A MULTI-LOOP GUIDANCE SCHEME USING SINGULAR PERTURBATION AND LINEAR QUADRATIC REGULATOR TECHNIQUES SIMULTANEOUSLY

by

Philip Merton Bushong

Dissertation submitted to the Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

in

Aerospace Engineering

APPROVED:

F. H. Lutze, Chairman

W. M. Mason

C. J. Ribbens

E. M. Cliff

W. C. Durham

July, 1991

Blacksburg, Virginia

A MULTI-LOOP GUIDANCE SCHEME USING SINGULAR PERTURBATION AND LINEAR QUADRATIC REGULATOR TECHNIQUES SIMULTANEOUSLY

by

Philip Merton Bushong

Committee Chairman: Frederick H. Lutze

Aerospace and Ocean Engineering

(ABSTRACT)

A design method for a multi-loop mixed discrete/continuous trajectory following pitch control algorithm for a generic aerospace vehicle is presented. This design methodology is facilitated by a time scale separation observed in the dynamical system. Two variations of this algorithm are considered, with features and drawbacks of both evaluated. The algorithm is then tested by simulations with two vehicles flying arbitrary trajectories. Results are presented for a thrust-vector controlled high-performance missile without atmospheric effects, and for a single-stage-to-orbit hypersonic vehicle with both elevator and thrust-vector control. It is shown that the control algorithm results in a pitch loop feedback controller that is robust and very stable, and is at least near optimal for the class of trajectories considered. No claims of optimality are made for the outer loop, but it is shown in the simulations that the outer loop tracker can do a reasonable job of following the prescribed nominal trajectory.

Acknowledgments

Of the many people that deserve a part of the credit for the completion of this work, many of whom will no doubt go unmentioned, I would like to thank my parents, Merton and Carolyn Bushong of Lucas, Texas, and my brothers and sister of around the globe, for having the confidence in me to push me on, and to stand by me both emotionally and financially during a long and, at times, seemingly endless effort.

For technical and professional help I would like to thank my advisor Dr. Frederick H. Lutze, Jr. for his support, advice and guidance during the course of study at VPI&SU. Without this advice and guidance this research work would not have progressed to this point. I would also like to thank Dr. Gene Cliff for his valuable assistance in this research effort, and for his instruction in engineering. Thanks go also to Dr. Wayne Durham for his valued help in this project. I should mention Dr. Rajiv Chowdry, who was instrumental in developing my interest in this particular topic, and was also very helpful in my gaining an early appreciation of it.

I would like to thank Dr. Cliff, Dr. Durham, Dr. Bill Mason, and Dr. Cal Ribbens for agreeing to be on my examination committee.

This work was partially supported by the Aerospace and Ocean Engineering Department of Virginia Polytechnic Institute & State University.

This material is based upon work supported under a National Science Foundation Graduate Fellowship.

SHORT CONTENTS

1.0 INTRODUCTION 1

2.0 INNER LOOP GUIDANCE ALGORITHM DEVELOPMENT 9

3.0 OUTER LOOP GUIDANCE ALGORITHM DEVELOPMENT 28

4.0 VEHICLE SIMULATIONS 38

5.0 CONCLUDING REMARKS 67

REFERENCES 69

FIGURES 72

VITA 150

Table of Contents

1.0 INTRODUCTION 1

1.1 Suboptimal Trajectories 3

1.2 The Multi-Loop Control Formulation 5

1.2.1 Inner Loop Control 7

1.2.2 Outer Loop Control 7

2.0 INNER LOOP GUIDANCE ALGORITHM DEVELOPMENT 9

2.1 Singular Perturbation Theory 9

2.2 Generic Aerospace Vehicle Pitch Control 15

3.0 OUTER LOOP GUIDANCE ALGORITHM DEVELOPMENT 28

3.1 Continuous Linear Quadratic Regulator 28

3.1.1 Terminal Weighting And Final Control Behavior 30

3.1.2 Choosing Weights For States and Controls 31

3.2 Discrete Linear Quadratic Regulator 32

3.3 On Continuing The Discrete System Realization 35

4.0 VEHICLE SIMULATIONS 38

4.1 Exo-Atmospheric Missile 38

4.1.1 Inner Loop Control 39

4.1.2 Modeling Error Effects 43

4.1.3 Nominal Trajectory 44

4.1.4 Inner Loop-Only Control Simulations 46

4.1.5 Outer Loop Control 47

4.1.5.1 Continuous Linear Quadratic Regulator 47

4.1.5.2 Continuous Regulator Control Simulations 48

4.1.5.3 Discrete Linear Quadratic Regulator 52

4.2 NASA LaRC Hypersonic Conical Accelerator 53

4.2.1 Inner Loop Implementation 54

4.2.2 Outer Loop Implementation 61

5.0 CONCLUDING REMARKS 67

REFERENCES 69

FIGURES 72

VITA 150

List of Figures

Figure 2.1 Aerospace Vehicle General Geometry 72

Figure 4.1 Simplified Model Configuration 73

Figure 4.2 Pole Travel With Control Effectiveness Parameter Change 74

Figure 4.3 Nominal Pitch Attitude (θ) 75

Figure 4.4 Nominal Pitch Rate (q) 76

Figure 4.5 Open Loop Range Error, Nominal Geometry 77

Figure 4.6 Open Loop Altitude Error, Nominal Geometry 78

Figure 4.7 Open Loop Velocity Error, Nominal Geometry 79

Figure 4.8 Open Loop Pitch Error and Control (β), Nominal Geometry 80

Figure 4.9 Open Loop Range Error, 0.1 m Thrust Offset 81

Figure 4.10 Open Loop Altitude Error, 0.1 m Thrust Offset 82

Figure 4.11 Open Loop Velocity Error, 0.1 m Thrust Offset 83

Figure 4.12 Closed Loop Mass Error
Nominal Geometry — Continuous Controller 84

Figure 4.13 Closed Loop Range Error
Nominal Geometry — Continuous Controller 85

Figure 4.14 Closed Loop Altitude Error
Nominal Geometry — Continuous Controller 86

Figure 4.15 Closed Loop Velocity Error
Nominal Geometry — Continuous Controller 87

Figure 4.16 Closed Loop Flight Path Angle Error
Nominal Geometry — Continuous Controller 88

Figure 4.17 Closed Loop Pitch Attitude Error
Nominal Geometry — Continuous Controller 89

Figure 4.18 Closed Loop Pitch Rate Error
Nominal Geometry — Continuous Controller 90

Figure 4.19 Applied Thrust Vector Control (β)
Nominal Geometry — Continuous Controller 91

Figure 4.20 Applied Throttle Deviation (η)
Nominal Geometry — Continuous Controller 92

Figure 4.21 Closed Loop Mass Error
0.1 m Thrust Offset — Continuous Controller 93

Figure 4.22 Closed Loop Range Error
0.1 m Thrust Offset — Continuous Controller 94

Figure 4.23 Closed Loop Altitude Error
0.1 m Thrust Offset — Continuous Controller 95

Figure 4.24 Closed Loop Velocity Error
0.1 m Thrust Offset — Continuous Controller 96

Figure 4.25 Closed Loop Flight Path Angle Error
0.1 m Thrust Offset — Continuous Controller 97

Figure 4.26 Closed Loop Pitch Attitude Error
0.1 m Thrust Offset — Continuous Controller 98

Figure 4.27 Closed Loop Pitch Rate Error
0.1 m Thrust Offset — Continuous Controller 99

Figure 4.28 Applied Thrust Vector Control (β)
0.1 m Thrust Offset — Continuous Controller 100

Figure 4.29 Applied Throttle Deviation (η)
0.1 m Thrust Offset — Continuous Controller 101

Figure 4.30 Closed Loop Mass Error
Nominal Geometry — Continuous Controller
Simultaneously Integrated Nominal 102

Figure 4.31 Closed Loop Range Error
Nominal Geometry — Continuous Controller
Simultaneously Integrated Nominal 103

Figure 4.32 Closed Loop Altitude Error
Nominal Geometry — Continuous Controller
Simultaneously Integrated Nominal 104

Figure 4.33 Closed Loop Velocity Error
Nominal Geometry — Continuous Controller
Simultaneously Integrated Nominal 105

Figure 4.34 Closed Loop Flight Path Angle Error
Nominal Geometry — Continuous Controller
Simultaneously Integrated Nominal 106

Figure 4.35 Closed Loop Pitch Attitude Error
Nominal Geometry — Continuous Controller
Simultaneously Integrated Nominal 107

Figure 4.36 Closed Loop Pitch Rate Error
Nominal Geometry — Continuous Controller
Simultaneously Integrated Nominal 108

Figure 4.37 Applied Thrust Vector Control (β)
Nominal Geometry — Continuous Controller
Simultaneously Integrated Nominal 109

Figure 4.38 Applied Throttle Deviation (η)
Nominal Geometry — Continuous Controller
Simultaneously Integrated Nominal 110

Figure 4.39 Discrete Range Gains — Set 1 111

Figure 4.40 Discrete Altitude Gains — Set 1 112

Figure 4.41 Discrete Velocity Gains — Set 1 113

Figure 4.42 Discrete Flight Path Angle Gains — Set 1 114

Figure 4.43 Closed Loop Mass Error
0.1 m Thrust Offset — Discrete Controller 115

Figure 4.44 Closed Loop Range Error
0.1 m Thrust Offset — Discrete Controller 116

Figure 4.45 Closed Loop Altitude Error
0.1 m Thrust Offset — Discrete Controller 117

Figure 4.46 Closed Loop Velocity Error
0.1 m Thrust Offset — Discrete Controller 118

Figure 4.47 Closed Loop Flight Path Angle Error
0.1 m Thrust Offset — Discrete Controller 119

Figure 4.48 Closed Loop Pitch Attitude Error
0.1 m Thrust Offset — Discrete Controller 120

Figure 4.49 Closed Loop Pitch Rate Error
0.1 m Thrust Offset — Discrete Controller 121

Figure 4.50 Applied Thrust Vector Control (β)
0.1 m Thrust Offset — Discrete Controller 122

Figure 4.51 Applied Pitch Attitude Deviation
0.1 m Thrust Offset — Discrete Controller 123

Figure 4.52 Applied Throttle Deviation (η)
0.1 m Thrust Offset — Discrete Controller 124

Figure 4.53 Complex Model Configuration Geometry 125

Figure 4.54 Open Loop Pitch Mode Eigenvalues — Real Part 126

Figure 4.55 Closed Loop Pitch Mode Eigenvalues — Real Part 127

Figure 4.56 β vs. θ Inner Loop Gain 128

Figure 4.57 β vs. q Inner Loop Gain 129

Figure 4.58 δ vs. θ Inner Loop Gain 130

Figure 4.59 δ vs. q Inner Loop Gain 131

Figure 4.60 Nominal Angle of Attack 132

Figure 4.61 Outer Loop Altitude Gains 133

Figure 4.62 Outer Loop Velocity Gains 134

Figure 4.63 Outer Loop Flight Path Angle Gains 135

Figure 4.64 Closed Loop Altitude Error at Nominal Time Points
Still Atmosphere 136

Figure 4.65 Closed Loop Velocity Error at Nominal Time Points
Still Atmosphere 137

Figure 4.66 Closed Loop Flight Path Angle Error at Nominal Time Points
Still Atmosphere 138

Figure 4.67 Closed Loop Altitude Error at 1 Second Interval
Still Atmosphere 139

Figure 4.68 Closed Loop Velocity Error at 1 Second Interval
Still Atmosphere 140

Figure 4.69 Closed Loop Flight Path Angle Error at 1 Second Interval
Still Atmosphere 141

Figure 4.70 Closed Loop Weight Error 142

Figure 4.71 Closed Loop Commanded Deviation In Throttle Ratio
Still Atmosphere 143

Figure 4.72 Closed Loop Commanded Deviation In Pitch Attitude
Still Atmosphere 144

Figure 4.73 Closed Loop Commanded Thrust Vector Angle
Still Atmosphere 145

Figure 4.74 Closed Loop Commanded Elevator Deflection Deviation
Still Atmosphere 146

Figure 4.75 Closed Loop Altitude Error at Nominal Time Points
Gust at 700 Seconds 147

Figure 4.76 Closed Loop Velocity Error at Nominal Time Points
Gust at 700 Seconds 148

Figure 4.77 Closed Loop Flight Path Angle Error at Nominal Time Points
Gust at 700 Seconds 149

1.0 INTRODUCTION

Over the past several years there has been a significant renewal of interest in the use of air-breathing hypersonic vehicles for low cost access to space. Much of this renewed interest is due to the astronomical cost of getting a payload into orbit, and the demonstrated unreliability of the currently available launch vehicles. The NASA Challenger disaster resulted in suspension of all NASA launch activities for an extended period of time, and the launch schedule will likely never resume the rates projected before that event. This suspension and subsequent launch schedule retrenchment has resulted in many payloads being delayed for years, and other payloads being canceled outright. The other major commercial launch service at the time of the Challenger accident was Ariannespace, the French consortium launch service company. Soon after the Challenger disaster, Ariannespace experienced several spectacular failures which resulted in a temporary suspension of launch activities. As a result of these U.S. and French launch failures there has been a huge increase in insurance premiums on their payloads. About this time the Soviet Union attempted to enter the commercial launch business but was stymied by American technology transfer restrictions. The Chinese also entered the launch business (for some reason without the technology transfer restrictions), but their launch location makes it impractical to launch into several desired orbits, and their vehicles are relatively unproven, which contributes to high insurance premiums on these launches also.

This tightening of launch capacity has come at a time when there are a number of major space projects in development, including the NASA Space Station Freedom and the U.S. Department of Defense Strategic Defense Initiative (SDI). Each of these projects will require unprecedented launch activities, with SDI possibly requiring the launch of hundreds of satellites into diverse orbits, and the Space Station requiring greatly expanded payload tonnage capability. The requirements of these two projects cannot be met by the shuttle

fleet alone. While SDI probably will not require significant increases in human space flight, the Space Station will necessarily require manned flights for assembly and operation. What is needed desperately at this time is an advanced low cost man-rated space launch system. With the burgeoning interest in the Earth environment, it has become a political as well economic necessity that the next major launch vehicle be sparing in the expenditure of natural resources, so a fully reusable vehicle is desirable. The technology necessary for the development of a single-stage-to-orbit air-breathing aerospace vehicle is now becoming available, and it would appear that there is a definite market for a vehicle of this type.

There are numerous operating constraints on a vehicle of this type, and several of these may be significantly altered through variations in the design of the vehicle. The current air-breathing engine of choice is the Scram-jet (*Supersonic Combustion* RAM-jet) which thrives on very high dynamic pressures and very high mass flow rates. This need of the engine for high pressures is in direct conflict with the airframe need for restricted dynamic pressures to minimize aerodynamic heating and direct loads on the structure. Design of a useful airbreathing launch vehicle will require a much more detailed, in-depth integration of components than the integration employed in the design of any vehicle built to date.

The successful development of a launch vehicle with this level of performance may also require simultaneous optimization of both vehicle and trajectory to minimize some mission cost function. Several researchers are presently developing procedures for this simultaneous optimization of vehicle and trajectory.[1–5] While this procedure may result in an outstanding vehicle for a particular mission, this design could result in an aircraft that can do only one thing well. It may be necessary in the initial design to reduce the specificity of the mission so as to reduce the penalty incurred from flying a non-standard mission. Each mission would then require its own specific trajectory, leading naturally to

the use of optimal control formulations for the trajectory problem. Unfortunately, because of the numerous constraints involved, optimal control problems of this type are extremely difficult to solve.

1.1 Suboptimal Trajectories

This emphasis on the optimality of trajectories, and the difficulty of actually solving the optimal trajectory problem, has resulted in much research into the solution of similar problems that are much easier to solve, and that provide a trajectory that is very close to the optimal. These are referred to, naturally, as sub-optimal trajectories, and they may be constructed in a number of different ways. One useful method for reducing the effort required in the determination of near optimal trajectories is *order reduction.* Order reduction involves the grouping of the states of the system into slow and fast states, with the fast states then behaving as controls for the slow states. An optimal (or possibly a sub-optimal) trajectory for the slow, reduced order system is then determined. A well developed theory that lends support to this order reduction approach is *singular perturbation theory* of ordinary differential equations. Singular perturbation theory's preëminent feature is reduction in order as a parameter goes to zero.[6]

While it is often possible to develop an optimal solution to a reduced order problem, a solution developed in this manner will not in general be optimal for the full order problem. The method of singular perturbations is attractive because it indicates that the solution to the reduced order problem, within certain limitations, will be close to the optimal full order solution; singular perturbation theory provides a process for approximating optimal *corrections* to the reduced order problem.

Even with order reduction the determination of optimal trajectories for a high performance aerospace vehicle is by no means trivial. A number of researchers are currently working on the development of techniques for efficiently solving this reduced order outer

problem.[7-10] We assume, for the purposes of this dissertation, that some method has been employed to develop a nominal point-mass trajectory for our model vehicle. We will restrict our interest to vertical plane motion over a non-rotating spherical Earth, and assume that the trajectory that is provided is at least close to the optimal for the point-mass model of the vehicle. It is assumed that the drag generated by the control surfaces in trimming the vehicle in pitch was considered during the development of this nominal trajectory.

It remains then to develop a control system for pitch so that the vehicle is stable in pitch and actually does follow the prescribed trajectory. For many lower performance aircraft, the penalty in operating cost or reduced mission capability is quite small even when the actual trajectory deviates significantly from the nominal sub-optimal trajectory. For the very high performance hypersonic vehicle this may not be true. Furthermore, very large penalties in range or fuel consumption may be incurred if the control system is not well designed, even when the nominal trajectory is followed closely.

The problem that appears with the hypersonic vehicle is that the drag penalty incurred in extending a control surface into a hypersonic flow can be severe. This drag penalty is to some extent accounted for with the drag addition from trim deflections, but does not account for the drag that is produced by control deflections in transients, or in stabilizing the vehicle. Because of the severe drag penalties imposed by elevator only control we seek to develop an additional, or possibly alternate, control—thrust vectoring. With thrust vectoring we seek to reduce the aerodynamic trim drag, and to reduce the fuel penalty for maneuvering control.

In constructing this alternate control we will formulate the *complete* vertical plane pitch control problem with multiple pitch control devices and then formally implement a singular perturbation order reduction, retaining the optimal character of the complete

problem in the reduced order trajectory problem and in the reduced order pitch dynamics problem. For this *new* pitch control to be optimal, it would be necessary to solve the reduced order trajectory problem while considering the multiple available controls, and then apply the reduced order pitch controller to this trajectory. What we will do is develop a reduced order pitch controller that in practice is capable of stabilizing the vehicle and of following almost any prescribed trajectory, but that will not in general be optimal (possibly not even near optimal) for that trajectory.

1.2 The Multi-Loop Control Formulation

This paper presents a dual-loop control system that seeks to minimize the penalties imposed on the performance index of the nominal problem due to the pitch control system, and that causes the vehicle to follow a prescribed nominal trajectory. This control system is formulated in two parts because of the order reduction process that we will impose. We will refer to the two loops as the inner loop, which involves the fast states *as* states, and the outer loop, which uses all or some of the fast states as controls for the slow states. The reason for these designations will be made clear later. The two control loops both attempt to track their respective states in the nominal trajectory. The inner loop control attempts to track the nominal fast states by applying physical controls to the vehicle; the outer loop control attempts to track the nominal slow states by applying perturbations to the nominal fast states. In this way all the states will be followed to reasonable accuracies.

Two formulations are presented in this paper. The first control system consists of continuous feedback formulations for both the inner and outer loops; the implementation of this control can be symbolically presented as

$$\mathbf{u}(t) = \mathbf{K}^{i}(t)\left(\mathbf{K}^{o}(t)\left(\mathbf{x}_n^{o}(t) - \mathbf{x}^{o}(t)\right) + \mathbf{x}^{i}(t) - \mathbf{x}_n^{i}(t)\right).$$

In our particular case

$$\mathbf{u} \in \Re^2, \quad \mathbf{x}^o \in \Re^4, \quad \mathbf{x}^i \in \Re^2, \quad \mathbf{K}^o \in \Re^{2\times 4}, \quad \mathbf{K}^i \in \Re^{2\times 2}.$$

$(*)_n$ indicates the nominal "correct" state, $(*)^i$ indicates the inner states or gains, and $(*)^o$ indicates the outer states or gains. It may be observed that this control system is actually a full state feedback single loop control system. This control system is presented as a dual loop control system because of the fundamentally different methods by which the inner and outer gains are determined.

The second control system formulation consists of a continuous inner feedback loop along with a discretely sampled and updated outer feedback loop. This can be symbolically represented as

$$\mathbf{u}(t) = \mathbf{K}^i(t)\left(\mathbf{K}^o(k)\left(\mathbf{x}_n^o(k) - \mathbf{x}^o(k)\right) + \mathbf{x}^i(t) - \mathbf{x}_n^i(t)\right),$$

where $(*)_n$ indicates the nominal "correct" state, $(*)^i$ indicates the inner states or gains, and $(*)^o$ indicates the outer states or gains; the inner nominal states, inner actual states, and the inner gains are continuous functions of time; the outer states and gains are sampled at discrete times k, $k+1$, $k+2, \ldots$ with a zero-order hold imposed on the sampled controls. This is implemented as a dual loop control system.

The inner loop gains for both implementations are identical, and are constructed from the singular perturbation order reduction of the full problem. The outer loop gains for the continuous/continuous controller result from employment of a continuous linear quadratic regulator, while the outer loop gains for the discrete/continuous controller result from employment of a discrete linear quadratic regulator.

1.2.1 Inner Loop Control

The basic requirements for a pitch control system are that the control system have sufficient resolution to trim the vehicle in steady flight, and in addition provide enough authority to flare the vehicle for landing and rotate the vehicle for takeoff. Several additional requirements for aircraft control systems are related to the flying qualities of the vehicle, and specify frequency response and damping ratios. We will not consider these requirements in this paper.

In this paper no effort is made to describe the response characteristics of actuators or actual control surfaces. It is assumed that the response of the actuators is fast relative to the pitch dynamics. (This assumption is one of the order reductions inherent in most optimal control formulations.)

The closed-loop control presented here is based entirely on the nominal model and trajectory and so is pre-computable. If a nominal trajectory is constructed in advance, then the pitch control gains can be determined before the flight and stored in the flight control computer for later use. These gains may be generated very efficiently. The application of these gains to the particular case presented in this paper results in a (usually) non-oscillatory, rapidly responding stable system that exhibits very large gain margins and a general insensitivity to certain system parameter variations. The result of this research is a control that very accurately follows a prescribed attitude/attitude-rate profile.

1.2.2 Outer Loop Control

Slight deviations in attitude and attitude rate away from the nominal values can cause significant deviations from the desired flight path.

If the nominal optimal trajectory is based on a point mass model, there will be an inherent error due to an assumption of no pitch moment of inertia. If other modeling errors or simplifying assumptions have been made, or if the atmosphere that the vehicle

flies in is not exactly as modeled, it is to be expected that an attitude only controller may allow very large position errors to develop over the course of the flight. As the inner-loop gain values, and therefore their optimality, are a function of the nominal trajectory, any major deviation from that trajectory will result in a reduction in the optimality of the inner loop gains and a subsequent cost penalty.

To alleviate this problem, an outer loop tracker that attempts to control errors in the trajectory position and velocity states was constructed. This feedback controller modifies the commanded pitch attitude that drives the inner loop control, and modifies the commanded throttle.

A number of different types of controllers could be constructed for this outer-loop control. Two controllers that were used in this project, and will be examined in more detail, are a Continuous Linear Quadratic Regulator operating on the linearized point mass system, and a Discrete Linear Quadratic Regulator operating on a discretized linearized point mass system.

The outer loop controllers presented here are optimal in that they minimize a quadratic cost function. This quadratic cost function is not directly related to the performance index presented in the original problem, so no claims of overall optimality will be made with regard to the outer-loop controllers. It is felt that the outer-loop controller has a relatively small effect on the overall optimality of the control system, and a controller that provides reasonable errors without excessive control excursions will be acceptable. An acceptable controller should also be efficient to implement and reasonably efficient to compute. Both of the described controllers meet these conditions.

2.0 INNER LOOP GUIDANCE ALGORITHM DEVELOPMENT

In this section we seek to construct a near optimal set of gains for an attitude controller using singular perturbation theory. To that end a brief overview of singular perturbation theory[11-13] for optimal control is presented.

2.1 Singular Perturbation Theory

The method of singular perturbations of ordinary differential equations is a method whose defining feature is a reduction in the order of the system as a parameter tends toward zero. This method then presents us with a reduced order system of differential equations with an increased number of controls and possibly several algebraic constraints relating the reduced states and the new controls (formerly states) and old controls. Lost in this reduced order solution is the ability to satisfy the initial and terminal conditions of the fast states. These initial and final conditions are recovered by modification of the reduced order system by a *boundary-layer* correction that is calculated by means of a system of differential equations also of lower order than the original system. In this paper we will be primarily interested in this boundary-layer correction; we will apply this boundary-layer correction continually along the trajectory as the fast states deviate from their prescribed nominal.

Consider the dynamic system

$$\dot{x} = f(x, y, u), \tag{2.1}$$

$$\epsilon \dot{y} = g(x, y, u). \tag{2.2}$$

With $\epsilon > 0$, x and y are state variables, u are control variables. Initial conditions are given by

$$x(t_0) = x_0,$$

$$y(t_0) = y_0.$$

Terminal constraints are imposed in the general form

$$\psi[x(t_f), t_f] = 0,$$

with a performance index given by

$$\Phi[x(t_f), t_f].$$

The Hamiltonian for this system is

$$\mathcal{H}(\lambda_x, \lambda_y, x, y, u) = \lambda_x^T f(x, y, u) + \lambda_y^T g(x, y, u),$$

from which we get the adjoint equations

$$\dot{\lambda}_x = -\frac{\partial \mathcal{H}}{\partial x},$$
$$\epsilon \dot{\lambda}_y = -\frac{\partial \mathcal{H}}{\partial y},$$

and the optimality condition

$$\frac{\partial \mathcal{H}}{\partial u} = 0 \qquad \left(\min_u \mathcal{H}\right).$$

From the transversality condition we get the boundary conditions

$$\lambda_x(t_f) = \frac{\partial \Phi}{\partial x} + \nu_x^T \frac{\partial \psi}{\partial x}$$
$$\lambda_y(t_f) = 0$$

where the ν's are determined so that the terminal constraint $\psi[x(t_f), t_f] = 0$ is satisfied. For $\epsilon = 0$, (the reduced problem) we have the dynamic system

$$\dot{x} = f(x, y, u),$$
$$0 = g(x, y, u).$$

so x are state variables, y and u are *linked* control variables, with $g(x, y, u) = 0$ algebraic constraints. To determine the solution y such that $g(x, y, u) = 0$, demands that $\left[\frac{\partial g}{\partial y}\right]$ have full rank.

This reduced problem has order of y fewer state variables than the full problem. This order reduction process also increases by the order of y the number of controls, but provides an equal number of algebraic constraints on those controls.

With $\epsilon = 0$, y and u are on equal footing as controls but are constrained by $g(x, y, u) = 0$, so we have as an optimality condition

$$\min_{y,u} \mathcal{H} \qquad \text{subject to} \qquad g(x, y, u) = 0,$$

which is amenable to treatment by classical Lagrange multiplier techniques. Assuming that y and u are thus determined, existence of an isolated local minimum requires that the second derivative of the Hamiltonian with respect to the controls be positive for all allowable variations. This requirement leads immediately to the strengthened Legendre-Clebsch condition,

$$\mathcal{H}_{uu} - g_u^T g_y^{-T} \mathcal{H}_{yu} - \mathcal{H}_{uy} g_y^{-1} g_u + g_u^T g_y^{-T} \mathcal{H}_{yy} g_y^{-1} g_u > 0$$

as a necessary condition for optimality of y and u.[14] If these conditions are met, then there is hope that the problem may be successfully attacked by singular perturbations.

This review provides a basic overview of singular perturbations as they may be applied to the solution of the reduced order outer problem. Solution of this problem is not in general a trivial problem, but should be relatively less difficult than the full problem because of the reduction in order. We will not solve this problem, but we implicitly assume that the nominal trajectory that is provided is in some sense close to the solution of this optimal control problem. This nominal trajectory is termed the outer solution.

Except for rare cases of good luck where $\dot{y}(t_0) = 0$, the value of y that is determined by $g(x(t_0), y(t_0), u(t_0)) = 0$ will not equal y_0, the prescribed initial conditions. We must therefore *fix* the missing initial condition. To the outer problem it appears that we require a jump discontinuity in y.

Consider the change of variables

$$\tau = \frac{t - t_0}{\epsilon}. \tag{2.3}$$

Then

$$\frac{d(*)}{d\tau} = \epsilon \frac{d(*)}{dt}, \tag{2.4}$$

and our state adjoint system becomes

$$\frac{dx}{d\tau} = \epsilon f(x, y, u), \tag{2.5}$$

$$\frac{d\lambda_x}{d\tau} = -\epsilon \frac{\partial \mathcal{H}}{\partial x}, \tag{2.6}$$

$$\frac{dy}{d\tau} = g(x, y, u),$$

$$\frac{d\lambda_y}{d\tau} = -\frac{\partial \mathcal{H}}{\partial y},$$

with the optimality condition

$$\frac{\partial \mathcal{H}}{\partial u} = 0.$$

Setting $\epsilon = 0$, (2.5) and (2.6) give

$$\frac{dx}{d\tau} = \frac{d\lambda_x}{d\tau} = 0,$$

or

$$x(\tau) = \overline{x}(t_0), \qquad \lambda_x(\tau) = \overline{\lambda}_x(t_0),$$

where the bar indicates the solution to the outer problem. The Hamiltonian of the inner problem is

$$\mathcal{H}(\lambda_x, \lambda_y, x, y, u) = \lambda_x^{0\,T} f(x^0, y, u) + \lambda_y^T g(x^0, y, u) \tag{2.7}$$

with $\lambda_x^0 = \overline{\lambda}_x(t_0)$ and $x^0 = \overline{x}(t_0)$ constant within the inner problem, and equal to the outer values at $t = t_0$.

(2.7) is the same Hamiltonian that would result from

$$\dot{y} = g(x^0, y, u)$$

with the performance index

$$J = \int_0^\infty {\lambda_x^0}^T f(x^0, y, u)\, d\tau\,.$$

The performance index of the inner problem is not arbitrary, but is inherited from the outer problem.

The boundary conditions for the inner problem are

$$y(0) = y_0, \qquad \text{(the prescribed initial conditions)}$$

and

$$\lim_{\tau\to\infty} y(\tau) = \overline{y}(t_0). \qquad \text{(the solution to } g(x^0, y, u) = 0).$$

Recall from (2.7) that $\mathcal{H}$ is a function of λ_x, λ_y, x, y, and u, where λ_x and x are constants from the outer solution, and u is determined by $\frac{\partial \mathcal{H}}{\partial u} = 0$ ($\min_u \mathcal{H}$), which leaves us with a coupled set of differential equations in λ_y and y. This coupled system of equations may be linearized about the stationary point $y = \overline{y}(t_0)$, $\lambda_y = \overline{\lambda}_y(t_0)$. This linearization gives

$$\begin{Bmatrix} \delta\dot{y} \\ \delta\dot{\lambda}_y \end{Bmatrix} = \begin{pmatrix} g_y & 0 \\ -\mathcal{H}_{yy} & -g_y^T \end{pmatrix} \begin{Bmatrix} \delta y \\ \delta\lambda_y \end{Bmatrix} + \begin{pmatrix} g_u \\ -\mathcal{H}_{yu} \end{pmatrix} \delta u\,. \tag{2.8}$$

We may determine δu by linearizing $\mathcal{H}_u = 0$, which gives

$$0 = \mathcal{H}_{uu}\delta u + \mathcal{H}_{uy}\delta y + \mathcal{H}_{u\lambda_y}\delta\lambda_y\,,$$

or, equivalently

$$\begin{aligned} \delta u &= -\mathcal{H}_{uu}^{-1}[\mathcal{H}_{uy}\delta y + \mathcal{H}_{u\lambda_y}\delta\lambda_y] \\ &= -\mathcal{H}_{uu}^{-1}[\mathcal{H}_{uy}\delta y + g_u^T\delta\lambda_y]\,. \end{aligned} \tag{2.9}$$

Substituting this representation of δu into (2.8) results in the closed loop coupled system

$$\begin{Bmatrix} \delta\dot{y} \\ \delta\dot{\lambda}_y \end{Bmatrix} = \begin{pmatrix} g_y - g_u\mathcal{H}_{uu}^{-1}\mathcal{H}_{uy} & -g_u\mathcal{H}_{uu}^{-1}g_u^T \\ -\mathcal{H}_{yy} + \mathcal{H}_{yu}\mathcal{H}_{uu}^{-1}\mathcal{H}_{uy} & -g_y^T + \mathcal{H}_{yu}\mathcal{H}_{uu}^{-1}g_u^T \end{pmatrix} \begin{Bmatrix} \delta y \\ \delta\lambda_y \end{Bmatrix}. \tag{2.10}$$

This system is of Hamiltonian form and can be written

$$\begin{Bmatrix} \delta\dot{y} \\ \delta\dot{\lambda}_y \end{Bmatrix} = \begin{pmatrix} \mathbf{A} & \mathbf{N} \\ \mathbf{K} & -\mathbf{A}^T \end{pmatrix} \begin{Bmatrix} \delta y \\ \delta\lambda_y \end{Bmatrix},$$

where it should be observed that **N** and **K** are symmetric.

Hamiltonian matrices have the special property that eigenvalues appear in positive-negative pairs as well as complex conjugate pairs, so an equilibrium point $y = \overline{y}(t_0)$, $\lambda_y = \overline{\lambda}_y(t_0)$ is not a stable equilibrium. We may show this symmetry as follows. Consider the orthogonal matrix **J** defined by

$$\mathbf{J} = \begin{pmatrix} \mathbf{0} & -\mathbf{I} \\ \mathbf{I} & \mathbf{0} \end{pmatrix}.$$

Let

$$\mathbf{H} = \begin{pmatrix} \mathbf{A} & \mathbf{N} \\ \mathbf{K} & -\mathbf{A}^T \end{pmatrix},$$

then

$$\mathbf{J}\mathbf{H}\mathbf{J}^T = -\mathbf{H}^T.$$

J is orthogonal, so **H** is similar to -**H**, and every eigenvalue of **H** is also an eigenvalue of -**H**. Therefore the eigenvalues must either be purely imaginary, or must appear in $\pm$ pairs.

The initial values of the adjoint variables λ_y must be chosen to suppress the unstable modes at the equilibrium point. The unstable modes may be suppressed if it is possible to choose initial values of λ_y such that the state-adjoint vector lies in the stable subspace of the Hamiltonian system. To find this space we construct the eigenvectors associated with the stable eigenvalues.

Consider the eigenvalue, eigenvector decomposition

$$\mathbf{H} = \begin{pmatrix} \mathbf{T}_{11} & \mathbf{T}_{12} \\ \mathbf{T}_{21} & \mathbf{T}_{22} \end{pmatrix} \begin{pmatrix} -\mathbf{\Lambda} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Lambda} \end{pmatrix} \begin{pmatrix} \mathbf{T}_{11} & \mathbf{T}_{12} \\ \mathbf{T}_{21} & \mathbf{T}_{22} \end{pmatrix}^{-1}$$

with real part of $-\mathbf{\Lambda} < 0$. We seek initial values of $\delta\lambda_{\mathbf{y}}$ such that

$$\begin{Bmatrix} \delta\mathbf{y} \\ \delta\lambda_{\mathbf{y}} \end{Bmatrix} = \begin{pmatrix} \mathbf{T}_{11} \\ \mathbf{T}_{21} \end{pmatrix} \mathbf{c}$$

is satisfied for some $\mathbf{c}$.

We have $$\delta \mathbf{y} = \mathbf{T}_{11}\mathbf{c},$$

or, equivalently $$\mathbf{c} = \mathbf{T}_{11}^{-1}\delta \mathbf{y},$$

Then

$$\delta \lambda_{\mathbf{y}} = \mathbf{T}_{21}\mathbf{T}_{11}^{-1}\delta \mathbf{y}. \tag{2.11}$$

Our final goal is a stable feedback law for δu in terms of δy. Substituting (2.11) into (2.9) finally results in

$$\delta u = -\mathcal{H}_{uu}^{-1}[\mathcal{H}_{uy} + g_u^T \mathbf{T}_{21}\mathbf{T}_{11}^{-1}]\delta y. \tag{2.12}$$

This expression provides an explicit perturbation feedback law that suppresses the unstable modes at the stationary point and minimizes the performance index inherited from the outer problem.

It should be noted that the derivatives indicated in (2.12) are all evaluated on the nominal trajectory, as are all the terms in (2.10). This fact then provides us with a linear feedback law that is precomputable.

We will apply this inner loop gain generation algorithm to our current problem.

2.2 Generic Aerospace Vehicle Pitch Control

This analysis begins with a generic aerospace vehicle acted on by the generic forces of Lift L, Drag D, Weight mg, Thrust T, and aerodynamic pitching Moment M. The general geometry that will be considered is shown in Figure 2.1. The vehicle has a mass m, acted on by an inverse square gravitational field g. We define a body reference axis that is fixed in the body and passes through the center of mass. The local vertical passes from the center of the earth through the center of mass of the vehicle. The local horizontal lies in the plane normal to the local vertical. The angle from the local horizontal to the

body reference axis is θ. The angle from the local horizontal and the velocity vector V is γ.

The motion considered here is vertical plane motion over a spherical non-rotating earth. For our general development we will consider a dynamic system with seven states and three controls. The states are mass m, downrange distance x, altitude h, velocity V, flight path angle γ, pitch attitude θ, and pitch rate q. The controls that we model are thrust vector angle β, elevator deflection angle δ, and throttle η. The equations of motion with coefficient dependencies are

$$
\begin{aligned}
\dot{m} &= -\frac{T(h,V,\eta)}{I_{sp}(h,V,\eta)} \qquad (I_{sp} \text{ in consistent units}) \\
\dot{x} &= V\cos\gamma \\
\dot{h} &= V\sin\gamma \\
\dot{V} &= \frac{T(h,V,\eta)\cos(\theta-\gamma+\beta) - D(h,V,\gamma,\theta,\delta)}{m} - g(h)\sin\gamma \\
\dot{\gamma} &= \frac{T(h,V,\eta)\sin(\theta-\gamma+\beta) + L(h,V,\gamma,\theta,\delta)}{mV} - \frac{g(h)\cos\gamma}{V} + \frac{V\cos\gamma}{r_e+h} \\
\epsilon\dot{\theta} &= q + \frac{V\cos\gamma}{r_e+h} \\
\epsilon\dot{q} &= \frac{T(h,V,\eta)}{I_{yy}(m)}(z\cos\beta - \ell\sin\beta) + \frac{M(h,V,\gamma,\theta,q,\delta)}{I_{yy}(m)}
\end{aligned}
$$

The term $\frac{V\cos\gamma}{r_e+h}$ is called centrifugal relief and is equal in magnitude to the rotation rate of the vehicle-centered coördinate system. Note that we neglect pitch rate effects on D and L. We make the implicit assumption that the pitch dynamic states θ and q are *fast* compared to the point mass states, with the ϵ parameter in the pitch dynamics equations motivating a time scale separation. With $\epsilon = 0$ the pitch dynamics are *instantaneous* and we have a point mass model with varying mass. This type of order-reduction is quite common in engineering.[6]

We construct the variational Hamiltonian for the system in two parts:

$$\mathcal{H}^0 = \lambda_m\Big(-\frac{T}{I_{sp}}\Big) + \lambda_x\Big(V\cos\gamma\Big) + \lambda_h\Big(V\sin\gamma\Big) + \lambda_V\Big(\frac{T\cos(\theta-\gamma+\beta) - D}{m} - g\sin\gamma\Big)$$
$$+ \lambda_\gamma\Big(\frac{T\sin(\theta-\gamma+\beta) + L}{mV} - \frac{g\cos\gamma}{V} + \frac{V\cos\gamma}{r_e+h}\Big)$$
$$\mathcal{H}^1 = \lambda_\theta\Big(q + \frac{V\cos\gamma}{r_e+h}\Big) + \lambda_q\Big(\frac{T}{I_{yy}}(z\cos\beta - \ell\sin\beta) + \frac{M}{I_{yy}}\Big)$$

The variational Hamiltonian for the full problem is given by $\mathcal{H} = \mathcal{H}^0 + \mathcal{H}^1$.

The optimality conditions for the outer point-mass problem are given by

$$g(x, y, u) = 0$$

$$\mathcal{H}_u = 0$$

$$\mathcal{H}_y = 0$$

This is seven nonlinear equations in the seven unknowns, η, δ, β, θ, q, λ_θ, and λ_q. As we are not actually interested at this time with solving the outer problem, we will simplify this system of equations by certain assumptions within the following procedure.

At this point we construct the adjoint differential equations. Since we are assuming that the point mass optimal trajectory is already known, we will ignore the adjoint differential equations associated with the point mass system and only deal with the pitch dynamic adjoint equations, which are

$$\epsilon\dot{\lambda}_\theta = -\frac{\partial\mathcal{H}}{\partial\theta} = \frac{\lambda_V}{m}\left(T\sin(\theta-\gamma+\beta) + \frac{\partial D}{\partial\theta}\right) - \frac{\lambda_\gamma}{mV}\left(T\cos(\theta-\gamma+\beta) + \frac{\partial L}{\partial\theta}\right) - \frac{\lambda_q}{I_{yy}}\frac{\partial M}{\partial\theta},$$
$$\epsilon\dot{\lambda}_q = -\frac{\partial\mathcal{H}}{\partial q} = -\lambda_\theta - \frac{\lambda_q}{I_{yy}}\frac{\partial M}{\partial q}.$$

If we set $\epsilon = 0$ and analyze first the zeroth-order approximation of the outer system, we see that the last two state equations and the two fast adjoint equations reduce to algebraic constraints, providing us with the four algebraic equations

$$0 = q^0 + \frac{V^0\cos\gamma^0}{r_e+h^0}; \tag{2.13}$$

$$0 = T\left(z\cos\beta^0 - \ell\sin\beta^0\right) + M^0; \tag{2.14}$$

$$0 = \frac{\lambda_V^0}{m}\left(T\sin(\theta^0-\gamma^0+\beta^0)+\frac{\partial D}{\partial\theta}\right)-\frac{\lambda_\gamma^0}{mV^0}\left(T\cos(\theta^0-\gamma^0+\beta^0)+\frac{\partial L}{\partial\theta}\right)-\frac{\lambda_q^0}{I_{yy}}\frac{\partial M}{\partial\theta}; \quad (2.15)$$

$$0 = -\lambda_\theta^0-\frac{\lambda_q^0}{I_{yy}}\frac{\partial M}{\partial q}. \quad (2.16)$$

The superscript ^{0} indicates that these are the nominal values from the outer problem and are held constant within the inner problem. The derivatives indicated in the preceding equations are evaluated on the nominal trajectory. As we are not constructing a solution to the outer problem, we will *unlink* the new and old controls. By this we are making the implicit assumption that the direct effects on the point-mass trajectory from the old controls β and δ are negligible compared to the direct effect on the point-mass trajectory of the new control θ. This assumption is probably reasonable . The optimality condition for the outer trajectory, using θ as control, and ignoring all others, is

$$\frac{\partial\mathcal{H}^0}{\partial\theta} = 0 = -\frac{\lambda_V^0}{m}\left(T\sin(\theta^0-\gamma^0+\beta^0)+\frac{\partial D}{\partial\theta}\right)+\frac{\lambda_\gamma^0}{mV^0}\left(T\cos(\theta^0-\gamma^0+\beta^0)+\frac{\partial L}{\partial\theta}\right),$$

$$0 = -\lambda_V^0\left(T\sin(\theta^0-\gamma^0+\beta^0)+\frac{\partial D}{\partial\theta}\right)+\frac{\lambda_\gamma^0}{V^0}\left(T\cos(\theta^0-\gamma^0+\beta^0)+\frac{\partial L}{\partial\theta}\right). \quad (2.17)$$

From equation (2.13) $\quad q^0 = -\dfrac{V\cos\gamma}{r_e+h}$;

from equation (2.15) and (2.17) $\quad \lambda_q^0 = 0$;

then equation (2.16) gives $\quad \lambda_\theta^0 = 0$.

Rearranging (2.17) gives $\quad \dfrac{\lambda_V^0}{\lambda_\gamma^0} = \dfrac{T\cos(\theta^0-\gamma^0+\beta^0)+\frac{\partial L}{\partial\theta}}{V^0\left(T\sin(\theta^0-\gamma^0+\beta^0)+\frac{\partial D}{\partial\theta}\right)}$,

so we can write

$$\lambda_V^0 = K^0\left(T\cos(\theta^0-\gamma^0+\beta^0)+\frac{\partial L}{\partial\theta}\right), \quad (2.18)$$

$$\lambda_\gamma^0 = K^0V^0\left(T\sin(\theta^0-\gamma^0+\beta^0)+\frac{\partial D}{\partial\theta}\right), \quad (2.19)$$

where K^0 is a proportionality *constant*, along an optimal trajectory. While the nominal trajectory that we are attempting to follow is suboptimal, we believe that this approximation from the optimality condition will work acceptably well.

A solution to the outer (point mass) problem is assumed to have been constructed by some method. It is also assumed that this solution has been developed using θ and η as controls. We assume that we have available aerodynamic devices and thrust vectoring for pitch attitude and rate control, designated by δ and β respectively. Were we actually constructing a true singular perturbation solution to the outer problem, the outer problem optimality conditions would be satisfied along the trajectory, with the resulting β and δ considered as the nominals β^0 and δ^0. The actual determination of the outer *nominal* trajectory and its associated control is not of interest to us here, and we will use whatever time history of β and δ are supplied with the nominal trajectory, even though they may be far from optimal. In our particular case (2.14) has been satisfied with $\beta = 0$ for the trajectory (that is, thrust vectoring was not considered).

We look now at the inner (fast dynamics) problem. Given the assumed separation in time scales, we will consider the outer states and adjoints to be constant over a small time interval in the inner problem. For notational convenience define $\xi^0 = \theta^0 - \gamma^0 + \beta^0$, and also $\xi = \theta - \gamma + \beta$. Note that this is the angle from the thrust line to the velocity vector. The optimality conditions for the inner problem are

$$\mathcal{H}_\beta = 0 = -\frac{\lambda_V^0}{m} T \sin\xi + \frac{\lambda_\gamma^0}{mV} T \cos\xi - \frac{\lambda_q}{I_{yy}} T(z \sin\beta + \ell \cos\beta), \tag{2.20}$$

and

$$\mathcal{H}_\delta = 0 = -\frac{\lambda_V^0}{m}\frac{\partial D}{\partial \delta} + \frac{\lambda_\gamma^0}{mV}\frac{\partial L}{\partial \delta} + \frac{\lambda_q}{I_{yy}}\frac{\partial M}{\partial \delta}. \tag{2.21}$$

First we will manipulate (2.20). Replace λ_V^0 and λ_γ^0 with the relationships of (2.18) and

(2.19), introduce $I_{yy} = m\rho^2$, and rearrange:

$$\begin{aligned}
\mathcal{H}_\beta = 0 &= -K^0\left(T\cos\xi^0 + \left.\frac{\partial L}{\partial\theta}\right|_0\right)T\sin\xi + K^0\left(T\sin\xi^0 + \left.\frac{\partial D}{\partial\theta}\right|_0\right)T\cos\xi \\
&\quad - \lambda_q\frac{1}{\rho^2}T(z\sin\beta + \ell\cos\beta) \\
&= K^0\left\{T\left(\sin\xi^0\cos\xi - \cos\xi^0\sin\xi\right) + \left.\frac{\partial D}{\partial\theta}\right|_0\cos\xi - \left.\frac{\partial L}{\partial\theta}\right|_0\sin\xi\right\} \\
&\quad - \lambda_q\frac{1}{\rho^2}(z\sin\beta + \ell\cos\beta) \\
&= K^0\left\{T\sin(\xi^0-\xi) + \left.\frac{\partial D}{\partial\theta}\right|_0\cos\xi - \left.\frac{\partial L}{\partial\theta}\right|_0\sin\xi\right\} - \lambda_q\frac{1}{\rho^2}(z\sin\beta + \ell\cos\beta).
\end{aligned}$$

We now linearize this equation about $\theta = \theta^0$, $\beta = \beta^0$, $\delta = \delta^0$, and $\lambda_q = 0$:

$$\begin{aligned}
0 = K^0\Big\{ &- T(\Delta\theta + \Delta\beta) - \left.\frac{\partial D}{\partial\theta}\right|_0\sin\xi^0(\Delta\theta + \Delta\beta) - \left.\frac{\partial L}{\partial\theta}\right|_0\cos\xi^0(\Delta\theta + \Delta\beta)\Big\} \\
&\qquad - \Delta\lambda_q\frac{1}{\rho^2}(z\sin\beta^0 + \ell\cos\beta^0).
\end{aligned}$$

This equation may be rearranged, noting that all non-perturbation quantities are evaluated at the nominal, to get

$$0 = -K^0\left\{T + \frac{\partial D}{\partial\theta}\sin\xi^0 + \frac{\partial L}{\partial\theta}\cos\xi^0\right\}(\Delta\theta + \Delta\beta) - \frac{1}{\rho^2}(z\sin\beta^0 + \ell\cos\beta^0)\Delta\lambda_q.$$

This last equation may finally be solved for $\Delta\beta$ to give the feedback law

$$\Delta\beta = -\Delta\theta - \frac{z\sin\beta^0 + \ell\cos\beta^0}{K^0\rho^2(T + \frac{\partial D}{\partial\theta}\sin\xi^0 + \frac{\partial L}{\partial\theta}\cos\xi^0)}\Delta\lambda_q. \tag{2.22}$$

Turning our attention now to (2.21) we again linearize about $\theta = \theta^0$, $\beta = \beta^0$, $\delta = \delta^0$, and $\lambda_q = 0$, and substitute (2.18) and (2.19), resulting in

$$\begin{aligned}
0 &= -\lambda_V^0\left(\frac{\partial^2 D}{\partial\delta\partial\theta}\Delta\theta + \frac{\partial^2 D}{\partial\delta^2}\Delta\delta\right) + \lambda_\gamma^0\frac{1}{V}\left(\frac{\partial^2 L}{\partial\delta\partial\theta}\Delta\theta + \frac{\partial^2 L}{\partial\delta^2}\Delta\delta\right) + \frac{1}{\rho^2}\frac{\partial M}{\partial\delta}\Delta\lambda_q \\
&= -K^0\left(T\cos\xi^0 + \left.\frac{\partial L}{\partial\theta}\right|_0\right)\left(\frac{\partial^2 D}{\partial\delta\partial\theta}\Delta\theta + \frac{\partial^2 D}{\partial\delta^2}\Delta\delta\right) \\
&\quad + K^0\left(T\sin\xi^0 + \left.\frac{\partial D}{\partial\theta}\right|_0\right)\left(\frac{\partial^2 L}{\partial\delta\partial\theta}\Delta\theta + \frac{\partial^2 L}{\partial\delta^2}\Delta\delta\right) + \frac{1}{\rho^2}\frac{\partial M}{\partial\delta}\Delta\lambda_q \\
&= -K^0\left\{\left(T\cos\xi^0 + \left.\frac{\partial L}{\partial\theta}\right|_0\right)\frac{\partial^2 D}{\partial\delta\partial\theta} - \left(T\sin\xi^0 + \left.\frac{\partial D}{\partial\theta}\right|_0\right)\frac{\partial^2 L}{\partial\delta\partial\theta}\right\}\Delta\theta \\
&\quad - K^0\left\{\left(T\cos\xi^0 + \left.\frac{\partial L}{\partial\theta}\right|_0\right)\frac{\partial^2 D}{\partial\delta^2} - \left(T\sin\xi^0 + \left.\frac{\partial D}{\partial\theta}\right|_0\right)\frac{\partial^2 L}{\partial\delta^2}\right\}\Delta\delta + \frac{1}{\rho^2}\frac{\partial M}{\partial\delta}\Delta\lambda_q,
\end{aligned}$$

which rearranged finally gives

$$\Delta\delta = -\frac{\left(T\cos\xi^0 + \frac{\partial L}{\partial\theta}\big|_0\right)\frac{\partial^2 D}{\partial\delta\partial\theta} - \left(T\sin\xi^0 + \frac{\partial D}{\partial\theta}\big|_0\right)\frac{\partial^2 L}{\partial\delta\partial\theta}}{\left(T\cos\xi^0 + \frac{\partial L}{\partial\theta}\big|_0\right)\frac{\partial^2 D}{\partial\delta^2} - \left(T\sin\xi^0 + \frac{\partial D}{\partial\theta}\big|_0\right)\frac{\partial^2 L}{\partial\delta^2}}\Delta\theta$$
$$+ \frac{\frac{\partial M}{\partial\delta}}{K^0\rho^2\left\{\left(T\cos\xi^0 + \frac{\partial L}{\partial\theta}\big|_0\right)\frac{\partial^2 D}{\partial\delta^2} - \left(T\sin\xi^0 + \frac{\partial D}{\partial\theta}\big|_0\right)\frac{\partial^2 L}{\partial\delta^2}\right\}}\Delta\lambda_q\,. \qquad (2.23)$$

(2.22) and (2.23) together correspond to the vector equation (2.9). The coefficients are evaluated along the nominal trajectory, so in effect all the derivatives are nominal and we can dispense with the $|_0$ designation. (2.22) and (2.23) provide us with a feedback law for the pitch controls $\Delta\beta$ and $\Delta\delta$ in terms of $\Delta\theta$ and $\Delta\lambda_q$. We therefore seek a stable closed form representation of $\Delta\lambda_q$ in terms of $\Delta\theta$ and Δq.

Let us use the time stretching transformation of (2.3), $\tau = (t - t^0)/\epsilon$, from which we get $d(*)/d\tau = \epsilon\, d(*)/dt$.

Now we construct the state-adjoint system of differential equations for the inner problem using this change of variables, designating $d(*)/d\tau$ as $(*)'$:

$$\theta' = q + \frac{V\cos\gamma}{r_e + h},$$
$$q' = \frac{1}{I_{yy}}\big(T(z\cos\beta - \ell\sin\beta) + M\big),$$
$$\lambda_\theta' = -\frac{\partial\mathcal{H}}{\partial\theta} = \lambda_V\frac{1}{m}\left(T\sin\xi + \frac{\partial D}{\partial\theta}\right) - \lambda_\gamma\frac{1}{mV}\left(T\cos\xi + \frac{\partial L}{\partial\theta}\right) - \lambda_q\frac{1}{I_{yy}}\frac{\partial M}{\partial\theta},$$
$$\lambda_q' = -\frac{\partial\mathcal{H}}{\partial q} = -\lambda_\theta - \lambda_q\frac{1}{I_{yy}}\frac{\partial M}{\partial q}.$$

The outer states and adjoints are held constant in this system. Manipulating the λ_θ' equation, taking care to separate the expansion of the terms from the outer variables and

those from the inner, we get

$$\begin{aligned}
\lambda'_\theta =& K^0\left(T\cos\xi^0 + \frac{\partial L}{\partial\theta}\bigg|_0\right)\frac{1}{m}\left(T\sin\xi + \frac{\partial D}{\partial\theta}\right)\\
&- K^0\left(T\sin\xi^0 + \frac{\partial D}{\partial\theta}\bigg|_0\right)\frac{1}{m}\left(T\cos\xi + \frac{\partial L}{\partial\theta}\right) - \lambda_q\frac{1}{I_{yy}}\frac{\partial M}{\partial\theta}\\
=& \frac{K^0}{m}\left\{T^2\left[\sin\xi\cos\xi^0 - \cos\xi\sin\xi^0\right]\right.\\
&+ T\left[\frac{\partial D}{\partial\theta}\cos\xi^0 - \frac{\partial D}{\partial\theta}\bigg|_0\cos\xi - \frac{\partial L}{\partial\theta}\sin\xi^0 + \frac{\partial L}{\partial\theta}\bigg|_0\sin\xi\right]\\
&\left.+ \left[\frac{\partial L}{\partial\theta}\bigg|_0\frac{\partial D}{\partial\theta} - \frac{\partial D}{\partial\theta}\bigg|_0\frac{\partial L}{\partial\theta}\right]\right\} - \lambda_q\frac{1}{I_{yy}}\frac{\partial M}{\partial\theta}\\
=& \frac{K^0}{m}\left\{T^2\left[\sin(\xi-\xi^0)\right] + T\left[\,*\,\right] + \left[\,*\,\right]\right\} - \lambda_q\frac{1}{I_{yy}}\frac{\partial M}{\partial\theta}\,.
\end{aligned}$$

The system of equations can now be written as

$$\begin{aligned}
\theta' &= q + \frac{V\cos\gamma}{r_e + h},\\
q' &= \frac{1}{I_{yy}}\left(T(z\cos\beta - \ell\sin\beta) + M\right),\\
\lambda'_\theta &= \frac{K^0}{m}\left\{T^2\left[\sin(\xi-\xi^0)\right]\right.\\
&\quad + T\left[\frac{\partial D}{\partial\theta}\cos\xi^0 - \frac{\partial D}{\partial\theta}\bigg|_0\cos\xi - \frac{\partial L}{\partial\theta}\sin\xi^0 + \frac{\partial L}{\partial\theta}\bigg|_0\sin\xi\right]\\
&\quad \left.+ \left[\frac{\partial L}{\partial\theta}\bigg|_0\frac{\partial D}{\partial\theta} - \frac{\partial D}{\partial\theta}\bigg|_0\frac{\partial L}{\partial\theta}\right]\right\} - \lambda_q\frac{1}{I_{yy}}\frac{\partial M}{\partial\theta},\\
\lambda'_q &= -\lambda_\theta - \lambda_q\frac{1}{I_{yy}}\frac{\partial M}{\partial q}.
\end{aligned} \tag{2.24}$$

These equations are appropriate for an optimal control problem with system dynamics given by θ' and q' and the Lagrange performance index $\mathcal{H}^0$. The Lagrange performance index can be given by

$$\mathcal{L} = \lambda_V^0\left(\frac{T^0\cos(\theta-\gamma^0+\beta) - D}{m^0}\right) + \lambda_\gamma^0\left(\frac{T^0\sin(\theta-\gamma^0+\beta) + L}{m^0V^0}\right).$$

Substituting the values of λ_V^0 and λ_γ^0 from (2.18) and (2.19) gives the Lagrange performance index

$$\mathcal{L} = \frac{K^0}{m^0}\left[\left(T\cos\xi^0 + \frac{\partial L}{\partial\theta}\right)(T^0\cos\xi - D) + \left(T\sin\xi^0 + \frac{\partial D}{\partial\theta}\right)(T^0\sin\xi + L)\right]$$

We can interpret this cost function as a measure of the alignment of the aerodynamic and thrust force vector with a vector representing the sensitivity of this vector to changes in θ. The control system then attempts to minimize this misalignment. Even though the original optimization problem has not been solved exactly (we are dealing with a sub-optimal trajectory), we feel that this is still the *correct* cost function to minimize.

