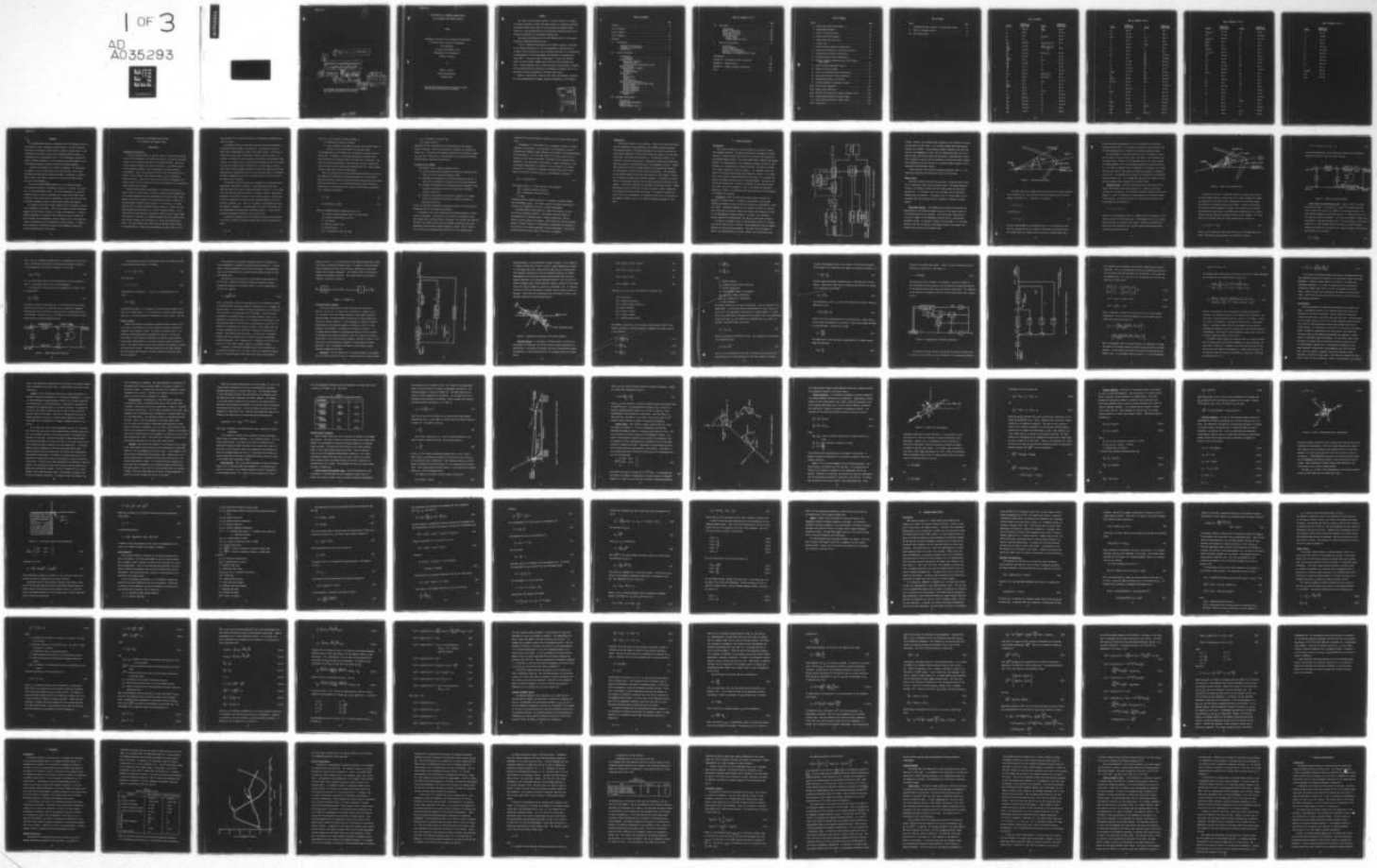
AD-A035 293 AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OHIO SCH--ETC F/G 19/5
APPLICATION OF AN EXTENDED KALMAN FILTER TO AN ADVANCED FIRE CO--ETC(U)
DEC 76 R N LUTTER

UNCLASSIFIED GE/EE/76-31 NL

1 OF 3
AD
A035293



GE/EE/76-31

(1)

(9) Master's thesis,

(6)

APPLICATION OF AN EXTENDED KALMAN FILTER TO AN ADVANCED FIRE CONTROL SYSTEM.

THESIS

(14) GE/EE/76-31

(10) Robert N. Lutter
Capt USAF

(11) Dec 76

(12) 247 p.

012 225

GE/EE/76-31

APPLICATION OF AN EXTENDED KALMAN FILTER TO AN ADVANCED FIRE CONTROL SYSTEM

THESIS

Presented to the Faculty of the School of Engineering

of the Air Force Institute of Technology

Air University

in Partial Fulfillment of the

Requirements for the Degree of

Master of Science

by

Robert N. Lutter

Graduate Engineering

December 1976

Preface

The intent of this study is twofold: the first intent is to satisfy the thesis requirement at AFIT; the second intent is to design and evaluate an extended Kalman Filter which, by incorporating the dominant dynamic effects on a non-thrusting missile, can provide an improved prediction and estimation capability for the missile tracking task.

The project was sponsored by the Laser Analysis Group of the Avionics Laboratory, Wright-Patterson AFB, Ohio.

I wish to express my appreciation to my sponsor, Captain J. Gary Reid of the Avionics Laboratory, for his responsiveness to inquiries and assistance in filter formulation, and to Professor Max Mintz, of the University of Pennsylvania, for his encouragement and assistance in developing the truth model. I would also like to thank Major P. Sproul and Professor Calico for their helpful comments and corrections while reviewing this text. I would especially like to thank my advisor, Captain Peter S. Maybeck, for the latitude which he allowed me in attacking the problem, and for the time which he spent assisting me in several problem areas.

Finally, I would like to thank my wife, Marty, and daughter, Christina, for their understanding and support during the completion of this thesis.

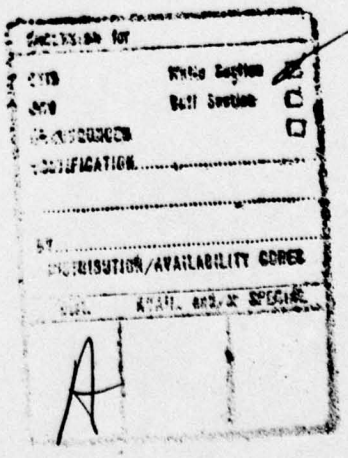


Table of Contents

Table of Contents (Cont'd)

List of Figures

List of Tables

List of Symbols

Symbol	Defined or First Used	Symbol	Defined or First Used
A	EQ 120	C_{TLOS}^{TB}	EQ 53
a_c	Fig 6	d	EQ 21-a
a_D	EQ 3	$e_{mean}(t_j)$	EQ 143
a_ℓ	EQ 2	E	EQ 89
$a_{m\alpha}^{I}$	EQ 98-b	$f[\underline{x}_n(t),\underline{u}(t),t]$	EQ 4
a_{mx}^{I}	EQ 46-a	$\dfrac{\partial \underline{f}[\underline{x}(t),\underline{u}(t),t]}{\partial \underline{x}}$	EQ 94
a_{mx}^{TLOS}	EQ 47-a		
a_{my}^{I}	EQ 46-b	$\underline{F}[t;\hat{\underline{x}}^{+}(t_i)]$	EQ 93
a_{Tx}^{I}	EQ 50-a	g	EQ 59
a_{mx}^{TLOS}	EQ 51	$G(t)$	EQ 88
a_{Ty}^{I}	EQ 50-b	h_1	EQ 33
a_{Ty}^{TB}	EQ 54	h_2	EQ 34
a_{T1}	Fig 11	h_3	Fig 10
a_{T2}	EQ 39	$\underline{h}[\underline{x}(t_i),t_i]$	EQ 90
C_D	EQ 22	$\underline{H}[t;x(t_i^-)]$	EQ 95
C_{DI}	EQ 3	I_{yy}	EQ 21-a
C_{DO}	EQ 3	K	EQ 23
C_I^{TLOS}	EQ 42	k	EQ 35
C_L	EQ 23	$K(t_i)$	EQ 95
$C_{L\alpha}$	EQ 24	K_1	EQ 48-a
$C_{M\alpha}$	EQ 21-a	K_2	EQ 48-a
$C_{M\delta}$	EQ 21-b	K_3	EQ 48-b
C_{Mq}	EQ 21-a	K_4	EQ 48-b
$C_{N\alpha}$	EQ 21-e	k_r	EQ 8
$C_{N\delta}$	EQ 21-c	L	EQ 26

List of Symbols (Cont'd)

Symbol	Defined or First Used	Symbol	Defined or First Used
L_α	EQ 18	S	EQ 3
L_δ	EQ 18	t	EQ 4
LOA	EQ 52-b	TS( )	EQ 85
m	EQ 3	u(t)	EQ 4
M	EQ 28	$\underline{v}(t_i)$	EQ 90
M_α	EQ 17	V_c	EQ 2
M_δ	EQ 17	V_m	EQ 3
M_q	EQ 17	$V^I_{m\alpha}$	EQ 98-a
n	EQ 1	$\dot{V}^I_{m\alpha}$	EQ 98-b
n'	EQ 16-a	V^I_{mx}	EQ 43-a
n_r	EQ 142	V^I_{my}	EQ 43-b
P_I	EQ 147-a	V_T	EQ 37
P_{TLOS}	EQ 147-a	$V^I_{T\alpha}$	EQ 98-a
$\underline{P}_o$	Pg 52	V^I_{Tx}	EQ 49-a
$\underline{P}(t_i^-)$	EQ 95	V^I_{Ty}	EQ 49-b
$\underline{P}(t/t_i)$	EQ 93	w(t)	EQ 88
q	EQ 17	w_3	EQ 100-c
$\bar{q}$	EQ 21-a	w_6	EQ 138
$\underline{Q}(t)$	EQ 89	w_7	EQ 139
R	EQ 55	w_x	EQ 113
$\dot{R}$	EQ 56	w_y	EQ 118
$\underline{R}(t_i)$	EQ 91	w_α	EQ 98-c
R_{obs}	EQ 102	$\hat{\underline{x}}(t_i^+)$	EQ 96
R_{OM}	Fig 17	$\hat{\underline{x}}_o$	Pg 52
R_{OT}	Fig 17	$\hat{\underline{x}}(t/t_i)$	EQ 92

List of Symbols (Cont'd)

Symbol	Defined or First Used	Symbol	Defined or First Used
$x_{comp_i}(t_j)$	EQ 142	$\eta_{\Lambda\alpha}$	EQ 99
x_m^I	EQ 43-a	η_R	EQ 102-a
$x_{mean}(t_j)$	EQ 142	η_R	EQ 102-b
$\underline{x}_n(t)$	EQ 4	θ	EQ 5
$\dot{\underline{x}}_n(t)$	EQ 4	$\dot{\theta}$	EQ 1
$x_{SD}(t_j)$	EQ 144	θ'	EQ 12
$x_{true}(t_j)$	EQ 143	$\dot{\theta}'$	EQ 11
x_T^I	EQ 48-a	θ_a	EQ 37
x_{t_m}	EQ 41	θ_o	EQ 52-c
y_d	EQ 40	$\dot{\theta}'_f$	EQ 73
y_m^I	EQ 43-b	$\ddot{\theta}'_f$	EQ 73
y_T^I	EQ 48-b	θ_ℓ	EQ 16-a
$\underline{z}(t_i)$	EQ 90	θ_m	EQ 5
α	EQ 17	θ_η	EQ 5
γ_m	EQ 43-a	θ_r	EQ 7
$\dot{\gamma}_m$	EQ 1	θ_{rb}	EQ 8
γ'_T	EQ 52-a	θ_t	EQ 40
δ	EQ 17	θ_T	EQ 42
δ_{ij}	EQ 91	λ	EQ 19
ε	EQ 5	$\dot{\Lambda}_\alpha$	EQ 98-a
ε'	EQ 7	$\Lambda_{\alpha obs}$	EQ 99
ζ	EQ 32	ρ	EQ 3
ζ_i	EQ 96	σ	EQ 141
η_1	EQ 82	$\sigma_{g/S}$	EQ 36
η_2	EQ 82	σ_R	EQ 36

List of Symbols (Cont'd)

Symbol	Defined or First Used
σ_T	EQ 36
τ_α	EQ 98-c
τ_1	EQ 10
τ_2	Fig 6
τ_3	EQ 100-c
τ_x	EQ 103-h
τ_y	EQ 103-i
ϕ_0	EQ 52-d
ϕ_1	EQ 52-e
ϕ_m	EQ 52-f
ω	EQ 32
$\omega_{I(TLOS)}$	Fig 17
ω_{LSy}	EQ 101
ω_{LSz}	EQ 101

Abstract

An extended Kalman Filter is developed to aid the tracking of an air-to-air missile from a maneuvering target aircraft. The filter exploits knowledge of the dominant dynamic effects acting on a missile that is non-thrusting and utilizing a proportional navigation guidance scheme, i.e. accelerations due to aerodynamic forces. It is designed to provide both dynamic tracking estimates in a local inertial frame and estimates of the proportional navigation constant and another pertinent parameter.

A feasibility analysis of the filter is conducted. Its performance is compared to a more conventional filter that utilizes a first order Gauss-Markov random process acceleration model. In addition, an evaluation is made of the filter's capability to recover from large initial errors in state estimates.

The study establishes the feasibility of the modelling approach. The estimates provided by the designed filter are, in general, less sensitive to system measurement noises. The filter performance is trajectory dependent, however, and the requirement for a higher order missile model within the filter system model is established (a zero-order model was used to develop as simple a filter as would provide adequate performance).

The results of the study strongly suggest that the navigation constant can be estimated by the filter. The recovery analysis provides additional insights into the filter's ability to estimate this parameter. It gives a general indication of the effects that varying the initial variance and noise strength (on the navigation constant channel) have on the tuning and recovery characteristics of the navigation constant estimate. A graphic filter analysis is included that portrays the estimation accuracy and recovery characteristics of the filter.

APPLICATION OF AN EXTENDED KALMAN FILTER TO AN ADVANCED FIRE CONTROL SYSTEM

I. Introduction

Background and Motivation

An ever increasing amount of research effort is being directed toward the air-to-air engagements of missiles and aircraft. An area of particular interest is the defensive capability of the target aircraft. While most current defensive measures are passive in nature, it is possible that an active defensive capability could be developed to increase aircraft survivability. One of the decisive factors in such a system would be the ability of the aircraft's fire control system to track a maneuvering target accurately. This report concerns itself with the development of a Kalman Filter to aid the tracking of an air-to-air missile from a maneuvering target aircraft (the tracker).

Several works have been accomplished in the general tracking area (Refs 3 and 10). In most cases, in order to simplify filter implementation, the tracking algorithms have been implemented in the line-of-sight frame. However, this does not appear to be an appropriate choice of frames for a multiple target, multiple sensor scenario. In such cases, it is highly probable that a transfer alignment would be required to combine the input data available and then route it to a remote tracking platform. Such a scheme would be computationally cumbersome if several line-of-sight frames were involved. Therefore, it is highly desirable that the tracking filter provide state estimates and predictions in a local inertial/stationary frame. Two other concerns that result from the multiple target scenario are identification of target type and knowledge of target status. Most

previous works in the tracking area have not specifically addressed these latter two areas.

The primary concern in these works has been the accurate modelling of missile/aircraft dynamics and kinematics. One approach has been to model target acceleration as a first order Markov Process. This model is widely used and has given satisfactory results in several studies. It provides for random, but time correlated acceleration of the vehicle being tracked. By a proper choice of process noise strength and time correlation, the first order Markov model is descriptive of a wide range of targets (Ref 10:326). While the model is generic in nature and appears particularly attractive for a maneuvering aircraft (i.e., man in the loop), it disregards available information in the case of several state of the art air-to-air and surface-to-air missiles.

The kinematics of most missiles are determined by a guidance package implementing proportional navigation. Easy implementation and high reliability make this form of guidance extremely attractive. In the linear case, with both the target and missile of constant velocity, proportional navigation is an optimal solution to the intercept problem. Several works have been accomplished to develop optimal guidance schemes for the more practical nonlinear case (Ref 14). In general, these schemes have been too complex to implement. In several cases, the form of the optimal solution approximates the proportional navigation solution - a time varying proportional navigation constant is one such approach.

Proportional navigation commands a missile turning rate proportional to the rate of change of the line-of-sight angle between the missile and target:

$$\dot{\gamma}_m = n\dot{\theta} \tag{1}$$

where $\dot{\gamma}_m$ = rate of change of missile heading, γ_m

n = proportional navigation constant

$\dot{\theta}$ = rate of change of the missile-to-target line-of-sight angle, θ, relative to an initial missile line-of-sight frame.

The implementation of the guidance law is discussed in greater detail in Chapter Two. In addition to utilizing proportional navigation, many current missiles are "boosted to mach" and then coast to intercept. For many of the current generation of missiles, then, it can be assumed that they are using a form of proportional navigation and, for a large segment of their flight time, they are non-thrusting.

These two characteristics provide additional information that may not only improve the estimation and prediction capability of the tracking filter, but provide a means of predicting missile identification and status. The two dominant affects on a non-thrusting missile using proportional navigation are the deceleration due to drag and the lateral acceleration due to commanded acceleration.

$$a_\ell = nV_c\dot{\theta} \tag{2}$$

$$a_D = \frac{1}{2}\rho Sm(C_{DO} + C_{DI})V_m^2 \tag{3}$$

where a_ℓ = missile lateral acceleration due to lift

V_c = relative closing velocity along the line-of-sight

a_D = missile acceleration due to drag

ρ = air density

S = reference surface area

m = mass of missile

C_{DO} = coefficient of zero-lift drag

C_{DI} = coefficient of induced drag

V_m = missile velocity.

Equations (2) and (3) model the total acceleration for the assumed missile conditions. The fact that gravity is neglected is discussed in Chapter Two. In addition, the equations embody three coefficients, namely C_{DI}, C_{DO}, and n, that could, if they themselves are accurately estimated, provide missile identification and status.

Statement of the Problem

The problem is then to design a Kalman Filter to

(1) provide desired state estimation and trajectory prediction capability in a local inertial/stationary frame,

(2) determine the feasibility of utilizing Equations (2) and (3) to model the acceleration of a generic tactical missile adequately,

(3) determine the feasibility of obtaining accurate estimates of the constants in Equations (2) and (3) or some other parameter related to them, and

(4) evaluate the estimation and prediction capability of a Kalman Filter that includes the additional information provided in Equations (2) and (3).

As part of the filter formulation, appropriate models of the three unknown parameters need to be proposed and validated.

In addition to the filter containing the unknown parameters, a second filter is formulated. The purpose of this filter is to provide a benchmark against which the first filter is compared. The second filter utilizes the first order Markov Process model for missile acceleration and comes essentially from the filter design presented by Landau (Ref 6). Both of the two filters are Extended Kalman Filters due to the nonlinearities

resulting from inertial models of several states and line-of-sight measurements.

Development of a "truth model" for the engagement scenario is another component of the problem. The truth model is a development of missile parameters and dynamics, target parameters and dynamics, and system kinematics and uncertainties. It is the "best model" of the real world in that it includes as many effects as possible, regardless of the resulting system complexity. The truth model is exercised in a computer simulation to provide nominal trajectories for analysis of the two filters. The nominal trajectory is a deterministic (reference) trajectory that starts from a known set of initial conditions, $\underline{x}_n(t_o)$, and propagates according to

$$\dot{\underline{x}}_n(t) = \underline{f}[\underline{x}_n(t), \underline{u}(t), t] \tag{4}$$

where $\underline{f}[\underline{x}_n(t), \underline{u}(t), t]$ = a known function of the arguments

$\underline{x}_n(t)$ = nominal reference trajectory

$\underline{u}(t)$ = deterministic forcing function

t = time.

Associated with the nominal trajectory is a sequence of nominal sampled data measurements, $\underline{z}_n(t_i)$. The filter utilizes the nominal measurements, corrupted by an additive white noise, as input measurements.

The final aspect of the problem is a Monte Carlo analysis of both filters over a nominal trajectory. The nominal trajectory chosen is representative of a moderate amount of missile maneuvering. The two filters are tuned against this nominal. Both filters will also be exercised against a "high-g" trajectory of the missile to demonstrate the sensitivity of the filter to the various extremes of missile maneuvering. It will also establish whether further tuning of the filter is required.

Organization

This thesis consists of five chapters. Chapter One outlines the motivation and background for investigating a more complex tracking algorithm. The algorithm exploits additional knowledge of the missile and models the three unknown parameters - proportional navigation constant, zero-lift drag coefficient, and induced drag coefficient - as additional states or as a function of the other system states. Chapter Two develops the system truth model and presents the assumptions made in the engagement scenario. Chapter Three introduces the basic Extended Kalman Filter Equations and presents the formulation of the two specific filters investigated. The development of the unknown parameter models are also included in this chapter. Chapter Four introduces the Monte Carlo analysis utilized. In addition, the tuning process and problems encountered in the tuning process are discussed in this chapter. Chapter Five discusses the graphical and numerical results. Having demonstrated concept feasibility (at least partially), several suggestions are presented for future studies of interest. The appendix contains portions of the filter development, the graphical results, and the description and computer listing of each of the computer programs developed for this study.

II. System Truth Model

Introduction

This chapter develops the realistic models of the missile, target, and engagement geometry. The models developed are simulated on a digital computer to provide nominal trajectories for the Monte Carlo analysis of the Extended Kalman Filters. The system models should represent the dominant dynamics and nonlinearities of the engagement scenario, and yet be representative of a wide spectrum of tactical missiles. The model is divided into several components, as shown in Figure 1. The remainder of the sections in this chapter will develop each component in detail. It is important to note that the primary objective of this report is to describe and analyze the dynamics and motion of an attacking missile. The missile subsystems and dynamics will, therefore, be modelled in much greater detail than other truth model components. Prior to discussing the individual components, however, the assumptions that have been made in the development of the model are presented.

Assumptions. Both the missile and target aircraft (tracker) are assumed to be point masses. In addition, measurements of the tracker dynamics are assumed to enter the filter algorithms as deterministic, noiseless inputs. Navigation systems that are currently available in aircraft have errors that are second or lower order effects when compared to errors in the measurements of missile dynamics. Therefore, this assumption greatly simplifies the system model, but does not affect model integrity.

All motion is assumed to occur in the horizontal plane (i.e. constant altitude) and gravity drop is ignored. These assumptions allow for simpler modelling of certain system parameters. The model could be extended to include the three-dimensional, altitude varying case if more detail were

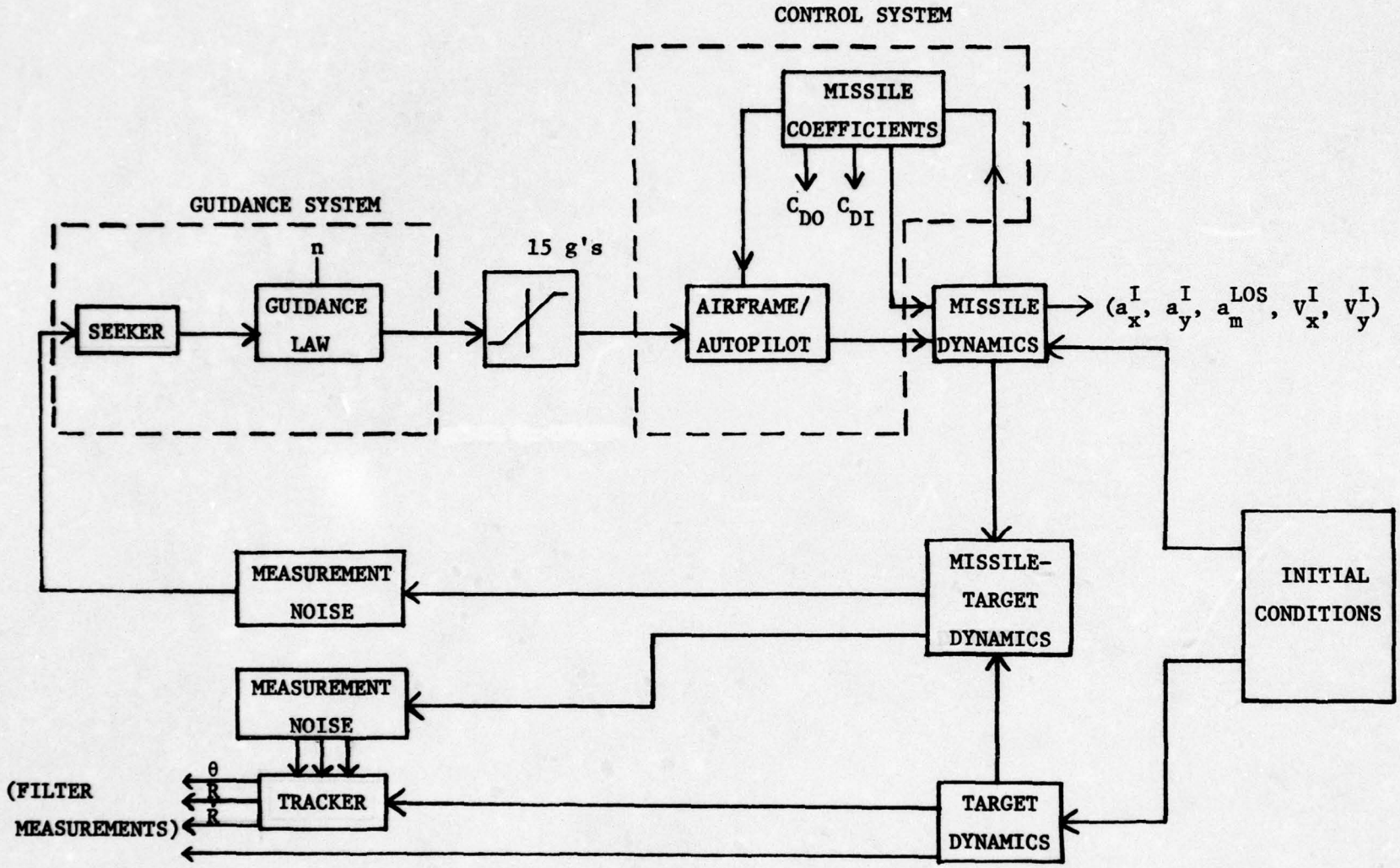


Figure 1. System Truth Model Block Diagram

desired. However, the increased model complexity is not required to achieve the objectives of this thesis. It is further assumed that meteorological effects can be ignored. This assumption appears reasonable since the time period of the engagement is short, both the missile and target are in the same air mass, and finally because the magnitudes of meteorological effects will in general be second order in comparison to missile and target dynamics. Other assumptions that are peculiar to a particular model component will be discussed in the applicable section.

Certain sections that follow are largely developed in (Ref 13). The appropriate sections are referenced at the end of the section.

Missile Seeker

The missile seeker serves the two functions of measuring target motion and of tracking the target with the antenna device. The system generally consists of a gimballed platform with the seeker antenna and rate gyros mounted on it. In the planar case, only one gimbal and its associated dynamics need be considered. The specific case of a radar seeker will be considered, but the principles could be extended to infrared or optical seekers.

Measurement Geometry. The fundamental measurement obtained from the homing sensor is the indicated angular position of the target relative to the antenna center line or boresight. The line-of-sight (LOS) angle or its rate of change, is the fundamental quantity used in a proportional navigation scheme. As depicted in Figure 2 the LOS angle, θ, is the angle between a line from the center of the seeker antenna to the target, and some arbitrary non-rotating reference frame.

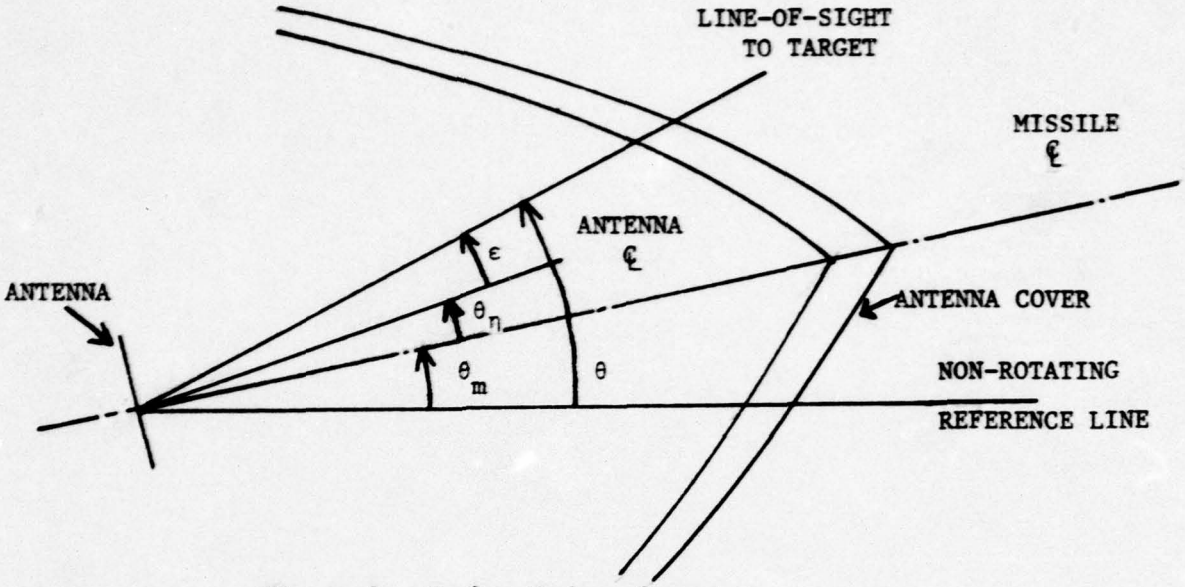


Figure 2. Seeker-Radome Geometry

The angle between the missile center line and the non-rotating reference line is defined as θ_m, and the angle between the missile center line and antenna centerline is θ_η. Therefore, θ is given by

$$\theta = \theta_m + \theta_\eta + \varepsilon \tag{5}$$

or solving for ε

$$\varepsilon = \theta - \theta_m - \theta_\eta \tag{6}$$

we have a relationship for the boresight error. It is important to note that the boresight error is a function of both missile attitude relative to inertial space and antenna position relative to missile center line.

To get the desired measurement of θ or $\dot{\theta}$, as opposed to an inertially stabilized seeker head, it is necessary to remove the missile motion from the LOS measurement data as measured by the missile guidance package.

One requirement of the seeker antenna is that it continues to point at the target in order to keep ε small. For small ε, the boresight error is nearly linear. However, if the boresight error is not small compared to antenna bandwidth, the error processing operation may become nonlinear. If ε is allowed to approach the half bandwidth of the antenna, the receiver detection unit will at some point lose lock and all guidance information will be lost. The nonlinearities involved are functions of the particular detection scheme used. It is assumed that the tracker is capable of keeping the boresight error processor in its linear region.

Aberration Error. The aberration angle error is the result of nonlinear distortions in the received energy as it passes the protective covering over the antenna. The distortion produces a false boresight error signal, ε', which is interpreted as target motion by the guidance system of the missile. As depicted in Figure 3, the indicated boresight error with aberration error present is

$$\varepsilon' = \theta + \theta_r - \theta_m - \theta_\eta \tag{7}$$

The size of the measurement error, θ_r, depends upon the orientation of the antenna relative to the antenna cover. The dependence of θ_r on θ_η couples body motion into the boresight error signal ("parasitic attitude loop"). This effect can drastically alter missile response characteristics and increase missile miss distance.

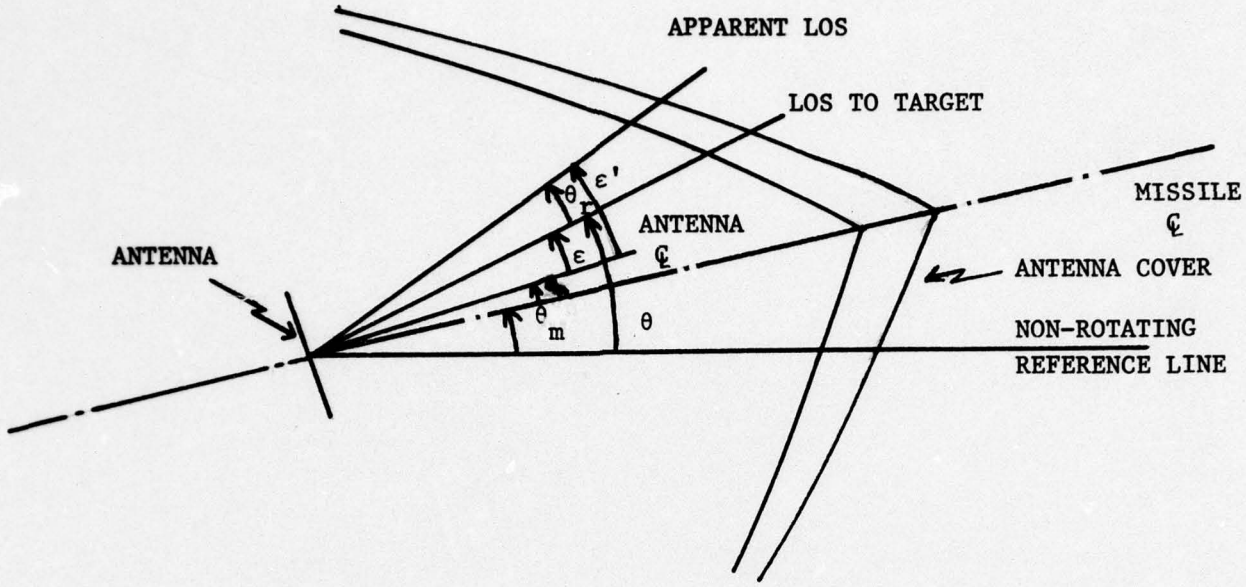


Figure 3. Effect of Aberration Error

Aberration error can be a nonlinear function of several error sources: the angle between missile centerline and LOS to target, physical and geometric properties of the antenna cover, polarization of the received signal, and erosion of the antenna surface during flight. Since the contributing error sources have wide ranges of variation, this research uses a constant aberration error slope model. The linear model is appropriate for a generic missile model and captures the dominant affect of body coupling. The linear model for the general aberration angle characteristics is given by

$$\theta_r \tilde{=} \theta_{rb} + (\theta - \theta_m)k_r \qquad (8)$$

where θ_{rb} is an aberration error angle bias and k_r is the aberration error slope. Substituting this expression into Equation (7) yields

$$\tilde{\varepsilon}' = (1 + k_r)(\theta - \theta_m) + \theta_{rb} - \theta_\eta \tag{9}$$

In the above expression, the boresight error bias has been considered negligible relative to the other errors present.

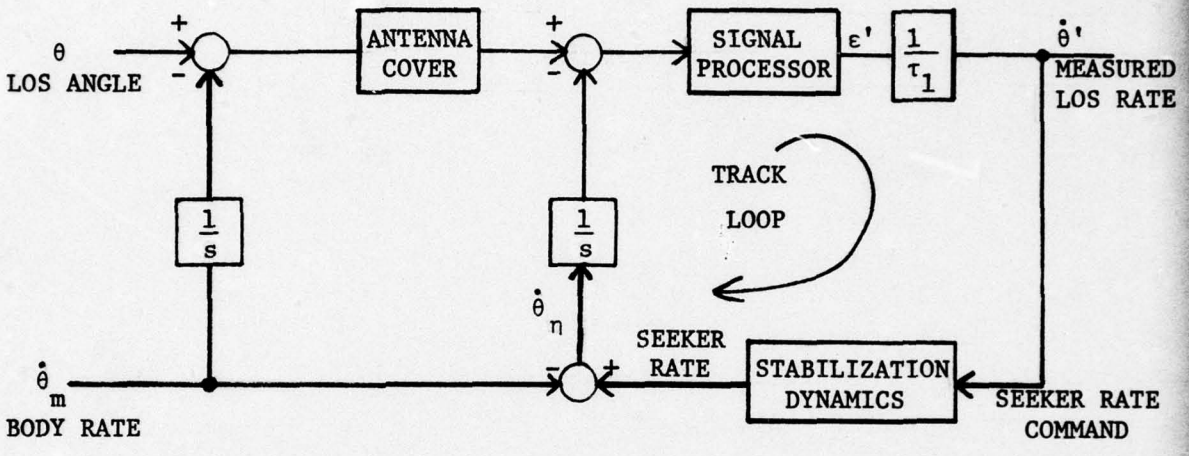


Figure 4. Seeker System Block Diagram

Seeker Track and Stabilization Loops. Figure 4 depicts the seeker system block diagram. The constant $\frac{1}{\tau_1}$ scales the indicated boresight error to obtain the desired rate command for the guidance system. The stabilization dynamics are typically comprised of a gimbal servo and rate gyro and has a very wide bandwidth. The track loop commands a gimbal rate proportional to measured boresight error. The loop attempts to drive boresight error to zero, thereby causing the antenna to track the target. Assuming unity gain for the antenna cover, signal processor, and stabilization dynamics, it is seen that

$$\frac{\varepsilon'}{\dot{\theta}} = \frac{\tau_1}{1 + s\tau_1} \tag{10}$$

For $\omega < \frac{1}{\tau_1}$, the indicated boresight error is proportional to the LOS rate. This is the desired relationship for classical proportional navigation. In the steady state case with $\dot{\theta}$ a constant, it can be seen

$$\varepsilon'_{max} = \tau_1 \dot{\theta}'_{max} \tag{11}$$

If τ_1 is sufficiently small in this relationship, it will be possible to keep ε' in the linear range of the received beamwidth.

The importance of aberration error is demonstrated in Figure 5 and the relationship

$$\frac{\theta'}{\theta_m} = \frac{-k_r}{1 + \tau_1 s} \tag{12}$$

From this it is seen that the LOS rate, $\dot{\theta}$, is corrupted by a term proportional to the body rate, $\dot{\theta}_m$. The body rate is a function of commanded acceleration, and the loop that is formed can have a destabilizing effect on the missile. Note that if $k_r = 0$ in Figure 5, the contribution of body rate cancels at the second summing junction.

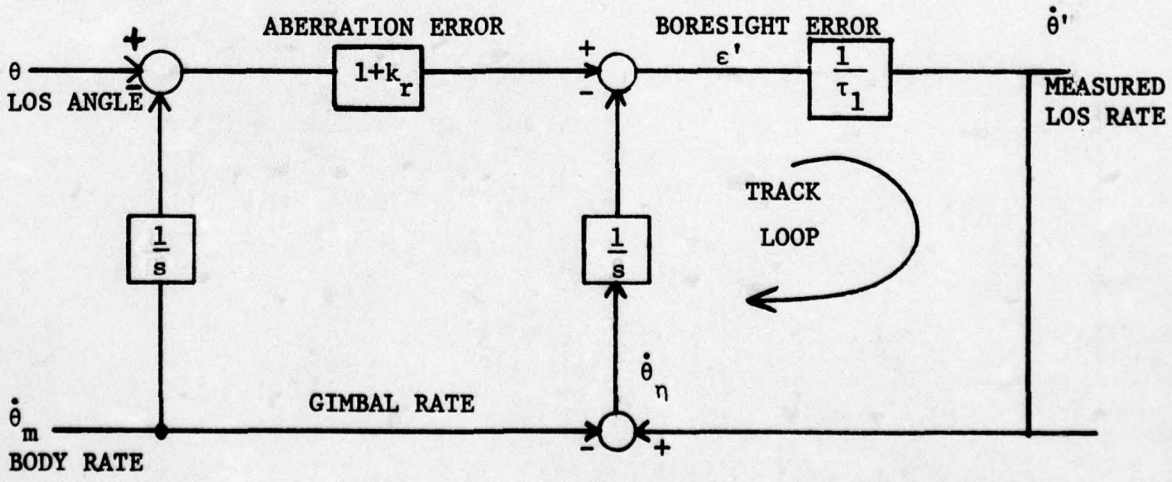


Figure 5. Seeker Model Plus Track Loop

The relationship between the measured LOS angle and measured LOS rate can be determined from Figure 5 by defining

$$\theta' = (1 + k_r)\theta - k_r\theta_m \tag{13}$$

and noting that

$$\varepsilon' = \theta' - \dot{\theta}'/s \tag{14}$$

Substituting the expression, $\varepsilon' = \tau_1\dot{\theta}'$, for ε' and solving for $\dot{\theta}'/\theta'$ yields

$$\frac{\dot{\theta}'}{\theta'} = \frac{s}{1 + \tau_1 s} \tag{15}$$

The measured LOS angle, θ', is depicted in the truth model diagram as the input to the seeker module. It is corrupted by an additive random measurement noise. The noise component is discussed in a later section. (Ref 13)

Missile Guidance

Before discussing the proportional navigation guidance scheme used in this study, it is important to note the distinction between "navigation", "guidance", and "control" systems. As defined by Blakelock (Ref 2), the navigation system automatically determines the position of the vehicle with respect to some reference frame while the guidance system generates the necessary command inputs to the control system to keep the vehicle on course. The control system, made up of the airframe and autopilot, controls the motion of the vehicle. The missile modelled in this study performs both the guidance and navigation functions, using the initial LOS frame for both purposes.

The operation of the guidance component consists of filtering the noisy measurements to command the missile lateral acceleration. A system using a constant bandwidth low-pass filter and having a fast measurement rate can be characterized as an analog on-board signal processor or high rate digital unit.

The output of the noise filter is a band limited representation of the LOS rate. Classical proportional navigation develops a lateral acceleration command for the missile normal to the LOS, proportional to the LOS rate, given by

$$a_\ell = \frac{n' V_c}{\cos(\theta_\ell - \theta)} \dot{\theta} \qquad (16\text{-a})$$

where θ_ℓ represents the missile lead angle and n' is a coefficient related to the proportional navigation constant of Equation (2) by $n = \frac{n'}{\cos(\theta_\ell - \theta)}$. The term $\cos(\theta_\ell - \theta)$ is included in the above expression since the velocity vector and LOS of the missile may not be coincident. Since the missile acceleration is developed normal to the missile velocity vector, it is necessary to use the term $\cos(\theta_\ell - \theta)$ to increase the acceleration so that the projection normal to the LOS will be proportional to the LOS rate, $\dot{\theta}$. However, if the guidance law functions properly, θ will remain small and θ_ℓ will be relatively small in addition to being well approximated as a constant. The result is that the $\cos(\theta_\ell - \theta)$ factor can be considered constant and can be combined with n' to form a redefined proportional navigation constant, n (i.e. Equation (2)). Therefore, in this study the value of "n" will remain constant unless it is changed by design.

In general, the navigation constants used by current missiles are in the range of three to six. The exact value of the constant is usually based on a tradeoff between line-of-sight rate noise and acceleration

saturation (Ref 2). As noted previously, some optimal schemes employ forms of the basic proportional guidance scheme. In addition, there are proportional navigation plus bias and proportional navigation plus dead space schemes that have been implemented. The guidance scheme in this study is classical proportional navigation. The assumed model for the guidance component is depicted in Figure 6.

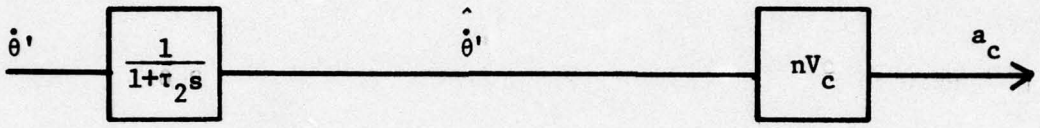


Figure 6. Guidance Law

Airframe/Autopilot Dynamics

The missile model adopted is based largely on the work done in (Refs 11, 12, and 13). The model that is simulated is a generic model of a cruciform, air-to-air missile employing a proportional navigation guidance scheme. The model assumes second order air frame dynamics and first order actuator dynamics. It also implements an adaptive autopilot. Consideration of an adaptive autopilot is realistic in view of current interest in such systems. The model used is depicted in Figure 7. It is assumed that the missile is non-thrusting during the time of interest. As indicated in the thesis introduction, this is not overly restrictive - particularly in an air-to-air missile scenario. Addition of thrust to the model would also add additional complexity to identifying the unknown parameters. Prior to analyzing the dynamics of the missile, certain additional assumptions are made.

Assumptions. For the purposes of the autopilot design, it is assumed that the control surface lift force is a linear function of control surface

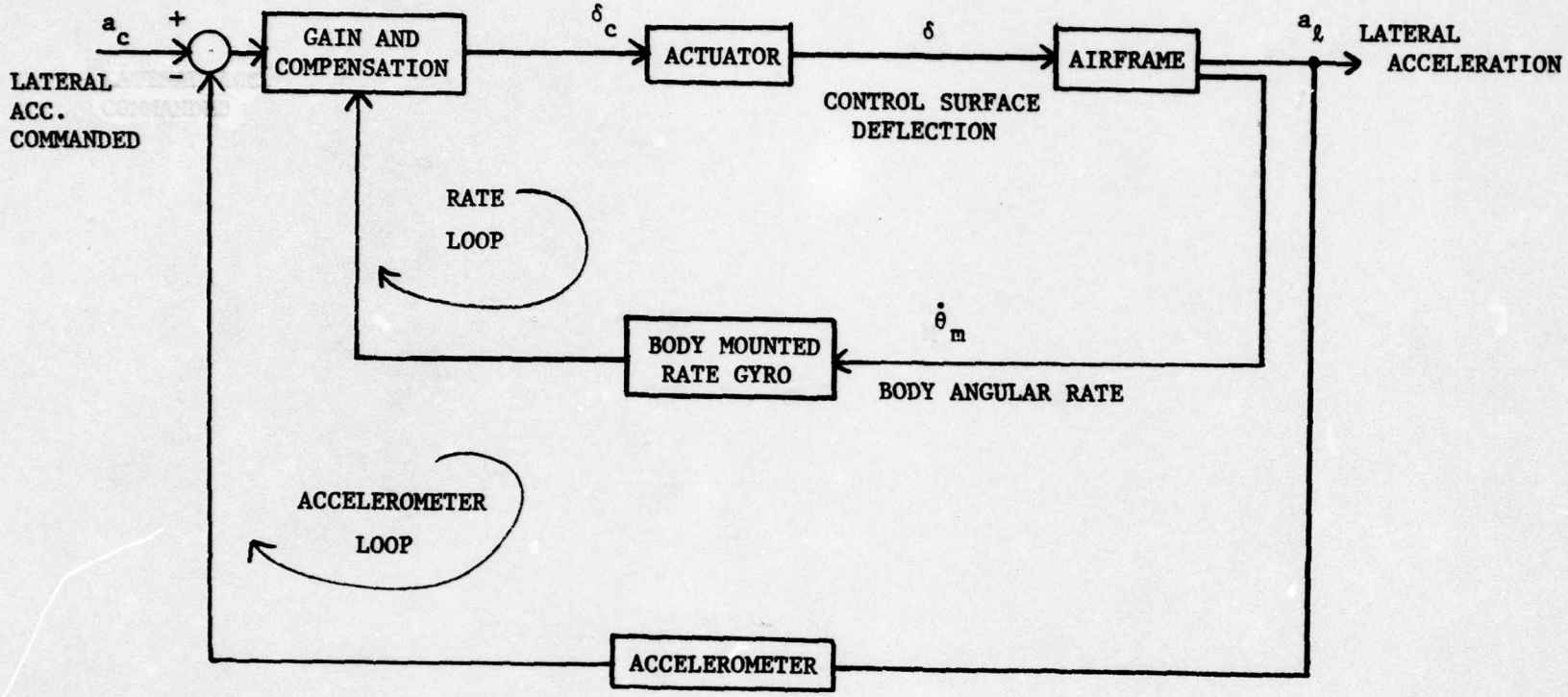


Figure 7. General Missile Autopilot Configuration

deflection angle, δ, and independent of angle of attack, α, as in Figure 8. It should be noted that in (Refs 12 and 13), linear techniques are applied to a nonlinear model since "experience has shown that the resulting autopilot response characteristics with the nonlinear airframe, are closely approximated by the linearized response characteristics near the given nominal conditions, for a properly designed autopilot" (Ref 13:A-16,A-17). Unlike the models used in these references, however, the model in this study allows for velocity change as a function of aerodynamic drag. In addition, the study in (Ref 13) also assumed a constant center of mass and center of pressure. These restrictive assumptions have been removed to allow a more realistic depiction of system characteristics.

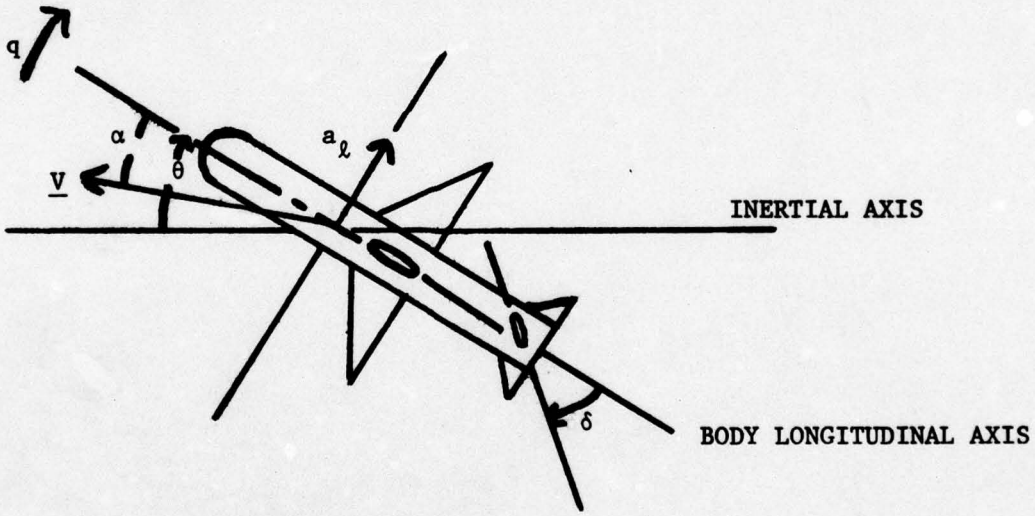


Figure 8. Geometrical Definitions of Yaw Plane Variables

Airframe Dynamics. By utilizing a cruciform missile, the pitch and yaw dynamics are identical, and the roll control system is primarily for stabilization. By looking at the planar case and by employing small angle approximations to remove nonlinearities, the following equations of motion result:

$$\dot{q}(t) = M_q q(t) + M_\alpha \alpha(t) + M_\delta \delta(t) \tag{17}$$

$$\dot{\alpha}(t) = q(t) - L_\alpha \alpha(t) - L_\delta \delta(t) \tag{18}$$

$$\dot{\delta}(t) = -\lambda \delta(t) + \lambda u(t) \tag{19}$$

$$a(t) = -V_m(\dot{\alpha}(t) - q(t)) \tag{20}$$

where M_q, M_α, M_δ, L_α, and L_δ are stability derivatives and

$q(t)$ = pitch rate

$a(t)$ = normal acceleration

$\delta(t)$ = control surface deflection

V_m = missile velocity

$\alpha(t)$ = angle of attack

$u(t)$ = control command

$1/\lambda$ = actuator time constant.

The stability derivatives can be expressed analytically in terms of aerodynamic coefficients, airframe parameters, airspeed, and dynamic pressure by the relations

$$M_q = \frac{\bar{q}Sd^2}{2I_{yy}V_m} C_{Mq} \tag{21-a}$$

$$M_\delta = \frac{\bar{q}Sd}{I_{yy}} C_{M\delta} \tag{21-b}$$

$$L_\delta = \frac{\bar{q}S}{mV_m} C_{N\delta} \tag{21-c}$$

$$M_\alpha = \frac{\bar{q}Sd}{I_{yy}} C_{M\alpha} \tag{21-d}$$

$$L_\alpha = \frac{\bar{q}S}{m_{Vm}} C_{N\alpha} \tag{21-e}$$

where

$\bar{q}$ = dynamic pressure

I_{yy} = moment of inertia about pitch axis

S = lifting surface area

d = characteristic length of the missile

C_{Mq}, $C_{M\alpha}$, $C_{M\delta}$ = moment coefficients

$C_{N\alpha}$, $C_{N\delta}$ = normal force coefficients

m = mass of missile.

The moment coefficients and normal coefficients vary as a function of the missile Mach. In this study they are approximated by a cubic curve fit of the plots for the appropriate coefficients of a generic missile. In addition to the coefficients in Equations (21-a) through (21-e), the coefficients of zero-lift drag, C_{DO}, and coefficient of induced drag, C_{DI}, need to be modelled. The sum of these terms yields

$$C_D = C_{DO} + C_{DI} \tag{22}$$

where C_D is the total coefficient of drag. This expression can be written in the equivalent form

$$C_D = C_{DO} + KC_L^2 \tag{23}$$

where C_L is the coefficient of lift and K is equal to the inverse of the partial derivative of the coefficient of lift with respect to the angle

of attack (approximately equal to the inverse of the partial derivative of the normal force coefficient with respect to the angle of attack), i.e.

$$K = \frac{1}{C_{L\alpha}} \approx \frac{1}{C_{N\alpha}} \tag{24}$$

The small angle approximation employed above is explained later in this chapter. Equating the right sides of Equations (22) and (23) and putting in the expression for K yields

$$C_{DI} = C_L^2 / C_{N\alpha} \tag{25}$$

The coefficient of lift, C_L, in the above expression is found by using the definition of lift

$$L \triangleq \frac{1}{2} \rho S C_L V_m^2 \approx Ma_\ell \tag{26}$$

where L is the lift developed and M is the missile Mach. Recall that a_ℓ is the lateral acceleration of the missile. Again a small angle approximation has been made. Solving for C_L yields

$$C_L = \frac{2Ma_\ell}{\rho S V_m^2} \tag{27}$$

The coefficient of zero-lift drag is approximated for a generic missile using the expression

$$C_{DO} = \frac{2}{\sqrt{M}} \tag{28}$$

where M is the missile mach number. Using the above relationships and the definition of drag force on the missile, D,

$$D = \frac{1}{2} \rho S C_D V_m^2 \tag{29}$$

the coefficient block of Figure 1 is simulated as depicted in Figure 9. The coefficients are held constant over the individual integration intervals of the simulation along with target and missile velocity/Mach. The remainder of the parameters in Equations (27) and (29) were given values from a generic missile.

