LOAD RELIEF OPTIMIZATION OF A FIRST STAGE BOOST VEHICLE
USING AN AUTOMATED COMPUTER DESIGN TOOL

by

CARLOS S. MORENO

S.B., Massachusetts Institute of Technology

(1986)

SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE
DEGREE OF

MASTER OF SCIENCE IN
AERONAUTICS AND ASTRONAUTICS

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 1988

© Carlos Sanchez Moreno, 1988

Signature of Author _____
Department of Aeronautics and Astronautics
May 6, 1988

Certified by _____
Mr. Gilbert S. Stubbs
Technical Advisor

Certified by _____
Prof. William K. Durfee
Thesis Advisor, Department of Mechanical Engineering

Accepted by _____
Prof. Harold Y. Wachman
Chairman, Departmental Graduate Committee

1

ABSTRACT

This thesis presents the development and application of an Automated Control Loop Design Tool (ACLDT). The ACLDT was applied to the digital autopilot of a first stage boost vehicle with a single gimballed engine at the maximum dynamic pressure point of the trajectory, with the purpose of maximizing the load relief performance of the control system. The purpose of load relief feedback for a boost vehicle is to reduce aerodynamic loads on the vehicle during atmospheric flight by minimizing the angle of attack.

The design tool automates the lengthy process of fine tuning control loop parameters through the use of linear programming methods. Subject to stability margin constraints, the ACLDT minimized the maximum vehicle angle of attack in a simulation of the vehicle transient response to a ramped wind input profile. Since the aerodynamic loads on the vehicle are proportional to the angle of attack, this maximum angle of attack excursion gave a reasonable measure of load relief performance.

Two different load relief configurations were compared. The first, or conventional, load relief design used only the measured change in velocity normal to the vehicle longitudinal axis, ΔV_N, as a load relief signal. The second, or alternative, load relief design used the ΔV_N signal, as well as the measured engine nozzle deflection angle, δ, for load relief feedback. In the six optimizations presented in this thesis, both the conventional and the alternative load relief designs improved their load relief performance by 26 to 52 percent over the designs at the start of the optimization runs. The optimized, alternative load relief designs provided better load relief performance than the optimized, conventional load relief, but the improvement was only 6 to 13 percent. Stability margins were not compromised by the improvement in load relief performance; moreover, the stability margins were not significantly reduced when uncertainties in vehicle thrust and aerodynamic coefficients were introduced.

ACKNOWLEDGEMENT

I hereby assign my copyright of this thesis to the Charles Stark Draper Laboratory, Inc., Cambridge, Massachusetts.

Carlos Sanchez Moreno

Permission is granted by the Charles Stark Draper Laboratory, Inc. to the Massachusetts Institute of Technology to reproduce any or all of this thesis.

TABLE OF CONTENTS

Appendix   **Page**

LIST OF ILLUSTRATIONS

Figure   **Page**

LIST OF TABLES

CHAPTER 1

INTRODUCTION

This thesis presents the development and application of a computer design tool to be used for the optimization of a control loop for a flexible space boost vehicle with a single gimballed engine. The computer algorithm, the Automated Control Loop Design Tool (ACLDT), automates the tedious and lengthy iterative process of fine tuning the control loop parameters.

This design tool applies linear programming methods to meet user-defined stability margin requirements and to maximize load relief performance. Load relief performance is measured by calculating the maximum increase in vehicle angle of attack in response to a wind input profile.

The overall program consists of four parts, as shown in Figure 1-1. The first part is the interface program which controls the entire design process. It calls up three subroutines which make up the other three parts: a frequency response program, a transient response program, and a linear programming optimizer. The frequency response subroutine incorporates a mathematical model of the control loop and calculates open loop gain and phase margins of the entire loop. The time domain transient response subroutine models the vehicle equations of motion and computes the dynamic response of the vehicle to a wind input at the maximum dynamic pressure point of the vehicle trajectory. The transient response simulation is limited to two dimensions in the trajectory (pitch) plane. Both of these programs contain models for the first seven bending modes of the flexible vehicle.

The interface program uses the frequency and transient response subroutines to create sensitivities of the control margins and of the dynamic response cost to the individual control

parameters. The linear optimizer subroutine takes the linearized sensitivities and, subject to gain and phase margin constraints, minimizes the cost. For this problem, the cost is defined to be the maximum magnitude of the vehicle angle of attack in response to the wind profile calculated in the transient response subroutine.

THE AUTOMATED DESIGN PROCESS

The process begins with an initial guess of values for the control system parameters supplied by the user. The interface program linearizes the problem by calculating sensitivities of each of the control margins and of the transient load relief performance to each of the respective control parameters. The values of these sensitivities are then passed to the linear optimizer subroutine.

The optimizer program uses a variation of the Simplex linear programming method (Ref 1-3) to first drive the system to meet user-defined phase and gain margin constraints. Once the constraints are all met, the program changes the values of the control system parameters in such a way as to improve load relief performance while maintaining the stability margin constraints. The problem is then relinearized about the control variable values and the process repeated until an optimum solution is reached.

ALTERNATIVE LOAD RELIEF

This thesis uses the automated design process to compare the performance of two control systems which employ alternative load relief methods. One method considered is the conventional approach of load relief which reduces the overall vehicle angle of attack by

$\underline{X}$,
MLPL,
Sensitivities

Interface
Program

$\underline{X}$

New $\underline{X}$

$\underline{X}$

Margin
Values

Cost

Linear
Optimizer

$\underline{X}$

Frequency
Response

Transient
Response

Margin
Values
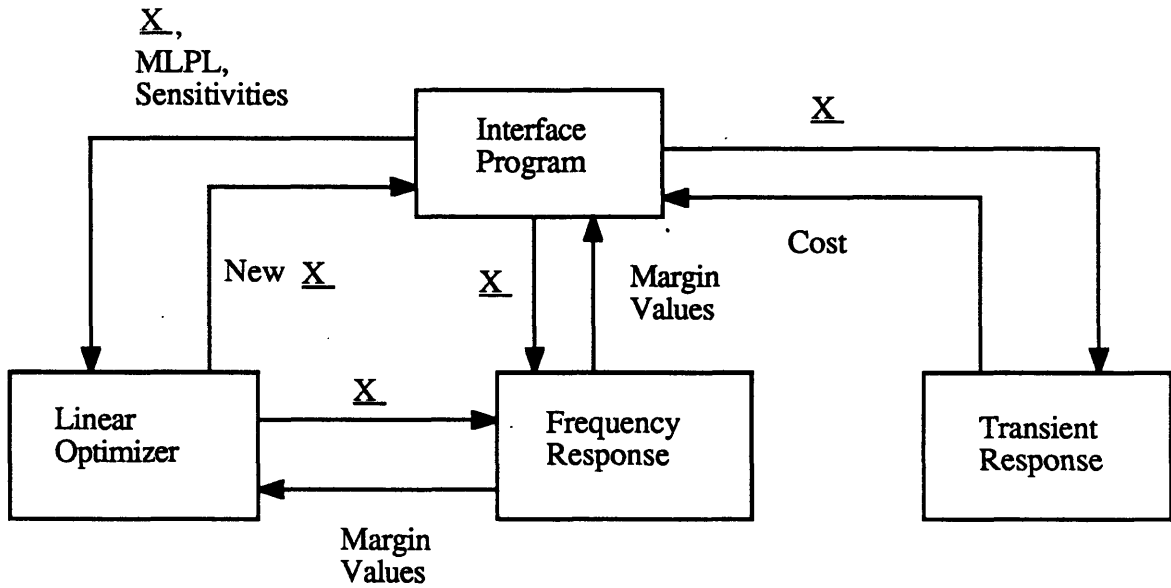
Figure 1-1. Program configuration

using the measured change in velocity normal to the vehicle longitudinal axis during the control sample period as a measure of the steady-state angle of attack. The other method of load relief considered adds the measured engine nozzle deflection as another measure of vehicle angle of attack in the feedback path. The weighted combination of these two signals is then used as an estimate of steady-state angle attack for load relief purposes.

Control loop optimizations were performed for six cases. The first two cases optimize a vehicle control loop which has a simple first order forward loop compensation for both types of load relief. The next two cases optimize a vehicle control loop that has a fourth order filter in the forward loop for both types of load relief. The final two cases also use a fourth order forward loop filter, but are designed for a trajectory having a maximum dynamic pressure much higher than the nominal case.

The alternative load relief control design gave better load relief performance than the conventional load relief design in each of the optimizations. The added information from the nozzle deflection angle made it possible to increase load relief performance with as large or larger stability margins. Nevertheless, the improvement using the alternative load relief over the conventional load relief was not dramatic. All of the optimized designs presented in this thesis give significantly better load relief performance than the initial starting point designs.

CHAPTER 2

SYSTEM DESCRIPTION

VEHICLE DESCRIPTION

The vehicle modelled in this thesis is the first stage of a multi-stage space boost vehicle which uses high thrust solid rocket motors and one gimballed engine nozzle per stage. The control loop optimization is carried out for the maximum dynamic pressure point in the trajectory during the first stage of flight. The vehicle is assumed at this point to be in steady-state conditions, with a constant center of pressure (cp) and center of gravity (cg). The nozzle hinge point (hp), the cg, and the cp are all assumed to lie on the vehicle roll axis. The vehicle dynamics are modelled in only a single (pitch) plane. The mass of the vehicle is assumed to be evenly distributed about the roll, pitch, and yaw axes. Consequently, all of the vehicle products of inertia are zero.

The vehicle is assumed to have no roll and no yaw motion, with the pitch axis of the gimballed nozzle perpendicular to the trajectory plane. Since vehicle motion is modelled only in the trajectory (pitch) plane, engine motion is limited to deflections about the pitch axis. The engine thrust is assumed to be constant, with a constant specific impulse. The nozzle deflection angle (δ) is limited to six degrees relative to the roll axis, and the nozzle deflection rate is limited to 40 deg/sec. The nozzle actuator dynamics are modelled using a linear second order model with a natural frequency of 10 Hz (62.8 rad/sec), and a damping ratio of 0.5.

The vehicle flexible-structure model includes terms for the first seven bending modes. However, in this thesis, vehicle stability margin constraints are computed for only the first

two bending modes. The first bending mode, which occurs at 10.775 Hz (67.7 rad/sec) is phase stabilized, and has a phase margin requirement of 60 degrees.[1] The second bending mode at 19.535 Hz (122.742 rad/sec) has a gain margin requirement of -10 db. The gain margins are defined as the open loop gain at -180 degrees phase, and the phase margins are defined as the phase at 0 db gain. The second bending mode gain margin is defined as the maximum open loop gain that occurs in the frequency range associated with the second bending mode natural frequency. The rigid body and bending mode stability margins are defined in Figure 2-1.

The on-board Inertial Measurement Unit (IMU) senses and provides values for vehicle attitude and changes in inertial velocity. A rate estimator, discussed below in the Control Loop Description, provides estimated values of angular rate.

---

[1] Note: The "first bending mode" in fact consists of two separate bending modes at frequencies of 10.775 and 10.800 Hz. Since these frequencies are so close, these two modes act effectively as one mode in the frequency response and are referred to as the first bending mode. The next bending mode at 19.535 Hz is termed the second bending mode in this thesis.

FREQUENCY RESPONSE

OPEN LOOP E/E
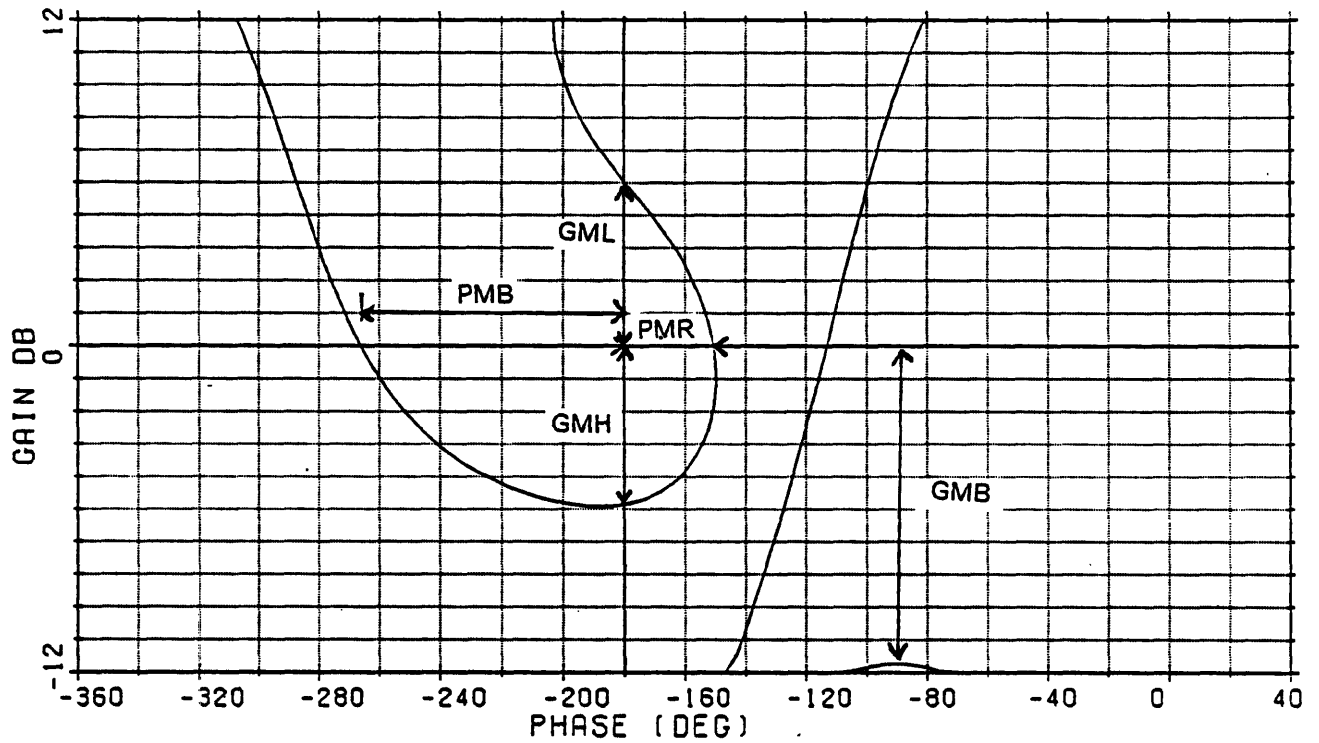
Figure 2-1. Stability Margin Constraints

GML = Low Frequency Rigid Body Gain Margin

PMR = Rigid Body Phase Margin

GMH = High Frequency Rigid Body Gain Margin

PMB = First Bending Mode Phase Margin

GMB = Second Bending Mode Gain Margin

VEHICLE DYNAMICS

The vehicle equations of motion sum the effects of three types of forces: the main engine thrust, the aerodynamic forces, and the reaction forces on the vehicle from angular acceleration of the engine nozzle, $\ddot{\delta}$, (or "tail wags dog" effect). These forces are shown in Figure 2-2. The effects of bending are shown in Figure 2-3.

The linear acceleration of the vehicle cg can be described by the following relationships:

$$a'_{cg,y} = \left[S\,Q\,C_N + m_n\,l_n\,\ddot{\delta} - F\left\{ \sin\delta + \sum_{i=1}^{7} \sigma_{NHL,i}\,q_i \right\} \right] / M$$

$$C_N = f_1\,(Mach\ number,\ \alpha)$$

$$a'_{cg,x} = [F\cos\delta - S\,Q\,C_A]/M$$

$$C_A = f_2\,(Mach\ number,\ \alpha)$$

$$a_{cg,y} = a'_{cg,y} - g\,\cos\theta$$

$$a_{cg,x} = a'_{cg,x} - g\,\sin\theta$$

The rigid body angular acceleration of the vehicle cg is described by the equation:

$$\ddot{\theta} = \left[F\, l_{cg} \sin\left(\delta \; - \; \sum_{i=1}^{7} \sigma_{NHL,i} q_i \right) \; - \; F \sum_{i=1}^{7} \phi_{NHL,i} q_i + S\, Q\, C_N\, l_{cp} + (m_n\, l_n\, l_{cg} + J_n)\ddot{\delta} \right] / I_{yy}$$

This can be rewritten using small angle approximations as

$$\ddot{\theta} = \left[F\, l_{cg} \sin \delta + S\, Q\, C_N\, l_{cp} + (m_n\, l_n\, l_{cg} + J_n)\ddot{\delta} - \sum_{i=1}^{7} F\, (l_{cg}\sigma_{NHL,i} + \phi_{NHL,i}) q_i \right] / I_{yy}$$

The variables and constants in the above equations are defined as follows:

$a'_{cg,y}$ = Linear Acceleration of CG in Body Y-direction not including Gravity

$a'_{cg,x}$ = Linear Acceleration of CG in Body X-direction not including Gravity

$a_{cg,y}$ = Total Acceleration of CG in Body Y-direction including Gravity

$a_{cg,x}$ = Total Acceleration of CG in Body X-direction including Gravity

$\ddot{\theta}$ = Rigid Body Angular Acceleration about the Vehicle CG

θ = Rigid Body Attitude Angle at CG

α = Angle of Attack

M = Vehicle Mass

F = Vehicle Thrust

g = Acceleration Due to Gravity

I_{yy} = Vehicle Moment of Inertia

S = Vehicle Reference Area

Q = Dynamic Pressure

C_N = Aerodynamic Normal Force Coefficient

C_A = Aerodynamic Axial Force Coefficient

δ = Engine Nozzle Deflection Angle

$\ddot{\delta}$ = Engine Nozzle Angular Acceleration

m_n = Engine Nozzle Mass

J_n = Engine Nozzle Moment of Inertia

l_n = Distance from Nozzle CG to Nozzle Hinge Line (NHL)

l_{cg} = Distance from Vehicle CG to Nozzle Hinge Line

l_{cp} = Distance from Vehicle CG to Center of Pressure

σ_{NHL} = Slope of Vehicle at NHL due to Bending

ϕ_{NHL} = Displacement of Vehicle at NHL due to Bending

q_i = Generalized Modal Displacement Coordinate

The sensed acceleration at the IMU including bending effects at the IMU is given in the following equations:

$$a_{IMU,x} = a'_{cg,x} - l_{IMU}\ \dot{\theta}^2$$

$$a_{IMU,y} = a'_{cg,y} + l_{IMU}\ \ddot{\theta} + \sum_{i=1}^{7} \sigma_{IMU,i}\ \ddot{q}_i$$

where

$$\ddot{q}_i = -2\ \zeta_{bend,i}\ \omega_{bend,i}\ \dot{q}_i - \omega_{bend,i}^2\ q_i - \frac{F\ \phi_{NHL,i}}{GM}\ \delta - \frac{\left[\ m_n\ l_n\ \phi_{NHL,i} - J_n\ \sigma_{NHL,i}\ \right]}{GM}\ \ddot{\delta}$$

Attitude and attitude rate at the IMU including the effects of bending rotation at the IMU are given by:

$$\theta_{IMU} = \theta + \sum_{i=1}^{7} \sigma_{IMU,i}\ q_i$$
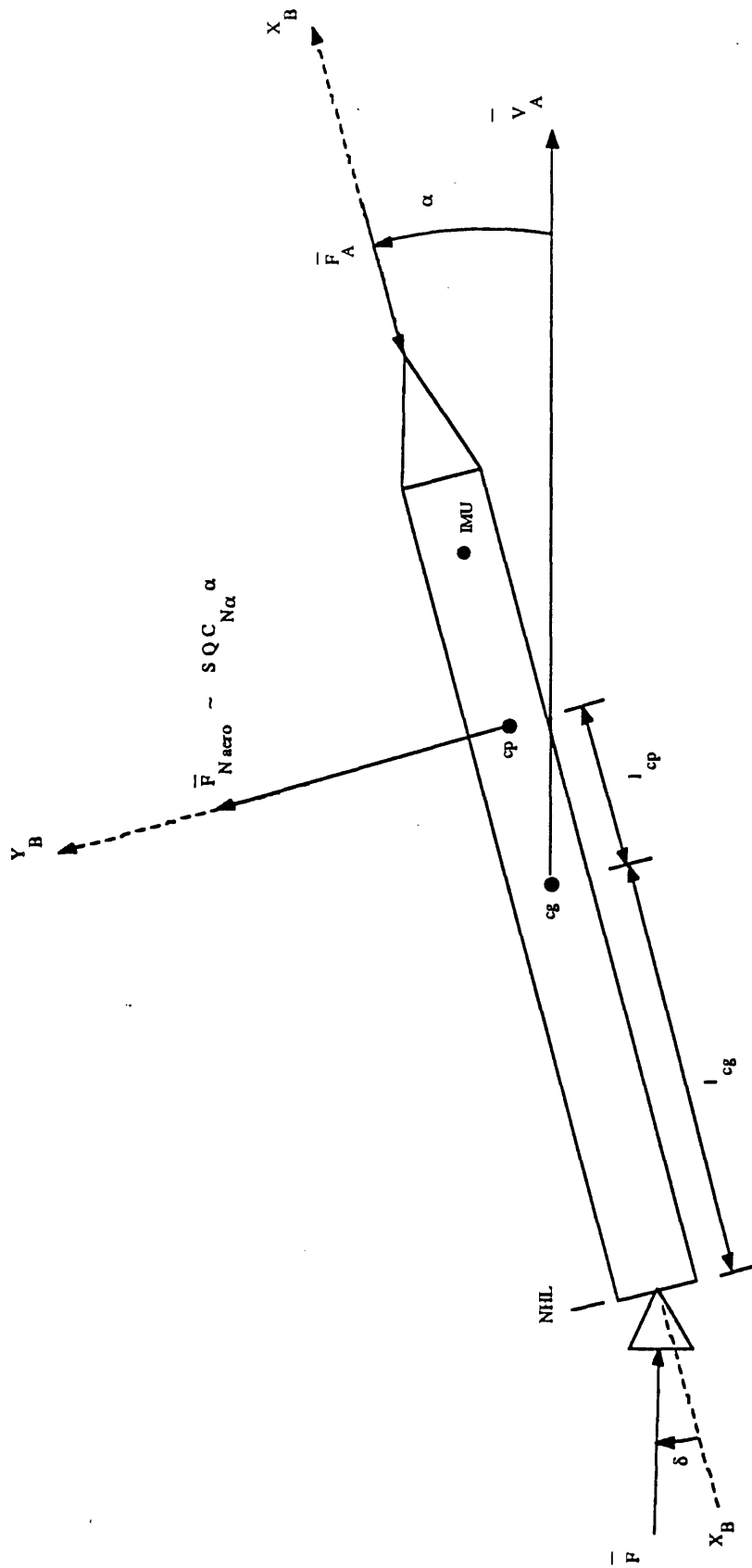
Figure 2-2. Forces on the Vehicle
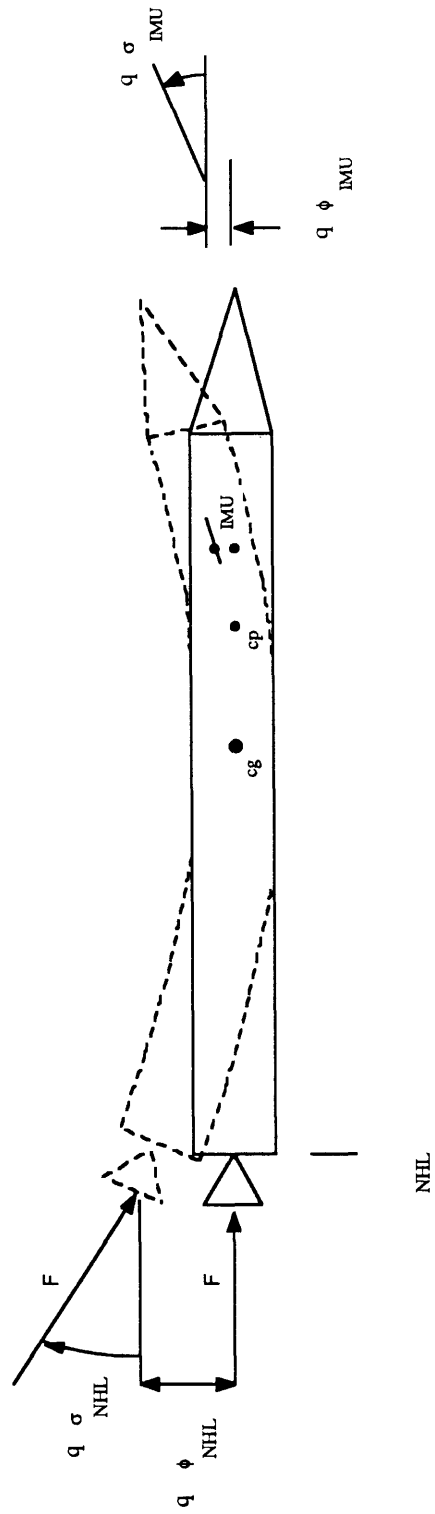
Figure 2-3. Effects of Bending at the IMU and NHL

$$\dot{\theta}_{IMU} = \dot{\theta} + \sum_{i=1}^{7} \sigma_{IMU,i} \, \dot{q}_i$$

Variables and constants of the above additional equations are defined as follows:

$a_{IMU,x}$ = Linear Acceleration at the IMU in Body X-Direction

$a_{IMU,y}$ = Linear Acceleration at the IMU in Body Y-Direction

θ = Rigid Body Attitude Angle at CG

$\dot{\theta}$ = Rigid Body Rate About CG

l_{IMU} = Distance from Vehicle CG to IMU

σ_{IMU} = Slope of Vehicle at IMU due to Bending

ϕ_{IMU} = Displacement of Vehicle at IMU due to Bending

$\zeta_{bend,i}$ = Damping Ratio of i th bending mode

$\omega_{bend,i}$ = Natural Frequency of i th bending mode

GM = Generalized Mass of Vehicle

q_i = Generalized Modal Displacement Coordinate

$\dot{q}_i$ = Generalized Modal Velocity

$\ddot{q}_i$ = Generalized Modal Acceleration

CONTROL LOOP DESCRIPTION

Any boost vehicle in atmospheric flight without a control system is inherently unstable. This is the case because the aerodynamic center of pressure lies forward of the vehicle center of gravity, as can be seen in Figure 2-4. In this configuration, a perturbation in the vehicle's aerodynamic angle of attack will result in a change in vehicle angular acceleration that tends to drive the angle of attack in the same direction as the perturbation. Thus, the prima-

ry purpose of the autopilot control loop is to stabilize the vehicle in the presence of perturbations in aerodynamic angle of attack (Δa) The secondary purpose of the control loop is to control the attitude of the vehicle to the orientation necessary to maintain the vehicle flight trajectory.