We now linearize (2.24) about $\theta = \theta^0$, $q = q^0$, $\lambda_\theta = \lambda_q = 0$, $\beta = \beta^0$, $\delta = \delta^0$, noting that all the derivatives are evaluated along the nominal trajectory. Care must be taken in the λ'_θ equation to separate those terms due to the outer adjoint variables from the other terms. The outer states and adjoints are held constant within the inner loop, so variations of their constituent terms are all identically zero.

$$
\begin{aligned}
\Delta\theta' &= \Delta q, \\
\Delta q' &= \frac{1}{I_{yy}}\left[-T\left(z\sin\beta + \ell\cos\beta\right)\Delta\beta + \frac{\partial M}{\partial\theta}\Delta\theta + \frac{\partial M}{\partial q}\Delta q + \frac{\partial M}{\partial\delta}\Delta\delta\right], \\
\Delta\lambda'_\theta &= \frac{K^0}{m}\Bigg\{T^2\Big[\Delta\theta + \Delta\beta\Big] \\
&\qquad + T\Bigg[\cos\xi^0\left(\frac{\partial^2 D}{\partial\theta^2}\Delta\theta + \frac{\partial^2 D}{\partial\delta\partial\theta}\Delta\delta\right) - \sin\xi^0\left(\frac{\partial^2 L}{\partial\theta^2}\Delta\theta + \frac{\partial^2 L}{\partial\delta\partial\theta}\Delta\delta\right) \\
&\qquad\quad + \frac{\partial L}{\partial\theta}\cos\xi^0(\Delta\theta + \Delta\beta) + \frac{\partial D}{\partial\theta}\sin\xi^0(\Delta\theta + \Delta\beta)\Bigg] \\
&\qquad + \frac{\partial L}{\partial\theta}\left(\frac{\partial^2 D}{\partial\theta^2}\Delta\theta + \frac{\partial^2 D}{\partial\delta\partial\theta}\Delta\delta\right) - \frac{\partial D}{\partial\theta}\left(\frac{\partial^2 L}{\partial\theta^2}\Delta\theta + \frac{\partial^2 L}{\partial\delta\partial\theta}\Delta\delta\right)\Bigg\} - \frac{1}{I_{yy}}\frac{\partial M}{\partial\theta}\Delta\lambda_q, \\
\Delta\lambda'_q &= -\Delta\lambda_\theta - \frac{1}{I_{yy}}\frac{\partial M}{\partial q}\Delta\lambda_q.
\end{aligned}
$$

Written in matrix notation, we have

$$
\begin{pmatrix} \Delta\theta' \\ \Delta q' \\ \Delta\lambda'_\theta \\ \Delta\lambda'_q \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ \frac{1}{I_{yy}}\frac{\partial M}{\partial\theta} & \frac{1}{I_{yy}}\frac{\partial M}{\partial q} & 0 & 0 \\ A_{31} & 0 & 0 & -\frac{1}{I_{yy}}\frac{\partial M}{\partial\theta} \\ 0 & 0 & -1 & -\frac{1}{I_{yy}}\frac{\partial M}{\partial q} \end{pmatrix} \begin{pmatrix} \Delta\theta \\ \Delta q \\ \Delta\lambda_\theta \\ \Delta\lambda_q \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ B_{21} & \frac{1}{I_{yy}}\frac{\partial M}{\partial\delta} \\ B_{31} & B_{32} \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \Delta\beta \\ \Delta\delta \end{pmatrix},
$$

where

$$A_{31} = \frac{K^0}{m}\left\{T\left[T + \cos\xi^0\left(\frac{\partial^2 D}{\partial\theta^2} + \frac{\partial L}{\partial\theta}\right) - \sin\xi^0\left(\frac{\partial^2 L}{\partial\theta^2} - \frac{\partial D}{\partial\theta}\right)\right] + \frac{\partial L}{\partial\theta}\frac{\partial^2 D}{\partial\theta^2} - \frac{\partial D}{\partial\theta}\frac{\partial^2 L}{\partial\theta^2}\right\},$$

$$B_{21} = -\frac{T}{I_{yy}}\left(z\sin\beta + \ell\cos\beta\right),$$

$$B_{31} = \frac{K^0 T}{m}\left(T + \cos\xi^0\frac{\partial L}{\partial\theta} + \sin\xi^0\frac{\partial D}{\partial\theta}\right),$$

$$B_{32} = \frac{K^0}{m}\left\{T\left(\cos\xi^0\frac{\partial^2 D}{\partial\delta\partial\theta} - \sin\xi^0\frac{\partial^2 L}{\partial\delta\partial\theta}\right) + \frac{\partial L}{\partial\theta}\frac{\partial^2 D}{\partial\delta\partial\theta} - \frac{\partial D}{\partial\theta}\frac{\partial^2 L}{\partial\delta\partial\theta}\right\}.$$

This last set of equations corresponds to (2.8) of section 2.1. Substituting in the values of $\Delta\beta$ and $\Delta\delta$ from equations (2.22) and (2.23) produces the *closed-loop* dynamic system

$$\begin{pmatrix} \Delta\theta' \\ \Delta q' \\ \Delta\lambda'_\theta \\ \Delta\lambda'_q \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ A'_{21} & \frac{1}{I_{yy}}\frac{\partial M}{\partial q} & 0 & A'_{24} \\ A'_{31} & 0 & 0 & -A'_{21} \\ 0 & 0 & -1 & -\frac{1}{I_{yy}}\frac{\partial M}{\partial q} \end{pmatrix} \begin{pmatrix} \Delta\theta \\ \Delta q \\ \Delta\lambda_\theta \\ \Delta\lambda_q \end{pmatrix}, \tag{2.25}$$

where

$$A'_{21} = \frac{1}{I_{yy}}\left\{\frac{\partial M}{\partial\theta} + T\left(z\sin\beta^0 + \ell\cos\beta^0\right) - \frac{\left(T\cos\xi^0 + \frac{\partial L}{\partial\theta}\right)\frac{\partial^2 D}{\partial\delta\partial\theta} - \left(T\sin\xi^0 + \frac{\partial D}{\partial\theta}\right)\frac{\partial^2 L}{\partial\delta\partial\theta}}{\left(T\cos\xi^0 + \frac{\partial L}{\partial\theta}\right)\frac{\partial^2 D}{\partial\delta^2} - \left(T\sin\xi^0 + \frac{\partial D}{\partial\theta}\right)\frac{\partial^2 L}{\partial\delta^2}}\frac{\partial M}{\partial\delta}\right\},$$

$$A'_{24} = \frac{1}{I_{yy}K^0\rho^2}\left\{\frac{T\left(z\sin\beta^0 + \ell\cos\beta^0\right)^2}{\left(T + \frac{\partial D}{\partial\theta}\sin\xi^0 + \frac{\partial L}{\partial\theta}\cos\xi^0\right)} + \frac{\frac{\partial M}{\partial\delta}^2}{\left(T\cos\xi^0 + \frac{\partial L}{\partial\theta}\right)\frac{\partial^2 D}{\partial\delta^2} - \left(T\sin\xi^0 + \frac{\partial D}{\partial\theta}\right)\frac{\partial^2 L}{\partial\delta^2}}\right\},$$

$$A'_{31} = \frac{K^0}{m}\left\{\left(T\cos\xi^0 + \frac{\partial L}{\partial\theta}\right)\frac{\partial^2 D}{\partial\theta^2} - \left(T\sin\xi^0 + \frac{\partial D}{\partial\theta}\right)\frac{\partial^2 L}{\partial\theta^2} - \frac{\left[\left(T\cos\xi^0 + \frac{\partial L}{\partial\theta}\right)\frac{\partial^2 D}{\partial\delta\partial\theta} - \left(T\sin\xi^0 + \frac{\partial D}{\partial\theta}\right)\frac{\partial^2 L}{\partial\delta\partial\theta}\right]^2}{\left(T\cos\xi^0 + \frac{\partial L}{\partial\theta}\right)\frac{\partial^2 D}{\partial\delta^2} - \left(T\sin\xi^0 + \frac{\partial D}{\partial\theta}\right)\frac{\partial^2 L}{\partial\delta^2}}\right\}.$$

(2.25) corresponds to (2.10) of section 2.1.

This system has, as expected, Hamiltonian structure, so the eigenvalues of the system are symmetrically distributed about the imaginary axis. Since we would like the feedback control system to be stable, we choose initial values of $\Delta\lambda_\theta$ and $\Delta\lambda_q$ so that

the initial conditions are within the eigenspace spanned by the eigenvectors associated with the stable eigenvalues. Problems arise only when the Hamiltonian system has pure imaginary eigenvalues. Due to the sparse structure of the closed-loop system of equations, we can construct a closed form solution for the eigenvalues. These values are given by

$$\lambda = \sqrt{\frac{1}{2}\left(\frac{M_q}{I_{yy}}\right)^2 + A'_{21} \pm \sqrt{\frac{1}{4}\left(\frac{M_q}{I_{yy}}\right)^4 + A'_{21}\left(\frac{M_q}{I_{yy}}\right)^2 - A'_{31}A'_{24}}} \; .$$

Note that if the aerodynamic terms are all set to zero (exo-atmospheric vehicle with thrust vector control), this system will have repeated real eigenvalues. In this case there will be only one linearly independent eigenvector for each repeated eigenvalue, so it will be necessary to construct a generalized eigenvector to complete the stable eigenspace.

In practice an efficient algorithm for the determination of eigenvalues and eigenvectors would be used instead of an analytic determination as described here. Note that only the eigenvectors associated with the stable eigenvalues need to be determined.

For our 4×4 system, there are two stable eigenvalues and two associated eigenvectors, call them v_1 and v_2. We seek initial values of $\Delta\lambda_\theta$ and $\Delta\lambda_q$ such that for known initial values of $\Delta\theta$ and Δq we have

$$\begin{pmatrix} \Delta\theta \\ \Delta q \\ \Delta\lambda_\theta \\ \Delta\lambda_q \end{pmatrix} = \alpha_1 \begin{pmatrix} \\ v_1 \\ \\ \end{pmatrix} + \alpha_2 \begin{pmatrix} \\ v_2 \\ \\ \end{pmatrix} .$$

Examination of the feedback law for $\Delta\beta$ and $\Delta\delta$ reveals that only $\Delta\lambda_q$ needs to be computed. From the previous equation we see that $\Delta\lambda_q = \alpha_1 v_{1_4} + \alpha_2 v_{2_4}$, or

$$\Delta\lambda_q = \begin{bmatrix} v_{1_4} & v_{2_4} \end{bmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix} .$$

We also have

$$\begin{pmatrix} \Delta\theta \\ \Delta q \end{pmatrix} = \begin{pmatrix} v_{1_1} & v_{2_1} \\ v_{1_2} & v_{2_2} \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix} ,$$

leading to, finally

$$\Delta\lambda_q = \begin{bmatrix} v_{1_4} & v_{2_4} \end{bmatrix} \begin{pmatrix} v_{1_1} & v_{2_1} \\ v_{1_2} & v_{2_2} \end{pmatrix}^{-1} \begin{pmatrix} \Delta\theta \\ \Delta q \end{pmatrix}.$$

Substituting this expression into the inner-loop feedback law produces the final representation of the pitch control law.

$$\Delta\beta = -\Delta\theta - \frac{z\sin\beta^0 + \ell\cos\beta^0}{K^0\rho^2(T + \frac{\partial D}{\partial\theta}\sin\xi^0 + \frac{\partial L}{\partial\theta}\cos\xi^0)} \begin{bmatrix} v_{1_4} & v_{2_4} \end{bmatrix} \begin{pmatrix} v_{1_1} & v_{2_1} \\ v_{1_2} & v_{2_2} \end{pmatrix}^{-1} \begin{pmatrix} \Delta\theta \\ \Delta q \end{pmatrix},$$

$$\Delta\delta = -\frac{(T\cos\xi^0 + \frac{\partial L}{\partial\theta}|_0)\frac{\partial^2 D}{\partial\delta\partial\theta} - (T\sin\xi^0 + \frac{\partial D}{\partial\theta}|_0)\frac{\partial^2 L}{\partial\delta\partial\theta}}{(T\cos\xi^0 + \frac{\partial L}{\partial\theta}|_0)\frac{\partial^2 D}{\partial\delta^2} - (T\sin\xi^0 + \frac{\partial D}{\partial\theta}|_0)\frac{\partial^2 L}{\partial\delta^2}}\Delta\theta$$

$$+\frac{\frac{\partial M}{\partial\delta}}{K^0\rho^2\{(T\cos\xi^0 + \frac{\partial L}{\partial\theta}|_0)\frac{\partial^2 D}{\partial\delta^2} - (T\sin\xi^0 + \frac{\partial D}{\partial\theta}|_0)\frac{\partial^2 L}{\partial\delta^2}\}} \begin{bmatrix} v_{1_4} & v_{2_4} \end{bmatrix}$$

$$\times \begin{pmatrix} v_{1_1} & v_{2_1} \\ v_{1_2} & v_{2_2} \end{pmatrix}^{-1} \begin{pmatrix} \Delta\theta \\ \Delta q \end{pmatrix}.$$

This last pair of equations corresponds to (2.12) of Section 2.1.

These feedback gains for $\Delta\beta$ and $\Delta\delta$ are in terms only of $\Delta\theta$ and Δq, which are physical states that hopefully might be measured easily. It appears that we still have in the feedback law the term K^0, which is dependent on the outer-loop adjoint variables. This term is canceled by components of the eigenvectors, and may, for convenience and with complete generality, be set to one. These gains are dependent only on the position along the nominal point-mass trajectory, so they may be precomputed, reducing the computational burden during flight.

Commonly singular perturbation fast dynamics control is thought of as a means of bringing a multiple time scale system from some prescribed initial conditions to initial conditions that are consistent with the algebraic constraints created by the setting of our perturbation operator ϵ to zero. This fast dynamics control is referred to as boundary layer control in that the control is thought of as all occurring in a very short time period at the beginning of some longer interval. In this problem the singular perturbation control is applied continuously over the entire time interval under consideration. As such we need

singular perturbation gains at every point along the trajectory. Providing continuous gains is clearly not feasible so the gains are computed at a finite number of points along the trajectory, and are linearly interpolated from one point to the next. If it happens that pure imaginary eigenvalues are encountered in the construction of the gains, then those points are deleted from the set and gains before and after are interpolated across the gap.

3.0 OUTER-LOOP GUIDANCE ALGORITHM DEVELOPMENT

The outer-loop control used in this project is based on a Linear Quadratic Regulator (LQR). The outer-loop equations of motion are

$$\dot{m} = -\frac{T[\eta]}{I_{sp}[\eta]}$$

$$\dot{x} = V\cos\gamma$$

$$\dot{h} = V\sin\gamma$$

$$\dot{V} = \frac{1}{m}\left(T[\eta]\cos(\theta-\gamma+\beta) - D\right) - g\sin\gamma$$

$$\dot{\gamma} = \frac{1}{mV}\left(T[\eta]\sin(\theta-\gamma+\beta) + L\right) - \frac{g}{V}\cos\gamma + \frac{V\cos\gamma}{r_e+h}$$

with the throttle setting η and the pitch attitude θ as controls. We may consider mass m to be a very slowly changing state, and implement another order reduction, holding mass constant during the working of the regulator control. As we do not control mass explicitly, and it is very slowly varying, we will ignore the mass equation in the formulation of the regulator.

3.1 Continuous Linear Quadratic Regulator

The LQR controller requires a linear system. Since we have (by assumption) a nominal trajectory, this outer-loop control can be posed as a problem of controlling the error in tracking, which naturally lends itself to a perturbation analysis. Symbolically the non-linear equations of motion may be written as

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}). \tag{3.1}$$

Linearization of these equations gives

$$\Delta\dot{\mathbf{x}} = \frac{\partial\mathbf{f}(\mathbf{x},\mathbf{u})}{\partial\mathbf{x}}\Delta\mathbf{x} + \frac{\partial\mathbf{f}(\mathbf{x},\mathbf{u})}{\partial\mathbf{u}}\Delta\mathbf{u} \tag{3.2}$$

where the derivatives are evaluated along the nominal trajectory.

We will assume that the dynamic equations are differentiable along the nominal trajectory, or at least may be approximated locally as differentiable. The actual means of evaluating the partial derivatives will be considered later. This linearization process provides a linear system of the form

$$\dot{\mathbf{x}}(t) = \mathbf{A}(t)\mathbf{x}(t) + \mathbf{B}(t)\mathbf{u}(t) \tag{3.3}$$

where the $\mathbf{x}$ and $\mathbf{u}$ are in actual fact the perturbation quantities $\Delta\mathbf{x}$ and $\Delta\mathbf{u}$. The *LQR* controller also requires a quadratic cost function, given by

$$J = \tfrac{1}{2}\mathbf{x}^T(t_f)\mathbf{H}\mathbf{x}(t_f) + \int_{t_0}^{t_f} \tfrac{1}{2}\left[\mathbf{x}^T(t)\mathbf{Q}(t)\mathbf{x}(t) + \mathbf{u}^T(t)\mathbf{R}(t)\mathbf{u}(t)\right] dt. \tag{3.4}$$

The variational Hamiltonian for this system is then

$$\mathcal{H} = \lambda^T(\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}) + \tfrac{1}{2}\left[\mathbf{x}^T\mathbf{Q}\mathbf{x} + \mathbf{u}^T\mathbf{R}\mathbf{u}\right], \tag{3.5}$$

with $\mathbf{x}$, $\mathbf{u}$, $\mathbf{A}$, $\mathbf{B}$, $\mathbf{Q}$, $\mathbf{R}$, and λ all functions of time. The optimality condition is

$$\frac{\partial \mathcal{H}}{\partial u} = \mathbf{0} = \mathbf{B}^T\lambda + \mathbf{R}\mathbf{u},$$

giving

$$\mathbf{R}\mathbf{u} = -\mathbf{B}^T\lambda,$$

and finally

$$\mathbf{u} = -\mathbf{R}^{-1}\mathbf{B}^T\lambda.$$

We will now assume that λ may be represented as $\lambda(t) = \mathbf{S}(t)\mathbf{x}(t)$ for some positive semi-definite matrix $\mathbf{S}(t)$. We then have

$$\mathbf{u} = -\mathbf{R}^{-1}\mathbf{B}^T\lambda = -\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S}\mathbf{x}, \tag{3.6}$$

which, when substituted into our perturbation dynamics equation gives the closed loop dynamic system

$$\dot{\mathbf{x}}(t) = \left[\mathbf{A}(t) - \mathbf{B}(t)\mathbf{R}^{-1}(t)\mathbf{B}^T(t)\mathbf{S}(t)\right]\mathbf{x}(t). \tag{3.7}$$

We also have the adjoint differential equations

$$\dot{\lambda} = -\frac{\partial \mathcal{H}}{\partial \mathbf{x}} = -\mathbf{A}^T\lambda - \mathbf{Q}\mathbf{x}. \tag{3.8}$$

But we have $\lambda = \mathbf{S}\mathbf{x}$, which then gives us

$$\begin{aligned}\dot{\lambda} &= \mathbf{S}\dot{\mathbf{x}} + \dot{\mathbf{S}}\mathbf{x} \\ &= \mathbf{S}\left(\mathbf{A} - \mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S}\right)\mathbf{x} + \dot{\mathbf{S}}\mathbf{x} = -\mathbf{A}^T\mathbf{S}\mathbf{x} - \mathbf{Q}\mathbf{x}\end{aligned}$$

or

$$0 = \left(\dot{\mathbf{S}} + \mathbf{S}\mathbf{A} + \mathbf{A}^T\mathbf{S} + \mathbf{Q} - \mathbf{S}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S}\right)\mathbf{x}. \tag{3.9}$$

Since (3.9) must be satisfied for all $\mathbf{x}$ we finally have

$$\dot{\mathbf{S}} = -\mathbf{S}\mathbf{A} - \mathbf{A}^T\mathbf{S} - \mathbf{Q} + \mathbf{S}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{S}. \tag{3.10}$$

This equation is known as the matrix Riccati equation. We desire a feedback controller for $\mathbf{x}$ so we need to evaluate $\mathbf{S}(t)$. With $\mathbf{S}(t)$ we may then compute the feedback gains

$$\mathbf{K}(t) = -\mathbf{R}^{-1}(t)\mathbf{B}^T(t)\mathbf{S}(t) \tag{3.11}$$

which we may then employ as $\mathbf{u}(t) = \mathbf{K}(t)\mathbf{x}(t)$.

From the transversality condition $\lambda(t_f) = \mathbf{H}\mathbf{x}(t_f)$. Since $\lambda(t) = \mathbf{S}(t)\mathbf{x}(t)$, $\mathbf{S}(t_f) = \mathbf{H}$. We now have a differential equation for $\mathbf{S}(t)$ with a final value of $\mathbf{H}$.

3.1.1 Terminal Weighting And Final Control Behavior

A problem with the finite time linear regulator is that there are often large changes in gains at the end of the trajectory. These changes are due to the choice of the final weighting matrix $\mathbf{H}$. Many conventional regulator problems present clear guides for the choice of weights, including the weights on the final error. In this problem it is not clear whether the final error should be weighted either more or less heavily than any other point

on the trajectory. What we desire here is a choice of weights that will result in "good" final dynamic characteristics.

It can be shown that for constant $\mathbf{A}$, $\mathbf{B}$, $\mathbf{Q}$, and $\mathbf{R}$ matrices (3.10) will converge to a constant value of $\mathbf{S}$, ($\dot{\mathbf{S}} = 0$) if integrated in reverse time. The value that $\mathbf{S}$ takes when (3.10) is integrated backward in time sufficiently is referred to as the steady state solution of the Riccati equation.

If $\mathbf{A}$ and $\mathbf{B}$ are relatively slowly varying near the final time, and $\mathbf{Q}$ and $\mathbf{R}$ are chosen to be smoothly varying near the final time, then a technique that will produce a smooth final control, without either dramatic reduction in final control, or a large final control input, is to compute the final weighting matrix $\mathbf{H}$ as the solution to the steady state Riccati equation with weights on states and controls of $\mathbf{Q}(t_f)$ and $\mathbf{R}(t_f)$.

3.1.2 Choosing Weights For States And Controls

We should recognize that the time varying Riccati equation is necessarily an explicit function of time along the trajectory. If we choose to key the gains to some other variable along the trajectory, say total specific energy, then the gains will tend to not be the "correct" gains if the nominal time-rate-of-change of energy and the actual time-rate-of-change of energy for a particular energy are different. This deviation from expected time scale of a key variable could be changed by transforming the gains into a function of energy, (i.e., of energy-to-go instead of time-to-go) but this type of transformation was not really deemed necessary. The purpose of the outer-loop feedback controller is to keep the actual trajectory near the nominal, so the choice of one error controller over another is arbitrary. That being so, a modification was made to the time varying controller—the Ricatti equation was solved to steady state at each of a number of points along the trajectory, and the gains were then interpolated between those points. So we have a locally infinite-time steady-state LQR that moves along the trajectory.

Choice of the state and control weights are made arbitrarily, with final weights and the associated gains chosen so that the dynamic behavior of the vehicle is satisfactory. It should be noted that the LQR produces a locally stable closed loop system if the state weighting matrix $\mathbf{Q}$ is non-negative definite, and the control weighting matrix $\mathbf{R}$ is positive definite.

Choices of the relative weights on the states and the controls in the LQR determines the position of the closed loop poles of the linearized point-mass system. Increasing the weights on the states relative to the controls results in the closed loop poles moving to the left in the complex plane. Recall that the closed loop pitch dynamics system has poles that are fixed by the singular perturbation procedure, i.e., there are no arbitrary parameters to adjust. As the enabling premise of the multi-loop control concept is that there is separation between the characteristic times of the slow and fast loops, it is necessary to make sure that the outer-loop poles are not pushed so far to the left that the time separation is lost.

This consideration leads to the requirement that the closed loop poles of the outer-loop system be checked after every adjustment in the weights on the states. Computing the closed loop poles of the system is a relatively simple and efficient procedure to implement during the computation of the outer-loop gains, but it is another step in the control formulation.

These outer-loop gains are based entirely on the prescribed nominal trajectory, and can be computed *off-line*, that is, on the ground, and stored until needed during the flight.

3.2 Discrete Linear Quadratic Regulator

It is supposed that a control system of this type will necessarily be implemented by use of a digital computer, controlling the vehicle so as to follow a trajectory that is described by tabulated data. We would presume that the trajectory data is stored

at discrete intervals, and that the controls are also updated at discrete intervals, not continuously as implied by the Continuous *LQR*. This effectively discrete controller leads us to a change of the outer-loop control from that of a Continuous Linear Quadratic Regulator to a Discrete Linear Quadratic Regulator formulation.

The discrete *LQR* also requires a linear dynamic system, but discrete instead of continuous,

$$\mathbf{x}(t_{k+1}) = \mathbf{A}(t_k)\mathbf{x}(t_k) + \mathbf{B}(t_k)\mathbf{u}(t_k), \tag{3.12}$$

and a discrete analog to the continuous cost function

$$J = \tfrac{1}{2}\mathbf{x}^T(t_N)\mathbf{H}\mathbf{x}(t_N) + \tfrac{1}{2}\sum_{k=1}^{N-1}\left[\mathbf{x}^T(t_k)\mathbf{Q}(t_k)\mathbf{x}(t_k) + \mathbf{u}^T(t_k)\mathbf{R}(t_k)\mathbf{u}(t_k)\right],$$

where $\mathbf{H}$ and $\mathbf{Q}(t_k)$ are real symmetric positive semi-definite $n \times n$ matrices, $\mathbf{R}(t_k)$ is a real symmetric positive definite $m \times m$ matrix, and N is the fixed number of time intervals greater than zero. For notational convenience let $\mathbf{x}(t_k) = \mathbf{x}_k$, $\mathbf{u}(t_k) = \mathbf{u}_k$, ...

Following the derivation in Kirk[15], consider the cost at $t = t_N$.

$$J_{N,N}(\mathbf{x}_N) = \tfrac{1}{2}\mathbf{x}_N^T\mathbf{H}\mathbf{x}_N = J^*_{N,N}(\mathbf{x}_N) \equiv \mathbf{x}_N^T\mathbf{P}_N\mathbf{x}_N, \tag{3.13}$$

where $\mathbf{P}_N \equiv \mathbf{H}$. The cost over the final interval $t_{N-1} - t_N$ is

$$J_{N-1,N}(\mathbf{x}_{N-1}, \mathbf{u}_{N-1}) = \tfrac{1}{2}\mathbf{x}_{N-1}^T\mathbf{Q}_{N-1}\mathbf{x}_{N-1} + \tfrac{1}{2}\mathbf{u}_{N-1}^T\mathbf{R}_{N-1}\mathbf{u}_{N-1} + \tfrac{1}{2}\mathbf{x}_N^T\mathbf{P}_N\mathbf{x}_N. \tag{3.14}$$

From (3.12), $\mathbf{x}_N = \mathbf{A}_{N-1}\mathbf{x}_{N-1} + \mathbf{B}_{N-1}\mathbf{u}_{N-1}$, so we may rewrite (3.14) as

$$\begin{aligned} J_{N-1,N}(\mathbf{x}_{N-1}, \mathbf{u}_{N-1}) &= \tfrac{1}{2}\mathbf{x}_{N-1}^T\mathbf{Q}_{N-1}\mathbf{x}_{N-1} + \tfrac{1}{2}\mathbf{u}_{N-1}^T\mathbf{R}_{N-1}\mathbf{u}_{N-1} \\ &+ \tfrac{1}{2}\left[\mathbf{A}_{N-1}\mathbf{x}_{N-1} + \mathbf{B}_{N-1}\mathbf{u}_{N-1}\right]^T\mathbf{P}_N\left[\mathbf{A}_{N-1}\mathbf{x}_{N-1} + \mathbf{B}_{N-1}\mathbf{u}_{N-1}\right]. \end{aligned} \tag{3.15}$$

We seek a control $\mathbf{u}_{N-1}$ that minimizes this cost. This is provided by $\mathbf{u}^*_{N-1}$ such that $\frac{\partial J_{N-1,N}}{\partial \mathbf{u}_{N-1}} = 0$. Thus $\mathbf{u}^*_{N-1}$ must satify

$$\begin{aligned} \frac{\partial J^*_{N-1,N}}{\partial \mathbf{u}^*_{N-1}} = 0 &= \mathbf{R}_{N-1}\mathbf{u}^*_{N-1} + \mathbf{B}_{N-1}^T\mathbf{P}_N\left[\mathbf{A}_{N-1}\mathbf{x}_{N-1} + \mathbf{B}_{N-1}\mathbf{u}^*_{N-1}\right] \\ &= \left[\mathbf{R}_{N-1} + \mathbf{B}_{N-1}^T\mathbf{P}_N\mathbf{B}_{N-1}\right]\mathbf{u}^*_{N-1} + \mathbf{B}_{N-1}^T\mathbf{P}_N\mathbf{A}_{N-1}\mathbf{x}_{N-1}. \end{aligned}$$

or, equivalently

$$\mathbf{u}^*_{N-1} = -\left[\mathbf{R}_{N-1} + \mathbf{B}^T_{N-1}\mathbf{P}_N\mathbf{B}_{N-1}\right]^{-1}\mathbf{B}^T_{N-1}\mathbf{P}_N\mathbf{A}_{N-1}\mathbf{x}_{N-1}$$

$$\equiv \mathbf{F}_{N-1}\mathbf{x}_{N-1}. \tag{3.16}$$

This extremal value of $J_{N-1,N}$ is a minimum if $\frac{\partial^2 J_{N-1,N}}{\partial \mathbf{u}^2_{N-1}}$ is positive definite at $\mathbf{u}_{N-1} = \mathbf{u}^*_{N-1}$. But

$$\frac{\partial^2 J_{N-1,N}}{\partial \mathbf{u}^2_{N-1}} = \mathbf{R}_{N-1} + \mathbf{B}^T_{N-1}\mathbf{P}_N\mathbf{B}_{N-1}.$$

Since $\mathbf{R}$ is positive definite, and $\mathbf{P}_N$ is positive semi-definite making $\mathbf{B}^T_{N-1}\mathbf{P}_N\mathbf{B}_{N-1}$ positive semi-definite, $\frac{\partial^2 J_{N-1,N}}{\partial \mathbf{u}^2_{N-1}}$ is positive definite, and $J^*_{N-1,N}$ is minimized at $\mathbf{u}^*_{N-1}$. Since $\mathbf{R} + \mathbf{B}^T\mathbf{P}_N\mathbf{B}$ is positive definite the inverse is guaranteed to exist. Substituting (3.16) into (3.15) gives

$$J_{N-1,N}(\mathbf{x}_{N-1}) = \tfrac{1}{2}\mathbf{x}^T_{N-1}\{\left[\mathbf{A}_{N-1} + \mathbf{B}_{N-1}\mathbf{F}_{N-1}\right]^T\mathbf{P}_N\left[\mathbf{A}_{N-1} + \mathbf{B}_{N-1}\mathbf{F}_{N-1}\right]$$

$$+ \mathbf{F}^T_{N-1}\mathbf{R}_{N-1}\mathbf{F}_{N-1} + \mathbf{Q}_{N-1}\}\mathbf{x}_{N-1} \tag{3.17}$$

$$\equiv \tfrac{1}{2}\mathbf{x}^T_{N-1}\mathbf{P}_{N-1}\mathbf{x}_{N-1}. \tag{3.18}$$

These equations lead immediately to the direct algorithm for computing the feedback gains for the discrete *LQR*. Since the cost at t_{N-1} is of exactly the same form as the cost at t_N then the same procedure will give the gains and the cost at t_{N-2} and t_{N-3} and so on to $k = 1$. We then define the recursive algorithm

$$\begin{aligned}\mathbf{F}_k &= -\left[\mathbf{R}_k + \mathbf{B}^T_k\mathbf{P}_{k+1}\mathbf{B}_k\right]^{-1}\mathbf{B}^T_k\mathbf{P}_{k+1}\mathbf{A}_k,\\ \mathbf{P}_k &= \left[\mathbf{A}_k + \mathbf{B}_k\mathbf{F}_k\right]^T\mathbf{P}_{k+1}\left[\mathbf{A}_k + \mathbf{B}_k\mathbf{F}_k\right] + \mathbf{F}^T_k\mathbf{R}_k\mathbf{F}_k + \mathbf{Q}_k.\end{aligned} \tag{3.19}$$

Substituting in (3.19) gives the discrete Riccati equation

$$\mathbf{P}_k = \mathbf{A}^T_k\left[\mathbf{P}_{k+1} - \mathbf{P}_{k+1}\mathbf{B}_k\left(\mathbf{R}_k + \mathbf{B}^T_k\mathbf{P}_{k+1}\mathbf{B}_k\right)^{-1}\mathbf{B}^T_k\mathbf{P}_{k+1}\right]\mathbf{A}_k + \mathbf{Q}_k. \tag{3.20}$$

The final behavior of the discrete regulator is similar to that of the continuous, and special consideration of the final state weights must be given if smooth final control

behavior is desired. As with the continuous regulator, a final state weight matrix may be constructed by stepping the discrete Riccati equation back in time until a constant value is observed. A number of techniques are available to accelerate the convergence to a steady state value.[6,7]

It may be noted that as with the Continuous *LQR* the closed-loop poles of the Discrete *LQR* are functions of the ratio of the weights on the states and the controls. Regulator theory tells us that the discrete regulator will be stable for a positive definite control weight **R** and a positive semi-definite state weight **Q**. These results are valid as long as the actual system may be reasonably represented by the discrete model. For reasonable system behavior under the control of a discrete regulator, the control update rate must be fast compared to the characteristic response of the system. It is also evident that for a system controlled by a stable discrete controller the characteristic times of the system can not be made faster than the update rate of the discrete system. If we can choose an update rate that is fast relative to the point-mass outer loop, then we should be able to control the continuous system with the discrete controller. If we also choose the update rate to be slow with respect to the inner-loop closed loop characteristic times, then the initial premise that there is a separation of time scales between the inner and outer loops will be guaranteed and we need not concern ourselves with the problem of interference between the two control loops.

Note that all the components of the outer loop controllers, both continuous and discrete, are dependent only on the nominal trajectory, and so may be precomputed. This feature allows for a relatively low computational workload in flight.

3.3 On Constructing The Discrete System Realization

The dynamic system that we are attempting to control is not discrete, but we may construct a discrete representation of it. It can be shown that the equations of motion,

(3.3), may be integrated over a period of time, say from $t = t_k$ to $t = t_{k+1}$ to produce the discrete representation

$$\mathbf{x}(t_{k+1}) = \mathbf{\Phi}(t_{k+1}, t_k)\mathbf{x}(t_k) + \int_{t_k}^{t_{k+1}} \mathbf{\Phi}(t_{k+1}, \tau)\mathbf{B}(\tau)\mathbf{u}(\tau)\, d\tau. \tag{3.21}$$

The term $\mathbf{\Phi}(t_{k+1}, t_k)\mathbf{x}(t_k)$ is the solution to the uncontrolled problem $\dot{\mathbf{x}}(t) = \mathbf{A}(t)\mathbf{x}(t)$ from t_k to t_{k+1}. We have made the assumption that this system is linear, so $\mathbf{A}(t)$ is a function of only time and not of $\mathbf{x}$. Now suppose instead we write

$$\dot{\mathbf{X}}(t) = \mathbf{A}(t)\mathbf{X}(t) \tag{3.22}$$

for any $n \times n$ matrix $\mathbf{X}(t)$. Then

$$\mathbf{X}(t_{k+1}) = \mathbf{\Phi}(t_{k+1}, t_k)\mathbf{X}(t_k). \tag{3.23}$$

Now suppose we let $\mathbf{X}(t_k) = \mathbf{I}$, then

$$\mathbf{X}(t_{k+1}) = \mathbf{\Phi}(t_{k+1}, t_k)\mathbf{I} \tag{3.24}$$

and we deduce that we may construct the state transition matrix by integrating $\dot{\mathbf{\Phi}}(t) = \mathbf{A}(t)\mathbf{\Phi}(t)$ from t_k to t_{k+1} with initial conditions of $\mathbf{\Phi}(t_k) = \mathbf{I}$. Note that this result is independent of whether $\mathbf{A}(t)$ is constant or time varying. We will designate the state transition matrix $\mathbf{\Phi}(t_{k+1}, t_k)$ as $\mathbf{\Phi}_k$ for the purposes of the fixed time step discrete problem.

The second term $\int_{t_k}^{t_{k+1}} \mathbf{\Phi}(t_{k+1}, \tau)\mathbf{B}(\tau)\mathbf{u}(\tau)d\tau$ can be handled as follows. We will assume for our purposes that the control $\mathbf{u}(t)$ will be a zero order hold (we can specify $\mathbf{u}_k$ to be whatever we desire); the control is constant over the interval. We may therefore remove it from the integral and write it as $\int_{t_k}^{t_{k+1}} [\mathbf{\Phi}(t_{k+1}, \tau)\mathbf{B}(\tau)d\tau]\, \mathbf{u}_k$, so the integrand is not a function of the controls or the states. Following our previous discussion of the construction of the state transition $\mathbf{\Phi}(t_{k+1}, t_k)$ we can easily construct transition matrices $\mathbf{\Phi}(t_{k+1}, \tau)$ for any values of τ. The control matrix $\mathbf{B}(t)$ is known, so it is a simple numeric problem to perform the integration. We usually designate this integral term $\mathbf{\Gamma}_k$, so

$$\mathbf{\Gamma}_k = \int_{t_k}^{t_{k+1}} \mathbf{\Phi}(t_{k+1}, \tau)\mathbf{B}(\tau)d\tau \tag{3.25}$$

and since Γ_k does not appear in the integral, we may integrate this using a trapezoid or Simpson's rule integration. Our final discretized system is then of the form

$$\mathbf{x}_{k+1} = \Phi_k \mathbf{x}_k + \Gamma_k \mathbf{u}_k \tag{3.26}$$

These matrices Φ_k and Γ_k correspond to the $\mathbf{A}$ and $\mathbf{B}$ in the discrete regulator discussion.

4.0 VEHICLE SIMULATIONS

The control algorithm was tested by simulation on two vehicles to determine whether it behaved as predicted. The two vehicles that were used were an arbitrary high performance vectored-thrust controlled exo-atmospheric missile, and a Winged-Cone Configuration Hypersonic Vehicle Simulation Model from NASA Langley Research Center.

4.1 Exo-Atmospheric Missile

The first of the simulations was based on a much simplified mathematical model. A drawing of the simplified model is presented in Figure 4.1. The modeled vehicle has thrust vector pitch control only and is not affected by aerodynamic lift, drag, or moment. This dynamic system and the pitch control derivation that follows are similar to that presented by Cliff.[16]

The nominal trajectory that we attempt to follow with this vehicle is not an optimal trajectory (at least it is not optimal in the sense of minimizing some known performance index), but is the byproduct of an arbitrarily imposed pitch attitude schedule in the point mass integration. This vehicle model was constructed as a simple and expedient means of determining whether this control concept would work acceptably well for an arbitrary vehicle and an arbitrary trajectory. There is no nominal cost function.

This vehicle has the following nominal properties:

$$\text{Initial mass} = 2000\,\text{kg}$$

$$\text{Final mass} = 500\,\text{kg}$$

$$\text{Nominal constant thrust} = 392{,}000\,\text{N}$$

$$\text{Nominal exhaust exit velocity } I_{sp} = 7840\,\text{m/s}$$

$$\text{Thrust vector moment arm } \ell = 3\,\text{m}$$

$$\text{Initial pitch moment of inertia} = 7835.9\,\text{kg m}^2$$

Final pitch moment of inertia = 3500 kg m^2

The vehicle is modeled as a missile that has nominal constant thrust, with weight varying as fuel is burned. Neglecting aerodynamics results in some interesting differences from the more complete control derivation presented previously.

4.1.1 Simplified Model Inner-Loop Control

The equations of motion for this vehicle are

$$\begin{aligned}
\dot{m} &= -\frac{T}{I_{sp}}, \\
\dot{x} &= V\cos\gamma, \\
\dot{h} &= V\sin\gamma, \\
\dot{V} &= \frac{T\cos(\theta-\gamma+\beta)}{m} - g\sin\gamma, \\
\dot{\gamma} &= \frac{T\sin(\theta-\gamma+\beta)}{mV} - \frac{g\cos\gamma}{V} + \frac{V\cos\gamma}{r_e+h}, \\
\epsilon\dot{\theta} &= q + \frac{V\cos\gamma}{r_e+h}, \\
\epsilon\dot{q} &= \frac{T}{I_{yy}}(z\cos\beta - \ell\sin\beta).
\end{aligned} \tag{4.1}$$

We may construct the Hamiltonian in two parts again:

$$\begin{aligned}
\mathcal{H}^0 &= -\lambda_m\frac{T}{I_{sp}} + \lambda_x V\cos\gamma + \lambda_h V\sin\gamma + \lambda_V\left(\frac{T\cos(\theta-\gamma+\beta)}{m} - g\sin\gamma\right) \\
&\quad + \lambda_\gamma\left(\frac{T\sin(\theta-\gamma+\beta)}{mV} - \frac{g\cos\gamma}{V} + \frac{V\cos\gamma}{r_e+h}\right), \\
\epsilon\mathcal{H}^1 &= \lambda_\theta\left(q + \frac{V\cos\gamma}{r_e+h}\right) + \lambda_q\frac{T}{I_{yy}}(z\cos\beta - \ell\sin\beta).
\end{aligned} \tag{4.2}$$

Let us assume that the normal thrust offset z is equal to zero. With $\epsilon = 0$, $\beta^0 = 0$, and with θ as the control, we have the outer loop optimality condition

$$\frac{\partial\mathcal{H}^0}{\partial\theta} = 0 = -\lambda_V\frac{T\sin(\theta-\gamma)}{m} + \lambda_\gamma\frac{T\cos(\theta-\gamma)}{Vm}.$$

This equation can be rearranged as

$$\frac{\lambda_V}{\lambda_\gamma} = \frac{T\cos(\theta-\gamma)}{VT\sin(\theta-\gamma)},$$

from which we can finally define

$$\begin{aligned} \lambda_V^0 &= K^0 \cos(\theta^0 - \gamma^0), \\ \lambda_\gamma^0 &= K^0 V \sin(\theta^0 - \gamma^0). \end{aligned} \tag{4.3}$$

Employing the time stretching transformation of Chapter 2, (2.3), we form the inner loop state-adjoint system of equations

$$\begin{aligned} \theta' &= q + \frac{V \cos\gamma}{r_e + h}. \\ q' &= -\frac{T\ell}{I_{yy}} \sin\beta. \\ \lambda_\theta' &= \lambda_V^0 \frac{T \sin(\theta - \gamma + \beta)}{m} - \lambda_\gamma^0 \frac{T \cos(\theta - \gamma + \beta)}{mV}. \\ \lambda_q' &= -\lambda_\theta. \end{aligned}$$

Substituting the values of λ_V^0 and λ_γ^0 from (4.3) into the λ_θ' equation gives

$$\begin{aligned} \lambda_\theta' &= \frac{KT}{m} \left[\cos(\theta^0 - \gamma^0) \sin(\theta - \gamma + \beta) - \sin(\theta^0 - \gamma^0) \cos(\theta - \gamma + \beta)\right] \\ &= \frac{KT}{m} \sin(\theta - \theta^0 - (\gamma - \gamma^0) + \beta), \end{aligned}$$

so the system of equations can finally be written

$$\begin{aligned} \theta' &= q + \frac{V \cos\gamma}{r_e + h}, \\ q' &= -\frac{T\ell}{I_{yy}} \sin\beta, \\ \lambda_\theta' &= \frac{KT}{m} \sin(\theta - \theta^0 - (\gamma - \gamma^0) + \beta), \\ \lambda_q' &= -\lambda_\theta. \end{aligned}$$

The inner loop optimality condition is

$$\begin{aligned} \frac{\partial \mathcal{H}}{\partial \beta} = 0 &= -\lambda_V^0 \frac{T \sin(\theta - \gamma + \beta)}{m} + \lambda_\gamma^0 \frac{T \cos(\theta - \gamma + \beta)}{mV} - \lambda_q \frac{T\ell}{I_{yy}} \cos\beta \\ &= -\lambda_V^0 \sin(\theta - \gamma + \beta) + \frac{\lambda_\gamma^0}{V} \cos(\theta - \gamma + \beta) - \frac{\lambda_q}{\rho^2} \ell \cos\beta \\ &= K \left[\sin(\theta^0 - \gamma^0) \cos(\theta - \gamma + \beta) - \cos(\theta^0 - \gamma^0) \sin(\theta - \gamma + \beta)\right] - \frac{\lambda_q}{\rho^2} \ell \cos\beta \\ &= K \sin(\theta - \theta^0 - (\gamma - \gamma^0) + \beta) + \frac{\lambda_q}{\rho^2} \ell \cos\beta. \end{aligned}$$

Linearize this equation about the nominal and solve for $\Delta\beta$:

$$\begin{aligned} 0 &= \cos(\theta-\theta^0-(\gamma-\gamma^0)+\beta)(\Delta\theta+\Delta\beta)+\Delta\lambda_q\frac{\ell}{\rho^2}\cos\beta \\ &= \Delta\theta+\Delta\beta+\frac{\ell}{\rho^2}\Delta\lambda_q, \\ \Delta\beta &= -\Delta\theta-\frac{\ell}{K^0\rho^2}\Delta\lambda_q. \end{aligned} \tag{4.4}$$

We now linearize the state-adjoint equations, evaluate along the nominal trajectory and substitute this value of $\Delta\beta$ to get the linearized closed loop system

$$\begin{pmatrix} \Delta\theta' \\ \Delta q' \\ \Delta\lambda'_\theta \\ \Delta\lambda'_q \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ \frac{T\ell}{I_{yy}} & 0 & 0 & \frac{T\ell^2}{K^0\rho^2 I_{yy}} \\ 0 & 0 & 0 & -\frac{T\ell}{I_{yy}} \\ 0 & 0 & -1 & 0 \end{pmatrix} \begin{pmatrix} \Delta\theta \\ \Delta q \\ \Delta\lambda_\theta \\ \Delta\lambda_q \end{pmatrix}.$$

The sparse structure of this matrix allows us to find the eigenvalues and eigenvectors in closed form to determine the optimal feedback gains. We notice from the structure that the eigenvalues of this matrix are repeated and are equal to $\pm\sqrt{\frac{T\ell}{I_{yy}}}$. We only need the stable values, so we use the value $-\sqrt{\frac{T\ell}{I_{yy}}}$. The one eigenvector for this eigenvalue is $\begin{bmatrix} 1 & -\sqrt{\frac{T\ell}{I_{yy}}} & 0 & 0 \end{bmatrix}^T$. A second vector is needed so we construct a generalized eigenvector for this eigenvalue, which may be given by $\begin{bmatrix} 0 & -1 & 2\frac{K^0\rho^2}{\ell} & 2\frac{K^0\rho^2}{\ell}\sqrt{\frac{I_{yy}}{T\ell}} \end{bmatrix}^T$.

We desire to satisfy the following equation for arbitrary initial values of $\Delta\theta$ and Δq

$$\begin{pmatrix} \Delta\theta \\ \Delta q \\ \Delta\lambda_\theta \\ \Delta\lambda_q \end{pmatrix} = \alpha_1 \begin{pmatrix} \\ v_1 \\ \\ \end{pmatrix} + \alpha_2 \begin{pmatrix} \\ v_2 \\ \\ \end{pmatrix}$$

where v_1 is the stable eigenvector and v_2 is the stable generalized eigenvector. We can write this as

$$\begin{pmatrix} \Delta\theta \\ \Delta q \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ -\sqrt{\frac{T\ell}{I_{yy}}} & -1 \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix}$$

and

$$\begin{pmatrix} \Delta\lambda_\theta \\ \Delta\lambda_q \end{pmatrix} = \begin{pmatrix} 0 & 2\frac{K\rho^2}{\ell} \\ 0 & 2\frac{K\rho^2}{\ell}\sqrt{\frac{I_{yy}}{T\ell}} \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix}.$$

From the first pair of equations we get

$$\begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ -\sqrt{\frac{T\ell}{I_{yy}}} & -1 \end{pmatrix} \begin{pmatrix} \Delta\theta \\ \Delta q \end{pmatrix},$$

which, combined with the second pair gives

$$\begin{pmatrix} \Delta\lambda_\theta \\ \Delta\lambda_q \end{pmatrix} = \begin{pmatrix} -2\frac{K\rho^2}{\ell}\sqrt{\frac{T\ell}{I_{yy}}} & -2\frac{K\rho^2}{\ell} \\ -2\frac{K\rho^2}{\ell} & -2\frac{K\rho^2}{\ell}\sqrt{\frac{I_{yy}}{T\ell}} \end{pmatrix} \begin{pmatrix} \Delta\theta \\ \Delta q \end{pmatrix}.$$

Substituting this into (4.4) gives the feedback law

$$\Delta\beta = \Delta\theta + 2\sqrt{\frac{I_{yy}}{T\ell}}\Delta q. \tag{4.5}$$

This schedule of gains is simple enough to compute during the simulation, so it is not precomputed. These gains are precisely those developed by Cliff[16] using a different outer loop velocity representation.

We may easily compute eigenvalues of the closed loop system. Recall

$$\begin{pmatrix} \Delta\dot{\theta} \\ \Delta\dot{q} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \Delta\theta \\ \Delta q \end{pmatrix} + \begin{pmatrix} 0 \\ -\frac{T\ell}{I_{yy}} \end{pmatrix} \Delta\beta.$$

But we now have

$$\Delta\beta = \Delta\theta + 2\sqrt{\frac{I_{yy}}{T\ell}}\Delta q,$$

so the closed loop system is

$$\begin{pmatrix} \Delta\dot{\theta} \\ \Delta\dot{q} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -\frac{T\ell}{I_{yy}} & -2\sqrt{\frac{T\ell}{I_{yy}}} \end{pmatrix} \begin{pmatrix} \Delta\theta \\ \Delta q \end{pmatrix}.$$

This dynamic system has the repeated real eigenvalue $\lambda = -\sqrt{\frac{T\ell}{I_{yy}}}$. The closed-loop system is non-oscillatory, so the errors go to zero asymptotically.

This control is seen to be dependent on the thrust T, the weight through I_{yy}, and the thrust moment arm ℓ. With constant T and I_{sp}, the weight is a function of time independent of the nominal trajectory. Therefore, the only trajectory dependencies in the inner loop are the nominal θ and q.

4.1.1.2 Modeling Error Effects

It is nice to know what the effect of modeling errors will be on the system. This system is governed by the value of the parameter group $\frac{T\ell}{I_{yy}}$. For notational convenience let us call this parameter group a. When computing the gains, let us assume that we have a nominal estimate of this group—call it a_0. The closed loop equation $\dot{\mathbf{x}} = (\mathbf{A} + \mathbf{BF})\mathbf{x}$ can be written

$$\begin{pmatrix} \Delta\dot{\theta} \\ \Delta\dot{q} \end{pmatrix} = \left\{ \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} + \begin{pmatrix} 0 \\ -a \end{pmatrix} \begin{pmatrix} 1 & 2\sqrt{\frac{1}{a_0}} \end{pmatrix} \right\} \begin{pmatrix} \Delta\theta \\ \Delta q \end{pmatrix}.$$

Written as a single matrix, this is

$$\begin{pmatrix} \Delta\dot{\theta} \\ \Delta\dot{q} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -a & -\frac{2a}{\sqrt{a_0}} \end{pmatrix} \begin{pmatrix} \Delta\theta \\ \Delta q \end{pmatrix}.$$

The eigenvalues of this closed loop dynamic system are the solution to the equation

$$\lambda\left(\lambda + \frac{2a}{\sqrt{a_0}}\right) + a = 0$$

which are given by

$$\begin{aligned} \lambda &= \frac{-\frac{2a}{\sqrt{a_0}} \pm \sqrt{\frac{4a^2}{a_0} - 4a}}{2} \\ &= -\frac{a}{\sqrt{a_0}} \pm \sqrt{\frac{a^2}{a_0} - a} \\ &= -\sqrt{a}\left\{\sqrt{\frac{a}{a_0}} \mp \sqrt{\frac{a}{a_0} - 1}\right\}. \end{aligned}$$

Replace a with a_0 times some constant c to get

$$\lambda = -\sqrt{c * a_0}\left\{\sqrt{c} \mp \sqrt{c-1}\right\}.$$

Note that $\lambda_0 = -\sqrt{a_0}$. If we now divide both sides by $\sqrt{a_0}$, we will get

$$\begin{aligned} \frac{\lambda}{-\lambda_0} &= -\sqrt{c}\left\{\sqrt{c} \mp \sqrt{c-1}\right\} \\ &= -c \pm \sqrt{c^2 - c}. \end{aligned}$$

This representation is the negative ratio (so the values are on the usual side of the imaginary axis) of the actual closed-loop poles to the modeled closed-loop poles. A negative value is still stable. Examination of this last relationship reveals what happens to the poles of the system when the actual parameters are different than the modeled parameters. We can see that the parameter group $\frac{T\ell}{I_{yy}}$ is a control effectiveness indicator. As this group gets larger than the nominal, that is, $c > 1$, the poles remain real but move apart on the negative real axis. We see that as $c \to \infty$, one pole goes to $2c\lambda_0$, and the other goes to $0.5\lambda_0$. Effectively the actual gain on pitch rate error is higher than would be optimal for the actual control effectiveness.

We also see that when $c < 1$ the eigenvalues are

$$\lambda = \lambda_0 \times \left(c \pm i\sqrt{c - c^2}\right).$$

The real part of these eigenvalues is the nominal λ_0 multiplied by the positive constant c. We see that this feedback law is stable for all variations in the parameter group $\frac{T\ell}{I_{yy}}$. (At least as long as all elements are non-negative, that is!) When the control effectiveness is much reduced, as when ℓ is much shorter than modeled, or when the pitch moment of inertia I_{yy} is much larger than modeled, the closed loop behavior may be oscillatory. In this case the actual gain on pitch rate error is lower than the optimal gain given the actual control effectiveness. If the control effectiveness is higher than modeled, the motion is asymptotically and monotonically stable.

The pole travel as c varies from 0 to ∞ is shown in Figure 4.2.

4.1.3 Simplified Model Nominal Trajectory

There was an explicit assumption in the derivation of the inner loop control law that the nominal trajectory was nearly optimal for some terminal cost function. In the construction of a nominal trajectory for the simplified problem this will be ignored and an arbitrary trajectory will be created.

