AAC/TM–2009–01

Adaptive Control of Generic Transport Model with Modelled Failures

Travis E. Gibson and Anuradha M. Annaswamy
Massachusetts Institute of Technology, Cambridge, Massachusetts

Sean P. Kenny
NASA Langley Research Center, Hampton VA

Active Adaptive Controls Laboratory
Massachusetts Institute of Technology

77 Massachusetts Avenue, Cambridge, MA 02139

January 2009

Abstract

This paper proposes an adaptive controller for a generic transport vehicle subject to center-of-gravity uncertainty and time-delays. The adaptive control architecture is based on a linearized model of the the aircraft dynamics. The adaptive algorithm specifically accommodates for actuator saturation and augments a baseline controller predicated on sequential loop closing techniques and integral anti–windup logic. The adaptive design is validated using the high-fidelity GTM SIMULNIK code developed at NASA Langley. The resilience of the adaptive algorithm is compared to that of the baseline controller for the uncertainties mentioned above by monitoring the structural loading and command tracking performance of the two controllers.

Contents

List of Figures

List of Tables

Preface

This study contains research conducted during an internship in the 2008 *Langley Aerospace Research Summer Scholar* (LARSS) program. The high fidelity SIMULINK model of the *Generic Transport Model* (GTM) was used for the testing and validation of two controllers. One controller was an in–house scheme designed by the Langley *Dynamic Systems and Controls* (DSC) group, and the other, an adaptive algorithm stemming from recent work in the *Active–Adaptive Controls* (AAC) lab at MIT. The in–house controller was designed using sequential loop closing techniques in order to design *Stability Augmented Systems* (SAS) and *Command Augmented Systems* (CAS). The adaptive controller then augments to the baseline, in–house, controller. In addition to the equations of motion for the aircraft, the GTM SIMULINK model also incorporates: actuator dynamics, actuator saturation limits, sensor dynamics, telemetry time delays and data processing time delays. The adaptive algorithm specifically accounts for the actuator saturation effects while remaining robust with respect to variations in telemetry and data processing time delays. Two different studies were conducted comparing the robustness and performance of the two control structures. The first study analyzed the g–force at the nose of the aircraft given a step command in the angle of attack while incorporating uncertain time delays in telemetry and uncertain *Center of Gravity* (CG) location. The second study analyzed the tracking performance of the control structures given a wave train in the angle of attack command while also having uncertainties in telemetry and CG location. The adaptive algorithm had similar g–loading characteristic to the nominal controller, and outperformed the nominal controller by a significant margin in the command tracking study.

Nomenclature

Acronyms and Abbreviations

AAC	Active–Adaptive Controls
ALT	Altitude
CAS	Command Augmented System
CG	Center of Gravity
DSC	Dynamic Systems and Controls
LARSS	Langley Aerospace Research Summer Scholar
LAT	Latitude
LON	Longitude
MAC	Mean Aerodynamic Cord
SAS	Stability Augmented System

Symbols

$\mathbf{A}$	Augmented state jacobian
$\mathbf{A_m}$	Reference model state jacobian
$\mathbf{A_p}$	Plant state jacobian
$\mathbf{B}$	Augmented input jacobian
$\mathbf{B_p}$	Plant input jacobian
$\mathbf{e}$	Reference model error
e_α	AOA error
$\mathbf{e_a}$	Augmented error
$\mathbf{e_\Delta}$	Saturation defect error
$\mathbf{F}$	Force vector
F_x	Force in x–direction
F_y	Force in y–direction
F_z	Force in z–direction
g	Gravitational constant
$\mathbf{H}$	Output selection matrix
$\mathbf{H_I^B}$	Transformation matrix, inertial earth frame to wind
I	Moment of inertia, Identity matrix
$\mathbf{K}$	Nominal feedback gain
$\mathbf{M}$	Moment vector
M_x	Moment in x–direction
M_y	Moment in y–direction
M_z	Moment in z–direction
p	Roll rate
$\mathbf{P}$	Solution to Lyapunov equation
q	Pitch rate
$\mathbf{Q}$	Rate parameter in Lyapunov equation
r	Yaw Rate
R_e	Integral error saturation function

$\mathbf{R_S}$	Rectangular saturation function
S_i	ith selection matrix
t	Time
u	Velocity in x–direction, input vector
$\mathbf{u}$	Perturbation input vector
$\mathbf{u_a}$	Adaptive input vector
$\mathbf{u_n}$	Nominal input vector
$\mathbf{u_p}$	Pilot input vector
$\mathbf{U}$	Input vector
v	Velocity in y–direction
V_a	Air speed
w	Velocity z–direction
$\mathbf{w}$	wind velocity vector
$\mathbf{x}$	Perturbation state vector
$\mathbf{x_p}$	Plant perturbation state vector
$\mathbf{X}$	State vector
α	Angle of attack
β	Side slip angle
δ	Control surface deflection angle
Δl	CG location movement in the units ft
$\boldsymbol{\epsilon}$	Trim state error vector
λ	Dimensionless CG uncertainty parameter
$\boldsymbol{\gamma}$	Adaptive rate parameter
ϕ	Euler angle x–direction
θ	Euler angle y–direction
$\boldsymbol{\theta}$	Adaptive feedback gain
$\boldsymbol{\sigma}$	Adaptive damping term
τ	Time
ψ	Euler angle z–direction

Subscript

0	Trim value
a	Aileron, adaptive
A	Wind axes
B	Bottom
cmd	Command
e	Elevator
F	Flap
I	Inboard
L	Left
O	Outboard
p	Plant, pilot
r	Rudder

R	Right
sp	Spoiler
st	Stabilizer
T	Top
w	Post washout filter state value
Δ	Perturbation from saturation limit

<u>Prefix</u>

Δ	Denotes perturbation from trim value

<u>Vector Notation</u>

There are an extensive number of variables introduced in this report. Great care was taken in developing appropriate nomenclature. Vector notation is used extensively as well. When bold font variables are introduced it is assumed that the variable contains more than one element. For instance, if a vector $\boldsymbol{\xi} \in \Re^n$ is introduced, then it is assumed that,

$$\boldsymbol{\xi} \triangleq \begin{bmatrix} \xi_1 \\ \xi_2 \\ \vdots \\ \xi_n \end{bmatrix}.$$

The same is assumed when dealing with matrices. Consider the following matrix $\mathbf{G} \in \mathbb{R}^{2\times 2}$, where

$$\mathbf{G} \triangleq \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix}.$$

Notice that each component of the 2×2 matrix is scalar and therefore not bold. Now, consider the partitioning of a different matrix, $\mathbf{Y} \in \mathbb{R}^{4\times 4}$

$$\mathbf{Y} \triangleq \begin{bmatrix} \mathbf{Y}_{11} & \mathbf{Y}_{12} \\ \mathbf{Y}_{21} & \mathbf{Y}_{22} \end{bmatrix}.$$

Notice that each of the 4 partitions of $\mathbf{Y}$ is still an element of $\mathbb{R}^{2\times 2}$ and as so is still bold when presented. This notation was recognized in works by Robert F. Stengel from Princeton, and several other prominent authors.

In the event that a vector is defined with a subscript in the name, such as $\mathbf{F_p}$, then it is assumed that the subscript in being part of the variable definition will be bold as well. When selecting a certain value within the vector, the usual integer notation is used as displayed in the following example.

$$\mathbf{F_p} \triangleq \begin{bmatrix} F_{p\,1} \\ F_{p_2} \\ \vdots \\ F_{p_n} \end{bmatrix}.$$

This notation will be avoided at all cost as sequential sub–indexing is not clean.

Indexing by variable will also be performed when necessary. Observe the following superscript and subscript declarations for selecting rows and columns. Let $\mathbf{u} \in \mathbb{R}^3$, $\mathbf{K} \in \mathbb{R}^{3\times 3}$, and $\boldsymbol{\xi} \in \mathbb{R}^3$ with the following relation,

$$\mathbf{u} \triangleq \begin{bmatrix} u_a \\ u_b \\ u_c \end{bmatrix} = \mathbf{K}\boldsymbol{\xi} \triangleq \begin{bmatrix} \leftarrow \mathbf{K}^a \rightarrow \\ \leftarrow \mathbf{K}^b \rightarrow \\ \leftarrow \mathbf{K}^c \rightarrow \end{bmatrix} \begin{bmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \end{bmatrix} \triangleq \begin{bmatrix} K^a_{\xi_1} & K^a_{\xi_2} & K^a_{\xi_3} \\ K^b_{\xi_1} & K^b_{\xi_2} & K^b_{\xi_3} \\ K^c_{\xi_1} & K^c_{\xi_2} & K^c_{\xi_3} \end{bmatrix} \begin{bmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \end{bmatrix}.$$

1 Introduction

Over the past several years NASA LARC has been developing the *Generic Transport Model* (GTM) under the Airborne Subscale Transport Aircraft Research(AirSTAR) project. The GTM is a 5.5% dynamically scaled transport vehicle that is flown wirelessly by radio frequency. A picture of the *Generic Transport Model–Turbine 1* (GTM–T1) can be seen in Figure 1.

Figure 1. Generic Transport Model Turbine–1

In addition to the physical GTM aircraft there exists a high fidelity Simulink based model. The Simulnk model of the aircraft is referred to as (GTM–S). The GTM-S environment contains the following:

- Experimentally obtained extended aero data set for high AOA and spins
- Sensor noise and sensor bias from flight data
- Telemetry uplink and downlink time delays
- Actuator dynamics with rate and position limits
- Sensor dynamics along with ADC and DAC latencies and quanitization
- Experimentally obtained aero data for damaged aircraft, i.e. missing tail section [1, 2]

The GTM–S was the test bed for this work.

In flight validation is always the crucial final hurdle for control algorithms. Most algorithms that depart in any way from classical control theory never make it to in–flight testing. This reluctance to experiment with nonlinear or robust control algorithms stems from a well known and tragic history.

The biggest set back to nonlinear control applications in avionics came from the X–15 program, the X–15 plane is pictured in Figure 2. The program began in 1954 and 199 test flights were performed. The X–15 was designed with several research goals in mind. The major goal was to understand the effects of high speed atmosphere reentry. In the process the X–15 broke altitude and

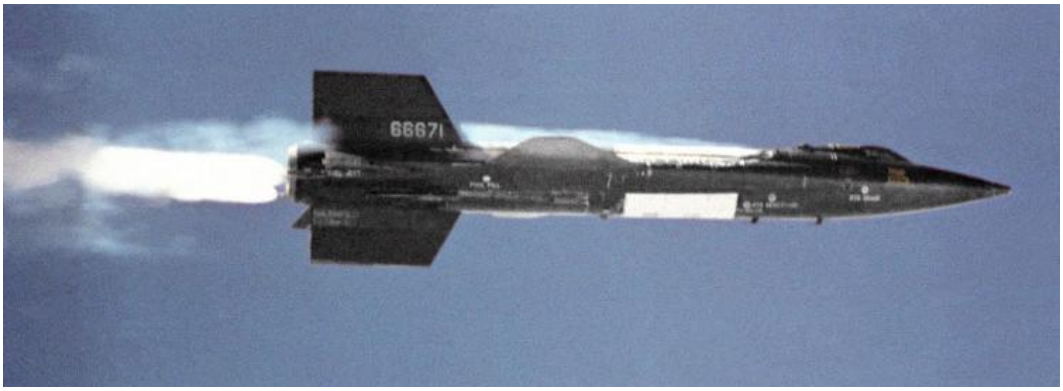

Figure 2. X-15

speed records with flights higher than 300,000 ft and at speeds in excess of Mach 6. [3]

The characteristic of the X–15 program most related to this work is the implementation of an adaptive algorithm in aircraft stabilization. The adaptive control algorithm implemented in the X–15 project was designed by Minneapolis Honeywell Corp. It was referred to as a "Self–Adaptive" control system. The Self–Adaptive control system had a variable feedback gain on euler rates in order to maintain attitude stability in flight. The variable feedback gains were adjusted so as to minimize the error between the actual attitude of the aircraft and some ideal reference attitude. The adaptive controller decreased the tuning time necessary to gain schedule a classic controller over the entire flight envelope. [4, 5]

The adaptive algorithm from Honeywell was truly ahead of its time in implementation. However, it lacked the mathematical tools necessary to prove stability in a rigorous manner and relied on "rule of thumb" ideologies instead. This ended in tragedy however. On November 15, 1967 test flight 191 of 199 crashed above Delamar Dry Lake. [3] Unbeknownst to the pilot there was an electrical malfunction and the aircraft began to deviate from the desired trajectory and a gross side–slip angle was building. Once off by 15° the pilot corrected for the mistake, then the aircraft drifted again, after several seconds of pilot corrections the aircraft interred a Mach 5 spin at an altitude of 230,000 ft. [3, 6] As the aircraft fell into more dense air it broke apart killing the pilot, Mike Adams. This crash put a holt on all adaptive control implementation for several decades, and not until recently has the idea been revisited.