For the simulations used in this thesis, a digital autopilot is used to generate discrete-time engine nozzle deflection commands (δ_c) for the engine nozzle actuators. The deflection of the engine thrust provides the torques on the vehicle necessary to counter the torques produced by aerodynamic perturbation forces, as well as to drive the vehicle to its commanded inertial attitude. New nozzle deflection commands are calculated every 10 msec by the autopilot based on measured and estimated data.

Since the simulations in this thesis are limited to pitch plane dynamics, all nozzle deflections and commands occur about the vehicle pitch axis only. Positive nozzle deflections are defined to produce positive angular accelerations about the vehicle pitch axis.

The control system block diagram for the boost vehicle in this thesis is shown in Figure 2-5. The loop represents a Single Input-Single Output (SISO) system in which the input is a commanded vehicle inertial attitude (θ_c), and the output is the sampled, measured vehicle inertial attitude (θ_m). Two other important measured quantities are the sampled, measured engine nozzle deflection angle (δ), and the sampled, measured change in normal velocity of the vehicle (ΔV_N). The vehicle attitude command (θ_c), is calculated by a steering loop every 500 msec. The purpose of θ_c is to produce the desired trajectory by steering the vehicle to some desired acceleration vector direction.
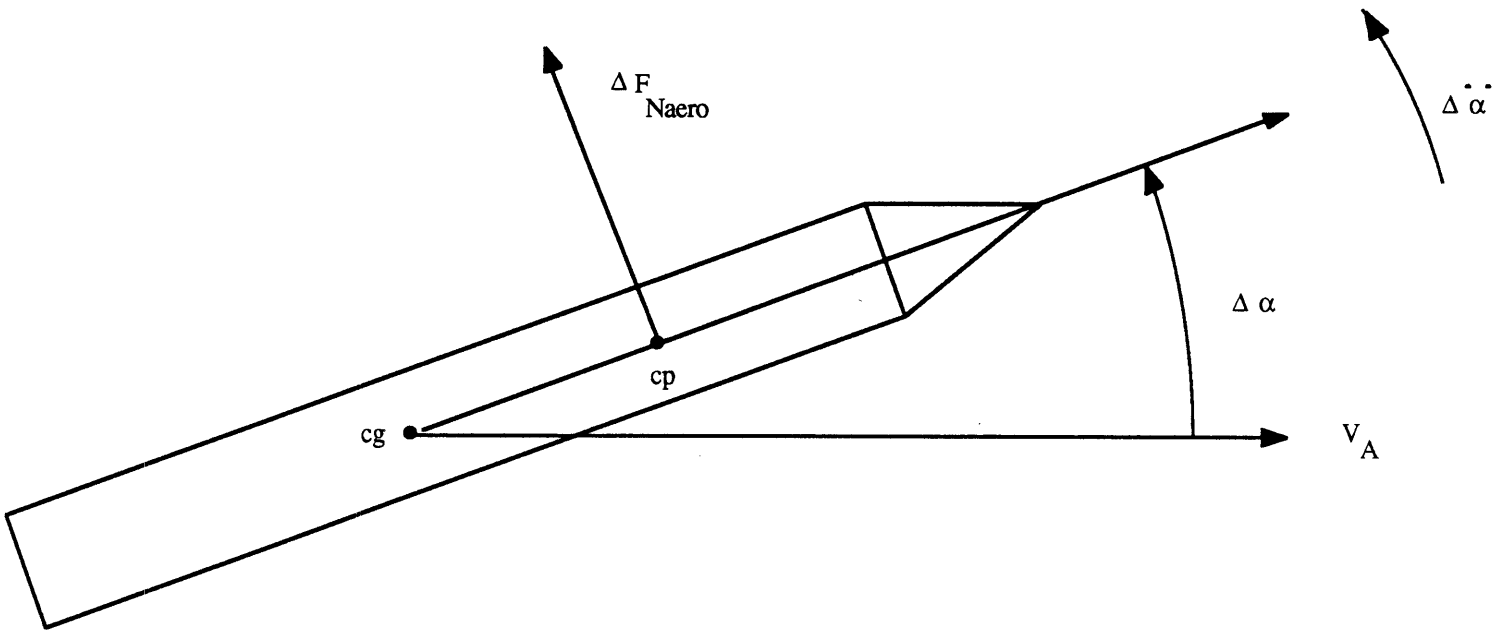
Figure 2-4. Effect of Aerodynamic Perturbations
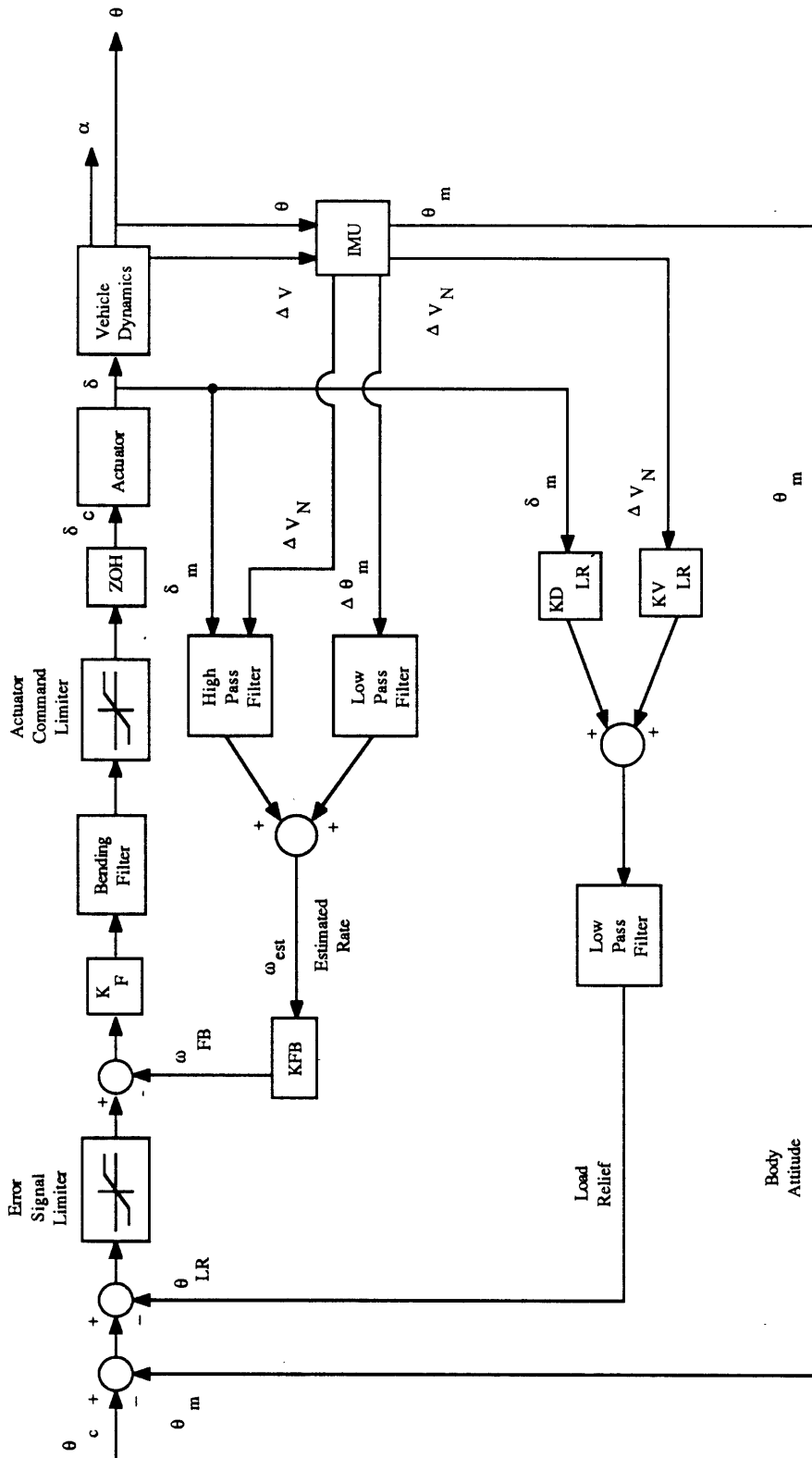
Figure 2-5. Control System Block Diagram

30

There are three different feedback signals used in the control system represented in Figure 2-5. The primary feedback signal is a simple, unmodified attitude feedback path which sends attitude measurements, θ_m, from the vehicle inertial measurement unit (IMU) to the flight control computer. The second feedback signal is from a rate estimator that combines θ_m, ΔV_N, and δ into an estimated rate signal (ω_{FB}). The third feedback signal is called the load relief signal. The purpose of this signal is to minimize structural loads on the vehicle by reducing the aerodynamic angle of attack (α). The two forms of load relief studied in this thesis use different types of load relief signals. The first form employs a conventional load relief feedback path that uses only the ΔV_N signal for load relief. The second form, called alternative load relief feedback, combines the ΔV_N and the δ signals for load relief purposes.

As can be seen in Figure 2-5, the vehicle attitude command signal is first modified by subtracting the current measured vehicle attitude to produce an error signal. Next, the load relief signal (θ_{LR}) is subtracted from the error signal, and the modified error is passed through a magnitude limiter. Then the rate signal, ω_{FB}, is subtracted from the limited error and this total error is multiplied by a constant forward path gain (K_F) and passed through a digital bending filter. The output of the bending filter is sent into a "bucket" limiter which limits the total commanded nozzle deflection to a value corresponding to the physical saturation limit of the engine nozzle actuator. The limiter output is the nozzle deflection command, δ_c, which is passed through a zero-order hold (ZOH) circuit to the pitch nozzle actuator. The ZOH maintains a constant δ_c until the autopilot calculates a new signal at the next 10 msec cycle.

LOAD RELIEF

The aerodynamic forces on a boost vehicle can become very large as the vehicle accelerates through the atmosphere. The normal aerodynamic force, F_{Naero}, on any vehicle traveling with an air-relative velocity, V_A through the atmosphere is equal to

$$F_{Naero} = 0.5\,\rho\,V_A^2\,S\,C_N$$

or

$$F_{Naero} \cong 0.5\,\rho\,V_A^2\,S\,C_{N\alpha}\,\alpha \quad \text{(for small values of } \alpha)$$

where

ρ = Local Air Density

V_A = Air-Relative Velocity Magnitude

S = Vehicle Reference Area

C_N = Aerodynamic Normal Force Coefficient

$C_{N\alpha}$ = Normal Force Stability Derivative

α = Aerodynamic Angle of Attack

As can be seen in the small-α equation, F_{Naero} increases with α, and becomes zero at zero angle of attack. The purpose of load relief is to prevent the combination of normal forces applied to the vehicle by both aerodynamic forces and the deflected thrust vector from creating damaging loads on the main structure. Since the only controllable variable in the normal aerodynamic force equation is α, the load relief feedback loop reduces structural loads by reducing the angle of attack.

The dynamic pressure, Q, is defined as

$$Q = 0.5 \, \rho \, V_A^2,$$

so that the above equation can be rewritten for small a values as

$$F_{Naero} \cong Q \, S \, C_{N\alpha} \, a.$$

As the vehicle accelerates and gains in altitude, its velocity increases while the local air density decreases. A point at which the dynamic pressure reaches its maximum, called the max-Q point of the trajectory, is reached during the first stage of boost flight. A typical dynamic pressure profile is shown in Figure 2-6. The load relief performance of the control system is most critical at this max-Q point since large structural loads on the vehicle can result from small variations in angle of attack.

Previous load relief feedback systems have used only one feedback signal, the ΔV_N signal, which represents the measured acceleration of the vehicle normal to the vehicle longitudinal axis. The justification for use of the ΔV_N signal for purposes of load relief is that it provides an estimate of the low frequency, or steady-state, angle of attack, a_{ss}. This estimate of a_{ss} is given by the following equation (derived in Appendix A):

$$a_{ss} \cong \frac{m \, l_{cg}}{S \, Q \, C_{N\alpha} \, (l_{cp} + l_{cg})T} \, \Delta V_N$$

where T is the control sampling period.

An alternative expression for estimated steady state angle of attack is given by the following expression (derived in Appendix A) in terms of the nozzle deflection, δ,

$$a_{ss} \cong -\frac{F \, l_{cg}}{S \, Q \, C_{N\alpha} l_{cp}} \, \delta$$

BOOST PLOT

DYNAMIC PRESSURE PROFILE
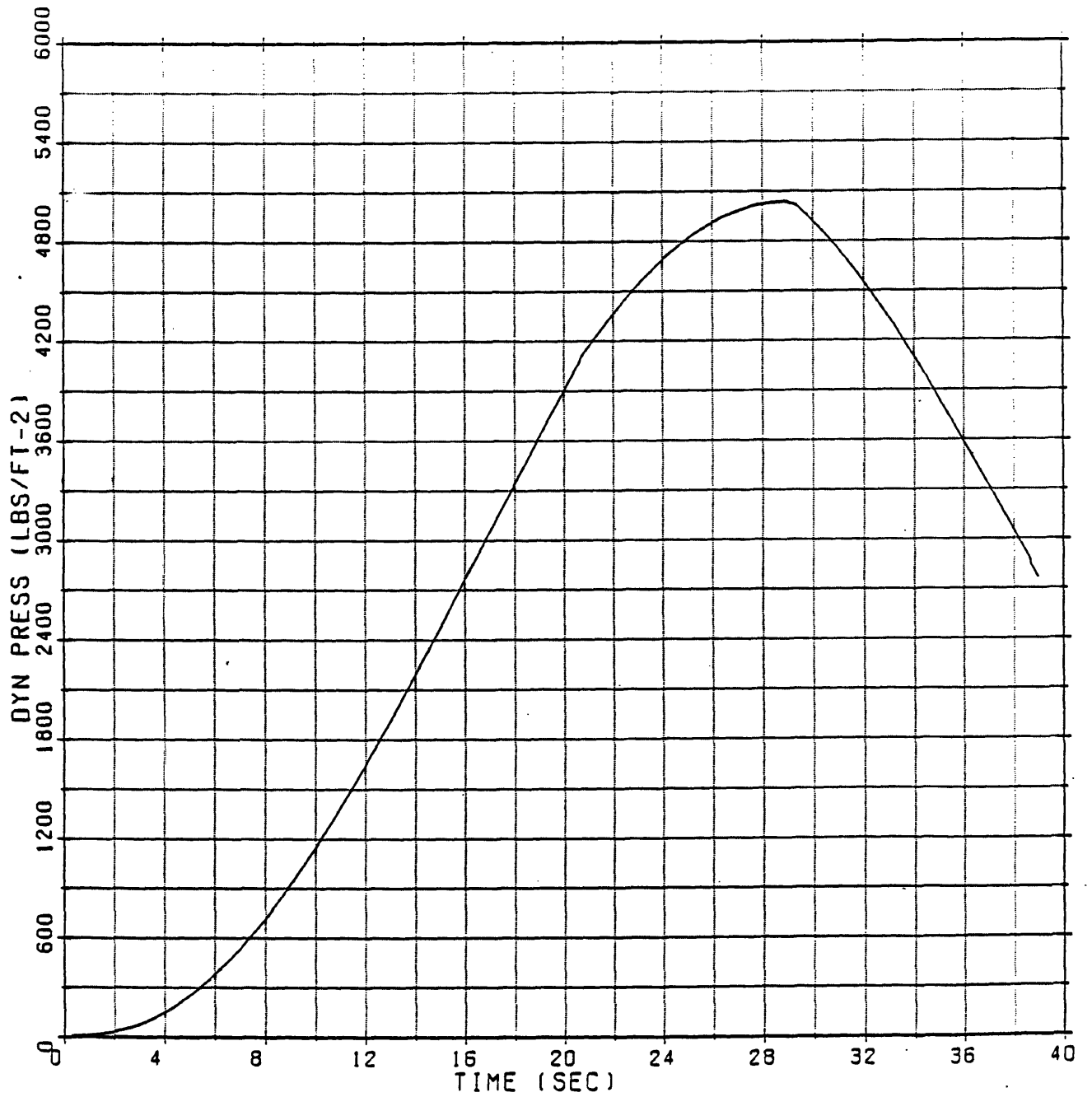
Figure 2-6. Dynamic Pressure Profile

Therefore, a mixture of the ΔV_N and δ signals in a linear combination represents a signal proportional to the steady state angle of attack and can be considered as an alternative to the use of the ΔV_N signal alone.

For either approach, the estimate is passed through a gain and a low-pass filter to generate the load relief feedback signal. The load relief gain and time constant as well as the other parameters in the control loop of the forward loop compensation and the rate estimator are chosen so that the specified stability margins are met and the load relief is maximized. The set of control parameters will be different for the combined ΔV_N and δ load relief signals than for the ΔV_N signal alone. Consequently, the control and load relief performance will differ for the two systems. The combined ΔV_N and δ load relief approach has the potential for improved performance over the approach using the ΔV_N signal alone. In any case, the optimum combination of ΔV_N and δ can do no worse than the ΔV_N signal alone since the latter approach represents a special case of the combined approach.

THE RATE ESTIMATOR

The rate estimator feedback loop utilizes all three of the vehicle variables (θ_m, δ, and ΔV_N) to calculate an estimate of the vehicle angular rate. A low frequency estimate of angular rate is computed using the change in measured body inertial attitude:

$$\dot{\theta}_{low} = \frac{\Delta \theta_m}{T}$$

where T is the autopilot sampling period (10 msec), and $\dot{\theta}_{low}$ is called the derived rate.

Because there are large fluctuations in the derived rate signals due to Inertial Measurement Unit (IMU) noise and quantization effects, the derived rate signal is passed through a

first-order low-pass filter. However, the low-pass filter introduces an undesirable lag in the estimate of angular rate.

In order to eliminate this lag, the output of the low-pass filter is summed with a high frequency estimate of $\dot{\theta}$ which has been passed through a "complementary" high-pass filter (i.e. the sum of the low-pass and high-pass filters equals unity). A linear combination of the ΔV_N and δ signals provides a high-frequency rate estimate of the *change* in the angular rate ($\Delta\dot{\theta}_{high}$) over the previous control sampling period. This relationship is derived from two vehicle equations of motion -- the angular acceleration equation and the equation for normal acceleration at the vehicle IMU. The high-frequency rate estimate is formed as the sum of the previous estimate and the estimated change in rate:

$$\dot{\theta}_{high}(t_i) = \dot{\theta}_{high}(t_{i-1}) + \Delta\dot{\theta}_{high}(t_i)$$

where t_i is the ith sampling period. The sum of the low and high frequency rate estimates provides a good approximation of the true vehicle angular rate with no filtering lags.

CHAPTER 3

THE AUTOMATED CONTROL LOOP DESIGN TOOL

THE FREQUENCY RESPONSE SUBROUTINE

Open loop frequency responses are calculated for the control system described in Chapter 2 in the frequency response subroutine. This control system, shown in Figure 2-5, represents a sampled-data feedback control system for the boost vehicle studied in this thesis.

The continuous time domain transfer functions in the control system are the engine nozzle actuator model and the vehicle dynamics model. The discrete time domain transfer functions include the load relief filter, the rate estimator filters, and the forward path bending filter compensation. The discrete time input to the zero-order hold is the commanded nozzle deflection, δ_c. The sampled, continuous outputs of the vehicle are measured engine nozzle deflection, δ, measured normal velocity, V_N, and measured inertial body attitude, θ_m. The sampling period of the control loop is 10 msec.

For a given, fixed set of vehicle and environmental conditions, frequency responses are computed for the transfer functions $\dfrac{\delta^*}{\delta_c^*}$, $\dfrac{V_N^*}{\delta_c^*}$, and $\dfrac{\theta_m^*}{\delta_c^*}$, where δ^*, V_N^*, and θ_m^* represent sampled values of the continuous quantities $\delta(t)$, $V_N(t)$, and $\theta_m(t)$. These transfer functions are based on the differential equation for the actuator response to the nozzle command δ_c and on the vehicle equations of motion in response to δ_c linearized about the given set of vehicle and environmental conditions. The frequency responses for the three sampled transfer functions are computed for frequencies expressed in the discrete w-domain. The responses for these functions are then stored in a permanent file.

When the frequency response subroutine is called up by either the interface or optimizer subroutines, it is given the values of the gains, time constants, poles, and zeroes associated with the discrete time filters. The frequency response subroutine then combines the gains and phases of the discrete time filters with the three stored frequency responses for the actuator and vehicle dynamics, and computes an overall open loop $\frac{e}{e}$ gain and phase.

The frequency response is computed only for frequencies near a gain or phase crossover point so as to decrease the number of computations. The range of frequencies over which the frequency response is computed must be specified by the user based on engineering judgement. The gain and phase margins are computed, stored, and then returned to the program that called the frequency response subroutine.