For ease of construction, the nominal trajectory was created by integration of the point-mass equations of motion after defining initial conditions, thrust, I_{sp}, and $\theta(t)$. The vertical plane point-mass equations of motion are

$$\begin{aligned}
\dot{m} &= -\frac{T}{I_{sp}}, \\
\dot{x} &= V\cos\gamma, \\
\dot{h} &= V\sin\gamma, \\
\dot{V} &= \frac{T}{m}\cos(\theta-\gamma) - g\sin\gamma, \\
\dot{\gamma} &= \frac{T}{mV}\sin(\theta-\gamma) - \frac{g}{V}\sin\gamma + \frac{V\cos\gamma}{r_e+h}.
\end{aligned} \tag{4.6}$$

Initial conditions for the states are

$$m_0 = 2000\,\text{kg}, \qquad x_0 = 0, \qquad h_0 = 0,$$

$$V_0 = 1\,\text{m/sec}, \qquad \gamma_0 = \frac{\pi}{2}\text{rad}.$$

I_{sp} is 7840 m/s. T is nominally 392,000 N. The Earth radius r_e is 6,370,000 m. $\theta(t)$ is defined by the equation

$$\theta(t) = a_1e^{-t} + a_2e^{-2t} + a_3e^{-3t} + a_4e^{-4t} + a_5$$

with the a_i defined such that $\theta(0) = \frac{\pi}{2}$, $\theta(0.2) = \frac{\pi}{2}$, $\theta(\infty) = 0.1$, $\dot{\theta}(0) = 0$, $\dot{\theta}(0.2) = 0$. These constraints, together with the defining equation, result in values of the a_i

$$\begin{aligned}
a_1 &= 6.534462275, \\
a_2 &= -10.8507062, \\
a_3 &= 7.981210895, \\
a_4 &= -2.19417064, \\
a_5 &= 0.1\,.
\end{aligned}$$

$\theta(t)$ is presented in Figure 4.3. $\dot{\theta}(t)$ is presented in Figure 4.4. In addition, ℓ = 3.0 m. The vehicle equations were integrated using this nominal θ for thirty seconds. During this time the missile climbs to about 23 km altitude, travels down-range 116 km, and reaches 10.5 km/sec velocity. The mass also changes from 2000 kg initial to 500 kg final.

4.1.4 Inner-Loop Only Control Simulations

This vehicle model was used in simulations with the inner-loop controller only, with the following results. Consider that the point-mass vehicle model can be derived by setting I_{yy} equal to zero. We would then expect the results to be somewhat different when I_{yy} is not zero. In our optimal control formulation of the problem, we have the condition $q^0 = -\frac{V\cos\gamma}{r_e+h}$. This nominal outer loop value does not work well in this case where, in the nominal trajectory, q achieves magnitudes in excess of 35°/sec. Since θ^0 is an analytic function of time, we can construct an analytic function of time for q^0. Using an analytic representation of q^0, integration of the full model with inner-loop only control results in errors in range, altitude, and velocity as presented in Figures 4.5, 4.6, and 4.7. Errors in θ, and the applied control β are presented in Figure 4.8. We see that the controller provides quite good tracking in this case, with a maximum relative altitude error at the end of less than 0.15%. This error is due to the inclusion of pitch moment of inertia in the equations, and the lag in the pitch attitude that this inclusion causes.

If we now impose a modeling error by offsetting the thrust line by 0.1 m, the resulting pitch equation is

$$\dot{q} = \frac{T}{I_{yy}}(0.1\cos\beta - 3.0\sin\beta) + \frac{V\cos\gamma}{r_e + h}.$$

Since we are not tracking anything but θ^0 and q^0, we expect to develop rather large errors with this modeling error. Open loop (open outer loop, closed inner loop) range error with this offset is presented in Figure 4.9, and we observe that the final error is over 2000 m. Altitude error is presented in Figure 4.10, and velocity error is presented in Figure 4.11. We note that the final relative altitude error approaches 35%.

The inner-loop controller effectively controlled the vehicle in the vicinity of the nominal trajectory when all the parameters were as modeled, but imposing a small thrust alignment deviation to the model causes very large final errors. Note that the closed-loop

system is stable, and is still very smooth in its motion. It should also be noted that a non-zero z does not affect the poles of the linearized closed loop system, only the accumulated error. It is clear that in the presence of modeling errors or modeling uncertainty there is a need for an outer loop controller to reduce the accumulation of error.

4.1.5 Simplified Model Outer-Loop Control

Both the Continuous LQR and the Discrete LQR were used with this model. Both outer loop controllers will be described, along with the behavior that they exhibit. The nominal trajectory is completely known for this model, so we may control whichever states we choose. With this simplified model we will control range x, altitude h, velocity V, and flight path angle γ, using pitch attitude θ and throttle η as controls. Nominally $\eta = 1$.

4.1.5.1 Continuous Linear Quadratic Regulator

The non-linear equations that were linearized for the regulator are

$$\begin{aligned}
\dot{x} &= V\cos\gamma, \\
\dot{h} &= V\sin\gamma, \\
\dot{V} &= \frac{T}{m}\cos(\theta-\gamma) - g\sin\gamma, \\
\dot{\gamma} &= \frac{T}{mV}\sin(\theta-\gamma) - \frac{g}{V}\cos\gamma + \frac{V\cos\gamma}{r_e+h}.
\end{aligned}$$

The matrix $\mathbf{A}$ was formed from $\frac{\partial \mathbf{f(x,u)}}{\partial \mathbf{x}}$, which evaluates as

$$\begin{pmatrix}
0 & 0 & \cos\gamma & -V\sin\gamma \\
0 & 0 & \sin\gamma & V\cos\gamma \\
0 & \frac{2g\sin\gamma}{r_e+h} & 0 & \frac{T\sin(\theta-\gamma)}{m} - g\cos\gamma \\
0 & \frac{2g\cos\gamma}{V(r_e+h)} - \frac{V\cos\gamma}{(r_e+h)^2} & -\frac{T\sin(\theta-\gamma)}{mV^2} + \frac{g\cos\gamma}{V^2} + \frac{\cos\gamma}{r_e+h} & -\frac{T\cos(\theta-\gamma)}{mV} + \frac{g\sin\gamma}{V} - \frac{V\sin\gamma}{r_e+h}
\end{pmatrix}. \tag{4.7}$$

The control matrix $\mathbf{B}$ was constructed from $\frac{\partial \mathbf{f(x,u)}}{\partial \mathbf{u}}$, and evaluates as

$$\begin{pmatrix}
0 & 0 \\
0 & 0 \\
-\frac{T\sin(\theta-\gamma)}{m} & \frac{\partial T}{\partial \eta}\frac{\cos(\theta-\gamma)}{m} \\
\frac{T\cos(\theta-\gamma)}{mV} & \frac{\partial T}{\partial \eta}\frac{\sin(\theta-\gamma)}{mV}
\end{pmatrix}.$$

Thrust in this model is given by $T = T_0 * \eta$, so $\frac{\partial T}{\partial \eta} = T$. $\mathbf{B}$ then becomes

$$\begin{pmatrix} 0 & 0 \\ 0 & 0 \\ -\frac{T\sin(\theta-\gamma)}{m} & \frac{T\cos(\theta-\gamma)}{m} \\ \frac{T\cos(\theta-\gamma)}{mV} & \frac{T\sin(\theta-\gamma)}{mV} \end{pmatrix} . \tag{4.8}$$

The weighting matrices $\mathbf{Q}$ and $\mathbf{R}$ were diagonal in this problem (this selection was arbitrary). The diagonal elements of these matrices were generated by a FORTRAN program, stored in a data file, and were adjusted until acceptable final errors and acceptable control behavior were simultaneously achieved. Conventional methods for constructing weights, such as using the inverse of the allowable squared error, were useful for starting conditions in the process of choosing acceptable weights.

The steady state Riccati equation was solved at each 0.1 second interval along the trajectory using Potter's method[17]. Gains were then computed as by (3.11). Trajectory simulations were computed using the gains generated in this way.

4.1.5.2 Continuous Regulator Control Simulations

Three weight sets were used in the simulations with the Continuous *LQR*. We will designate these sets as `ws1`, `ws2`, and `ws3`. These weights were generated as follows. The weight at time zero is the same as the weight at 0.1 second. For `ws1` the values in the $\mathbf{R}$ matrix were constant. The weight on $\Delta\theta$ was 1.0, and the weight on $\Delta\eta$ was 0.1. The weights on Δx and Δh were given by $\frac{0.001}{t^2}$, the weight on ΔV was $\frac{0.01}{t^2}$, and the weight on $\Delta\gamma$ was $\frac{0.1}{t}$. Entries in `ws2` were the same as `ws1` to $t = 5.0$ sec, after which the weights on Δx, Δh, and ΔV were held constant. The only change from `ws2` to `ws3` was to change the weight on $\Delta\eta$ to 100.

This type of weighting was motivated by an observation that acceptable errors in the states would reasonably be roughly proportional to the elapsed time along the trajectory. Of true importance in the weights is the ratio of the control weights to those

of the states. In `ws1` and `ws2` deviations in θ are weighted more heavily than deviations in η, while in `ws3` the opposite is true. This weighting bias was motivated completely by prejudice. This being the case, the weighting functions were adjusted until it appeared that the gains would be usable.

Simulations with these weightings were computed with both nominal parameters and with a normal thrust offset. Trajectory errors produced by the three weight sets, with all parameters at their nominal values, are presented in Figures 4.12–4.18. Required controls are presented in Figures 4.19 and 4.20. It is observed that there is little apparent difference in the behaviors with the three weight sets.

Trajectory errors with a 0.1 meter lateral thrust offset are presented in Figures 4.21–4.29. It is observed that all three weight sets produce about the same final mass error, but that the intermediate behavior is noticeably different. We may note that the mass deviation using `ws3`, which more heavily weights the use of throttle, is greater initially than for the other two weight sets. It is also noticed that overall `ws1` uses slightly more fuel than either `ws2` or `ws3`. We can observe that there is little difference in final results between `ws2` and `ws3`. Of more concern to us should be the large discrepancies that appear between the accumulated mass error, Figure 4.21, and the thrust variation, Figure 4.29. We observe that the commanded η for `ws1` and `ws2` is everywhere shown to be negative, but the mass error, which is the actual simulated weight less the nominal weight, is also negative, which implies either integration error, or some other throttle usage that is not being graphed.

The integration routine used for these simulations is a 5^{th} order *Runge-Kutta-Fehlberg*[18] variable time step algorithm. With inner-loop control only, the actual time step used by the algorithm very quickly approaches the maximum allowed, which is the output time step of 0.1 second. With the combined control the integration time step

collapsed to very small values, even with relatively large error bounds. Further analysis of this phenomena resulted in the discovery that the control η was oscillating, with both positive and negative values during every output time interval.

This behavior is a specific symptom of a more general problem that was encountered on the project. This problem has to do with the precision with which the nominal data is recorded, and the way that it is interpolated. It was discovered early in this investigation that much better integration results were achieved if the nominal trajectory was stored in full double precision (approximately 16 decimal digits) in a binary file. In the initial work with the inner-loop only control, and then more so with combined inner and outer loop control, significant control fluctuations were observed that were traced to the representation of the nominal states and controls to five significant digits in an ASCII file. From Figure 4.13–4.15 we may observe that even with our interpolation difficulties, range is tracked to within 0.7 m absolute error, and altitude and velocity are tracked to within 0.1 m and 0.06 m/s respectively. Consider that with a terminal range of 116,000 m, five decimal digits of resolution in the nominal would present the controller with a level of uncertainty in the nominal that is more than an order of magnitude greater than the error that might easily be controlled. When the representation of the nominal trajectory was changed to full double precision representation, the major fluctuations disappeared.

In the simplified problem with outer-loop control, the nominal states and controls were read from a binary file, and stored in a double precision array. A nominal representation of the pitch rate q was constructed from the nominal θ by a finite difference approximation, and also stored in an array. During the actual integration the intermediate values of the nominal states were estimated by linear interpolation of the stored values. While the data are relatively close together, and a linear interpolation would be thought to be sufficient, in this case it results in an unacceptable amount of control ac-

tivity. This interpolation error causes significant control activity because the interpolated nominal states and nominal controls are not kinematically consistent in the interpolated interval. The interpolation errors enter into the control calculations and induce spurious excitement of the controls that are not consistent with the actual trajectory errors.

To demonstrate the improvement that is possible with a better interpolation of both the nominal states and controls, simulations were computed with simultaneous integration of the nominal point-mass equations and the full order controlled equations. In effect we have an exact (to the finite precision resolution of the computer) interpolation of the nominal trajectory. Nominal values for both θ and q are determined analytically. Results of these simulations for the three weight sets are presented in Figures 4.30–4.38. Comparing Figure 4.30 to Figure 4.12 indicates that the total mass flow is not appreciably affected by the change in interpolation. A comparison of Figures 4.13–4.15 with Figures 4.31–4.33 respectively shows that the overall accuracy of the trajectory following improves markedly with the improved interpolation. Comparison of Figures 4.16–4.19 with Figures 4.34–4.37 respectively shows very little difference. Figures 4.20 and 4.38 show the most pronounced difference, with the control presented in Figure 4.38 being consistent with Figure 4.30. This consistent behavior is much more indicative of a smooth, non-oscillatory control.

An alternate method of suppressing the oscillatory behavior, at least as it affects the outer control is to impose a sampled control on the outer loop. It is felt that the discrete controller is a logical choice for the outer-loop, since it is likely that a control algorithm of this sort would be implemented with a digital computer, so the control would be discrete in reality, even if not in theory. In the construction of a discrete controller, the sample rate of the controller is considered, and the stability of the closed loop system is based on this sample rate. Imposing a discrete update schedule on a nominally continuous

controller can have devestating effects on the stability of the closed loop system.

4.1.5.3 Discrete Linear Quadratic Regulator

The Discrete LQR was constructed as described in Section 3.2, using the $\mathbf{A}$ and $\mathbf{B}$ matrices developed for the continuous LQR. The state transition matrix $\mathbf{\Phi}$ was constructed by exponentiation of $\mathbf{A}\Delta t$, rather than by integration of $\mathbf{\Phi}$ from $\mathbf{\Phi_0} = \mathbf{I}$.

The discrete gains were generated in two forms, as steady state gains at each point on the trajectory and as a time varying regulator with the final gains determined as steady state gains. These two methods produce gains that are very different during the early part of the trajectory, but result in trajectories that have almost indistinguishable final errors. The same weight sets as used for the Continuous LQR were used for the Discrete LQR. The discrete gains were based on a time step of 0.1 second. The steady-state discrete equation was solved by use of a non-recursive eigenvalue decomposition algorithm (a discrete analog to Potter's method) presented in Anderson.[19]

A comparison of the gains generated by the steady-state regulator method and the time-varying regulator method is presented in Figures 4.39–4.42. The gains are the same at the final time since the end condition for the time-varying regulator is the steady-state regulator at final time. The gains at previous points on the trajectory are different.

Simulations using the three weight sets with both types of gains were computed. Errors incurred in the simulations with steady-state type weights and a lateral thrust offset of 0.1 m are presented in Figures 4.43–4.49. Comparisons of these errors to the Continuous LQR errors of Figures 4.21–4.27 indicates that the move to the discrete outer loop controller has relatively little effect on the accumulated error. The feedback controls are presented in Figures 4.50–4.52. It should be noted that the maximum β in the discrete controller is significantly larger than the maximum value in the continuous controller. The control also appears to take longer to reach a steady value, and approaches the steady

value from the opposite direction. Whether this is in fact the case is difficult to tell, as inner loop control operates on a much faster time scale than the time update (and the print output rate), so possibly much of the inner loop control activity is not displayed in the figures.

The outer loop controls, $\Delta\theta_c$ and $\Delta\eta$, are zero-order-hold controls with the discrete LQR, so the values output at the time updates are the true values over the update interval. We therefore expect the slope of the mass error to be proportional to the negative of the applied control $\Delta\eta$. This fact is born out in Figures 4.43 and 4.52.

We see that we can follow an arbitrary trajectory with very little accumulated error with either the continuous inner and outer loop controls, or the continuous inner and discrete outer loop controls. There is relatively little overall difference in the errors for the two controllers, while there may be a significant computational advantage to using a discrete controller during flight.

It should be noted that while we claim that our inner loop controller is in certain respects an *optimal* controller, no claims of optimality can be made for this controller in this example. This lack of optimality is precisely because we have chosen to use a purely arbitrary nominal trajectory that is not in any reasonable sense optimal. Therefore it would be specious to then claim that our controller was optimal. Our claim here is that this controller is very effective at stabilizing this example vehicle, and provides very good trajectory following behavior.

4.2 NASA LaRC Hypersonic Conical Accelerator

The second vehicle simulation in this study used the Winged-Cone Configuration Hypersonic Vehicle Simulation Model from NASA Langley Research Center (LaRC). A two-view drawing of the vehicle is presented in Figure 4.53. This vehicle is a generic horizontal-takeoff, single-stage to orbit (SSTO) aircraft with airbreathing engines. The

aerodynamic model is generic, and does not model an actual current vehicle, so some liberties can be taken with certain aerodynamic limits. The vehicle model is presented as tabular data and comprises the requisite information necessary to construct a full six degree-off-freedom simulation of the aircraft. As the present effort has to date only involved vertical plane motion, a significant portion of the model was neglected.

Documentation for the model[20], a preliminary prepublication version, indicates that the data are constructed so that good simulation results can be achieved while using linear interpolation of the data.

The nominal trajectory used in this part of this study was the result of an optimization simulation with the program *POST*. The trajectory data are provided approximately every ten seconds, except at flight phase changes where there are intermediate points provided. In total, two-hundred data points along the trajectory are provided. This trajectory begins at an altitude of 39 ft at 0.4 Mach heading north from Cape Canaveral, Florida, and ends 1943 seconds later at an altitude of 681,356 ft at 27.1 Mach over southern Siberia.

Main engine thrust is dependent on dynamic pressure and Mach number, and shuts off at 1350 seconds, where the Mach number is 24.2 and the dynamic pressure is rapidly falling through 172 psf. Dynamic pressure very quickly goes to virtually zero, and the modeled control system loses all effectiveness. Because of this condition the simulations in this effort are ended at 1350 seconds into the flight. The nominal altitude at this point is 199,082 ft.

4.2.1 Inner Loop Implementation

The inner loop gains were computed at each point on the nominal trajectory at which data are provided. This provided gains about every ten seconds. These gains were linearly interpolated between the computed points.

From (2.25) and its supporting definitions we find that we need to estimate a

number of aerodynamic derivatives to be able to determine the optimal inner loop gains. The aerodynamic data that are provided are in coefficient form and referenced to a fixed geometric point on the vehicle. We require first partial derivatives of the aerodynamic moment M with respect to the pitch rate q, the pitch attitude θ, and the elevator deflection δ. We see that holding γ constant and varying θ is exactly the same as simply varying the angle of attack α. By extension, all partial derivatives with respect to θ may be treated as partial derivatives with respect to α. The only aerodynamic data involving q is an explicit moment derivative $\frac{\partial C_M}{\partial q}$. We do not model lift or drag as functions of q. We need first and second derivatives of lift L and drag D with respect to θ and δ.

This vehicle model also incorporates a retractable Canard that is used at Mach numbers less than 0.9. As no nominal Canard deflections are provided, and the vehicle exceeds 0.9 Mach in less than twenty seconds, the Canard is ignored in this analysis.

The data provided for the model are such that the complete representation of the drag coefficient C_D is given by

$$C_D = C_{Da} + C_{D,de} + C_{D,da} + C_{D,dr} + C_{D,dc},$$

and the drag D is given by

$$D = \bar{q} S_{\text{ref}} C_D.$$

The term C_{Da} is the basic drag presented as a function of Mach number and α. This term is presented in a two dimensional table of values. The terms $C_{D,da}$ and $C_{D,de}$ are incremental drags for right and left elevon respectively, and are presented as functions of Mach, α, and δ. As we are only interested here in symmetric flight, we assume that these will always be the same, so only one table was used and all values were doubled. $C_{D,dr}$ is the increment for rudder, but since we are in symmetric flight, rudder deflection is zero and this term is ignored. Canard drag $C_{D,dc}$ is also ignored. This leaves only two terms in the expression for C_D: C_{Da} and $C_{D,de}$. C_L is provided in a similar set of tables,

with the respectively same final terms being retained. C_M is provided in the same group of tables, plus the additional pitch rate table. It should be noted that the derivatives needed in equation 2.25 are dimensional derivatives.

Estimation of the derivatives that are needed is done by finite differences in the α and δ dimensions of the tables. These estimates are then linearly interpolated in the Mach direction. No derivatives in the Mach direction are needed in the inner loop. The one dimensional finite difference approximation used a four point finite difference to estimate the functional value and the first and second derivative. The two dimensional derivatives are constructed by multiple applications of the one dimensional approximation, followed by a one-dimensional finite difference approximation on the previous results.

For two dimensional tables, where the data are in terms of Mach and α, two four-point finite difference approximations are required, one for each of the closest two Mach values. Each of these approximations involves the solution of three four-by-four Vandermonde-type systems of equations. This solution provides a function value and a first and second derivative.

For three dimensional tables, where the data are in terms of Mach, α, and δ, twelve four-point finite difference approximations are required. With these calculations, a function value, two first derivatives and three second derivatives are provided.

We have the drag coefficient C_D as

$$C_D = C_{Da}(\text{Mach}, \alpha) + 2 * C_{D,de}(\text{Mach}, \alpha, \delta),$$

and

$$D = \overline{q} S_{\text{ref}} C_D,$$

so

$$\frac{\partial D}{\partial \theta} = \overline{q} S_{\text{ref}} \left[\frac{\partial C_{Da}}{\partial \alpha} + 2 * \frac{\partial C_{D.de}}{\partial \alpha} \right];$$

$$\frac{\partial^2 D}{\partial \theta^2} = \overline{q} S_{\text{ref}} \left[\frac{\partial^2 C_{D\alpha}}{\partial \alpha^2} + 2 * \frac{\partial^2 C_{D.de}}{\partial \alpha^2} \right];$$
$$\frac{\partial^2 D}{\partial \theta \partial \delta} = 2 * \overline{q} S_{\text{ref}} \frac{\partial^2 C_{D,de}}{\partial \alpha \partial \delta};$$
$$\frac{\partial^2 D}{\partial \delta^2} = 2 * \overline{q} S_{\text{ref}} \frac{\partial C_{D,de}}{\partial \delta^2}.$$

The required lift derivatives are constructed in the same way. The moment derivative C_M is

$$C_M = C_{M\alpha}(\text{Mach}, \alpha) + 2 * C_{M,de}(\text{Mach}, \alpha, \delta) + C_{M_q} \left(\frac{qc}{2V} \right).$$

The term $\frac{qc}{2V}$ is the non-dimensional pitch rate. The aerodynamic moment about the moment reference center (mrc) is

$$\overline{M}_{\text{mrc}} = \overline{q} c S_{\text{ref}} C_M.$$

The total moment about the center of gravity (cg) is given by

$$\overline{M} = \overline{M}_{\text{mrc}} - x_{\text{cg}} Z, \tag{4.9}$$

where Z is the z-axis force and is given by

$$Z = -D \sin \alpha - L \cos \alpha. \tag{4.10}$$

This configuration has no pitching moment due to the x-axis force as the cg and the moment reference center are both located on the x-axis.

The moment derivative $\frac{\partial M}{\partial q}$ then is given by

$$\frac{\partial \overline{M}}{\partial q} = \frac{\partial \overline{M}_{\text{mrc}}}{\partial q} = \overline{q} c S_{\text{ref}} \frac{\partial C_M}{\partial q},$$
$$= \overline{q} c S_{\text{ref}} C_{M_q} \left(\frac{c}{2V} \right),$$

which finally gives

$$\frac{\partial \overline{M}}{\partial q} = \frac{\overline{q} c^2 S_{\text{ref}}}{2V} C_{M_q}.$$

C_{M_q} was provided in the form of a two-dimensional table as a function of Mach and α. As no further derivatives of this term are required this term is evaluated by linear interpolation of the table.

From equation (4.9) and (4.10) we infer that

$$\begin{aligned}\frac{\partial \overline{M}}{\partial \alpha} &= \frac{\partial \overline{M}_{\text{mrc}}}{\partial \alpha} - x_{\text{cg}} \frac{\partial Z}{\partial \alpha} \\ &= \left[\frac{\partial C_{Ma}}{\partial \alpha} + 2\frac{\partial C_{M,de}}{\partial \alpha}\right] \bar{q} c S_{\text{ref}} + x_{\text{cg}} \left[\frac{\partial D}{\partial \alpha} \sin\alpha + \frac{\partial L}{\partial \alpha} \cos\alpha + D\cos\alpha - L\sin\alpha\right] \\ &= \left[\frac{\partial C_{Ma}}{\partial \alpha} + 2\frac{\partial C_{M,de}}{\partial \alpha}\right] \bar{q} c S_{\text{ref}} + x_{\text{cg}} \left[\left(\frac{\partial D}{\partial \alpha} - L\right) \sin\alpha + \left(D + \frac{\partial L}{\partial \alpha}\right) \cos\alpha\right],\end{aligned}$$

and

$$\begin{aligned}\frac{\partial \overline{M}}{\partial \delta} &= \frac{\partial \overline{M}_{\text{mrc}}}{\partial \delta} - x_{\text{cg}} \frac{\partial Z}{\partial \delta} \\ &= 2\frac{\partial C_{M,de}}{\partial \delta} \bar{q} c S_{\text{ref}} + x_{\text{cg}} \left[\frac{\partial D}{\partial \delta} \sin\alpha + \frac{\partial L}{\partial \delta} \cos\alpha\right].\end{aligned}$$

Two additional parameters that are dependent on the point along the trajectory are I_{yy} and x_{cg}. These values are provided in one-dimensional tables as functions of weight. They are linearly interpolated. The thrust level is also needed in several of the coefficients, and was read directly from the nominal trajectory.

It can be noted that the aerodynamic data are presented in a completely regular grid. That is, all of the two-dimensional data tables have the same Mach and α gridding, and all the three-dimensional tables have the same Mach, α, and δ gridding. Further the three-dimensional tables have the same Mach and α grid as the two-dimensional table. Because of this grid arrangement the actual computational requirement for estimating all the derivatives is the solution of six four-by-four Vandermonde-type systems of equations to generate interpolation coefficients.

An alternative to the use of finite differences for the derivative estimation is the use of cubic splines. It was determined that using cubic splines in this situation resulted in what was believed to be poorer estimates of the derivatives, and in this situation the

spline generation and evaluation required significantly more computational effort, and much more computer memory. For these reasons splines were not used here.

The linearization of the pitch equations of motion is

$$\begin{pmatrix} \Delta\dot{\theta} \\ \Delta\dot{q} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ \frac{1}{I_{yy}}\frac{\partial M}{\partial \theta} & \frac{1}{I_{yy}}\frac{\partial M}{\partial q} \end{pmatrix} \begin{pmatrix} \Delta\theta \\ \Delta q \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ -\frac{T\ell}{I_{yy}} & \frac{1}{I_{yy}}\frac{\partial M}{\partial \delta} \end{pmatrix} \begin{pmatrix} \Delta\beta \\ \Delta\delta \end{pmatrix}.$$

The open loop eigenvalues of this system are presented in Figure 4.54. Since the system is unstable over a large part of the trajectory, it is clear that some form of pitch stabilization control is necessary.

With estimates of the derivatives we may construct the Hamiltonian matrix. It has been shown that the eigenvalues of this matrix are independent of the value of K^0. It can be shown that the eigenvectors are also independent of the value of K^0. To provide the best balanced matrix that is reasonable, the value of K^0 that was used was determined to be the value such that $A'_{31} = A'_{24}$. This is accomplished by evaluating A'_{31} and A'_{24} without K^0 and then computing

$$K^0 = \sqrt{\frac{A'_{24}}{A'_{31}}},$$

and then

$$A'_{24} = \frac{A'_{24}}{K^0},$$

$$A'_{31} = A'_{31} * K^0 = A'_{24}.$$

Eigenvalues and eigenvectors of this matrix were computed with routines from the Eispack library. The eigenvectors are sorted into those associated with stable eigenvalues, and those associated with unstable eigenvalues. Under proper conditions exactly half the eigenvalues should be stable and half should be unstable. At several points on the trajectory this fails, as there are two real eigenvalues and two pure imaginary eigenvalues. If this situation occurs the computation of the gains at this point is skipped, and the

gains are simply interpolated across the gap. In our abbreviated trajectory, this condition occurs at times of 70, 80, and 90 seconds.

Estimates of the real part of the closed loop inner loop eigenvalues are presented in Figure 4.55. This system is second order so there are two eigenvalues; if the real components are distinct, then the roots are real and the system behavior is non-oscillatory. The closed loop roots are real for the entire trajectory except for a short period from about 10 seconds to 60 seconds, and from 1310 seconds to 1320 seconds. Note that these occurrences are not at the same times as when the Hamiltonian system has a wrong number of stable eigenvalues.

It should be noted that the closed loop stability analysis was developed using the same estimates of the aerodynamic derivatives that were used in the determination of the gains. It is therefore expected that the system will appear to be very stable. This analysis is really only as good as our ability to estimate the required derivatives. The real test of the gains is in the non-linear simulation (or an actual flight test).

Inner loop gains are presented in Figures 4.56–4.59. This trajectory involves a pull-up to climb at the beginning of the trajectory, a cruise-climb-accel segment tracking 2000 psf dynamic pressure from about 150 seconds to 1200 seconds, then a pull-up (zoom) to orbital altitude beginning at 1200 seconds. This vehicle is open loop unstable through the largest part of the trajectory, and requires relatively large negative elevator deflections to trim. As the Mach number rises the effectiveness of the elevator declines, and the gains increase to compensate. At 1200 seconds the zoom begins and the angle of attack increases from 0.2° to 1.9°. This attitude results in a much greater elevator pitch effectiveness, and the gains drop immediately. The angle of attack then drops to a negative value before increasing in a somewhat oscillatory manner toward 10° at 1340 seconds. Nominal angle of attack is presented in Figure 4.60. During this period from 1200 seconds

to 1340 seconds the altitude changes from 130,000 ft to 200,000 ft. The dynamic pressure drops from 2000 psf to 280 psf and the aerodynamic control becomes relatively ineffective. This reduction in effectiveness results in the gain generation algorithm reducing the gains, since use of the control is relatively ineffective. Over the same time interval the thrust is reduced from 105,000 lbs to 57,500 lbs which results in reduced thrust vector pitch control effectiveness, so those gains are also reduced. Very small variations in the nominal trajectory may have relatively large effects on the gains.

4.2.2 Outer Loop Implementation

A Continuous LQR was not implemented for the complex model that included aerodynamics. The discrete controller was implemented using the equations presented in Section 3.2, both in steady-state form and in time-varying form. While the simplified model controlled four states in the outer loop, the complex model nominal trajectory does not provide range, so only the three variables (h, V, and γ) are controlled in this implementation. Possibly better tracking could be obtained if a range variable was included in the nominal.

Since we only need to linearize the equations that will be controlled, there will be only three equations, which can be written as

$$\begin{pmatrix} \Delta\dot{h} \\ \Delta\dot{V} \\ \Delta\dot{\gamma} \end{pmatrix} = \begin{pmatrix} \frac{\partial\dot{h}}{\partial h} & \frac{\partial\dot{h}}{\partial V} & \frac{\partial\dot{h}}{\partial\gamma} \\ \frac{\partial\dot{V}}{\partial h} & \frac{\partial\dot{V}}{\partial V} & \frac{\partial\dot{V}}{\partial\gamma} \\ \frac{\partial\dot{\gamma}}{\partial h} & \frac{\partial\dot{\gamma}}{\partial V} & \frac{\partial\dot{\gamma}}{\partial\gamma} \end{pmatrix} \begin{pmatrix} \Delta h \\ \Delta V \\ \Delta\gamma \end{pmatrix} + \begin{pmatrix} \frac{\partial\dot{h}}{\partial\theta} & \frac{\partial\dot{h}}{\partial\eta} \\ \frac{\partial\dot{V}}{\partial\theta} & \frac{\partial\dot{V}}{\partial\eta} \\ \frac{\partial\dot{\gamma}}{\partial\theta} & \frac{\partial\dot{\gamma}}{\partial\eta} \end{pmatrix} \begin{pmatrix} \Delta\theta \\ \Delta\eta \end{pmatrix}.$$

Several techniques for estimating the elements of the linearized state and control matrices were evaluated. One technique involved computing the individual derivatives as in the inner-loop formulation. This procedure required repeated application of the chain-rule for several of the derivatives, and resulted in several very complex expressions which

can lead to errors. As an example, let us look at the term $\frac{\partial\dot{\gamma}}{\partial V}$.

$$\begin{aligned}\frac{\partial\dot{\gamma}}{\partial V} &= -\frac{T\sin(\theta-\gamma+\beta)+L}{mV^2}+\frac{g\cos\gamma}{V^2}+\frac{\cos\gamma}{r_e+h}+\frac{\partial T}{\partial V}\frac{\sin(\theta-\gamma+\beta)}{mV}+\frac{\partial L}{\partial V}\frac{1}{mV}\\ &= -\frac{T\sin(\theta-\gamma+\beta)+L}{mV^2}+\frac{g\cos\gamma}{V^2}+\frac{\cos\gamma}{r_e+h}\\ &\quad+\left(\overline{q}\frac{\partial C_T}{\partial M}\frac{\partial M}{\partial V}+\overline{q}\frac{\partial C_T}{\partial\overline{q}}\frac{\partial\overline{q}}{\partial V}+C_T\frac{\partial\overline{q}}{\partial V}\right)\frac{\sin(\theta-\gamma+\beta)}{mV}\\ &\quad+\left(\overline{q}S_{\text{ref}}\frac{\partial C_L}{\partial M}\frac{\partial M}{\partial V}+S_{\text{ref}}C_L\frac{\partial\overline{q}}{\partial V}\right)\frac{1}{mV}.\end{aligned}$$

This method is relatively efficient, but resulted in a system matrix with elements that changed with an almost random bent.

Another method is to construct finite difference derivatives of the actual non-linear equation. As an example, again consider $\frac{\partial\dot{\gamma}}{\partial V}$.

$$\frac{\partial\dot{\gamma}}{\partial V}=\frac{\dot{\gamma}(V+\Delta V)-\dot{\gamma}(V)}{\Delta V}.$$

It must be possible to evaluate the state equations at arbitrary states, so we may evaluate them along the nominal trajectory, then individually perturb each of the states and controls by a small amount, and construct the linearized state and control matrices with six evaluations of the state equations. In this evaluation all the table lookups are by linear interpolation. This method is more computationally intensive than the previous method, but it does result in a system matrix that varies somewhat more uniformly over time than by the previously described method.

Several outer loop update frequencies were simulated with very similar results. Update times of 0.5 seconds, 1.0 second and 2.0 seconds were used to test the sensitivity to the time step. An update time of 1.0 second was used for most of the simulations.

Numerous weight schedules were tried in an attempt to get a reasonably smooth trajectory that follows the nominal to within acceptable limits. Of the several weight schedules that were tried, the one that seems to provide the best overall performance is

given by the following equations.

$$Q(1,1) = \frac{\log\sqrt{\frac{t+2}{t+1}}}{100},$$

$$Q(2,2) = \log\sqrt{\frac{t+2}{t+1}},$$

$$Q(3,3) = 0.001,$$

$$R(1,1) = 10,$$

$$R(2,2) = 1.$$

Gains were generated as both steady-state gains at each time, and as time-varying gains as previously described. With this model the time-varying gains tended to vary excessively from one time step to the next in the latter parts of the trajectory while the steady-state gains seemed to be much more consistent. Both time-varying and steady-state gains are presented in Figures 4.61–4.63. Note that the steady-state and time-varying gains are very nearly indistinguishable over the largest part of the trajectory.

Simulations were computed with gains generated from numerous weight schedules and with several different schemes for interpolating state and control values along the nominal trajectory. At this point it is necessary to discuss the interpolation of the nominal trajectory itself. Recall that the nominal trajectory was provided at ten second intervals. This time interval is much too long to be used as the outer loop discrete control time step. Therefore, the nominal trajectory must be interpolated in some way to create intermediate nominal trajectory points to follow. It turns out that this problem is one of the more intractable problems that was encountered on this program.

Simulations require that the error in the inner loop be computed virtually continuously. The inner loop requires knowledge of the nominal θ, q, δ, and β. Unfortunately neither θ nor q are provided in the nominal trajectory. Since $\theta = \gamma + \alpha$ we can construct θ and q from γ, α, $\dot{\gamma}$, and $\dot{\alpha}$. We need these variables continuously along the trajectory, so we need them smoothly interpolated. Since we have γ and $\dot{\gamma}$, at each end of each

ten second interval, we can construct a unique cubic polynomial that satisfies these four boundary conditions. The same is true for α, so θ and $\dot{\theta}$ are available along the trajectory in this manner.

A problem that becomes apparent when this procedure is done over the entire trajectory is that solving the equations in this way produces a trajectory of γ and θ that is not truly smooth. What is produced is a curve that has a very slight waviness. This waviness is not visible in a plot of the curves but it causes increasingly large oscillations in altitude and velocity errors on the trajectory. Two ways that are available to eliminate the problem are to add to the $\dot{\gamma}$ term at each point in the interpolation equation a quantity corresponding to the rotation rate of the coördinate system, and to linearly interpolate the data when this waviness becomes a problem. From a computational perspective the linear interpolation route is better, and it also results in a smoother trajectory, so it was used for this problem.

The nominal value of β is everywhere zero since the nominal trajectory does not use thrust vectoring for control. The nominal value of δ is more difficult to construct since δ is a control and does not need to be continuous. The nominal δ was linearly interpolated for want of an obvious better alternative.

It should be noted that this nominal control will not be consistent with the nominal states over each time interval. That is, integration of the equations of motion from initial conditions with this control will not in general result in achieving the nominal conditions at the next given data point. To achieve better trajectory following would require that the nominal states and controls be more consistent.

This points up a deficiency in the nominal trajectory as supplied by Nasa LaRC. The supplied trajectory was never intended to be a nominal for a numeric simulation, and is not nearly dense enough for a good simulation. For accurate trajectory following

the nominal values of the outer loop states and controls should be provided at the points where the discrete updates are made. It should be acceptable to provide the inner loop states and controls at the same points.

With these limitations in mind, results for two simulations will be presented. The first simulation is with a still atmosphere. Figures 4.64–4.66 present altitude, velocity, and γ errors at every ten seconds, which approximately corresponds to the times in the nominal trajectory. These figures indicate that the error is relatively small at the nominal trajectory points, which we would hope would be acceptable. Figures 4.67–4.69 present the same variable errors at a time interval of every one second. These figures present the errors that are driving the outer loop control system. This unwanted excitement of the controls demonstrates that there are serious deficiencies in the interpolation of the nominal trajectory, but even so the controller is able to respectably follow the nominal. It is hypothesized that given a better representation of the nominal trajectory, a much smoother and more accurately controlled trajectory would result.

The primary purpose of developing this control algorithm is to construct a pitch control algorithm that minimizes the fuel penalty imposed by the controls. The weight deviation for this simulation is presented in Figure 4.70. This figure indicates that this control algorithm requires less fuel flow until just past seven hundred seconds into the trajectory. This conclusion is in some contrast to the throttle deviation presented in Figure 4.71. This figure indicates that the throttle control is slightly higher than nominal for most of the trajectory. This discrepancy points up the difficulty in generating accurate comparisons with disparate integration routines. Determination of the actual effectiveness of this control algorithm would require simulations using the same integration routine.

The other outer loop control, the commanded pitch attitude deviation $\Delta\theta$, is presented in Figure 4.72. These two controls are outer loop discrete controls and are

affected by the interpolation problems previously discussed.

The inner loop controls β and δ are presented in Figures 4.73–4.74. The figures indicate the same interpolation problems as the outer loop control. Observing the behavior of the control over each one second interval indicates that the control quickly approaches zero after each large jump. It is felt that a better interpolation scheme, or a denser nominal trajectory would dramatically reduce the control activity.

Altitude, velocity, and flight path angle errors with a one-hundred fps vertical gust starting at 700 seconds and continuing to 880 seconds are presented in Figures 4.75–4.77. These figures indicate that this gust has no significant affect on the trajectory. The weight error is not noticeably different than with the still atmosphere.

5.0 CONCLUDING REMARKS

We have developed an inner loop control algorithm that is reasonably computationally efficient and produces a controller that generates a very stable closed loop system. In the event that the nominal trajectory was constructed as a singular perturbation reduced order trajectory, the inner loop feedback gains are optimal. Nominal trajectories constructed by some alternative method result in inner loop gains that are sub-optimal. These gains may be far from the optimal, but the gains still generate a stable pitch mode controller.

The pitch mode controller is developed with no tunable parameters, that is, the performance index is derived directly from the full order optimal control problem. This derived performance index generates a nominally optimal pitch controller. In the two implementations presented, the singular perturbation pitch control algorithm produced a closed loop controller that was very stable. The gains developed for this controller are much higher than what would be expected from a more classical closed loop controller. Because of these high gains, the singular perturbation pitch mode controller was very sensitive to errors in the estimate of the nominal trajectory, and it is expected that it would also be sensitive to errors in the state estimates of an actual vehicle.

A possible remedy for these concerns might be the development of a stochastic singular pertubation controller. Whether this is feasible is not known to the author. This approach may be similar to the multi-loop optimal output control developed by Calise.[21]

The outer loop control algorithms described herein provide stable closed loop trajectory following controllers. We make no claim that these controllers provide optimal control in the sense of minimizing the original performance index, but these controllers are tunable so that the dynamic behavior of the system can be made acceptable. The effectiveness of these controllers is also dependent on the accuracy of the estimation of the

the nominal trajectory, and the accuracy of the estimation of the actual vehicle state. It is believed that the outer loop control has relatively little overall impact on the measured performance of the complete system.

REFERENCES

[1] Adams, N. J., Hattis, P. D., "An Integrated Configuration and Control Analysis Technique for Hypersonic Vehicles," *Proceedings of the American Control Conference*, June, 1989.

[2] Hattis, P. D., Smolskis, R. K., "Optimal Trajectory Generation and Design Trades for Hypersonic Vehicles," *Proceedings of the American Control Conference*, June, 1989.

[3] Grallert, H. K. H., Furniss, S. G., Hewitt, F. A., "An Investigation of Future European Winged Launcher Concepts," *International Astronautical Congress, IAF Paper 89-216*, October, 1989.

[4] Furniss, S. G., "Single-Stage-To-Orbit Design Optimization," *International Astronautical Congress, IAF Paper 89-222*, October, 1989.

[5] **Плохих В. П.**, "Sensitivity Analysis of Single-Stage-to-Orbit Reusable Vehicle Parameters," *International Astronautical Congress, IAF Paper 89-223*, October, 1989.

[6] Kelley, H. J., "Aircraft Manœuver Optimization by Reduced-Order Approximation," *Control and Dynamic Systems*, C. T. Leondes (ed.), Vol 10, Academic Press, New York, 1973.

[7] Murthy, S. N. B., Czysz, P., "Energy Analysis of High Speed Vehicle Systems," *28^{th} Aerospace Sciences Meeting, AIAA Paper 90-0089*, January, 1990.

[8] Corban, J. E., Calise, A. J., Flandro, G. A., "Optimal Guidance and Propulsion Control for Transatmospheric Vehicles," *AIAA Guidance, Navigation, and Control Conference, AIAA Paper 89-3617*, August, 1989.

[9] Calise, A. J., Flandro, G. A., Corban, J. E., "A Real-time Guidance Algorithm For Aerospace Plane Optimal Ascent To Low Earth Orbit," *Proceedings of the*

American Control Conference, Vol 3, June, 1989.

[10] Calise, A. J., Flandro, G. A., Corban, J. E., *Trajectory Optimization and Guidance Law Development For National Aerospace Plane Applications, Final Report, NASA-CR-182994*, December, 1988.

[11] Wasow, W., *Asymptotic Expansions for Ordinary Differential Equations*, Interscience, New York, 1965.

[12] Ardema, M. D., "Singular Perturbations in Systems and Control," *International Center for Mechanical Sciences (CISM) Course and Lectures No. 280*, Springer-Verlag, New York, 1983.

[13] Ardema, M. D., *Singular Perturbations in Flight Mechanics, NASA TM X-62,380*, 1977.

[14] Bryson, A. E., Ho, Y. C., *Applied Optimal Control*, Hemisphere, New York, 1975.

[15] Kirk, D. E., *Optimal Control Theory*, Prentice-Hall, Inc., Englewood Cliffs, N. J., 1970.

[16] Cliff, E. M., "A Singular Perturbation Approach to Pitch-Loop Design," *Proceedings of the American Control Conference*, June, 1990.

[17] Maine, R. E., Iliff, K. W., *User's Manual for MMLE3, a General FORTRAN Program for Maximum Likelihood Parameter Estimation, NASA Technical Paper 1563*, 1980.

[18] Gerald, C. F., Wheatley, P. O., *Applied Numerical Analysis*, Addison-Wesley Publishing Company, Inc., Reading, MA., 1985

[19] Anderson, B. D. O., Moore, J. B., *Optimal Filtering*, Prentice-Hall, Inc., Englewood Cliffs, N. J., 1979.

[20] Shaughnessy, J. D., et. al., *Hypersonic Vehicle Simulation Model: Winged-Cone Configuration (Preliminary)*, NASA Langley Research Center, 1990.

[21] Calise, A. J., Jonnalagadda, R. P., Siciliano, B., "Optimal Output Feedback in Two-Time Scale Systems," *Proceedings of the 26th IEEE Conference on Decision and Control, Vol 2*, December, 1987.

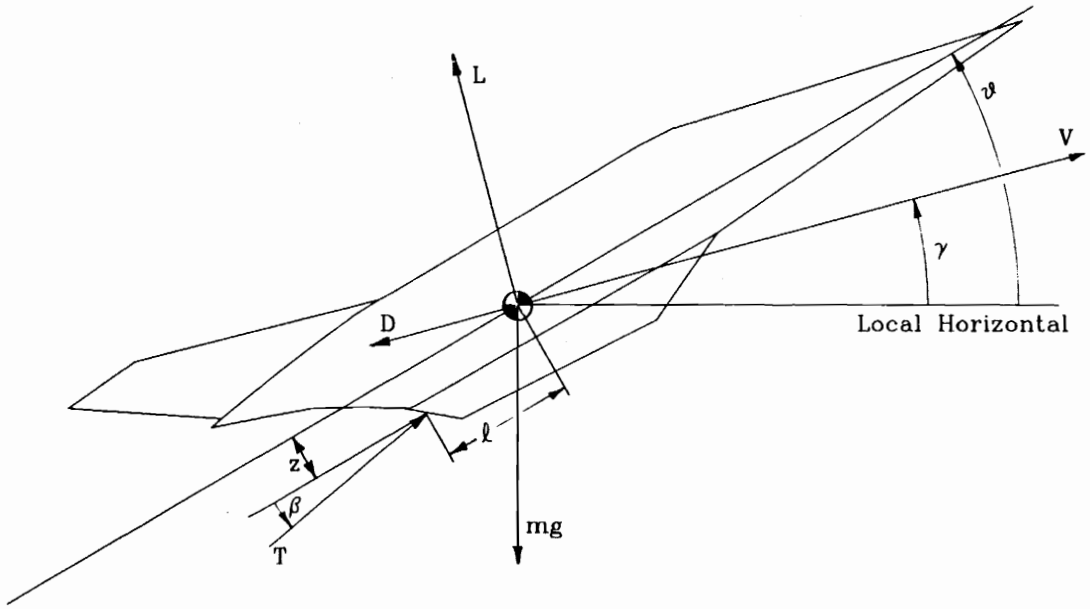

Figure 2.1 Aerospace Vehicle General Geometry

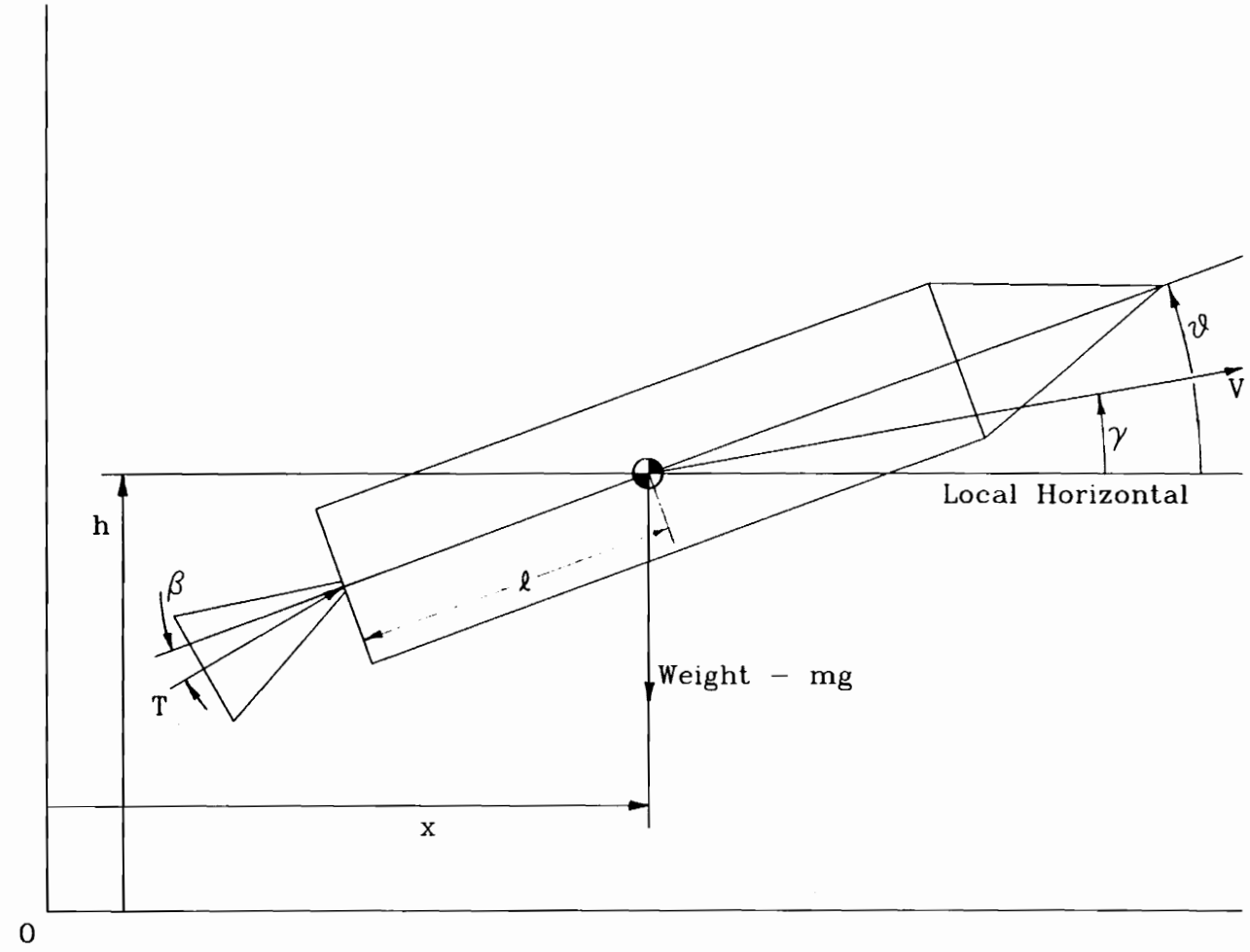

Figure 4.1 Simplified Model Configuration

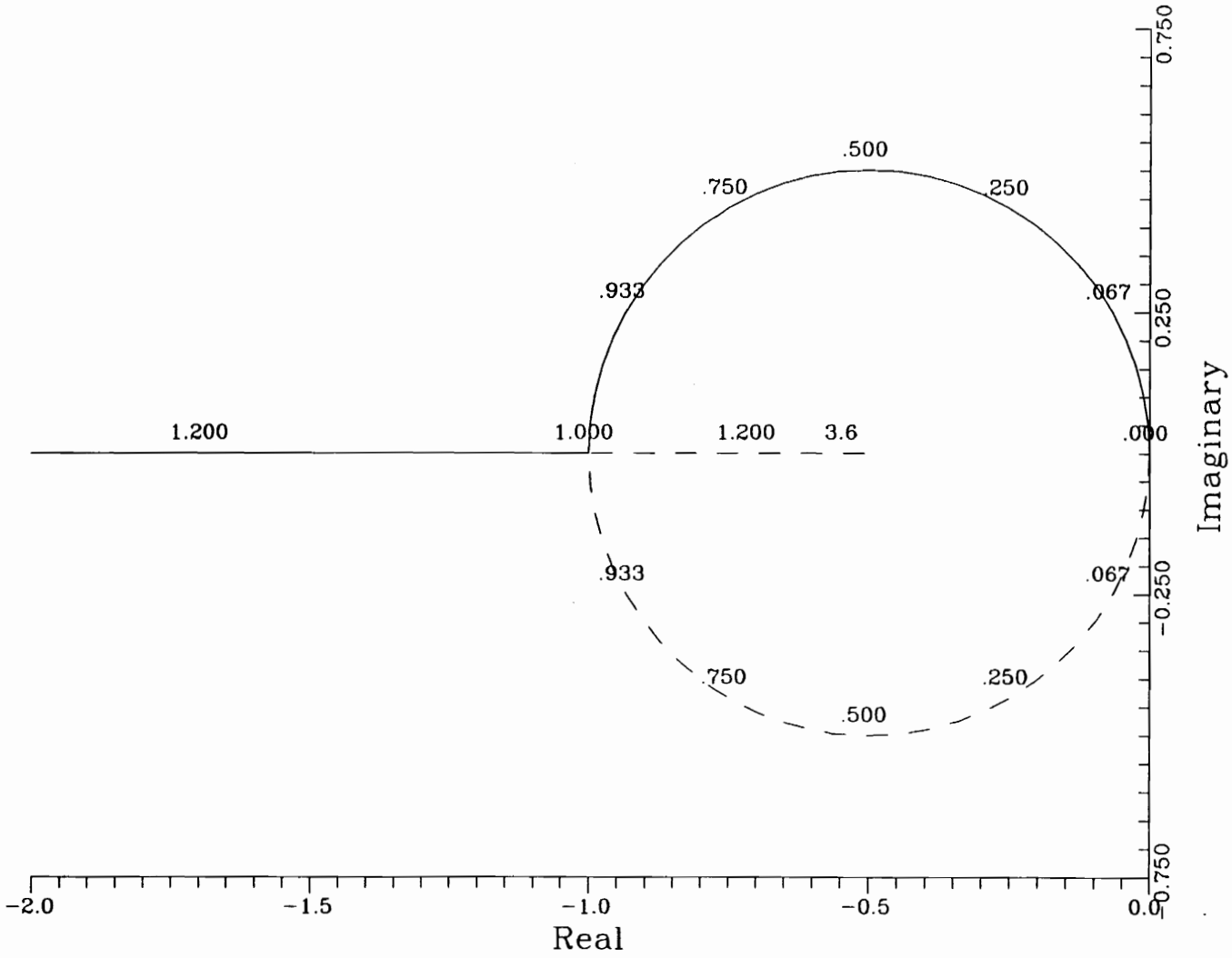

Figure 4.2 Pole Travel With Control Effectiveness Parameter Change

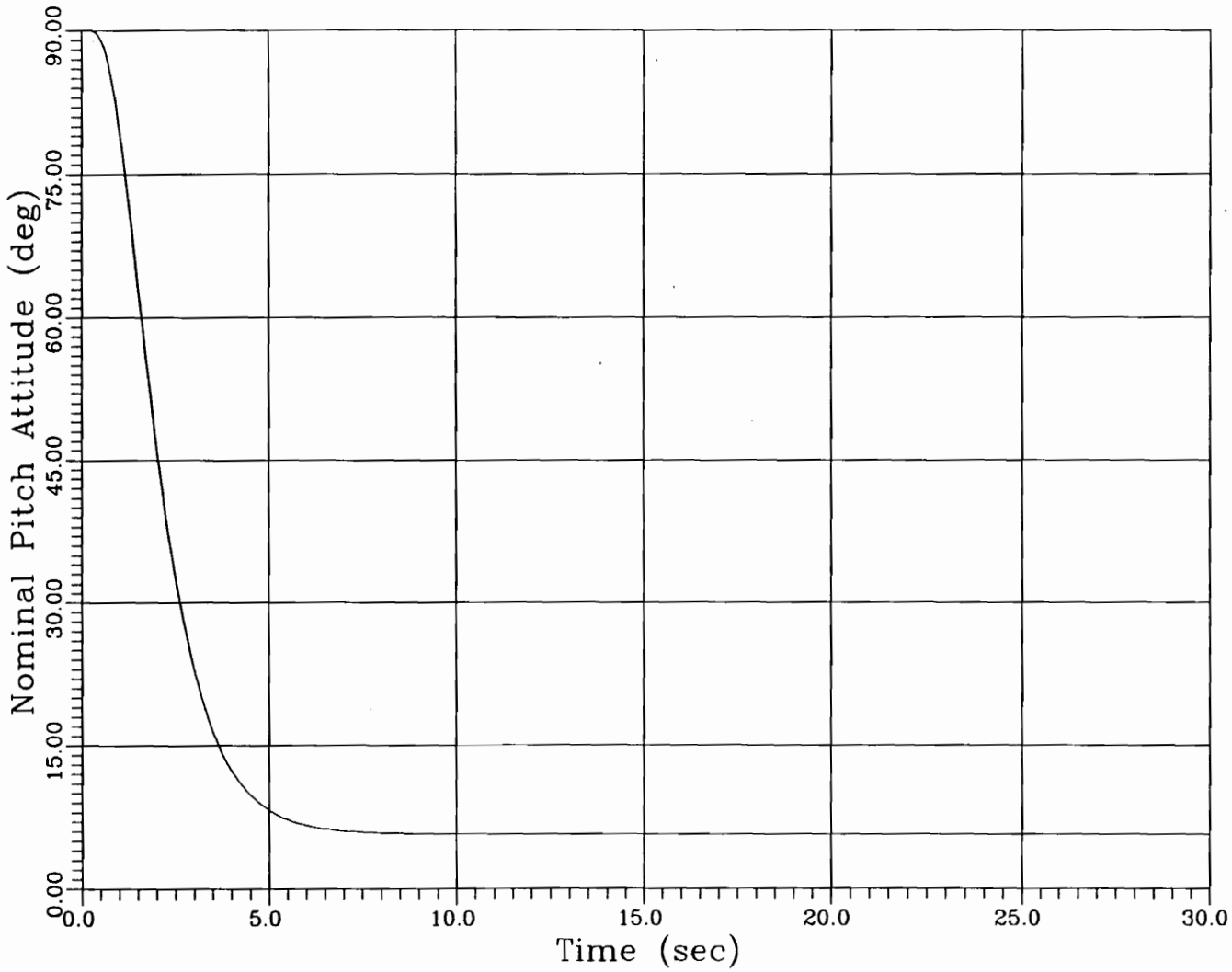

Figure 4.3 Nominal Pitch Attitude (θ)

