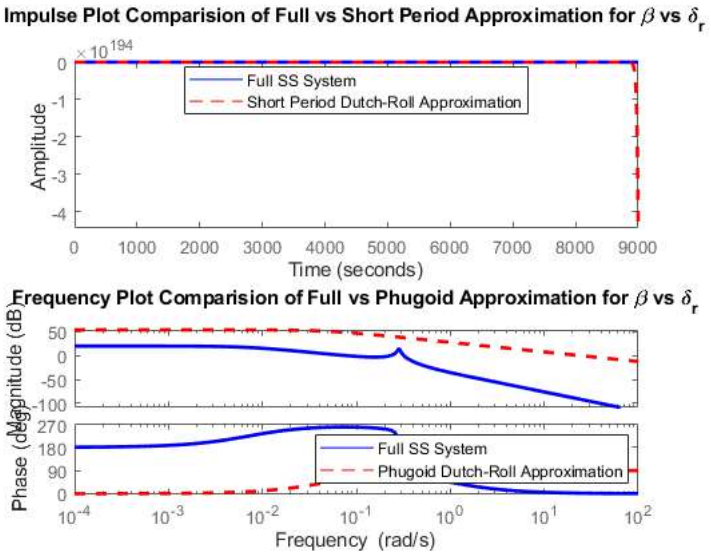


Dutch-Roll Mode Approximation

Output C Matrix for Dutch-Roll Approximation (p)

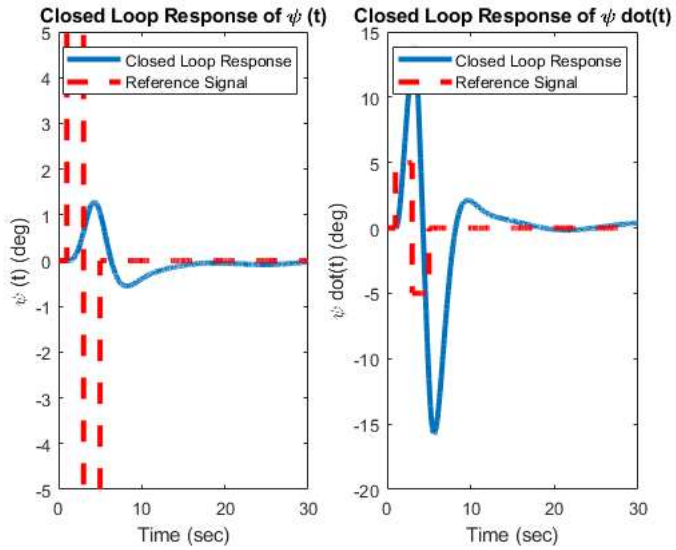
```
C_lat_p = [0 0 0 1 0];
```

```
% Create TF from Lat-Dir SS System
[r_numerator_delr,r_denominator_delr] = ss2tf(A_lat,B_lat,C_lat_p,D_lat,1);
[r_numerator_dela,r_denominator_dela] = ss2tf(A_lat,B_lat,C_lat_p,D_lat,2);
dutchroll_numerator_r = tf(r_numerator_delr,r_denominator_delr);
dutchroll_denominator_delr = tf(r_numerator_dela,r_denominator_dela);

% Roll Approximation Transfer Function
dutchroll_approx_TF = tf([Ndelr*(U1 -Yb) + Nb*Ydelr], [1 -(Nr +(Yb/U1) + (Nb+(1/U1)*(Yb*Nr-Nb*Yr)))]);

% Graph and compare Full SS response vs. Short Period Approximation for
% Dutch-Roll Mode
figure(9),
subplot(2,1,1);
impulse(dutchroll_denominator_delr,'b');
hold on
impulse(dutchroll_approx_TF,'r--');
legend('Full SS System','Short Period Dutch-Roll Approximation','location','best');
title('Impulse Plot Comparision of Full vs Short Period Approximation for \beta vs \delta_r')
set(findall(gcf,'type','line'),'linewidth',2)

figure(9),
subplot(2,1,2);
bode(dutchroll_denominator_delr,'b');
hold on
bode(dutchroll_approx_TF,'r--');
legend('Full SS System','Phugoid Dutch-Roll Approximation','location','best');
title('Frequency Plot Comparision of Full vs Phugoid Approximation for \beta vs \delta_r')
set(findall(gcf,'type','line'),'linewidth',2)
```