40 years have passed now, and several advances have been made in control theory and time domain analysis. The main theoretical tool of adaptive algorithms stems from work by Lyapunov in 1892 [7] however these techniques were not translated into English until the 40's. [8] By the 1960's several control theorist were able to construct rigorous methods for stability proofs, some prominent figures were Bellman [9], LaSalle [10], Coppel [11], Hahn [12, 13],

Krasovskii [14] and Anderson [15]. [16]

In this work an adaptive control algorithm is introduced which is then validated on the GTM SIMULINK model. The adaptive algorithm is designed to augment a nominal baseline controller. The nominal baseline controller is comprised of a *Control Augmented System* (CAS) for the pitch axis and a *Stability Augmented System* (SAS) for both the yaw and roll axes. The adaptive architecture explicitly accounts for actuator saturation limits and the anti–windup logic present in the CAS. The specfifc algorithm used in this report stems from work first introduced by Karason and Annaswamy in [17] with multi dimensional extensions by Schwager and Jang in [18–23]. The robustness of the above algorithm with respect to uncertain center of gravity location and time delays is studied here in.

2 Dynamics

In this section the aerodynamics, actuator dynamics, actuator saturation, and sensor dynamics. of the GTM-S are explored.

2.1 Dynamics of Aircraft Motion

The standard conservation equations [24] for a flat–earth symmetric aircraft describe the dynamics of u, v, and w, the body-fixed aircraft velocities; p, q, and r, the roll, pitch, and yaw rates; as well as the Euler angles ϕ, θ, and ψ. The aircraft's equations of motion are given by:

$$\dot{u} = \frac{F_x}{m} - g\sin\theta - qw + rv \tag{1}$$

$$\dot{v} = \frac{F_y}{m} + g\cos\theta\sin\phi - ru + pw \tag{2}$$

$$\dot{w} = \frac{F_z}{m} + g\cos\theta\cos\phi + qu - pv \tag{3}$$

$$\dot{p} = \frac{I_{zz}}{I_D}\left[M_x + I_{xz}pq - (I_{zz} - I_{yy})\,qr\right] + \frac{I_{xz}}{I_D}\left[M_z - I_{xz}qr - (I_{yy} - I_{xx})\,pq\right] \tag{4}$$

$$\dot{q} = \frac{1}{I_{yy}}\left[M_y - (I_{xx} - I_{zz})\,pr - I_{xz}\left(p^2 - r^2\right)\right] \tag{5}$$

$$\dot{r} = \frac{I_{xz}}{I_D}\left[M_x + I_{xz}pq - (I_{zz} - I_{yy})\,qr\right] + \frac{I_{xx}}{I_D}\left[M_z - I_{xz}qr - (I_{yy} - I_{xx})\,pq\right] \tag{6}$$

$$\dot{\phi} = p + q\sin\phi\tan\theta + r\cos\phi\tan\theta \tag{7}$$

$$\dot{\theta} = q\cos\phi - r\sin\phi \tag{8}$$

$$\dot{\psi} = (qsin\phi + r\cos\phi)\sec\theta, \tag{9}$$

where $I_D = I_{xx}I_{zz} - I_{xz}^2$. The aerodynamic forces are represented as $\mathbf{F} \triangleq [F_x\ F_y\ F_z]^T$ and the aerodynamic moments as $\mathbf{M} \triangleq [M_x\ M_y\ M_z]^T$. The gross vehicle mass is denoted by m and the components of the inertial tensor are listed as I_{xx}, I_{yy}, I_{zz} and I_{xz}.

The following navigation equations determine x and y, the positions of the aircraft in the north and east directions respectively, as well as the altitude $h{=}{-}z$:

$$\begin{aligned}\dot{x} =& u\cos\theta\cos\psi + v(-\cos\phi\sin\psi + \sin\phi\sin\theta\cos\psi) + \\ & w(\sin\phi\sin\psi + \cos\phi\sin\theta\cos\psi),\end{aligned} \tag{10}$$

$$\begin{aligned}\dot{y} =& u\cos\theta\sin\psi + v(\cos\phi\cos\psi + \sin\phi\sin\theta\sin\psi) + \\ & w(-\sin\phi\cos\psi + \cos\phi\sin\theta\sin\psi),\end{aligned} \tag{11}$$

$$\dot{z} = -u\sin\theta + v\sin\phi\cos\theta + w\cos\phi\cos\theta. \tag{12}$$

It is often convenient to replace the body-fixed velocities with the actual velocities by accounting for wind. Letting the subindex $(\cdot)_A$ represent components of the velocity vector corrected for wind and $\mathbf{w}$ representing the wind velocity vector, the following relation can be defined.

$$\begin{bmatrix} u_A \\ v_A \\ w_A \end{bmatrix} = \begin{bmatrix} u \\ v \\ w \end{bmatrix} - \mathbf{H}_\mathbf{I}^\mathbf{B}\mathbf{w}, \tag{13}$$

where,

$$\mathbf{H}_\mathbf{I}^\mathbf{B} = \begin{bmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ (-\cos\phi\sin\psi + \sin\phi\sin\theta\cos\psi) & (\cos\phi\cos\psi + \sin\phi\sin\theta\sin\psi) & \sin\phi\cos\theta \\ (\sin\phi\sin\psi + \cos\phi\sin\theta\cos\psi) & (-\sin\phi\cos\psi + \cos\phi\sin\theta\sin\psi) & \cos\phi\cos\theta \end{bmatrix}. \tag{14}$$

Once the wind adjusted velocity is attained, then the total velocity of the aircraft, V_A, the angle of attack, α, and the side slip angle, β, can be defined as,

$$V_A = \sqrt{u_A^2 + v_A^2 + w_A^2}, \tag{15}$$

$$\tan\alpha = \frac{w_A}{u_A}, \tag{16}$$

$$\sin\beta = \frac{v_A}{V_A}. \tag{17}$$

2.2 Control Authority

The forces and moments represented by $\mathbf{F}$ and $\mathbf{M}$ are generated by the varied air pressure across different surface of the aircraft and from the the thrust generated by the engines. In this study the engine was not given as a control input and therefore will not be controlled. There will simply be a constant throttle setting for the engines with later work considering throttle control in more detail. The surfaces of the aircraft are adjusted in order to impart moments and forces on the aircraft and are referred to as *control surfaces*. A visual representation of the control surfaces on the GTM are shown in Figure 13.

There are 6 distinct types of control surfaces:

- *Elevator*–Located on the rear of the empennage and controls the pitch of the aircraft.
- *Aileron*–Located outside of the flaps and controls roll.

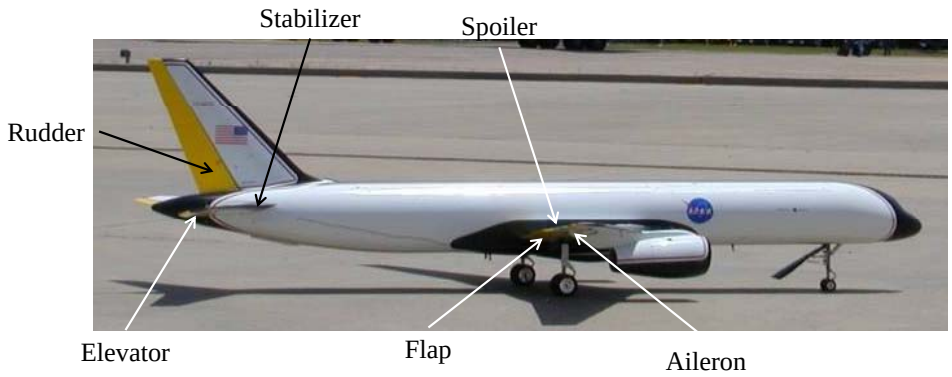

Figure 3. Aircraft control surfaces

- *Rudder*–Vertical controller structure on the rear of the empennage, and controls yaw.

- *Flaps*–Located inboard on the underside of the wings. When extended out and down the coefficient of lift for the aircraft is increased. Necessary for low–speed maneuvers such as take off and landing.

- *Stabilizer*–Located on the leading edge of the empennage and is usually adjusted for various trim conditions.

- *Spoilers*– Located on the top of the aircraft wing. These control surfaces can greatly affect lift. When deployed straight up the aircraft can loose altitude without increasing speed. Spoilers are essentially air brakes.

Within the 6 different types of control surfaces there may be several independent components of each. For instance there are 4 independent flap control surfaces on the GTM. There are left–inboard and left–outboard as well as right–inboard and right–outboard flaps. This leads to an important topic relating to symmetry and control authority. On the GTM there are actually 17 distinct control surfaces that can be controlled independently, consider the

following defenition,

$$\mathbf{U}_{\text{control surface}} \triangleq \begin{bmatrix} \delta_{a_L} \\ \delta_{a_R} \\ \delta_{sp_{LO}} \\ \delta_{sp_{LI}} \\ \delta_{sp_{RO}} \\ \delta_{sp_{RI}} \\ \delta_{e_{LO}} \\ \delta_{e_{LI}} \\ \delta_{e_{RO}} \\ \delta_{e_{RI}} \\ \delta_{r_T} \\ \delta_{r_B} \\ \delta_{st} \\ \delta_{F_{LO}} \\ \delta_{F_{LI}} \\ \delta_{F_{RO}} \\ \delta_{F_{RI}} \end{bmatrix} \begin{array}{l} \text{: L aileron} \\ \text{: R aileron} \\ \text{: L out spoiler} \\ \text{: L in spoiler} \\ \text{: R out spoiler} \\ \text{: R in spoiler} \\ \text{: L out elevator} \\ \text{: L in elevator} \\ \text{: R out elevator} \\ \text{: R in elevator} \\ \text{: top rudder} \\ \text{: bottom rudder} \\ \text{: stabilizer} \\ \text{: L out flap} \\ \text{: L in flap} \\ \text{: R out flap} \\ \text{: R in flap} \end{array} . \tag{18}$$

In this work δ will be the variable that represents deflection angles for control surfaces. At the time of this work, not all of the independent control surfaces were available for control input authority. For this reason the total number of independent control surfaces available was constrained to 8. The independent control deflection vector used in this study is defined as,

$$\mathbf{U} \triangleq \begin{bmatrix} \delta_{a_L} \\ \delta_{a_R} \\ \delta_{sp_L} \\ \delta_{sp_R} \\ \delta_e \\ \delta_r \\ \delta_{st} \\ \delta_F \end{bmatrix} \begin{array}{l} \text{: L aileron} \\ \text{: R aileron} \\ \text{: L spoiler} \\ \text{: R spoiler} \\ \text{: elevator} \\ \text{: rudder} \\ \text{: stabilizer} \\ \text{: flaps} \end{array} . \tag{19}$$

From a pilot standpoint, however, the control inputs are lumped together. Pulling back on the control stick will simultaneously initiate the deflection of all 4 elevators. So the pilot input vector will be much different. The pilot stick will control the aileron elevator and rudder, while auxiliary logic for different flight scenarios will engage the stabilizers, flaps and spoilers. Stemming from

this ideology the pilot reference inputs could simply be defined as,

$$\mathbf{R}_{\text{pilot}} \triangleq \begin{bmatrix} \delta_{a,p} \\ \delta_{sp,p} \\ \delta_{e,p} \\ \delta_{r,p} \\ \delta_{st,p} \\ \delta_{F,p} \end{bmatrix} \begin{array}{l} \text{: aileron} \\ \text{: spoiler} \\ \text{: elevator} \\ \text{: rudder} \\ \text{: stabilizer} \\ \text{: flaps} \end{array} . \tag{20}$$

Specific details surrounding the engagement of the control surfaces and the interaction between the actual control surface position and the pilot's commanded position will be covered in great detail in the following sections. In general, the pilot stick is not always directly linked to a specific control surface. Different axes on the pilot stick might command control surfaces to move through the generation of state trajectory reference commands, thus indirectly affecting the control surfaces through error states and feedback control.

