The Pennsylvania State University
The Graduate School
College of Engineering

FLIGHT CONTROL DESIGN OF TANDEM DUCTED FAN AIRCRAFT USING REDUNDANT CONTROL EFFECTORS

A Thesis in
Aerospace Engineering
by
Gurbuz Taha Ozdemir

© 2010 Gurbuz Taha Ozdemir

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Master of Science

May 2010

The thesis of Gurbuz Taha Ozdemir was reviewed and approved* by the following:

Joseph F. Horn
Associate Professor of Aerospace Engineering
Thesis Advisor

Edward C. Smith
Professor of Aerospace Engineering

Dennis K. McLaughlin
Professor of Aerospace Engineering

George A. Lesieutre
Professor of Aerospace Engineering
Head of the Department of Aerospace Engineering

*Signatures are on file in the Graduate School

ABSTRACT

Controllability and stability of ducted fan air vehicles is a challenging problem due to their complex nonlinear aerodynamics and dynamic behavior. At the same time, the combination of vanes and rotor pitch controls can provide unique control characteristics for these vehicles. A dynamic inversion controller is designed for a tandem ducted fan air vehicle to achieve desired response characteristics across the flight envelope. The controller includes an inner loop that controls the attitude and an outer loop that controls translational motion. In this study, it is desired for the vehicle to perform translational maneuvers with minimum pitch and roll angles. This is obtained by modifying the control mixing and the model inversion controller to use a combination of cyclic pitch and vanes to generate a lateral or longitudinal propulsive force. Simulation results of the controller show that the use of vanes decreased the pitch and roll angles considerably for lateral and longitudinal translation maneuvers.

TABLE OF CONTENTS

LIST OF FIGURES..... vi

LIST OF TABLES..... ix

LIST OF SYMBOLS..... x

ACKNOWLEDGEMENTS..... xii

Chapter 1 Introduction..... 1

1.1 Brief History..... 1

1.2 Literature Review..... 3

1.3 Research Objectives..... 5

Chapter 2 Tandem Ducted Fan Model..... 8

2.1 Mathematical Model..... 8

2.1.1 Model Description..... 8

2.1.2 State Space Representation..... 11

2.1.3 Control Mixing..... 12

2.1.4 Control Vane Deflections..... 16

2.2 Physical Properties of Aircraft Modeled..... 17

2.3 Updates of Simulation Model..... 18

Chapter 3 Basic Control System..... 21

3.1 Open Loop Response..... 21

3.2 Inner Loop of the Controller..... 24

3.3 Outer Loop of the Controller..... 28

3.4 Simulation Model with Controller..... 30

3.5 Preliminary Results..... 31

3.5.1 Forward Motion Command Tracking..... 31

3.5.2 Sideward Motion Command Tracking..... 34

Chapter 4 Method for Including Redundant Control Effectors to Model Following Controller..... 36

4.1 Control Allocation..... 36

4.2 Effect of Vane Control..... 37

4.2.1 Forward Velocity Tracking..... 37

4.2.2 Sideward Velocity Tracking..... 37

4.2.3 Pitch Attitude Variation for Varying Forward Velocities..... 39

Chapter 5 Dynamic Maneuvers and Gust Rejection..... 41

5.1 Gust Response..... 41

5.2 Hover Turn..... 44

5.3 Square Flight Path Maneuver..... 45

5.4 Circular Flight Path Maneuver..... 47

5.5 Circular Flight Path Maneuver with Gusts..... 49

Chapter 6 Conclusions..... 52

Appendix A..... 54

Appendix B..... 65

B.1 Matlab Code..... 65

B.2 Embedded Matlab Codes in Simulink Diagrams..... 83

Appendix C..... 86

Bibliography..... 88

LIST OF FIGURES

Figure 1-1 Piasecki VZ-8 Flying Geep..... 1
Figure 1-2 Piasecki X-49A “SpeedHawk”..... 2
Figure 1-3 Urban Aeronautics X-Hawk..... 3
Figure 1-3 Urban Aeronautics X-Hawk..... 3

Figure 2-1 Inflow Model Illustration..... 10
Figure 2-2 Exit Control Vane Indices..... 12
Figure 2-3 Control Mixing..... 13
Figure 2-4 Front View with Deflections of Vanes 1 and 2 (Vane 1 is at behind)..... 14
Figure 2-5 Back View with Deflections of Vanes 5 and 6 (Vane 6 is at behind)..... 15
Figure 2-6 Right Side View with Deflections of Vanes 4 and 8..... 15
Figure 2-7 Left Side View with Deflections of Vanes 3 and 7..... 15
Figure 2-8 Angle of Attack Experienced on the Undeflected Vane in Forward Motion..... 17
Figure 2-9 Tandem Ducted Fan Aircraft Schematic..... 18

Figure 3-1 Eigenvalues of the Simulation Model..... 22
Figure 3-2 Open Loop Response of the Generic Ducted Fan Simulation Model for 5% Longitudinal Control Impulse..... 23
Figure 3-3 Open Loop Response of the Generic Ducted Fan Simulation Model for 1% Longitudinal Control Step Input..... 23
Figure 3-4 Open Loop Response of the Generic Ducted Fan Simulation Model for 5% Longitudinal Control Step Input..... 24
Figure 3-5 The Inner Loop..... 25
Figure 3-6 Model Follower Design for Inner Loop..... 27
Figure 3-7 Model Follower Design for Outer Loop..... 29
Figure 3-8 The Block Diagram of the Controller..... 31
Figure 3-9 On Axis Response for Forward Velocity Tracking of 15 ft/s..... 32
Figure 3-10 Off Axis Response for Forward Velocity Tracking of 15 ft/s..... 32
Figure 3-11 On Axis Response for Forward Velocity Tracking of 20 ft/s..... 33
Figure 3-12 Off Axis Response for Forward Velocity Tracking of 20 ft/s..... 33

Figure 3-13 On Axis Response for Sideward Velocity Tracking of 10 ft/s..... 34
Figure 3-14 Off Axis Response for Sideward Velocity Tracking of 10 ft/s..... 35

Figure 4-1 Control Allocation for Redundant Control Effectors..... 37
Figure 4-2 Effect of Vane Control on Pitch Attitude for 10 ft/s..... 38
Figure 4-3 Effect of Vane Control on Pitch Attitude for 15 ft/s..... 38
Figure 4-4 Effect of Vane Control on Roll Attitude for 10 ft/s..... 39
Figure 4-5 Effect of Vane Control on Pitch Attitude for Varying Forward Velocities..... 40

Figure 5-1 On Axis Response for North Gust of 15 ft/s Mean Velocity..... 42
Figure 5-2 Off Axis Response for North Gust of 15 ft/s Mean Velocity..... 42
Figure 5-3 Diagonal Gust Response of Longitudinal Dynamics for 20 ft/s
Gust Mean Velocity..... 43
Figure 5-4 Diagonal Gust Response of Lateral Dynamics for 20 ft/s Gust Mean Velocity..... 43
Figure 5-5 Yaw and Vane Responses for Hover Turn Maneuver..... 44
Figure 5-6 Velocities and Attitudes for Hover Turn Maneuver..... 45
Figure 5-7 Motion around a Square..... 46
Figure 5-8 Pitch Attitudes of the Aircraft for the Motion around a Square..... 46
Figure 5-9 Roll Attitudes of the Aircraft for the Motion around a Square..... 47
Figure 5-10 Circular Paths..... 48
Figure 5-11 Attitudes of the Aircraft for Circling Motion with a Yaw Rate of 10 deg/s..... 48
Figure 5-12 Attitudes of the Aircraft for Circling Motion with a Yaw Rate of 20 deg/s..... 49
Figure 5-13 Circling Maneuver with Gust..... 50
Figure 5-14 Velocities during Circling Maneuver with Gust..... 50
Figure 5-15 Attitudes and Vane Deflections during Circling Maneuver with Gust..... 51
Figure 5-16 Yaw Rate Command Tracking during Circling Maneuver with Gust 51

Figure A-1 The Simulink Block Diagram of the Simulation..... 55
Figure A-2 Outer Loop Block..... 56
Figure A-3 Command Block..... 56
Figure A-4 Model Follower and Feedback..... 57
Figure A-5 Model Follower Block in Model Follower and Feedback Block..... 58

Figure A-6 Roll Command Filter Block..... 58
Figure A-7 Pitch Command Filter Block..... 58
Figure A-8 Yaw Rate Command Filter Block..... 59
Figure A-9 Inversion Model Block in Model Follower and Feedback Block..... 59
Figure A-10 Control Mixing Block..... 60
Figure A-11 Long. Mixing Block in Control Mixing Block..... 60
Figure A-12 Lat. Mixing Block in Control Mixing Block..... 61
Figure A-13 Yaw Mixing Block in Control Mixing Block..... 61
Figure A-14 Vane Mixing Block in Control Mixing Block..... 62
Figure A-15 Actuators Block..... 63
Figure A-16 All Subsystem Blocks in the Actuators Block..... 63
Figure A-17 Gust Block..... 64
Figure A-18 States Block..... 64

LIST OF TABLES

Table 2-1 Tandem Ducted Fan Geometric Configuration Properties..... 18
Table 2-2 Inflow Lookup Table Parameters..... 19

Table 3-1 Command Filter Variables and Compensator Gains..... 26

LIST OF SYMBOLS

u,v,w	Body Axis Velocities (Longitudinal, Lateral, Vertical)
p, q, r	Body Angular Rates (Roll, Pitch, Yaw)
φ,θ,ψ	Euler Angles (Roll, Pitch, Yaw)
X,Y,Z	Body Axis Forces (Longitudinal, Lateral, Vertical)
L,M,N	Body Axis Moments (Roll, Pitch, Yaw)
A_D	Rotor Disk Area
α	Angle of Attack
$\overline{T}$	Air Density and Rotor Disk Area Normalized Thrust
e	Tracking Error
D	Momentum Drag
δ_{lat}	Lateral Cyclic Control
δ_{long}	Longitudinal Cyclic Control
δ_{coll}	Collective Control
δ_{pdl}	Yaw Pedal Control
δ_X	Longitudinal Vane Control
δ_Y	Lateral Vane Control
δ_{vane}	Corresponding Vane Deflection Command
g	Acceleration due to Gravity
K_P	Proportional Control Gain
K_D	Derivative Control Gain
K_I	Integral Control Gain
k_{aug}	Thrust Augmentation Factor
k_X	Flow Turning Efficiency
χ	Wake Skew Angle
$\mu_{inplane}$	Inplane Advance Ratio
μ_Z	Vertical Advance Ratio
Ω	Rotor Speed
R	Rotor Radius
ρ	Air Density
s	Laplacian Operator

T	Total Vertical Thrust
T_R	Thrust from Rotor
T_D	Thrust from Duct
θ_{1s}	Rotor Lateral Cyclic Control
θ_{1c}	Rotor Longitudinal Cyclic Control
θ_0	Rotor Collective Control
ω_n	Natural Frequency
ξ	Damping Ratio
τ	Time Constant
x	State Vector
υ_i	Induced Velocity
$\upsilon_X,\upsilon_Y,\upsilon_Z$	Translational Acceleration Pseudo Commands (Forward, Sideward, Vertical)
$\upsilon_p,\upsilon_q,\upsilon_r$	Angular Acceleration Pseudo Commands (Roll, Pitch, Yaw)
V	Velocity

Subscripts

0	Freestream
1	Rotor 1
2	Rotor 2
∞	Far-Wake
R	Disk Rotor
cmd	Pilot or Autopilot Commands
c	Filtered Commands
fwd	Forward
side	Sideward
Z	Vertical

ACKNOWLEDGEMENTS

I would like to acknowledge my advisor, Dr. Joe Horn, for his support and encouragement throughout my study. I would also like to recognize my fellow student, Wei Guo, for his resources and patience with my frequent questions. I would like to thank Eric Tobias for his contributions to this research and the Office of Naval Research (ONR) technical monitors John Kinzer and Micheal Fallon.

Most importantly, I would like to acknowledge my parents for their support, confidence and love. My graduate work would not have been possible without such blessings and I am forever grateful.

This project was funded by the Office of Naval Research (ONR) Grant Number N00014-06-1-0205.

Chapter 1

Introduction

1.1 Brief History

A helicopter with an open rotor system typically fulfills the requirements for most vertical take off and landing (VTOL) aircraft missions, but there are certain advantages to ducted rotor systems that make them well suited for niche operations. For example, ducted fans are popular in VTOL unmanned air vehicles (UAVs), because in hover they have higher thrust/power ratio for a given diameter than an open propeller, as shown by Pereira [1]. Pereira performed experiments on Micro Air Vehicles for the comparison of an open rotor and a shrouded rotor and up to 94% increase in thrust obtained with the same power consumption [1]. This allows the aircraft design to be more compact. Also they provide impact protection for the blades and personnel safety due to its enclosed fan. There are many examples of ducted fan UAVs in the market. Using ducted fans for UAVs is becoming more common, but use of a ducted fan for a full scale manned aircraft has not seen wide spread use. There are several examples of experimental ducted fan aircraft over the years. One example, the Piasecki VZ-8 Flying Geep developed in the late 1950s, is shown in Figure 1-1.

Figure 1-1 Piasecki VZ-8 Flying Geep

At that time, the handling qualities and controllability of ducted fan aircraft were limited by the current state of technology in electronic control systems, and the demand for a non-conventional V/STOL was very low. Due to improvements in control theory and technology, ducted fan vehicles are a more viable solution for manned aircraft. The Piasecki X-49A "SpeedHawk", shown in Figure 1-2, is a Vector Thrust Ducted Propelled (VTDP) compound helicopter with a ducted propeller instead of a tail rotor to provide anti-torque and yaw control [2]. The Urban Aeronautics X-Hawk, as seen in Figure 1-3, is an example for the interest in using ducted fan vehicles for operations in complex urban environments, such as using for transportation of medical personnel to an emergency fast and safe [3]. By the April of 2010 Urban Aeronautics achieved sustained tethered automatic hovering flight of AirMule, an unmanned air vehicle [4]. Figure 1-4 shows the AirMule during the hover test.

Figure 1-2 Piasecki X-49A "SpeedHawk"

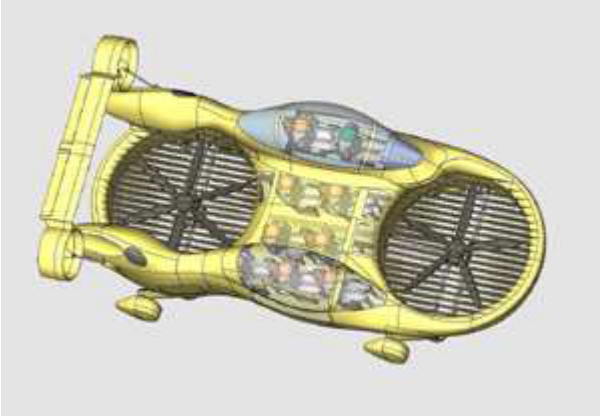

Figure 1-3 Urban Aeronautics X-Hawk

Figure 1-4 Urban Aeronautics AirMule in Hover Test (© Urban Aeronautics)

1.2 Literature Review

A considerable amount of recent research has been devoted to simulation and control of ducted fans. Tobias and Horn developed a mathematical model for a tandem ducted fan aircraft [5]. The ducted fan model was validated against experimental data of a 29-inch single ducted fan UAV. Also the usage of vanes for primary and redundant control was demonstrated. They pointed out that vanes can be used to reduce the pitch angle during the forward motion in addition to their usage for yawing.

Stability analysis of different ducted fan model have been studied and tested using many different methods. Avanzini *et al* present a model created by numerical analysis of

the flowfield around the body of the ducted fan vehicle and study performance and stability characteristics [6]. The study by De Divitiis focused on the complex flow interactions between the rotor and the duct and the aerodynamics coefficients were calculated by using two different models [7]. One of the models was used for the duct only and the other model is used for the rotor. Stability characteristics were then studied for different flight conditions. Both of the studies showed that the existence of large amount of pitching moment has an impact on the performance and stability, since the operational range is limited.

Avanzini *et al* presented development and testing of linear controllers for an unmanned ducted fan vehicle. [8] The linear controller developed were tested and checked how they meet the rotorcraft handling qualities specifications. The flight control system was also tested in a simulation environment with different aircraft configurations and good handling qualities in real-time simulation testing were observed. Johnson and Turbe studied a small ducted-fan GTSpy and found the vehicle exhibited nonlinear aerodynamics and was difficult to control. The study focused on using dynamic inversion controller with neural-network adaptation [9]. The use of such controller reduced the complexity of aircraft dynamics throughout the entire flight envelope without the use of gain scheduling. Johnson and Kannan studied an autonomous helicopter where the dynamics are more complex due to the operations on urban environment [10]. Flight control is separated into inner and outer loops, where inner loop is used to control attitude and outer loop is used to control translational trajectory. It is stated that this kind of control separation is common in autonomous helicopters. Dynamic model inversion and neural network based adaptation are implemented to increase performance of attitude and translational dynamics. Also the model error is minimized with adaptation which leads to a more accurate position tracking.

Sahani [11] developed an adaptive model inversion based controller with Attitude Command Attitude Hold (ACAH) response type for roll and pitch axes and Rate Command Attitude Hold (RCAH) response type for yaw axis. A non-linear simulation model of the UH-60A helicopter was used to test the controller and it was observed that model inversion controller with a linear model at a single design point was adequate for

off-design tracking, but use of an adaptive neural network improved the tracking performance significantly.

Hess and Bakthiari-Nejad [12] studied a sliding mode control design for use in an unmanned ducted fan aircraft, which is applied to a nonlinear, unstable and coupled model in different flight conditions. As a result of the studies, the controller, with the inclusion of gain scheduling, shown to be capable of controlling a nonlinear and unstable vehicle. Another application of dynamic inversion on small ducted fan UAV is performed by Spaulding *et al* [13], where the research focused on the control robustness, gust rejection and vehicle velocity control. The inversion model is completed with experimental wind tunnel data and state feedback.

Several researchers have investigated model inversion type controllers for use on VTOL aircraft and aircraft with redundant control effectors. Horn and Bridges [14] used a model inversion controller flight control design methodology for gust rejection during shipboard operations in order to reduce pilot workload. Horn and Guo [15] used a similar control design methodology for rotorcraft with variable rotor speed capability, which is treated as a redundant control effecter for the heave axis. A gain scheduled model following/model inversion controller is used to control the roll, pitch, yaw, heave and rotor RPM degrees of freedom. Another example of usage of model inversion controller with redundant control effectors is the study by Geiger [16]. In this study a method for calculation of the optimal control deflections of a fully compounded helicopter in trim and maneuvering flight conditions was developed.

1.3 Research Objectives

In the light of the previous studies, this study aims to design a model inversion controller to improve the stability and the controllability of the tandem ducted fan. The dynamic model of the aircraft is based on the study of Tobias and Horn [5]. The controller is separated into two loops as inner and outer loops. The inner loop is used to control attitude and the outer loop is used to control translational motion. For attitude control, pitch and roll motions are controlled by a model inversion controller with attitude

command attitude hold response type and yaw motion is controlled by a model inversion controller with rate command attitude hold response.

For the control of the translational motion, this study aims to develop a control law that minimizes roll and pitch attitude during translational motion, where the driving force for forward and sideward motion is generated from the deflection of the vanes. With this controller, vanes are going to be deflected in a way to generate propulsive force. When the vanes reach their authority limit, the vehicle smoothly transitions to use attitude change for translational motion. This is what makes this study original in this field. Use of vanes as the primary control surface, not the conventional rotor cyclic pitch control, is not a common way of implementing the control law. With the use of vanes, attitude changes can be sustained lower than the conventional control methods. Maintaining a more level attitude might be useful for an unmanned vehicle carrying a sensor payload which would achieve better tracking and performance on a level platform. Reducing attitude excursions might also be useful for a vehicle operating in a constrained urban environment. The control allocation method presented in this study could be modified to optimize the vehicle attitude for other applications as well.

The model inversion controller determines 6 control variables. In addition to the four traditional control inputs of a VTOL aircraft (lateral, longitudinal, vertical, and yaw control axes), the controller calculates X and Y axis vane controls. A control mixing system was developed by Tobias that converts the 6 control inputs into a combination of 14 control surface deflections, including rotor collective pitch, lateral cyclic pitch and longitudinal cyclic pitch for each rotor, as well as 8 different vane deflections. Roll control is primarily achieved by lateral cyclic pitch; pitch control is primarily achieved by differential collective pitch, heave control by symmetric collective pitch, and yaw by differential deflection of vanes on each rotor. In addition the vanes can be deflected symmetrically to primarily produce longitudinal and lateral forces on the vehicle (the X and Y axis vane controls). The vane deflection inputs and lateral and longitudinal commands are generated in a synchronized manner. Both X and Y axes vanes are deflected to a certain limit without any roll or pitch command. Pitch and roll commands are only used when vanes hit their saturation limits.

The mathematical model is implemented in MATLAB® code and embedded in a SIMULINK® diagram for rapid development of control laws. This study presents the methods for development of such a control system described above and the results of this controller on the tandem ducted fan configuration for different flight conditions and maneuvers.

Chapter 2

Tandem Ducted Fan Model

A generic ducted fan simulation model was developed by Tobias and Horn [5] and implemented in MATLAB for ease of development and user-friendly environment. The generic ducted fan simulation model by Tobias and Horn [5] is used in this study also. This chapter documents the basic theory of the generic ducted fan simulation model and describes the physical properties of the aircraft used in the model. The updates of the model, such as development of the inflow lookup table and improving the code vectorization, are also described in this chapter.

2.1 Mathematical Model

2.1.1 Model Description

The rotors are modeled using a rigid blade element model with uniform dynamic inflow as described in detail by Tobias and Horn [5]. The rotor inflow model used momentum theory modified to account for the thrust augmentation and flow turning effects of the ducts. An illustration of the velocity and force vectors of the inflow model is shown in Figure 2-1 by Tobias and Horn [5].

Thrust augmentation is due the effect of duct around the rotor. It is assumed that the total force vector, **T,** consists of vertical thrust component, T, and a horizontal momentum drag component, D_m. The thrust component of the total force vector is the total of rotor thrust and duct thrust.

$$T = T_R + T_D = (1 + k_{aug})T_R \tag{2.1}$$

The thrust augmentation factor, k_{aug}, represents the increase in the total thrust due to duct lift effect. The thrust augmentation factor set to be 0.3 as in the simulation model by Tobias and Horn [5].

The model accounts for the turning of freestream velocity vector $\mathbf{V_0}$ rather than only considering a reduction in the horizontal component. The velocity vectors upstream of the rotor, at the disk rotor and far downstream of the rotor are shown below from Tobias and Horn [5].

$$\begin{aligned}
\mathbf{V_0} &= V_0 \cos\alpha.\mathbf{i} - V_0 \sin\alpha.\mathbf{j} \\
\mathbf{V_R} &= V_0 \cos\left(\alpha + k_{XR}\left(\frac{\pi}{2} - \alpha\right)\right)\mathbf{i} - \left(V_0 \sin\left(\alpha + k_{XR}\left(\frac{\pi}{2} - \alpha\right)\right) + \upsilon_i\right)\mathbf{j} \\
\mathbf{V_E} &= V_0 \cos\left(\alpha + k_{X\infty}\left(\frac{\pi}{2} - \alpha\right)\right)\mathbf{i} - \left(V_0 \sin\left(\alpha + k_{X\infty}\left(\frac{\pi}{2} - \alpha\right)\right) + \upsilon_i\right)\mathbf{j} \\
\mathbf{V_\infty} &= V_0 \cos\left(\alpha + k_{X\infty}\left(\frac{\pi}{2} - \alpha\right)\right)\mathbf{i} - \left(V_0 \sin\left(\alpha + k_{X\infty}\left(\frac{\pi}{2} - \alpha\right)\right) + \upsilon_\infty\right)\mathbf{j}
\end{aligned} \tag{2.2}$$

The amount of flow turned before and after the rotor is a function of the rotor flow turning efficiency factors, denoted by k_{XR} and $k_{x\infty}$ respectively. These factors determine deflection of inflow velocity just before and downstream of the rotor respectively. In the simulation model k_{XR} was set to be 0.6 and $k_{x\infty}$ was set to be 0.8. This means, the flow is turned 60% just before the rotor and 80% downstream of the rotor. The affected angle of attacks are denoted by α^R and α^∞ that are shown below from Tobias and Horn [5]

$$\begin{aligned}
\alpha^R &= \alpha + k_{XR}\left(\frac{\pi}{2} - \alpha\right) \\
\alpha^\infty &= \alpha + k_{X\infty}\left(\frac{\pi}{2} - \alpha\right)
\end{aligned} \tag{2.3}$$

The corresponding wake skew angles at the exit duct and far downstream are given as

$$\begin{aligned}
\chi &= \tan^{-1}\left(\frac{V_0 \cos\alpha^R}{V_0 \sin\alpha^R + \upsilon_i}\right) \\
\chi_\infty &= \tan^{-1}\left(\frac{V_0 \cos\alpha^\infty}{V_0 \sin\alpha^\infty + \upsilon_\infty}\right)
\end{aligned} \tag{2.4}$$

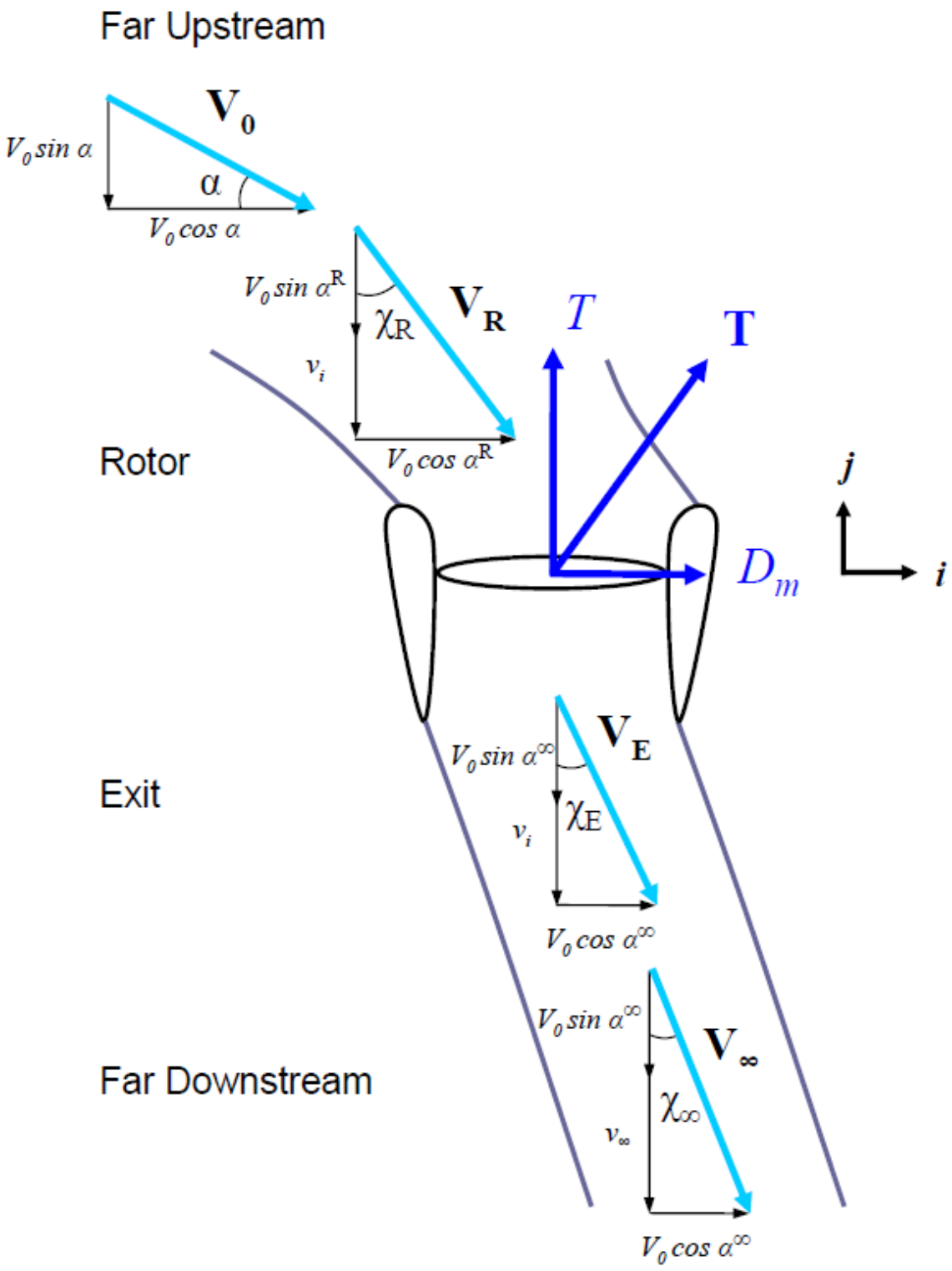

Figure 2-1 Inflow Model Illustration

Following the calculations performed by Tobias and Horn [5], the quartic expression for the induced velocity, υ_i, formulated in Equation 2.5, where for given thrust, T, freestream velocity, V_0, and angle of attack, α, the induced velocity, υ_i, needed to be solved. Calculations for obtaining the momentum equation are given in Appendix C.

$$\frac{1}{1+k_{aug}}\left[\begin{array}{l}\upsilon_i-\sin(\alpha)V_0+\sin(\alpha^R)V_0-\sin(\alpha)k_{aug}V_0\\ +\frac{1}{2}\left[\begin{array}{l}8(-\sin(\alpha)+\sin(\alpha^\infty))(1+k_{aug})V_0(\upsilon_i+\sin(\alpha^R)V_0)\\ +4(\upsilon_i+(\sin(\alpha^R)-\sin(\alpha^\infty)-\sin(\alpha^\infty)k_{aug})V_0)^2\end{array}\right]^{\frac{1}{2}}\end{array}\right] \qquad (2.5)$$
$$-\frac{T}{\rho A_D\sqrt{\upsilon_i^{\,2}+2\sin(\alpha^R)\upsilon_i V_0+V_0^{\,2}}}=0$$

Blade Element Momentum Theory (BEMT) is used to determine the distribution of inflow and lift on the blades. BEMT is combination of the principles of Blade Element Theory (BET) and the Momentum Theory. BET is used in order to make the assumption that each rotor blade section is a quasi-two-dimensional airfoil producing aerodynamic forces and moments and momentum theory provides the calculation of rotor thrust, power and induced velocity [17]. Detailed description of each theory and how they are used in the simulation model are stated by Tobias and Horn [5]. In addition to the rotor model, Tobias and Horn describe how vanes, fuselage and duct pitching moment modeled.