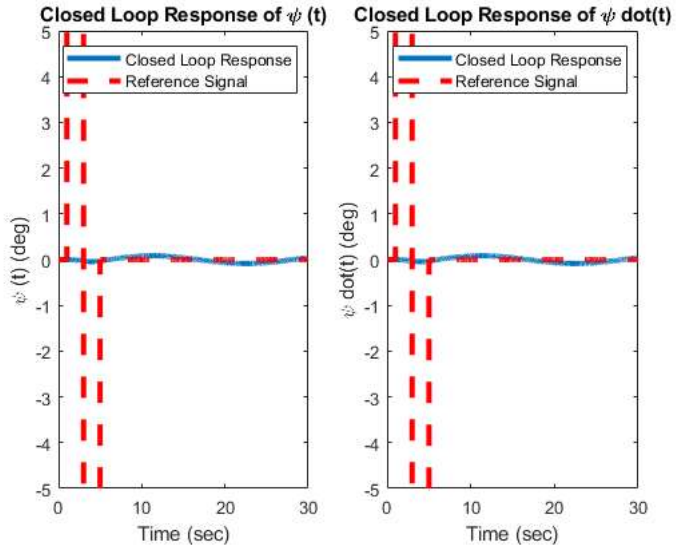
Plot Simulink Yaw Response (Heading Angle via Aileron Control)

```
T_dragrudder_eval = sim("TaillessAircraftControl_Simulink.slx");
figure(11),         % Phi response
subplot(1,2,1)
plot(T_dragrudder_eval.T_Psi(:,1), T_dragrudder_eval.T_Psi(:,2));
hold on
plot(T_dragrudder_eval.T_Reference(:,1), T_dragrudder_eval.T_Reference(:,2),'r--');
legend('Closed Loop Response', 'Reference Signal');
xlabel('Time (sec)');
ylabel('\psi (t) (deg)');
title('Closed Loop Response of \psi (t)');
set(gca,'fontsize',10);
set(findall(gcf,'type','line'),'linewidth',3);

figure(11),         % Phi_dot response
subplot(1,2,2)
plot(T_dragrudder_eval.T_Psi_dot(:,1), T_dragrudder_eval.T_Psi_dot(:,2));
hold on
plot(T_dragrudder_eval.T_Reference(:,1), T_dragrudder_eval.T_Reference(:,2),'r--');
legend('Closed Loop Response', 'Reference Signal');
xlabel('Time (sec)');
ylabel('\psi dot(t) (deg)');
title('Closed Loop Response of \psi dot(t)');
set(gca,'fontsize',10);
set(findall(gcf,'type','line'),'linewidth',3);
```



Plot Simulink Yaw Response (Heading Angle via Drag Rudder Control)

```
figure(12),         % Phi response
subplot(1,2,1)
plot(T_dragrudder_eval.T_Psi_0(:,1), T_dragrudder_eval.T_Psi_0(:,2));
hold on
plot(T_dragrudder_eval.T_Reference(:,1), T_dragrudder_eval.T_Reference(:,2),'r--');
legend('Closed Loop Response', 'Reference Signal');
xlabel('Time (sec)');
ylabel('\psi (t) (deg)');
title('Closed Loop Response of \psi (t)');
set(gca,'fontsize',10);
set(findall(gcf,'type','line'),'linewidth',3);

figure(12),         % Phi_dot response
subplot(1,2,2)
plot(T_dragrudder_eval.T_Psi_dot_0(:,1), T_dragrudder_eval.T_Psi_dot_0(:,2));
hold on
plot(T_dragrudder_eval.T_Reference(:,1), T_dragrudder_eval.T_Reference(:,2),'r--');
legend('Closed Loop Response', 'Reference Signal');
xlabel('Time (sec)');
ylabel('\psi dot(t) (deg)');
title('Closed Loop Response of \psi dot(t)');
set(gca,'fontsize',10);
set(findall(gcf,'type','line'),'linewidth',3);
```