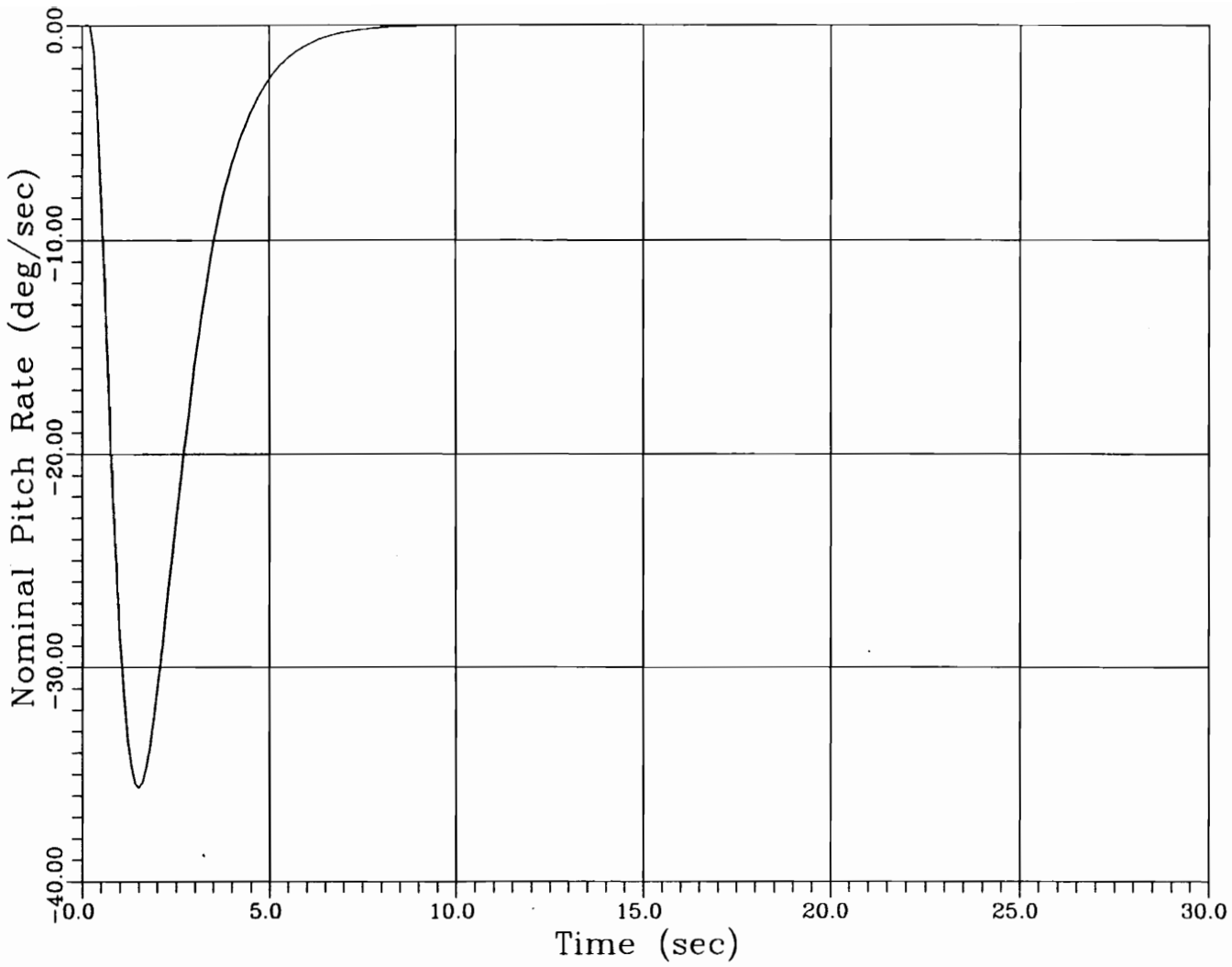

Figure 4.4 Nominal Pitch Rate (q)

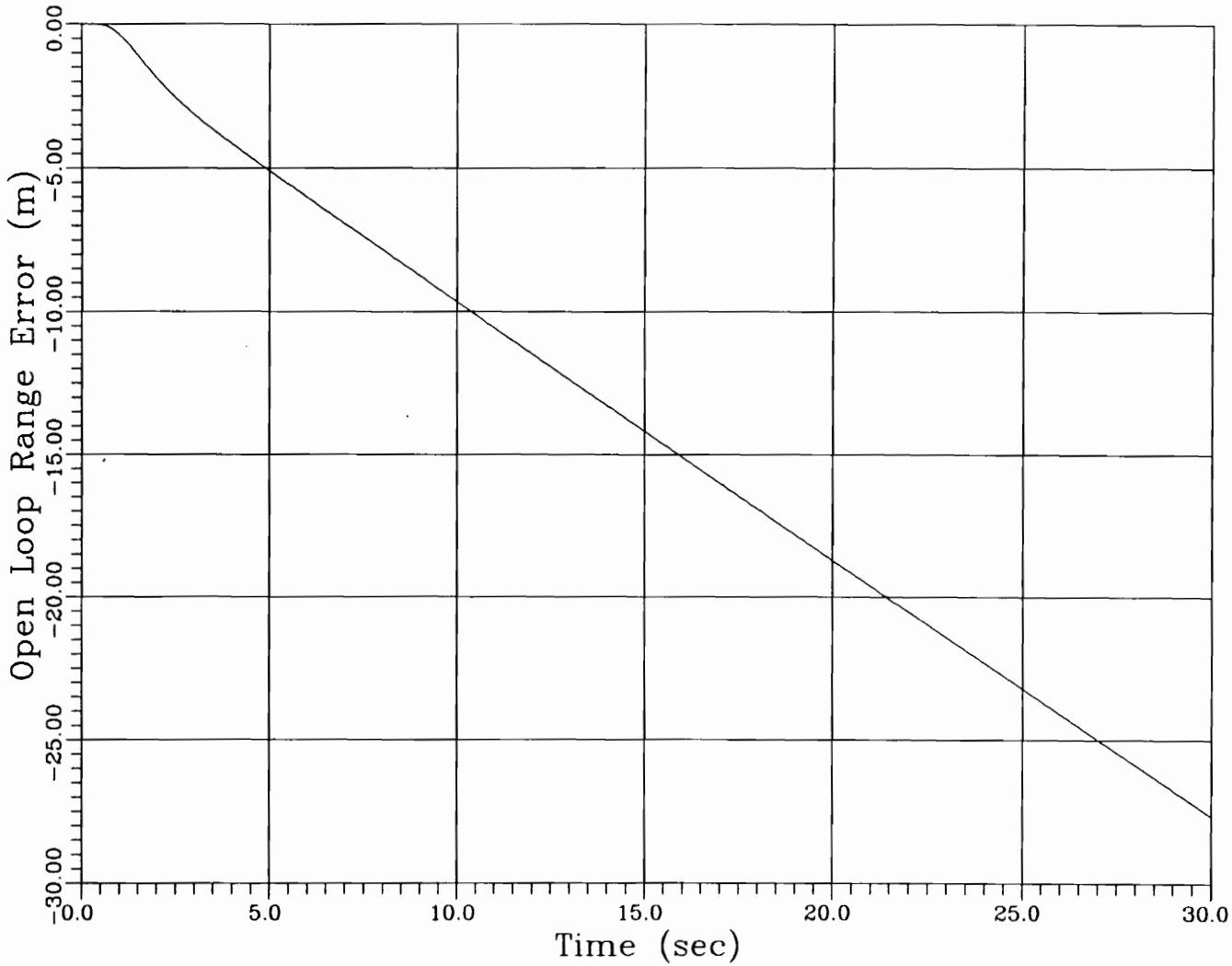

Figure 4.5 Open Loop Range Error
Nominal Geometry

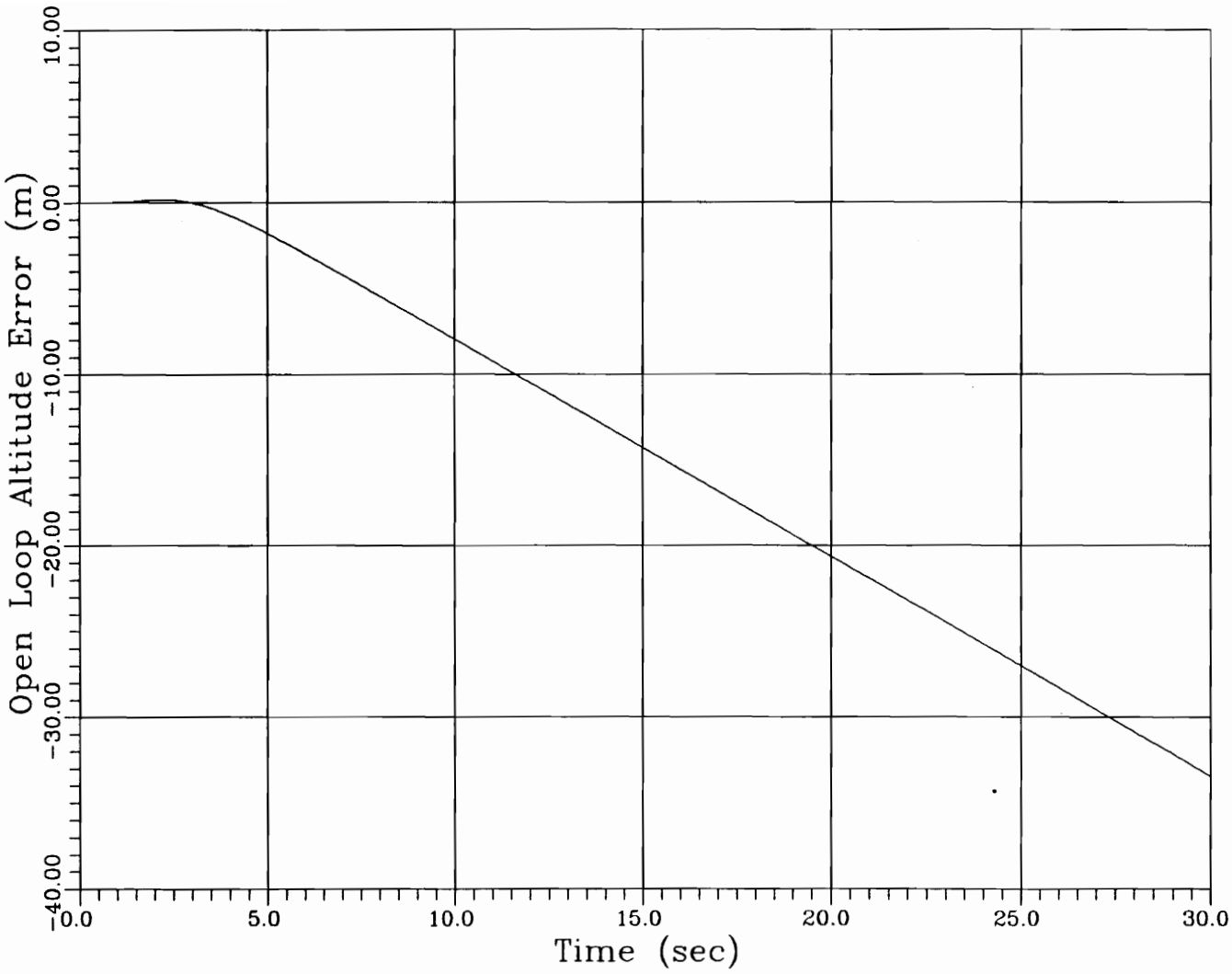

Figure 4.6 Open Loop Altitude Error
Nominal Geometry

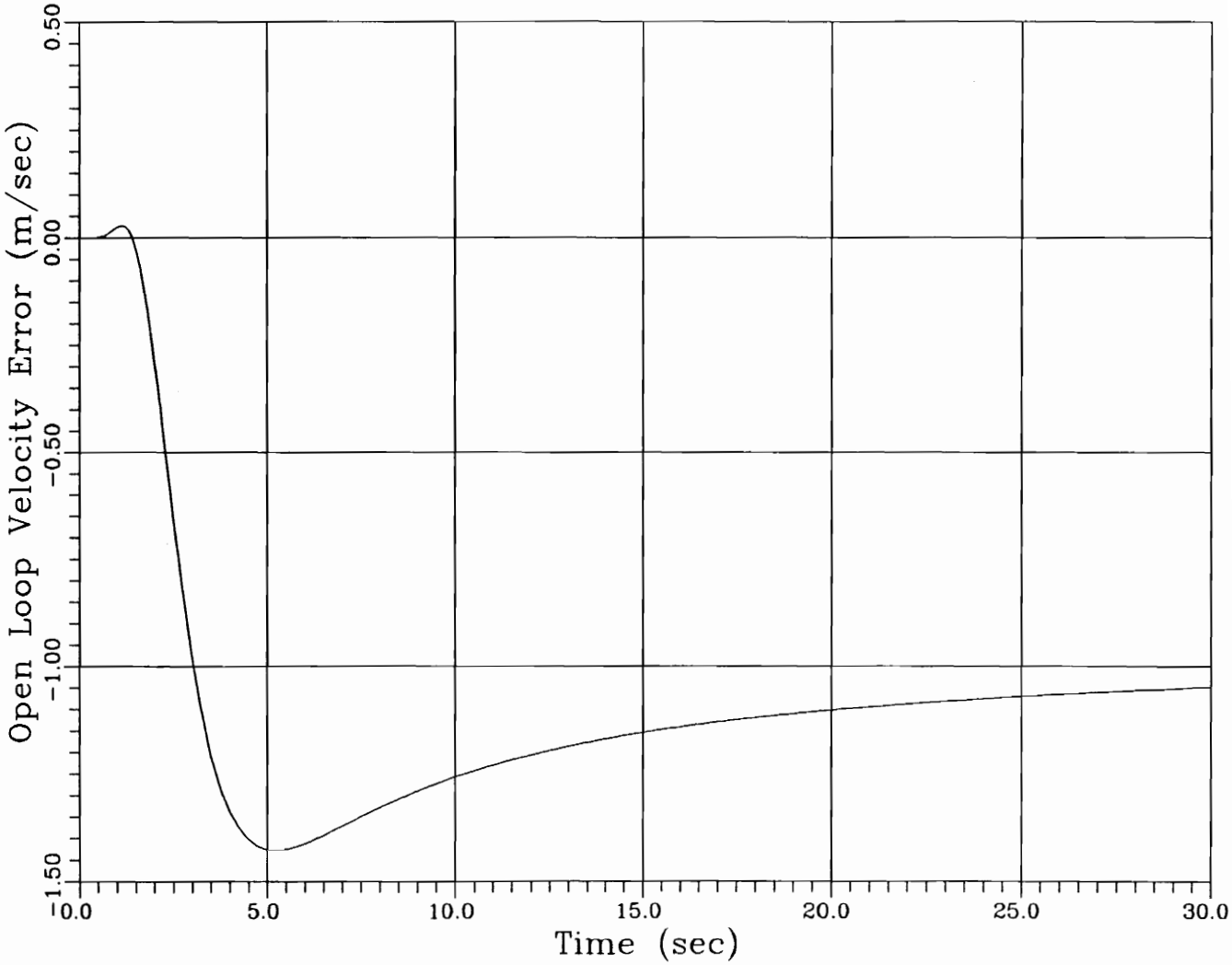

Figure 4.7 Open Loop Velocity Error
Nominal Geometry

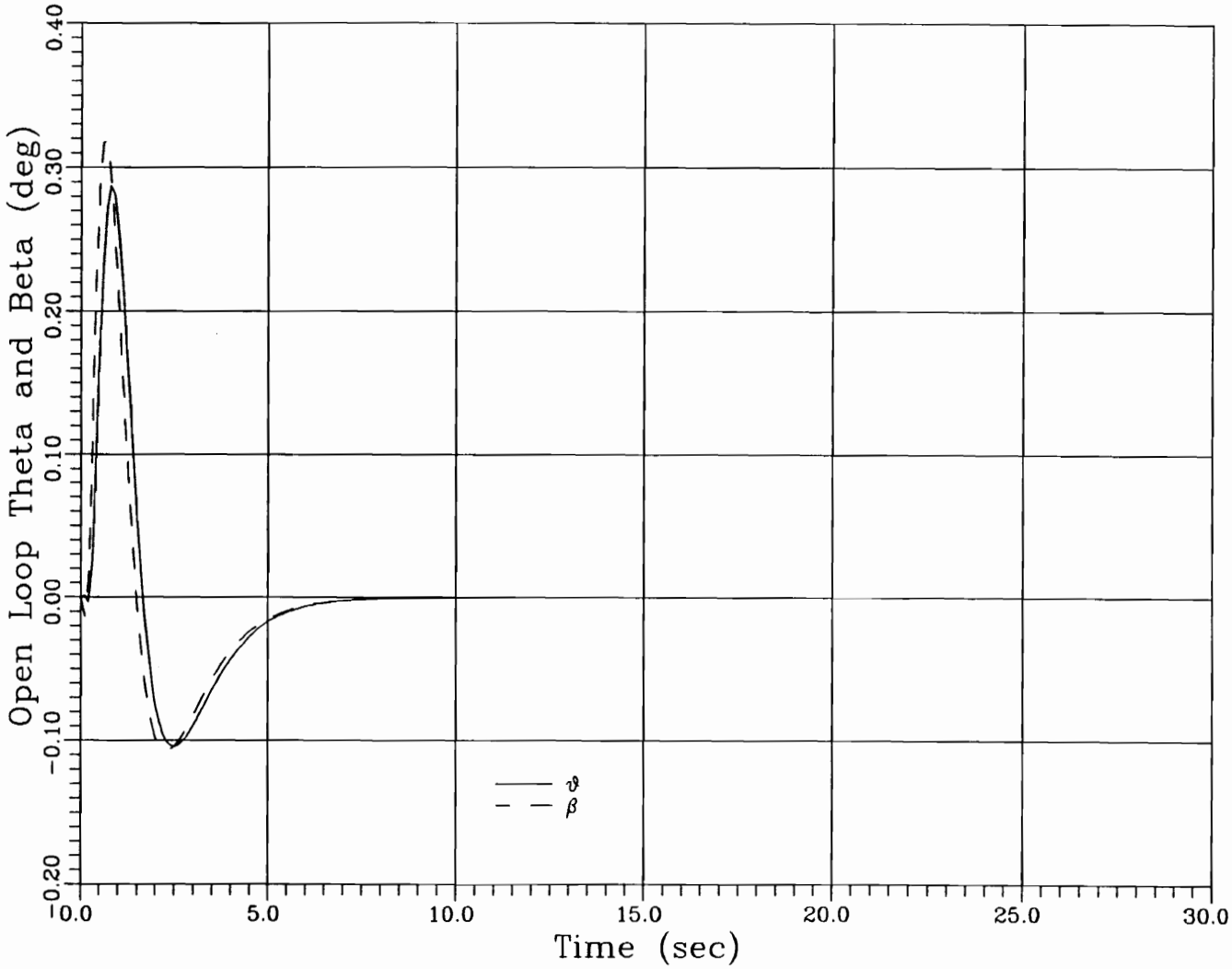

Figure 4.8 Open Loop Pitch Error and Control (β)
Nominal Geometry

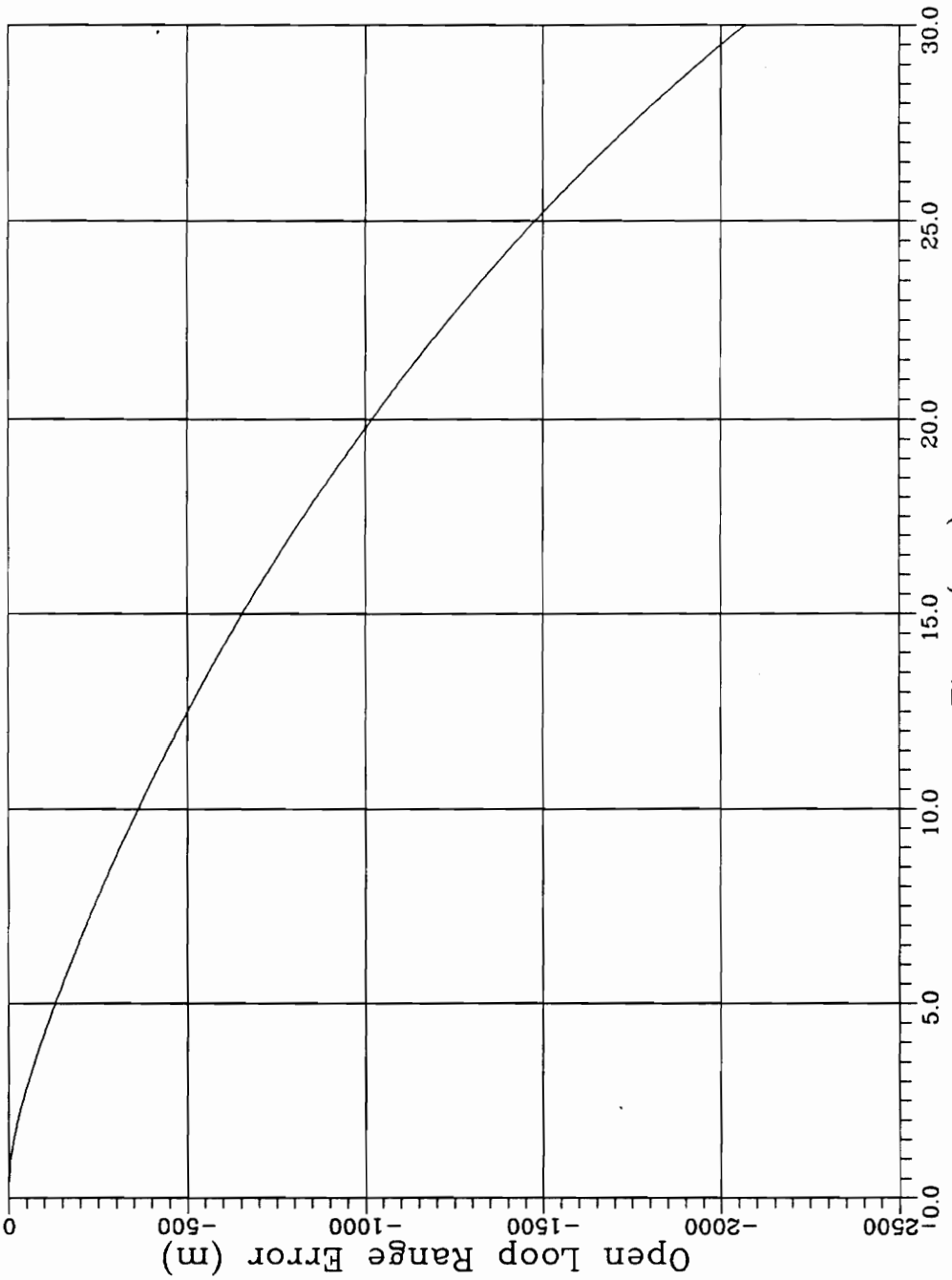

Figure 4.9 Open Loop Range Error
0.1 m Thrust Offset

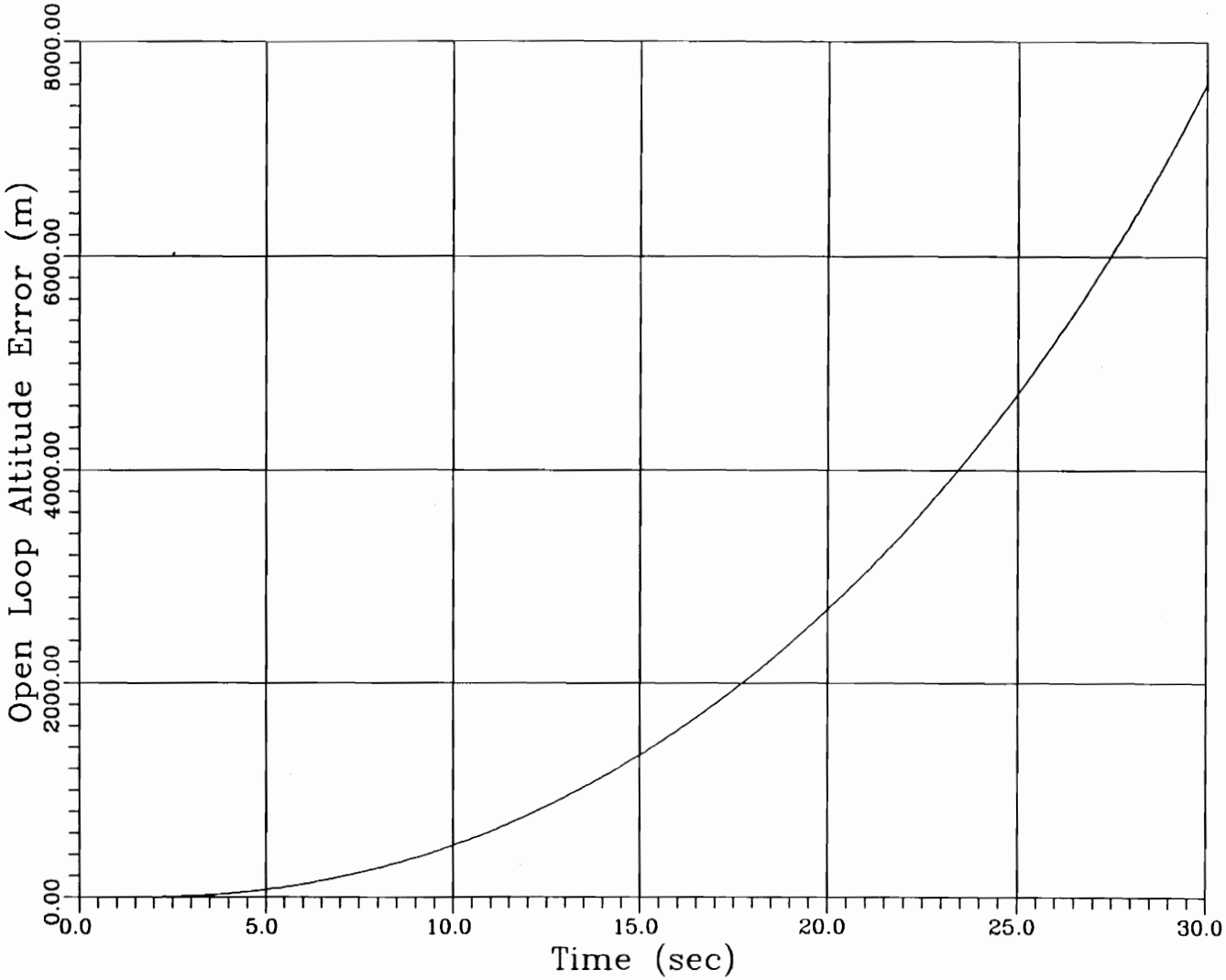

Figure 4.10 Open Loop Altitude Error
0.1 m Thrust Offset

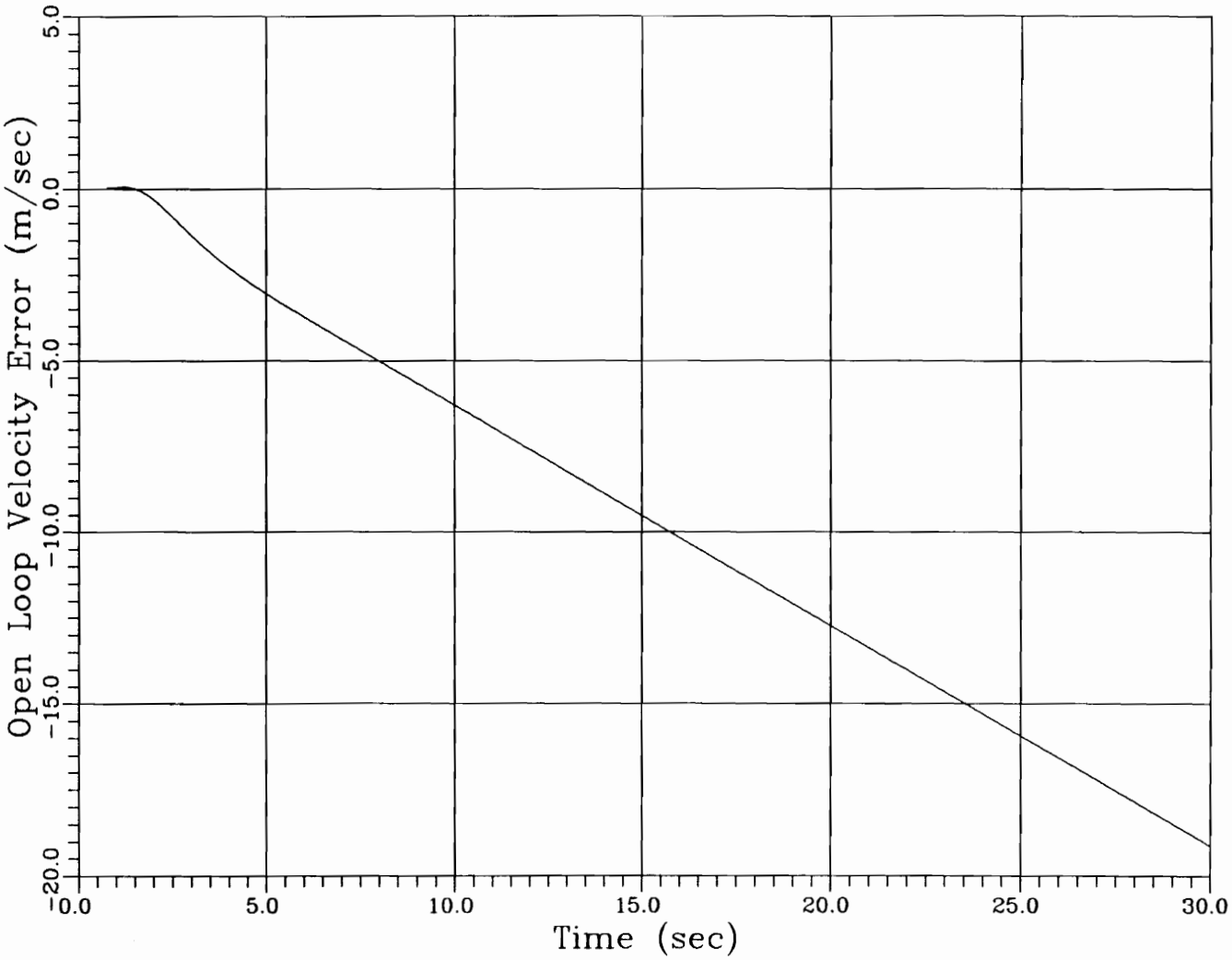

Figure 4.11 Open Loop Velocity Error
0.1 m Thrust Offset

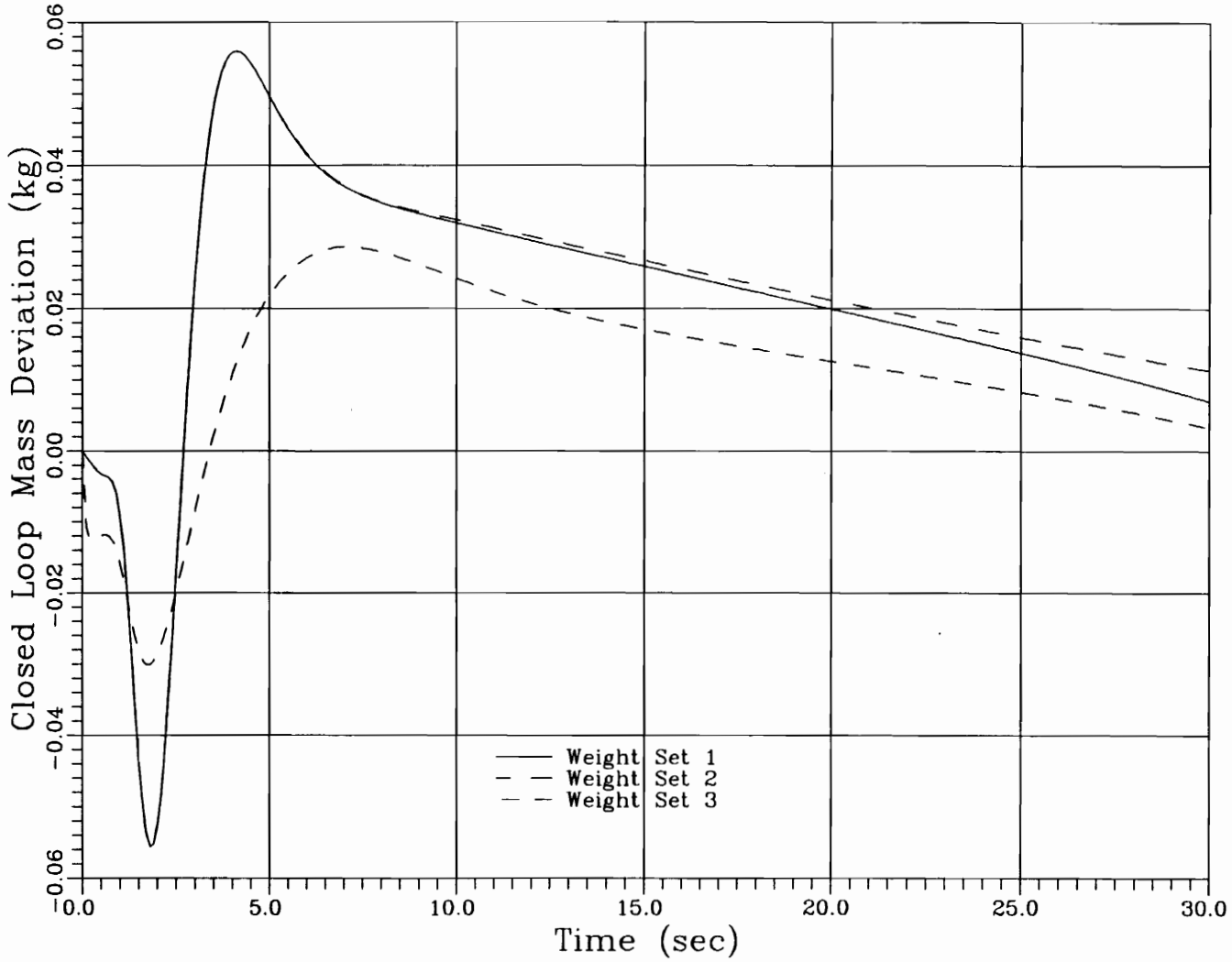

Figure 4.12 Closed Loop Mass Error
Nominal Geometry — Continuous Controller

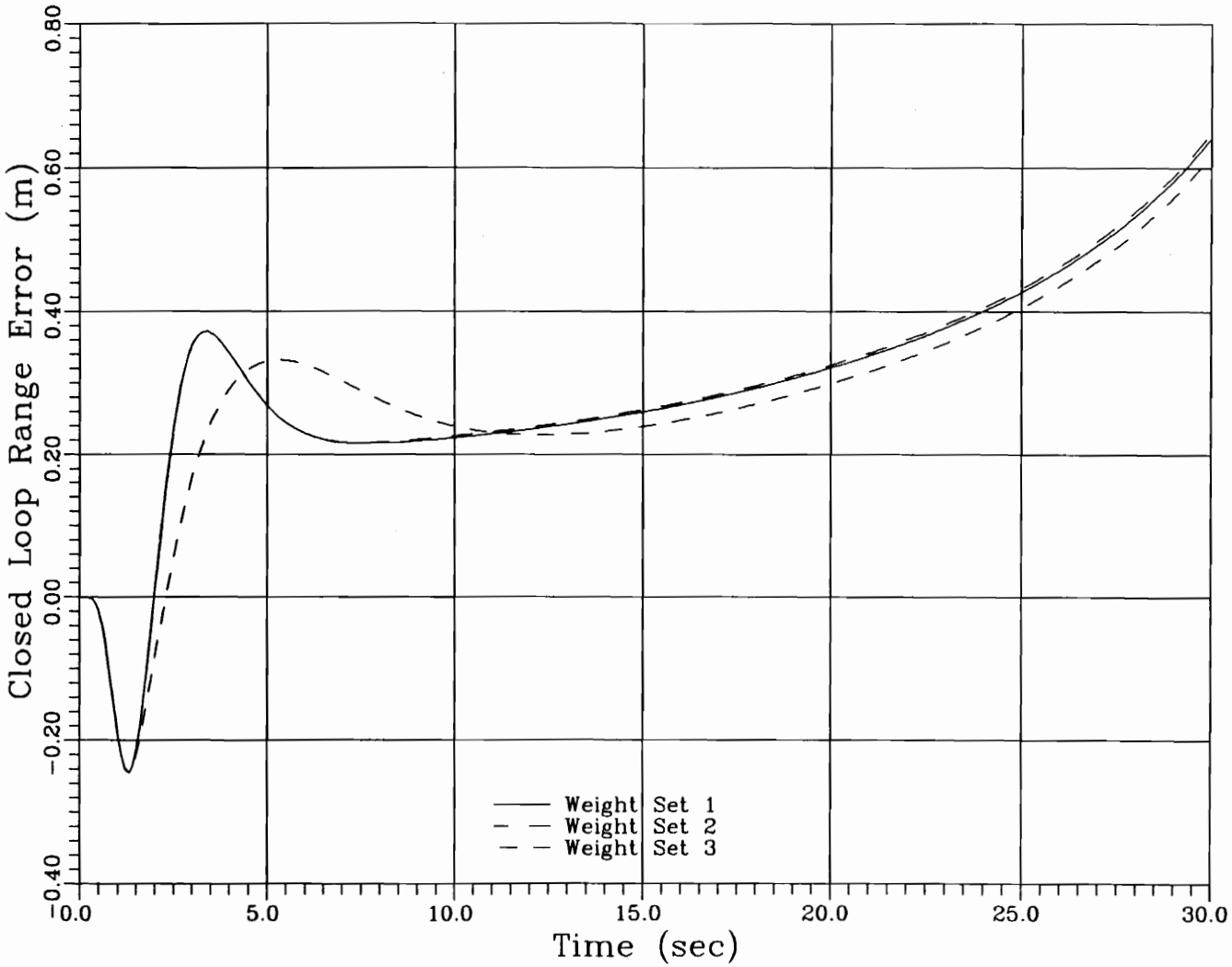

Figure 4.13 Closed Loop Range Error
Nominal Geometry — Continuous Controller

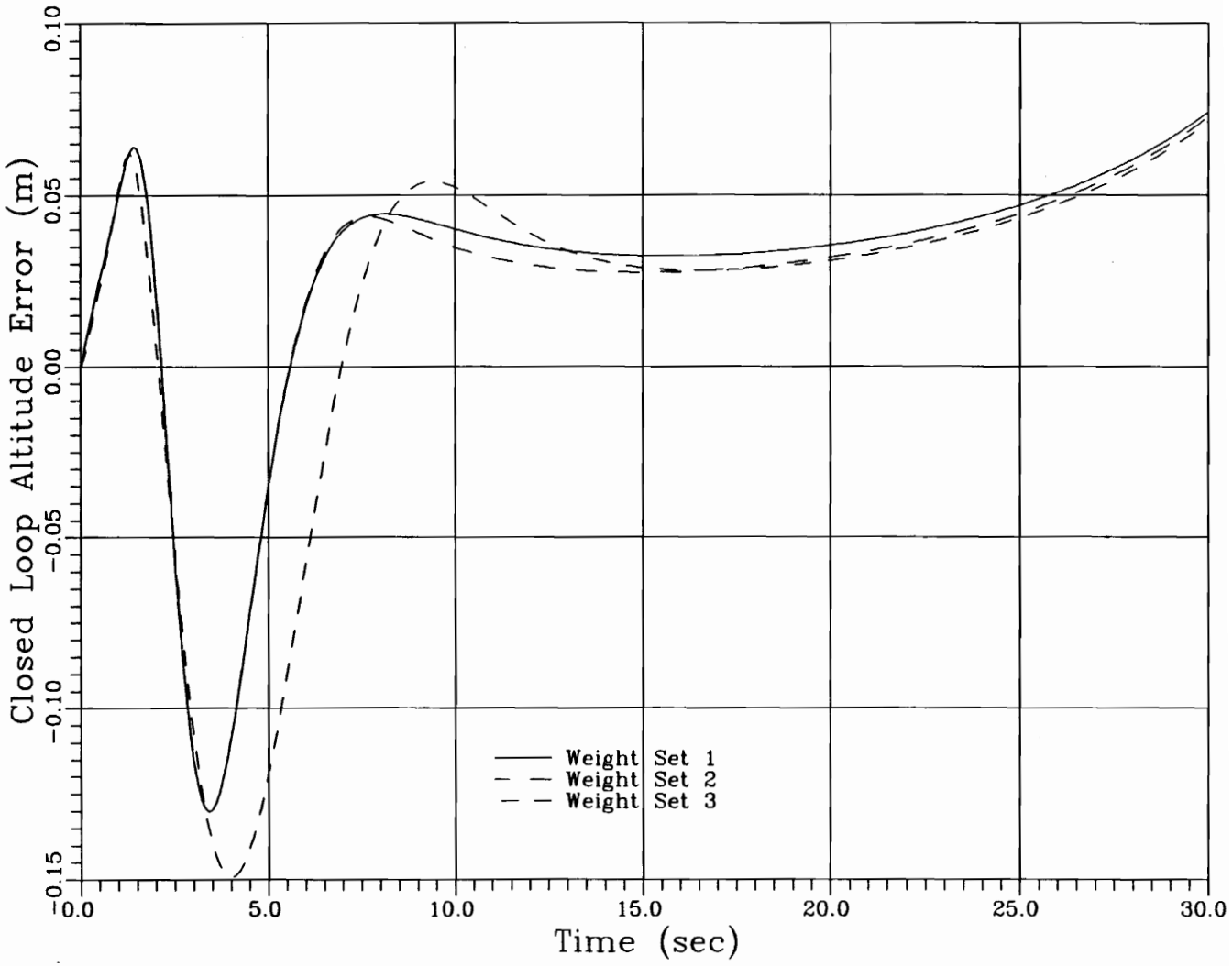

Figure 4.14 Closed Loop Altitude Error
Nominal Geometry — Continuous Controller

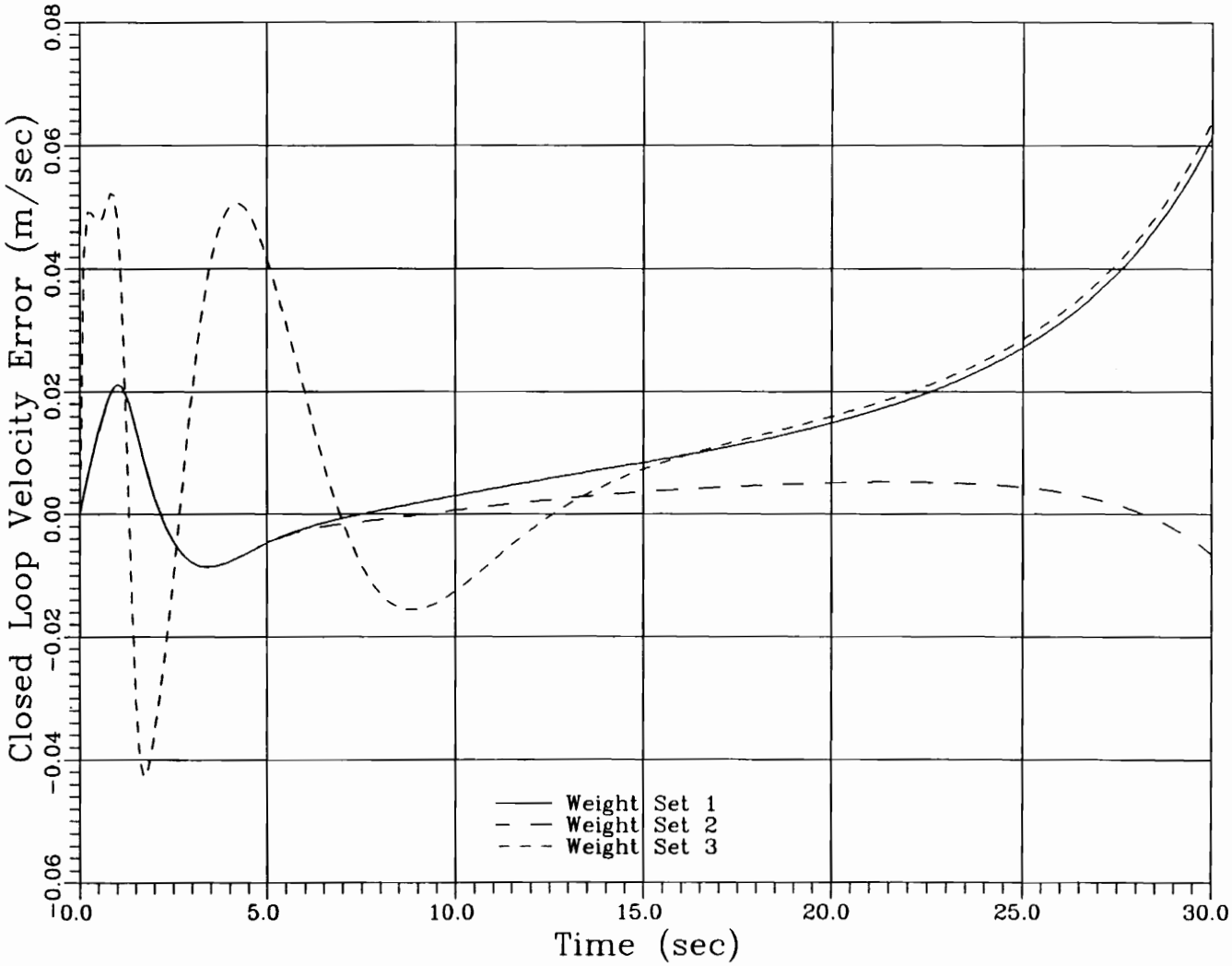

Figure 4.15 Closed Loop Velocity Error
Nominal Geometry — Continuous Controller

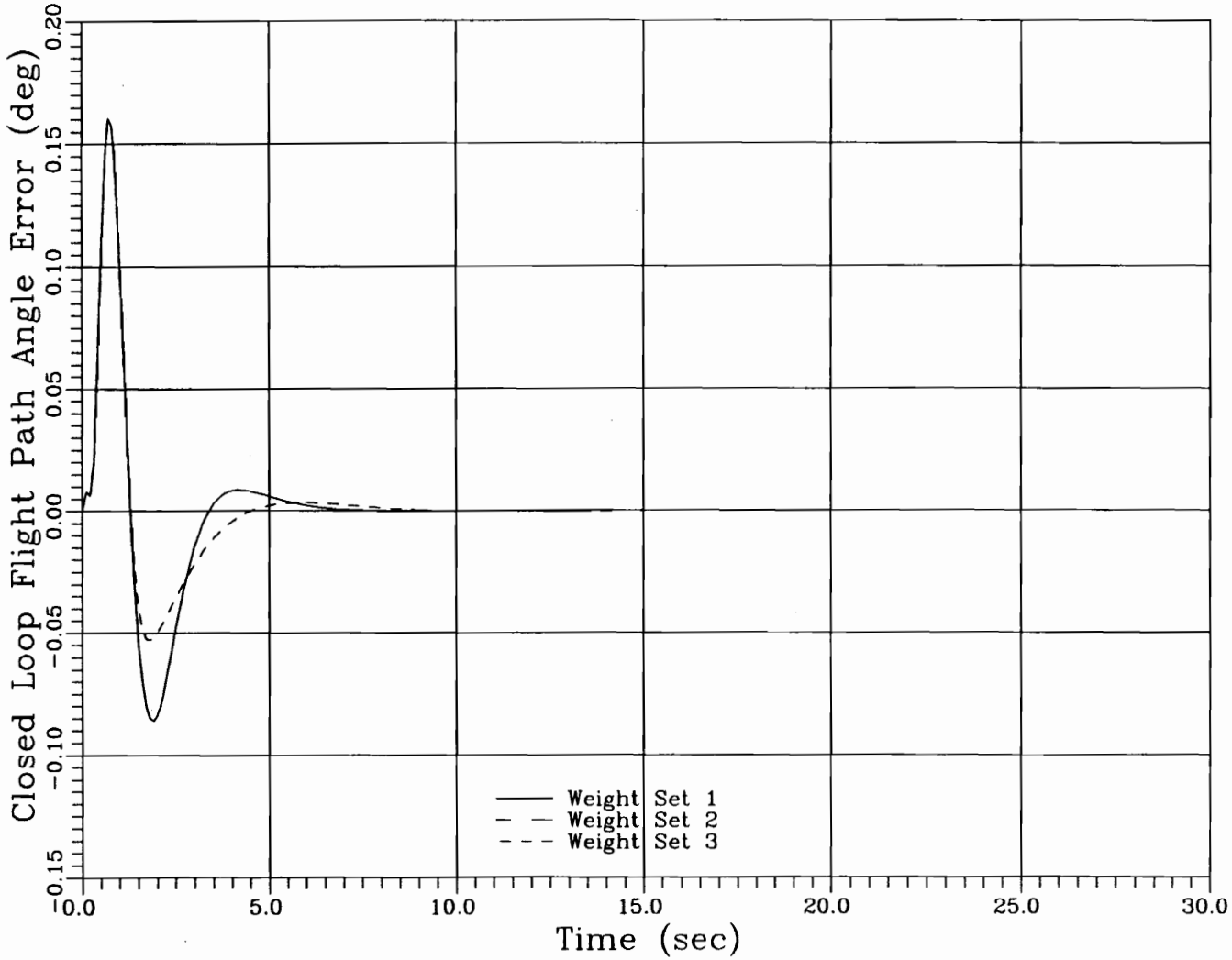

Figure 4.16 Closed Loop Flight Path Angle Error
Nominal Geometry — Continuous Controller

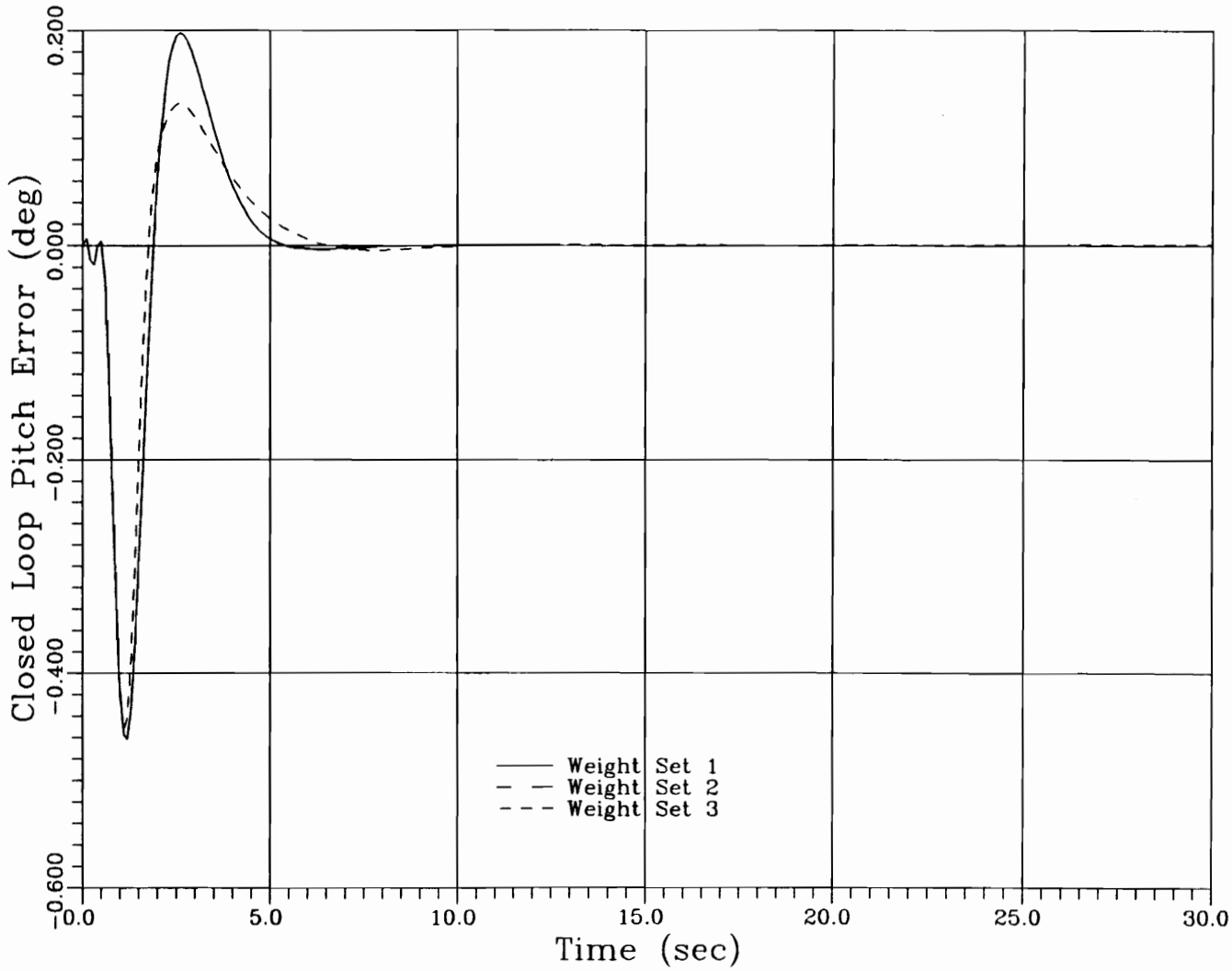

Figure 4.17 Closed Loop Pitch Attitude Error
Nominal Geometry — Continuous Controller