THE TRANSIENT RESPONSE SUBROUTINE

The transient response subroutine is based on the vehicle equations of motion described in Chapter 2. The program implements a two-dimensional simulation of the vehicle's response to a wind input profile at the maximum dynamic pressure point of the trajectory. When this subroutine is called up, the vehicle is initialized to be in a steady-state condition, with zero angle of attack (a), zero angle of attack rate ($\dot{a}$), and zero angular acceleration ($\ddot{\theta}$). There is also no steering loop command error, because when the simulation begins, the vehicle is assumed to be at the desired inertial attitude. Interactions between the steering and control loops, which in reality have a small effect on the vehicle stability margins, have been neglected in both the frequency and transient responses. In fact, the steering command was run open loop in the transient response subroutine. The steering/control loop interactions are second order effects, and their examination is beyond the scope of this thesis. It is not unreasonable to expect that these interactions would be

similar for both the conventional and alternative load relief autopilots, and would therefore have no effect on the comparison of the two systems.

The wind input profile is a simplified one, in which the angle of attack due to wind is ramped up from zero to one degree over a period of 2.5 seconds and then ramped back down to zero for the same period, giving a total simulation time of 5 seconds. The vehicle equations of motion are integrated using a fourth-order Runge-Kutta integration algorithm every 5 msec. These equations include models for seven bending modes, the "Tail wags dog" effect, aerodynamic forces, and the deflected engine thrust. Deflection commands to the nozzle actuator, δ_c, are calculated from the control system variables, using the values of the control parameters in the forward loop and feedback compensation supplied by the interface program. A typical transient response is shown in Figure 3-1.

THE INTERFACE PROGRAM

The Automated Control Loop Design Tool (ACLDT) employs a linear optimizer routine to solve the nonlinear problem of computing an optimal set of control parameters. The intelligent link between the frequency response subroutine, the transient response subroutine, and the linear optimizer is known as the interface program. The interface program computes approximate linearizations of the nonlinear relationships between the control parameters and the stability margins, as well as between the control parameters and the load relief performance. These linearizations are referred to as the margin and cost sensitivities, respectively.

The interface program also controls the execution of the two iterative cycles which are called the Major and Minor Loop Cycles. A Major Loop Cycle can include up to six Minor

TRANSIENT RESPONSE

LOAD RELIEF PERFORMANCE
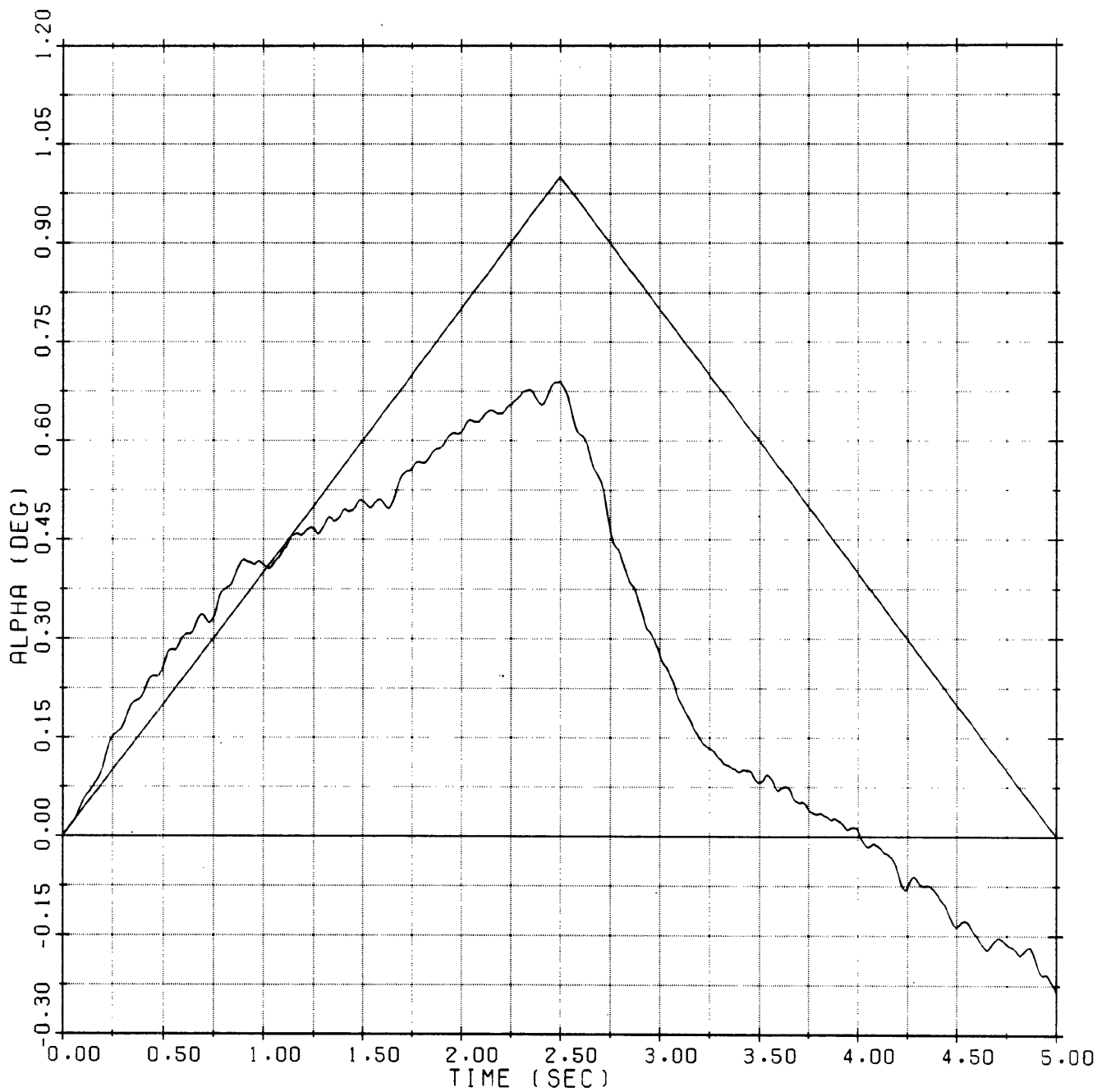
Figure 3-1. Typical Vehicle Transient Response

Loop Cycles, each of which use the same set of linearized sensitivities. These sensitivities are recomputed at the start of each Major Loop Cycle to insure sufficiently accurate modelling of the parameter relationships. Similar programs have been developed by Machles and Hauser (Reference 4).

The Major Loop Cycle

The first Major Loop Cycle begins at a starting point provided by the user. This starting point could be an initial guess at possible control loop parameter values, or it could be an existing design which needs to be improved. It begins by calling up the frequency response subroutine in order to determine the initial values of the control margins. Next, it uses the transient response subroutine to calculate the load relief performance, or cost. Once the current values of the control margins have been established, the Interface program checks each margin to see if the current value is greater than or equal to the specified margin constraint. If so, then the margin is considered to be meeting its constraint. A counter keeps track of how many of the control margins meet the constraints at any given time.

Linearizing the Parameter Relationships

Next, the interface program performs one of its primary functions by linearizing the control parameter relationships. It accomplishes this function by executing a loop in which each control parameter is incremented by a small amount. For the optimization study performed in this thesis, the best linearization increment was found to be 1.5 percent of the current value.

In the linearization routine, the interface program sends the perturbed value of each control parameter over to the frequency and transient response subroutines. It then com-

pares the new values of the control margins and maximum angle of attack (Max a) to the old ones and calculates a finite difference partial derivative, or sensitivity. Thus, with each execution of the linearization loop, the sensitivity of each control margin and of the cost to one control parameter is computed. For example, if there were five control margin constraints and eight control parameters, the linearization loop would be executed eight times, each time producing five constraint sensitivities and one cost sensitivity. Since one set of frequency response computations and one transient response simulation must be executed for each control parameter, the computational costs for each linearization are relatively high, and these costs increase with the number of parameters that are allowed to vary in the optimizaton.

The Minor Loop Cycle

Once the linearization is complete, the interface program begins a Minor Loop Cycle. The Minor Loop is extremely important because it is the key to maximizing the amount by which the control variables can change for any given linearization. In each Minor Loop Cycle, the interface program limits the allowed increments of the control parameters by specifying the *Minor Loop Percentage Limit* (MLPL), defined as the maximum percentage of their current values by which the control loop parameters are allowed to change. The MLPL starts at a small value and is increased by one percent each Minor Loop Cycle until a specified limit has been reached at which the linearized sensitivities are no longer valid.

The first Minor Loop Cycle begins by saving the number of constraints that have been met, and setting the MLPL equal to 2 percent of their current values. It then takes the absolute upper and lower limits for the control parameters which are provided by the user, and passes them to the optimizer subroutine. It also transmits the sensitivities calculated in the linearization loop, the current values of the control parameters, and the MLPL to the optimi-

zer. The Linear Optimizer described in Chapter 4 then executes a variation of a Simplex linear program, producing the optimal solution using the provided MLPL and linearized sensitivities. The new values for the control parameters are then used to perform frequency and transient responses. If the number of met constraints has increased, or if the number of met constraints has remained unchanged with an improvement in the cost, another Minor Loop Cycle is executed with a one percent larger MLPL. However, if the number of met constraints decreases, the control parameters are set back to the values computed in the previous Minor Loop cycle, and the interface program exits the Minor Loop. If and when the value of the MLPL reaches a limit of 7 percent, the Minor Loop is terminated, and the interface program decides whether to terminate the optimization, or to execute another Major Loop Cycle.

Optimization Termination

After the Minor Loop produces the optimal solution for a given linearization, the Interface program compares the results of the Minor Loop with the results of the last Major Loop Cycle. If all of the constraints were met for both the last and the present Major Cycles, and the improvement in load relief performance is below a given criterion, the optimization stops. However, if the performance is significantly improved, the program executes another Major Loop, relinearizing the problem around the new solution.

CHAPTER 4

THE LINEAR OPTIMIZER

LINEAR PROGRAMMING THEORY

Linear programming is a proven method for optimizing a linear function of several variables when the solution is subject to several linear constraints. The methods described here were developed extensively by G.B. Dantzig in the 1940's, and were originally used as a method of allocating limited resources in an optimal way. More detailed discussions of linear programming theory can be found in References 1-3.

The Linear Problem

The basic linear programming problem can be stated as follows:

Optimize

$$Z = c_1 x_1 + c_2 x_2 + \ldots + c_n x_n$$

Subject to

$$g_1 = a_{11} x_1 + a_{12} x_2 + \ldots + a_{1n} x_n$$
$$g_2 = a_{21} x_1 + a_{22} x_2 + \ldots + a_{2n} x_n$$
$$\vdots$$
$$g_m = a_{m1} x_1 + a_{m2} x_2 + \ldots + a_{mn} x_n$$

$$x_j \geq 0$$

The variables $x_1, x_2, ... x_n$ are defined as the decision variables. The function Z being optimized is called the objective cost function, and the c_n's are the cost sensitivity coefficients. The functions $g_1, g_2, ... g_m$ are the equality constraints, and the coefficients a_{ij} are called the constraint sensitivities. For a linear problem these sensitivities have constant numerical values.

Theorems and Definitions

The following definitions are provided for clarity.

1. A feasible solution to a linear programming problem is a vector *x* which satisfies the m equality constraints with all members of $x \geq$ zero

2. An optimal solution is a feasible solution that minimizes the objective cost function.

3. A basic feasible solution is a feasible solution in which m variables are greater than zero, and n-m variables are equal to zero.

4. A basic feasible optimal solution is a basic feasible solution that optimizes the objective cost function Z.

5. The basis is the set of nonzero decision variables in a basic feasible solution.

6. A convex solution space is a set of solutions in which for any two points in the set, a and b, all points on the line connecting a and b are also in the solution space (see Figure 4-1).

Vertex

Convex   Space

A

B

Local Minimum

Global   Minimum
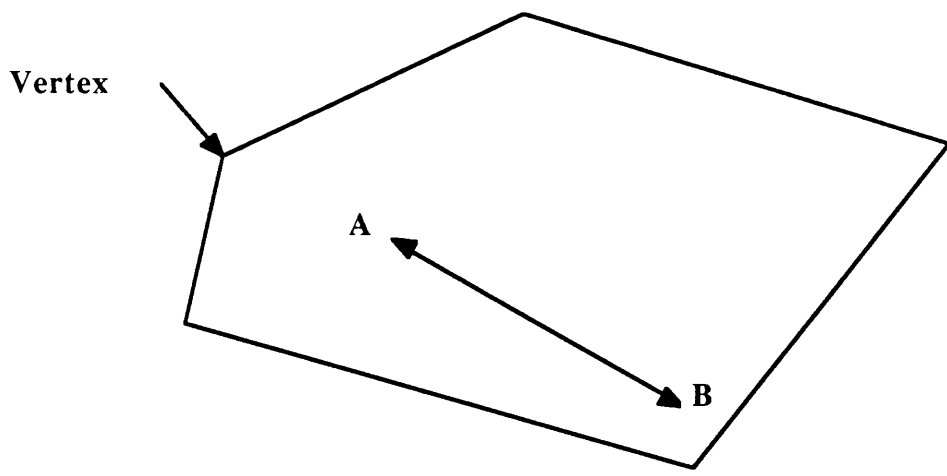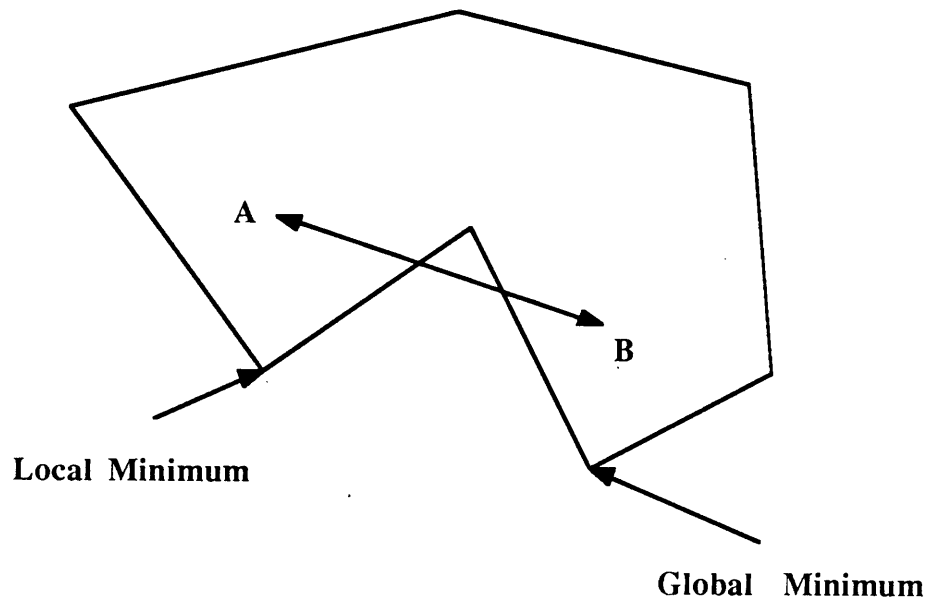
Non-Convex   Space

Figure 4-1. Solution Space Convexity

7.   A vertex of a solution space is an extrema or corner of the space as shown in Figure 4-1.

The following theorems are stated without proof.

Theorem 1. The set of feasible solutions to a linear programming problem is convex.

Theorem 2. The optimal solution to a linear programming problem occurs at a vertex of the convex solution set of feasible solutions.

Theorem 3. If x is a basic feasible solution, it is a vertex of the feasible solution space.

Theorem 4. If x is a vertex of the feasible solution space, it is a basic feasible solution.

The Simplex Algorithm

The Simplex method for solving linear problems is a proven and efficient algorithm. It must begin the solution process with a basic feasible solution, since the simplex routine does not have the capability of converting a non-basic feasible solution to a basic feasible solution.

If no basic feasible solution is immediately available at the start of the linear problem, artificial variables are often used. These artificial variables usually have constraint sensitivity coefficients that form an identity matrix. In such a case, each artificial variable is numerically equal to one of the equality constraints. However, the artificial variables also have arbitrarily assigned cost sensitivities which are chosen to make them very unattractive sol-

ution variables. Consequently, the simplex algorithm will replace the artificial variables with real variables in order to reduce the cost and obtain the optimal solution.

For a nonlinear problem such as the one studied in this thesis, however, artificial variables cannot be used as an initial solution because the cost and margin sensitivities vary with the values of the decision variables. The method used to meet the constraints for the nonlinear problem is discussed below in the section on the Inner Loop.

The simplex routine works by moving from one vertex of the solution space to another. It starts at one vertex and searches in all possible directions for the one that produces the maximum reduction in cost. The program does this by calculating a cost gradient for each of the variables that is not in the basis. The cost gradient represents the total reduction in cost resultant from an incremental increase in the non-basic variable, and it is calculated in the following manner.

Given a vector, G_m, of equality constraints, a matrix, A, of the constraint sensitivities for the variables in the basis, and a vector, x_b, of the values of the basic variables, the following equation represents the constraints:

$$G_m = A \; x_b$$

For a given increment, or stepsize, in a non-basic variable, dx_i, it is desired that the change in the equality constraints, dG_m, be zero. Thus, if the column vector, a_i, is the set of constraint sensitivities associated with the non-basic variable, x_i, then

$$dG_m = A \; dx_b + a_i \; dx_i = 0$$
$$A \; dx_b = - a_i \; dx_i$$

Define

$$dxdx = dx_b/dx_i$$

Then

$$dxdx = -A^{-1}a_i.$$

The quantity dxdx is the sensitivity to a perturbation in the non-basic variable x_i that represents the required increments in the basic variables in order to maintain the equality constraints. Next, define the vector

$$dcdx = dc/dx_b$$

and the scalar

$$c_i = dc/dx_i$$

The vector dcdx represents the cost sensitivity to increments in each of the basic variables, and c_i represents the cost sensitivity to an increment in the non-basic variable x_i. The total cost gradient with respect to the peturbed non-basic variable x_i is given by the following relationship.

Cost Gradient = dcdx . dxdx + c_i

The simplex algorithm then performs an exchange in which the variable that produces the maximum cost benefit is "invited" into the basis, and one of the basic variables is driven to zero, or "excluded," from the basis. The stepsize, dx_i, is defined as the amount that the invited decision variable must be incremented so as to drive one of the basic variables exactly to zero. The invited decision variable must be incremented by such a stepsize that none of the basic decision variables is driven to be negative. Consequently, separate stepsizes are calculated corresponding to each of the basic variables, and the minimum stepsize is chosen. The excluded variable is the variable which is driven to zero by the minimum cal-

culated stepsize, dx_i. Once the stepsize has been calculated, the vector of variables in the basis, x_b, is incremented using the following equation.

$$x_b = x_b + dxdx\ dx_i$$

The invited non-basic variable x_i is set equal to the stepsize.

$$x_i = dx_i$$

Exchanges continue and the basis is continually rearranged until no possible exchange produces any cost benefit. When this is the case, the algorithm has reached the optimal basic feasible solution, and the program stops.

The overall process can be summarized as follows:

- STEP 1: Begin with a basic feasible solution.

- STEP 2: Calculate a cost gradient for each non-basic variable.

- STEP 3: If there are no positive cost gradients, the solution is optimal, stop.

- STEP 4: Invite the non-basic variable with the largest positive cost gradient.

- STEP 5: Calculate a stepsize for the invited decision variable such that one of the basic variables is driven to zero, and none of the basic variables is driven negative.

- STEP 6: Increment decision variables by the amount necessary to maintain the constraints.

- STEP 7: Return to Step 2.

THE LINEAR OPTIMIZER SUBROUTINE

The linear optimizer subroutine used in this thesis draws from the work of Francis Hauser (Reference 5). Since the optimizer subroutine applies linear programming methods to a nonlinear problem, it is very different from a classical linear program. The problem that the optimizer must try to solve is based on a first-order Taylor Series expansion around the current solution. The problem can be stated as follows.

Optimize:

$$Z(x_0 + \Delta x) = Z(x_0) + \sum_{i=1}^{n} (c_i) \Delta x_i$$

Subject to

$$G_m(x_0 + \Delta x) = G_i(x_0) + \sum_{i=1}^{n} (a_i) \Delta x_i$$

$$x_{lo,i} \leq x_i \leq x_{up,i}$$

One of the major differences between the optimizer and a conventional Simplex program is that it must be able to start the optimization process without a basic feasible solution. For the nonlinear problem studied in this thesis, none of the decision variables are equal to zero at the start of the iteration. Moreover, not all of the constraints have necessarily been met when the problem begins. Thus, not only is the initial solution not a basic feasible solution,

but it is not even a feasible solution. The linear optimizer must take the initial solution and drive it to a feasible solution that satisfies the constraints. Once the solution is feasible, it must try to minimize the cost and achieve an optimal solution.

Furthermore, since the problem is nonlinear, the constraint and cost sensitivities are not constant. The linearized sensitivities provided by the interface program are accurate only for a small area of the solution space near the solution about which the problem was linearized. As a result, variable exchanges and increments do not exactly produce the expected changes in the control margins. In this thesis, control margins are the resultant gain and phase margins computed by the frequency response subroutine for given values of the control variables, x. Equality or margin constraints are the desired values of these control margins (e.g. 6 db rigid body gain margin or 30 degrees rigid body phase margin).

The linear optimizer begins its execution with the information that is provided by the Interface program. The linear optimizer receives the following data:

- The current values of the control variables, x.

- The current values of the control margins, G_m.

- A vector of margin sensitivities, a_i, for each control variable.

- A vector of cost sensitivities, c_i.

The optimizer next checks to see if the control margins meet the equality constraints. The criterion used is an arbitrary one determined by the user. In this thesis, a gain margin constraint is considered to be met if:

current gain margin + 0.05 db $\geq$ gain margin constraint.

The phase margin constraint is considered met if:

current phase margin + 0.05 degrees $\geq$ phase margin constraint.

Thus, the constraints are in fact one-sided inequality constraints; any margin which exceeds the margin constraints is considered to be met.

The added 0.05 db or 0.05 degrees is called the constraint error range. The addition of a constraint error range plays an extremely important role in determining the ability of the optimizer to allow exchanges. Each time an exchange is made, and the values of x are incremented, a frequency response run is made and the control margin values are evaluated. The control margin errors are checked, and if all of the margins are not met within the allowed constraint error range, the exchange with the non-basic decision variable is not allowed. The current control margin values, G_m, are always evaluated using a full frequency response simulation, not merely by an extrapolation using the linearized sensitivities.

In general, when an exchange is made to improve the load relief performance, the resultant solution has smaller stability margins. If, before the exchange, the stability margins exactly meet the constraints, the extra constraint error range will allow a .05 db or .05 deg decrease in the control margins. In this way an exchange that causes a decrease in a stability margin below the constraint value is not summarily rejected. The next time that the linear optimizer is called up the inner loop (described below) is executed, and the solution is incremented so as to meet the original stability margin constraints.

The Inner Loop

The inner loop is the first part of the linear optimizer that is executed. The loop is entered only one time, and it is limited to four iterations. In the inner loop, the optimizer program takes the difference between the control margin values and the margin constraint values, and forms the difference vector, **DeltaM**. A matrix **A** is formed from the vectors of margin sensitivities, a_i, associated with each of the basic variables. It then calculates the amount. **DeltaX**, that *the basic variables* (and only the basic variables) must be incremented in order to satisfy the constraints, where

$$DeltaX = A^{-1}DeltaM$$

The control variables are then incremented by **DeltaX**, and a frequency response run is made to determine the true control margin values, **G**m. Since the problem is nonlinear, the control margins will not exactly meet the constraints and the inner loop must be repeated. If, after four iterations, all of the control margin constraints have not been met, the constraint error ranges are loosened to be equal to the current error in any control margin that has not been met. For example, if the current difference vector

DeltaM = (.01 db, .23 deg, .12 db, .03 deg, .04 db),

then the second and third components exceed the allowed error range of .05. In this case, the loosened constraint error ranges would be

Ranges = (.05 db, .23 deg, .12 db, .05 deg, .05 db).