2.1.2 State Space Representation

A state space representation is used in the generic ducted fan simulation model. The state vector of the mathematical model of the tandem ducted fan has 14 states. The particular states used in the generic ducted fan simulation model are given by the following state vector:

$$\vec{x}=\{u \quad v \quad w \quad p \quad q \quad r \quad \phi \quad \theta \quad \psi \quad X \quad Y \quad Z \quad \upsilon_{i1} \quad \upsilon_{i2}\}^T \qquad (2.6)$$

The particular states can be described as; u, v and w are the body axis velocities, p, q and r are the body axis attitudes, Φ, θ and ψ are the Euler angles, X,Y and Z are the positions and υ_{i1}, υ_{i2} are the inflow velocities for each rotor.

The control vector generated from the controller has 6 components, which are then sent to the mixer to determine 14 control surfaces. The input commands are given by the following input vector:

$$\vec{u} = \{\delta_{lat} \quad \delta_{long} \quad \delta_{coll} \quad \delta_{pedal} \quad \delta_X \quad \delta_Y\}^T \tag{2.7}$$

The inputs δ_{lat} and δ_{long} are the lateral and longitudinal controls, that are used to control roll and pitch, respectively. δ_{coll} is the collective control and δ_{pedal} is the yaw pedal control, where δ_{coll} control vertical velocity and δ_{pedal} controls yaw. Finally, δ_X and δ_Y are the longitudinal and lateral vane controls and they control longitudinal and lateral translation.

Collective control is used for controlling the collective pitch angle of rotor blades, in order to control altitude. Longitudinal control is used for controlling the differential collective pitch angle of rotor blades, in order to obtain a pitch attitude. Lateral control is used for controlling longitudinal cyclic pitch angle of rotor blades, in order to control roll attitude. In addition to the conventional use of controls, control mixing provides control authority on the vanes. Together with the vane controls, lateral and pedal controls mix to generate deflection angle commands for each vane. Especially, pedal control is used for controlling differential lateral vane deflections, in order to control yaw attitude.

2.1.3 Control Mixing

The control mixing converts 6 control inputs into 14 control surface commands. The control effectors on the vehicle include collective, lateral cyclic and longitudinal cyclic pitch control for each of the ducted rotors and 8 control vanes, as seen in Figure 2-2, in the exit flow.

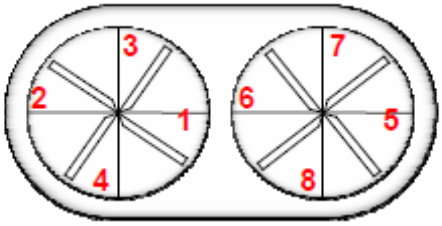

Figure 2-2 Exit Control Vane Indices

In this study, the rotor RPM is assumed to be regulated by engine RPM governor, and assumed to be approximately constant. The control mixing was previously embedded in Tobias and Horn's [5] MATLAB code implementation of the vehicle model, but in this study, control mixing extracted from the MATLAB included and implemented in Simulink for ease of modification.

Control inputs, in percent, are manipulated by gains, biases and summing junctions that convert control inputs into control deflections on the aircraft in degrees. Figure 2-3 shows the change of control inputs into control surface commands. The detailed schematic of control mixing block can be seen in Appendix A.

Collective control input would correspond to an increase in the collective pitch of both rotors equally. A 100% collective input corresponds to 20 degrees of pitching at the root of the blade. It must be noted that an increase in collective does not create any yaw moment. Counter rotating tandem rotors cancel the increase in torque reaction by each rotor, therefore no net moments is achieved.

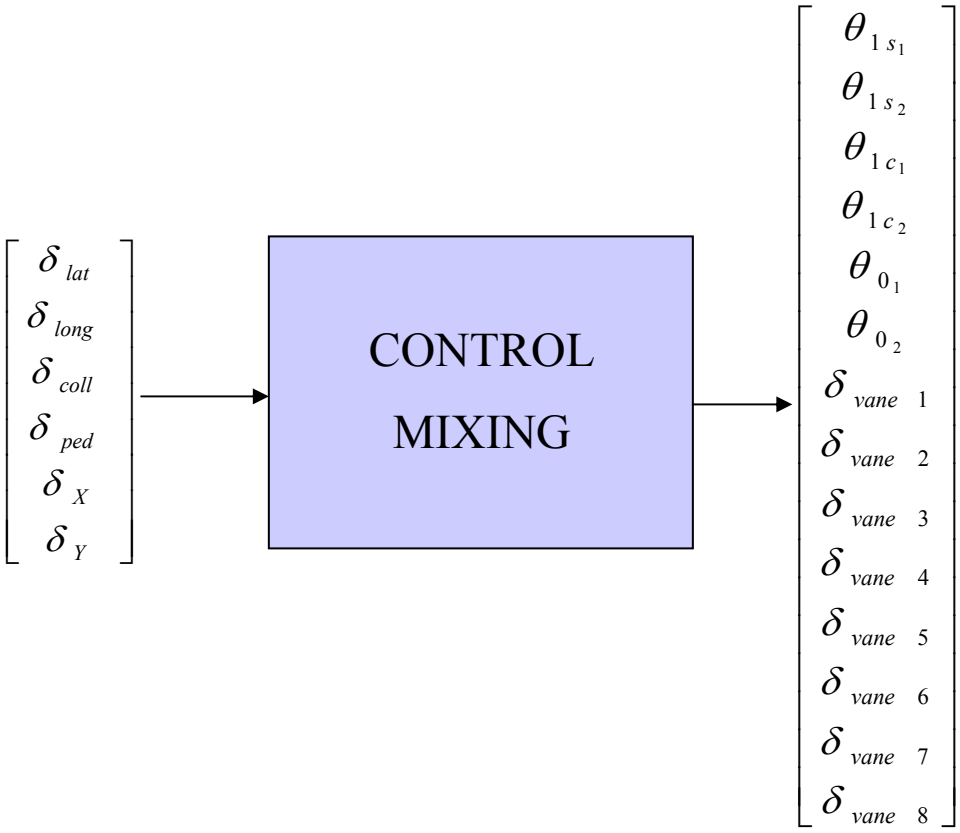

Figure 2-3 Control Mixing

Longitudinal control input control differential collective pitch on the rotors. The pitch control is achieved by this method; however, differential collective pitch results in yaw moment because increase in collective pitch in one of the rotors increases the torque reaction on that rotor and therefore a net yaw moment is observed.

Roll moment is achieved by 4 contributors, lateral cyclic pitch, pedal and vane control inputs. Lateral control input controls the lateral cyclic pitch on both rotors, thus the roll control of aircraft is achieved. Lateral control input, pedal control input and vane control inputs are all mixed to create 8 vane deflection angle commands. The main purposes of vane deflections are producing yaw moment and propulsive force for translational motion, but since vanes are located below the center of gravity of the aircraft, cross-coupling occurs between lateral vane deflections and roll moment. Thus, roll moment is experienced by the lateral vane deflections.

The longitudinal and lateral vane deflections have 2 simultaneous uses. Differential deflection of lateral vanes result in yaw moment and symmetric deflection of vanes results in propulsive force for translational motion in forward and sideward directions. The lateral vane deflections are shown with their positive deflections in Figures 2-4 and 2-5. Similar to the lateral vane deflections, the longitudinal vane deflections are shown with their positive deflections in Figures 2-6 and 2-7.

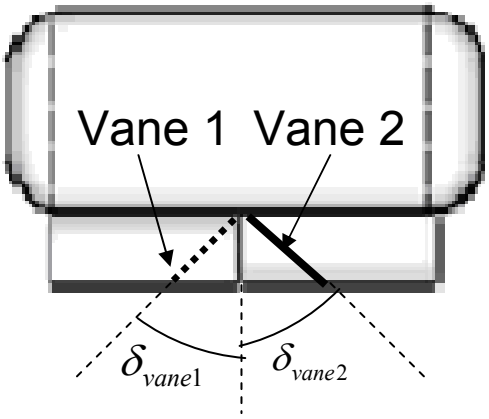

Figure 2-4 Front View with Deflections of Vanes 1 and 2 (Vane 1 is at behind)

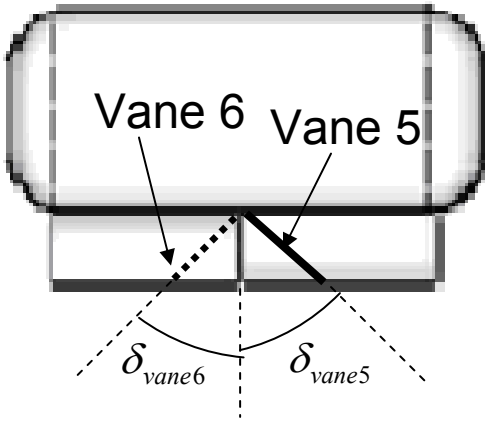

Figure 2-5 Back View with Deflections of Vanes 5 and 6 (Vane 6 is at behind)

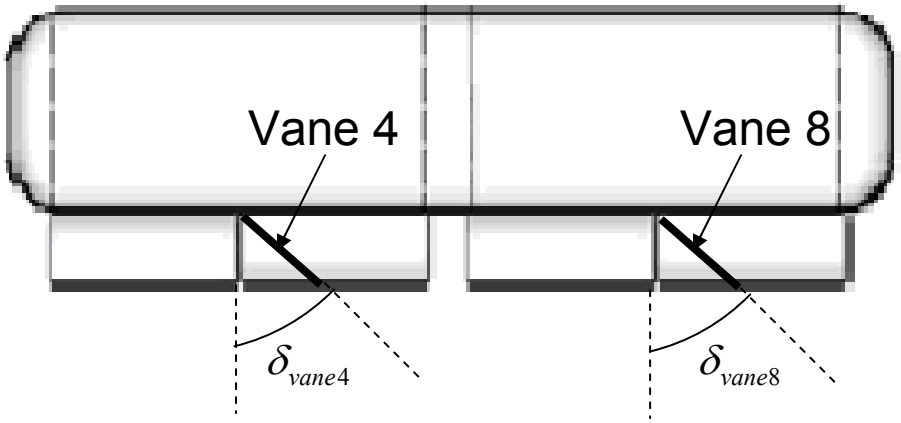

Figure 2-6 Right Side View with Deflections of Vanes 4 and 8

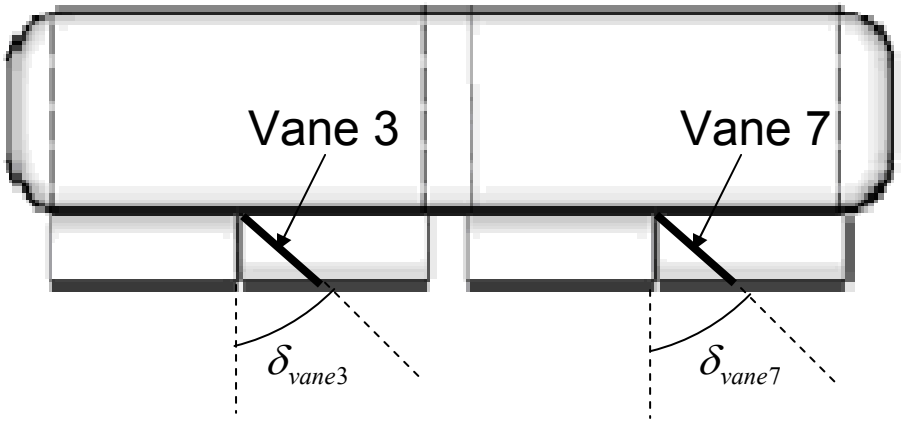

Figure 2-7 Left Side View with Deflections of Vanes 3 and 7

2.1.4 Control Vane Deflections

The control vane deflections are determined by calculating three different contributors. Equation 2.8 and 2.9 show the longitudinal and lateral directional vane deflections, respectively.

$$\delta_X = \delta_{X_{Sched}} + \delta_{X_{control}} + \delta_{X_{swirl}} \tag{2.8}$$

$$\delta_Y = \delta_{Y_{Sched}} + \delta_{Y_{control}} + \delta_{Y_{swirl}} \tag{2.9}$$

The contributions to vane deflection include the scheduled deflection, control deflection, and swirl compensation deflection.

For forward velocities, the flow passing through the fan will be skewed in the direction of the oncoming flow causing the vane to experience an angle of attack even when it is not deflected, as seen in Figure 2-8. In research by Tobias and Horn [5] a linear schedule of δ_{Xsched} with forward velocity was found with a slope of 0.26°/fps. This scheduled deflection is applied to longitudinal vanes only, in order to avoid stall occurring on the vanes at high speeds.

Since the spinning rotor creates a drag force, the flow tends to swirl within the duct. The flow has a rotational speed due to this swirl effect that is proportional to the rotor torque [18]. Therefore the swirl effects will produce undesired angle of attack on the vanes, so δ_{Xswirl} and δ_{Yswirl} are constant biases to the deflection to minimize swirl effects. Tobias and Horn showed that 8° of deflection was appropriate to reduce the swirl effect [5].

The control deflections, $\delta_{Xcontrol}$ and $\delta_{Ycontrol}$ are then used to provide the desired yaw moment, X force, or Y force control effect.

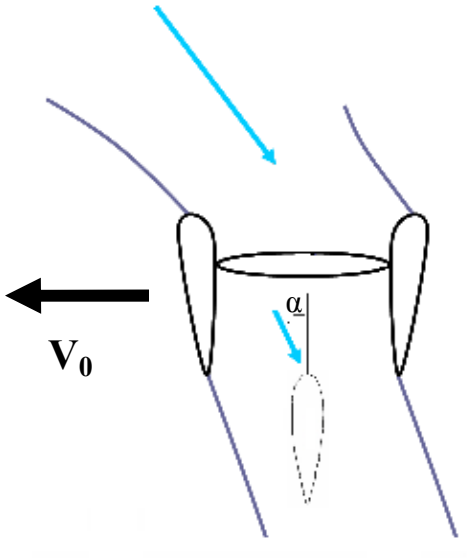

Figure 2-8 Angle of Attack Experienced on the Undeflected Vane in Forward Motion

2.2 Physical Properties of Aircraft Modeled

This study focuses on a tandem ducted fan configuration. Because differential thrust provides higher longitudinal control authority than a single ducted fan, the tandem configuration is more widely considered for manned operations. The aircraft modeled in the generic ducted fan simulation model is a full-size tandem ducted fan with four control vanes placed in the exit flow of each duct. Rotors are modeled to be rigid and with a front rotor rotating counter-clockwise and an aft rotor rotating clockwise. A generic tandem ducted fan aircraft suitably sized for manned and unmanned flight, is used for this study. A schematic of the configuration of the vehicle can be seen in Figure 2-9 and the geometric properties of the tandem ducted fan configuration are given at Table 2-1.

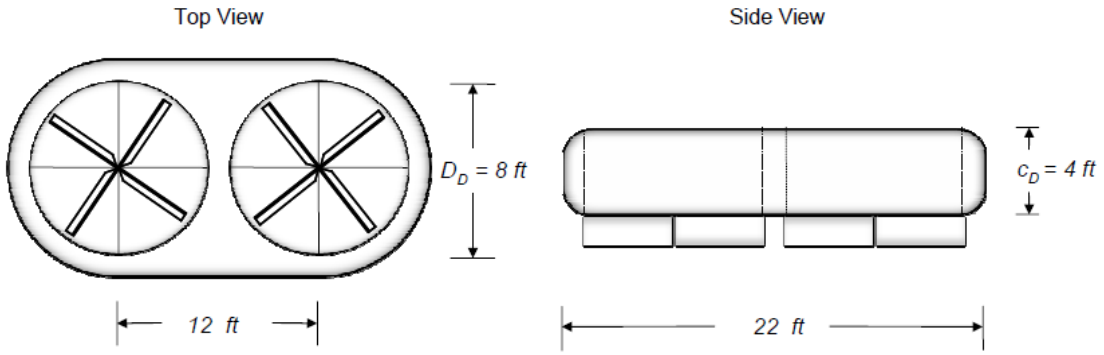

Figure 2-9 Tandem Ducted Fan Aircraft Schematic

Property	Value	Unit
Weight of the Aircraft	4000	lbs
Rotor RPM	188.5	rad/sec
Rotor Blade Twist Angle	-10.0	deg
Rotor Blade Radius	4.0	ft
Rotor Blade Chord	0.6	ft
Each Control Vane Surface Area	9.0	ft^2
Number of Blade	6	-

Table 2-1 Tandem Ducted Fan Geometric Configuration Properties

2.3 Updates of Simulation Model

The design of the control law in Simulink for the generic ducted fan simulation model required particular modifications on the model generated by Tobias and Horn [5]. The need of these modifications came up from the desire of speeding up the simulations performed in Simulink, because some elements of the generic ducted fan simulation model require considerable amount of computation time.

An important part of the generic ducted fan simulation model is the inflow calculation. Ducted rotors have significantly different inflow behavior compared to an

open rotor. The duct influences the inflow calculation both by producing an additional lift, which affects wake contraction and the flow turning effect. The inflow model developed by Tobias and Horn [5] is a modification of basic momentum theory inflow of an open rotor, which is described in Section 2.1.1.

The method developed to calculate inflow by Tobias and Horn [5] was an iterative numerical method using Newton-Raphson iteration process, which had specific weaknesses. For example, for some desired trim conditions, the numerical solution can diverge. Another issue was that the execution time required for a single iteration. In every time step, the simulation model requires on the order of 100-200 of iterations to solve. Repeating calculations over and over again is unnecessary; therefore, in order to speed up the simulation, the inflow calculation was performed offline for all possible conditions and a lookup table is generated. In order to obtain a wider use of the lookup table, non-dimensional inplane and vertical velocities with respect to rotors ($\mu_{inplane}$ and μ_z) and vertical thrust which is normalized with air density and rotor disk area ($\overline{T}$) are used. It must be noted that normalized thrust is not non-dimensional, for the ease of calculation thrust is not normalized with the square of tip velocity. Equations 2.10 to 2.12 define variables used in the lookup table and Table 2-2 shows the range of the variables.

$$\mu_{inplane} = \frac{V_{inplane}}{\Omega R} \tag{2.10}$$

$$\mu_Z = \frac{V_Z}{\Omega R} \tag{2.11}$$

$$\overline{T} = \frac{T}{\rho A_D} \tag{2.12}$$

Variable	Minimum Value	Maximum Value	Step Size
$\mu_{inplane}$	0	0.1	0.005
μ_z	-0.1	0.1	0.005
$\overline{T}$ (ft^2/s^2)	5000	50000	500

Table 2-2 Inflow Lookup Table Parameters

Although it took a long time to generate the lookup table for various cases, the simulation model speeded up for all subsequent calculations. MATLAB's interpolation script is modified and embedded in the simulation model for the use of lookup table generated.

In order to increase the speed of the generic ducted fan simulation code vectorization is applied. Code vectorization is a method in which instead of loops to form arrays, vector and matrix multiplications are used. MATLAB is a matrix processing language that utilizes very efficient built-in operations for data contained in arrays [19]. Vectorization of the code can sometimes be difficult to develop and understand, rather than the using loops, but it increases the efficiency and performance of the program considerably.

Chapter 3

Basic Control System

This chapter describes how Translational Rate Command (TRC) response achieved in lateral, longitudinal and vertical axes and how Rate Command Heading Hold (RCHH) response achieved in yaw axis using the basic closed loop control system designed for the generic ducted fan simulation model. The designed controller has two loops; outer and inner. Outer loop is used to control translational motion and inner loop is used to control the attitude of the aircraft.

3.1 Open Loop Response

Before designing the controller, the open loop response was studied. Open loop eigenvalues for the linearized model of the generic ducted fan simulation model for hover and 20 ft/s forward and sideward flight are shown in Figure 3-1. The linearized model obtained from the non-linear model about the operating point using the small perturbation method. It must be noted that the inflow dynamics are excluded from the figure, since the inflow dynamics are lot faster than the rest of the dynamics and stable. So this study concentrates on the rest of the dynamics of the aircraft. It can be observed that open loop response has unstable eigenvalues and unexpectedly eigenvalues move more to positive side with increasing velocity, unlike the dynamics of a conventional helicopter [20]. This is a result of the nonlinear dynamics of the ducted fan aircraft and the high pitch and roll moments created by the duct around the rotor. Dynamics with the higher frequency unstable eigenvalues correspond to the lateral dynamics of the aircraft and the lower frequency unstable eigenvalues correspond to the longitudinal dynamics.

The open loop response of the simulation model to a 5% longitudinal impulse input is studied first. Figure 3-2 shows the behavior of the aircraft for an impulsive disturbance. It can be observed that open loop response is statically unstable. The unstable oscillations have a period about 10 seconds that can be compared with the eigenvalue of the

longitudinal dynamics. Imaginary part of the unstable longitudinal dynamics is 0.433 and the corresponding period of oscillations is 14.5 seconds, which is comparable with the response obtained from the simulation model. Figure 3-3 and 3-4 shows the response of the vehicle for 1% and 5% longitudinal step input, respectively, starting from a trimmed hovering condition. Clearly the vehicle dynamics are highly unstable, and the vehicle reaches high pitch attitudes in short time.

The longitudinal control sensitivity is also very high. The vehicle needs a large range of control power to trim in forward flight conditions to overcome the pitching moment effects on the ducts, but the excess control power also results in excessive control sensitivity when operating around a trim point.

The vehicle would clearly benefit from a feedback augmentation to stabilize the dynamics and produce predictable response characteristics.

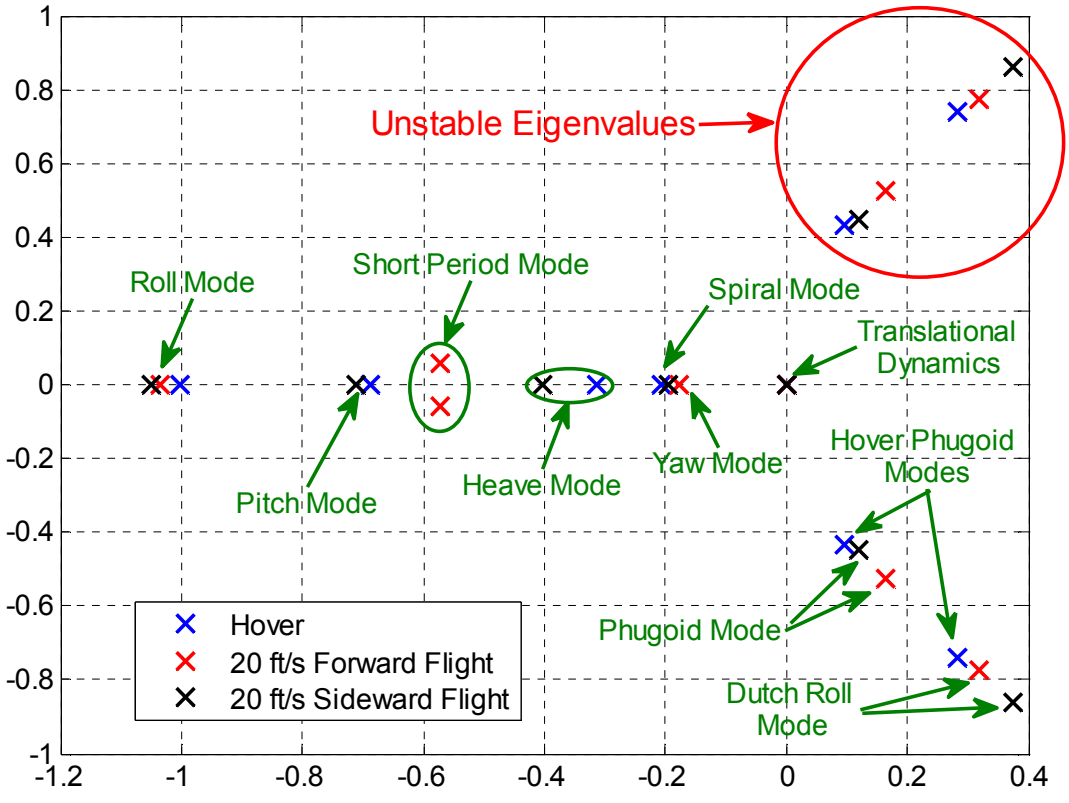

Figure 3-1 Eigenvalues of the Simulation Model

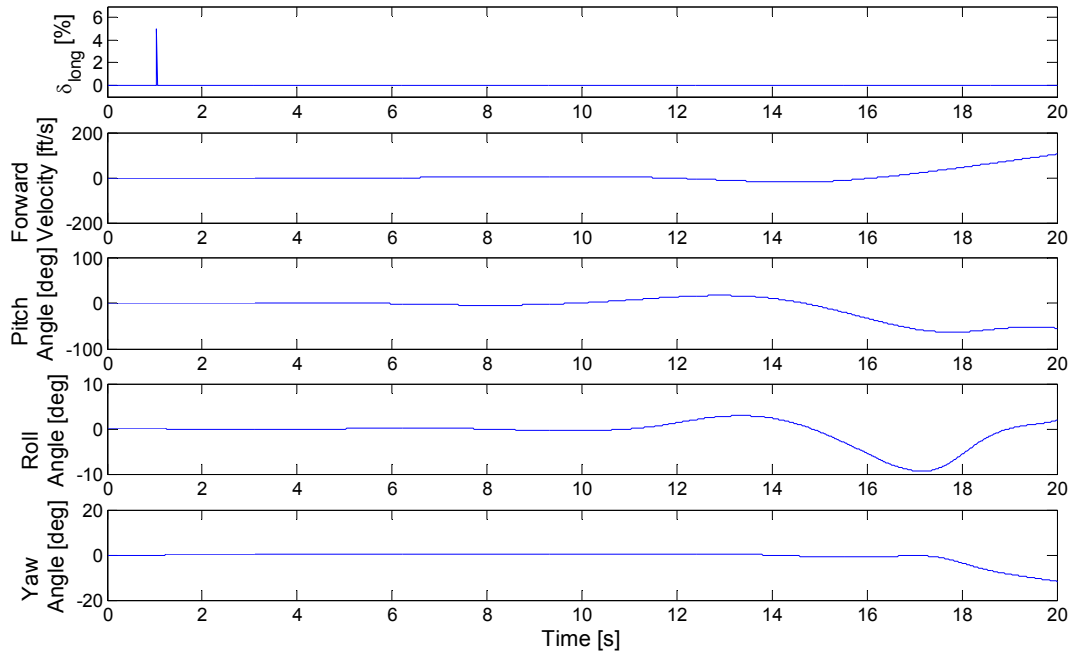

Figure 3-2 Open Loop Response of the Generic Ducted Fan Simulation Model for 5% Longitudinal Control Impulse