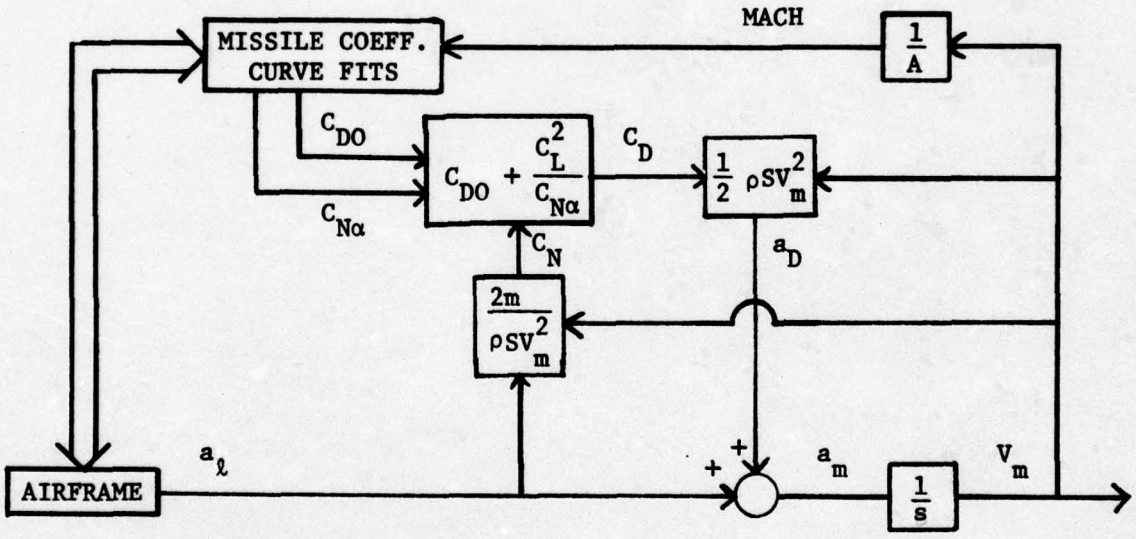


Figure 9. Determination of Missile Coefficients

By neglecting actuator dynamics and setting the actuator feedback gain, h_3, equal to zero in Figure 10, the determination of the airframe parameters

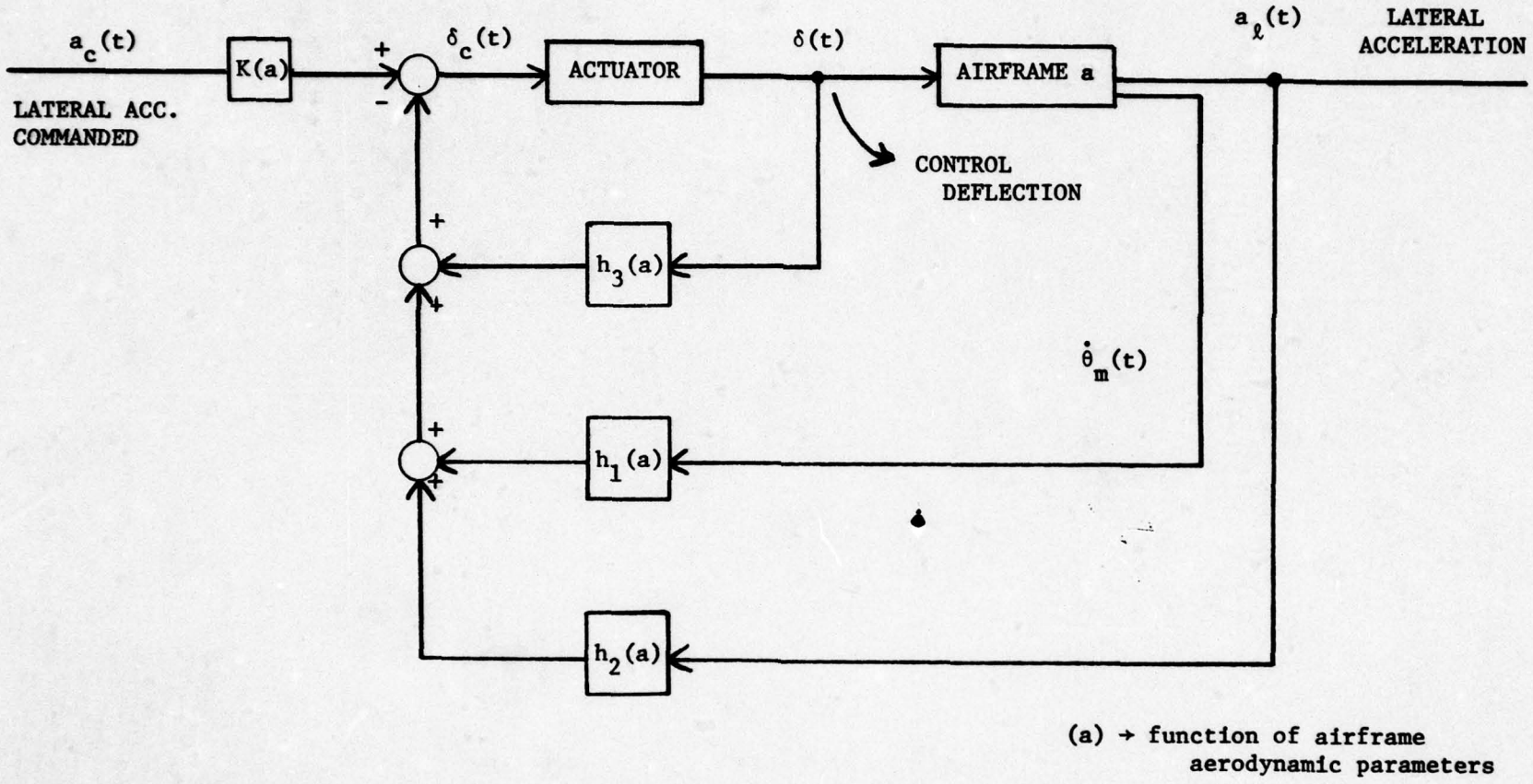


Figure 10. Autopilot Adaptive Gain Configuration

most important for establishing the autopilot feedback gains is greatly simplified. This is justified when the actuator response characteristics are much faster than those specified for the autopilot. This simplification implies that, $\lambda = \infty$, and $u(t) = \delta(t)$ in Equation (19), and that the equations of motion become

$$\begin{bmatrix} \dot{q}(t) \\ \dot{\alpha}(t) \end{bmatrix} = \begin{bmatrix} M_q & M_\alpha \\ 1 & -L_\alpha \end{bmatrix} \begin{bmatrix} q(t) \\ \alpha(t) \end{bmatrix} + \begin{bmatrix} M_\delta \\ -L_\delta \end{bmatrix} \delta(t) \tag{30-a}$$

$$\delta(t) = -h_1 q(t) - h_2 a(t) + kv(t) \tag{30-b}$$

$$a(t) = -V_m(\dot{\alpha}(t) - q(t)) \tag{30-c}$$

where k represents a varying input gain and v(t) is the input command acceleration. By use of Laplace Transforms, it can be shown that the closed loop poles are the roots of

$$p(s) = s^2 + \left[\frac{h_1 M_\delta - h_2 V_m L_\delta M_q - (M_q - L_\alpha)}{1 + h_2 V_m L_\delta} \right] s$$

$$+ \left[\frac{(-h_1 - h_2 V_m)(L_\delta M_\alpha - M_\delta L_\alpha) - M_q L_\alpha - M_\alpha}{1 + h_2 V_m L_\delta} \right] \tag{31}$$

Under the assumption that the actuator dynamics are negligible, the dominant poles of the missile response are those associated with the airframe dynamics. For the second order model this response is determined by the damping ratio, ζ, and undamped natural frequency, ω, in the relationship

$$p(s) = s^2 + 2\zeta\omega s + \omega^2 \tag{32}$$

By equating terms of like powers in Equations (31) and (32) these expressions can be solved for the feedback gains, h_1 and h_2 as

$$h_1 = \frac{-L_\alpha[M_\alpha - \frac{M_\delta}{L_\delta} L_\alpha + \omega^2 + (2\zeta\omega + M_q)(\frac{M_\delta}{L_\delta} + M_q)}{M_\delta(M_\alpha - \frac{M_\delta}{L_\delta} L_\alpha + \omega^2) + (M_q + 2\zeta\omega)(L_\delta M_\alpha - M_\delta L_\alpha)} \tag{33}$$

$$h_2 = \frac{L_\delta M_\alpha(L_\alpha - 2\zeta\omega - M_q) + M_\delta[L_\alpha(2\zeta\omega - L_\alpha) - \omega^2 - M_\alpha]}{V_m L_\delta[M_\delta(M_\alpha - \frac{M_\delta}{L_\delta} L_\alpha + \omega^2) + (M_q + 2\zeta\omega)(L_\delta M_\alpha - M_\delta L_\alpha)]} \tag{34}$$

Finally, by specifying ζ and ω and having the stability derivatives evaluated, the closed loop poles of the autopilot can be determined. For this study ζ was set equal to 0.707 and ω equal to 7.07 rad/sec. A damping ratio of .707 results in a minimum settling time for a second order system. In addition, these values for the damping ratio and natural frequency are realistic values for a missile with pitch-rate and acceleration feedback. (The values for both quantities tend to be much lower for a missile having no autopilot (Ref 12:D-3).

To achieve a unity closed loop autopilot gain (i.e. lateral acceleration of missile equals acceleration commanded), one final parameter, the input gain, is developed. Looking at the steady state solution for the case in which the commanded acceleration, a_c, equals one, the value of k is adjusted to make the lateral acceleration, a_ℓ, equal one. This gives the expression for k as

$$k = \frac{h_1}{V} + h_2 - \frac{1}{V}\left(\frac{M_\alpha + L_\alpha M_q}{L_\alpha M_\delta - M_\alpha L_\delta}\right) \tag{35}$$

At this point the airframe dynamics and autopilot component blocks have been completely specified as functions of the appropriate dynamic and design parameters. Physical systems, however, are subject to structural and instrument constraints. To account for these real world constraints an acceleration limiter with a value of ±15 g's is included in the truth model. As will be noted in the simulation, the limiter will not be exercised. This is true for this study only and would not be the case in certain encounter geometries.

System Noises

This section models the random noise components of the system truth model. As depicted in Figure 1, four separate random noise sources are considered in the system truth model: missile seeker LOS angle noise, tracker LOS angle noise, range noise, and range rate noise. The noise input into the seeker creates an error in the missile commanded lateral acceleration, while the three measurement noises create tracking errors in the tracker system. It is important to recall that, in the case of the tracker, perfect tracking is accomplished by the servo system. The noises that actually corrupt the four physical measurements are themselves a complex mixture of individual system disturbances. For the tracker angle, range, and range rate noises, Ref (3) classifies the various sources of noise as radar-dependent tracking error, radar-dependent translation errors, target-dependent tracking error, and propagation error (Ref 9:326,373, 400). Ref (13), however, categorizes the noises according to their dependency on missile-to-target range. This study will consider the noise types to fall into three categories: "glint", "scintillation", and "thermal"

noise. This approach is consistent with, and includes, the salient features of the references cited in this study. A description of the various noise types follow.

"Glint". Glint is defined in this study as angle scintillation, as opposed to amplitude scintillation, to be discussed subsequently. It refers to the disturbances in apparent angle of arrival of the return signal due to interferences (i.e. phase distortions). Physically it represents the wandering of the apparent center of the radar reflection. As pointed out in (Ref 1), the center of reflection may at times lie well outside the physical limits of the target (Ref 15:47). The importance of glint is readily apparent, in as much as a large, abrupt variation in measured radar angle will be interpreted as a change in angular velocity of the target.

It is found that a first order Gauss-Markov process, i.e. the output of a first order lag driven by white Gaussian noise, provides a good curve fit to the ensemble statistics of the glint spectrum. This model proves suitable for both large, slow moving and small, rapidly moving targets. The modelling difference in the two cases lies in the proper choice of lag time constant and signal strength of the input white noise. In addition, however, glint is dependent on both range and target aspect angle. In general, glint will vary inversely with both range and the instantaneous cross section of the target. The effect of varying aspect ratio is the appearance of large spikes in the spectral density at points of phase discontinuities. A model for the spike characteristic is developed in (Ref 15). However, for the purposes of this study, the glint will be modelled as a simple first order Markov Process. It will be assumed that it is independent of both range and aspect angle. This approach is taken in view of time and computer constraints. By assuming a point mass missile, the

latter assumption is reasonable. The range assumption is reasonable for the missile noise since the missile seeker is in general a passive or semi-active system. In either case, the noise can be justified as range independent (Ref 13:A-14). Therefore, the strength of the input white noise for the glint model is assumed to be constant.

"Scintillation". Scintillation, or more specifically amplitude scintillation, refers to amplitude distortions of the received signal. In general, scintillation is directly proportional to range (i.e. decreasing signal-to-noise ratio). For the ranges considered in this report, this dependence can be eliminated since the amplitude of received signal will not vary significantly. Two additional features of scintillation should be noted. First, it is also dependent on the target aspect angle. Ref (15) states that the phase and amplitude distortions due to aspect angle are negatively correlated. As in the case of glint, however, this effect is not modelled for the assumed point mass. Secondly, the frequency spectrum of the scintillation can be assumed to be identical to that of the glint spectrum (Ref 1:290). This allows for a "lumping" of the glint and scintillation spectrums into one first order Markov Process.

"Thermal". The final noise component to be considered is the thermal noise. This form of noise is generated by the background of "black body" radiation - both environmental and receiver noise. A good example of this form of noise is "shot noise" in electronic tubes. Thermal noise is inversely proportional to the signal-to-noise ratio, which in turn is inversely proportional to range. It is characteristically modelled as a white noise over the bandwidth of the systems considered (Ref 13:A-11). The assumed model for this study is a white Gaussian noise of a constant strength. While this model neglects the effect of range dependency, it should not seriously degrade the results of the study.

Based upon the above descriptions of the noise types, all four of the system noises are modelled as the sum of an exponentially correlated Gaussian noise added to a Gaussian white noise. The correlated portion of the noise models the glint and scintillation error components while the white noise source represents the thermal component. The summing of the noise types to achieve the total noise for each system noise is based upon the statistical description of the various noises as completely uncorrelated with each other. It has been found in practice that this assumption is valid (Ref 1:324). Using the range measurement noise as an example, each system noise can be described by an autocorrelation

$$E\{\eta_R(t)\eta_R(t+\tau)\} = \sigma^2_{g/S} e^{-|\tau|/\tau R} + \sigma^2_T \delta(\tau) \qquad (36)$$

where $\sigma^2_{g/S}$ = strength of glint/scintillation noise corrupting the range measurement

σ^2_T = strength of thermal noise corrupting the range measurement.

Table I lists the standard deviation, σ, and correlation time, τ, for the various system noises. The values of the noises are obtained from a composite of figures received from AFAL and the values used in (Ref 15). The values of the individual seeker noise components were determined by using the one sigma values from (Ref 5) as the total noise strength and using the same ratio of thermal to glint/scintillation strength as found in the tracker angle measurement channel.

Noise Generator. Each of the eight components of noise was produced using a call to Subroutine NOISE (see Appendix C). The essential procedure is to generate a unit variance Gaussian noise and to multiply it by the strength desired. For a white noise the strength is σ^2, while

for an exponentially correlated noise the strength of the input white noise is equal to $2\lambda\sigma^2$ where $\lambda = \frac{1}{\tau}$. (Ref 9:345)

Table 1
Strengths and Time Constants for Measurement Noises

MEASUREMENT	σ	τ
θ_{MS}(radians)		
THERMAL	.003	--
GL/SCINT	.000894	.1(sec)
θ_T(radians)		
THERMAL	.00126	--
GL/SCINT	.00168	.5(sec)
R_T(feet)		
THERMAL	11.7	--
GL/SCINT	10.0	.5(sec)
$\dot{R}_T$(feet/sec)		
THERMAL	7.0	--
GL/SCINT	4.242	1(sec)

Missile-Tracker Kinematics

Three reference frames are used to describe the dynamics of the engagement scenario. The first of these, the initial missile line-of-sight frame (IMLOS), is instrumented by the missile guidance equipment to accomplish target intercept. The second and third frames are the tracker inertial frame (I) and tracker line-of-sight frame (TLOS). Both the TLOS and I frame are used in determining tracker/missile kinematics. The tracker measurements, taken in the TLOS frame, are used to update the tracking filter algorith, while the desired outputs of the filter algorithm are coordinatized in the I-frame. The motivation for using two tracker frames was given in Chapter One.

Initial Missile Line-of-Sight Frame. Both the motivation for, and development of, the IMLOS frame are well developed in (Refs 5 and 13). Unlike these sources, however, both the tracker and missile accelerations

and velocities will be allowed to vary - the trackers in an unspecified manner, and the missiles as a result of aerodynamic drag affects. The intercept geometry is depicted in Figure 11. The IMLOS frame does not rotate, but does translate with the missile. It is assumed that an off-boresight launch capability is available. Having assumed this capability the desired missile lead angle, θ_ℓ, is given by

$$\theta_\ell = \sin^{-1}\left(\frac{V_T}{V_m}\sin\theta_a\right) \tag{37}$$

where V_T and V_m are the velocities of the tracker vehicle and missile respectively, and θ_a is the angle the tracker makes with IMLOS as depicted in Figure 11. The angle θ_ℓ varies as

$$\theta_\ell = \frac{a_\ell}{V_m} \tag{38}$$

The initial target angle, θ_α, is set by initial conditions at the time of missile tracking initiation, and varies according to

$$\dot{\theta}_a = \frac{a_{t_2}}{V_t} \tag{39}$$

where a_{t_2} is the tracker acceleration perpendicular to the V_t vector. Both a_ℓ and a_{t_2} are defined as positive in the direction that results in an increasing θ_ℓ and θ_a respectively. While defining positive a_ℓ in this manner is adequate for the IMLOS frame, it is not compatible with the TLOS frame. This inconsistency is easily solved by a sign change in the computer simulation as will be discussed later.

The change in the relative velocity normal to the LOS is

$$\dot{y}_d = V_t\sin\theta_t - V_m\sin\theta_\ell \tag{40}$$

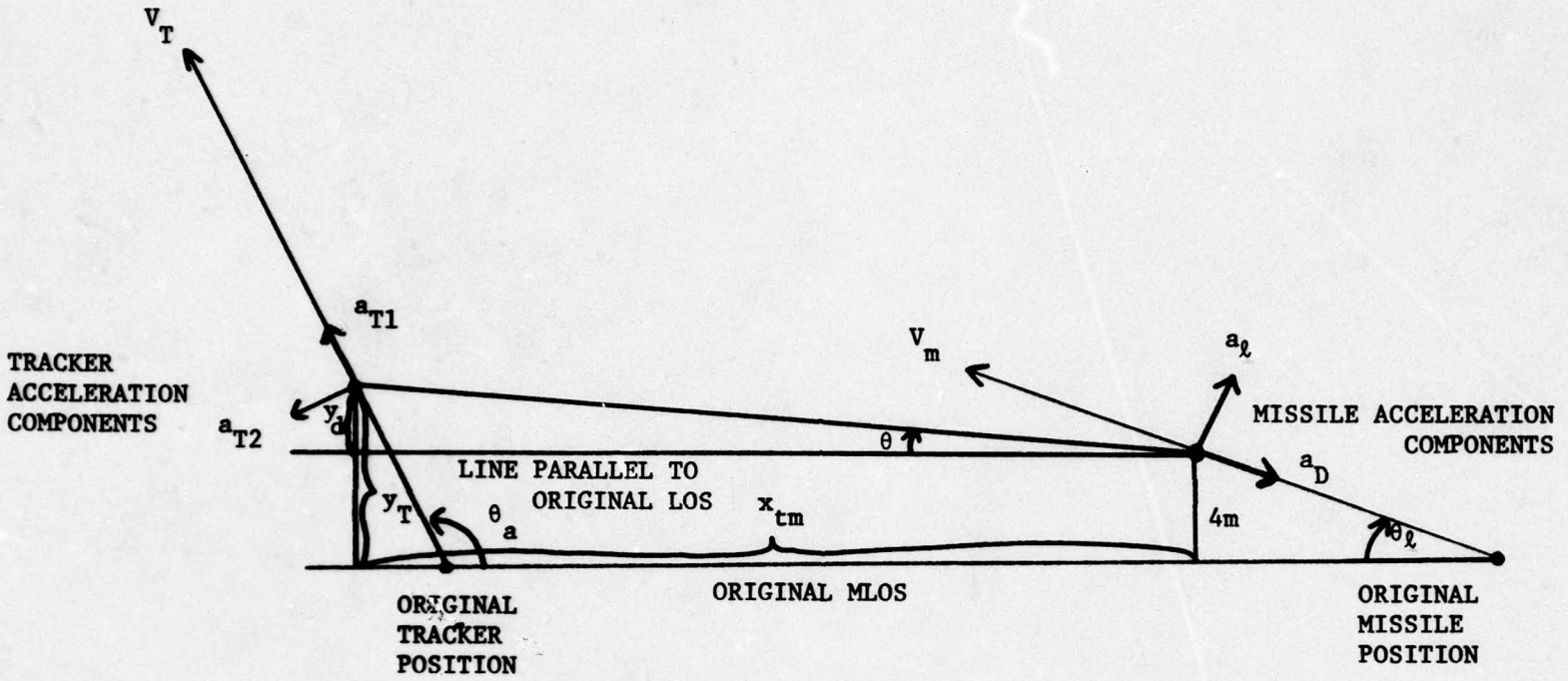


Figure 11. Missile Intercept Geometry/Initial Line-of-Sight Frame (IMLOS)

where y_d is the lateral distance between the tracker and missile. Finally, it is seen that (referring to Fig 11):

$$\theta = \tan^{-1}\left(\frac{y_d}{x_{t_m}}\right) \cong \frac{y_d}{x_{t_m}} \tag{41}$$

where x_{t_m} is the projection of missile-to-target range on the original line-of-sight direction. The approximation in Equation (41) is valid for a properly designed guidance scheme (i.e. the case in which y_d, a good approximation of miss distance, is kept small). Note also that as x_{t_m} approaches zero, the angle, θ, gets large. This is the source of the terminal high-g condition so common to proportional navigation.

Tracker Frames. Both reference frames associated with the tracker are depicted in Figure 12. It will be assumed that the aircraft-instrumented navigation frame is a stationary/local inertial frame. This will be valid for the short duration of the engagement scenario. The choice of this frame is convenient since several of the parameters that are required in filter implementation are available from the instruments required to mechanize the frame. Among these are rate-integrating gyros on the antenna axis and accelerometers on the platform. From Figure 12 it is readily seen that the direction cosine matrix for transforming from the I to the TLOS frame

$$C_I^{TLOS}\begin{bmatrix} \cos\theta_T & \sin\theta_T & 0 \\ -\sin\theta_T & \cos\theta_T & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{42}$$

The antenna centerline is defined as the x^{TLOS} axis. It is assumed that the tracker is capable of instantaneous, errorless response to commanded signals (i.e. errors in θ_T are due only to errors in system measurements).

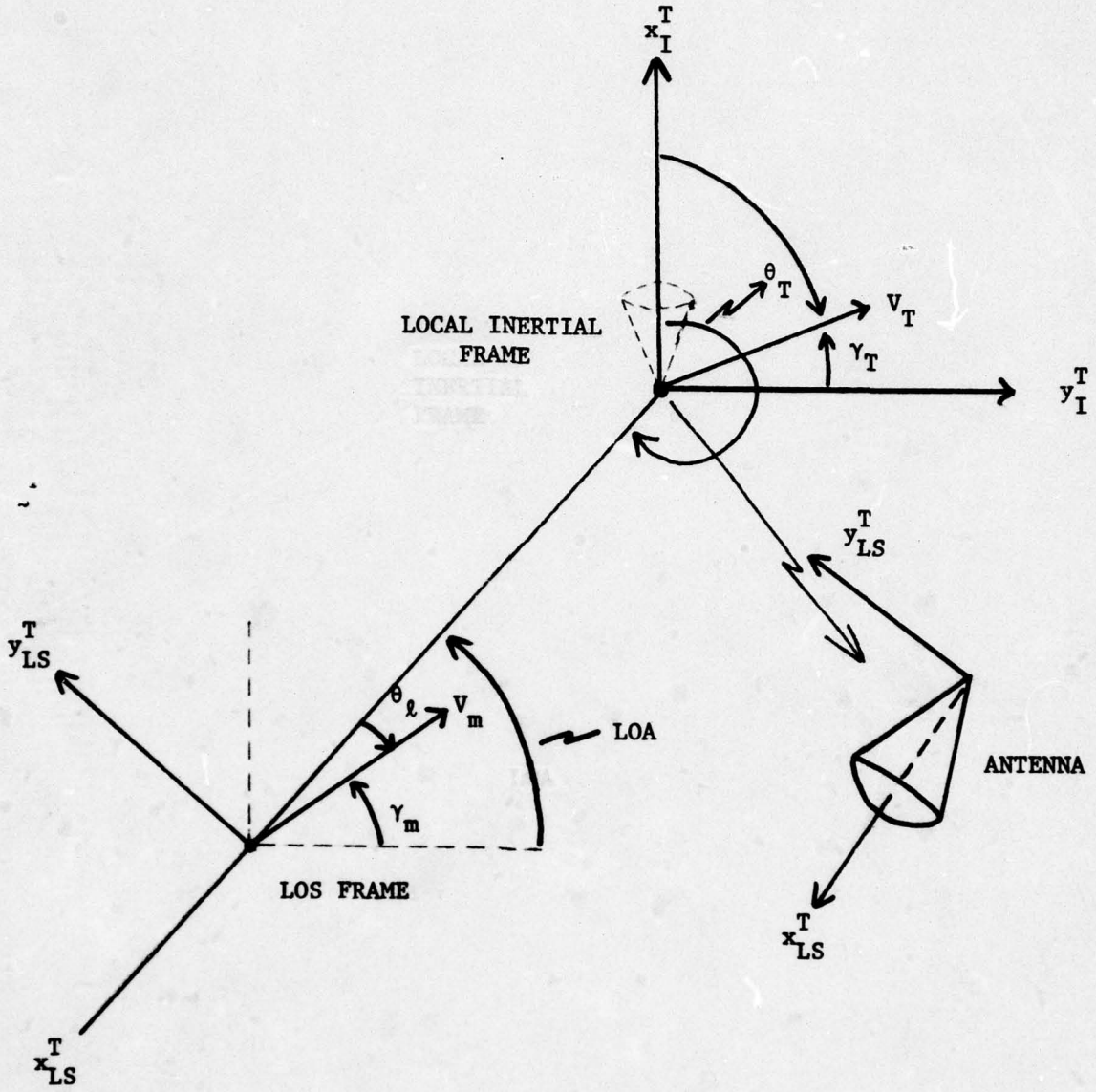


Figure 12. Missile-Tracker Engagement Geometry

This simplification reduces system complexity while still allowing analysis of the dominant features of the missile motion.

Missile Kinematics. By utilizing the piecewise continuous assumption, very simple kinematic relations can be established. Significantly, missile behavior is well enough behaved that, despite using this assumption, relatively sophisticated models can be used for missile dynamics, parameters, and coefficients. Figure 12 illustrates the engagement geometry. The components of the inertially referenced missile velocity are given by

$$V_{mx}^{I} = \dot{x}_{m}^{I} = V_{m}\sin\gamma_{m} \tag{43-a}$$

$$V_{my}^{I} = \dot{y}_{m}^{I} = V_{m}\cos\gamma_{m} \tag{43-b}$$

where

V_{mx}^{I}, V_{my}^{I} = x and y components respectively of missile velocity in I frame

x_{m}^{I}, y_{m}^{I} = x and y position of missile in I frame

$\gamma_{m} = \tan^{-1}\left(\frac{V_{mx}^{I}}{V_{my}^{I}}\right)$

This convention will hold throughout the remainder of this study. To develop the missile acceleration it is necessary to examine certain aerodynamic quantities.

Figure 13 is a free body diagram of the non-thrusting missile. The forces acting on the missile are lift and drag. It is assumed that the velocity of the missile is aligned with the longitudinal axis of the missile. Therefore, by definition, the drag will be along the longitudinal axis of the missile and opposite in direction to the velocity. It follows that the missile lift force is normal to the longitudinal axis. These

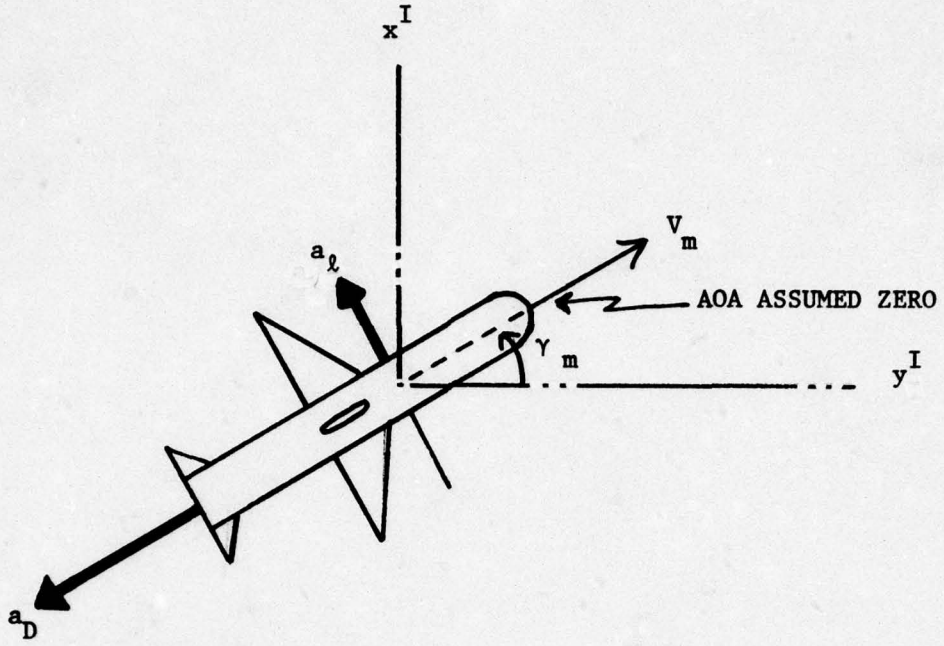


Figure 13. Missile Free Body Diagram

assumptions imply that the angle-of-attack, α, of the missile is zero. This will not be true in the presence of missile deceleration and/or maneuvering. However, due to a combination of high airspeeds and structural g-limitations, missiles are generally designed for angles-of-attack of less than five degrees. For the Mach range considered in this study, a small angle approximation is valid. Using the definitions listed in Equations (26) and (29) it follows directly that the accelerations due to drag and lift are given by

$$a_D = \frac{1}{2} \rho S C_D V_m^2 / m \tag{44}$$

and

$$a_\ell = \frac{1}{2} \rho S C_L V_m^2 / m \tag{45}$$

From Figure 13 it can be seen that

$$a_{mx}^{I} = \cos\gamma_m \cdot a_\ell - \sin\gamma_m \cdot a_D \tag{46-a}$$

and

$$a_{my}^{I} = -\sin\gamma_m \cdot a_\ell - \cos\gamma_m \cdot a_D \tag{46-b}$$

where a_{mx}^{I} and a_{my}^{I} represent the x and y-accelerations respectively of the missile coordinatized in the I frame. Equations (46-a) and (46-b) utilize a positive a_ℓ as depicted in Figure 13. The sense of the a_ℓ vector is correct since a missile acceleration in the direction depicted creates a positive rate of change of the TLOS angle, i.e., $+\dot{\theta}_T$. Note, however, that in the IMLOS frame a_ℓ is positive in the opposite direction. The sign difference is compensated for by commanding the missile with a minus times the rate of change of the TLOS angle. Finally, from Equations (42), (46-a), and (46-b), it is seen that the missile acceleration along the instantaneous missile-to-tracker LOS, a_{mx}^{TLOS}, can be represented by

$$a_{mx}^{TLOS} = \cos\theta_T a_{mx}^{I} + \sin\theta_T a_{my}^{I} \tag{47-a}$$

or

$$a_{mx}^{TLOS} = \cos\theta_T(\cos\gamma_m a_\ell - \sin\gamma_m a_D) + \sin\theta_T(-\sin\gamma_m a_\ell - \cos\gamma_m a_D) \tag{47-b}$$

Tracker Kinematics. The motion of the tracker aircraft is described by a set of closed form equations. The actual motion of a target aircraft would, in general, be best described by a random process. The chosen mathematical description, however, is generic in nature and does allow for several important types of evasive aircraft maneuvers such as high "g" turns or "jinking" maneuvers. It is assumed that the tracker is located on the target aircraft. This assumption is required since the tracker could be carried on a remote "area defense" vehicle. The tracker position equations are

$$x_T^I = K_1 t + K_2 \cos\omega t \tag{48-a}$$

$$y_T^I = K_3 t + K_4 \sin\omega t \tag{48-b}$$

where

x_T^I, y_T^I = x and y positions of tracker in I frame

K_1, K_2, K_3, K_4 = chosen constants

ω = tracker rate of rotation.

It follows from successive differentiations that

$$V_{Tx}^I = K_1 - K_2 \omega \sin\omega t \tag{49-a}$$

$$V_{Ty}^I = K_3 + K_4 \omega \cos\omega t \tag{49-b}$$

and

$$a_{Tx}^I = -K_2 \omega^2 \cos\omega t \tag{50-a}$$

$$a_{Ty}^{I} = -K_4\omega^2 \sin\omega t \tag{50-b}$$

where V_{Tx}^{I} and V_{Ty}^{I} are the x and y-velocity components of the tracker and a_{Tx}^{I} and a_{Ty}^{I} the x and y-acceleration components of the tracker. Finally from Equations (42), (50-a), and (50-b) it is seen that

$$a_{Tx}^{TLOS} = \cos\theta_T(-K_2\omega^2\cos\omega t) + \sin\theta_T(-K_4\omega^2\sin\omega t) \tag{51}$$

Relative Kinematics. The use of three reference frames requires the transformation of missile accelerations into the TLOS frame and into the I frame. One additional transformation, not previously discussed, is required to transform tracker acceleration into a tracker body frame in order to determine a_t in Equation (39). Prior to establishing this additional transformation, however, the set of angles that describe the relationships between various system frames and vectors quantities, as depicted in Figures 12 and 14, are listed:

$$\gamma_T' = 90 - \sin^{-1}(V_{Tx}^{I}/V_T) \tag{52-a}$$

$$\theta_T = 270° - \text{LOA} \tag{52-b}$$

$$\theta_0 = 270° - \text{LOA}(0) \tag{52-c}$$

$$\phi_0 = \theta - \gamma_T' = \theta_a(0) \tag{52-d}$$

$$\phi_1 = 360° - \theta_a \tag{52-e}$$

$$\phi_m = \theta_\ell \tag{52-f}$$

$$\gamma_m = LOA - \phi_m \qquad (52\text{-}g)$$

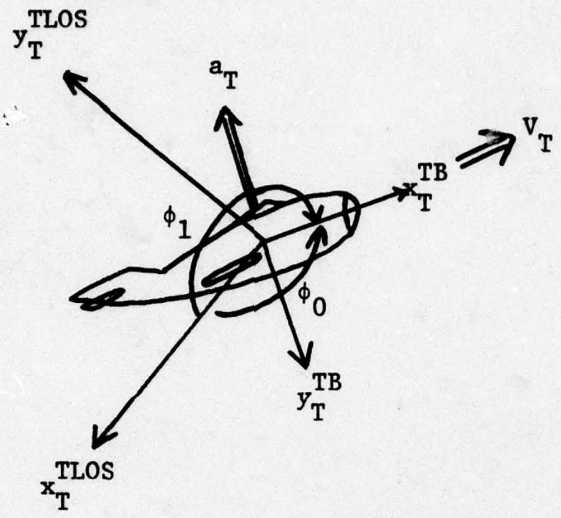


Figure 14. TLOS to Tracker Body Frame Transformation

The angles defined in Equations (52-a) through (52-g) restrain the initial conditions of the simulation. With the I frame centered at the target at time equal to zero, the missile initial position must lie in the third quadrant, (i.e. both x_m^I and y_m^I must be less than or equal to zero), as in Figure 15. Though appearing restrictive, the initial conditions allow for all engagements from head-on attacks to tail attacks and all angles between. This is achieved by either varying initial position in the third quadrant and/or initial tracker heading.

The angle ϕ_1 is used in the transformation of the tracker acceleration in the TLOS frame into the tracker body frame

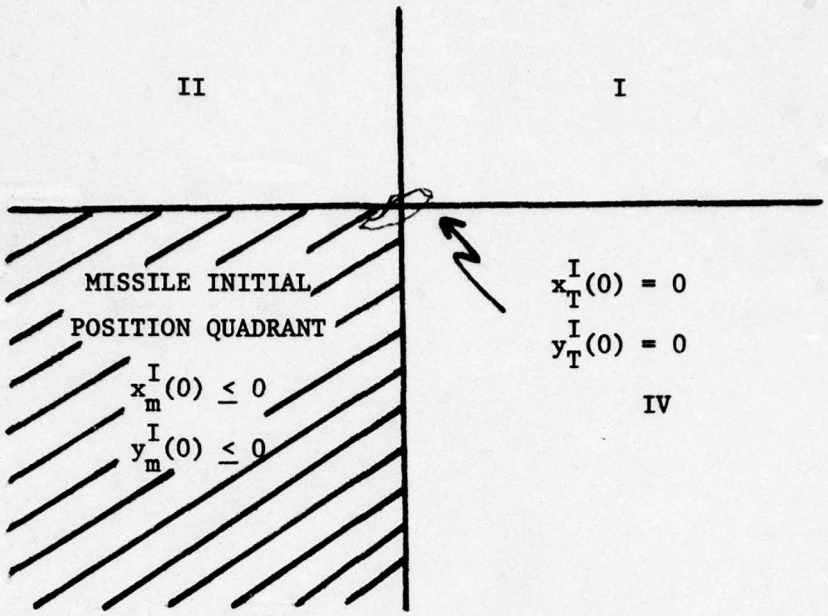


Figure 15. Truth Model Position Initial Conditions

$$C_{TLOS}^{TB} = \begin{bmatrix} \cos\phi_1 & \sin\phi_1 & 0 \\ -\sin\phi_1 & \cos\phi_1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{53}$$

from which it is seen

$$a_t = -a_{Ty}^{TB} = \sin\phi_1 a_{Tx}^{TLOS} - \cos\phi_1 a_{Ty}^{TLOS} \tag{54}$$

This relationship is depicted in Figure 14. Note that the x-axis of the tracker body frame is aligned with the tracker velocity.

The missile inertial quantities were obtained by generating a_ℓ and a_D and then resolving them into inertial coordinates using the angle the missile velocity vector makes with the inertial y-axis, γ_m. With both tracker and missile kinematics in the inertial frame, it can be seen from Figure 12 that the range equals

$$R = [(x_T^I - x_m^I)^2 + (y_T^I - y_m^I)2]^{\frac{1}{2}} \tag{55}$$

Range rate is defined as the relative velocity along the instantaneous line-of-sight

$$V_c = -\dot{R} \tag{56}$$

or expressed differently

$$V_c = (V_{xT}^I - V_{xm}^I)\text{sinLOA} + (V_{yT}^I - V_{ym}^I)\text{cosLOA} \tag{57}$$

A complete listing and sequencing of the various transformations can be found in the computer flowchart and listing in Appendix C.

System Equations

This section develops the equations of the missile airframe, autopilot, and dynamics. The state space format is used for these equations, but no attempt is made to describe the whole of the truth model in a state space representation. There are fourteen equations developed - the first twelve of which are developed in (Ref 5) and are repeated here only for completeness. The section ends with a listing of those quantities to be estimated by the Kalman filters.

Prior to developing the equations, it is convenient to define the states and the several quantities used in the development. Definitions of several of these quantities can be found elsewhere in the study, but are repeated here for clarity. The 14 states are

$x_1 = y_d$, missile-to-target lateral distance

$x_2 = \theta_\ell$, missile lead angle

$x_3 = \frac{1}{R}$, reciprocal of missile-to-target range

$x_4 = \theta_m$, angle between missile centerline and nonrotating reference line

$x_5 = \dot{\theta}_m$, missile pitch rate

$x_6 = a_\ell$, missile lateral acceleration

$x_7 = \delta$, elevator deflection

$x_8 = a_c$, missile commanded acceleration

$x_9 = \varepsilon' - \theta'$, defined state, where ε' = boresight error signal and θ' = measured LOS angle

$x_{10} = \theta_a$, tracker angle wrt IMLOS

$x_{11} = a_t$, tracker acceleration normal to IMLOS

x_{12} = missile seeker noise

$x_{13} = V_{mx}^{IMLOS}$, x-velocity component of missile in IMLOS frame

$x_{14} = V_{my}^{IMLOS}$, y-velocity component of missile in IMLOS frame

In addition

h_1, h_2 = autopilot feedback gains

V_m/V_t = missile/tracker velocity

K = autopilot input gain

k_r = aberration error slope

M_α, M_δ, M_q, L_α, L_δ = stability derivatives

$q(t)$ = pitch rate

$1/\lambda$ = actuator time constant

τ_1 = track loop time constant

τ_2 = guidance loop time constant

θ' = measured LOS angle

θ_f' = filtered LOS angle

$u(t)$ = input to autopilot

The equations for $\dot{x}_1$ and $\dot{x}_2$ are derived directly from Equations (38) and (40)

$$\dot{x}_1 = V_t \sin x_{10} - V_m \sin x_2 \tag{58}$$

$$\dot{x}_2 = (g/V_m) x_6 \tag{59}$$

The "g" in Equation (59) is present since all accelerations in these equations are in terms of g's. The third "state variable" equation is

$$\dot{x}_3 = -\frac{1}{R^2}\dot{R} = -x_3^2 \dot{R} \tag{60}$$

But using Equation (56) this can be written as

$$\dot{x}_3 = -x_3^2 V_c \tag{61}$$

In Equation (61), V_c is considered a positive quantity. The equation for $\dot{x}_4$ is

$$\dot{x}_4 = x_5 \tag{62}$$

The equation for $\dot{x}_5$ is developed by utilizing the expression

$$a_\ell(t) = V_m(L_\alpha \alpha(t) + L_\delta \delta(t)) \tag{63}$$

for acceleration. Solving for the angle of attack

$$\alpha(t) = \frac{a_\ell(t) - V_m L_\delta \delta(t)}{V_m L_\alpha} \tag{64}$$

and substituting this expression into Equation (17) and recognizing $q(t) = \dot{\theta}_m = x_5$, the result is

$$\dot{x}_5 = M_q x_5 + \frac{qM_\alpha}{V_m L_\alpha} x_6 + (M_\delta - \frac{L_\delta}{L_\alpha} M_\alpha)x_7 \tag{65}$$

The next equation is developed by taking the derivatives of Equation (62) and substituting in for $\dot{\alpha}(t)$ from Equation (18) and $\alpha(t)$ from Equation (64)

$$\dot{a}_\ell(t) = L_\alpha q(t) - L_\alpha a_\ell(t) - \lambda V_m L_\delta \delta(t) + \lambda V_m L_\delta u(t) \tag{66}$$

Finally, from Figure 10, u(t) can be seen to be

$$u(t) = -h_1 q(t) - h_2 a_\ell(t) + K a_c(t) \tag{67}$$

therefore

$$\dot{x}_6 = \frac{1}{g}(V_m L_\alpha - h_1 \lambda V_m L_\delta)x_5 - (L_\alpha + h_2 \lambda V_m L_\delta)x_6$$

$$- \frac{1}{g} \lambda V_m L_\delta x_7 + \lambda V_m L_\delta K x_8 \tag{68}$$

The equation for $\dot{x}_7$ comes from Equations (19) and (67) which yield

$$\dot{x}_7 = -h_1 \lambda x_5 - g h_2 \lambda x_6 - \lambda x_7 + g \lambda K x_8 \tag{69}$$

From Figure 6, the transfer function of $a_c/\dot{\theta}'$ can be written

$$\frac{a_c}{\dot{\theta}'} = \frac{nV_c}{1 + \tau_2 s} \tag{70}$$

therefore

$$\dot{x}_8 = \frac{nV_c}{\tau_2}\dot{\theta}' - \frac{1}{\tau_2}x_8 \tag{71}$$

The relationship for $\dot{\theta}'$ can be derived from Equation (15)

$$\dot{\theta}' = \tau_1\ddot{\theta}_f' + \dot{\theta}_f' \tag{72}$$

The expression for $\dot{x}_9$ is then described as

$$\dot{x}_9 = \tau_1\ddot{\theta}_f' - \dot{\theta}' = -\dot{\theta}_f' \tag{73}$$

and this yields

$$x_9 = \tau_1\dot{\theta}_f' - \theta' \tag{74}$$

Note that $\tau_1\dot{\theta}_f'$ is the boresight error from Equation (11). By putting the expression for $\dot{x}_9$ in Equation (74) and rearranging terms

$$\dot{x}_9 = -\frac{1}{\tau_1}(x_9 + \theta') \tag{75}$$

but from Figure 5 it can be seen that

$$\theta' = x_{12} + (1 + k_r)x_1x_3 - k_rx_4 \tag{76}$$

putting this into Equation (75) yields

$$\dot{x}_9 = \frac{1}{\tau_1}\{k_rx_4 - x_9 - x_{12} - (1 + k_r)x_1x_3\} \tag{77}$$

Putting this expression $\dot{x}_9$, with a minus sign, back into Equation (71) gives

$$\dot{x}_8 = -\frac{nV_c}{g\tau_1\tau_2}\{k_r x_4 - x_9 - x_{12} - (1 + k_r)x_1 x_3\} - \frac{1}{\tau_2} x_8 \tag{78}$$

The equation for $\dot{x}_{10}$ is

$$\dot{x}_{10} = \frac{g x_{11}}{V_t} \tag{79}$$

where x_{11}, a_t, is described as

$$x_{11} = C_{TLOS}^{TB}\, a_T^{TLOS} \tag{80}$$

where a_T^{TLOS} is the total tracker acceleration vector in the TLOS frame. The expression for $\dot{x}_{11}$ is

$$\dot{x}_{11} = C_{TLOS}^{TB}\, C_I^{TLOS}\, \dot{a}_T^I \tag{81}$$

where $\dot{a}_T^I$ can be expressed in a closed form equation. In the actual simulation, the rectangular integration scheme made it unnecessary to use $\dot{a}_T^I$. The expression for $\dot{x}_{12}$ is seen to be

$$\dot{x}_{12} = -\lambda x_{12} + \sqrt{2\lambda}\,\eta_1 + \eta_2 \tag{82}$$

where η_1 and η_2 are white Gaussian noises of appropriate strengths. Finally, from Figure 11, $\dot{x}_{13}$ and $\dot{x}_{14}$ are seen to be

$$\dot{x}_{13} = (\cos\theta_\ell \cdot a_D + \sin\theta_\ell \cdot x_6)g \tag{83}$$

$$\dot{x}_{14} = (\sin\theta_{\ell} a_D - \cos\theta_{\ell} \cdot x_6)g \tag{84}$$

Recall that a_D is the acceleration due to drag, computed in Equation (44).

A total of ten truth model parameters are to be estimated by the two Extended Kalman Filters. These truth model quantities are defined as "true states" and are to be compared to the filter state estimates. The first five true states are common to both filters

$$TS(1) = \theta \tag{85-a}$$

$$TS(2) = V_{mx}^{I} \tag{85-b}$$

$$TS(3) = V_{my}^{I} \tag{85-c}$$

$$TS(4) = R \tag{85-d}$$

$$TS(5) = \dot{R} \tag{85-e}$$

For the "Landau Filter" the last three states are

$$TS(6) = a_{mx}^{TLOS} \tag{86-a}$$

$$TS(7) = a_{mx}^{I} \tag{86-b}$$

$$TS(8) = a_{my}^{I} \tag{86-c}$$

For the "Landau Filter", Chapter Three shows that, in the planar case, not all eight of the states are independent. Further discussion of this is deferred to Chapter Three. For the "Unknown Parameter Filter", the final two states are

$$TS(6) = n \tag{87-a}$$

$$TS(7) = m/S \tag{87-b}$$

where n is the proportional navigation constant and m/S is the ratio of the missile mass to the reference surface area.

Summary. Chapter Two has developed the real world model of the engagement scenario's systems, parameters, and noises. By utilizing a piecewise continuous assumption, it has been possible to describe a complex, nonlinear system and yet simulate the system using a simple rectangular integration scheme. The modelling emphasis has been on the missile system and its associated parameters and coefficients.

The system equations were developed throughout the chapter. The truth model flowchart and computer listing are in Appendix C and are labeled subroutine TRAJ. Finally, the "true states" were generated for the Monte-Carlo analysis of the two filters.

III. Extended Kalman Filter

Introduction

This section presents the Extended Kalman Filter equations and develops the "Landau" and "Unknown Parameter" Filters. As will be seen, both of these filters are highly nonlinear. Therefore, the conventional Kalman Filter is not an appropriate estimator for this particular system. The Extended Kalman Filter (EKF), however, is a good initial choice as a means of solving the nonlinear system estimation and prediction problem. The EKF gain and covariance propagation equations have the same form as the Kalman Filter equations, but are linearized about the current state estimates. The linearization is a first term approximation to a Taylor series expansion about the current state estimate. Higher order, more exact approximations to the optimal nonlinear filter can be achieved by using more terms of the Taylor series expansion for the nonlinearities, and by deriving recursive relations for the higher order moments of the state vector, $\underline{x}$ (Ref 4:184, Ref 9:252). When compared to the EKF, however, the higher order filters are both more complex and more costly in terms of computer implementation. The higher order filters are not considered in this thesis. In general, when nonlinearities are significant, neglecting the higher order terms results in biased estimates.

It is important, therefore, to remember that the system model within the EKF is an approximation to the actual nonlinear system. The fact that it is an approximation implies that the filter's predictions and estimates will be limited by the "correctness" of the system modelling assumptions. More specifically, large differences between an assumed nominal trajectory and the true trajectory can result in loss of tracking accuracy, or loss of track completely. In general, the greater the system nonlinearities, the poorer the EKF performance. Not only might the degree of nonlinearity

prove the EKF to be an inadequate filter form, it also creates a filter design disadvantage for the EKF - and higher order filters. Unlike the conventional Kalman filter, the EKF gains and estimation error covariance matrices depend on the time history of $\hat{\underline{x}}(t)$, i.e. estimation accuracy is trajectory dependent (Ref 4:187). This requires that the actual filter performance be verified by a Monte-Carlo simulation. In general, this form of analysis is both more time consuming and more costly in terms of computer usage than is a covariance analysis. Despite these factors, the EKF has been used, and has been found to yield accurate estimates, in a number of important applications. An example which includes parameter estimation can be found in (Ref 9) (Ref 9:189). The derivation of the EKF equations is not included in this thesis. Instead the equations are stated, and (Refs 4 and 9) are given as sources for the filter derivation.

Equations and Explanation

The equations for the EKF can be divided into three categories: system equations upon which the filter is based, propagation equations, and update equations. The system equations can be expressed in the form

$$\dot{\underline{x}}(t) = \underline{f}[\underline{x}(t),\underline{u}(t),t] + G(t)\underline{w}(t) \tag{88}$$

where $\underline{w}(t)$ is a zero mean white Gaussian noise process of strength $\underline{Q}(t)$, i.e.

$$E\{\underline{w}(t)\underline{w}^T(t+\tau)\} = Q(t)\delta(\tau) \tag{89}$$

and where $\underline{x}(t_o)$ is modelled as a Gaussian random variable with mean $\hat{\underline{x}}_o$ and covariance $\underline{P}_o$. In Equation (88), $\underline{u}(t)$ represents a deterministic forcing

function. Note that the dynamic driving noise is assumed to enter in a linear additive fashion (Ref 9:179). In addition, the discrete measurement equations can be expressed as

$$\underline{z}(t_i) = \underline{h}[\underline{x}(t_i), t_i] + \underline{v}(t_i) \tag{90}$$

where $\underline{v}(t_i)$ is a white Gaussian noise sequence of zero mean and covariance kernel

$$E\{\underline{v}(t_i)\underline{v}^T(t_j)\} = R(t_i)\delta_{ij} \tag{91}$$

$\underline{R}(t_i)$ represents the strength of the $\underline{v}(t_i)$ noise process. It is assumed that $\underline{w}(t_i)$ and $\underline{v}(t_i)$ are independent of each other. The specific system equations for the "Landau" and "Unknown Parameter" Filters are developed later in this chapter.

The filter propagation equations are

$$\dot{\hat{\underline{x}}}(t/t_i) = \underline{f}[\hat{\underline{x}}(t/t_i), \underline{u}(t), t]; \; \hat{\underline{x}}(t_i/t_i) = \hat{\underline{x}}(t_i^+) \tag{92}$$

where the notation $\hat{\underline{x}}(t/t_i)$ means the optimal estimate of the state, $\underline{x}$, at time, t, given the updated estimates up to and including time t_i. In addition, (the covariance is propagated approximately by)

$$\dot{\underline{P}}(t/t_i) = \underline{F}[t;\hat{\underline{x}}^+(t_i)]\underline{P}(t/t_i) + \underline{P}(t/t_i)\underline{F}[t;\hat{\underline{x}}^+(t_i)]^T$$

$$+ \underline{G}(t)\underline{Q}(t)\underline{G}(t)^T; \underline{P}(t_i/t_i) = \underline{P}(t_i^+) \tag{93}$$

where $\underline{P}$ is the filter covariance matrix and $\underline{F}$ is the matrix of partial derivatives of $\underline{f}$ with respect to $\underline{x}$, evaluated along the nominal trajectory

$$\underline{F}[t;\hat{\underline{x}}^{+}(t_i)] = \left.\frac{\partial \underline{f}[\underline{x}(t),\underline{u}(t),t]}{\partial \underline{x}}\right|_{\underline{x}(t) = \hat{\underline{x}}_n^{+}(t_i)} \tag{94}$$

The notation $[t;\hat{\underline{x}}^{+}(t_i)]$ implies that the linearized F matrix is evaluated at the values of the states just after a measurement update (Ref 7:19,20). This is an approximation to the actual EKF equation in which the value elements are evaluated as a function of $\hat{\underline{x}}(t/t_i)$ for all $t\varepsilon[t_i,t_{i+1})$. Since the measurement update period for this problem is one-fiftieth of a second, this should be a good approximation (i.e. the value of the states remain relatively unchanged over the period). This assumption helps to reduce computer simulation time. Note that the dependency of the covariance propagation on the current state estimate can be seen in Equation (93).

The requirement for the current state estimate is also evident in the covariance update equation. The measurement update is given by

$$\underline{K}(t_i) = \underline{P}(t_i^{-})\underline{H}^{T}[t;\hat{\underline{x}}(t_i^{-})]\{\underline{H}[t_i;\hat{\underline{x}}(t_i^{-})]\underline{P}(t_i^{-})\underline{H}^{T}[t_i;\hat{\underline{x}}(t_i^{-})] + \underline{R}(t_i)\}^{-1} \tag{95}$$

$$\hat{\underline{x}}(t_i^{+}) = \hat{\underline{x}}(t_i^{-}) + \underline{K}(t_i)[\underline{\zeta}_i - \underline{h}[\underline{x}(t_i^{-}),t_i]] \tag{96}$$

$$\underline{P}(t_i^{+}) = \underline{P}(t_i^{-}) - \underline{K}(t_i)\underline{H}[t;\hat{\underline{x}}(t_i^{-})]\underline{P}(t_i^{-}) \tag{97}$$

where

$\underline{K}(t_i)$ = Extended Kalman gain matrix

$\underline{R}(t_i)$ = measurement noise covariance matrix (see Equation (91))

$\underline{H}[t;\hat{\underline{x}}(t_i^{-})]$ = matrix of partial derivatives of $\underline{h}$ with respect to x

$\underline{\zeta}_i$ = values of actual measurements taken at time t_i.