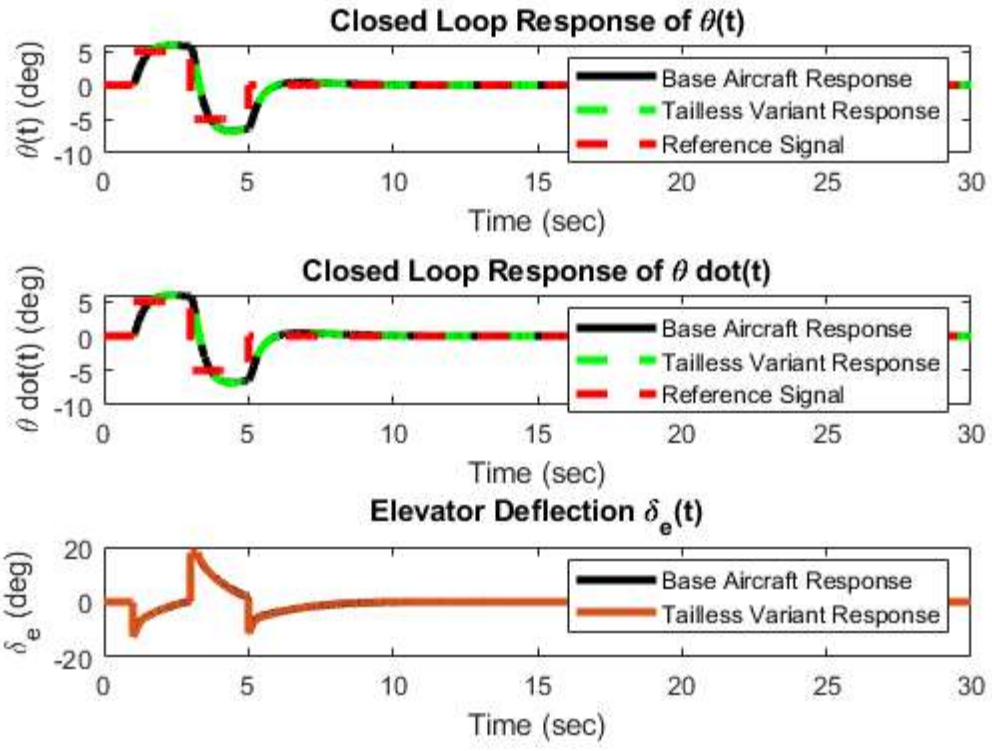
Published with MATLAB® R2024a

Contents

Plot Comparison Between Base Aircraft (Black) and Tailless Variant (Green)

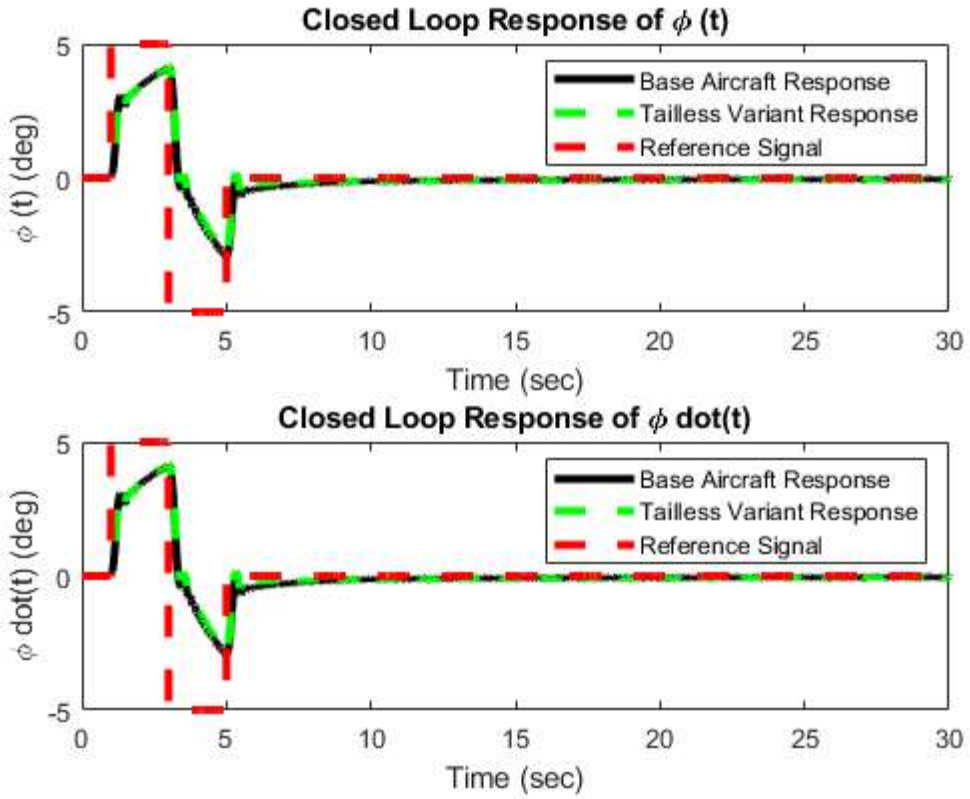
Plot the Closed Loop Responses

```
figure(1),          % Theta response
subplot(3,1,1)
plot(elevator_eval.Theta_CL(:,1), elevator_eval.Theta_CL(:,2),'k');
hold on
plot(T_elevator_eval.T_Theta_CL(:,1), T_elevator_eval.T_Theta_CL(:,2),'g--');
hold on
plot(T_elevator_eval.T_Reference(:,1), T_elevator_eval.T_Reference(:,2),'r--');
legend('Base Aircraft Response', 'Tailless Variant Response','Reference Signal');
xlabel('Time (sec)');
ylabel('\theta(t) (deg)');
title('Closed Loop Response of \theta(t)');
set(gca,'fontsize',10);
set(findall(gcf,'type','line'),'linewidth',3);

figure(1),          % Theta_dot response
subplot(3,1,2)
plot(elevator_eval.Theta_dot(:,1), elevator_eval.Theta_dot(:,2),'k');
hold on
plot(T_elevator_eval.T_Theta_dot(:,1), T_elevator_eval.T_Theta_dot(:,2),'g--');
hold on
plot(T_elevator_eval.T_Reference(:,1), T_elevator_eval.T_Reference(:,2),'r--');
legend('Base Aircraft Response', 'Tailless Variant Response','Reference Signal');
xlabel('Time (sec)');
ylabel('\theta dot(t) (deg)');
title('Closed Loop Response of \theta dot(t)');
set(gca,'fontsize',10);
set(findall(gcf,'type','line'),'linewidth',3);

figure(1),          % Elevator Deflection
subplot(3,1,3)
plot(elevator_eval.dele(:,1), elevator_eval.dele(:,2),'k');
hold on
plot(T_elevator_eval.T_dele(:,1), T_elevator_eval.T_dele(:,2));
hold on
legend('Base Aircraft Response', 'Tailless Variant Response');
xlabel('Time (sec)');
ylabel('\delta_e (deg)');
title('Elevator Deflection \delta_e(t)');
set(gca,'fontsize',10);
set(findall(gcf,'type','line'),'linewidth',3);
```