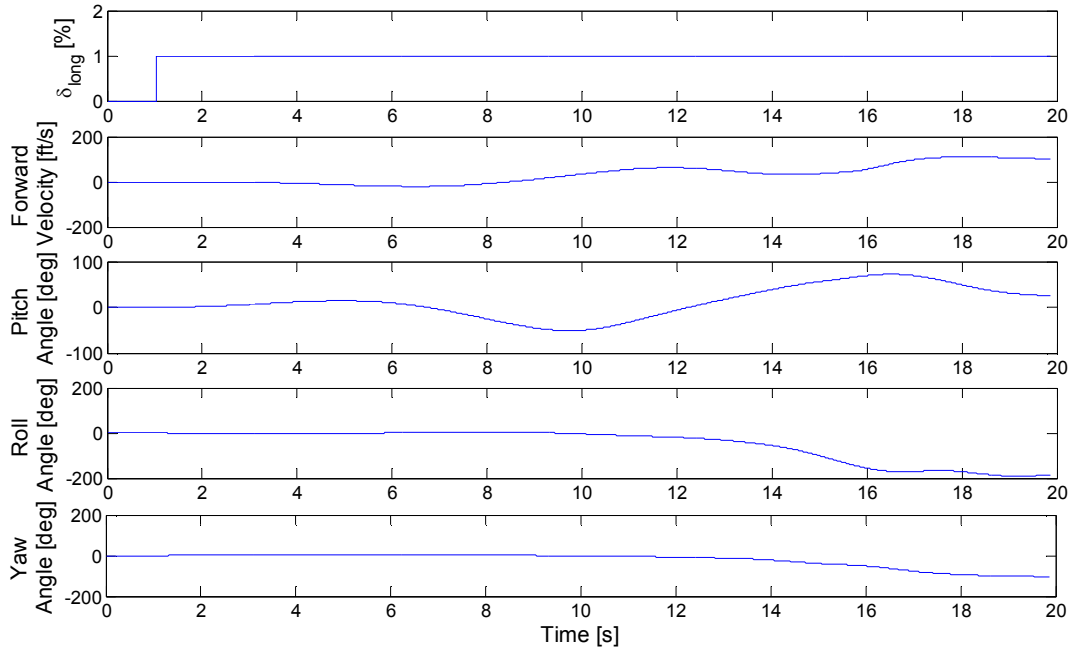

Figure 3-3 Open Loop Response of the Generic Ducted Fan Simulation Model for 1% Longitudinal Control Step Input

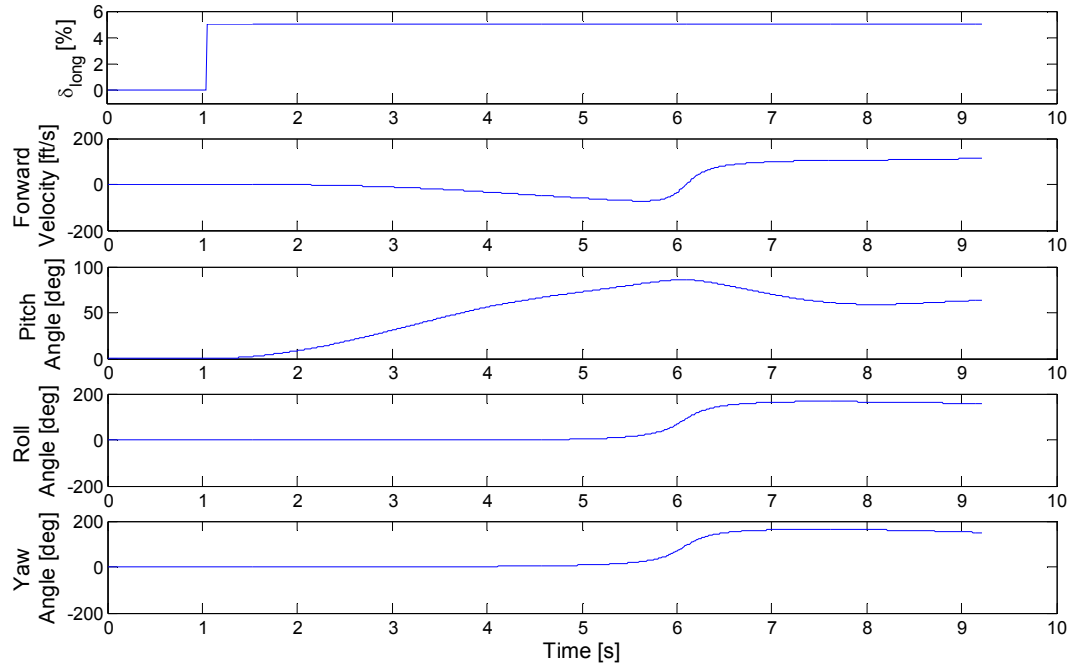

Figure 3-4 Open Loop Response of the Generic Ducted Fan Simulation Model for 5% Longitudinal Control Step Input

3.2 Inner Loop of the Controller

The inner loop is responsible for attitude dynamic and consists of model inversion and state feedback controllers, which are implemented as shown in Figure 3-5. Commands for roll and pitch attitude, heave velocity command and vane commands are inputs to the inner loop. These inputs are normally generated from the outer loop control as discussed in Section 3.3. Yaw rate command would be provided by a pilot in a manned vehicle but could be provided by a heading command outer loop control law on an unmanned aircraft. The aircraft attitude and angular rates are fed back to the inner loop. The linear model of the open loop dynamics of tandem ducted fan was obtained from the mathematical model developed by Tobias and Horn [5]. The linearized model includes the inflow dynamics, which is faster than the flight dynamics. In order to reduce the order of the model, faster dynamics assumed to reach steady state immediately, therefore inflow terms can be removed from the state vector.

$$\vec{x} = \begin{bmatrix} \vec{x}_f \\ \vec{x}_i \end{bmatrix} \tag{3.1}$$

where

$$\vec{x}_f = \{u \quad v \quad w \quad p \quad q \quad r \quad \phi \quad \theta \quad \psi \quad X \quad Y \quad Z\}^T \tag{3.2}$$

$$\vec{x}_i = \{\upsilon_{i_1} \quad \upsilon_{i_2}\}^T \tag{3.3}$$

The linearized model is written as in Equation 3.4.

$$\begin{bmatrix} \dot{\vec{x}}_f \\ \dot{\vec{x}}_i \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} \vec{x}_f \\ \vec{x}_i \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} \vec{u} \tag{3.4}$$

Since we assume $\dot{\vec{x}}_i$ is equal to zero, because of its fast dynamics, we can obtain 12-by-12 model by removing the inflow dynamics. The new A and B matrices can be calculated by simple algebra and shown in Equation 3.5.

$$\begin{aligned} A_r &= A_{11} - A_{12}\left(A_{22}\right)^{-1} A_{21} \\ B_r &= B_1 - A_{12}\left(A_{22}\right)^{-1} B_2 \end{aligned} \tag{3.5}$$

From the reduced model above, angular rate dynamics and body Z velocity dynamics were picked and a fourth order model with lateral, longitudinal, pedal (yaw) and collective inputs for control variables as shown in Equation 3.6 was developed. Also the linearized model includes the effect of the vanes as redundant effectors.

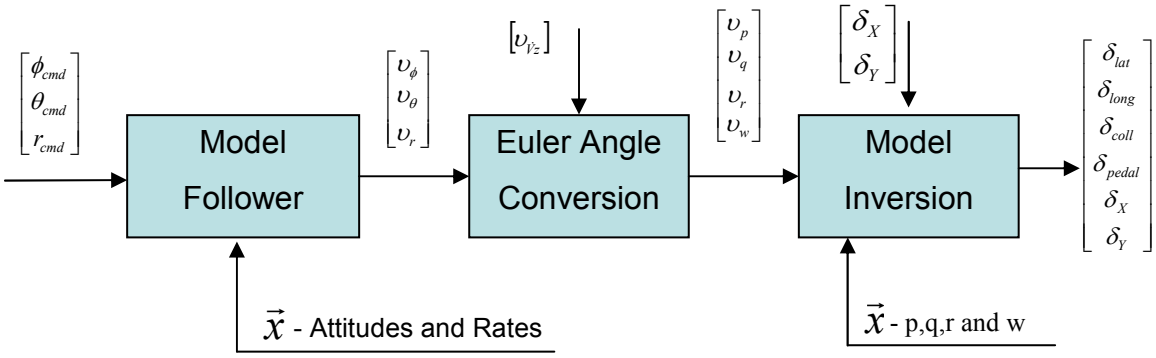

Figure 3-5 The Inner Loop

$$\begin{bmatrix}\dot{p}\\ \dot{q}\\ \dot{r}\\ \dot{w}\end{bmatrix}=\begin{bmatrix}L_p & L_q & L_r & L_w\\ M_p & M_q & M_r & M_w\\ N_p & N_q & N_r & N_w\\ Z_p & Z_q & Z_r & Z_w\end{bmatrix}\begin{bmatrix}p\\ q\\ r\\ w\end{bmatrix}+$$
$$\begin{bmatrix}L_{\delta_{lat}} & L_{\delta_{long}} & L_{\delta_{pedal}} & L_{\delta_{coll}}\\ M_{\delta_{lat}} & M_{\delta_{long}} & M_{\delta_{pedal}} & M_{\delta_{coll}}\\ N_{\delta_{lat}} & N_{\delta_{long}} & N_{\delta_{pedal}} & N_{\delta_{coll}}\\ Z_{\delta_{lat}} & Z_{\delta_{long}} & Z_{\delta_{pedal}} & Z_{\delta_{coll}}\end{bmatrix}\begin{bmatrix}\delta_{lat}\\ \delta_{long}\\ \delta_{pedal}\\ \delta_{coll}\end{bmatrix}+\begin{bmatrix}L_{\delta_X} & L_{\delta_Y}\\ M_{\delta_X} & M_{\delta_Y}\\ N_{\delta_X} & N_{\delta_Y}\\ Z_{\delta_X} & Z_{\delta_Y}\end{bmatrix}\begin{bmatrix}\delta_x\\ \delta_y\end{bmatrix} \quad (3.6)$$

Model follower block has the command filters, calculating the pseudo commands for the model inversion. The model follower is governed by second order command filters for roll and pitch axes and first order command filter for yaw axis as shown in Figure 3-6. The attitude responses are obtained as roll and pitch angles and as yaw rate. The pseudo commands are calculated as shown in Equations 3.7 to 3.9. A second order command filter is also used for the desired vertical acceleration in the outer loop (See Section 3.3). Natural frequency, damping ratio and time constant of the command filters are selected by the roll and pitch guidance from ADS-33. Values used for command filters and PID compensator gains are tabulated in Table 3-1.

In the roll and pitch axes, second derivatives of roll and pitch Euler angles must be converted to body axis angular accelerations as well as the desired vertical acceleration in the inertial frame to body axis acceleration using the transformation equations shown in Equations 3.10 to 3.12 [15].

	ω_n (rad/s)	ξ	τ (s)	K_P	K_I	K_D
Roll Attitude	3	0.9	-	9	-	5.4
Pitch Attitude	3	0.8	-	9	-	4.8
Yaw Rate	2.5	-	0.4	4.5	6.25	-

Table 3-1 Command Filter Variables and Compensator Gains

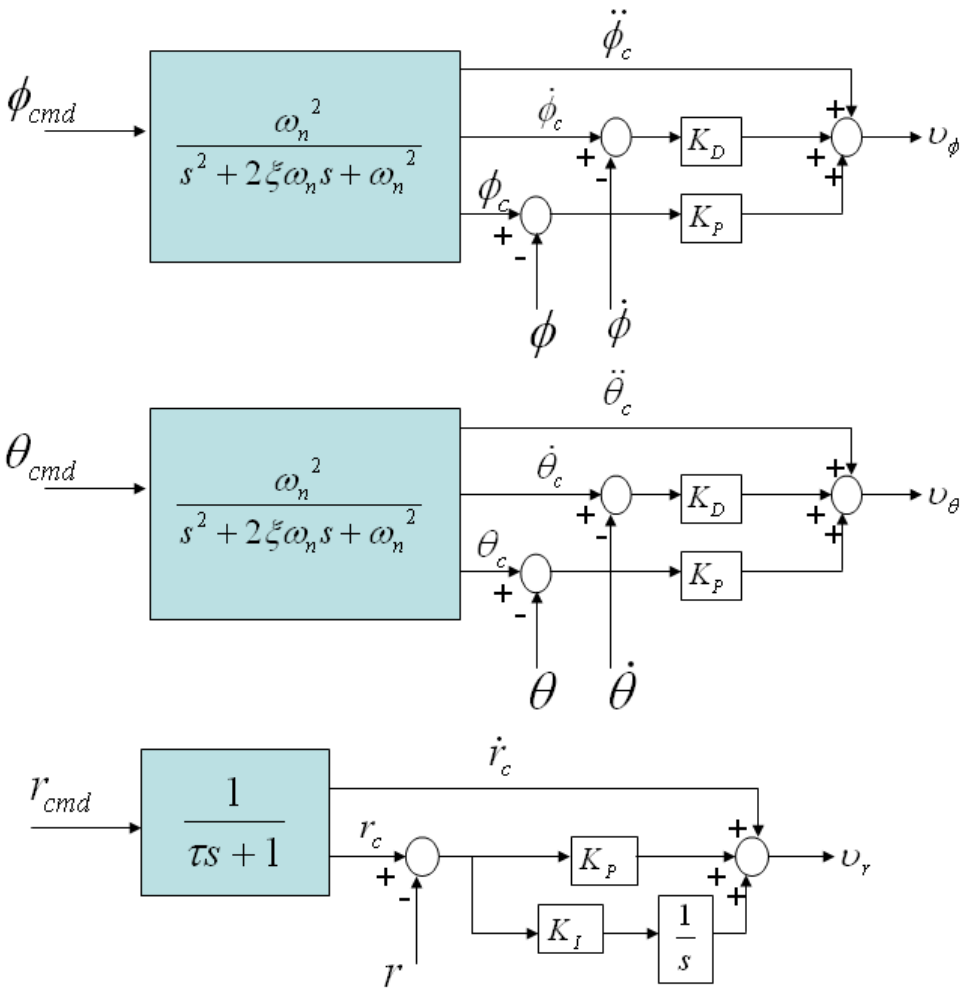

Figure 3-6 Model Follower Design for Inner Loop

$$\upsilon_\phi = \ddot{\phi}_c + K_P(\phi_c - \phi) + K_D(\dot{\phi}_c - \dot{\phi}) \tag{3.7}$$

$$\upsilon_\theta = \ddot{\theta}_c + K_P(\theta_c - \theta) + K_D(\dot{\theta}_c - \dot{\theta}) \tag{3.8}$$

$$\upsilon_r = \dot{r}_c + K_P(r_c - r) + K_I\int(r_c - r)dt \tag{3.9}$$

$$\upsilon_p = \upsilon_\phi - \upsilon_\theta \tan\phi\tan\theta + \upsilon_r\frac{\tan\theta}{\cos\phi} + \dot{\psi}\dot{\theta}\frac{1}{\cos\theta} + \dot{\theta}\dot{\phi}\tan\theta + \dot{\psi}\dot{\phi}\sin\theta\tan\phi \tag{3.10}$$

$$\upsilon_q = \upsilon_\theta\frac{1}{\cos\phi} + \upsilon_r\tan\phi + \dot{\psi}\dot{\phi}\frac{\cos\theta}{\cos\phi} \tag{3.11}$$

$$\upsilon_w = \frac{\upsilon_{\dot{V}_z} + u\dot{\theta}\cos\theta}{\cos\theta\cos\phi} \tag{3.12}$$

The model inversion yields the control law given in Equation 3.13.

$$\begin{bmatrix} \delta_{lat} \\ \delta_{long} \\ \delta_{pedal} \\ \delta_{coll} \end{bmatrix} = [B_1]^{-1} \left(\begin{bmatrix} \upsilon_p \\ \upsilon_q \\ \upsilon_r \\ \upsilon_w \end{bmatrix} - [A] \begin{bmatrix} p \\ q \\ r \\ w \end{bmatrix} - [B_2] \begin{bmatrix} \delta_X \\ \delta_Y \end{bmatrix} \right) \tag{3.13}$$

The inversion model for the inner loop can be solved for each control commands. The system is fourth order and there are 4 controls to be solved. There is a unique solution; therefore there is no control allocation problem needed to be solved for the inner loop.

If this was a perfect representation of the aircraft dynamics then the tracking error, defined as Equation 3.14, would have dynamics governed by a simple set of differential equations, as shown in Equation 3.15. By a proper selection of gains, well-damped and stable tracking can be assured.

$$\begin{aligned} e_\phi &= \phi_c - \phi \\ e_\theta &= \theta_c - \theta \\ e_r &= r_c - r \end{aligned} \tag{3.14}$$

$$\begin{aligned} \ddot{e}_\phi + K_D \dot{e}_\phi + K_P e_\phi &= 0 \\ \ddot{e}_\theta + K_D \dot{e}_\theta + K_P e_\theta &= 0 \\ \ddot{e}_r + K_p \dot{e}_r + K_I e_r &= 0 \end{aligned} \tag{3.15}$$

3.3 Outer Loop of the Controller

The outer loop control law has inputs of forward, sideward and vertical velocities, which might be provided by the pilot (for a manned aircraft with translation rate command control response) or by an autopilot/path planner (for an unmanned aircraft).

The outer loop uses a model following inversion type of controller, where acceleration pseudo-commands ($\upsilon_{\dot{V}_X}$, $\upsilon_{\dot{V}_Y}$ and $\upsilon_{\dot{V}_Z}$) are calculated by a model follower. As shown in Figure 3-7, the model follower uses a first order command filter, with 2 seconds

of time constant (τ = 2 seconds), designed to meet desired response characteristics and a proportional-integral (PI) compensator (K_p=0.9 and K_I=0.25) to regulate tracking error. Compensator gains and command filter time constant is selected with the guidance of ADS-33 requirements. The command filter produces the desired state and state derivative response. The state tracking error is passed through a PI compensator, which is added to the desired response of the state derivative.

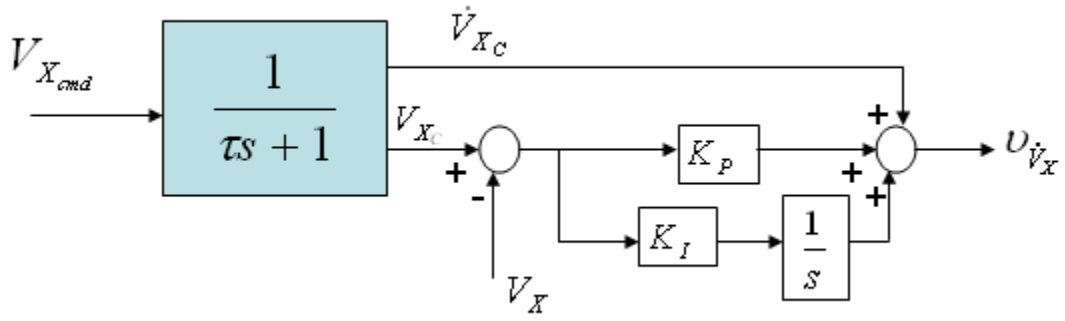

Figure 3-7 Model Follower Design for Outer Loop

For forward and sideward velocities, the acceleration pseudo-commands obtained from the model follower are sent to the outer loop inversion control, but the heave axis pseudo-command is sent directly to the inner loop, where it is converted into body coordinates and included in the inversion control law which is used to determine the desired collective command.

The forward and sideward motion dynamics are governed by Equation 3.16. In the inversion control law the acceleration terms are replaced by the pseudo-commands as shown in Equation 3.17. This equation must then be solved to find the combination of roll and pitch attitudes and longitudinal and lateral vane deflections to achieve the desired forward and sideward accelerations. However, the system of equations is an underdetermined system, where the unknowns are more than the number of equations. This leads to a need for control allocation for redundant control effectors. The control allocation method generated to overcome this problem is described in Chapter 4. The

control law will reduce to Equation 3.18, if vane control is taken out of the controller, so pitch and roll commands can be calculated without the need of a control allocation.

$$\begin{bmatrix}\dot{u}\\ \dot{v}\end{bmatrix}=\begin{bmatrix}X_u & X_v & X_w\\ Y_u & Y_v & Y_w\end{bmatrix}\begin{bmatrix}u\\ v\\ w\end{bmatrix}+ \\ \begin{bmatrix}X_{\delta_{lat}} & X_{\delta_{long}} & X_{\delta_{coll}}\\ Y_{\delta_{lat}} & Y_{\delta_{long}} & Y_{\delta_{coll}}\end{bmatrix}\begin{bmatrix}\delta_{lat}\\ \delta_{long}\\ \delta_{coll}\end{bmatrix}+\begin{bmatrix}X_{\delta_X} & X_{\delta_Y}\\ Y_{\delta_X} & Y_{\delta_Y}\end{bmatrix}\begin{bmatrix}\delta_X\\ \delta_Y\end{bmatrix}+\begin{bmatrix}0 & -g\\ g & 0\end{bmatrix}\begin{bmatrix}\phi_{cmd}\\ \theta_{cmd}\end{bmatrix} \tag{3.16}$$

$$[B_1]\begin{bmatrix}\delta_X\\ \delta_Y\end{bmatrix}+[B_2]\begin{bmatrix}\phi_{cmd}\\ \theta_{cmd}\end{bmatrix}=\begin{bmatrix}\upsilon_X\\ \upsilon_Y\end{bmatrix}-[A_1]\begin{bmatrix}u\\ v\\ w\end{bmatrix}-[A_2]\begin{bmatrix}\delta_{lat}\\ \delta_{long}\\ \delta_{coll}\end{bmatrix} \tag{3.17}$$

$$\begin{bmatrix}\phi_{cmd}\\ \theta_{cmd}\end{bmatrix}=[B_2]^{-1}\left(\begin{bmatrix}\upsilon_X\\ \upsilon_Y\end{bmatrix}-[A_1]\begin{bmatrix}u\\ v\\ w\end{bmatrix}-[A_2]\begin{bmatrix}\delta_{lat}\\ \delta_{long}\\ \delta_{coll}\end{bmatrix}\right) \tag{3.18}$$

3.4 Simulation Model with Controller

The controller consists of inner and outer loop, the combination of both and the generic ducted fan simulation model are implemented in Simulink, and the block diagram of the system is shown in Figure 3-8. The actual Simulink diagrams can be found in detail at Appendix A .

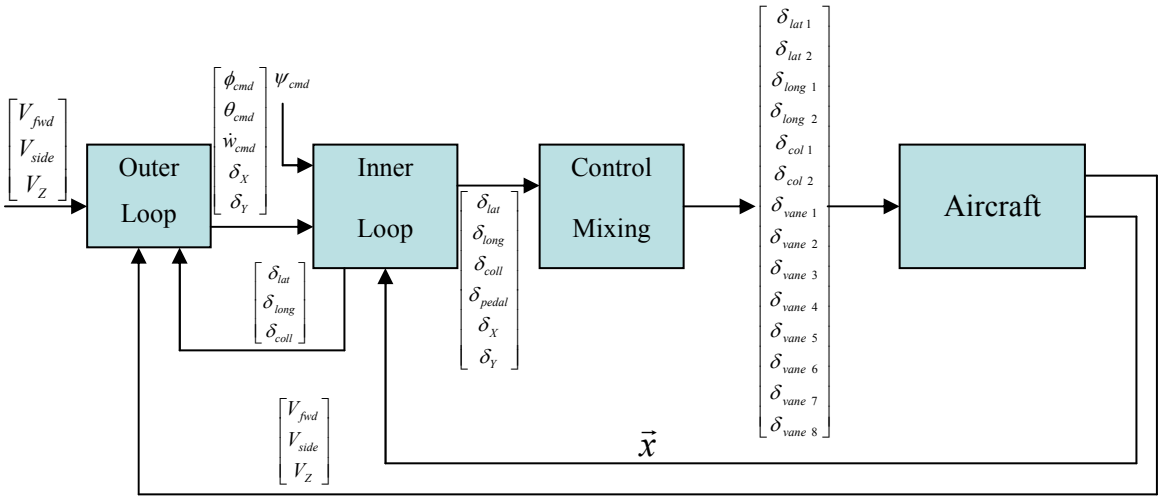

Figure 3-8 The Block Diagram of the Controller

3.5 Preliminary Results

This section shows the results of the closed loop controller designed for the generic ducted fan simulation model with the vanes effective.

3.5.1 Forward Motion Command Tracking

The controller was analyzed for tracking of a commanded change in forward velocity. The tracking for 15 ft/s and 20 ft/s commands are observed to be performed satisfactorily. Figures 3.9 and 3.11 show the on axis responses for 15 ft/s and 20 ft/s forward velocity commands, whereas Figures 3.10 and 3.12 show the off axes responses. Achieving a good tracking performance for a desired command and low off axis response shows that vane control can be useful. One might ask, what is the benefit of using a controller which employs vanes instead of a controller which does not use vanes? This question is answered in Chapter 4.

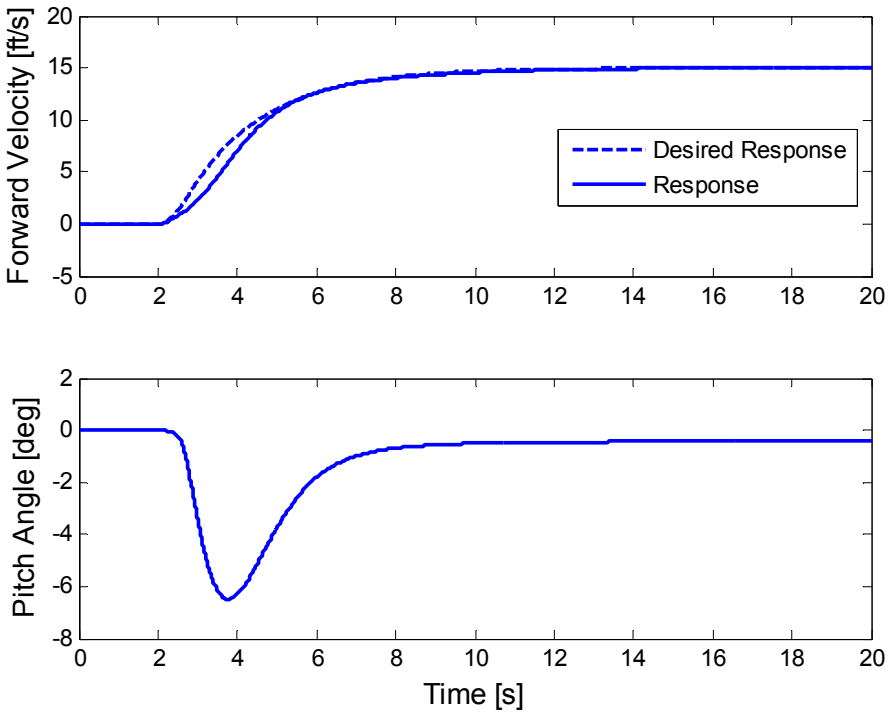

Figure 3-9 On Axis Response for Forward Velocity Tracking of 15 ft/s

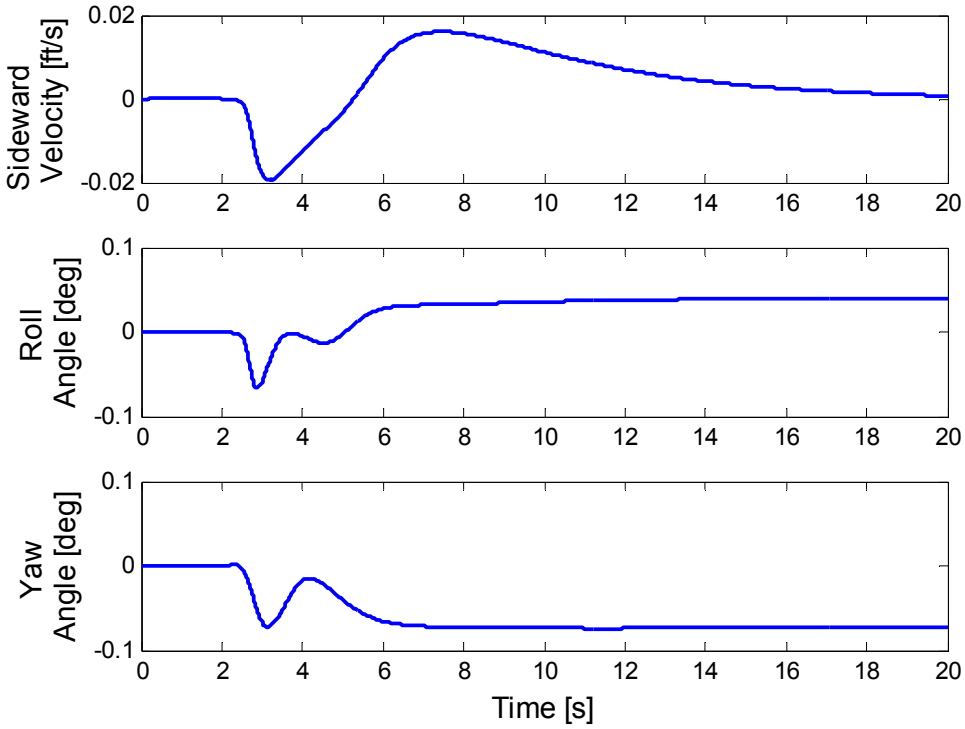

Figure 3-10 Off Axis Response for Forward Velocity Tracking of 15 ft/s

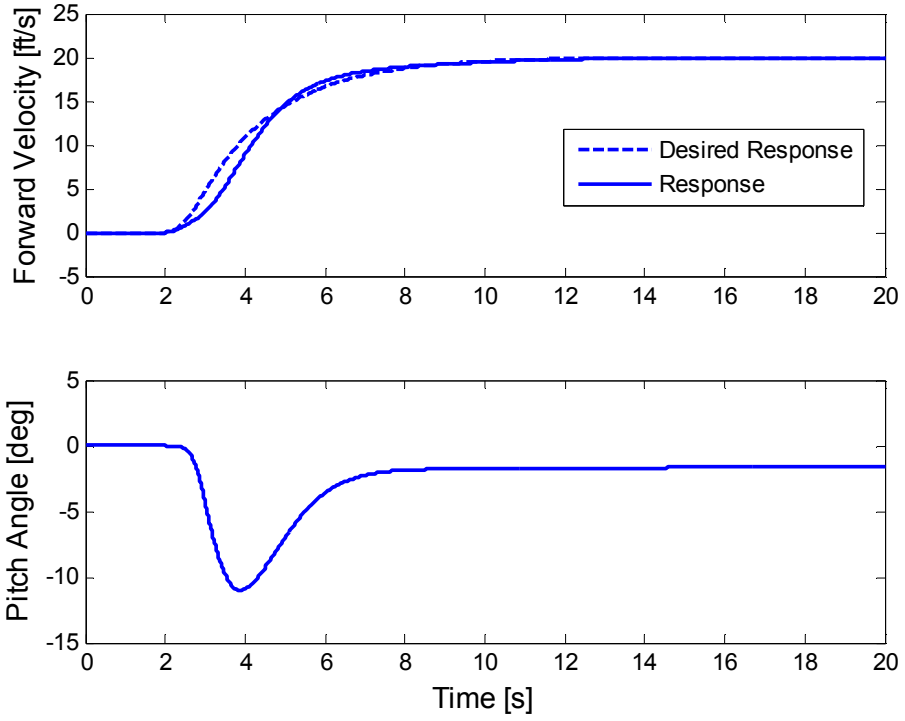

Figure 3-11 On Axis Response for Forward Velocity Tracking of 20 ft/s

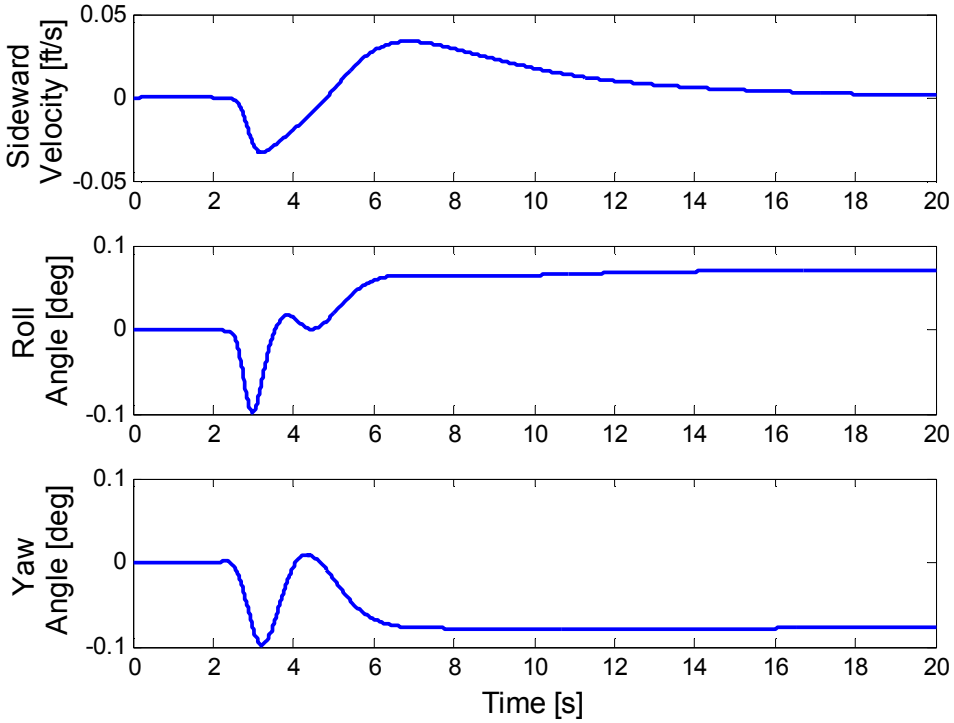

Figure 3-12 Off Axis Response for Forward Velocity Tracking of 20 ft/s

3.5.2 Sideward Motion Command Tracking

Similar to forward flight, sideward flight is also analyzed for on and off axes responses. Results are illustrated in Figures 3-13 and 3-14, where a 10 ft/s step input is commanded for sideward velocity. As observed, sideward flight tracking capabilities are also satisfactory like the forward flight case and the off axes responses are very low.