The notation t_i^- implies the value of the quantity at the instant prior to the update at time t_i, and t_i^+ is the value of the quantity just after the update. The notation used thus far, except as noted for the F matrix, is developed and used in (Ref 9). The system equations for this thesis are developed in the next two sections. Together with the propagation and update equations listed in this section and evaluated at the proper values, the Extended Kalman Filters are developed. The m-vector function, $\underline{h}$, for both filters is linear. The F matrices of partial derivatives for the two filters are found in Appendix A.

"Landau Filter"

In his paper, "Radar Tracking of Airborne Targets", (Ref 6), M. I. Landau deals with several of the problems that are basic to this study. Despite the fact that Landau considers the single target track environment, the resulting algorithm should be applicable to the multiple target environment. The filter system equations developed in Landau's paper are combined and modified, in accordance with the assumptions made in Chapter Two, to form the system equations for an Extended Kalman Filter. This filter is termed the "Landau Filter" of L-Filter, and serves as a benchmark with which to compare the "Unknown Parameter Filter".

The L-Filter estimates the missile attitude direction cosine (Λ), velocity (V_m), and acceleration (a_m) vectors in stabilized/"inertial" (I) coordinates. The equations defining the missile kinematics are

$$\dot{\Lambda}_\alpha = -\frac{\dot{R}}{R}\Lambda_\alpha + \frac{(V^I_{m\alpha} - V^I_{T\alpha})}{R} \tag{98-a}$$

$$\dot{V}^I_{m\alpha} = a^I_{m\alpha} \tag{98-b}$$

$$\dot{a}^{I}_{m\alpha} = -\frac{1}{\tau_\alpha} a^{I}_{m\alpha} + w_\alpha \qquad (98\text{-c})$$

where

α = a subscript that assumes the values of x for north, y for east, and z for down

Λ_α = direction cosine in the α direction (e.g., $R_N = \Lambda_N R$, R = range from missile to tracker)

τ_α = correlation time of acceleration process

w_α = white Gaussian driving noise for the acceleration process

$\dot{R}$ = range rate along the instantaneous LOS between missile and tracker

$V^{I}_{T\alpha}$ = α-component of tracker/ownship velocity coordinatized in the I frame.

The basic observation for this configuration is

$$\Lambda_{\alpha obs} = \Lambda_\alpha + \eta_{\Lambda\alpha} \qquad (99)$$

where $\eta_{\Lambda\alpha}$ is the white Gaussian noise corrupting the Λ_α observation. Notice that the acceleration for this system of equations is modelled by the first order Markov Process of Equation (98-c). The range and range rate terms in Equation (98-a) are supplied as "aiding terms" from a separate range and range rate tracking algorithm. Landau uses this technique to maintain the linear properties of the tracking algorithm for simpler mechanization. The equations for the range (R) and range rate ($\dot{R}$) kinematics are expressed in the LOS coordinates as

$$\dot{R} = V_c \qquad (100\text{-a})$$

$$\dot{V}_c = \omega^2 R + a_{mx}^{TLOS} - a_{Tx}^{TLOS} \tag{100-b}$$

$$\dot{a}_{mx}^{TLOS} = -\frac{1}{\tau_3} a_{mx}^{TLOS} + w_3 \tag{100-c}$$

where

$$\omega^2 = \omega_{LSy}^2 + \omega_{LSz}^2 \tag{101}$$

and

ω_{LSy}, ω_{LSz} = missile to tracker line-of-sight rates about the y and z LOS coordinates

a_{mx}^{TLOS} = missile acceleration along the LOS coordinatized in the TLOS frame

a^{TLOS} = tracker acceleration along the line-of-sight coordinatized in the TLOS frame

τ_2 = correlation time of missile acceleration process

w_3 = white Gaussian driving noise for the acceleration process

V_c = relative closing velocity between missile and tracker along the instantaneous LOS.

Again the acceleration has been modelled as a first order Markov Process. Note also that Equations (98-a) and (100-b) contain tracker kinematic terms. Assuming perfect measurements of these terms (reference Chapter Two), $V_{T\alpha}^I$ and a_{Tx}^{TLOS} are treated as deterministic forcing functions. The observations for the range and range rate configuration are

$$R_{obs} = R + \eta_R \tag{102-a}$$

$$\dot{R}_{obs} = \dot{R} + \eta_{\dot{R}} \tag{102-b}$$

where η_R and $\eta_{\dot{R}}$ are white Gaussian noise terms (reference Equation (91)) that corrupt the range and range rate measurements respectively. Landau's development is for a three dimensional problem. For the planar case, and by combining the two separate sets of equations, the following system of equations result

$$-\dot{\theta}_T \sin\theta_T = -\frac{\dot{R}}{R}\cos\theta_T + \frac{V^I_{mx} - V^I_{Tx}}{R} \tag{103-a}$$

$$\dot{\theta}_T \cos\theta_T = -\frac{\dot{R}}{R}\sin\theta_T + \frac{V^I_{my} - V^I_{Ty}}{R} \tag{103-b}$$

$$\dot{V}^I_{mx} = a^I_{mx} \tag{103-c}$$

$$\dot{V}^I_{my} = a^I_{my} \tag{103-d}$$

$$\dot{R} = V_c \tag{103-e}$$

$$\dot{V}_c = \dot{\theta}^2_T R + a^{TLOS}_{mx} - a^{TLOS}_{Tx} \tag{103-f}$$

$$\dot{a}^{TLOS}_{mx} = -\frac{1}{\tau_3} a^{TLOS}_{mx} + w_3 \tag{103-g}$$

$$\dot{a}^I_{mx} = -\frac{1}{\tau_x} a^I_{mx} + w_x \tag{103-h}$$

$$\dot{a}^I_{my} = -\frac{1}{\tau_y} a^I_{my} + w_y \tag{103-i}$$

where θ_T is the LOS angle as measured by the tracker (reference Figure 12). It is apparent that the $\sin\theta_T$ and $\cos\theta_T$ are not independent. Therefore, θ_T is chosen as the state variable to avoid observability problems. In addition θ_T can be generated as a solution to

$$\dot{\theta}_T = \frac{\dot{R}}{R}\cot\theta_T + \frac{V^I_{Tx} - V^I_{mx}}{R}\csc\theta_T \qquad (104\text{-a})$$

or

$$\dot{\theta}_T = -\frac{\dot{R}}{R}\tan\theta_T + \frac{V^I_{my} - V^I_{Ty}}{R}\sec\theta_T \qquad (104\text{-b})$$

Either of the two forms is correct. To prevent $\dot{\theta}_T$ from growing unbounded, Equation (104-a) is used when $\sin\theta_T \geq .707$ and Equation (104-b) is used when $\sin\theta_T < .707$. Furthermore, it can be seen from Equations (104-a) and (104-b) that V^I_{mx} and V^I_{my} are not independent. By setting the two expressions for $\dot{\theta}_T$ equal to each other and solving for V^I_{my}

$$V^I_{my} = \frac{(\dot{R}\cos^2\theta_T + V^I_{Tx}\cos\theta_T - V^I_{mx}\cos\theta_T)}{\sin\theta_T} + \dot{R}\sin\theta_T + V^I_{Ty} \qquad (105\text{-a})$$

Equation (105-a) is used when the $\sin\theta_T \geq .707$ and

$$V^I_{mx} = \frac{(\dot{R}\sin^2\theta_T + V^I_{Ty}\sin\theta_T - V^I_{my}\sin\theta_T)}{\cos\theta_T} + \dot{R}\cos\theta_T + V^I_{Tx} \qquad (105\text{-b})$$

is used for $\sin\theta_T < .707$. The use of either Equation (105-a) or (105-b) eliminates the requirement to estimate a^I_{my} or a^I_{mx} respectively. By setting

$$x_1 = \theta_T \qquad x_5 = \dot{R}$$
$$x_2 = V_{mx} \qquad x_6 = a^{TLOS}_{mx}$$
$$x_3 = V_{my} \qquad x_7 = a^I_{mx}$$
$$x_4 = R \qquad x_8 = a^I_{my} \qquad (106\text{-a-h})$$

and considering the case of $\sin\theta_T \geq .707$, the system equations for the L-Filter are

$$\dot{x}_1(t) = f_1[\underline{x}(t),\underline{u}(t),t] = \frac{x_5(t)}{x_4(t)}\cot x_1(t) + \frac{V_{Tx}^I - x_2(t)}{x_4(t)}\csc x_1(t) \quad (107)$$

$$\dot{x}_2(t) = f_2[\underline{x}(t),\underline{u}(t),t] = x_7(t) \quad (108)$$

$$\dot{x}_3(t) = f_3[\underline{x}(t),\underline{u}(t),t] = 0 \quad (V_{my} \text{ is not estimated for } \sin\theta_T \geq .707 \text{ - reference Equation (105-a))} \quad (109)$$

$$\dot{x}_4(t) = f_4[\underline{x}(t),\underline{u}(t),t] = x_5(t) \quad (110)$$

$$\dot{x}_5(t) = f_5[\underline{x}(t),\underline{u}(t),t] = (\dot{x}_1)^2 x_4(t) + x_6(t) - a_{Tx}^{TLOS} \quad (111)$$

$$\dot{x}_6(t) = f_6[\underline{x}(t),\underline{u}(t),t] + w_3(t) = \frac{1}{\tau_3} x_6(t) + w_3 \quad (112)$$

$$\dot{x}_7(t) = f_7[\underline{x}(t),\underline{u}(t),t] + w_x(t) = \frac{1}{\tau_x} x_7 + w_x \quad (113)$$

$$\dot{x}_8(t) = f_8[\underline{x}(t),\underline{u}(t),t] = 0 \quad (a_{my}^I \text{ is not estimated for } \sin\theta_T \geq .707) \quad (114)$$

When $\sin\theta_T < .707$

$$\dot{x}_2(t) = f_2[\underline{x}(t),\underline{u}(t),t] = 0 \quad (115)$$

$$\dot{x}_3(t) = f_3[\underline{x}(t),\underline{u}(t),t] = x_8(t) \quad (116)$$

$$\dot{x}_7(t) = f_7[\underline{x}(t),\underline{u}(t),t] = 0 \quad (117)$$

$$\dot{x}_8(t) = f_8[\underline{x}(t),\underline{u}(t),t] + w_y(t) = \frac{1}{\tau_y} x_8 + w_y \quad (118)$$

All other equations remain unchanged. In the equation for $\dot{x}_5(t)$ the expression for $\dot{x}_1(t)$ is considered a parameter. This approximation is made to reduce the number of partials required in the $\underline{F}$ matrix. It assumes that $\dot{\theta}_T$ remains constant over the propagation period. With the frequency of updates, .02 seconds, this should be a good approximation. It is a better approximation at long and intermediate ranges to the target than at short range. All of the original set of equations from Landau's paper are easily understood with the possible exceptions of Equations (107) and (111). These two equations are developed in Appendix A. In addition to these and the $\underline{F}$ matrix development, the values for the $\underline{G}$, $\underline{H}$, $\underline{Q}$, $\underline{R}$, $\hat{\underline{x}}_o$, and $\underline{P}_o$ matrices are given in Appendix A.

The L-Filter system equations are clearly a set of coupled, non-linear differential equations. The "forcing functions" are the driving noises of the first order Markov Processes for missile acceleration and the tracker's velocity and acceleration. While this filter adequately models the missile kinematics, it does not utilize some important knowledge that is available from the total system model.

"Unknown Parameter Filter"

The Unknown Parameter Filter, UP-Filter, is so named since it incorporates the dynamics and coefficients represented in Equations (2) and (3). Specifically, the three unknown parameters, C_{DI}, C_{DO}, and n, must be estimated or modelled to describe the acceleration of the missile as the sum of the acceleration due to drag and acceleration due to lift. As discussed previously, the inertial acceleration can be described by resolving the a_ℓ and a_D vectors of Figure 13 into the x^I and y^I axes. Equations (46-a) and (46-b) are repeated for convenience.

$$a^I_{mx} = +\cos\gamma_m \cdot a_\ell - \sin\gamma_m \cdot a_D \tag{46-a}$$

$$a^I_{my} = -\sin\gamma_m \cdot a_\ell - \cos\gamma_m \cdot a_D \tag{46-b}$$

Equations (46-a) and (46-b) are the necessary expressions required to describe the rate of change of velocity in the inertial coordinate frame. To solve for these quantities, expressions are necessary for a_D and a_ℓ. Equations (44) and (2) are repeated here for these quantities

$$a_D = \frac{1}{2}\rho S C_D V_m^2/m \tag{44}$$

$$a_\ell = nV_c\dot{\theta} \tag{2}$$

Unfortunately, the values of n, m, S, C_L, and C_D are not known apriori by the tracking filter. What information that is available would be known from intelligence information on possible threats. At best this form of information can only be considered relatively accurate. Therefore, to implement a tracker mechanization utilizing the dynamics of the missile, as described by Equations (46-a) and (46-b), estimates of the parameters must be made. As will be seen in the subsequent development, both the parameters themselves and the way they enter the state equations introduce further nonlinearities into the system.

As stated in Chapter Two, the proportional navigation constant, n, is considered to be constant unless the user commands a change in flight. Therefore, the navigation constant might initially be modelled as a random bias

$$\dot{n}(t) = 0 \tag{119}$$

where $n(t_o)$ is a Gaussian random variable of mean, m_o, and variance P_o. Mathematically, the bias model tells the filter that the constant does not change in time, but its value is not known apriori. The filter will determine its estimate of the constant on early measurements and disregard measurements that come later (i.e. the Kalman gain goes to zero on the bias channel). Since the value of the navigation constant can be changed, an inability to re-estimate its new value would be highly undesirable. Therefore, pseudonoise is added to the navigation constant channel to ensure the gain does not go to zero (Ref 9:159). In addition, and based upon the discussion in the Guidance Section of Chapter Two, a navigation constant limiter with a range of three to six is implemented in the filter.

The coefficient of zero-lift drag can be described as

$$C_{DO} = \frac{2\sqrt{A}}{\sqrt{V_m}} \tag{120}$$

where the expression, V_m/A, has been substituted for the Mach, M, in Equation (28). A is the speed of sound at the appropriate altitude. In addition, to give an expression for C_{DI}, Equation (25) is repeated

$$C_{DI} = C_L^2/C_{N\alpha}$$

From a curve fit to a generic missile, $C_{L\alpha}$ can be expressed as

$$C_{N\alpha} = \frac{K_{CL\alpha}}{M} \approx \frac{A}{V_m} \tag{121}$$

where the constant, $K_{CL\alpha}$, is approximately equal to one and the substitution of V_m/A was made for the Mach. An expression for C_L is found in

Equation (27)

$$C_L = \frac{2ma_\ell}{\rho S V_m^2}$$

Substituting Equations (27) and (121) into Equation (25) yields

$$C_{DI} = \left(\frac{4m^2}{\rho^2 S^2 A}\right) \frac{a_\ell^2}{V_m^3} \tag{122}$$

This expression for C_{DI} is intuitively pleasing. As expected, the induced drag, vis-a-vis C_{DI}, of the missile increases as lateral acceleration increases. The V_m^3 tern in the denominator accounts for the fact that greater lateral acceleration is required for control at lower velocities. Substituting the expressions for C_{DI} and C_{DO} into the expression for C_D in Equation (44) yields

$$a_D = \frac{1}{2} \rho S/m \left[\frac{2A^{1/2}}{V_m^{1/2}} + \frac{4m^2}{\rho^2 S^2 A}\left(\frac{a_\ell^2}{V_m^3}\right)\right] V_m^2 \tag{123-a}$$

By substituting in the expression for a_ℓ from Equation (2) and expanding, this becomes

$$a_D = \rho A^{1/2} \left(\frac{S}{m}\right) V_m^{3/2} + \left(\frac{2}{\rho A}\right)\left(\frac{m}{S}\right) \frac{n^2 V_c^2 \dot{\theta}_T^2}{V_m} \tag{123-b}$$

In Equation (123-b), the factors, $\rho A^{1/2}$ and $2/\rho A$ are assumed to be deterministic constants. This is a valid assumption for the constant altitude case. Even with removal of the constant altitude assumption, ρ and A will vary little during a typical air-to-air engagement. If desired, and if warranted by performance enhancement, a more sophisticated

model could include the variation in these parameters. The quantities $(\frac{m}{S})$, n, and V_c in Equation (123-b) are defined as states while V_m and $\dot{\theta}_T$ are defined as parameters. The rationale for the decision about the latter two quantities follows the discussion of $\dot{\theta}_T$ in the L-Filter development. The ratio of $\frac{m}{S}$ is modelled as a random bias

$$(\dot{\frac{m}{S}})(t) = w(t) \tag{124}$$

Conceptually, this makes sense for a non-thrusting missile. It is assumed that $[\frac{m}{S}](t_o)$ is a Gaussian random variable of mean m_{o1} and variance P_{o1}. Again, a pseudonoise, w(t), is added to the model to ensure a non-zero gain on the $[\frac{m}{S}]$ channel. In the extreme case, this parameter could detect a change in target status (i.e. an abrupt change in this parameter could be indicative of target damage or destruction). The first five states of the UP-Filter are identical to the corresponding states in the L-Filter. Three of the system equations, however, take a radically different form. States two and three, V^I_{mx} and V^I_{my}, now vary according to

$$\dot{V}^I_{mx} = -a_D \sin\gamma_m + a_\ell \cos\gamma_m \tag{125}$$

$$\dot{V}^I_{my} = -a_D \cos\gamma_m - a_\ell \sin\gamma_m \tag{126}$$

Substituting in Equations (123-b) and (2) for a_D and a_ℓ respectively yields

$$\dot{V}^I_{mx} = -[\rho A^{1/2} V_m^{3/2} (\frac{S}{m}) + (\frac{2}{\rho A})(\frac{m}{S}) \frac{n^2 V_c^2 \dot{\theta}^2}{V_m}] \sin\gamma_m + n V_c \dot{\theta} \cos\gamma_m \tag{127}$$

$$\dot{V}_{my}^{I} = -[\rho A^{1/2}(\frac{S}{m})V_m^{3/2} + (\frac{2}{\rho A})(\frac{m}{S})\frac{n^2V_c^2\dot{\theta}^2}{V_m}]\cos\gamma_m - nV_c\dot{\theta}\sin\gamma_m \quad (128)$$

State five, the range rate ($\dot{R}$), requires an expression for the missile line-of-sight acceleration, a_{mx}^{TLOS}. This can be obtained by taking the x-component of

$$\underline{a}_m^{TLOS} = C_I^{TLOS}\,\underline{a}_m^I \quad (129)$$

where $\underline{a}_m^{TLOS}$ and $\underline{a}_m^I$ are the expressions for the missile acceleration expressed in the TLOS and I frames respectively. Substituting Equations (42), (127), and (128) into this expression yields

$$\underline{a}_m^{TLOS} = \begin{bmatrix} a_{mx}^I\cos\theta_T + a_{my}^I\sin\theta_T \\ -a_{mx}^I\sin\theta_T + a_{my}^I\cos\theta_T \\ 0 \end{bmatrix} \quad (130)$$

and thus

$$a_{mx}^{TLOS} = a_{mx}^I\cos\theta_T + a_{my}^I\sin\theta_T \quad (131)$$

Substituting Equation (131) into the fifth system equation of the L-Filter, and using Equations (127) and (128) for a_{mx}^I and a_{my}^I respectively yields

$$\dot{V}_c = \dot{\theta}_T^2 R + \{[-\rho A^{1/2}(\frac{S}{m})V_m^{3/2}\sin\gamma_m - (\frac{2}{\rho A})(\frac{m}{S})\frac{n^2V_c^2\dot{\theta}^2}{V_m}\sin\gamma_m + nV_c\dot{\theta}\cos\gamma_m]\cos\theta_T + [-\rho A^{1/2}(\frac{S}{m})V_m^{3/2}\cos\gamma_m - (\frac{2}{\rho A})(\frac{m}{S})\frac{n^2V_c^2\dot{\theta}^2}{V_m}\cos\gamma_m - nV_c\dot{\theta}\sin\gamma_m]\sin\theta_T\} - a_{Tx}^{TLOS} \quad (132)$$

as the fifth system equation for the UP-Filter. The angle, $\dot{\theta}$, and range rate, $\dot{R}$, equations are the same as described in the L-Filter. The final two states of the filter are the proportional navigation constant n and the (m/S) coefficient. The system equations of the UP-Filter are summarized by (for $\sin\theta_T \geq .707$)

$$\dot{x}_1(t) = f_1[\underline{x}(t),\underline{u}(t),t] = \frac{x_5(t)}{x_4(t)} \cot x_1(t) + \frac{V_{Tx} - x_2(t)}{x_4(t)} \csc x_1(t) \tag{133}$$

$$\dot{x}_2(t) = f_2[\underline{x}(t),\underline{u}(t),t] = -(\rho A^{1/2} V_m^{3/2} \sin\gamma_m)(\frac{1}{x_7}) - \frac{2\dot{\theta}^2 \sin\gamma_m}{\rho A V_m} (x_7 x_6^2 x_5^2) + (x_6 x_5)\dot{\theta}\cos\gamma_m \tag{134}$$

$$\dot{x}_3(t) = f_3[\underline{x}(t),\underline{u}(t),t] = -(\rho A^{1/2} V_m^{3/2} \cos\gamma_m)(\frac{1}{x_7}) - \frac{2\dot{\theta}^2 \cos\gamma_m}{\rho A V_m} (x_7 x_6^2 x_5^2) - (x_6 x_5)\dot{\theta}\sin\gamma_m \tag{135}$$

$$\dot{x}_4(t) = f_4[\underline{x}(t),\underline{u}(t),t] = x_5(t) \tag{136}$$

$$\dot{x}_5(t) = f_5[\underline{x}(t),\underline{u}(t),t] = \dot{\theta}_T^2 x_4 + \{[-\rho A^{1/2} V_m^{3/2} \sin\gamma_m (\frac{1}{x_7}) - \frac{2\dot{\theta}^2 \sin\gamma_m}{\rho A V_m} (x_7 x_5^2 x_6^2) + \dot{\theta}\cos\gamma_m (x_5 x_6)]\cos x_1 + [-\rho A^{1/2} V_m^{3/2} \cos\gamma_m (\frac{1}{x_7}) - \frac{2\dot{\theta}^2 \cos\gamma_m}{\rho A V_m} (x_7 x_5^2 x_6^2) - \dot{\theta}\sin\gamma_m (x_5 x_6)]\sin x_1\} - a_{Tx}^{TLOS} \tag{137}$$

$$\dot{x}_6(t) = f_6[\underline{x}(t),\underline{u}(t),t] + w_6(t) = w_6(t) \tag{138}$$

$$\dot{x}_7(t) = f_7[\underline{x}(t),\underline{u}(t),t] + w_7(t) = w_7(t) \tag{139}$$

where

$x_1 = \theta$ $\qquad$ $x_5 = V_c$

$x_2 = V^I_{mx}$ $\qquad$ $x_6 = n$

$x_3 = V^I_{my}$ $\qquad$ $x_7 = m/S$

$x_4 = R$

Note that

$$\dot{\theta}_T = \dot{x}_1;\ V_m = (x_2^2 + x_3^2)^{1/2};\ \gamma_m = \tan^{-1}(\frac{x_2}{x_3}) \tag{140}$$

These quantities are treated as constants when the elements of the F-matrix are calculated. This approach avoids further complexity in both the system equations and the F-matrix. Unlike the L-Filter, the two velocities, V^I_{mx} and V^I_{my}, can both be estimated - even in the planar case. The structure of the system equations allows for this, despite the fact the two velocities are not independent. In addition, the same switching mechanism is utilized for the $\dot{\theta}_T$ equation. Of the three parameters, n, C_{DO}, C_{DI}, only the first is estimated directly by the UP-Filter. It is obvious, however, that the quantities required to calculate C_{DI} and C_{DO} are available in the filter. Consideration had been given to modelling the parameters C_{DI} and C_{DO} as random walks. However, this approach neglects the dominant effects of the lateral acceleration and missile velocity on these two parameters and thus on the acceleration of the missile. The UP-Filter embodies a highly nonlinear, coupled set of differential equations. The primary assumptions made in the filter

development are: (1) the geometry of the forces acting on the missile, (2) the defining relations, Equations (122) and (28), for the parameters C_{DI} and C_{DO}, (3) the zero order model for the missile (i.e. instantaneous time response) - also true in L-Filter, and (4) $\dot{\theta}_T$, γ_m, and V_m being treated as constant quantities over a propagation period. In addition to the F matrix, the $\underline{G}$, $\underline{Q}$, $\underline{R}$, $\underline{H}$, $\hat{\underline{x}}_o$, and $\underline{P}_o$ matrices for this filter are listed in Appendix A.

Having developed the system equations for the two Extended Kalman Filters, and provided with a nominal trajectory, it is now possible to evaluate the prediction and estimation capability of the two filters. Chapter Four discusses the analysis approach and "tuning" techniques used in the evaluation.

IV. Discussion

Introduction

The primary objective of this study is to determine the feasibility of the modelling approach taken in the Unknown Parameter Filter. To accomplish this objective, it is necessary to establish the ability of the filter to track an incoming missile and to establish some degree of confidence in its ability to recover from initial state estimate errors. To determine the ability of the filter to track the missile, its performance is compared against that of a more conventional system model (i.e. the Landau Filter). The analysis of the two Extended Kalman Filters is carried out entirely as a Monte Carlo computer simulation. The truth model provides nominal measurements to the filters (reference Chapter One). The filters use the measurements to generate estimates of the system states. The "true states" are compared to the estimated states and the sample statistics generated to describe the resulting error process.

In this chapter, the choice of the trajectory profiles and initial conditions are discussed. In addition, the criteria used to analyze and compare the two filters are introduced. Finally, the filter tuning philosophy is presented. Prior to discussing the trajectory profiles, it is important to recall that the emphasis in this study is not to generate tuned filters, but rather to determine the feasibility of the UP-Filter, and to conduct a general comparison of its performance against that of the L-Filter. Therefore, as will be seen subsequently, the major analysis effort is involved with the Unknown Parameter Filter.

Nominal Trajectories

The basis for the comparison and recovery analysis are the two nominal trajectories generated by the truth model. The choice of

trajectories has been limited to the region in which the sine of the LOS angle, θ_T, is greater than .707 (reference Figure 12). This eliminates any problems associated with switching and reduces the tuning requirements in the study. In addition, the truth model allows for missile acceleration up to, but not exceeding 15 g's. The two trajectories chosen do not exercise the full dynamic range of the missile. Due to time constraints, a full study of "worst case" trajectories is not feasible. Therefore, as depicted in Figure 16, the two trajectories chosen represent a low-g and moderate-g trajectory. Table II lists those parameters that determine the character of the two respective trajectories [reference Figure 12 and Equations (48-a) and (48-b)].

Table II
Time (t_o) Parameter Values

Parameter	Low-g Trajectory	High-g Trajectory
R, range	10,000 ft	10,000 ft
V_m, missile velocity	3,051 ft/sec	3,051 ft/sec
θ_ℓ, missile lead angle	3.5 deg	0 deg
γ_m, missile inertial heading	20 deg	30 deg
γ_T, tracker inertial heading	32 deg	0 deg
K_1	500	0
K_2	2864.51	0
K_3 tracker dynamics	800	920
K_4	0	0
ω	.17455	.17455
n, pro-nav constant	4.0	4.0

While this approach does not fully exercise the algorithms developed, it is consistent with the objective of establishing concept feasibility.

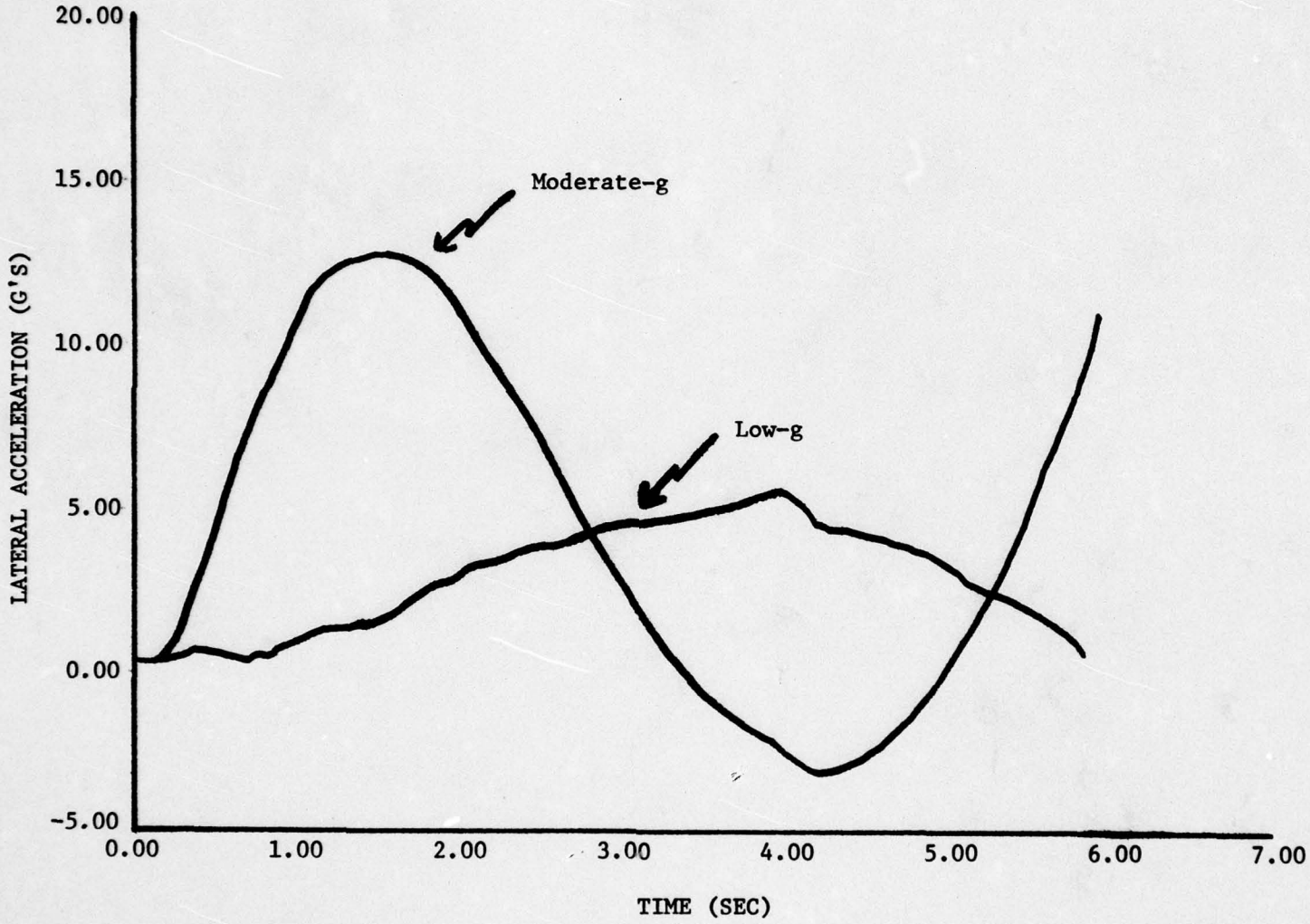


Figure 16. Nominal Trajectory Profiles

As will be seen in Chapter Five, the recovery analysis of the UP-Filter is accomplished using the low-g trajectory.

Filter Initialization

In addition to establishing the trajectory profiles, it is necessary to initialize the filter matrices. (The values for the $\underline{R}$, $\underline{Q}$, $\underline{P}_o$ and $\hat{\underline{x}}_o$ matrices are listed as part of Appendix A.) First, the initial values for both the state estimates, $\hat{\underline{x}}(t_o)$, and covariance, $\underline{P}(t_o)$, must be specified. As discussed in the introduction to this chapter, filter recovery from bad initial state estimates constitutes a major portion of the analysis effort. To establish a basis for comparison between the two filters, however, the initial state estimates are chosen equal to the true values of the states (i.e. zero initial error). In general, this is a good initialization procedure for a systematic investigation of the filter capabilities. While assuming zero initial errors is appropriate for an initial feasibility study, it is still necessary to consider practical initialization methods. This is done in order to choose values for the $\underline{P}_o$ matrix. Two methods of initializing state estimates are the use of time (t_o^-) measurements of the missile launch vehicle and missile, and the use of intelligence data collected on possible threats. Time (t_o^-) measurements refer to measurements (made by the tracker) of both missile launch vehicle and missile prior to initialization of the missile guidance function. Assuming an on (or nearly on) boresight launch implies that the missile will depart the launch vehicle with the same approximate inertial heading that the launch vehicle has been maintaining. Recall that, from Chapter Two, missile guidance initiation is assumed to occur simultaneously with the termination of missile thrusting. It is assumed that the launch heading is known with a relatively high degree of accuracy.

Combined with raw range-rate measurements, this heading information provides the initial estimates of the inertial velocity vectors, V^I_{mx} and V^I_{my}. The initialization of the estimates of the range, range-rate, and angle states is straightforward since they are measured directly by system sensors (i.e. raw measurements are used as estimates). The last three dynamic states considered in the two filters, x, y, and LOS accelerations, can be initialized by averaging the time rate of change of a small number of velocity/range-rate measurements. Finally, information about the unknown parameters, n and (m/S), can come from intelligence sources and knowledge of the current state-of-the-art in missile design. The values of the initial variances are chosen to reflect the estimation accuracy given by these initialization procedures.

In this study, it is assumed that $\underline{P}_o$ is a diagonal matrix. This assumption is made for expediency and reflects the lack of a priori knowledge of any cross coupling terms. The variances for the measured states, range, range-rate, and angle, can be calculated by summing the low frequency portion of the power spectral densities of the thermal and glint/scintillation components of the respective measurements. Recall that the thermal and glint/scintillation components are considered independent. The acceleration states of the L-Filter utilize a standard deviation that is consistent with the values listed in Table III on page 76. Since the initial values of the velocity states are computed using other estimates, the values would be known with less certainty. Therefore, the variances of these states are chosen to reflect less confidence than that of the measured range-rate state, but of the same order of magnitude. The inertial velocity variances are set equal to 900 ft^2/sec^2 as compared to 100 ft^2/sec^2 for the range-rate variance.

The unknown parameters present a different problem. Information about the unknown parameters will come from intelligence sources and knowledge of the current state-of-the-art. A priori knowledge that the navigation constant is between three and six is used to determine an appropriate standard deviation, i.e. $\sigma = .5$. Therefore, the initial variance for the navigation constant is .25. The (m/S) state is not as well defined as the navigation constant. Certainly the lower absolute limit is zero. The upper limit, however, can vary over a wide range. Confidence in the intelligence available is the primary influence on the initial covariance value. For this study, the (m/S) of a generic missile is used as the mean value of the range of possible values for this parameter. The "standard deviation" is computed using this mean value and the lower limit of zero to define a "three-sigma" interval length.

In addition to establishing initial estimates and variances of the states, it is necessary to determine the strengths of the dynamic driving noises, $\underline{w}(t)$, and measurement noises, $\underline{v}(t)$, i.e. the elements in the appropriate covariance kernels. As in the case of the truth model measurement noises, two types of noise models are used in the filters. The exponentially time-correlated noise model is used in the L-Filter to model the x, y, and LOS accelerations. Recall that this model is a first order lag driven by a white Gaussian noise. The required strength of the noise for this model is (Ref 9:178)

$$q = 2\lambda\sigma^2 \tag{141}$$

where

q = strength of white Gaussian noise driving the first order lag

= reciprocal of the time constant

= standard deviation of the output of the lag.

It is apparent from this expression that both the time constant and the standard deviation determine the strength of the required driving noise. Table III, listed here for completeness, lists general values for these quantities (Ref 10:323, 5:30)

Table III
Acceleration Types

Type	$\sigma(ft/sec^2)$	τ(sec)
small accel./slowly varying	10	3
large accel./slowly varying	64	10
large accel./rapidly varying	64	1
accel. for randomly maneuvering aircraft	96	10

The application of the values in this table are discussed in the last section of this chapter. The two pseudonoises of the UP-Filter previously mentioned and pseudonoises to be added to the range-rate and angle channels of the UP-Filter are all modelled as white Gaussian noises, the strength of which is denoted as q. The values selected for these noises should in general reflect the expected rate of change in the parameter/state being estimated. The unknown parameters are expected to be essentially constant, but adding a pseudonoise can be justified in order to keep the associated Kalman gains from going to zero. The effect of different values of noise strength for the navigation constant are discussed in Chapter Five. The initial noise strengths for the unknown parameters are kept small (i.e. a noise strength of .001 for the navigation constant and .0001 for (m/S)). The noises added to the range-rate and angle

channels are added to ensure that possible degradation of the system model over time is offset by "forcing" the filter to look harder at update measurements (i.e. higher variances on these channels).

Finally, values for the filter measurement noise, $\underline{v}(t)$, covariance matrix must be selected. The initial values for the matrix can be generated by equipment specifications and are obtained in the same manner as used for the initial state variances for range, range-rate, and angle. In addition, the $\underline{R}$ matrix is assumed diagonal for the same reason that $\underline{P}_o$ was.

Performance Criteria

The results of this study are divided into two parts, both of which are located in Appendix B. The first part consists of plots of mean estimation errors with a one standard deviation envelope about these errors. These quantities are calculated as sample statistics obtained from repeated runs of the simulation. Each run is driven by a different noise sample from a random noise generator. The formulae used to compute the mean error are

$$x_{mean}(t_j) = \frac{1}{n_r} \sum_{i=1}^{n_r} x_{comp_i}(t_j) \tag{142}$$

$$e_{mean}(t_j) = x_{true}(t_j) - x_{mean}(t_j) \tag{143}$$

where n_r is the number of runs, $x_{comp_i}(t_j)$ is the filter computed value of the state of interest at time t_j, (on the i-th simulation run), and $x_{true}(t_j)$ is the true or truth model value of the state of interest at time t_j. The notation $e_{mean}(t_j)$ indicates the error in the mean at time (t_j) (Ref 7:16).

The standard deviation is then given by

$$x_{SD}(t_j) = \left[\frac{1}{n_r - 1} \sum_{i=1}^{n_r} \{x_{comp_i}(t_j) - x_{mean}(t_j)\}^2 \right]^{1/2} \quad (144)$$

It is important to note the use of $\left(\frac{1}{n_r - 1}\right)$ in this expression instead of $\left(\frac{1}{n_r}\right)$. It can be shown that the quantity inside the brackets represents an unbiased estimate of the variance of x(t). However, the expression for $x_{SD}(t_j)$ is a biased estimate of the standard deviation for small sample sizes, such as $n_r = 5$ in this study. While it is biased, it is still a better estimate than would have been obtained by utilizing the term $\left(\frac{1}{n_r}\right)$. Methods for eliminating the bias in the standard deviation estimate are presented in Reference 9 (Ref 9:8, 8:269). These statistical characteristics provide a good, though only partial, description of the error characteristics of the filters under investigation.

The second part of the results are plots that depict the tuning of the different channels of each of the two filters. This is accomplished by comparing the filter estimate of the standard deviation, $\sqrt{P_{kk}(t_j)}$, and the computed standard deviation, $x_{SD}(t_j)$. The objective of the tuning process is to minimize the computed standard deviation. As a rule of thumb, good tuning is achieved when the estimated standard deviation is approximately equal to the computed standard deviation. While precise tuning is desirable for eventual implementation, it is not essential for a feasibility study. In addition, precise filter tuning by the use of Monte Carlo techniques is very costly in terms of effort and time. However, this part of the results provides confidence that the filter performance noted in the error of the mean results are representative of the filter's fundamental capabilities. In addition, the tuning results may provide insights into the reasons for performance differences between

the two filters, and may also give insights for filter performance enhancements.

Tuning Philosophy

The tuning effort is limited due to time constraints and the overall objective of this study. It is assumed a priori that case (1) in Table III, namely small, slowly varying acceleration, is an inappropriate model for the dynamics of a missile. The remaining three cases are all considered as possible models.

Landau Filter. The general tuning procedure for this filter involves the choice of an acceleration model which best represents the ensemble of expected missile accelerations. The actual filter performance will be degraded to the extent that the actual target acceleration violates the assumed acceleration model. In actual implementation, the filter model may require adaptive settings of the dynamic driving noise. Due to the time constraints and the scope of this study, adaptive techniques are not considered, but their potential benefits can be discerned from performance sensitivity analysis initiated in this study. The tuning process is concentrated on the low-g trajectory.

For the Landau Filter the tuning process consists of varying the standard deviation and time constants of the three accelerations, a_{mx}^{I}, a_{my}^{I}, and a_{m}^{TLOS}. In addition, the standard deviation of the angle measurement noise component was varied. The three standard deviation values that were tested are listed in Table III. The lamda ($\frac{1}{\tau}$) values were varied from ten to one-tenth (i.e. time constants of one-tenth of a second to ten seconds). It should be noted that this original tuning was accomplished by evaluating large quantities of error values on computer printouts. The use of plots for the tuning is preferable to

this method, but was not utilized due to a lack of computer plotting availability during the period in which tuning was being considered. The initial values of the time constant and standard deviation used were indicative of case (3), large, rapidly varying, acceleration, in Table III. The time constant for the x and LOS acceleration were varied from one second to five seconds and the value of the standard deviation was set equal to 96 ft/sec^2. Both of the acceleration models displayed improved performance as the time constant was increased (i.e. the magnitude of the errors over the majority of the trajectory were decreased). For lower time constants, the x-acceleration model displayed slightly better performance over the higher g portion of the trajectory, but degraded performance over a large portion of the trajectory. With the same standard deviation, the time constants of the two accelerations were varied from five seconds to ten seconds. The five second time constant for the LOS acceleration model continued to provide the best performance. A time constant of seven seconds was found to provide the best performance for the x-acceleration model. The same analysis (i.e. time constant variation) for the two acceleration models was accomplished using a standard deviation of 64 ft/sec^2 as suggested in Reference 10. The values finally chosen are: σ = 96 ft/sec^2, λ_3 = .2, and λ_x = .14. As indicated in Table III, these values describe a missile with large, slowly to moderately varying, acceleration. This is consistent with the profile of the low-g trajectory in Figure 16.

During the initial tuning efforts, it was found that the velocity state not coupled to the angle state is not tunable. Neither it nor its associated random acceleration model is coupled in any way to the other filter states. (The value of this state is computed as a function of

the other estimated values [reference Equation (105-a)]. As will be seen, this is not a problem in the Unknown Parameter Filter.

Due to the dependency of the velocity estimate on the angle channel, it was decided to conduct a limited tuning effort with the angle measurement noise strength. The noise strength was varied from $.5 \times 10^{-6}$ rad^2 to 4×10^{-6} rad^2. The final value chosen is 1×10^{-6} rad^2.

Unknown Parameter Filter. In the UP-Filter, as originally developed in Chapter Three, the filter has no dynamic driving noises if both the proportional navigation constant, n, and (m/S) parameter are modelled as random bias. Recall that, in Chapter Three, pseudonoises are suggested for both parameters. In view of the assumptions made in modelling the UP-Filter, it would be unrealistic to model the system with no dynamic driving noises. No system noises imply that the system model reflects, with absolute certainty, the true system state. It is equally undesirable that the uncertainty in the dynamic states, (i.e. angle, velocity, range, and range-rate states), be so dominant that the coupling effects of the two parameters are not readily observable. The objective then is to tune the model so that good estimates of the parameters are obtainable, but that if poor estimates of the parameters are obtained, the estimates of the dynamic states are not severely degraded. The philosophy adopted is to couple the two parameters into good velocity and range rate estimates. To assure good estimates of the velocity and range-rate states, irrespective of the quality of the parameter estimates, an additive white Gaussian noise is added to the range-rate and angle states. The range-rate noise is added to account for the uncertainties in the model of the state. The angle noise is added to account for uncertainties in the angle model and to ensure that the angle estimates remain "good". As stated in the preceding section on the L-Filter, the velocity states have significant coupling to

the angle state. Therefore, the accuracy of the velocity state estimate is enhanced by a strong coupling to the angle state, but the dynamics of the state can still be sensitive to parameter coupling (i.e. the velocity state is driven by coupling, not noise directly). The values of the additional noises are indicated in the system noise covariance matrix, $\underline{Q}$, in Appendix A.

An alternative to adding noise to the angle channel would be to add a pseudonoise directly to the x-velocity channel. In order to keep the x-velocity estimate sensitive to the navigation constant channel, it was decided that the value of this pseudonoise should be kept small. However, when a noise of strength 100 ft^2/sec^2 was added to the x-velocity channel, there was a negligible effect on filter performance. Due to this result, a time constraint on the research effort, and the fact that the other method had already indicated a limited performance capability, it was decided to pursue the method presented in the previous paragraph in greater depth and to not add a pseudonoise to the x-velocity channel.

Finally, it should be noted that, unlike the L-Filter, both velocity states are tunable for all values of $\sin\theta_T$. However, the state that is not coupled into the angle channel displayed relatively poor estimation capability. Therefore, both for comparison with the L-Filter and for expediency, the state not coupled to the angle channel was computed using the estimates of the other states and not estimated [reference Equation (105-a)].

This chapter has established the techniques and criteria by which the two filters will be compared and evaluated. The chapter ended with a discussion of the tuning philosophy used in the two filters. The results of the tuning are found in the second part of Appendix B. Chapter Five discusses the results of the filter comparison and analysis, and discusses the findings of this study.

V. Results and Conclusions

Introduction

The Unknown Parameter Filter is evaluated using the criteria and tuning philosophy discussed in Chapter Four. The evaluation consists of both a performance comparison between the UP-Filter and the more conventional L-Filter and an evaluation of the UP-Filter's capability to recover from initial errors in state estimates. The results of the comparison and recovery analysis are presented graphically in Appendix B.

In Appendix B, the results are divided into two sections. The first section contains plots of the mean estimation errors (± one standard deviation) while the second section contains the tuning plots; both types of plots are described in Chapter Four. There is a one-to-one correspondence between the plots of the two sections. (The figure number of the tuning plot which corresponds to a plot of section one can be determined by adding 40 to the figure number of the plot in section one.)

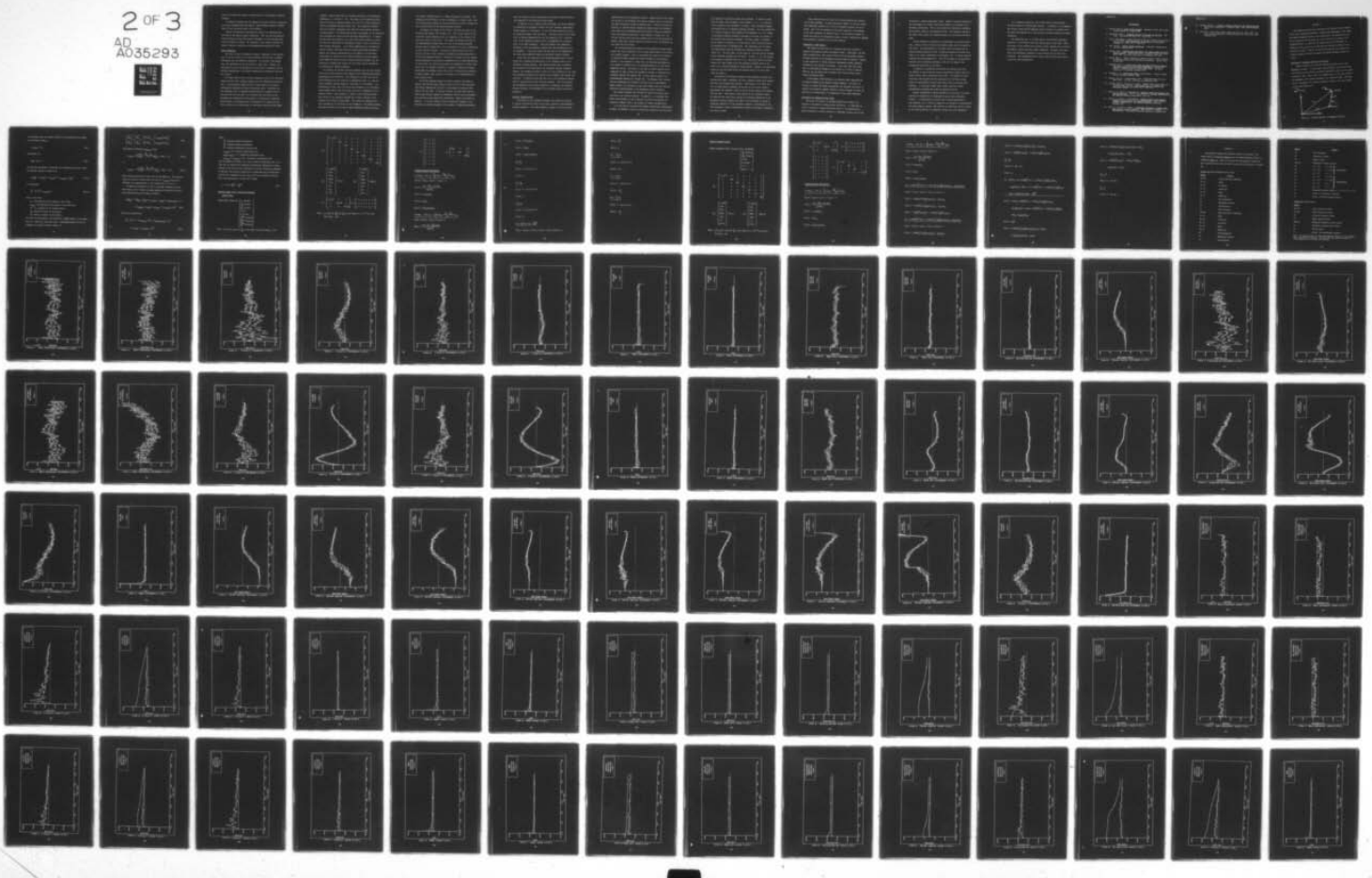
The results provide valuable insights into the capabilities of the UP-Filter. Most importantly, they establish the feasibility of utilizing the UP-Filter to track an incoming missile. As will be seen subsequently, the more detailed modelling of missile dynamics clearly enhances the filter's tracking capability. The estimates provided by the UP-Filter are, in general, less sensitive to system measurement noises because of its more complete internal system model.

Further, the results strongly suggest that the proportional navigation constant can be estimated by the UP-Filter. The recovery analysis provides additional insights into the filter's ability to estimate this parameter. It gives a general indication of the effects that varying the initial variance and noise strength (on the navigation constant channel)

AD-A035 293 AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OHIO SCH--ETC F/G 19/5
APPLICATION OF AN EXTENDED KALMAN FILTER TO AN ADVANCED FIRE CO--ETC(U)
DEC 76 R N LUTTER

UNCLASSIFIED GE/EE/76-31 NL

2 OF 3
AD
AD35293



have on the tuning and recovery characteristics of the navigation constant estimate.

In addition, insights into the adequacy of several modelling assumptions and tuning techniques have been gained. These insights provide a valuable basis for any future work done on the UP-Filter.

Finally, the results of this study are found to be consistent with work done by The Applied Science Corporation. This fact and the graphical results obtained in this study provide confidence in the truth model upon which the analysis is based. The remainder of this chapter discusses the specific results, conclusions, and recommendations generated by study.

Filter Comparison

The first 14 plots of Appendix B present a comparison of the performance of the two filters for the low-g trajectory and the next 14 compare performance over the moderate-g trajectory. In the prior, less dynamic case, the performance of the two filters is about equal. However, for the moderate-g case, the performance of both filters is degraded, and the UP-Filter demonstrates a far greater degradation of performance than does the L-Filter.

The "bias" noted in the x-velocity estimation errors of the UP-Filter, Figures (21) and (35), provides insights into the degradation of performance. A characteristic common to both of these figures is the nearly instantaneous appearance of positive estimation error. Recalling that the estimation error is computed by subtracting the estimated value of the state from the true value, it is obvious that the filter is overestimating the deceleration of the missile in the positive x direction. The difference in deceleration can be accounted for by noting that the filter utilizes a zero order model for the missile, while the truth model

utilizes a second order model and responds according to its performance parameters, $\zeta = .707$ and $\omega = .707$. The effect of this modelling difference can be seen by comparing Figures (21) and (35) to their respective lateral acceleration profiles, Figure (16). The comparison indicates that the x-velocity estimation error leads the actual missile acceleration/deceleration. The higher line-of-sight rates present in the moderate-g case accentuate the effects of this modelling simplification. It should be noted that the decision to use a zero-order missile model was consistent with the design goal of developing as simple a filter as would provide adequate performance. As is clearly demonstrated by the graphical results, however, this model does not provide good enough performance. Despite this apparent undermodelling of the missile in the UP-Filter system model, an analysis of the results obtained in the less dynamic low-g comparison and the recovery evaluation provide insights into the difference in capabilities between the two filters and the feasibility of estimating the unknown parameters.

The narrower one standard deviation envelopes about the mean estimation errors of the UP-Filter plots suggest both an advantage and possible disadvantage of the UP-Filter. The advantage is the fact that the UP-Filter estimates are less sensitive to measurement noise, i.e. that the filter provides a greater degree of consistency in its estimates. The disadvantage is that, while the filter has confidence in it's estimates, these estimates may in fact have large errors due to unmodelled or mismodelled effects. This condition is common to any filtering problem - the accuracy of the estimates obtained are in direct relation to the accuracy of the filter's system model. The narrow one standard deviation envelopes for the UP-Filter are not unexpected. The random acceleration model utilized in the L-Filter admits unmodelled affects and keeps

the filter's bandwidth open (i.e. higher uncertainty is present). The advantage of the UP-Filter is that it eliminates, at least in part, some of the uncertainty inherent in the model within the L-Filter through its more extensive model of the missile guidance function.

In general, the tuning of the L-Filter is better than that of the UP-Filter [Ref Figures (58) through (71)]. This is most obvious in the x-velocity tuning comparison, and in a comparison of the tuning of the final two states of each filter. This result is probably an effect of the undermodelling already noted, but also of the tuning philosophy decision made in Chapter Four (i.e. the decision not to add a pseudonoise to the x-velocity channel). As noted, the system equation for the x-velocity (as well as the y-velocity and range-rate system equations) do in fact have unmodelled effects. Therefore, it appears that the adopted tuning philosophy has increased the degradation due to modelling mismatch. In other words, the filter has been given the capability to adjust effectively only the parameter estimates in response to differences between measurement and estimate values, when in fact it required the capability to adjust dynamic state estimates to account for the undermodelling in these states as well.

The strong coupling between the velocity channels (discussed in Chapter Four) and the angle channel is seen to flow in both directions. Both filters demonstrate this fact in the moderate-g trajectory [Ref Figures (32) and (33)]. In general, however, the angle state and the other states directly updated by measurements reflect better performance than the other states during both trajectories - again, this result is expected. It is seen, however, that the range-rate estimates of the UP-Filter are severely degraded in the presence of significant dynamics

since the range-rate model incorporates the erroneous lateral acceleration component of the zero-order missile model.