2.3 Actuator and Filter Dynamics

The control inputs from the pilot as well as the control inputs from the control logic will pass through the actuators before the deflection of the control surfaces occur. For this reason, actuator dynamics are included into the SIMULINK model. The actuator dynamics are modeled by: band limiting the control input, instituting maximum and minimum deflection angles for the control surfaces and rate limiting the movement of the control surfaces. A screen capture of the servo model in the SIMULINK environment is shown in Figrue 4 with the various control limits for the six different types of control surfaces detailed in Table 2.

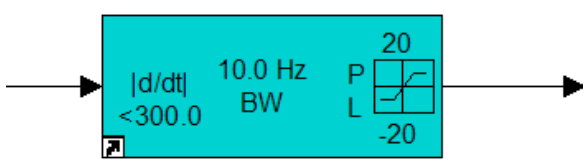

Figure 4. Visual representation of servo dynamics

In order to further increase the accuracy of the SIMULINK model, low pass filters are applied to the states for α, p, q and r. In the real 9 foot scaled remote control GTM there will be sensors in place to determine the above listed states. A visual representation of the low pass filters is shown in Figure 5. While the low pass filters have been added in order to increase the accuracy of the model, washout filters are added for a completely different reason. Washout filters are high pass filters and do not pass steady state values through. Washout filters have been added to the SIMULINK model for control

Table 1. Actuator dynamics and servo limits.

Surface	BW [Hz]	Rate Limit [deg/sec]	Max [deg]	Min [deg]
δ_a	10.0	300	20	-20
δ_{sp}	10.0	300	45	0
δ_e	10.0	300	20	-30
δ_r	8.4	300	30	-30
δ_{st}	10.0	300	20	-20
δ_F	10.0	300	20	-20

design purposes. Feedback on washed out states is preferable for maneuvers where a sustained pitch yaw or roll rates is maintained. More detail will be given in the control design as to why this is desirable.

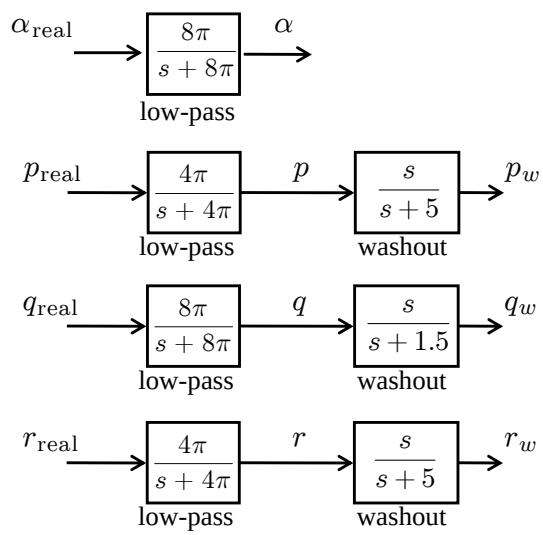

Figure 5. Filters

At the time of this work the specific types of sensors used to measure the states of the GTM were not given. However it is known that digital to analog conversion and can introduce time delays into a system. Therefore, all of the states of the GTM are delayed by τ_{filter}. A visual representation of the filter time delays is shown in Figure 6. Two other time delay variables are also present in the SIMULINK model τ_{up} and τ_{down}. These time delays values represent the time for wireless radio communication between the GTM aircraft and the ground pilot.

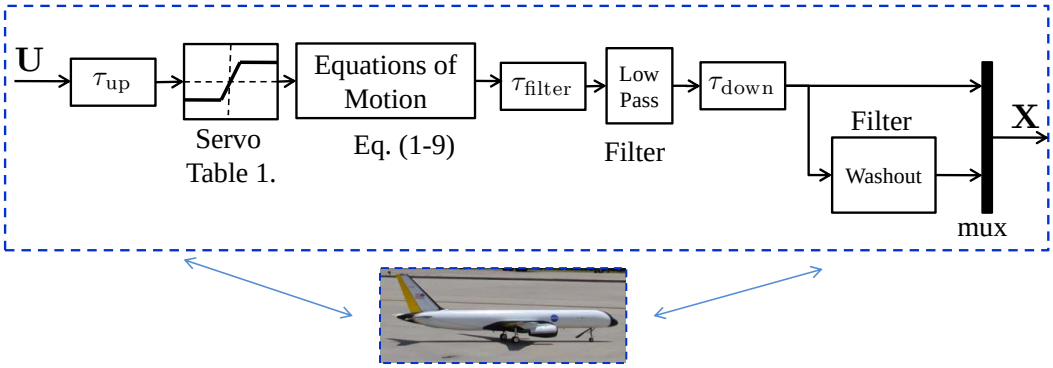

Figure 6. Entire GTM plant.

Table 2. Actuator dynamics and servo limits.

Delay Variable	Time [ms]
τ_{filter}	40
τ_{up}	10
τ_{down}	10

.

3 Baseline Control Design

The control design is comprised of both a baseline controller and an adaptive algorithm. The baseline controller is comprised of three sequential loop closing designs. A Control Augmented System (CAS) is designed for the elevator–pitch loop, and Stability Augmented Systems (SAS) are designed for the aileron–roll and rudder–yaw loops. The CAS incorporates an integral error augmentation for angle of attack command following. In order to increase the robustness of the CAS system, an integral anti–windup saturation logic is introduced. The adaptive control system is based on Model Reference Adaptive Control and explicitly accounts for actuator saturation and the time varying saturation limit imposed by the CAS system. The anti–windup logic is expressed in state space form as best as possible so as to ensure analytically tractable representations necessary for stability proofs. Before continuing, a detailed discussion on linearization must be performed.

3.1 Linearized Dynamics

The plant to be controlled involves the aircraft dynamics as outlined in Equations (1)-(9), along with the filter dynamics as illustrated in Figure 5. The control inputs to the plant were first defined in Equation (19) (but are repeated here in as well),

$$\mathbf{U} \triangleq [\delta_{a_L}\ \delta_{a_R}\ \delta_{sp_L}\ \delta_{sp_R}\ \delta_e\ \delta_r\ \delta_{st}\ \delta_F]^T,$$

and the states of the plant are defined as,

$$\mathbf{X} = [\alpha\ \beta\ \phi\ \theta\ \psi\ p\ q\ r\ p_w\ q_w\ r_w]^T.$$

Notice that $V_A \notin \mathbf{X}$. Under normal conditions the velocity would most certainly be a state variable. During this study however, the throttle inputs were not given as control inputs. It was therefore assumed impossible to simultaneously control aircraft attitude and velocity without throttle control. Noting the above input and state variable definitions along with the assumed dynamic model described above, the nonlinear system has the following form,

$$\dot{\mathbf{X}} = \boldsymbol{f}(\mathbf{X}, \mathbf{U}). \tag{21}$$

So that linearization can be performed on Equation (21), and to ensure that the system remains full states accessible, the servo and low–pass filter dynamics are ignored, along with all of the modeled time delays.[1] Figure 7 illustrates the above discussion explicitly showing the components of the aircraft that are neglected during linearization.

[1] The internal servo dynamics can not be measured, so in light of remaining a full states accessible approach there internal dynamics are ignored; and since the input to the gyros, for example, are not measurable the same ideology is extended to the low–pass filters as well.

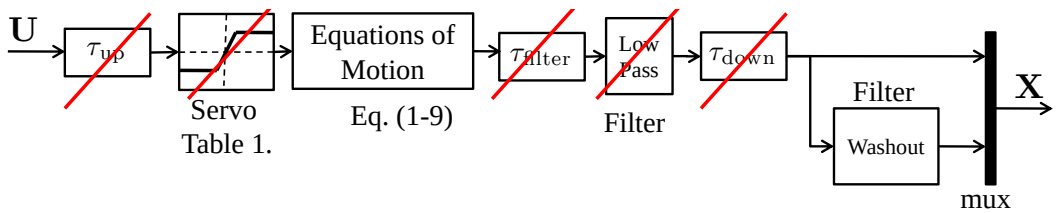

Figure 7. Visual representation of the dynamics included in linearization

The system will be linearized at the trim state $\mathbf{X}_0$ and trim input $\mathbf{U}_0$ satisfying

$$\dot{\mathbf{X}} = \boldsymbol{f}(\mathbf{X}_0, \mathbf{U}_0) = 0. \tag{22}$$

The resulting linear system is then:

$$\dot{\mathbf{x}}_\mathbf{p} = \mathbf{A}_\mathbf{p}\mathbf{x}_\mathbf{p} + \mathbf{B}_\mathbf{p}\mathbf{u} + \boldsymbol{\varepsilon}(t) \tag{23}$$

where,

$$\mathbf{A}_\mathbf{p} = \left.\frac{\partial \boldsymbol{f}(\mathbf{X}, \mathbf{U})}{\partial \mathbf{X}}\right|_{\substack{\mathbf{X}=\mathbf{X}_0 \\ \mathbf{U}=\mathbf{U}_0}} \tag{24}$$

$$\mathbf{B}_\mathbf{p} = \left.\frac{\partial \boldsymbol{f}(\mathbf{X}, \mathbf{U})}{\partial \mathbf{U}}\right|_{\substack{\mathbf{X}=\mathbf{X}_0 \\ \mathbf{U}=\mathbf{U}_0}} \tag{25}$$

$$\mathbf{x}_\mathbf{p} = \mathbf{X} - \mathbf{X}_0 \tag{26}$$

$$\mathbf{u} = \mathbf{U} - \mathbf{U}_0 \tag{27}$$

$$\boldsymbol{\varepsilon} : \text{linearization error} \tag{28}$$

It is assumed that $\boldsymbol{\varepsilon}$ is small. $\mathbf{x}_\mathbf{p}$ is the perturbation plant state vector and $\mathbf{u}$ is the perturbation input vector. When selecting scalar components of the aforementioned vectors the following $\Delta(\cdot)$ notation will be used,

$$\mathbf{u} = [\Delta\delta_{a_L}\ \Delta\delta_{a_R}\ \Delta\delta_{sp_L}\ \Delta\delta_{sp_R}\ \Delta\delta_e\ \Delta\delta_r\ \Delta\delta_{st}\ \Delta\delta_F]^T,$$

$$\mathbf{x}_\mathbf{p} = [\Delta\alpha\ \Delta\beta\ \Delta\phi\ \Delta\theta\ \Delta\psi\ \Delta p\ \Delta q\ \Delta r\ \Delta p_w\ \Delta q_w\ \Delta r_w]^T.$$

In the nominal control design $\Delta\alpha$ will be commanded to follow a reference input $\Delta\alpha_{\text{cmd}}$. Therefore, a selection vector $\mathbf{H}$ is introduced so that,

$$\Delta\alpha = \mathbf{H}\mathbf{x}_\mathbf{p}, \tag{29}$$

and then an alpha error signal e_α is introduced as,

$$e_\alpha = \int_0^t \Delta\alpha - \Delta\alpha_{\text{cmd}}\ \mathrm{d}t \tag{30}$$

Augmenting the error dynamics of Equation (30) to the plant dynamics from Equation (23) the open loop linear system then becomes,

$$\underbrace{\begin{bmatrix}\dot{\mathbf{x}}_\mathbf{p}\\ \dot{e}_\alpha\end{bmatrix}}_{\dot{\mathbf{x}}} = \underbrace{\begin{bmatrix}\mathbf{A_p} & 0\\ \mathbf{H} & 0\end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix}\mathbf{x_p}\\ e_\alpha\end{bmatrix}}_{\mathbf{x}} + \underbrace{\begin{bmatrix}\mathbf{B_p}\\ 0\end{bmatrix}}_{\mathbf{B}_1} \mathbf{u} + \underbrace{\begin{bmatrix}0\\ -1\end{bmatrix}}_{\mathbf{B}_2} \Delta\alpha_{\text{cmd}} \tag{31}$$

or more compactly

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}_1\mathbf{u} + \mathbf{B}_2\Delta\alpha_{\text{cmd}}. \tag{32}$$

