A Comparison of Two Methods Used to Deal with Saturation of Multiple, Redundant Aircraft Control Effectors

Mark D. Nelson

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Aerospace Engineering

Dr. Wayne Durham - chair
Dr. Frederick Lutze
Dr. Chris Hall

August 2001
Blacksburg, Virginia

Keywords: Control Allocation, Moment Direction Preservation, Moment Prioritization, Control Saturation, Redundant Aircraft Controls

A Comparison of Two Methods Used to Deal with Saturation of Multiple, Redundant Aircraft Control Effectors

Mark D. Nelson

(ABSTRACT)

A comparison of two methods to deal with allocating controls for unattainable moments in an aircraft was performed using a testbed airframe that resembled an F/A-18 with a large control effector suite. The method of preserving the desired moment direction to deal with unattainable moments is currently used in a specific control allocator. A new method of prioritizing the pitch axis is compared to the moment-direction preservation. Realtime piloted simulations are completed to evaluate the characteristics and performance of these methods.

A direct comparison between the method of preserving the moment direction by scaling the control solution vector and prioritizing the pitching moment axis is performed for a specific case. Representative maneuvers are flown with a highly unstable airframe to evaluate the ability to achieve the specific task. Flight performance and pilot interpretation are used to evaluate the two methods.

Pilot comments and performance results favored the method of pitch-axis prioritization. This method provided favorable flight characteristics compared to the alternative method of preserving the moment direction for the specific tasks detailed in this paper.

Acknowledgments

I would like to thank first and foremost my parents for their support and unending advice that have helped me to make the most of myself and live my life to the fullest. To my sister Julie and my brother Daniel with whom I have the fortune of knowing. Julie's pursuit to make the most of life in the face of constant hurdles, and Dan's quest to never be in the shadow of older siblings has made our relationships both exhaustive and rewarding.

I would like to thank my advisor Dr. Wayne Durham for his persistence and patience with me and my academic career. He has given me the perfect work environment and has constantly pushed me, with much resistance, to achieve all that I could. To the remaining members of my commitee, Dr. Fred Lutze and Dr. Chris Hall, thank you for making my education at Virginia Tech a wonderful experience and for sharing everyday life experiences with me.

I must thank my friends that have made my life at Virginia Tech so enjoyable. To Mike Henry, Kevin Waclawicz, Dan Hart, Todd Norell, Trevor Wallace, Roger Beck, Josh Durham, Bill Oetjens, and others that mean so much to me. To the friend that I lost, and the ones that I have recently gained, I will always remember you. Finally, I would like to thank the few friends back home who have offered me support from a long ways away and have never lost touch.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Classically, aircraft flight controls have been designed to use a single control effector to produce a desired moment about a specific rotational degree of freedom. For instance, the elevators are used to control the pitch-moment, while the ailerons and rudder are primarily used for roll moment and yaw moment control respectively. With this arrangement of one effector per rotational degree of freedom a unique solution can be determined to generate a desired moment. However, many modern tactical fighter aircraft have more than three independent moment generating effectors. These additional effectors arise from freeing opposing moment generators (effectors) and controlling them independently. For instance, left and right horizontal tails can be used individually to produce roll, pitch or yaw moments. Non-traditional effectors are also added, such as thrust vectoring, canards, and elevons, to name a few. Each effector on the aircraft is treated as a separate control. With this arrangement the number of control effectors on an aircraft could total close to 20.

The problem of allocating these redundant control effectors to achieve the desired objective is complex. In general there will be an infinite number of solutions to the problem. However, most of these solutions are not realistic for an actual aircraft due to control limits created by the physical geometry of the effector and aerodynamic constraints on the surfaces. Therefore, the solution to the problem must incorporate these effector limits and provide an admissible control solution.

1.2 Allocation Problem

The distribution of controls to obtain a certain objective is the general allocation problem. In an aircraft specific problem it is to find a control solution that generates the necessary forces and moments to alter the aircraft's acceleration. The controls enter into the aircraft's equations of motion as seen in equation 1.1. These equations come from Newton's Second Law of Motion and are frequently linearized about some reference condition and written

$$\dot{x} = Ax + Bu \tag{1.1}$$

Where $\dot{x}$ are the aircraft angular accelerations, Ax are the inherent dynamics of the aircraft, and $\mathbf{B}u$ is the effect of the controls on the angular accelerations. The problem is solved by finding a control vector to produce desired aircraft dynamics. To solve for the control vector, equation 1.1 becomes

$$\mathbf{B}u = \dot{x_d} - Ax = m_d \tag{1.2}$$

where $\dot{x_d}$ are the desired dynamics of the aircraft and m_d is defined as the desired moment vector for this problem. Some inertia terms are divided out of m_d such that units will match. The problem then becomes the determination of the controls u that generate the desired moments m_d. Consider m control effectors in the control vector u that produce n objectives m_d, usually 3 moments. These objectives are generated through a linear mapping $\mathbf{B} : \mathbf{R^m} \ \rightarrow \mathbf{R^n}$ using matrix multiplication as described in equation 1.2.

$\mathbf{B}$ is the control effectiveness matrix which describes the effect each control has on a specific moment direction. The controls are constrained values between the minimum and maximum values which defines the admissible control subset. Moments which can be generated using some combination of admissible controls will be referred to as attainable moments.

A problem that could occur with control allocation is that the control law may command moments which are impossible to achieve. It is important that any control allocation method implemented in a real aircraft be able to deal with a situation when no valid solution exists. This research examines such a method in dealing with these unattainable moments.

Note that other difficulties arise in control allocation such as the fact that the linearized equation $m_d = Bu$ is only an approximation. As aircraft flight conditions change, nonlinear effects can introduce error into the problem. A control allocator should be able to effectively deal with these changes to reduce the errors associated with them.

1.3 Allocation Methods

A schematic of the general arrangement of a control system, with control law and control allocator, is presented in Figure 1.1. In the figure, δ_{stick} represents the stick deflection and rudder pedals from the pilot station which is converted into desired moments m_d by the control law. A dynamic inversion control law was chosen so that desirable dynamics could be specified for the test-bed airframe used in this research. Dynamic inversion allows one to tailor the shape of the responses by calculating the forces and moments required of the controls. These desired moments are then allocated to form a control vector that will produce the desired flight dynamics.

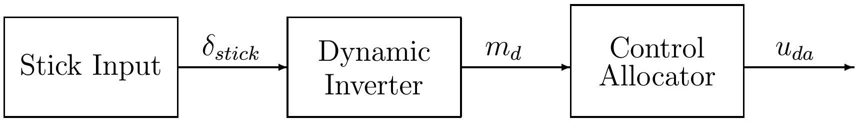

Figure 1.1: Control Loop Block Diagram

Early control allocation methods were computationally simple yet restrictive in that the entire moment-generating capabilities of the control effectors were not utilized. Some of these methods of control allocation are presented in Bordignon [4]. In general, there are an infinite number of ways to allocate controls. These range from a simple control interconnect to several mathematical algorithms for determining the control solution. Some methods such as daisy chaining and generalized inverses are often used [4]. A common method is the minimum norm solution, often referred to as the pseudoinverse, which minimizes control energy. All of these methods provide a simple linear approach to solving the allocation problem. Even with additional methods to improve the solution, it is shown that these methods can not obtain the maximum moment generating capabilities [1].

Another subset of mathematical solutions is referred to as the optimal solution. This method determines the unique control vector solution that corresponds to the maximum attainable moment in the direction of the desired moment, and scales the solution as necessary to yield the desired moment. One method that has been developed is solving the problem using cascaded generalized inverses (CGI) [4]. This method calculates several pseudoinverses, each of decreasing order. This method will generally produce admissible solutions for problems with attainable desired objectives. However, for cases where the objective is not attainable, large errors may be produced in the solution.

A second method developed by Durham is presented in several papers [1–4]. This algorithm is a brute force method of searching the entire subset of attainable moments given by the geometry of the problem to produce a solution. This method was proven [4, 9] to work, however it also was computationally intensive and therefore not practical for implementation into real-time flight control computers. This method was replaced by a more computationally efficient method of solving the control allocation problem using an optimal solution.

Durham [6] introduced the Bisecting, Edge-Searching Algorithm (BESA) to efficiently allocate controls. This method is a computationally efficient method that produces near-optimal solutions to the linear three-objective control problem. Further work by Durham and Scalera [7] explores this algorithm and compares the performance to an existing F-15 ACTIVE (Advanced Control Technology for Integrated VEhicles) allocator using real-time pilot-in-the-loop simulations. The research done by Durham and Scalera focused on the restoring aspects of allocation as well as control reconfiguration in the event of a surface failure. Their research provides an investigation into the performance qualities of the allocator by direct comparison to an allocation scheme already in use using aggressive tasks and edge-of-the-envelope flight conditions. Durham and Scalera demonstrate that this method of control allocation provides a guarantee that the full moment capability is utilized in the control solution. The theory behind this control allocator is provided in Chapter 2.

1.4 Attainable Moments

The term attainable refers to the set of moments that can be generated given the physical limits of the effectors. Control vector solutions that are within the physical limits are admissible solutions. The geometry of the set of attainable moments is the projection of an m-dimensional box using a linear mapping, where m is the number of effectors, into three-dimensional space (moment space) to form a polytope. Each vertex of the m-dimensional box is formed by the effectors being at a maximum or minimum limit. The linear mapping is defined by the geometry and aerodynamic characteristics of each individual effector known as the control effectiveness matrix **B**. The calculation of the attainable moment subset (AMS) is simple and defines individual facets of the polytope [5]. The AMS defines the subset of maximum attainable moments in any direction. Figure 1.2 is an example of an AMS calculated from a 4-control problem. One facet is defined by two controls free to vary, with all other controls being saturated. This means that a vertex of the AMS represents a moment that is generated by all 4 controls being at a limit, while an edge represents one control being free to vary [5].

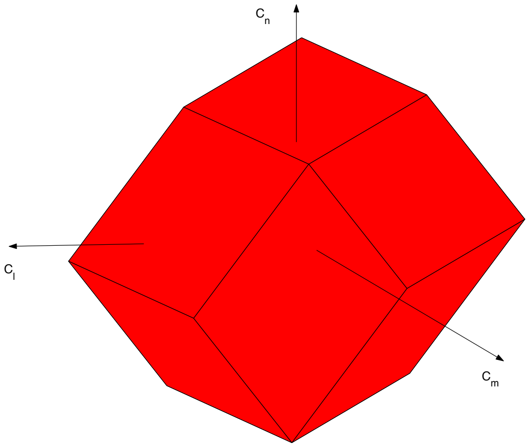

Figure 1.2: General Attainable Moment Subset for a 4 Control Problem

An additional constraint to the control allocation problem could involve the rate limits for each effector. To include the rate capabilities of each effector the control allocation problem is done on a frame wise basis. This method is known as Moment Rate Allocation (MRA), and it accounts for both position and rate limits of each individual effector. This method was used by Bolling in batch mode simulation [9]. The method, strictly speaking, is the same as allocating on a global basis (ignoring rate limits), except that the AMS calculated in MRA is a smaller subset of the global AMS. The smaller AMS results from the limits of the problem being reset to account for the rate of travel for each time frame, whereas the global allocation allows for an effector to travel from limit to limit in one time step. Any of the allocation routines explained above are capable of being used on a frame wise basis. This research however utilizes the BESA method on a global basis, ignoring rate limits of each effector.

Unattainable moments occur when the control law asks for a desired moment outside of the AMS. This leads to an inadmissible control vector, meaning that one or more individual controls exceed the maximum or minimum limit. Control laws could produce unattainable moments because they are not inherently designed with the knowledge of the limits for each effector. A specific effector could also fail or be damaged leading to changed capabilities from reduced limits.

There are several ways in dealing with inadmissible control solutions. The two most common are scaling and prioritization. The problem is most easily dealt with by preserving the moment direction. This involves scaling the control vector such that none of the controls exceed a limit. This method is used most often, however it ignores the effects on the resulting angular accelerations. Figure 1.3 shows the scaling technique to preserve the moment direction. The black arrow indicates the desired moment while the red arrow corresponds to the moment produced with the scaled control vector. Another method is to prioritize a certain objective while neglecting a less important objective. This less important objective(s) is sacrificed until the control effectors are no longer commanded past their limits. In this method the problem is broken up into two parts. The first part is the calculation of the control solution that provides the full amount of the priority objective. The second part of the problem uses the remaining control capabilities to acheive the secondary objectives. Figure 1.4 shows the prioritization technique. The method prioritizes the pitch-axis such that all of the demanded pitch moment is used, this moment is indicated by the red circles. Note that this method attains almost twice the amount of pitch moment that the scaled version attains (yellow circles). Prioritization is rarely used and little is written on the matter. Buffington has done some work on the decomposition of the control law and the prioritization of certain control law parts [10]. The prioritization in the research performed in this paper was of the actual objectives, or moments, of the airframe.

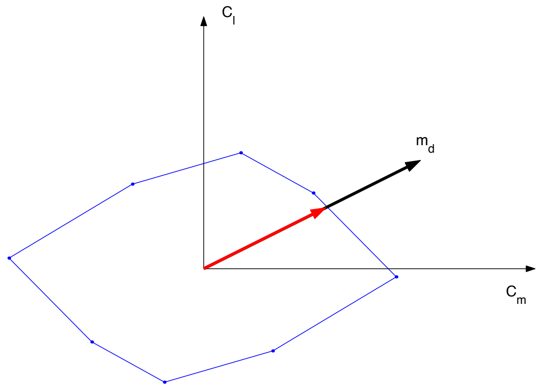

Figure 1.3: Method of scaling control solution vector to preserve moment direction

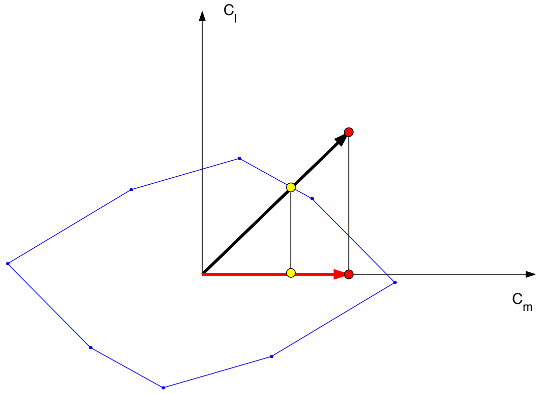

Figure 1.4: Method of prioritizing the pitch axis

1.5 Research Objectives

It was the purpose of this research to provide an insight into the problem of unattainable moments. A method of pitch prioritization was investigated to allocate controls in the case of an unattainable moment and compared with the method of scaling the moment direction. Pitch prioritization allocates controls while preserving the stability augmentation in the longitudinal axis of an unstable airframe. An F-18-like airframe was used as a testbed to perform the research and evaluate the performance of prioritization.

Prioritization of a single objective during situations that ask for unattainable moments was compared to the method of scaling the control vector and preserving the moment direction. The prioritization performed was of the pitch-moment. This means that for a set of desired moments that are unattainable, the desired rolling and yawing moments are sacrificed so that the amount of desired pitch-moment was attained to the greatest extent possible. This strategy is reasonable for the case in which the airframe is unstable in the longitudinal axis. For instance, relaxing the static margin of the aircraft is popular because it reduces the trim-drag of an aircraft. This causes the aircraft to be more unstable in the longitudinal axis, creating the need to produce the desired pitch-moments. Scaling down the necessary pitch moment may cause the aircraft to diverge.

Preserving a commanded direction has been shown to provide a small degradation of aircraft performance [2] in some cases. Buffington has made efforts to show that in other cases preserving direction of a solution can cause premature degradation. For this case, Buffington [10] attempts to develope a prioritization structure that provides more control over the performance degradation of an airframe for control deficient situations.

Buffington's research focuses on breaking a sliding mode control law into control partitions. He showed that it may be preferable to maintain direction and magnitude of one control part while altering the other to satisfy effector limits. His research showed improved large command performance compared to a direction preservation command limiting [10]. The research contained in this paper does not prioritize control law parts, it instead puts the highest priority on the pitch-moment, so that the desired longitudinal characteristics can be maintained.

All research done in this paper focused on the powered approach mode (PA) of the airframe during an aircraft carrier landing. Aggressive maneuvers were performed such that the full capabilities of the effectors were utilized and were sometimes exceeded. The two methods of dealing with unattainable moments, scaling and prioritization, were both used separately for the same maneuver. Results from each method were compared to determine the effect of each on the flying qualities and performance of the aircraft during these aggressive maneuvers.

Chapter 2

Control Allocation

2.1 Introduction

Before the days of redundant control effectors and complex control laws the control allocation problem was simply dealt with by the use of mechanical linkages between the pilot's stick and pedals, and the control surfaces. Control systems since then have increased in complexity such as using an aileron/rudder interconnect to offset the adverse yaw effect during a turn. Even more complexity comes about when additional controls are developed and added to the airframe in an attempt to provide more maneuverability. Redundancy in aircraft controls has significantly altered the complexity of the control allocation problem. Several algorithms have been developed over the years to solve this problem. This chapter provides a background to the methods for allocating controls, and focuses on the one used in this research.

2.2 Background

Less complex algorithms such as generalized inverses have been used successfully in the past. The most popular of these control allocation algorithms is the Minimum Norm solution. This solution arranges the controls to produce the desired objectives while minimizing control energy. This method simply uses the pseudo-inverse solution to minimize the 2-norm of the control vector solution. There are several well-known ways to derive the pseudo-inverse solution that may be found in any text dealing with the underdetermined systems of linear equations. Another popular routine is the method of Daisy Chaining. This method breaks the control set up into multiple subsets of controls. One subset of controls is used to generate as much as possible of the desired objectives, with the second set being used if the first set is unable to acheive the desired objectives. Each successive grouping of controls is used only when the previous group is unable to acheive the desired objectives. This method uses a generalized inverse to solve for the control solution, much like the minimum norm solution. This method is described in Bordignon [4].

It has been shown that these generalized inverse routines, including minimum norm solutions and daisy chaining, cannot guarantee admissible control solutions for all attainable moments [4]. The methods described as optimal solutions, however, are able to provide a unique admissible solution for all attainable moments. There are three methods of optimal allocation algorithms in use today. They are the Cascading Generalized Inverses, Facet Seraching Algorithm, and the Bisecting Edge Searching Algorithm. Each of these methods is described below.

2.3 Cascading Generalized Inverses

One method called Cascading Generalized Inverse (CGI) was developed by Bordignon [4]. This method uses the idea that if a generalized inverse commands a solution in which one or more controls exceed a limit, so long as not all are saturated, the controls are set at the limits, and the rest of the controls are redistributed to achieve the desired objectives. In other words, if a control is commanded past its limit, it is set at the limit, and the effects of the control at saturation are subtracted from the desired objectives. The problem is solved again for the new desired objective without the saturated control. The method is repeated every time a control position is exceeded. If all controls become saturated, the moment is unattainable using this method. A derivation and explanation of the method is found in [4].

2.4 Facet Search

The allocation method used in this research is based on the original research by Durham [1,2] and later refined by Bordignon and Durham [3, 4]. This method of control allocation takes the control effectiveness matrix **B**, the set of admissible controls, and some desired objective

m_d, and determines the intersection of m_d with the AMS by checking each facet. After finding the correct facet the controls that generate the point on the facet are calculated [7]. Remember that a facet is defined by two controls free to vary, with the remaining controls at limits. The solution corresponds to a unique control vector given the mapping between control space and moment space. If the desired objective does not lie on the boundary of the AMS, meaning that the objective is inside the AMS, the solution is scaled back appropriately. The description to this direct allocation solution can be found in Scalera [7].

The one problem that has always been associated with the Facet Searching algorithm is the computational complexity. Section 2.5 describes a method designed to greatly reduce the complexity of the Facet Searching algorithm and provide an optimal control allocation scheme with the capabilities of real time implementation.

2.5 Bisecting Edge Searching Algorithm

The method described above is more or less a brute force method of determining the intersection of m_d with the surface of the AMS. The method involved looking at a single facet defined by a pair of controls and determining whether the desired moment direction pointed toward this facet. However, there is no guarantee that the intersection would be found before the last facet was tested. The potential of searching the entire AMS meant that the number of floating point operations and computation time could be very high with the addition of control effectors. With aircraft becoming more advanced with more control surfaces, a computationally efficient method of determining the correct facet is needed. Work by Durham [6] introduces this computationally efficient control allocation method called Bisecting Edge Searching Algorithm (BESA). Scalera [7] shows that the computational complexity of this method increases linearly with m whereas the brute force method of facet searching was on the order of m^2. The theory of the algorithm is not the point of this paper, and is only briefly described here. The reader is referred to Durham and Scalera [6, 7] for the theory behind the algorithm and a comparison of the performance with other allocators.

2.5.1 Two-Dimensional Problem

The two dimensional problem is introduced as part of the solution to the three dimensional problem. That is, the three dimensional problem can be solved by systematically repeated solutions of the two dimensional projections of the three dimensional problem.

For this problem consider m_d to be a set of two objectives. **B** will then consist of two rows, with each element describing the effect each control has on the objectives. Figure 2.1 depicts an attainable moment subset, AMS, created from a four-control two-objective problem. Being able to describe the edges of the attainable moment subset is a big step towards the solution of the problem, since each edge corresponds to a single control. The vertex with the maximum x-component is identified on the AMS. This vertex is shown in figure 2.1 as a star. To find the first edge, the vertex that lies next to the vertex with the maximum x-component must be found. To do this, the axes of the problem are rotated. At a specific rotation angle a new vertex with the maximum x-coordinate is identified. As the angle is increased and each subsequent vertex is found, each edge of the AMS can be defined. These edges are associated with a single control, providing the necessary information to construct the AMS [6].