In addition to the coupling already mentioned, the errors committed in estimating the navigation constant and (m/S) parameter demonstrate a strong coupling, or correlation. In all the system equations in which the parameters appear, they both appear. This may account for the "negative" correlation demonstrated by the two parameters (i.e. a positive error in the navigation constant error is accompanied with a negative error in the (m/S) parameter). This also suggests that independent estimates of both the navigation constant and (m/S) parameters may not be obtainable. It may also account for some part of the insensitivity the other states exhibited toward the navigation constant estimate.

Recall that the objective of the tuning philosophy used in the UP-Filter was to enhance the sensitivity of the x-velocity state to the navigation constant estimate. The results indicate that, in general, the x-velocity is not sensitive to the navigation constant estimates. However, the navigation constant is seen to be very sensitive to the other state estimates (primarily the velocity state estimate) [Ref Figures (29) and (43)]. It should be noted that the error in the navigation constant tends to lag the error in the x-velocity estimate. This lag decreases, and the magnitude of the errors committed in estimating the navigation constant increase, as the dynamics of the tracking problem become more pronounced.

Recovery Characteristics

This portion of the evaluation considers the UP-Filter's ability to recover from initial errors in the range, x-velocity, and navigation constant estimates. The primary emphasis, however, is on the recovery

characteristics of the navigation constant. Figures (46) and (47) depict the recovery of the UP-Filter from initial estimate errors in x-velocity and range respectively (from two separate runs). While the recovery is positive, it requires several measurement updates before the estimates come within the one standard deviation envelope of the mean estimation error plots generated previously. When compared to Figure (57), the L-Filter recovery from an initial estimate error in the LOS acceleration channel, these plots further support the assertion that the UP-Filter is less tuned than the L-Filter.

The recovery/sensitivity analysis of the navigation constant demonstrates the lack of sensitivity of the other filter states to the navigation constant estimate. Prior to discussing this point and the effects of varying the noise strength and initial variance of the navigation constant, it is necessary to recall that the estimation performance on this channel is "biased" due to the undermodelling noted in the Filter Comparison Section. The effect of the bias is readily apparent from a comparison of cases in which the navigation constant is initially overestimated, Figures (48) through (50), to those cases in which it is underestimated, Figures (51) through (54). The biasing "aids" recovery from an overestimate and "prevents" recovery from an underestimate. Again, however, it is possible to gain some insights into the recovery/sensitivity characteristics of the navigation constant parameter.

The series of Figures (48) through (55) depict the several cases evaluated in the recovery analysis. In general, it is indicated that, given a good estimate of the x-velocity, the filter can estimate the navigation constant, even in the presence of initial estimate error.

It is seen that by increasing the dynamic driving noise (pseudonoise) strength, the estimate is made to converge more rapidly on the true value,

but exhibits an oscillatory steady state response. It should be noted that too high a noise strength on this channel, i.e. q = .5, caused the filter estimate of that parameter to diverge. These statements suggest the possible use of an adaptive g-setting technique. One such possibility is to monitor trends in the estimates of the parameter (i.e. if a number of successive estimates continue to increase or decrease, the values of the driving noise would be increased). The exact number of estimates and the amount of increase in q to be used are values that would need to be determined prior to implementation. The "switching case" in Figure (55) demonstrates both the divergent and convergent effects of larger values of q. For this case, q is set equal to .5 and the initial estimate of the navigation constant has zero error. At time, t = 2.5, the navigation constant is switched from 4.0 to 5.2. The errors noted in the plot reflect the "bias" common to all the navigation constant plots in the study (i.e. x-velocity/trajectory related errors). However, the significant points are the more rapid growth and decrease noted in the error when using the larger q.

An increase in the initial variance of the navigation constant also provided more rapid convergence, but also a smoother steady state response than did the increased noise strength. Additionally, it is found that, of the tuning efforts attempted, an increase in the initial variance demonstrated the "greatest" effect on the x-velocity channel [Ref Figures (56) and (96)]. The results of this section of the evaluation support the idea that the other filter states are relatively insensitive to the navigation constant estimate, but that it may be possible to obtain an estimation capability for the navigation constant - particularly if the (m/S) parameter is made a deterministic quantity by choosing a representative value a priori.

Time constraints did not allow for a recovery/sensitivity analysis of the (m/S) parameter. As noted previously, however, it does not appear that independent estimates of it and the navigation constant can be obtained. Due to the low dynamics in the low-g trajectory, the estimates of the (m/S) parameter are generally good. The correlation between its errors and those of the navigation constant, however, significantly degraded the precision of estimates of the latter quantity.

Comparison to TASC Results

The results of this study are consistent with those obtained in studies by The Applied Science Corporation (TASC). Essentially, the TASC study concluded that, with current state-of-the-art measurement sensors, accurate estimates of the navigation constant cannot be obtained. However, it also stated that if higher precision measurements become available, estimation of the constant is possible. In any case, they have found that, by including the dynamics of the missile in a filter of adequate fidelity, one can obtain better estimates than those obtained using a random acceleration model.

In their model, TASC utilized an acceleration model comprised both of missile dynamics and an additive random component. The latter is included to account for model uncertainties, but degrades the ability to estimate the navigation constant (beyond a certain strength of noise, the degradation is proportional to the magnitude of the random component).

Conclusions and Suggested Future Study

This study has proposed an Extended Kalman Filter design to be utilized in the missile tracking problem (UP-Filter). In comparison to a filter utilizing a random acceleration model, it has demonstrated a limited capability to obtain estimates of comparable accuracy and of less

sensitivity to system measurement noises. However, the study demonstrates the requirement for a higher order missile model to be included in the UP-Filter system model. This in turn requires an additional, preferably more precise, tuning of the remodelled filter. The graphical results of the study clearly depict the mismatch between the truth model and filter model.

Several methods are available for accomplishing the required modelling. Either a first or second order missile model could be included in the filter. The uncertainties of the model could be accounted for by using the method employed by TASC, utilizing an adaptive g-setting technique, or by using a combination of techniques, (i.e. to include adaptive g-setting for some pseudonoises). The noise strength of the appropriate channels could be adjusted according to the rate of change of the acceleration calculated in the filter.

In addition to remodelling the missile and retuning the filter, the addition of bias correction terms, as generated by higher order nonlinear filters, may be appropriate (Ref 9:255). The addition of bias correction terms becomes even more appropriate if the filter tuning results in weak dynamic driving noises (which tend to accentuate bias problems). If warranted, higher order filters could be utilized (implementation problems may become the limiting factors).

Besides establishing a limited capability of the filter, the study supports the feasibility of estimating the proportional navigation constant. It demonstrates an overall lack of system sensitivity to this parameter, but a "correct" response of its estimate to tuning efforts. The study suggests, however, that the lack of sensitivity may be a result of the coupling between the navigation constant and (m/S) parameters.

It is suggested, therefore, that future efforts consider making the (m/S) parameter a deterministic quantity. In addition, it is suggested that further tuning efforts emphasize precise tuning of the dynamic driving states and possible use of the previously mentioned adaptive q-setting techniques.

The feasibility and, to a lesser degree, the potential performance capability of the UP-Filter have been demonstrated in this study. While the results of the analysis have provided several insights into the filter performance, several questions remain unanswered. To answer these questions, the performance capability of the UP-Filter must be more fully determined. The recommendations for future study from this work provide a basis for this determination.

Bibliography

1. Barton, David K. Radar System Analysis. Englewood Cliffs, New Jersey: Prentice-Hall, Inc., April 1965.

2. Blakelock, John H. Automatic Control of Aircraft and Missiles. New York: John Wiley and Sons, Inc., April 1965.

3. Fitts, John Murray. "Aided Tracking as Applied to High Accuracy Pointing Systems", IEEE Transactions on Aerospace and Electronics Systems, Vol. AES-9, No. 3, May 1973.

4. Gelf, Arthur. Applied Optimal Estimation. Cambridge, Massachusetts: The M.I.T. Press, 1974.

5. Kjesbo, Lee A. Terminal Miss Sensitivity of a Radar Guided Air-to-Air Missile System. Unpublished thesis. Wright-Patterson Air Force Base, Ohio: Air Force Institute of Technology, June 1974.

6. Landau, Mark I. "Radar Tracking of Airborne Targets", Paper presented at National Aerospace and Electronics Conference: Dayton, Ohio, 19 May 1976.

7. Lindberg, Eric K. A Radar Error Model and Kalman Filter for Predicting Target States in an Air-to-Air Environment. Unpublished thesis. Wright-Patterson Air Force Base, Ohio: Air Force Institute of Technology, December 1974.

8. Lindgren, B. W. Statistical Theory, (2nd Edition). Toronto, Canada: Collier-MacMillan Limited, 1968.

9. Maybeck, Peter S. Lecture Notes, AFIT. Wright-Patterson Air Force Base, Ohio: Air Force Institute of Technology, 1974.

10. Pearson, John B., III and E. B. Stead. "Kalman Filter Applications in Airborne Radar Tracking", IEEE Transactions on Aerospace and Electronic Systems, Vol AES-10, No. 3, May 1974.

11. Price, C. F. and W. D. Koeningsburg. Adoptive Control and Guidance for Tactical Missiles. Vol II. TASC TR-170-1. Reading, Massachusetts: The Analytic Sciences Corporation, June 1970.

12. Price, C. F. and W. D. Koeningsburg. Adoptive Control With Explicit Parameter Identification for Tactical Missiles. TASC TR-170-3. Reading, Massachusetts: The Applied Sciences Corporation, December 1971.

13. Price, C. F. and R. S. Warren. Performance Evaluation of Homing Guidance Laws for Tactical Missiles. TASC TR-170-4. Reading, Massachusetts: The Analytic Sciences Corporation, January 1973.

14. Stockum, Larry A. A Study of Guidance Controllers for Homing Missiles. Unpublished dissertation. Columbus, Ohio: Ohio State University, 1974.

15. TR-73-287. "Air-to-Air Control Exposition Phase III (Expo III)", Air Force Avionics Laboratory, Wright-Patterson Air Force Base, Ohio, September 1973.

Appendix A

This appendix contains the development and listing of those elements of the filter algorithms required to complete the development of the Landau and Unknown Parameter Filters. More specifically, the equations for the rate of change of the LOS angle, θ_T, and for the rate of change of the closing velocity, V_c, (both for the Landau Filter) are developed. In addition, the $\underline{P}_o$, $\hat{\underline{x}}_o$, $\underline{R}$, $\underline{H}$, $\underline{G}$, and $\underline{Q}$ matrices required as inputs to the filters are listed. Finally, the partial derivatives that comprise the $\underline{F}$ matrix elements for both the Landau and Unknown Parameter Filter are listed.

Derivation of Equations (103-a/b) and (103-f)

Figure 17 depicts the "inertial" geometry (planar case) of the engagement scenario. In this figure, R^I_{OT} and R^I_{OM} are the inertial position vectors of the tracker and missile respectively, and R is the target-to-missile range. Recall that it is assumed that, for the time period of the engagement, the tracker instruments the "inertial" frame. The range vector is assumed to lie along the positive x-axis of the line-of-sight at all times (Figure 17).

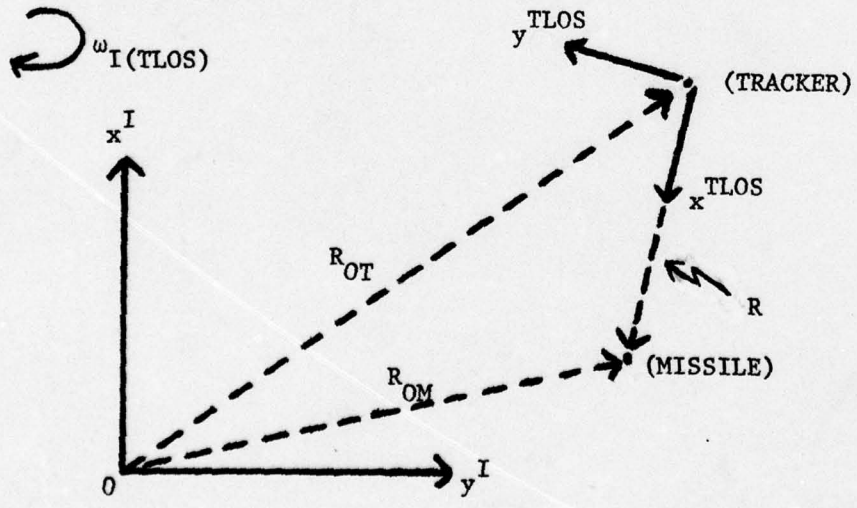


Figure 17. Inertial Geometry of Engagement Scenario

In the planar case, the angular velocity of the TLOS-frame with respect to the inertial frame is

$$\omega_{I(TLOS)} = \dot{\theta}_T \tag{145}$$

From Figure (17)

$$R_{OM} = R_{OT} + R \tag{146}$$

By utilizing the Theorem of Coriolis, and coordinatizing in the I-frame, the inertial velocity is found to be

$$(p_I R^I_{OM})^I = (p_I R_{OT})^I + (p_{TLOS} R)^I + (\omega_{I(TLOS)} \times R_{OM})^I \tag{147-a}$$

or equivalently

$$V^I_m = V^I_T + \dot{R}^I + \omega_{I(TLOS)} R^I \tag{147-b}$$

where in the above

p_I = time derivative with respect to the I-frame

p_{TLOS} = time derivative with respect to the TLOS frame

$\dot{R} = -V_c$ = negative of the closing velocity

V^I_m = inertial velocity of the missile

V^I_T = inertial velocity of the tracker.

Note that the cross product term results in a simple product in the planar case (i.e. employ the right-hand rule). Expressing Equation (147-b) in component form yields (reference Figure 17)

$$\begin{bmatrix} V^I_{mx} \\ V^I_{my} \end{bmatrix} = \begin{bmatrix} V^I_{Tx} \\ V^I_{Ty} \end{bmatrix} + \begin{bmatrix} \dot{R}\cos\theta_T \\ \dot{R}\sin\theta_T \end{bmatrix} + \begin{bmatrix} -\omega_{I(TLOS)} R\sin\theta_T \\ \omega_{I(TLOS)} R\cos\theta_T \end{bmatrix} \tag{148}$$

and finally, solving for $\omega_{I(TLOS)}$ yields

$$\omega_{I(TLOS)} = \frac{\dot{R}}{R}\frac{\cos\theta_T}{\sin\theta_T} + \frac{V^I_{Tx} - V^I_{mx}}{R}\frac{1}{\sin\theta_T} \; ; \; \sin\theta_T \geq .707 \tag{149-a}$$

or

$$\omega_{I(TLOS)} = -\frac{\dot{R}}{R}\frac{\sin\theta_T}{\cos\theta_T} + \frac{V^I_{my} - V^I_{Ty}}{R}\frac{1}{\cos\theta_T} \; ; \; \sin\theta_T < .707 \tag{149-b}$$

These equations describe the angle rate system equations. The alternate forms are required to ensure that the value for $\omega_{I(TLOS)}$ does not go unbounded as θ_T approaches $N\pi$ or $N\pi/2$ respectively, N = 1,2,3,$\cdots$.

By taking the derivative of each of the terms in Equation (147-a) with respect to the I-frame, and again applying the Theorem of Coriolis and coordinatizing in the I-frame, we obtain

$$(p^2_I R_{OM})^I = [p^2_I R_{OT} + p^2_{TLOS} R + \omega_{I(TLOS)} \times p_{TLOS} R + p_{TLOS}\omega_{I(TLOS)} \times R$$

$$+ \omega_{I(TLOS)} \times p_{TLOS} R + \omega_{I(TLOS)} \times (\omega_{I(TLOS)} \times R)]^I \tag{150}$$

This can be rewritten as

$$a^I_m = a^I_T + \ddot{R}^I + 2(\omega_{I(TLOS)} \times \dot{R})^I + (p_{TLOS}\omega_{I(TLOS)} \times R)^I$$

$$+ (\omega_{I(TLOS)} \times (\omega_{I(TLOS)} \times R))^I \tag{151}$$

where

a_m^I = inertial missile acceleration

a_T^I = inertial tracker acceleration

$\ddot{R}^I$ = relative acceleration along the LOS

$\omega_{I(TLOS)} \times R$ = Coriolis acceleration term

$P_{TLOS}\omega_{I(TLOS)} \times R$ = tangential acceleration term

$\omega_{I(TLOS)} \times (\omega_{I(TLOS)} \times R)$ = centripetal acceleration term.

From the geometry of Figure (17), and by using the right-hand rule, it is seen that the Coriolis and tangential acceleration components are perpendicular to the LOS and do not contribute to the LOS acceleration component. In addition, the a_m^I and a_T^I terms must be transformed into the TLOS-frame and only the x-component of each used in the LOS acceleration equation. After this manipulation, solving for $\ddot{R}^I$ yields

$$\ddot{R} = \omega^2 R + a_{Tx}^{TLOS} - a_{mx}^{TLOS} \tag{152}$$

Extended Kalman Filter (Defining Parameters)

Landau Filter

Landau Filter states are $\begin{bmatrix} \theta_T \text{ (radians)} \\ V_{mx}^I \text{ (ft/sec)} \\ V_{my}^I \text{ (ft/sec)} \\ R \text{ (ft)} \\ \dot{R} \text{ (ft/sec)} \\ a_m^{TLOS} \text{ (ft/sec}^2\text{)} \\ a_{mx}^{TLOS} \text{ (ft/sec}^2\text{)} \\ a_{my}^{TLOS} \text{ (ft/sec}^2\text{)} \end{bmatrix}$

*Note: The Q(3,3) term for a_{my}^I is set equal to zero for $\sin\theta_T \geq .707$.

$$\underline{P}_o = \begin{bmatrix} 1 \times 10^{-6} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 900 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 900 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 256 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 100 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 4100 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 4100 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4100 \end{bmatrix}$$

$$\hat{\underline{x}}_o = \begin{bmatrix} 4.36334 \\ 1225. \\ 2793. \\ 10000. \\ -2127. \\ 193. \\ -66. \\ -180. \end{bmatrix} \text{(low g)} \qquad \hat{\underline{x}}_o = \begin{bmatrix} 4.18879 \\ 1525. \\ 2642. \\ 10000. \\ -2254. \\ 196.2 \\ -98. \\ -169.7 \end{bmatrix} \text{(high g)}$$

$$R = \begin{bmatrix} 1 \times 10^{-6} & 0 & 0 \\ 0 & 278.89 & 0 \\ 0 & 0 & 201.64 \end{bmatrix} \qquad H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

*Note: P_o terms for V^I_{my} and a^I_{my} are set equal to 1×10^{-8} for runs with $\sin\theta_T \geq .707$.

$$G = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \qquad Q = \begin{bmatrix} 3600 & 0 & 0 \\ 0 & 2256 & 0 \\ 0 & 0 & 3600 \end{bmatrix}$$

F-matrix partial derivatives:

If $\sin\theta_T \geq .707$, $\dot{x}_1 = \frac{x_5}{x_4}\cot x_1 + \frac{V_{Tx}^I - x_2}{x_4}\csc x_1$

$F(1,3) = F(1,6) = F(1,7) = F(1,8) = 0$

$$F(1,1) = \frac{-x_5 - (V_{Tx}^I - x_2)\cos x_1}{x_4 \sin^2 x_1}$$

$$F(1,2) = -1/(x_4 \sin x_1)$$

$$F(1,4) = -\dot{x}_1/x_4$$

$$F(1,5) = \cos x_1/(x_4 \sin x_1)$$

If $\sin\theta_T < .707$, $\dot{x}_1 = -\frac{x_5}{x_4}\tan x_1 + \frac{(x_3 - V_{Ty}^I)}{x_4}\csc x_1$

$F(1,2) = F(1,6) = F(1,7) = F(1,8) = 0$

$$F(1,1) = \frac{-x_5 + (x_3 - V_{Ty}^I)\sin x_1}{x_4 \cos^2 x_1}$$

$F(1,3) = 1/(x_4 \cos x_1)$

$F(1,4) = -\dot{x}_1/x_4$

$F(1,5) = -\sin x_1/(x_4 \cos x_1)$

$\underline{\dot{x}_2 = x_7}$

$F(2,j) = 0 \quad \text{for all } j \neq 7$

$F(2,7) = 1$

$\underline{\dot{x}_3 = x_8}$

$F(3,j) = 0 \quad \text{for all } j \neq 8$

$F(3,8) = 1$

$\underline{\dot{x}_4 = x_5}$

$F(4,j) = 0 \quad \text{for all } j \neq 5$

$F(4,5) = 1$

$\underline{\dot{x}_5 = (\dot{x}_1)^2 x_4 + x_6 - a_{Tx}^{TLOS}}$

$F(5,1) = F(5,2) = F(5,3) = F(5,5) = F(5,7) = F(5,8) = 0$

$F(5,4) = \dot{x}_1^2$

$F(5,6) = 1$

$\dot{x}_6 = -\frac{1}{\tau_3} x_6$

$F(6,7) = 0$ for all $j \neq 6$

$F(6,6) = -\frac{1}{\tau}$

$\dot{x}_7 = -\frac{1}{\tau_x} x_7$

$F(7,j) = 0$ for all $j \neq 7$

$F(7,7) = -\frac{1}{\tau_x}$

$\dot{x}_8 = -\frac{1}{\tau_y} x_8$

$F(8,j) = 0$ for all $j \neq 8$

$F(8,8) = -\frac{1}{\tau_y}$

Unknown Parameter Filter

Unknown Parameter Filter states are $\begin{bmatrix} \theta_T \text{ (radians)} \\ V^I_{mx} \text{ (ft/sec)} \\ V^I_{my} \text{ (ft/sec)} \\ R \text{ (ft)} \\ \dot{R} \text{ (ft/sec)} \\ n \text{ (--)} \\ (m/S) \text{ (--)} \end{bmatrix}$

$$\underline{P}_o = \begin{bmatrix} 1 \times 10^{-6} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 900 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 900 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 256 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 100 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & .25 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 100 \end{bmatrix}$$

$$\hat{\underline{x}}_o = \begin{bmatrix} 4.36334 \\ 1225. \\ 2793. \\ 10000. \\ -2127. \\ 4.0 \\ 29.197 \end{bmatrix} \text{ (low g)} \qquad \hat{\underline{x}}_o = \begin{bmatrix} 4.18879 \\ 1525. \\ 2642. \\ 10000. \\ -2254. \\ 4.0 \\ 29.197 \end{bmatrix} \text{ (high g)}$$

*Note: The $P_o(3,3)$ term for V^I_{my} is set equal to 1×10^{-8} for the case of $\sin\theta_T \geq .707$.

$$R = \begin{bmatrix} 4 \times 10^{-6} & 0 & 0 \\ 0 & 278.89 & 0 \\ 0 & 0 & 201.64 \end{bmatrix} \qquad H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad Q = \begin{bmatrix} 1 \times 10^{-5} & 0 & 0 & 0 \\ 0 & 100 & 0 & 0 \\ 0 & 0 & .001 & 0 \\ 0 & 0 & 0 & 1 \times 10^{-4} \end{bmatrix}$$

F-matrix partial derivatives::

If $\sin\theta_T \geq .707$, $\dot{x}_1 = \frac{x_5}{x_4}\cot x_1 + \frac{V_{Tx}^I - x_2}{x_4}\csc x_1$

$F(1,3) = F(1,6) = F(1,7) = F(1,8) = 0$

$$F(1,1) = \frac{-x_5 - (V_{Tx}^I - x_2)\cos x_1}{x_4 \sin^2 x_1}$$

$F(1,2) = -1/(x_4 \sin x_1)$

$F(1,4) = -\dot{x}_1/x_4$

$F(1,5) = \cos x_1/(x_4 \sin x_1)$

If $\sin\theta_T < .707$, $\dot{x}_1 = -\frac{x_5}{x_4}\tan x_1 + \frac{(x_3 - V_{Ty}^I)}{x_4}\sec x_1$

$F(1,2) = F(1,6) = F(1,7) = F(1,8) = 0$

$$F(1,1) = \frac{-x_5 + (x_3 - V_{Ty}^I)\sin x_1}{x_4 \cos^2 x_1}$$

$F(1,3) = 1/(x_4 \cos x_1)$

$F(1,4) = \dot{x}_1/x_4$

$F(1,5) = -\sin x_1/(x_4 \cos x_1)$

$\dot{x}_2 = -[.0338\, V_m^{3/2}(1/x_7) + (.0390\,(\dot{x}_1)^2 x_7 x_6^2 x_5^2)/V_m]\sin\gamma_m - x_6 x_5 (\dot{x}_1)\cos\gamma_m$

$F(2,1) = F(2,2) = F(2,3) = F(2,4) = F(2,8) = 0$

$F(2,5) = (.0780(\dot{x}_1)^2 x_7 x_6^2 x_5 \sin\gamma_m)/V_m + x_6 \dot{x}_1 \cos\gamma_m$

$F(2,6) = (-.0780(\dot{x}_1)^2 x_7 x_5^2 x_6 \sin\gamma_m)/V_m - x_5 \dot{x}_1 \cos\gamma_m$

$F(2,7) = .0338 V_m^{3/2} \sin\gamma_m / x_7^2 - (.0390(\dot{x}_1)^2 x_6^5 x_5^2 \sin\gamma_m)/V_m$

$\dot{x}_3 = -[.0338 V_m^{3/2}(1/x_7) + (.0390(\dot{x}_1)^2 x_7 x_6^2 x_5^2)/V_m]\cos\gamma_m + x_6 x_5 \dot{x}_1 \sin\gamma_m$

$F(3,1) = F(3,2) = F(3,3) = F(3,4) = F(3,8) = 0$

$F(3,5) = (.0780(\dot{x}_1)^2 x_7 x_6^2 x_5 \cos\gamma_m)/V_m - x_6 \dot{x}_1 \sin\gamma_m$

$$F(3,6) = (-.0780(\dot{x}_1)^2 x_7 x_5^2 x_6 \cos\gamma_m)/V_m + x_5 \dot{x}_1 \cos\gamma_m$$

$$F(3,7) = .0338V_m^{3/2}\cos\gamma_m/x_7^2 - (.0390(\dot{x}_1)^2 x_6^2 x_5^2 \cos\gamma_m)/V_m$$

$$\underline{\dot{x}_4 = x_5}$$

$$F(4,j) = 0 \quad \text{for } j \neq 5$$

$$F(4,5) = 1$$

$$\dot{x}_5 = (\dot{x}_1)^2 x_4 + \{[-(.0338V_m^{3/2}/x_7) + (.0390(\dot{x}_1)^2 x_7 x_5^2 x_6^2/V_m)]\sin\gamma_m$$

$$- x_6 x_5 \dot{x}_1 \cos\gamma_m\} \cos x_1 + \{[-(.0338V_m^{3/2}/x_7) + (.0390(\dot{x}_1)^2 x_7 x_5^2 x_6^2/V_m)]$$

$$\cos\gamma_m + x_5 x_6 \dot{x}_1 \sin\gamma_m\}\sin x_1 - a_{Tx}^{TLOS}$$

$$F(5,1) = -\sin x_1[-(.0338V_m^{3/2}/x_7 + .0390(\dot{x}_1)^2 x_7 x_5^2 x_6^2/V_m)\sin\gamma_m$$

$$-x_6 x_5 \dot{x}_1 \cos\gamma_m] + \cos x_1[-(.0338V_m^{3/2})/x_7 + .0390(\dot{x}_1)^2 x_7 x_5^2 x_6^2/V_m)$$

$$\cos\gamma_m + x_6 x_5 \dot{x}_1 \sin\gamma_m]$$

$$F(5,4) = (\dot{x}_1)^2$$

$$F(5,5) = (.0780(\dot{x}_1)^2 x_7 x_5 x_6^2 \sin\gamma_m/V_m)(\cos x_1 + \sin x_1)$$

$$+ (x_6 \dot{x}_1 \cos\gamma_m)(\cos x_1 - \sin x_1)$$

$$F(5,6) = -(.0780(\dot{x}_1)^2 x_7 x_6 x_5^2 \sin\gamma_m / V_m)(\cos x_1 + \sin x_1)$$

$$+ (x_5 \dot{x}_1 \cos\gamma_m)(\sin x_1 - \cos x_1)$$

$$F(5,7) = (.0338 V_m^{3/2} \sin\gamma_m / x_7^2 - .0390(\dot{x}_1)^2 x_5^2 x_6^2 / V_m)$$

$$\sin\gamma_m(\cos x_1 + \sin x_1)$$

$\dot{x}_6 = - 0$

$F(6,j) = 0$ for all j

$\dot{x}_7 = 0$

$F(7,j) = 0$ for all j

Appendix B

This Appendix contains the graphical results of the study. The Landau Filter is designated Filter 1 and the Unknown Parameter Filter is designated Filter 2. There is a one-to-one correspondence between the mean estimation error plots of Section One and the tuning plots of Section Two.

Section One (Mean Estimation Error Plots)

Figure	Subject
[18-31]	Low-g Trajectory Comparison
18, 19	Angle
20, 21	x-velocity
22, 23	y-velocity
24, 25	Range
26, 27	Range-rate
28	LOS-acceleration
29	Navigation constant
30	x-acceleration
31	(m/S) parameter
[32-45]	High-g Trajectory Comparison
32, 33	Angle
34, 35	x-velocity
36, 37	y-velocity
38, 39	Range
40, 41	Range-rate
42	LOS-acceleration
43	Navigation constant
44	x-acceleration

Figure	Subject	
45	(m/S) parameter	
46	x-velocity recovery	
47	Range recovery	
[48-54]	Navigation constant recovery	
48	$P_o = .25$ $Q = .001$	overestimate
49	$P_o = 2.25$ $Q = .001$	overestimate
50	$P_o = .25$ $Q = .1$	overestimate
51	$P_o = .25$ $Q = .001$	underestimate
52	$P_o = 2.25$ $Q = .001$	underestimate
53	$P_o = .25$ $Q = .1$	underestimate
54	$P_o = .25$ $Q = .5$	underestimate
55	Navigation Constant Switch; t = 2.5 (n = 4.0 → 5.2)	
56	Effect on x-velocity of (Fig 50)	
57	Filter 1 LOS-acceleration recovery	

Section Two (Tuning Plots)

Figure	
[58-71]	Low-g trajectory tuning
[72-85]	High-g trajectory tuning
86	x-velocity recovery tuning
87	Range recovery tuning
[88-94]	Navigation constant recovery tuning
95	Navigation constant switch tuning
96	Fig 56 tuning
97	Filter 1 LOS-acceleration recovery

Note: The scale on both the range and range-rate plots for Filter Number One in 18 units per division. The scale for the respective plots of Filter Number Two is 30 units per division.