In the subsequent sections the logic behind the control perturbation vector $\mathbf{u}$ will be introduced. The control will be broken into nominal feedback, adaptive feedback and pilot direct surface commands.

$$\mathbf{u} = \underbrace{\mathbf{u_n}}_{\text{nominal}} + \underbrace{\mathbf{u_a}}_{\text{adaptive}} + \underbrace{\mathbf{S}_2\boldsymbol{\delta}_\mathbf{p}}_{\text{direct surface, } \mathbf{u_s}} \tag{33}$$

where,

$$\begin{aligned}
\mathbf{u_n} &\triangleq \begin{bmatrix}\Delta\delta_{a_L,n} & \Delta\delta_{a_R,n} & 0 & 0 & \Delta\delta_{e,n} & \Delta\delta_{r,n} & 0 & 0\end{bmatrix}^T\\
\mathbf{u_a} &\triangleq \begin{bmatrix}\Delta\delta_{a_L,a} & \Delta\delta_{a_R,a} & \Delta\delta_{sp_L,a} & \Delta\delta_{sp_R,a} & \Delta\delta_{e,a} & \Delta\delta_{r,a} & \Delta\delta_{st,a} & \Delta\delta_{F,a}\end{bmatrix}^T\\
\boldsymbol{\delta}_\mathbf{p} &\triangleq \begin{bmatrix}\delta_{a,p} & \delta_{e,p} & \delta_{r,p}\end{bmatrix}^T \text{ and,}
\end{aligned}$$

$$\mathbf{S}_2 = \begin{bmatrix}-20 & 0 & 0\\ 20 & 0 & 0\\ 0 & 0 & 0\\ 0 & 0 & 0\\ 0 & 0 & 0\\ 0 & 0 & 30\\ 0 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}. \tag{34}$$

The first thing to notice is that the nominal perturbation input, $\mathbf{u_n}$, only has feedback for the aileron, elevator, and rudder; where as the the adaptive perturbation input command, $\mathbf{u_a}$, has feedback to as many independent control surfaces as possible. The pilot perturbation command is similar to the nominal control structure in that it only affects the aileron, elevator and rudder. Note that if a single input to the aileron is given, then it is implied that $\Delta\delta_{a_L,n} = -1 \cdot \Delta\delta_{a,n}$ and $\Delta\delta_{a_R,n} = +1 \cdot \Delta\delta_{a,n}$. The same is not true for the adaptive system where asymmetric commands can be give to the ailerons. The purpose of the selection matrix $\mathbf{S}_2$ is to distinguish between left and right aileron input commands, to select the components of the pilots input that directly affect control surfaces and scale the pilot stick inputs in order to map them to deflection angles on the actual control surfaces. Later, $\mathbf{S}_1$ will be

introduced as the matrix that selects the elevator command and directs it to the CAS system. The subscript $(\cdot)_p$ denotes normalized pilot inputs, bounded between -1 and 1, and the subscript $(\cdot)_s$ denotes the actual surface deflection that is being commanded with feed–forward input. To review, the rudder and ailerons are directly affected by the pilots command stick, in contrast to the elevator pilot input, which passes through auxiliary logic.

As an exercise in understanding, consider the total input to the GTM:

$$\begin{aligned} \mathbf{U} &= \mathbf{U}_0 + \mathbf{u} \\ &= \mathbf{U}_0 + \underbrace{\mathbf{u_n}}_{\text{nominal}} + \underbrace{\mathbf{u_a}}_{\text{adaptive}} + \underbrace{\mathbf{u_s}}_{\text{direct surface}} \end{aligned} \tag{35}$$

so that the total input for the rudder for instance would consist of:

$$\delta_r = \delta_{r,\text{trim}} + \Delta\delta_{r,n} + \Delta\delta_{r,a} + 30\delta_{r,p}$$

where the subscript notation $(\cdot)_{\text{trim}}$ denotes a component of the trim vector $\mathbf{U}_0$.

3.2 Sequential Loop Closing Controller

In the baseline control design, three control loops are used for command following and aircraft stabilization. The Control Augment System (CAS) elevator to pitch loop is shown in Figure 8.

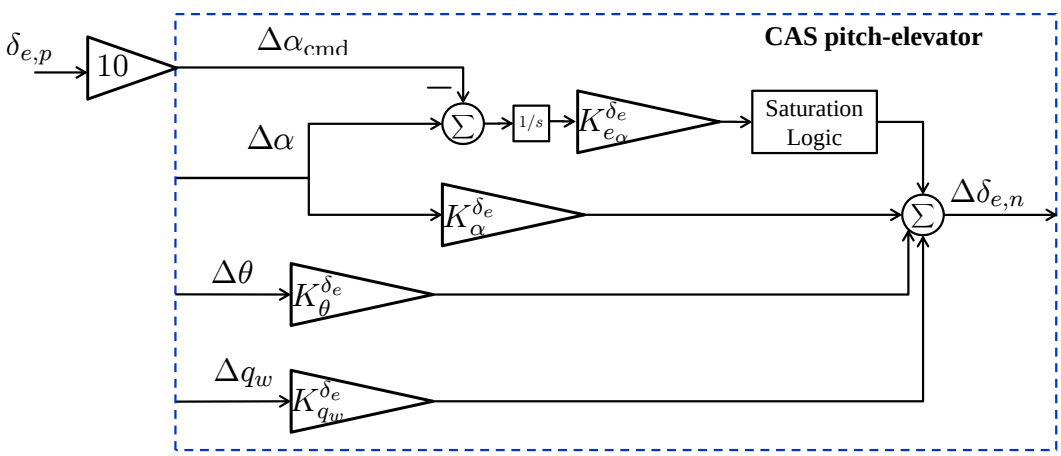

Figure 8. CAS for pitch–elevator loop.

In the CAS design the pilot stick controlling the elevator deflection is denoted as δ_{cmd}, and is subsequently scaled by a factor of 10 in order to obtain the angle of attack command signal

$$\Delta\alpha_{cmd} = 10\delta_{e,p}. \tag{36}$$

The elevator control stick indirectly affects the elevator through the generation of the alpha command and the subsequent generation of e_α as shown in Equation (30). The four states e_α $\Delta\alpha$, $\Delta\theta$ and Δq_w are then multiplied by the scalar components of the nominal state feedback gain $\mathbf{K}$. The sum of the four control signals generates the nominal perturbation input for the elevator, $\Delta\delta_{e,\mathrm{n}}$,

$$\Delta\delta_{e,n} = \begin{bmatrix} K_{e_\alpha}^{\delta_e} & K_\alpha^{\delta_e} & K_\theta^{\delta_e} & K_{q_w}^{\delta_e} \end{bmatrix} \begin{bmatrix} e_\alpha \\ \Delta\alpha \\ \Delta\theta \\ \Delta q_w \end{bmatrix}. \tag{37}$$

The subscript $(\cdot)_n$ will be used in this work to denote nominal. Note that there is a saturation logic block shown in Figure 8. Equation (38) is not entirely correct once saturation in the elevator occurs. A more accurate representation of the nominal control law is as follows:

$$\Delta\delta_{e,n} = \begin{bmatrix} K_{e_\alpha}^{\delta_e} & K_\alpha^{\delta_e} & K_\theta^{\delta_e} & K_{q_w}^{\delta_e} \end{bmatrix} \begin{bmatrix} R_e(e_\alpha, \delta_e(t)) \\ \Delta\alpha \\ \Delta\theta \\ \Delta q_w \end{bmatrix}. \tag{38}$$

In the revamped nominal control law a saturation function R_e is introduced. Before going into the structure of the integral anti–windup saturation function another form of saturation must be reviewed.

In addition to the integral saturation and anti–windup logic, each of the control surfaces on the aircraft has a maximum angle of deflection. Let the scalar saturation function, R_s, representing this effect be defined as:

$$R_s(u_i) = \begin{cases} u_i & \text{if } \|u_i\| \leq u_{i,\max} \\ u_{i,\max}\ \mathrm{sign}(u_i) & \text{if } \|u_i\| > u_{i,\max} \end{cases} \tag{39}$$

where the subscript i runs through the length of the control input vector, $n_u = 8$.[2] It is important to make a distinction between control inputs that are present for stabilization and those necessary for command following. Consider the following parsing of the baseline control input for the elevator.[3,4]

$$\begin{aligned} \delta_e(t) &= \underbrace{\delta_{e,\mathrm{trim}} + K_{q_w}^{\delta_e}\Delta q_w + K_\theta^{\delta_e}\Delta\theta + K_\alpha^{\delta_e}\Delta\alpha}_{\text{stabilization}} + \underbrace{K_{e_\alpha}^{\delta_e} R_s(e\alpha, \delta_e(t))}_{\text{command following}} \\ &= \delta_{e,\mathrm{stab}} + \underbrace{K_{e_\alpha}^{\delta_e} R_s(e_\alpha, \delta_e(t))}_{\text{command following}} \end{aligned} \tag{40}$$

[2]This saturation function assumes that the saturation limit is symmetric about zero. This need not be the case, but does increase the analytical tractability of the formulation.

[3]When the adaptive algorithm is introduced there will be extra terms present in Equation (40)

[4]The adaptive components of the stabilization and command following must also be taken into account, but have been left out for increased intuition when dealing with the baseline control alone.

In the CAS system priority is given to maintaining stability. Then, the alpha command following ensues, however, the integration error is monitored so as to not instigate integration windup.

When the elevator begins to saturate e_α is adjusted. Consider the following saturation function and state resetting definition for $e_\alpha \geq 0$,

$$R_e(e_\alpha) = \begin{cases} e_\alpha & \text{if } e_\alpha \leq e_{\text{available}} \\ e_{\text{available}} & \text{if } e_\alpha > e_{\text{available}} \end{cases} \tag{41}$$

$$e_{\text{available}} = \begin{cases} \max\left\{0, \left(R_s(\delta_e) - \delta_{e,\text{stab}}\right)/K_{e_\alpha}^{\delta_e}\right\} & \text{if } \delta_e \geq 0 \\ \min\left\{0, \left(R_s(\delta_e) - \delta_{e,\text{stab}}\right)/K_{e_\alpha}^{\delta_e}\right\} & \text{if } \delta_e < 0 \end{cases} \tag{42}$$

$$e_\alpha = \begin{cases} e_\alpha & \text{if } \dot{e}_\alpha \geq 0 \text{ or } e_\alpha \leq e_{\text{available}} \\ e_{\text{available}} & \text{if } \dot{e}_\alpha < 0 \text{ and } e_\alpha > e_{\text{available}} \end{cases}. \tag{43}$$

Equation (41) is a simple rectangular saturation function, which is bounded from above by the available integration error, $e_{\text{available}}$. The amount of available error is then simply the maximum signal value that when multiplied by the integration error gain will not induce saturation of the elevator. If for instance, the stability component of the elevator input is demanding the maximum deflection angle of the elevator, then $e_{\text{available}} = 0$. Equation (43) represents the anti–windup logic. When the integration error is positive, and begins to head back into the direction of the maximum available error the error signal is reset to the available limit. Thus short circuiting the unwinding process. A visual representation of the anti–windup logic is shown in Figure 9.

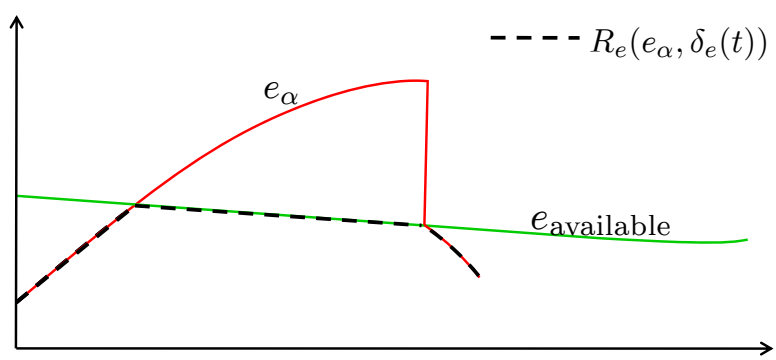

Figure 9. Integral error saturation and anti–windup

The above integration saturation and anti–windup logic were for positive integration errors. For completeness, the replacements for Equations (41) and

(43) when $e_\alpha < 0$ are listed as well,

$$R_e(e_\alpha) = \begin{cases} e_\alpha & \text{if } e_\alpha \geq e_{\text{available}} \\ e_{\text{available}} & \text{if } e_\alpha < e_{\text{available}} \end{cases} \tag{44}$$

$$e_\alpha = \begin{cases} e_\alpha & \text{if } \dot{e}_\alpha \leq 0 \text{ or } e_\alpha \geq e_{\text{available}} \\ e_{\text{available}} & \text{if } \dot{e}_\alpha > 0 \text{ and } e_\alpha < e_{\text{available}} \end{cases} . \tag{45}$$

This completes the CAS baseline control design. Later when the adaptive system is introduced some of the above definitions are redefined to include the adaptive effects. It is redundant to redefine equations, however, with the adaptive components included some of the equations become less intuitive at first glance.