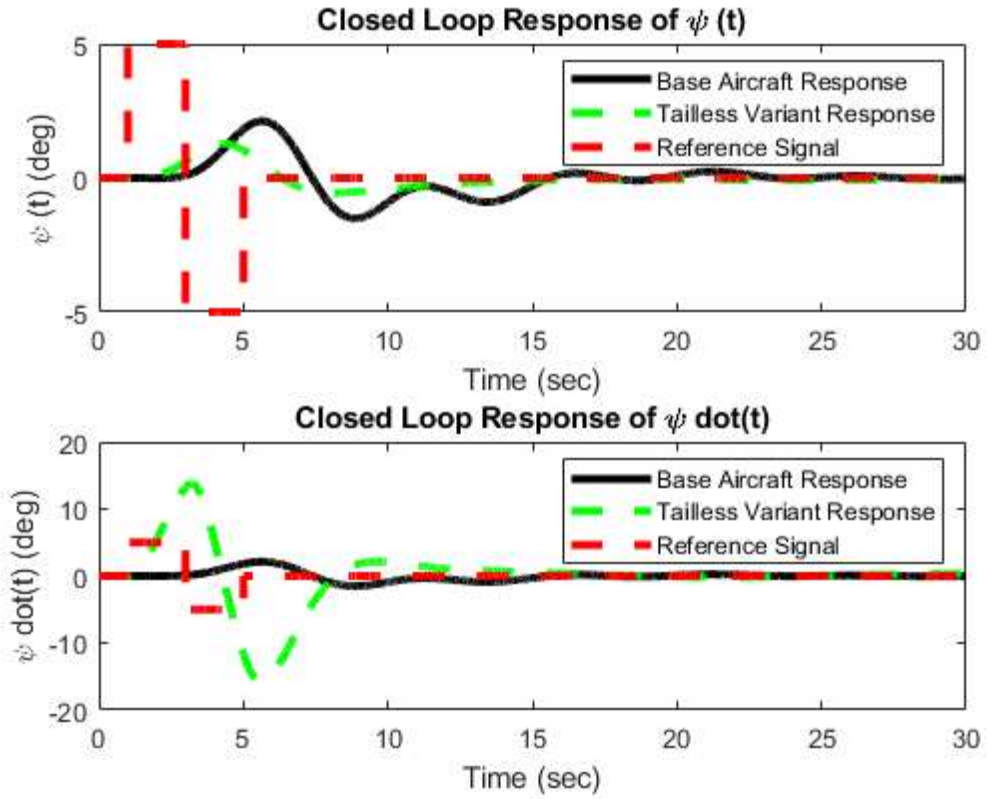
Plot Simulink Roll Response

```
T_aileron_eval = sim("TaillessAircraftControl_Simulink.slx");
figure(2),          % Phi response
subplot(2,1,1)
plot(aileron_eval.Phi(:,1), aileron_eval.Phi(:,2),'k');
hold on
plot(T_aileron_eval.T_Phi(:,1), T_aileron_eval.T_Phi(:,2),'g--');
hold on
plot(T_aileron_eval.T_Reference(:,1), T_aileron_eval.T_Reference(:,2),'r--');
legend('Base Aircraft Response', 'Tailless Variant Response','Reference Signal');
xlabel('Time (sec)');
ylabel('\phi (t) (deg)');
title('Closed Loop Response of \phi (t)');
set(gca,'fontsize',10);
set(findall(gcf,'type','line'),'linewidth',3);

figure(2),          % Phi_dot response
subplot(2,1,2)
plot(aileron_eval.Phi_dot(:,1), aileron_eval.Phi_dot(:,2),'k');
hold on
plot(T_aileron_eval.T_Phi_dot(:,1), T_aileron_eval.T_Phi_dot(:,2),'g--');
hold on
plot(T_aileron_eval.T_Reference(:,1), T_aileron_eval.T_Reference(:,2),'r--');
legend('Base Aircraft Response', 'Tailless Variant Response','Reference Signal');
xlabel('Time (sec)');
ylabel('\phi dot(t) (deg)');
title('Closed Loop Response of \phi dot(t)');
set(gca,'fontsize',10);
set(findall(gcf,'type','line'),'linewidth',3);
```