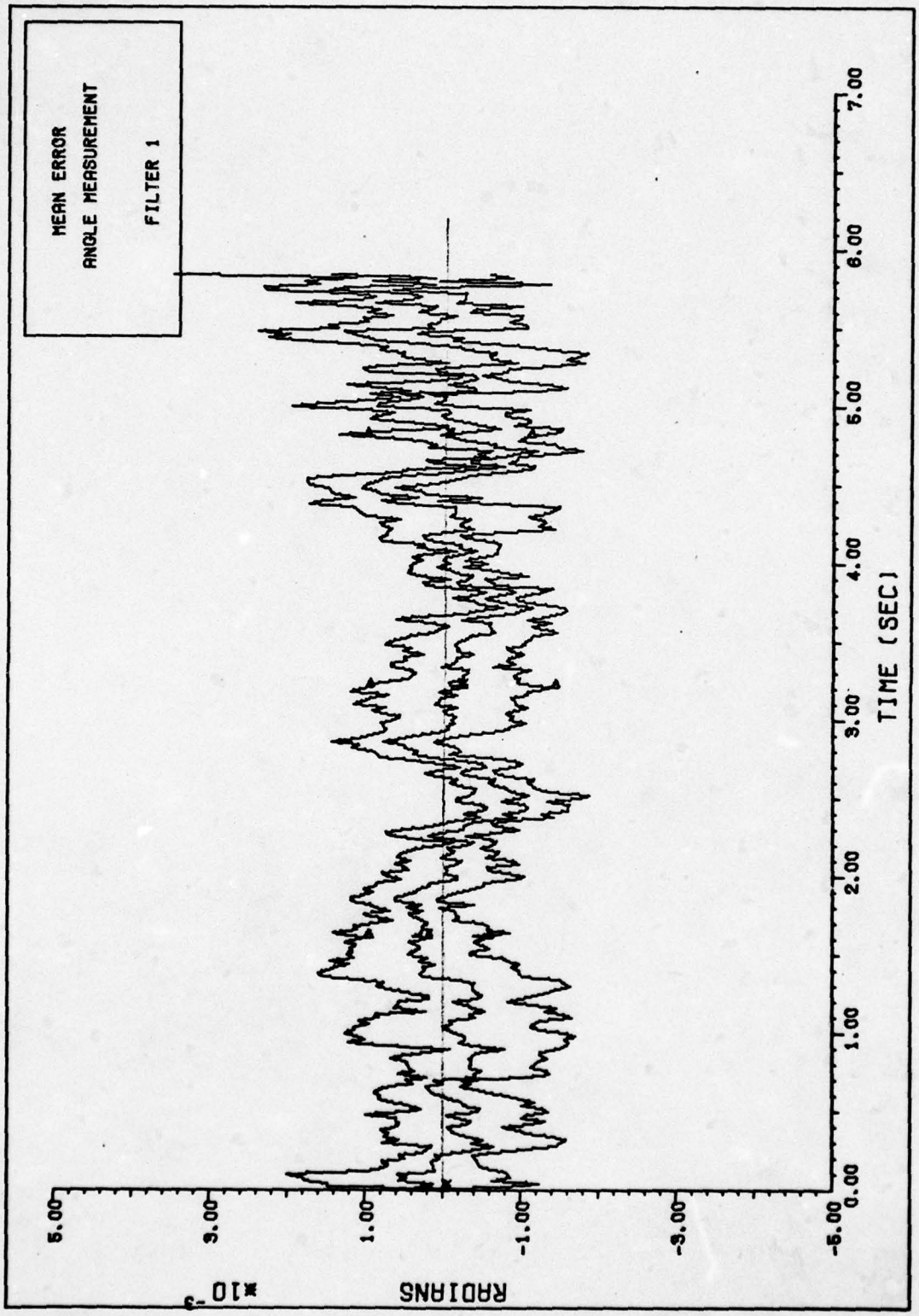


FIGURE 18. ANGLE MEASUREMENT PERFORMANCE FILTER 1

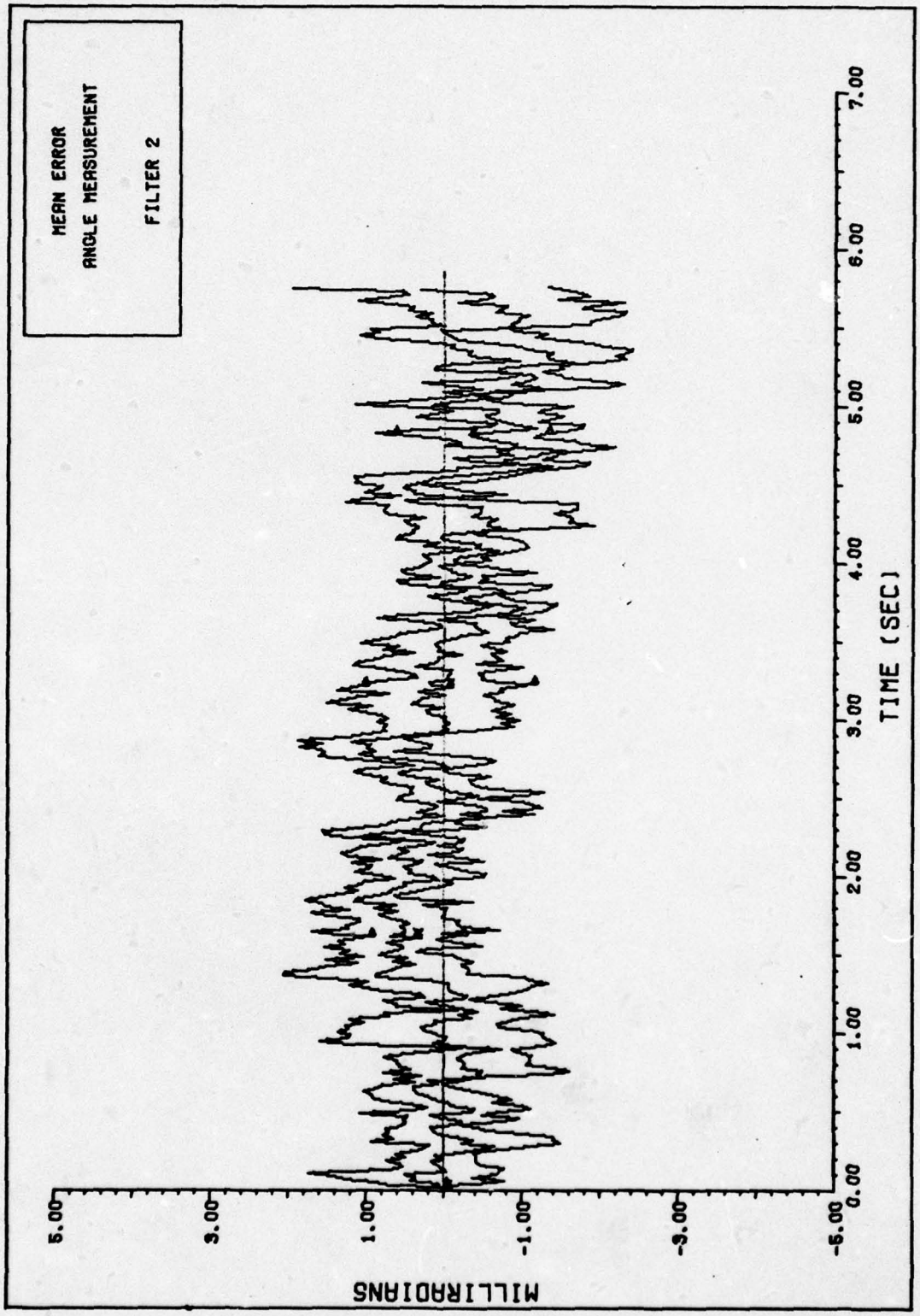


FIGURE 19. ANGLE MEASUREMENT PERFORMANCE FILTER 2

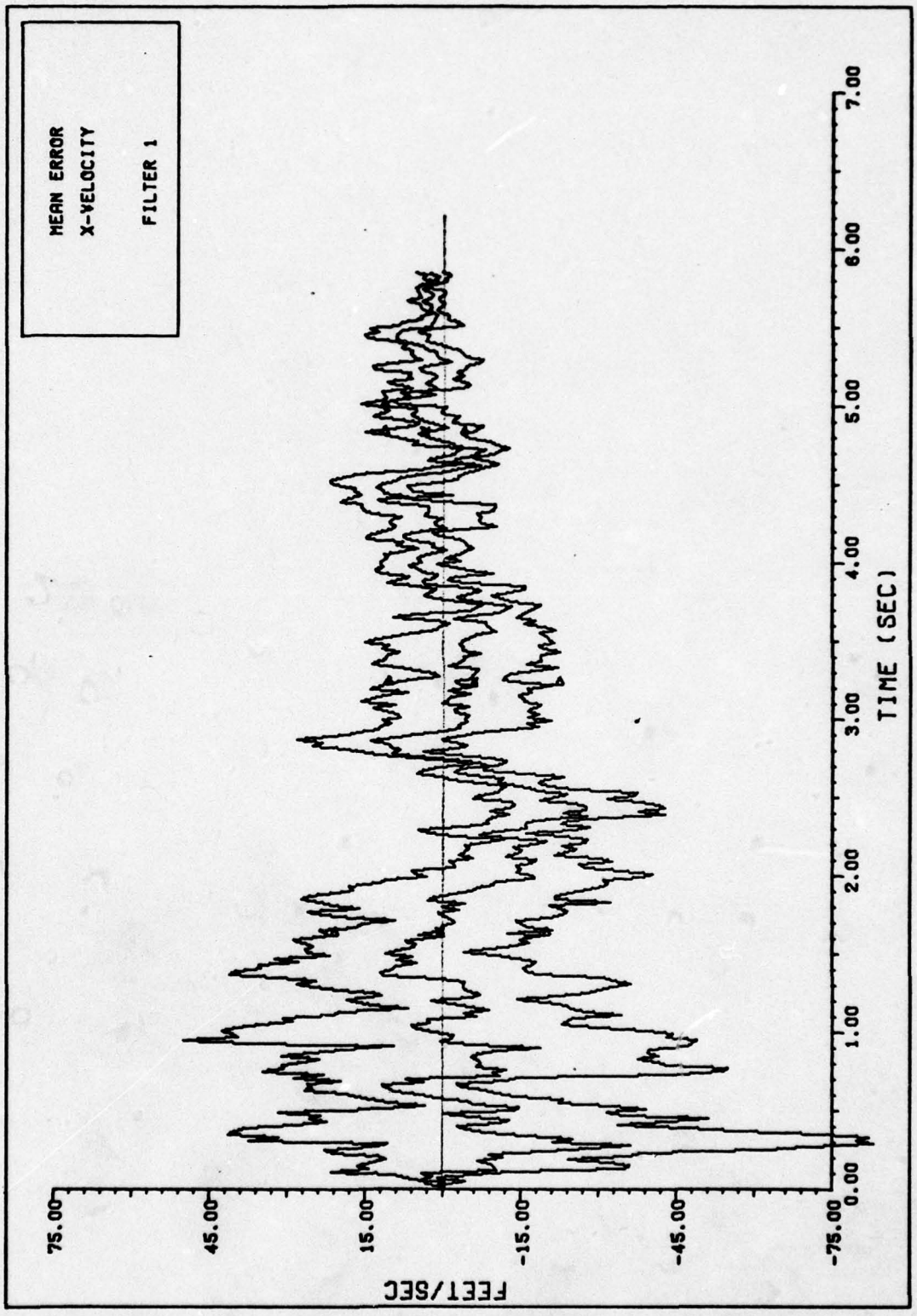


FIGURE 20. X-VELOCITY PERFORMANCE FILTER 1

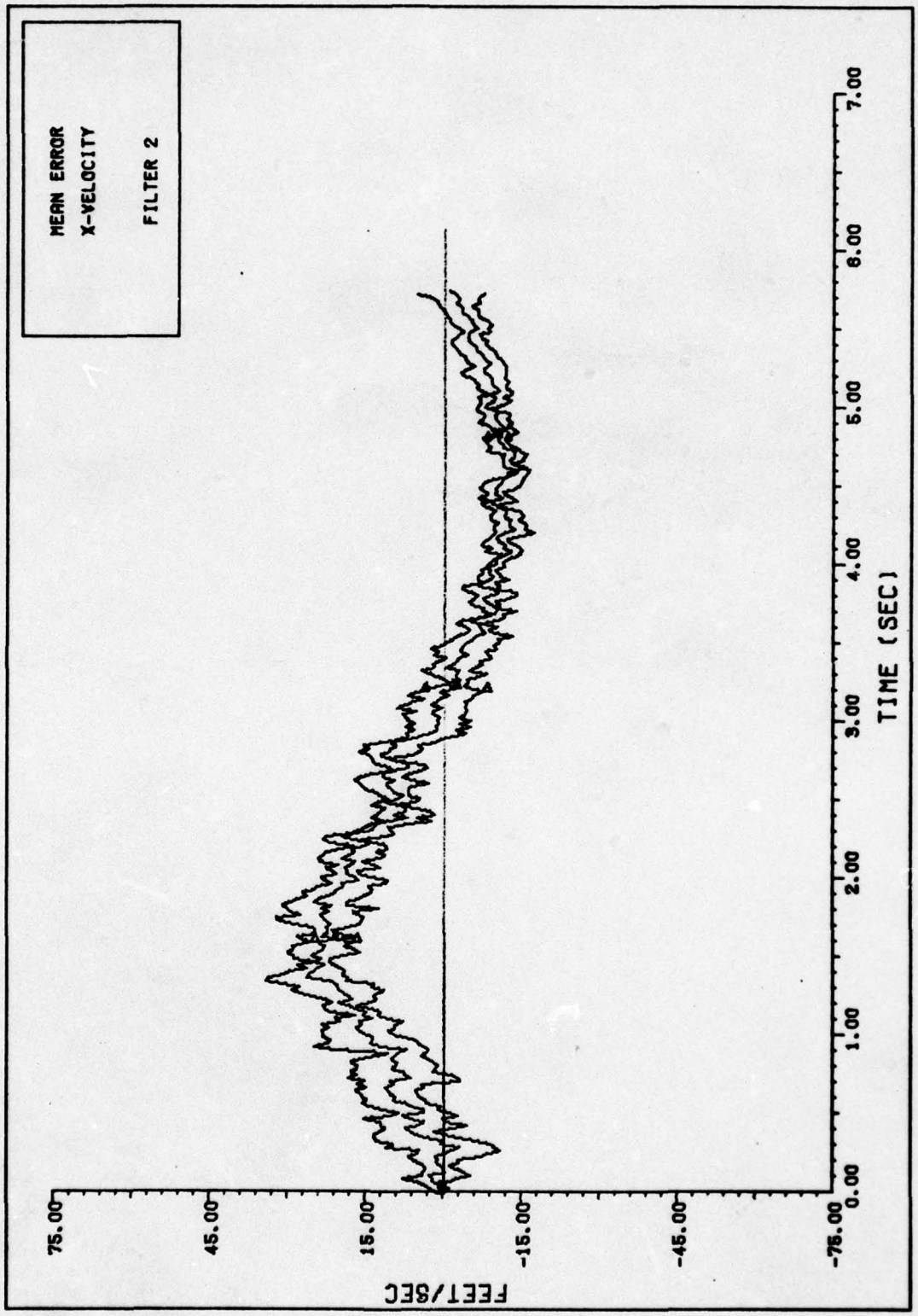


FIGURE 21. X-VELOCITY PERFORMANCE FILTER 2

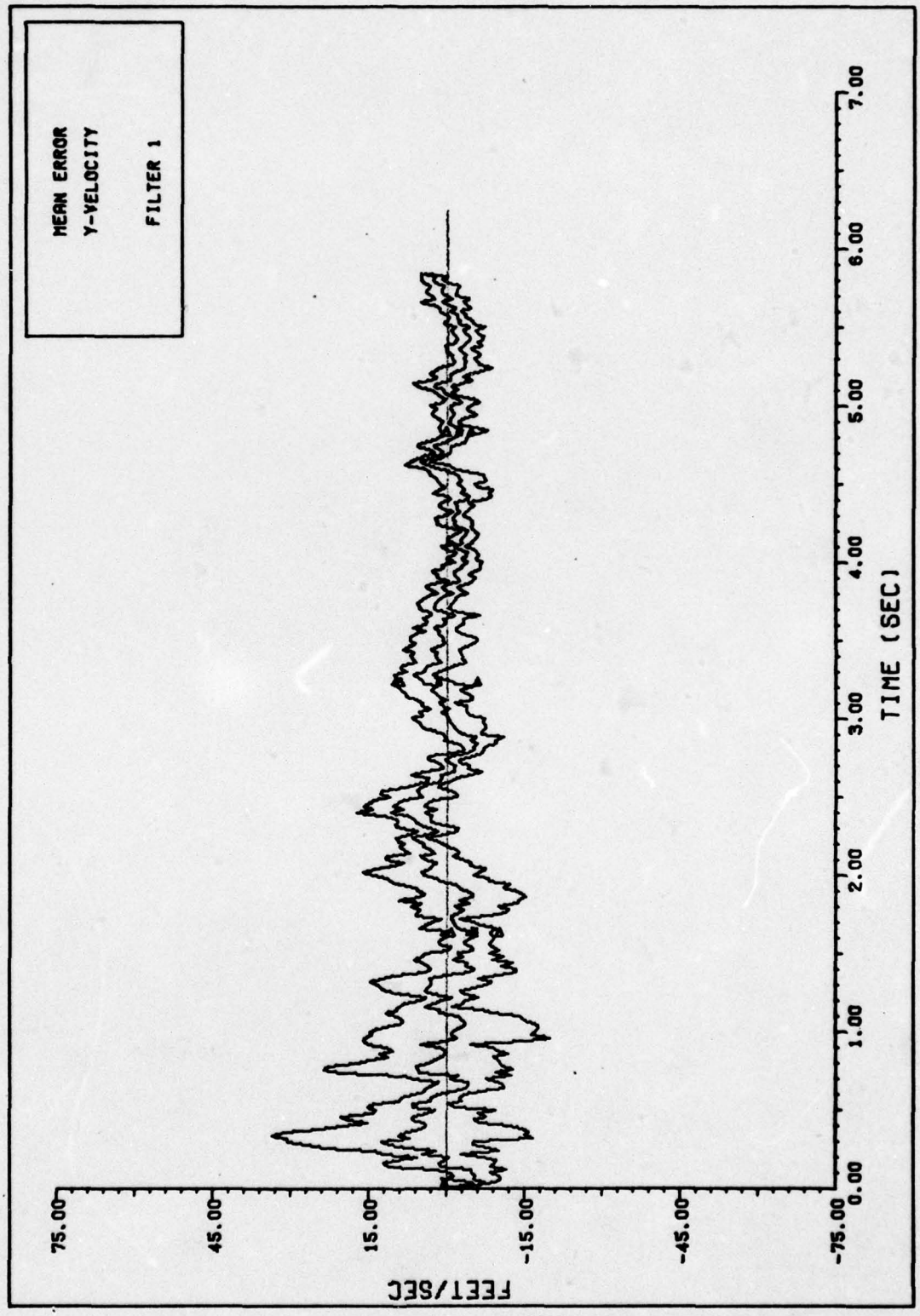


FIGURE 22. Y-VELOCITY PERFORMANCE FILTER 1

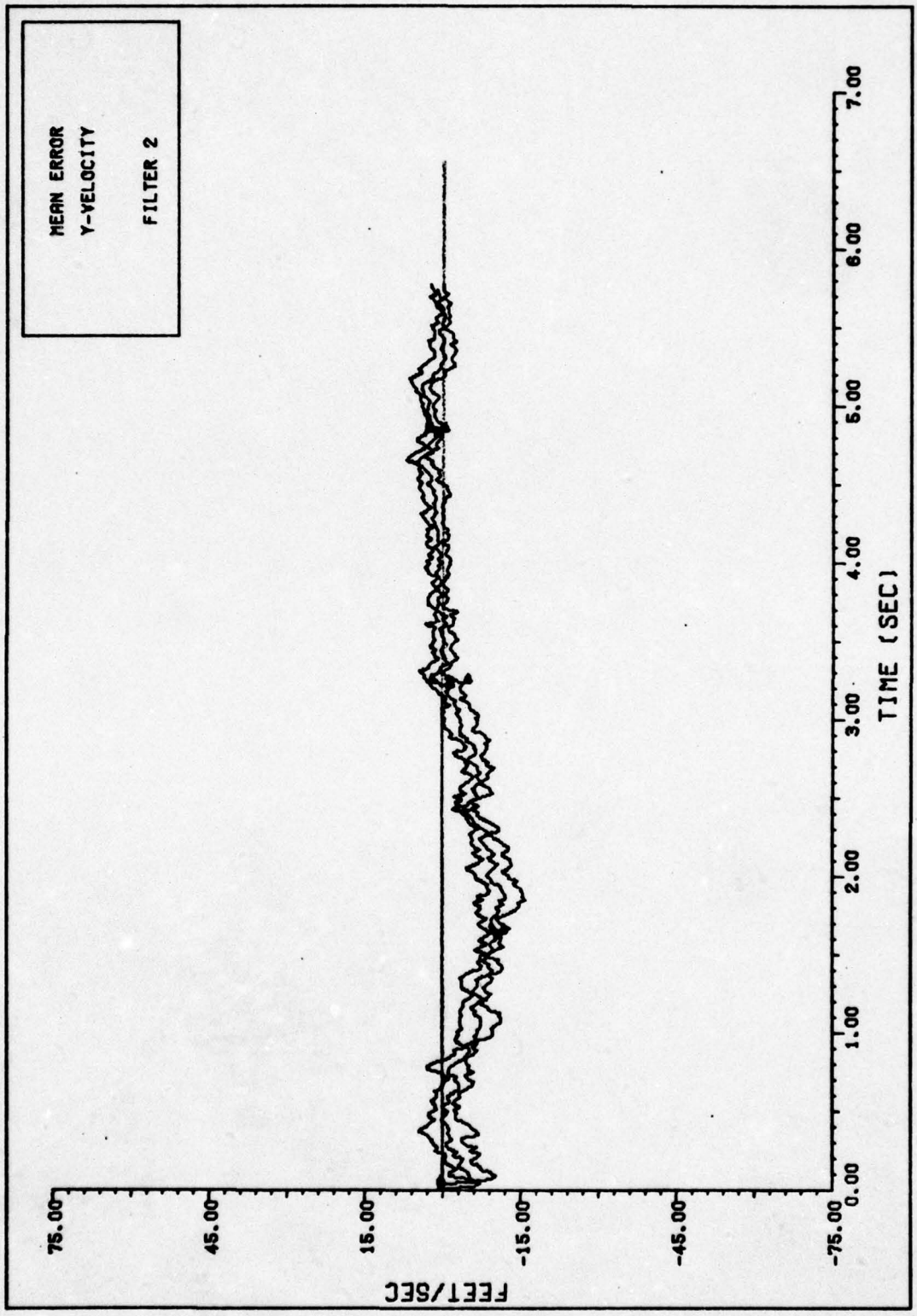


FIGURE 23. Y-VELOCITY PERFORMANCE FILTER 2

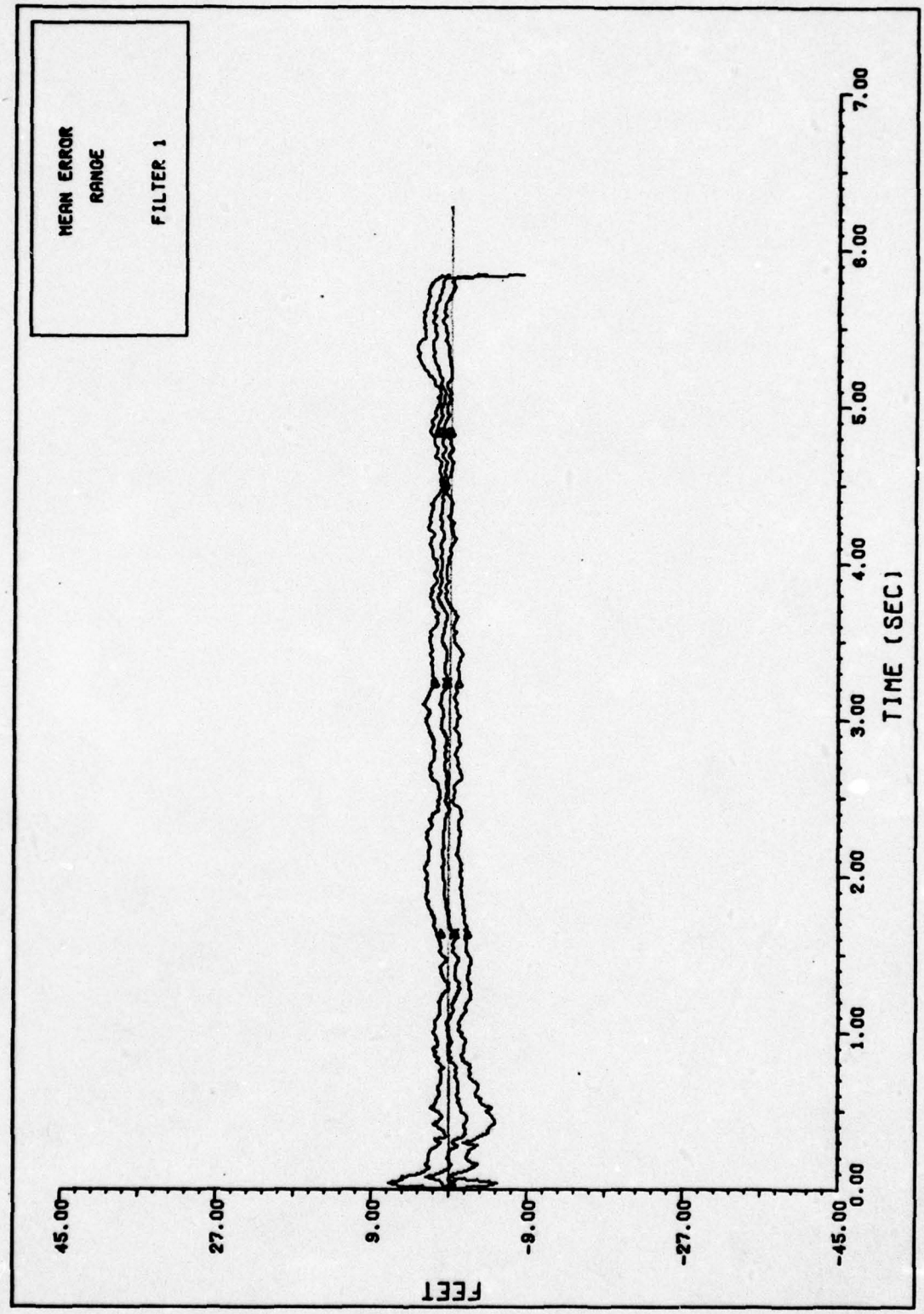


FIGURE 24. RANGE PERFORMANCE FILTER.1

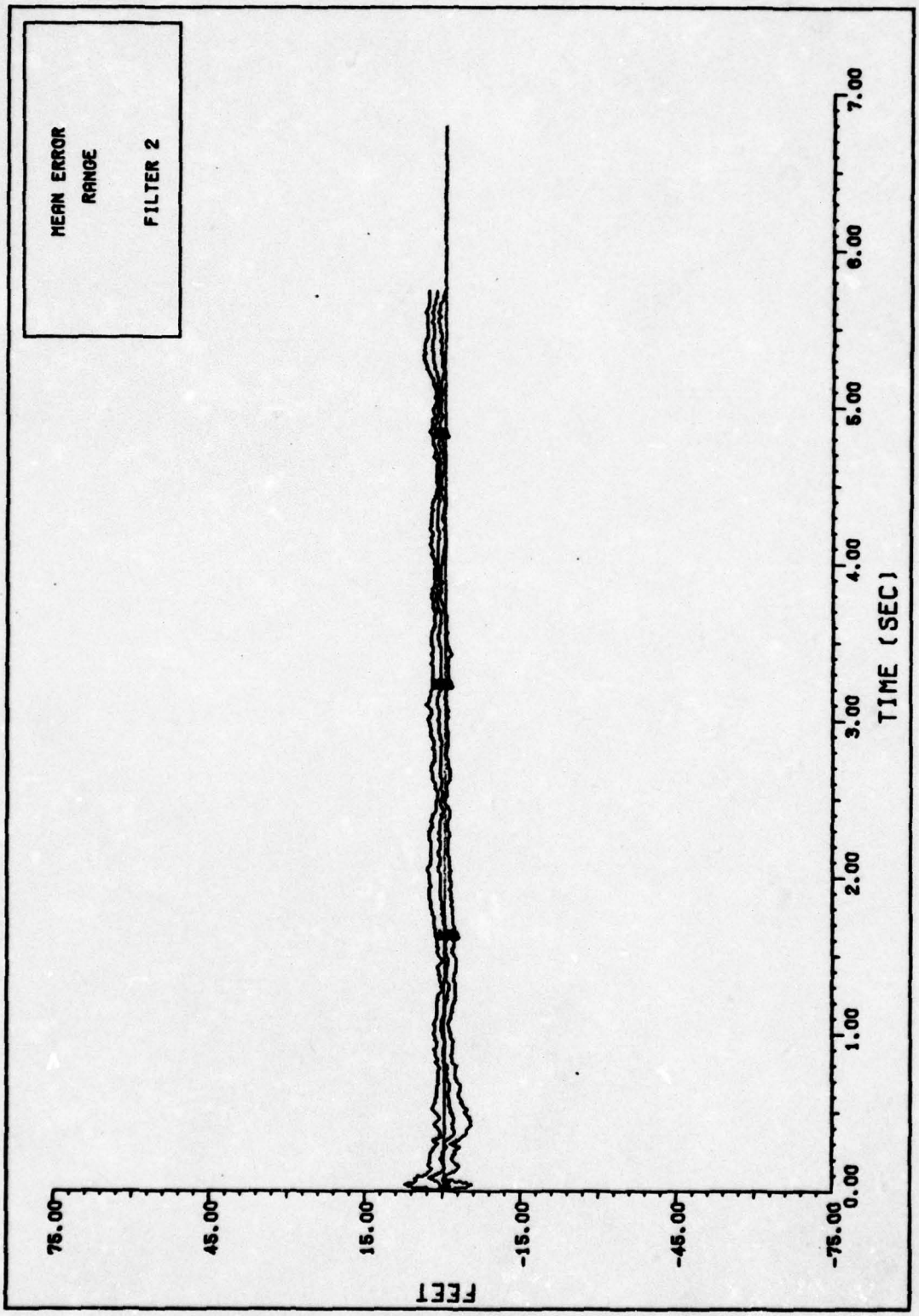


FIGURE 25. RANGE PERFORMANCE FILTER 2

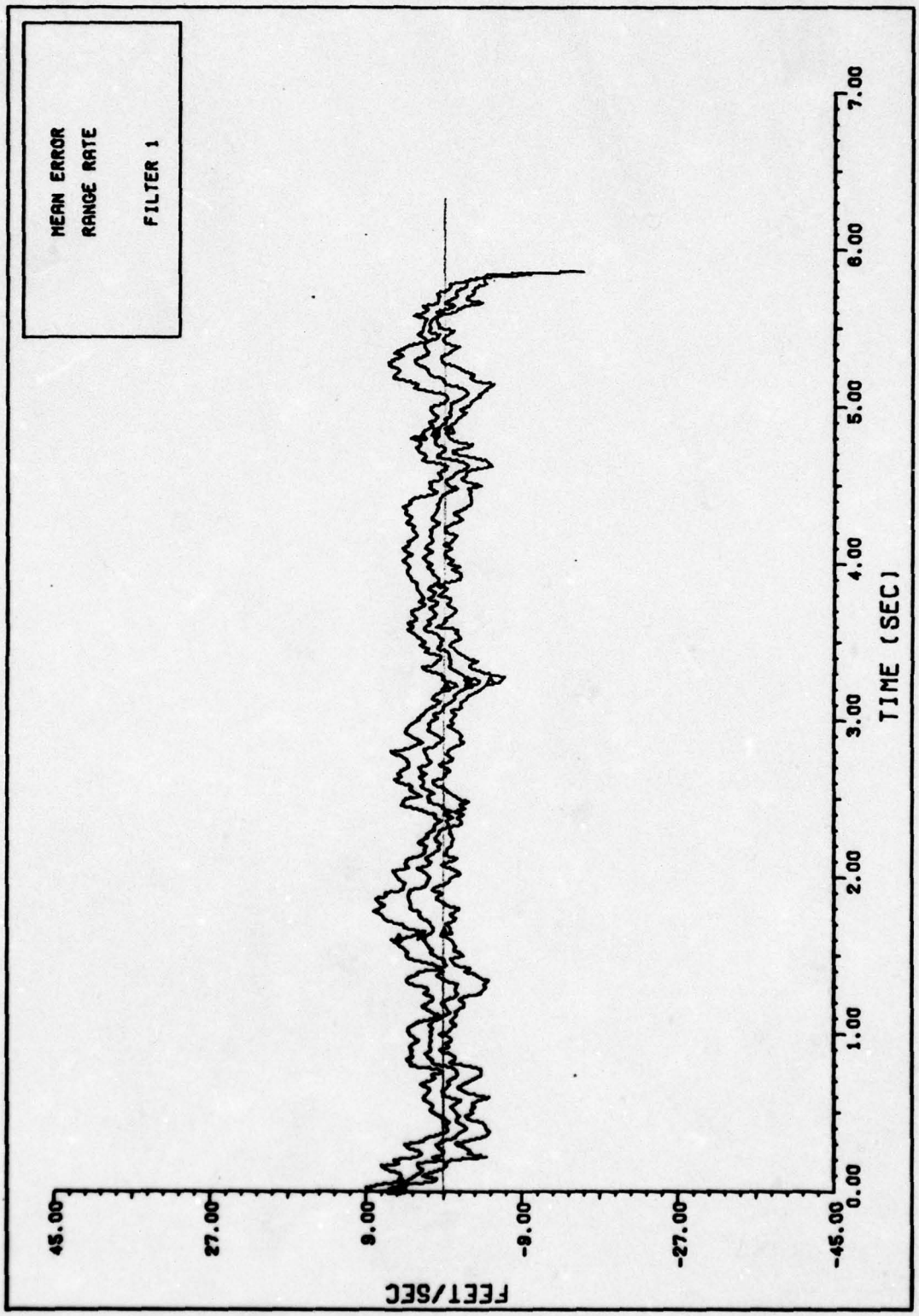


FIGURE 26. RANGE RATE PERFORMANCE FILTER 1

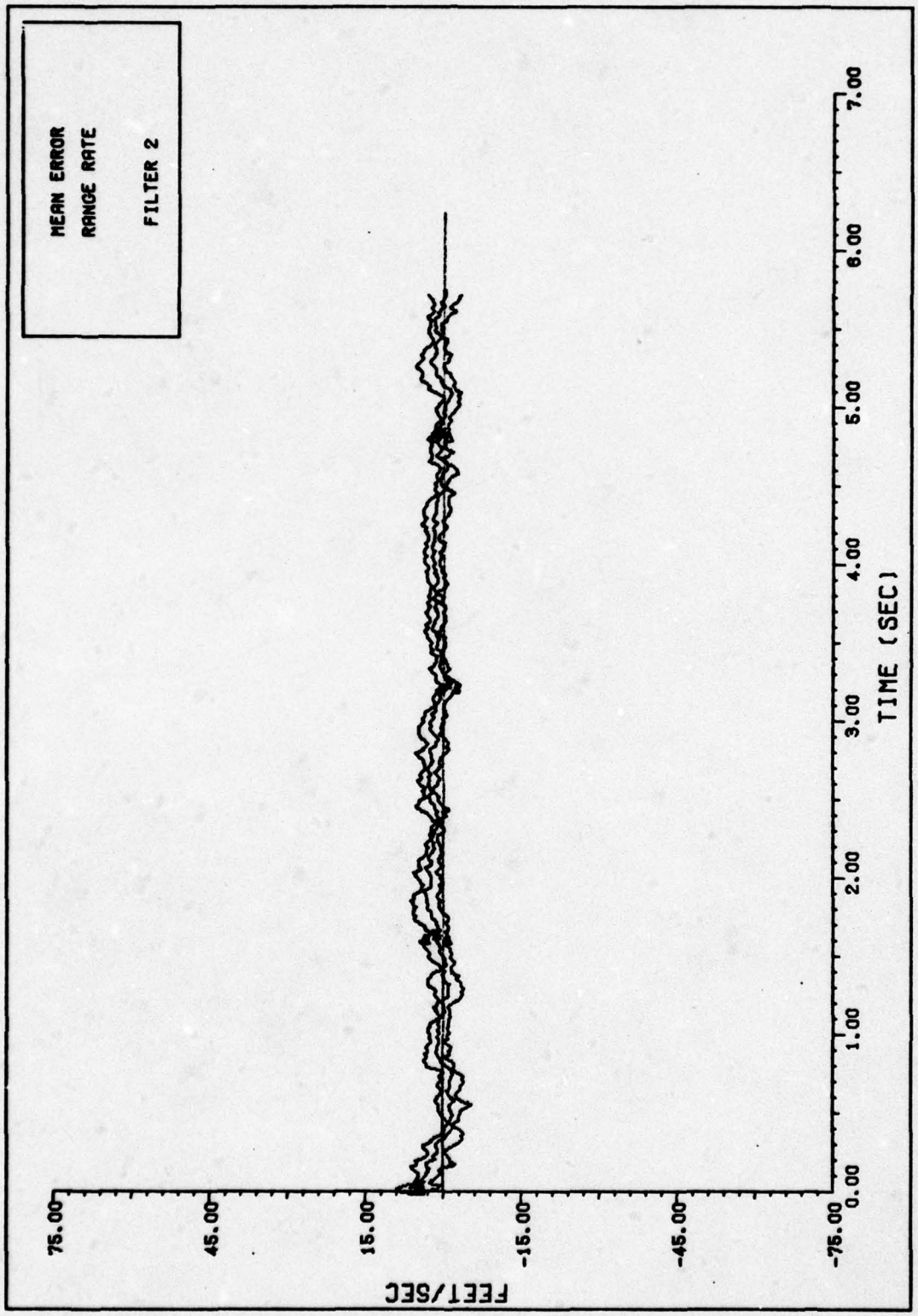


FIGURE 27. RANGE RATE PERFORMANCE FILTER 2

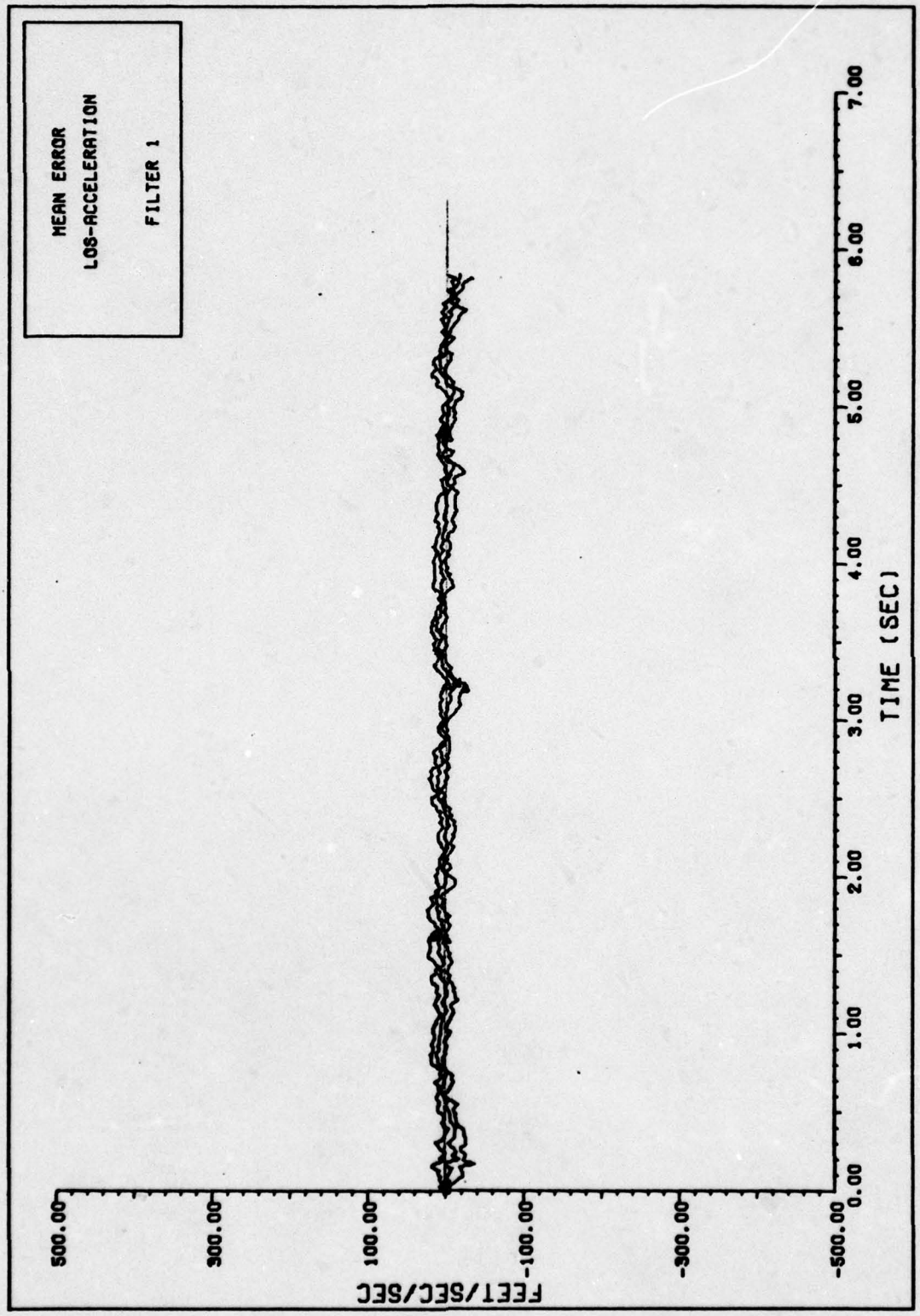


FIGURE 28. LOS-ACCELERATION PERFORMANCE FILTER 1

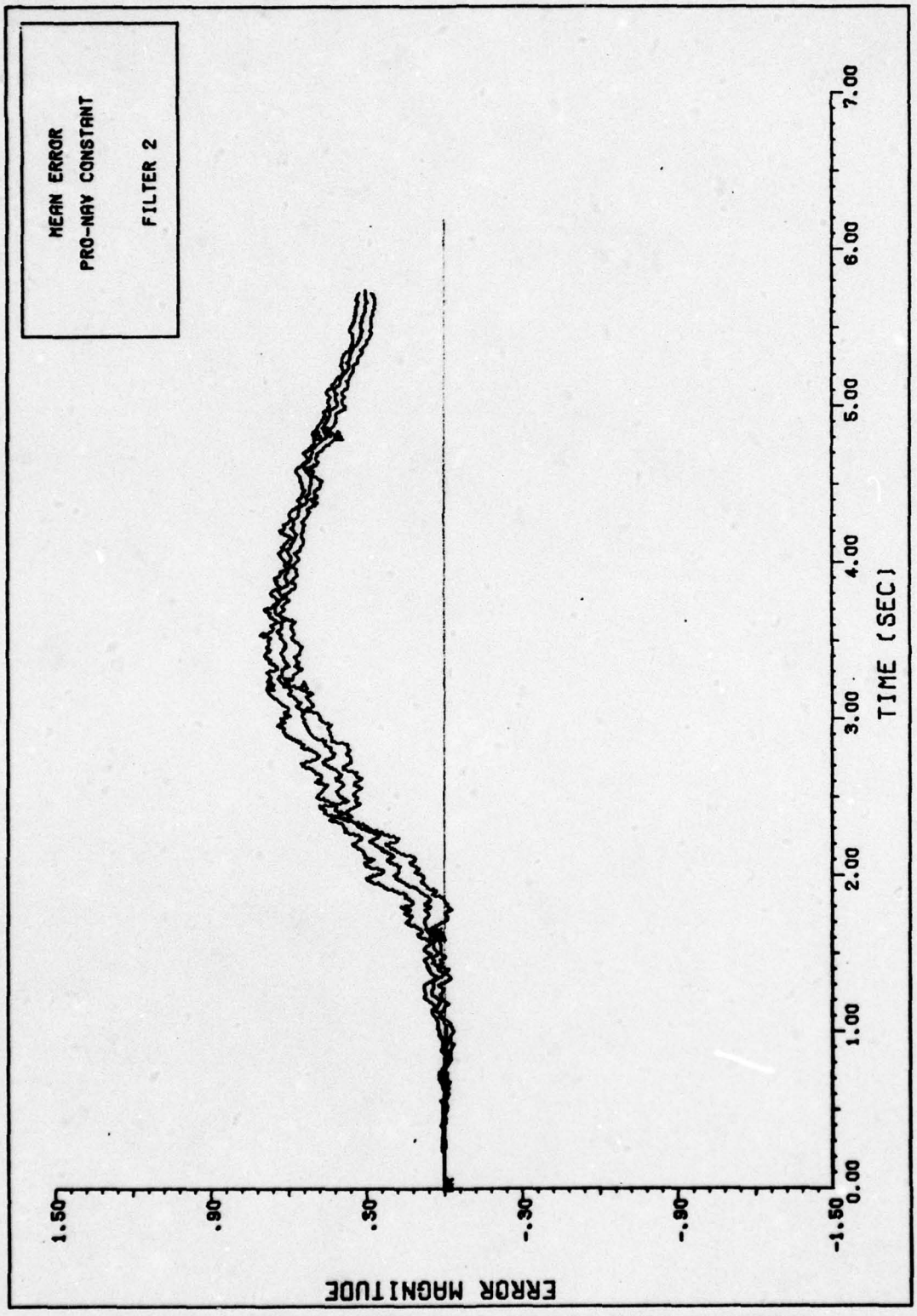


FIGURE 29. PRO-NAV CONSTANT PERFORMANCE FILTER 2

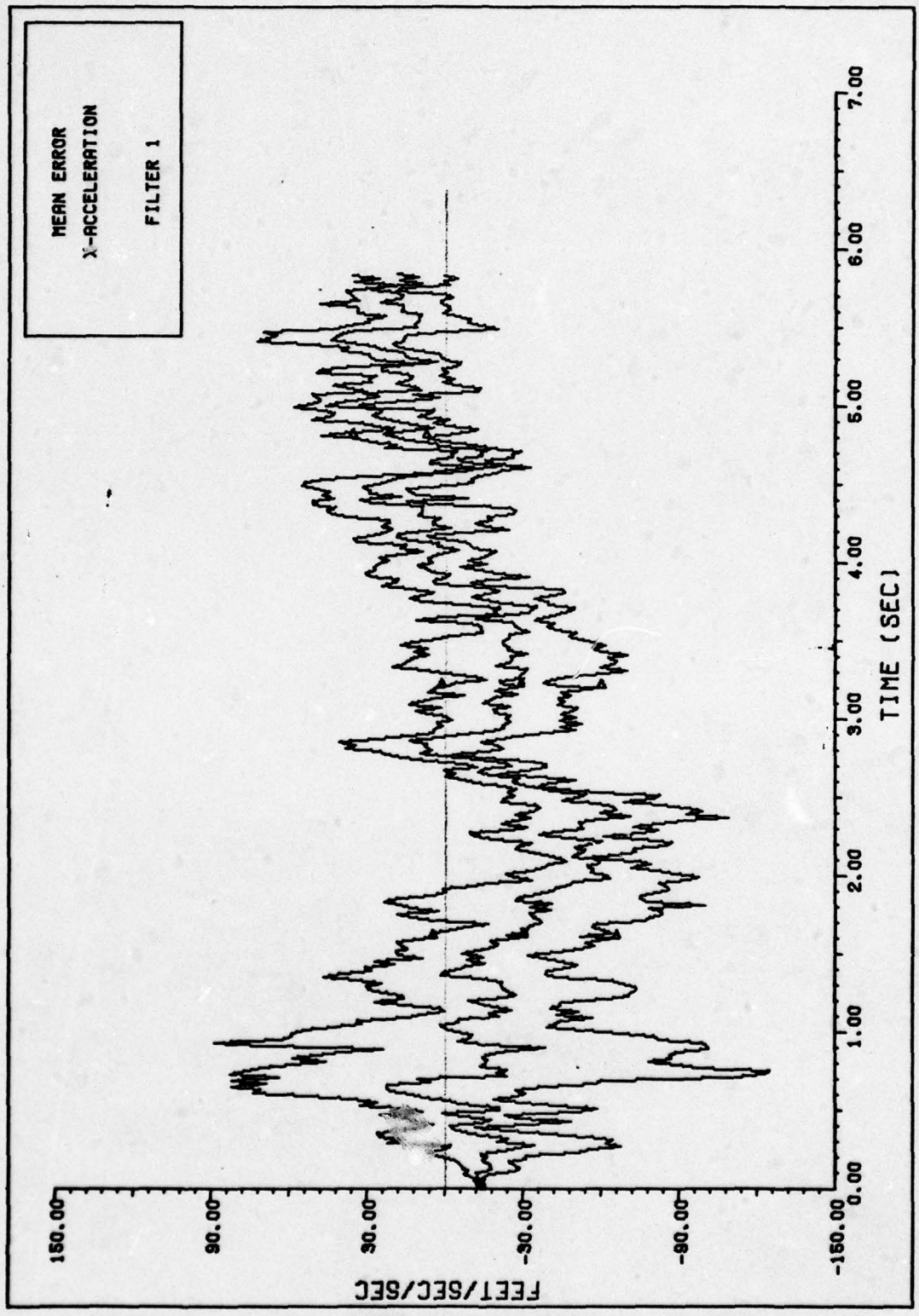


FIGURE 30. X-ACCELERATION PERFORMANCE FILTER 1

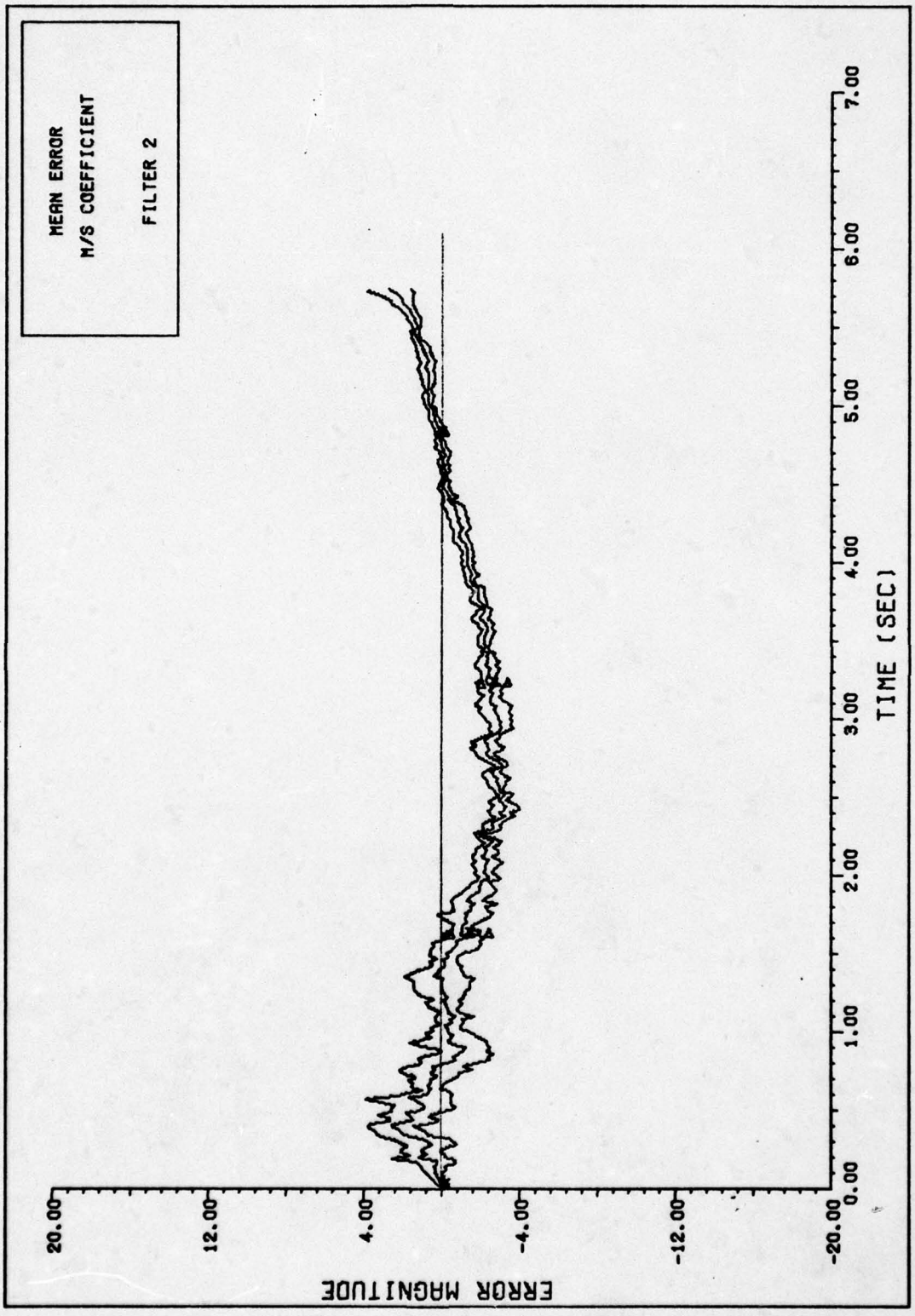


FIGURE 31. M/S COEFFICIENT PERFORMANCE FILTER 2

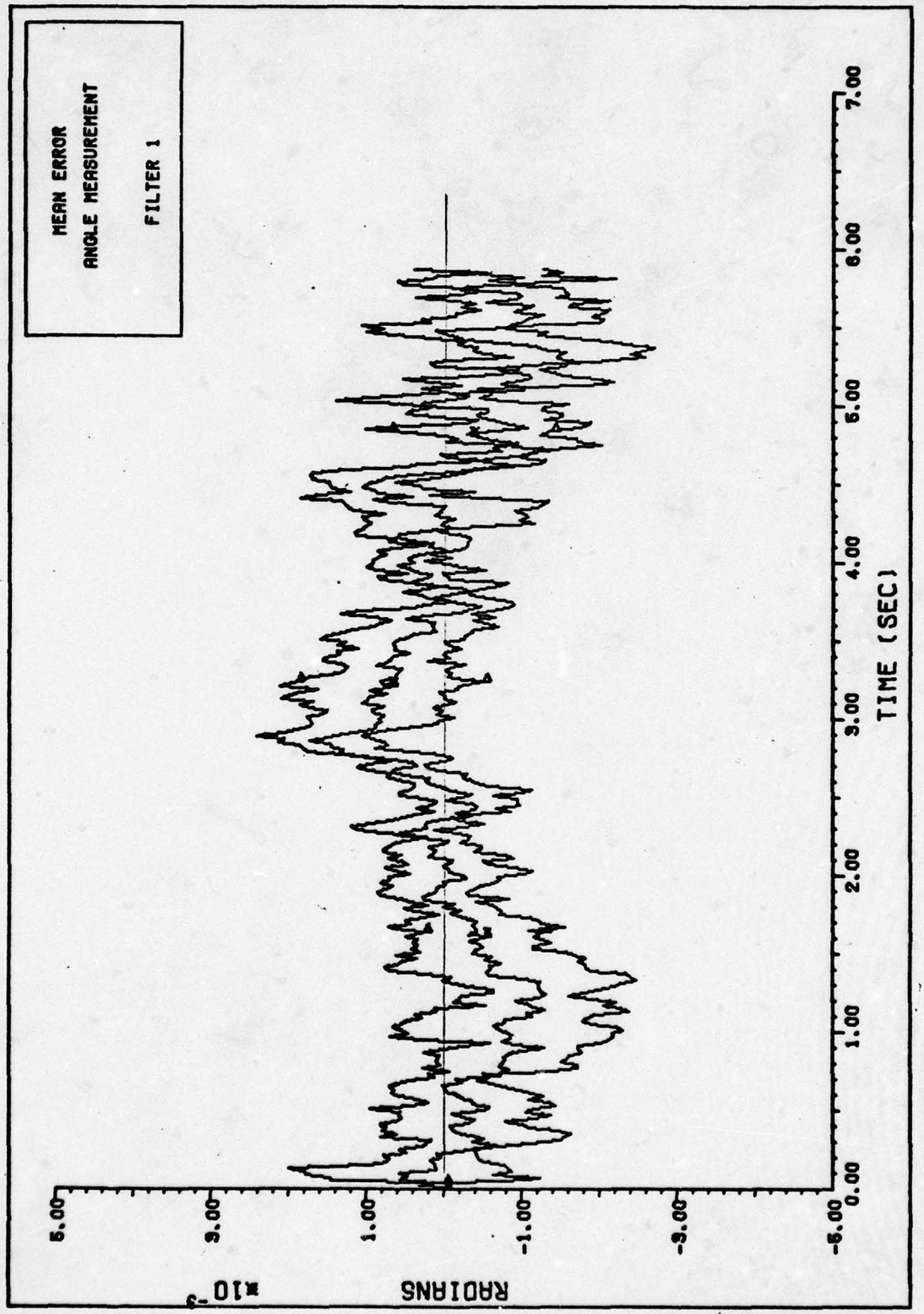


FIGURE 32. ANGLE MEASUREMENT PERFORMANCE FILTER 1

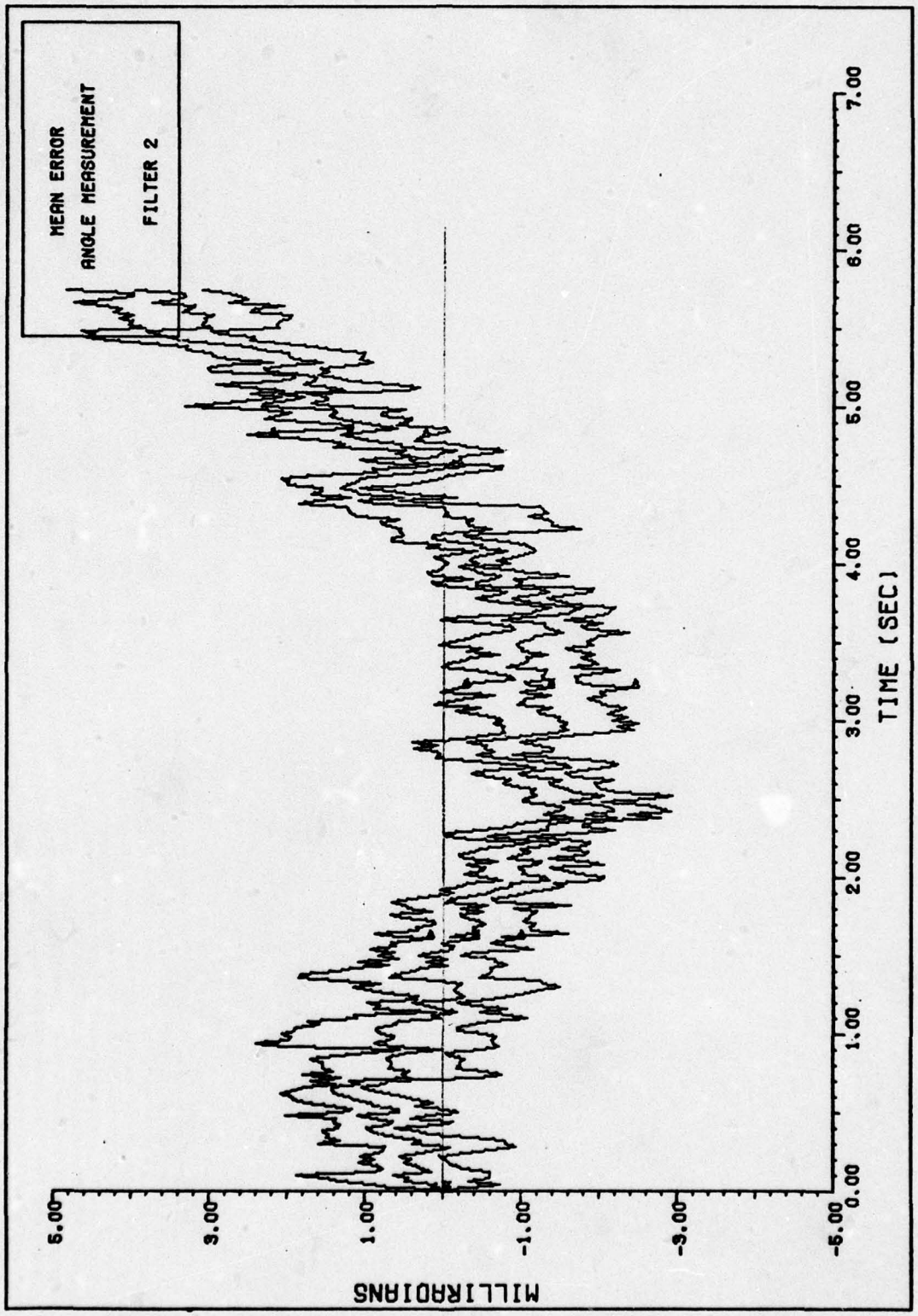