The Stability Augmented Systems (SAS) for yaw and roll are much simpler than the CAS system. The two SAS control systems are shown in Figures 10(a) and 10(b).

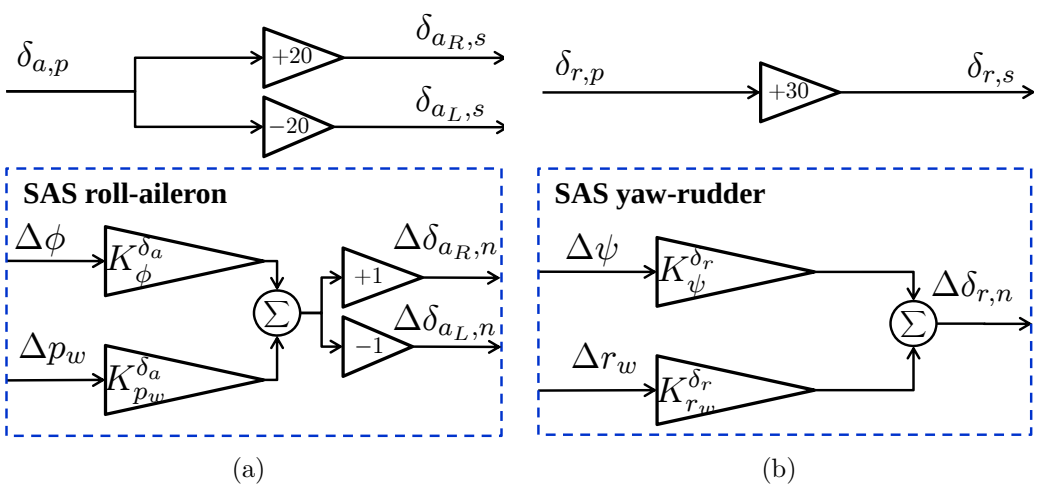

Figure 10. (a) Baseline aileron control loop, (b) Baseline rudder control loop.

The nominal perturbation inputs for the SAS systems are listed below:

$$\Delta\delta_{a,n} = K_{p_w}^{\Delta_a}\Delta p_w + K_\phi^{\delta_a}\Delta\phi \tag{46}$$

$$\Delta\delta_{r,n} = K_{r_w}^{\delta_r}\Delta r_w + K_\psi^{\delta_r}\Delta\psi \tag{47}$$

The SAS systems are essentially yaw and roll dampers. Combining the three sequential loop closing systems together leads to an overall baseline control design as shown in Figure 11 with a further simplified representation shown in Figure 14.

The next step in the control design is to generate a closed loop state space model of the GTM. This is not a straightforward task. Given the nonlinearities in the control design we arrive to the conclusion that $\mathbf{u_n} \neq \mathbf{Kx}$.

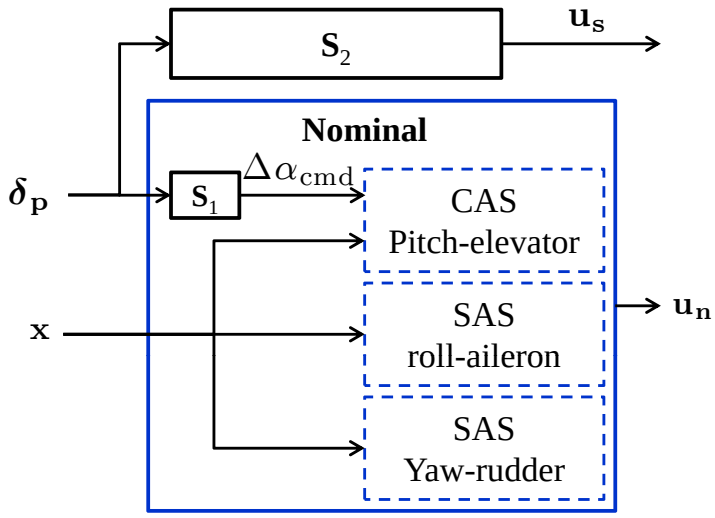

Figure 11. SAS and CAS control systems.

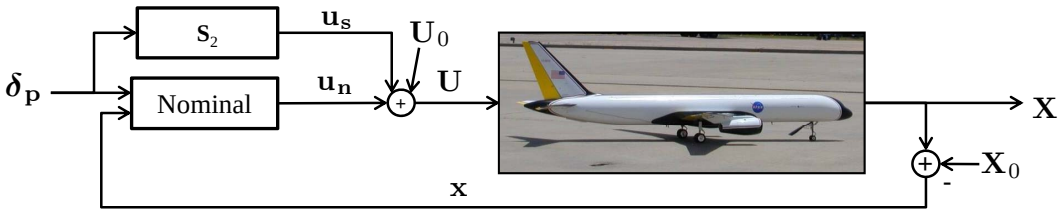

Figure 12. Baseline control structure.

3.3 Representing Nonlinearities in a Linear Context

Even though the nominal controller does not have a simple linear relation with the feedback gain $\mathbf{K}$ a pseudo linear representation can be realized. The only nonlinearities in the baseline controller are the control surface saturation function,

$$\mathbf{R_s}(\mathbf{u}) \triangleq \begin{bmatrix} R_s(u_1) & R_s(u_2) & \cdots & R_s(u_8) \end{bmatrix}^T \tag{48}$$

and the integration saturation function R_s.

Using the linear dynamics as shown in Equation (31) with the baseline controller defined as $\mathbf{u} = \mathbf{u_n} + \mathbf{u_s}$ the following closed loop dynamics can be

generated,[5]

$$\underbrace{\begin{bmatrix}\dot{\mathbf{x}}_\mathbf{p}\\ \dot{e}_\alpha\end{bmatrix}}_{\dot{\mathbf{x}}} = \underbrace{\begin{bmatrix}\mathbf{A_p}+\mathbf{B_p}\mathbf{K}_1 & \mathbf{B_p}\mathbf{K}_2\\ \mathbf{H} & 0\end{bmatrix}}_{\mathbf{A}+\mathbf{B}_1\mathbf{K}}\underbrace{\begin{bmatrix}\mathbf{x_p}\\ e_\alpha\end{bmatrix}}_{\mathbf{x}} + \underbrace{\begin{bmatrix}\mathbf{B_p}\\ 0\end{bmatrix}}_{\mathbf{B}_1}\mathbf{S}_1\boldsymbol{\delta}_\mathbf{p} + \underbrace{\begin{bmatrix}0\\ -1\end{bmatrix}}_{\mathbf{B}_2}\mathbf{S}_2\boldsymbol{\delta}_\mathbf{p} - \underbrace{\begin{bmatrix}\mathbf{B_p}\\ 0\end{bmatrix}}_{\mathbf{B}_1}\mathbf{u}_\Delta - \underbrace{\begin{bmatrix}\mathbf{B_p}\\ 0\end{bmatrix}}_{\mathbf{B}_1}\mathbf{K}_2 e_{\alpha,\Delta} \tag{49}$$

where the saturation defect signals are denoted as $(\cdot)_\Delta$, and defined as

$$e_{\alpha,\Delta} = e_\alpha - R_e(e_\alpha, \delta_e) \tag{50}$$

$$\mathbf{u}_\Delta = \mathbf{u} - \mathbf{R_s}(\mathbf{u}). \tag{51}$$

The saturation defect signals are introduced in order to have an analytically tractable closed loop dynamic model of the GTM. Equation (49) is deemed pseudo–linear because the feedback matrix $\mathbf{K}$ still portrays a linear relationship with $\mathbf{x}$, and the nonlinearities are captured entirely by the defect signals, which can be thought of as exogenous inputs. Note that this work is ongoing and was a first attempt at characterizing the nonlinearities in the anti–windup scheme in a tractable fashion.

[5]Notice that the following subscript notation was used on $\mathbf{K}$ as explained in the Nomenclature section of this report, $\mathbf{K} = \begin{bmatrix}\mathbf{K}_1 & \mathbf{K}_2\end{bmatrix}$

4 Uncertainty

Two forms of uncertainty are discussed in this work. Unknown time delay and uncertain center of gravity location. The uncertain time delay will appear in the system as shown in Figure 4. The uncertain time delay will be another time delay in addition to the three known time delays.

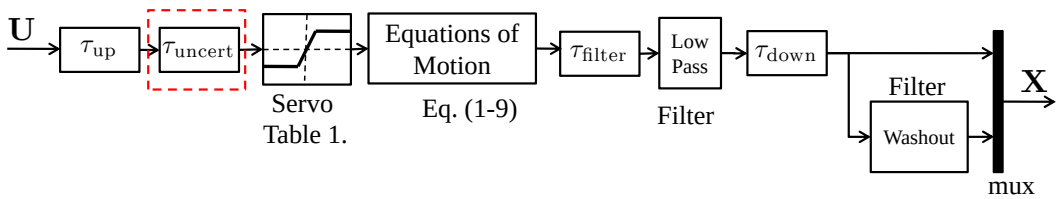

Figure 13. Time delay uncertainty

Uncertain center of gravity affects are also studied within this work. During simulations the center of gravity is simply injected into the aircraft model as a design variable. The effects that center of gravity shifts have on the equations of motion of an aircraft were studied in great detail in [2]. The center of gravity will change the state jacobian matrix A_p as first introduced in Equation (23). Let the uncertain center of gravity be characterized by the variable λ, where

$$\lambda = \frac{\Delta l}{\text{MAC}} \tag{52}$$

Δl is the perturbation from the assumed position of the center of gravity in ft, and MAC is the mean aerodynamics chord. Therefore, λ is essentially the percent change in the center of gravity location in relation to the length of the cross section of the wing at the fuselage. Positive values of λ assume that the center of gravity was moved toward the nose of the aircraft and negative values are towards the tail of the aircraft. The uncertain state jacobian matrix is defined as

$$A_{p,\ \text{uncertain}} = A_p(\lambda). \tag{53}$$

Combining the closed loop system dynamics with saturation in (49) with the uncertain jacobian matrix (53), results in

$$\begin{aligned}\underbrace{\begin{bmatrix}\dot{\mathbf{x}}_\mathbf{p}\\ \dot{e}_\alpha\end{bmatrix}}_{\dot{\mathbf{x}}} = \underbrace{\begin{bmatrix}\mathbf{A_p}(\lambda)+\mathbf{B_p}\mathbf{K}_1 & \mathbf{B_p}\mathbf{K}_2\\ \mathbf{H} & 0\end{bmatrix}}_{\mathbf{A}(\lambda)+\mathbf{B}_1\mathbf{K}}\underbrace{\begin{bmatrix}\mathbf{x_p}\\ e_\alpha\end{bmatrix}}_{\mathbf{x}} + \underbrace{\begin{bmatrix}\mathbf{B_p}\\ 0\end{bmatrix}}_{\mathbf{B}_1}\mathbf{S}_1\boldsymbol{\delta}_\mathbf{p}\\ + \underbrace{\begin{bmatrix}0\\ -1\end{bmatrix}}_{\mathbf{B}_2}\mathbf{S}_2\boldsymbol{\delta}_\mathbf{p} - \underbrace{\begin{bmatrix}\mathbf{B_p}\\ 0\end{bmatrix}}_{\mathbf{B}_1}\mathbf{u}_\Delta - \underbrace{\begin{bmatrix}\mathbf{B_p}\\ 0\end{bmatrix}}_{\mathbf{B}_1}\mathbf{K}_2 e_{\alpha,\Delta}\end{aligned} \tag{54}$$

or in compact form,

$$\dot{\mathbf{x}} = (\mathbf{A}(\lambda) + \mathbf{B}_1\mathbf{K})\mathbf{x} + \mathbf{B}_1(\mathbf{S}_1\boldsymbol{\delta}_\mathbf{p} - \mathbf{u}_\Delta - \mathbf{K}_2 e_{\alpha,\Delta}) + \mathbf{B}_2\mathbf{S}_2\boldsymbol{\delta}_\mathbf{p} \tag{55}$$

Given the uncertainty above, an adaptive controller is introduced in order to improve the performance of the baseline controller.