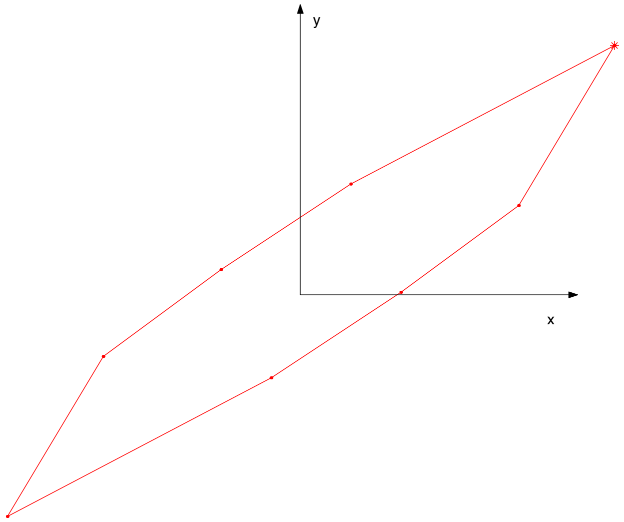

Figure 2.1: Two-dimensional AMS. Star represents the starting vertex of the AMS. x and y axes correspond to arbitrary moment directions.

The computational complexity of this method can be decreased if there was a way to solve the problem by only defining part of the AMS where the solution of the problem lies. To begin this solution the desired objective, m_d, is lined up with the x-axis direction by rotating the problem. Once rotated, the problem is solved by determining which edge the x-axis intersects. This edge is identified by a change in sign of the y-component as the verticies are traversed from one to the next. If the y-component of the original vertex is positive, as in figure 2.1, the edges are traversed clockwise until the sign of the y-component of the next vertex changes.

2.5.2 Three-Dimensional Problem

For the three-dimensional problem, consider m_d to consist of three objectives, causing the control effectiveness matrix to have three rows. Figure 2.2 is a random attainable moment subset orientated to see the x-axis passing through the AMS. This problem is transformed, as in the two-dimensional problem, so that m_d is along the $+x$-axis. Once this transformation is performed, the AMS is viewed looking down the $+z$-axis. From this view point the problem resembles the two-dimensional problem and is solved in the same way. Note that figure 2.3 is identical to figure 2.1 except for the facets associated with the third dimension. An edge that crosses the $+x$-axis is determined by solving the two-dimensional problem, however, this edge is not necessarily part of the solution facet, nor is it sufficient to define a whole facet.

There is an angle where the AMS can be rotated about the $+x$-axis such that the intersection of m_d, which is aligned with the $+x$-axis, with the AMS will lie exactly on a limb of the AMS as viewed from the $+z$-axis. This means that the plane of the facet that contains the intersection will be parallel to the z-axis. Figure 2.3 shows an AMS oriented such that the intersection lies directly on a limb. Small perturbations in this angle will identify two edges that define the facet that contains the intersection. By rotating the AMS with a small angle in the positive direction, an edge of the facet that contains the intersection will constitute the edge of the two dimensional AMS figure. This edge is solved for with the two dimensional problem and is "behind" the intersection of the $+x$-axis with the AMS because it has a negative z component. A small rotation in the opposite direction will reveal the edge that is in "front" of the intersection, having a positive z component. These edges are identified exactly as was done for the two dimensional problem [6].

The rotation angle can not be calculated analytically, however, the angle is not needed, just the edges that constitute the solution facet. The solution starts by identifying the edge of the two dimensional figure that crosses the x-axis for the arbitrarily oriented AMS. The z-component of this edge is calculated. The AMS is then rotated a predefined angle about the x-axis and another edge is identified. This method is repeated until a sign change occurs in the z component of the identified edge. The last two edges identified, one "behind" and one in "front", could possibly define the solution facet. If the edges do define a single facet, the solution is found. If the edges do not lie on the same facet, the rotation direction of the AMS is reversed and the predefined angle is reduced. This process is repeated until a solution facet is found.

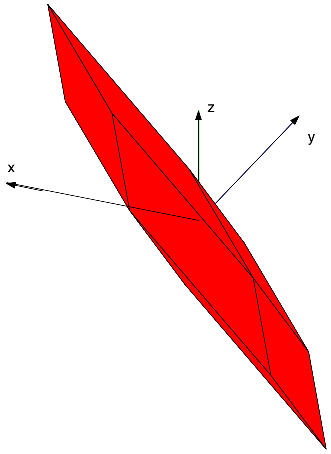

Figure 2.2: 3 dimensional AMS orientated such that the x-axis (desired objective) is viewed coming out of a specific facet.

The allocation method is limited to a maximum number of bisections, or switches in rotational direction. If an answer is not produced after these bisections an estimator is used which linearly interpolates the answer using the y and z components of the vertices that define the last two edges that were found. Scalera [7] presents an error analysis for the method of estimation being used.

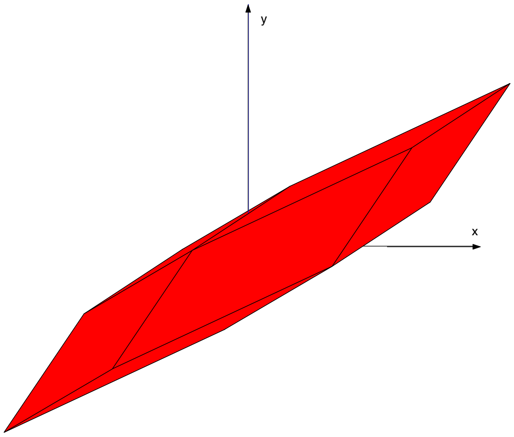

Figure 2.3: Three-dimensional AMS rotated such that the view point is looking down the z-axis of the problem.

Chapter 3

Airframe Simulation and Implementation

3.1 Introduction

This chapter explains the F/A-18 testbed's dynamic-inversion control law that was used for this research and the method of coding used to implement the control allocator into the control law. The simulation environment is also introduced.

3.2 F/A-18 Testbed

The testbed airframe used in this research was based off of an F/A-18A airframe simulation. The greatest area in which the test-bed simulation differs from the original F/A-18A airframe is in the treatment of control effectors. There are essentially two sets of control effectors that the testbed airframe uses:

1. The original control effectors in the F/A-18 airframe model. These effectors were retained to make the simulation compatible with existing F/A-18 aircraft simulations. These controls are only used for initial trim and subsequent scheduling. The F/A-18 in PA mode schedules leading-edge flaps based on angle-of-attack. Trailing-edge flaps and ailerons are scheduled with dynamic pressure. The rudders are scheduled

differentially as a function of angle-of-attack, with full toe-in (30 degrees) for low angles-of-attack, and half-flare for high angles-of-attack. This rudder scheduling augments the longitudinal static stability for the carrier landing task [13].

2. A set of 20 control effectors with linear effectiveness. This control set provided a challenging 20 effector control allocation problem. Constant linear effectiveness was provided so that the differences between prioritization and scaling would not be obscured by nonlinearities.

The rationale for incorporating this second set of linear control effectors was to provide a constant, linear control effectiveness matrix with flexibility for future variations and modifications. This second set of linear control effectors contains 20 redundant control effectors to provide control power for attaining desired objectives. The effectiveness of these controls was based on the linearization of the original six control effectors. The original control set was linearized in a PA mode using a linearization method borrowed from Stevens and Lewis [12]. This method calculates Jacobian matricies for a set of non-linear state equations. The algorithm chooses sequentially smaller pertubations in the independent variable (single control effector) and compares three successive approximations to find the particular partial derivative. The final product of this algorithm produced a linear control effectiveness matrix for the original six-control set of the F/A-18.

To provide a control effectiveness matrix for the entire set of 20 control effectors, an additional 14 columns were added to the control effectiveness matrix for the six controls that were first linearized. These additional controls were scaled down from the original six columns based on their "virtual" position on the airframe. For instance, two left rudders were added to the airframe. The control effectivness for the newly added left rudder was determined by scaling the column corresponding to the original left rudder, making it a smaller surface compared to the original left rudder. The end result was a 3x20 control effectiveness matrix that produced linear aerodynamics for the testbed airframe.

This set of 20 virtual control effectors is referred to as the UGLOBAL array. The twenty controls with their description and limits are in table 3.1 along with the six non-linear controls. As described above, there are two sets of effectors that control this airframe. Only the set of six original controls were used for trimmed flight and control scheduling. Control scheduling will move the controls to a certain position based on aircraft states at a certain flight condition of the aircraft. For instance, at an angle-of-attack of 8.1 degrees, the rudders are toed in 0.5 degrees. In a trimmed flight condition the six controls will be set appropriately with all 20 UGLOBAL controls set to zero. Forces and moments are calculated from this set of six control effectors using a non-linear aerodynamic database for the F/A-18. The second set of control effectors, UGLOBAL, is allocated using the linear control effectiveness data previously obtained. This set of effectors is used to attain as much of the desired moments from the control law as possible.

The control system for this testbed was arranged such that it could incorporate the two distinct sets of controls described above and superimpose their aerodynamic effects.

The trim/schedule controls are based purely on an aerodynamic trim condition, or are driven by schedule from state variables such as angle-of-attack. The aerodynamic effect from the first set is added to the aerodynamics created by the UGLOBAL controls that produce the desired moment. Figure 3.1 is a block diagram of the logic explained above.

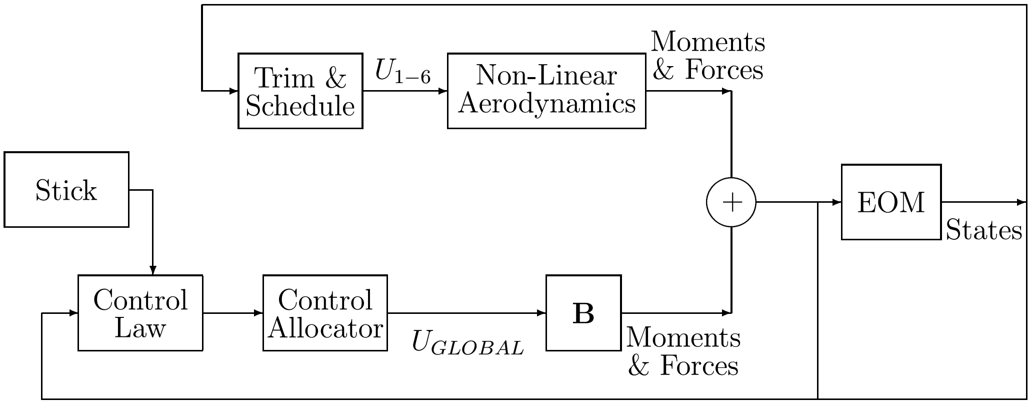

Figure 3.1: Block Diagram of the Testbed Airframe Simulation Structure

The original set of control effectors goes through the non-linear aerodynamic database before being added to the aerodynamics from the UGLOBAL controls which are calculated in the allocator with the linear control effectiveness matrix. Note that the control effectiveness matrix **B** in this case includes rows for the force coefficients as well as the moment coefficients. These partial derivatives for the force coefficients with respect to each control effector are found in the same manner as the partials for the moment coefficients.

Section 3.3 provides a complete description of each routine used in the simulation of the aircraft.

Table 3.1: Control Surfaces on the testbed airframe.

Control	Description	Deflection Limits
DHTL	Left Horizontal Tail	-24.0/10.5
DHTR	Right Horizontal Tail	-24.0/10.5
DAL	Left Aileron	-25.0/45.0
DAR	Right Aileron	-25.0/45.0
DRUDL	Left Rudder	-30.0/30.0
DRUDR	Right Rudder	-30.0/30.0
UGLOBAL(1)	Left Horizontal Tail	-24.0/10.5
UGLOBAL(2)	Right Horizontal Tail	-24.0/10.5
UGLOBAL(3)	Left Aileron	-25.0/45.0
UGLOBAL(4)	Right Aileron	-25.0/45.0
UGLOBAL(5)	Left Rudder	-30.0/30.0
UGLOBAL(6)	Right Rudder	-30.0/30.0
UGLOBAL(7)	Left Horizontal Tail	-24.0/10.5
UGLOBAL(8)	Right Horizontal Tail	-24.0/10.5
UGLOBAL(9)	Left Aileron	-25.0/45.0
UGLOBAL(10)	Right Aileron	-25.0/45.0
UGLOBAL(11)	Left Rudder	-30.0/30.0
UGLOBAL(12)	Right Rudder	-30.0/30.0
UGLOBAL(13)	Left Aileron	-25.0/45.0
UGLOBAL(14)	Right Aileron	-25.0/45.0
UGLOBAL(15)	Left Aileron	-25.0/45.0
UGLOBAL(16)	Right Aileron	-25.0/45.0
UGLOBAL(17)	Left Horizontal Tail	-24.0/10.5
UGLOBAL(18)	Right Horizontal Tail	-24.0/10.5
UGLOBAL(19)	Left Rudder	-30.0/30.0
UGLOBAL(20)	Right Rudder	-30.0/30.0

3.3 Airframe Simulation

There are six files that were used in the simulation of the airframe: AERO.F, AEROPA.F, CONTROL.F, CONSTANTS.F, ENGINE.F, AND ALLOC.F. This section explains the purpose of each routine. Some of these subroutines are found in Appendix A.

3.3.1 Aero.f

AERO.F is an executive subroutine that calls subroutines based on the flight phase of the aircraft. In this specific case AERO.F calls AEROPA.F to calculate the aerodynamics of the scheduled/trimmed flight condition. The code then combines the aerodynamics from the non-linear scheduled/trimmed flight condition with the aerodynamics from the UGLOBAL array to produce the total aerodynamic forces and moments.

3.3.2 Aeropa.f

AEROPA.F is taken from the F/A-18 simulation and modified slightly to mesh with AERO.F and to include proper control scheduling. These modifications are in Appendix A. This is the only code that gives the airframe F/A-18 like characteristics.

3.3.3 Control.f

Stick and rudder pedal commands are taken as inputs and converted into an angle-of-attack command α_{cmd}, sideslip command β_{cmd}, and a roll rate command p_{cmd}. These commands are input to a simple dynamic inversion control law that generates desired moments for the control allocation subroutine. First, α_{cmd} and β_{cmd} are converted to desired accelerations $\dot{w}_{des}$ and $\dot{v}_{des}$:

$$\begin{aligned}
w_{cmd} &= u \tan \alpha_{cmd} \\
\dot{w}_{des} &= \lambda_w \left(w - w_{cmd} \right) \\
v_{cmd} &= V \sin \beta_{cmd} \\
\dot{v}_{des} &= \lambda_v \left(v - v_{cmd} \right)
\end{aligned}$$

where λ_w and λ_v are control gains. Next, $\dot{w}_{des}$ and $\dot{v}_{des}$ are applied to inversions of the body-axis force equations (treating q and r as controls):

$$\begin{aligned} q_{cmd} &= \tfrac{\dot{w}_{des}+pv-g\cos\theta\cos\phi-Z/m}{u} \\ r_{cmd} &= \tfrac{-\dot{v}_{des}+pw+g\cos\theta\sin\phi+Y/m}{u} \end{aligned}$$

These two inversions are made as perfect as possible by using actual aircraft states, and the last calculated values of the body-axis forces Y and Z from the aerodynamic calculations. First-order responses are specified for the desired angular accelerations,

$$\begin{aligned} \dot{p}_{des} &= \lambda_p\left(p-p_{cmd}\right) \\ \dot{q}_{des} &= \lambda_q\left(q-q_{cmd}\right) \\ \dot{r}_{des} &= \lambda_r\left(r-r_{cmd}\right) \end{aligned}$$

where $\lambda_p, \lambda_q, \lambda_r$ are all control gains. Finally, the desired body-axis moments, required to obtain the desired accelerations, are calculated from inversions of the body-axis moment equations:

$$\begin{aligned} C^c_{\ell_{des}} &= -C^a_\ell + \tfrac{I_{xx}\dot{p}_{des}-I_{xz}\dot{r}_{des}+(I_{zz}-I_{yy})qr-I_{xz}pq}{\bar{q}Sb} \\ C^c_{m_{des}} &= -C^a_m + \tfrac{I_{yy}\dot{q}_{des}+(I_{xx}-I_{zz})pr+I_{xz}\left(p^2-r^2\right)}{\bar{q}S\bar{c}} \\ C^c_{n_{des}} &= -C^a_n + \tfrac{-I_{xz}\dot{p}_{des}+I_{zz}\dot{r}_{des}+(I_{yy}-I_{xx})pq+I_{xz}qr}{\bar{q}Sb} \end{aligned}$$

The moment coefficients C^a_ℓ, C^a_m, and C^a_n are the last calculated values of the body-axis moments. Since control-generated moments are superimposed on these values, they are the moments generated by the first control set (bare-airframe plus scheduled control deflections). The attained moments are calculated next using the control deflections from the last time step for comparison purposes with the desired moments. A flag is set to determine whether the airframe will use pitch prioritization or control scaling to deal with the case of unattainable moments. The desired moments, along with the required inputs, are input either to the scaling allocator routine or the prioritization allocating routine, depending on the allocator flag, to produce the required control deflections.

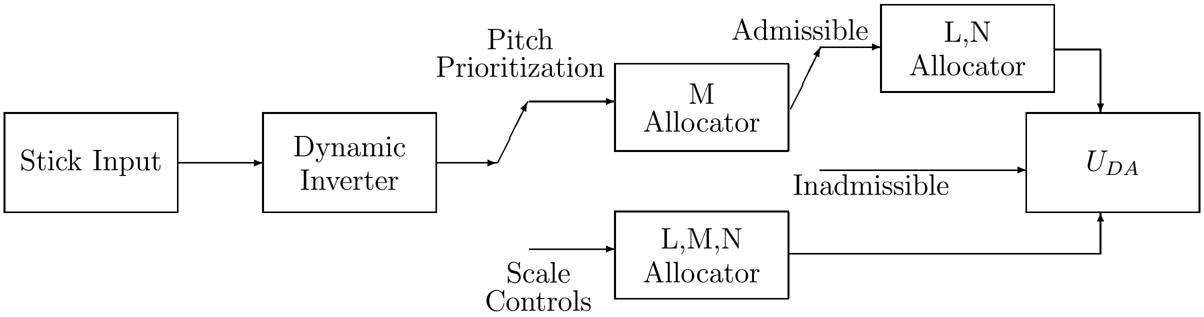

Figure 3.2: Control Loop Block Diagram

If the method of dealing with unattainable moments is to scale the control solution the BESA allocator is run one time with the vector of desired moments. The allocator routine is designed to scale the solution back if it is determined to be inadmissible. This scaled solution is then returned from the allocator to the airframe.

Using pitch prioritization to deal with unattainable moments requires the allocator to run two separate times. Since the pitch moment is the objective that is being prioritized the allocator is run with only the desired pitching moment while roll and yaw are set to zero. The method attempts to satisfy the pitching moment requirement using as much control power as needed. If the solution is admissible the rolling and yawing moments are treated as a separate problem and the origin is shifted to the desired pitching moment. The remaining control availability is used to obtain as much of the desired rolling and yawing moment as possible. This solution and the solution that produces the pitching moment are added to form the total control solution. However, if the pitching moment is unattainable, the solution is scaled and rolling and yawing moment are ignored. Chapter 4 provides a description of how each method of dealing with unattainable moments works, and how it is implemented into the code. Figure 3.2 shows a block diagram of the control structure of this testbed airframe.

3.3.4 Constants.f

This section of code sets the model specific contants.

3.3.5 Engine.f

The engine model is from an existing F/A-18 simulation flying in PA mode.

3.3.6 Alloc.f

This code is the BESA control allocator that produces required control deflections for desired moments. The BESA method was explained in Chapter 2.

3.4 Simulation Environment

Naval Air Systems Command provided the original F/A-18 simulation code in FORTRAN that was implemented on a Silicon Graphics Origin 2000^{TM} Deskside System with two CPU's running at 180 MHz with 256 MB of RAM and a 4 GB disk. The whole airframe was made compatible with CASTLE (Control Analysis and Simulation Test Loop Environment). CASTLE is a 6-degree-of-freedom non-linear aircraft simulation architecture developed by Naval Air Warfare Center's Manned Flight Simulator branch (MFS) [11]. The CASTLE structure is implemented on the Virgina Tech modified 2F122A motion-based flight simulator.

All simulations are run at 100 Hz in either a batch mode or a real-time piloted simulation. CASTLE has the ability in batch mode to input canned maneuvers or a previously recorded time history of data using the built-in function MANGEN. The real-time simulations utilize a pilot-in-the-loop that performs a specific defined task. The visual scene is calligraphic depicting a dusk or night environment. The visual database includes scenes for a carrier landing approach, and a naval air station. The stick feel is produced using an electronic control loader and can be modified through software to produce a realistic feel of any actual aircraft stick. The stick being used is modeled on the characteristics of the F/A-18 E/F.

3.5 Airframe Validity

The airframe in this research was modeled after the F-18A airframe for ease of implementation on other users computers that already had the original F-18 airframe. The modifications that were made were simply added subroutines to transform the airframe as needed. The testbed airframe is not meant to resemble the actual F-18 airframe simulation, but is meant to provide an airframe capable of producing reasonable flight characteristics.

Chapter 4

Pitch Prioritization

4.1 Introduction

The method of prioritization has rarely been used for problems such as unattainable moments that lead to inadmissible control solutions. The use of pitch prioritization provides one method of dealing with unattainable moment commands. As previously stated, pitch axis prioritization was chosen because of the popularity of relaxing the static margin of an aircraft to reduce trim drag. To stabilize the airframe the control law requires a certain amount of pitching moment. Prioritization the pitching-moment requirement will ensure that the maximum amount of pitch can be attained during any maneuver. Scaling back the control solution to preserve moment direction may not produce the required amount of pitch moment to keep the airframe stable.

This chapter explains how pitch prioritization is performed, and how it is implemented into the airframe model. The attainable moment subset was also sized to provide cases in which desired moments would be unattainable so that the method of pitch prioritization could be used.

4.2 Sizing the Attainable Moment Subset

Control effectiveness was determined by the linearization of the F/A-18 aerodynamic database described in Chapter 3. The controls that were added to the testbed airframe were given the same control limits and a reduced control effectiveness, for the most part on the same order as the original six controls. This yielded a large attainable moment subset, to the point where the airplane had excessive control power compared to the normal amount of control power for an F/A-18. The options for producing a **B** matrix which yielded more reasonable control power were either adjusting the effectiveness of each control one by one, or separately scaling each moment axis of the matrix. Considering that this aircraft is only intended to provide a testbed airframe and not be characteristic of an actual aircraft, each moment axis was scaled appropriately to provide a control effectiveness matrix **B** that was useful in this research.