Since the first, fourth, and fifth margins meet the original constraint error range criteria, the values of these error ranges are not changed from the nominal value ⸳ .05. This loosening of the margin constraints "fools" the linear optimizer into believing that all of the constraints

have been met for that particular Minor Loop cycle. In this way, none of the control margin errors will be allowed to increase when the control variables are incremented.

When the linear optimizer is called up again for the next Minor Loop cycle, the constraint error ranges are reset to their original values, the inner loop is executed again, and the constraint error ranges are recalculated. In this way, the constraint error ranges tighten each time the optimizer is called up, until finally all of the constraints are met. After the optimizer has tried to drive the solution to meet its constraints and the constraint error ranges have been calculated, the optimizer exits the inner loop.

Decision Variable Limits

The decision variable limits control the amount by which the decision variables are allowed to vary using the current linearized margin and cost sensitivities. The decision variables all have absolute upper and lower limits ($absx_{up}$ and $absx_{lo}$) which are constant. Since the linearized model of the problem is valid only for a region near the current solution, the Minor Loop Percentage Limit (MLPL), described in Chapter 3, controls the upper and lower decision variable limits (x_{up} and x_{lo}). The limits are evaluated as follows for a particular decision variable, x.

$$x_{up} = Minimum(absx_{up}, \ x \ (1 + MLPL))$$

$$x_{lo} = Maximum(absx_{lo}, \ \frac{x}{(1 + MLPL)})$$

Thus, unless an absolute variable limit is within the minor loop percentage limit of the current value of the decision variable x, the MLPL will determine the decision variable upper and lower bounds.

56

Once decision variable limits are established, the linear optimizer follows a routine similar to the Simplex algorithm. The main difference is that each time a stepsize is calculated for the invited non-basic variable, it is determined by the amount that will drive a basic variable to either its upper or lower limit, instead of to zero. Moreover, non-basic variables are generally nonzero and are equal to either their upper or lower limit.

The overall operation of the linear optimizer may be summarized as follows:

- STEP 1: Begin with an arbitrarily chosen set of basic and non-basic variables.

- STEP 2: Calculate cost gradients for each of the non-basic variables.

- STEP 3: If there are no positive cost gradients, choose a different set of basic and non-basic variables, return to Step 2.

- STEP 4: Invite the non-basic variable with the largest positive cost gradient.

- STEP 5: Calculate a stepsize for the invited decision variable such that only one of the basic variables is driven to either its upper or lower limit.

- STEP 6: The basic variable driven to its limit becomes a non-basic variable, and the invited decision variable becomes a basic variable.

- STEP 7: Increment decision variables by the amount necessary to maintain the constraints (based on the linearized margin sensitivities).

- STEP 8: Perform a frequency response run to calculate the true control margins, G_m

- STEP 9: If the margins are not still met, return the decision variables to their old values.

- STEP 10: Return to Step 2.

LIMITS OF THE LINEAR APPROACH

There are several limits to the linear approach used in this thesis to optimize the non linear problem studied here. The five most important limitations have to do with

- Solution Space Convexity

- Choice of Cost Function

- Basis Initialization

- Constraint Convergence

- Computational Costs

Solution Space Convexity

As discussed in the section on linear programming theory, linear programming can guarantee an optimal solution only if the solution space is convex. The Simplex method of linear programming uses what is basically a version of a simple gradient search. Therefore, if the program finds itself in a local minimum, it has no way to work its way out in order to

find the global minimum. The program developed in this thesis compensates for this fact by allowing the cost to actually increase for one major loop.

For example, if the cost on the sixth major loop cycle is greater than the cost of the fifth cycle, a seventh major loop cycle will still be executed. The program will compare the cost of the fifth major loop cycle with that of the seventh. If the cost of the seventh cycle is less than that of the fifth, the program will go on, having in essence escaped a local minimum (an example of this is given in Appendix B). However, if the cost increases for two major loops, the program will return to the minimum cost solution and terminate. This logic does not guarantee that every local minimum will be avoided, but it does at least provide a chance to do so.

The reason that the cost may increase from one major cycle to the next is due to the fact that the program attempts to iterate toward the margin constraints each time the linear optimizer subroutine is called up. It is often the case that the actual gain margin is less than the margin constraint but still lies within an acceptable constraint error range. For instance, if the gain margin constraint is 6.05 db with a constraint error range of .05 db, the actual gain margin may be 6.02 db. When the program tries to increase that gain margin to 6.05 db, the resultant solution will most likely produce a higher cost (i.e. worse load relief performance). The program will next try to perform exchanges that improve the load relief. If however, there are no positive cost gradients to allow exchanges, none will be made. The resultant solution would then be much closer to the specified constraints, but would give a solution with a higher cost than the previous cycle.

Basis Initialization

Another weakness of applying linear programming to this non-linear problem is that none of the variables are actually zero as they would be in a classic linear problem. Since this is the case, there is no rational way to determine what is initially a basic variable and what is a non-basic variable. The choice of basis initialization is not determined by any rule, but by the engineering judgement of the user.

However, the choice of basis initialization has a significant impact on the optimization procedure. If the initial basis is chosen so that none of the non-basic variables produces a positive cost gradient, the program may be unable to produce exchanges that improve the cost, and the program could terminate prematurely. Consequently, engineering judgement must be used to select variables that can improve the load relief performance (e.g. KD_{LR} or KV_{LR}) to be initially outside of the basis.

Choice of Cost Function

As with any optimization scheme, the choice of cost function will dramatically affect what type of solution is obtained. The cost function used in this thesis represents the maximum angle of attack for a typical wind gust that may occur at the maximum dynamic pressure point in the trajectory. The solutions obtained from this cost function might not be optimal for other types of wind inputs (e.g. wind angle of attack step functions or impulses).

It is conceivable that several different wind profiles could be used for the transient response subroutine, and a weighted sum of the performance criteria from each profile be used to determine the total cost. Limits on computational costs and time, however, prevented this from being implemented.

Constraint Convergence

A fourth limitation of the linear approach lies in the iteration in the inner loop used to meet the stability margin constraints. Whenever some of the stability margins are significantly different from the margin constraints (approximately > 10%), the inner loop of the linear optimizer attempts to increment the control variables *in the basis* by a significant amount. This creates two problems. First, since the problem is not linear, the margin sensitivities that the increments are based on are neither constant nor valid for any size of variable increment. As a result, attempts to alter stability margins by large amounts can lead to divergent iterations and system instability. Consequently, a correction factor had to be applied to the increments calculated in the inner loop that prevent the inner loop from incrementing the basic variables by more than 5 %. Unfortunately, this slows down the convergence of the constraint iterations.

Secondly, the inner loop does not increment any of the non-basic variables. This is the case because in order to calculate the inverse of the margin sensitivity matrix for the decision variable increments in the inner loop, this matrix must be square. As a result, the number of variables affected by the inner loop is limited to be equal to the number of constraints in the problem (i.e. the number of basic variables). Little success was achieved when attempts were made to utilize a psuedo-inverse of the non-square matrix of margin sensitivites for all of the decision variables. However, this avenue of calculating inner loop variable increments is a possible area of future investigation.

Computational Costs

A major problem with the linear approach is that of computation time and costs. Since variables were allowed to be incremented by only a few percent of their current values, pro-

gress was slow. Furthermore, in order to maintain the stability margin constraints, frequency response runs were made each time the control variables were changed by either the inner loop or from a normal optimization exchange. The computational costs also increase with the number of variables allowed in the optimization. When the fourth order bending filter optimization cases were run, four of the seven variables in the bending filter were held constant in order to avoid intolerable computation costs.

CHAPTER 5

OPTIMIZATION RESULTS

The results of six control loop optimizations are presented in this chapter. Three of the optimizations employ conventional load relief, and the other three use alternative load relief methods. In each case the optimization run was initialized with a set of control parameters that met, or came close to meeting, the stability margin constraints. The values of control parameters, stability margins, and maximum aerodynamic angle of attack, Max a, are presented in this chapter for the initial and final conditions of each of the six control optimization runs. In all of these runs, it will be seen that the optimization process drives the margins to meet their constraints, and substantially reduces the value of Max a as well.

FIRST ORDER OPTIMIZATION

In Case 1, the control loop employs conventional load relief, using ΔV_N feedback, and in Case 2, the control loop uses the alternative load relief with ΔV_N and δ feedback. Both of these cases were run using a first order forward loop bending filter. The first order bending filter used for these two optimizations, D(w), is defined as

$$D(w) = \frac{(w + a)}{(w + b)}$$

For the frequency response subroutine, the w-domain representations of the bending filter, rate estimator, and load relief filters, are used because it is more convenient for computation of gain and phase. In the transient response subroutine, however, the outputs of the filters are computed from difference equations obtained from the z-transform representations of the filters, obtained from the w-transforms by the substitution:

$$w = \frac{1 - z^{-1}}{1 + z^{-1}}$$

The optimization for Cases 1 and 2 was performed for a nominal trajectory with a maximum dynamic pressure (Q) of 4670 lbs/ft². Due to the limitations of the first order bending filter, no constraint was applied to the second bending mode gain margin. Consequently, there were only four stability margin constraints, which are given in Table 5-1.

Table 5-1. Margin Constraints for Cases 1 and 2

GML (db)	PMR (deg)	GMH (db)	PMB (deg)
6.05	30.05	- 6.05	65.05
Constraint Error Ranges			
.05	.05	.05	.05

GML = Low Frequency Rigid Body Gain Margin

PMR = Rigid Body Phase Margin

GMH = High Frequency Rigid Body Gain Margin

PMB = First Bending Mode Phase Margin

Conventional Load Relief Case (Case 1)

The initial conditions for the conventional load relief optimization are given in Table 5-2, and the final solution for this optimization are given in Table 5-3. In each of these tables, the first two rows show the transient and frequency responses that are determined by the control parameters in the third and fourth rows. The control parameter symbols are defined as follows:

K_F = Forward Loop Gain

Pole = b in definition of D(w)

Zero = a in definition of D(w)

τ_{est} = Rate Estimator Time Constant

KFB = Rate Feedback Gain

KV_{LR} = ΔV_N Load Relief Gain

KD_{LR} = δ Load Relief Gain

τ_{LR} = Load Relief Time Constant

Table 5-2. Case 1 Initial Conditions

Max σ (deg)	GML (db)	PMR (deg)	GMH (db)	PMB (deg)
.76060	6.003	33.76	- 6.106	65.34

K_F	Pole	Zero	T_{est} (sec)	KFB	KV_{LR}	KD_{LR}	T_{LR} (sec)
2.0634	.02508	.03408	.2415	.15102	.1931	0.000	1.0392

Table 5-3. Case 1 Final Conditions

Max σ (deg)	GML (db)	PMR (deg)	GMH (db)	PMB (deg)
.38499	6.003	30.65	- 6.024	66.84

K_F	Pole	Zero	T_{est} (sec)	KFB	KV_{LR}	KD_{LR}	T_{LR} (sec)
2.2883	.02302	.04882	.1774	.21658	1.584	0.000	5.9590

Comparing Tables 5-2 and 5-3, it can be seen that (1) the stability margin constraints are satisfied at both the beginning and the end of the optimization run, and (2) the optization process substantially reduces Max a from 0.76060 degrees to 0.38499 degrees, amounting to a 49 percent reduction. A plot of the transient response using the final solution for conventional load relief is shown in Figure 5-1.

Alternative Load Relief Case (Case 2)

An optimization using the same vehicle and environmental conditions, as well as the same stability margin constraints was performed using the alternative load relief. The initial and final conditions of this run are given in Tables 5-4 and 5-5 respectively.

Table 5-4. Case 2 Initial Conditions

Max a (deg)		GML (db)		PMR (deg)		GMH (db)		PMB (deg)
.67065		6.001		27.88		- 6.551		67.94

K_F	Pole	Zero	T_{est} (sec)	KFB	KV_{LR}	KD_{LR}	T_{LR} (sec)
2.3514	.02540	.04092	.2450	.15000	.1920	- .1800	0.9950

TRANSIENT RESPONSE

LOAD RELIEF PERFORMANCE
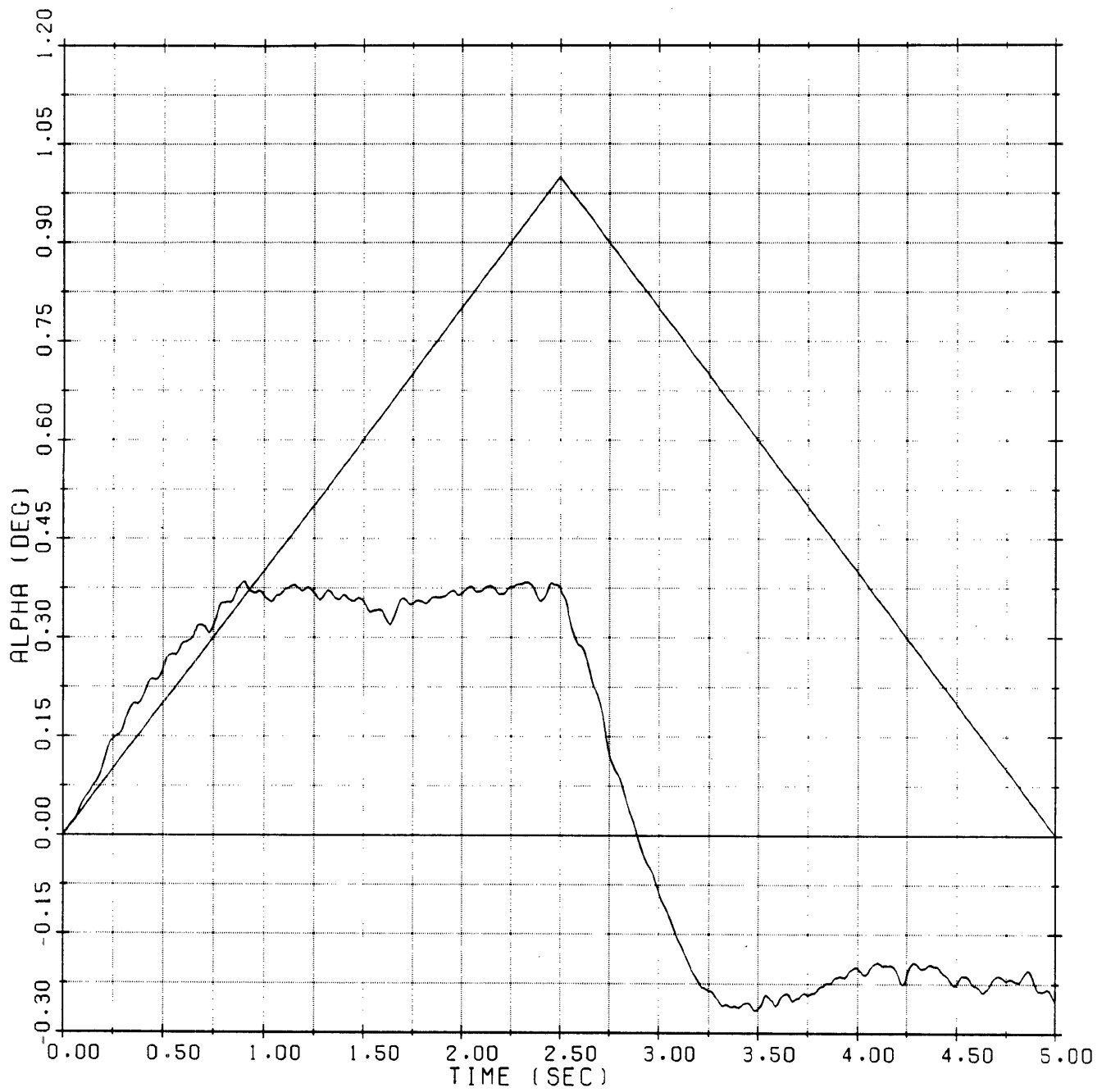
Figure 5-1. Conventional Load Relief with First Order Filter

Table 5-5. Case 2 Final Conditions

Max a (deg)	GML (db)	PMR (deg)	GMH (db)	PMB (deg)
.33537	6.140	30.00	- 6.081	68.69

K_F	Pole	Zero	T_{est} (sec)	KFB	KV_{LR}	KD_{LR}	T_{LR} (sec)
2.1878	.09611	.18837	.7126	.25700	- 0.7015	-11.5804	4.9865

Comparing Tables 5-4 and 5-5, it can be seen that (1) the rigid body phase margin, PMR, was increased in order to meet its constraint, and (2) the value of Max a is reduced from 0.67065 to 0.33537 degrees, amounting to appoximately a 50 percent reduction. The alternative load relief case produced a final Max a that was .05 degrees less (approximately 13%) than the conventional load relief case. A plot of the transient response using the final solution for the alternative load relief is shown in Figure 5-2.

A perspective on the significance of the gain factors multiplying the δ and ΔV_N signals in the alternative load relief can be obtained by noting that these two signals each provide a measure of the steady state or low frequency angle of attack corresponding to their values. Therefore, the combination of these two signals may be described as an overall load relief gain, K_{LR} , times a linear combination of the two steady state angle of attack estimates, $a_{ss}(\delta)$ and $a_{ss}(\Delta V_N)$:

$$a_{ss} = R \ a_{ss}(\Delta V_N) + (1 - R) \ a_{ss}(\delta)$$

TRANSIENT RESPONSE

LOAD RELIEF PERFORMANCE
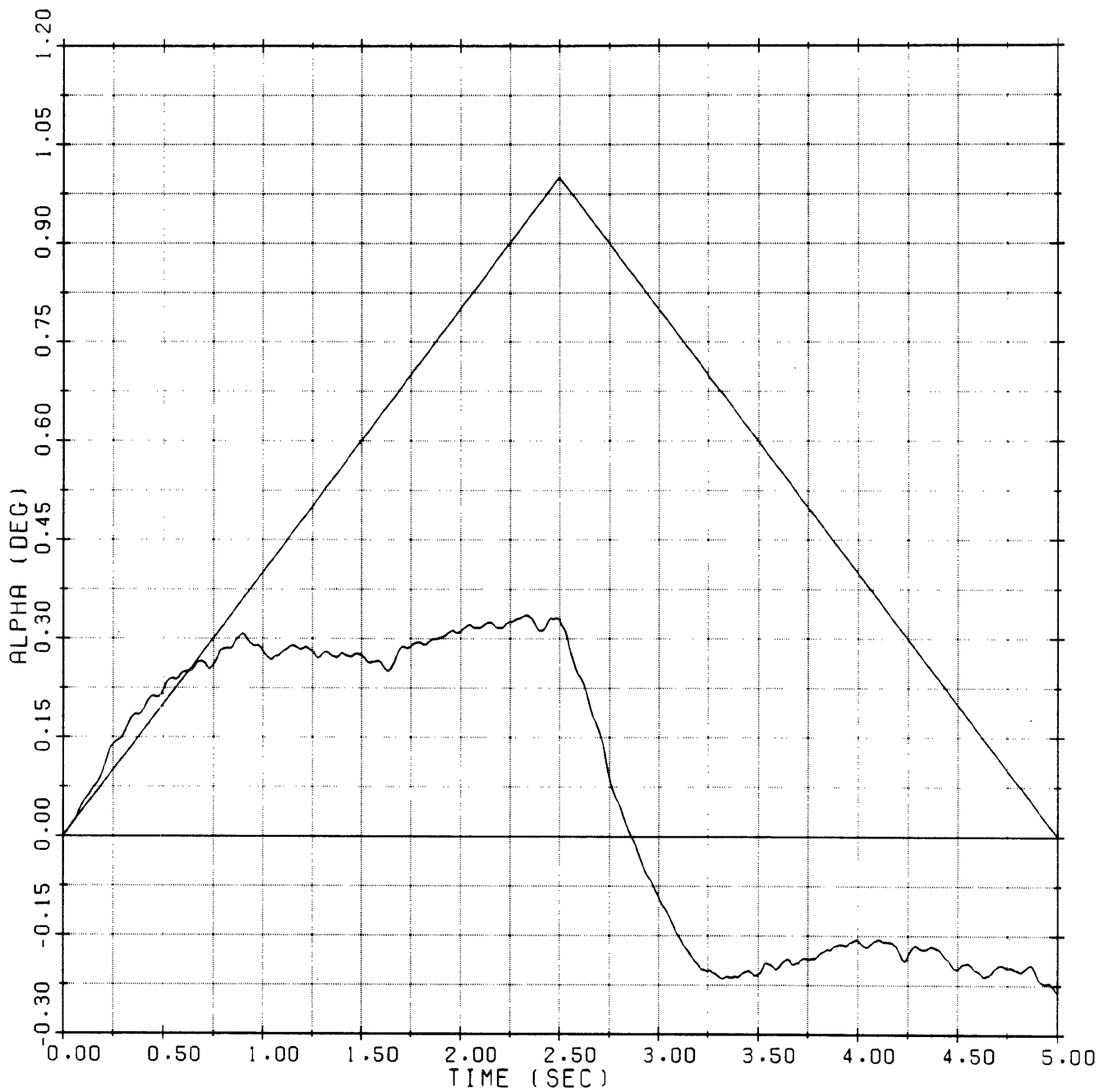
Figure 5-2. Alternative Load Relief with First Order Filter

Where R is a weighting factor. A diagram of this low frequency angle of attack feedback is shown in Figure 5-3. The load relief feedback, however, was actually implemented as is shown in Figure 5-4. In terms of the relative weighting that these two signals have, it is necessary to convert the gains KV_{LR} and KD_{LR} using the two equations for a_{ss}

$$a_{ss}(\delta) = \frac{-F\, I_{cg}}{S\, Q\, C_{N\alpha}\, I_{cp}} \quad \delta$$

$$a_{ss}(\Delta V_N) = \frac{M\, I_{cg}}{S\, Q\, C_{N\alpha}\, (I_{cp} + I_{cg})\, T} \quad \Delta V_N$$

The relationship for determining the weighting factor, R, from the two gains KV_{LR} and KD_{LR} is given below (see Appendix C for derivation).

$$R = \frac{F\, (I_{cp} + I_{cg})T\ KV_{LR}}{(F\, (I_{cp} + I_{cg})\, T\ KV_{LR} - M\, I_{cp}\, KD_{LR})}$$

For this alternative load relief case, the KV_{LR} gain was driven negative, producing a negative weighting factor (R = - 0.427). Consequently, the angle of attack estimate would be weighted as

$$a_{ss} = 1.427\ a_{ss}(\delta) - 0.427\ a_{ss}(\Delta V_N)$$

$$\frac{m\, l_{cg}}{S\, Q\, C_{N\alpha}\, (l_{cp} + l_{cg})}$$

$$\frac{-F\, l_{cg}}{S\, Q\, C_{N\alpha}\, l_{cp}}$$

R

1 - R

α_{ss}

K_{LR}

$$\frac{1}{\tau s + 1}$$

θ_{LR}

$\frac{\Delta V}{T}$

δ
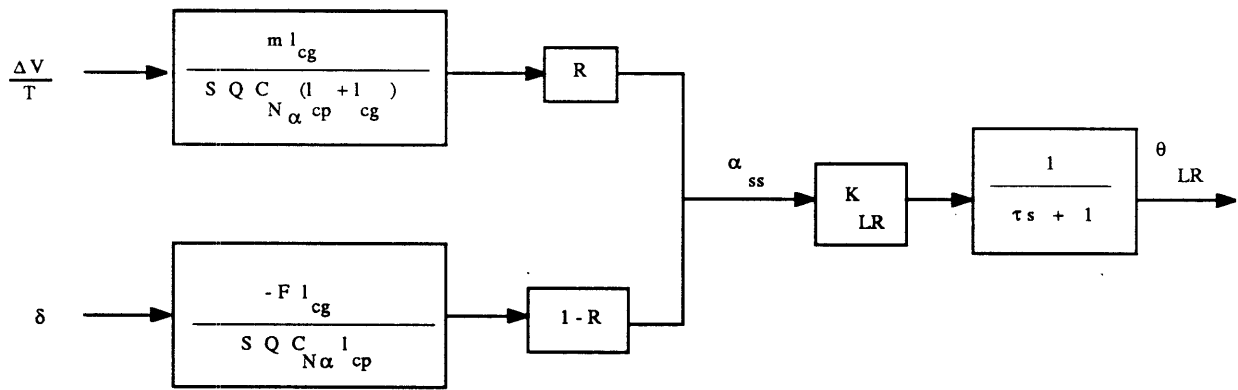
Figure 5-3. Angle of Attack Weighting

$KV_{LR}\, T$

KD_{LR}

$$\frac{1}{\tau s + 1}$$

θ_{LR}

$\frac{\Delta V}{T}$

δ
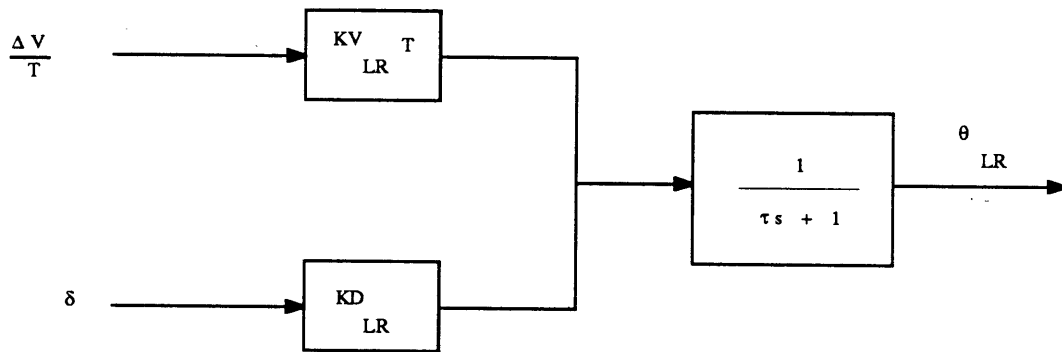
Figure 5-4. Load Relief Implementation

FOURTH ORDER OPTIMIZATION

The four cases presented in this section were run using a fourth order forward loop bending filter. The fourth order bending filter, D(w), like the first order filter used in Cases 1 and 2, is defined in the discrete w-domain.

$$D(w) = \frac{(w + a)\,(w^2 + 2\zeta w_n + w_n^2)}{(w + b)\,(w + c)\,(w^2 + 2\zeta' w'_n + w'^2_n)}$$

The two second order compensation terms constitute a notch filter at the second bending mode natural frequency of 122 rad/sec. The equivalent w-domain frequency is approximately 0.7. In order to reduce computational costs and improve solution time, these two terms were held constant for the four optimization runs. As a result, there were only three variables in the forward loop compensation: one real zero, $-a$, and two real poles, $-b$ and $-c$. The values for the notch filter parameters are given below in Table 5-6.