5 Adaptive Control Design

Recall the total input to the plant as first shown in (35), and repeated here in,

$$\mathbf{U} = \mathbf{U}_0 + \underbrace{\mathbf{u_n}}_{\text{nominal}} + \underbrace{\mathbf{u_a}}_{\text{adaptive}} + \underbrace{\mathbf{u_s}}_{\text{direct surface}}$$

The adaptive input $\mathbf{u_a}$ augments naturally to the nominal control structure. A visual representation of the complete control structure is shown in Figure 5 Consider the uncertain closed loop dynamics with the baseline control input

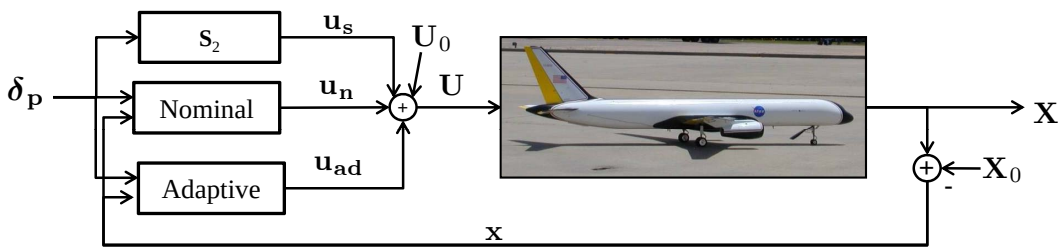

Figure 14. Full control design.

as first shown in (55), and repeated here,

$$\dot{\mathbf{x}} = (\mathbf{A}(\lambda) + \mathbf{B}_1\mathbf{K})\mathbf{x} + \mathbf{B}_1(\mathbf{S}_1\boldsymbol{\delta}_\mathbf{p} - \mathbf{u}_\Delta - \mathbf{K}_2 e_{\alpha,\Delta}) + \mathbf{B}_2\mathbf{S}_2\boldsymbol{\delta}_\mathbf{p}$$

Notice that with the uncertain parameter λ there is no guarantee that the eigen values of $\mathbf{A}(\lambda) + \mathbf{B}_1\mathbf{K}$ will remain in the left half plane. The motivation for the adaptive component continues as follows. Collect the bounded and unbounded terms in the above expression as:

$$\mathbf{u_b} = \mathbf{S}_1\boldsymbol{\delta}_\mathbf{p} \tag{56}$$

$$\mathbf{u_u} = \mathbf{u}_\Delta + \mathbf{K}_2 e_{\alpha,\Delta} \tag{57}$$

and recall that the direct surface commands were previously defined as $\mathbf{u_s} = \mathbf{S}_2\boldsymbol{\delta}_\mathbf{p}$. Substituting the above simplications into (55),

$$\dot{\mathbf{x}} = (\mathbf{A}(\lambda) + \mathbf{B}_1\mathbf{K})\mathbf{x} + \mathbf{B}_1(\mathbf{u_b} - \mathbf{u_u}) + \mathbf{B}_2\mathbf{u_s}. \tag{58}$$

The adaptive parameter is then introduced as a term that will premultiply $\mathbf{B}_1$, so that the closed loop dynamics are now of the form,

$$\dot{\mathbf{x}} = (\mathbf{A}(\lambda) + \mathbf{B}_1\mathbf{K})\mathbf{x} + \mathbf{B}_1(\mathbf{u_b} - \mathbf{u_u} + \mathbf{u_a}) + \mathbf{B}_2. \tag{59}$$

The adaptive component of the controller is a time varying state feedback gain,

$$\mathbf{u_a} \triangleq \boldsymbol{\theta}^T(t)\mathbf{x}. \tag{60}$$

Substituting the adaptive law from (60) into (59) yields the closed loop dynamical expression,

$$\dot{\mathbf{x}} = (\mathbf{A}(\lambda) + \mathbf{B}_1(\mathbf{K} + \boldsymbol{\theta}^T))\mathbf{x} + \mathbf{B}_1(\mathbf{u_b} - \mathbf{u_u}) + \mathbf{B}_2\mathbf{u_s}. \tag{61}$$

Notice that the eigen values of the expression $\mathbf{A}(\lambda)+\mathbf{B}_1(\mathbf{K}+\boldsymbol{\theta}^T)$ can be placed anywhere, given a controllable pair $(\mathbf{A}, \mathbf{B}_1)$. The adaptive parameter $\boldsymbol{\theta}$ will be adjusted by comparing the performance of the closed loop GTM with that of a reference model. We define the reference model as,

$$\dot{\mathbf{x}}_\mathbf{m} = \mathbf{A_m}\mathbf{x_m} + \mathbf{B}_1\mathbf{u_b} + \mathbf{B}_2\mathbf{u_s} \tag{62}$$

The reference model Jacobian $\mathbf{A_m}$ is chosen to be Hurwitz, so that given bounded inputs $\mathbf{u_b}$ and $\mathbf{u_s}$ the output $\mathbf{x_m}$ is globally bounded. The model following error

$$\mathbf{e} = \mathbf{x} - \mathbf{x_m} \tag{63}$$

is now introduced along with the following assumption. For all λ under consideration, there exist $\boldsymbol{\theta}^*$ such that, $\mathbf{A}(\lambda) + \mathbf{B}_1(\mathbf{K} + \boldsymbol{\theta}^{*T}) = \mathbf{A_m}$. In addition to the matching condition assumption, an adaptive error term is introduced,

$$\tilde{\boldsymbol{\theta}} = \boldsymbol{\theta} - \boldsymbol{\theta}^*. \tag{64}$$

If the adaptive term $\boldsymbol{\theta}$ approaches $\boldsymbol{\theta}^*$, then the adaptive error term $\tilde{\boldsymbol{\theta}}$ will converges to zero, and the dynamics of the closed loop GTM will match the dynamics of the reference model. The above scencario is not completely true however when saturation occurs. Consider the time derivative of the model followin error from (64),

$$\dot{\mathbf{e}} = \mathbf{A_m}\mathbf{e} - \mathbf{B}_1\mathbf{u_u} + \mathbf{B}_1\tilde{\boldsymbol{\theta}}^T\mathbf{x} \tag{65}$$

Notice that the unbounded term $\mathbf{u_u}$ still apears and specifically has a non zero value when saturation occurs. For this reason another error signal is generated. The error defect signal, $\mathbf{e}_\Delta$ has a time derivative as

$$\dot{\mathbf{e}}_\Delta = \mathbf{A_m}\mathbf{e}_\Delta - \mathbf{B}_1\mathbf{u_u}. \tag{66}$$

The error defect signal uses the unbounded input in order to estimate the amount of error in the reference model that occurs from saturation alone. From the reference model error and the defect error, the augmented error, $\mathbf{e_a} = \mathbf{e} - \mathbf{e}_\Delta$ now has a time derivative of the following form.

$$\dot{\mathbf{e}}_\mathbf{a} = \mathbf{A_m}\mathbf{e_a} + \mathbf{B}_1\tilde{\boldsymbol{\theta}}^T\mathbf{x} \tag{67}$$

The form of equation (67) has already been studied in great detail and previous work suggest that a stable tuning law for $\boldsymbol{\theta}$ exist,

$$\dot{\boldsymbol{\theta}} = -\boldsymbol{\Gamma}\mathbf{x}\mathbf{e_a}^T\mathbf{P}\mathbf{B}_1 - \boldsymbol{\sigma}\boldsymbol{\theta} \tag{68}$$

where $\mathbf{A_m}^T\mathbf{P} + \mathbf{P}\mathbf{A_m} = -\mathbf{Q}$ and $\mathbf{Q} = \mathbf{Q}^T > 0$, the parameter $\mathbf{\Gamma}$ is a positive diagonal matrix that controls the rate of adaptation and $\boldsymbol{\sigma}$ is a possitive diagonal matrix that adds robustness to the adaptive law.

There are three extensions however that are necessary for a complete stability proof of the adaptive law above. The integral saturation law introduces time varying saturation limits, which have not been studied before. Also, anti–windup logic has not been studied in great detail in the context of model reference adaptive control, and with a short literature search the authors of this work have found no stability proof for anti–windup logic involving simple state resetting. Another issue considering stability is how the authors of this work chose to deal with the large known time delays in the system. Historically and at present, time delay robustness has been of major concern for adaptive systems. Resent works such as, [22], have gone through pain staking analysis in order to determine analytical time delays margins for adaptive systems. In this work there are several large "known" time delays. The known time delays were added to the reference model as shown in Figure 15, and then the only time delay discrepancy between the GTM and the reference model is the "unknown" component of the time delay.

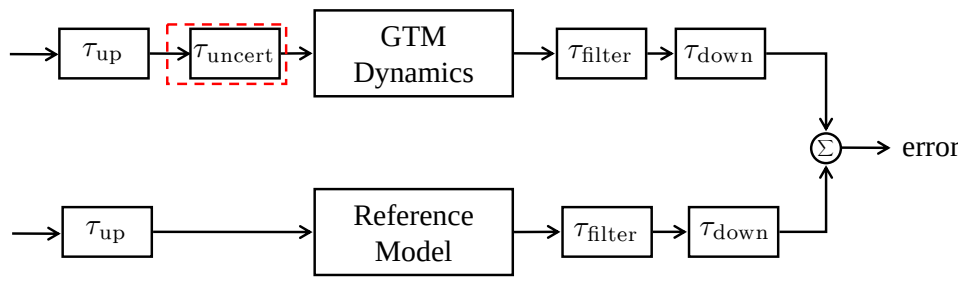

Figure 15. Time delay in reference model

With the time delays added to the reference model, the reference model becomes less stable, yet the task of model matching is simplified for the adaptive parameter. The burden of robustness with respect to time delay is shifted from the adaptive parameter space to the reference model. The implications of introducing time delays into the reference model have never been studied by any adaptive control groups.

6 High Fidelity Simulation Studies

Two different sets of simulation studies were performed in this section. The first set of simulations study the impact of uncertain CG locations and time delays on transient performance. The performance metric in the first study is the maximum acceleration of the fuselage. In the second study an angle of attack wave train is commanded and the time response of the nominal and adaptive controllers is compared over a 35 second time window. For all of the studies presented in this section the GTM was trimmed with an airspeed of 87 Knots at and angle of attack of 3 degrees. For the studies below the fixed step Euler 1 solver was used at 500 Hz.

6.1 Loading Factor Study

For the first study an angle of attack command was given as shown in Figure 16. At time zero the center of gravity is perturbed and the control input signal is delayed by the uncertain and known time delay amount. The z–direction loading factor (N_z=$|\ddot{z}/g|$) is constantly monitored in these studies and when the loading factor exceeds 5, then the simulations are stopped. Simulations were stopped at loading factors of 5 because at that value critical support structures within the wings of transport vehicles begin to plastically yield. The purpose of this study was to compare the transient response of the nominal and adaptive systems. Adaptive structures are notoriously aggressive, and as so, could possibly induce large loading factors. The largest spikes in the loading factor occur at two time instances in these studies. At time zero there is an immediate impulse response from the system. This is do to the fact that the aircraft begins each simulation at a non–trim condition. The second spike in the transients occurs at 2 seconds when the step command is given in the angle of attack.