The scaling factors for each axis are unimportant, just the fact that there is enough control power to stabilize and control the airframe, but not so much that the airframe is able to attain any desired moment during aggressive maneuvers. Figure 4.1 shows the AMS calculated from the original unscaled **B** matrix.

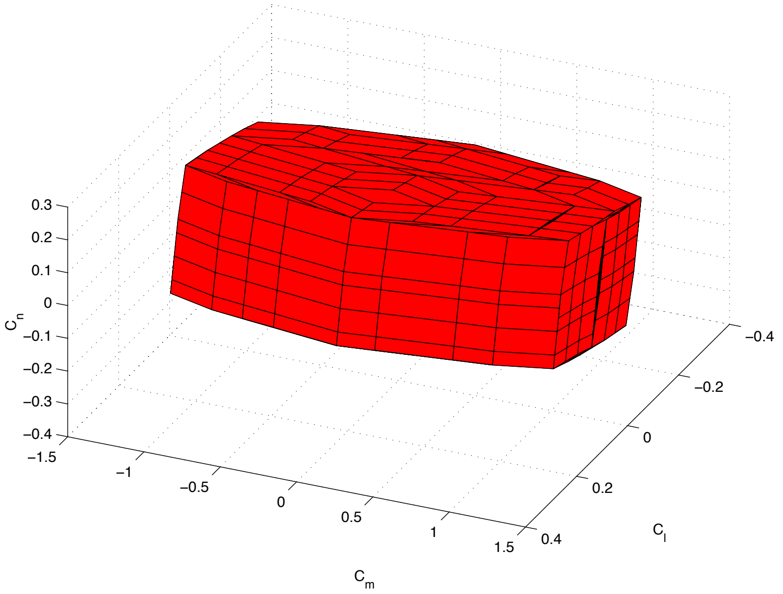

Figure 4.1: AMS described by the original Control Effectivness Matrix **B**.

The control effectiveness matrix was scaled appropriately by flying an aggressive offset approach task that used the full capabilities of all the control effectors. The time histories of the desired moments were recorded and compared to the AMS. Figure 4.2 clearly shows that the original control effectiveness matrix provided an abundance of control power with respect to the desired moments of the task. The blue trace represents the moment time histories of the maneuver plotted inside the full AMS. Note that the axes in figure 4.2 are not square. The shape of the AMS would be stretched in the C_m direction if the axes were equal.

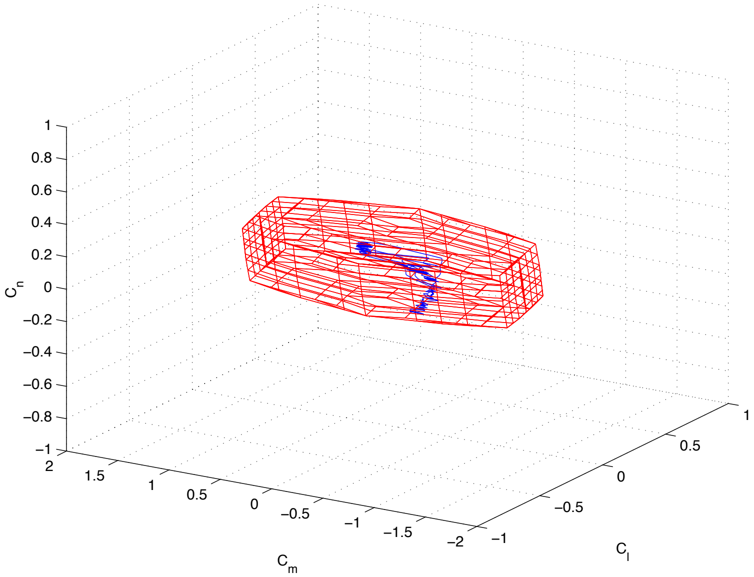

Figure 4.2: AMS described by the original Control Effectivness Matrix **B** plotted with the desired moments of the offset approach task.

With fast and aggressive maneuvers, the desired moments in roll and yaw should be able to exceed the AMS so that pitch prioritization can be compared against scaling the control solution. Using this desired moment time history, each axis of the AMS was scaled to provide an appropriate amount of control power such that some portion of the desired moments were unattainable. The final AMS for the testbed airframe is shown in figure 4.3 with the same time history of desired moments that were plotted in figure 4.2. Again, the axes in figure 4.3 are not square, making the C_l and C_n axes appear fatter than they actually are. Note that this scale is used in the rest of the figures in this chapter. The final scaling was 25% in the lateral and directional axes, and 50% in the pitch axis of the original AMS. Figure 4.4 shows the original AMS with the scaled AMS inside. This new control effectiveness matrix provided the opportunity to use pitch prioritization to allocate controls for cases of unattainable moments. This new attainable moment subset is referred to as the AMS for the remainder of the paper unless otherwise noted.

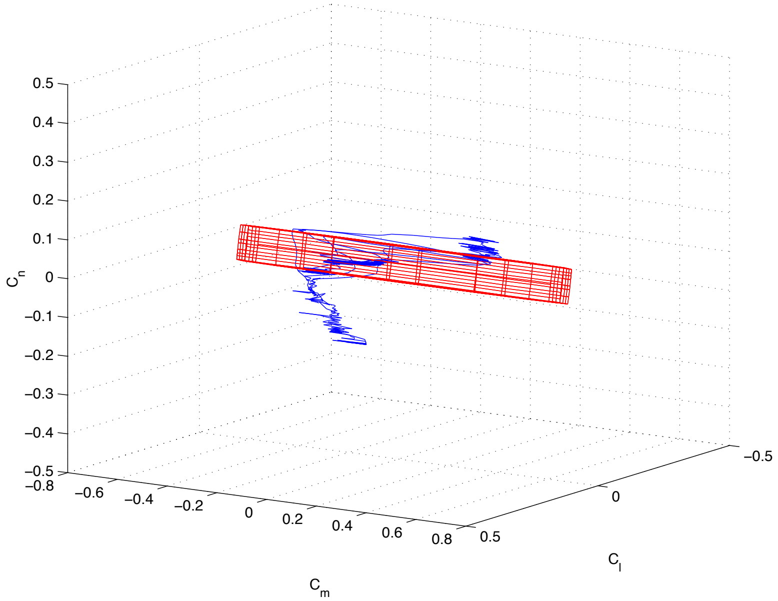

Figure 4.3: AMS described by the scaled Control Effectivness Matrix **B** plotted with the desired moments of the offset approach task.

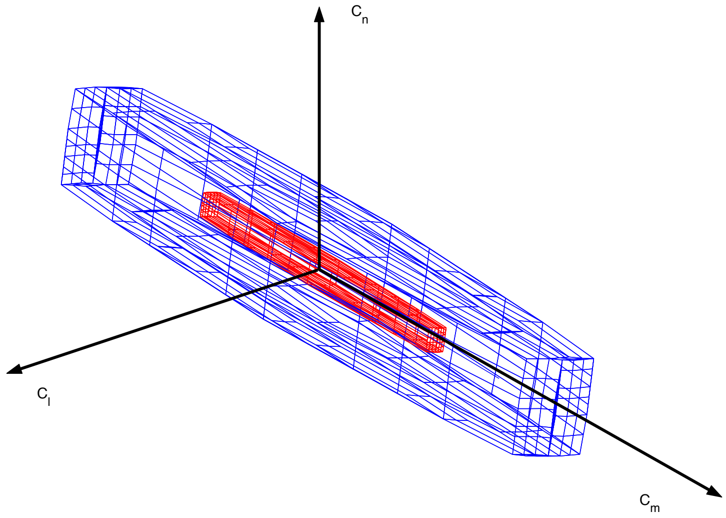

Figure 4.4: AMS described by the original Control Effectivness Matrix **B** with Scaled Final **B** used in the Research.

4.3 Method of Pitch Prioritization and Moment Direction Preservation

As explained before, pitch prioritization is used to preserve as much of the pitch component of the desired moments as possible. Scaling the control solution vector preserves the moment direction while decreasing the magnitude. Figure 4.5 is the AMS in figure 4.3 as viewed down the $+x$-axis, corresponding to roll (C_l). With this view one can see that some of the desired moments are outside the AMS.

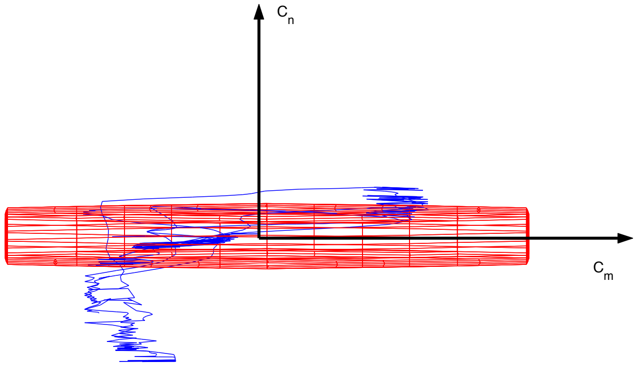

Figure 4.5: AMS as viewed down the $+x$-axis with the desired moments of the offset approach maneuver superimposed.

For such cases this research compared the two options of pitch prioritization and scaling the control solution which preserves the moment direction. Preserving the moment direction will simply decrease the magnitude of the moment vector, using the solution on the boundary of the AMS. Figure 4.6 shows how this solution is performed. The light green arrow indicates the desired moment for a specific time during the maneuver. Since this desired moment is unattainable, the control vector is scaled and the moment direction preserved. The dark green arrow indicates the solution on the boundary of the AMS. Note that the amount of pitching moment attained is less than half of the desired pitching moment.

Pitch prioritization in this case would attain all of the desired pitching moment by sacrificing some of the requirements for roll and yaw. Figure 4.7 shows the implementation of prioritization in the pitch axis. As before the light green arrow indicates the desired moment at a certain point in the maneuver. Prioritizing the pitch axis, the control effectors are able to provide the full amount of desired pitching moment. However, the controls are now limited

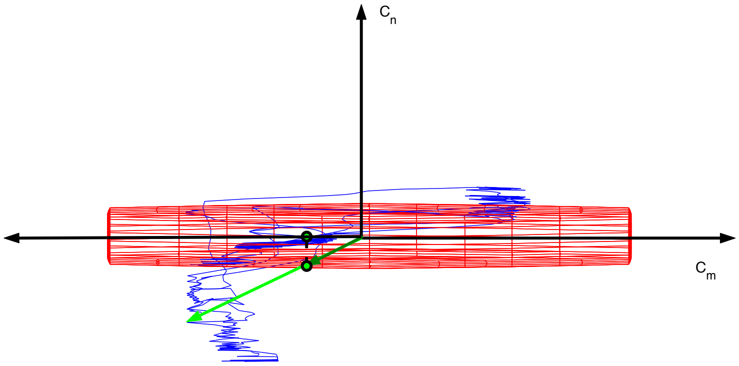

Figure 4.6: AMS as viewed down the $+x$-axis. The method of control vector scaling is implemented.

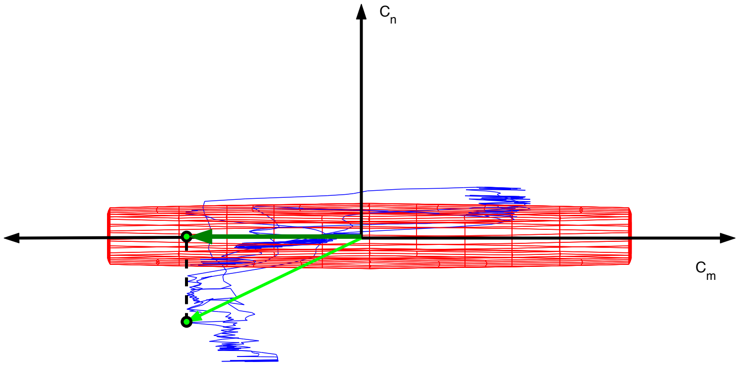

Figure 4.7: AMS as viewed down the $+x$-axis. The method of pitch prioritization is implemented.

in their capabilities to provide the desirable amount of roll and yaw.

Once the pitching moment is attained, the origin of the problem is recentered by adjusting the limits on the controls corresponding to the control vector solution from the pitch moment problem. This recentering of the problem will produce a different AMS since the control limits that define the m dimensional box in control space are different. The desired rolling and yawing moments are allocated and the control vector is added to the solution obtained from allocating the pitching moment. In this case, not all of the desired rolling and yawing moment is attainable, therefore the control solution for this secondary problem is scaled to preserve direction as described above.

Should the desired moment history include a desired pitching moment that is unattainable with the AMS, only the pitching moment is allocated and the controls are scaled to preserve direction. Consider the hypothetical situation in figure 4.8. The problem determines that the desired amount of pitching moment exceeds the maximum available pitching moment. The problem is solved by finding the solution for only the desired pitching moment and scaling back the control vector so that the solution is on the boundary of the AMS. For this case, the desired rolling and yawing moments are ignored completely.

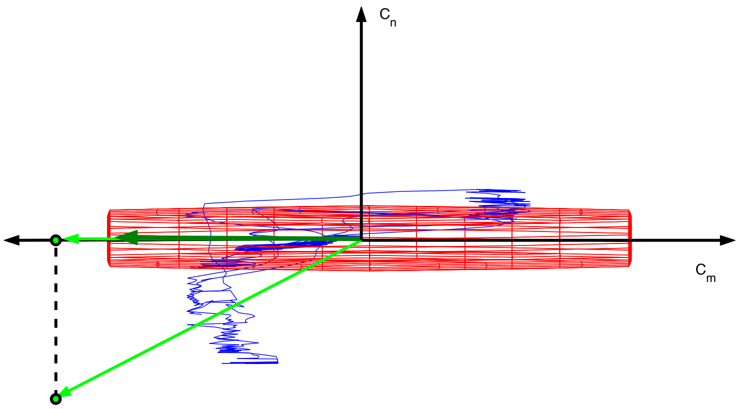

Figure 4.8: AMS as viewed down the $+x$-axis. The method of pitch prioritization is implemented during the case where control power is insufficient for attaining the desired amount of pitching moment.

Chapter 5

Representative maneuver

5.1 Background

maneuvers that cause the control law to require unattainable moments and require the majority of the controls to be saturated are used to investigate the comparison between the two methods of dealing with unattainable moments. An important aspect of the results in this research was how the airplane felt to the pilot in addition to the hard data that was plotted. One evaluation pilot was used to fly all the data represented in this paper. Several flights were flown by other pilots, however those results are not included here. The test pilot has a background in Navy fighters, carrier suitability, and flight testing. He served as a test pilot instructor at the Navy Test Pilot School at Patuxent River Naval Air Station.

5.2 Offset Carrier Approach maneuver

Aerodynamic control effectiveness reduces with a decrease in the dynamic pressure. As a result, an aircraft requires larger control deflections to produce the desired amount of force and moment. Since the dynamic pressure is a function of velocity, a task in which the aircraft is at a low velocity puts the aircraft in a high control deflection maneuver. An offset powered approach to an aircraft carrier was chosen for evaluation in this research since it provided the low dynamic pressure flight condition as well as large stick inputs and a high pilot workload.

5.2.1 Offset Carrier Approach maneuver: Description

For this maneuver the static margin of the airframe was relaxed so that the airframe was unstable in the longitudinal axis. The control law kept the airframe stable during normal flight with small stick inputs. However, with large stick inputs and large desired moments, the control effectors are saturated, or close to limits, and there is not enough control power to stablize the longitudinal axis and the airframe will depart during aggressive maneuvers.

The pilot was to fly a constant glide-slope approach which was parallel to the desired approach. The approach was always offset horizontally from the desired glide-slope with frequent vertical offsets. The pilot then realigned the airframe with the glide-slope at a predetermined distance from the aircraft carrier using an aggresive stick maneuver resulting in unattainable moment requirements.

Figure 5.1 contains plots of the path of the airframe during the offset landing approach. In this case the airframe was offset high and left. The correction is applied at -4000 ft in the x direction (xcgloc). A sharp right turn correction is easily seen, along with the increase in the rate of descent depicted by the larger slope in the altitude versus x location plot.

Different offset approaches were used in the evaluation of the pitch prioritization method. Two approaches were selected for the evaluation of pitch prioritization based on several preparatory flights. These two offsets were chosen because they required large moment requirements to reacquire the desired approach. A high left offset was chosen over a low left offset so that the pilot would not lose sight of the carrier during a hard nose-up correction. Also, pilot comments suggested that an aggressive nose-down correction was more comfortable than nose-up, since low speed nose-up corrections place the aircraft near stall.

- Offset Left - Airframe was offset to the left of centerline eight degrees and required more lateral maneuvering than pitch to correct the offset. The approach was offset eight degrees to the left of the centerline. The pilot began the task 3/4 of a statute mile from the carrier.

- Offset High Left - This maneuver required a greater amount of correction in the longitudinal axis, as well as keeping approximately the same amount of lateral correction as the left offset approach. The approach was offset eight degrees left of centerline and three degrees above the desired glide-slope. The pilot began the maneuver at the same point.

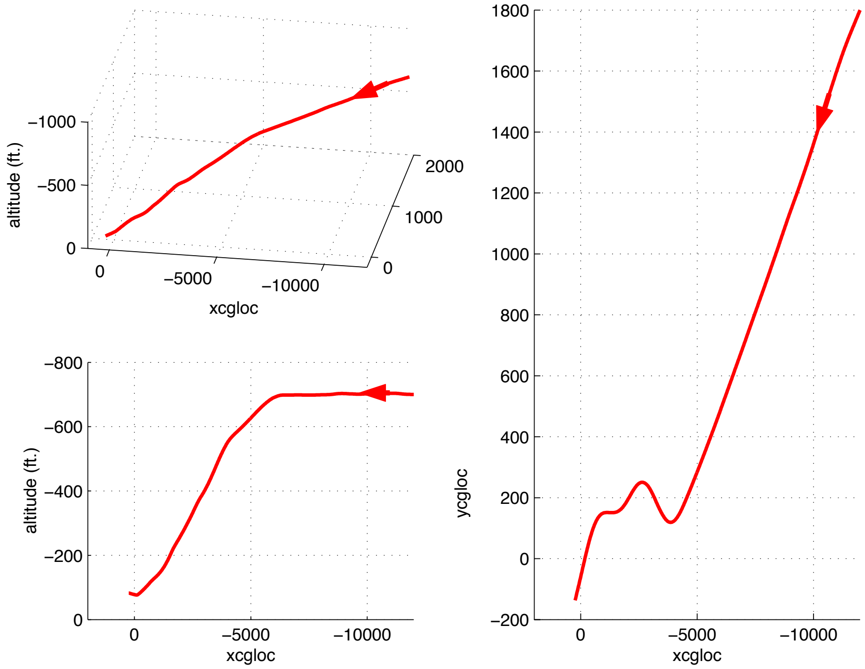

Figure 5.1: Path of the airframe during a high left offset carrier landing approach.

In order to ensure that the pilot flew the same offset each time, a cockpit Instrument Landing System (ILS) was used. The ILS consists of a horizontal and vertical line (needles) that tracks the glide-slope indicating the position of the airframe relative to the glide-slope and centerline. Offsetting the approach with the desired glide-slope was then accomplished by setting the needles to track a path offset from the desired approach. The pilot was required to follow the ILS until a certain distance from the carrier. At this point the pilot acquired visual contact with the carrier and performed the corrective maneuver to realign the airframe with the center line of the carrier deck and the glide-slope. To prevent the pilot from unintentionally correcting early, a cloud bank was added to the simulation such that the pilot had no choice but to follow the needles until the clouds broke and the pilot was able to see the carrier. This enabled the pilot to begin the corrective maneuvers at the same location as previous flights.

During these approaches the speed of the aircraft was held constant. Implementing a constant speed into the task allowed the pilot to concentrate on reacquiring the desired glide-slope without worrying about adjusting the throttles during the aggressive maneuvers. Flying at a constant speed meant that the dynamic pressure would not vary and change the effectiveness of the control effectors. Constant speed helped to make each task more repeatable, eliminating a variable from the comparison of the two methods.

The control power for the airframe during these runs was scaled as explained in chapter 4. The static margin was relaxed to the point where the airframe would feel the instability if the desired amount of pitching moment was not attained. In some cases the instability of the aircraft and the lack of pitch moment control power would cause the airframe to depart. The cg location of the airframe was set back to 68% of the wing chord as compared to the reference cg of 25%. This made the static margin of the aircraft 43% unstable.

Making the airframe this unstable was necessary to investigate the differences between pitch prioritization and scaling the control solution vector. With the airframe more stable it required less pitch-moment control power, hiding a lot of the qualities that pitch prioritization provided. The cg location was arrived at by several preparatory flights testing the stability of the airframe. With the cg so far aft, the differences between the two methods of dealing with unattainable moments were apparent. The following section describes how the airframe reacted during two specific offset approaches.

5.2.2 Offset Left Carrier Approach maneuver: Results

The following data represents two different offset left approaches flown from the same initial conditions. One approach implemented the method of scaling the moment direction while the other used the method of pitch prioritization. One way of comparing the two methods is to look at the saturation of controls and investigate how much control power is lost in each axis. The saturation of a control refers to the control being at a minimum or maximum limit. Figure 5.2 plots the saturation of each method for an offset left maneuver. The yellow hatching indicates when the controls are saturated. Note that in pitch prioritization mode the longitudinal axis requirement is never saturated, meaning that the airframe was able to achieve the total amount of pitching moment desired. Using as much control power as needed to acheive the desired amount of pitch moment meant that there would be a reduced amount of control authority to attain the desired roll and yaw moments. The third saturation plot in figure 5.2 shows where the lateral-directional solution is saturated. The area of saturation for the lateral-directional requirement occured when the pitch requirement was near saturation indicating that most of the controls had been used for pitching moment.

The offset left approach that was flown with the scaling method was saturated 13.6% of the time measured from the beginning of the maneuver when the pilot broke out of the clouds. This saturation corresponded to losing on average over the whole task 10.4% of the longitudinal control. Likewise, lateral-directional control power was also affected. Roll and yaw control was decreased by 11.6% and 12.2% respectively. These averages were calculated by determining the amount of desired moment that was not attained due to the scaling of the control vector solution and comparing it to the original desired moment. The determination of these averages assumes that each moment can be attained individually with the given amount of control power.