Table 5-6. Notch Filter Parameters

ζ	w_n	ζ'	w'_n
.11168	.70088	.22042	.70530

In cases 3 and 4, the control loop employs both conventional load relief, (using ΔV_N feedback), and the alternative load rr'ief with (ΔV_N and δ feedback) for a nominal trajectory with a maximum dynamic pressure of Q = 4670 lbs/ft² . In cases 5 and 6, both the conven-

tional and the alternative load relief are employed for a "worst case" trajectory with a maximum dynamic pressure of Q = 6000 lbs/ft² .

Nominal Trajectory Conditions

For the nominal trajectory cases the following constraints and constraint error ranges were used:

Table 5-7. Margin Constraints for Cases 3 and 4

GML (db)	PMR (deg)	GMH (db)	PMB (deg)	GMB (db)
6.10	30.10	- 6.10	80.00	- 11.80
Constraint Error Ranges				
0.10 (db)	0.10 (deg)	0.10 (db)	0.30 (deg)	0.30 (db)

GML = Low Frequency Rigid Body Gain Margin

PMR = Rigid Body Phase Margin

GMH = High Frequency Rigid Body Gain Margin

PMB = First Bending Mode Phase Margin

GMB = Second Bending Mode Gain Margin

These constraints reflect the fact that the addition of the notch filter and the extra pole to the first order bending filter makes it possible to meet much larger bending mode stability margins.

Conventional Load Relief Case (Case 3)

The initial conditions for the conventional load relief optimization are given in Table 5-8, and the final solution for this optimization are given in Table 5-9. In each of these tables, the first two rows show the transient and frequency responses that are determined by the control parameters in the third and fourth rows. The control parameter symbols are as defined below.

K_F = Forward Loop Gain

Zero = a in definition of D(w)

Pole1 = b in definition of D(w)

Pole2 = c in definition of D(w)

τ_{est} = Rate Estimator Time Constant

KFB = Rate Feedback Gain

KV_{LR} = ΔV_N Load Relief Gain

KD_{LR} = δ Load Relief Gain

τ_{LR} = Load Relief Time Constant

Table 5-8. Case 3 Initial Conditions

Max a (deg)	GML (db)	PMR (deg)	GMH (db)	PMB (deg)	GMB (db)
.59915	6.027	31.92	- 6.033	80.04	-11.482

K_F	Pole1	Pole2	Zero	T_{est} (sec)	KFB	KV_{LR}	KD_{LR}	T_{LR} (sec)
1.9824	.02231	1.3784	.05009	.3335	.27960	.2949	0.000	1.0507

Table 5-9. Case 3 Final Conditions

Max a (deg)	GML (db)	PMR (deg)	GMH (db)	PMB (deg)	GMB (db)
.44185	6.009	30.16	- 6.021	79.72	-11.649

K_F	Pole1	Pole2	Zero	T_{est} (sec)	KFB	KV_{LR}	KD_{LR}	T_{LR} (sec)
2.0973	.01360	1.2603	.05009	.3282	.45792	.8086	0.000	2.7039

Comparing Tables 5-8 and 5-9, it can be seen that the stability margin constraints are satisfied within the constraint error ranges at both the start and the end of the optimization run, and that the value of Max a was reduced from .59915 to .44185 degrees, a 26 percent improvement. A plot of the transient response using the final solution for conventional load relief is shown in Figure 5-5.

Alternative Load Relief Case (Case 4)

An optimization using the same vehicle and environmental conditions, as well as the same stability margin constraints as in Case 3 was performed using the alternative load relief. The initial and final conditions of this run are given in Tables 5-10 and 5-11 respectively.

Table 5-10. Case 4 Initial Conditions

Max a (deg)	GML (db)	PMR (deg)	GMH (db)	PMB (deg)	GMB (db)
.56836	6.123	29.79	- 6.163	81.36	-11.501

K_F	Pole1	Pole2	Zero.	T_{est} (sec)	KFB	KV_{LR}	KD_{LR}	T_{LR} (sec)
1.9944	.02353	1.4620	.05607	.2868	.29937	.2516	0.3712	0.9950

TRANSIENT RESPONSE

LOAD RELIEF PERFORMANCE
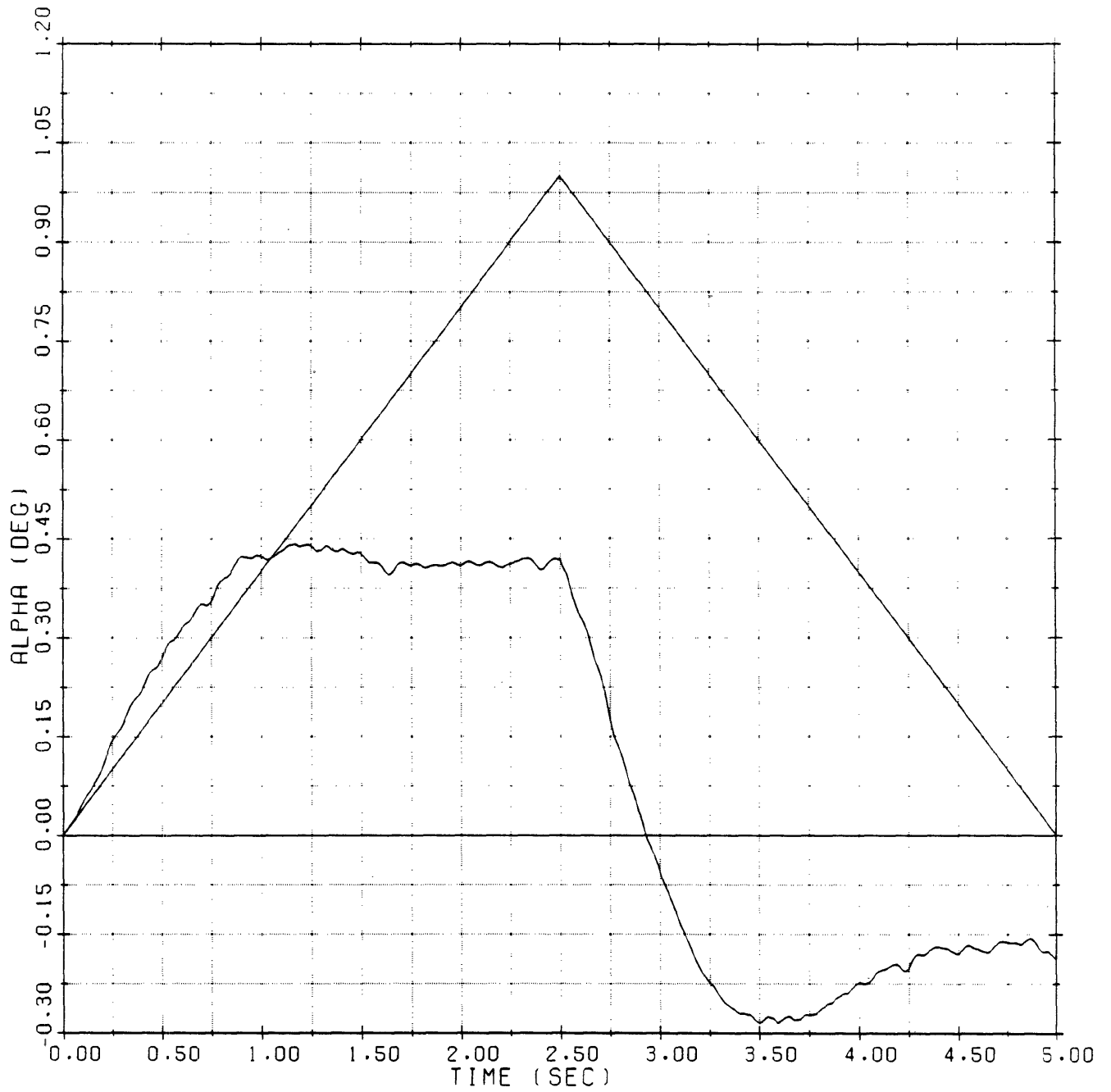
Figure 5-5. Conventional LR with 4th Order Filter and Nominal Q

Table 5-11. Case 4 Final Conditions

Max a (deg)	GML (db)	PMR (deg)	GMH (db)	PMB (deg)	GMB (db)
.41496	6.103	30.10	- 6.100	80.00	-11.800

K_F	Pole1	Pole2	Zero	T_{est} (sec)	KFB	KV_{LR}	KD_{LR}	T_{LR} (sec)
2.3171	.01386	1.1263	.04662	.4498	.38332	.7169	0.8840	3.1324

Comparing Tables 5-10 and 5-11, it can be seen that as in previous runs, the value of Max a was decreased. In this case it was reduced from .56836 to .41496 degrees, a 27 percent improvement.

A plot of the transient response using the final solution for the alternative load relief is shown in Figure 5-6. For this alternative load relief case, the values of KV_{LR} and KD_{LR} produced an angle of attack weighting factor of R = 0.8001. Consequently, the angle of attack estimate would be weighted as

$$a_{ss} = 0.200 \ a_{ss}(\delta) + 0.800 \ a_{ss}(\Delta V_N)$$

The alternative load relief case produced a final Max a of 0.415 degrees, which was .027 degrees less (or 6 %) than the 0.442 degrees from the conventional load relief case. The solution favored the ΔV_N load relief signal in the angle of attack estimate, with a weighting

TRANSIENT RESPONSE

LOAD RELIEF PERFORMANCE
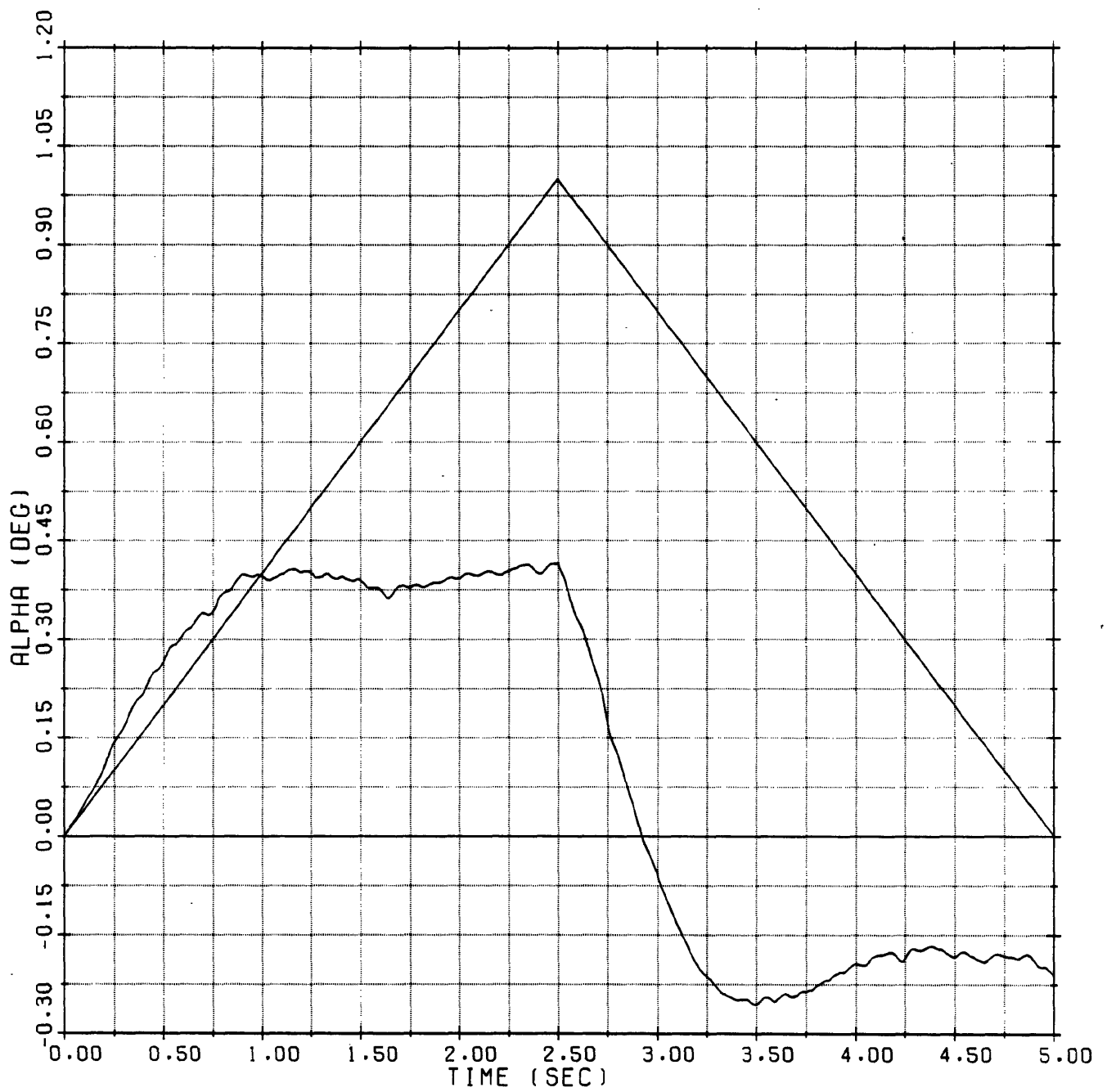
Figure 5-6. Alternative LR with 4th Order Filter and Nominal Q

factor of 80 %. Nevertheless, the combination of signals outperformed the conventional load relief using the ΔV_N signal alone.

Worst Case Trajectory Conditions

For the worst case trajectories the following constraints and constraint error ranges were used:

Table 5-12. Margin Constraints for Cases 5 and 6

GML (db)	PMR (deg)	GMH (db)	PMB (deg)	GMB (db)
5.75	27.05	- 5.75	80.05	- 11.75
Constraint Error Ranges				
0.10 (db)	0.10 (deg)	0.10 (db)	0.10 (deg)	0.25 (db)

The rigid body gain and phase margin constraints were decreased for these last two cases. This was done because it was not possible to meet the original rigid body stability margin constraints using a trajectory with a maximum Q of 6000 lbs/ft². As can be seen below, the lowering of the rigid body constraints allowed a better load relief performance to be achieved in the worst case trajectory runs.

Conventional Load Relief Case (Case 5)

The initial conditions for the conventional load relief optimization are given in Table 5-13, and the final solution for this optimization are given in Table 5-14.

Table 5-13. Case 5 Initial Conditions

Max α (deg)		GML (db)		PMR (deg)		GMH (db)		PMB (deg)		GMB (db)
.75795		5.867		27.93		- 5.964		88.40		-12.101
K_F	Pole1	Pole2	Zero	T_{est} (sec)	KFB	KV_{LR}	KD_{LR}	T_{LR} (sec)
2.4184	.02061	1.1332	.03093	.3450	.15000	.1693	0.000	0.9950

Table 5-14. Case 5 Final Conditions

Max a (deg)	GML (db)	PMR (deg)	GMH (db)	PMB (deg)	GMB (db)
.38116	5.842	27.75	- 5.656	82.90	-11.769

K_F	Pole1	Pole2	Zero	τ_{est} (sec)	KFB	KV_{LR}	KD_{LR}	τ_{LR} (sec)
2.2808	.03684	1.0876	.06709	1.6553	.19141	2.2408	0.000	9.0211

Comparing Tables 5-13 and 5-14, it can be seen that the stability margin constraints are satisfied at both the start and the end of the optimization run, and that the value of Max a was reduced from .75795 to .38116 degrees, a 50 percent improvement. A plot of the transient response using the final solution for traditional load relief is shown in Figure 5-7.

Alternative Load Relief Case (Case 6)

An optimization using the same vehicle and environmental conditions, as well as the same stability margin constraints as in Case 5 was performed using the alternative load relief. The initial and final conditions of this run are given in Tables 5-15 and 5-16 respectively.

TRANSIENT RESPONSE

LOAD RELIEF PERFORMANCE
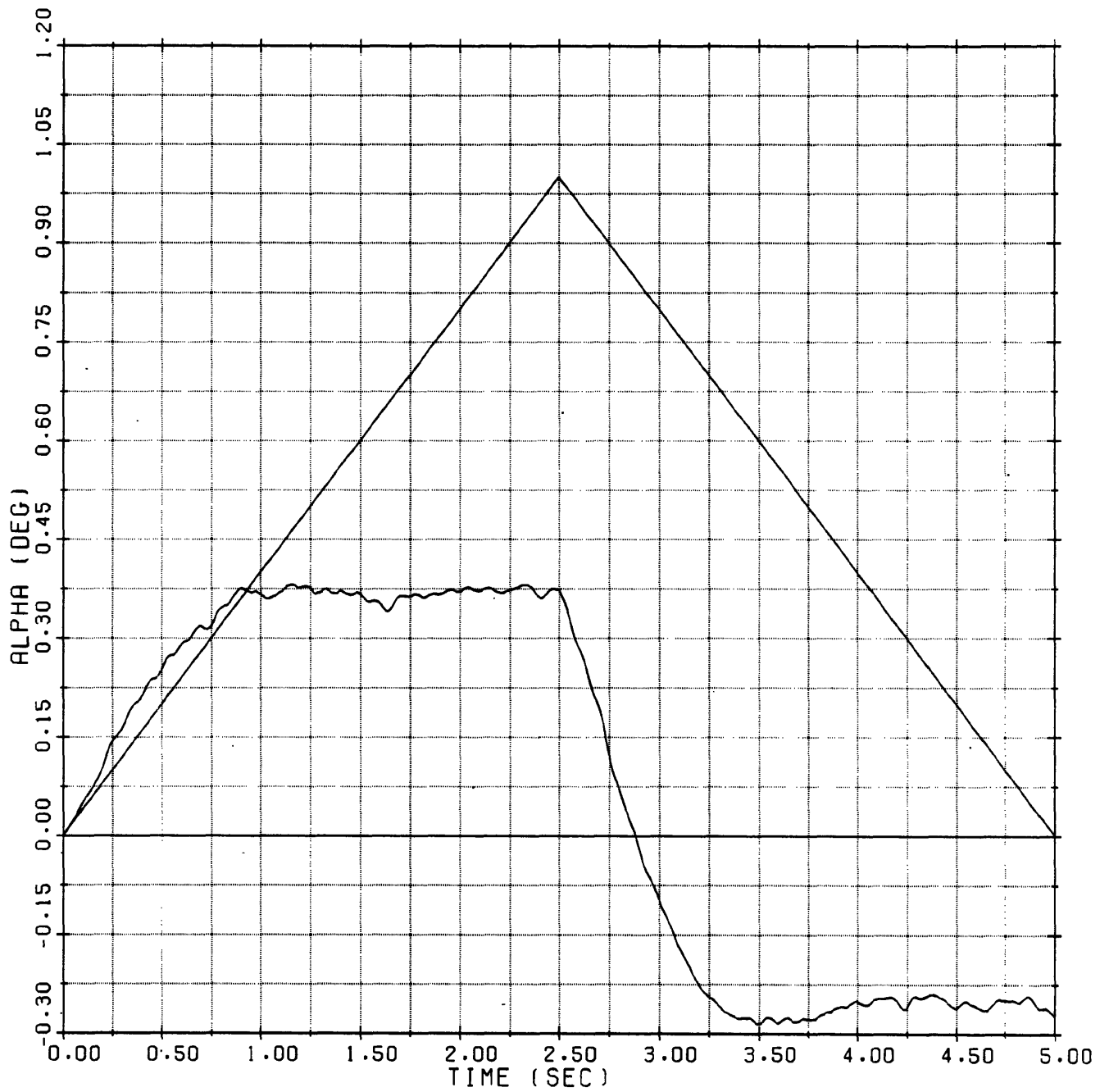
Figure 5-7. Conventional LR with 4th Order Filter and Worst Q

Table 5-15. Case 6 Initial Conditions

Max σ (deg)	GML (db)	PMR (deg)	GMH (db)	PMB (deg)	GMB (db)			
.71663	5.765	27.01	- 6.101	88.67	-12.226			
K_F	Pole1	Pole2	Zero	T_{est} (sec)	KFB	KV_{LR}	KD_{LR}	T_{LR} (sec)
2.4772	.02041	1.1223	.03129	.4062	.15000	.2002	0.0600	1.3191

Table 5-16. Case 6 Final Conditions

Max σ (deg)	GML (db)	PMR (deg)	GMH (db)	PMB (deg)	GMB (db)			
.34738	5.722	27.02	- 5.761	80.14	-11.756			
K_F	Pole1	Pole2	Zero	T_{est} (sec)	KFB	KV_{LR}	KD_{LR}	T_{LR} (sec)
2.8293	.01427	0.9501	.04892	.9626	.35315	1.3411	9.1884	11.113

Comparing Tables 5-15 and 5-16, it can be seen once again that the stability margin constraints are satisfied at both the start and the end of the optimization run, and that the value of Max a was reduced from .71663 to .34738 degrees, a 52 percent load relief improvement.