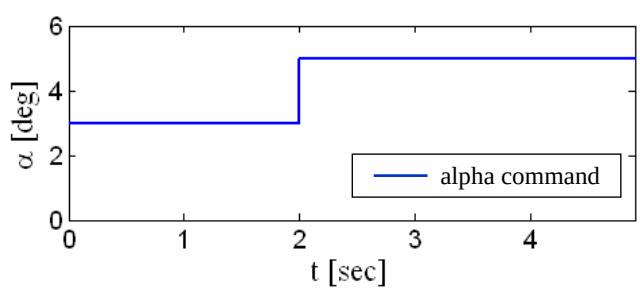

Figure 16. AOA path for study 1

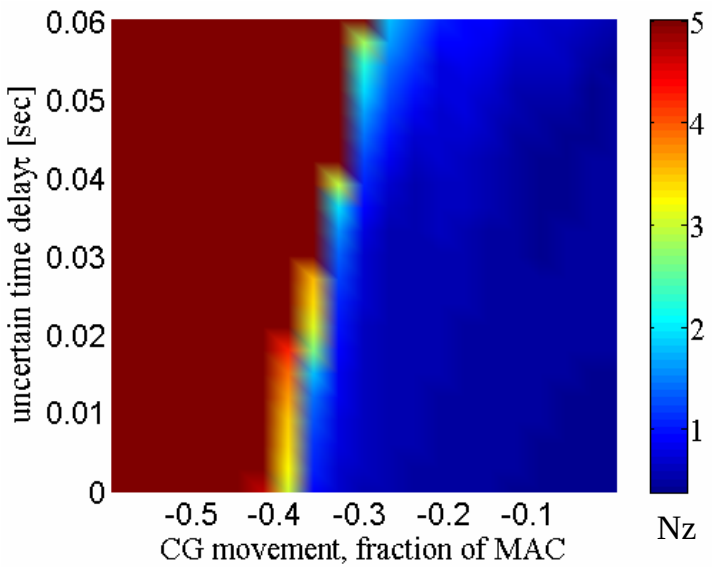

Figure 17. Loading factor results baseline controller

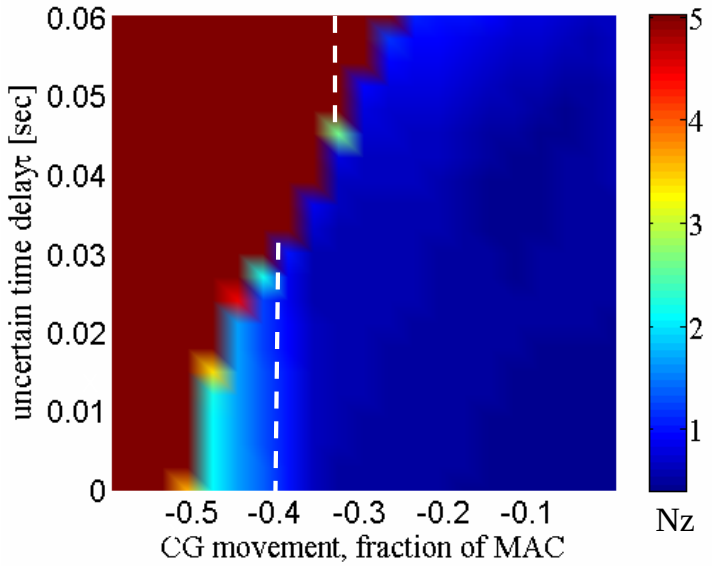

Figure 18. Loading factor study baseline + adaptive

For this study, uncertain time delays from 0 to 60 ms and uncertain CG locations of λ equal to $0 \sim -0.60$ were simulated.[6] The results for the nominal controller alone are shown in Figure 17, and the results for the nominal with adaptive controller are shown in Figure 18. It was seen that the adaptive algorithm does not generate excessively high loading factors during these

[6]Note that for each simulation there are allready 60 ms of known time delay incorporated. Therefore, a simulation with 60 ms of uncertain time delay has a total time delay of 120 ms.

transients, and in fact has lower loading factors for a larger uncertainty set than the nominal controller alone. In figure 18 the boundary for the nominal controller is shown as a white dashed line. From comparing the nominal and adaptive controller it is seen that that for smaller time delays the adaptive controller is more robust to uncertain CG location. For large time delays the adaptive controller is less robust than the nominal controller as shown by the dashed line at the top of Figure 18. However, at uncertain time delays of 60 ms the total time delay in the system is greater than 120 ms. Time delays of that magnitude encroach on data link drop out time scales and are considered to be well outside the range of normal conditions.

6.2 Doublet Command Following

Six different uncertain scenarios were simulated for the doublet command trajectory given in Figure 19. The six studies are shown in Table 3. The studies are broken into two groups; time delay studies (TD1–2–3) and center of gravity studies (CG1–2–3). The time delays studies have a fixed CG uncertainty of $\lambda = -0.30$, and the uncertain time delay is increased from 0 ms to 40 ms. In the center of gravity studies the uncertain time delay is held at 20 ms and the uncertain CG location is moved from $\lambda = -0.30 \sim -0.40$.

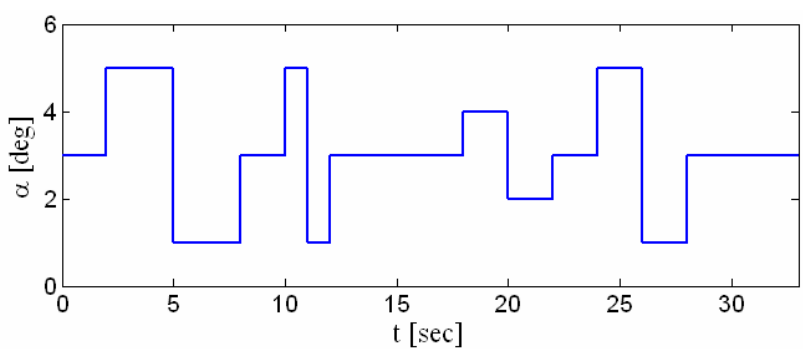

Figure 19. Time simulation study angle of attack doublets

Table 3. Time simulation study chart.

Test Name	Uncertain Delay [ms]	Total Delay [ms]	Uncertain CG (λ)
TD1	0	60	-0.30
TD2	20	80	-0.30
TD3	40	100	-0.30
CG1	20	80	-0.20
CG2	20	80	-0.35
CG3	20	80	-0.40

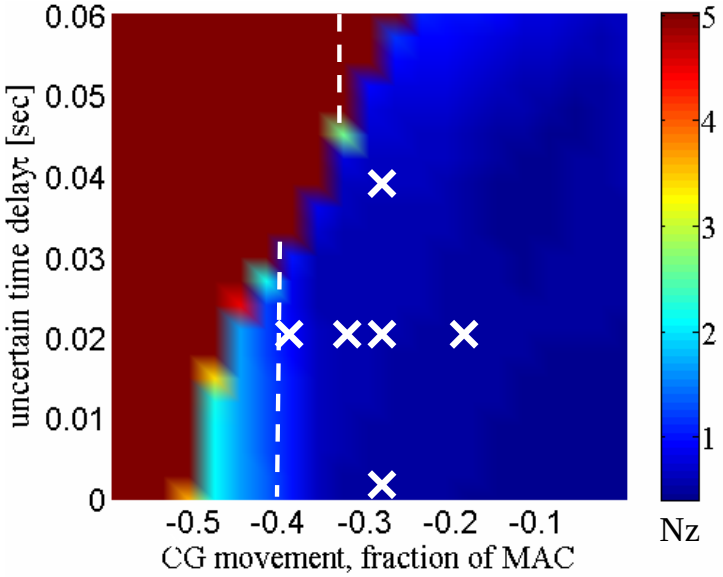

Figure 20. Time simulation study map

A visual map of the uncertainties is given in Figure 20, where the uncertain TD and CG studies are shown as superimposed white ×'s on the loading factor study for the adaptive controller.

The results for TD1 are shown in Figure 21. For this study the time delay was 0 ms and the uncertain CG location was not large enough to cause drastic difference in the dynamic response of the closed loop GTM when compared to the reference system. For this reason the adaptive and nominal responses are very similar. This is an ideal result. When there is little uncertainty in the system the adaptive system should remain relatively dormant. Notice that at $t = 0$ there are large transients in the angle of attack. This is do to the unstable initial condition instituted by the uncertain CG location. Thus an impulse response is obtained. This impulse response occurs in every scenario, and because the TD and CG uncertain scenarios are in the stable regions of the loading factor study, it is known apriori that the the loading factor obtained from these transients are below 5 for each of the six studies. The results for TD2 are shown in Figure22. The differences in the nominal and adaptive systems are becoming more clear as the overshoot is significantly smaller in the adaptive system. Study TD3 has similar characteristics to that of TD2 as shown in Figure 23. The adaptive controller once again has $\sim$ 50% reduction in overshoot. In general the time delay studies illustrate that for reasonable time delays 100 ms the adaptive controller and nominal controller have similar characteristics with the adaptive controller illustrating smaller overshoots, while the robustness with respect to time delay is not visibly depredated.

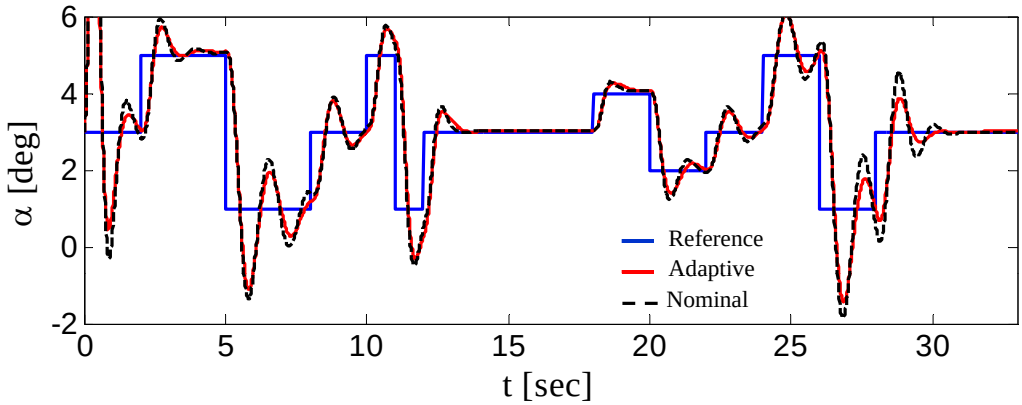

Figure 21. TD1 $\tau_{\text{uncert}} = 0.0ms$ $\lambda = -0.3$

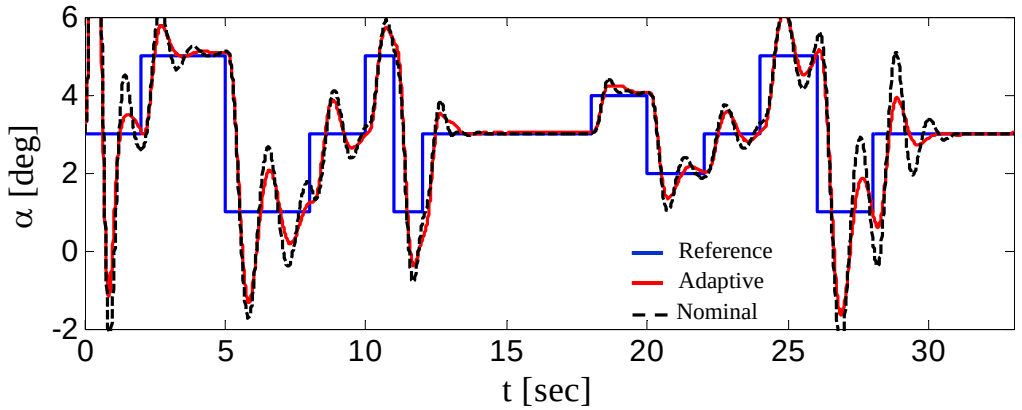

Figure 22. TD2 $\tau_{\text{uncert}} = 0.2ms$ $\lambda = -0.3$

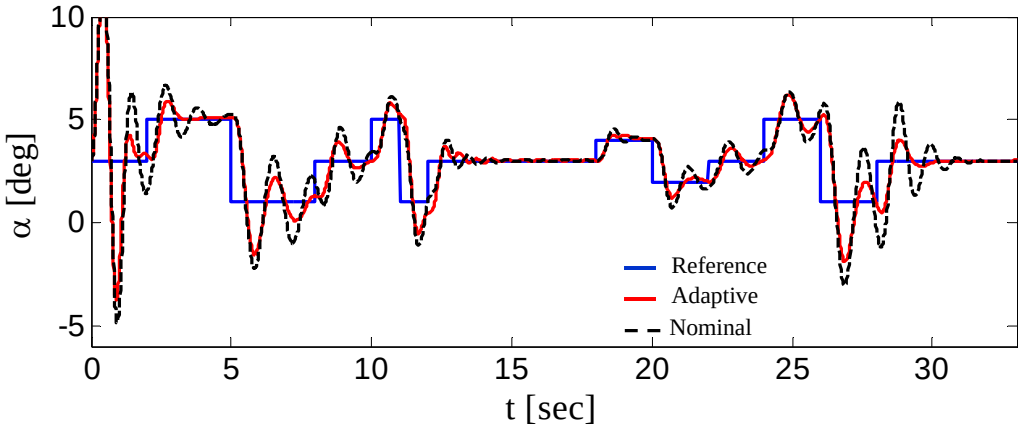

Figure 23. TD3 $\tau_{\text{uncert}} = 0.4ms$ $\lambda = -0.3$

The CG1–2–3 are shown in Figures 24, 25 and 26 respectively. The three studies have the same uncertain time delay of $\tau_{\text{uncert}} = 0.20ms$ and the center of gravity is moved toward the aft of the aircraft with larger and larger uncertainty in each simulation. In CG1, the uncertainties are similar to that of TD1 and there simply is not enough discrepancy between the closed loop dynamics of the GTM and the reference model to instigate noticeable adaptation. This is not the case in CG2 and CG3. From Figures 25 and 26 there is a stark difference between the nominal and adaptive systems. The nominal controllers allow for large excursions from the reference signal and in both uncertain scenarios the nominal controller allows the GTM to hit the ground at $\sim$ 10s.