FIGURE 33. ANGLE MEASUREMENT PERFORMANCE FILTER 2

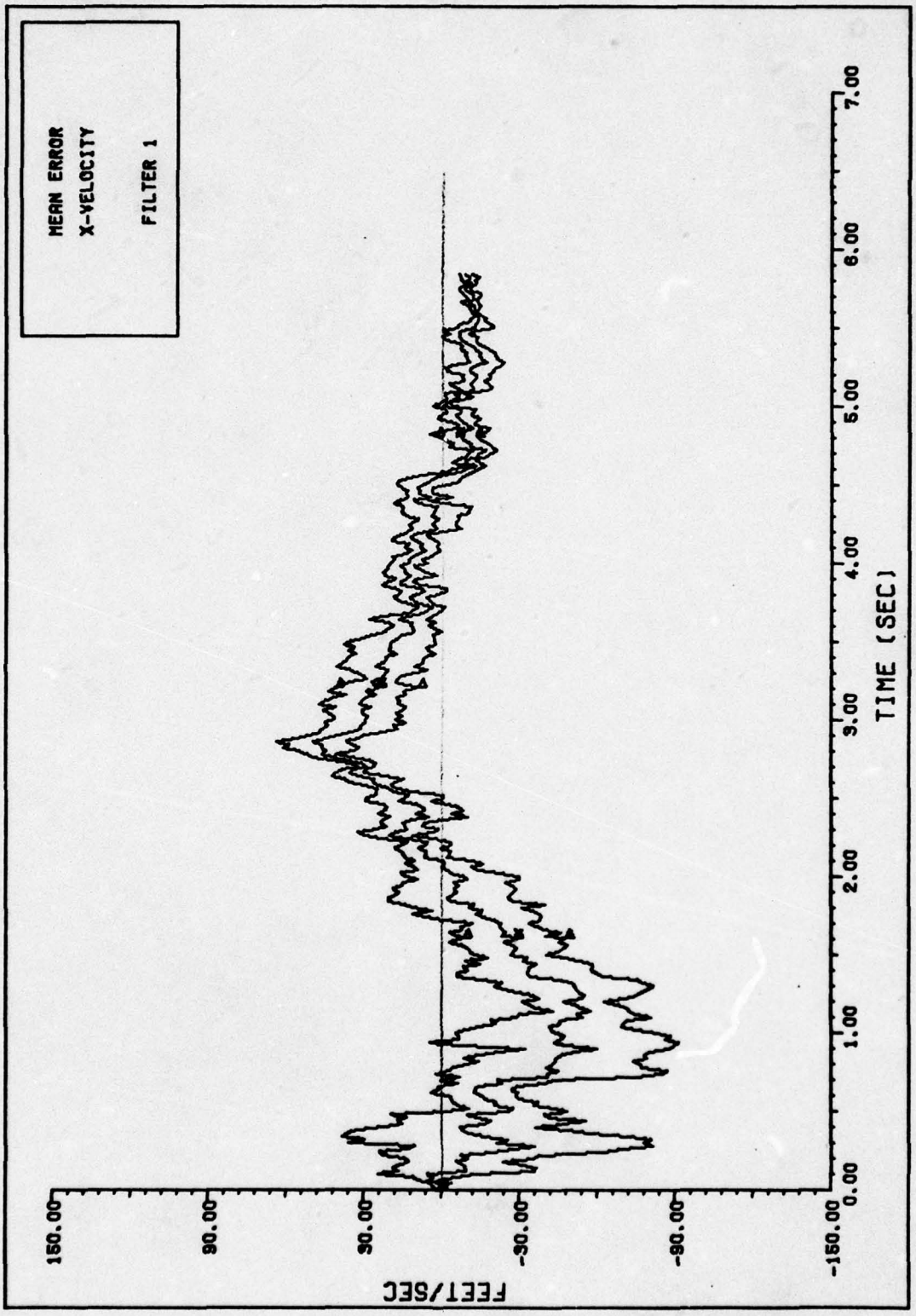


FIGURE 34. X-VELOCITY PERFORMANCE FILTER 1

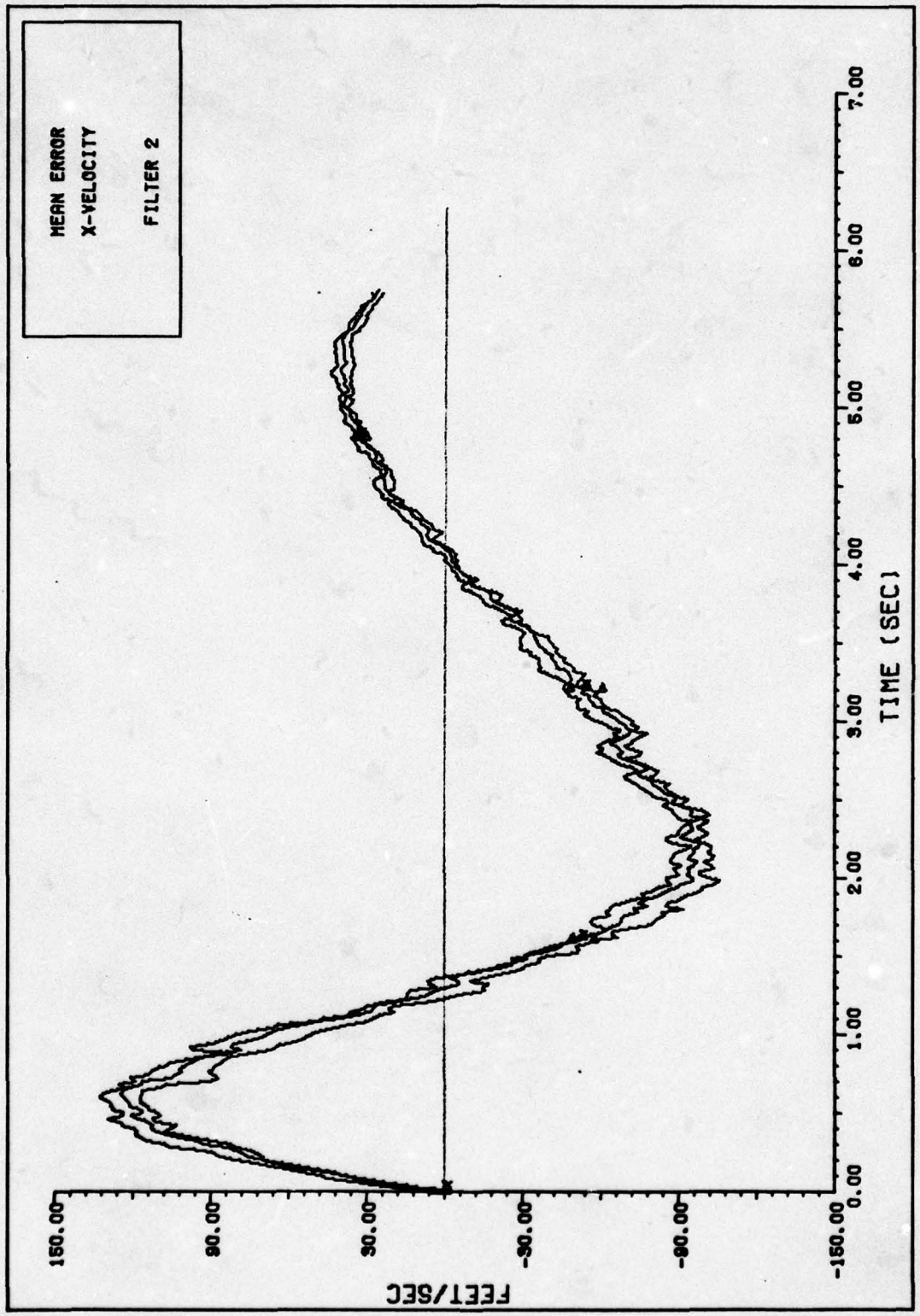


FIGURE 35. X-VELOCITY PERFORMANCE FILTER 2

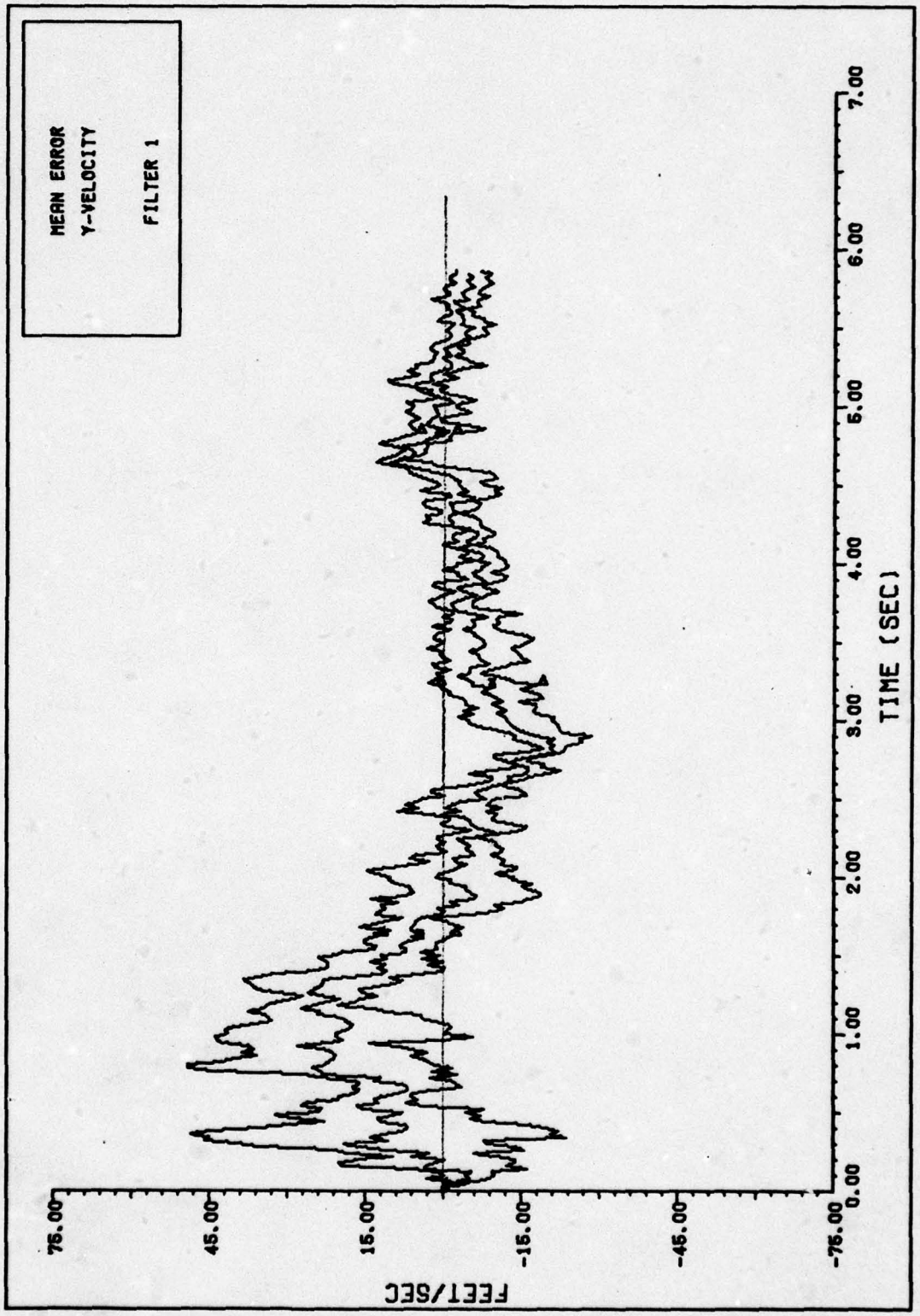


FIGURE 36. Y-VELOCITY PERFORMANCE FILTER 1

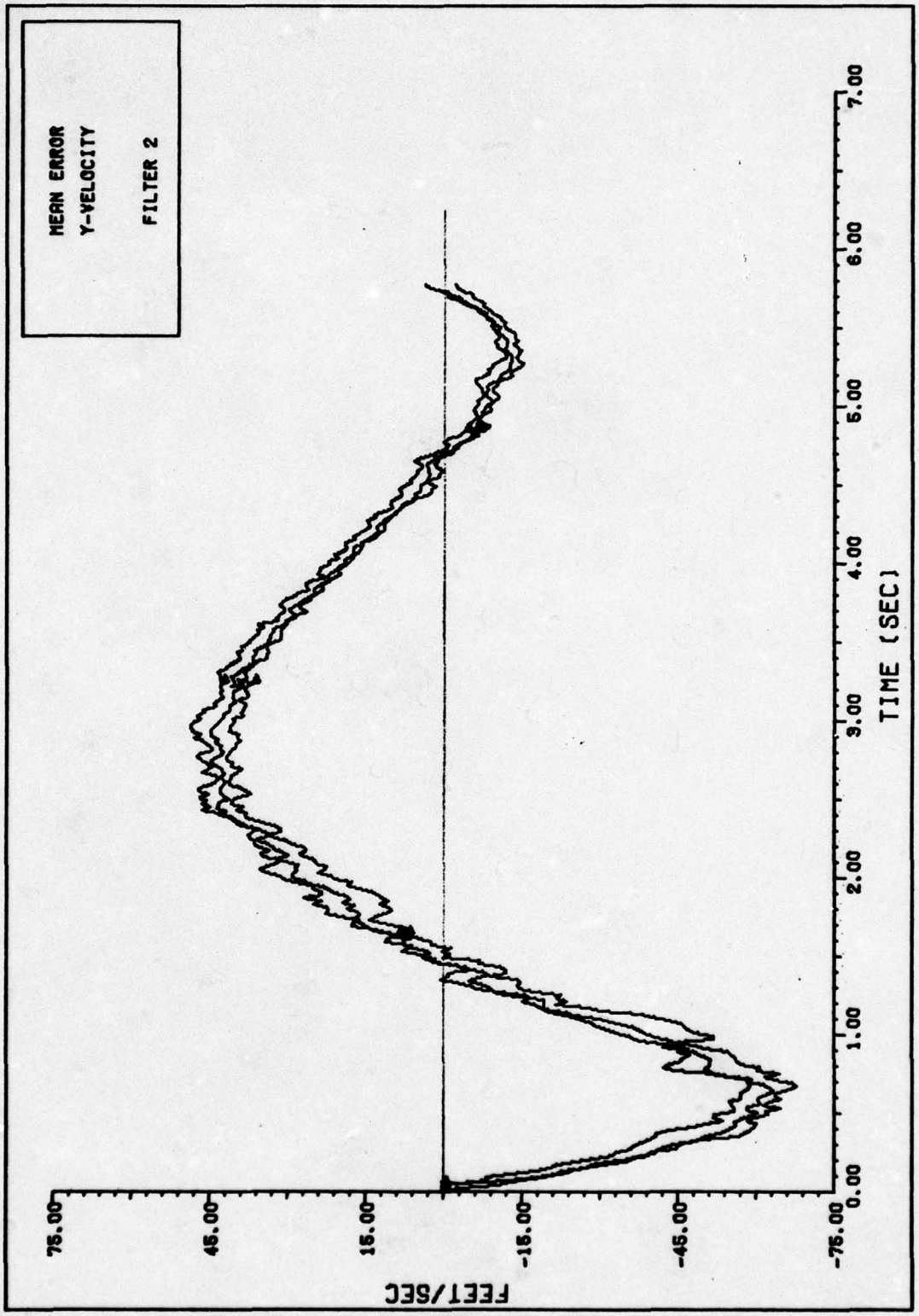


FIGURE 37. Y-VELOCITY PERFORMANCE FILTER 2

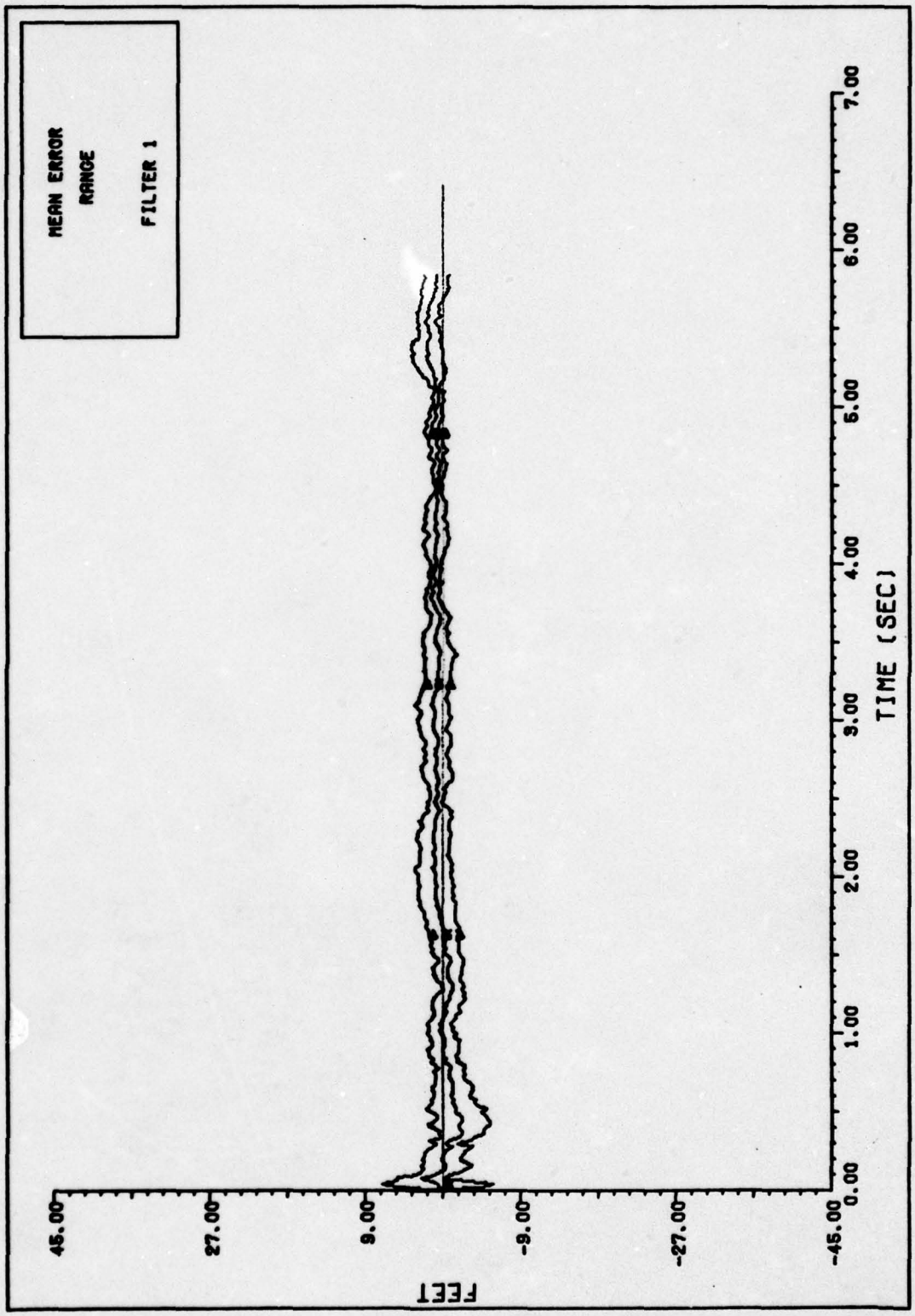


FIGURE 38. RANGE PERFORMANCE FILTER 1

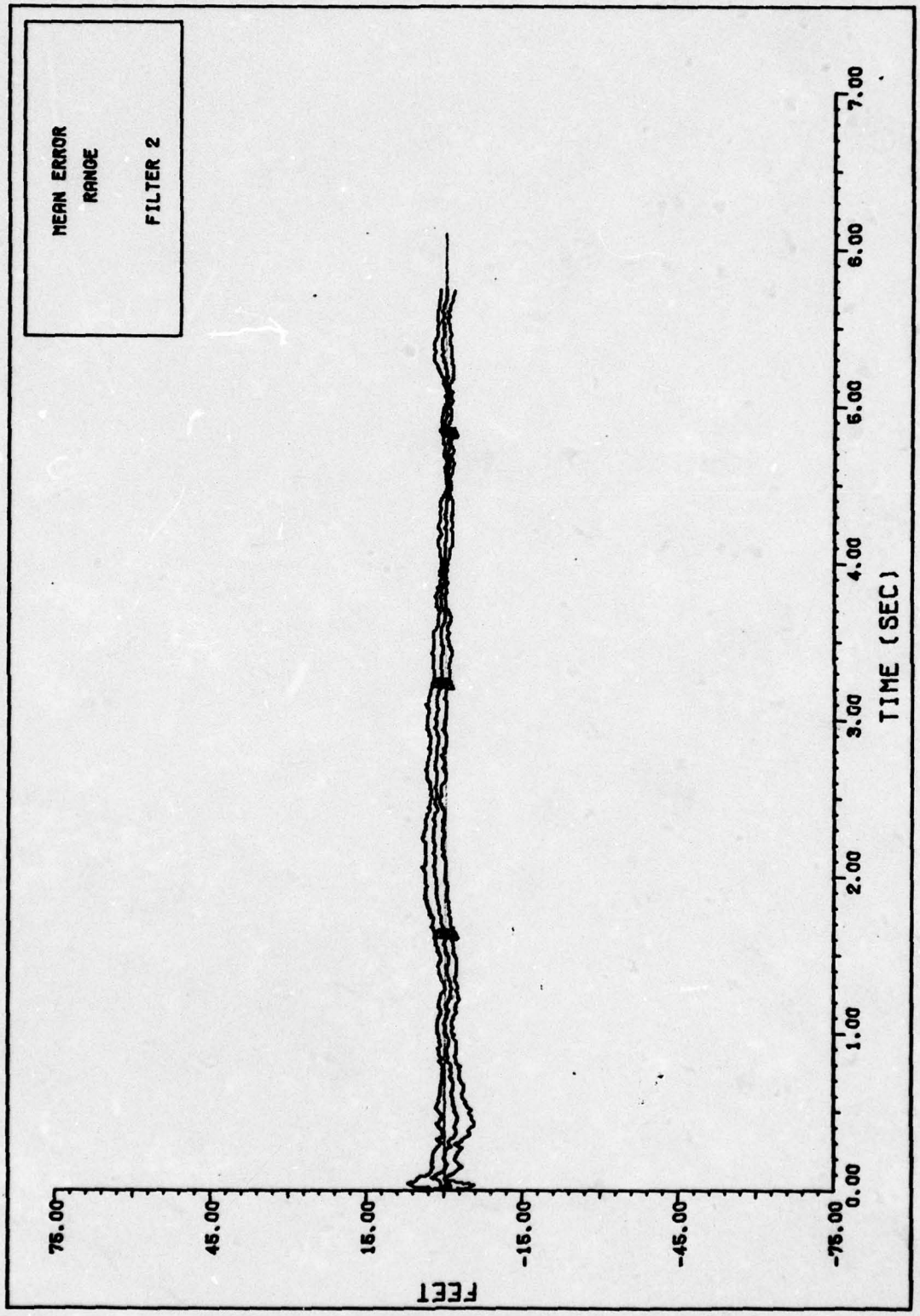


FIGURE 39. RANGE PERFORMANCE FILTER.2

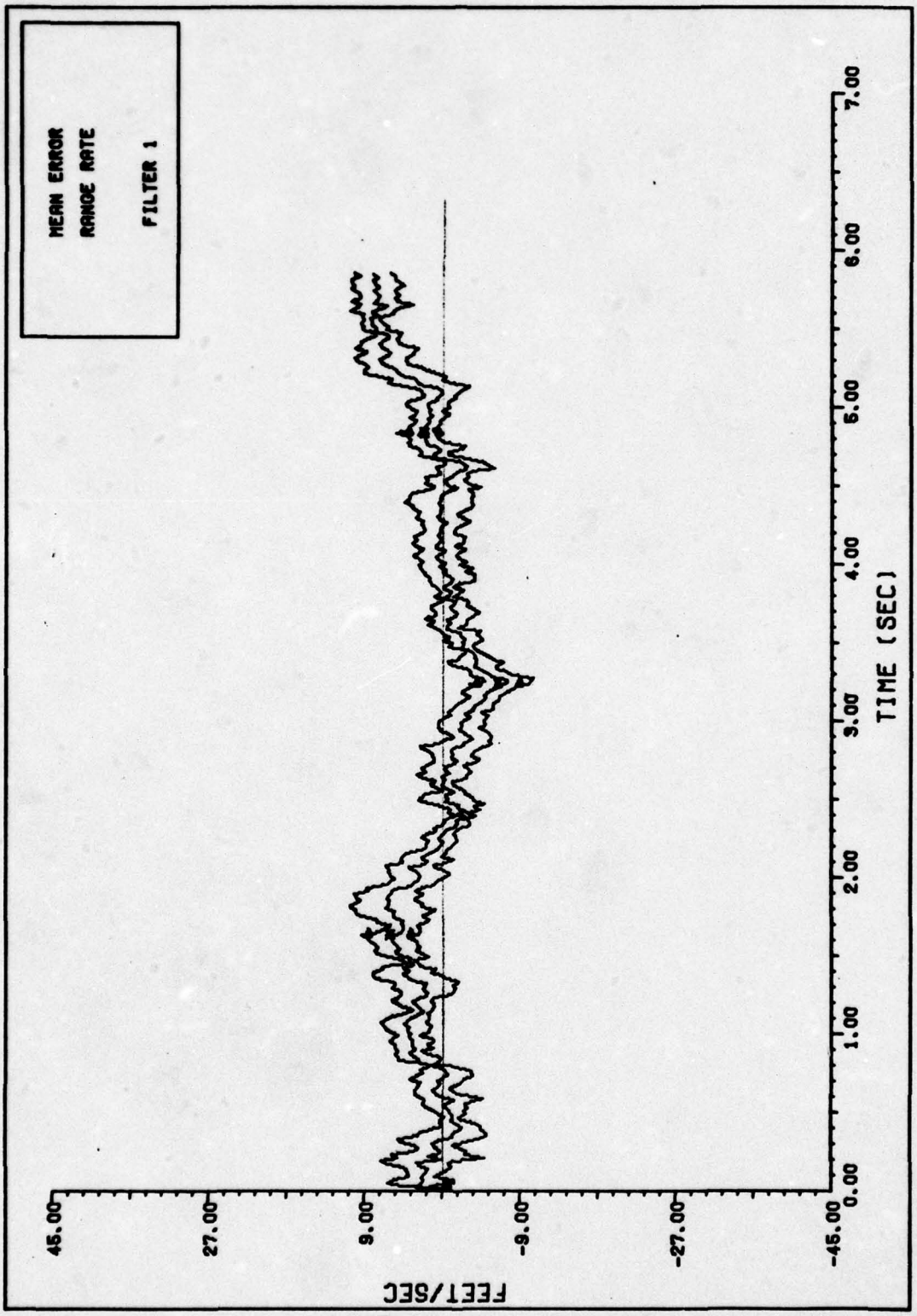


FIGURE 40. RANGE RATE PERFORMANCE FILTER 1

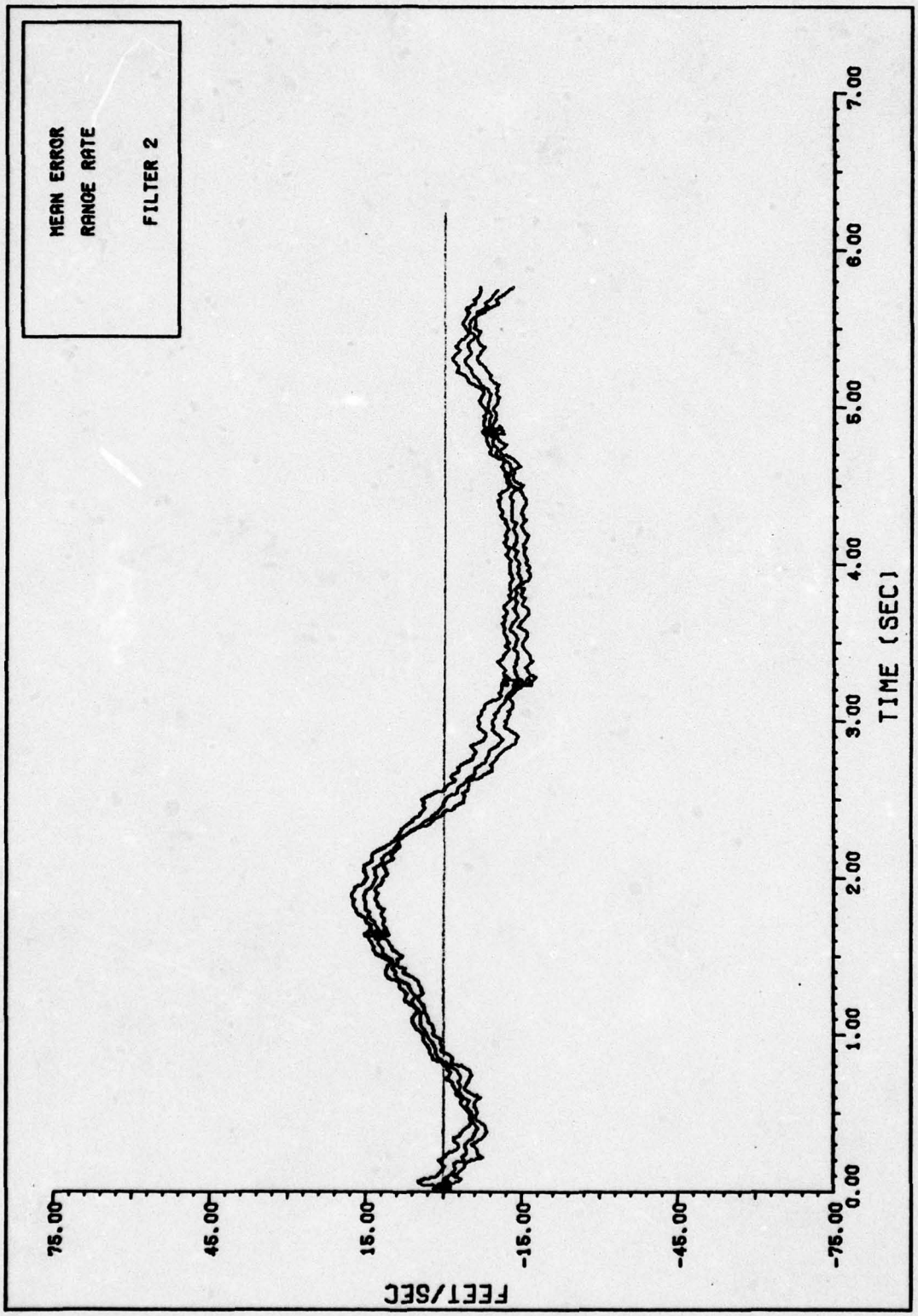


FIGURE 41. RANGE RATE PERFORMANCE FILTER 2

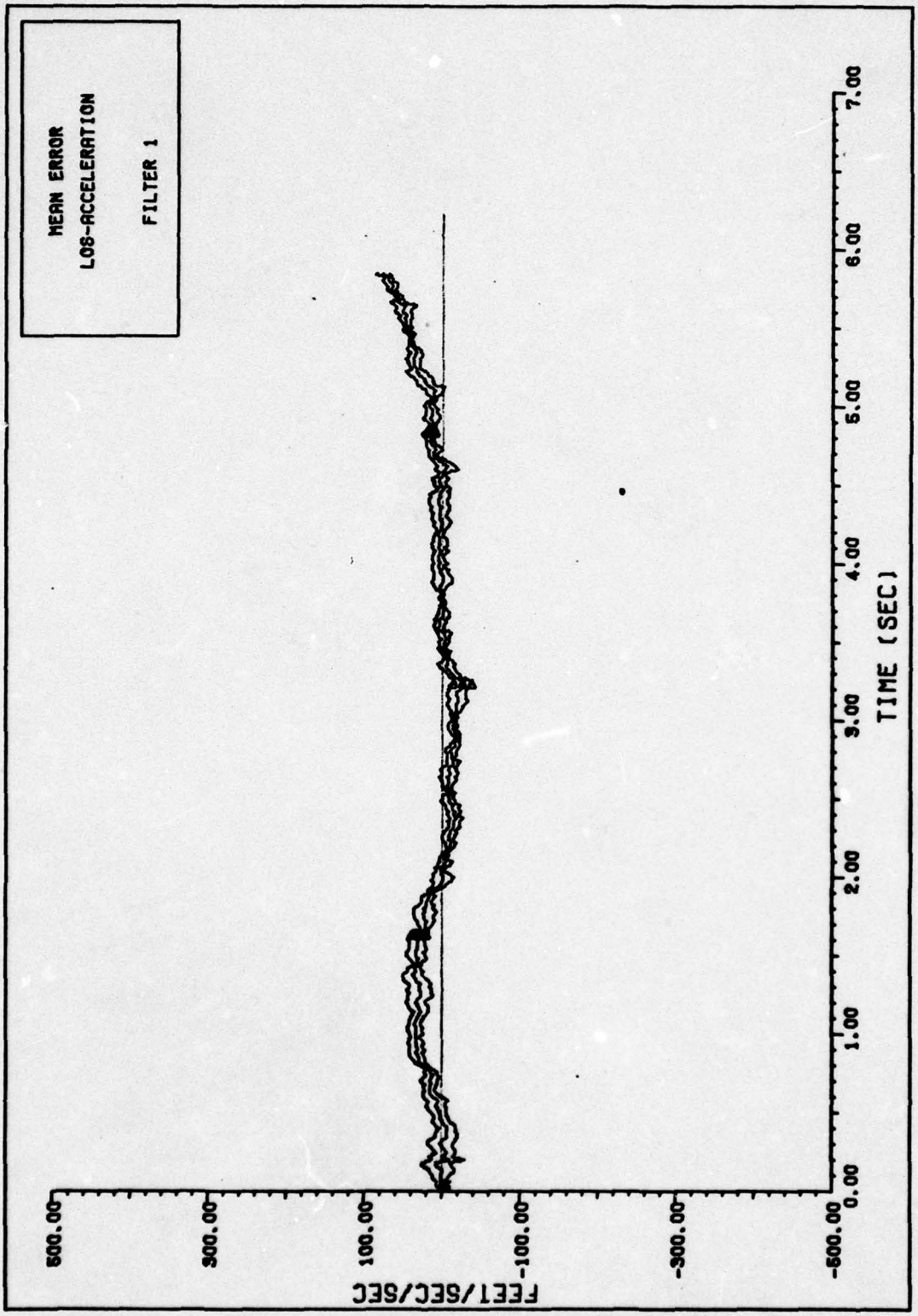


FIGURE 42. LOS-ACCELERATION PERFORMANCE FILTER 1

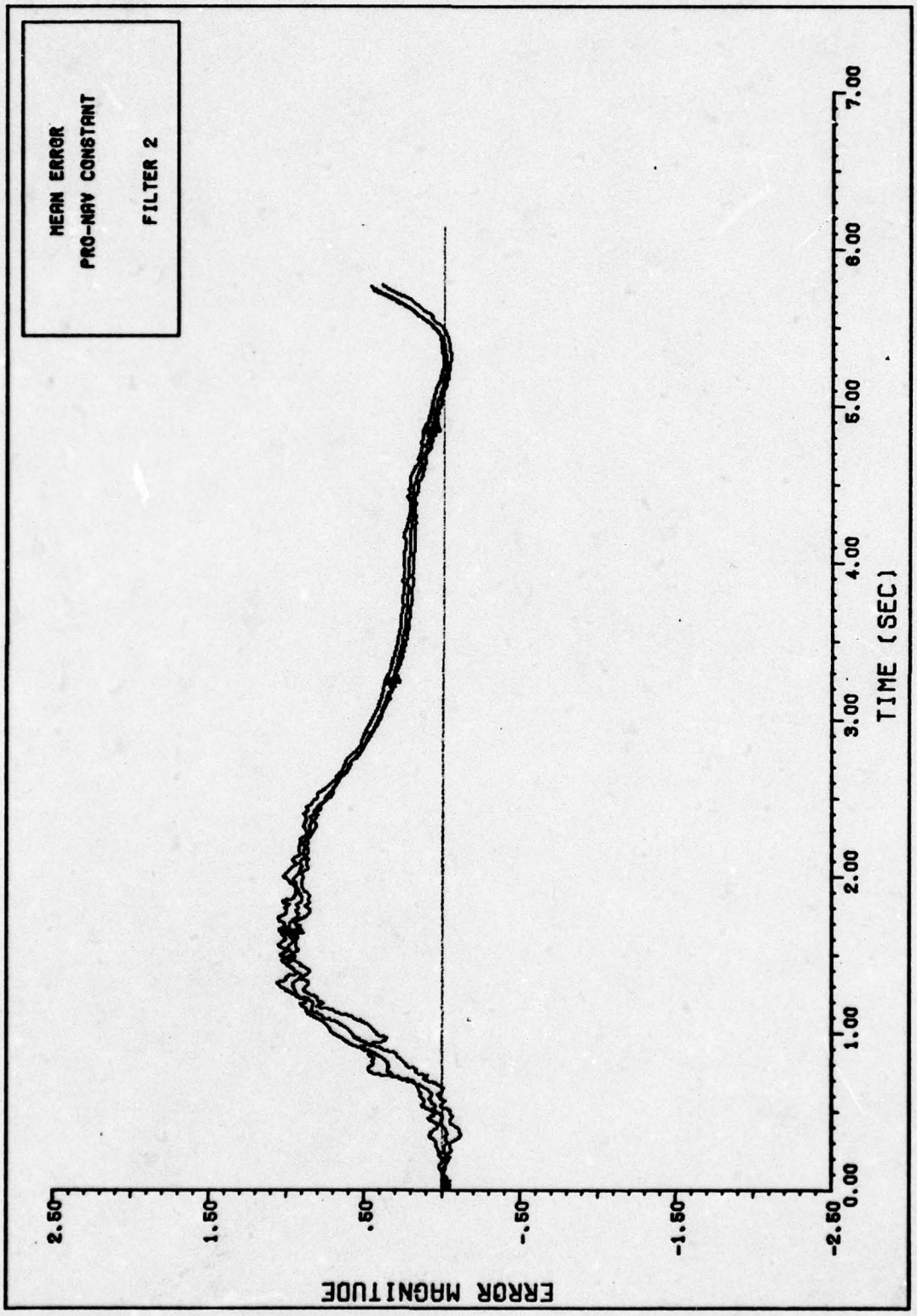


FIGURE 43. PRO-NAV CONSTANT PERFORMANCE FILTER 2

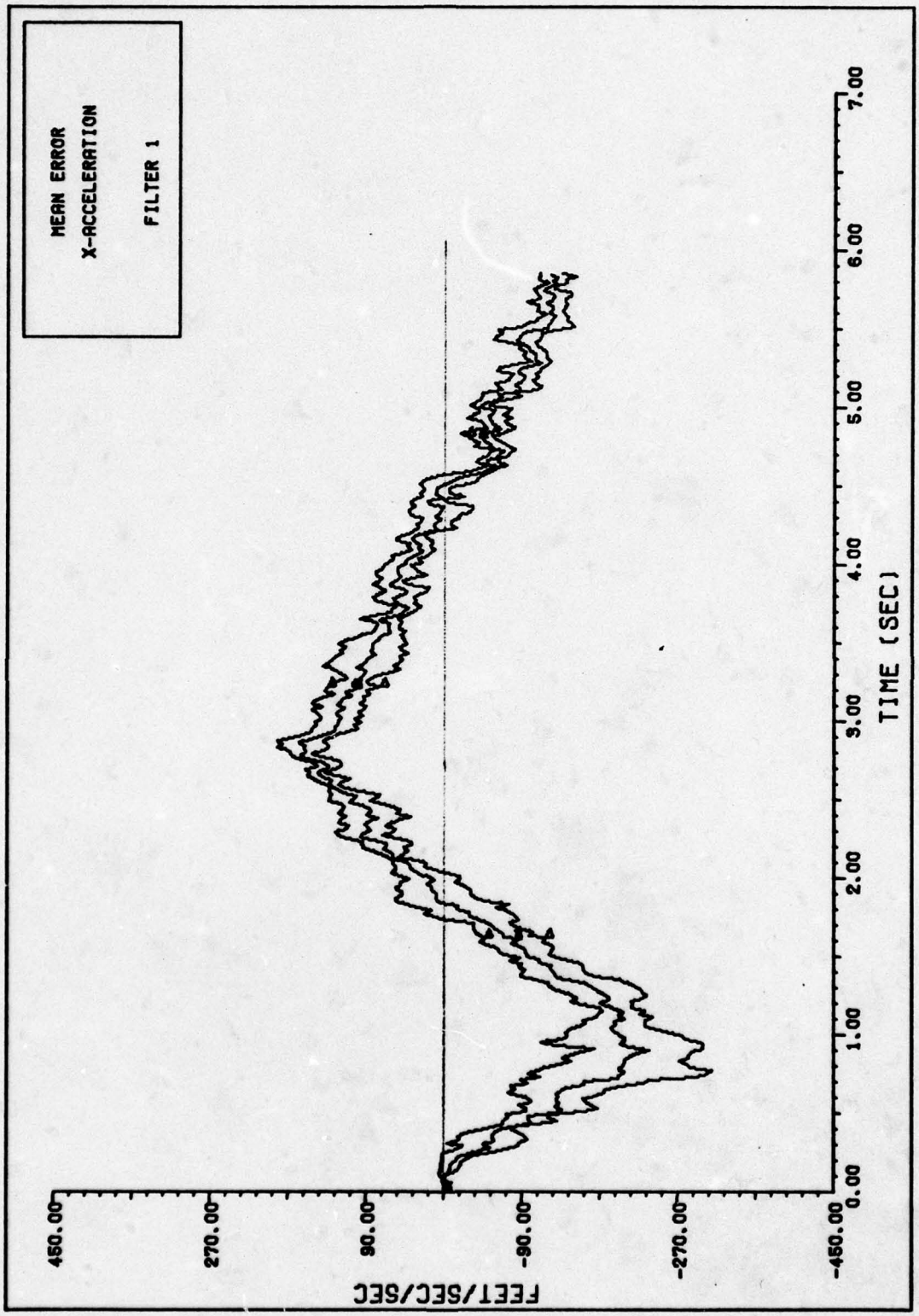


FIGURE 44. X--ACCELERATION PERFORMANCE FILTER 1

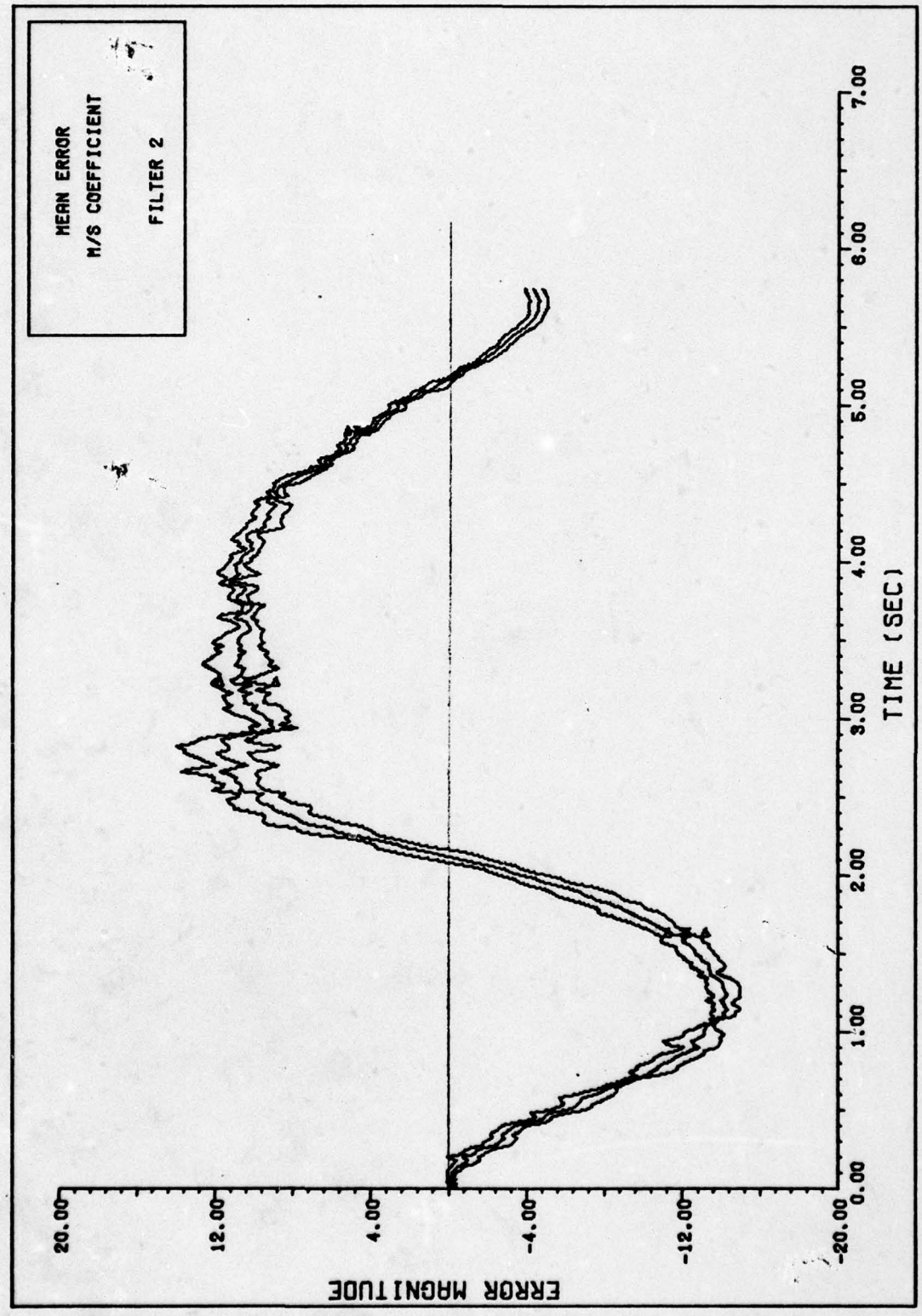


FIGURE 45. M/S COEFFICIENT PERFORMANCE FILTER 2

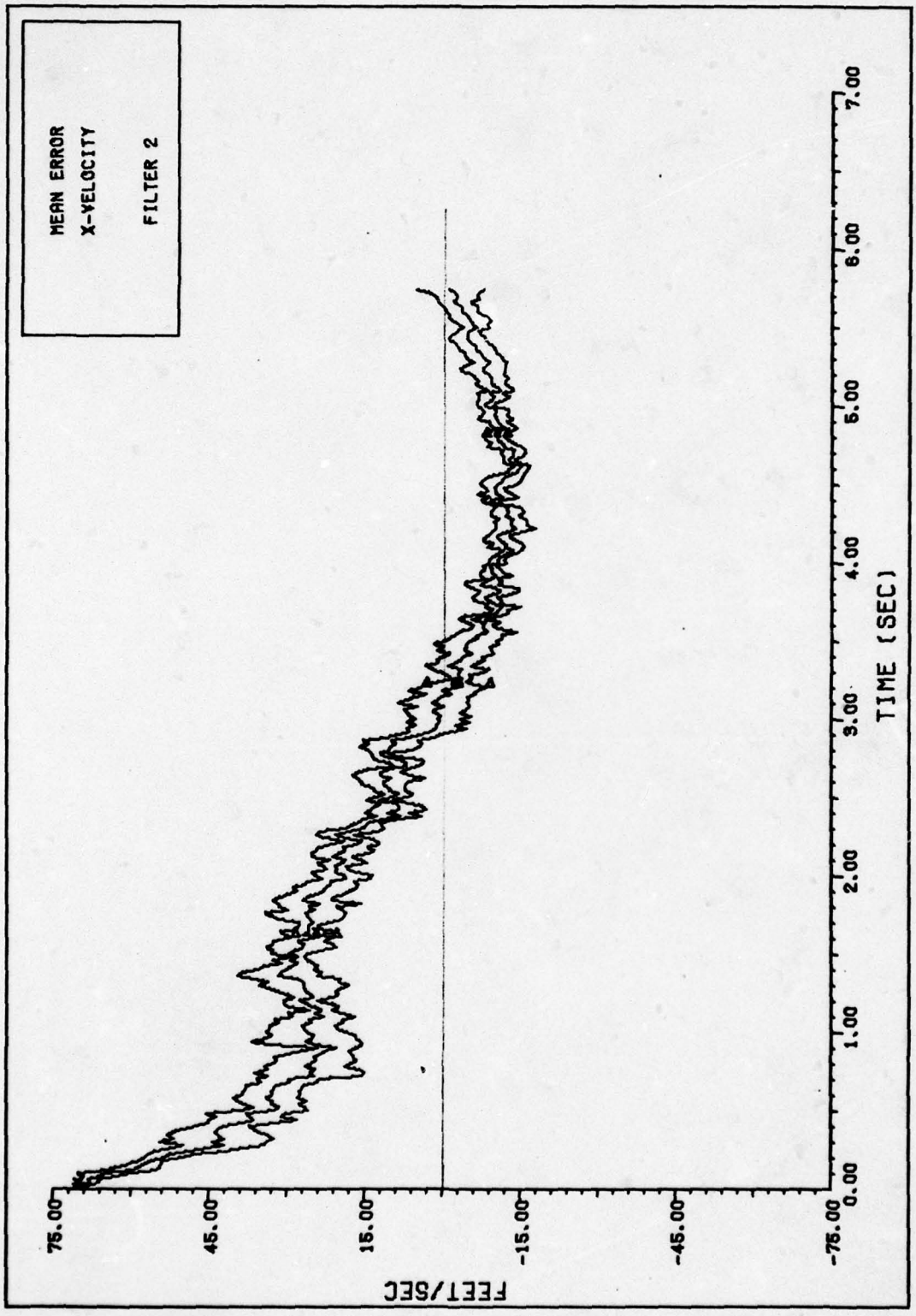


FIGURE 46. X-VELOCITY PERFORMANCE FILTER 2

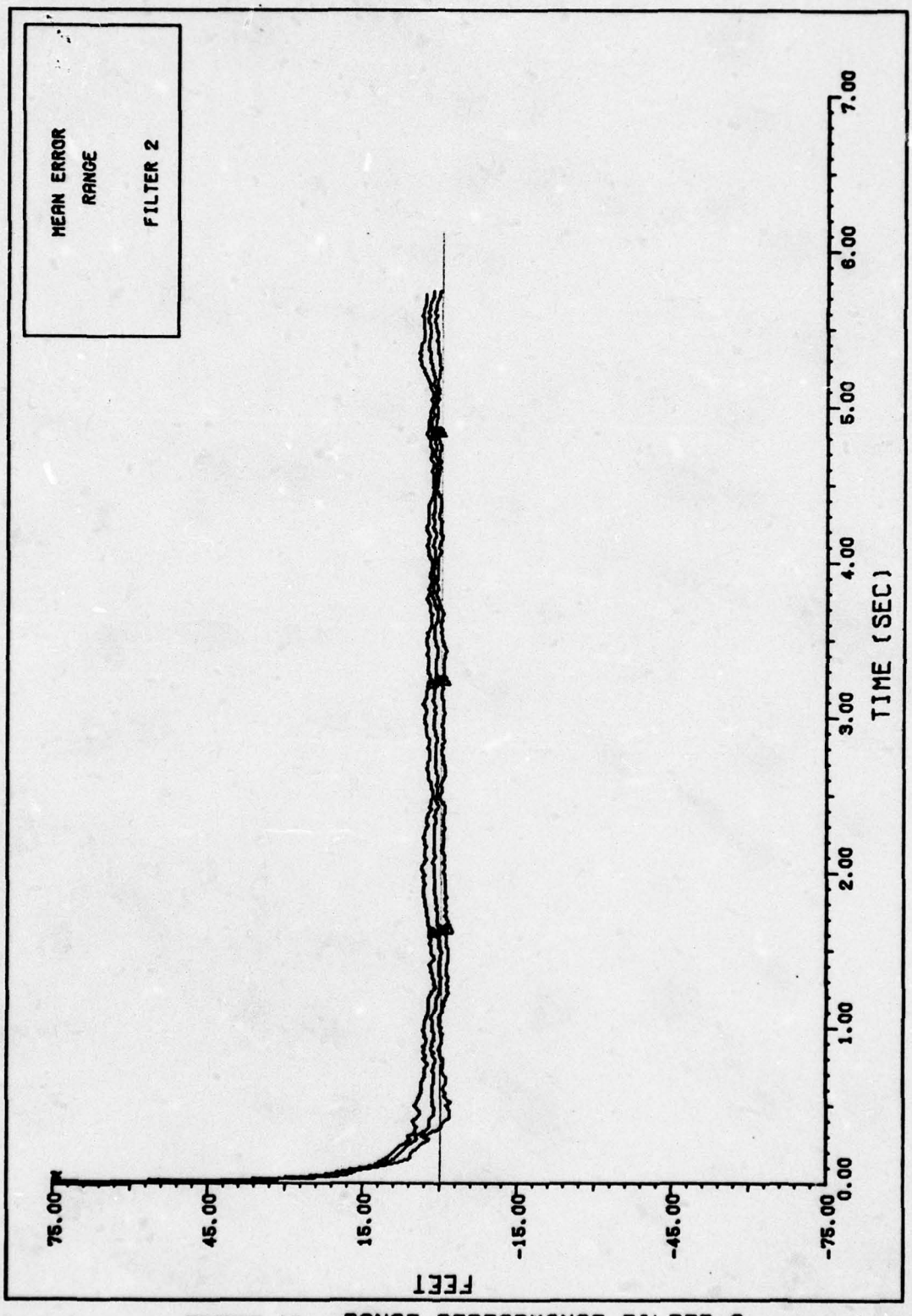


FIGURE 47. RANGE PERFORMANCE FILTER 2

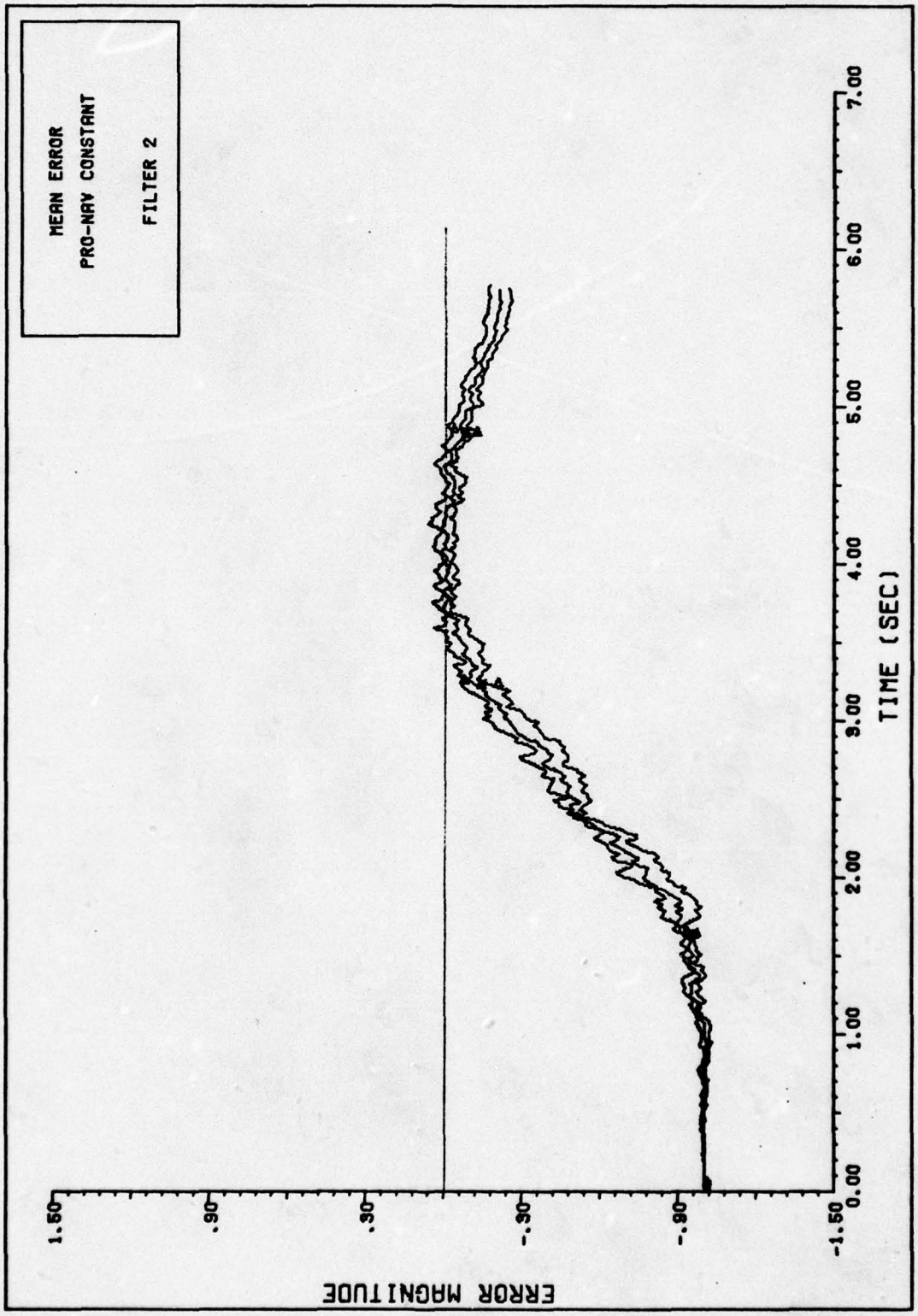


FIGURE 48. PRO-NAV CONSTANT PERFORMANCE FILTER 2

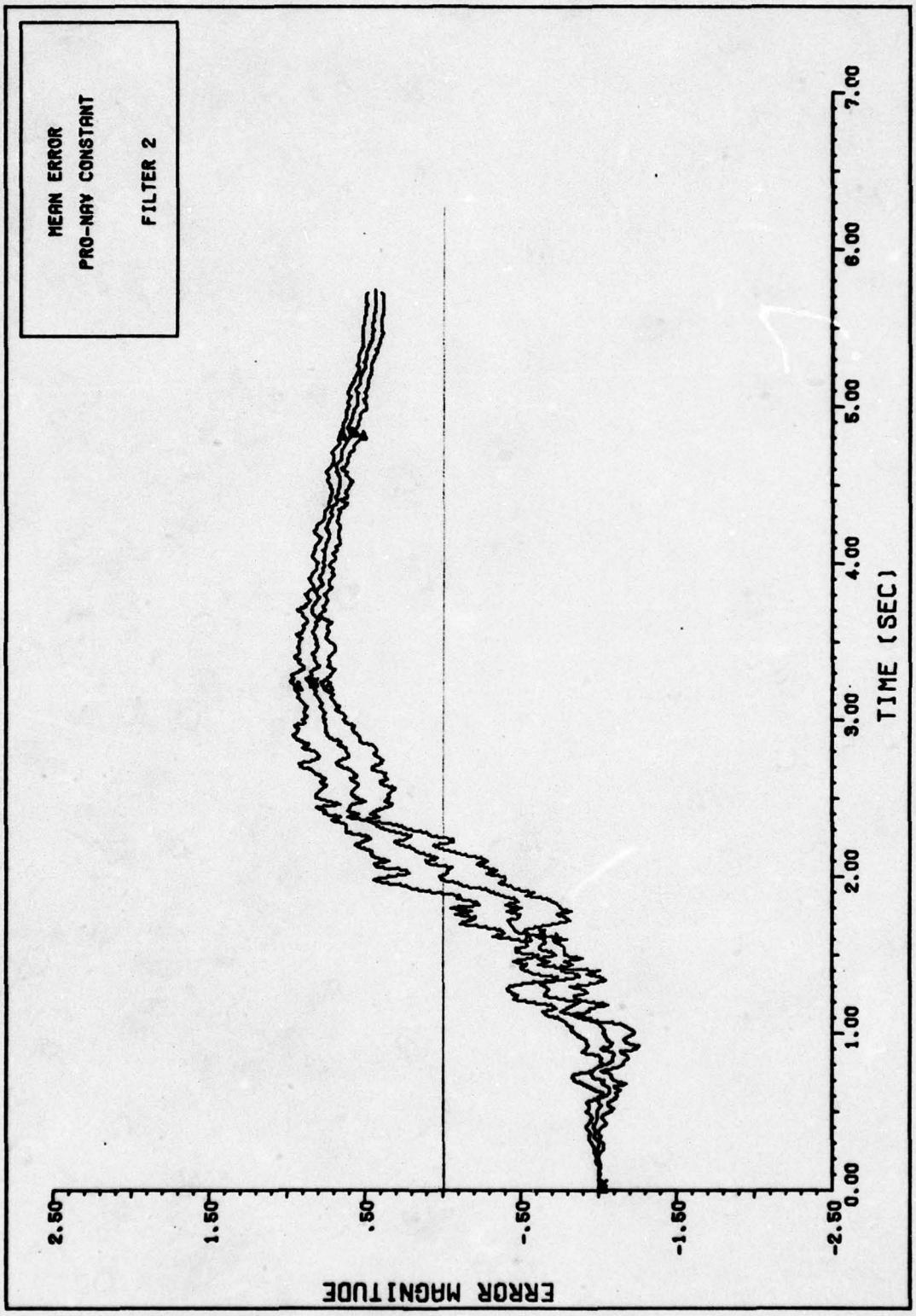