A plot of the transient response using the final solution for the alternative load relief is shown in Figure 5-8. For this alternative load relief case, the values of KV_{LR} and KD_{LR} produced an angle of attack weighting factor of R = 0.4188. Consequently, the angle of attack estimate would be weighted as

$$a_{ss} = 0.5812 \; a_{ss}(\delta) + 0.4188 \; a_{ss}(\Delta V_N)$$

The alternative load relief case produced a final Max a of 0.347 degrees, which was .034 degrees less (or 9 %) than the 0.381 degrees from the conventional load relief case. In this instance, the solution favored the δ load relief signal in the angle of attack estimate, with a weighting factor of 58 %. Once again, the combination of signals outperformed the conventional load relief using the ΔV_N signal alone. This fact is to be expected, since the addition of the δ signal provides more flexibility for the optimization.

TRANSIENT RESPONSE

LOAD RELIEF PERFORMANCE
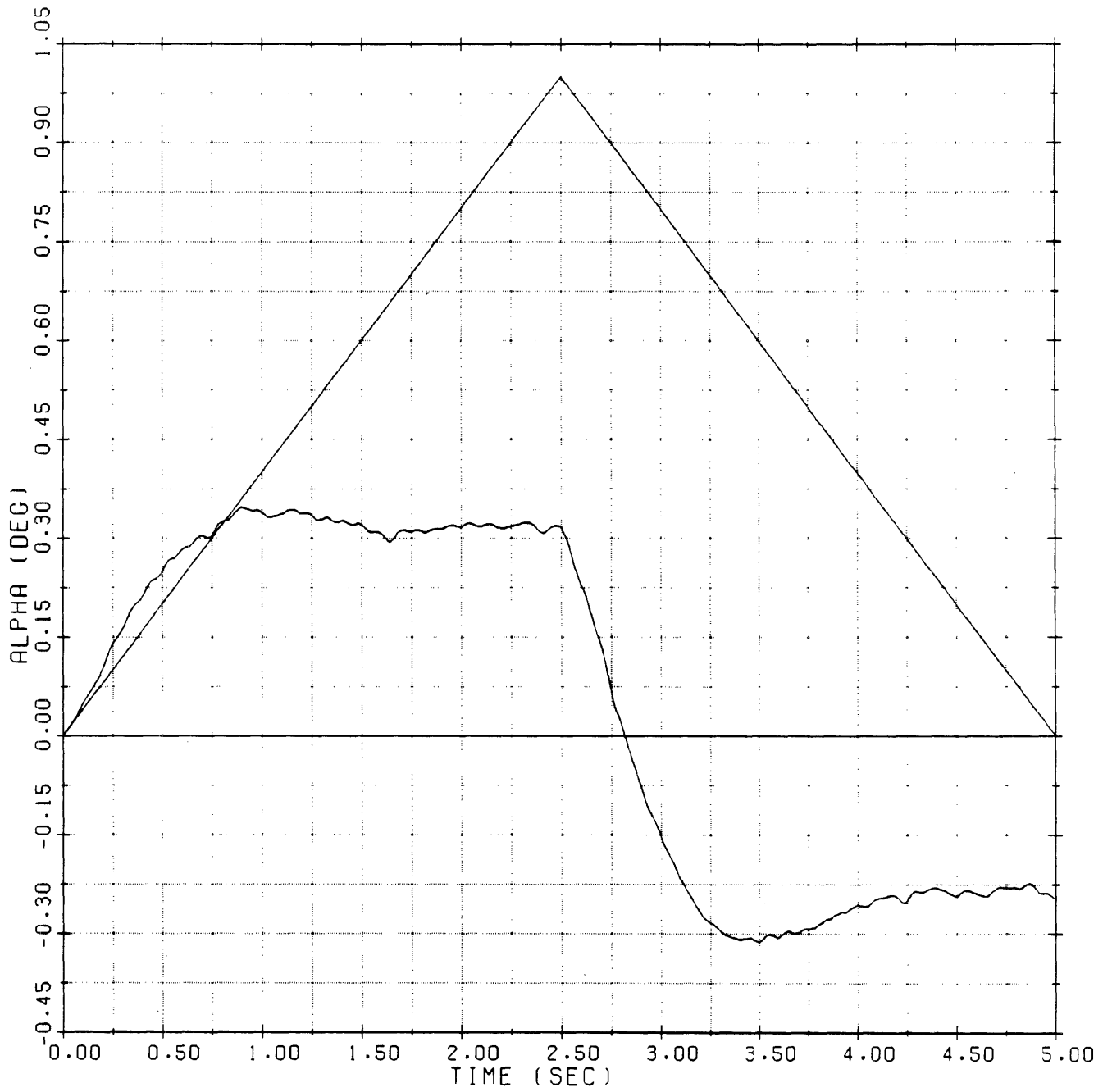
Figure 5-8. Alternative LR with 4th Order Filter and Worst Q

DESIGN ROBUSTNESS

Each of the optimization cases studied in this thesis was designed for specific vehicle and environmental conditions, with presumed knowledge of aerodynamic and mass properties of the vehicle and its environment. Due to the possibility of significant uncertainties in vehicle parameters, the sensitivity of these designs to parameter uncertainties is important to overall design robustness. Frequency and transient response runs were made for each of the six cases presented in this thesis for four off nominal situations:

High Vehicle Thrust

Low Vehicle Thrust

High Aerodynamic Forces

Low Aerodynamic Forces.

The six optimization cases may be described as follows:

Case 1: First Order Optimization with Conventional Load Relief

Case 2: First Order Optimization with Alternative Load Relief

Case 3: Fourth Order Optimization with Conventional Load Relief

Case 4: Fourth Order Optimization with Alternative Load Relief.

Case 5: Fourth Order Optimization with Worst Case Trajectory Conditions and Conventional Load Relief

Case 6: Fourth Order Optimization with Worst Case Trajectory Conditions and Alternative Load Relief

Tables 5-17 to 5-22 describe the effects of the four off nominal conditions on the stability margins and maximum angle of attack excursion for the six optimization cases. The results

of the off nominal cases were mixed in general. Neither the conventional nor the alternative load relief control systems proved clearly more robust than the other. Nevertheless, none of the six cases showed great sensitivity to the off nominal conditions. The rigid body phase margin was never compromised by more than 2 degrees, and the rigid body gain margin was not reduced by more than 0.85 db. Each set of off nominal conditions is considered below in terms of both the variations in vehicle and trajectory parameters assumed, as well as in the effects of the particular off nominal conditions on the stability margins and load relief performance of the six optimized control systems.

High Thrust Conditions

For the increased thrust scenario, the vehicle parameters were varied in the following manner.

Thrust, F (lbs)	+ 5 %
Mass, M (slugs)	- 2.5 %
Moment of Inertia, I_{yy} (slug-ft^2)	- 2.5 %
Distance to cg, l_{cg} (ft)	+ 1.5 %
Velocity, V (ft/sec)	+ 5 %
Dynamic Pressure, Q (lbs/ft^2)	+ 10 %

These vehicle parameters are all perturbed in such a way as to simulate the vehicle conditions for an off nominal, high thrust trajectory.

Applying the high thrust conditions to the six optimization cases produced a reduction in the low frequency rigid body gain margin, GML, and the rigid body phase margin, PMR. The higher frequency margins (GMH, PMB, and GMB), however, generally improved. This trend

of reduction in GML and PMR with an increase in GMH, PMB, and GMB, was true for most other off nominal cases as well. The sum of the low frequency rigid body gain margin and the high frequecy rigid body gain margin, called the total rigid body gain margin, decreased for five of the six cases. Cases 5 and 6 showed the largest losses of total rigid body gain margin of approximately 0.5 db and 0.75 db respectively.

Low Thrust Conditions

For the decreased thrust conditions, the vehicle parameters were varied in a manner opposite to the high thrust case.

Thrust, F (lbs)	- 5 %
Mass, M (slugs)	+ 2.5 %
Moment of Inertia, I_{yy} (slug-ft^2)	+ 2.5 %
Distance to cg, I_{cg} (ft)	- 1.5 %
Velocity,V (ft/sec)	- 5 %
Dynamic Pressure, Q (lbs/ft^2)	- 10 %

For the low thrust conditions, the only stability margin which was reduced in the six optimization cases was the GML. All of the other stability margins were improved, and for Case 6, even the GML was improved. The total rigid body gain margin was reduced only in Cases 3 and 4.

High Aerodynamic Forces

For the six optimization cases with increased aerodynamic forces, the vehicle parameters were changed as follows:

Normal Force Coefficient, C_N + 10 %

Distance to cp, l_{cp} (ft)   + 10 %

Dynamic Pressure, Q (lbs/ft²)   + 10 %

The stability characteristics of the increased aerodynamic parameter cases were similar to those of the high thrust cases. The GML and PMR margins were reduced in general, and the higher frequency GMH, PMB, and GMB margins increased slightly. There was no general trend to the total rigid body gain margin, which was reduced in three of the six cases.

Low Aerodynamic Forces

For the six optimization cases with increased aerodynamic forces, the vehicle parameters were changed as follows:

Normal Force Coefficient, C_N - 10 %

Distance to cp, l_{cp} (ft)   - 10 %

Dynamic Pressure, Q (lbs/ft²)   - 10 %

As in the low thrust cases, only the GML margin was adversely affected by these off nominal conditions. Moreover, all of the stability margins, including the GML margin, improved for Case 6. The results of the total rigid body gain margin were again mixed.

Off Nominal Load Relief

Throughout this thesis, the aerodynamic angle of attack has been used as a measure of the load relief performance of the vehicle. As can be seen in Figure 5-9, the total normal force on the vehicle, F_N, is given by the expression:

$$F_N = -F\delta + SQC_{N\alpha}\,\alpha$$

Assuming steady-state conditions and equating the sum of the torques to zero gives,

$$F\delta\,l_{cg} + SQC_{N\alpha}\,\alpha\,l_{cp} = 0$$

Solving for $F\delta$ and substituting into the equation for F_N results in:

$$F_N = \frac{SQC_{N\alpha}\,\alpha\,l_{cp}}{l_{cg}} + SQC_{N\alpha}\,\alpha.$$

Or,

$$F_N = SQC_{N\alpha}\left(1 + \frac{l_{cp}}{l_{cg}}\right)\alpha.$$

For each of the transient response runs based on nominal vehicle and aerodynamic conditions, the same set of values was used for the parameters Q, $C_{N\alpha}$, l_{cp}, and l_{cg}. In the off nominal runs, however, a different set of parameter values was used for each run. Consequently, in these cases, the use of angle of attack alone is not sufficient for comparing load relief effectiveness. A more accurate measure of the normal load, F_N, can be obtained by defining a modified, "effective" angle of attack as follows,

$$\alpha_{eff} = \left[\frac{SQ'C'_{N\alpha}\left(1 + \dfrac{l'_{cp}}{l'_{cg}}\right)}{SQC_{N\alpha}\left(1 + \dfrac{l_{cp}}{l_{cg}}\right)}\right]\alpha'$$

where the primed quantities refer to the values ob†- ned from the off nominal runs and the unprimed quantities are the reference set of values obtained from the nominal run. The quantity, α_{eff}, is directly proportional to the normal force in all cases.

Using the effective angle of attack as a measure of load relief, the cases with high thrust and high aerodynamic forces tended to result in worse load relief, and the cases with low thrust and low aerodynamic forces tended to produce improved load relief. The key source of these trends seems to be the change in dynamic pressure. Larger aerodynamic loads produce larger torques on the vehicle which must be balanced by increased nozzle deflection angles. The net result is larger loads on the vehicle structure.
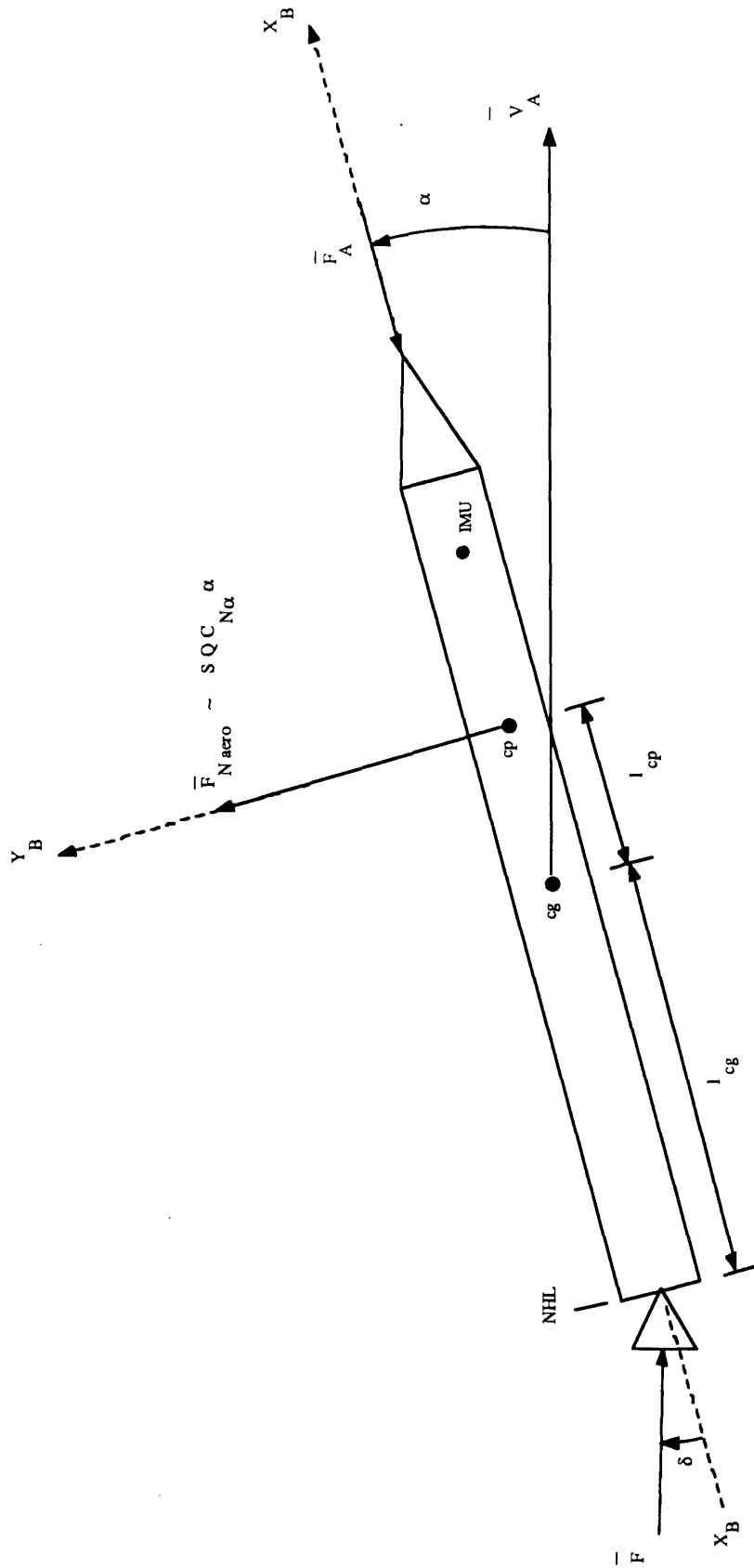
Figure 5-9. Simplified Forces on Vehicle

OFF NOMINAL RESULTS

A summary of results for the stability margins and load relief effectiveness for the off nominal cases is given in Tables 5-17 to 5-22.

Table 5-17. Case 1 Off Nominal Results

Case	Max a_{eff} (deg)	GML (db)	PMR (deg)	GMH (db)	PMB (deg)
Nominal	.3850	6.00	30.65	- 6.02	66.84
High Thrust	.3942 + 2.4 %	5.88 - 2.0 %	28.21 - 8.0 %	- 6.09 + 1.2 %	67.82 + 1.5 %
Low Thrust	.2846 - 26.1%	5.98 - 0.3 %	32.20 + 5.1 %	- 6.13 + 1.8 %	70.44 + 5.4 %
High Aero	.2939 - 23.7%	5.90 - 1.7 %	29.60 - 3.4 %	- 6.12 + 1.7 %	69.14 + 3.4 %
Low Aero	.3411 - 11.4%	5.97 - 0.5 %	30.95 + 1.0 %	- 6.10 + 1.3 %	69.12 + 3.4 %
Note: Percentages represent change from nominal value					

95

Table 5-18. Case 2 Off Nominal Results

Case	Max a_{eff} (deg)	GML (db)	PMR (deg)	GMH (db)	PMB (deg)
Nominal	.3354	6.14	30.00	- 6.08	68.69
High Thrust	.5040 + 50.3%	5.91 - 3.8 %	28.56 - 4.8 %	- 6.13 + 0.8 %	69.33 + 0.9 %
Low Thrust	.4306 + 28.4%	6.10 - 0.7 %	30.82 + 2.7 %	- 6.14 + 1.0 %	71.67 + 4.3 %
High Aero	.4941 + 47.6%	5.94 - 3.3 %	29.53 - 1.6 %	- 6.14 + 1.0 %	70.50 + 2.6 %
Low Aero	.3717 + 10.8%	6.08 - 1.0 %	30.06 + 0.2 %	- 6.12 + 0.7 %	70.49 + 2.6 %

Note: Percentages represent change from nominal value

Table 5-19. Case 3 Off Nominal Results

Case	Max α_{eff} (deg)	GML (db)	PMR (deg)	GMH (db)	PMB (deg)	GMB (db)
Nominal	.4419	6.01	30.16	- 6.02	79.72	-11.65
High Thrust	.4712 + 6.6 %	6.00 - 0.2 %	28.76 - 4.6 %	- 6.10 + 1.3 %	79.70 - 0.0 %	-11.17 - 4.1 %
Low Thrust	.4123 - 6.7 %	5.22 - 13.1%	31.02 + 2.9 %	- 6.04 + 0.3 %	82.60 + 3.6 %	-12.47 + 7.0 %
High Aero	.5807 + 31.4%	6.01 0.0 %	29.64 - 1.7 %	- 6.08 + 1.0 %	81.16 + 1.8 %	-11.81 + 1.4 %
Low Aero	.4154 - 6.0 %	5.10 - 15.1 %	30.32 + 0.5 %	- 6.06 + 0.7 %	81.15 + 1.8 %	-11.81 + 1.4 %
Note: Percentages represent change from nominal value						

Table 5-20. Case 4 Off Nominal Results

Case	Max a_{eff} (deg)	GML (db)	PMR (deg)	GMH (db)	PMB (deg)	GMB (db)
Nominal	.4150	6.10	30.10	- 6.10	80.00	-11.80
High Thrust	.4510 + 8.7 %	5.57 - 8.7 %	28.53 - 5.2 %	- 6.18 + 1.3 %	79.86 - 0.2 %	-11.30 - 4.2 %
Low Thrust	.3735 - 10.0%	5.78 - 5.3 %	31.06 + 3.2 %	- 6.11 + 0.2 %	82.80 + 3.5 %	-12.60 + 6.8 %
High Aero	.5272 + 27.0%	5.59 - 8.4 %	29.49 - 2.0 %	- 6.16 + 1.0 %	81.34 + 1.7 %	-11.94 + 1.2 %
Low Aero	.4033 - 2.8 %	5.70 - 6.6 %	30.32 + 0.7 %	- 6.14 + 0.7 %	81.33 + 1.7 %	-11.94 + 1.2 %

Note: Percentages represent change from nominal value

Table 5-21. Case 5 Off Nominal Results

Case	Max a_{eff} (deg)	GML (db)	PMR (deg)	GMH (db)	PMB (deg)	GMB (db)
Nominal	.3812	5.84	27.75	- 5.66	82.90	-11.77
High Thrust	.3852 + 1.0 %	5.20 - 11.0%	26.66 - 3.9 %	- 5.80 + 2.5 %	81.80 - 1.3 %	-11.22 - 4.7 %
Low Thrust	.3796 - 0.4 %	6.22 + 6.5 %	28.66 + 3.3 %	- 5.72 + 1.1 %	84.91 + 2.4 %	-12.53 + 6.5 %
High Aero	.4848 + 27.2%	5.21 - 10.8%	27.48 - 1.0 %	- 5.77 + 1.9 %	83.33 + 0.5 %	-11.87 + 0.9 %
Low Aero	.3732 - 2.1 %	6.26 + 7.2 %	28.02 + 1.0 %	- 5.75 + 1.6 %	83.32 + 0.5 %	-11.89 + 1.0 %

Note: Percentages represent change from nominal value

Table 5-22. Case 6 Off Nominal Results

Case	Max α_{eff} (deg)	GML (db)	PMR (deg)	GMH (db)	PMB (deg)	GMB (db)
Nominal	.3474	5.72	27.02	- 5.76	80.14	-11.76
High Thrust	.3639 + 4.7 %	4.94 - 13.6%	25.81 - 4.5 %	- 5.80 + 0.7 %	78.62 - 1.9 %	-11.11 - 5.5 %
Low Thrust	.3348 - 3.6 %	6.47 + 13.1%	27.96 + 3.5 %	- 5.71 + 0.9 %	81.70 + 2.0 %	-12.42 + 5.6 %
High Aero	.4161 + 19.8%	6.466 + 13.0%	26.67 - 1.3 %	- 5.78 + 0.4 %	80.17 + 0.0 %	-11.76 0.0 %
Low Aero	.3354 - 3.5 %	6.50 + 13.6%	27.29 + 1.0 %	- 5.76 0.0 %	80.16 + 0.0 %	-11.76 0.0 %

Note: Percentages represent change from nominal value

CHAPTER 6

ANALYSIS OF OPTIMIZATION RESULTS

In Chapter 5, results were presented from runs in which control loop parameters were optimized to maximize load relief while maintaining desired stability margins. The optimized parameters differ from the non-optimized parameters in three important respects. First, the load relief gain and time constant were increased significantly in each run. Second, the rate estimator gain was increased in each optimization, and the rate estimator time constant increased in four of the six optimizations. Third, the forward path compensation was modified to produce increased separation between the low frequency pole and the real zero.

LOAD RELIEF FEEDBACK ANALYSIS

An exact term by term analysis of the load relief performance of the entire vehicle and control system is beyond the scope of this thesis. However, an approximate analysis can provide some insight into the characteristics of the final solutions and explain the general trends of the six optimizations. The effects of the optimization process on vehicle response will be examined in terms of Case 3 in which conventional load relief is used in conjunction with fourth order forward path compensation for a nominal trajectory.

At first, one might expect that a higher load relief time constant would bring about a slower response. However, it can be shown that the effective load relief time constant is in fact given approximately by the ratio

$$\tau_{eff} = \frac{\tau_{LR}}{1 + K_{LR} K_o}$$

where τ_{LR} is the load relief time constant, K_{LR} is the load relief gain applied to the estimate of steady-state angle of attack as defined in Figure 5-3, and K_o is a function of the forward loop gain and vehicle parameters. Specifically,

$$K_o = \frac{K_F F I_{cg}}{K_F F I_{cg} - S Q C_{N\alpha} I_{cp}}$$

Thus, if the load relief gain, K_{LR}, is increased along with the time constant, τ_{LR}, as was the case in the optimization runs, then the effective time constant τ_{eff} may not be drastically altered. An approximate analysis showing the role of K_{LR} and τ_{LR} in determining the response to wind disturbances is presented below.

The wind disturbance normal to the vehicle velocity vector produces a direct change in angle of attack, $\Delta\alpha_w$, from the nominal trajectory value. Consequently, there is a change in aerodynamic normal force that produces a rotation rate of the velocity vector and an angular acceleration of the vehicle about its center of gravity. Similar effects are produced by the deflection of the engine nozzle from the closed loop control system response to aerodynamic forces. These dynamic effects produced by aerodynamic and thrust forces produce additional change in the angle of attack, $\Delta\alpha_{nw}$. The total change in angle of attack, $\Delta\alpha_{tot}$, caused by the wind disturbance is the sum of $\Delta\alpha_w$ and $\Delta\alpha_{nw}$.