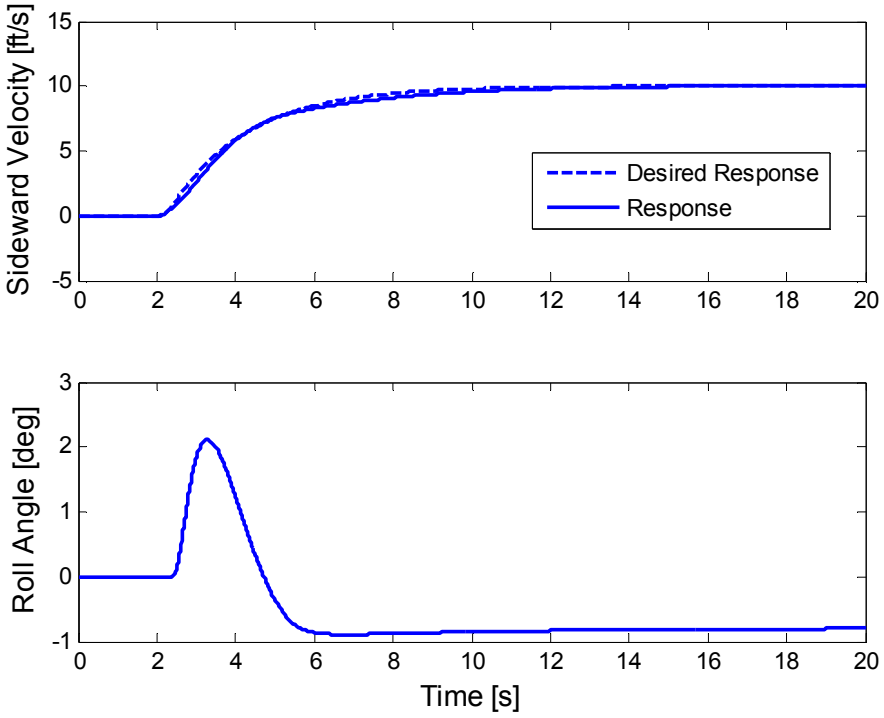

Figure 3-13 On Axis Response for Sideward Velocity Tracking of 10 ft/s

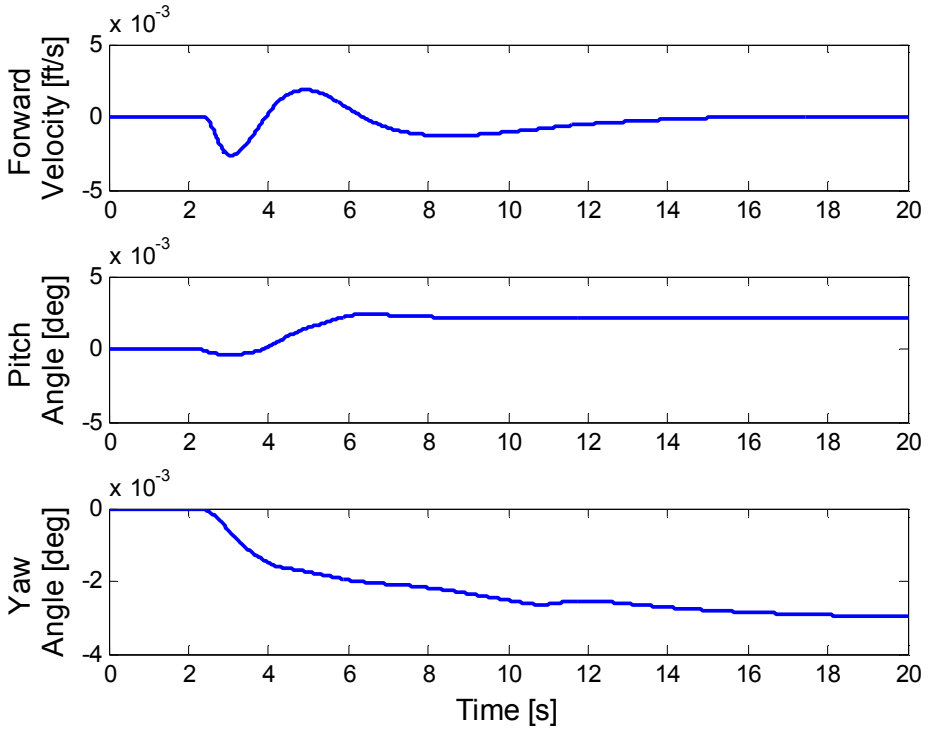

Figure 3-14 Off Axis Response for Sideward Velocity Tracking of 10 ft/s

Chapter 4

Method for Including Redundant Control Effectors to Model Following Controller

This chapter describes the control allocation developed to include the redundant control effectors to the model following controller designed for the generic ducted fan simulation model. Also the comparison between a controller with vane control and a controller without vane control is shown in this chapter.

4.1 Control Allocation

The object of this study is to obtain forward and sideward motion with minimum pitch and roll attitude change, and possibly even no attitude change for low-speed translation. The algorithm generated for this controller, therefore, aims to obtain desired pseudo accelerations with minimum attitude commands by using redundant effectors.

In order to achieve this goal, the controller sets pitch command and roll command to zero and calculates the lateral and longitudinal vane commands. The control component of the vane deflections ($\delta_{X_{control}}$) is limited to ±10° to avoid stalling the control surface. When a vane reaches this limit, the controller sets δ_X or δ_Y to the limit value and calculates the pitch or roll command to achieve the commanded acceleration. Therefore, command for attitude change is avoided until some speed where vane deflections are not enough anymore. The control allocation for redundant control effectors can be achieved by this algorithm, which is also shown in Figure 4-1.

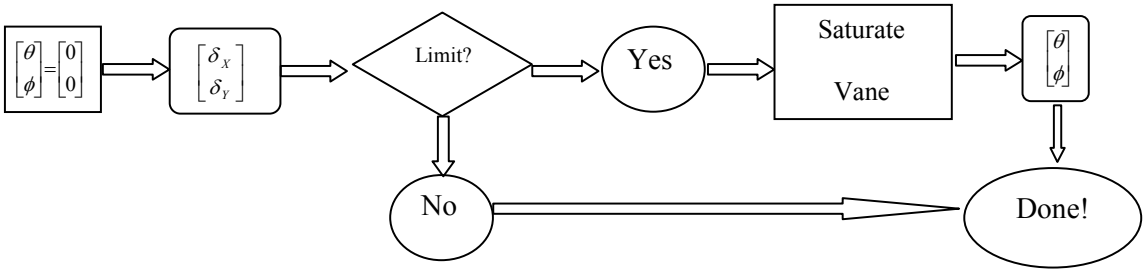

Figure 4-1 Control Allocation for Redundant Control Effectors

4.2 Effect of Vane Control

In this section the effect of vane control is studied. Similar to the previous chapter, forward velocity and sideward velocity tracking performed for both with vane control and without vane control. Also variation of pitch attitude for different forward velocities studied.

4.2.1 Forward Velocity Tracking

Figure 4-2 and 4-3 show the tracking of forward velocity command with and without vane control. It can be observed that commanded forward speed is achieved with considerably lower pitch attitude with vane control. Vanes are saturated at 10 degrees of deflection, so variation at pitch attitude is observed after saturation.

4.2.2 Sideward Velocity Tracking

Figure 4-4 shows the tracking of sideward velocity command with and without vane control. It can be observed that commanded sideward speed is achieved with considerably lower roll attitude with vane control, similar to the case with forward velocity command. Vanes are saturated at 10 degrees of deflection, so variation at roll attitude is observed after saturation.

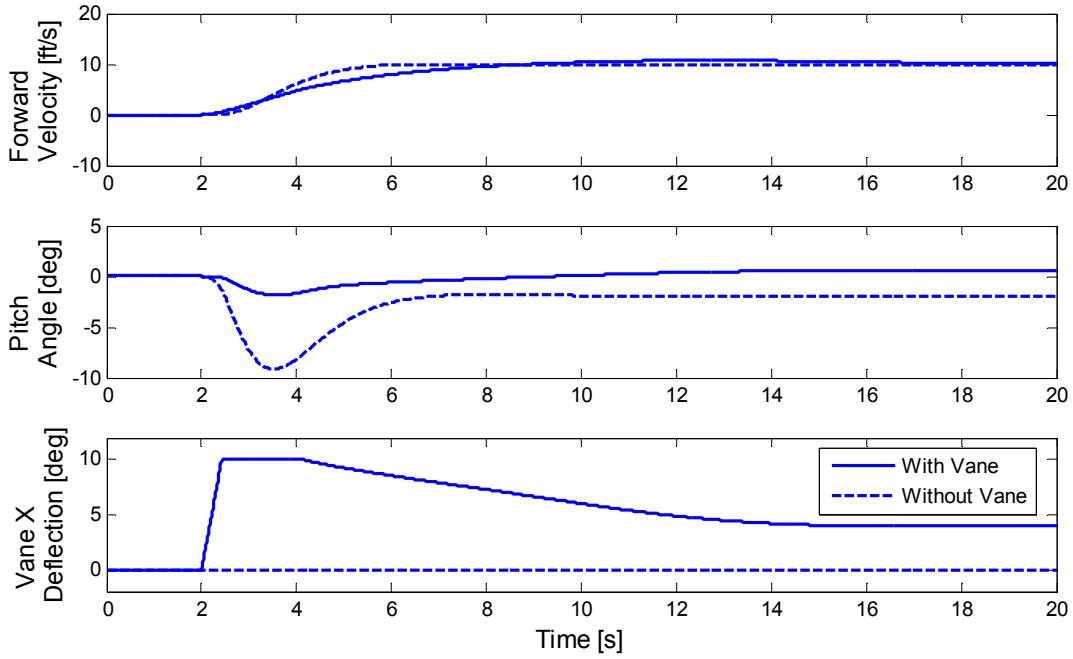

Figure 4-2 Effect of Vane Control on Pitch Attitude for 10 ft/s

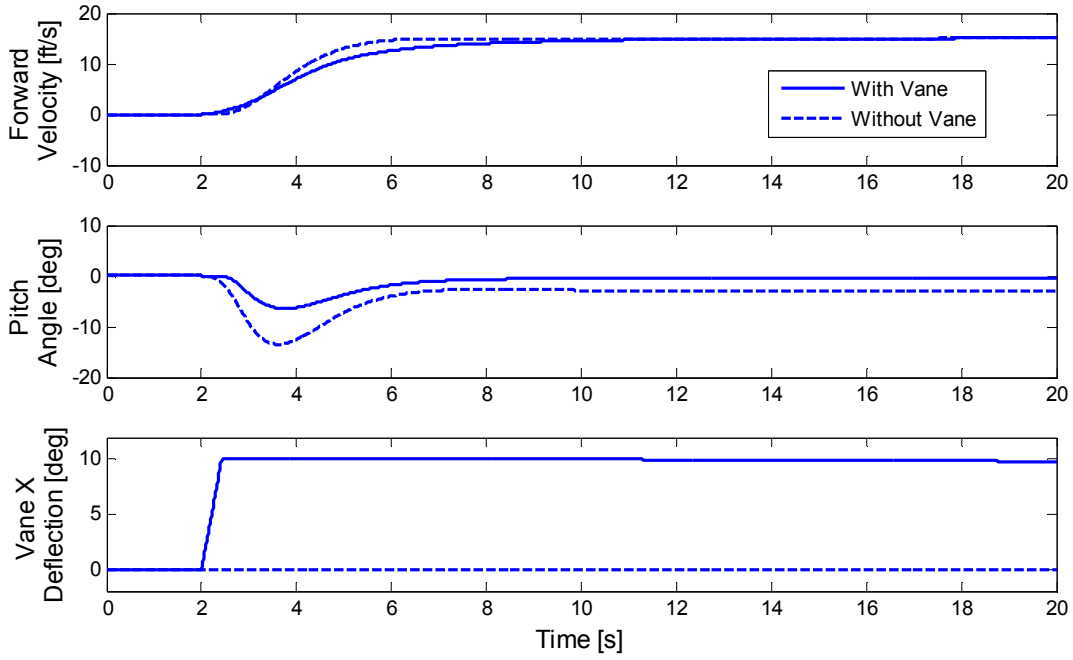

Figure 4-3 Effect of Vane Control on Pitch Attitude for 15 ft/s

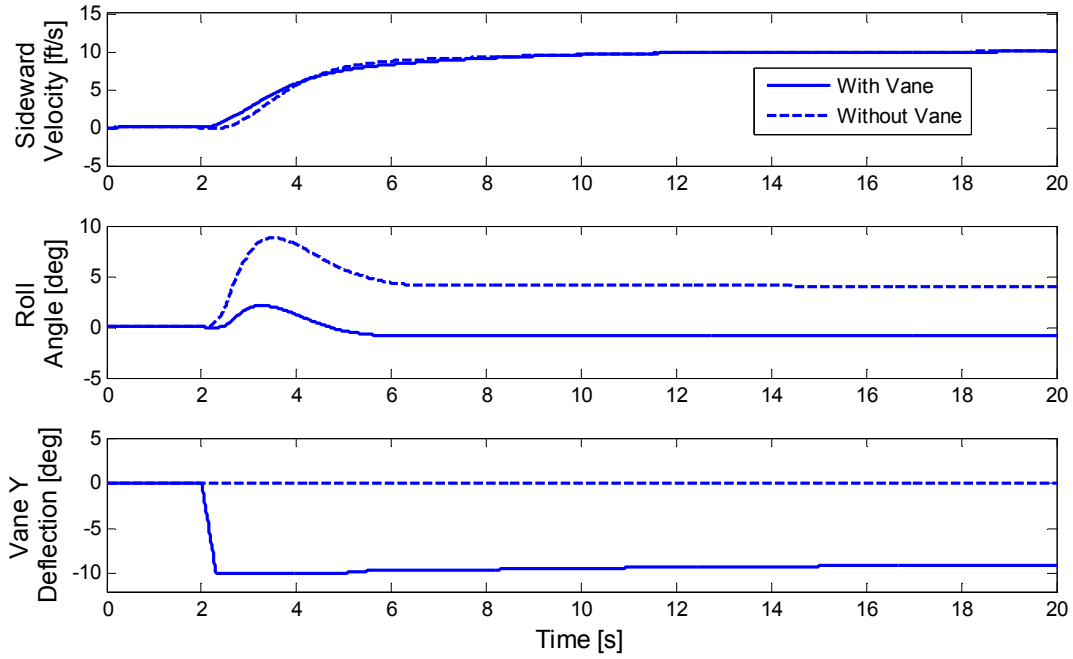

Figure 4-4 Effect of Vane Control on Roll Attitude for 10 ft/s

4.2.3 Pitch Attitude Variation for Varying Forward Velocities

The variation of the pitch attitude for varying forward velocity step inputs was studied in this section. It can be observed on Figure 4-5, by using vane control, up to 20 ft/s, the pitch attitude required is less than 5°. It must be noted that, in Figures 4-2, 4-3 and 4-4 the vanes seem to be saturated at 10°, the total vane deflections can be higher than that value. Recall that vane deflections are defined in Equations 2.8 and 2.9, and the values shown here are the control vane deflection ($\delta_{X_{control}}$). However, vanes are not stalled since other component account for swirl wake skew.

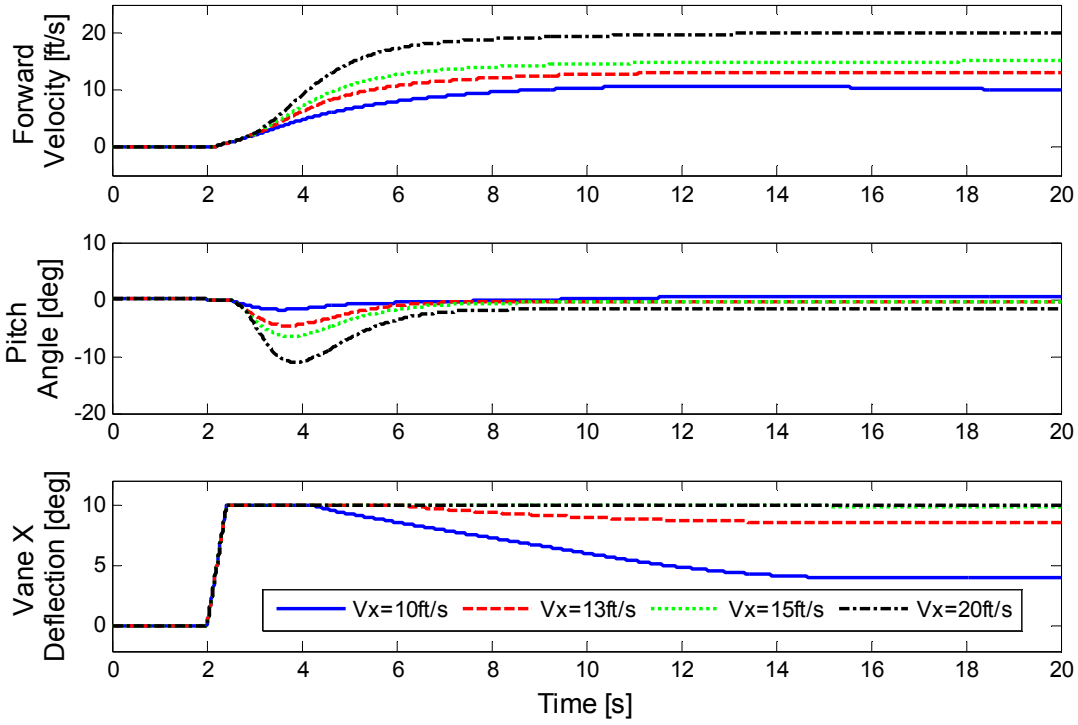

Figure 4-5 Effect of Vane Control on Pitch Attitude for Varying Forward Velocities

Chapter 5

Dynamic Manuevers and Gust Rejection

5.1 Gust Response

The controller for the generic ducted fan simulation is designed to reject gusts due to typical wind and atmospheric turbulence. In order to observe the performance of gust rejection 2 different gust conditions studied. One is with gust only from north and the other one is with gust from northeast, so gust velocity disturbs both longitudinal and lateral dynamics. For both cases, the aircraft has an initial heading towards the North direction. Therefore, for the first case, the gust is from the front of the aircraft (headwind) and for the second case, the gust is from 45° right of the aircraft.

Figure 5-1 shows the on axis response of a hovering tandem ducted fan aircraft for turbulent gust of 15 ft/s mean velocity from front of the aircraft (headwind) with and without vane control. Figure 5-2 shows the off axis response for the same aircraft and for the same gust and flight condition. Gust rejection observed to be better with vane control for on axis response, but for off axis response vane control case has higher roll and yaw attitude, but they are very small and can be neglected. Also it must be noted that aircraft is trimmed without the effect of gust and it is assumed gust suddenly starts with the start of the simulation, which is the reason of the transient at the beginning of the simulation.

A more challenging case with gust mean velocity of 20 ft/s from 45° diagonal direction is analyzed with the vane control on for the generic ducted fan simulation. The longitudinal and lateral responses are shown in Figures 5-3 and 5-4. It can be observed that controller successfully sustains the initial condition of the aircraft. Same as the previous case, gust is assumed as starting suddenly that is the reason of the transient observed at the beginning of the simulation.

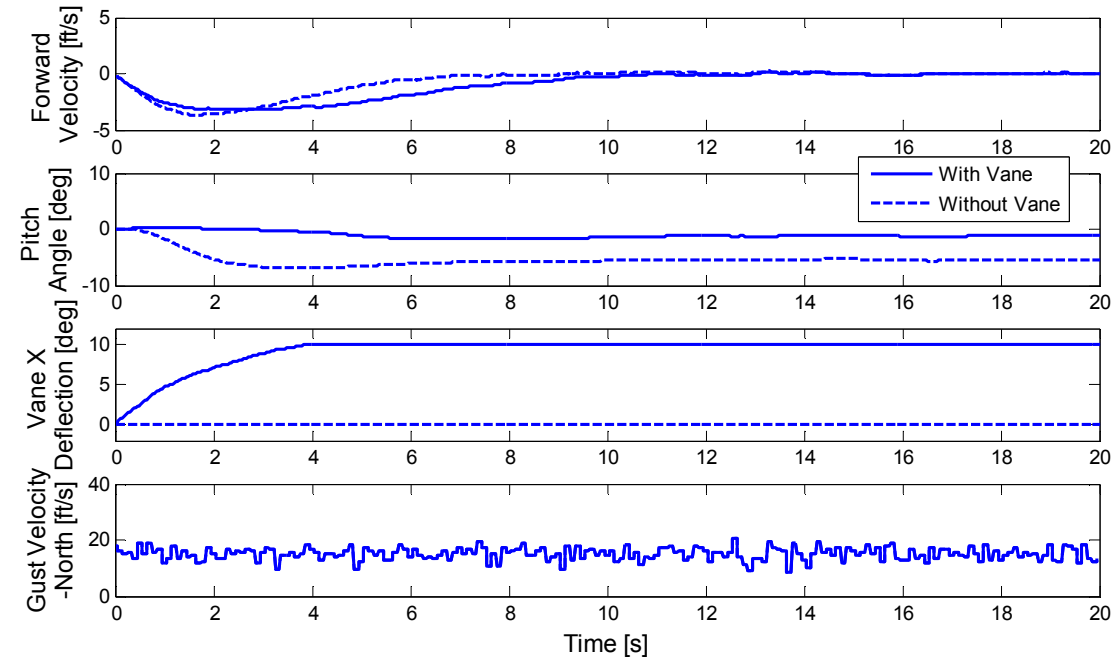

Figure 5-1 On Axis Response for Longitudinal Gust of 15 ft/s Mean Velocity

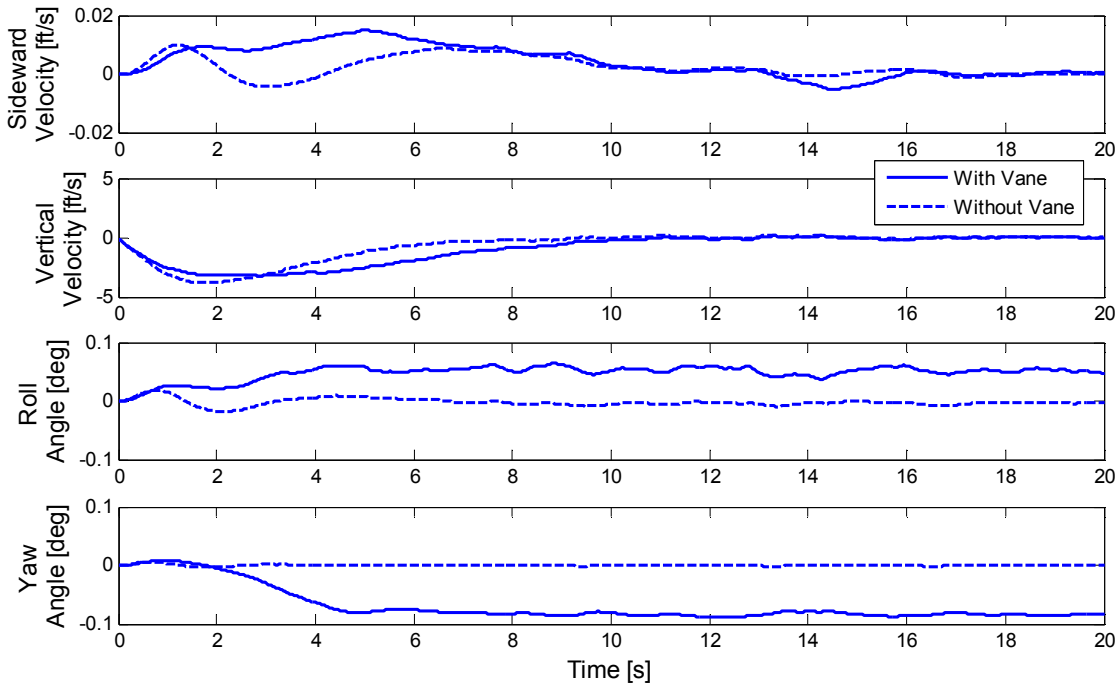

Figure 5-2 Off Axis Response for Longitudinal Gust of 15 ft/s Mean Velocity

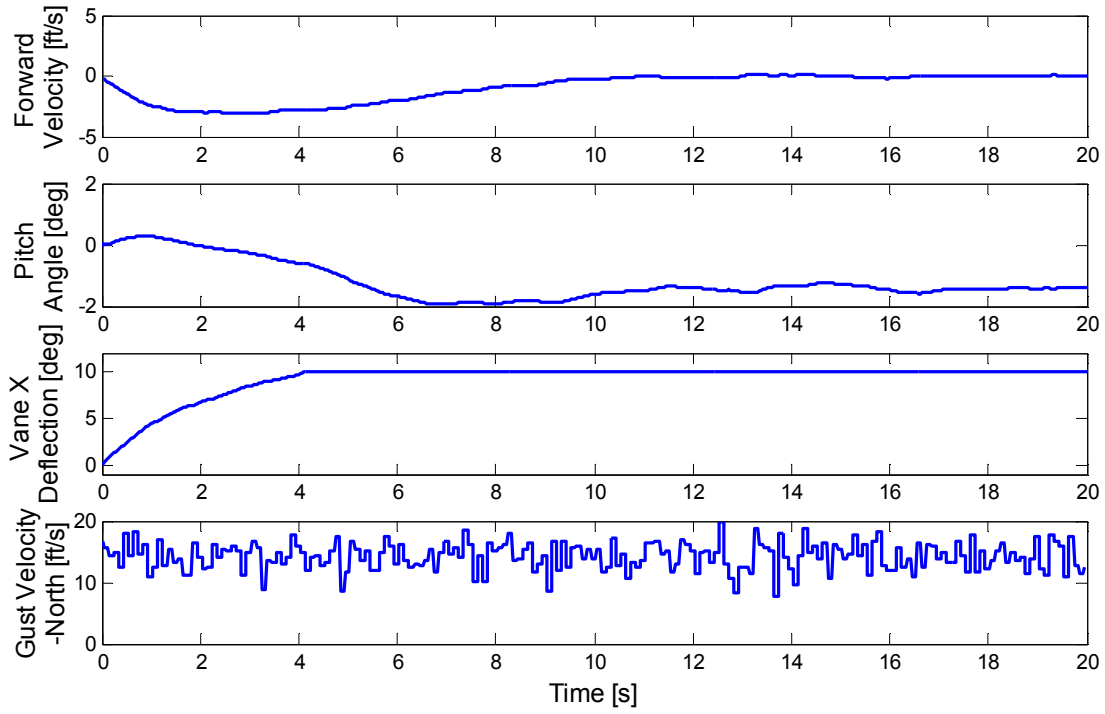

Figure 5-3 Diagonal Gust Response of Longitudinal Dynamics for 20 ft/s Gust Mean Velocity