By prioritizing the pitch axis, the airframe was able to attain all of the desired pitch moment for the offset left approach. However, the lateral-directional requirements suffered. Roll and yaw requirements were saturated 17.4% of the flight time. This resulted in 19.8% of roll control power loss and 16.6% loss in yaw control. From comparing just the saturation amounts of each method we can see that pitch prioritization provides the ability to achieve the required pitch moments, only by sacrificing lateral-directional control authority.

Plotting the stick movement gives an indication of the pilot workload for each flight. Figure 5.3 shows the time histories of the longitudinal and lateral stick movement for the offset left approach. Rudder pedals were not used because the control law converted pedal inputs into pure sideslip (with no roll) which was not desired for the approach. Positive longitudinal and lateral stick corresponds to aft and right stick respectively. There is a slight difference in the longitudinal stick near the end of the tasks, however, it is difficult to conclude from this plot whether one method reduced the pilot workload or not. There are sharp spikes in the scaling method plot that indicate areas where the pilot felt the airframe make an unstable divergence due to the lack of control power, and corrected with rapid stick movements. The lateral stick plot shows that both methods required the full range of lateral stick with neither method requiring a more rapid stick motion. However, in prioritization mode the stick was held at a stop for a length of time which is a good indication that the pilot felt that the airframe was not producing enough lateral control power. This corresponds to the pilot comment that the airframe seemed sluggish towards the end of the task.

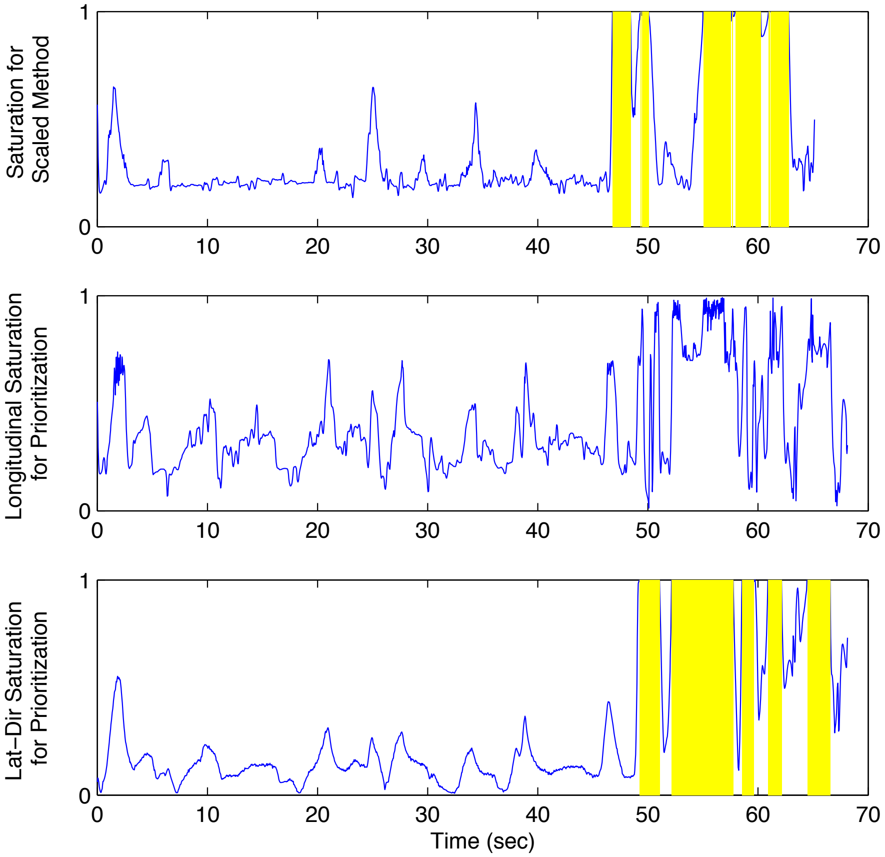

Figure 5.2: Comparison of the saturation of control surfaces between scaling the moment direction and prioritizing the pitch axis for an offset left maneuver.

Plotting the angle of attack time histories is a good indication of when the airframe starts to exhibit unstable characteristics. Figure 5.4 is the angle of attack time histories for each method. On speed angle-of-attack for the cg position used was around five degrees. It can be seen that with the scaling method the angle of attack makes abrupt peaks indicating an unstable divergence. These peaks occur at the same time the controls are saturated in figure 5.2. This divergence can be contributed to the lack of necessary control authority to stabilize the airframe longitudinally. In the prioritization method the airframe is able to attain the required pitching moments to stabilize itself while allowing the pilot to make the desired approach corrections.

Figure 5.5 is a time history of the aircraft pitch and roll rates plotted with the commanded rates. The first two plots are the actual pitch rate and commanded pitch rate for both the

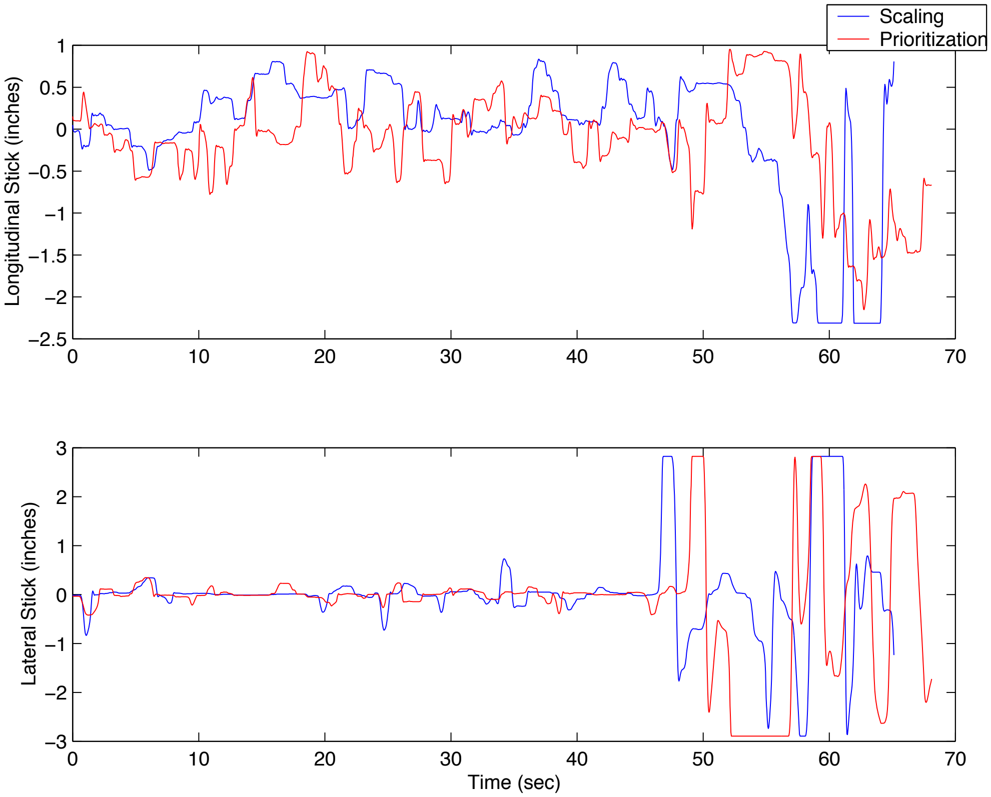

Figure 5.3: Longitudinal and lateral stick time histories for the offset left approach using both methods.

scaling and prioritization methods. During scaling it is evident that the airframe is not achieving the commanded pitch rates that the control law required, whereas prioritization did. In the scaling method plot the areas where actual and commanded pitch rates are opposite of each other are indicative instability, while the pitch prioritization plot shows commanded and actual rates in phase indicating stability. The scaled method was able to achieve most of the roll rate commanded unless there was a large pitch rate that dominated. However, it is clear that in prioritization after achieving everything in the longitudinal axis, the roll rate suffers, giving the airframe the sluggish feel in lateral control described earlier.

Figure 5.6 plots the desired moments for each case to compare the relative magnitude for each method. This figure represents the two different approaches with the same setup. One difference between the two methods is the amount of pitching moment that each demanded. The method of scaling the control solution vector required much more pitch moment than

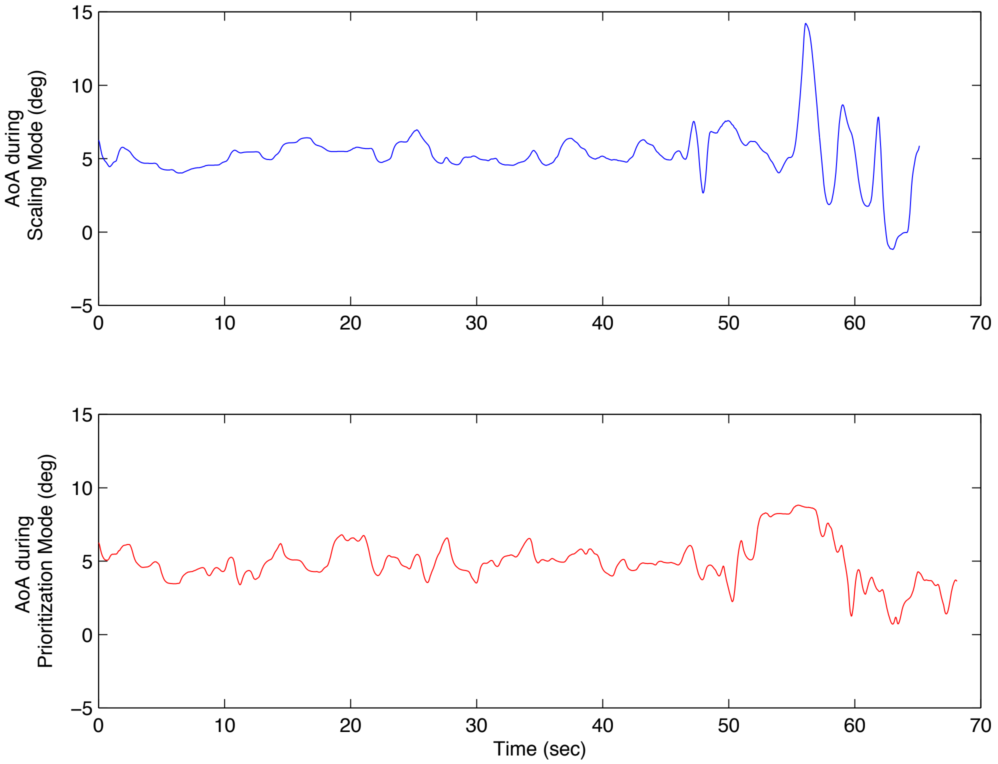

Figure 5.4: Angle of attack time histories for the offset left approach using both methods.

that of the prioritization method. This large desired pitching moment was caused by the airframe starting to diverge, requiring a large negative pitching moment to correct. Roll and yaw moments for each method were about the same magnitude.

Comparing the desired moment to the moment that the airframe actually attains provides a lot of insight into the performance of the two methods. Figure 5.7 plots desired moments with attained moments for the method of scaling the control solution vector. The points in the time history where the controls are saturated are obvious. These are the instances where the attained moment cannot achieve all of the desired moment. In the pitching moment plot there were two instances when the deficiency in pitching moment resulted in unacceptable dynamic responses. These were when the airframe started to diverge, requiring a certain amount of pitch control that the scaling method could not provide. This method scales back the control solution vector such that none of the desired moments are fully attained.

Figure 5.8 shows the prioritization method results and how it sacrifices roll and yaw requirements to attain the full amount of pitching moment. At the points where a large amount of moment is desired in all axes, the roll and yaw are sacrificed such that the airframe can attain all the desired pitching moment. This provided the airframe with enough pitch moment such that the pilot did not sense airframe instability. However, the lack of roll and yaw attained caused objectionable flying qualities during the offset approach.

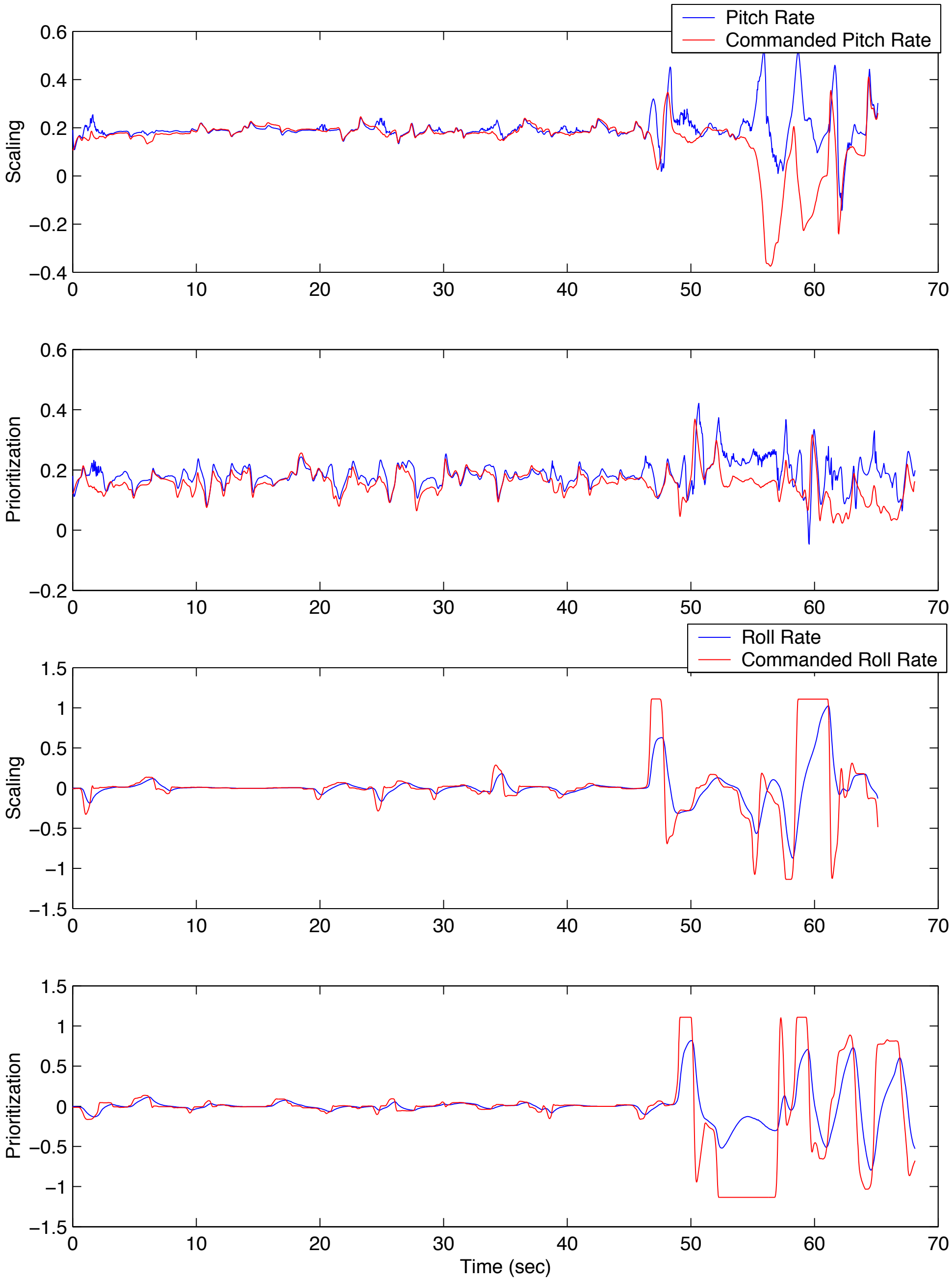

Figure 5.5: Aircraft pitch and roll rate time histories for the offset left approach using both methods.

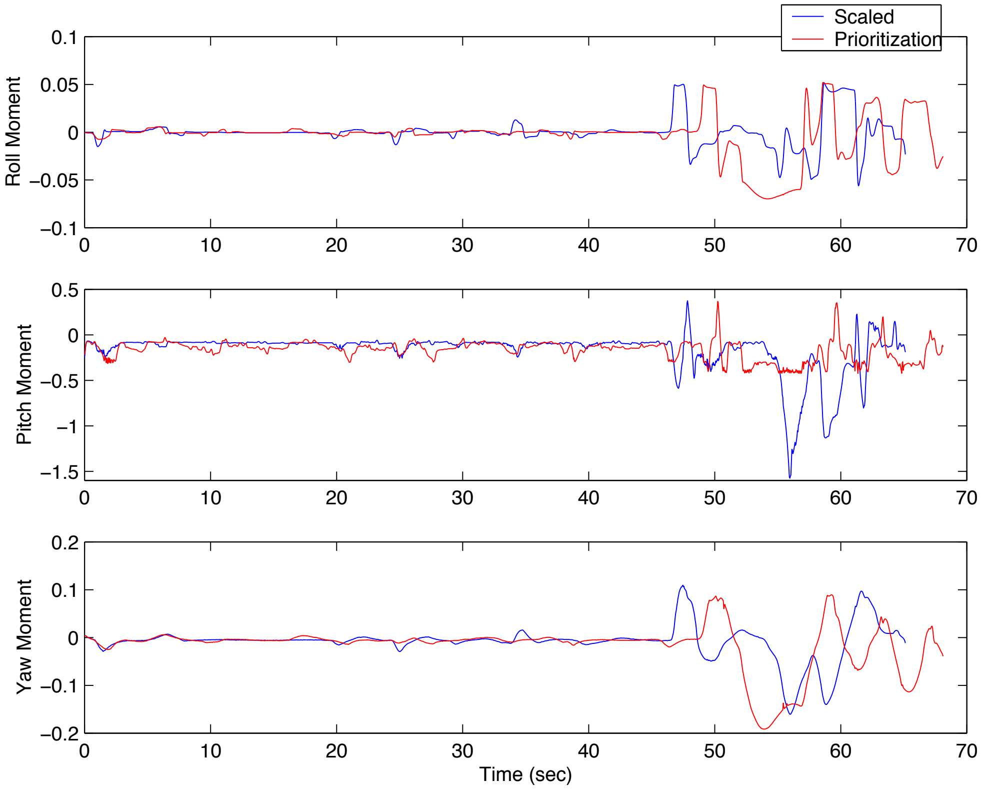

Figure 5.6: Desired moment time histories for the offset left approach using both methods.

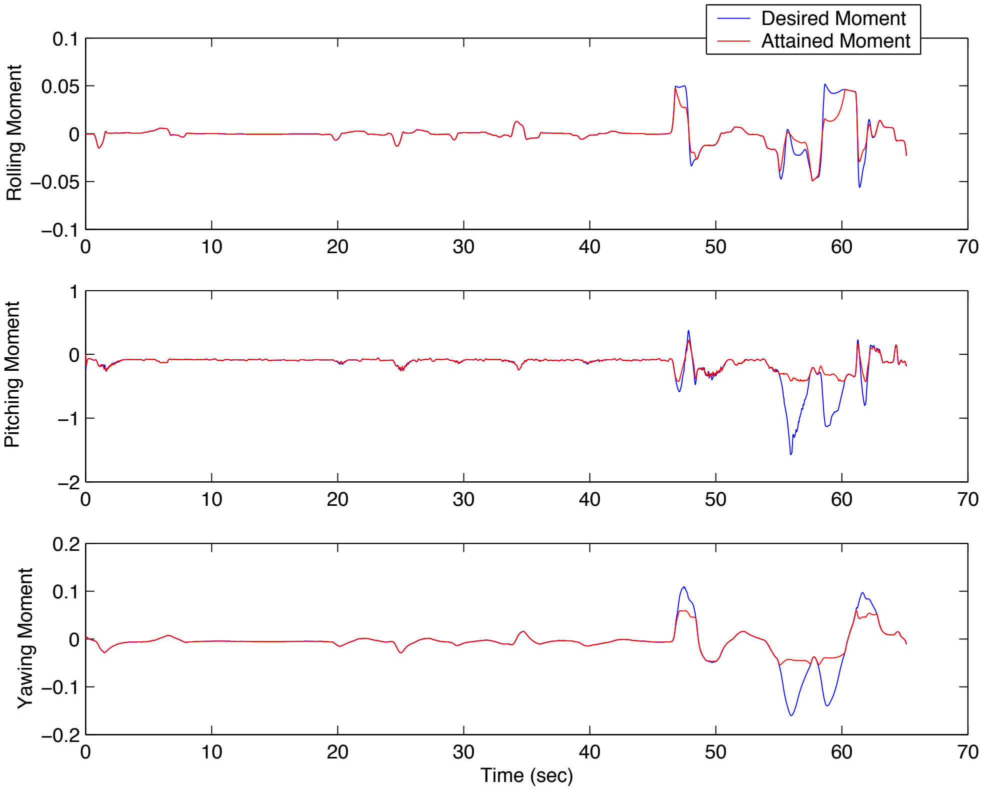

Figure 5.7: Desired moment time histories plotted with attained moments for the offset left approach using the method of scaling the moment direction.