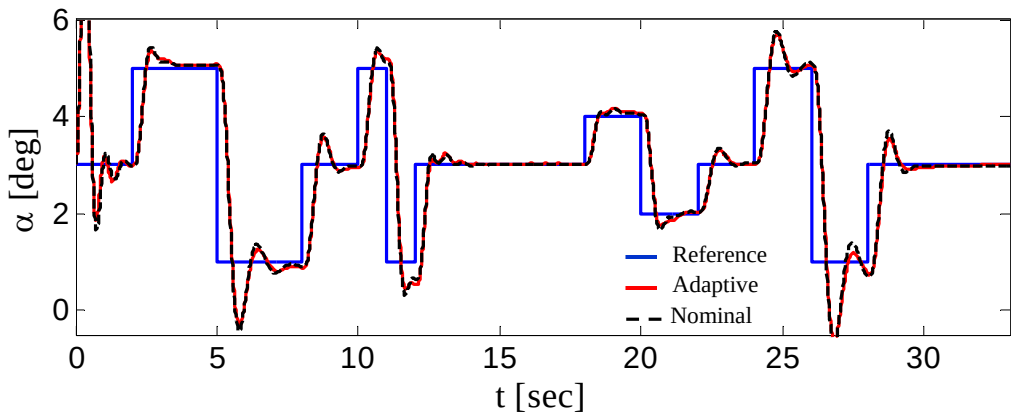

Figure 24. CG1 $\tau_{\text{uncert}} = 0.2ms$ $\lambda = -0.20$

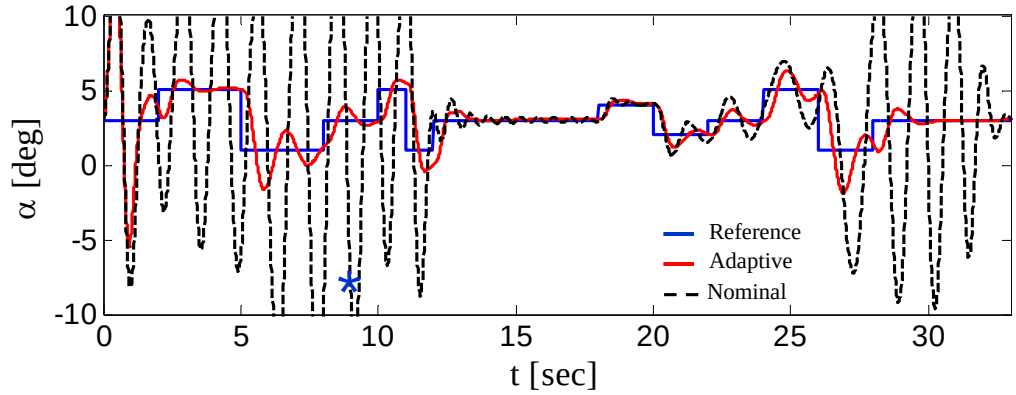

Figure 25. CG2 $\tau_{\text{uncert}} = 0.2ms$ $\lambda = -0.35$

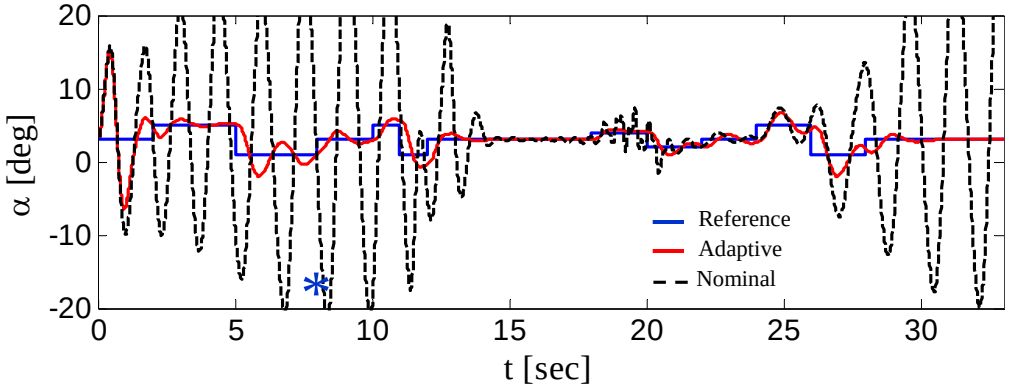

Figure 26. CG3 $\tau_{\text{uncert}} = 0.2ms$ $\lambda = -0.40$

.

7 Concluding Remarks

The adaptive algorithm showed promising results on the high fidelity SIMULINK model. This architecture is suitable for further evaluation. The following are areas of concern and future work suggestions:

- A stability proof for the state resetting should be constructed.
- The implications of incorporating known time delays into the reference model must be analyzed.
- Extensive testing should be conducted at the resetting limit of the CAS system.
- Consider what will happen in the adaptive system if the saturation limit of the actuator is poorly known.
- Explore adaptive control with rate saturation

References

1. Shah, Gautam H.: Aerodynamic Effects and Modeling of Damage to Transport Aircraft. *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, AIAA-2008-6203, August 2008.

2. Bacon, Barton J.; and Gregory, Irene M.: General Equations of Motion for a Damaged Asymmetric Aircraft. *AIAA Atmospheric Flight Mechanics Conference and Exhibit*, AIAA-2007-6306, Aug. 20-23 2007, Hilton Head, South Carolina.

3. Jenkins, Dennis R.: *Hypersonic Before the Shuttle: A concise History of ther X-15 Research Plane*. Monographs In Aerospace History 18, National Aeronautics and Space Administration, Washington, D.C., 2000, SP-2000-4518.

4. Stillwell, Wendell H.: *X–15 Research Results*. Sp 60, National Aeronautics and Space Administration, Washington, D.C., 1965.

5. Dydek, Zachary T.: Guaranteed margins and performance for an adaptive flight control system and application on the X-15 research airplane. Master's thesis, Massachusetts Institute of Technology, 2007.

6. Thompson, M. O.: At the edge of Space: The X-15 Flight Program. Smithsonian Institution Press, Washington DC, 1992.

7. Lyapunov, A.M.: The General Problem of Motion Stability (orig. 1892 Russian). *Reprinted in Annals of Mathematics*, , no. 17, 1949.

8. Slotine, Jean-Jacques E.; and Li, Weiping: *Applied Nonlinear Control*. Prentice Hall Inc., Upper Saddle River NJ, 1991.

9. Bellman, R.: *Stability of Differential Equations*. Dover, New York NY, 1953.

10. LaSalle, J.P.; and Lefschetz, S.: *Stability of Lyapunov's Direct Method*. Academic Press, New York NY, 1961.

11. Coppel, W.A.: *Stability and Asymptotic Behavior of Differential Equations*. McGraw-Hill Book Company, New York NY, 1965.

12. Hahn, W.: *Theory and Application of Lyapunov's Direct Method*. Prentice Hall, Inc., Englewood Cliffs NJ, 1963.

13. Hahn, W.: *Stability of Motion*. Springer–Verlag, New York NY, 1967.

14. Krasovskii, N.N.: *Stability of Motion*. Stanford University Press, Stanford CA, 1963.

15. Anderson, B.D.O.: *Network Analysis and Synthesis: A Modern Systems Theory Approach*. Prentice Hall, Inc., Englewood Cliffs NJ, 1973.

16. Narendra, Kumpati S.; and Annaswamy, Anuradha M.: *Stable Adaptive Systems*. Dover Publication, Inc., Mineola New, York, 1989.

17. Karason, S. P.; and Annaswamy, A. M.: Adaptive Control in the Presence of Input Constraints. *IEEE Transactions on Automatic Control*, vol. 39, May-June 1994, pp. 2325–2330.

18. Schwager, Mac; and Annaswamy, Anuradha M.: Direct Adaptive Control of Multi-Input Plants with Magnitude Saturation Constraints. *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference*, Seville, Spain, December 12-15 2005.

19. Schwager, Mac: Towards Verifiable Adaptive Control for Safety Critical Applications. Master's thesis, Massachusetts Institute of Technology, 2005.

20. Jang, J.; Annaswamy, A. M.; and Lavretsky, E.: Adaptive Flight Control in the Presence of Multiple Actuator Anomalies. *Proceedings of the 2007 American Control Conference*, New York, NY,20-23 August 2007, pp. 3300–3305.

21. Jang, Jinho: Adaptive Control Design with Guaranteed Margins for Nonlinear Plants. Ph.D. thesis, Massachusetts Institute of Technology, 2008.

22. Annaswamy, Anuradha M.; and Jang, Jinho: Stability Margins for Adaptive Gain-Scheduled Controllers in the Presence of Time-Delay. *AIAA Guidance, Navigation, and Control Conference and Exhibit*, 18-21 August 2008, Honolulu,Hawaii.

23. Annaswamy, Anuradha M.; and Jang, Jinho: Adaptive Gain-Scheduled Controllers in the Presence of Actuator Anomalies. *AIAA Guidance, Navigation, and Control Conference and Exhibit*, 18-21 August 2008, Honolulu,Hawaii.

24. Stengel, Robert F.: *Flight Dynamics*. Princeton University Press, Princeton and Oxford, 2004.

Appendix A

Control Design Free Parameters

The design parameters for the CAS control loop (angles in degrees):

$$K_{e_\alpha}^{\delta_e} = 0.9 * 4.6357 \tag{A1}$$
$$K_{\alpha}^{\delta_e} = 0.7489 \tag{A2}$$
$$K_{\theta}^{\delta_e} = 0 \tag{A3}$$
$$K_{q_w}^{\delta_e} = 0.1841 * 0.7 \tag{A4}$$

The design parameter for the SAS roll loop (angles in degrees):

$$K_{p_w}^{\delta_a} = 0.1 \tag{A5}$$
$$K_{\phi}^{\delta_a} = 0.1 \tag{A6}$$

The design parameters for the SAS yaw loop (angles in degrees):

$$K_{r_w}^{\delta_r} = 0.3 \tag{A7}$$
$$K_{\psi}^{\delta_r} = 0.1 \tag{A8}$$

Adaptive design parameters:

$$\mathbf{\Gamma} = 1 \times 10^{1.8} * \mathrm{eye}(12, 12) \tag{A9}$$
$$\boldsymbol{\sigma} = 100 * \mathrm{eye}(12, 12) \tag{A10}$$
$$\mathbf{Q} = \mathrm{diag}([10 \;\; 0.1 \;\; 1e^{-2} \;\; 1e^{-2} \;\; 1e^{-2} \;\; 1e^{-1} \;\; 1e^{-2} \;\; 1e^{-1} \;\; 1e^{-2} \;\; 1e^{-2} \;\; 1e^{-2} \;\; 10]') \tag{A11}$$