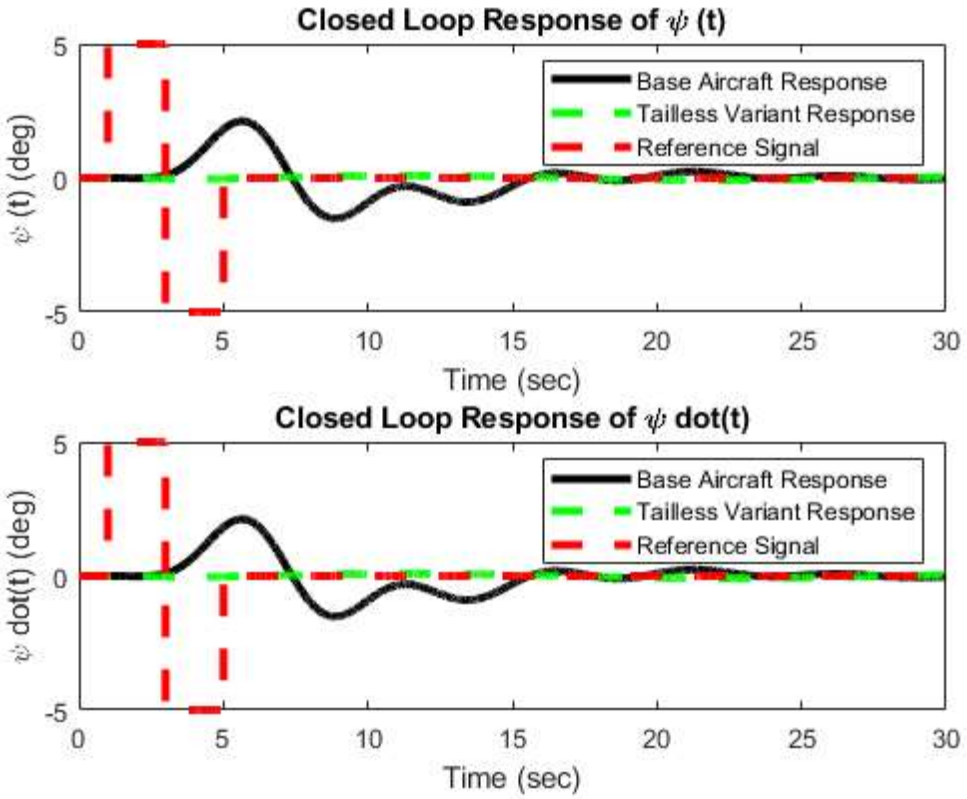
Plot Simulink Yaw Response (Heading Angle via Roll Control)

```
T_dragrudder_eval = sim("TaillessAircraftControl_Simulink.slx");
figure(3),          % Phi response
subplot(2,1,1)
plot(rudder_eval.Psi(:,1), rudder_eval.Psi(:,2),'k');
hold on
plot(T_dragrudder_eval.T_Psi(:,1), T_dragrudder_eval.T_Psi(:,2),'g--');
hold on
plot(T_dragrudder_eval.T_Reference(:,1), T_dragrudder_eval.T_Reference(:,2),'r--');
legend('Base Aircraft Response', 'Tailless Variant Response','Reference Signal');
xlabel('Time (sec)');
ylabel('\psi (t) (deg)');
title('Closed Loop Response of \psi (t)');
set(gca,'fontsize',10);
set(findall(gcf,'type','line'),'linewidth',3);

figure(3),          % Phi_dot response
subplot(2,1,2)
plot(rudder_eval.Psi_dot(:,1), rudder_eval.Psi_dot(:,2),'k');
hold on
plot(T_dragrudder_eval.T_Psi_dot(:,1), T_dragrudder_eval.T_Psi_dot(:,2),'g--');
hold on
plot(T_dragrudder_eval.T_Reference(:,1), T_dragrudder_eval.T_Reference(:,2),'r--');
legend('Base Aircraft Response', 'Tailless Variant Response','Reference Signal');
xlabel('Time (sec)');
ylabel('\psi dot(t) (deg)');
title('Closed Loop Response of \psi dot(t)');
set(gca,'fontsize',10);
set(findall(gcf,'type','line'),'linewidth',3);
```