FIGURE 49. PRO-NAV CONSTANT PERFORMANCE FILTER 2

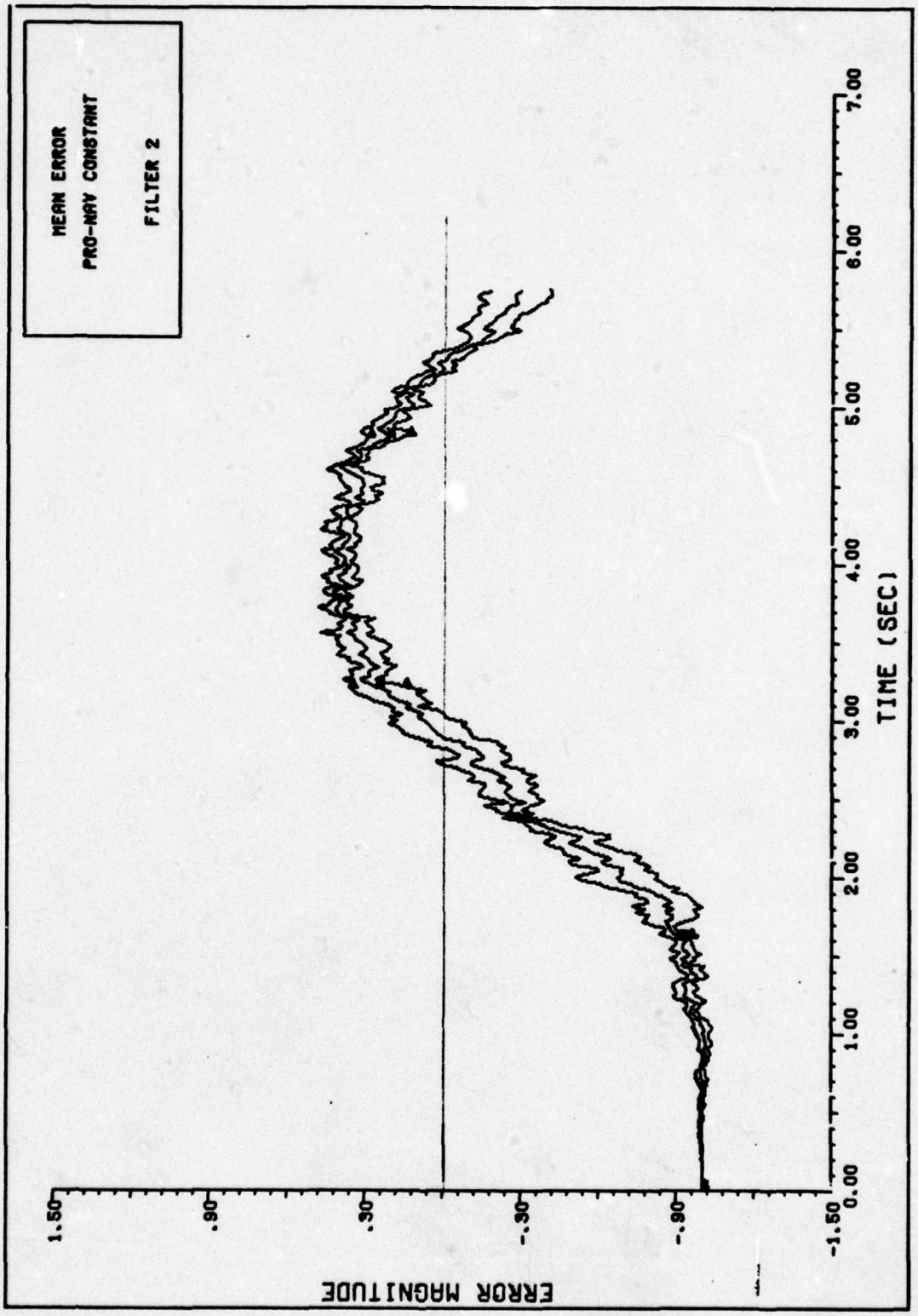


FIGURE 50. PRO-NAV CONSTANT PERFORMANCE FILTER 2

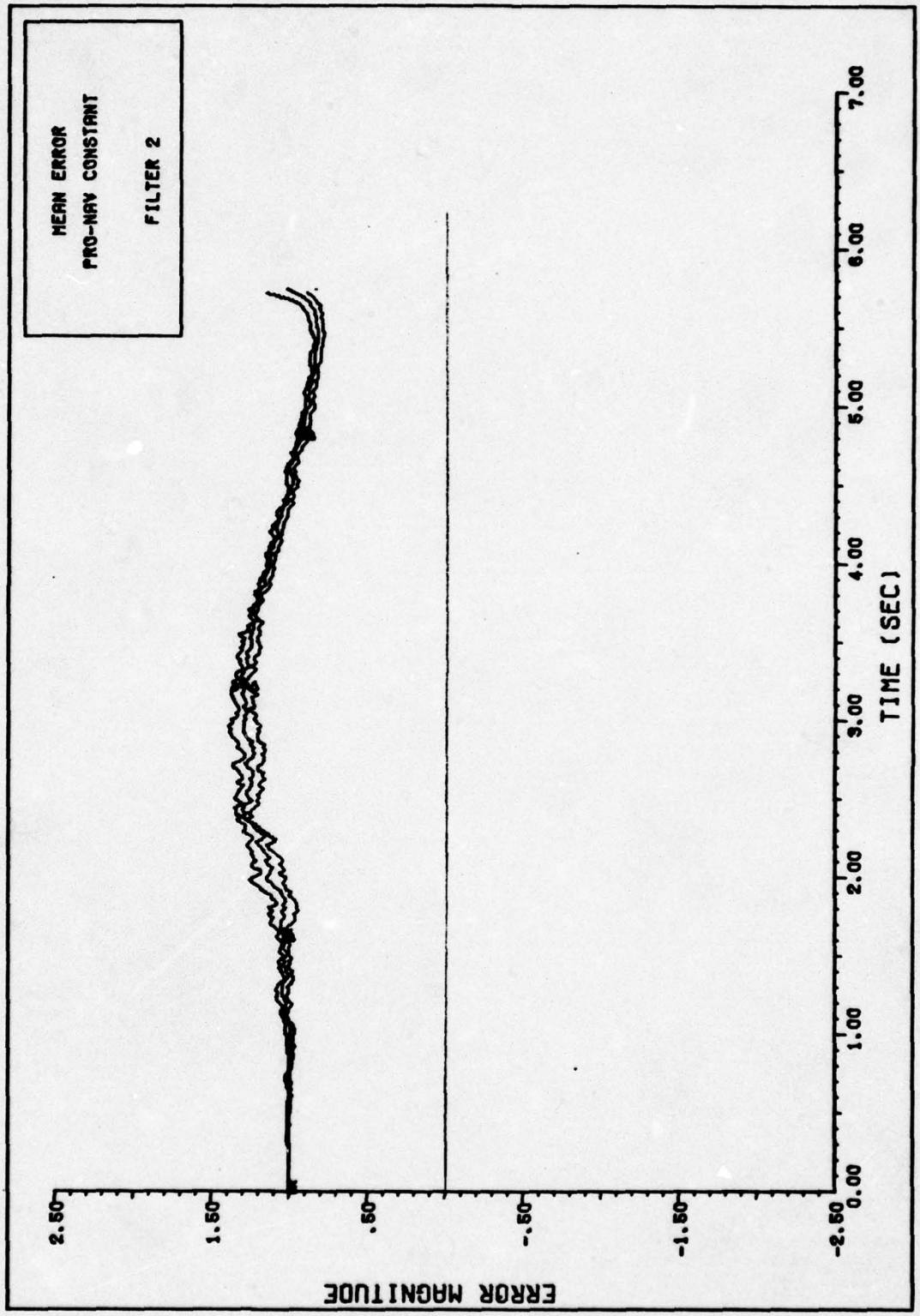


FIGURE 51. PRO-NAV CONSTANT PERFORMANCE FILTER 2

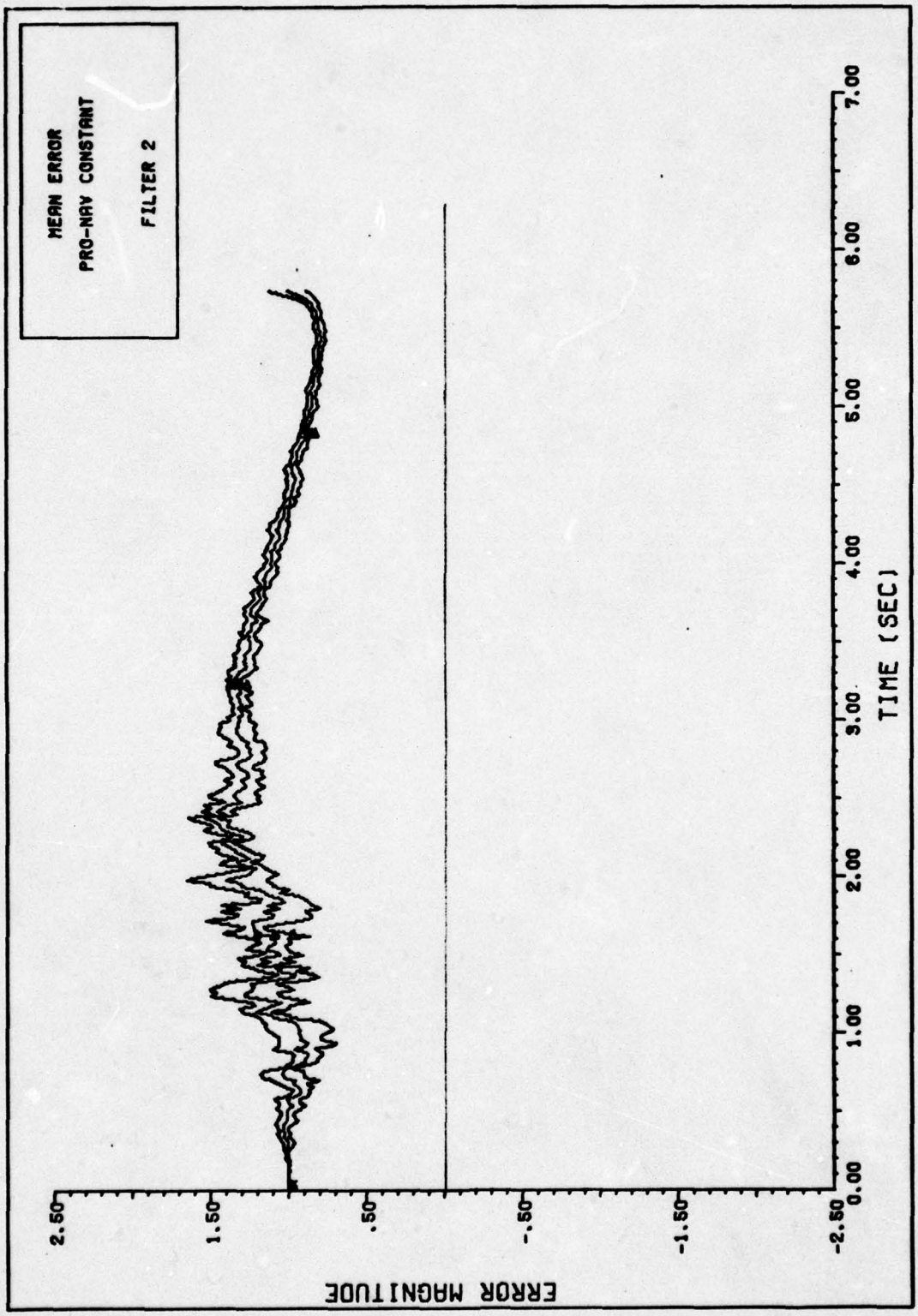


FIGURE 52. PRO-NAV CONSTANT PERFORMANCE FILTER 2

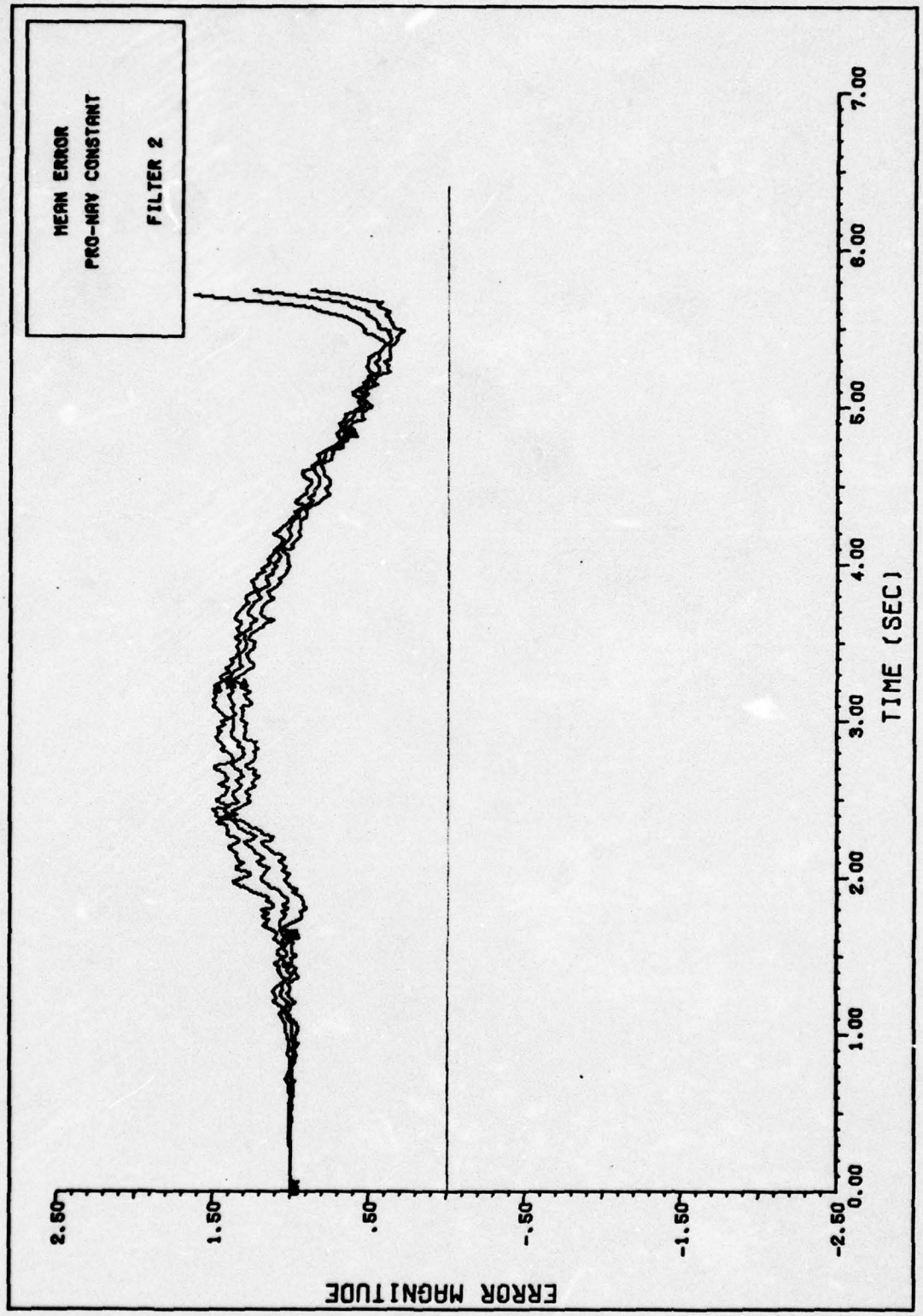


FIGURE 53. PRO-NAV CONSTANT PERFORMANCE FILTER 2

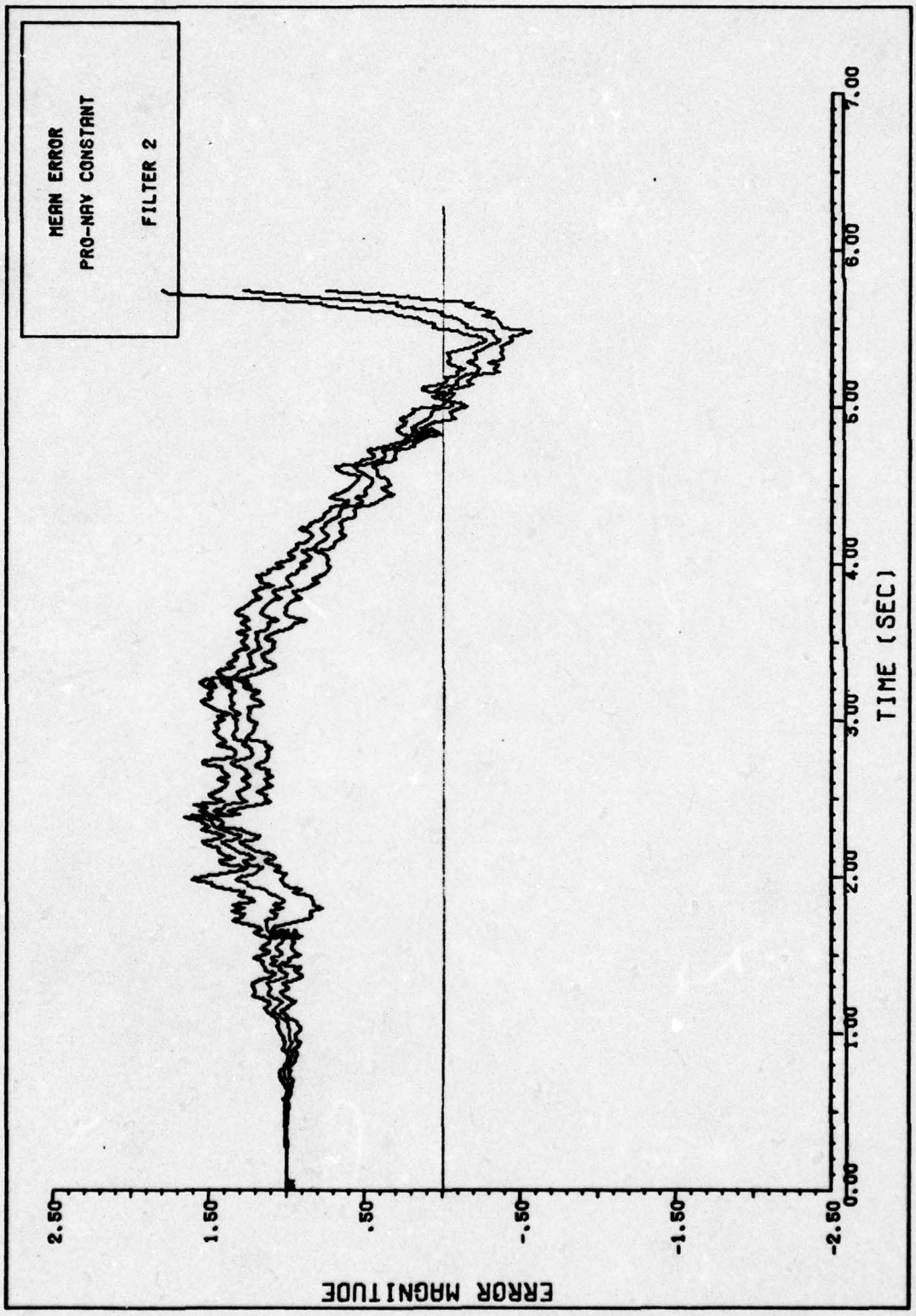


FIGURE 54. PRO-NAV CONSTANT PERFORMANCE FILTER 2

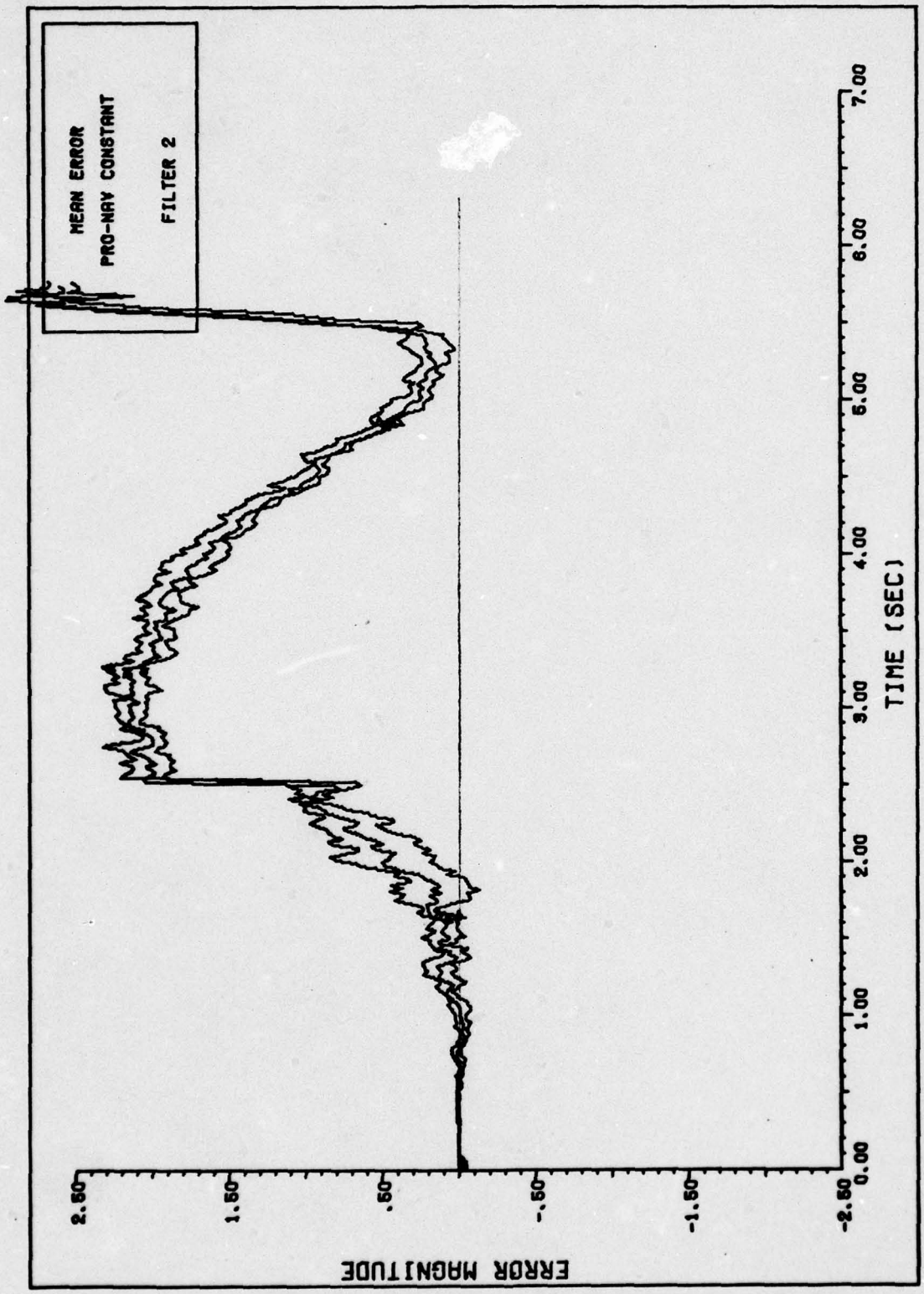


FIGURE 55. PRO-NAV CONSTANT PERFORMANCE FILTER 2

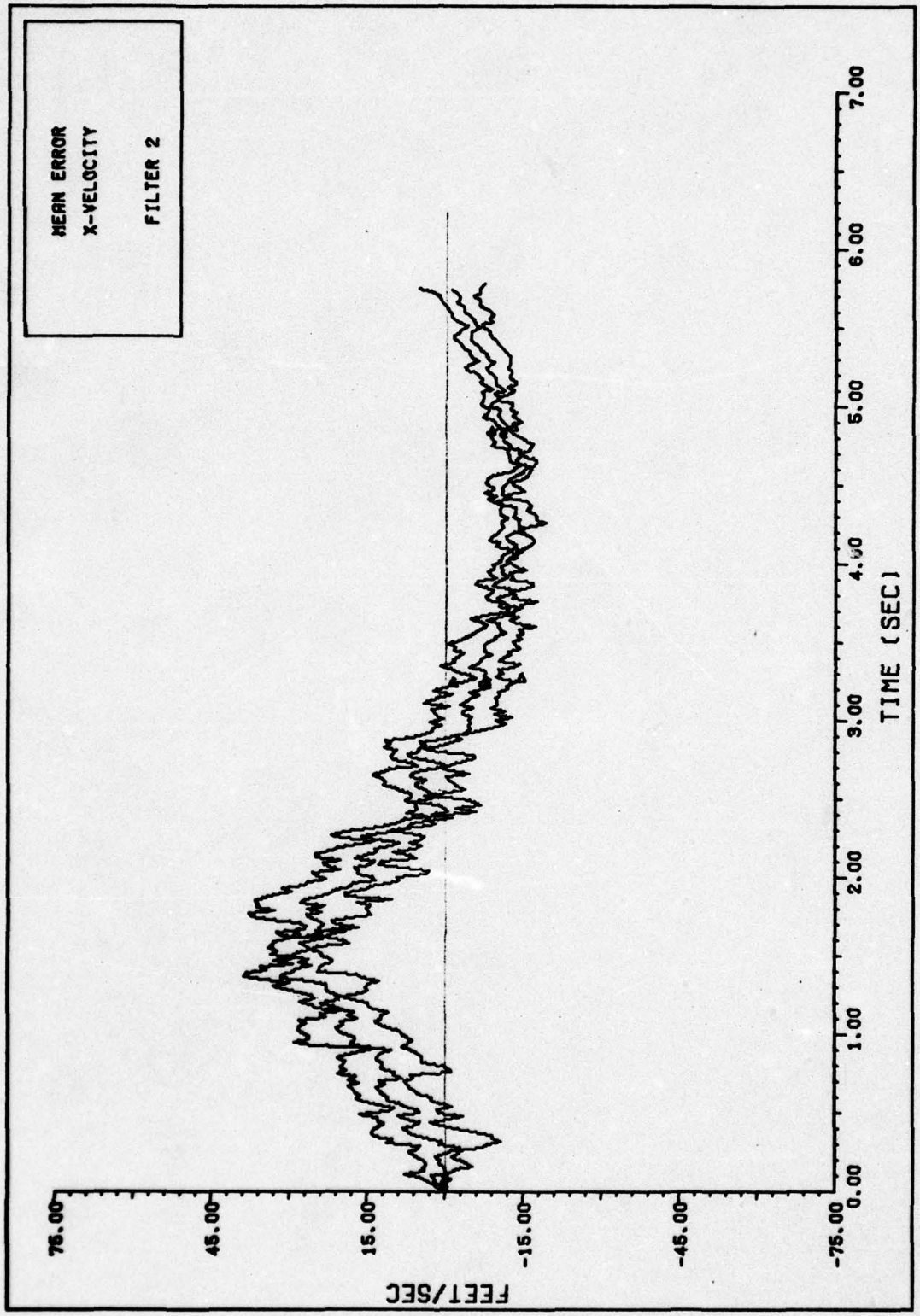


FIGURE 56. X-VELOCITY PERFORMANCE FILTER 2

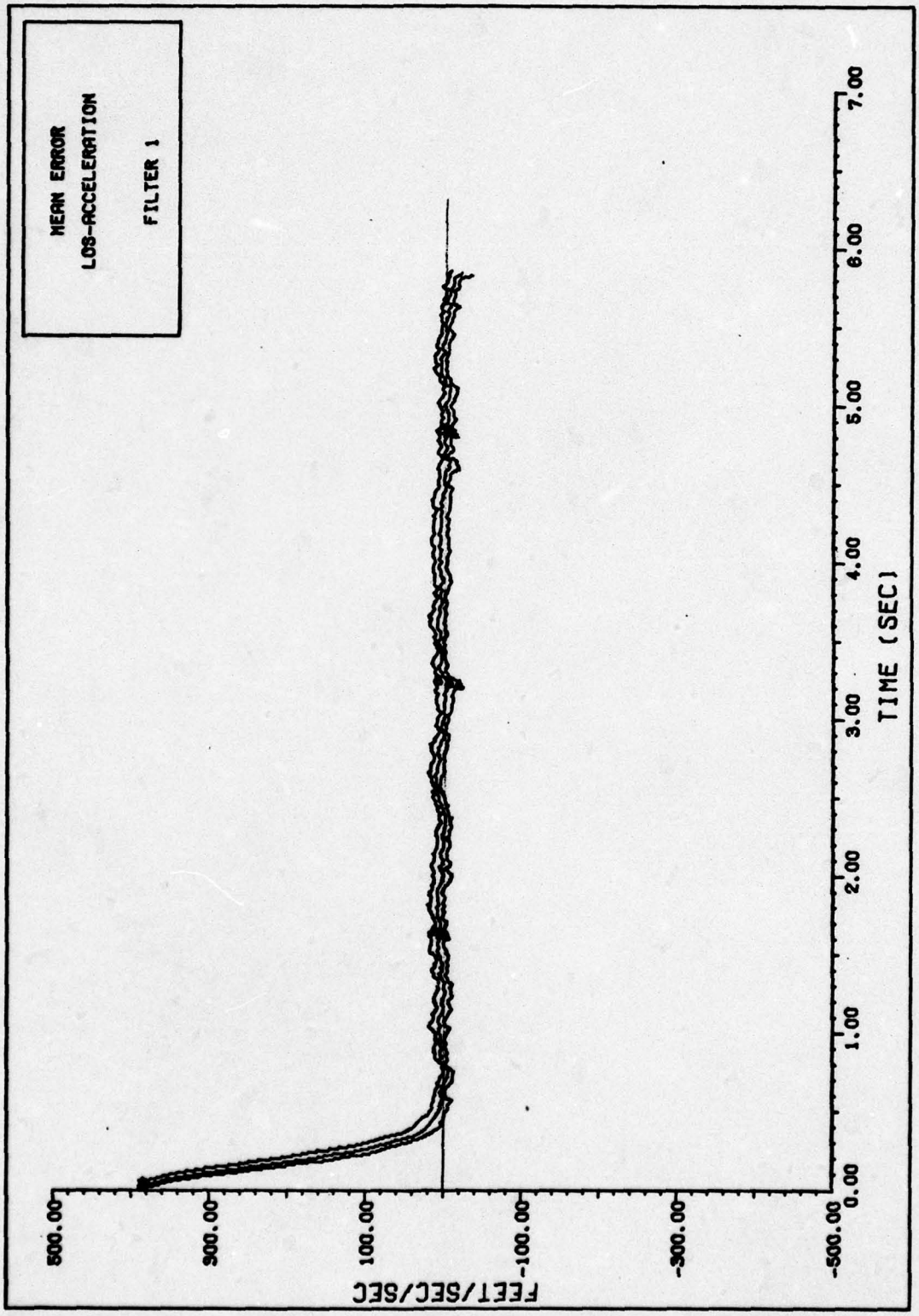


FIGURE 57. LOS-ACCELERATION PERFORMANCE FILTER 1

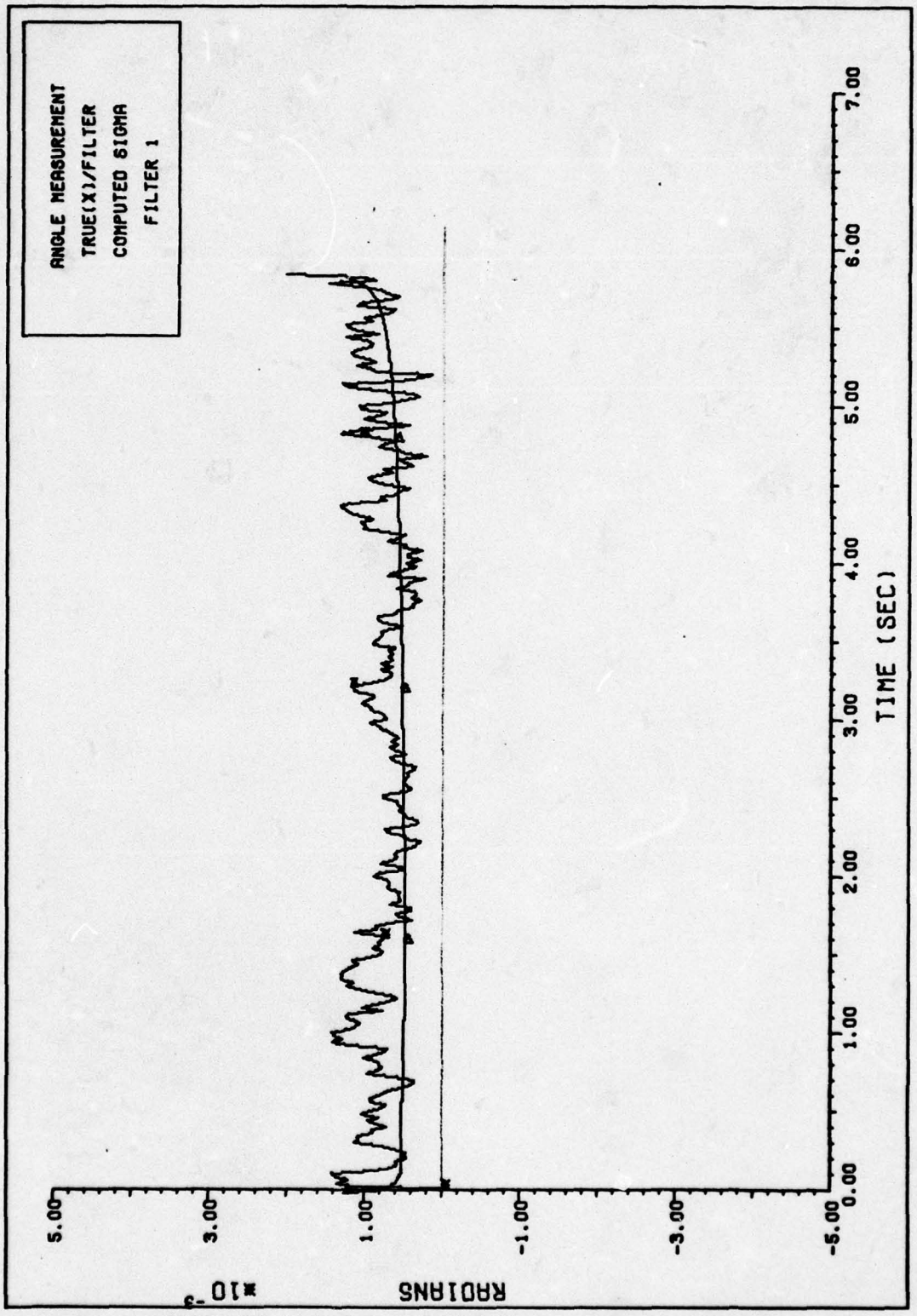


FIGURE 58. ANGLE MEASUREMENT TUNING FILTER 1

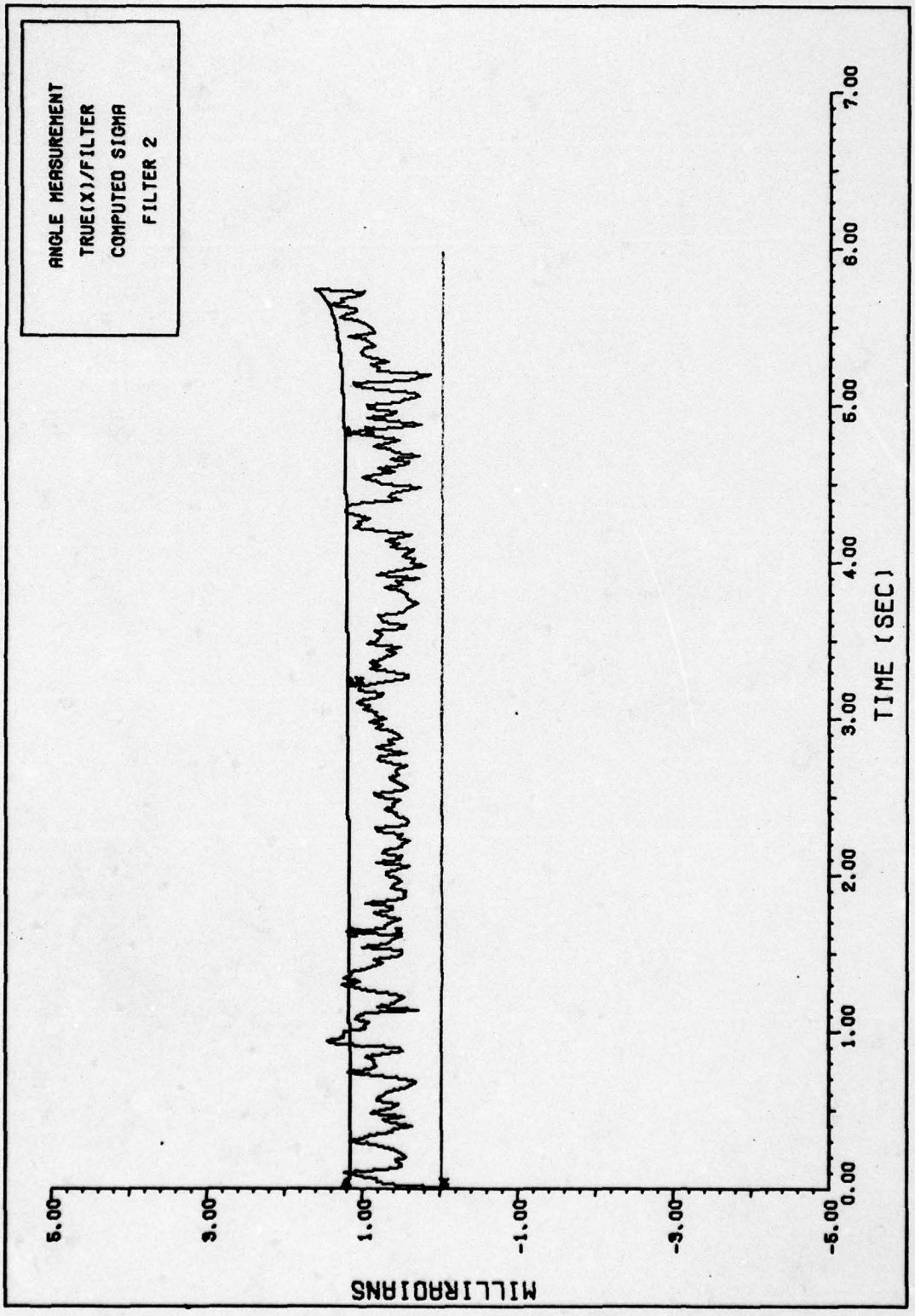


FIGURE 59. ANGLE MEASUREMENT TUNING FILTER 2

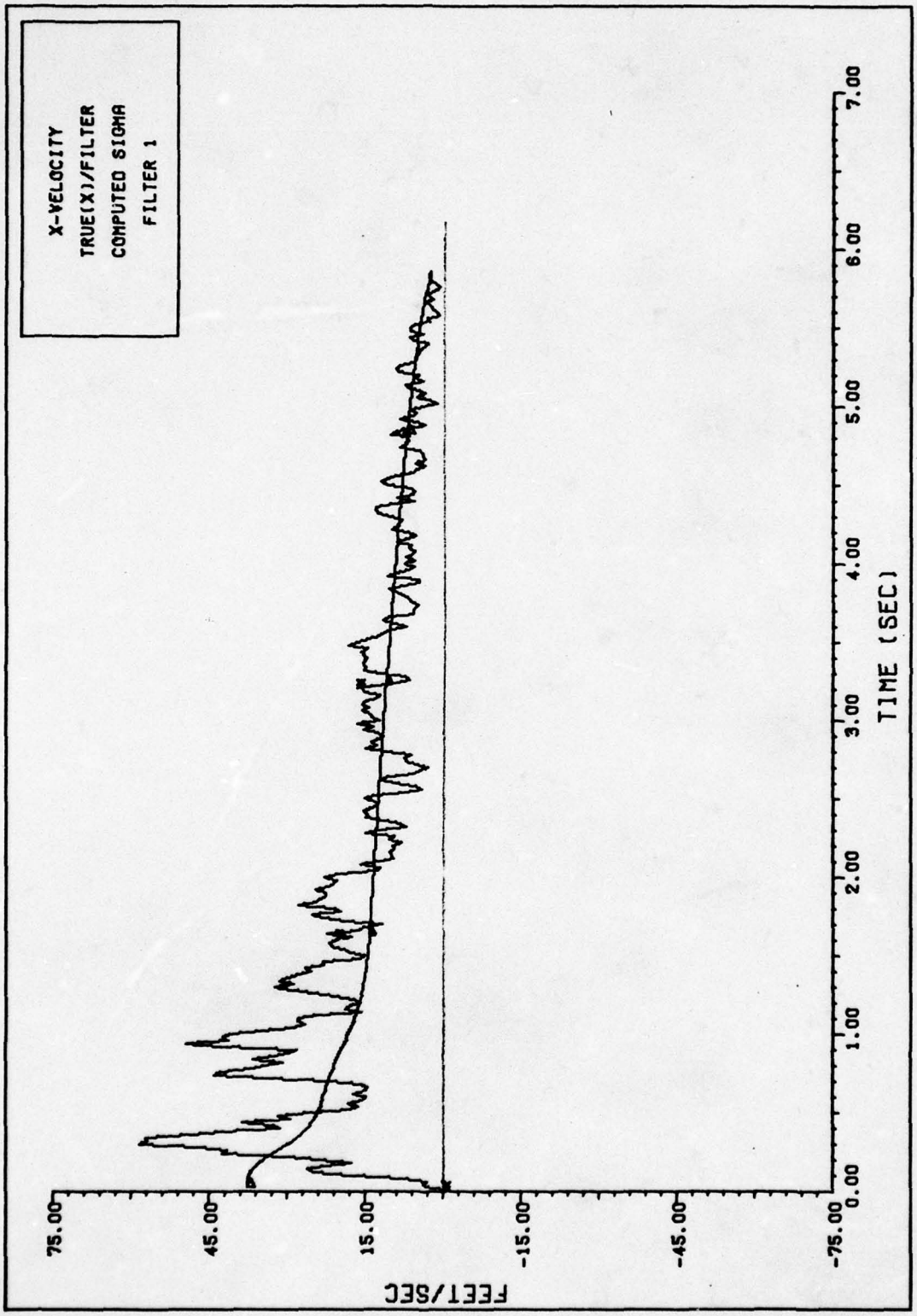


FIGURE 60. X-VELOCITY TUNING FILTER 1

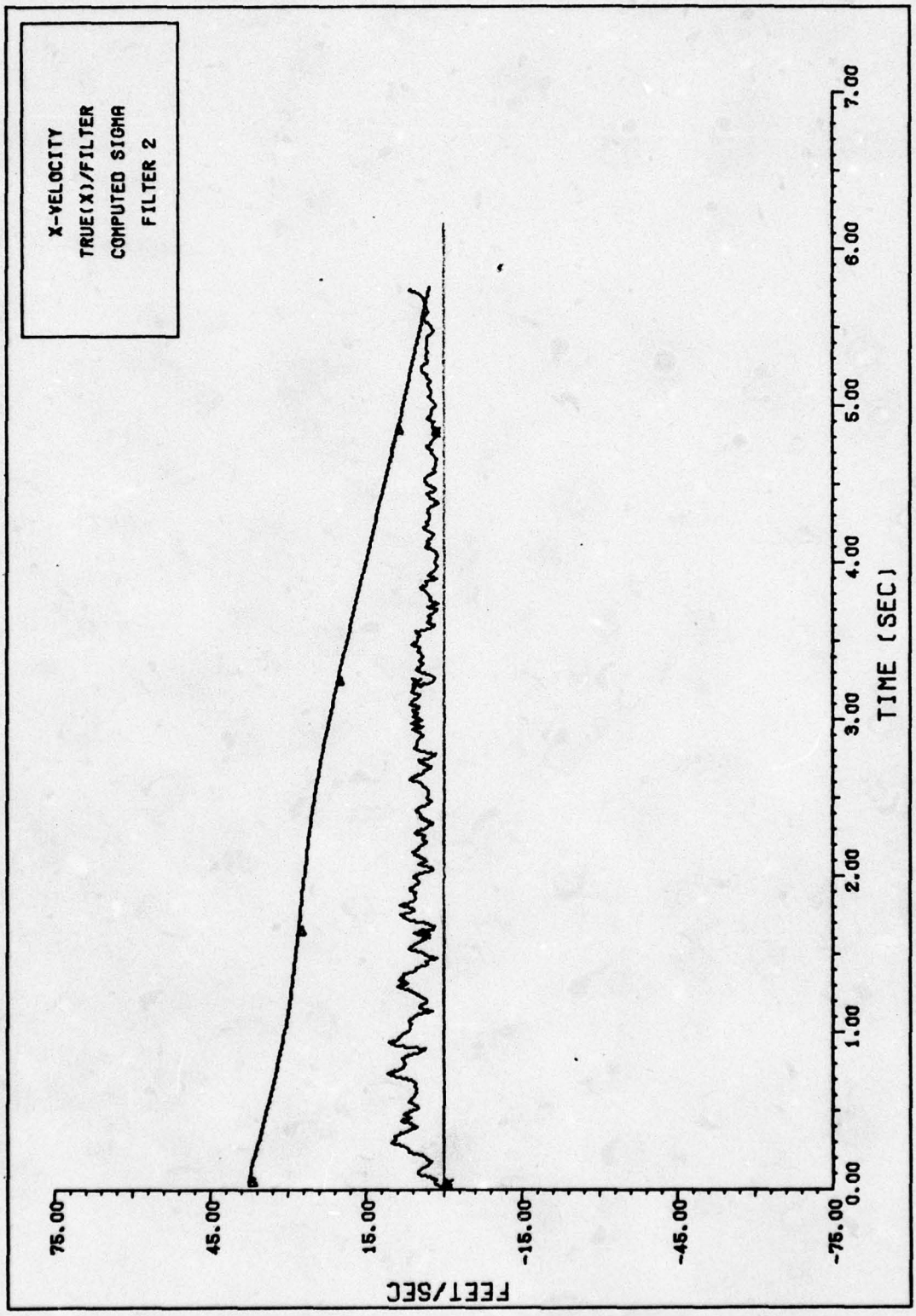


FIGURE 61. X-VELOCITY TUNING FILTER'2

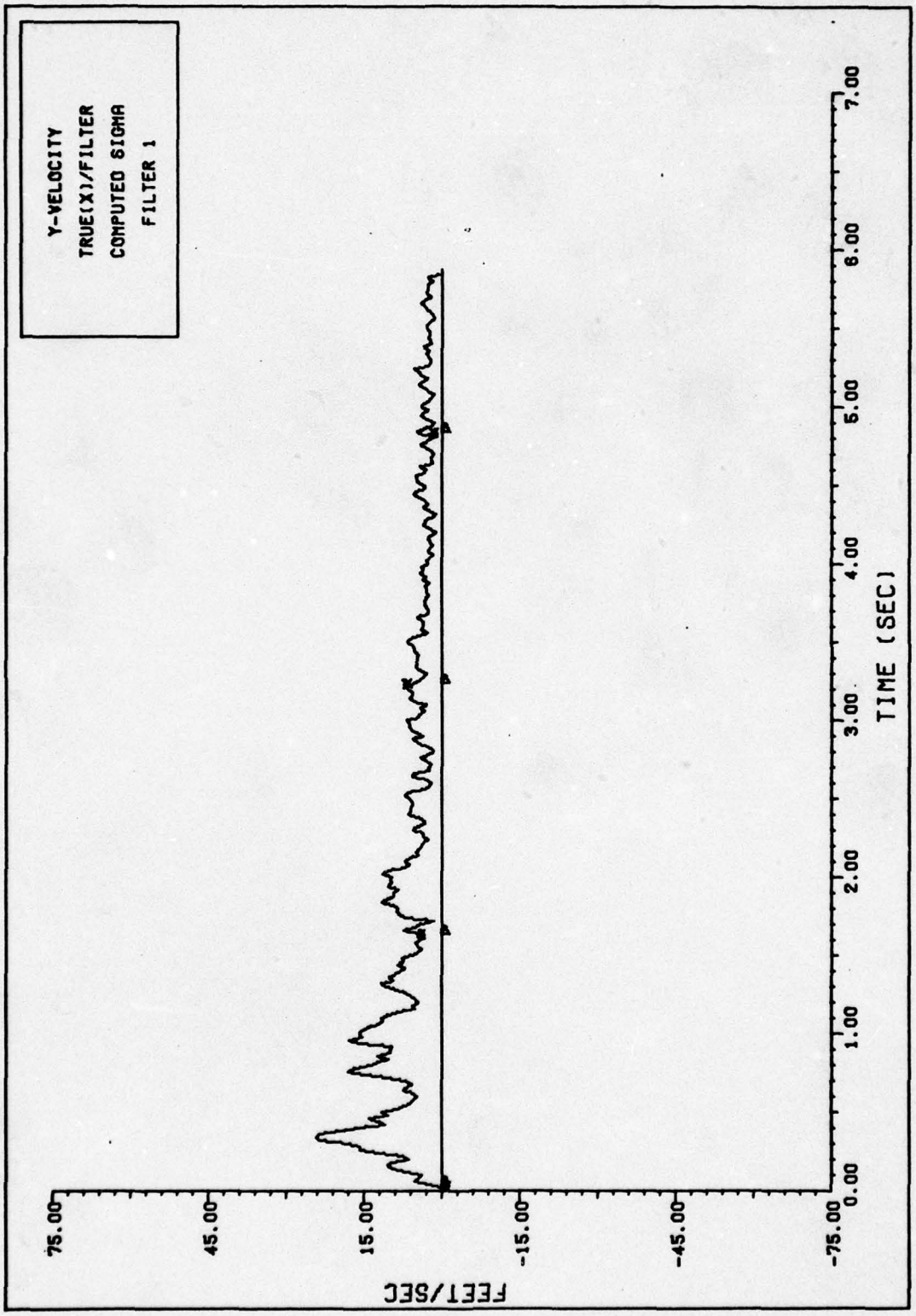


FIGURE 62. Y-VELOCITY TUNING FILTER 1

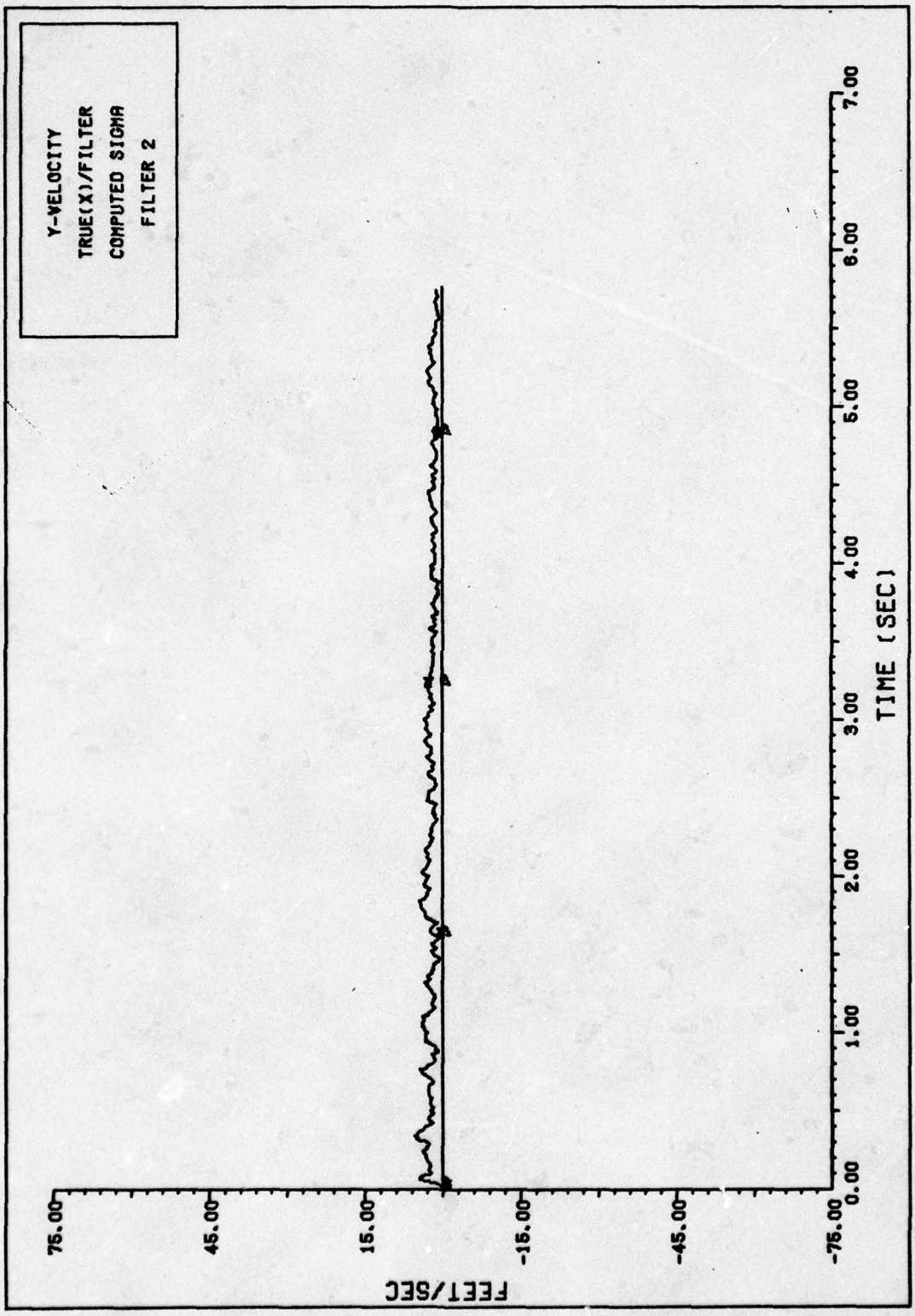


FIGURE 63. Y-VELOCITY TUNING FILTER 2

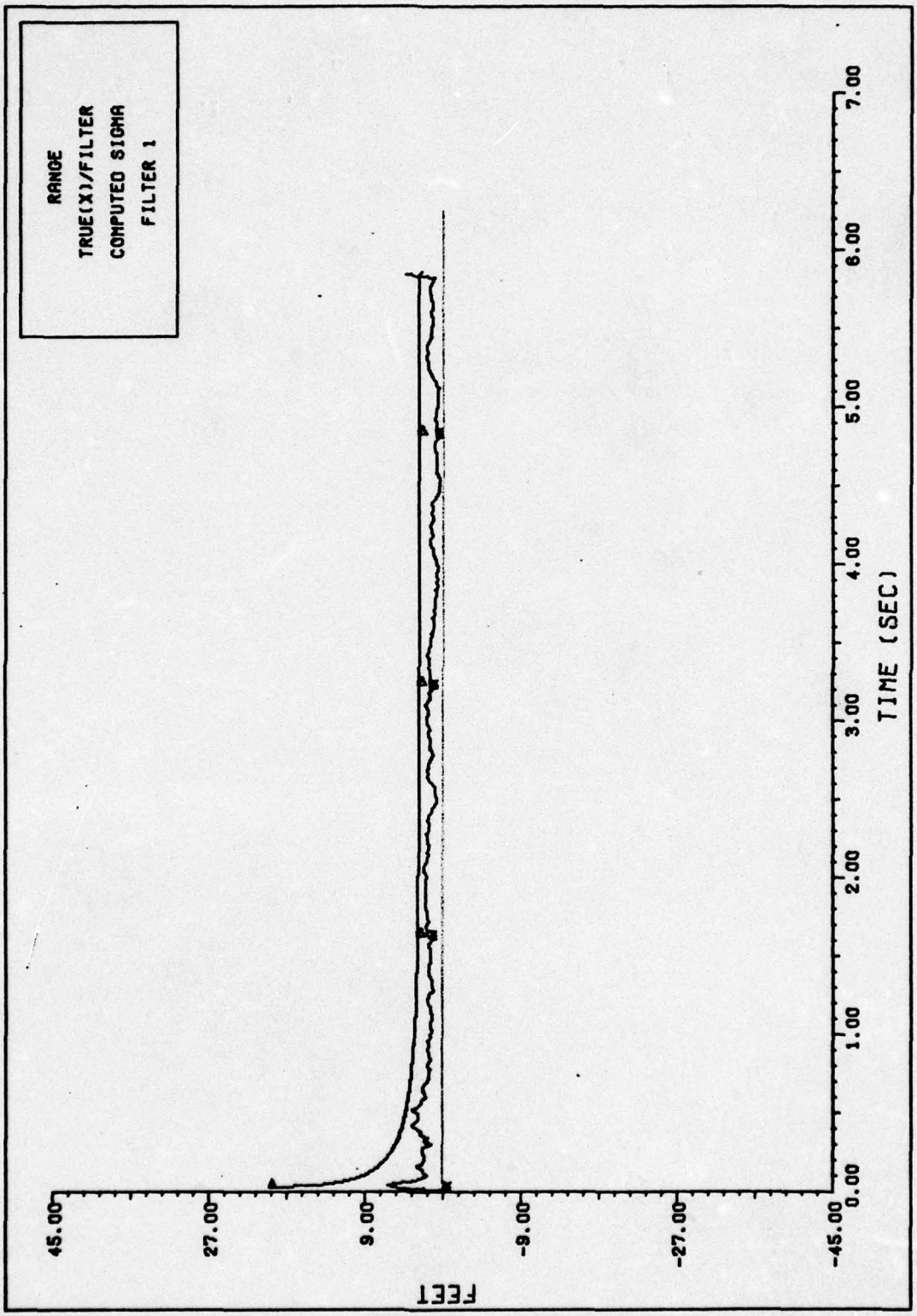


FIGURE 64. RANGE TUNING FILTER 1 .