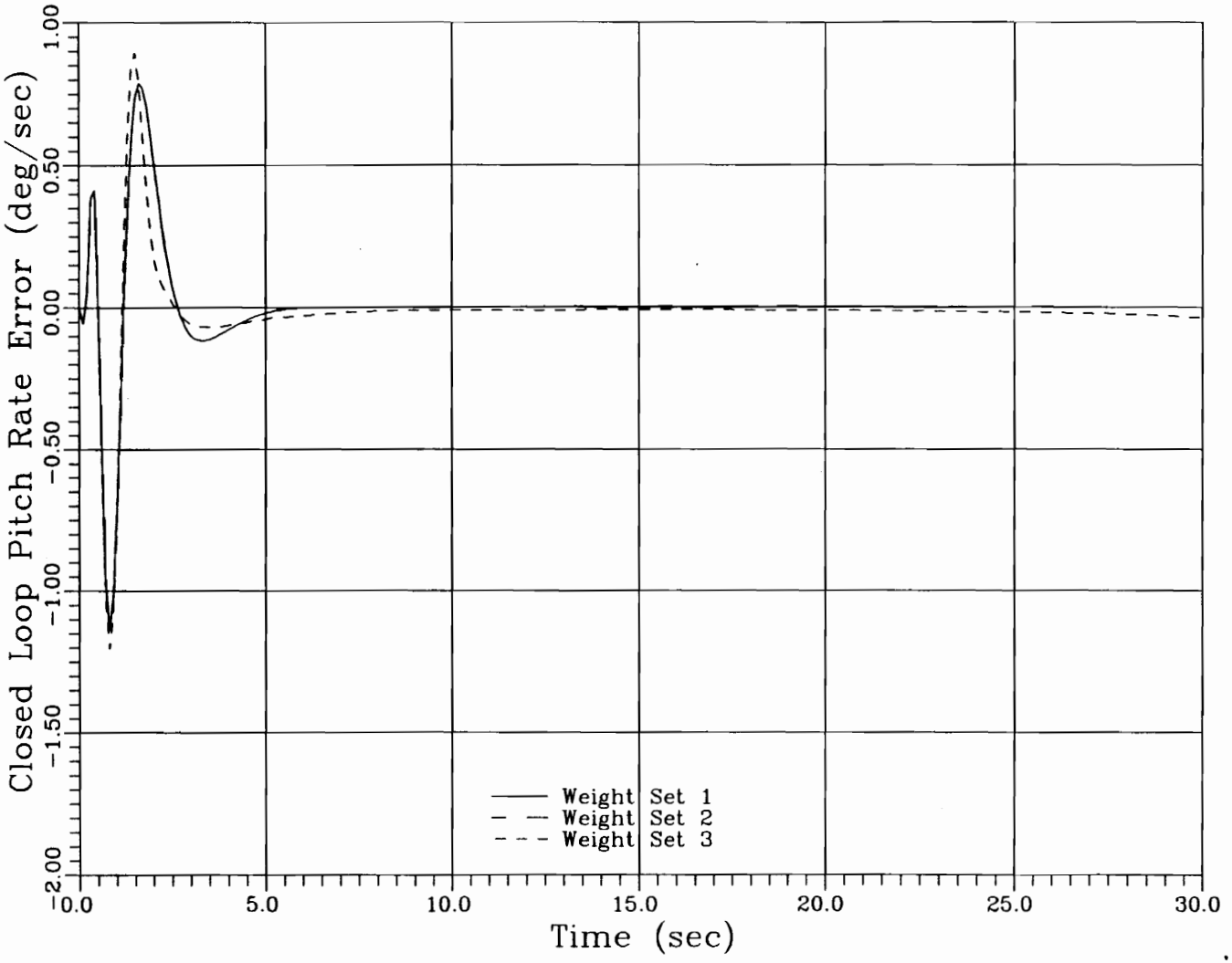

Figure 4.18 Closed Loop Pitch Rate Error
Nominal Geometry — Continuous Controller

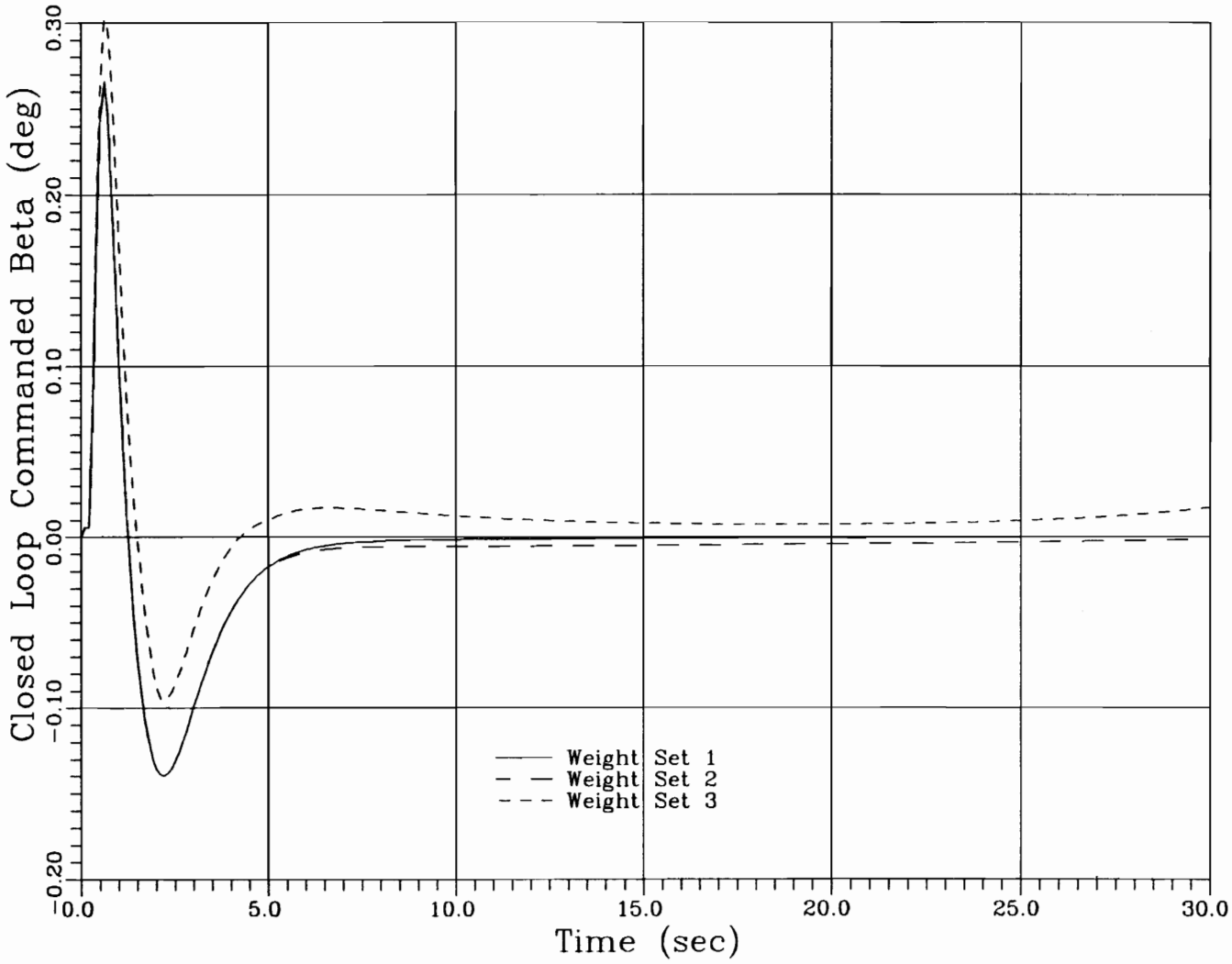

Figure 4.19 Applied Thrust Vector Control (β)
Nominal Geometry — Continuous Controller

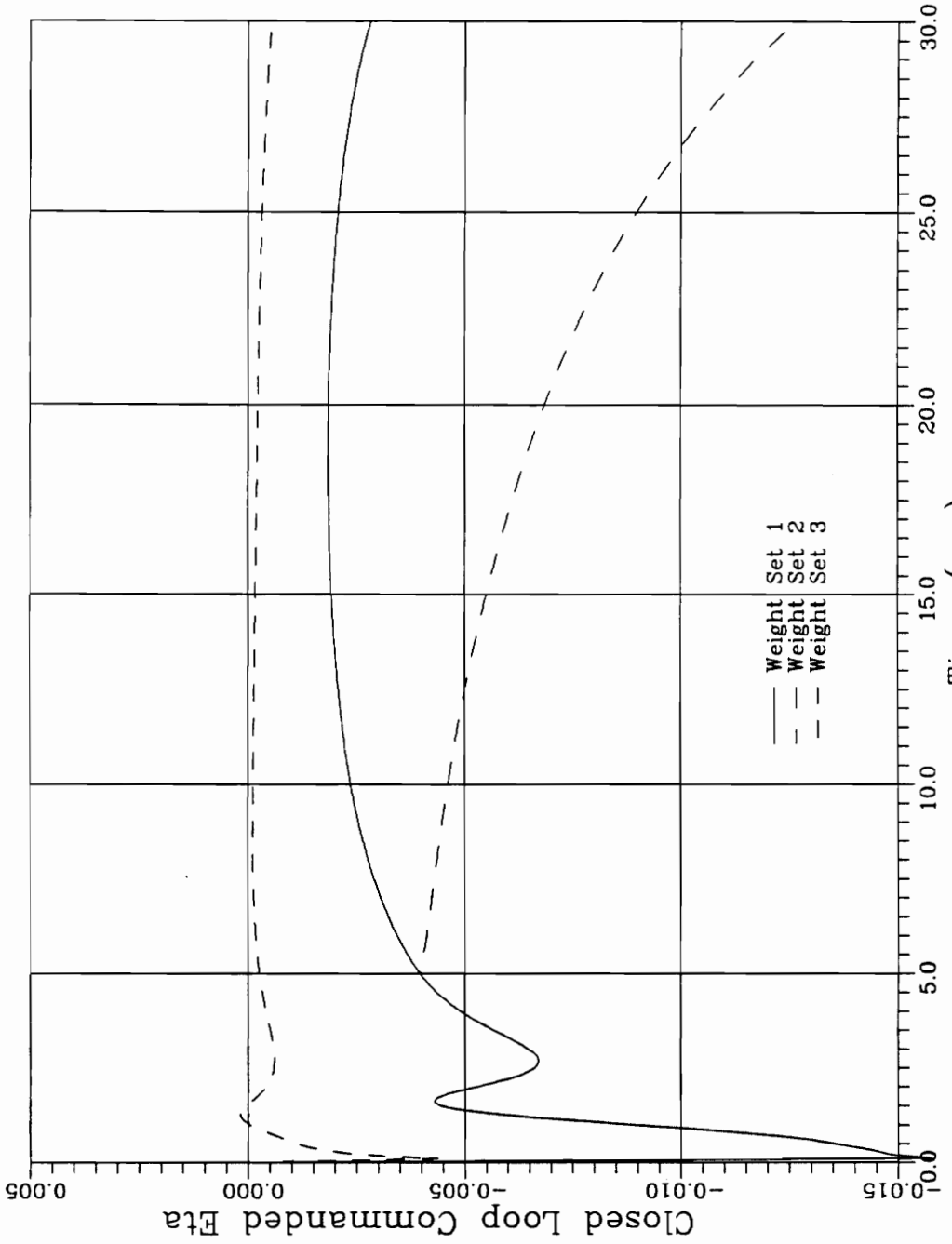

Figure 4.20 Applied Throttle Deviation (η)
Nominal Geometry — Continuous Controller

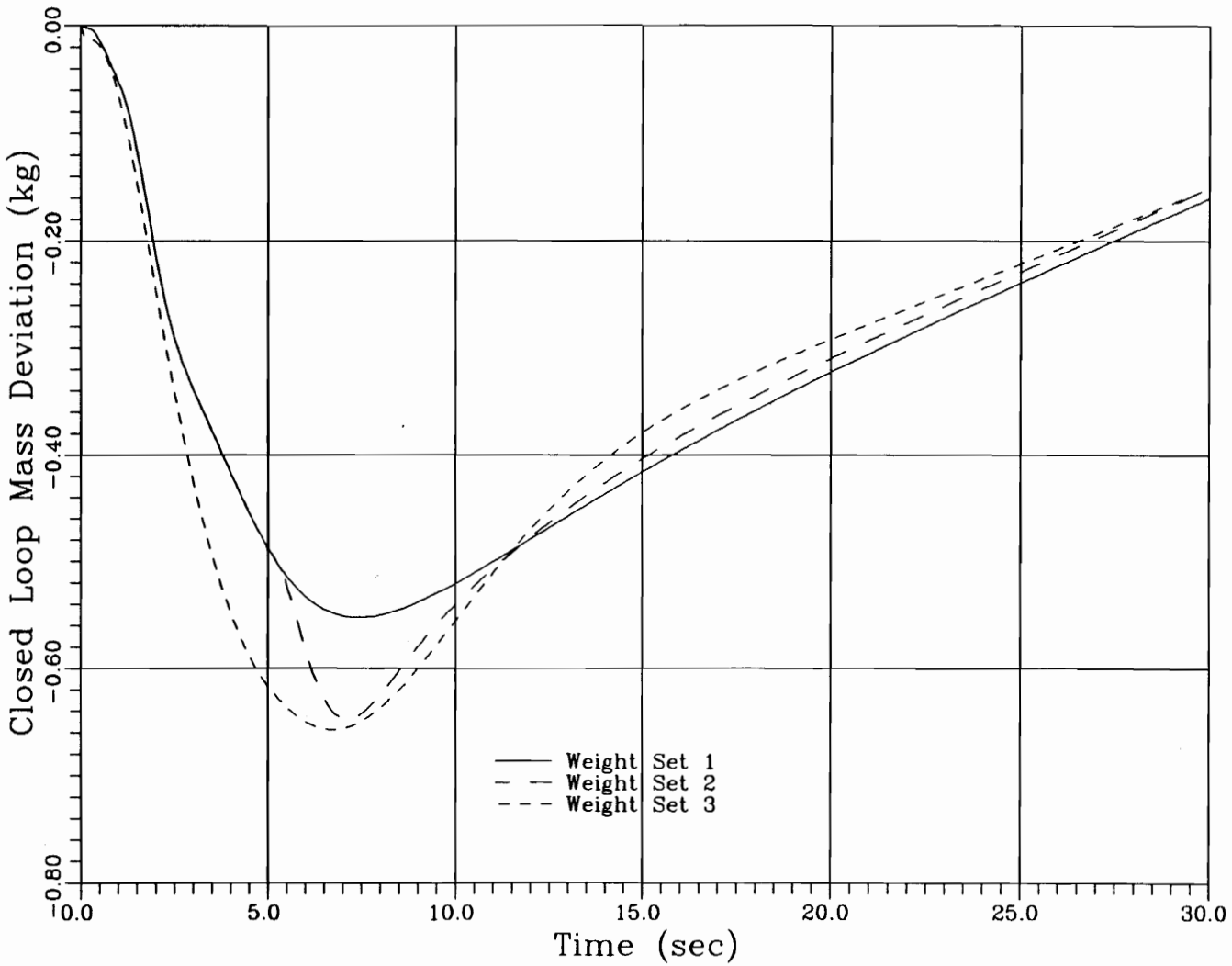

Figure 4.21 Closed Loop Mass Error
0.1 m Thrust Offset — Continuous Controller

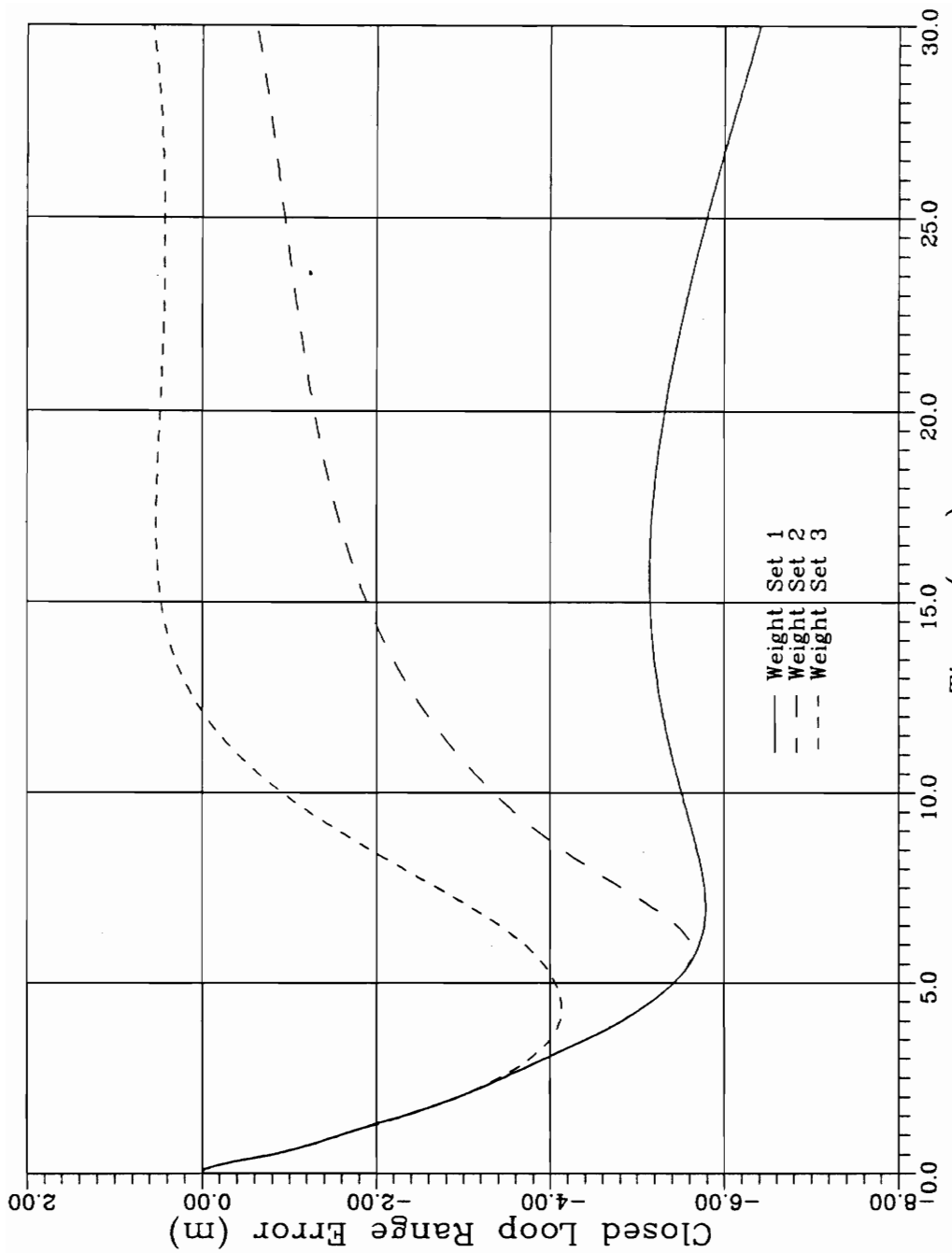

Figure 4.22 Closed Loop Range Error
0.1 *m* Thrust Offset — Continuous Controller

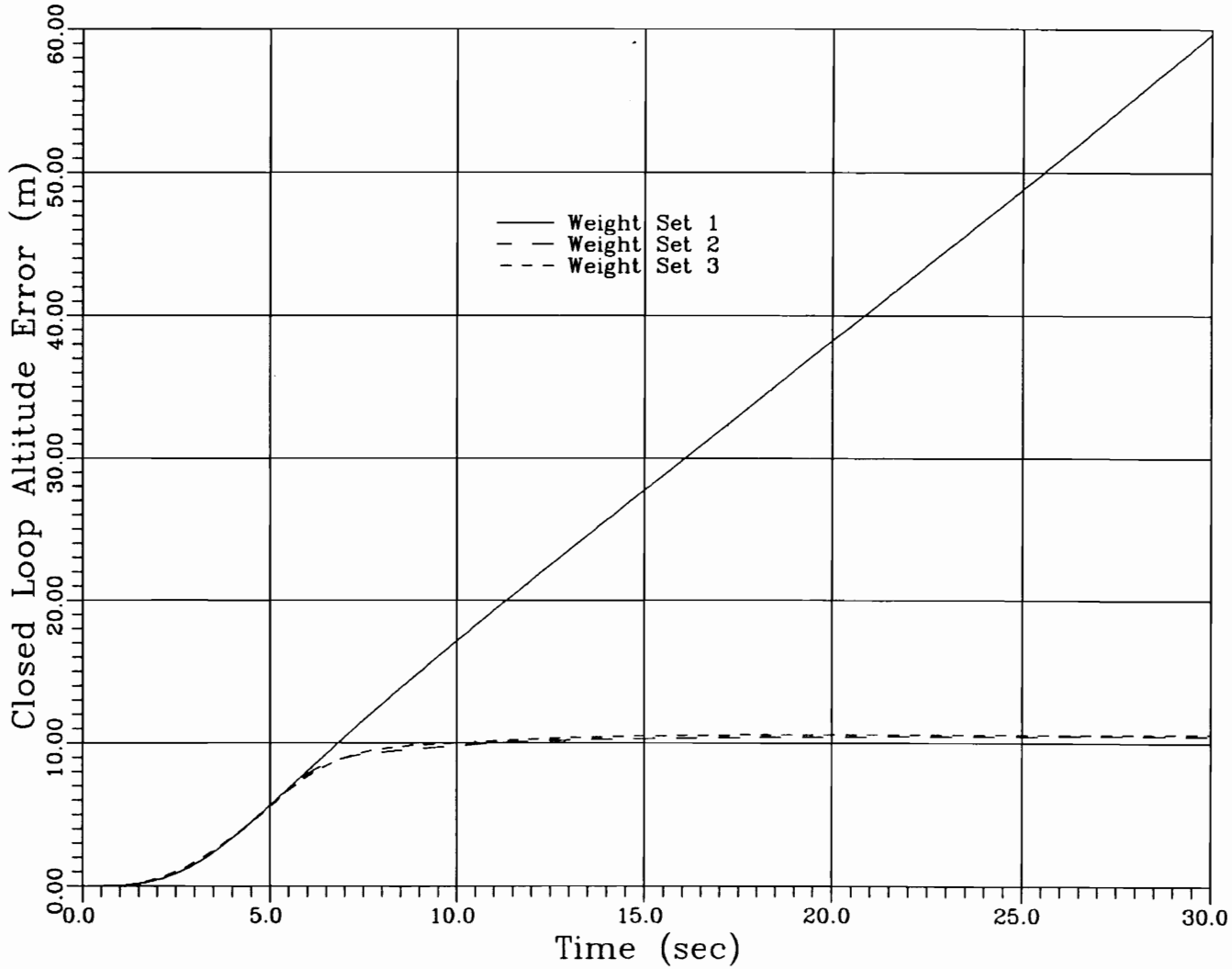

Figure 4.23 Closed Loop Altitude Error
0.1 *m* Thrust Offset — Continuous Controller

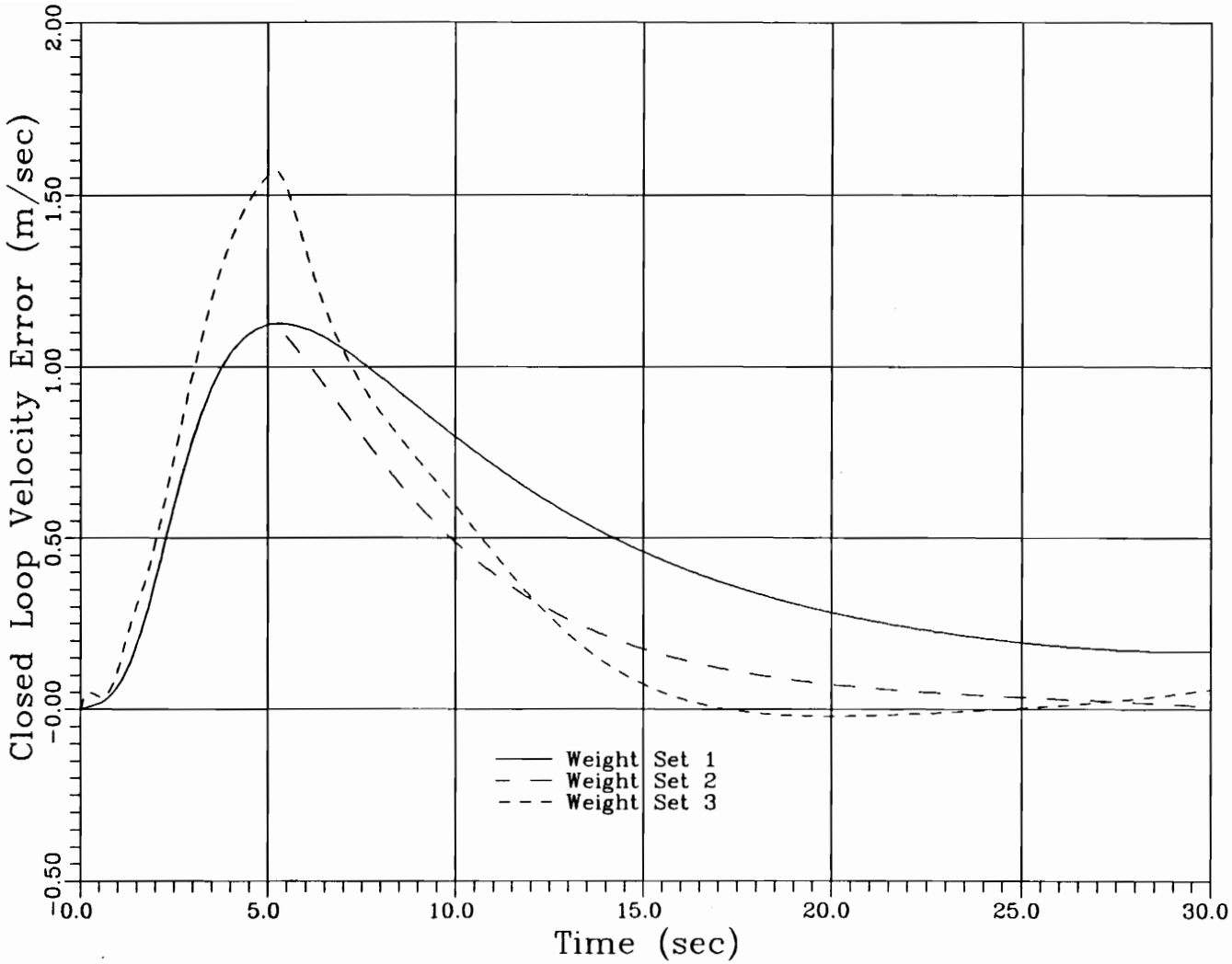

Figure 4.24 Closed Loop Velocity Error
0.1 m Thrust Offset — Continuous Controller

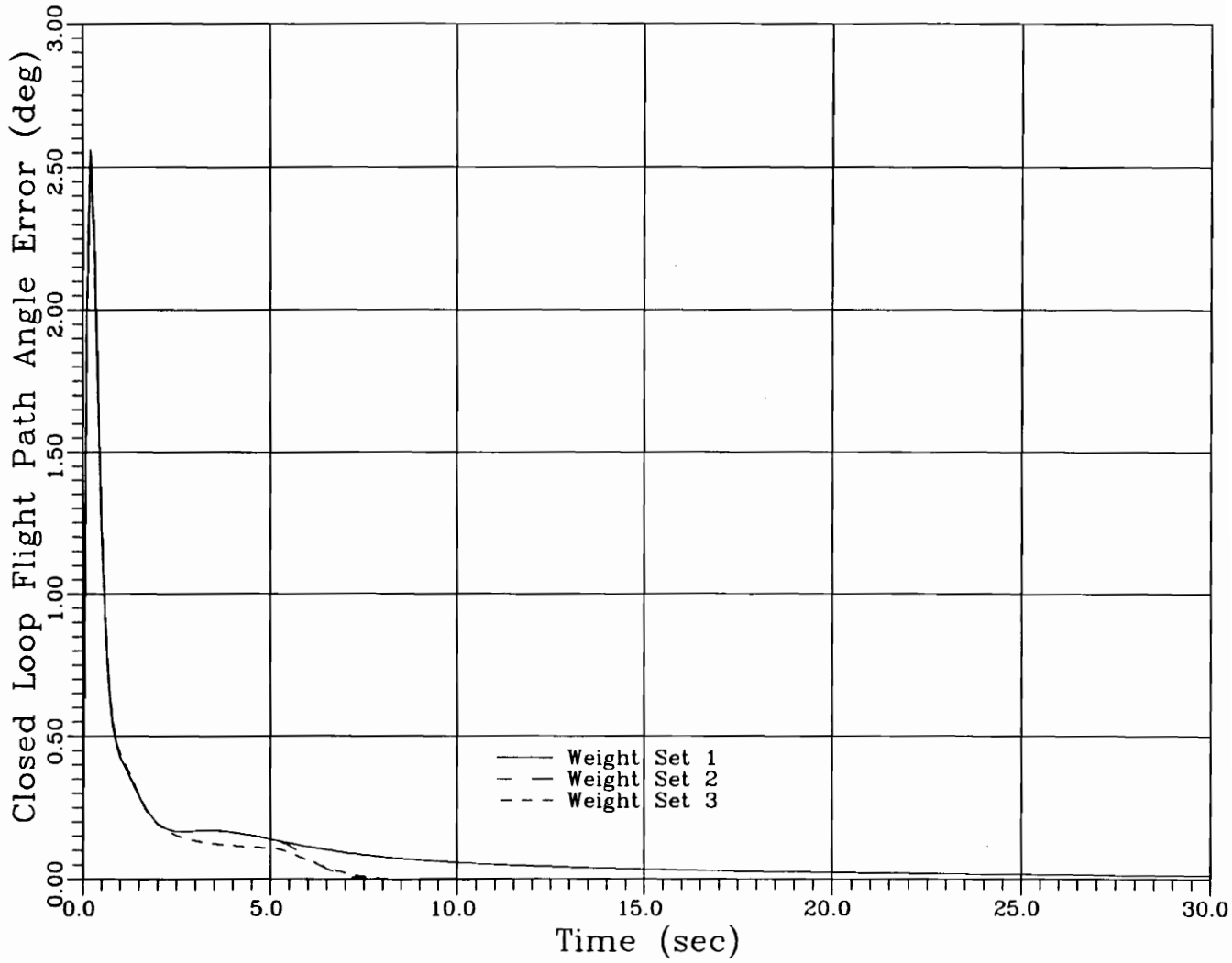

Figure 4.25 Closed Loop Flight Path Angle Error
0.1 m Thrust Offset — Continuous Controller

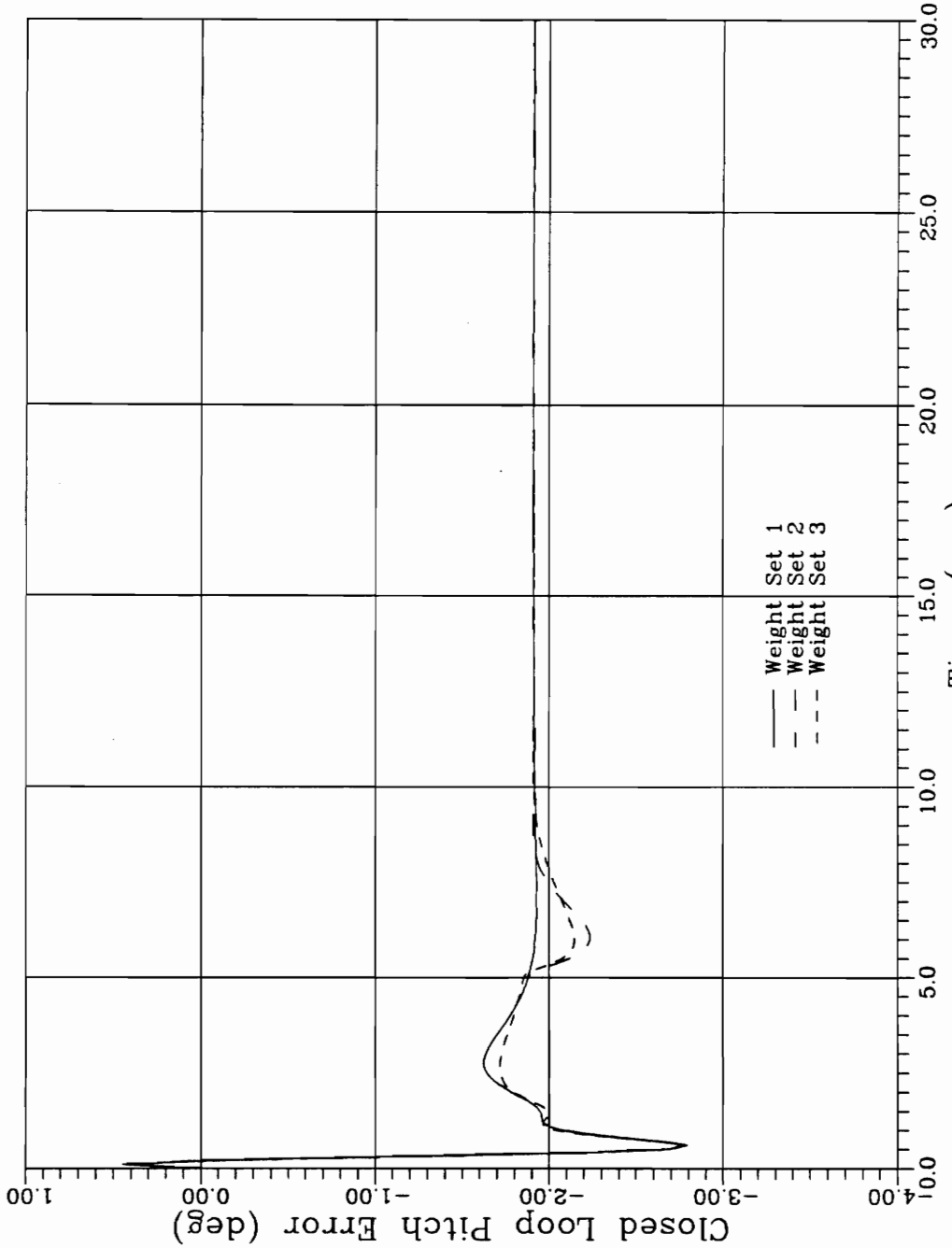

Figure 4.26 Closed Loop Pitch Attitude Error
0.1 *m* Thrust Offset — Continuous Controller

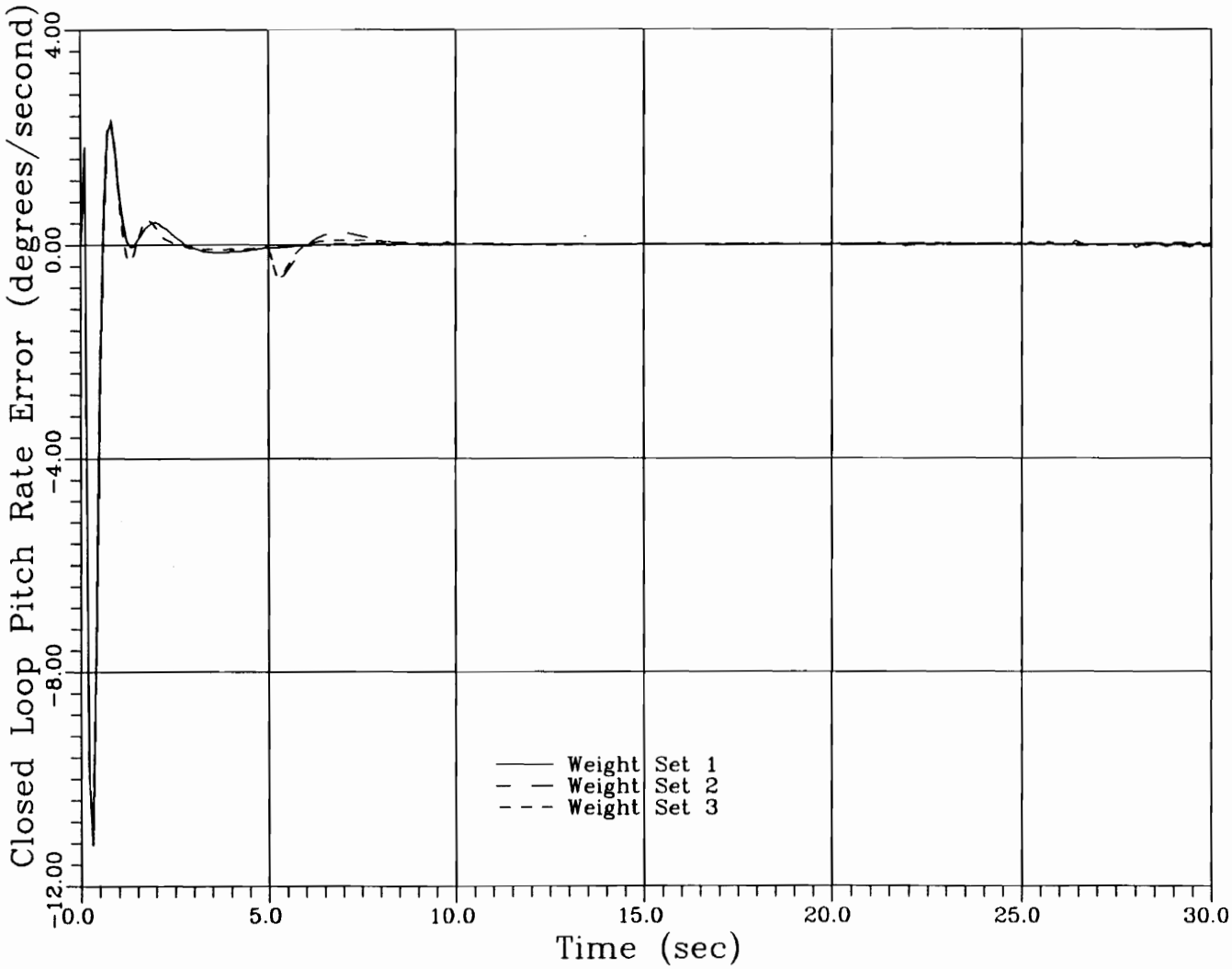

Figure 4.27 Closed Loop Pitch Rate Error
0.1 m Thrust Offset — Continuous Controller

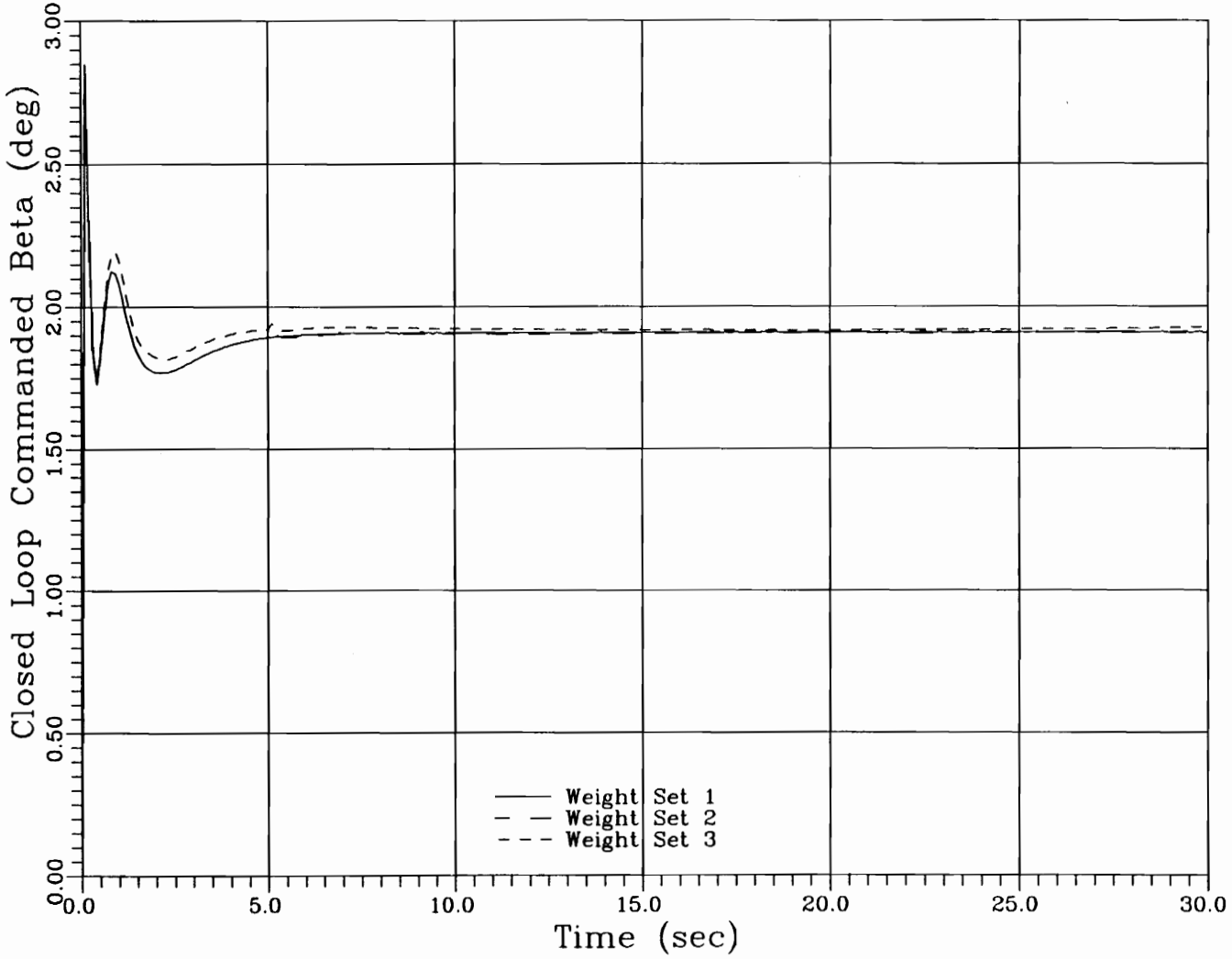

Figure 4.28 Applied Thrust Vector Control (β)
0.1 m Thrust Offset — Continuous Controller

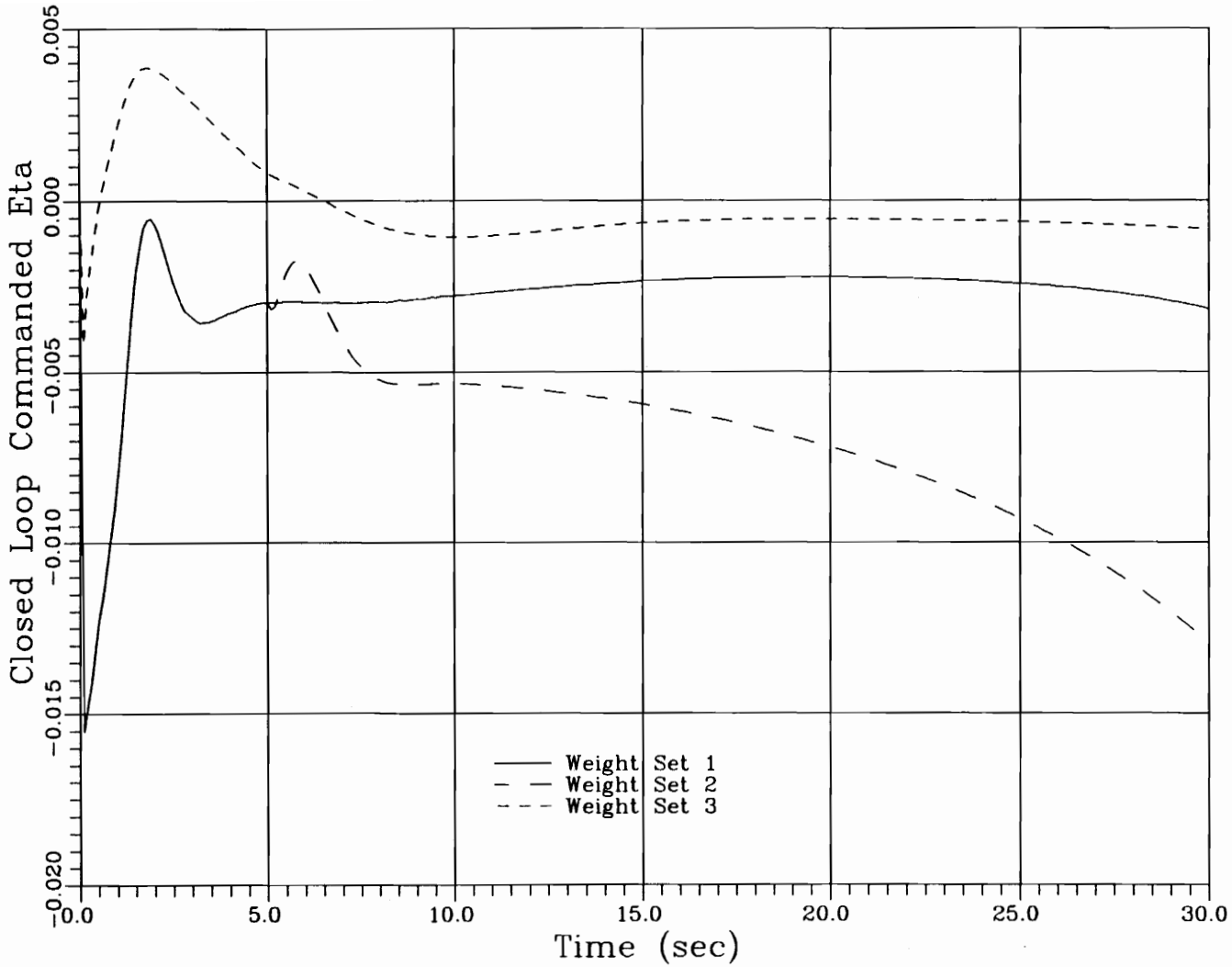

Figure 4.29 Applied Throttle Deviation (η)
0.1 m Thrust Offset — Continuous Controller

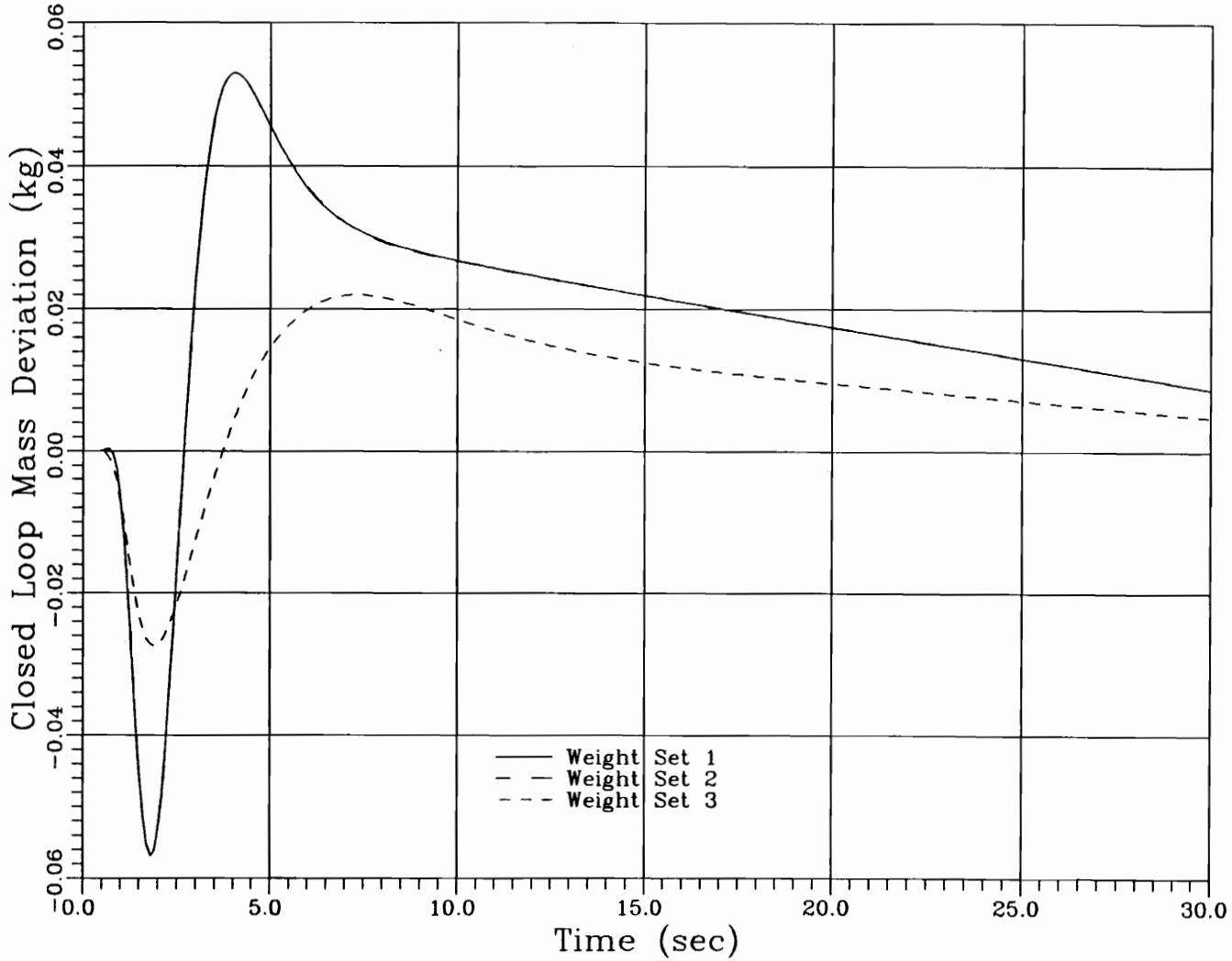

Figure 4.30 Closed Loop Mass Error
Nominal Geometry — Continuous Controller
Simultaneously Integrated Nominal

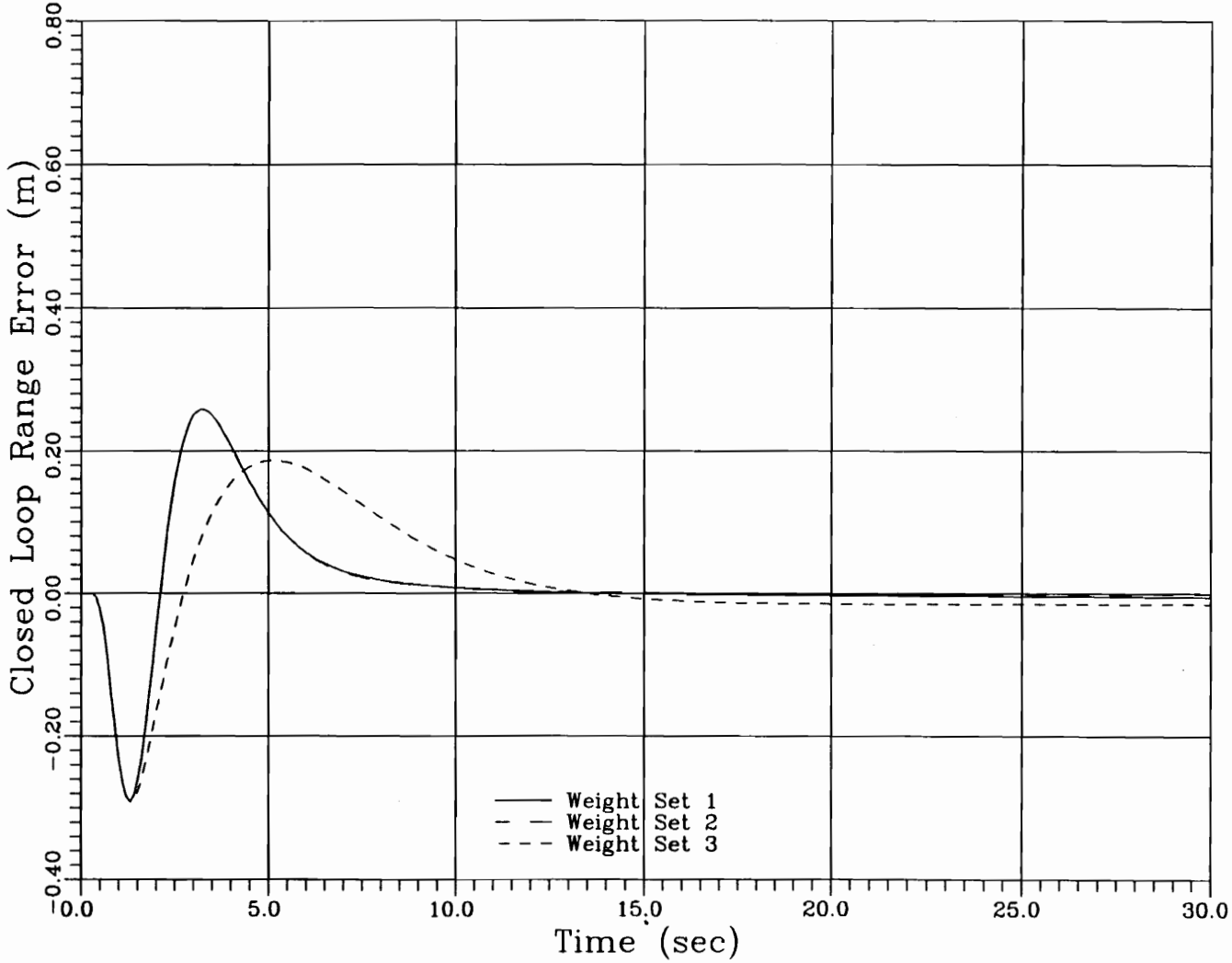

Figure 4.31 Closed Loop Range Error
Nominal Geometry — Continuous Controller
Simultaneously Integrated Nominal

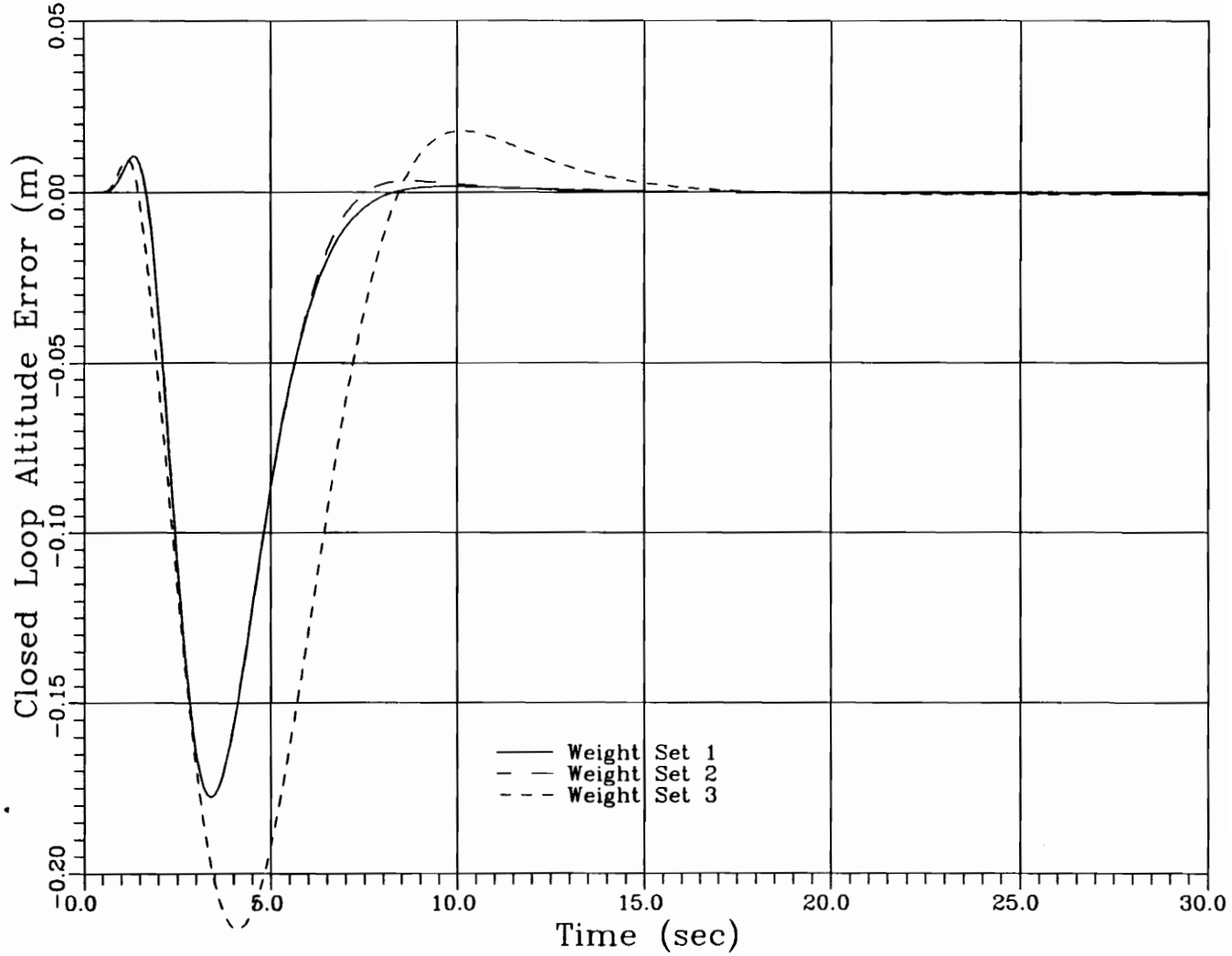

Figure 4.32 Closed Loop Altitude Error
Nominal Geometry — Continuous Controller
Simultaneously Integrated Nominal