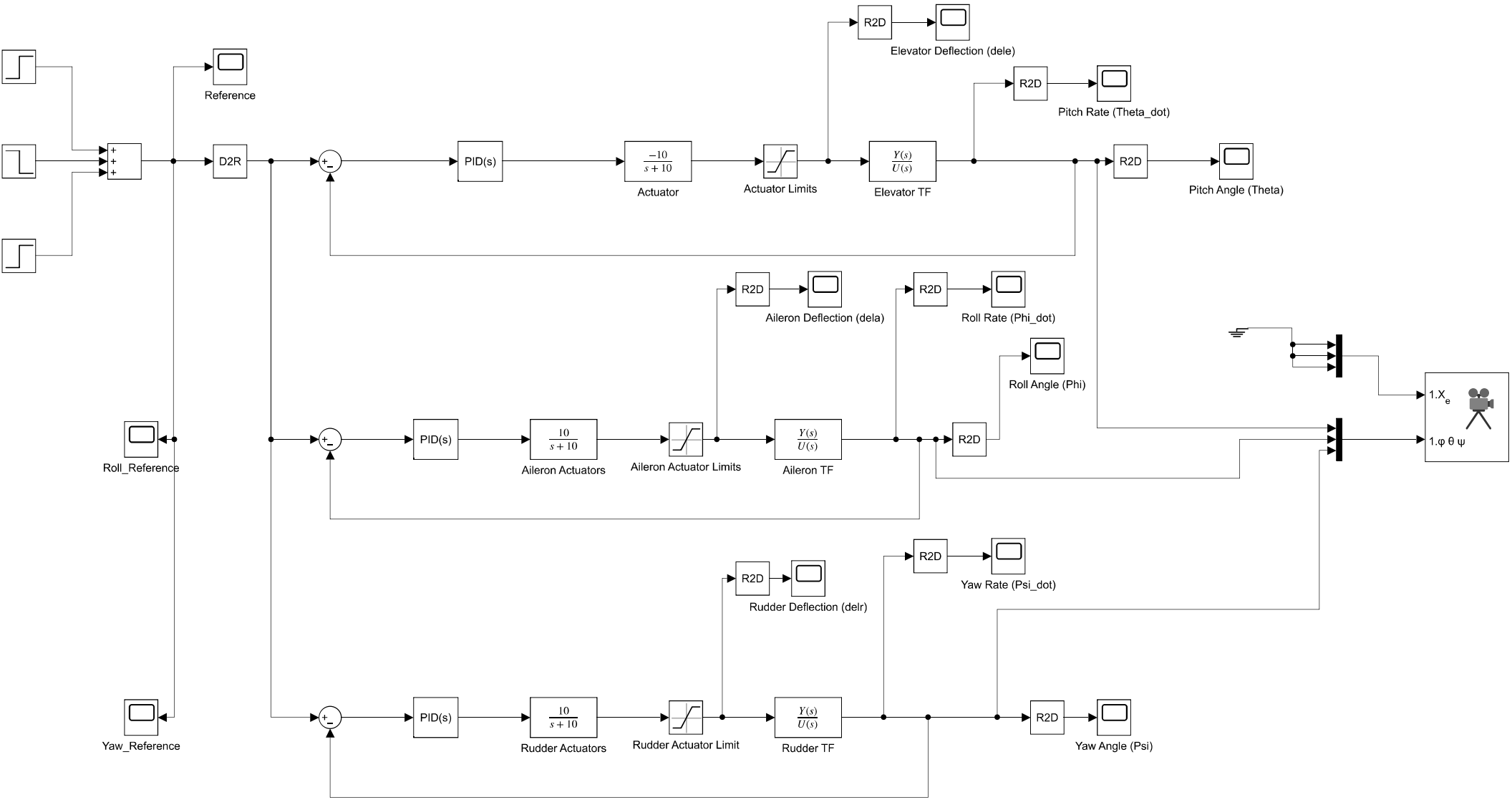


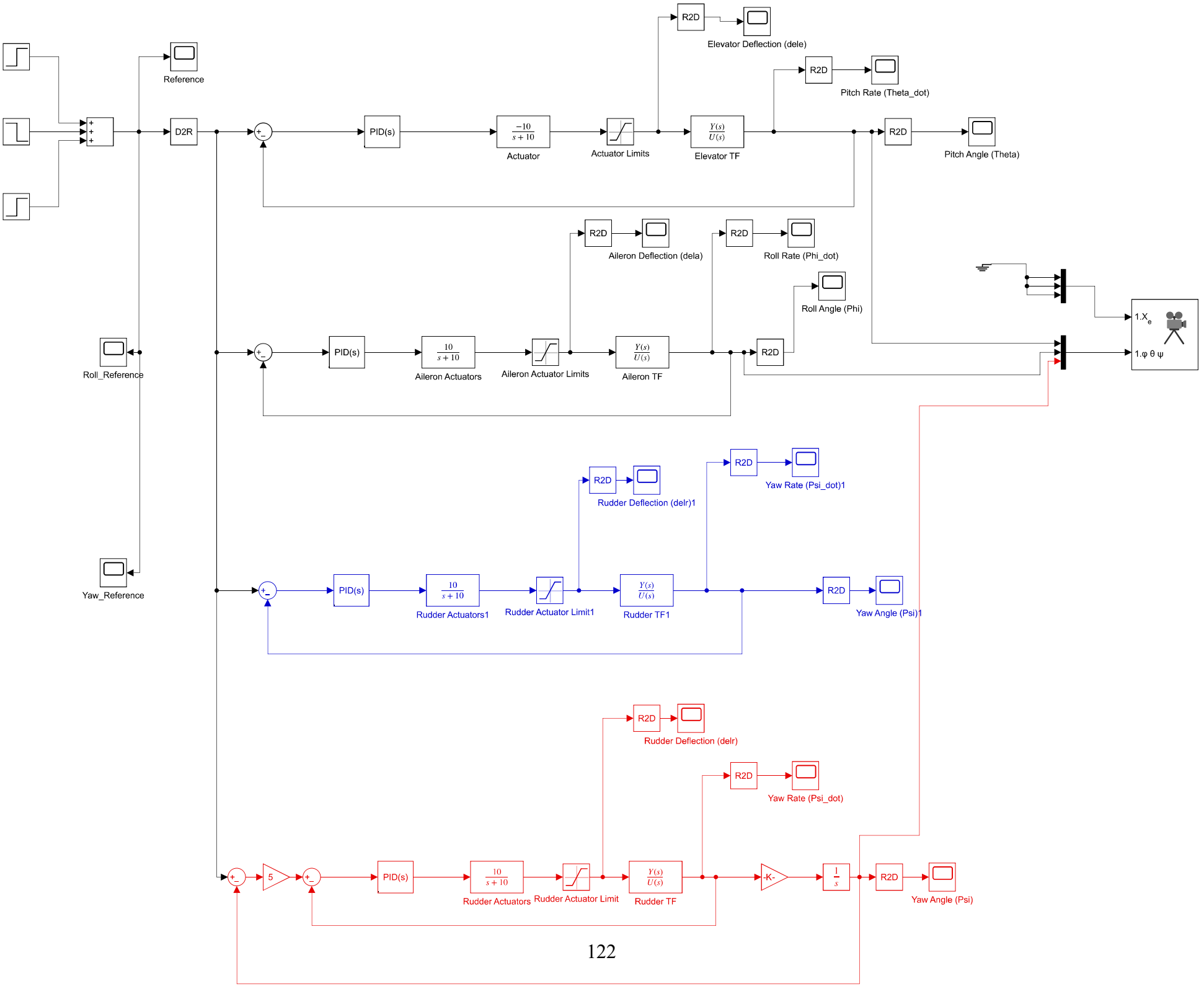
Plot Simulink Yaw Response (Heading Angle via Yaw Control)

```
figure(4),          % Phi response
subplot(2,1,1)
plot(rudder_eval.Psi(:,1), rudder_eval.Psi(:,2),'k');
hold on
plot(T_dragrudder_eval.T_Psi_0(:,1), T_dragrudder_eval.T_Psi_0(:,2),'g--');
hold on
plot(T_dragrudder_eval.T_Reference(:,1), T_dragrudder_eval.T_Reference(:,2),'r--');
legend('Base Aircraft Response', 'Tailless Variant Response','Reference Signal');
xlabel('Time (sec)');
ylabel('\psi (t) (deg)');
title('Closed Loop Response of \psi (t)');
set(gca,'fontsize',10);
set(findall(gcf,'type','line'),'linewidth',3);

figure(4),          % Phi_dot response
subplot(2,1,2)
plot(rudder_eval.Psi_dot(:,1), rudder_eval.Psi_dot(:,2),'k');
hold on
plot(T_dragrudder_eval.T_Psi_dot_0(:,1), T_dragrudder_eval.T_Psi_dot_0(:,2),'g--');
hold on
plot(T_dragrudder_eval.T_Reference(:,1), T_dragrudder_eval.T_Reference(:,2),'r--');
legend('Base Aircraft Response', 'Tailless Variant Response','Reference Signal');
xlabel('Time (sec)');
ylabel('\psi dot(t) (deg)');
title('Closed Loop Response of \psi dot(t)');
set(gca,'fontsize',10);
set(findall(gcf,'type','line'),'linewidth',3);
```