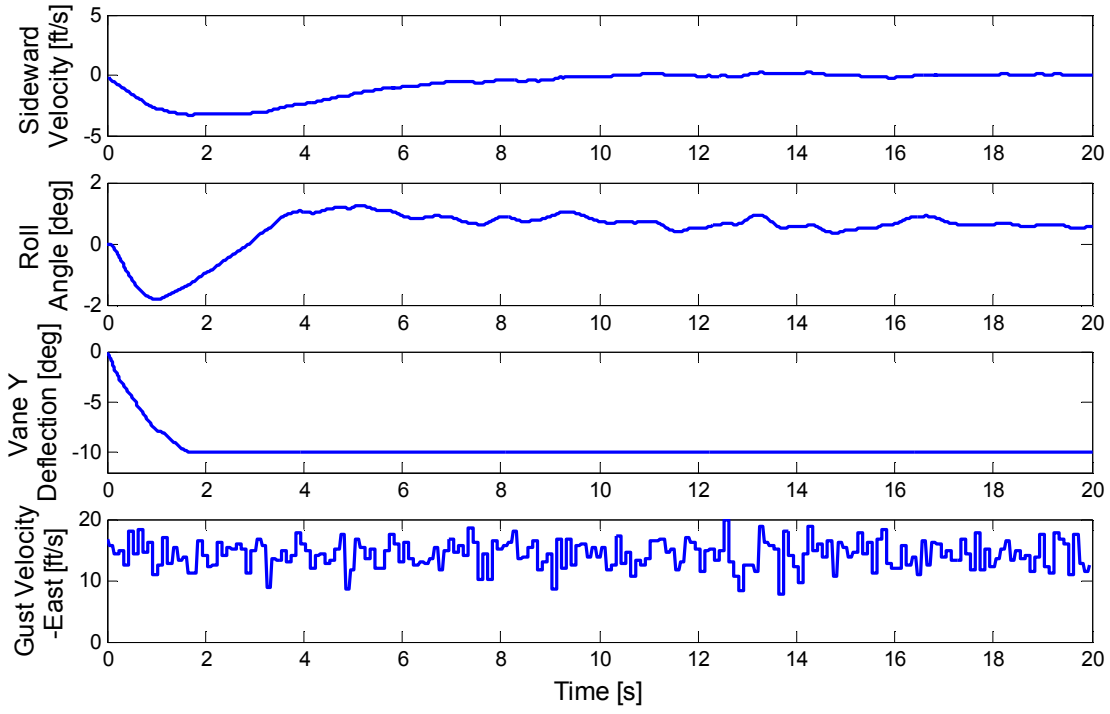

Figure 5-4 Diagonal Gust Response of Lateral Dynamics for 20 ft/s Gust Mean Velocity

5.2 Hover Turn

360 degrees of yaw is performed at hover both with and without vane control. Figure 5.5 shows the yaw and vane responses and Figure 5.6 shows the velocities and attitudes of the aircraft for the hover turn maneuver. Both with and without vane control turn, maneuver completed almost at the same time. Yaw motion is controlled by the pedal command only, which causes an asymmetric deflection of vanes. Different than the symmetric deflection; asymmetric deflections create yaw moment and no net propulsive force, since this maneuver does not involve any translational motion, longitudinal and lateral deflection of vanes are very small. Therefore, both with and without vane control, aircraft responded similarly.

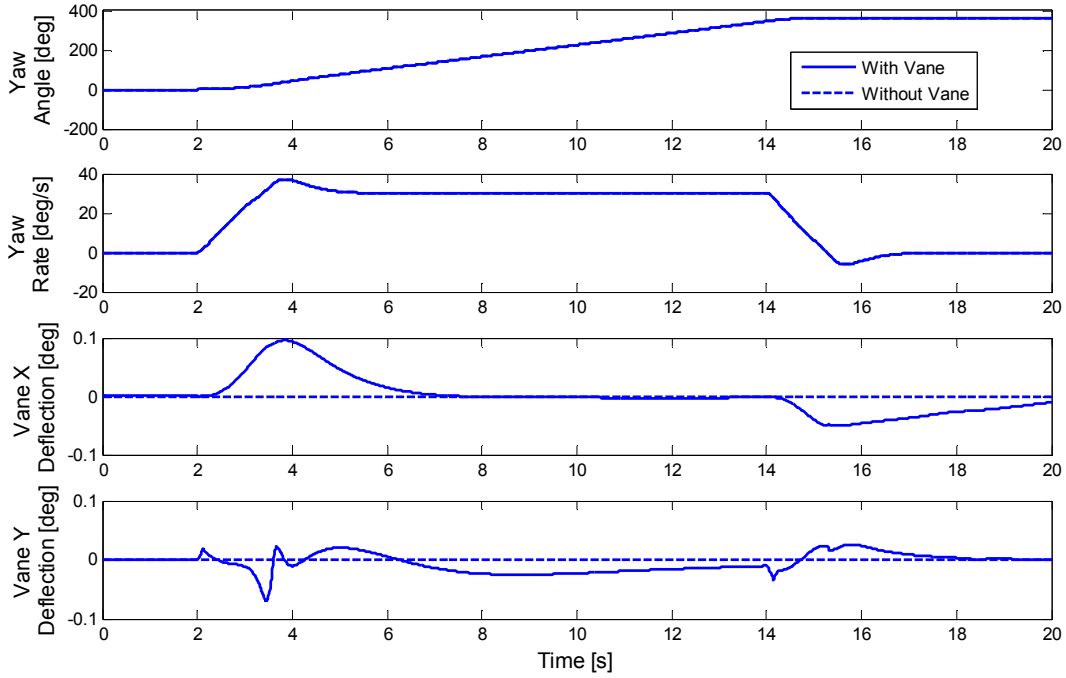

Figure 5-5 Yaw and Vane Responses for Hover Turn Maneuver

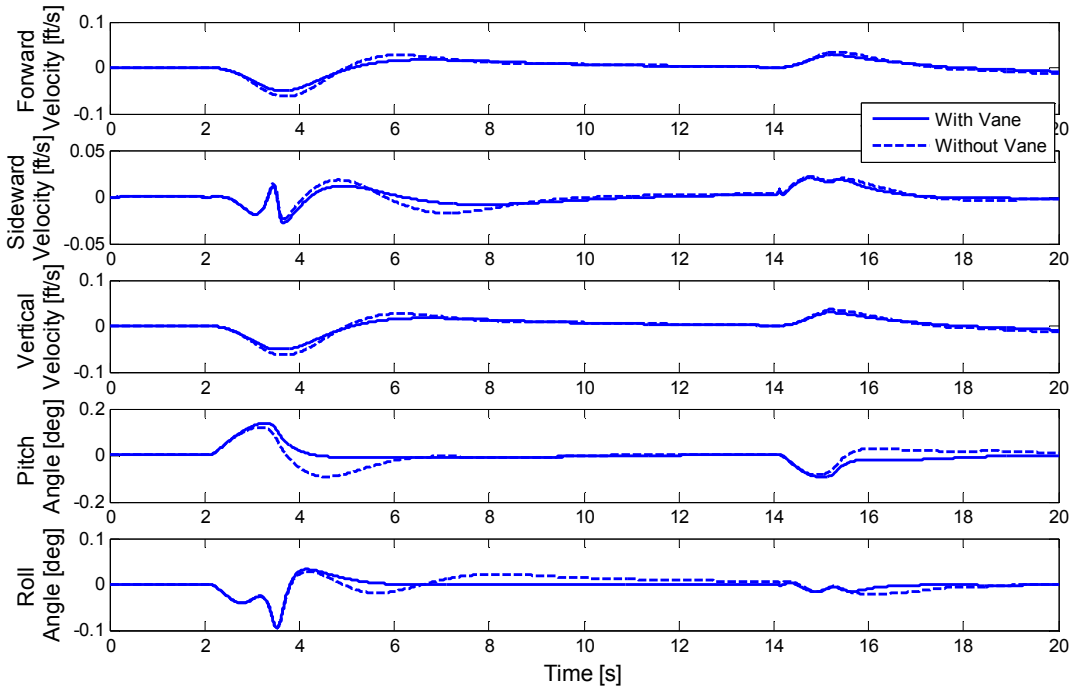

Figure 5-6 Velocities and Attitudes for Hover Turn Maneuver

5.3 Square Flight Path Maneuver

The generic ducted fan simulation model seemed to be work well for separate flight conditions, like forward flight and sideward flight. A more complex maneuvering flight condition was analyzed for a flight around a square as illustrated in Figure 5-7, where the aircraft will hover for 5 seconds at each corner.

The heading of the aircraft did not change during this motion. Figures 5-8 and 5-9 show how vane controls are used in this maneuvering flight. It is observed that pitch and roll attitudes are less than 6°.

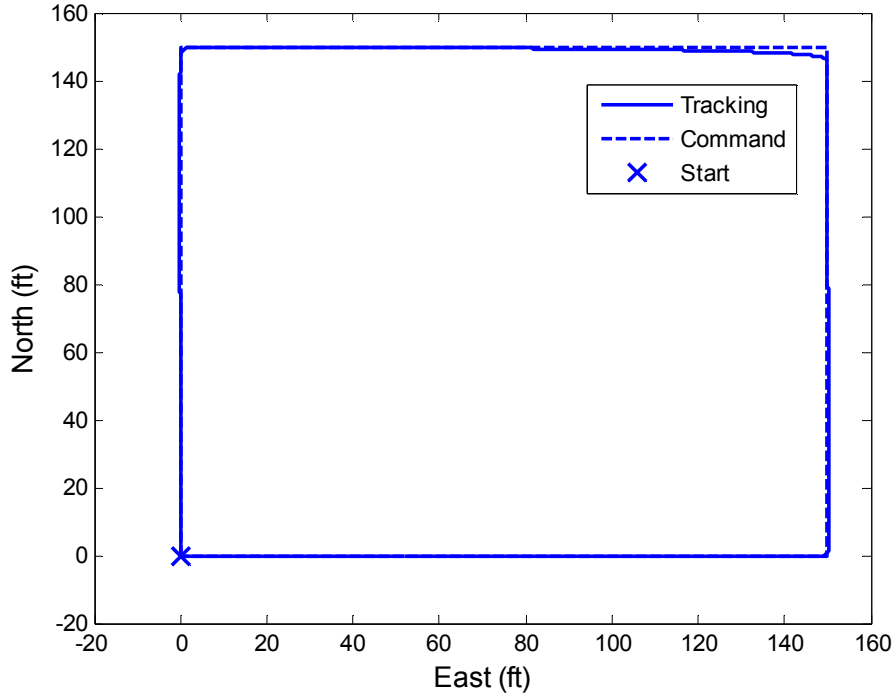

Figure 5-7 Motion around a Square

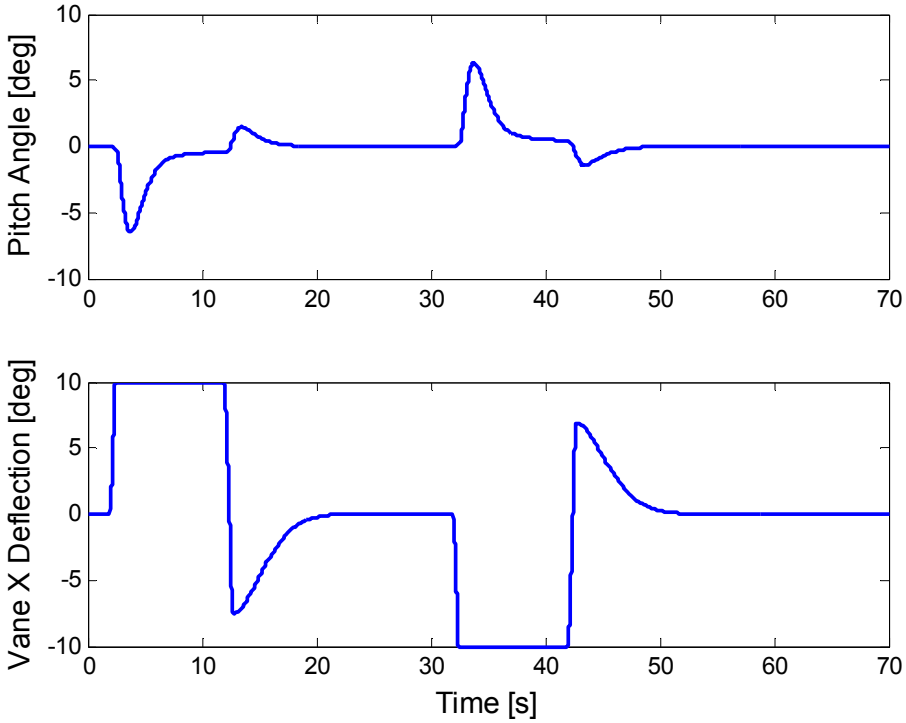

Figure 5-8 Pitch Attitudes of the Aircraft for the Motion around a Square

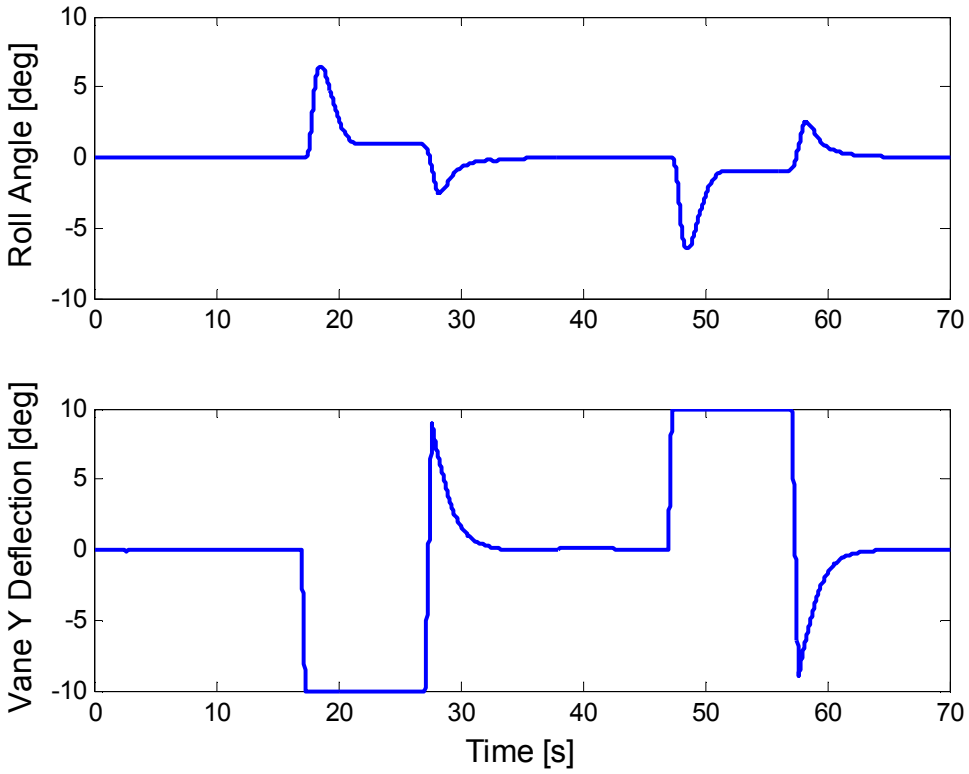

Figure 5-9 Roll Attitudes of the Aircraft for the Motion around a Square

5.4 Circular Flight Path Maneuver

The controller was also tested for a circular flight path starting from hover and ending at hover. For this condition 2 different flight speeds and 2 different yaw rates are used. 10 ft/s and 15 ft/s forward speed maintained in each circling motion and 10 deg/s and 20 deg/s yaw rates are maintained. Figure 5-10 shows the circling paths of the vehicle for different forward speeds and different yaw rates. Attitudes and vane deflections are shown in Figures 5-11 and 5-12, where Figure 5-11 is for a yaw rate of 10 deg/s and Figure 5-12 is for a yaw rate of 20 deg/s.

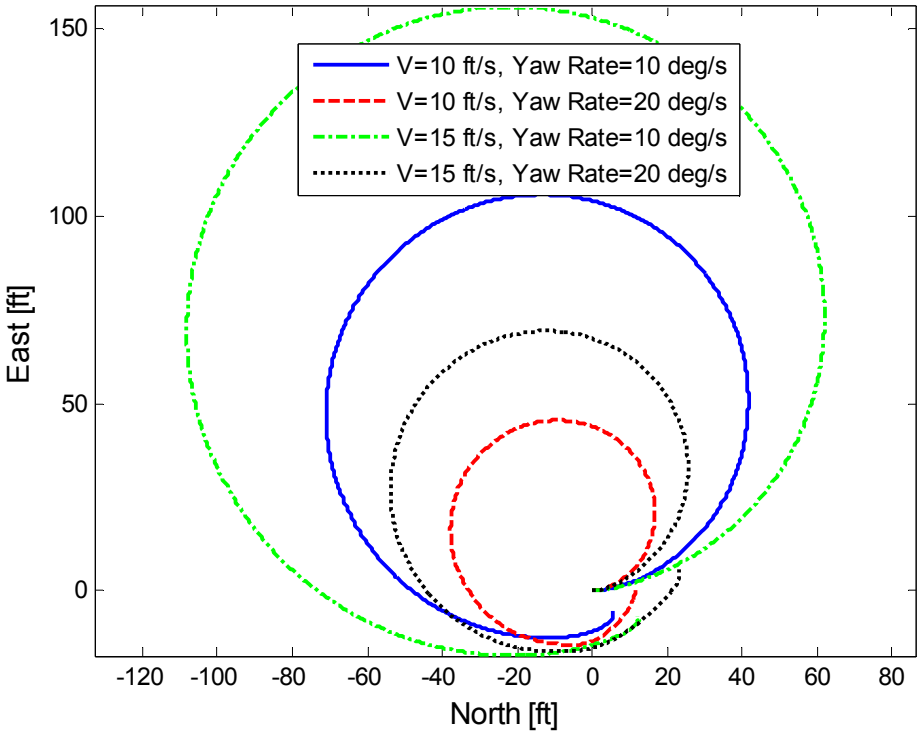

Figure 5-10 Circular Paths

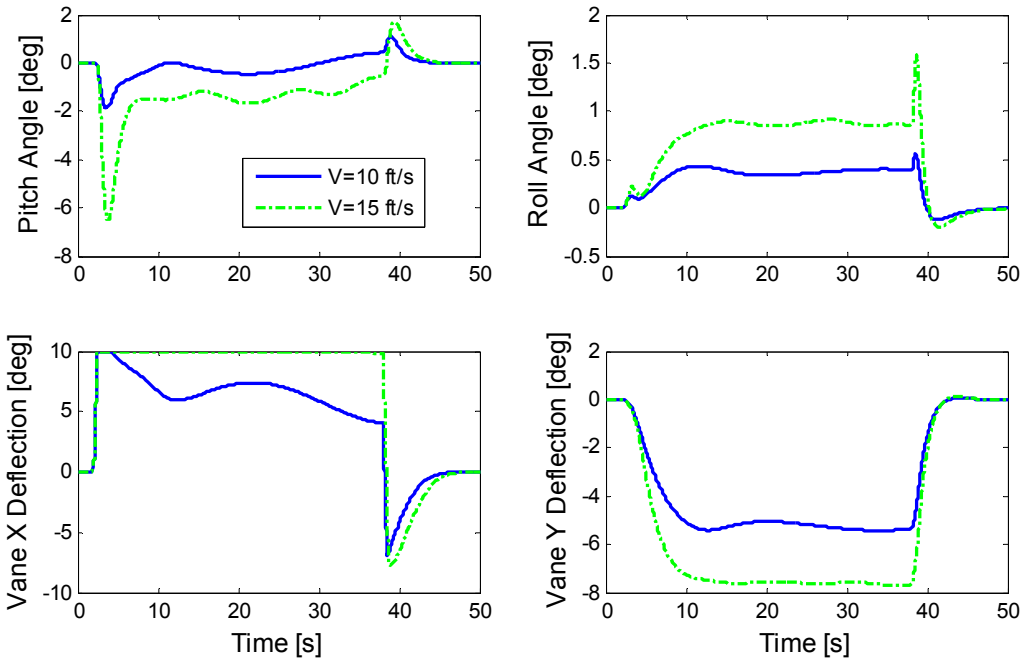

Figure 5-11 Attitudes of the Aircraft for Circling Motion with a Yaw Rate of 10 deg/s

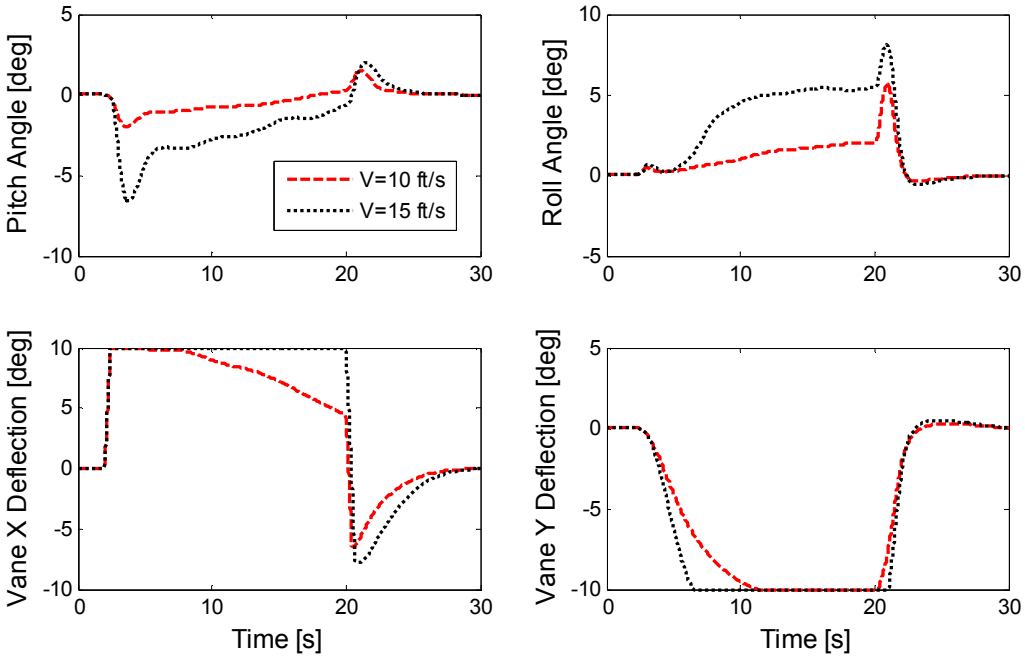

Figure 5-12 Attitudes of the Aircraft for Circling Motion with a Yaw Rate of 20 deg/s

5.5 Circular Flight Path Maneuver with Gusts

The controller was also tested for a circular flight path in gust condition while using vane control. Under a constant 15 ft/s north gust, aircraft followed a circular flight path with 10 ft/s forward speed and 20 deg/s yaw rate. Figure 5-13 shows the flight path of the aircraft and the heading of the aircraft throughout the maneuver, where a circular flight path is desired, however the gust shifted the path. Therefore aircraft could not track the circular path as it did in previous section, under the influence of 15 ft/s gust, aircraft moved 70 feet away from the start position, whereas without gust it was only 10 feet. Figures 5-14 shows how the all three directional velocities varied and the gust velocity. Also in Figure 5-14, how the forward velocity command is tracked under the influence of gust can be seen. In Figure 5-15 the yaw rate command tracking is shown. In Figure 5-16, pitch and roll attitudes can be seen together with longitudinal and lateral vane deflections.