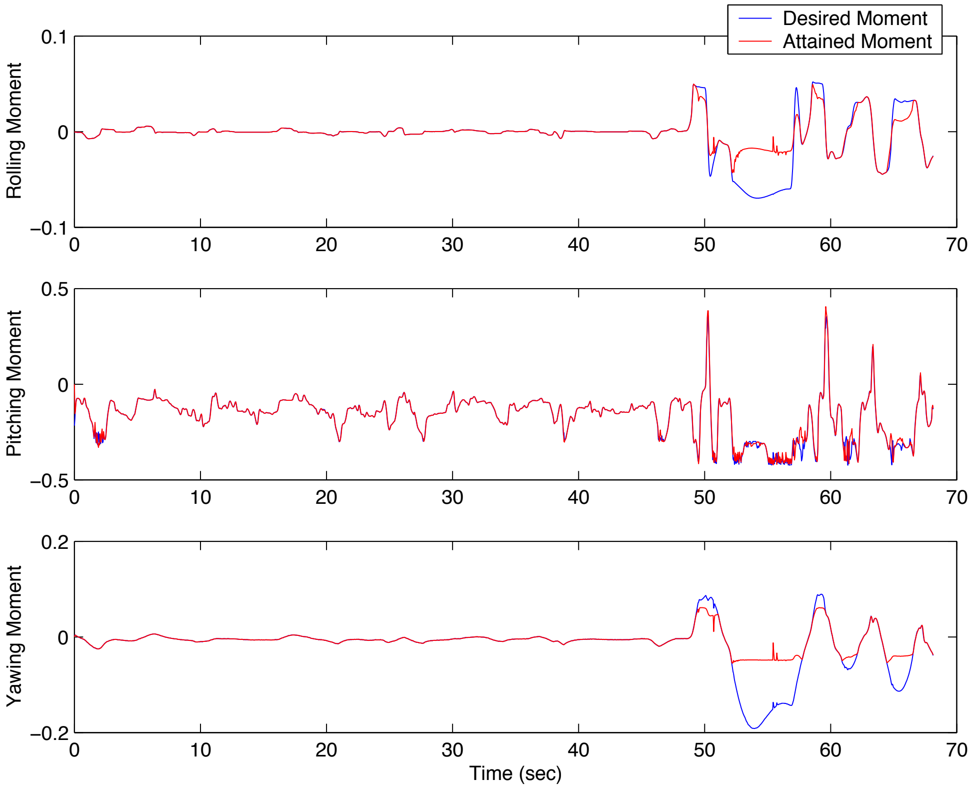

Figure 5.8: Desired moment time histories plotted with attained moments for the offset left approach using the method of pitch prioritization.

5.2.3 Offset High Left Carrier Approach maneuver: Results

An additional offset approach was used to further evaluate the two methods of dealing with unattainable moments. Data was taken with a more aggressive maneuver to see if the results from section 5.2.2 were repeatable and to determine whether pitch prioritization could handle larger unattainable moments with the same level of instability. For the most part, the conclusions drawn from this data set resemble those from the previous section. Figure 5.9 is the saturation plot for this maneuver. The prioritization mode was able to achieve all of the pitching moment without saturating any of the controls on the airframe. There is a large area of saturation in the lateral-directional axes however during the time when the airframe was demanding a large amount of pitching moment. This was caused by a more aggressive maneuver that required more pitch moment than the previous maneuver. The saturation plot for the scaling method ends early because the airframe diverged in pitch as soon as the corrective maneuver was applied. The following plots for the scaling method are all short because of this early unstable divergence. Because the scaling flight ended early, the percentages of saturation do not hold much merit. However, it is clearly seen that using the method of scaling for this magnitude of corrective maneuver caused the airframe to diverge from normal flight, becoming impossible to control.

During the prioritization mode all of the pitch moment was satisfied without saturating the controls. This caused the lateral-directional axes to be saturated 21.4% of the time over the entire task. This related to an 18.1% loss in roll control, and an 18.5% loss in yaw control. These percentages are calculated as previously explained in section 5.2.2. This loss of control in the lateral-directional axes was the direct trade-off of keeping the airframe stable throughout the entire flight.

Figure 5.10 plots the stick movement throughout the entire flight using both methods. Looking at the longitudinal and lateral stick, both methods have approximately the same stick history which produces approximately the same moment time history, which will be seen later. The major difference between these flights occur when the corrective maneuver is applied. Shortly after the initial lateral and longitudinal stick input were applied, the airframe using the scaling method diverged wildly, becoming unable to control, while the prioritization method kept the airframe stabilized by providing the required pitching moment. Figure 5.11 shows the unstable divergence using the scaling method by plotting the angle of attack. Note that in prioritization mode the angle of attack did not make any fast changes, indicating

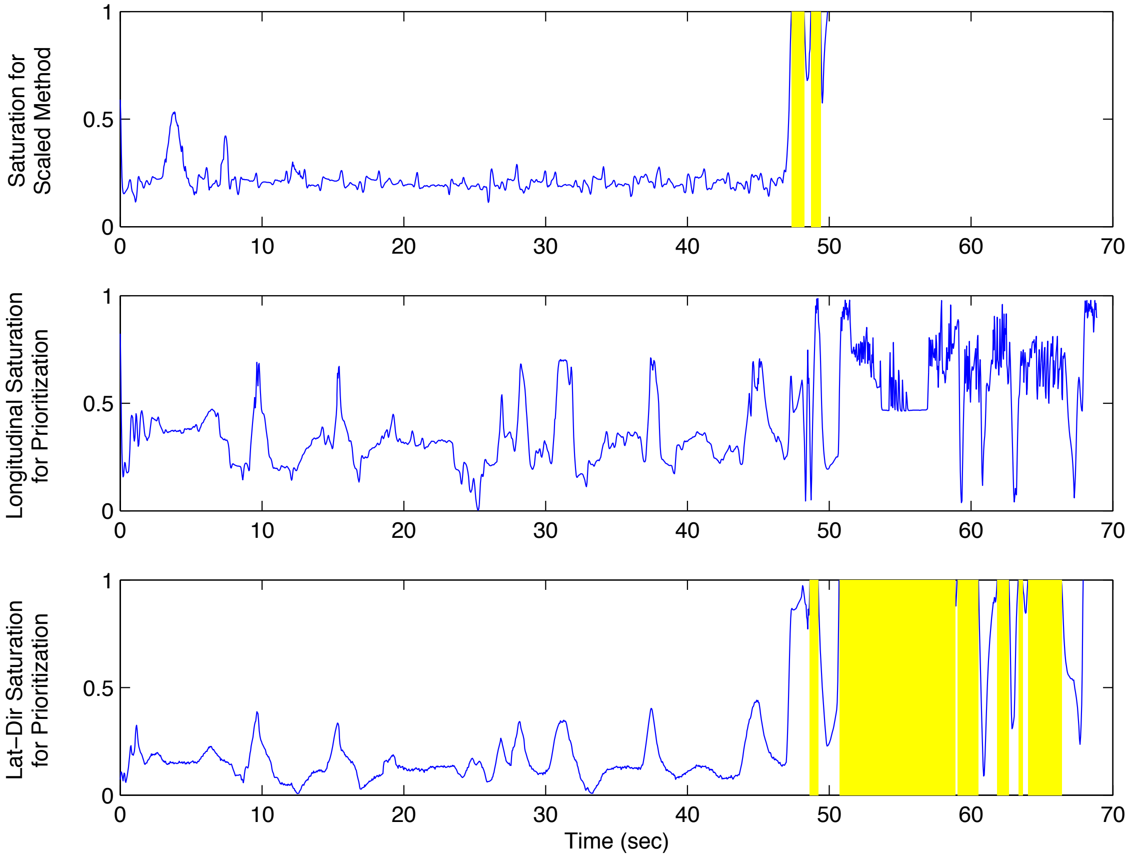

Figure 5.9: Comparison of the saturation of control surfaces between scaling the moment direction and prioritizing the pitch axis for an offset high left maneuver.

that the airframe did not exhibit unstable characteristics.

Figure 5.12 shows the time histories of the commanded and actual pitch and roll rates. This figure indicates that when most of the control power is used to achieve the pitching moment, the roll rate requirement is not met. The yaw rate exhibits the same characteristics as the roll rate does and is left out of the figure. There is a period of time at the end of the flight using the method of pitch prioritization when the desired pitch rate was not attained. However, during this time the desired pitching moment was attained. This could be related to control windup, where the controls are near limits and do not have the necessary authority to attain the desired rate. This unattained pitch rate was noted by the pilot's concern with the nose very slow in reacting to the correction.

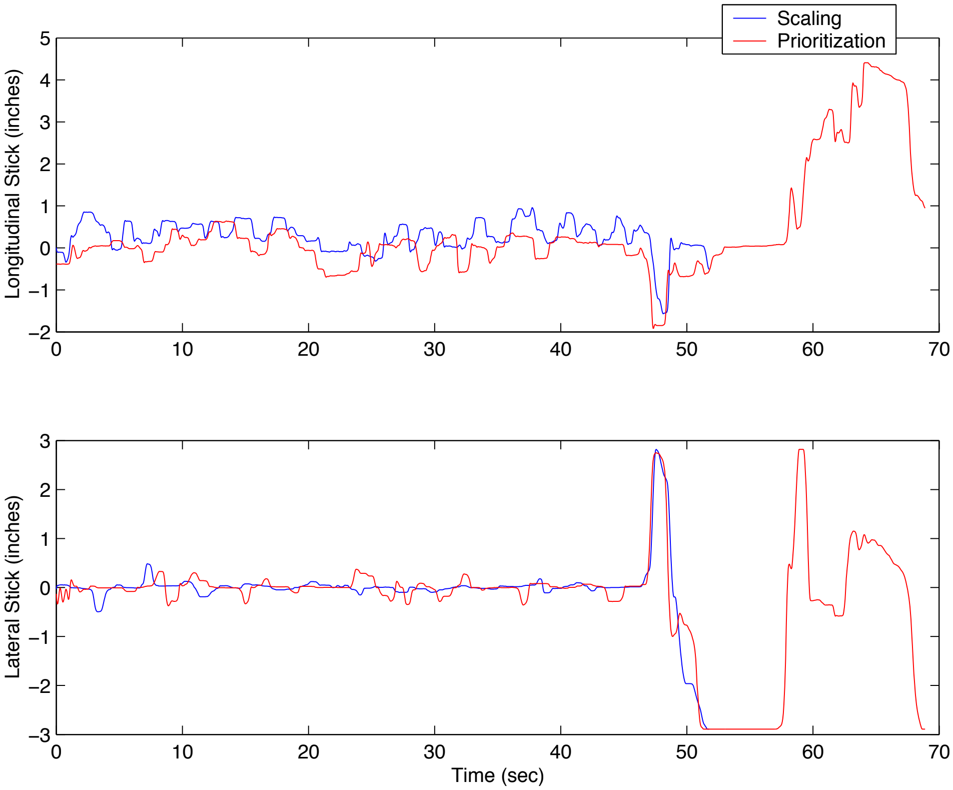

Figure 5.10: Longitudinal and lateral stick time histories for the offset high left approach using both methods.

Comparing the desired moment time histories in figure 5.13 for the two different methods reveals that the required moments are almost the same. This shows that the flights were nearly identical in that they both required approximately the same magnitude of moment. However, figure 5.14 shows that about 47 seconds into the flight the scaled method was unable to attain the required pitching moment to keep the airframe stable. The airframe departs from normal flight about three seconds after the initial correction. With the airframe so unstable and the airframe performing such aggressive tasks, any amount of desired pitching moment that is not attained could prove fatal to the airframe and the pilot. This was the case for the flight using the method of scaling the desired moment direction.

Figure 5.15 shows the necessary pitching requirements in the prioritization method are met to keep the airframe stable. Prioritization in this case provided enough control power to meet the requirements of the control law in stabilizing the airframe. Again, the trade off is

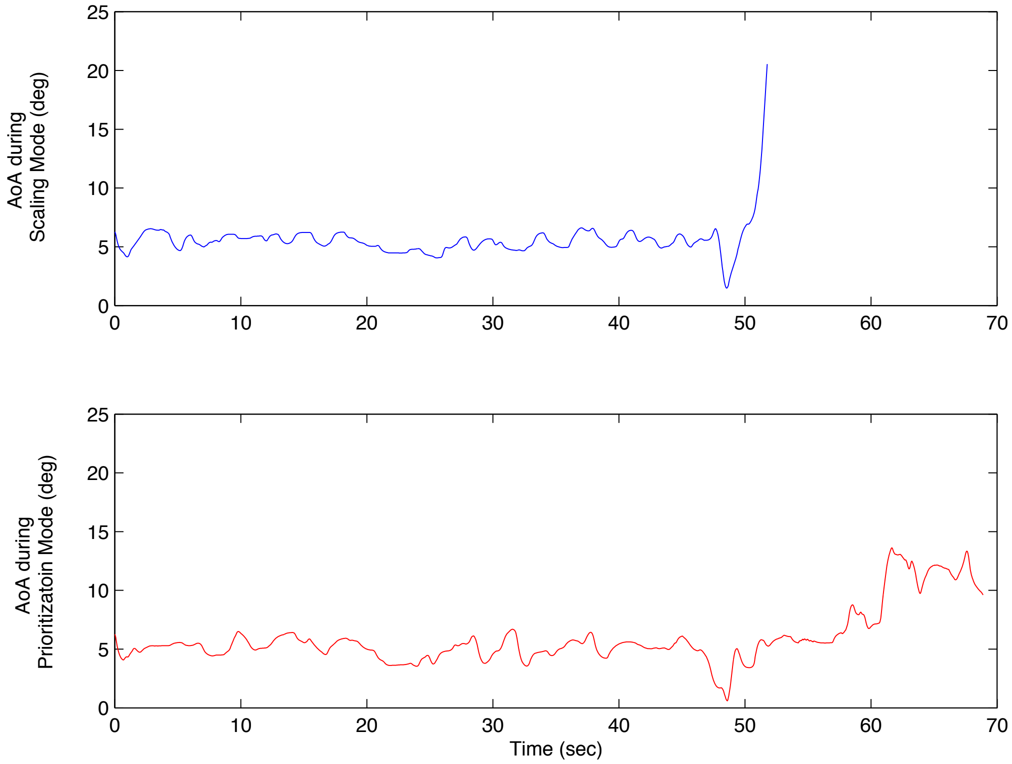

Figure 5.11: Angle of attack time histories for the offset high left approach using both methods.

in the rolling and yawing moments while pitch is being prioritized.

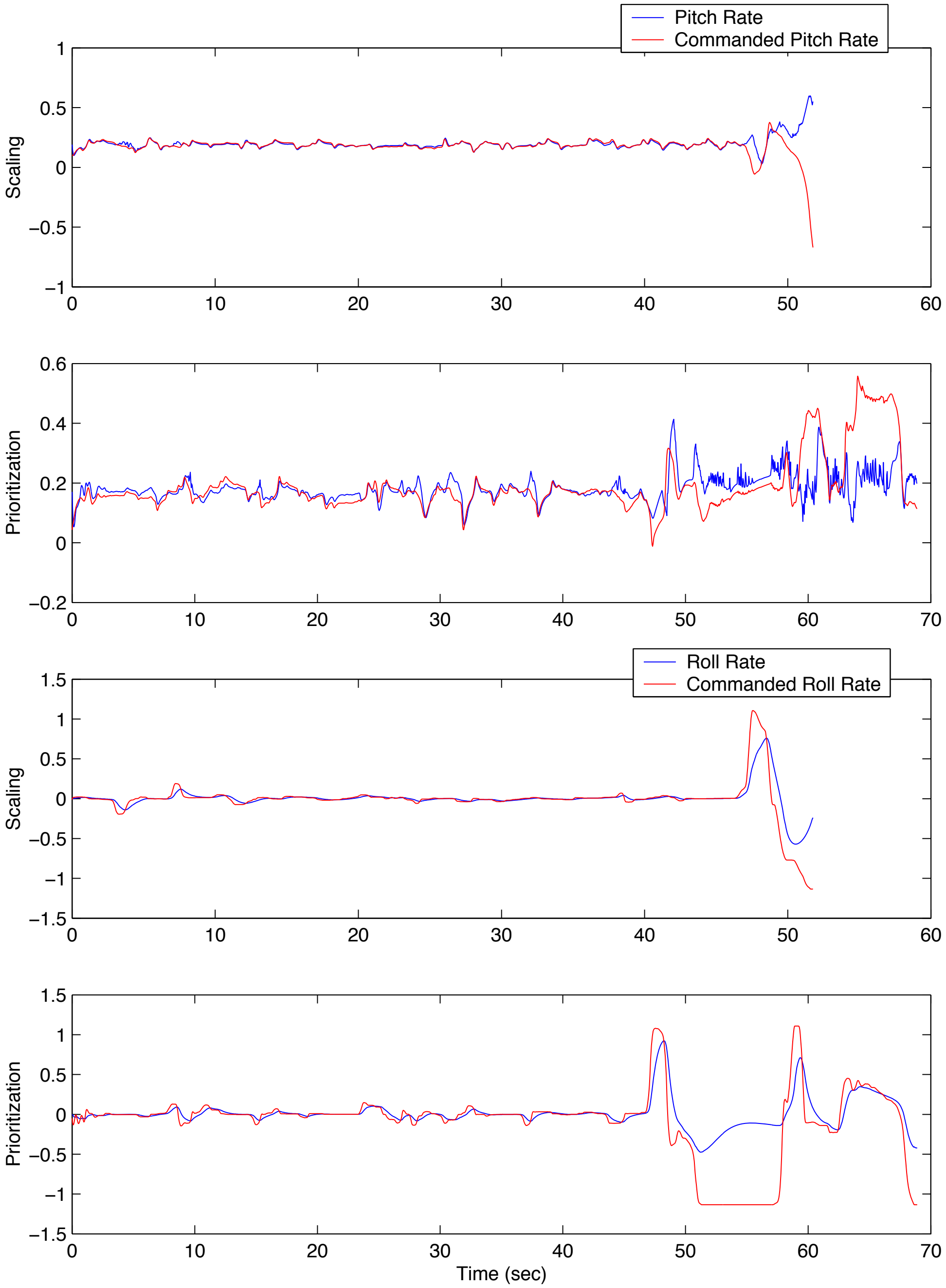

Figure 5.12: Aircraft pitch and roll rate time histories for the offset high left approach using both methods.

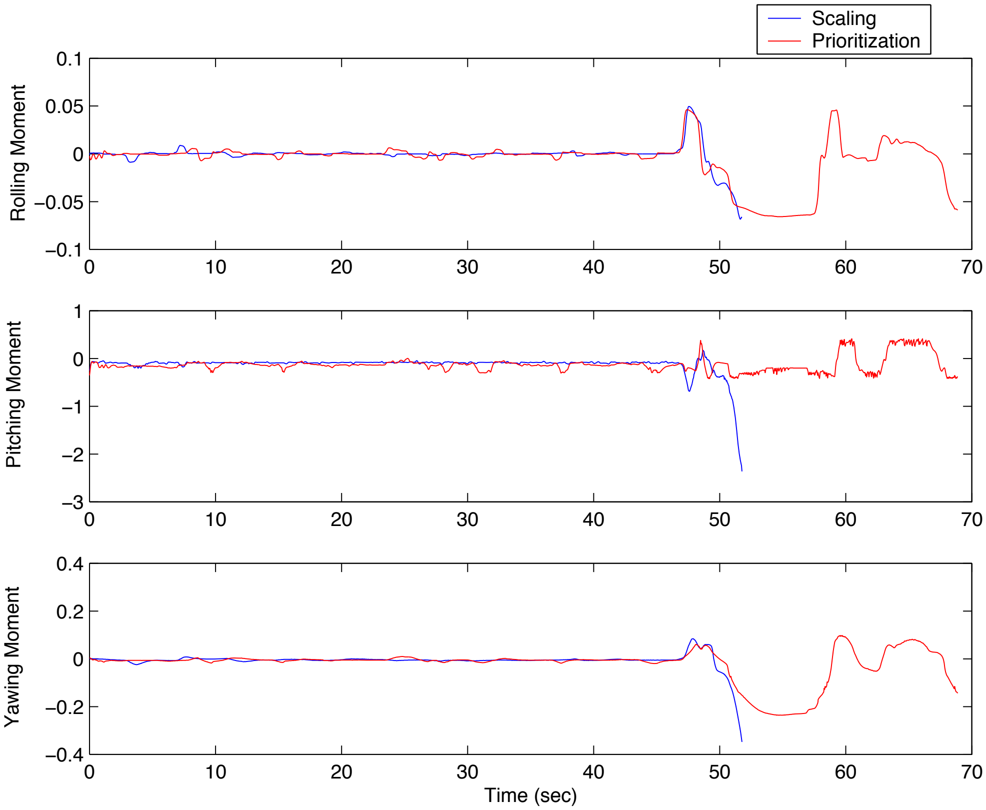

Figure 5.13: Desired moment time histories for the offset high left approach using both methods.

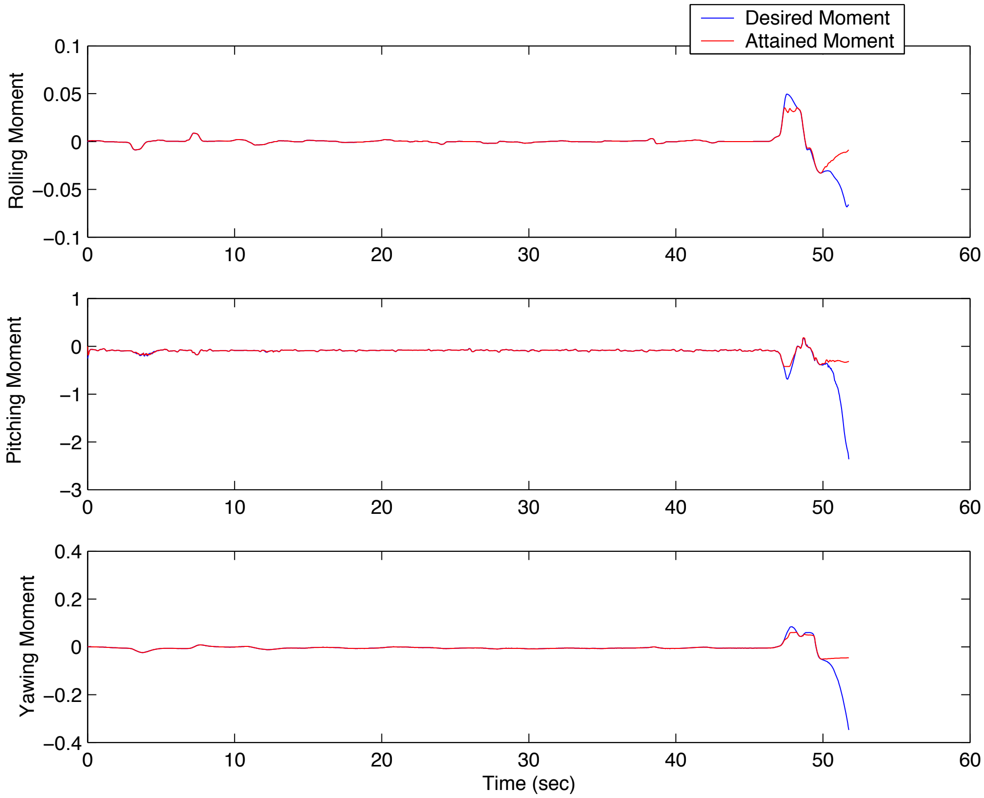

Figure 5.14: Desired moment time histories plotted with attained moments for the offset high left approach using the method of preserving the moment direction.