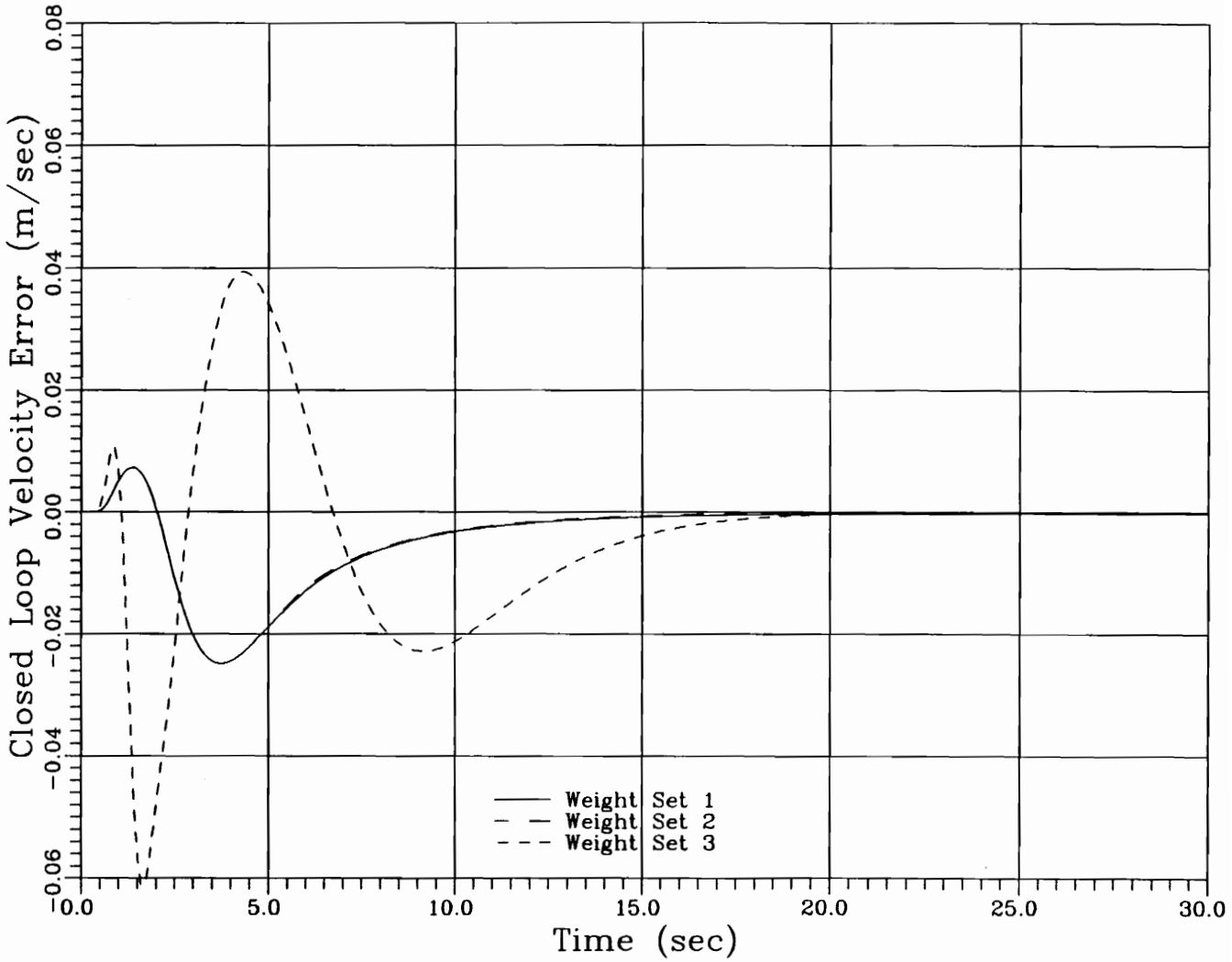

Figure 4.33 Closed Loop Velocity Error
Nominal Geometry — Continuous Controller
Simultaneously Integrated Nominal

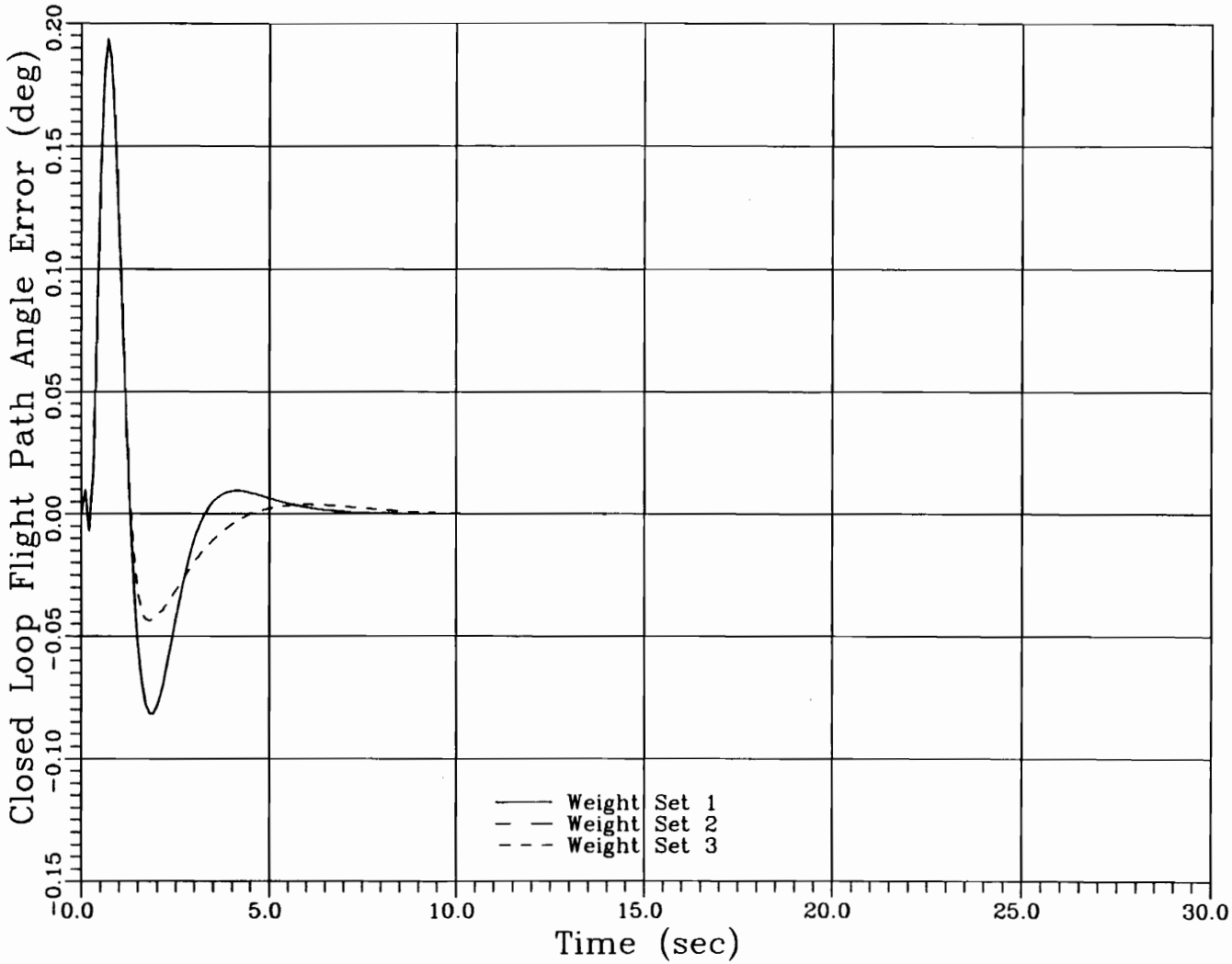

Figure 4.34 Closed Loop Flight Path Angle Error
Nominal Geometry — Continuous Controller
Simultaneously Integrated Nominal

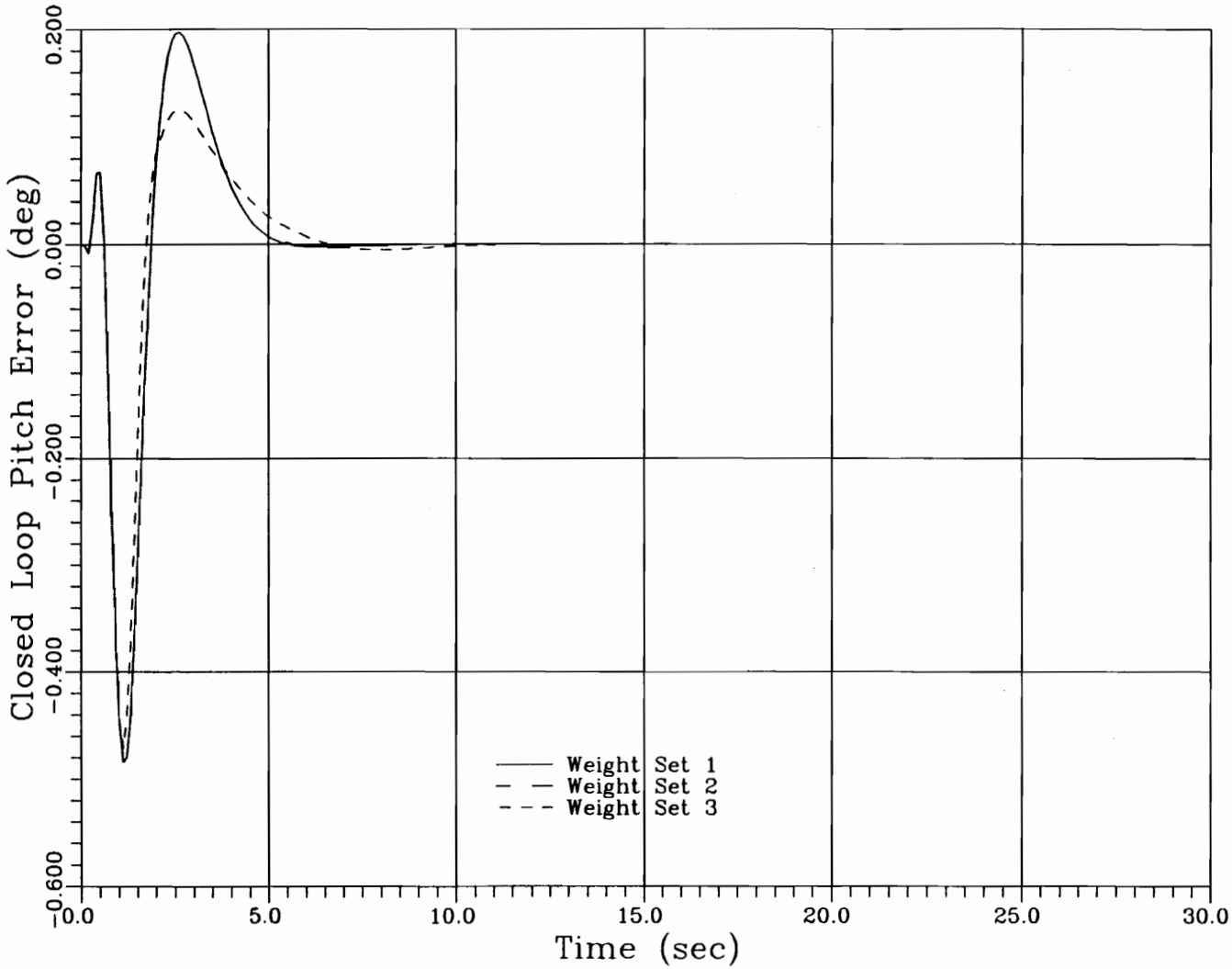

Figure 4.35 Closed Loop Pitch Attitude Error
Nominal Geometry — Continuous Controller
Simultaneously Integrated Nominal

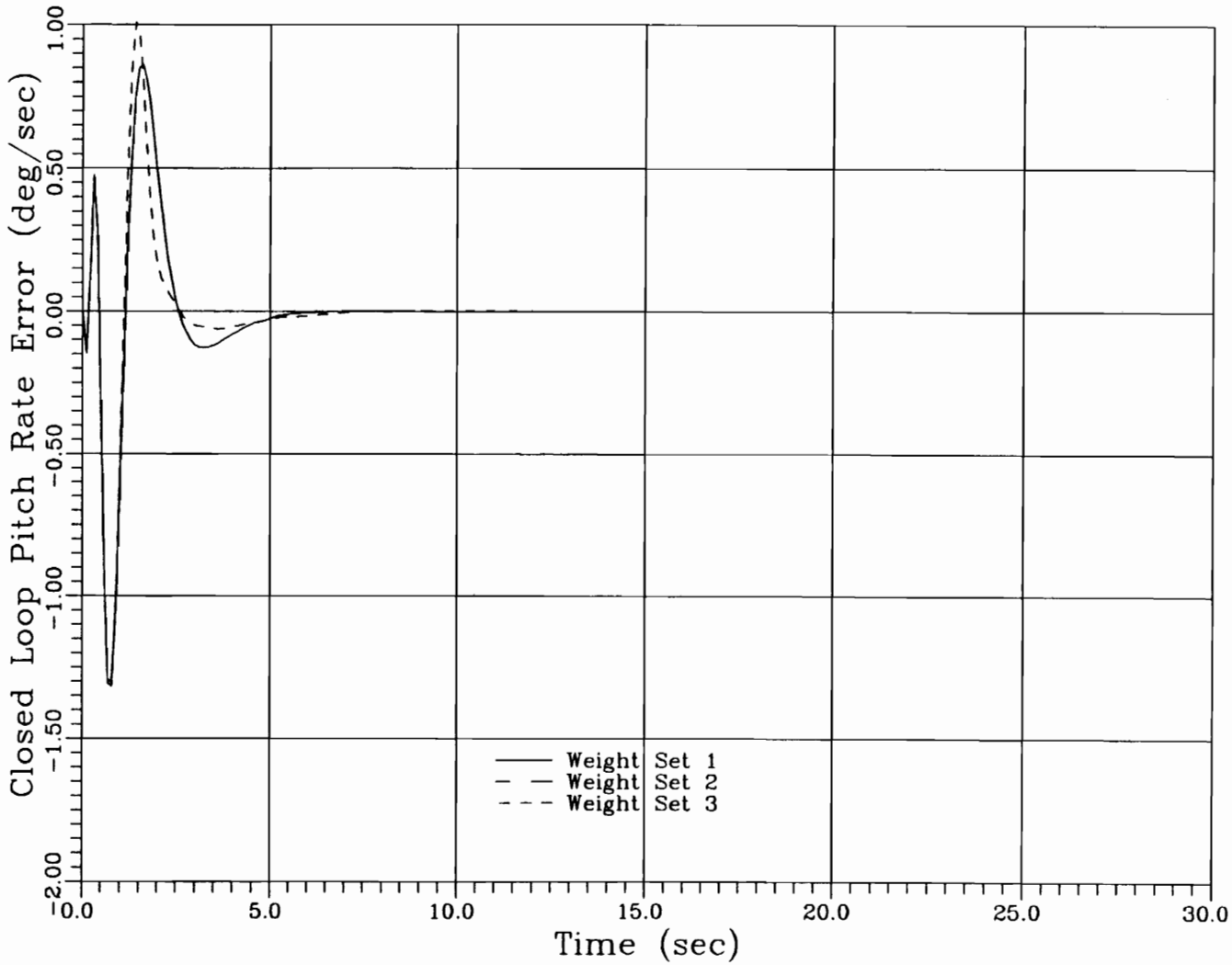

Figure 4.36 Closed Loop Pitch Rate Error
Nominal Geometry — Continuous Controller
Simultaneously Integrated Nominal

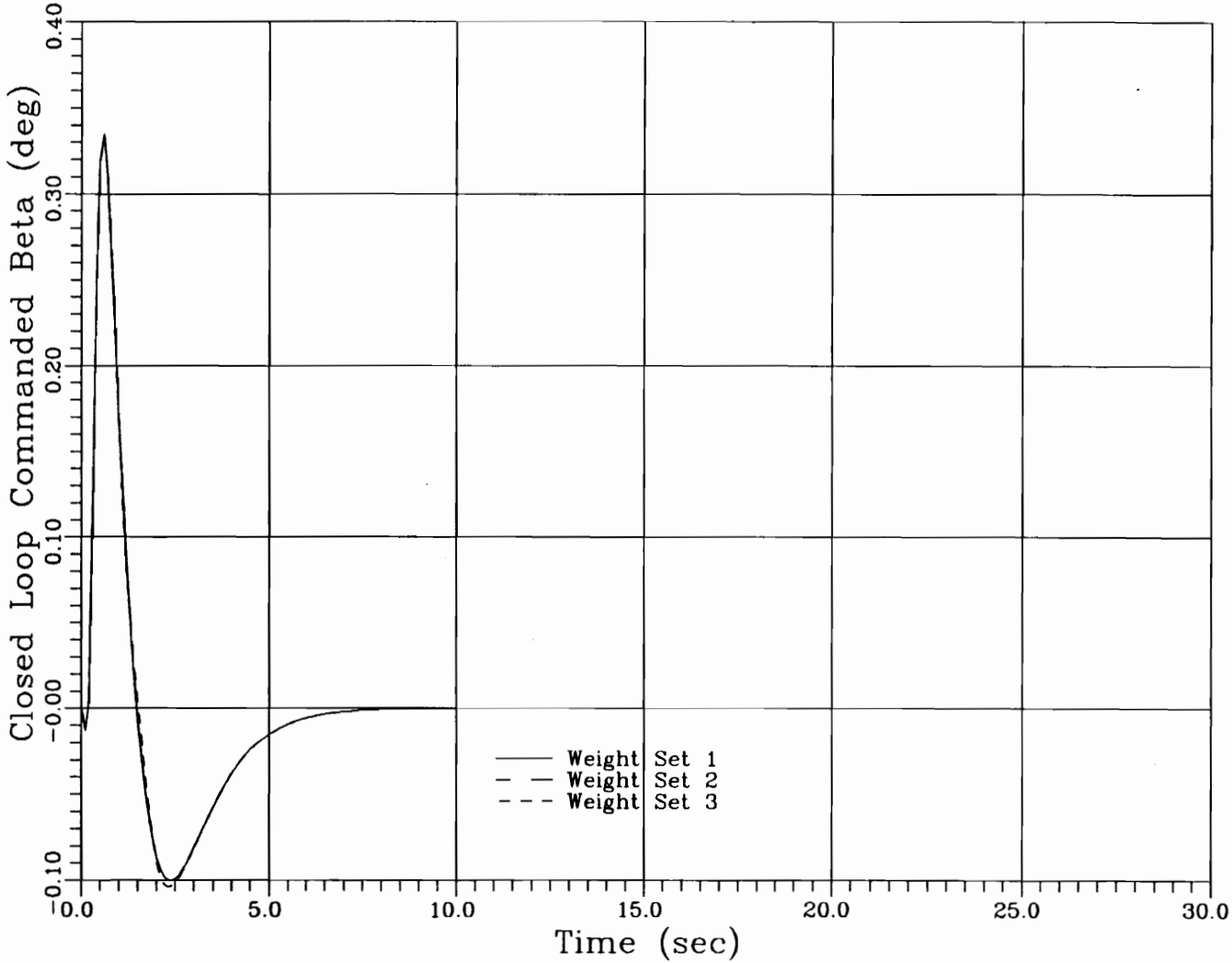

Figure 4.37 Applied Thrust Vector Control (β)
Nominal Geometry — Continuous Controller
Simultaneously Integrated Nominal

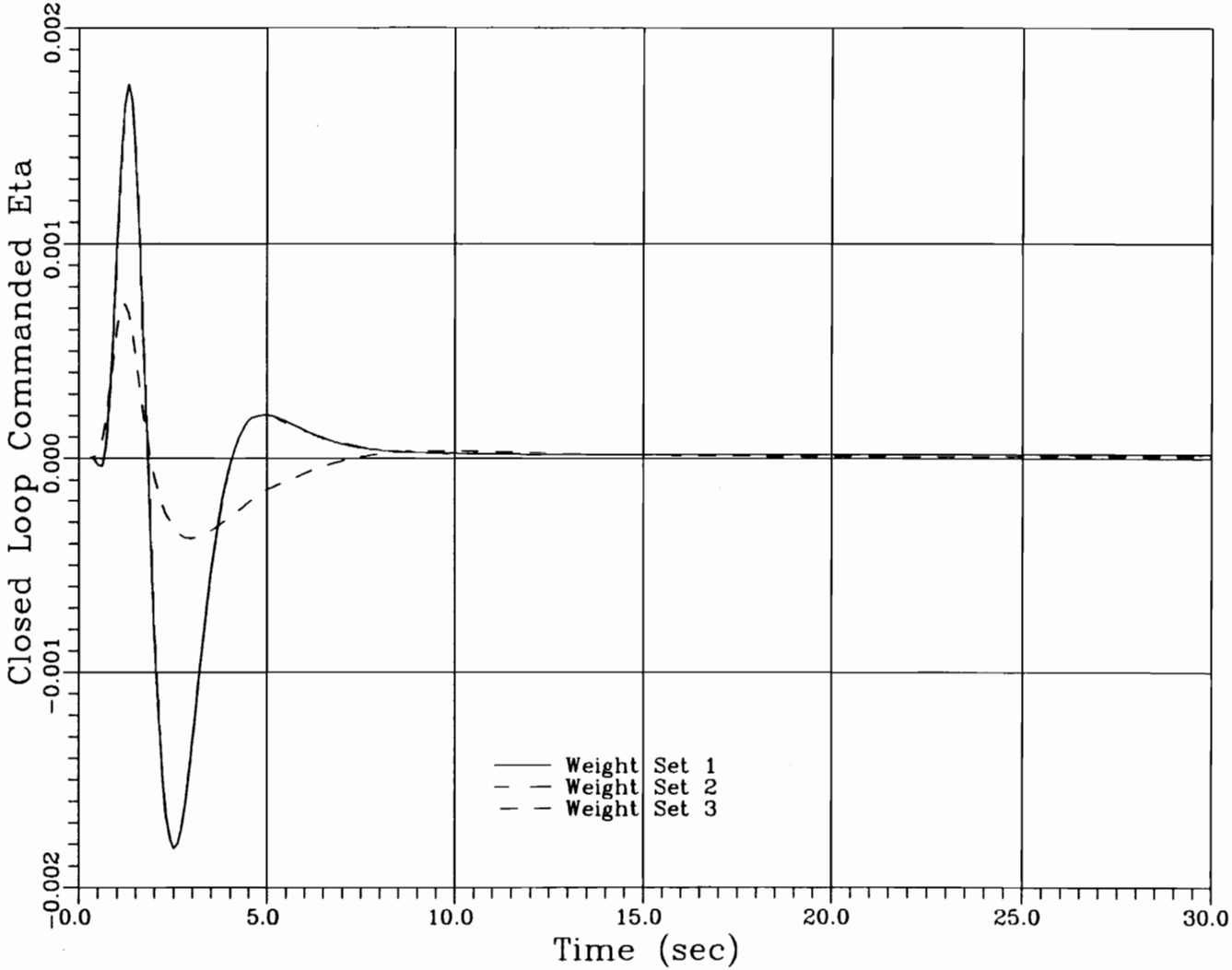

Figure 4.38 Applied Throttle Deviation (η)
Nominal Geometry — Continuous Controller
Simultaneously Integrated Nominal

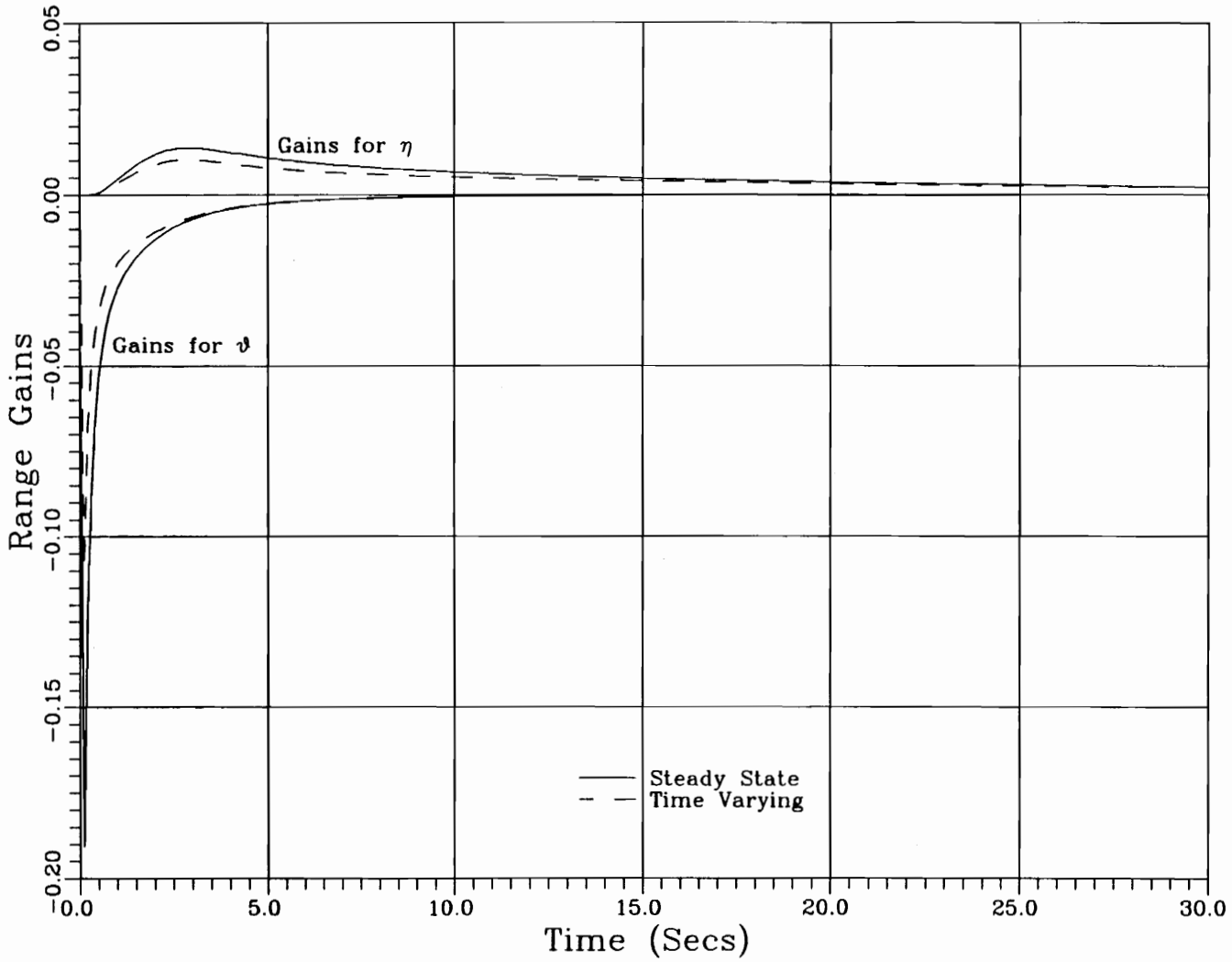

Figure 4.39 Discrete Range Gains — Set 1

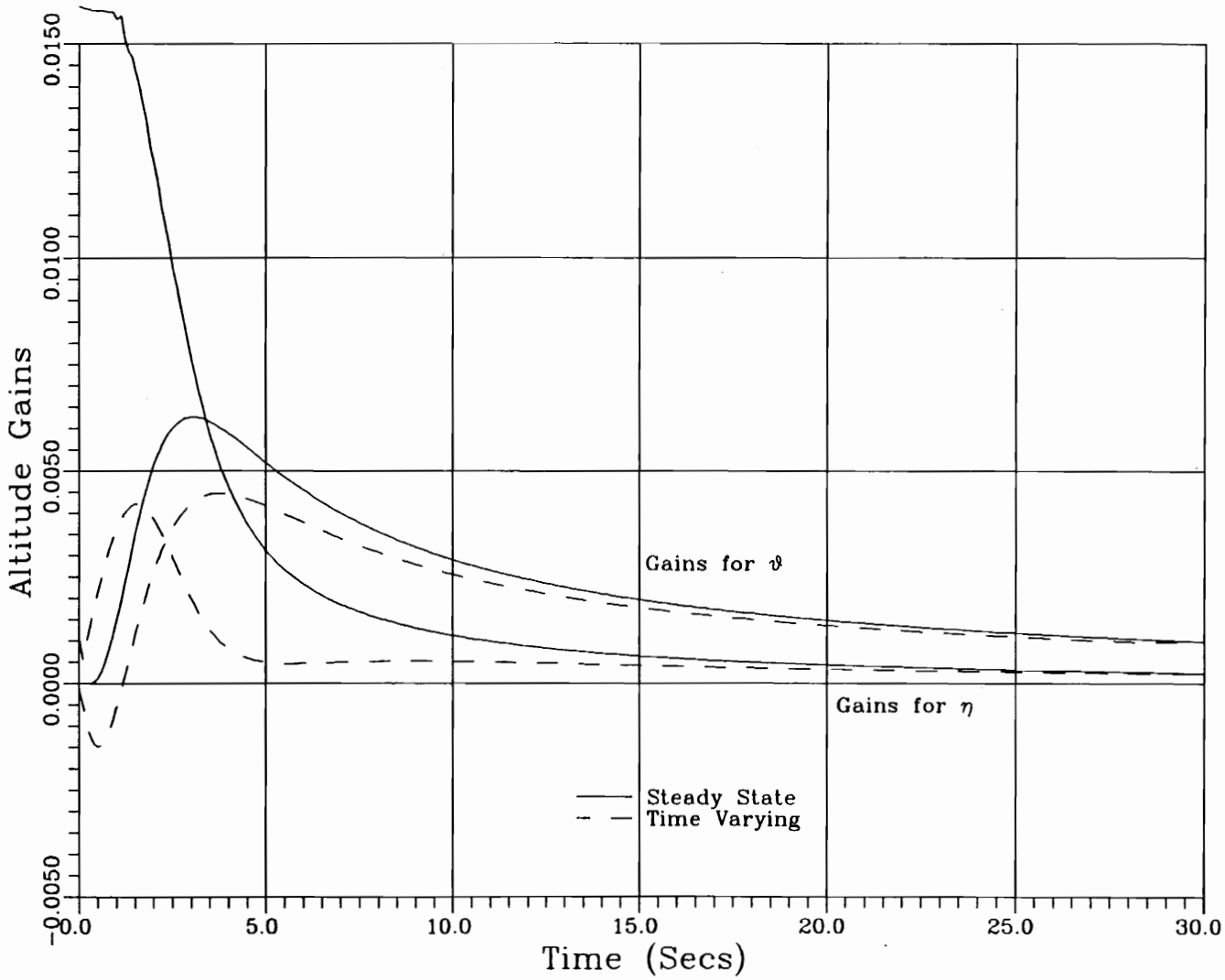

Figure 4.40 Discrete Altitude Gains — Set 1

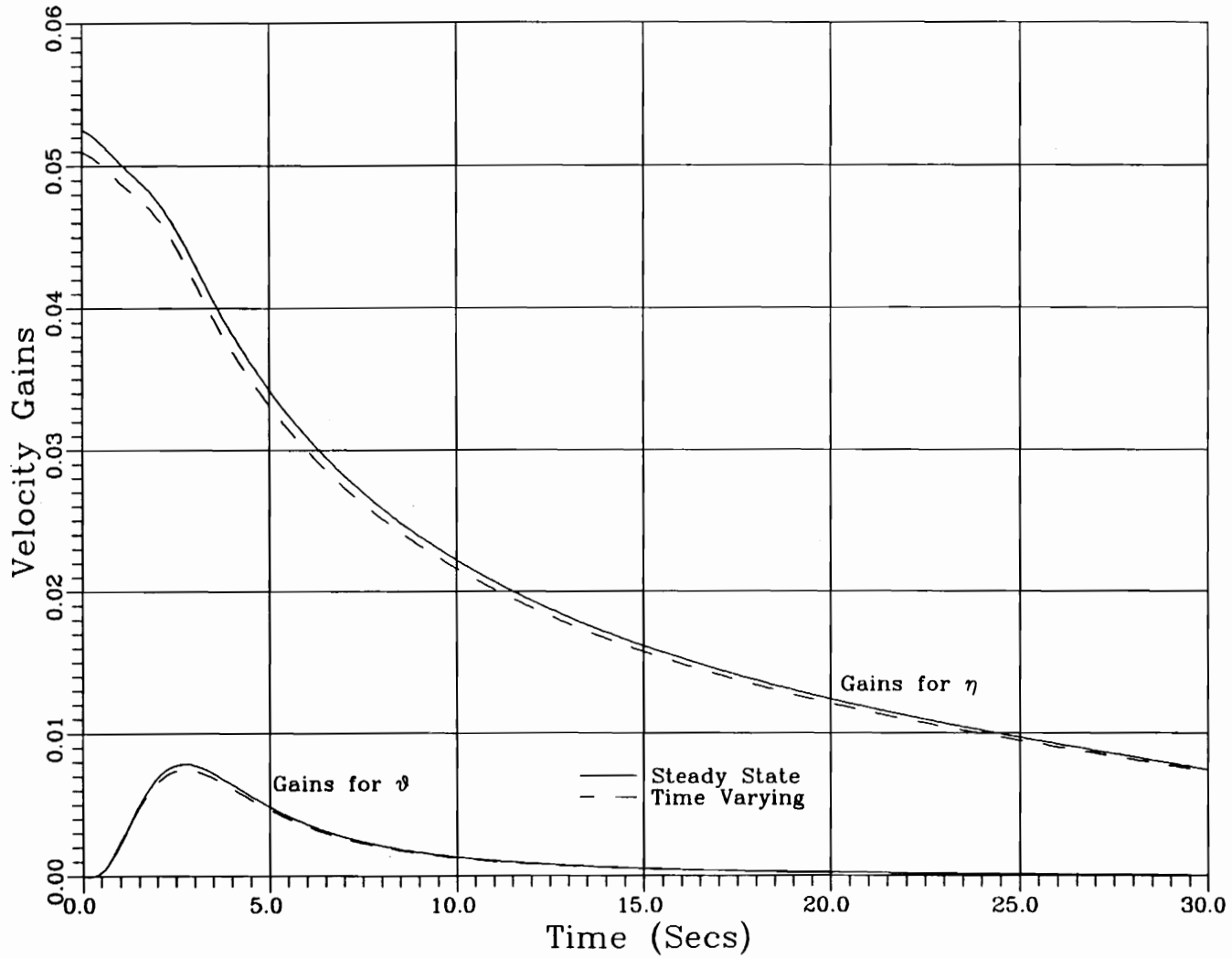

Figure 4.41 Discrete Velocity Gains — Set 1

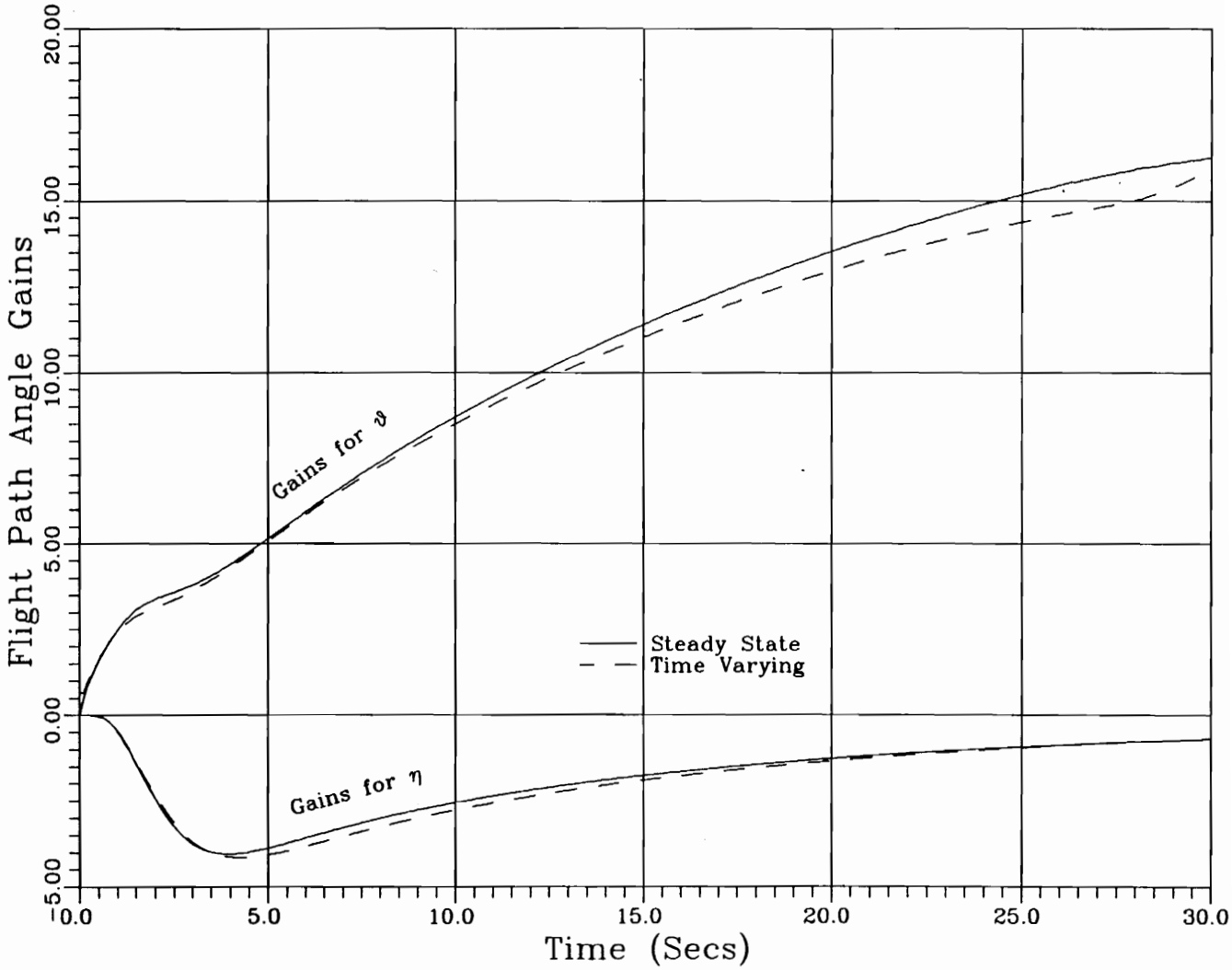

Figure 4.42 Discrete Flight Path Angle Gains — Set 1

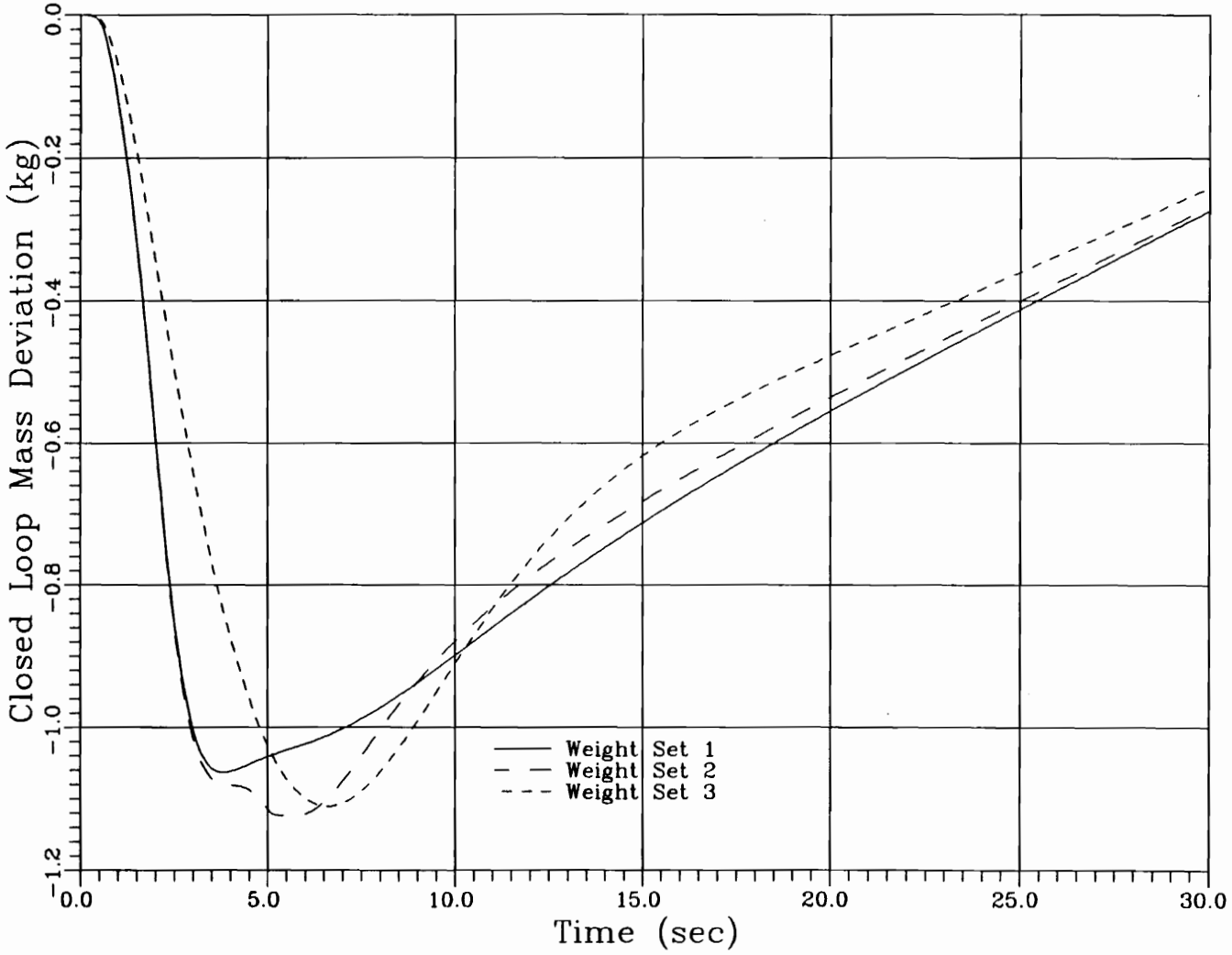

Figure 4.43 Closed Loop Mass Error
0.1 m Thrust Offset — Discrete Controller

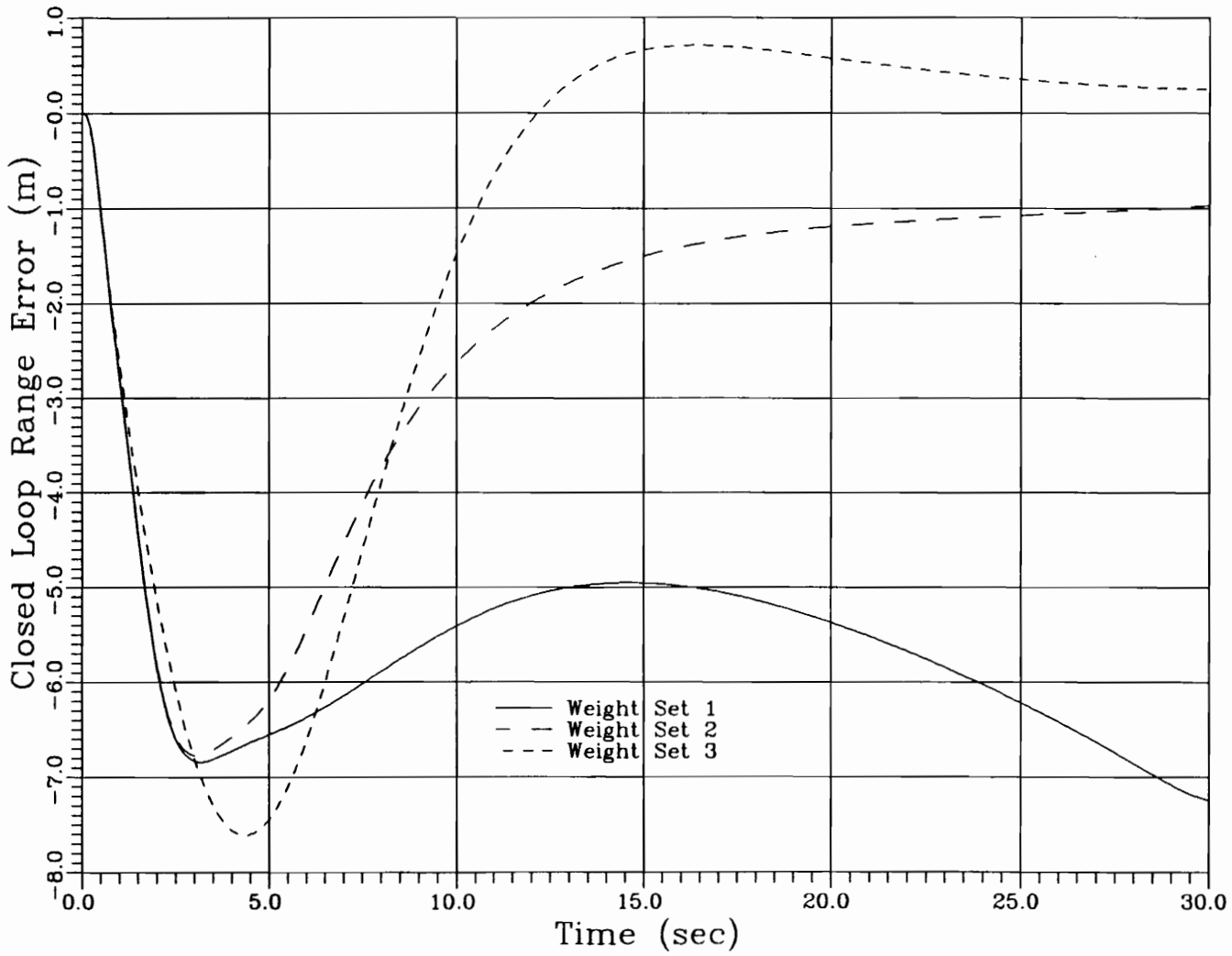

Figure 4.44 Closed Loop Range Error
0.1 m Thrust Offset — Discrete Controller

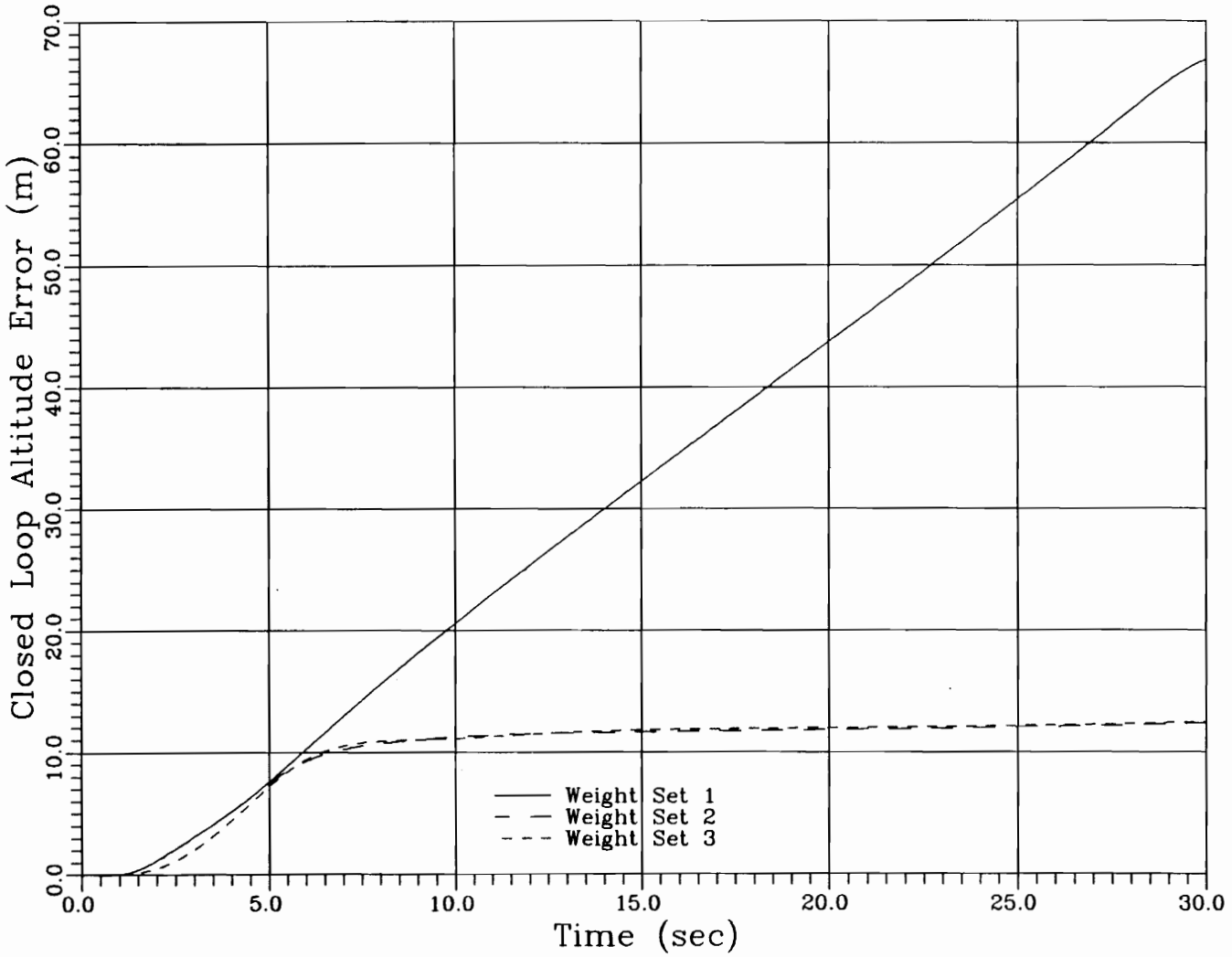

Figure 4.45 Closed Loop Altitude Error
0.1 m Thrust Offset — Discrete Controller

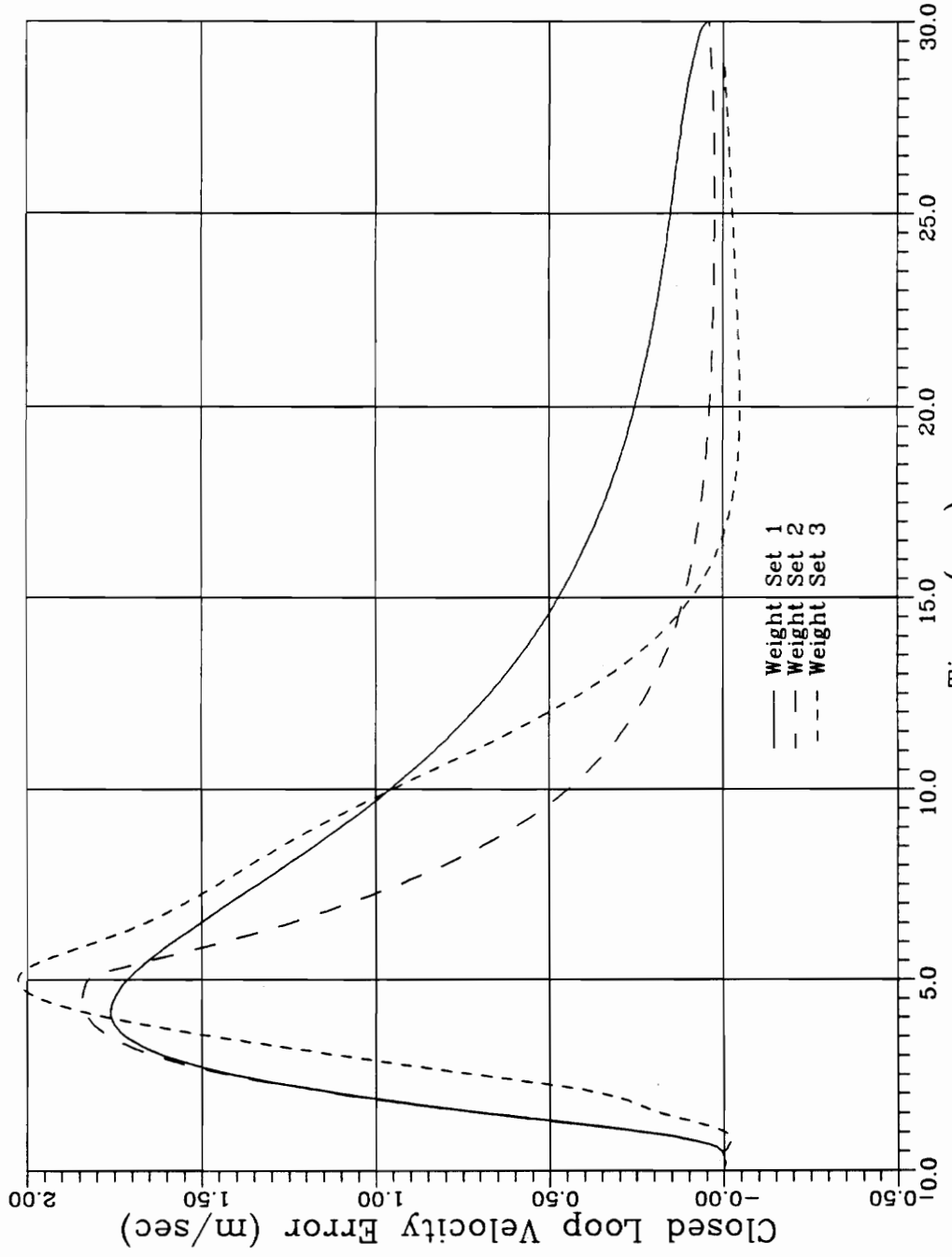

Figure 4.46 Closed Loop Velocity Error
0.1 *m* Thrust Offset — Discrete Controller

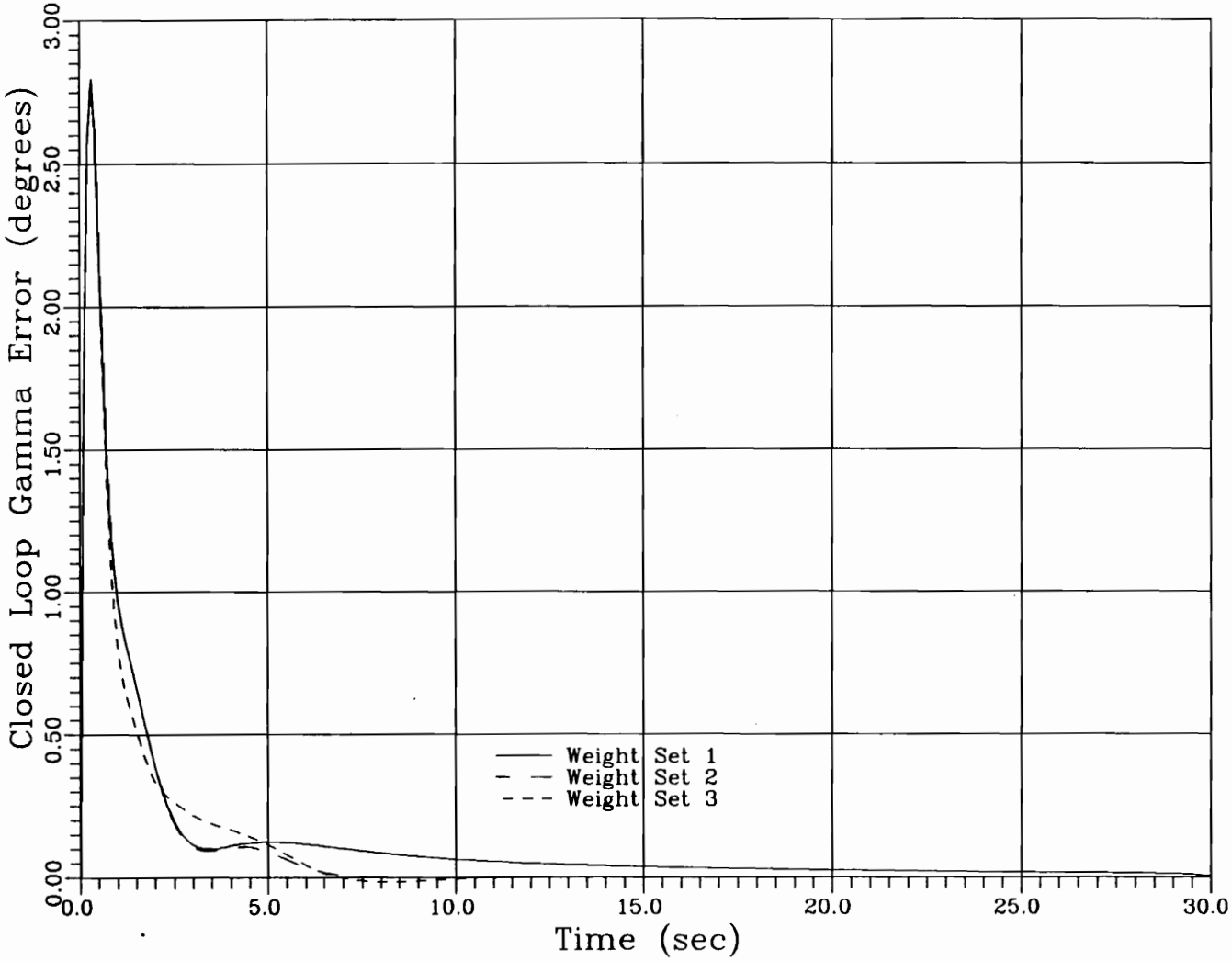

Figure 4.47 Closed Loop Flight Path Angle Error
0.1 m Thrust Offset — Discrete Controller