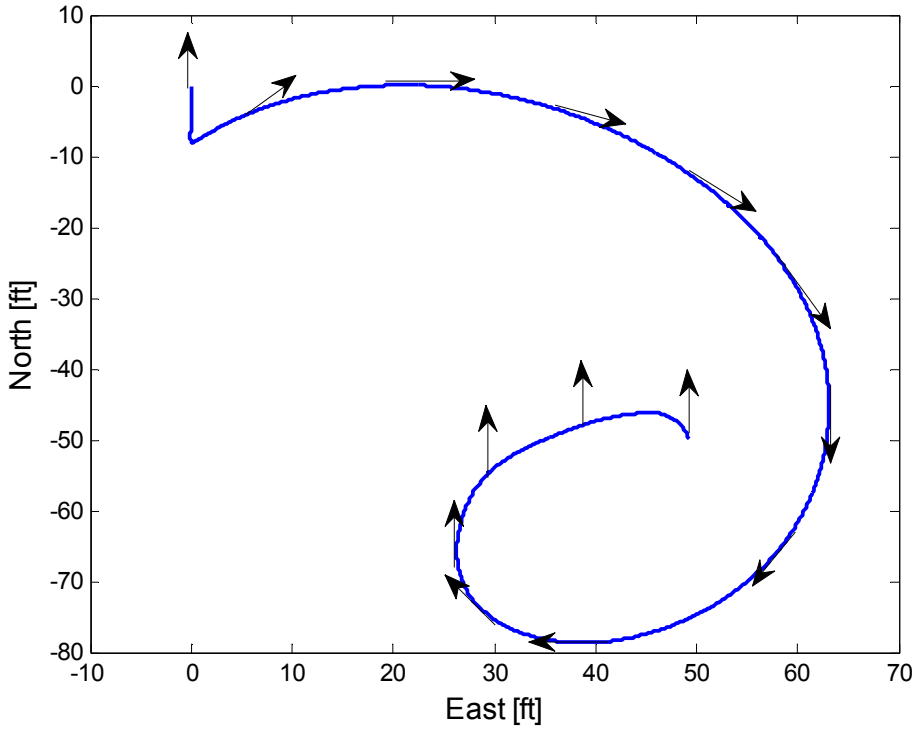

Figure 5-13 Circling Maneuver with Gust

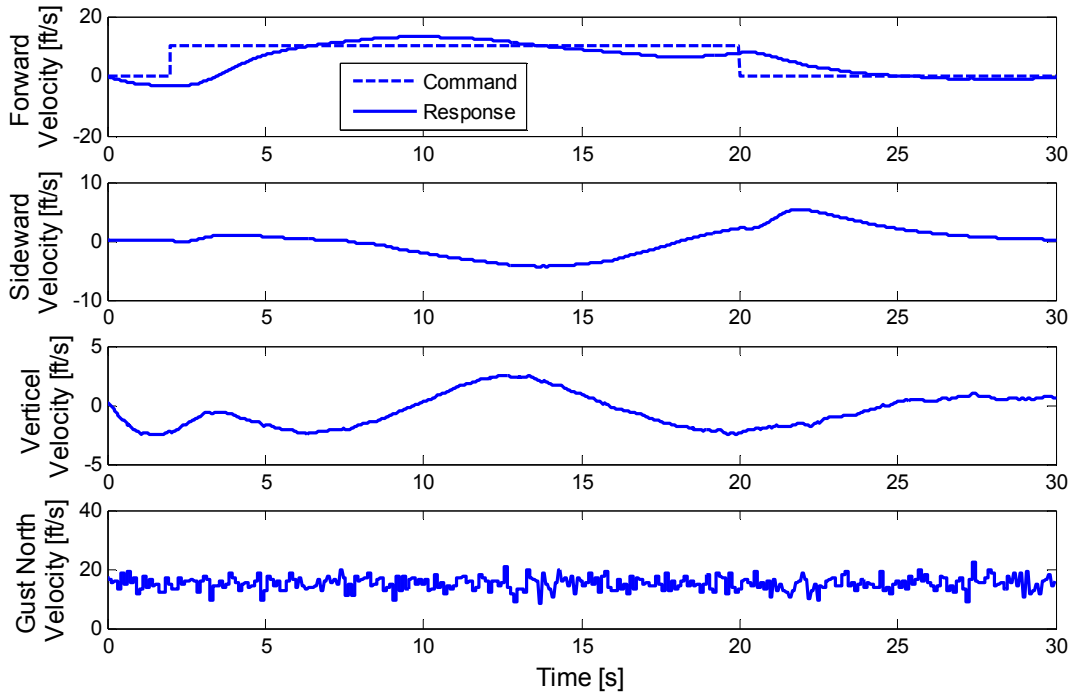

Figure 5-14 Velocities during Circling Maneuver with Gust

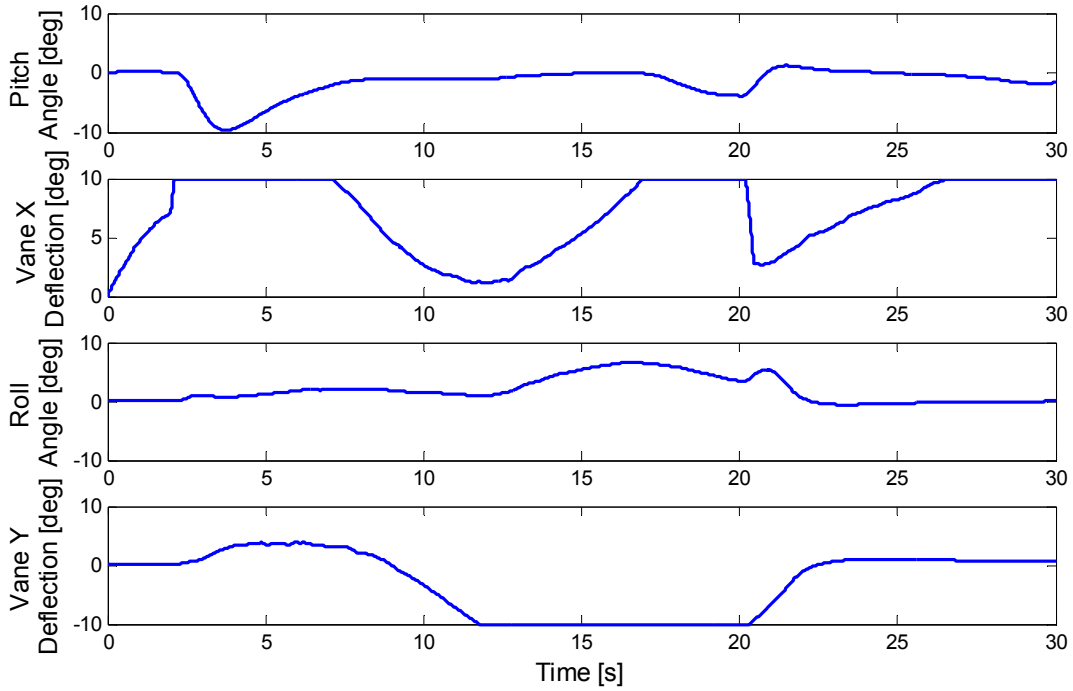

Figure 5-15 Attitudes and Vane Deflections during Circling Maneuver with Gust

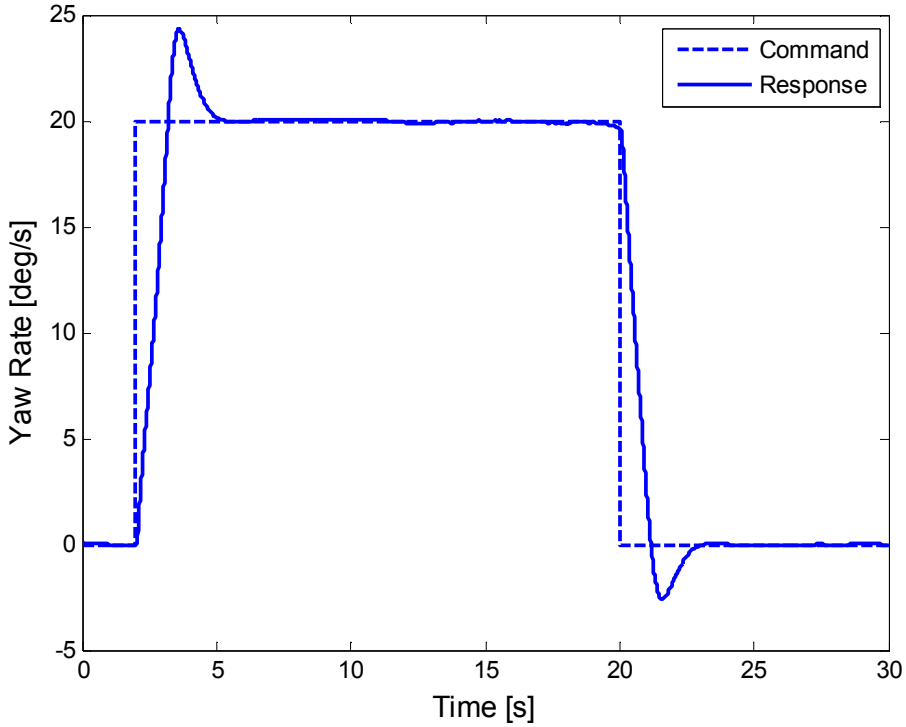

Figure 5-16 Yaw Rate Command Tracking during Circling Maneuver with Gust

Chapter 6

Conclusions

The goal of this study was to develop a controller for a tandem ducted fan aircraft with redundant control allocation and observe the response of the aircraft under different flight conditions. A control allocation method was developed to minimize attitude changes. The design method was developed and implemented using SIMULINK®.

In forward and sideward flight, the control allocation for redundant control effectors was found to be useful for reduction of pitch and roll attitude during translational motion. For low-speed translational motions the vehicle maneuvered with no attitude change, where all propulsive forces are supplied from vanes. Except for an initial attitude change in forward flight, very low pitch attitudes, that are lower then 3 degrees, are obtained for velocities up to 20 ft/s. As a result, vane control is found to be useful for reducing pitch attitude during translational motion.

The gust rejection characteristics of the controller were investigated and it was observed that use of vane controls is sufficient for gust rejection. 15 ft/s headwind gust was tolerated using only vane deflections. More complex gust condition, where sideward gust is also included, was tolerated with the controller developed.

The controller was also tested for four more complex maneuvering cases. First case was the hover turn maneuver, where 360° turn was performed in hover. During the yaw motion, ducted fan vehicle preserved its location, altitude and roll and pitch attitudes. In another case the aircraft is tracking a square motion without changing the heading. Another case is a circular trajectory while keeping the heading aligned with the flight path. At low speeds, these maneuvers achieved with attitude changes less than 5 degrees. For higher speeds and higher yaw rates roll attitude reached up to 6 degrees. Finally a circular maneuvering case under the influence of gust in tested and controller performed the maneuver, but lost its position and altitude. It must be noted that the controller is only

trying to regulate velocity; it is not trying to track a path. However, pitch and roll attitudes are kept below 10 degrees throughout the maneuver.

The results proved that the forward and sideward translation of this type of vehicle can be controlled independently of the roll and pitch attitude with the control authority limits of the vanes. A larger range of flight conditions could be achieved with zero attitude variation if the number or size of the exit vanes were increased. This capability might be useful for an unmanned ducted fan vehicle carrying a sensor system, since the sensor might more able to track a target if the attitude excursion of the platform were minimized. The capability might also be useful for a vehicle operating in an urban environment close to structures. Large attitude changes might not be desirable for safety reasons. A ducted fan aircraft might even be able to push against buildings to insert or retrieve people in a military or rescue mission. The control allocation methodology could be modified to optimize the vehicle attitude for other applications as well.

Appendix A

Simulink Diagram of the Controller

Figures A-1 through A-18 shows the Simulink block diagrams of the generic ducted fan simulation model used in this study.