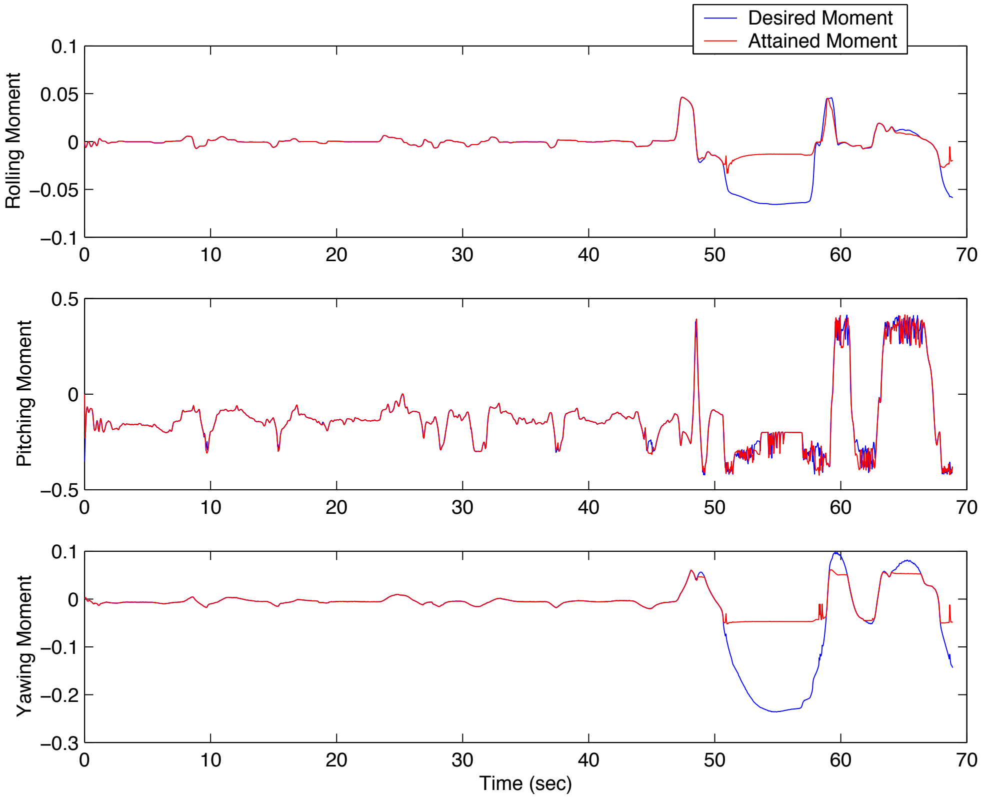

Figure 5.15: Desired moment time histories plotted with attained moments for the offset high left approach using the method of pitch prioritization.

5.2.4 Offset Carrier Approach maneuvers: Pilot Interpretation

The main trend seen in all the test flights was the pilot adaptation to how the aircraft handled. This learning curve made it difficult to investigate all of the differences between the two different methods. The pilot found himself changing techniques to avoid an unstable situation. To better understand the differences it was necessary to remove the pilot adaptation by requiring the pilot to make hard aggressive maneuvers each time to regain the desired glide-slope as quickly as possible.

During the flights the pilot felt that the aircraft was uncomfortable to fly in any situation. He compared the handling to that of an aircraft with some sort of control deficient situation such as a hydraulics failure or damage to the airframe. Overall the pilot preferred the method of pitch prioritization because the method prevented the aircraft from having an unstable feel. The one disconcerting tendency of the airframe while in pitch prioritization mode was the continuously changing amount of lateral control power during flight. However, the pilot was willing to deal with the situation if it meant that the airframe would not display unstable characteristics.

The method of scaling the desired moment direction gave the aircraft an unstable feel to the pilot during any kind of aggressive maneuver. In an extreme maneuver such as the high left offset, the airplane would diverge. The pilot also commented on the lack of longitudinal control power, saying that there was not enough, especially nose down control authority. This is because the desired pitching moment is reduced due to moment direction preservation as explained in chapter 4.

Prioritization in the pitch axis provided an airframe in which stability issues were not a question. The pilot did not feel any instability in the airframe using the same aggresive corrective maneuvers that were applied with the scaling method. However, the pilot was able to feel the trade off between lateral-directional control and pitch augmentation. Initial lateral corrections were responsive but became a little more sluggish as time and pitching moment requirements increased. Overall, the airframe with pitch prioritization was “more smooth” compared to the flight with the method of scaling the moment direction implemented.

Prioritization had the tendency to make the airframe feel sluggish at times to the pilot. During the high left offset there was a time in which the full amount of pitch rate was not satisfied. This unattained pitch rate gave the airframe a sluggish feel in the longitudinal axis and could be attributed to the control wind-up problem mentioned earlier. A sluggish feel in the lateral control was also noticed. This tendency was due to the airframe not having the required control authority to achieve the desired lateral characteristics after providing the required longitudinal dynamics.

Chapter 6

Summary and Conclusions

A comparison of two methods to deal with unattainable moments in an aircraft was performed using a testbed airframe that resembled an F/A-18 with a highly redundant control effector suite. Control allocation utilizing the Bisecting, Edge Searching Algorithm was implemented on this airframe and a dynamic inversion control law was used to produce the desired aircraft handling qualities. Real-time piloted simulations were used to evaluate the performance characteristics of each method. Simulation results indicated that the method of pitch prioritization implemented with a control allocation scheme provided a different approach to allocating controls for a situation in which the desired moment was unattainable. In extreme unstable cases prioritizing the pitch axis produced favorable characteristics with respect to the alternative of preserving the moment direction by scaling the control solution vector that the allocator produces.

Representative maneuvers that offered real-time evaluation of the two methods were flown with a pilot-in-the-loop. These maneuvers were chosen such that the two methods under investigation could be evaluated in an aggresive pilot task. The stability of the airframe was reduced such that at any time a desired pitch moment could not be produced, an unstable divergence would occur. These tasks were flown at the edge of the flight envelope in a low dynamic pressure flight condition. The validity of these tasks were subject to pilot compensation, to the point where the pilot was required to remove any learning of how the aircraft feels from the desired task. This provided a task that was flown the same way each time, providing repeatable data to compare the two methods.

The results in this paper have demonstrated the capabilities of the method of pitch prioritization as compared to scaling the moment direction for an unstable aircraft performing a specialized task. In this special case it was seen that prioritizing the pitch axis was the better option compared to preserving the moment direction. In extreme cases pitch prioritization was able to keep the airframe flying while the alternative method rendered the airframe uncontrollable.

The advantage to having the pilot-in-the-loop during these simulations is that it provides an interpretation that cannot be gained by plotting data sets. A pilot also provides stick and rudder inputs that are realistic. Using canned inputs seldom yields airplane trajectories representative of real life. The pilot in this research provided valuable insight to how these methods differ in the feel of the airframe. Overall the pilot preferred to fly the prioritization for an aircraft in the situation that was given. Prioritization provided a feeling of stability in the aircraft that was not felt while the method of scaling the control solution vector was used. The pilot felt that trading the uneasy feeling of constantly changing lateral-directional control power for the aircraft remaining stable was acceptable.

Much future work into the investigation of unattainable moments is neccessary before a concrete conclusion as to the best way of dealing with this situation is needed. This research provides a background investigation into a specific case of dealing with unattainable moments. While the method presented in this paper is not meant to be the best way of dealing with unattainable moments for all cases, it provides an improved way of allocating control effectors to achieve favorable flight characteristics for the specific case in this paper.

Bibliography

[1] Durham, W., "Constrained Control Allocation" *Journal of Guidance, Control, and Dynamics,* Volume 16, Number 4, 1993, pp. 717-725.

[2] Durham, W., "Constrained Control Allocation: Three Moment Problem" *Journal of Guidance, Control, and Dynamics,* Volume 17, Number 2, 1994, pp. 330-336.

[3] Durham, W., and Bordignon, Kenneth A., "Closed-Form Solutions to Constrained Control Allocation Problem" *Journal of Guidance, Control, and Dynamics,* Volume 18, Number 5, 1995, pp. 1000-1007.

[4] Bordignon, K. *Constrained Control Allocation for Systems with Redundant Control Effectors,* Doctoral Dissertation, Virginia Polytechnic Institute and State University, Blacksburg, VA, 1996.

[5] Durham, W., "Attainable Moments for the Constrained Control Allocation Problem" *Journal of Guidance, Control, and Dynamics,* Volume 17, Number 6, 1994, pp. 1371-1373.

[6] Durham, W. C., "Computationally Efficient Control Allocation" AIAA-99-4214,1999.

[7] Scalera, K. *A Comparison of Control Allocation Methods for the F-15 ACTIVE Research Aircraft Utilizing Real-Time Piloted Simulations,* Master's Thesis, Virginia Polytechnic Institute and State University, Blacksburg, VA, 1999.

[8] Leedy, J. *Real-Time Moment Rate Constrained Control Allocation for Aircraft with a Multiply-Redundant Control Suite,* Master's Thesis, Virginia Polytechnic Institute and State University, Blacksburg, VA, 1998.

[9] Bolling, J. *Implementation of Constrained Control Allocation Techniques Using an Aerodynamic Model of an F-15 Aircraft,* Master's Thesis, Virginia Polytechnic Institute and State University, Blacksburg, VA, 1998.

[10] J. M. Buffington and Y. B. Shtessel, "Saturation Protection for Feedback Linearizable Systems using Sliding Mode Theory," Proceedings of the 1998 American Control Conference, Philadelphia PA, pp.1028-1032, Jun. 1998.

[11] Nichols, James H., Magyar, Thomas J. and Schug, Eric C., "The Platform-Independent Aircraft Simulation Environment at Manned Flight Simulator" AIAA-98-4179,1998.

[12] Stevens, Brian L. and Frank L. Lewis, *Aircraft Control and Simulations* , John Wiley & Sons, Inc., New York, 1992, pp. 593-607.

[13] McDonnell Douglas Corporation. "F/A-18 Flight Control System Design Report," Report Number MDC A7813, 1982.

Appendix A

F/A-18 Testbed Simulation Code

A.1 AERO.F

A.1.1 Description

AERO.F is an executive subroutine that calls subroutines based on the flight phase of the aircraft. In this specific case AERO.F calls AEROPA.F to calculate the aerodynamics of the scheduled/trimmed flight condition. The code then combines the aerodynamics from the non-linear scheduled/trimmed flight condition with the aerodynamics from the UGLOBAL array to produce the total aerodynamic forces and moments.

A.1.2 Code

```
C*****************************************************************
C
C TITLE: AERO
C
C-----------------------------------------------------------------
C
C FUNCTION: F18PA AERODYNAMIC MODEL
C Derived from Stevens and Lewis.
C
C-----------------------------------------------------------------
C
```

```
C DESIGNED BY: Drew Robbins
C
C CODED BY: Bull
C
C MAINTAINED BY:
C
C-----------------------------------------------------------------
C
C MODIFICATION HISTORY:
C
C DATE PURPOSE BY
C ==== ======= ==
C  1 FEB 98 Creation ACR
C  7 FEB 98       Removed CG Moment Corrections
C! Done in Shell_Accel.f ACR
C       16 JUN 98       Limited Velocity to VRW=0 problemS      REB
C                        Combined Calculations in non
C                         dimensional rate coeffiecinets
C July 2000 Made into F18PA Bull
C-----------------------------------------------------------------
C
C GLOSSARY
C ========
C
C ASSIGNMENTS:
C
C  NONE
C
C-----------------------------------------------------------------
C
C INPUTS:
C
C  ALFA       Angle Of Attack                          DEG          ANU
C  AREA       Wing Area                                FT2          N/A
C  BDIAL_L    Scale Roll axis of AMS                   --------     --------
C  BDIAL_M    Scale Pitch axis of AMS                  --------     --------
C  BDIAL_N    Scale yaw axis of AMS                    --------     --------
C  BETA       Sideslip Angle                           DEG          ANL
C  CALPH      Cosine Of Angle Of Attack                N/A          ANU
C  CG         C.g. Position W/r/t L.e. Of Mac          PCT          AFT
C  CHORD      Mean Aerodynamic Chord                   FT           N/A
C  IMODE      Sim. mode: -2=init,-1=reset,0=hold,1=ru  --------     --------
```

```
C  PB         Aircraft Roll Velocity, Body Frame      RAD/SEC      RWD
C  QB         Aircraft Pitch Velocity, B-frame        RAD/SEC      ANU
C  QBAR       Dynamic Pressure                        LB/FT2       N/A
C  R2D        57.2957795 Rad To Deg Conversion Factor DEG/RAD      --------
C  RB         Aircraft Yaw Velocity, B-frame          RAD/SEC      ANR
C  SALPH      Sine Of Angle Of Attack                 N/A          ANU
C  SPAN       Wing Span                               FT           N/A
C  THET       Pitch Euler Angle, L (local) Frame      DEG          ANU
C  U_GLOBAL(1)
C!            Global Ctl Deflections                  --------     --------
C  VRW        Velocity w.r.t. wind (true airspeed)    FT/SEC       N/A
C
C-----------------------------------------------------------------
C
C OUTPUTS:
C
C  C1U        C1 due to Ctls                          --------     --------
C  CD         Coefficient Of Drag                     N/A          AFT
C  CL         Coefficient Of Lift                     N/A          UP
C  CLL        Total Rolling Moment Coef., X-body      N/A          RWD
C  CLM        Total Pitch Moment Coef., Y-body        N/A          ANU
C  CLN        Total Yawing Moment Coef., Z-body       N/A          ANR
C  CMU        Cm due to Ctls                          --------     --------
C  CNU        Cn due to Ctls                          --------     --------
C  CX         Aero X-body Force Coeff                 N/A          FWD
C  CXU        CX due to Ctls                          --------     --------
C  CY         Aero Y-body Side Force Coeff            N/A          RT
C  CYU        CY due to Ctls                          --------     --------
C  CZ         Aero Z-body Force Coeff                 N/A          DOWN
C  CZU        CZ due to Ctls                          --------     --------
C  DXCG       Cg Displacement From Reference          FT           FWD
C  FAX        Total Aero Force, X-body                LB           FWD
C  FAY        Total Aero Force, Y-body                LB           RT
C  FAZ        Total Aero Force, Z-body                LB           DOWN
C  TAL        Aero Rolling moment, X-body             FT-LB        RWD
C  TAM        Aero pitch moment, Y-body               FT-LB        ANU
C  TAN        Aero Yawing moment, Z-body              FT-LB        ANR
C
C-----------------------------------------------------------------
C
C LOCALS:
C
```

```
C  B          Description Not Available                --------     --------
C
C-----------------------------------------------------------------

      SUBROUTINE AERO

C-----------------------------------------------------------------
C
C DECLARATION SECTION
C
C-----------------------------------------------------------------
      IMPLICIT NONE
C ** Parameters


C ** INPUTS:

INTEGER*4 IMODE
REAL*4 AREA, BDIAL_L, BDIAL_M, BDIAL_N, BETA, CALPH, CG, CHORD
REAL*4 ALFA, PB, QB, QBAR, R2D, RB, SALPH, SPAN, THET
REAL*4 U_GLOBAL(20), VRW

C ** OUTPUTS:

REAL*4 C1U, CD, CL, CLL, CLM, CLN, CMU, CNU, CX, CXU, CY, CYU
REAL*4 CZ, CZU, DXCG, FAX, FAY, FAZ, TAL, TAM, TAN

C ** LOCALS:

REAL*4 B(6,20)

C ** OTHER LOCALS:

BYTE CONPAR, F15ACTIVE, F16PAR, KONSTANTS, LCLBUF, SIMPAR
BYTE XF18PA
LOGICAL*4 INITIALIZED, SCALED
INTEGER*4 K, M, J, I
REAL*4 BDUM(6,20), CGREF, BDIAL

C-----------------------------------------------------------------
C
C COMMON SECTION
```

```
C
C-----------------------------------------------------------------

COMMON/ BK_AERO / LCLBUF(480)
        COMMON/ SHELL1 / CONPAR(424)
        COMMON/ SHELL2 / SIMPAR(1024)
        COMMON/ KONSTANT / KONSTANTS(40)







COMMON/ XF18PA / XF18PA(2092)

C-----------------------------------------------------------------
C
C EQUIVALENCE SECTION
C
C-----------------------------------------------------------------

C ** INPUTS:

EQUIVALENCE( SIMPAR(97), ALFA)
EQUIVALENCE( SIMPAR(557), AREA)
EQUIVALENCE( XF18PA(1921), BDIAL_L)
EQUIVALENCE( XF18PA(1925), BDIAL_M)
EQUIVALENCE( XF18PA(1929), BDIAL_N)
EQUIVALENCE( SIMPAR(101), BETA)
EQUIVALENCE( SIMPAR(125), CALPH)
EQUIVALENCE( SIMPAR(573), CG)
EQUIVALENCE( SIMPAR(565), CHORD)
EQUIVALENCE( CONPAR(1), IMODE)
EQUIVALENCE( SIMPAR(145), PB)
EQUIVALENCE( SIMPAR(149), QB)
EQUIVALENCE( SIMPAR(549), QBAR)
EQUIVALENCE( KONSTANTS(9), R2D)
EQUIVALENCE( SIMPAR(153), RB)
EQUIVALENCE( SIMPAR(121), SALPH)
EQUIVALENCE( SIMPAR(561), SPAN)
EQUIVALENCE( SIMPAR(5), THET)
EQUIVALENCE( XF18PA(49), U_GLOBAL(1))
EQUIVALENCE( SIMPAR(205), VRW)

C ** OUTPUTS:

EQUIVALENCE( XF18PA(141), C1U)
EQUIVALENCE( SIMPAR(377), CD)
EQUIVALENCE( SIMPAR(373), CL)
EQUIVALENCE( SIMPAR(457), CLL)
EQUIVALENCE( SIMPAR(461), CLM)
EQUIVALENCE( SIMPAR(465), CLN)
EQUIVALENCE( XF18PA(145), CMU)
EQUIVALENCE( XF18PA(149), CNU)
EQUIVALENCE( SIMPAR(381), CX)
EQUIVALENCE( XF18PA(129), CXU)
EQUIVALENCE( SIMPAR(385), CY)
EQUIVALENCE( XF18PA(133), CYU)
EQUIVALENCE( SIMPAR(389), CZ)
EQUIVALENCE( XF18PA(137), CZU)
EQUIVALENCE( SIMPAR(605), DXCG)
EQUIVALENCE( SIMPAR(393), FAX)
EQUIVALENCE( SIMPAR(397), FAY)
EQUIVALENCE( SIMPAR(401), FAZ)
EQUIVALENCE( SIMPAR(469), TAL)
EQUIVALENCE( SIMPAR(473), TAM)
EQUIVALENCE( SIMPAR(477), TAN)

C ** LOCALS:

EQUIVALENCE( LCLBUF(1), B)

C----------------------------------------------------------------
C
C DATA SECTION
C
C----------------------------------------------------------------

      DATA M / 20 /
DATA INITIALIZED  /  .FALSE. /
```

```
      DATA (B(1,I),I=1,20)/1.8477e-03,    1.8477e-03,  -2.3546e-04,
     &  -2.3546e-04, 0., 0., 1.7477e-03,  1.7477e-03,
     &  -2.2546e-04,-2.2546e-04, 0., 0.,  1.7477e-03,  1.7477e-03,
     &  -2.2546e-04,-2.2546e-04, 0., 0., -2.2546e-04,-2.2546e-04/


      DATA (B(2,I),I=1,20)/-2.6178e-04,   2.6178e-04,   1.1510e-04,
     &  -1.1510e-04,   4.1521e-03,   4.1521e-03,
     &  -2.5178e-04,   2.5178e-04,   1.0510e-04,
     &  -1.0510e-04,   4.0521e-03,   4.0521e-03,
     &  -2.5178e-04,   2.5178e-04,   1.0510e-04,
     &  -1.0510e-04,   4.0521e-03,   4.0521e-03,
     &   1.0510e-04,  -1.0510e-04/

      DATA (B(3,I),I=1,20)/-5.6102e-03,  -5.6102e-03,  -4.2584e-04 ,
     &  -4.2584e-04, 0., 0., -5.5102e-03,  -5.5102e-03,
     &  -4.1584e-04 , -4.1584e-04, 0., 0.,  -5.5102e-03,  -5.5102e-03,
     &  -4.1584e-04 , -4.1584e-04, 0., 0.,  -4.1584e-04 , -4.1584e-04/



      DATA (B(4,I),I=1,20)/5.2100e-04,  -5.2200e-04,  -7.4500e-04,
     &  7.4600e-04,   2.5300e-04,   2.5400e-04, 4.2100e-04, -4.2200e-04,
     &-6.4500e-04,  6.4600e-04,   1.5300e-04,   1.5400e-04, 4.2100e-04,
     &-4.2200e-04,  -6.4500e-04,  6.4600e-04,   1.5300e-04,   1.5400e-04,
     &  -6.4500e-04,  6.4600e-04   /

      DATA (B(5,I),I=1,20)/-6.6900e-03,  -6.7000e-03,  -5.6600e-05,
     &  -5.6700e-05, 0., 0., -6.5900e-03,  -6.6000e-03,  -5.5600e-05,
     &  -5.5700e-05, 0., 0., -6.5900e-03,  -6.6000e-03,  -5.5600e-05,
     &  -5.5700e-05, 0., 0., -5.5600e-02,  -5.5700e-02/

      DATA (B(6,I),I=1,20)/2.3700e-04,  -2.3800e-04,   -2.7600e-04,
     &  2.7700e-04,  -1.1700e-03, -1.1800e-03, 1.3700e-04, -1.3800e-04,
     &   -1.7600e-04,  1.7700e-04,  -1.0700e-03, -1.0800e-03,
     &  0.13700e-04, -0.13800e-04,  -1.7600e-04,  1.7700e-04,
     &   -1.0700e-03, -1.0800e-03,  -1.7600e-04,  1.7700e-04/



C-----------------------------------------------------------------
```

```
C
C INITIALIZATION SECTION
C
C-----------------------------------------------------------------

      IF( (IMODE.EQ.-2))THEN
        U_GLOBAL(1) = 0.
        U_GLOBAL(2) = 0.
        U_GLOBAL(3) = 0.
        U_GLOBAL(4) = 0.
        U_GLOBAL(5) = 0.
        U_GLOBAL(6) = 0.
        U_GLOBAL(7) = 0.
        U_GLOBAL(8) = 0.
        U_GLOBAL(9) = 0.
        U_GLOBAL(10) = 0.
        U_GLOBAL(11) = 0.
        U_GLOBAL(12) = 0.
        U_GLOBAL(13) = 0.
        U_GLOBAL(14) = 0.
        U_GLOBAL(15) = 0.
        U_GLOBAL(16) = 0.
        U_GLOBAL(17) = 0.
        U_GLOBAL(18) = 0.
        U_GLOBAL(19) = 0.
        U_GLOBAL(20) = 0.
      DO I=1,6
   DO J=1,M
      BDUM(I,J)=B(I,J)
   ENDDO
ENDDO
      ENDIF
C-----------------------------------------------------------------
C
C RESET SECTION
C
C-----------------------------------------------------------------

      IF ((IMODE.LE.0)) THEN
       Initialized = .TRUE.
        U_GLOBAL(1) = 0.
        U_GLOBAL(2) = 0.
        U_GLOBAL(3) = 0.
        U_GLOBAL(4) = 0.
        U_GLOBAL(5) = 0.
        U_GLOBAL(6) = 0.
        U_GLOBAL(7) = 0.
        U_GLOBAL(8) = 0.
        U_GLOBAL(9) = 0.
        U_GLOBAL(10) = 0.
        U_GLOBAL(11) = 0.
        U_GLOBAL(12) = 0.
        U_GLOBAL(13) = 0.
        U_GLOBAL(14) = 0.
        U_GLOBAL(15) = 0.
        U_GLOBAL(16) = 0.
        U_GLOBAL(17) = 0.
        U_GLOBAL(18) = 0.
        U_GLOBAL(19) = 0.
        U_GLOBAL(20) = 0.

      DO I=1,6
   DO J=1,M
      B(I,J)=BDUM(I,J)
   ENDDO
ENDDO

      SCALED=.FALSE.
      END IF

C-----------------------------------------------------------------
C
C RUN SECTION
C
C-----------------------------------------------------------------

      CALL  AEROPA
      IF (.NOT. SCALED) THEN
      SCALED=.TRUE.


   DO J=1,M
      B(1,J)=BDIAL_L*B(1,J)
      B(2,J)=BDIAL_M*B(2,J)
      B(3,J)=BDIAL_N*B(3,J)
      B(4,J)=BDIAL_L*B(4,J)
      B(5,J)=BDIAL_M*B(5,J)
      B(6,J)=BDIAL_N*B(6,J)
   ENDDO
      ENDIF
      CXU=0.
      CYU=0.
      CZU=0.
      C1U=0.
      CMU=0.
      CNU=0.
      DO I=1,M
       CXU=CXU+B(1,I)*U_GLOBAL(I)
       CYU=CYU+B(2,I)*U_GLOBAL(I)
       CZU=CZU+B(3,I)*U_GLOBAL(I)
       C1U=C1U+B(4,I)*U_GLOBAL(I)
       CMU=CMU+B(5,I)*U_GLOBAL(I)
       CNU=CNU+B(6,I)*U_GLOBAL(I)
      ENDDO
C
      CX =   CL*SALPH - CD*CALPH
      CZ = -(CL*CALPH + CD*SALPH)

C-----------------------------------------------------------------
C BODY AXES
C-----------------------------------------------------------------

      FAX =  QBAR*AREA*(CX + CXU)
      FAY =  QBAR*AREA*(CY + CYU)
      FAZ =  QBAR*AREA*(CZ + CZU)

      CGREF       =  0.25
      DXCG     =  (CGREF-CG)*CHORD

      TAL = QBAR*AREA*SPAN*(CLL + C1U)
      TAM = QBAR*AREA*CHORD*(CLM + CMU)
      TAN = QBAR*AREA*SPAN*(CLN + CNU)

C-----------------------------------------------------------------
C
C END OF AERO MODULE
```

```
C
C------------------------------------------------------------------
       RETURN
       END
```