Published with MATLAB® R2024a


Reference
D2R
PID(s)
$\frac{-10}{s+10}$
Actuator
Actuator Limits
$\frac{Y(s)}{U(s)}$
Elevator TF
R2D
Elevator Deflection (dele)
R2D
Pitch Rate (Theta_dot)
R2D
Pitch Angle (Theta)
R2D
Aileron Deflection (dela)
R2D
Roll Rate (Phi_dot)
R2D
Roll Angle (Phi)
Roll_Reference
PID(s)
$\frac{10}{s+10}$
Aileron Actuators
Aileron Actuator Limits
$\frac{Y(s)}{U(s)}$
Aileron TF
R2D
1.X_e
1.φ θ ψ
R2D
Rudder Deflection (delr)
R2D
Yaw Rate (Psi_dot)
Yaw_Reference
PID(s)
$\frac{10}{s+10}$
Rudder Actuators
Rudder Actuator Limit
$\frac{Y(s)}{U(s)}$
Rudder TF
R2D
Yaw Angle (Psi)


Reference
D2R
PID(s)
$\frac{-10}{s+10}$
Actuator
Actuator Limits
$\frac{Y(s)}{U(s)}$
Elevator TF
R2D
Elevator Deflection (dele)
Pitch Rate (Theta_dot)
Pitch Angle (Theta)
Roll_Reference
Aileron Actuators
Aileron Actuator Limits
Aileron TF
Aileron Deflection (dela)
Roll Rate (Phi_dot)
Roll Angle (Phi)
1.X_e
1.φ θ ψ
Yaw_Reference
Rudder Actuators1
Rudder Actuator Limit1
Rudder TF1
Rudder Deflection (delr)1
Yaw Rate (Psi_dot)1
Yaw Angle (Psi)1
5
Rudder Actuators
Rudder Actuator Limit
Rudder TF
Rudder Deflection (delr)
Yaw Rate (Psi_dot)
-K-
$\frac{1}{s}$
Yaw Angle (Psi)