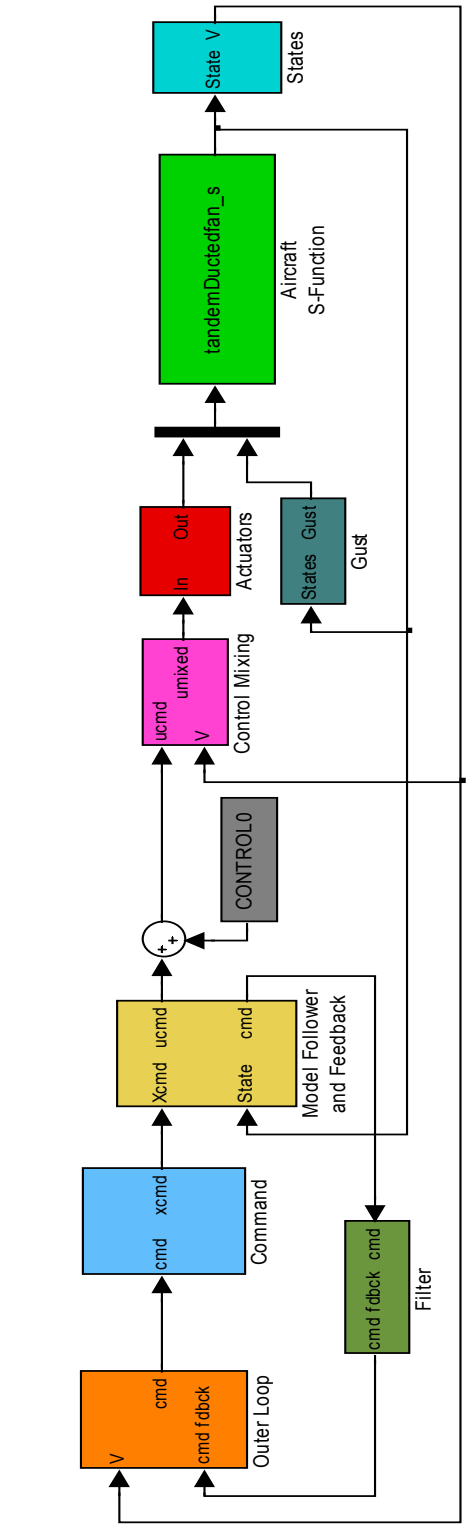

Figure A-1 The Simulink Block Diagram of the Simulation

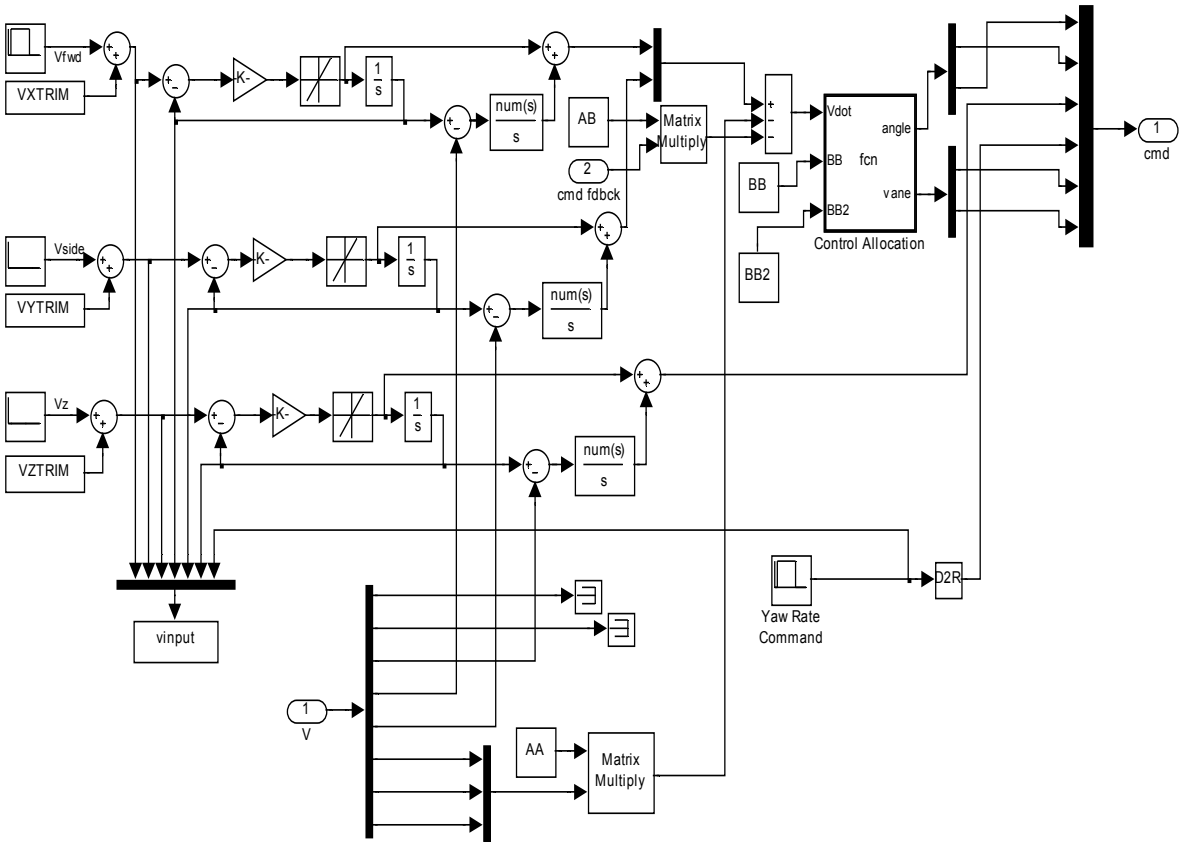

Figure A-2 Outer Loop Block

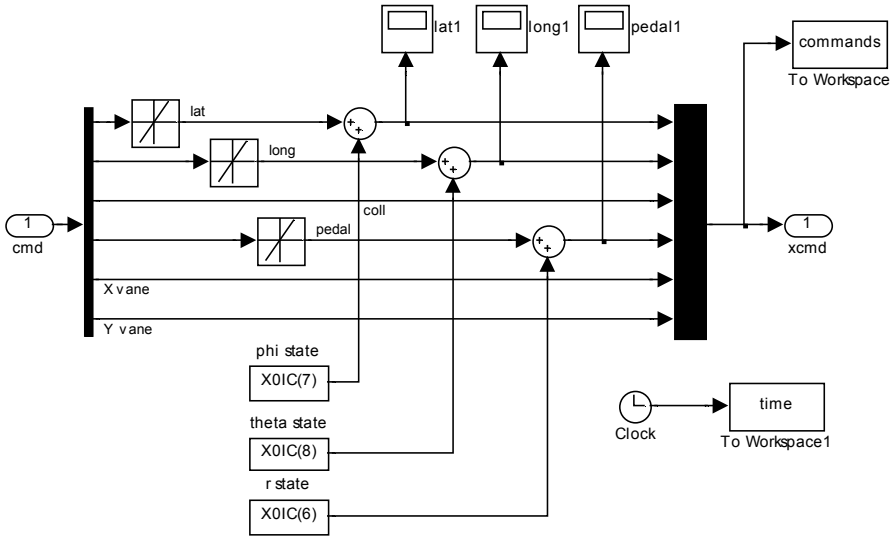

Figure A-3 Command Block

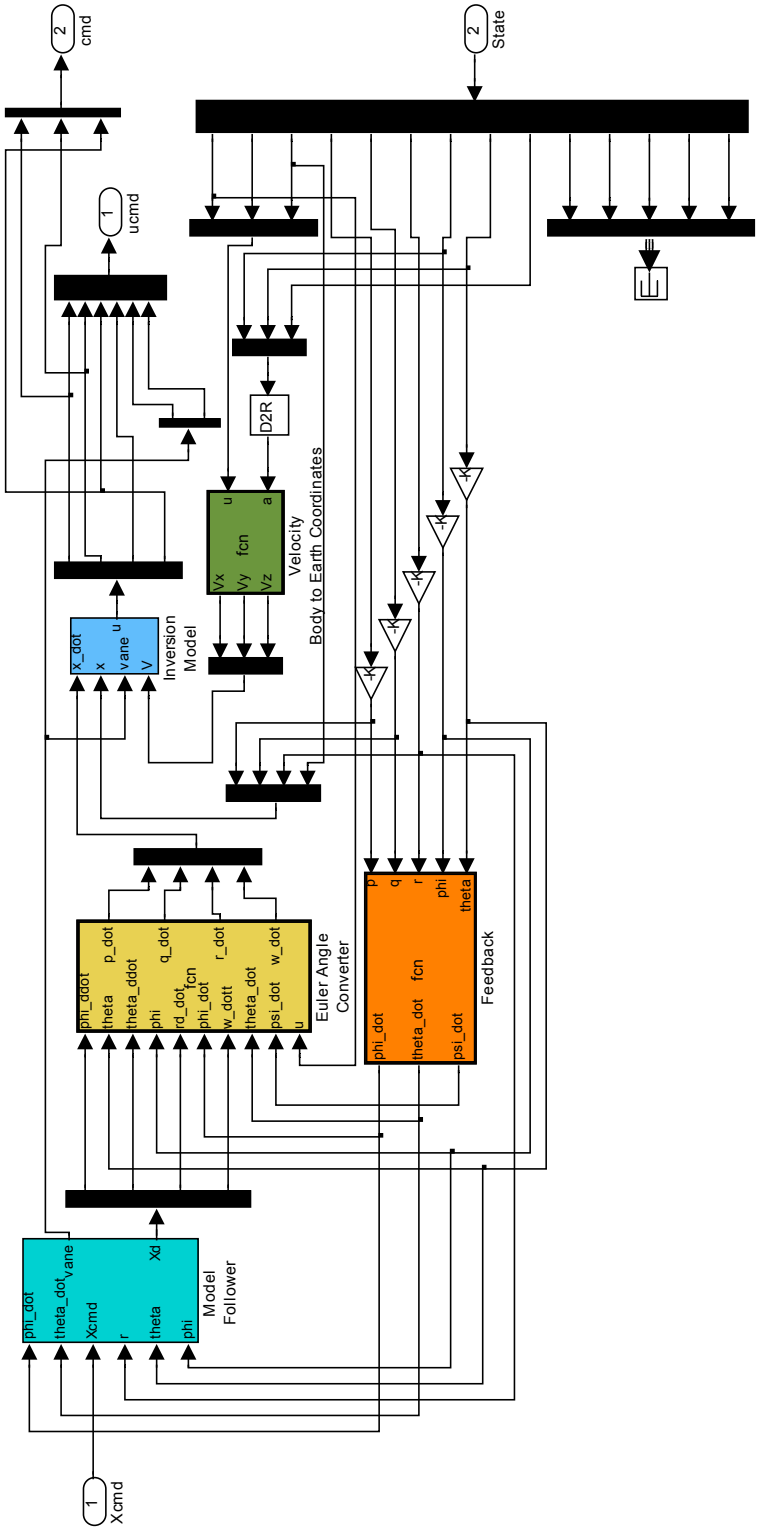

Figure A-4 Model Follower and Feedback

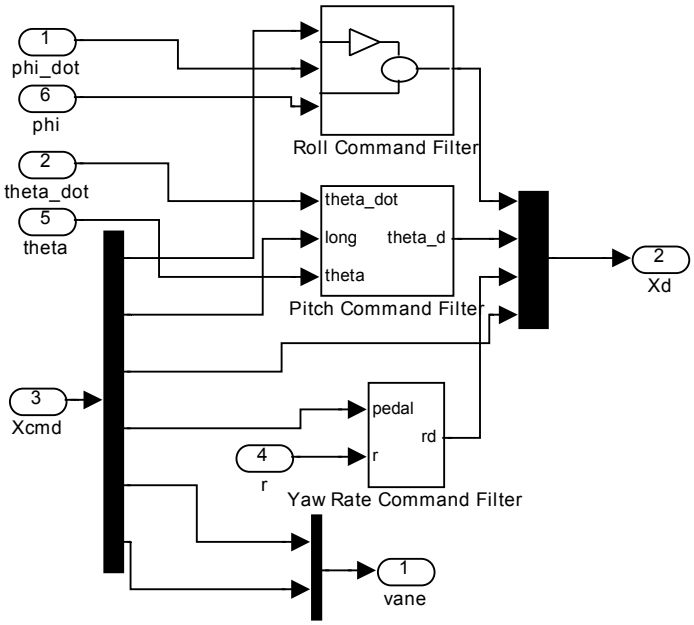

Figure A-5 Model Follower Block in Model Follower and Feedback Block

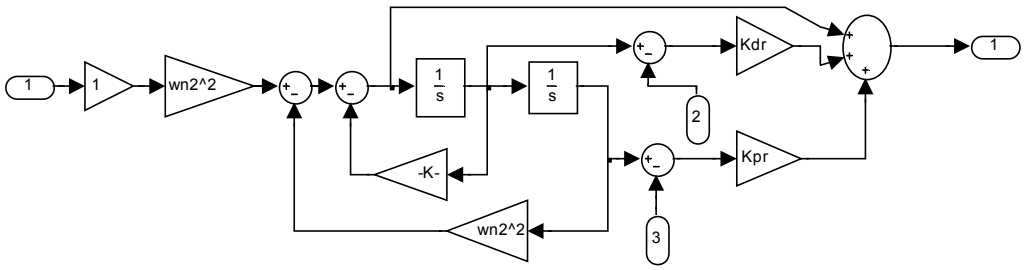

Figure A-6 Roll Command Filter Block

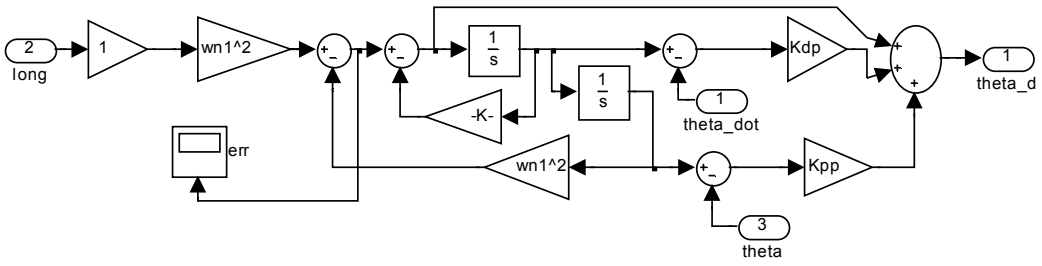

Figure A-7 Pitch Command Filter Block

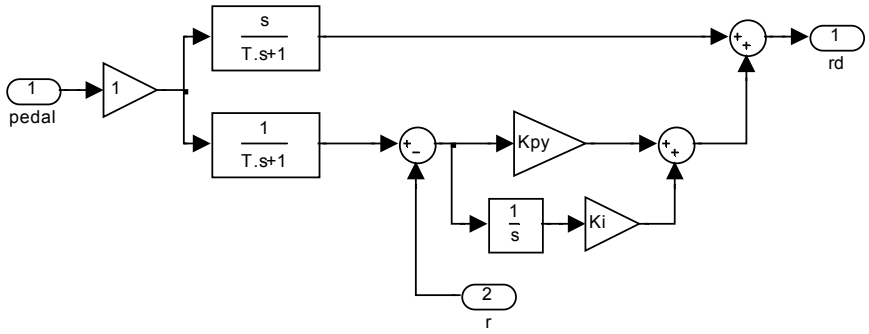

Figure A-8 Yaw Rate Command Filter Block

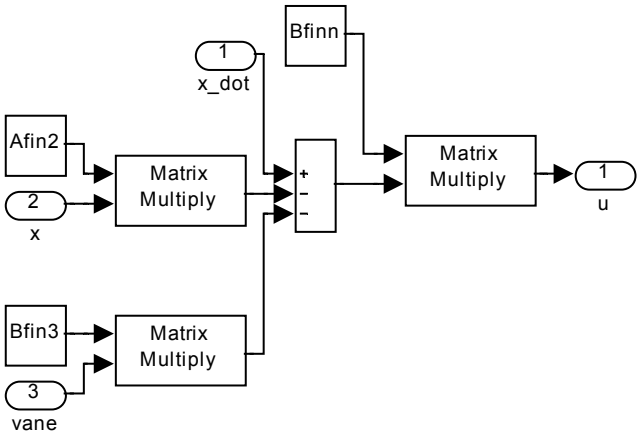

Figure A-9 Inversion Model Block in Model Follower and Feedback Block

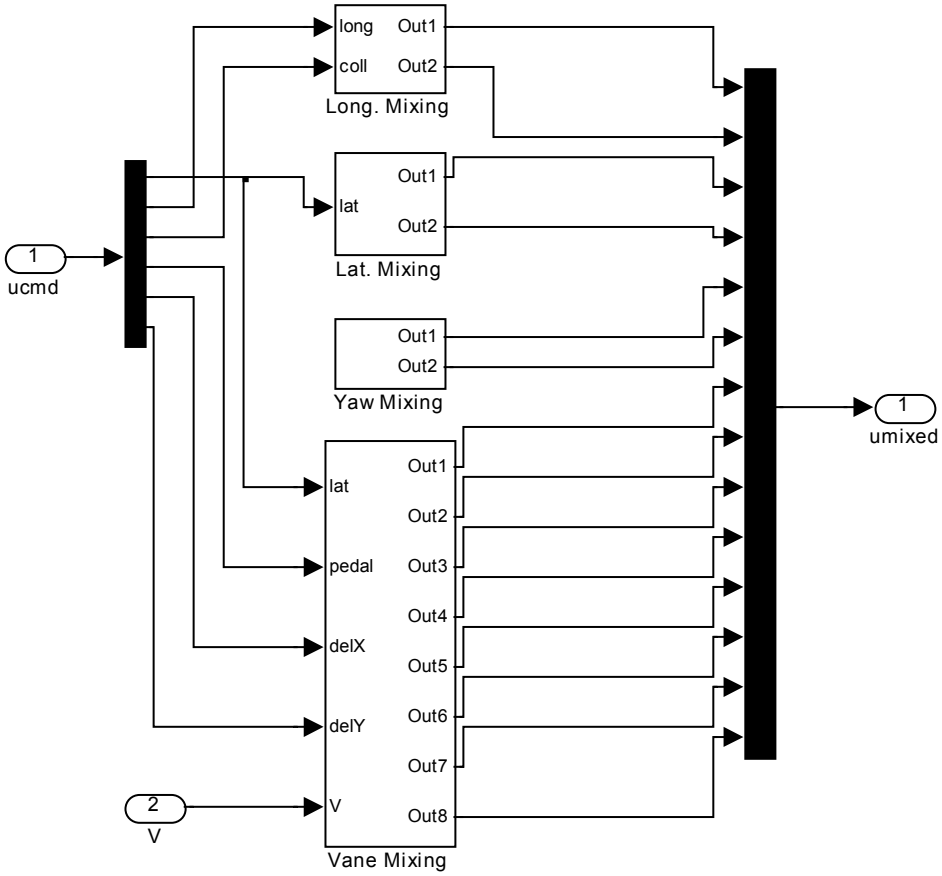

Figure A-10 Control Mixing Block

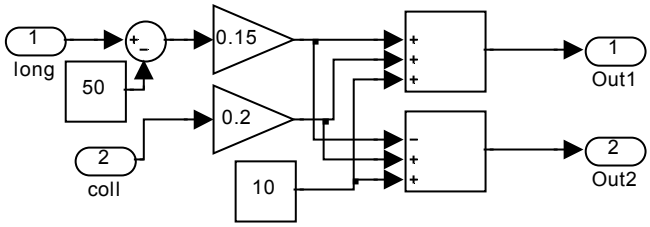

Figure A-11 Long. Mixing Block in Control Mixing Block

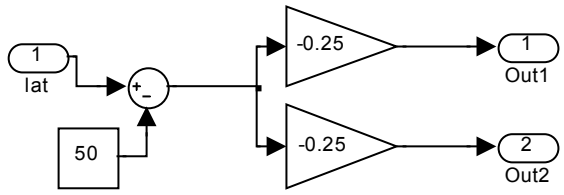

Figure A-12 Lat. Mixing Block in Control Mixing Block

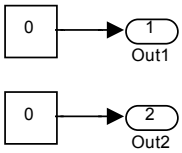

Figure A-13 Yaw Mixing Block in Control Mixing Block

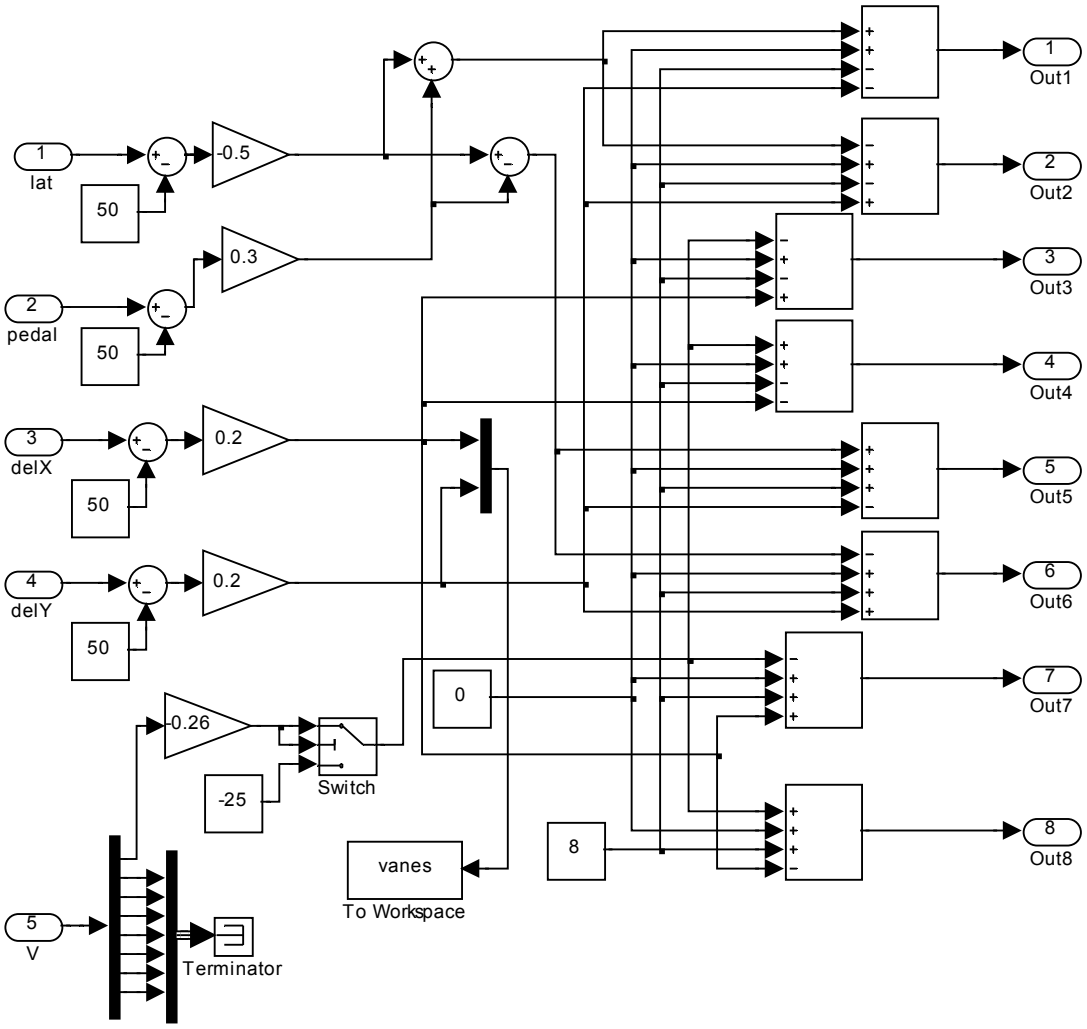

Figure A-14 Vane Mixing Block in Control Mixing Block

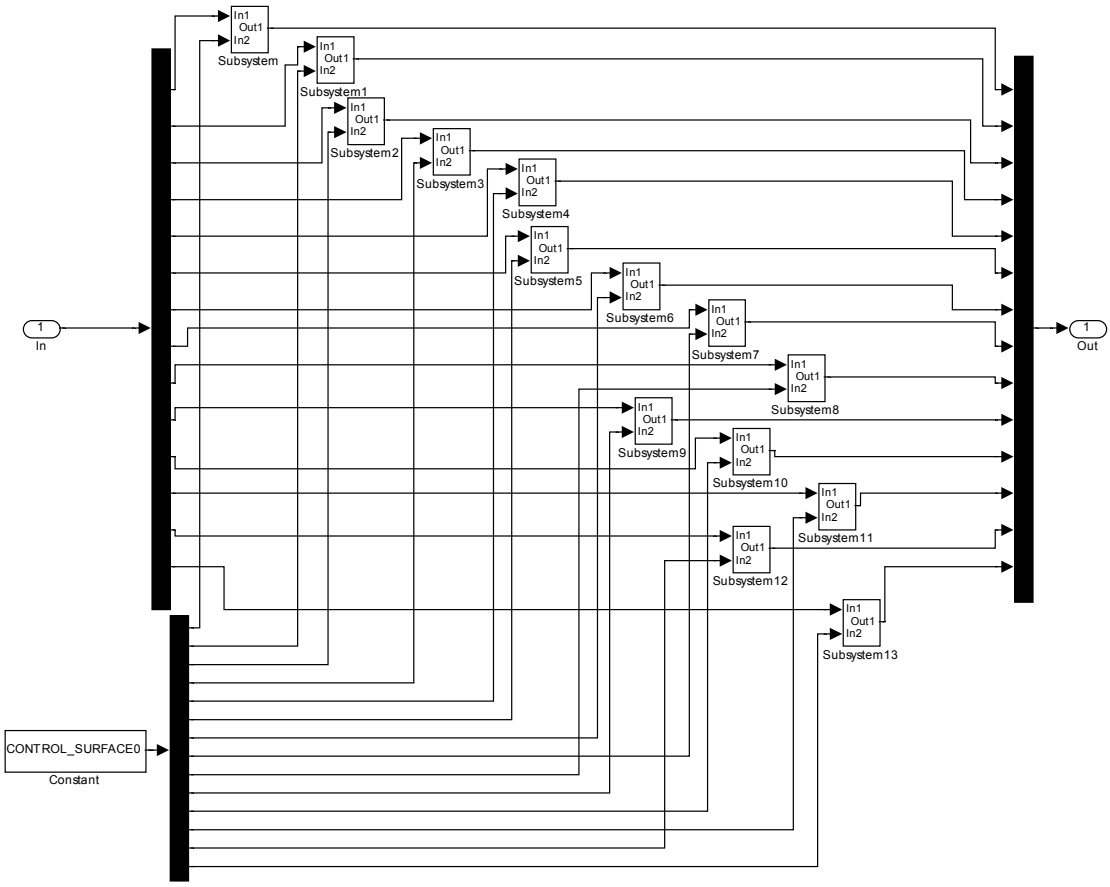

Figure A-15 Actuators Block

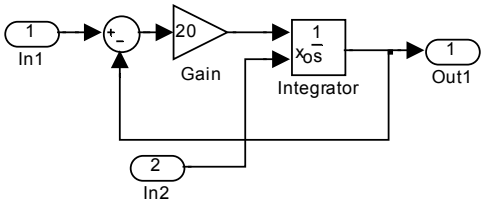

Figure A-16 All Subsystem Blocks in the Actuators Block

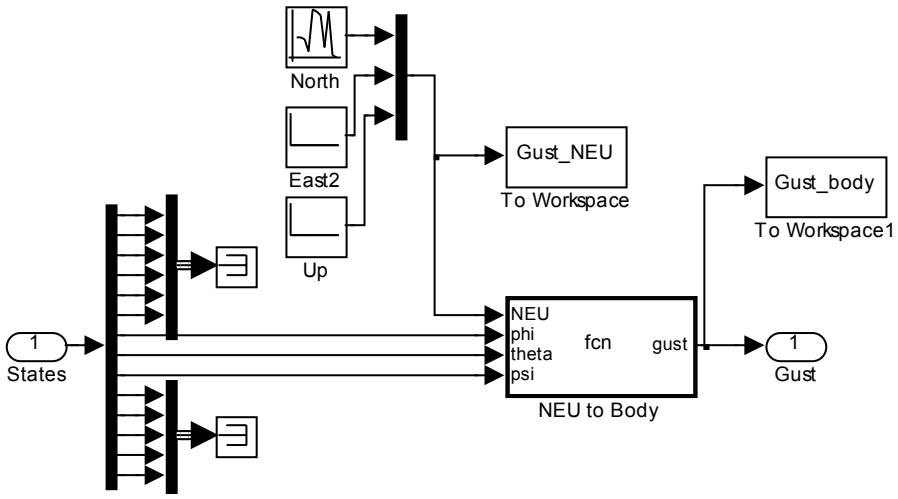

Figure A-17 Gust Block

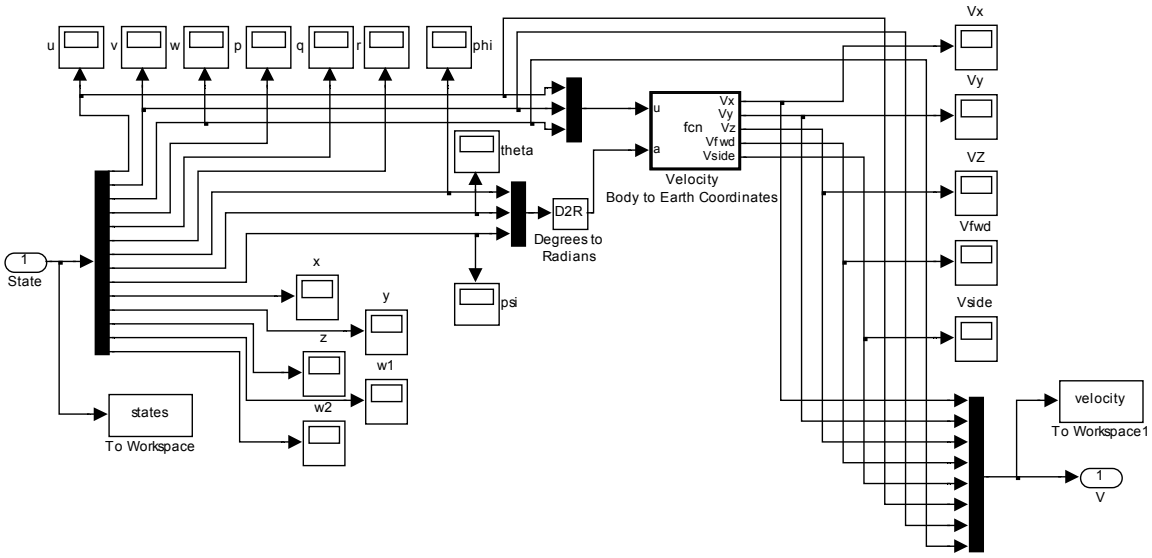

Figure A-18 States Block

Appendix B

B.1 Matlab Code

```
global  INIT VISC RHO RADIUS CHORD NB RCO BTL HROT XROT YROT ISHAFT
TWIST IDIRROT RETAB ALTAB CLTAB CDTAB D2R R2D DELT PSIV SV VANEFF XVANE
YVANE ZVANE NVANES CHORDV CLVTAB CDVTAB ALVTAB REVTAB DELDRAG DXCG DYCG
DZCG OMEGAREF G MASS IX IY IZ IXZ NROTORS NSTATES NFSTATES NRSTATES
NCTRLS TRIMVARS XSCALE DELXLIN DELCLIN VXTRIM VYTRIM VZTRIM PSIDTRIM
IUNITS OUTDATA X0IC CONTROL0 FM FLAGG vi viqss1 viqss2 Tdata Vzdata
Vindata nrows ncols npages;

IUNITS=1;

% -------------- Load Model Parameters --------------%
INIT=1;

%-- Time step --%
DELT=0.01;

%-- Constants --%
D2R=pi/180.;
R2D=180./pi;

%---- Atmospheric propeties ----%
RHO=0.002377;
VISC=3.74e-7;
IATMOS=0;  % Set IATMOS =1 to use standard atmo model,
           % else the value of RHO above will be used regardless
           % of altitude or temperature

%---- Mass properties ----%
%Accel of gravity (ft/s^2)
G=32.174;
%Weight (lbs)
weight = 4000.;
%Aircraft Mass (slugs)
MASS=weight/G;
%Moments of Inertia (slug-ft^2)
IX=1208;
IZ=6083;
IY=5208;
IXZ=0.0;

%---- Rotor properties ----%
NROTORS=2;
RADIUS=4; % (ft)
OMEGAREF=188.5; % (rad/sec)
% DCHORD=4; % duct chord length (ft)          (not used)
CHORD=0.6; %Blade chord (ft)
```

```
RCO=0.5; % Root cutout (ft)
NB=6; % number of blades
BTL=0.98; % nominal tip loss factor
XROT=[6.; -6.]; %Rotor center position rel. to fuse center (ft)
HROT=[0.; 0.];
YROT=[0.; 0.];
ISHAFT=[0.; 0.]; % shaft incidence
TWIST=-10.; % blade twist (deg)
IDIRROT=[1; -1]; % direction of rotation,1 CCW, -1 CW, 0 Counter
Rotating
%This script definmes NACA0012 airfoil data
naca0012;
load('inflow_table_7.mat')
load('GS_af2_1.mat')
load('GS_bf3_1.mat')
load('GS_bfn_1.mat')
[nrows,ncols,npages] = size(vi);
viqss1=[0 0];
viqss2=[0 0];
%---- Vane propeties ----%
NVANES=4;
VANEFF=1; % vane efficiency factor
%initialize vane properties
PSIV=zeros(NVANES,NROTORS); % Azimuthal Orientation of vanes around the
rotor disk
XVANE=zeros(NVANES,NROTORS); % X-location of the center of pressure on
the vane relative to the center of the rotor
YVANE=zeros(NVANES,NROTORS); % Y-location " "
ZVANE=zeros(NVANES,NROTORS); % Z-location " "
SV=9*ones(NVANES,NROTORS); % area of each vane (ft^2)
CHORDV=2.25*ones(NVANES,NROTORS); %vane chord (ft)

PSIV(1,1)=0.;
PSIV(2,1)=180.;
PSIV(3,1)=90.;
PSIV(4,1)=270.;
PSIV(1,2)=0.;
PSIV(2,2)=180.;
PSIV(3,2)=90.;
PSIV(4,2)=270.;

XVANE(1,1)=-0.5*RADIUS;
XVANE(2,1)=0.5*RADIUS;
XVANE(3,1)=0;
XVANE(4,1)=0;
YVANE(1,1)=0;
YVANE(2,1)=0;
YVANE(3,1)=0.5*RADIUS;
YVANE(4,1)=-0.5*RADIUS;
ZVANE(1,1)=3;
ZVANE(2,1)=3;
ZVANE(3,1)=3;
ZVANE(4,1)=3;
```

```
XVANE(1,2)=-0.5*RADIUS;
XVANE(2,2)=0.5*RADIUS;
XVANE(3,2)=0;
XVANE(4,2)=0;
YVANE(1,2)=0;
YVANE(2,2)=0;
YVANE(3,2)=0.5*RADIUS;
YVANE(4,2)=-0.5*RADIUS;
ZVANE(1,2)=3;
ZVANE(2,2)=3;
ZVANE(3,2)=3;
ZVANE(4,2)=3;

%Just use NACA0012 properties for vanes as well
CLVTAB=CLTAB;
CDVTAB=CDTAB;
ALVTAB=ALTAB;
REVTAB=RETAB;
%---- Rotor airfoil drag increment ----%
DELDRAG=0.0;

DXCG=0.;
DYCG=0.;
DZCG=0.;

 VXTRIM = 0.01;
 VYTRIM = 0.01;
 VZTRIM = 0.0;
 PSIDTRIM = 0.0;

%Number of state variables for fuselage degrees of freedom, rotor,
propulsion system, and controls.
NFSTATES=12;
NRSTATES=2;
NSTATES=NFSTATES+NRSTATES;
%Number of controls
NCTRLS=6;

%INDEX OF STATE VARIABLES
%Fuselage States
IDXF=[1:NFSTATES];
%Rotor States
IDXR=[NFSTATES+1:NFSTATES+NRSTATES];

IFREEZE=0.;

%Initialize guess of states
xf=zeros(NFSTATES,1);

gust=[0; 0; 0];

u0=[50;50;50;50;50;50;gust];
```

```
%Initial guess for trim solution
xf(1)=VXTRIM;
xf(2)=VYTRIM;
xf(3)=VZTRIM;
xf(9)=0;
xr=[45.;45.];
x0=[xf;xr];

%Set up trim variables
%State scale factors - used to weight relative value of state variables
for trim and numerical integration
XSCALE=1./[1. 1. 1. 57.3*ones(1,6) 1. 1. 1. 1. 1.]';

%Perturbations for trim and linearization
DELCLIN=[1. 1. 1. 1. 1. 1.];
DELXLIN=0.1*XSCALE;

%Define Trim Variables
TRIMVARS=[1:8 IDXR NSTATES+1:NSTATES+4];

INIT=1;
%Run trim
current_aircraft='tandemDuctedfan';
[x0_new,u0_new,itrim]=trimmer(current_aircraft,x0,u0);
x0=x0_new;
u0=u0_new;

X0IC=x0;
CONTROL0=u0;
[CONTROL_SURFACE0]=sim2(CONTROL0);
CONTROL_SURFACE0=CONTROL_SURFACE0';
xf=x0(IDXF);
XF0IC=x0(IDXF);
xr=x0(IDXR);
XR0IC=x0(IDXR);
current_aircraft='tandemDuctedfan';
```

```
global x0 u0 Afin Bfin;

[A,B]=linearize(current_aircraft,x0,u0);

A11=A(1:12,1:12);
A12=A(1:12,13:14);
A21=A(13:14,1:12);
A22=A(13:14,13:14);
B1=B(1:12,1:6);
B2=B(13:14,1:6);

Afin=A11-A12*(A22^-1)*A21;
Bfin=B1-A12*(A22^-1)*B2;
Afin2=[Afin(4:6,4:6) Afin(4:6,3);Afin(3,4:6) Afin(3,3)];
Bfin2=[Bfin(4:6,[1 2 4]) Bfin(4:6,3);Bfin(3,4:6) Bfin(3,3)];
Bfin3=[Bfin(4:6,[5 6]);Bfin(3,[5 6])];
Cfin2=eye(3,3);
Dfin2=zeros(3,3);
Bfinn=Bfin2^-1;
Zw=A(3,3);
Zcol=B(3,3);
B22=B(4:6,5:6);
Xx=B(1,5);
Yy=B(2,6);

AA=A(1:2,1:3);
AB=B(1:2,1:3);
BB=B(1:2,5:6);
BB2=[-32.1 0; 0 32.1];

%pitch
wn1=3.;
zeta1=0.8;
Kpp=wn1^2;
Kdp=2*zeta1*wn1;

%roll
wn2=3.;
zeta2=0.9;
Kpr=wn2^2;
Kdr=2*zeta2*wn2;

%yaw
wn3=4;
zeta3=0.9;
T=0.4;
Ki=wn3^2;
Kpy=2*zeta3*wn3;

%Translational Controls
T2=2;
Kp2=0.9;
Ki2=0.25;
```

```
function [x_dot]=tandemDuctedfan(x,u3)

% Input: time, t
%        aircraft state vector, x
%        pilot control input, u
% Output: state derivative, x_dot

global IFREEZE NFSTATES NRSTATES NSTATES OUTDATA GEARTR QACC IROTOR
OMEGAREF X0IC FM FLAGG;

%Partition state vector
% Fuselage states
xf=x(1:NFSTATES);
%Rotor states
xr=x(NFSTATES+1:NFSTATES+NRSTATES);

xr_dot=zeros(NRSTATES,1);

omega=[OMEGAREF; OMEGAREF];

if numel(u3)==17
    gust=u3(15:17);
    u=u3(1:14);
else
    u=u3(1:6);
    gust=[0; 0; 0];
end

if numel(u)==6
    FLAGG=0;
end
%Control System Mixing
%[theta0,theta1s,theta1c,deltav]=ductfan_mixing2_vane_only(u);

if FLAGG==0
    [u2]=sim2(u);
    u=u2;
end

    theta0(1)=u(1);
    theta0(2)=u(2);
    theta1s(1)=u(3);
    theta1s(2)=u(4);
    theta1c(1)=u(5);
    theta1c(2)=u(6);
    deltav(1,1)=u(7);
    deltav(2,1)=u(8);
    deltav(3,1)=u(9);
    deltav(4,1)=u(10);
    deltav(1,2)=u(11);
    deltav(2,2)=u(12);
    deltav(3,2)=u(13);
    deltav(4,2)=u(14);
```

```
xf2=xf;
xf2(1:3)=xf2(1:3)+gust;

[F1,M1,Qr1,xr_dot(1)] =
ductfan_vec(xf2,xr,omega,theta0,theta1s,theta1c,deltav,1);

[F2,M2,Qr2,xr_dot(2)] =
ductfan_vec(xf2,xr,omega,theta0,theta1s,theta1c,deltav,2);

% Sum forces and Moments
FM=[F1+F2;M1+M2];
Qreq=Qr1+Qr2;

%Equations of Motion
xf_dot=eqnmot(xf,FM);
if (IFREEZE==1);
   xf_dot(1:8)=zeros(8,1);
end

%Total state derivative
x_dot=[xf_dot;xr_dot];

%OUTDATA=[x;u];

 return
```

```
function [F,M,Qr,xrdot] = ductfan_vec(xf,xr,omega_vec,theta0_vec,
                                theta1s_vec,theta1c_vec,deltav,irotor)

%Ducted fan model
%Inputs:
%xf  =  fuselage states
%xr  =  rotor states (just inflow for now)
%omega  =  rotor speed
%theta0  =  collectiuve pitch
%theta1s,theta1c =  cyclic pitch
%deltav  =  vane deflections
%irotor  =  rotor index, for multiple rotors

global  INIT VISC RHO RADIUS CHORD NB RCO BTL HROT XROT YROT ISHAFT
TWIST IDIRROT RETAB ALTAB CLTAB CDTAB D2R R2D PSIV SV VANEFF XVANE
YVANE ZVANE NVANES CHORDV CLVTAB CDVTAB ALVTAB REVTAB DELDRAG DXCG
DYCG DZCG AD X0IC FM  viqss1 viqss2 Tdata Vzdata Vindata OMEGAREF vi
nrows ncols npages;

persistent NR NPSI DR RSEG DPSI TSHAFT KAUG KCHI KCHI_R V0VITAB VTAB
DXTDTAB KAUGRED KVINDRAG TAUINFLOW STALLTAB INFLOW_MODEL;

%Initialize
if (INIT == 1)

    %Inflow Model Version (select 1 or 2 or 3)
    % 3 uses the lookup table
    INFLOW_MODEL = 2;

    %Define radial blade elements (use equal annular areas)
    NR = 6;
    DR = zeros(NR,1);
    RSEG = zeros(NR,1);
    AnnularArea = pi*(RADIUS^2-RCO^2)/NR;

    RSEG(1) = 0.5*RCO+sqrt((0.5*RCO)^2+AnnularArea/(4*pi));
    DR(1) = AnnularArea/(2*pi*RSEG(1));
    for ir = 2:NR;
        rinner = RSEG(ir-1)+0.5*DR(ir-1);
        RSEG(ir) = 0.5*rinner+sqrt((0.5*rinner)^2+AnnularArea/(4*pi));
        DR(ir) = AnnularArea/(2*pi*RSEG(ir));
    end

    %Define azimuthal step
    NPSI = 24;
    DPSI = 2*pi/NPSI;
    %Disk area
    AD = pi*RADIUS^2;
    %Transformation used for rotor shaft inclination relative to body
axes
    cis = cos(ISHAFT(irotor)*D2R);
    sis = sin(ISHAFT(irotor)*D2R);
    TSHAFT = [cis 0 sis; 0 1 0; -sis 0 cis];
```

```
    %Max. Thrust Augmentation Factor (augmentation in hover conditions)
    KAUG = 0.3;

   %Turning efficiency of duct (Freestream to infinity, apprev. of
KCHI_INF)
    KCHI = 0.8;

    %Portion of total turned flow that is turned before the rotor
    %(e.g. if KCHI_R = KCHI then all the flow is turned before the
rotor)
    KCHI_R = 0.6;

    %These parameters represent an empirical model to calculate momemnt
due the
    %duct due to assymetric flow in forward or sideward flight.  Also
approximate stall
    %effects using "Stall Factor".

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    %This section used with Flightlab duct thrust offset data
    VTAB =     [0. 5. 10. 15. 20. 25. 30. 35. 40. 45. 50. 55. 60. 65.
70. 75. 80. 85. 90. 95. 100. 105. 110. 115. 120. 125. 130. 135. 140.
200.];
    STALLTAB = [0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0
0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0];
    DXTDTAB =  [0.0 0.069 0.138 0.208 0.277 0.346 0.393 0.386 0.363
0.335 0.288 0.199 0.076 -0.014 -0.105 -0.181 -0.253 -0.324 -0.397 -
0.473 -0.521 -0.549 -0.572 -0.591 -0.605 -0.617 -0.626 -0.634 -0.639 -
0.639];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

    %Factor to model reduction in thrust augmentation due to stall
    KAUGRED = 0.5; % 50% reduction in thurst augmentation when leading
lip is completely stalled.

    %Inflow time constant
    TAUINFLOW = 0.1;
    %Induced drag coefficient of vanes, 1/(pi*e*AR), AR = Aspect ratio
of
    %vanes
    KVINDRAG = 1./(pi*0.8*RADIUS/CHORDV(1,1));
%%%%%% changed AR from 10 to RADIUS/CHORDV  (4-22-08)

    %Initialize forces, moments, torques to zero
    F = zeros(3,1);
    M = zeros(3,1);
    Qh = 0.;
    xrdot = 0;
end

theta0 = theta0_vec(irotor);
```

```
theta1s = theta1s_vec(irotor);
theta1c = theta1c_vec(irotor);
omega = omega_vec(irotor);

%Location of rotor hub relative to CG
hrot = HROT(irotor)+DZCG;
xrot = XROT(irotor)-DXCG;
yrot = YROT(irotor)-DYCG;
%Parameter represents rotation direction of rotor
idir = IDIRROT(irotor);
SignIdir = sign(idir);

%Define states
u = xf(1);
v = xf(2);
w = xf(3);
p = xf(4);
q = xf(5);
r = xf(6);
vi0 = xr(irotor);

%Calculate hub/wind axis velocities and angular rates
Vh = TSHAFT*[(u-q*hrot-r*yrot);(v+p*hrot+r*xrot);w-q*xrot+p*yrot];
Vtot = norm(Vh);
Vinplane = max(sqrt(Vh(1)^2+Vh(2)^2),1e-12);
Vz = -Vh(3);
alphad = atan2(Vz,Vinplane);
cpsiw = Vh(1)/Vinplane;
spsiw = Vh(2)/Vinplane;
psiw = atan2(Vh(2),Vh(1));
%This wind axis transformation
Tw2h = [cpsiw -spsiw;spsiw cpsiw];
omega2 = omega*omega;

%Vparx and Vparz parameters used to estimate shroud moment, stall
effects,
%max thurst augmentation
vparx = Vinplane/max(vi0,0.001);
vparz = Vz/max(vi0,0.001);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%
%This section used with Flightlab duct thrust offset data
%Shroud pitching moment parameter (x_TD/R)
xtdor = interp1(VTAB,DXTDTAB,Vtot);
%Stall factor
stallfac = interp1(VTAB,STALLTAB,Vtot);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%

%Time averaged integration of forces and moments over rotor disk
% Assumes rigid rotor
Tr = 0;
Lr = 0;
```

```
Mr = 0;
Qr = 0;
Xr = 0;
Yr = 0;

psi = [0:DPSI:2*pi-DPSI];
spsi = sin(psi);
cpsi = cos(psi);

% blade pitch
thetab = D2R*(theta0+SignIdir*theta1s*spsi+theta1c*cpsi);

%Blockage factor due to stall of duct lip, for now set to 0
blockage = zeros(1,NPSI);

%Tip loss, constant for now, can be function of stall effects
Btl = BTL*RADIUS*ones(1,NPSI);

%Blade segment pitch
thetaseg = ones(NR,1)*thetab + TWIST*D2R*((RSEG-RCO)/(RADIUS-
RCO))*ones(1,NPSI);
%Local velocity at the blade segment


chifact1 = (KCHI^2-2*KCHI);
chifact2 = 1.-KCHI;


if (INFLOW_MODEL == 2)

    %New Model
    VRx = Vtot*cos(alphad+KCHI_R*(pi/2-alphad));
    VRz = Vtot*sin(alphad+KCHI_R*(pi/2-alphad))+vi0;

    VEx = Vtot*cos(alphad+KCHI*(pi/2-alphad));
    VEz = Vtot*sin(alphad+KCHI*(pi/2-alphad))+vi0;
else
    %Old Model
    VRx = Vinplane*(1.-KCHI_R);
    VRz = Vz+vi0;

    VEx = Vinplane*(1.-KCHI);
    VEz = VRz;
end

Vr = sqrt(VRx^2+VRz^2);

%Perpindicular
%Vp = -Vz-vi0+RSEG*(p*spsi+q*cpsi);
Vp = -VRz+RSEG*(p*spsi+q*cpsi);
%Tangential (note term Vinplane*sin(psi+psiw)  =
%Vinplane*(spsi*cpsiw+cpsi*spisw) acounts for direction of relative
```

```
%wind.  SignIdir accounts for direction of rotation

Vt = RSEG*(omega+(1.-blockage)*(-
SignIdir*r))+VRx*ones(size(RSEG))*(spsi*cpsiw+cpsi*spsiw*SignIdir);

%Velocity and angle of attack and Re #
Vseg = sqrt(Vp.^2+Vt.^2);
alphaseg = mod(R2D*(thetaseg+atan2(Vp,Vt)),360);
Renumseg = max(RHO*Vseg*CHORD/VISC,160000);
Renumseg = min(Renumseg,5e6);

%Tip Loss Factor
tlfactor = (ones(NR,1)*Btl-(RSEG-
0.5*DR)*ones(1,NPSI))./(DR*ones(1,NPSI));
tlfactor = max(min(tlfactor,1.),0.);
% CL and CD table lookup for all blade elements at each azimuth
location
Clseg = interp2(RETAB,ALTAB,CLTAB,Renumseg,alphaseg).*tlfactor;
Cdseg = interp2(RETAB,ALTAB,CDTAB,Renumseg,alphaseg)+DELDRAG;
Fpseg = 0.5*RHO*CHORD*(DR*ones(1,NPSI)).*Vseg.*(Clseg.*Vt+Cdseg.*Vp);
Ftseg = 0.5*RHO*CHORD*(DR*ones(1,NPSI)).*Vseg.*(Cdseg.*Vt-Clseg.*Vp);

% Sum forces and moments over the blade
Fpblade = sum(Fpseg);
Ftblade = sum(Ftseg);
Mfblade = sum(Fpseg.*(RSEG*ones(1,NPSI)));
Mlblade = sum(Ftseg.*(RSEG*ones(1,NPSI)));

%Time averaged forces and moments of rotor
Lr = sum(-SignIdir*(Mfblade.*spsi))*NB/NPSI;
Mr = sum(Mfblade.*cpsi)*NB/NPSI;
Qr = sum(Mlblade)*NB/NPSI;
Tr = sum(Fpblade)*NB/NPSI;
Xr = sum(-Ftblade.*spsi)*NB/NPSI;
Yr = sum(-SignIdir*(Ftblade.*cpsi))*NB/NPSI;

%Inflow and duct effects

%Max thrust augmentation
if ( (chifact1*vparx^2+2*vparz+1) > 0)
   Kaug_max = (2*(vparz+1))/(chifact1*vparx^2+2*vparz+1)-1;
else
   Kaug_max = 999.;
end

%Kaug_eff = min(KAUG,Kaug_max)*(1.-stallfac*KAUGRED);
Kaug_eff = KAUG;

%Total thurst
Tduct = Kaug_eff*Tr;
T = Tduct+Tr;

%Momentum Drag (Ram Drag)
```

```
if (INFLOW_MODEL == 2)
    %New Model
 Dm = RHO*AD*Vr*(cos(alphad)-cos(alphad+0.5*(-2*alphad+pi)*KCHI))*Vtot;
else
    %Old Model
    Dm = RHO*AD*Vr*KCHI*Vinplane;
end

%Inflow Calculation

%augfact = 1./(1+Kaug_eff);
%viqs = vi0;
%infloweqn = 999.;
%iter = 0;
%iter_max = 100;
CT1=500:50:4000;
mu_z1=-0.1:0.005:0.1;
mu1=0:0.005:0.1;

    %Inflow table lookup

    del_CT=50;
    del_mu=0.005;

    CT=T/(RHO*AD);
    mu=Vinplane/425;
    mu_z=Vz/425;

    xx = 1 + (mu_z-mu_z1(1))/del_mu;
    yy = 1 + (mu-mu1(1))/del_mu;
    zz = 1 + (CT-CT1(1))/del_CT;

    % Check for out of range values of xx and set to 1
    sout = find((xx<1)|(xx>ncols));
    if ~isempty(sout), xx(sout) = ones(size(sout)); end

    % Check for out of range values of yy and set to 1
    tout = find((yy<1)|(yy>nrows));
    if ~isempty(tout), yy(tout) = ones(size(tout)); end

    % Check for out of range values of zz and set to 1
    wout = find((zz<1)|(zz>npages));
    if ~isempty(wout), zz(wout) = ones(size(wout)); end

    nw = nrows*ncols;
    ndx = floor(yy)+floor(xx-1)*nrows+floor(zz-1)*nw;

    % Compute intepolation parameters, check for boundary value.
    if isempty(xx), d = xx; else d = find(xx==ncols); end
    xx(:) = (xx - floor(xx));
    if ~isempty(d), xx(d) = xx(d)+1; ndx(d) = ndx(d)-nrows; end
```

```
    % Compute intepolation parameters, check for boundary value.
    if isempty(yy), d = yy; else d = find(yy==nrows); end
    yy(:) = (yy - floor(yy));
    if ~isempty(d), yy(d) = yy(d)+1; ndx(d) = ndx(d)-1; end

    % Compute intepolation parameters, check for boundary value.
    if isempty(zz), d = zz; else d = find(zz==npages); end
    zz(:) = (zz - floor(zz));
    if ~isempty(d), zz(d) = zz(d)+1; ndx(d) = ndx(d)-nw; end

    % Now interpolate.
    viqs =  (( vi(ndx).*(1-yy) + vi(ndx+1).*yy ).*(1-xx) +
    ( vi(ndx+nrows).*(1-yy) + vi(ndx+(nrows+1)).*yy ).*xx).*(1-zz) +
    (( vi(ndx+nw).*(1-yy) + vi(ndx+1+nw).*yy ).*(1-xx) +
    ( vi(ndx+nrows+nw).*(1-yy) + vi(ndx+(nrows+1+nw)).*yy ).*xx).*zz;

    % Now set out of range values to ExtrapVal.
    if ~isempty(sout), viqs(sout) = fsolve(@(viqs)
      inflow_func(viqs,CT,mu_z,mu),45,options); end
    if ~isempty(tout), viqs(tout) = fsolve(@(viqs)
      inflow_func(viqs,CT,mu_z,mu),45,options); end
    if ~isempty(wout), viqs(wout) = fsolve(@(viqs)
      inflow_func(viqs,CT,mu_z,mu),45,options); end

xrdot = 1./TAUINFLOW*(viqs-vi0);

%Swirl angular velocity
omega_swirl = Qr/(RHO*AD*Vr*0.5*RADIUS^2)*SignIdir;

    %Vane center of pressure position vector
r_v = [XVANE(1,irotor) XVANE(2,irotor) XVANE(2,irotor) XVANE(2,irotor);
    YVANE(1,irotor) YVANE(2,irotor) YVANE(3,irotor) YVANE(4,irotor);
    ZVANE(1,irotor) ZVANE(2,irotor) ZVANE(3,irotor) ZVANE(4,irotor)];
    %velocity vector in hub-wind axes
    Vv_vec1 = [VEx*cpsiw;VEx*spsiw;-VEz];
    cpsiv = cos(PSIV(:,irotor)*D2R);
    spsiv = sin(PSIV(:,irotor)*D2R);
   %Add swirl component
    Vtheta=zeros(1,4);
Vtheta(:) = omega_swirl*[norm(r_v(1:2,1)) norm(r_v(1:2,2))
            norm(r_v(1:2,3)) norm(r_v(1:2,4))];
    Vv_vec(1,:) = Vv_vec1(1)-Vtheta.*spsiv';
    Vv_vec(2,:) = Vv_vec1(2)-Vtheta.*cpsiv';
    Vv_vec(3,:) = Vv_vec1(3)*[1 1 1 1];

    %Add angular motion
    Vv_vec = Vv_vec+[cross([p;q;r],r_v(:,1)) cross([p;q;r],r_v(:,2))
            cross([p;q;r],r_v(:,3)) cross([p;q;r],r_v(:,4))];
    Vv_tot = [norm(Vv_vec(:,1)) norm(Vv_vec(:,2)) norm(Vv_vec(:,3))
            norm(Vv_vec(:,4))];
    qv = 0.5*RHO*Vv_tot(:).^2;

    %Transformation from hub-wind axes to vane axes
```

```
    % uv  =  chordwise vel
    % vv  =  spanwise vel
    % wv  =  norm vel
    Th2v=zeros(3,3,4);
    Th2v(:,:,1) = [0 0 -1;-1 0 0;0 1 0]*[cpsiv(1) -spsiv(1) 0;spsiv(1)
      cpsiv(1) 0; 0 0 1];
    Th2v(:,:,2) = [0 0 -1;-1 0 0;0 1 0]*[cpsiv(2) -spsiv(2) 0;spsiv(2)
      cpsiv(2) 0; 0 0 1];
    Th2v(:,:,3) = [0 0 -1;-1 0 0;0 1 0]*[cpsiv(3) -spsiv(3) 0;spsiv(3)
      cpsiv(3) 0; 0 0 1];
    Th2v(:,:,4) = [0 0 -1;-1 0 0;0 1 0]*[cpsiv(4) -spsiv(4) 0;spsiv(4)
      cpsiv(4) 0; 0 0 1];

    temp = [Th2v(:,:,1)*Vv_vec(:,1) Th2v(:,:,2)*Vv_vec(:,2)
      Th2v(:,:,3)*Vv_vec(:,3) Th2v(:,:,4)*Vv_vec(:,4)];
    uv = temp(1,:);
    vv = temp(2,:);
    wv = temp(3,:);

    %AOA and sideslip of vane
        alphav = atan2(wv(:),uv(:));
        betav = asin(vv(:)./Vv_tot(:));
    %Transform from vane wind axes back to vane axes
    Tw2v=zeros(3,3,4);
    Tw2v(:,:,1) = [cos(alphav(1)) 0 -sin(alphav(1));0 1
      0;sin(alphav(1)) 0 cos(alphav(1))]*[cos(betav(1)) -sin(betav(1))
      0;sin(betav(1)) cos(betav(1)) 0; 0 0 1];
    Tw2v(:,:,2) = [cos(alphav(2)) 0 -sin(alphav(2));0 1
      0;sin(alphav(2)) 0 cos(alphav(2))]*[cos(betav(2)) -sin(betav(2))
      0;sin(betav(2)) cos(betav(2)) 0; 0 0 1];
    Tw2v(:,:,3) = [cos(alphav(3)) 0 -sin(alphav(3));0 1
      0;sin(alphav(3)) 0 cos(alphav(3))]*[cos(betav(3)) -sin(betav(3))
      0;sin(betav(3)) cos(betav(3)) 0; 0 0 1];
    Tw2v(:,:,4) = [cos(alphav(4)) 0 -sin(alphav(4));0 1
      0;sin(alphav(4)) 0 cos(alphav(4))]*[cos(betav(4)) -sin(betav(4))
      0;sin(betav(4)) cos(betav(4)) 0; 0 0 1];

    %Lift and drag of vanes
    %deltav defined positive trailing edge deflected in clockwise
    %direction
    Renumv = RHO*Vv_tot.*CHORDV(:,irotor)'/VISC;
    Renumv = max(min(Renumv(:),5e6),160000);
    alphavtab = mod(R2D*alphav(:)-deltav(:,irotor),360);
    Clv = interp2(REVTAB,ALVTAB,CLVTAB,Renumv(:),alphavtab(:))*VANEFF;
    Cdv =
interp2(REVTAB,ALVTAB,CDVTAB,Renumv(:),alphavtab(:))+KVINDRAG*Clv(:).^2
;
    Dvane = Cdv.*qv.*SV(:,irotor);
    Lvane = Clv.*qv.*SV(:,irotor);
    temp = [Tw2v(:,:,1)*[-Dvane(1);0;-Lvane(1)] Tw2v(:,:,2)*[-
      Dvane(2);0;-Lvane(2)] Tw2v(:,:,3)*[-Dvane(3);0;-Lvane(3)]
      Tw2v(:,:,4)*[-Dvane(4);0;-Lvane(4)]];

%Transform vane forces and moments into hub axes
```

```
    Fvane = [Th2v(:,:,1)'*temp(:,1) Th2v(:,:,2)'*temp(:,2)
      Th2v(:,:,3)'*temp(:,3) Th2v(:,:,4)'*temp(:,4)];
    Mvane = cross(r_v,Fvane);
    Xv = sum(Fvane(1,:));
    Yv = sum(Fvane(2,:));
    Zv = sum(Fvane(3,:));
    Lv = sum(Mvane(1,:));
    Mv = sum(Mvane(2,:));
    Nv = sum(Mvane(3,:));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Calculate drag on duct
[Ff,Mf] = fuselage_drag([u;v;w]);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Forces and moments in hub system
Fh = zeros(3,1);
Mh = zeros(3,1);

Fh(1) = Xr-Dm*cpsiw+Xv;
Fh(2) = Yr-Dm*spsiw+Yv;
Fh(3) = -T+Zv;

Mh(1) = Lr-Tduct*xtdor*RADIUS*spsiw+Lv;
Mh(2) = Mr+Tduct*xtdor*RADIUS*cpsiw+Mv;
Mh(3) = Qr*SignIdir+Nv;

%Transform to body systems
F = TSHAFT'*Fh+Ff;
M = TSHAFT'*Mh+Mf+cross([xrot;0;-hrot],F);

return
```

```
function [xdot]=eqnmot(x,FM)

%Equations of Motion Module

%Inputs: Fuselage State Vector, x
%        Total Aerodynamic Forces and Moments, FM
%Outputs: Fuselage State Vector Derivative, xdot

%This module simulates the standard aircraft, rigid body equations of
motion

global MASS G IX IY IZ IXZ;

u=x(1);
v=x(2);
w=x(3);
p=x(4);
q=x(5);
r=x(6);
phi=x(7);
theta=x(8);
psi=x(9);

X=FM(1);
Y=FM(2);
Z=FM(3);
L=FM(4);
M=FM(5);
N=FM(6);

cphi=cos(phi);
sphi=sin(phi);
cthe=cos(theta);
sthe=sin(theta);
cpsi=cos(psi);
spsi=sin(psi);

xdot=zeros(12,1);


%The following equations are given in Padfield pages 92 and 173-178
% Calculate state derivatives
%Velocities
%udot eqn.
xdot(1)=X/MASS-G*sthe-q*w+r*v;
%vdot eqn.
xdot(2)=Y/MASS+G*cthe*sphi-r*u+p*w;
%wdot eqn.
xdot(3)=Z/MASS+G*cthe*cphi-p*v+q*u;

%Angular rates
gam=IX*IZ-IXZ^2;
%pdot eqn.
```

```
xdot(4)=(IZ*L+IXZ*N+IXZ*(IX-IY+IZ)*p*q-(IZ^2-IY*IZ+IXZ^2)*q*r)/gam;
%qdot eqn.
xdot(5)=(M+(IZ-IX)*p*r-IXZ*(p^2-r^2))/IY;
%rdot eqn.
xdot(6)=(IX*N+IXZ*L-IXZ*(IX-IY+IZ)*q*r+(IX^2-IX*IY+IXZ^2)*p*q)/gam;

%Attitudes
%phi dot eqn.
xdot(7)=p+q*sphi*sthe/cthe+r*cphi*sthe/cthe;
%theta dot eqn.
xdot(8)=q*cphi-r*sphi;
%psi dot eqn.
xdot(9)=q*sphi/cthe+r*cphi/cthe;

%Position
xdot(10)=u*cthe*cpsi+v*(sphi*sthe*cpsi-
cphi*spsi)+w*(cphi*sthe*cpsi+sphi*spsi);
xdot(11)=u*cthe*spsi+v*(sphi*sthe*spsi+cphi*cpsi)+w*(cphi*sthe*spsi-
sphi*cpsi);
xdot(12)=-u*sthe    +v*sphi*cthe                 +w*cphi*cthe;

return;



function [u2]=sim2(u)

T=[1:1:1]';

UT=[T,u(1),u(2),u(3),u(4),u(5),u(6)];

%mix is the simulink diagram with only control mixing block in it.
[T,X,Y]=sim('mix',[],[],UT);
u2=Y(1,:);

return
```