Several simplifying assumptions were made to show the approximate effect of the wind disturbance angle of attack, α_w, on the total vehicle angle of attack, α_{tot}. First, it was assumed that $\Delta\alpha_{nw} \cong \Delta\theta$, the change in inertial attitude. Second, bending effects were assumed to be negligible since this analysis is limited to low frequencies. Third, the change in flight path angle, $\Delta\gamma$, was assumed to be zero (γ is shown along with the other angles in Figure 6-1). Fourth, the forward path compensation was replaced by its DC gain value. Finally, digital sampling effects were neglected in order to approximate the system in the

continuous s-domain, where the load relief feedback path is represented as a continuous low pass filter with a load relief gain, K_{LR}.

Figure 6-1. Angle of Attack Due to Wind

The perturbation in angular acceleration of the vehicle from the nominal trajectory value produced by the wind input is given by the expression

$$\Delta\ddot{\theta} = \omega_v^2 \Delta\alpha_{tot} + K_v \Delta\delta$$

where

$$\omega_v^2 = \frac{S Q C_{N\alpha} l_{cp}}{I}$$

and

$$K_v = \frac{F l_{cg}}{I}.$$

Using Laplace notation and replacing Δa_{tot} by $(\Delta a_{nw} + \Delta a_w)$, the above equation can be written as

$$s^2 \Delta\theta = \omega_v^2 (\Delta a_{nw} + \Delta a_w) + K_v \Delta\delta$$

or

$$s^2 \Delta a_{nw} = \omega_v^2 (\Delta a_{nw} + \Delta a_w) + K_v \Delta\delta$$

The above equation can be rewritten to give

$$\Delta a_{nw} = \Delta\theta = \left[\frac{\omega_v^2}{s^2 - \omega_v^2} \right] \Delta a_w + \left[\frac{K_v}{s^2 - \omega_v^2} \right] \Delta\delta$$

Representing the sum of the attitude and rate feedback as θ (1 + KFB s), and implementing the above equation into the control loop block diagram results in Figure 6-2. Closing the inner loop in Figure 6-2 results in Figure 6-3.
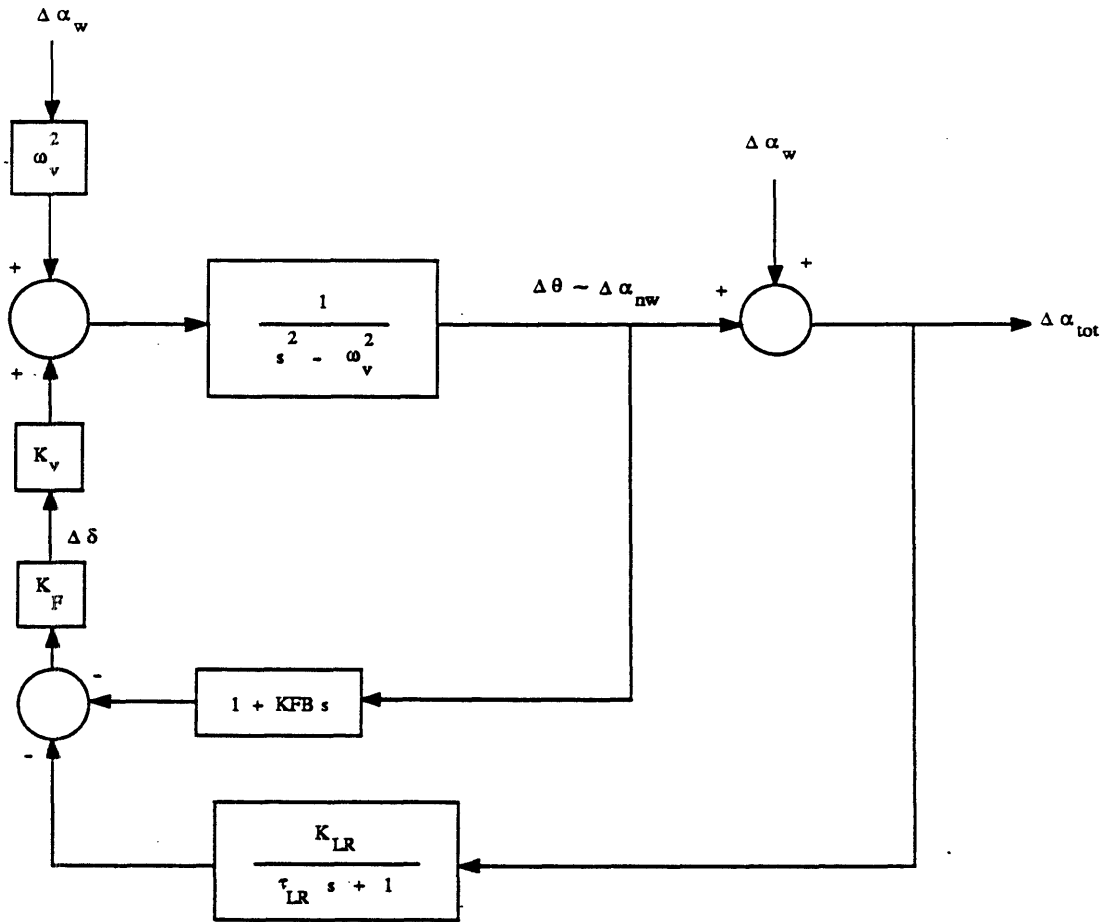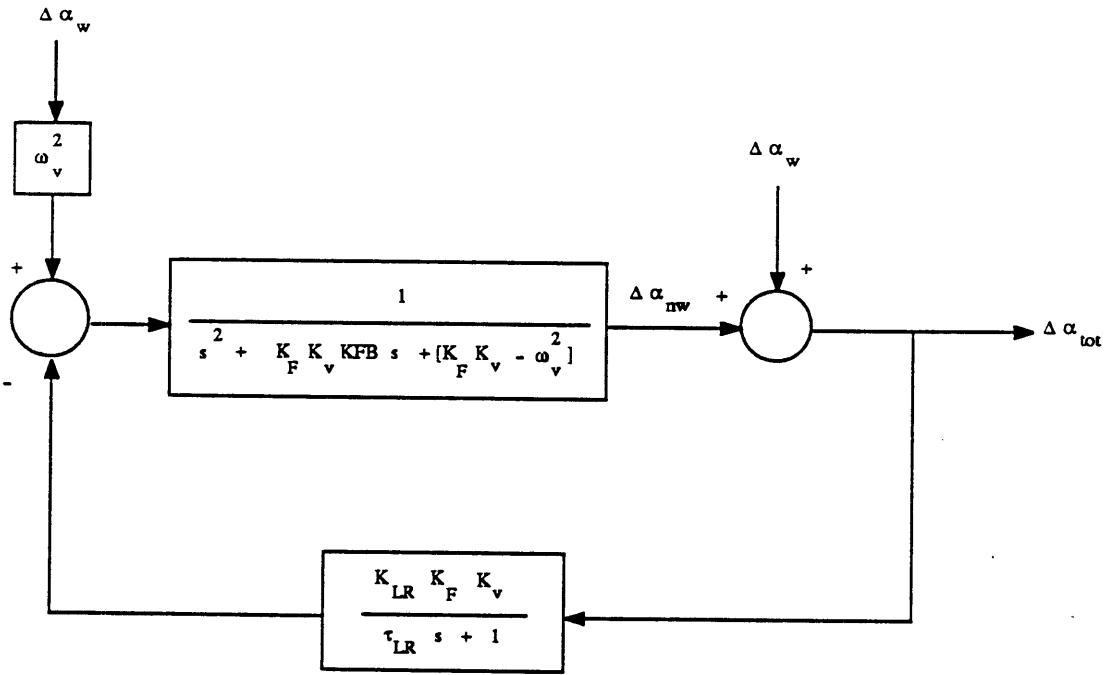
Figure 6-2. Block Diagram 1

$$\Delta\alpha_w$$

$$\omega_v^2$$

$$+$$

$$\frac{1}{s^2 + K_F K_v KFB\, s + [K_F K_v - \omega_v^2]}$$

$$\Delta\alpha_{nw}$$

$$\Delta\alpha_w$$

$$+$$

$$\Delta\alpha_{tot}$$

$$\frac{K_{LR} K_F K_v}{\tau_{LR}\, s + 1}$$

Figure 6-3. Block Diagram 2

Solving for the transfer function from α_w to α_{tot} results in the equation

$$\frac{\alpha_{tot}}{\alpha_w} = 1 + \frac{\omega_v^2(s + 1/\tau_{LR}) - K_F K_v K_{LR}/\tau_{LR}}{\left[(s + 1/\tau_{LR})(s^2 + K_F K_v KFB\, s + (K_F K_v - \omega_v^2)) + K_F K_v K_{LR}/\tau_{LR}\right]}$$

Substituting a ramped wind input of $\frac{0.4}{s^2}$ for α_w, and taking the inverse Laplace transform of this function results in a time domain equation of the form

$$\alpha_{tot}(t) = A + Bt - Ce^{-Dt} - Fe^{-Gt}\sin(\omega t + \phi).$$

The response to a ramped wind input indicated by the above equation was compared to the results produced by the transient response program for the initial and final control parameters of Case 3. For this comparison, the wind input profile otherwise used in the transient response program was replaced by a single ramped wind input for which the approximate analytical solution was obtained. Also, the noise from the IMU was zeroed in

order to provide a smoother response for comparison purposes. The comparisons of the approximate analytical results and the more accurate simulation results are shown in Figures 6-4 and 6-5, where the simulation results are represented by the continuous curve and the analytical results are shown as computed dots.

The first comparison of analytical and simulation results, shown in Figure 6-4, employed the final control parameters from the Case 3 optimization. In this case, with $K_{LR} = 3.48$, and $\tau_{LR} = 2.704$, the theoretical slope of the steady state response from the above analysis is .098 deg/sec, while the actual slope as measured directly from Figure 6-4 is approximately .100 deg/sec. This agreement is reasonably good, given all of the simplifying assumptions.

The second comparison, shown in Figure 6-5, used the initial set of control parameters from the Case 3 optimization run, with $K_{LR} = 1.27$, and $\tau_{LR} = 1.05$. The slope of the steady state response as measured directly from Figure 6-5 is approximately 0.15 deg/sec, while the theoretical slope from the above simplified analysis is 0.22 deg/sec. A possible reason for the less accurate modelling of the unoptimized, initial Case 3 parameters is that with less load relief, the assumptions that $\Delta a_{nw} \cong \Delta\theta$ and $\Delta\gamma \cong 0$ break down. This may be due to the fact that large normal forces on the vehicle rotate the Earth relative velocity vector, producing a change in the flight path angle, γ, and in a_{nw}, while the attitude control system maintains a relatively constant value of θ .

TRANSIENT RESPONSE

LOAD RELIEF PERFORMANCE
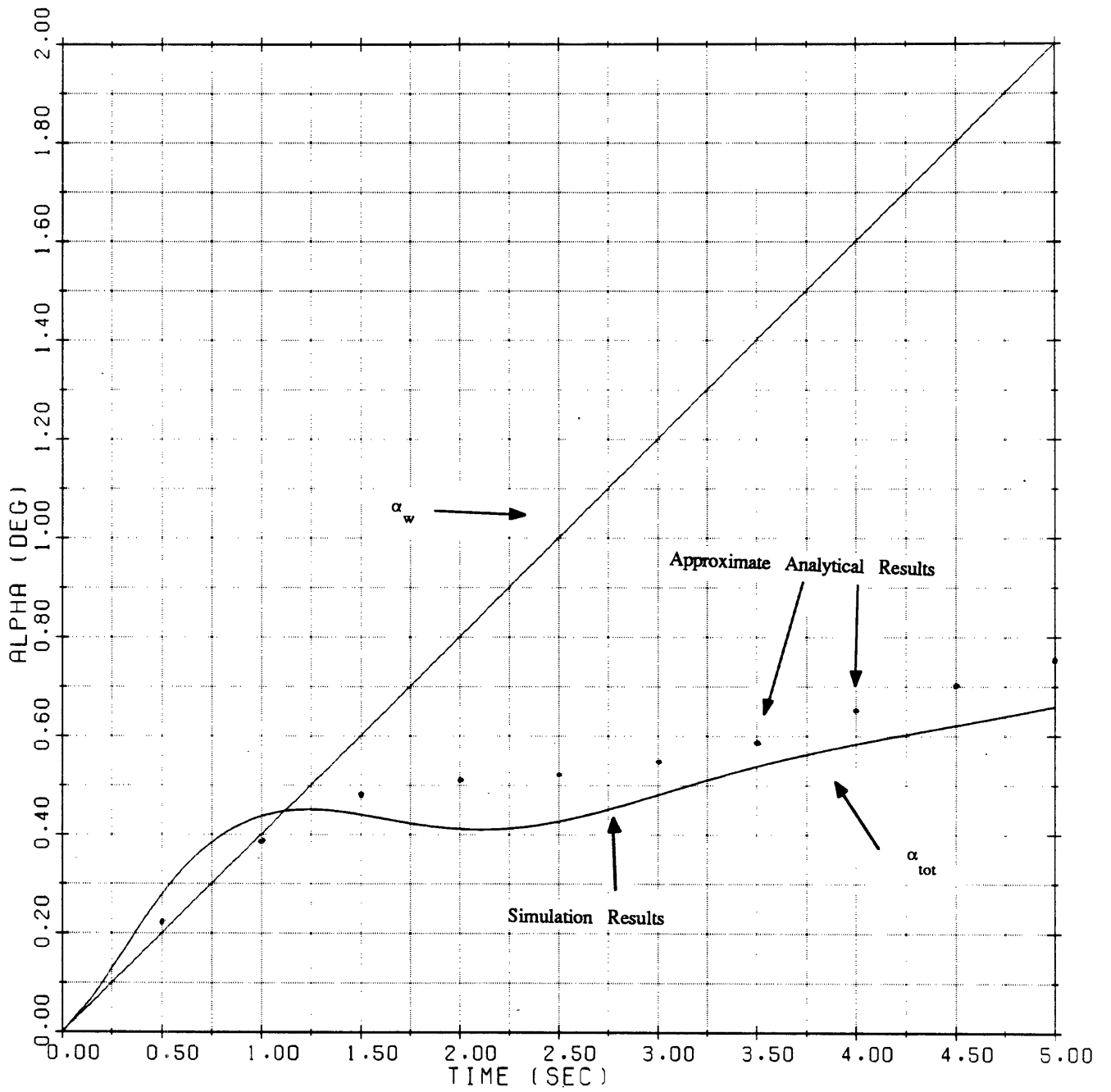
Figure 6-4. Ramp Response For Case 3 Final Control Parameters

TRANSIENT RESPONSE

LOAD RELIEF PERFORMANCE

Figure 6-5. Ramp Response For Case 3 Initial Control Parameters

If one introduces a further approximation into the analysis and neglects higher frequencies, one can redraw the block diagram of Figure 6-3 as shown in Figure 6-6. Manipulating Figure 6-6 results in Figure 6-7, which gives the closed loop transfer function relating α_{tot} to α_w .

Figure 6-6. Block Diagram 3

110

Figure 6-7. Block Diagram 4

Finally, by defining K_o as

$$K_o = \frac{K_F K_v}{K_F K_v - \omega_v^2} = \frac{K_F F l_{cg}}{K_F F l_{cg} - S Q C_{N\alpha} l_{cp}}$$

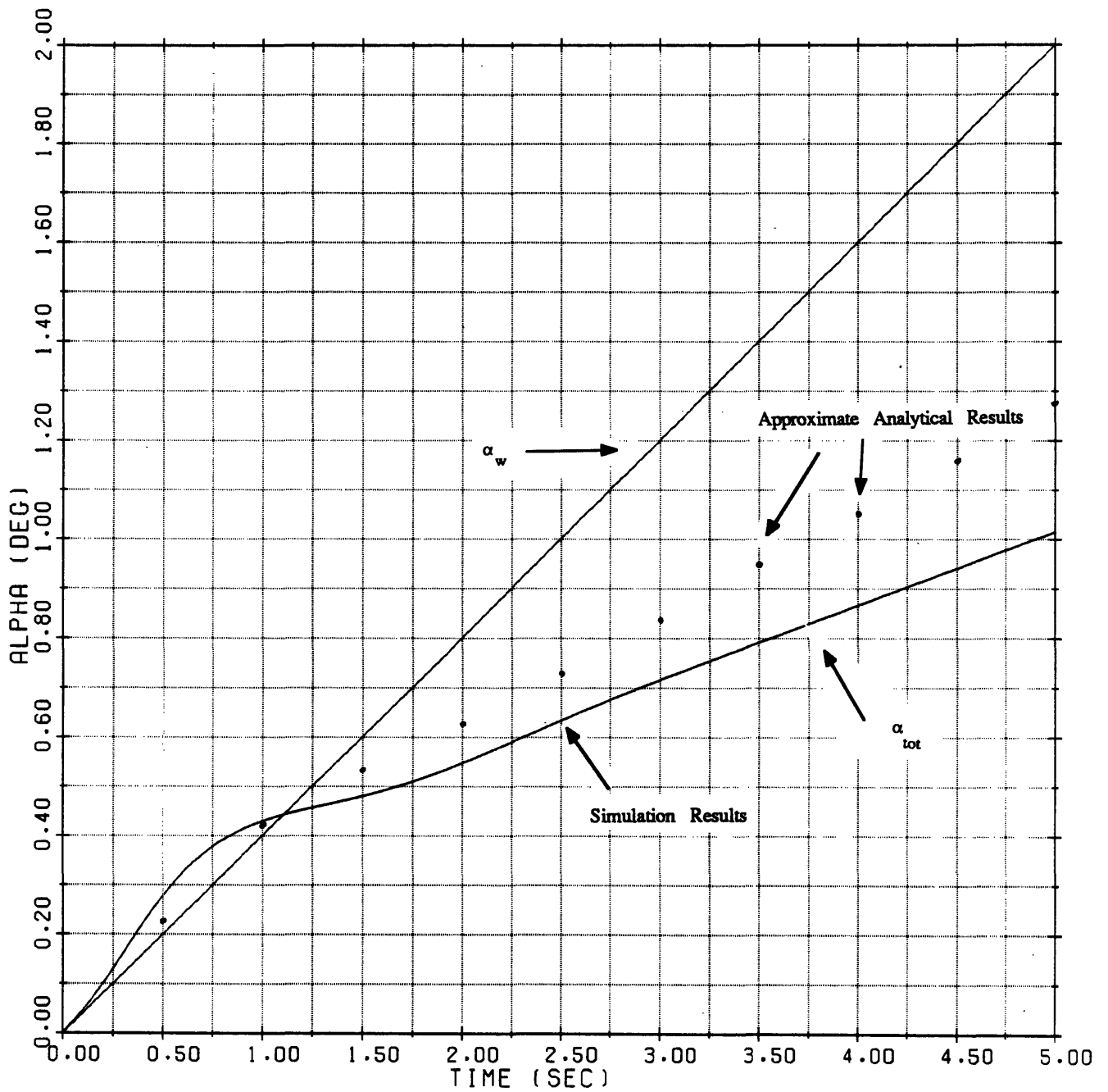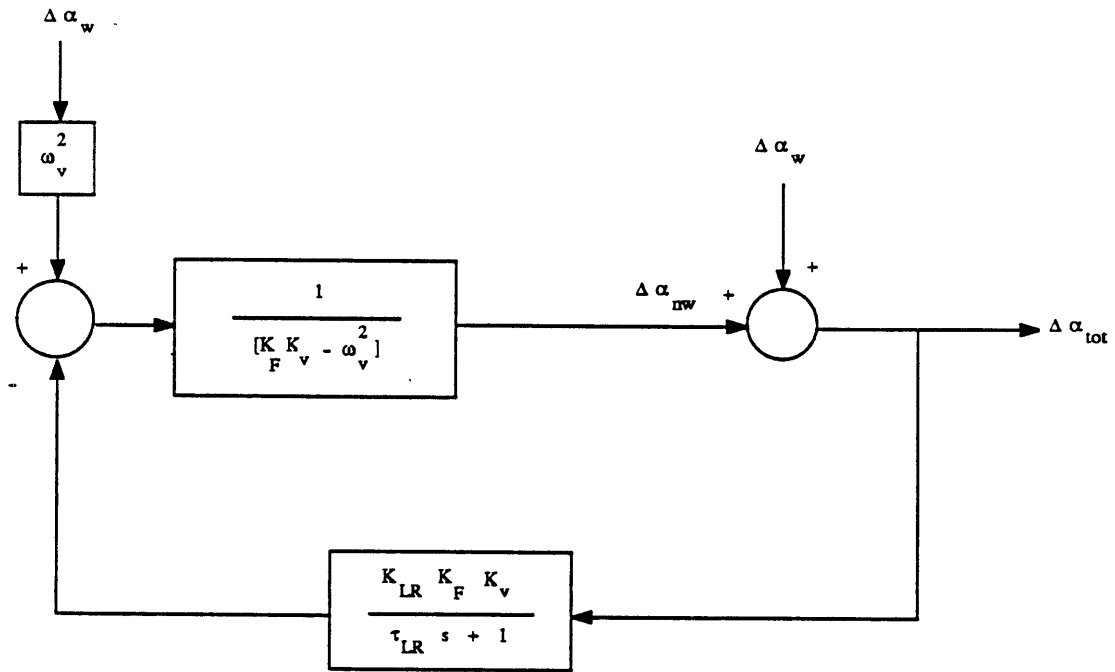one can write the approximate low frequency transfer function from a wind disturbance to the total angle of attack as

$$\frac{\alpha_{tot}}{\alpha_w} = \frac{K_o(\tau_{LR} s + 1)}{\tau_{LR} s + (1 + K_o K_{LR})}.$$

In order to obtain the time domain response of this control system to a ramped wind input of 0.4 deg/sec, we set $\alpha_w = 0.4/s^2$, and solve for α_{tot} :

$$\alpha_{tot}(s) = \frac{0.4}{s^2} \left[\frac{K_o(\tau_{LR} s + 1)}{\tau_{LR} s + (1 + K_o K_{LR})} \right]$$

$$= \frac{0.4 K_o}{s \left[s + (\dfrac{1 + K_o K_{LR}}{\tau_{LR}}) \right]} + \frac{0.4 K_o}{\tau_{LR} s^2 \left[s + (\dfrac{1 + K_o K_{LR}}{\tau_{LR}}) \right]}$$

Taking the inverse Laplace transform gives the function for α_{tot} in the time domain,
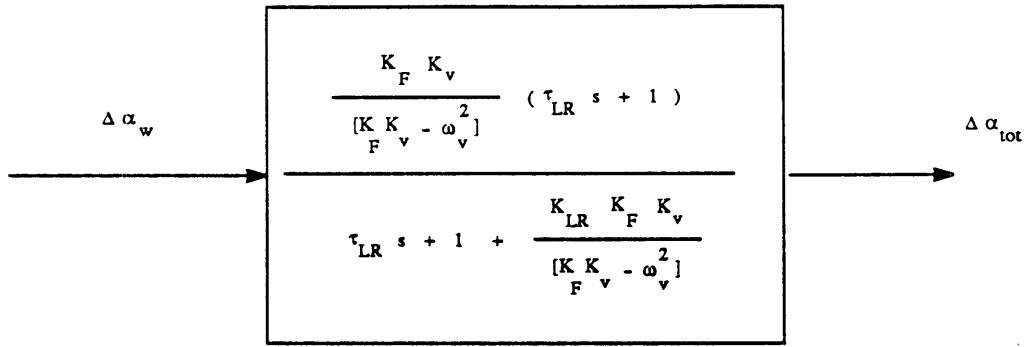
111

Figure 6-7. Block Diagram 4

Finally, by defining K_o as

$$K_o = \frac{K_F K_v}{K_F K_v - \omega_v^2} = \frac{K_F F l_{cg}}{K_F F l_{cg} - S Q C_{N\alpha} l_{cp}}$$

one can write the approximate low frequency transfer function from a wind disturbance to the total angle of attack as

$$\frac{\alpha_{tot}}{\alpha_w} = \frac{K_o(\tau_{LR} s + 1)}{\tau_{LR} s + (1 + K_o K_{LR})}.$$

In order to obtain the time domain response of this control system to a ramped wind input of 0.4 deg/sec, we set $\alpha_w = 0.4/s^2$, and solve for α_{tot} :

$$\alpha_{tot}(s) = \frac{0.4}{s^2} \left[\frac{K_o(\tau_{LR} s + 1)}{\tau_{LR} s + (1 + K_o K_{LR})} \right]$$

$$= \frac{0.4 K_o}{s \left[s + (\dfrac{1 + K_o K_{LR}}{\tau_{LR}}) \right]} + \frac{0.4 K_o}{\tau_{LR} s^2 \left[s + (\dfrac{1 + K_o K_{LR}}{\tau_{LR}}) \right]}$$

Taking the inverse Laplace transform gives the function for α_{tot} in the time domain,

$$a_{tot}(t) = \frac{0.4 K_o \tau_{LR}}{1 + K_o K_{LR}} \left\{ 1 - e^{-t(\frac{1 + K_o K_{LR}}{\tau_{LR}})} \right\} +$$

$$\frac{0.4 K_o \tau_{LR}}{(1 + K_o K_{LR})^2} \left\{ t(\frac{1 + K_o K_{LR}}{\tau_{LR}}) - 1 + e^{-t(\frac{1 + K_o K_{LR}}{\tau_{LR}})} \right\}$$

Rearranging terms results in

$$a_{tot}(t) = \frac{0.4 K_o t}{1 + K_o K_{LR}} + \frac{0.4 \tau_{LR} K_o^2 K_{LR}}{(1 + K_o K_{LR})^2} \left(1 - e^{-t(\frac{1 + K_o K_{LR}}{\tau_{LR}})} \right).$$

The steady state response to this ramped wind input should be an increasing ramp with a slope of approximately:

$$\text{steady state slope} = \frac{0.4 K_o}{1 + K_o K_{LR}}$$

It is important to note that this slope depends primarily on the load relief gain, K_{LR}. since the constant K_o does not vary significantly.