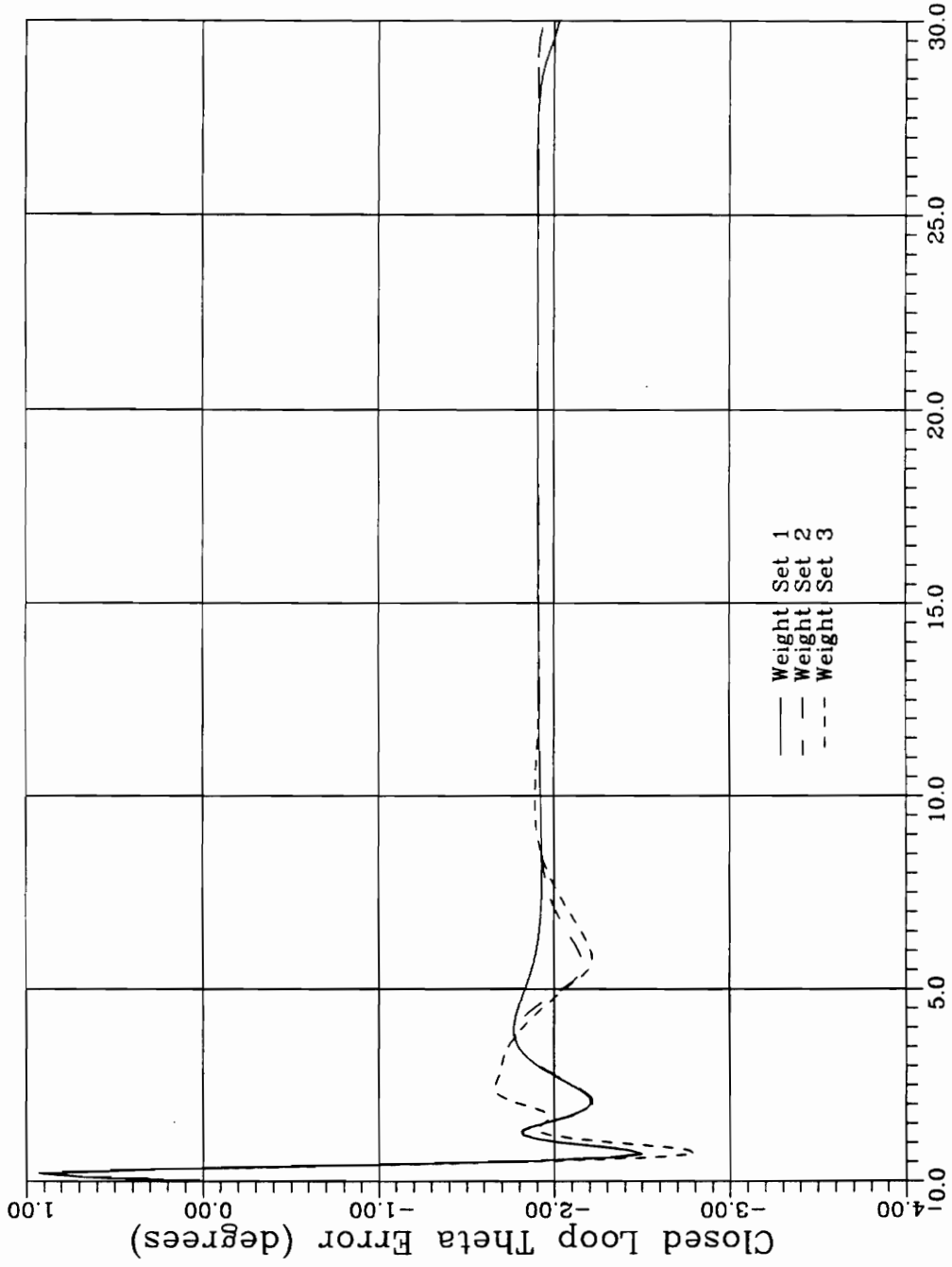

Figure 4.48 Closed Loop Pitch Attitude Error
0.1 m Thrust Offset — Discrete Controller

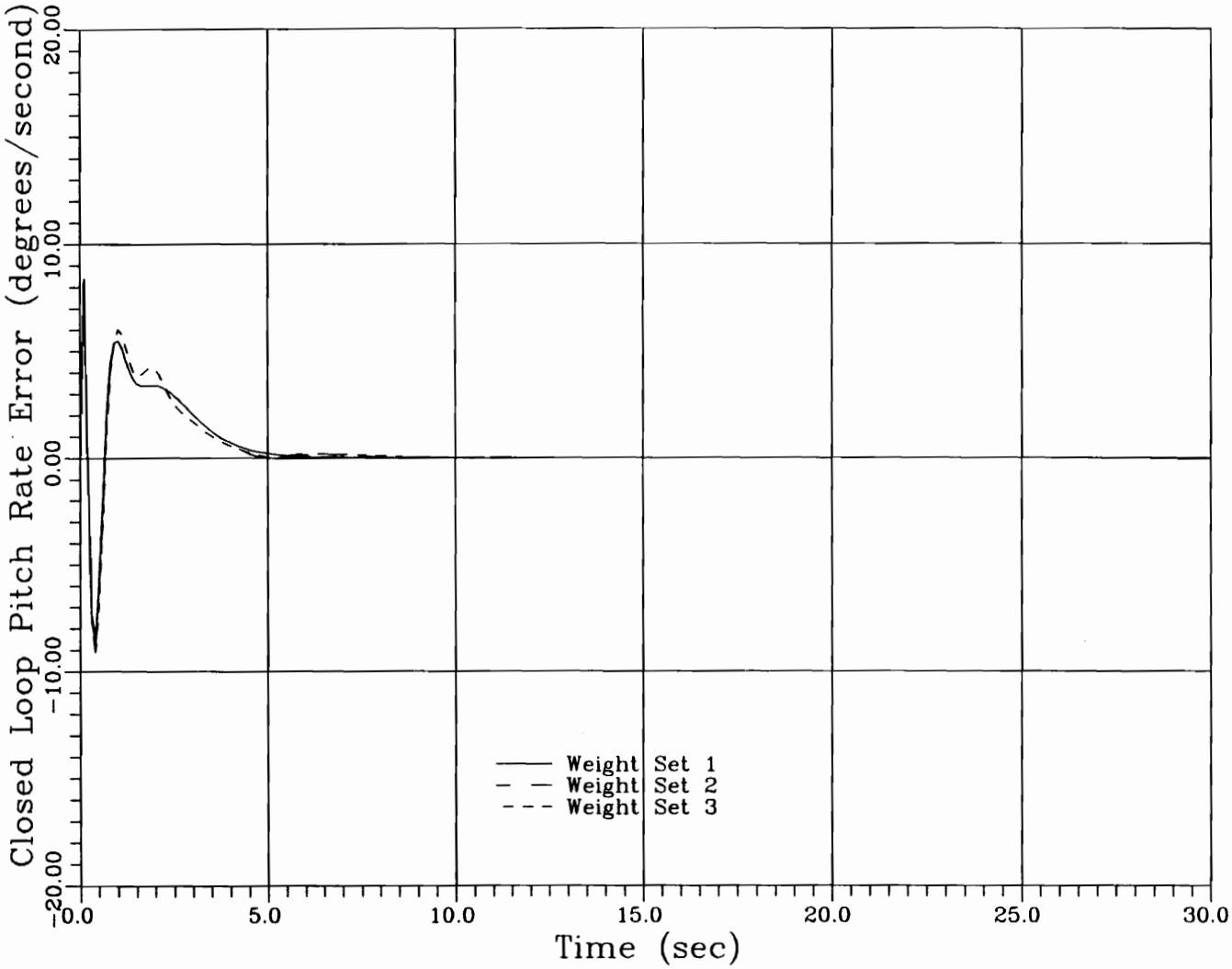

Figure 4.49 Closed Loop Pitch Rate Error
0.1 m Thrust Offset — Discrete Controller

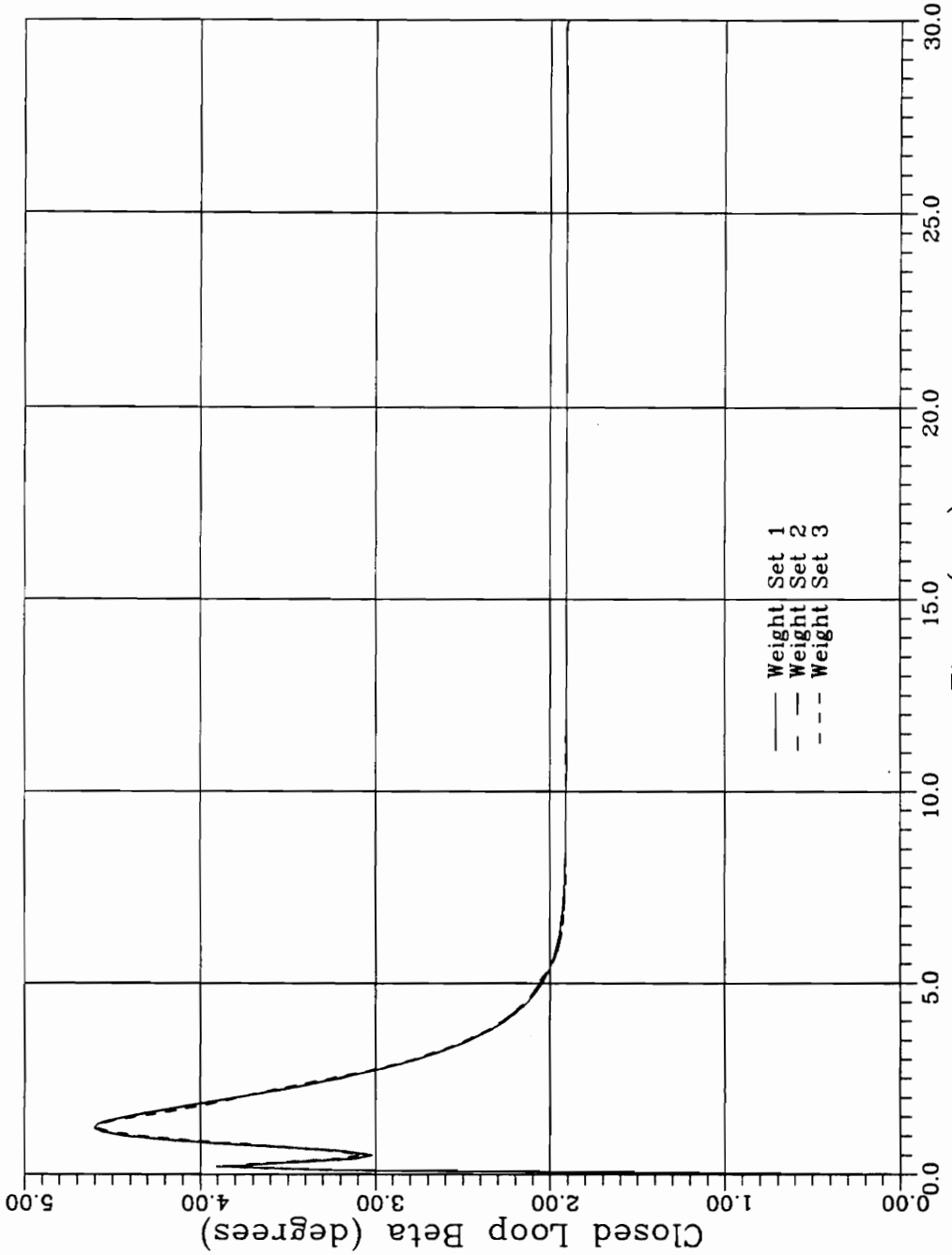

Figure 4.50 Applied Thrust Vector Control (β)
0.1 m Thrust Offset — Discrete Controller

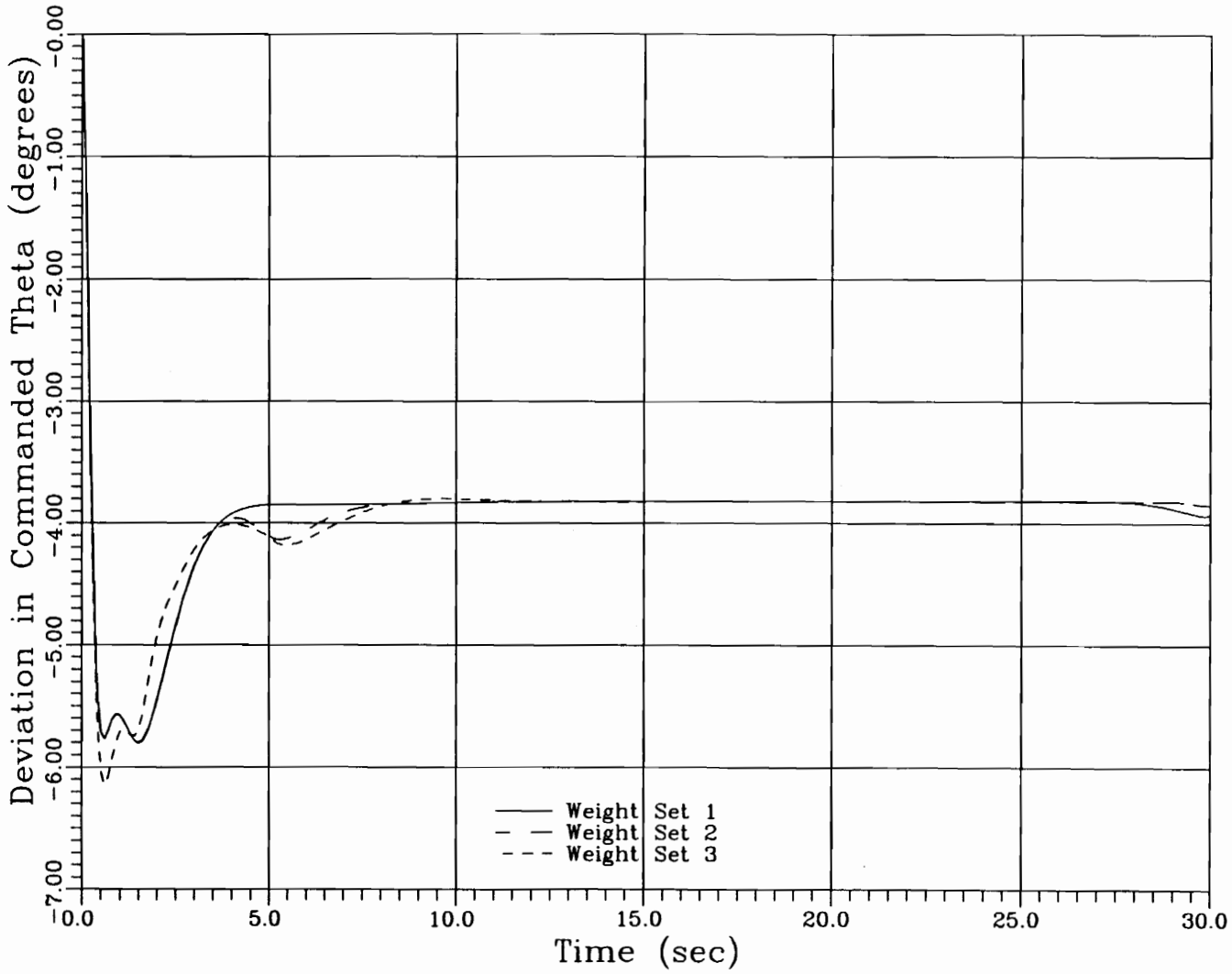

Figure 4.51 Applied Pitch Attitude Deviation
0.1 m Thrust Offset — Discrete Controller

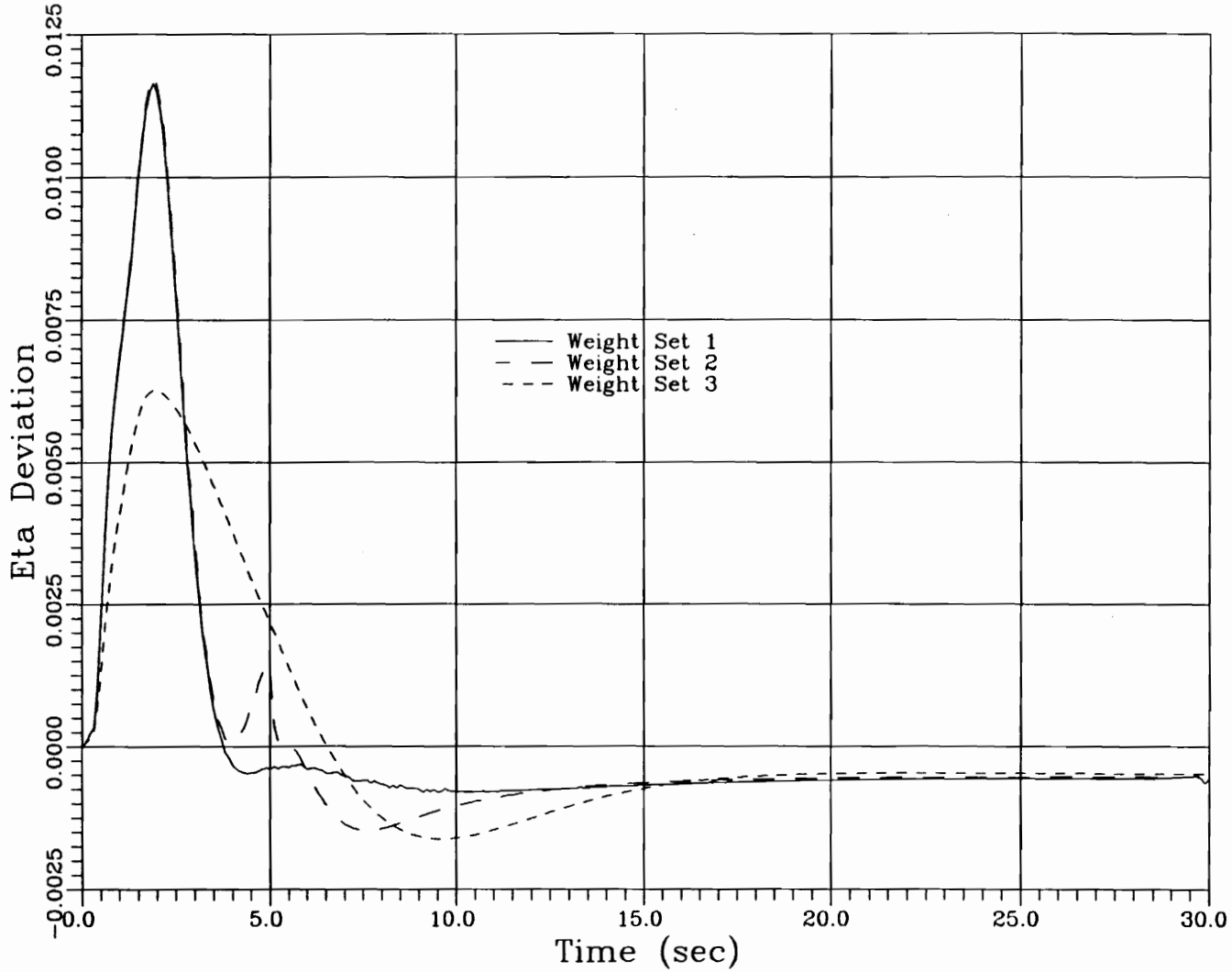

Figure 4.52 Applied Throttle Deviation (η)
0.1 m Thrust Offset — Discrete Controller