B.2 Embedded Matlab Codes in Simulink Diagrams

`Control Allocation` at `Outer Loop` Block:

```
function [angle, vane] = fcn(Vdot,BB,BB2)

angle=[0;0];
vane=BB^-1*(Vdot-BB2*angle);

if vane(1)<-50
    vane(1)=-50;
    if vane(2)<-50
        vane(2)=-50;
    elseif vane(2)>50
        vane(2)=50;
    end
    angle=BB2^-1*(Vdot-BB*vane);
elseif vane(1)>50
    vane(1)=50;
    if vane(2)<-50
        vane(2)=-50;
    elseif vane(2)>50
        vane(2)=50;
    end
    angle=BB2^-1*(Vdot-BB*vane);
else
    if vane(2)<-50
        vane(2)=-50;
        angle=BB2^-1*(Vdot-BB*vane);
    elseif vane(2)>50
        vane(2)=50;
        angle=BB2^-1*(Vdot-BB*vane);
    else
        angle=[0;0];
    end
end

%vane=[0;0];
%angle=BB2^-1*(Vdot-BB*vane);
```

`Euler Angle Converter` at `Model Follower and Feedback` Block:

```
function [p_dot,q_dot,r_dot,w_dot] = fcn(phi_ddot,theta,theta_ddot,phi,
                        rd_dot,phi_dot,w_dott,theta_dot,psi_dot,u)

p_dot = phi_ddot-(theta_ddot*sin(phi)*sin(theta)+rd_dot*sin(theta)+
      psi_dot*theta_dot*cos(phi)+theta_dot*phi_dot*cos(phi)*sin(theta)
+psi_dot*phi_dot*sin(theta)*sin(phi)*cos(theta))/(cos(phi)*cos(theta));

q_dot = (theta_ddot+rd_dot*sin(phi)+
      psi_dot*phi_dot*cos(theta))/cos(phi);

r_dot = rd_dot;

w_dot = (w_dott+u*theta_dot*cos(theta))/(cos(theta)*cos(phi));
```

`Feedback` at `Model Follower and Feedback` Block:

```
function [phi_dot,theta_dot,psi_dot] = fcn(p,q,r,phi,theta)

phi_dot=p+q*sin(phi)*tan(theta)+r*cos(phi)*tan(theta);

theta_dot=q*cos(phi)-r*sin(phi);

psi_dot=q*sin(phi)*sec(theta)+r*cos(phi)*sec(theta);
```

`Velocity Body to Earth Coordinates` at `Model Follower and Feedback` Block:

```
function [Vx,Vy,Vz] = fcn(u,a)

Vx = u(1)*cos(a(2))*cos(a(3))+u(2)*(-cos(a(1))*sin(a(3))
      +sin(a(1))*sin(a(2))*cos(a(3)))+u(3)*(sin(a(1))*sin(a(3))
      +cos(a(1))*sin(a(2))*cos(a(3)));
Vy = u(1)*cos(a(2))*sin(a(3))+u(2)*(cos(a(1))*cos(a(1))
      +sin(a(1))*sin(a(2))*sin(a(3)))+u(3)*(-sin(a(1))*cos(a(3))
      +cos(a(1))*sin(a(2))*sin(a(3)));
Vz = -(u(1)*sin(a(2))-u(2)*sin(a(1))*cos(a(2))-
      u(3)*cos(a(1))*cos(a(2)));
```

`NEU to Body` at `Gust` Block:

```
function gust = fcn(NEU,phi,theta,psi)

cphi=cosd(phi);
sphi=sind(phi);
cthe=cosd(theta);
sthe=sind(theta);
cpsi=cosd(psi);
spsi=sind(psi);

transmatrix=[cthe*cpsi (sphi*sthe*cpsi-cphi*spsi)
      (cphi*sthe*cpsi+sphi*spsi);
    cthe*spsi (sphi*sthe*spsi+cphi*cpsi) (cphi*sthe*spsi-sphi*cpsi);
    sthe sphi*cthe cphi*cthe];

gust = transmatrix'*NEU;
```

`Velocity Body to Earth Coordinates` at `States` Block:

```
function [Vx,Vy,Vz,Vfwd,Vside] = fcn(u,a)

Vx = u(1)*cos(a(2))*cos(a(3))+u(2)*(-cos(a(1))*sin(a(3))
      +sin(a(1))*sin(a(2))*cos(a(3)))+u(3)*(sin(a(1))*sin(a(3))
      +cos(a(1))*sin(a(2))*cos(a(3)));
Vy = u(1)*cos(a(2))*sin(a(3))+u(2)*(cos(a(1))*cos(a(3))
      +sin(a(1))*sin(a(2))*sin(a(3)))+u(3)*(-sin(a(1))*cos(a(3))
      +cos(a(1))*sin(a(2))*sin(a(3)));
Vz = -(u(1)*sin(a(2))-u(2)*sin(a(1))*cos(a(2))-
      u(3)*cos(a(1))*cos(a(2)));
Vfwd = Vx*cos(a(3))+Vy*sin(a(3));
Vside = -Vx*sin(a(3))+Vy*cos(a(3));
```

Appendix C

C.1 Calculation of Momentum Equation

The mass flow rate of air through the duct is determined by Equation C.1.

$$\dot{m} = \rho A_D \left|\mathbf{V_R}\right| \tag{C.1}$$

where ρ is the local air density and A_D is the duct area at the plane of the rotor. The velocity vectors, shown in Figure 2-1 and given in Equation 2.2, are used in the derivation of the momentum equation. Conservation of momentum is then applied across the stream tube, from freestream flow, $\mathbf{V_0}$, to the far-wake flow, $\mathbf{V}_\infty$, is calculated by Equation C.2. The reason why the far-wake flow velocity is used in determining the control volume for the conservation of momentum, instead of the velocity right after the rotor disk, is to avoid the wake effects right at the downstream of the rotor.

$$\mathbf{T} = \dot{m}\left(\mathbf{V}_\infty - \mathbf{V_0}\right) \tag{C.2}$$

Considering only the vertical component of Equation C.2, the thrust can be found.

$$T = \dot{m}\left(V_0 \sin\alpha^\infty + \upsilon_\infty - V_0 \sin\alpha\right) = \dot{m}\left(\left(-\sin\alpha + \sin\alpha^\infty\right)V_0 + \upsilon_\infty\right) \tag{C.3}$$

It is assumed that energy enters the system through the rotor thrust, so conversation of energy can be applied as in Equation C.4.

$$\begin{aligned}
&\mathbf{T_R}.\mathbf{V_R} = \frac{1}{2}\dot{m}\left(\mathbf{V}_\infty \cdot \mathbf{V}_\infty - \mathbf{V_0} \cdot \mathbf{V_0}\right) \\
&T_R\left(V_0 \sin\alpha^R + \upsilon_i\right) = \frac{1}{2}\dot{m}\left(\left(V_0 \cos\alpha^\infty\right)^2 + \left(V_0 \sin\alpha^\infty + \upsilon_\infty\right)^2 - {V_0}^2\right) \\
&\frac{T}{1+k_{aug}}\left(V_0 \sin\alpha^R + \upsilon_i\right) = \frac{1}{2}\dot{m}\left(\left(V_0 \cos\alpha^\infty\right)^2 + \left(V_0 \sin\alpha^\infty + \upsilon_\infty\right)^2 - {V_0}^2\right)
\end{aligned} \tag{C.4}$$

Substituting the thrust found in C.3 into C.4 would result in below equation.

$$\frac{\dot{m}\left(\left(-\sin\alpha+\sin\alpha^{\infty}\right)V_0+\upsilon_{\infty}\right)}{1+k_{aug}}\left(V_0\sin\alpha^{R}+\upsilon_i\right)$$
$$=\frac{1}{2}\dot{m}\left(V_0{}^2\cos^2\alpha^{\infty}+V_0{}^2\sin^2\alpha^{\infty}+2V_0\sin\alpha^{\infty}\upsilon_{\infty}+\upsilon_{\infty}{}^2-V_0{}^2\right)=\frac{1}{2}\dot{m}\left(2V_0\sin\alpha^{\infty}\upsilon_{\infty}+\upsilon_{\infty}{}^2\right)$$

(C.5)

From Equation C.5, a quadratic expression for υ_{∞} can be obtained as below,

$$\upsilon_{\infty}{}^2+2\left(V_0\sin\alpha^{\infty}\frac{V_0\sin\alpha^{R}+\upsilon_i}{1+k_{aug}}\right)\upsilon_{\infty}-\frac{2}{1+k_{aug}}\left[\left(-\sin\alpha+\sin\alpha^{\infty}\right)\left(V_0\sin\alpha^{R}+\upsilon_i\right)V_0\right]=0$$

(C.6)

The induced velocity far downstream from the duct, υ_{∞}, can be solved using the quadratic equation C.6. Substituting the terms υ_{∞} and $\dot{m}$ into the equation C.3, would give us the Equation 2.5, which is a complex nonlinear algebraic equation.

Bibliography

[1] Pereira, Jason L. 2008, “Hover and Wind-Tunnel Testing of Shrouded Rotors for Improved Micro Air Vehicle Design”, PhD Thesis, University of Maryland, College Park.

[2] Huber, M 2007, *Piasecki compound X-49 makes first flight*, news article, Aviation International News, viewed 27 April 2010, <http://www.ainonline.com/ain-and-ainalerts/aviation-international-news/single-publication-story/browse/0/article/piasecki-compound-x-49-makes-first-flight-9185/?no_cache=1&tx_ttnews [story_pointer]=3&tx_ttnews[mode]=1>

[3] Yoeli, R., “Ducted Fan Utility Vehicles and Other Flying Cars,” *Collection of Technical Papers – 2002 Biennial International Powered Lift Conference and Exhibit,* Williamsburg, VA, United States, 2002, pp. 1-6.

[4] Egozi, A 2010, *Urban’s Air Mule achieves sustained tethered hover*, news article, Flight International, viewed 27 April 2010, <http://www.flightglobal.com/articles/2010/04/26/340979/picture-urbans-air-mule-achieves-sustained-tethered-hover.html>

[5] Tobias, E. and Horn, J.,”Simulation Analysis of the Controllability of a Tandem Ducted Fan Aircraft”, *Proceedings of AIAA Atmospheric Flight Mechanics Conference and Exhibit* 2008-6700

[6] Avanzini, G., D’Angelo, S., and De Matteis, G., “Performance and Stability of Ducted-Fan Uninhabited Aerial Vehicle Model,” *Journal of Aircraft*, Vol. 40, No. 1, 2003, pp. 86-93.

[7] De Divitiis, N., "Performance and Stability Analysis of a Shrouded-Fan Unmanned Aerial Vehicle," *Journal of Aircraft*, Vol. 43, No. 3, 2006, pp. 681-691.

[8] Avanzini, G., Ciniglio, U., and De Matteis, g., "Full-Envelope Robust Control of a Shrouded-Fan Unmanned Vehicle," *Journal of guidance, Control and Dynamics*, Vol. 29, No. 2, 2006, pp. 435-443.

[9] Johnson, E. and Turbe, M., "Modeling, Control, and Flight Testing of a Small Ducted-Fan Aircraft," *Journal of Guidance, Control and Dynamics*, Vol. 29, No. 4, 2006, pp. 769–779.

[10] Johnson, E. and Kannan, S., "Adaptive Trajectory Control for Autonomous Helicopters", *Journal of Guidance, Control, and Dynamics,* Vol. 28, No. 3, 2005, pp. 524-538.

[11] Sahani, N. and Horn, J., "Adaptive Model Inversion Control of a Helicopter with Structural Load Limiting", *Proceedings of AIAA Guidance, Navigation, and Control Conference and Exhibit* 2004-4753

[12] Hess, R. and Bakhtiari-Nejad, M., "Sliding Mode Control of a Nonlinear Ducted-Fan UAV Model" *Collection of Technical Papers - AIAA Guidance, Navigation, and Control Conference* 2006 , Vol. 2, Keystone, CO, United States, 2006, pp. 748–762.

[13] Spaulding, C., Mansur, M., Tischler, M., Hess, R., and Franklin, J., "Non-linear Inversion Control for a Ducted Fan UAV", *Proceedings of AIAA Atmospheric Flight Mechanics Conference and Exhibit* Vol. 2, San Francisco, CA, United States, 2005, pp. 1209–1234.

[14] Horn, J.F. and Bridges, D.O., "A Model Following Controller Optimized for Gust Rejection during Shipboard Operations," *Proceedings of the American Helicopter Society 63rd Annual Forum*, Virginia Beach VA, May 1-3, 2007.

[15] Horn, J. F., Guo, W., "Flight Control Design for Rotorcraft with Variable Rotor Speed," *Proceedings of the American Helicopter Society 64th Annual Forum, 2008*

[16] Geiger, B., "Flight Control Optimization on a Fully Compounded Helicopter with Redundant Control Effectors," M.Sc. Thesis, Aerospace Engineering, Penn State University, State College, PA, 2005.

[17] Leishman, J.G., *Principles of Helicopter Aerodynamics*, Cambridge University Press, New York, NY, United States, 2nd ed., 2006.

[18] Phillips, W., "Propeller Momentum Theory with Slipstream Rotation," *Journal of Aircraft*, Vol. 39, No. 1, 2002, pp. 184-187

[19] Birkbeck, N., Levesque, J., and Amaral, J. N., "A Dimension Abstraction Approach to Vectorization in Matlab," *International Symposium on Code Generation and Optimization, CGO 2007,* San Josa, CA, United States, 2007, pp. 115-127.

[20] Padfield, G.D., *Helicopter Flight Dynamics: The Theory and Application of Flying Qualities and Simulation Modelling*, AIAA Education Series, Blacksburg, VA, United States, 2nd ed., 2007.