The above equation also shows (with the simplifying assumptions made) that the effective time constant of the transient response to a wind disturbance, τ_{eff}, is given by

$$\tau_{eff} = \frac{\tau_{LR}}{1 + K_o K_{LR}}.$$

Consequently, the approximate value of effective load relief time constant can be maintained at roughly the same level if τ_{LR} and K_{LR} increase correspondingly. For the final set of control parameters in Case 3, $K_o = 1.71$, $K_{LR} = 3.48$, and $\tau_{LR} = 2.70$, resulting in a τ_{eff} of 0.39 seconds. The initial set of control parameters for Case 3, using $K_o = 1.79$, $K_{LR} = 1.27$, and $\tau_{LR} = 1.05$, produce an effective time constant of 0.32 sec.

RATE ESTIMATOR DISCUSSION

Large values of KFB without changes in the forward path compensation can adversely affect stability. In order to maintain stability, it appears that the optimization program increased the separation between the low frequency pole and real zero in the forward path compensation so as to provide the shaping of the magnitude vs. frequency characteristics necessary to accommodate the larger rate feedback gain. Thus, both the rate feedback and forward paths were modified in order to allow an increased load relief gain and time constant.

The effects of the increased rate estimator gain on load relief performance were examined in two ways. First, the initial set of parameters in Case 3 were changed by substituting the load relief gain and time constant of the final parameters in Case 3. The result was a Max a of 0.4091 degrees, which was less than the 0.4419 degrees resulting from the set of parameters in Case 3. However, the total rigid body gain margin (i.e. the sum of the maginitudes of the GML and GMH margins) was compromised from 12 db to 11.4 db. These results imply that the primary role of optimizer adjustments in the forward path compensation and rate feedback filter is to prevent unacceptable degradations in stability margins rather than to improve load relief performance. Second, the load relief gain was zeroed for the initial and final parameters of Case 3, and a transient response was performed. The initial set of parameters, which had a KFB = .28, produced a Max a of 1.416 degrees, while the final Case 3 parameters with a KFB = .46, resulted in a Max a of 1.379 degrees. This result shows that an increased rate feedback gain by itself does not greatly affect load relief performance, and it is, in fact, the load relief path alone which produces improvements in load relief performance.

The time constant of the rate estimator increased in four of the six optimizations. An increased rate estimator time constant implies heavier weighting to the high frequency component of estimated rate based on the δ and ΔV_N signals. One possible reason why the rate estimator time constant may have been increased so significantly in certain cases (e.g. Cases 5 and 6), is that τ_{est} has a unique property over all the other control parameters. The rate estimator time constant has a large impact on the first bending mode phase margin, a significantly smaller effect on the rigid body phase margin, and almost no effect on all other margins. As a result, the optimizer can adjust τ_{est} to selectively reduce or increase the value of the PMB margin as necessary. Another possible reason for the increase in the value of τ_{est} is that the rate estimator used in this thesis uses the latest value of δ instead of an average of δ over the last two control cycles. As a result, the high frequency rate estimate introduces a slight lead into the system.

A significantly increased τ_{est} may be undesirable because it increases the rate estimate's sensitivity to vehicle parameters since the computation of the high frequency rate estimate used coefficients that are functions of vehicle parameters; moreover, it may increase the gain applied to the error sources in the δ and ΔV_N signals. However, the error from noise in the $\Delta \theta$ signal would be decreased as τ_{est} goes up. An optimal value of τ_{est} would require a trade-off between these error sources, which would require modification of the cost function used in the optimization program.

CHAPTER 7

CONCLUSIONS AND RECOMMENDATIONS

The development and use of the Automated Control Loop Design Tool has produced the six optimized control systems presented in this thesis. Both the conventional and alternative optimized control systems showed 26 to 52 percent improvement in load relief performance over current typical control system designs. The comparison of conventional load relief with alternative load relief systems has shown that the addition of the δ signal gives the flexibility to achieve enhanced load relief performance over the conventional load relief systems using ΔV_N feedback alone. However, the alternative load relief design's improvement over the conventional design's performance was less than dramatic, from 6 to 13 percent.

The ACLDT showed itself to be a very useful design tool, and produced some unanticipated results. By simultaneously manipulating the forward path compensation parameters, the ACLDT produced stable designs with feedback gains which were previously considered too high to be feasible. The large increases in the load relief and rate feedback gains and time constants while still maintaining required stability margin constraints was significant. All of the six optimizations produced designs which gave a greater weighting to the feedback signals to improve load relief response. In the case of the rate feedback signal, a substantial increase in feedback gain was made possible by increasing the separation between the low frequency pole and zero of the forward path compensation. These designs are significantly different from any designs previously constructed from intuitive synthesis approaches.

One possible drawback of the increased feedback gains is a reduced response of the vehicle attitude to the command supplied by the guidance/steering system. This in turn could result in a reduced capacity of the guidance/steering system to meet trajectory shap-

115

ing objectives. The possibility of drawbacks in these designs is a reflection of the limitation of any type of optimization process --- that is, that the optimizer used is limited by the choice of cost function provided by the user. The program has no engineering judgement, and pursues its single-minded decrease of cost without regard to other factors which may occur to the user after the optimization has been completed. Besides being sensitive to the choice of cost function, the ACLDT is also sensitive to the choice of parameters to be optimized and the control system feedback configuration. These factors give rise to the areas where much work remains to be done.

RECOMMENDATIONS FOR FUTURE STUDY

Not only are there a number of ways in which the design tool developed in this thesis could be improved, but there are also several other applications for which it could be used. First, an alternative cost function could be used that would reflect the trade-off between load relief optimization and steering response optimization. Second, the use of non-linear optimzation subroutines instead of linear programming could be studied to attempt to reduce computational costs and convergence time. Third, a method to perturb *all* of the control parameters in order to meet the stability margin constraints, instead of just the control parameters that are in the basis of the linear program, should be investigated. The use of a psuedo-inverse of the non-square matrix of margin sensitivities is a possibility. Fourth, the ACLDT could be applied to several other critical points in the trajectory in addition to the maximum dynamic pressure point considered in this thesis. For example, there is a major advantage to improving load relief at the critical time of thrust tail off at the end of the first stage, when minimization of angle of attack is crucial to successful staging. Fifth, an additional filter in the rate feedback path could be added in order to compensate for increased rate feedback gains without affecting the attitude control in the forward path. Finally, the

possibility of adverse effects of increased load relief on the overall shape of the trajectory needs to be investigated.

APPENDIX A

ANGLE OF ATTACK ESTIMATES

In order to derive an estimate of the vehicle steady-state angle of attack, the following assumptions are made. First, the high frequency effects from bending mode deflections and from "Tail wags dog" forces are neglected. Thus, the vehicle is assumed to be a rigid body with only two forces acting on it. As is shown in Figure A-1, these forces consist of the noz-zle-deflected engine thrust and the aerodynamic forces centered at the vehicle center of pressure (cp). Second, the vehicle is assumed to be in steady-state flight with zero angular accelerations about the vehicle center of gravity (cg). Third, since the maximum engine noz-zle deflection, δ is only six degrees, it is assumed that $\sin(\delta) \cong \delta$.

Taking the sum of the moments about the vehicle cg to be zero gives

$$F\, \delta\, l_{cg} + S\, Q\, C_{N\alpha}\, \alpha_{ss}\, l_{cp} = 0$$

$$\alpha_{ss} \cong - \frac{F\, l_{cg}}{S\, Q\, C_{N\alpha}\, l_{cp}}\ \delta$$

Obtaining a steady-state angle of attack estimate in terms of the change in velocity nor-mal to the vehicle roll axis, ΔV_N , requires summing all of the forces on the vehicle normal to the roll axis. The normal acceleration of the vehicle, a_N , is approximately equal to the mea-sured change in normal velocity divided by the sampling time, T.

$$a_N \cong \frac{\Delta V_N}{T}$$

Summing the normal accelerations produced by aerodynamic and thrust forces,
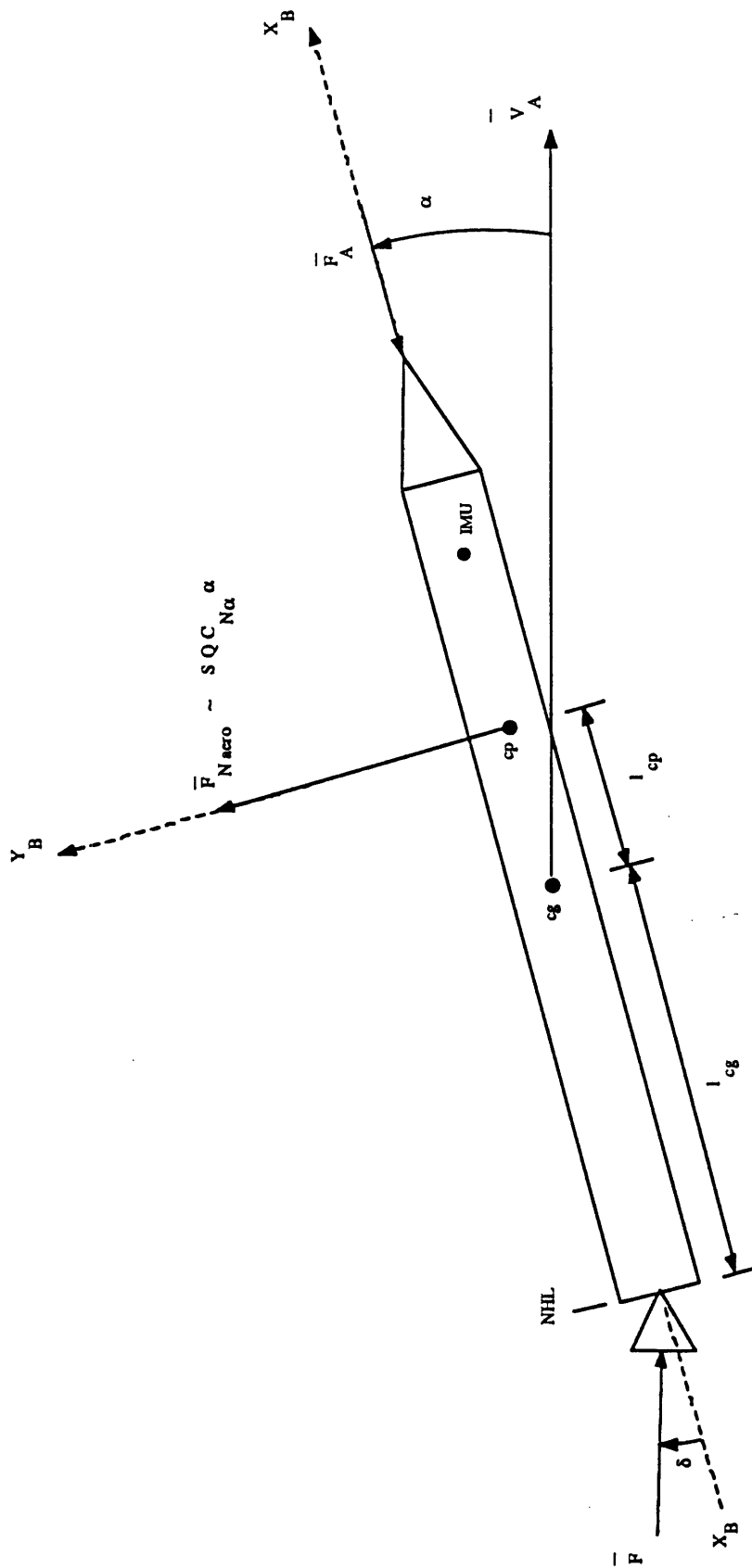
119

X_B

$\bar{V}_A$

α

$\bar{F}_A$

IMU

$\bar{F}_{N\,aero} \sim SQC_{N\alpha}\,\alpha$

cp

Y_B

l_{cp}

cg

l_{cg}

NHL

δ

$\bar{F}$

X_B

Figure A-1. Simplified Vehicle Forces

$$\frac{\Delta V_N}{T} \cong -\frac{F}{m}\delta + \frac{S\ Q\ C_{Na}\ a_{ss}}{m}$$

Substituting

$$\delta \cong -\frac{S\ Q\ C_{Na}\ a_{ss}\ l_{cp}}{F\ l_{cg}}$$

gives

$$\frac{\Delta V_N}{T} \cong \frac{S\ Q\ C_{Na}}{m}\left[\frac{l_{cp}}{l_{cg}} + 1\right] a_{ss}$$

or

$$a_{ss} \cong \frac{m\ l_{cg}}{S\ Q\ C_{Na}\ (l_{cp} + l_{cg})\ T}\quad \Delta V_N.$$

APPENDIX B

OPTIMIZATION RUN EXAMPLE

This sample optimization was performed using a fourth-order forward loop bending filter and alternative load relief, for worst-case maximum dynamic pressure environmental conditions. The rigid body gain margin constraints were $\pm$ 5.75 db. The rigid body phase margin constraint was 27 degrees. The bending mode phase margin constraint was 80 degrees, and the bending mode gain margin constraint was -11.75 db. In the sample below, the stability margin vector, **G** is defined as

$$\mathbf{G} = [\text{GML, PMR, GMH, PMB, GMB}]$$

Where

GML = Low Frequency Rigid Body Gain Margin

PMR = Rigid Body Phase Margin

GMH = High Frequency Rigid Body Gain Margin

PMB = First Bending Mode Phase Margin

GMB = Second Bending Mode Gain Margin

The control parameter vector, **X**, is defined as

$$\mathbf{X} = [K_F, Zero, Pole1, Pole2, \tau_{est}, \text{KFB}, KV_{LR}, KD_{LR}, \tau_{LR}]$$

Where

K_F = Forward Loop Gain

Zero = - Forward Loop Zero (w-domain)

Pole1 = - Forward Loop Pole (w-domain)

Pole2 = - Forward Loop Pole (w-domain)

τ_{est} = Rate Estimator Time Constant

KFB = Rate Estimator Gain

KV_{LR} = ΔV_N Load Relief Gain

KD_{LR} = δ Load Relief Gain

τ_{LR} = Load Relief Time Constant

Maxalpha is defined as the maximum angle of attack calculated in the transient response subroutine.

In the following synopsis of the optimization run, values for Maxalpha, **G**, and **X** are given at the beginning of each Major Loop Cycle. These values are also given for the beginning of each Minor Loop Cycle of the first Major Loop Cycle.

1st Major Loop

Maxalpha = .71663

G = {5.7651, 27.014, -6.1011, 88.6739, -12.2264}

X = (2.4772, 0.03129, 0.02041, 1.12225, 0.4062, 0.15000, 0.2002, 0.0600, 1.3191)

2nd Minor Loop

Maxalpha = .71398

G = {5.7657, 27.008, -6.0836, 88.2985, -12.2068}

X = (2.5113, 0.03129, 0.02016, 1.10846, 0.44718, 0.15110, 0.2002, 0.0769, 1.3576)

3rd Minor Loop

Maxalpha = .70938

G = {5.7667, 27.028, -6.0811, 88.2804, -12.2047}

X = (2.5154, 0.03129, 0.02007, 1.10314, 0.44771, 0.15278, 0.2002, 0.1060, 1.3983)

4th Minor Loop

Maxalpha = .69881

G = {5.7669, 27.015, -6.0870, 88.3100, -12.2132}

X = (2.5227, 0.03223, 0.02005, 1.09501, 0.43116, 0.15889, 0.2029, 0.1502, 1.4542)

5th Minor Loop

Maxalpha = .68309

G = {5.8194, 27.118, -6.0607, 88.2463, -12.1978}

X = (2.5198, 0.03385, 0.02039, 1.08235, 0.41202, 0.16683, 0.2092, 0.2076, 1.5269)

6th Minor Loop

Maxalpha = .66606

G = {5.9247, 27.276, -5.9862, 87.7019, -12.1430}

X = (2.5475, 0.03588, 0.02064, 1.05631, 0.42908, 0.17684, 0.2172, 0.2789, 1.6311)

2nd Major Loop

Maxalpha = .64677

G = {6.1084, 27.477, -5.8668, 87.0768, -12.0639}

X = (2.5744, 0.03839, 0.02100, 1.02651, 0.44212, 0.18922, 0.2272, 0.3685, 1.7577)

3rd Major Loop

Maxalpha = .60183

G = {6.4122, 27.300, -5.7799, 85.3701, -12.1317}

X = (2.6913, 0.04472, 0.01904, 0.93072, 0.61260, 0.24571, 0.2542, 0.8016, 2.4169)

4th Major Loop

Maxalpha = .42864

G = {5.7026, 27.020, -5.7357, 82.5630, -11.8520}

X = (2.8347, 0.04506, 0.01619, 0.94182, 0.75821, 0.27694, 0.5442, 1.8710, 3.6217)

5th Major Loop

Maxalpha = .37589

G = {5.8001, 27.016, -5.7582, 80.2975, -11.7836}

X = (2.8484, 0.04178, 0.01525, 0.95529, 1.20713, 0.31642, 0.9102, 4.1271, 6.6574)

6th Major Loop

Maxalpha = .37639

G = {5.8549, 27.184, -5.7220, 80.0027, -11.7498}

X = (2.8401, 0.04829, 0.01519, 0.95152, 1.22450, 0.32272, 0.8849, 4.4980, 6.9097)

7th Major Loop

Maxalpha = .37176

G = {5.8604, 27.197, -5.7189, 80.0002, -11.7485}

X = (2.8391, 0.04835, 0.01508, 0.94430, 1.19170, 0.32914, 0.8456, 5.2785, 7.3349)

8th Major Loop

Maxalpha = .36400

G = {5.8559, 27.176, -5.7216, 80.0060, -11.7494}

X = (2.8336, 0.04854, 0.01501, 0.94037, 1.16588, 0.33447, 0.8470, 6.0674, 7.8662)

9th Major Loop

Maxalpha = .34738

$G = \{5.7222, 27.017, -5.7608, 80.1367, -11.7556\}$

$X = (2.8293, 0.04892, 0.01427, 0.95013, 0.96264, 0.35315, 1.3411, 9.1884, 11.113)$

10th Major Loop

Maxalpha $= .34916$

$G = \{5.7501, 27.052, -5.7500, 80.0506, -11.7501\}$

$X = (2.8330, 0.04892, 0.01426, 0.94947, 0.98282, 0.35330, 1.3634, 9.3062, 11.399)$

At End Of 10th Major Loop

Maxalpha $= .35675$

Since the maximum angle of attack increased in value for two consecutive major loop cycles, the program reverts to the solution at the start of the 9th Major Loop Cycle and terminates.

Final Solution

Maxalpha $= .34738$

$G = \{5.7222, 27.017, -5.7608, 80.1367, -11.7556\}$

$X = (2.8293, 0.04892, 0.01427, 0.95013, 0.96264, 0.35315, 1.3411, 9.1884, 11.113)$

APPENDIX C

DERIVATION OF WEIGHTING FACTOR

As can be seen from Figures 5-3 and 5-4,

$$KV_{LR} = \frac{m\ I_{cg}}{S\ Q\ C_{Na}\ (\ I_{cp} + I_{cg}\ )\ T}\ \ R\ K_{LR}$$

and

$$KD_{LR} = \frac{-\ F\ I_{cg}}{S\ Q\ C_{Na}\ I_{cp}}\ \ (1 - R)\ K_{LR.}$$

Dividing the two expressions results in

$$\frac{KV_{LR}}{KD_{LR}} = \frac{m\ I_{cp}}{-\ F\ (\ I_{cp} + I_{cg}\ )\ T}\ \ \frac{R}{(1 - R)}\ .$$

Rearranging gives

$$-\ (1 - R)\ F\ (\ I_{cp} + I_{cg}\ )\ T\ KV_{LR} = R\ (\ m\ I_{cp}\ KD_{LR})\ .$$

or

$$R\ \{\ F\ (\ I_{cp} + I_{cg}\ )\ T\ KV_{LR} -\ m\ I_{cp}\ KD_{LR}\ \} = F\ (\ I_{cp} + I_{cg}\ )\ T\ KV_{LR.}$$

Finally,

$$R = \frac{F\ (\ I_{cp} + I_{cg}\ )\ T\ KV_{LR}}{\{\ F\ (\ I_{cp} + I_{cg}\ )\ T\ KV_{LR} -\ m\ I_{cp}\ KD_{LR}\ \}}\ .$$

LIST OF REFERENCES

1. Pfaffenberger, Roger C., and David A. Walker, *Mathematical Programming for Economics and Business*, Iowa State University Press Ames, Iowa, 1976, Pages 3-130.

2. Jeter, Melvyn W., *Mathematical Programming: An Introduction to Optimization*, Marcel Dekker, Inc. New York, NY, 1986, Pages 81-117.

3. Rao, S. S., *Optimization: Theory and Applications*, John Wiley & Sons New York, NY, 1984, Pages 86-130.

4. Machles, Gary W. and Francis D. Hauser, "Applications of Nonlinear Programming for Automated Multivariable Control Design", *Journal of Guidance and Control*, July-August, 1985, Pages 525-529.

5. Hauser, Francis D., "Nonlinear Programming Algorithm For The Automated Design and Optization of Flexible Space Vehicle Autopilots", *AIAA Paper 73-892*, August, 1973.