A.2 AEROPA.F

A.2.1 Description

This section includes only the modifications made to AEROPA.F.

A.2.2 Code

```
      ABET = ABS(BETA)
      DDLEF = 0.
      DNGEAR = 1.
      DNSB = 0.
      FEO = 0
      FLARUD = 0.
      HAT = ALT-65.
      ICLTNK = 0
      INTRDC = 0
      KAIM9 = 0
      SBETAF = SIGN(1.,BETA)

      TOE_IN = 40.-5.*ALFA
      DRUDL = MIN(- TOE_IN,30.)
      DRUDL = MAX(DRUDL,-30.)
      DRUDR = MIN(TOE_IN,30.)
      DRUDR = MAX(DRUDR,-30.)

        QCLIM = QBAR
        IF (QCLIM.LT. 68.) QCLIM = 68.
        IF (QCLIM.GT.285.) QCLIM = 285.
        F26 = 65. - 0.334*QCLIM + 5.8E-04*QCLIM*QCLIM
      DADROOP = MIN(F26,42.)
        F69A = ALFA
        IF (F69A.LT.4.) F69A = 4.
        IF (F69A.GT.20.) F69A = 20.
        F69 = 6.5 + 1.375*F69A
      DLEFL = F69
      DLEFR = F69
      DRAD = DADROOP
      DLAD = DADROOP
      DTEFL = F26
      DTEFR = F26

        DHTD = (DRHTD+DLHTD)/2.    !MDN change 09/06/00
DRO2F = (DRUDL + DRUDR)*0.5
        DRUDD = 0.
        DTEFD = F26
        DLEFD = F69
        DDTD = (DRHTD-DLHTD)    !MDN change 09/06/00
        FLARUD = DRUDL - DRUDR
        DAD = 0.
        DTEFDI = 0.
        DDLEF = 0.


DLEFP = ABS(DDLEF)

FLAR1 = -FLARUD*0.5
```

A.3 CONTROL.F

A.3.1 Description

This subroutine calculates the desired moments for the airframe based on stick position and airframe states. These desired moments are then input into the control allocator based on the chosen method of dealing with unattainable moments. This code also includes the routine to keep aircraft velocity constant during the task and the method of reducing the attainable moment subset (AMS).