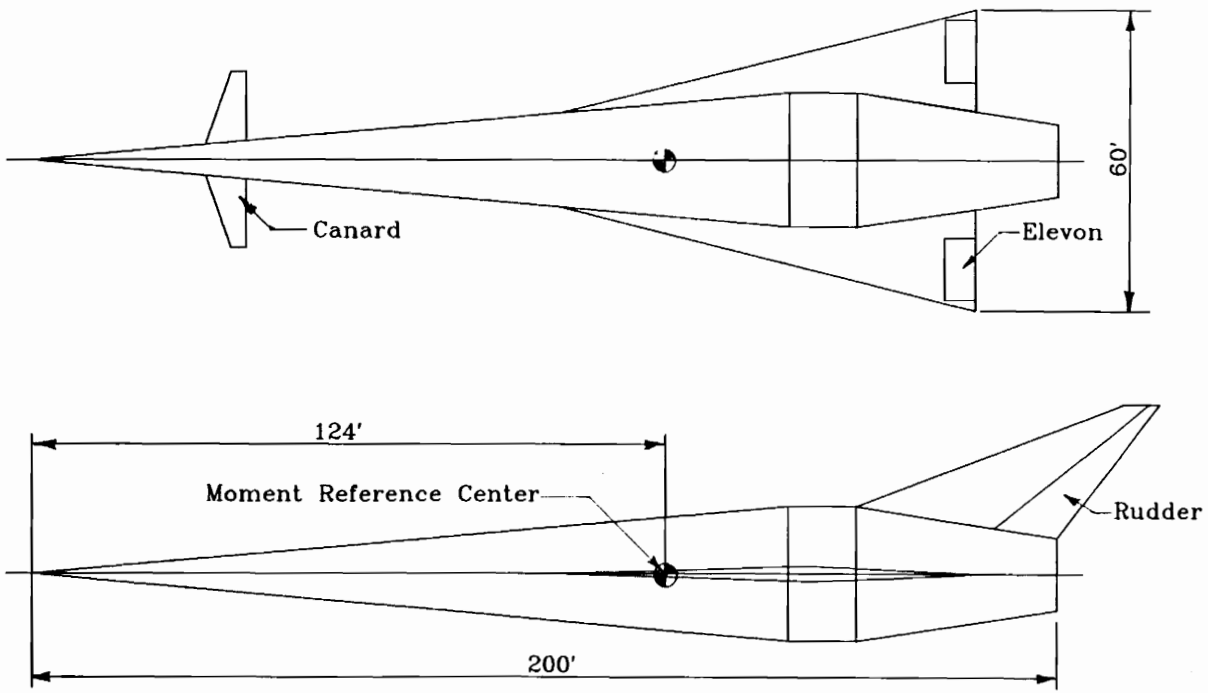

Figure 4.53 Complex Model Configuration Geometry

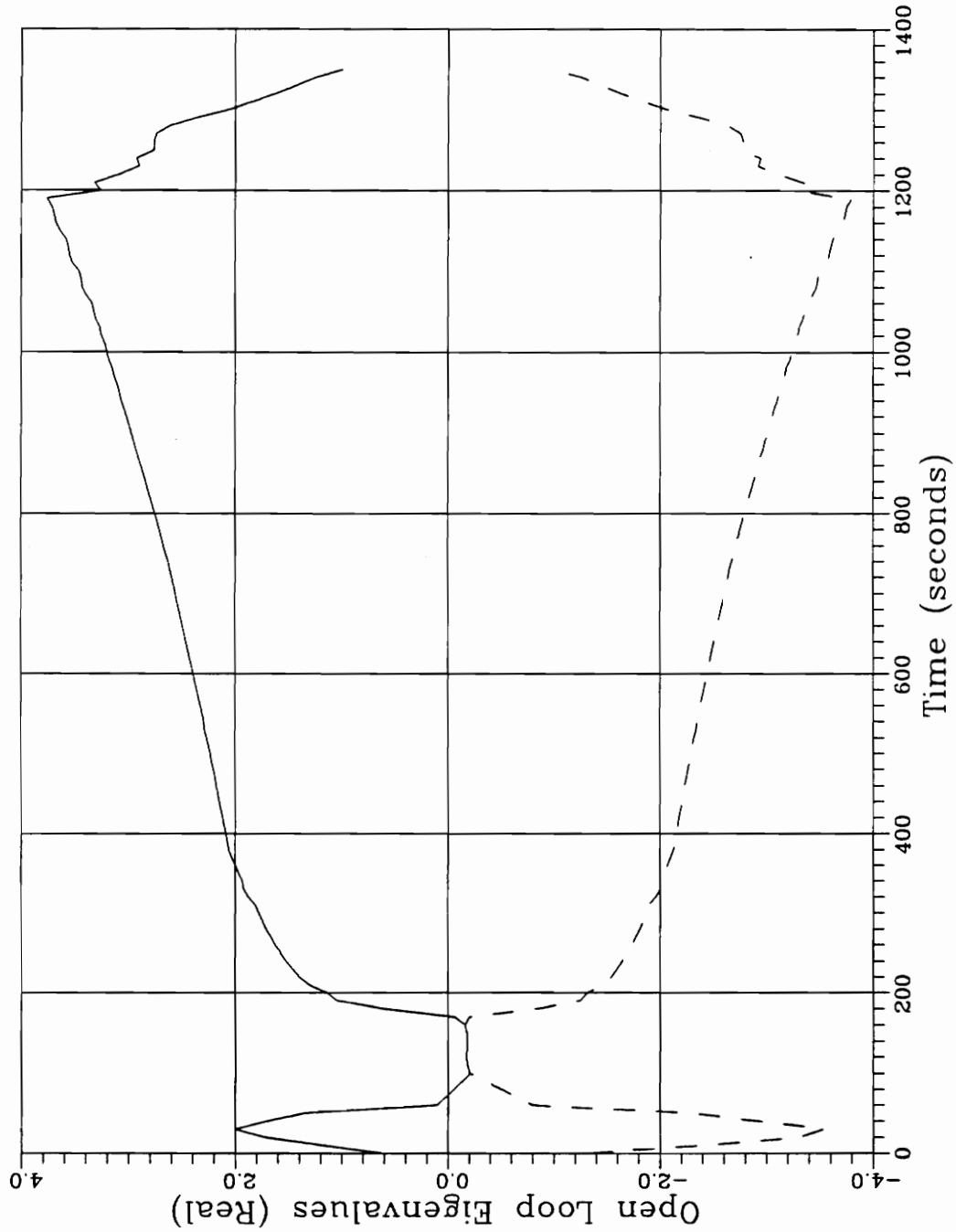

Figure 4.54 Open Loop Pitch Mode Eigenvalues — Real Part

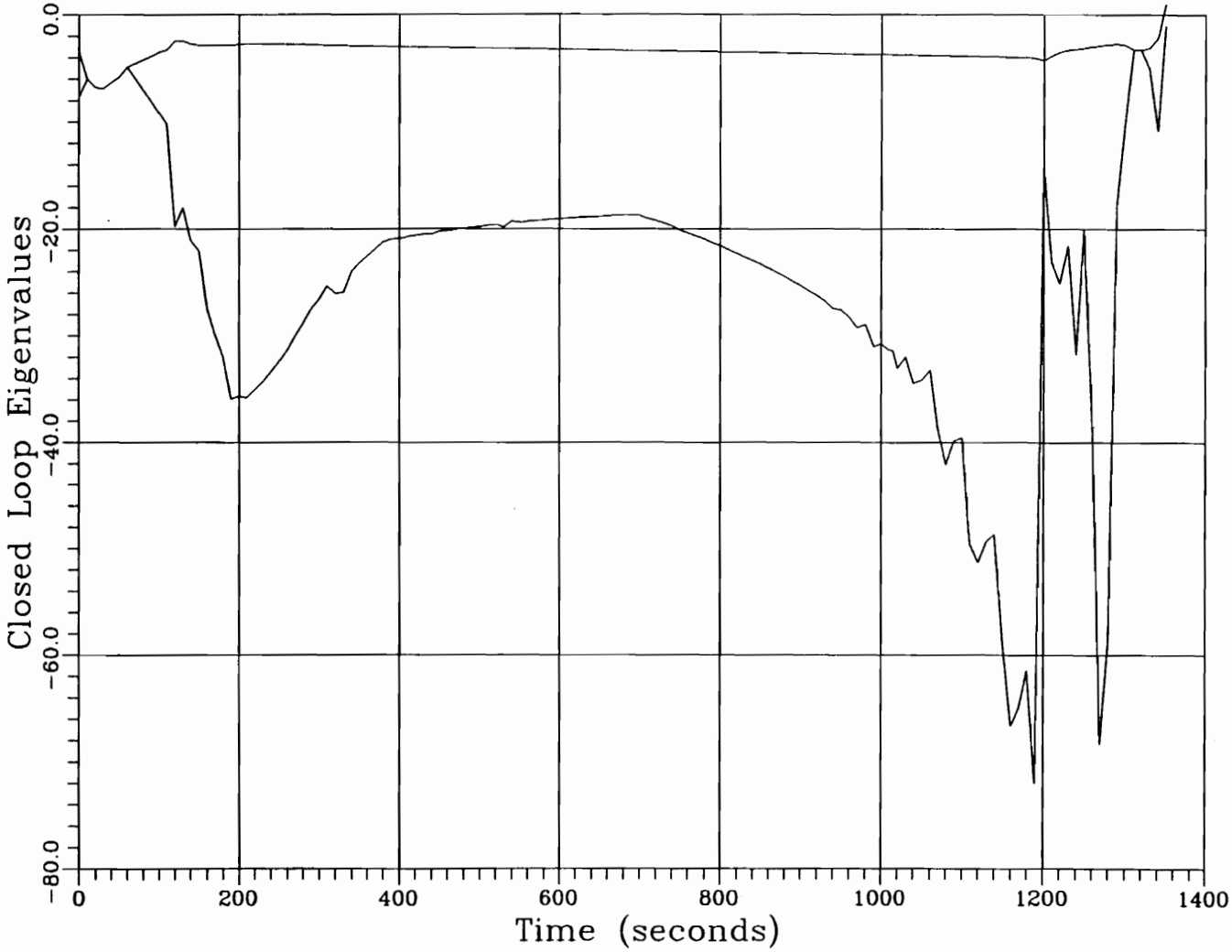

Figure 4.55 Closed Loop Pitch Mode Eigenvalues — Real Part

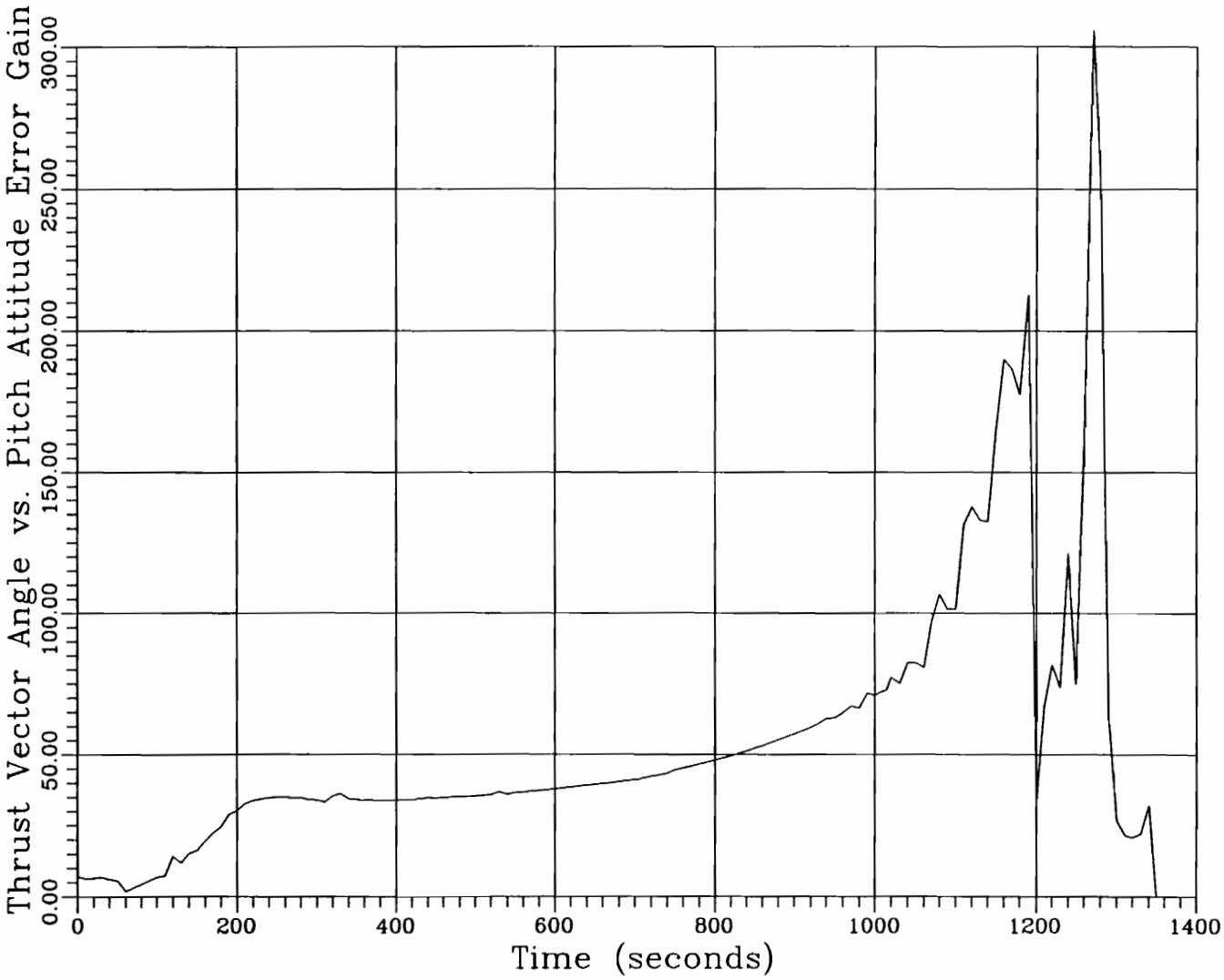

Figure 4.56 β vs. θ Inner Loop Gain

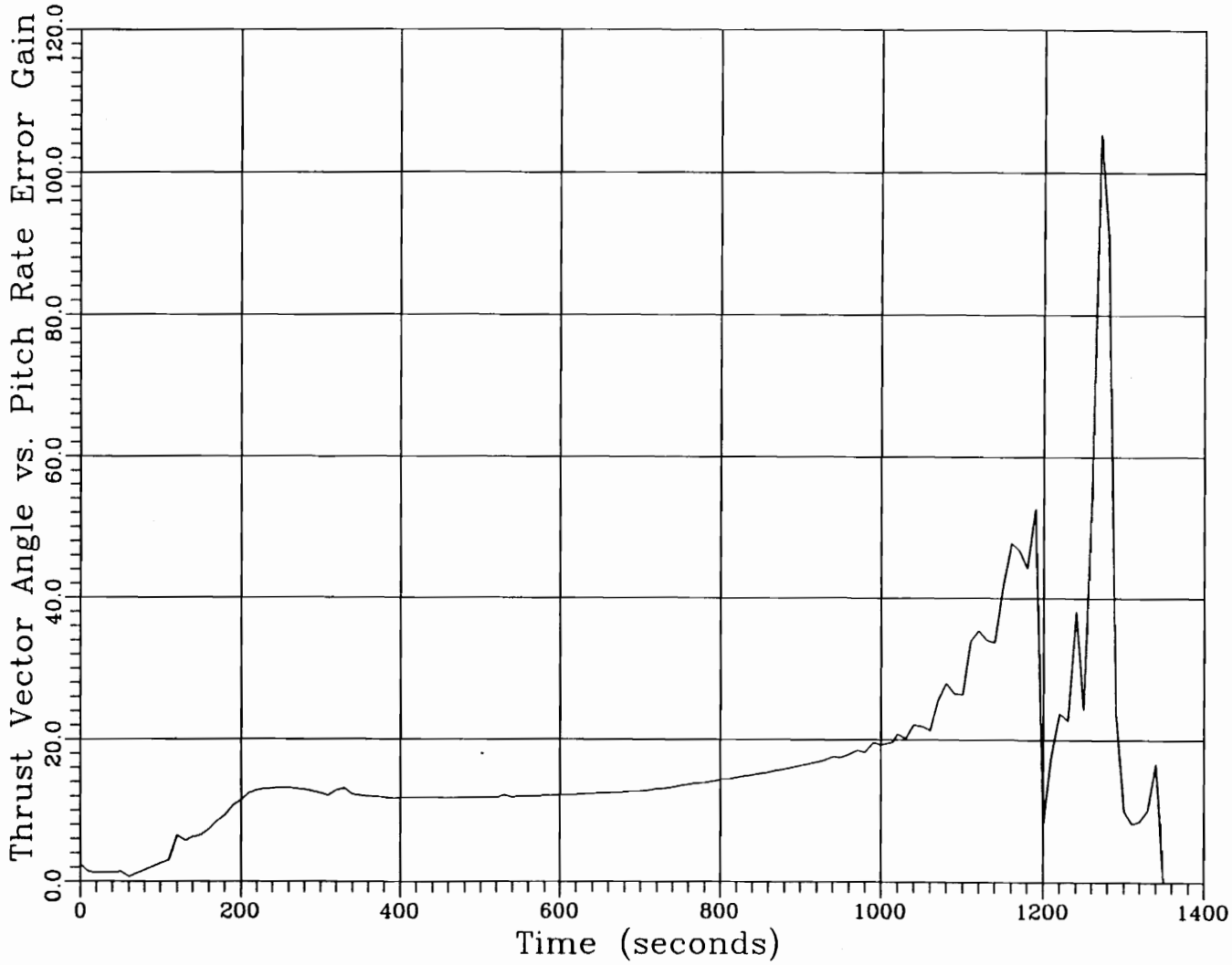

Figure 4.57 β vs. q Inner Loop Gain

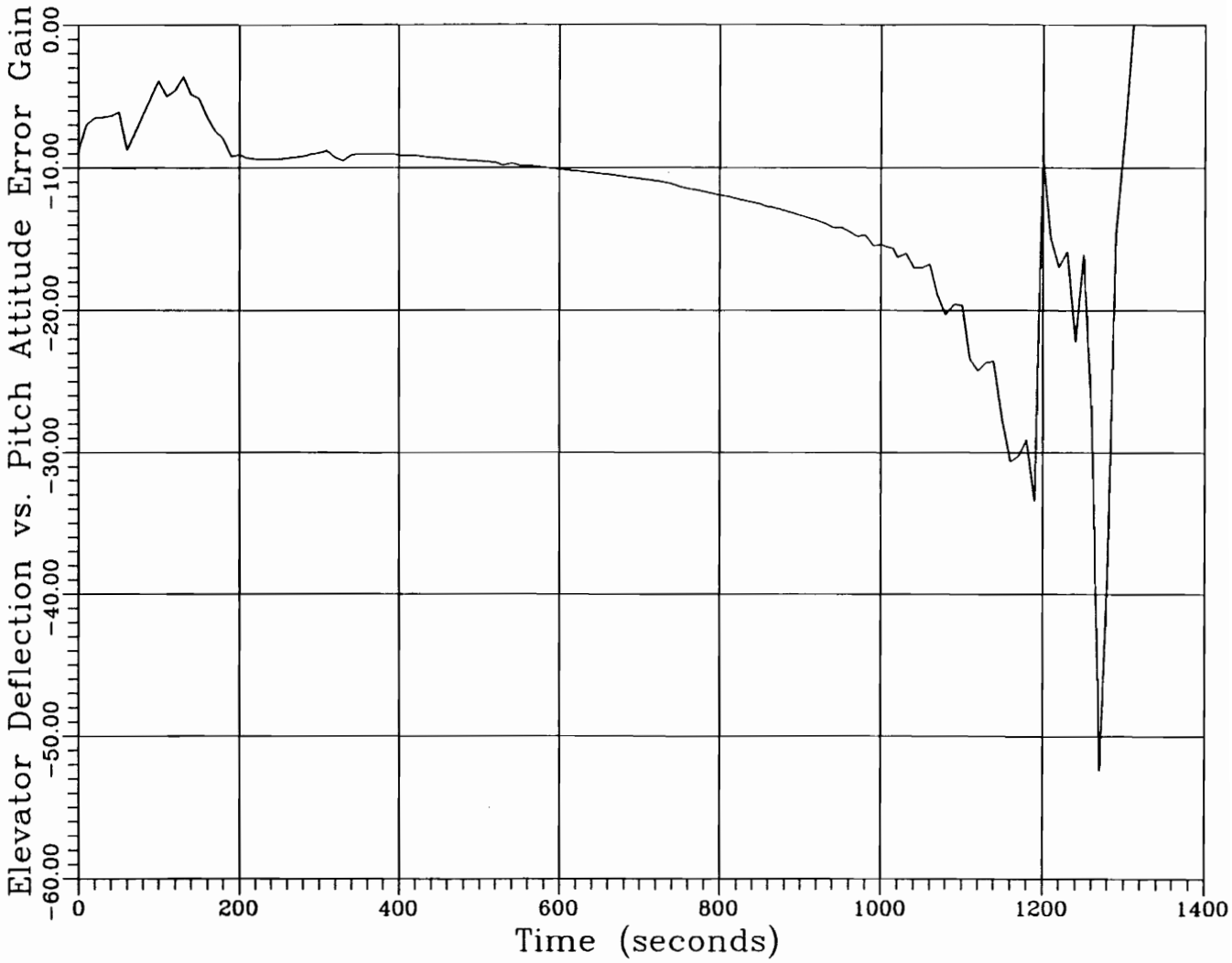

Figure 4.58 δ vs. θ Inner Loop Gain

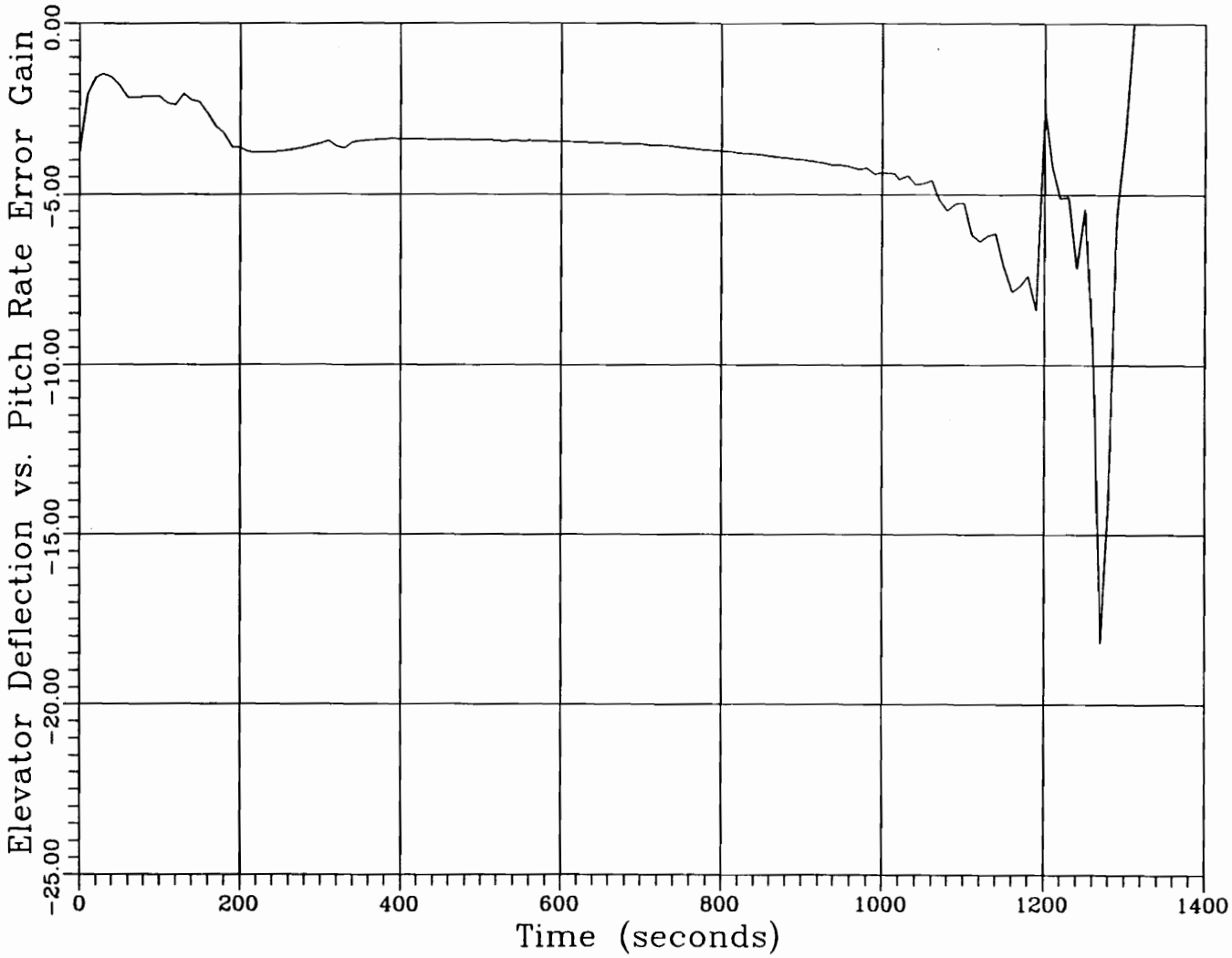

Figure 4.59 δ vs. q Inner Loop Gain

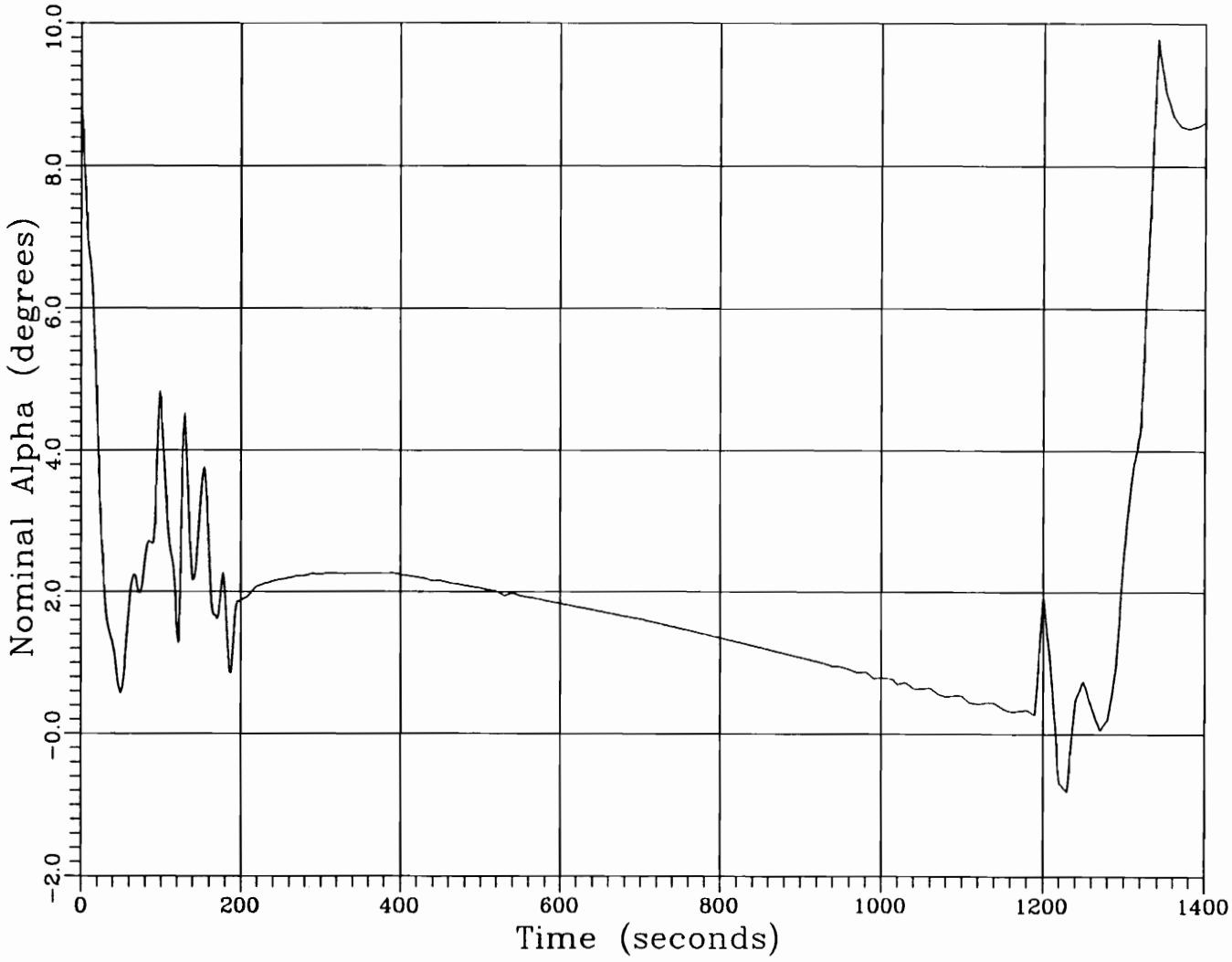

Figure 4.60 Nominal Angle of Attack

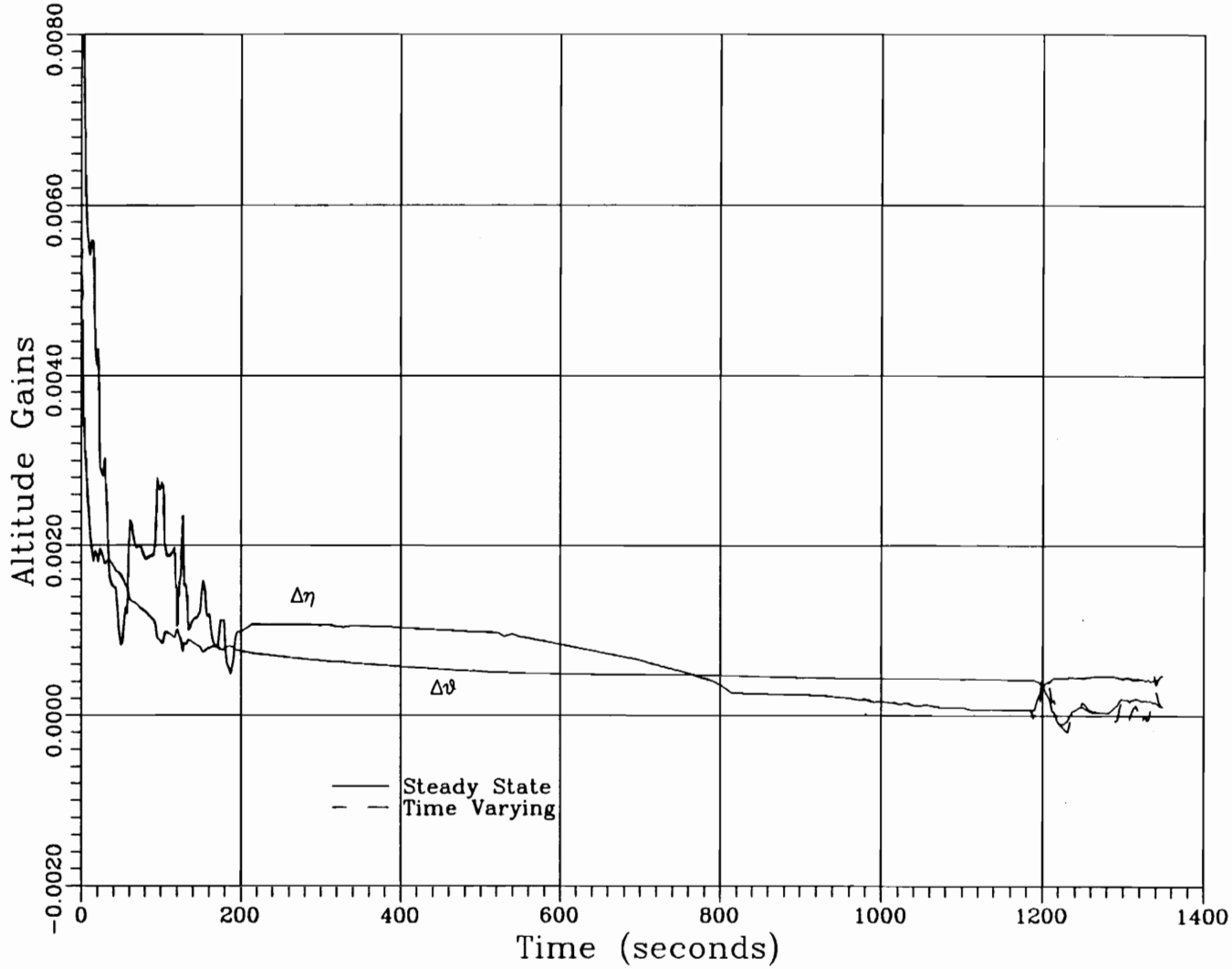

Figure 4.61 Outer Loop Altitude Gains

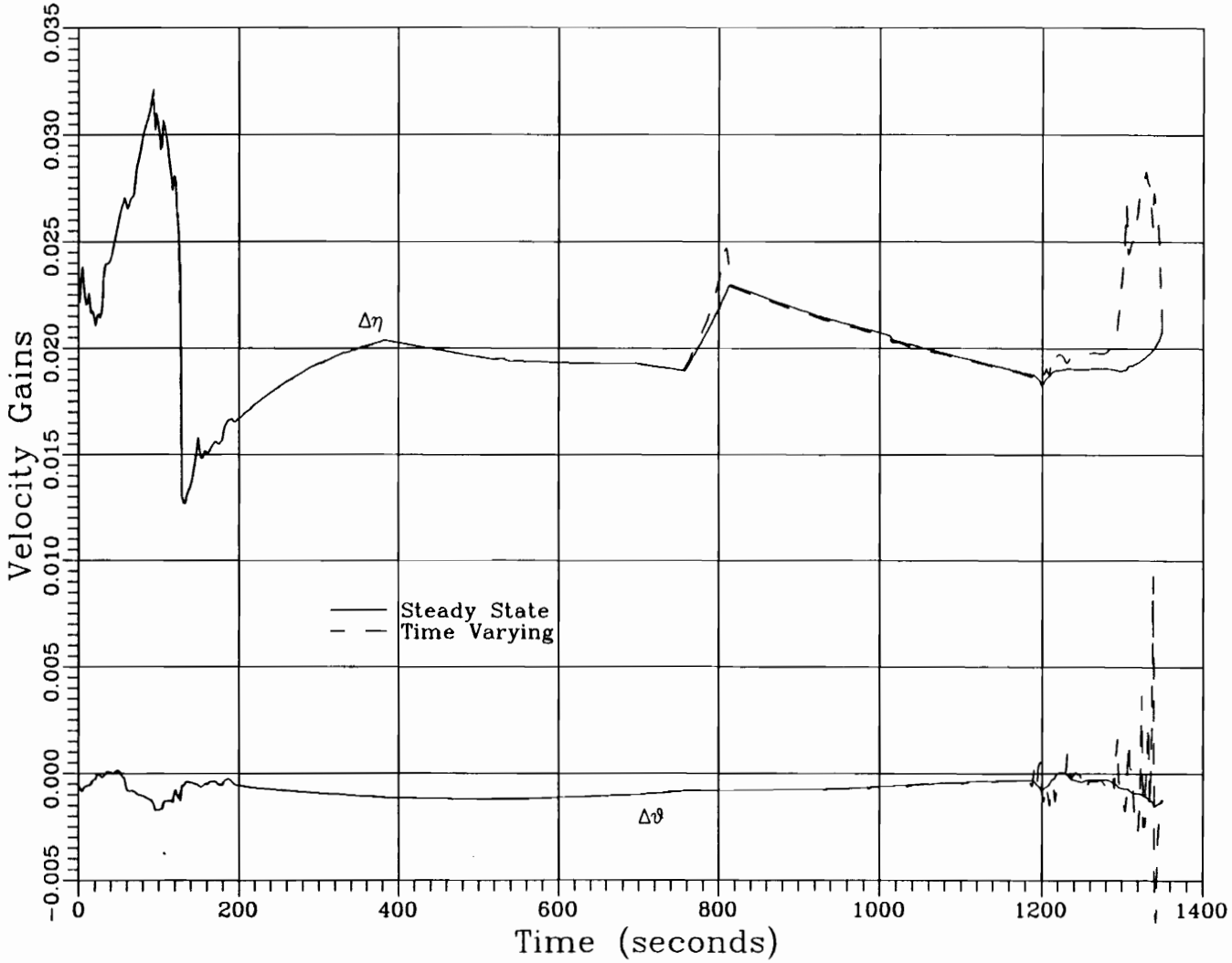

Figure 4.62 Outer Loop Velocity Gains

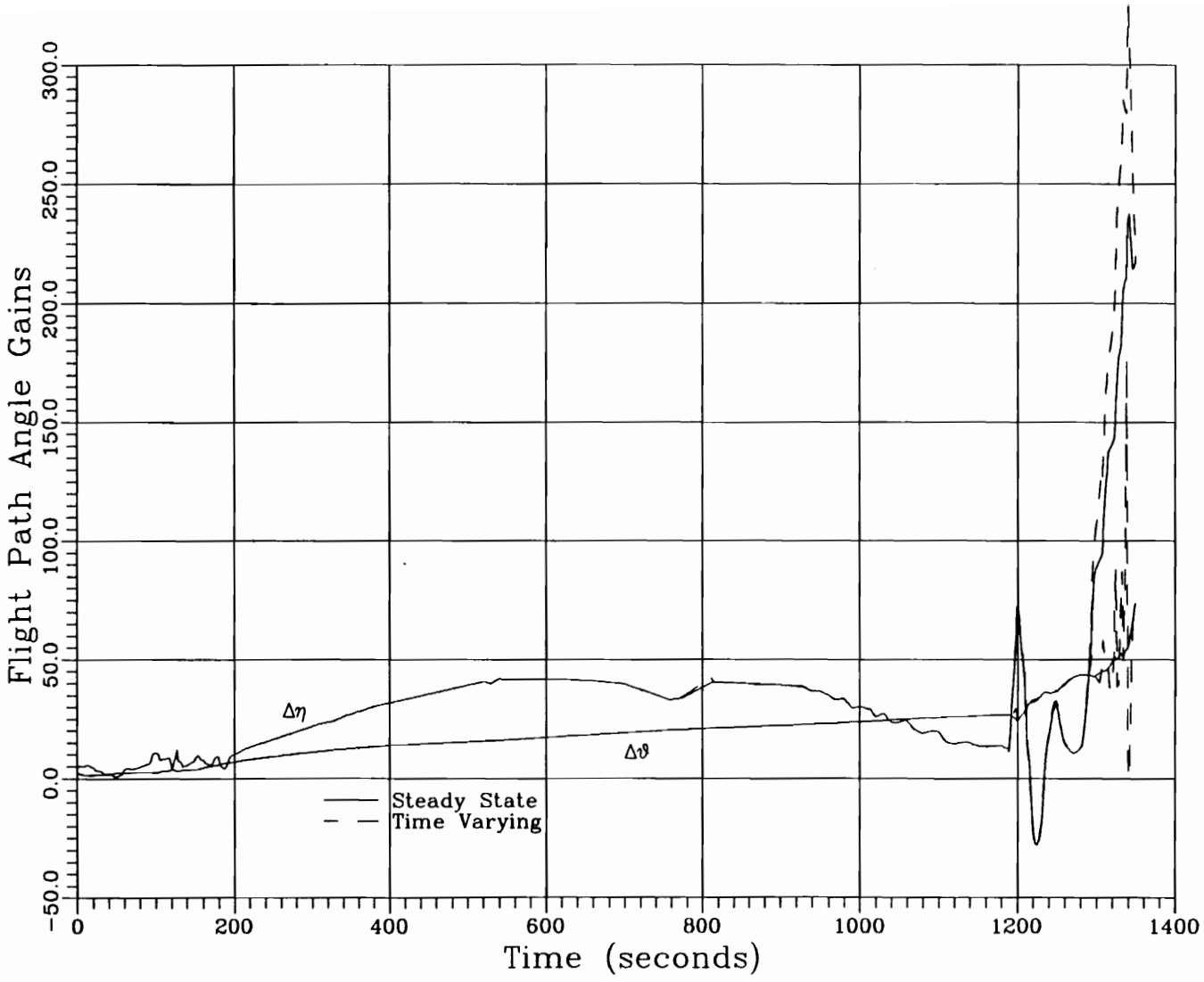

Figure 4.63 Outer Loop Flight Path Angle Gains

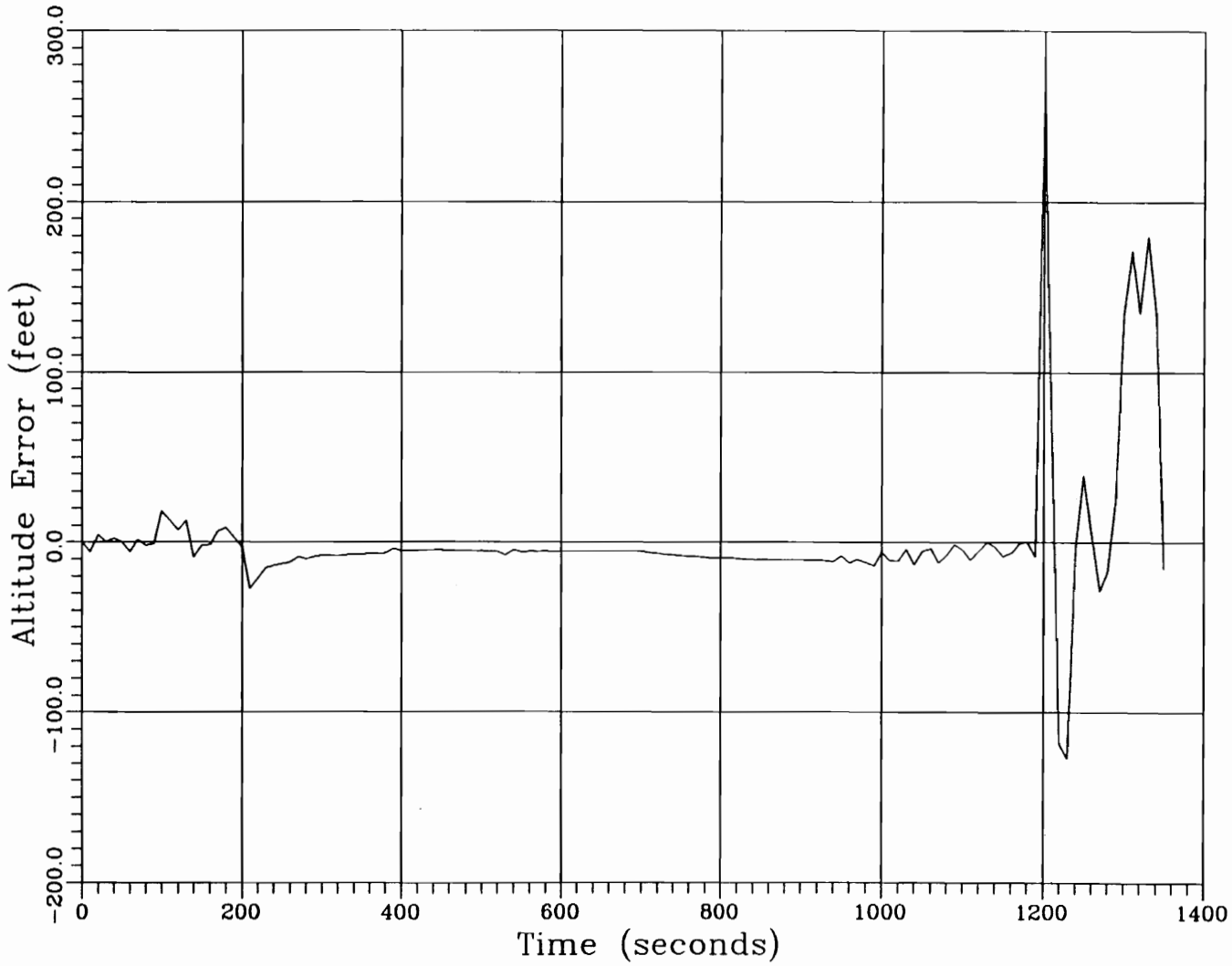

Figure 4.64 Closed Loop Altitude Error at Nominal Time Points Still Atmosphere

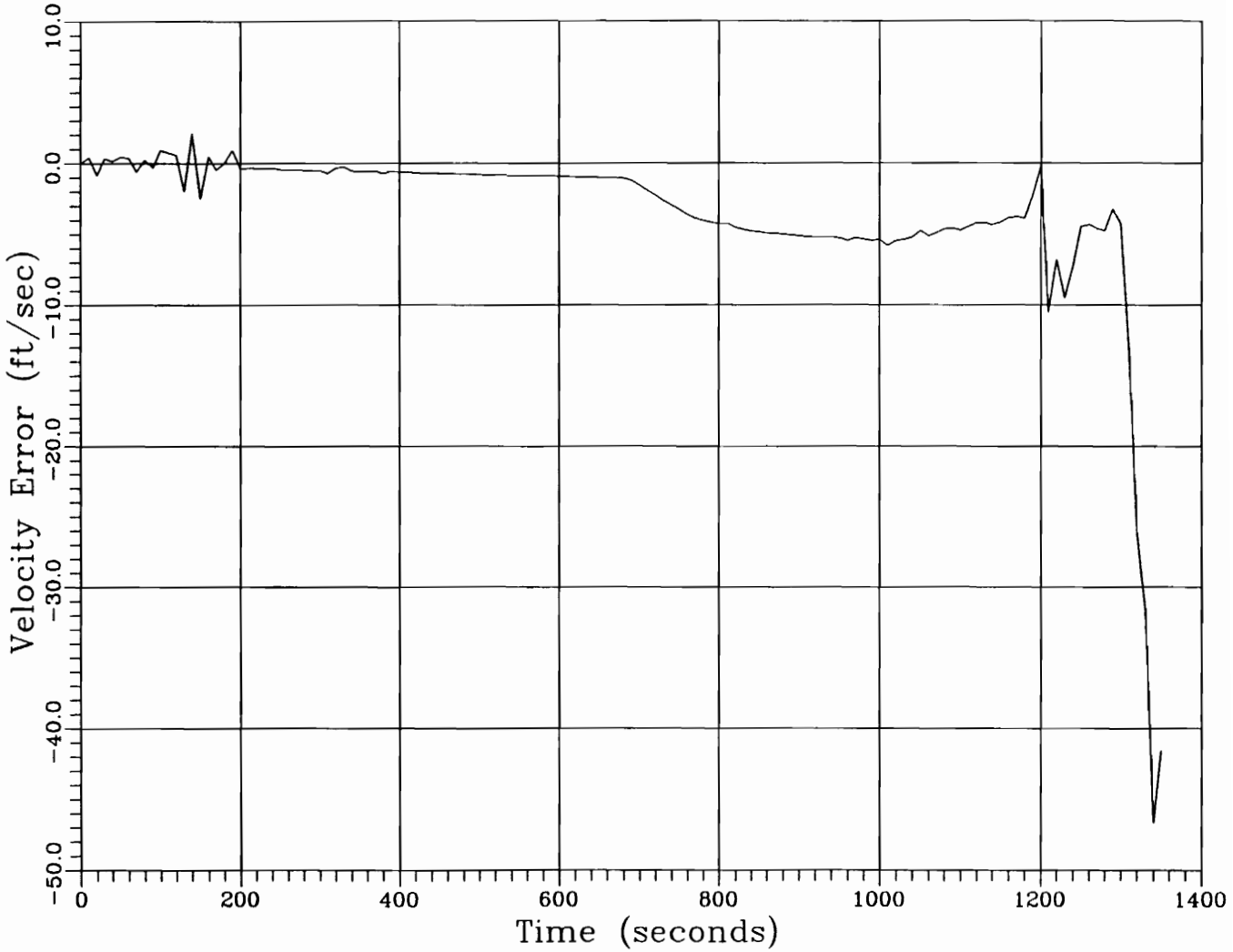

Figure 4.65 Closed Loop Velocity Error at Nominal Time Points Still Atmosphere

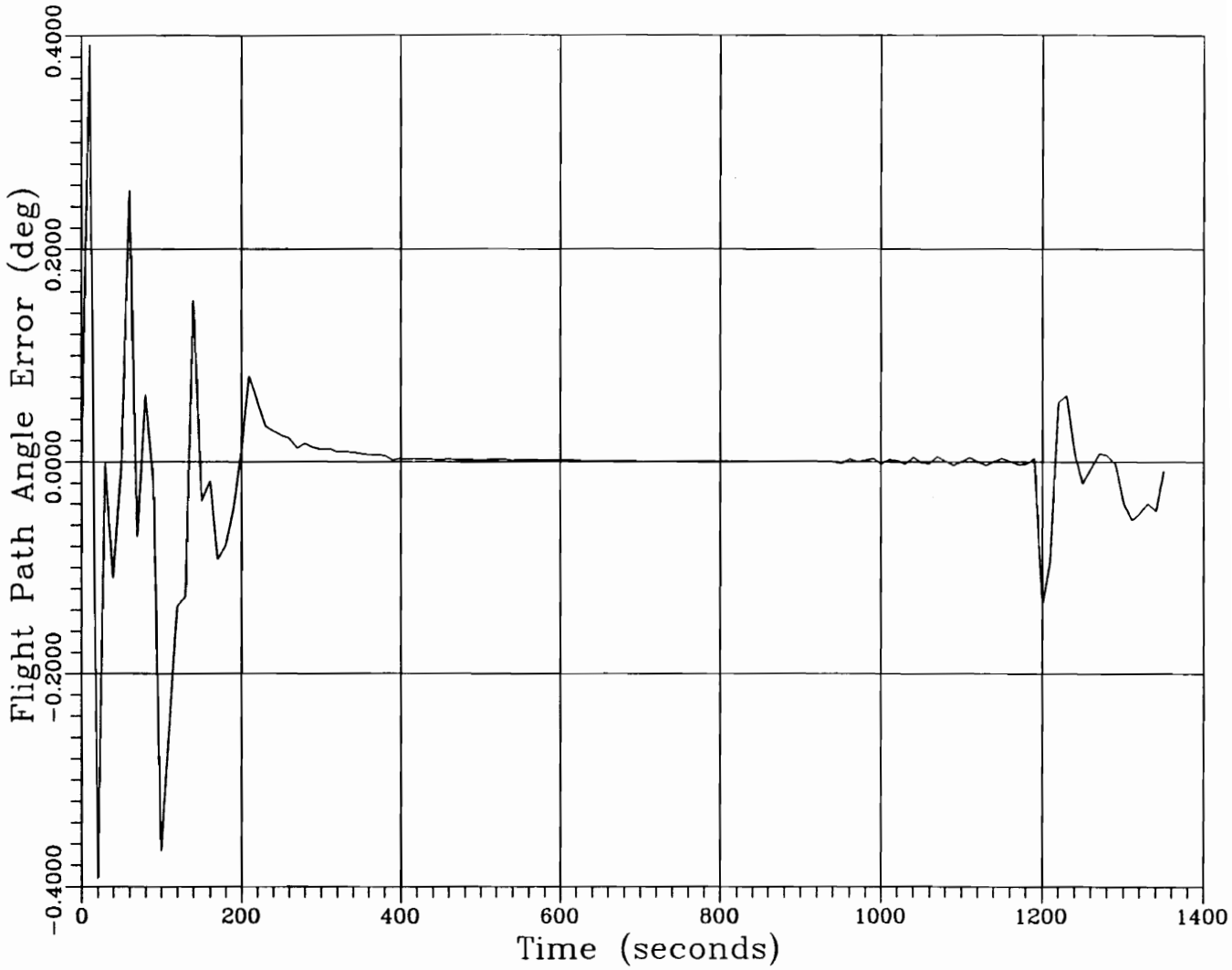

Figure 4.66 Closed Loop Flight Path Angle Error at Nominal Time Points Still Atmosphere

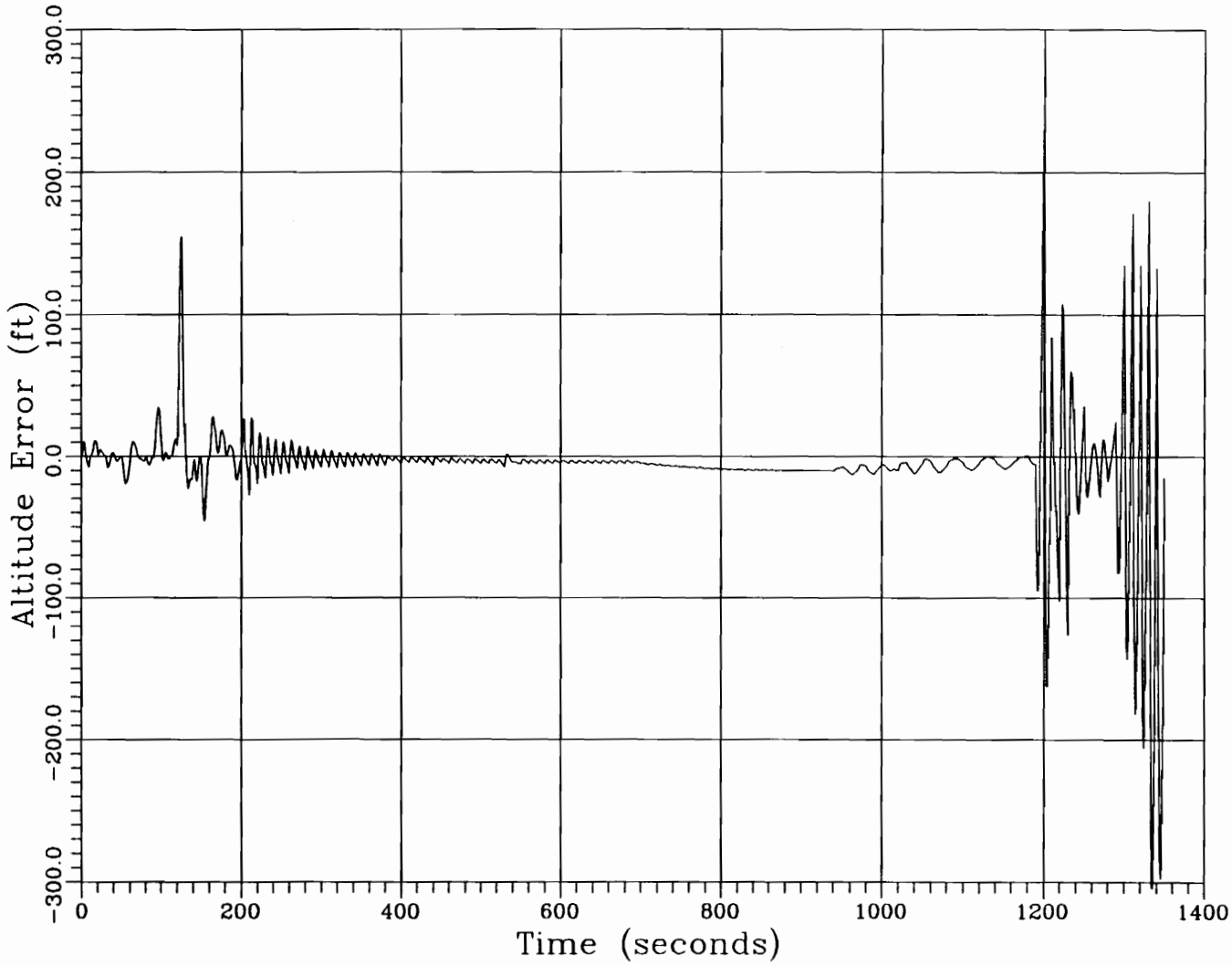

Figure 4.67 Closed Loop Altitude Error at 1 Second Interval Still Atmosphere

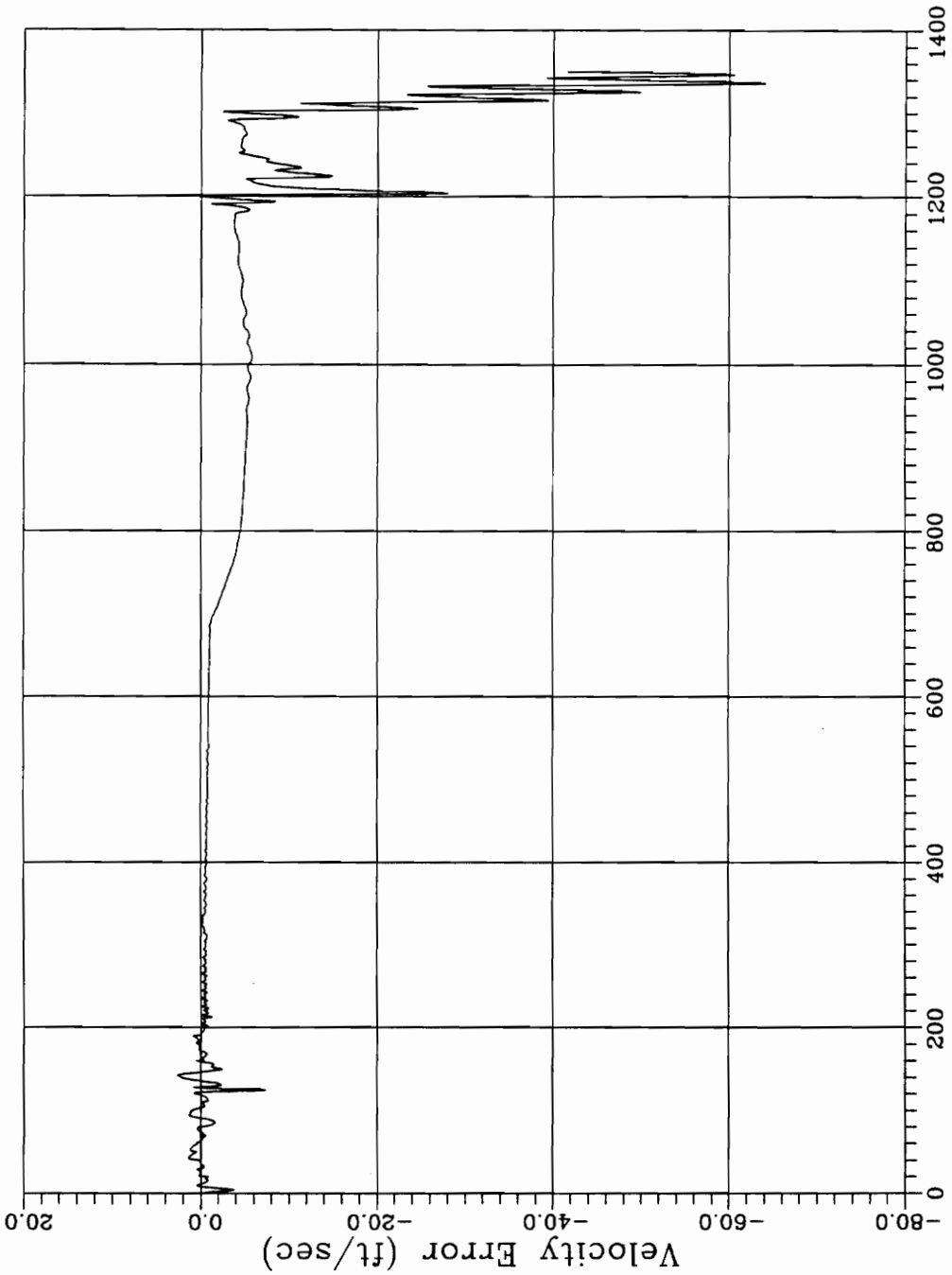

Figure 4.68 Closed Loop Velocity Error at 1 Second Interval Still Atmosphere

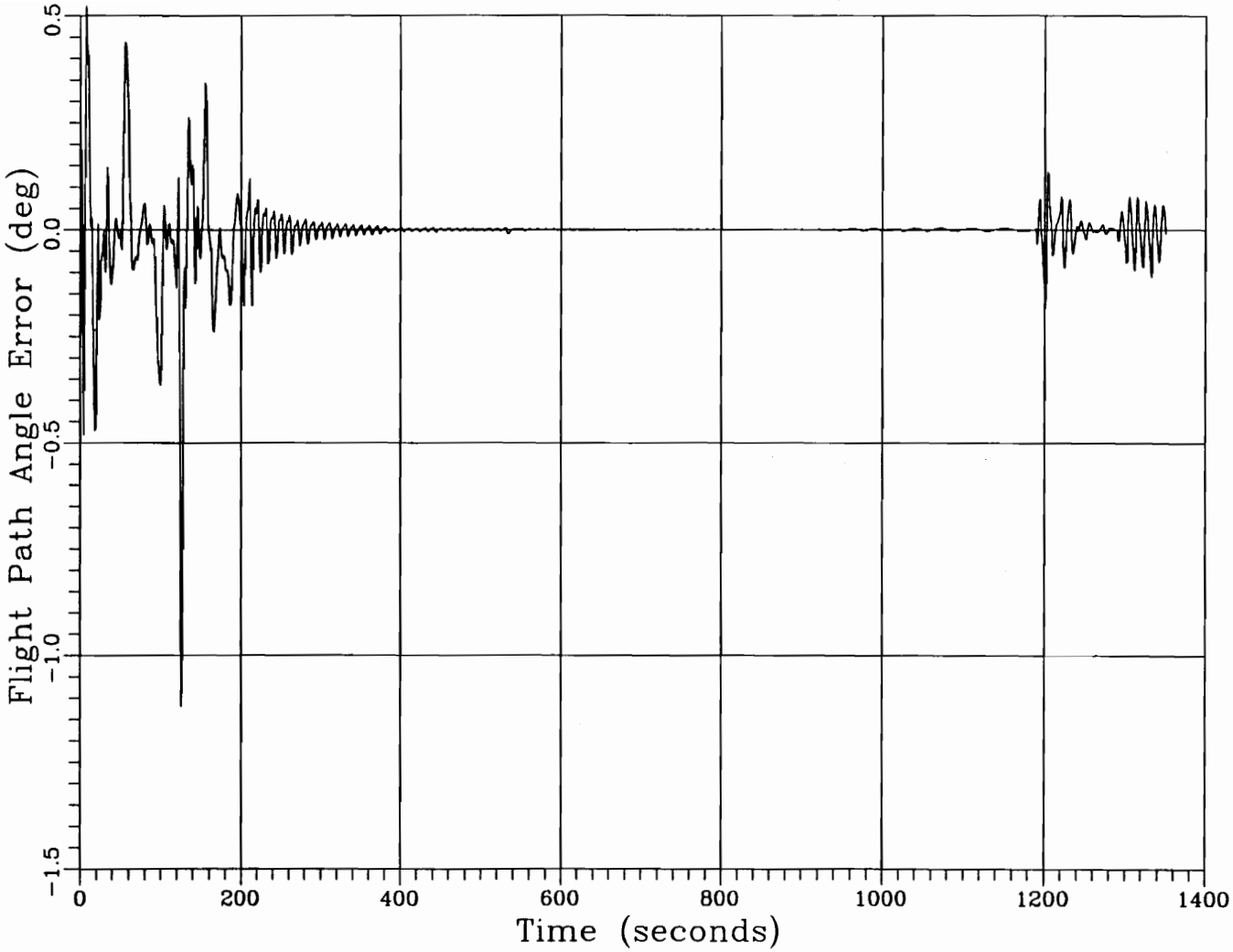

Figure 4.69 Closed Loop Flight Path Angle Error at 1 Second Interval Still Atmosphere

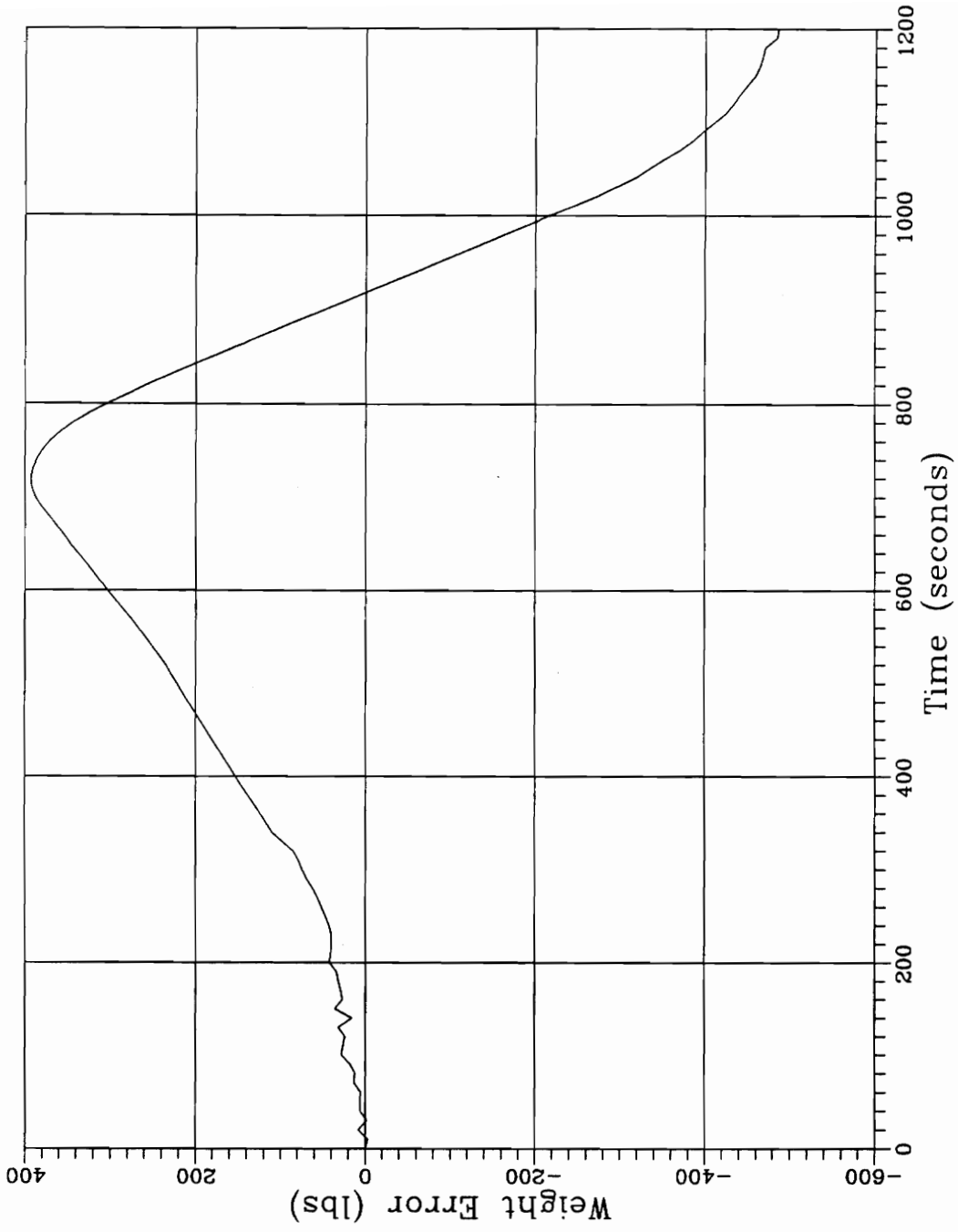

Figure 4.70 Closed Loop Weight Error

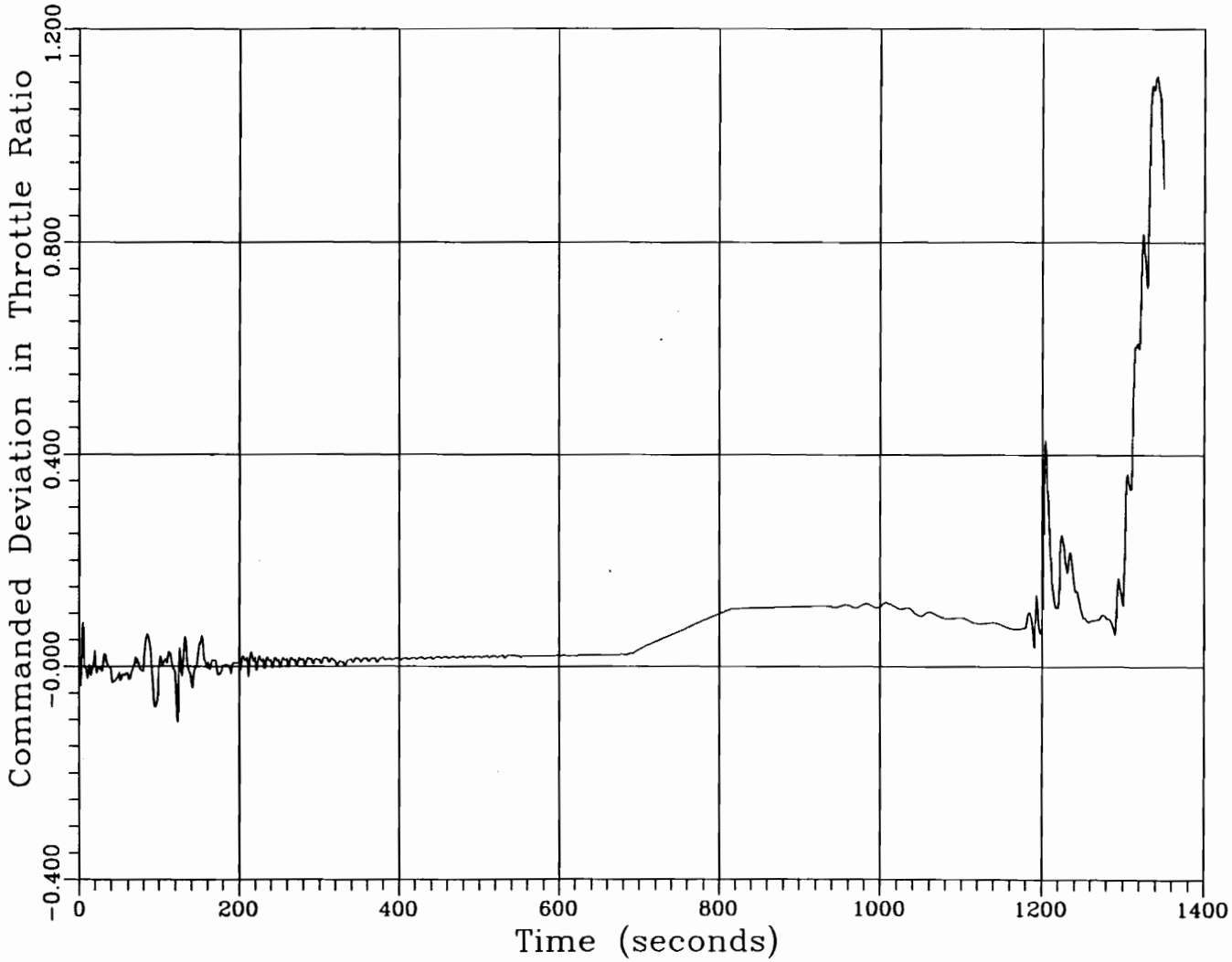

Figure 4.71 Closed Loop Commanded Deviation In Throttle Ratio Still Atmosphere

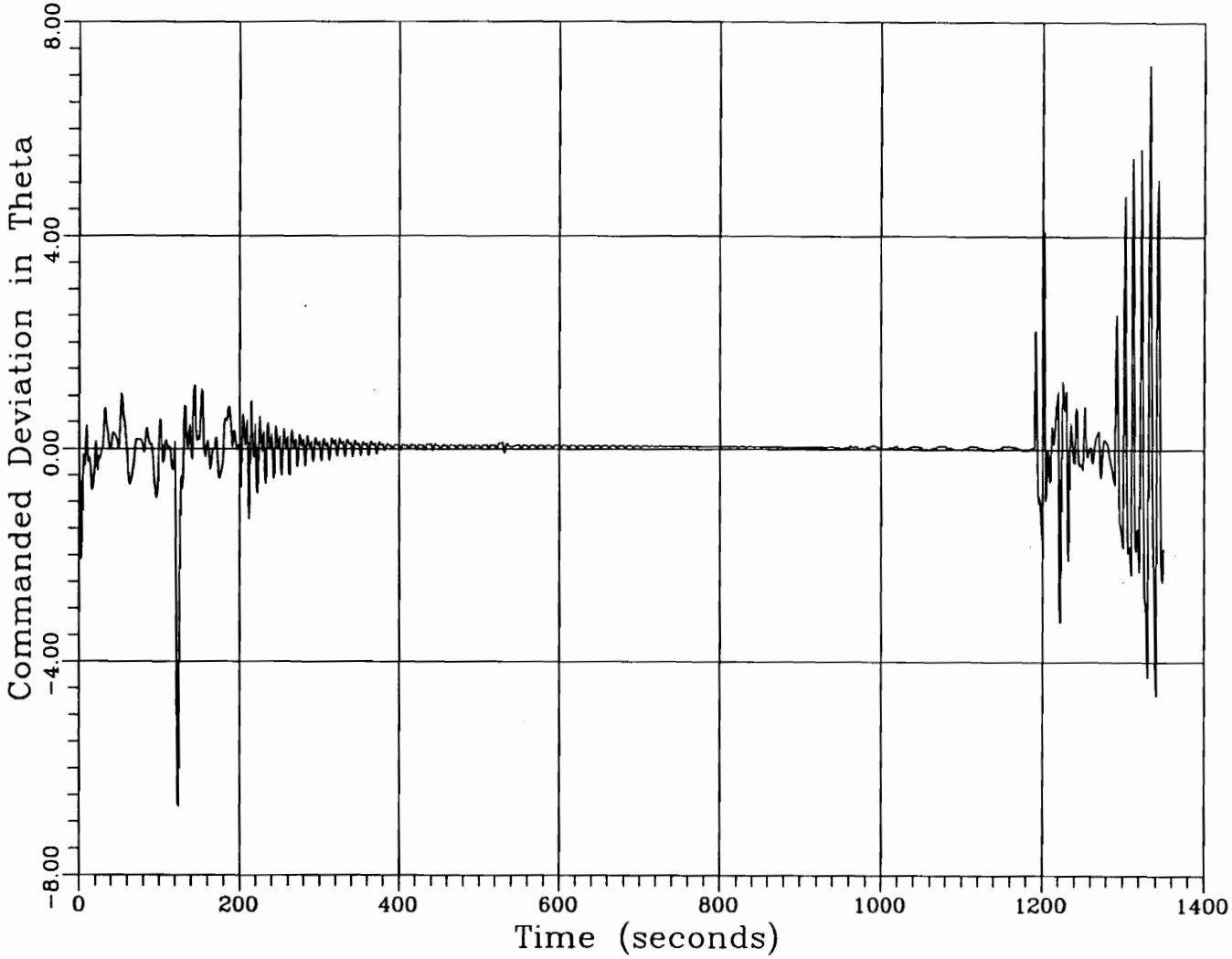

Figure 4.72 Closed Loop Commanded Deviation In Pitch Attitude Still Atmosphere

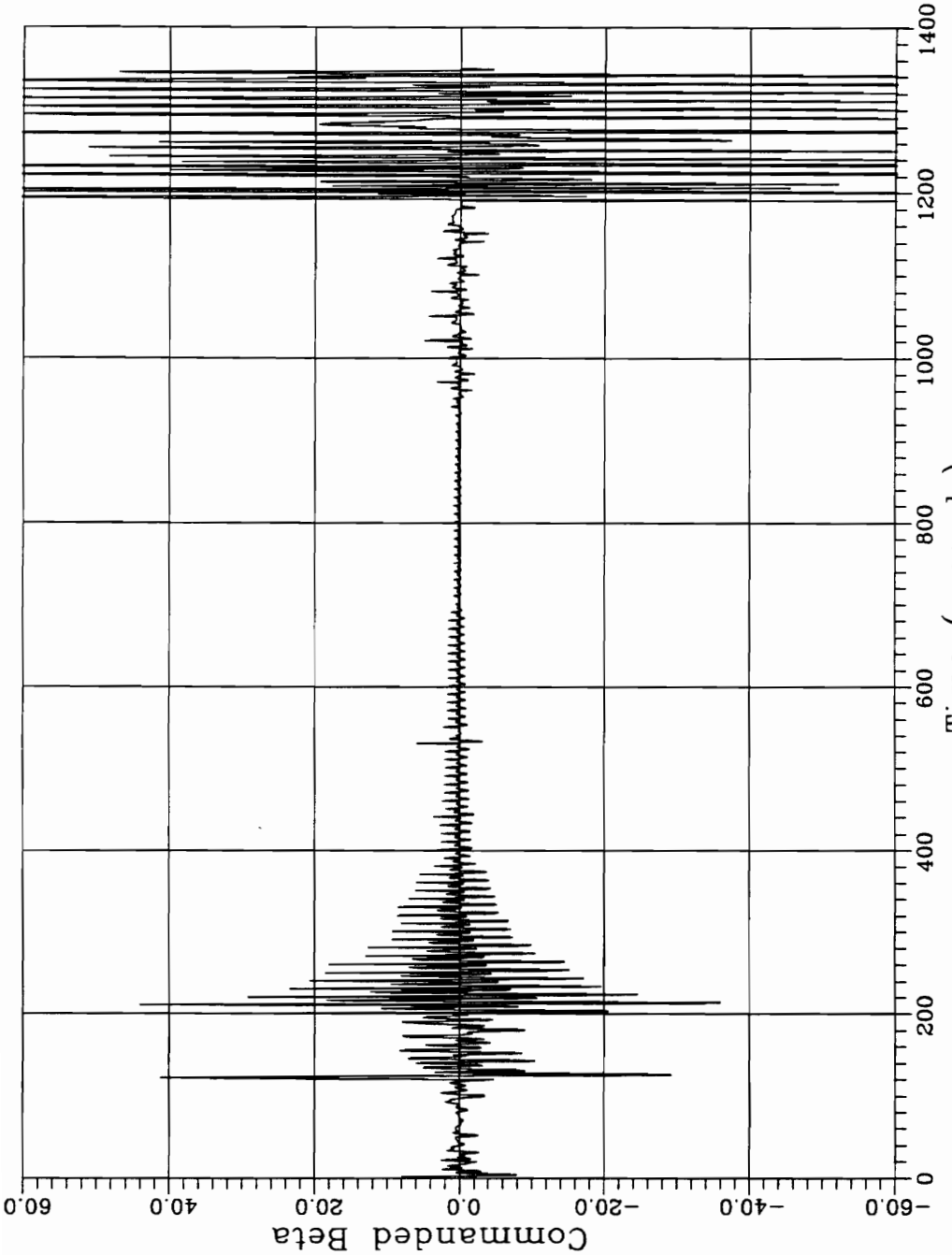

Figure 4.73 Closed Loop Commanded Thrust Vector Angle
Still Atmosphere

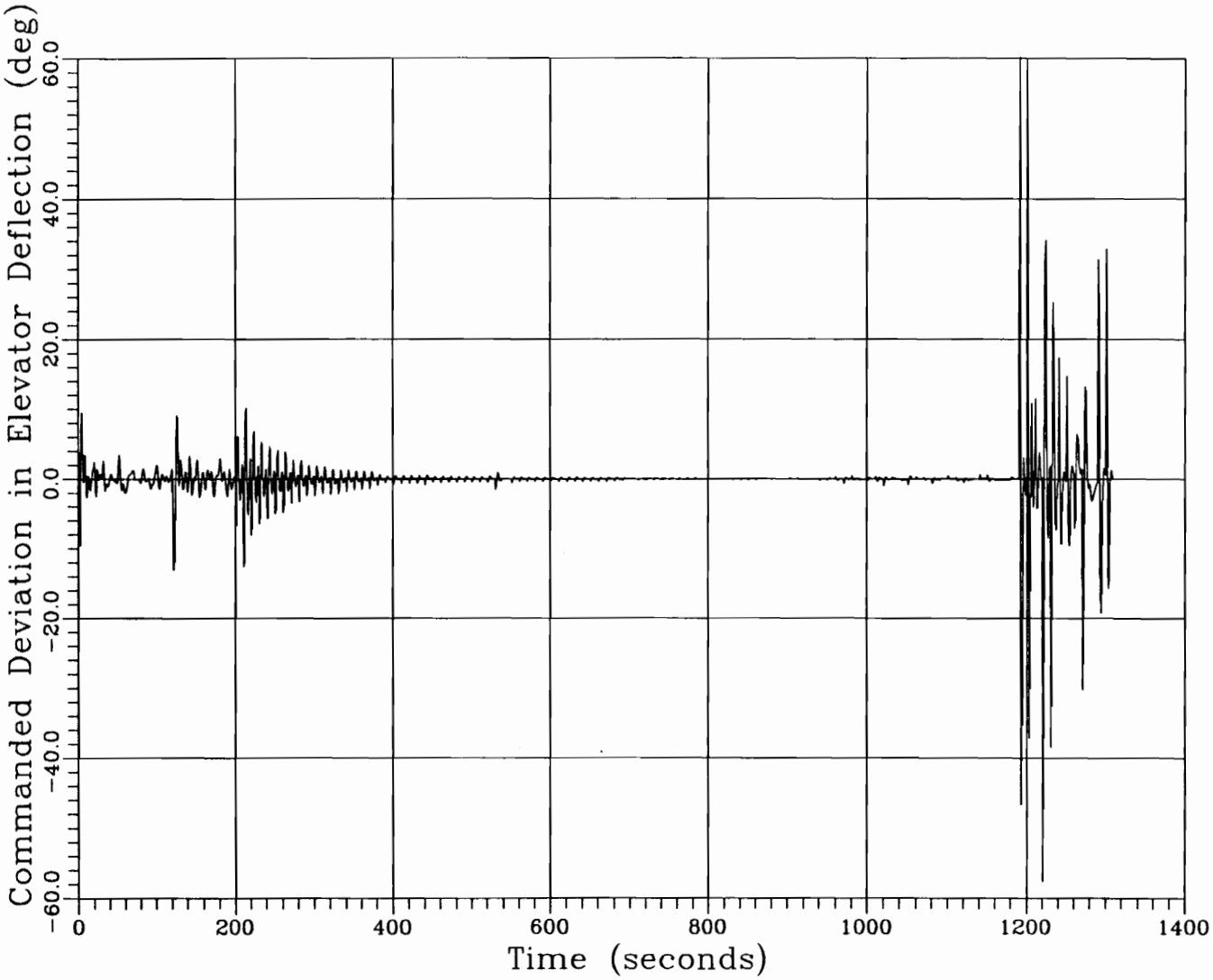

Figure 4.74 Closed Loop Commanded Elevator Deflection Deviation Still Atmosphere

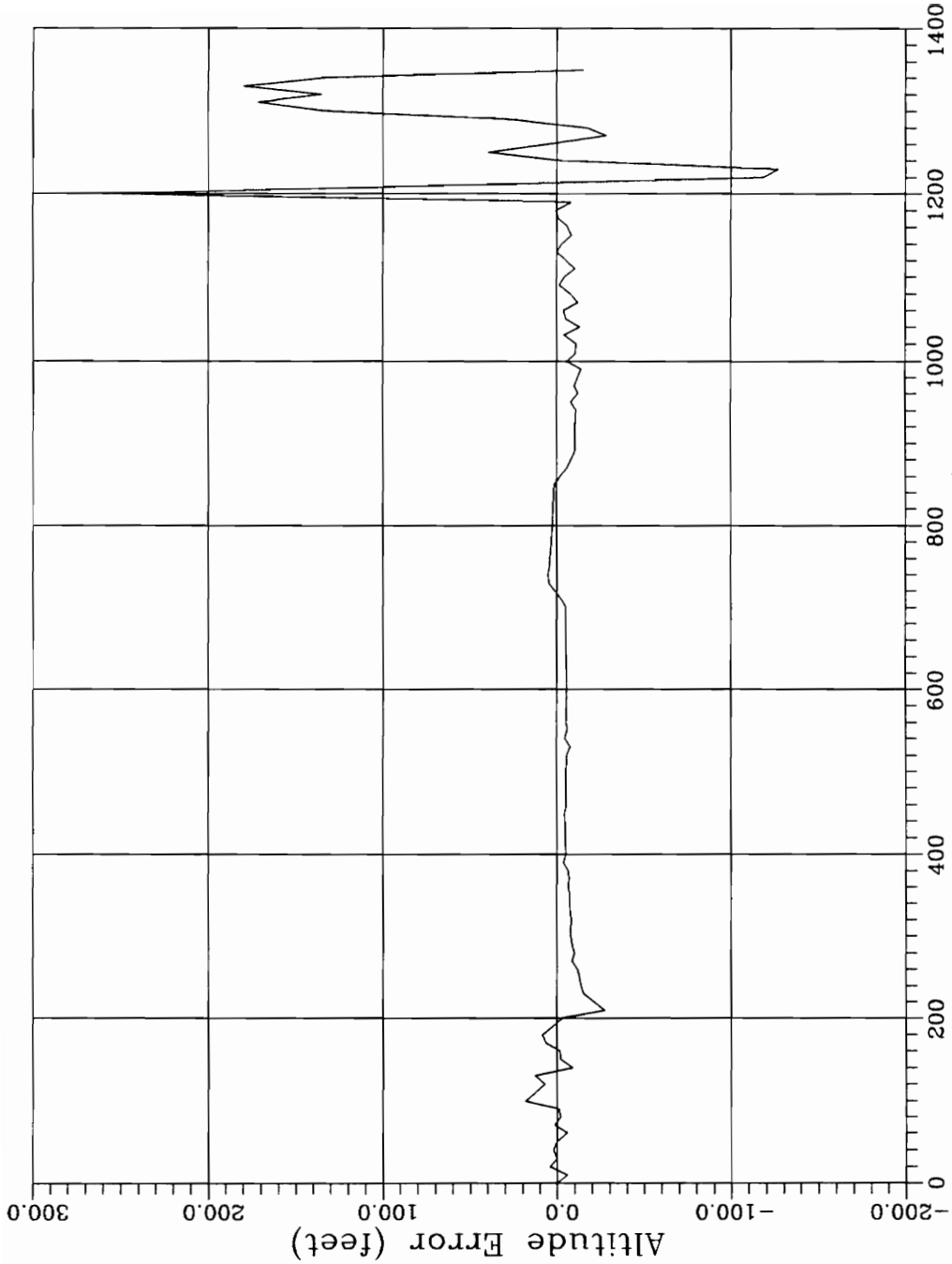

Figure 4.75 Closed Loop Altitude Error at Nominal Time Points Gust at 700 Seconds

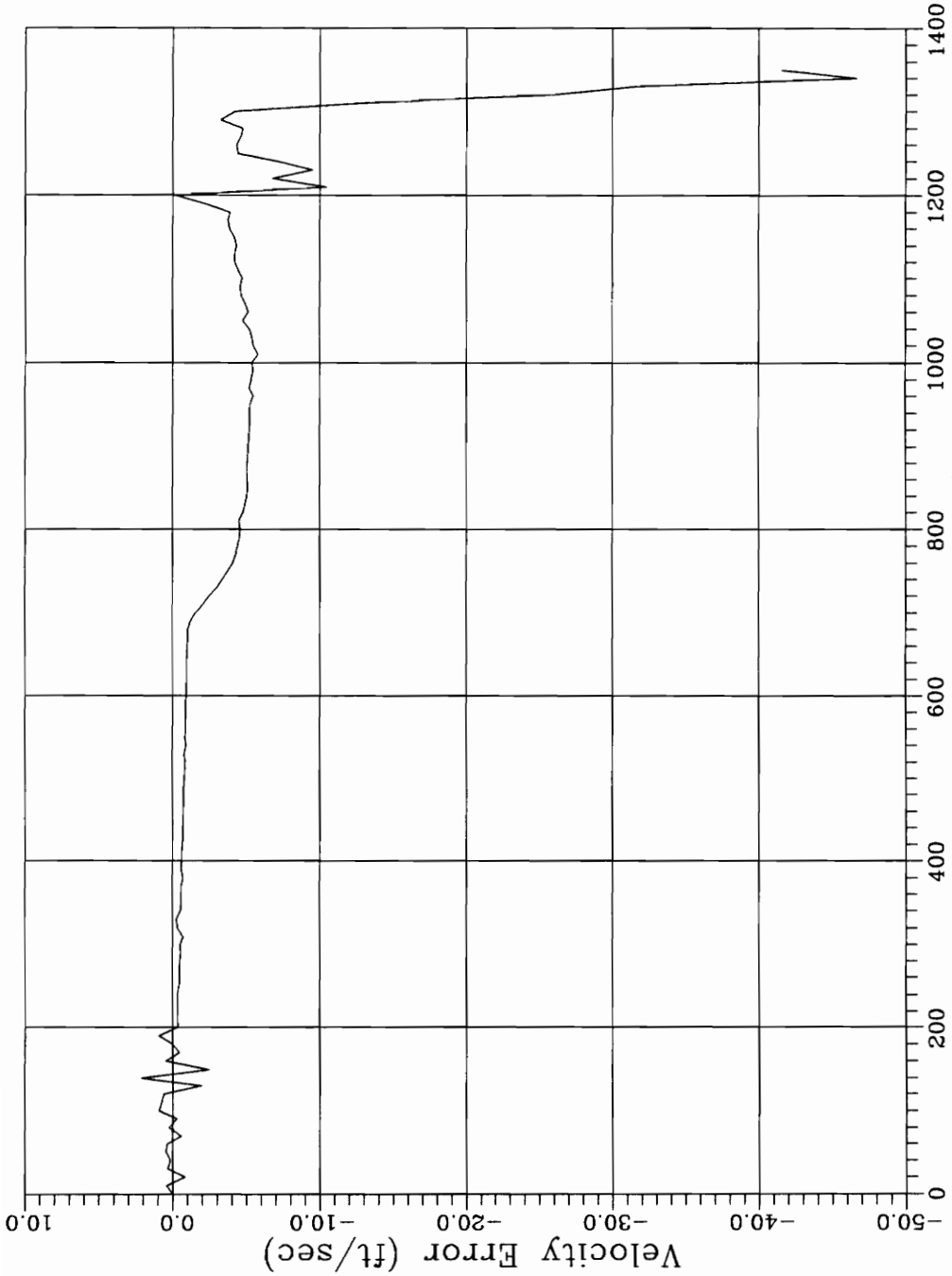

Figure 4.76 Closed Loop Velocity Error at Nominal Time Points Gust at 700 Seconds

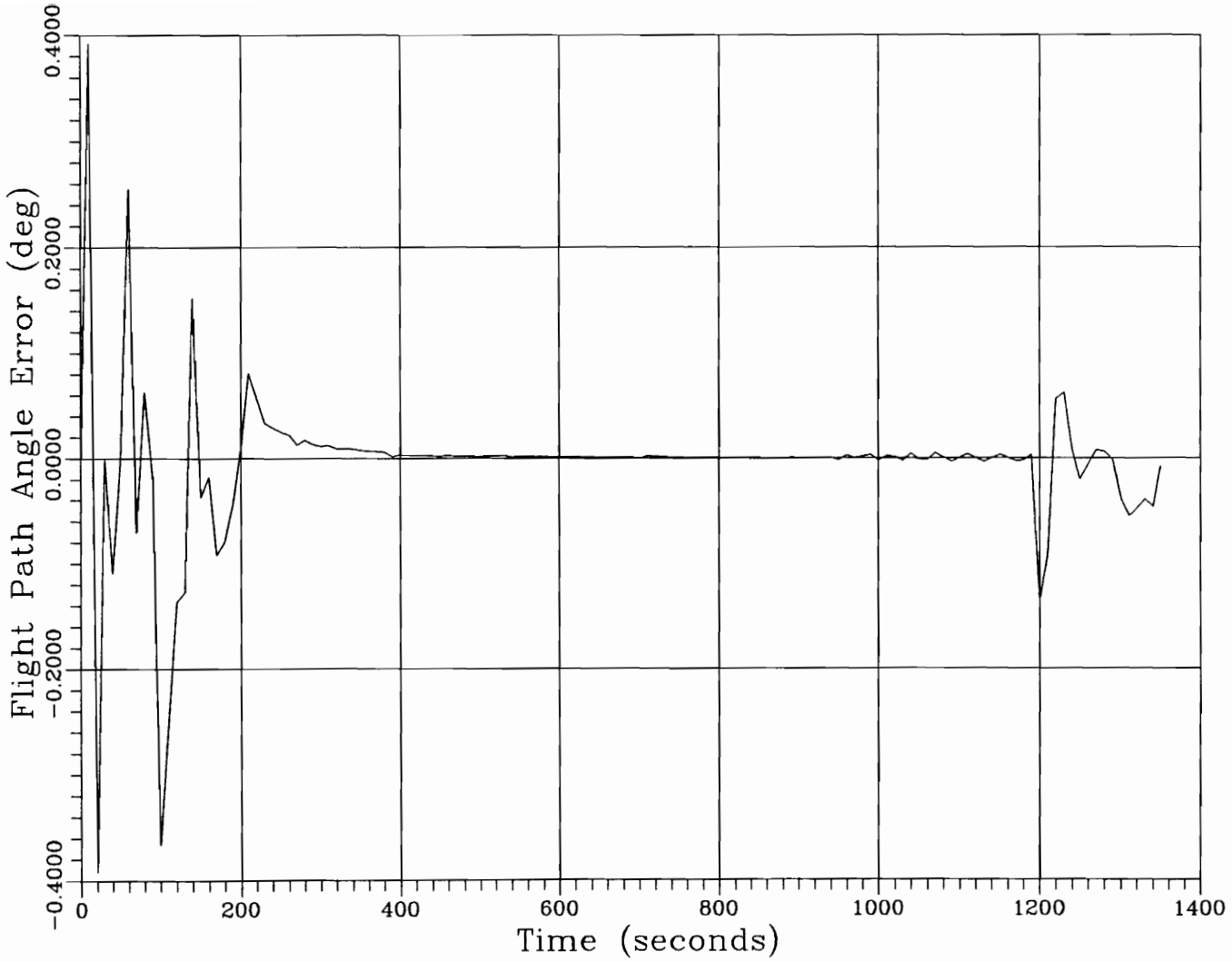

Figure 4.77 Closed Loop Flight Path Angle Error at Nominal Time Points Gust at 700 Seconds

VITA

Philip Merton Bushong was born 14 December 1957 in Elkart, Indiana. He received his Bachelor of Science in Aerospace Engineering from Texas A&M University in December of 1986, after gaining valuable engineering experience as a Coöp Engineer at General Dynamics / Fort Worth Division during his undergraduate studies. He returned there as an engineer after graduation. In August 1987 he returned to Texas A&M to begin graduate studies and received a Master of Science in Aerospace Engineering from that University in August of 1988. Work was begun in January of 1989 on coursework and research toward a Doctor of Philosophy in Aerospace Engineering, for which this dissertation is a requirement.