A.3.2 Code

```
C*****************************************************************
C
C TITLE: CONTROL
C
C-----------------------------------------------------------------
C
C FUNCTION: Control Law
C
C
C-----------------------------------------------------------------
C DESIGNED BY: Bull
C
C CODED BY: Bull
C
C MAINTAINED BY: VPI SIMULATIONS
C
C-----------------------------------------------------------------
C MODIFICATION HISTORY:
C
C       DATE            PURPOSE                   BY
C       ====            =======                   ==
C
C       20 June 2000    Orig                     Bull
C
C
C-----------------------------------------------------------------
C GLOSSARY
```

```
C ========
C
C ASSIGNMENTS:
C
C  NONE
C
C-----------------------------------------------------------------
C INPUTS:
C
C  ALFA       Angle Of Attack                          DEG          ANU
C  AREA       Wing Area                                FT2          N/A
C  BDIAL_L    Scale Roll axis of AMS                   --------     --------
C  BDIAL_M    Scale Pitch axis of AMS                  --------     --------
C  BDIAL_N    Scale yaw axis of AMS                    --------     --------
C  B_FORCE(1) Force Coeff. Ctl. Eff.                   --------     --------
C  B_MOM(1)   Moment Coeff. Ctl. Eff.                  --------     --------
C  CALPH      Cosine Of Angle Of Attack                N/A          ANU
C  CHORD      Mean Aerodynamic Chord                   FT           N/A
C  CLL        Total Rolling Moment Coef., X-body       N/A          RWD
C  CLM        Total Pitch Moment Coef., Y-body         N/A          ANU
C  CLN        Total Yawing Moment Coef., Z-body        N/A          ANR
C  CPHI       Cosine Of Euler Roll Angle               N/A          RWD
C  CPSI       Cosine Of Euler Yaw Angle                N/A          ANR
C  CTHT       Cosine Or Euler Pitch Angle              N/A          ANU
C  CX         Aero X-body Force Coeff                  N/A          FWD
C  CXU        CX due to Ctls                           --------     --------
C  CY         Aero Y-body Side Force Coeff             N/A          RT
C  CYU        CY due to Ctls                           --------     --------
C  CZ         Aero Z-body Force Coeff                  N/A          DOWN
C  CZU        CZ due to Ctls                           --------     --------
C  DADROOP    Aileroon droop (average deflection)      DEG          TED
C  DIAGS      Do DA3 Diagnostics                       --------     --------
C  DLHTD      Left Horizontal Tail                     DEG          TED
C  DRHTD      Right Horizontal Tail                    DEG          TED
C  DRUDL      Left rudder deflection                   DEG          TEL
C  DRUDR      Right rudder deflection                  DEG          TEL
C  DT         Simulation time step (frame time)        SEC          --------
C  DXCG       Cg Displacement From Reference           FT           FWD
C  FAX        Total Aero Force, X-body                 LB           FWD
C  FAY        Total Aero Force, Y-body                 LB           RT
C  FAZ        Total Aero Force, Z-body                 LB           DOWN
C  G          Acceleration Due To Gravity              FT/S2        --------
```

```
C  IMODE      Sim. mode: -2=init,-1=reset,0=hold,1=ru  --------     --------
C  IREST      Restoring, 0=No, 1=Min. Norm             --------     NLT
C  JOVDCK(1)  Wheel over deck flag                     --------     --------
C  LATSTK     Lateral Stick Deflection                 --------     --------
C  LONSTK     Longitudinal Stick Deflection            --------     --------
C  M          Number of Ctls (max 20)                  --------     --------
C  NBI        Number of Bisections (max 20)            --------     --------
C  P(1)       Pseudoinverse matrix                     --------     --------
C  PB         Aircraft Roll Velocity, Body Frame       RAD/SEC      RWD
C  PI         Constant = pi = 3.14159                  --------     --------
C  PLATRM     Average PLA trim value                   PCT          --------
C  QB         Aircraft Pitch Velocity, B-frame         RAD/SEC      ANU
C  QBAR       Dynamic Pressure                         LB/FT2       N/A
C  RB         Aircraft Yaw Velocity, B-frame           RAD/SEC      ANR
C  RUDPED     Rudder Pedal Deflection                  --------     --------
C  SALPH      Sine Of Angle Of Attack                  N/A          ANU
C  SPAN       Wing Span                                FT           N/A
C  SPHI       Sine Of Euler Roll Angle                 N/A          RWD
C  STHT       Sine Of Euler Pitch Angle                N/A          ANU
C  T11        CTHT*CPSI                  (L To B)      N/A          --------
C  T12        CTHT*SPSI                  (L to B)      N/A          --------
C  T13        -STHT                      (L to B)      N/A          --------
C  T21        SPHI*STHT*CPSI - CPHI*SPSI (L To B)      N/A          --------
C  T22        SPHI*STHT*SPSI + CPHI*CPSI (L to B)      N/A          --------
C  T23        SPHI*CTHT                  (L to B)      N/A          --------
C  T31        CPHI*STHT*CPSI + SPHI*SPSI (L to B)      N/A          --------
C  T32        CPHI*STHT*SPSI - SPHI*CPSI (L to B)      N/A          --------
C  T33        CPHI*CTHT                  (L to B)      N/A          --------
C  TIME       Time Since Start Of Operate Mode         SEC          N/A
C  TRIMMING   Simulation is currently trimming         --------     --------
C  UB         X-velocity B-frame                       FT/SEC       FWD
C  VB         Y-velocity B-frame                       FT/SEC       RT
C  VDW        Steady state wind Velocity Down          FT/SEC       DOWN
C  VEW        Steady state wind velocity East          FT/SEC       EAST
C  VNW        Steady state wind velocity North         FT/SEC       NORTH
C  VRW        Velocity w.r.t. wind (true airspeed)     FT/SEC       N/A
C  WB         Z-velocity B-frame                       FT/SEC       DWN
C  XIXX       Vehicle X - Moment Of Inertia about Cg   SLUG-FT2     N/A
C  XIXZ       Vehicle Xz Prod Of Inertia about Cg      SLUG-FT2     N/A
C  XIYY       Vehicle Y - Moment Of Inertia about Cg   SLUG-FT2     N/A
C  XIZZ       Vehicle Z - Moment Of Inertia about Cg   SLUG-FT2     N/A
C  XMASS      Mass Of Vehicle (incl Fuel)              SLUG         N/A
```

```
C
C-----------------------------------------------------------------
C OUTPUTS:
C
C  ALFACMD    Commanded angle-of-attack                --------     NLT
C  ALT        Altitude Of Aircraft Above Sea Lvl       FT           UP
C  BETACMD    Commanded sideslip angle                 --------     NLT
C  DTH        Throttle, percent                        PRCNT        +=More
C  D_MDES(1)  Delta = (Desired - Attained) Moment      None         --------
C  IERR       Allocator return code 0=OK, 1=Est.       --------     NLT
C  IERRA      0=Normal ; 1=Estimator                   --------     --------
C  IERRB      0=Normal ; 1=Estimator                   --------     --------
C  LATSTK_LFTSTP
C!            Lateral Left Stop                        Unknown      --------
C  LATSTK_RGTSTP
C!            Lateral Right Stop                       Unknown      --------
C  LONSTK_AFTSTP
C!            Longitudinal Aft Stop                    Unknown      --------
C  LONSTK_FWDSTP
C!            Longitudinal Forward Stop                Unknown      --------
C  MDES(1)    Desired Moment                           None         --------
C  M_ATT(1)   Attained Moment                          None         --------
C  M_TRIM(1)  Trimmed Moment                           None         --------
C  PCMD       Commanded roll rate                      --------     NLT
C  PDDES      P_Dot Desired                            RAD/SEC2     --------
C  QCMD       Cmd Pitch Rate                           DEG/SEC      --------
C  QDDES      Q_Dot Desired                            RAD/SEC2     --------
C  QS         Q_Bar*Area                               DEG/SEC      --------
C  RCMD       Cmd Yaw Rate                             DEG/SEC      --------
C  RDDES      R_Dot Desired                            RAD/SEC2     --------
C  RUDLFTSTP  Rudder Left Stop                         IN           +NL
C  RUDRGTSTP  Rudder Right Stop                        IN           +NL
C  SAT        Control allocation saturation            --------     NLT
C  SATA       Saturation for pitch moment being        --------     --------
C!            prioritized
C  SATB       Saturation for roll and yaw while pitch  --------     --------
C!            moment is being prioritized
C  SCALE      Restoring Scale Parameter                None         --------
C  UDA(1)     Ctl Vector from DA3                      None         --------
C  UDAP(1)    Controls that describe pitch moment in   --------     --------
C!            pitch prior.
C  UDAPP(1)   Controls that describe roll and yaw in   --------     --------
```

```
C!           pitch prior.
C  UMAX_DA(1) Max Framewise Limits                     None         --------
C  UMAX_GLOB(1)
C!           Max Global Limits                        None         --------
C  UMIN_DA(1) Min Framewise Limits                     None         --------
C  UMIN_GLOB(1)
C!           Min Global Limits                        None         --------
C  URATE_MAX(1)
C!           Max Rate Limits                          None         --------
C  URATE_MIN(1)
C!           Min Rate Limits                          None         --------
C  UTRIM(1)   Trimmed Ctl Deflections                  None         --------
C  UXMAX(1)   Max Ctl Deflections                      None         --------
C  UXMIN(1)   Min Ctl Deflections                      None         --------
C  U_GLOBAL(1)
C!           Global Ctl Deflections                   --------     --------
C  VD         Downward Velocity Toward Earth Ctr.      FT/SEC       DOWN
C  VE         Eastward Velocity Over Fixed Earth       FT/SEC       EAST
C  VN         Northward Velocity Over Earth            FT/SEC       NORTH
C
C-----------------------------------------------------------------
C LOCALS:
C
C  CONSTSPEED Description Not Available                --------     --------
C  CXT        X-Force Coefficient, aero+ctl            --------     --------
C  CYT        Y-Force Coefficient, aero+ctl            --------     --------
C  CZT        Z-Force Coefficient, aero+ctl            --------     --------
C  GCTH       Description Not Available                --------     --------
C  LAMBDA_P   Control law parameter                    --------     NLT
C  LAMBDA_Q   Control law parameter                    --------     NLT
C  LAMBDA_R   Control law parameter                    --------     NLT
C  LAMBDA_V   Control law parameter                    --------     NLT
C  LAMBDA_W   Control law parameter                    --------     NLT
C  QSB        Description Not Available                --------     --------
C  RMQS       Description Not Available                --------     --------
C  VDOTDES    Description Not Available                --------     --------
C  WDOTDES    Description Not Available                --------     --------
C  XSCALE     Further scaling for restoring 0-1        --------     --------
C
C-----------------------------------------------------------------
C OTHER LOCALS:
C
```

```
C I
C J
C K
C
C-----------------------------------------------------------------
      SUBROUTINE CONTROL

C-----------------------------------------------------------------
C DECLARATION SECTION
C
C-----------------------------------------------------------------
IMPLICIT NONE
C ** Parameters
C
C


C ** INPUTS:

LOGICAL*4 DIAGS, JOVDCK(5), TRIMMING
INTEGER*4 IMODE, IREST, M, NBI
REAL*4 CALPH, CHORD, CLL, CLM, CLN, CPHI, CPSI, CTHT, CX, CXU
REAL*4 CY, CYU, CZ, CZU, DADROOP, ALFA, DLHTD, DRHTD, DRUDL
REAL*4 DRUDR, DT, DXCG, FAX, FAY, FAZ, G, BDIAL_M, BDIAL_N
REAL*4 AREA, LATSTK, LONSTK, B_FORCE(3,20), B_MOM(3,20)
REAL*4 P(3,20), PB, PI, PLATRM, QB, QBAR, RB, RUDPED, SALPH
REAL*4 SPAN, SPHI, STHT, T11, T12, T13, T21, T22, T23, T31, T32
REAL*4 T33, TIME, BDIAL_L, UB, VB, VDW, VEW, VNW, VRW, WB, XIXX
REAL*4 XIXZ, XIYY, XIZZ, XMASS

C ** OUTPUTS:

INTEGER*4 IERR, IERRA, IERRB
REAL*4 DTH, D_MDES(3), ALFACMD, ALT, BETACMD, LATSTK_LFTSTP
REAL*4 LATSTK_RGTSTP, LONSTK_AFTSTP, LONSTK_FWDSTP, MDES(3)
REAL*4 M_ATT(3), M_TRIM(3), PCMD, PDDES, QCMD, QDDES, QS, RCMD
REAL*4 RDDES, RUDLFTSTP, RUDRGTSTP, SAT, SATA, SATB, SCALE
REAL*4 UDA(20), UDAP(20), UDAPP(20), UMAX_DA(20), UMAX_GLOB(20)
REAL*4 UMIN_DA(20), UMIN_GLOB(20), URATE_MAX(20), URATE_MIN(20)
REAL*4 UTRIM(20), UXMAX(20), UXMIN(20), U_GLOBAL(20), VD, VE
REAL*4 VN
```

```
C ** LOCALS:

REAL*4 CXT, CYT, CZT, GCTH, LAMBDA_P, LAMBDA_Q, LAMBDA_R
REAL*4 LAMBDA_V, LAMBDA_W, QSB, RMQS, VDOTDES, WDOTDES, XSCALE
LOGICAL*4 CONSTSPEED

C ** OTHER LOCALS:

BYTE ATMPAR, CONPAR, GEARPAR, KONSTANTS, LCLBUF, SIMPAR, VPI
BYTE XF18PA
LOGICAL*4 SCALED
INTEGER*4 I, J, K
REAL*4 PUMIN_DA(20), P_MDES(20), BDUM(3,20), CHMOM(3)
REAL*4 UCURR(20), VCONST, PDUM(20,3), VRAT, PUMAX_DA(20)

C-----------------------------------------------------------------
C COMMON SECTION
C
C-----------------------------------------------------------------
COMMON/ SHELL1 / CONPAR(424)
COMMON/ SHELL2 / SIMPAR(1024)
COMMON/ KONSTANT / KONSTANTS(40)
COMMON/ VPI / VPI(1324)

COMMON/ BK_CONTROL / LCLBUF(60)




COMMON/ XF18PA / XF18PA(2092)

COMMON/ XATMOS / ATMPAR(908)

COMMON/ XGEAR / GEARPAR(2312)

C-----------------------------------------------------------------
C EQUIVALENCE SECTION
C
C-----------------------------------------------------------------
C ** INPUTS:

EQUIVALENCE( SIMPAR(97), ALFA)
EQUIVALENCE( SIMPAR(557), AREA)
EQUIVALENCE( XF18PA(1921), BDIAL_L)
EQUIVALENCE( XF18PA(1925), BDIAL_M)
EQUIVALENCE( XF18PA(1929), BDIAL_N)
EQUIVALENCE( XF18PA(153), B_FORCE)
EQUIVALENCE( XF18PA(393), B_MOM)
EQUIVALENCE( SIMPAR(125), CALPH)
EQUIVALENCE( SIMPAR(565), CHORD)
EQUIVALENCE( SIMPAR(457), CLL)
EQUIVALENCE( SIMPAR(461), CLM)
EQUIVALENCE( SIMPAR(465), CLN)
EQUIVALENCE( SIMPAR(41), CPHI)
EQUIVALENCE( SIMPAR(57), CPSI)
EQUIVALENCE( SIMPAR(49), CTHT)
EQUIVALENCE( SIMPAR(381), CX)
EQUIVALENCE( XF18PA(129), CXU)
EQUIVALENCE( SIMPAR(385), CY)
EQUIVALENCE( XF18PA(133), CYU)
EQUIVALENCE( SIMPAR(389), CZ)
EQUIVALENCE( XF18PA(137), CZU)
EQUIVALENCE( XF18PA(1), DADROOP)
EQUIVALENCE( XF18PA(633), DIAGS)
EQUIVALENCE( XF18PA(1777), DLHTD)
EQUIVALENCE( XF18PA(1781), DRHTD)
EQUIVALENCE( XF18PA(5), DRUDL)
EQUIVALENCE( XF18PA(9), DRUDR)
EQUIVALENCE( CONPAR(145), DT)
EQUIVALENCE( SIMPAR(605), DXCG)
EQUIVALENCE( SIMPAR(393), FAX)
EQUIVALENCE( SIMPAR(397), FAY)
EQUIVALENCE( SIMPAR(401), FAZ)
EQUIVALENCE( KONSTANTS(1), G)
EQUIVALENCE( CONPAR(1), IMODE)
EQUIVALENCE( XF18PA(45), IREST)
EQUIVALENCE( GEARPAR(1941), JOVDCK(1))
EQUIVALENCE( XF18PA(885), LATSTK)
EQUIVALENCE( XF18PA(889), LONSTK)
EQUIVALENCE( XF18PA(637), M)
EQUIVALENCE( XF18PA(641), NBI)
EQUIVALENCE( XF18PA(645), P)
EQUIVALENCE( SIMPAR(145), PB)
EQUIVALENCE( KONSTANTS(25), PI)
EQUIVALENCE( XF18PA(1897), PLATRM)
EQUIVALENCE( SIMPAR(149), QB)
EQUIVALENCE( SIMPAR(549), QBAR)
EQUIVALENCE( SIMPAR(153), RB)
EQUIVALENCE( XF18PA(893), RUDPED)
EQUIVALENCE( SIMPAR(121), SALPH)
EQUIVALENCE( SIMPAR(561), SPAN)
EQUIVALENCE( SIMPAR(37), SPHI)
EQUIVALENCE( SIMPAR(45), STHT)
EQUIVALENCE( SIMPAR(61), T11)
EQUIVALENCE( SIMPAR(73), T12)
EQUIVALENCE( SIMPAR(85), T13)
EQUIVALENCE( SIMPAR(65), T21)
EQUIVALENCE( SIMPAR(77), T22)
EQUIVALENCE( SIMPAR(89), T23)
EQUIVALENCE( SIMPAR(69), T31)
EQUIVALENCE( SIMPAR(81), T32)
EQUIVALENCE( SIMPAR(93), T33)
EQUIVALENCE( SIMPAR(577), TIME)
EQUIVALENCE( CONPAR(161), TRIMMING)
EQUIVALENCE( SIMPAR(169), UB)
EQUIVALENCE( SIMPAR(173), VB)
EQUIVALENCE( ATMPAR(849), VDW)
EQUIVALENCE( ATMPAR(845), VEW)
EQUIVALENCE( ATMPAR(841), VNW)
EQUIVALENCE( SIMPAR(205), VRW)
EQUIVALENCE( SIMPAR(177), WB)
EQUIVALENCE( SIMPAR(345), XIXX)
EQUIVALENCE( SIMPAR(357), XIXZ)
EQUIVALENCE( SIMPAR(349), XIYY)
EQUIVALENCE( SIMPAR(353), XIZZ)
EQUIVALENCE( SIMPAR(369), XMASS)

C ** OUTPUTS:

EQUIVALENCE( XF18PA(33), ALFACMD)
EQUIVALENCE( SIMPAR(241), ALT)
EQUIVALENCE( XF18PA(37), BETACMD)
EQUIVALENCE( XF18PA(1773), DTH)
EQUIVALENCE( XF18PA(949), D_MDES(1))
EQUIVALENCE( XF18PA(41), IERR)
EQUIVALENCE( XF18PA(1793), IERRA)
EQUIVALENCE( XF18PA(1797), IERRB)
EQUIVALENCE( VPI(1013), LATSTK_LFTSTP)
EQUIVALENCE( VPI(1009), LATSTK_RGTSTP)
EQUIVALENCE( VPI(961), LONSTK_AFTSTP)
EQUIVALENCE( VPI(957), LONSTK_FWDSTP)
EQUIVALENCE( XF18PA(925), MDES(1))
EQUIVALENCE( XF18PA(937), M_ATT(1))
EQUIVALENCE( XF18PA(961), M_TRIM(1))
EQUIVALENCE( XF18PA(29), PCMD)
EQUIVALENCE( XF18PA(897), PDDES)
EQUIVALENCE( XF18PA(909), QCMD)
EQUIVALENCE( XF18PA(901), QDDES)
EQUIVALENCE( XF18PA(921), QS)
EQUIVALENCE( XF18PA(913), RCMD)
EQUIVALENCE( XF18PA(905), RDDES)
EQUIVALENCE( VPI(1053), RUDLFTSTP)
EQUIVALENCE( VPI(1057), RUDRGTSTP)
EQUIVALENCE( XF18PA(25), SAT)
EQUIVALENCE( XF18PA(1785), SATA)
EQUIVALENCE( XF18PA(1789), SATB)
EQUIVALENCE( XF18PA(917), SCALE)
EQUIVALENCE( XF18PA(973), UDA(1))
EQUIVALENCE( XF18PA(1933), UDAP(1))
EQUIVALENCE( XF18PA(2013), UDAPP(1))
EQUIVALENCE( XF18PA(1053), UMAX_DA(1))
EQUIVALENCE( XF18PA(1293), UMAX_GLOB(1))
EQUIVALENCE( XF18PA(1133), UMIN_DA(1))
EQUIVALENCE( XF18PA(1213), UMIN_GLOB(1))
EQUIVALENCE( XF18PA(1373), URATE_MAX(1))
EQUIVALENCE( XF18PA(1453), URATE_MIN(1))
EQUIVALENCE( XF18PA(1533), UTRIM(1))
EQUIVALENCE( XF18PA(1613), UXMAX(1))
EQUIVALENCE( XF18PA(1693), UXMIN(1))
EQUIVALENCE( XF18PA(49), U_GLOBAL(1))
EQUIVALENCE( SIMPAR(189), VD)
EQUIVALENCE( SIMPAR(185), VE)
EQUIVALENCE( SIMPAR(181), VN)

C ** LOCALS:

EQUIVALENCE( LCLBUF(1), CXT)
EQUIVALENCE( LCLBUF(5), CYT)
EQUIVALENCE( LCLBUF(9), CZT)
EQUIVALENCE( LCLBUF(13), GCTH)
EQUIVALENCE( LCLBUF(17), LAMBDA_P)
EQUIVALENCE( LCLBUF(21), LAMBDA_Q)
EQUIVALENCE( LCLBUF(25), LAMBDA_R)
EQUIVALENCE( LCLBUF(29), LAMBDA_V)
EQUIVALENCE( LCLBUF(33), LAMBDA_W)
EQUIVALENCE( LCLBUF(37), QSB)
EQUIVALENCE( LCLBUF(41), RMQS)
EQUIVALENCE( LCLBUF(45), VDOTDES)
EQUIVALENCE( LCLBUF(49), WDOTDES)
EQUIVALENCE( LCLBUF(53), XSCALE)
EQUIVALENCE( LCLBUF(57), CONSTSPEED)

C-----------------------------------------------------------------
C DATA SECTION
C

      DATA M / 20 /
      DATA NBI / 9 /

C B                    /   Roll, Pitch, Yaw    /
C             Aileron Cl changed sign for left and right.


      DATA (B_MOM(I,1),I=1,3)/ 0.52100e-03, -6.6900e-03,  0.23700e-03/
      DATA (B_MOM(I,2),I=1,3)/-0.52200e-03, -6.7000e-03,  -0.23800e-03/

      DATA (B_MOM(I,3),I=1,3)/ -0.74500e-03,-5.6600e-05,-0.27600e-03/
      DATA (B_MOM(I,4),I=1,3)/ 0.74600e-03, -5.6700e-05,  0.27700e-03/

      DATA (B_MOM(I,5),I=1,3)/ 0.25300e-03,  0.,   -1.17000e-03/
      DATA (B_MOM(I,6),I=1,3)/ 0.25400e-03,  0.,   -1.18000e-03/

      DATA (B_MOM(I,7),I=1,3)/ 0.42100e-03, -6.5900e-03, 0.13700e-03/
      DATA (B_MOM(I,8),I=1,3)/-0.42200e-03, -6.6000e-03, -0.13800e-03/

      DATA (B_MOM(I,9),I=1,3)/ -0.64500e-03, -5.5600e-05, -0.17600e-03/
      DATA (B_MOM(I,10),I=1,3)/0.64600e-03, -5.5700e-05, 0.17700e-03/

      DATA (B_MOM(I,11),I=1,3)/ 0.15300e-03,  0.,   -1.07000e-03/
      DATA (B_MOM(I,12),I=1,3)/ 0.15400e-03,  0.,   -1.08000e-03/
      DATA (B_MOM(I,13),I=1,3)/ 0.42100e-03, -6.5900e-03,  0.13700e-03/
      DATA (B_MOM(I,14),I=1,3)/-0.42200e-03, -6.6000e-03,  -0.13800e-03/

      DATA (B_MOM(I,15),I=1,3)/ -0.64500e-03, -5.5600e-05,  -0.17600e-03/
      DATA (B_MOM(I,16),I=1,3)/0.64600e-03, -5.5700e-05, 0.17700e-03/

      DATA (B_MOM(I,17),I=1,3)/ 0.15300e-03,  0.,  -1.07000e-03/
      DATA (B_MOM(I,18),I=1,3)/ 0.15400e-03,  0.,   -1.08000e-03/

      DATA (B_MOM(I,19),I=1,3)/ -0.64500e-03, -5.5600e-05,  -0.17600e-03/
      DATA (B_MOM(I,20),I=1,3)/0.64600e-03, -5.5700e-05, 0.17700e-03/



C-------------Remove rate limits for globalization----------------

      DATA (UMIN_GLOB(I),I=1,20)/-24., -24.,  -25.,  -25., -30., -30.,
     &  -24., -24., -25., -25., -30., -30., -25., -25., -25., -25., -24.
     & ,-24., -30., -30./
      DATA (UMAX_GLOB(I),I=1,20)/ 10.5, 10.5,  45.,   45.,  30.,  30.,
     &  10.5, 10.5,  45.,   45.,  30.,  30., 45., 45., 45., 45., 10.5,
     &  10.5, 30., 30./


C-----------------------------------------------------------------
C INITIALIZATION SECTION
C
C-----------------------------------------------------------------

      IF (IMODE.EQ.-2) THEN
      LAMBDA_P = -2.
      LAMBDA_Q = -4.
      LAMBDA_R = -4.
      LAMBDA_W = -3.
      LAMBDA_V = -3.

      XSCALE = 0.2

U_GLOBAL(1) = 0.
        U_GLOBAL(2) = 0.
        U_GLOBAL(3) = 0.
        U_GLOBAL(4) = 0.
        U_GLOBAL(5) = 0.
        U_GLOBAL(6) = 0.
        U_GLOBAL(7) = 0.
        U_GLOBAL(8) = 0.
        U_GLOBAL(9) = 0.
        U_GLOBAL(10) = 0.
        U_GLOBAL(11) = 0.
        U_GLOBAL(12) = 0.
        U_GLOBAL(13) = 0.
        U_GLOBAL(14) = 0.
        U_GLOBAL(15) = 0.
        U_GLOBAL(16) = 0.
        U_GLOBAL(17) = 0.
        U_GLOBAL(18) = 0.
        U_GLOBAL(19) = 0.
        U_GLOBAL(20) = 0.
        BDIAL_L=1.
        BDIAL_M=1.
        BDIAL_N=1.
      DO I=1,3
   DO J=1,M
      BDUM(I,J)=B_MOM(I,J)
   ENDDO
ENDDO

      DO I=1,M
   DO J=1,3
      PDUM(I,J)=P(I,J)
   ENDDO
      ENDDO

      ENDIF


C-----------------------------------------------------------------
C RESET SECTION
C
C-----------------------------------------------------------------
      IF (IMODE.LE.0) THEN
  U_GLOBAL(1) = 0.
        U_GLOBAL(2) = 0.
        U_GLOBAL(3) = 0.
        U_GLOBAL(4) = 0.
        U_GLOBAL(5) = 0.
        U_GLOBAL(6) = 0.
        U_GLOBAL(7) = 0.
        U_GLOBAL(8) = 0.
        U_GLOBAL(9) = 0.
        U_GLOBAL(10) = 0.
        U_GLOBAL(11) = 0.
        U_GLOBAL(12) = 0.
        U_GLOBAL(13) = 0.
        U_GLOBAL(14) = 0.
        U_GLOBAL(15) = 0.
        U_GLOBAL(16) = 0.
        U_GLOBAL(17) = 0.
        U_GLOBAL(18) = 0.
        U_GLOBAL(19) = 0.
        U_GLOBAL(20) = 0.


      DO I=1,3
   DO J=1,M
      B_MOM(I,J)=BDUM(I,J)
   ENDDO
ENDDO

      DO I=1,M
   DO J=1,3
      P(I,J)=PDUM(I,J)
   ENDDO
ENDDO

        SCALED=.FALSE.
 IF (TRIMMING) THEN
  DRHTD = DLHTD
          U_GLOBAL(1) = 0.
          U_GLOBAL(2) = 0.
          U_GLOBAL(3) = 0.
          U_GLOBAL(4) = 0.
          U_GLOBAL(5) = 0.
          U_GLOBAL(6) = 0.
          U_GLOBAL(7) = 0.
          U_GLOBAL(8) = 0.
          U_GLOBAL(9) = 0.
          U_GLOBAL(10) = 0.
          U_GLOBAL(11) = 0.
          U_GLOBAL(12) = 0.
          U_GLOBAL(13) = 0.
          U_GLOBAL(14) = 0.
          U_GLOBAL(15) = 0.
          U_GLOBAL(16) = 0.
          U_GLOBAL(17) = 0.
          U_GLOBAL(18) = 0.
          U_GLOBAL(19) = 0.
          U_GLOBAL(20) = 0.
        GOTO 194
        ENDIF
        CONSTSPEED=.TRUE.
        ENDIF

C-----------------------------------------------------------------
C RUN SECTION
C
C-----------------------------------------------------------------
C       Keeping Velocity constant

        IF (JOVDCK(1).GT.0) CONSTSPEED = .FALSE.

        IF (CONSTSPEED) THEN

        VCONST=231.5

VRAT=VCONST/VRW

UB=UB*VRAT
VB=VB*VRAT
WB=WB*VRAT

C       TRANSFORM INTO INERTIAL REFERENCE FRAME


   VN = T11*UB + T21*VB + T31*WB + VNW
   VE = T12*UB + T22*VB + T32*WB + VEW
   VD = T13*UB + T23*VB + T33*WB + VDW
```

```
        ENDIF



C        Scaling the B Matrix to assure unattainable moments

      IF (.NOT. SCALED) THEN
      SCALED=.TRUE.
   DO J=1,M
      B_MOM(1,J)=BDIAL_L*B_MOM(1,J)
      B_MOM(2,J)=BDIAL_M*B_MOM(2,J)
      B_MOM(3,J)=BDIAL_N*B_MOM(3,J)
   ENDDO
      ENDIF
C
C Limit Stick Positions - Stops are in constants.f
        LONSTK = MIN(MAX( LONSTK, LONSTK_FWDSTP), LONSTK_AFTSTP )
        LATSTK = MIN(MAX( LATSTK, LATSTK_LFTSTP), LATSTK_RGTSTP )
RUDPED = MIN(MAX( RUDPED, RUDLFTSTP), RUDRGTSTP )

C Determine pilot commands
      ALFACMD = (8.1+20.0*LONSTK/(LONSTK_AFTSTP-LONSTK_FWDSTP))*PI/180.
      PCMD    =     180.0*LATSTK/(LATSTK_RGTSTP-LATSTK_LFTSTP)*PI/180.
      BETACMD =     (10.0*RUDPED/(RUDLFTSTP-RUDRGTSTP))*PI/180.
C
      CXT =   CX + CXU
      CYT =   CY + CYU
      CZT =   CZ + CZU


C These controls correspond to the last time step -
         UTRIM(1) = DLHTD
         UTRIM(2) = DRHTD
         UTRIM(3) = DADROOP
         UTRIM(4) = DADROOP
         UTRIM(5) = DRUDL
         UTRIM(6) = DRUDR
         UTRIM(7) = 0.
         UTRIM(8) = 0.
         UTRIM(9) = 0.
         UTRIM(10) = 0.
         UTRIM(11) = 0.
         UTRIM(12) = 0.
         UTRIM(13) = 0.
         UTRIM(14) = 0.
         UTRIM(15) = 0.
         UTRIM(16) = 0.
         UTRIM(17) = 0.
         UTRIM(18) = 0.
         UTRIM(19) = 0.
         UTRIM(20) = 0.

      DO I=1,3
        M_TRIM(I) = 0.
        DO J=1,M
          M_TRIM(I) = M_TRIM(I)+B_MOM(I,J)*UTRIM(J)
        ENDDO
      ENDDO

C  The total amount of physical control left over.

      DO I = 1,M


        UMIN_DA(I) =
     &            UMIN_GLOB(I) - UTRIM(I)

        UMAX_DA(I) =
     &            UMAX_GLOB(I) - UTRIM(I)

      ENDDO



      QS = QBAR*AREA
      QSB = QS*SPAN
      RMQS = QS/XMASS
      GCTH = G*CTHT

      VDOTDES = LAMBDA_V*(VB-VRW*SIN(BETACMD))
      WDOTDES = LAMBDA_W*(WB- UB*TAN(ALFACMD))
      QCMD  = (WDOTDES + PB*VB - GCTH * CPHI - RMQS * CZT)/UB
      RCMD  = (PB*WB - VDOTDES + GCTH * SPHI + RMQS * CYT)/UB

      PDDES = LAMBDA_P*(PB-PCMD)
      QDDES = LAMBDA_Q*(QB-QCMD)
      RDDES = LAMBDA_R*(RB-RCMD)


!Put CG offset corrections into dynamic inverter


  MDES(1) =   -CLL +
     & ( XIXX*PDDES-XIXZ*RDDES + (XIZZ - XIYY)*QB*RB - XIXZ*PB*QB)
     &  /QSB


!  FAZ * DXCG is the aircraft dynamics offset
!  CZU * DXCG is controls (B matrix) offset

  MDES(2) = -CLM - (FAZ * DXCG)/QS/CHORD +
     & (XIYY*QDDES + (XIXX - XIZZ)*PB*RB + XIXZ*(PB*PB - RB*RB))
     &  /(QS*CHORD)


!  FAY * DXCG is the aircraft dynamics offset
!  CYU * DXCG is controls (B matrix) offset

  MDES(3) =   -CLN + (FAY * DXCG)/QSB +
     & (-XIXZ*PDDES+XIZZ*RDDES + (XIYY - XIXX)*PB*QB + XIXZ*QB*RB)
     &  /QSB


      DO I=1,3
        M_ATT(I)=0.
        DO J=1,M
          M_ATT(I)=M_ATT(I)+B_MOM(I,J)*U_GLOBAL(J)!/1000.
        ENDDO
         D_MDES(I)   = 1000.*MDES(I) /1000.  !1000*
      ENDDO

IF (IREST.EQ.0) THEN

  CALL DA3(UDA, SAT, IERR,
     &     B_MOM, D_MDES, UMIN_DA, UMAX_DA, M, NBI, TIME, DIAGS)

        DO I=1,M
           U_GLOBAL(I) = UDA(I)

        ENDDO


C      We are prioritizing the Pitch Axis in this routine.

      ELSEIF (IREST.EQ.1) THEN

          P_MDES(1)=0.
  P_MDES(2)=D_MDES(2)   ! This will solve for desired pitch
  P_MDES(3)=0.

  CALL DA3(UDAP, SATA, IERRA,
     &     B_MOM, P_MDES, UMIN_DA, UMAX_DA, M, NBI, TIME, DIAGS)


C       Recenter the origin at the position of desired pitch

       IF(SATA .LT. 1) THEN
          DO I=1,M

     PUMIN_DA(I) = UMIN_DA(I) - UDAP(I)
     PUMAX_DA(I) = UMAX_DA(I) - UDAP(I)

  ENDDO

C        Solve for the other Moment directions now, at new origin

          P_MDES(1)=D_MDES(1)
  P_MDES(2)=0.
  P_MDES(3)=D_MDES(3)

  CALL DA3(UDAPP, SATB, IERRB,
     &     B_MOM, P_MDES, PUMIN_DA, PUMAX_DA, M, NBI, TIME, DIAGS)

          ELSE

  DO I=1,M
```

```
     UDAPP(I)=0.
  ENDDO

       ENDIF

      DO I=1,M
   U_GLOBAL(I)=UDAP(I) + UDAPP(I)
      ENDDO


      DO I=1,M
        IF (U_GLOBAL(I).LT.(UMIN_GLOB(I)-UTRIM(I))) U_GLOBAL(I) =
     &                                                     UMIN_DA(I)
        IF (U_GLOBAL(I).GT.(UMAX_GLOB(I)-UTRIM(I))) U_GLOBAL(I) =
     &                                                     UMAX_DA(I)
      ENDDO


 194  CONTINUE
C
C-----------------------------------------------------------------
C END OF CONTROL MODULE
C
C-----------------------------------------------------------------
      RETURN
      END
```

Vita

Mark Nelson was born in the summer of 1977 to David and Terry Nelson in Fairfax, Virginia. He spent most of his younger days involved in the community with sports and clubs. In time, Mark grew wise in the ways of math and science. In 1995 he moved into the peaceful mountains of southwestern Virginia where he spent four years obtaining his dual Bachelor of Science degrees in Aerospace Engineering and Ocean Engineering. The next two years of his life were spent fine tuning his knowledge of Aerospace Engineering in the Flight Simulation Labratory where he was surrounded by only the greatest of scholars. After two years of hard work intermittent with times of relaxation and leisure activities, Mark was rewarded with a Master of Science in Aerospace Engineering during the summer of 2001.