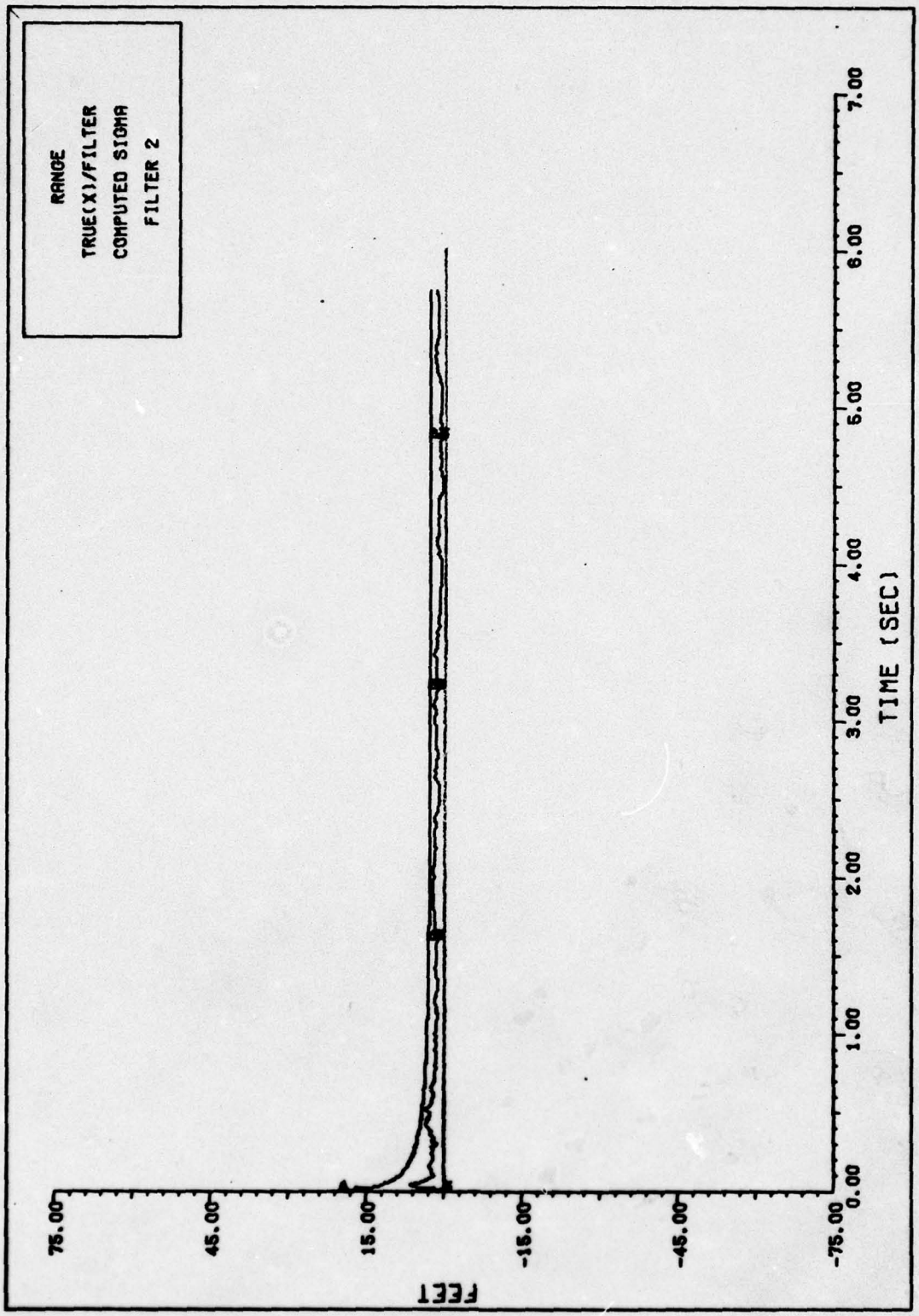


FIGURE 65. RANGE TUNING FILTER 2

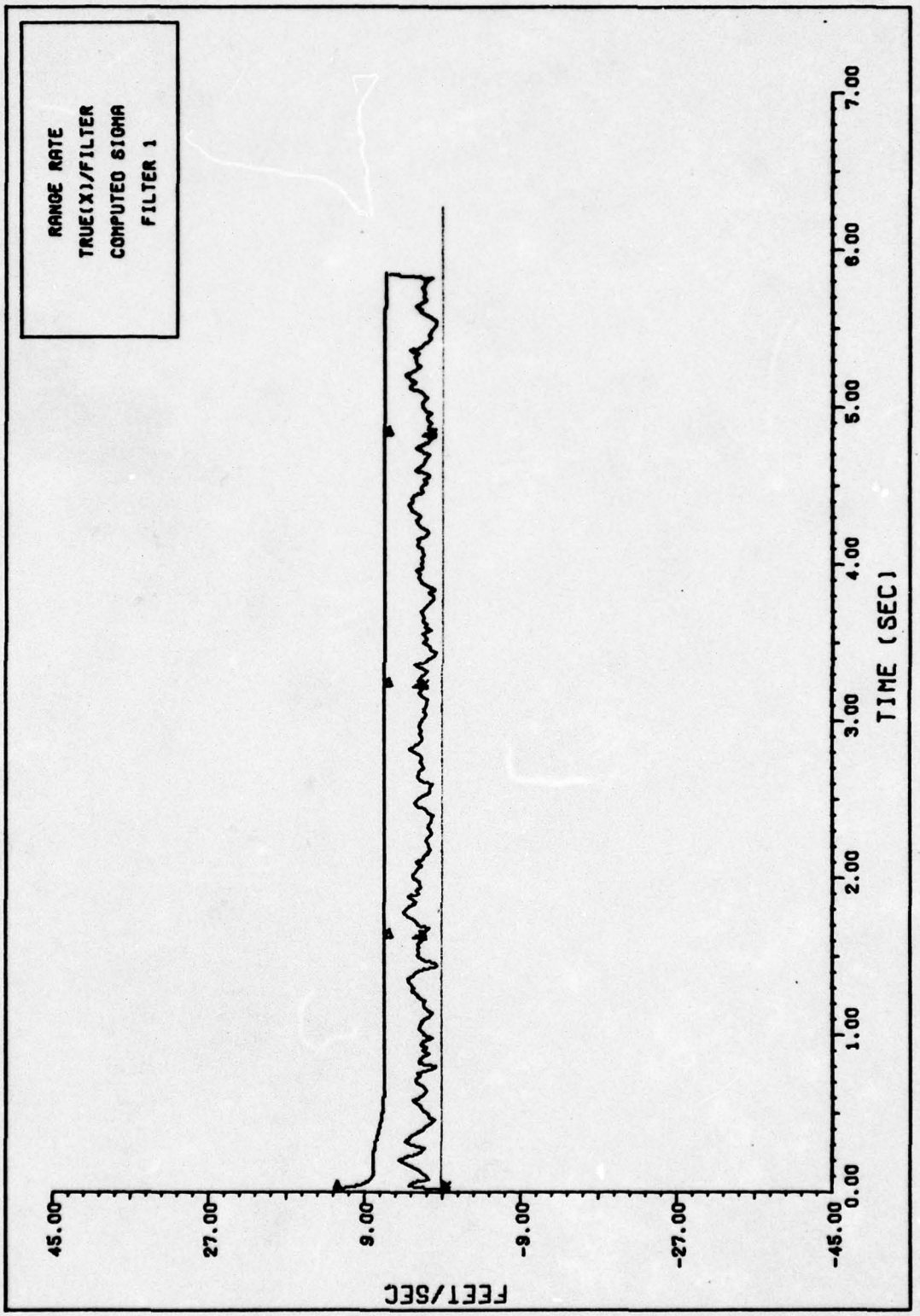


FIGURE 66. RANGE RATE TUNING FILTER 1

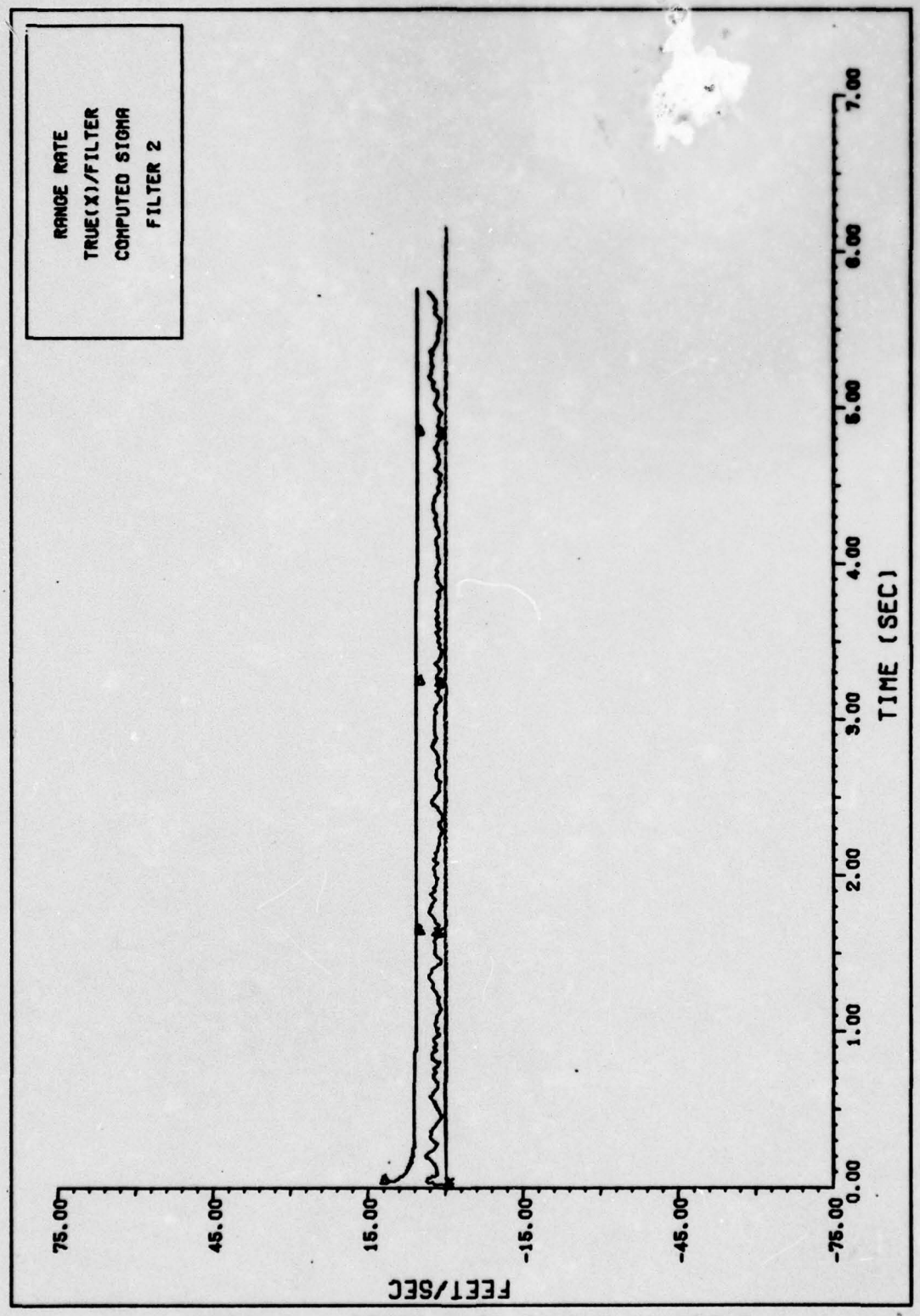


FIGURE 67. RANGE RATE TUNING FILTER.2

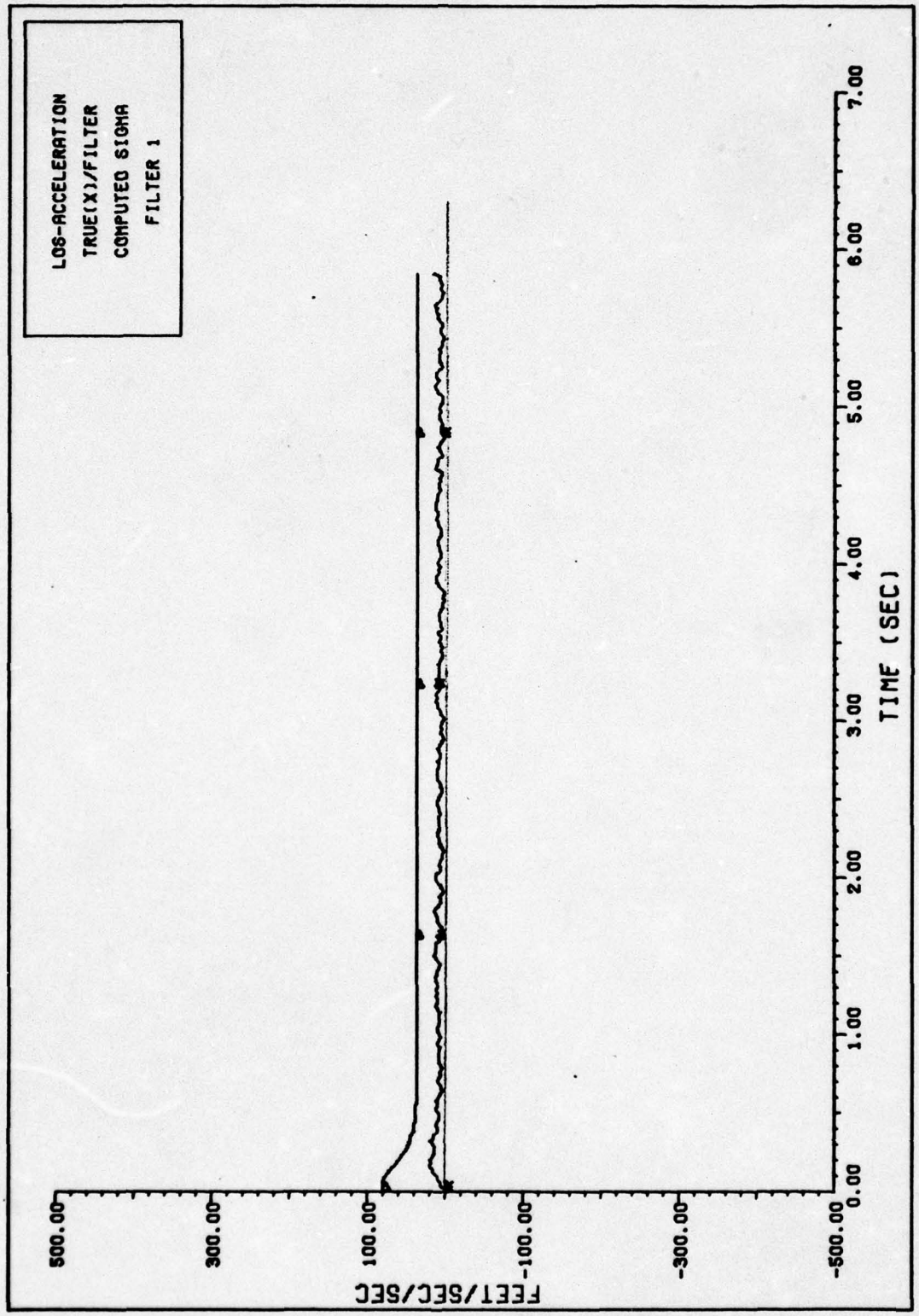


FIGURE 68. LOS-ACCELERATION TUNING FILTER 1

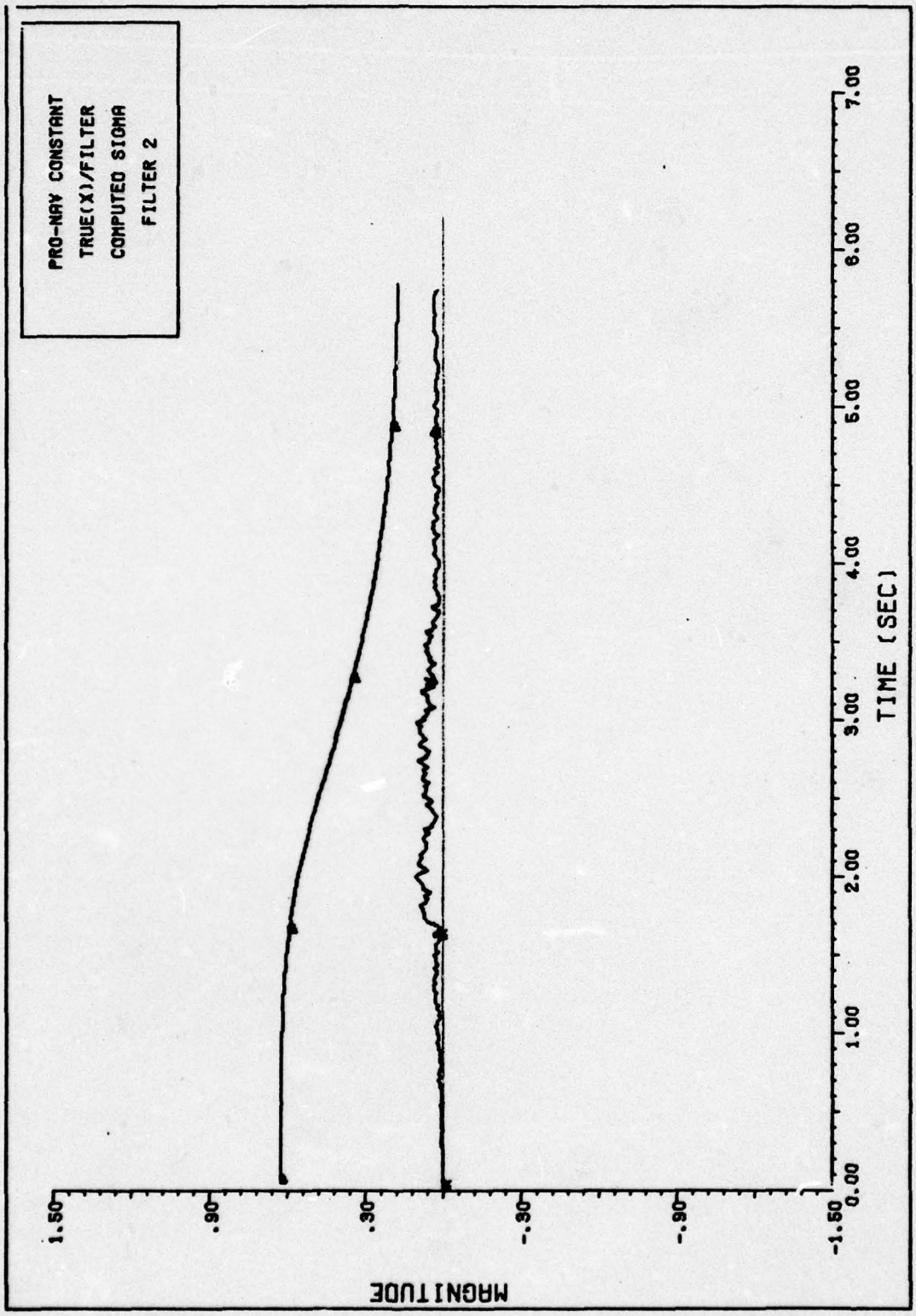


FIGURE 69. PRO-NAV CONSTANT TUNING FILTER 2

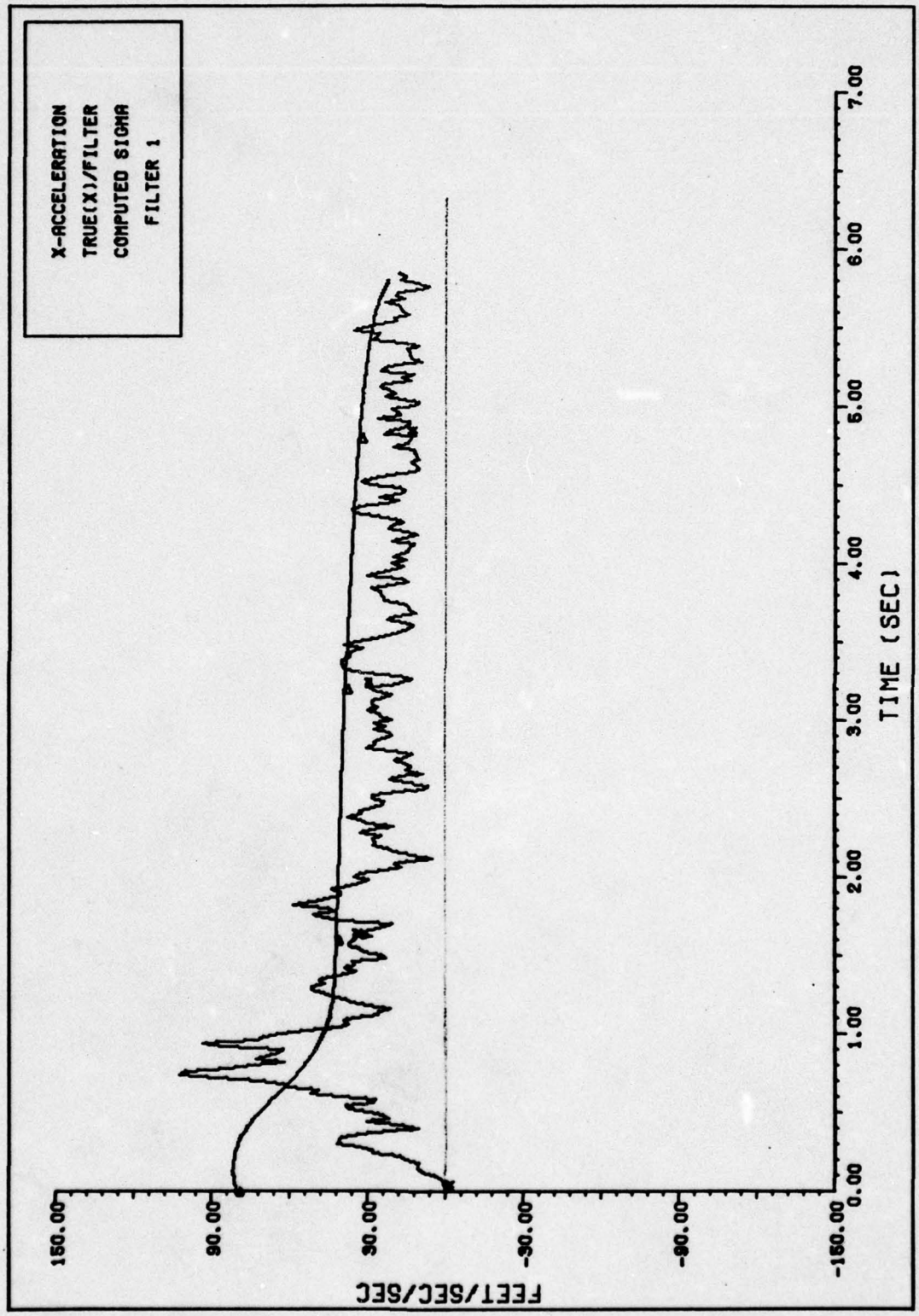


FIGURE 70. X-ACCELERATION TUNING FILTER 1

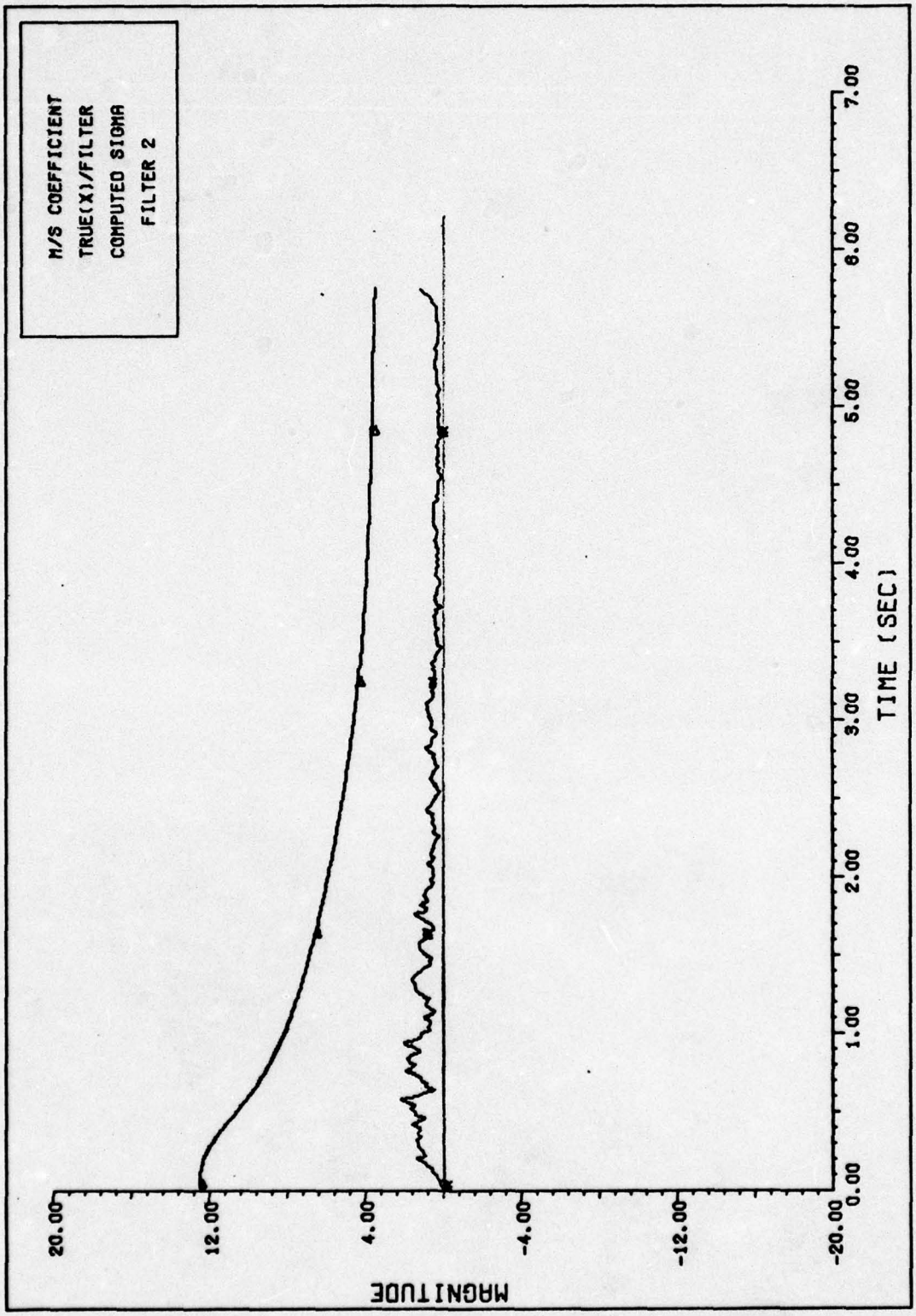


FIGURE 71. M/S COEFFICIENT TUNING FILTER 2

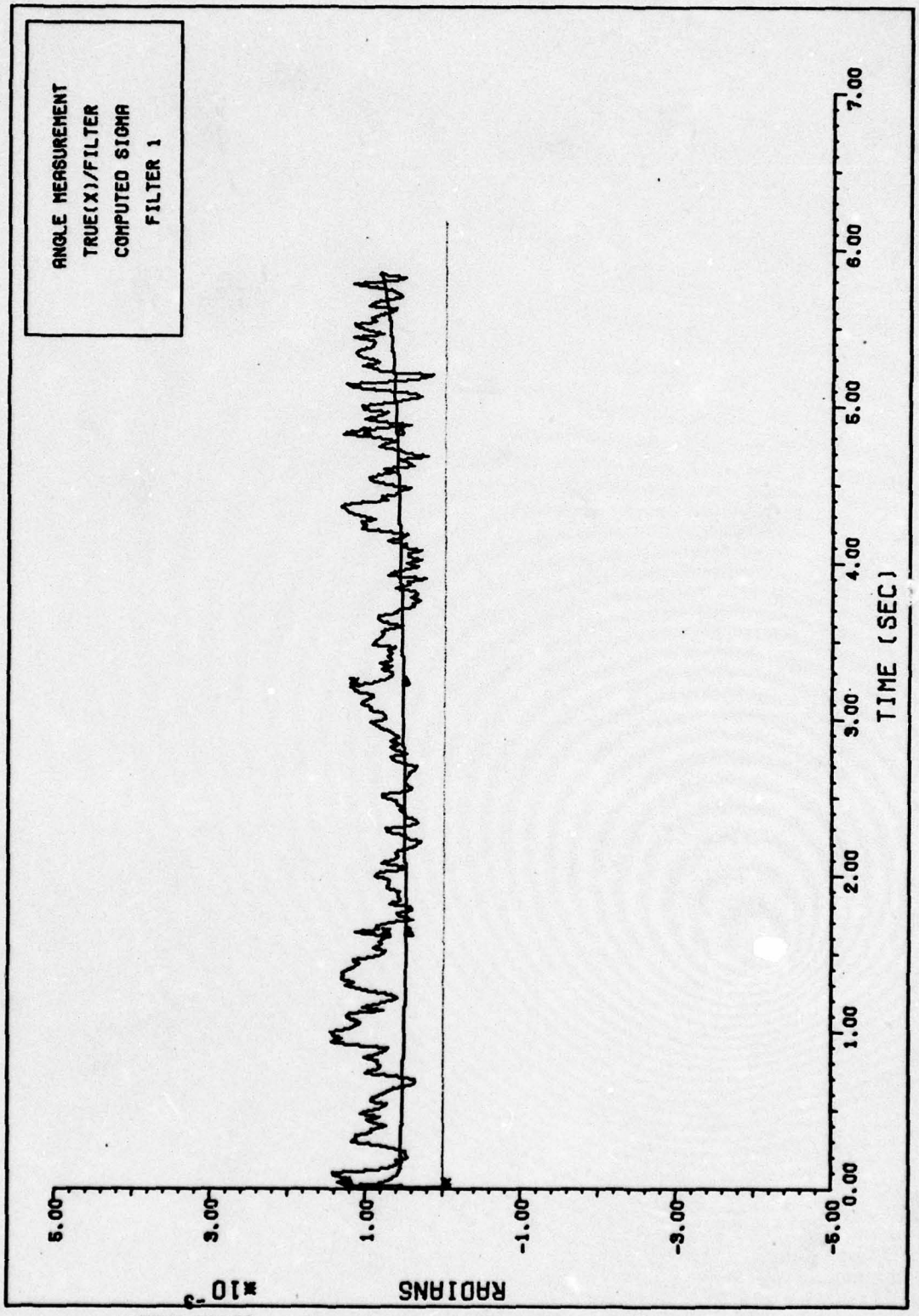


FIGURE 72. ANGLE MEASUREMENT TUNING FILTER 1

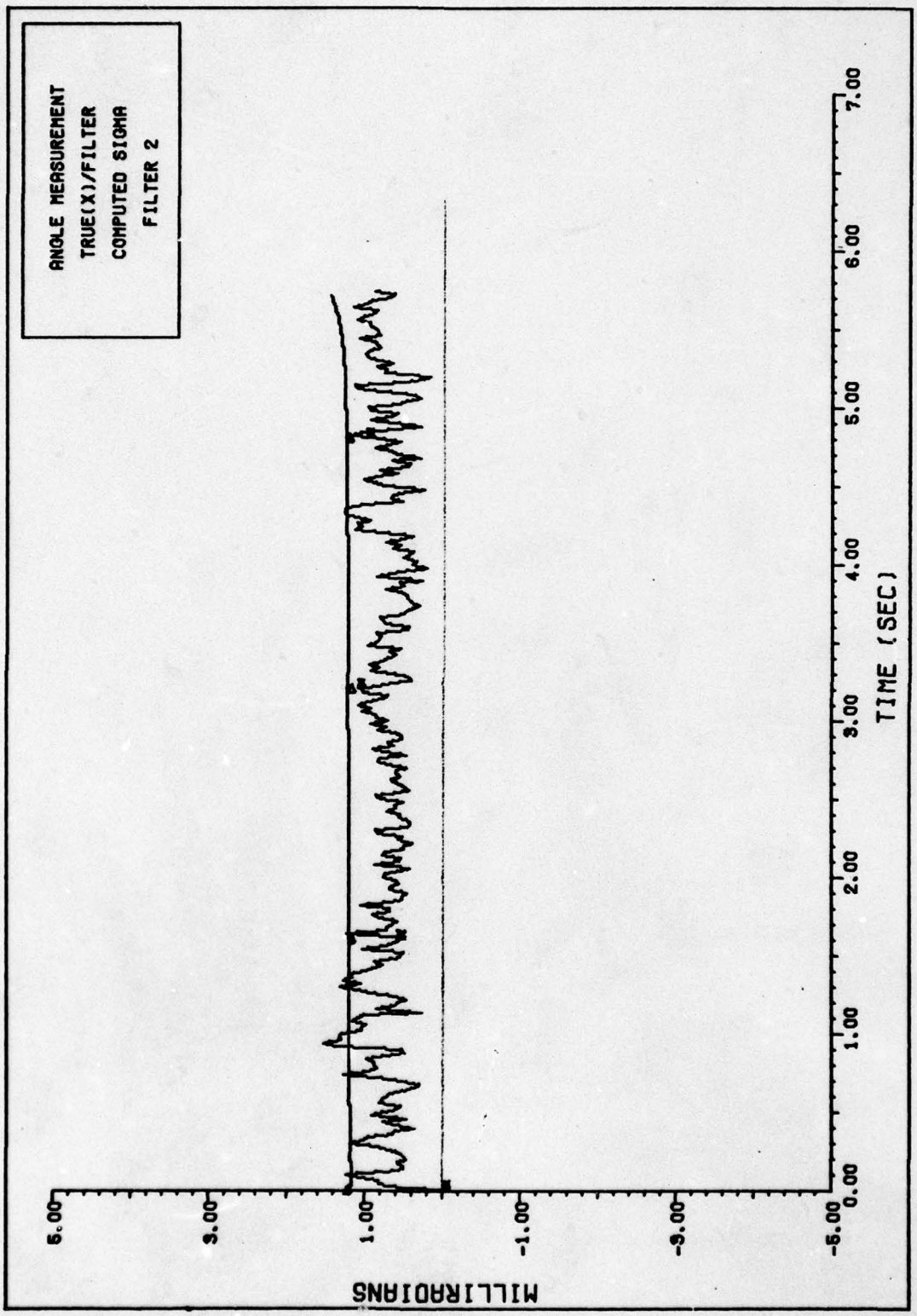


FIGURE 73. ANGLE MEASUREMENT TUNING FILTER 2

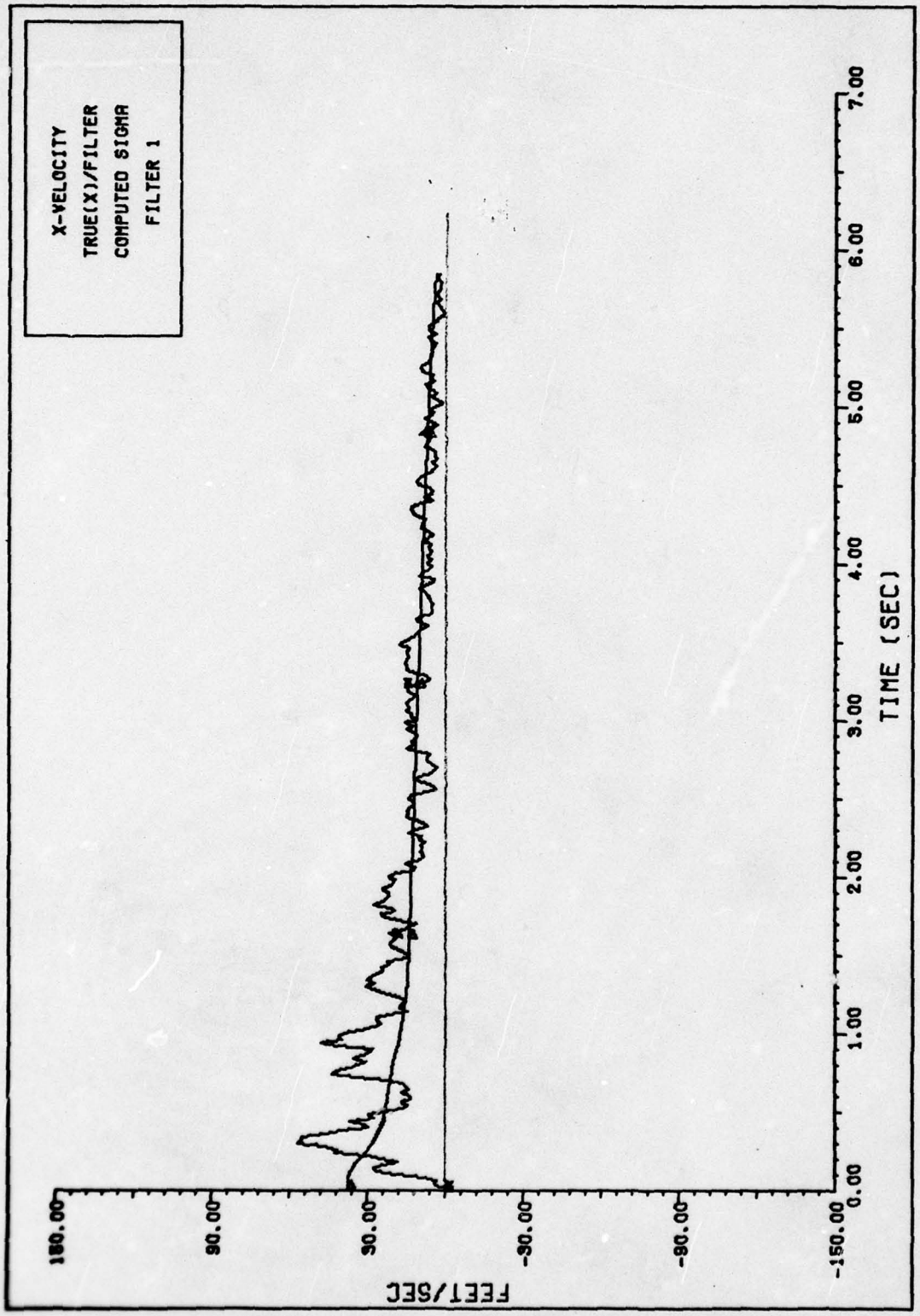


FIGURE 74. X-VELOCITY TUNING FILTER 1

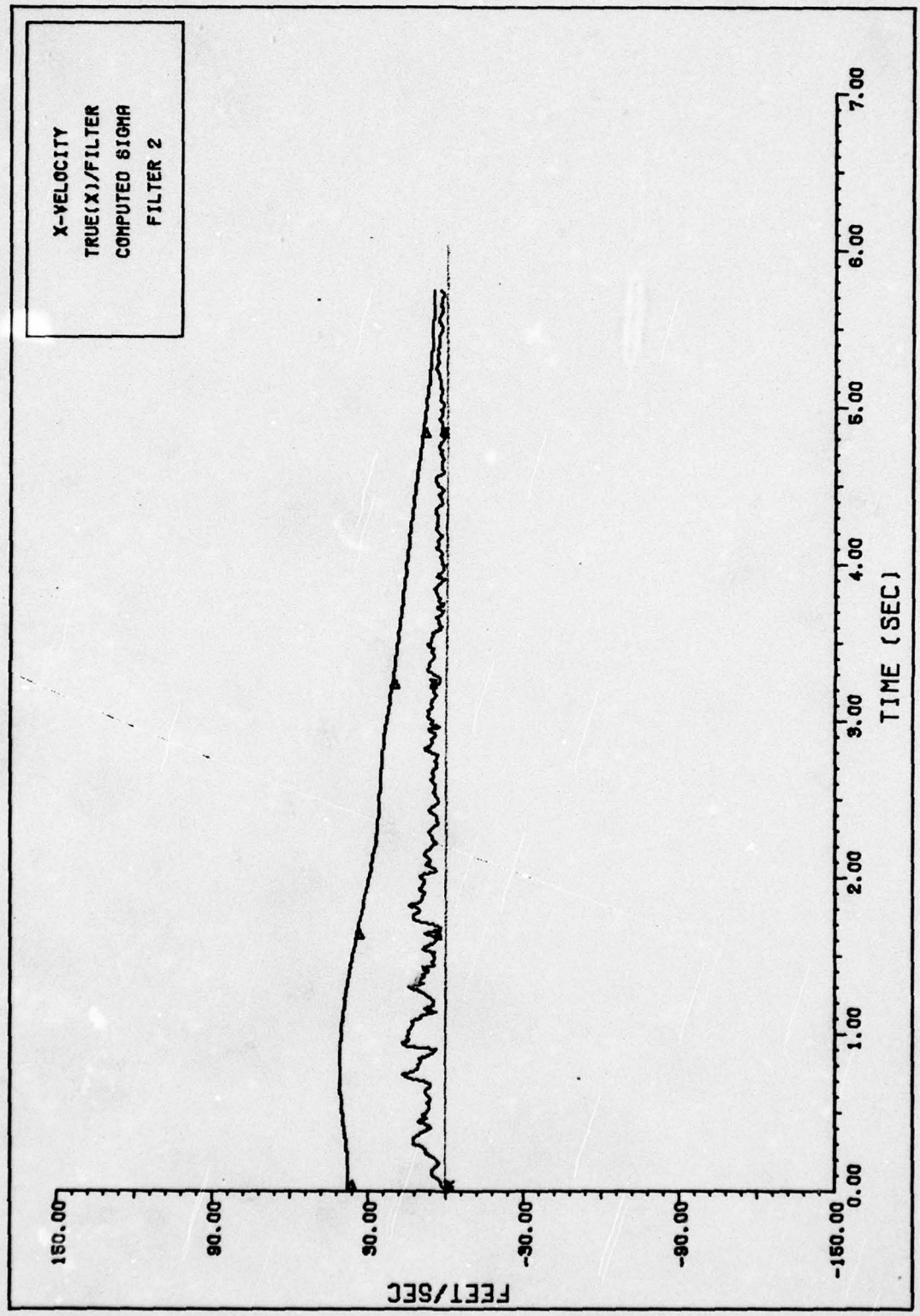


FIGURE 75. X-VELOCITY TUNING FILTER.2

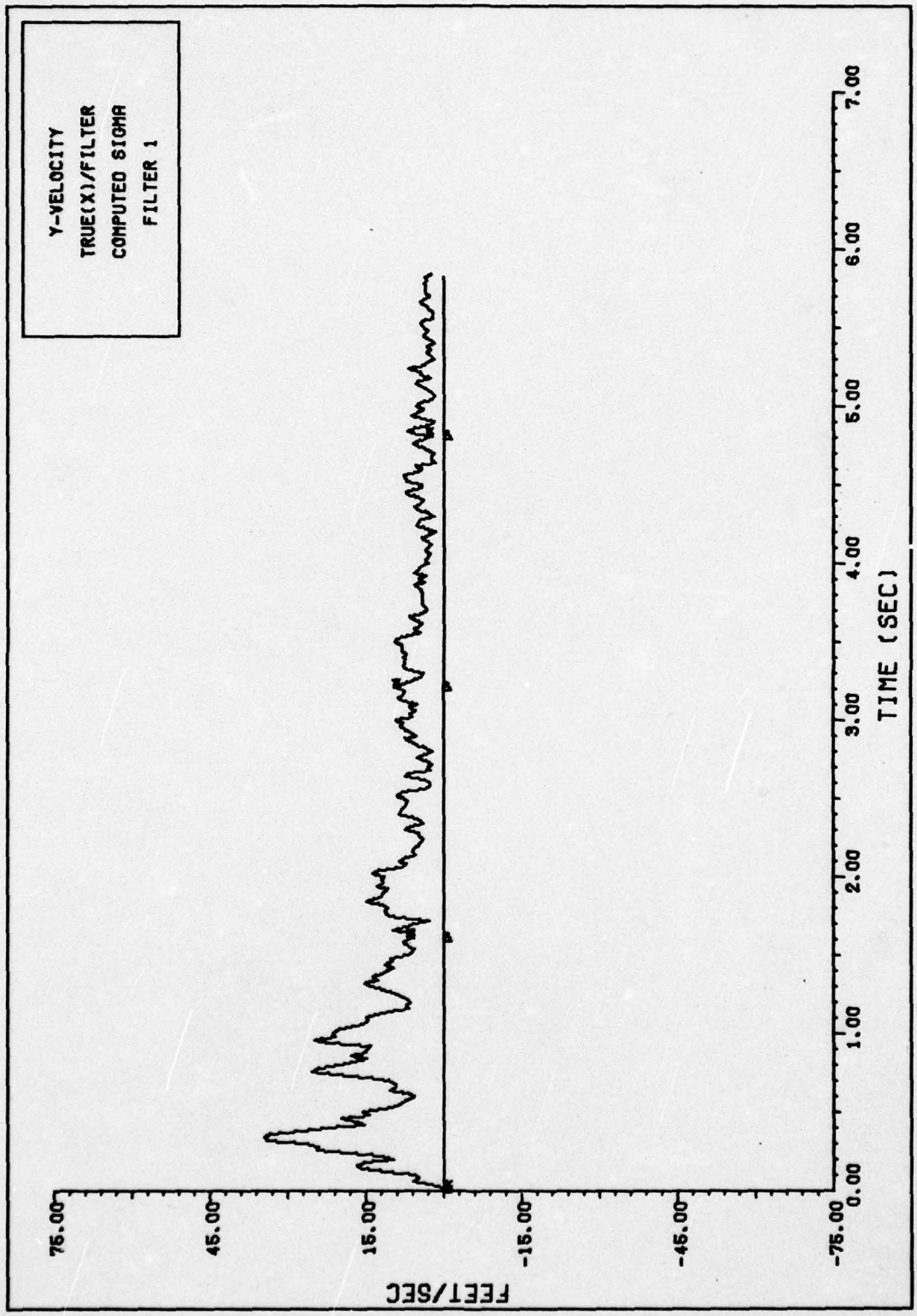


FIGURE 76. Y-VELOCITY TUNING FILTER 1

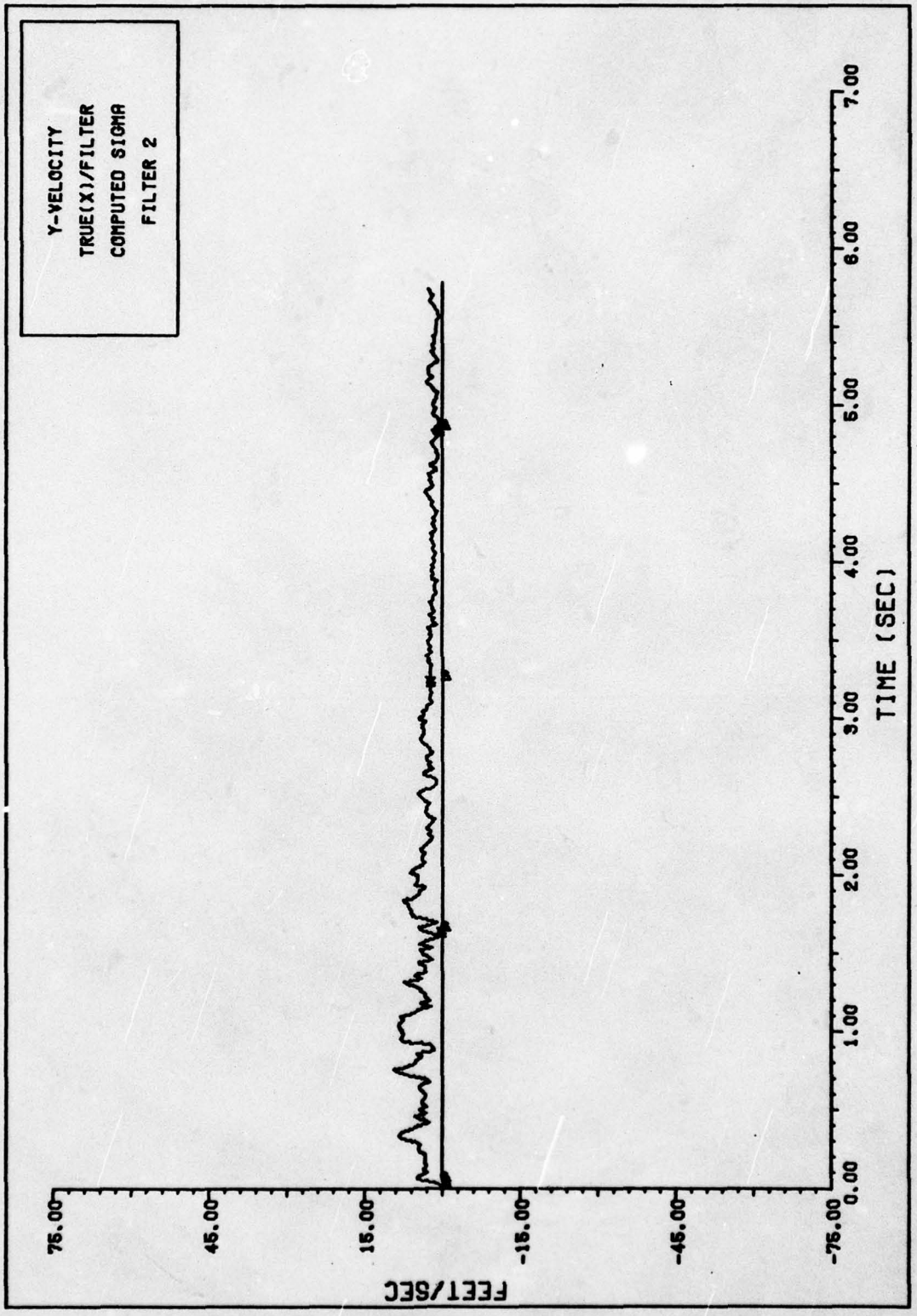


FIGURE 77. Y-VELOCITY TUNING FILTER 2

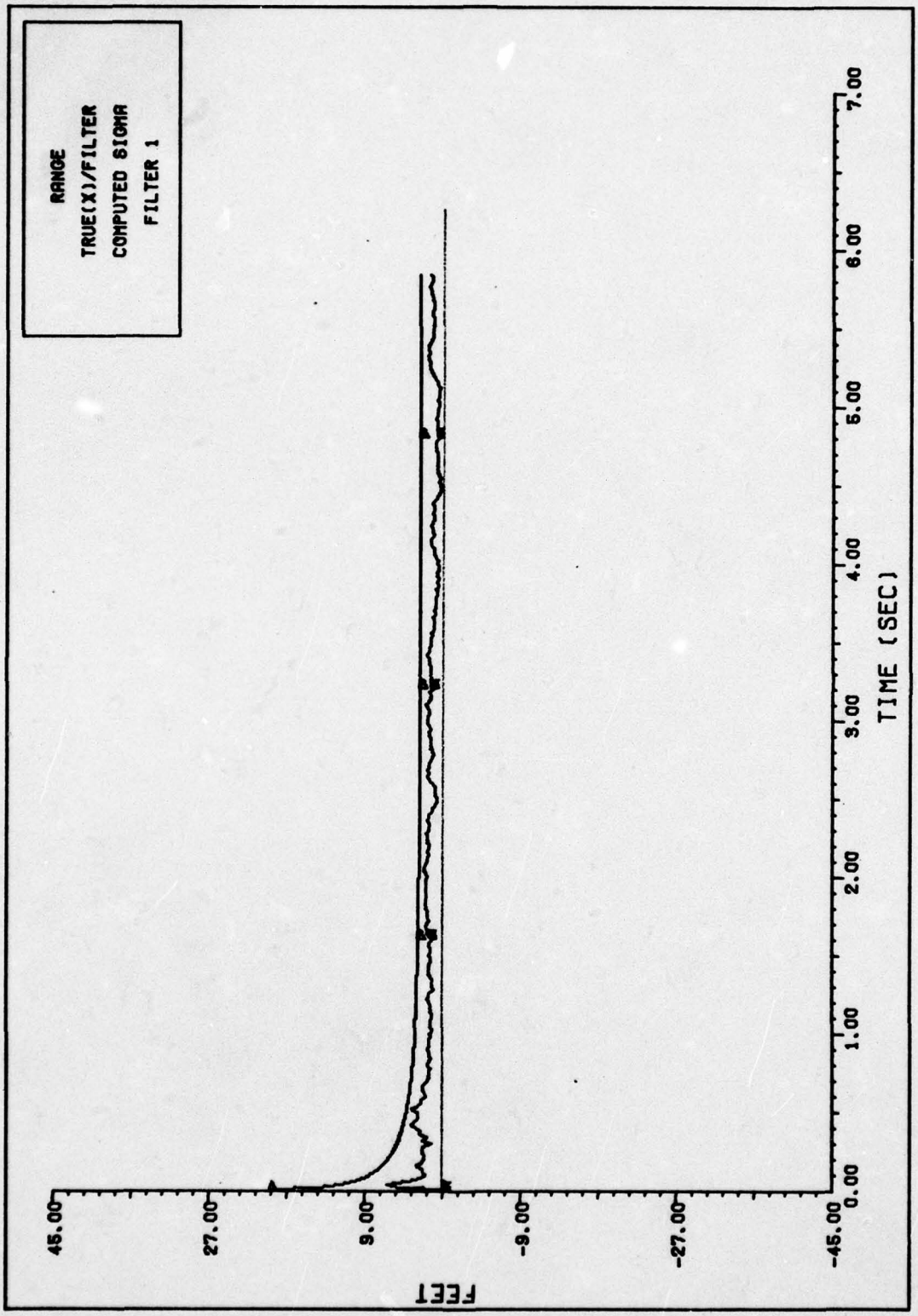


FIGURE 78. RANGE TUNING FILTER 1

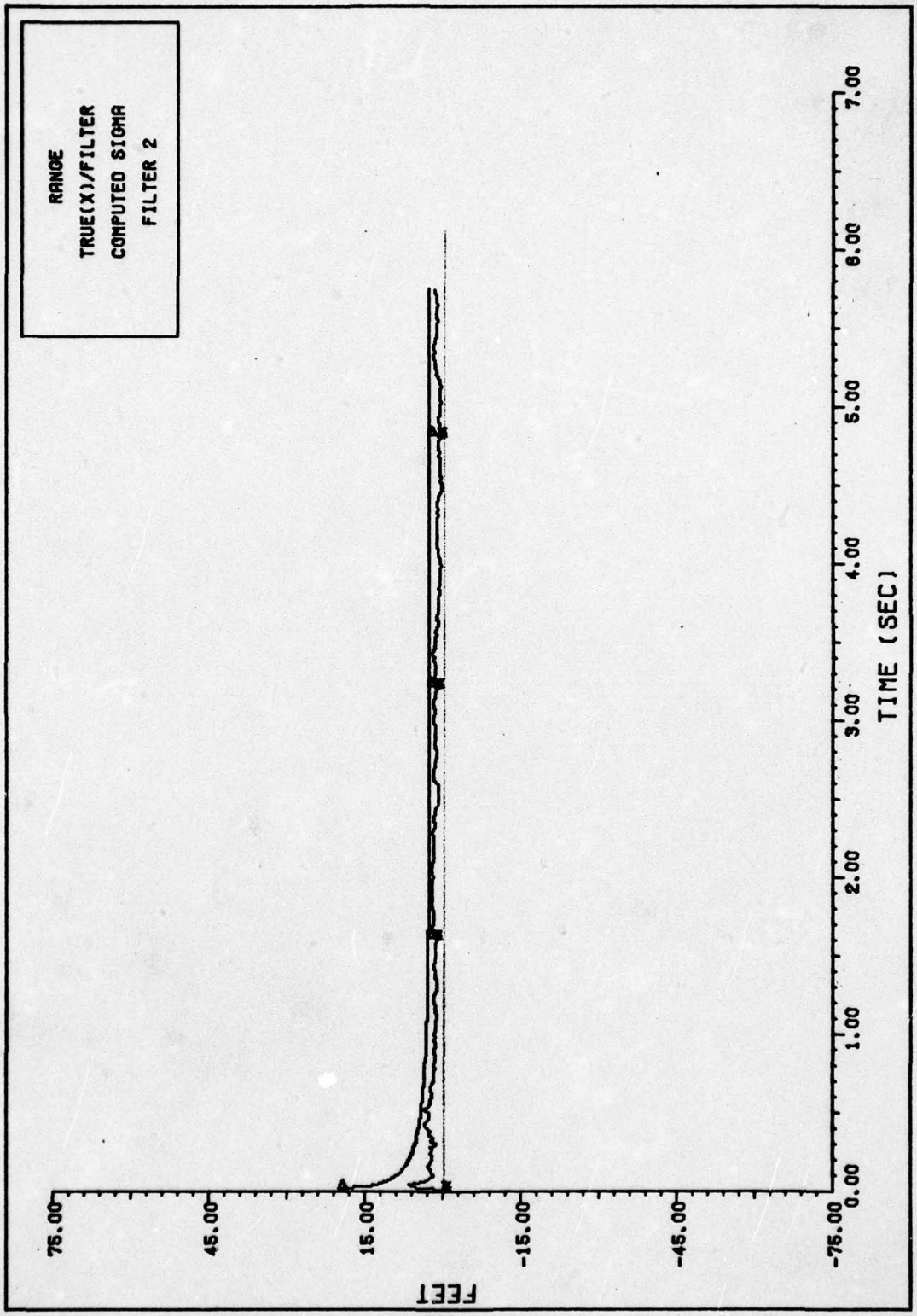


FIGURE 79. RANGE TUNING FILTER 2

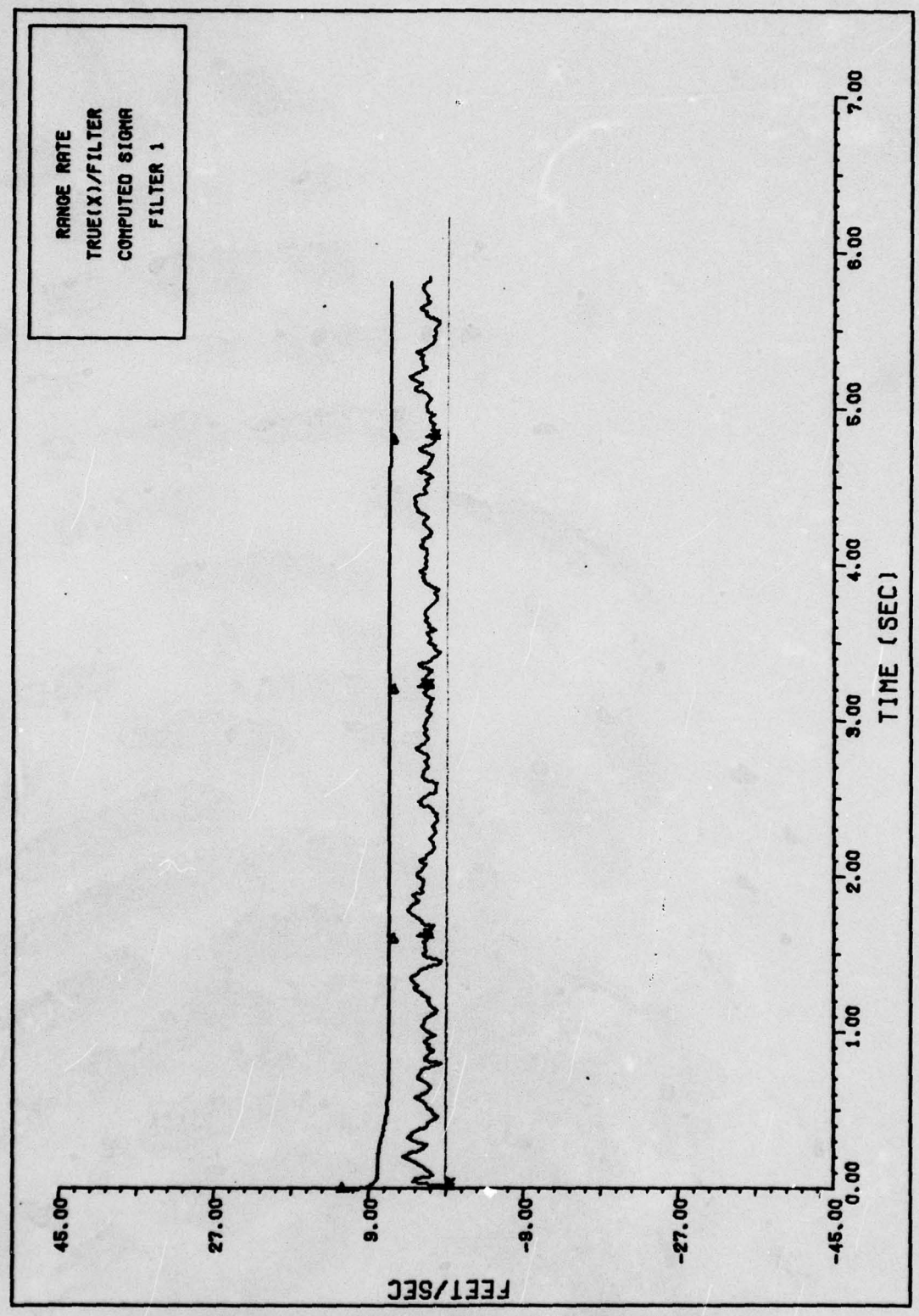


FIGURE 80. RANGE RATE TUNING FILTER 1

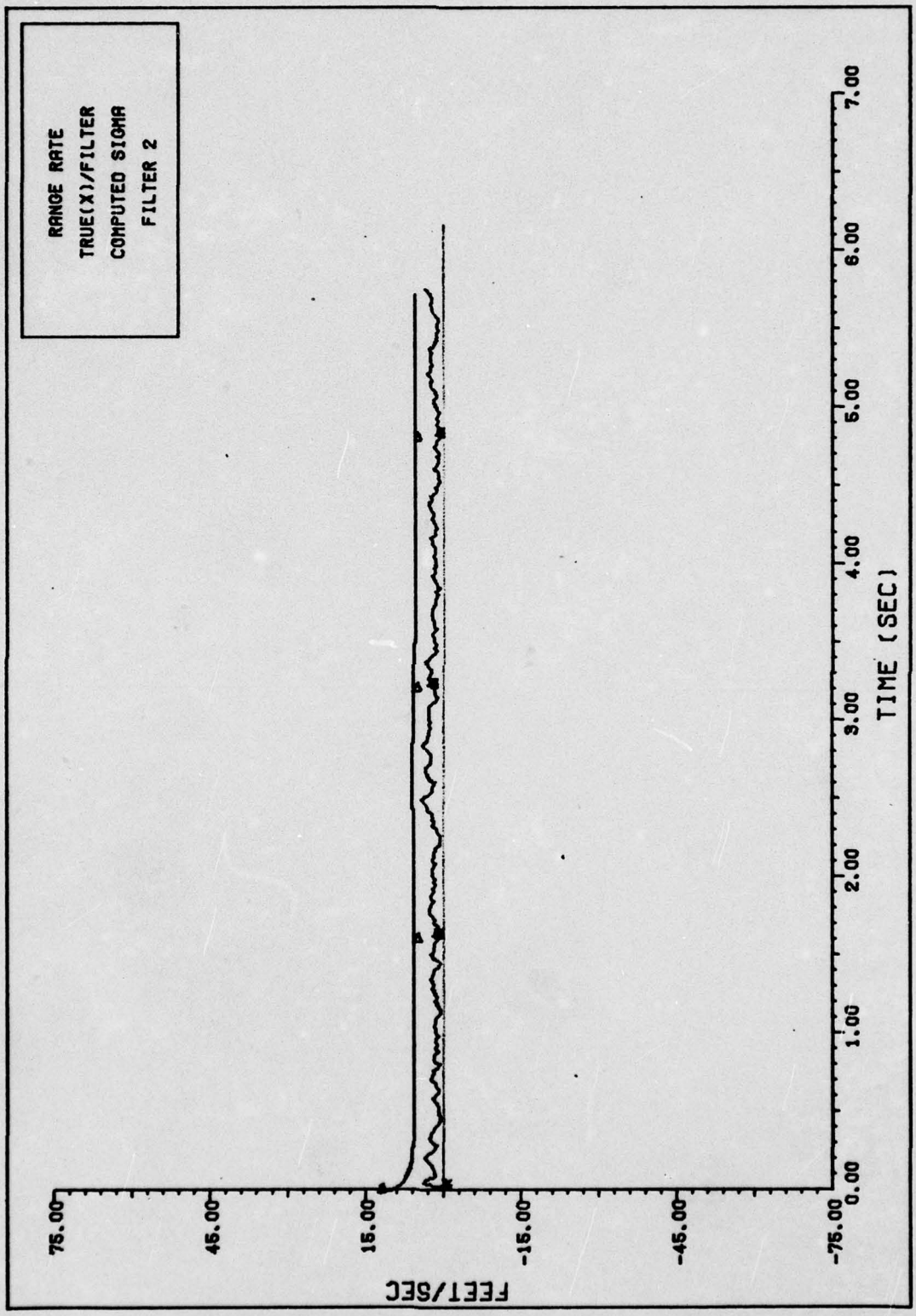


FIGURE 81. RANGE RATE TUNING FILTER.2

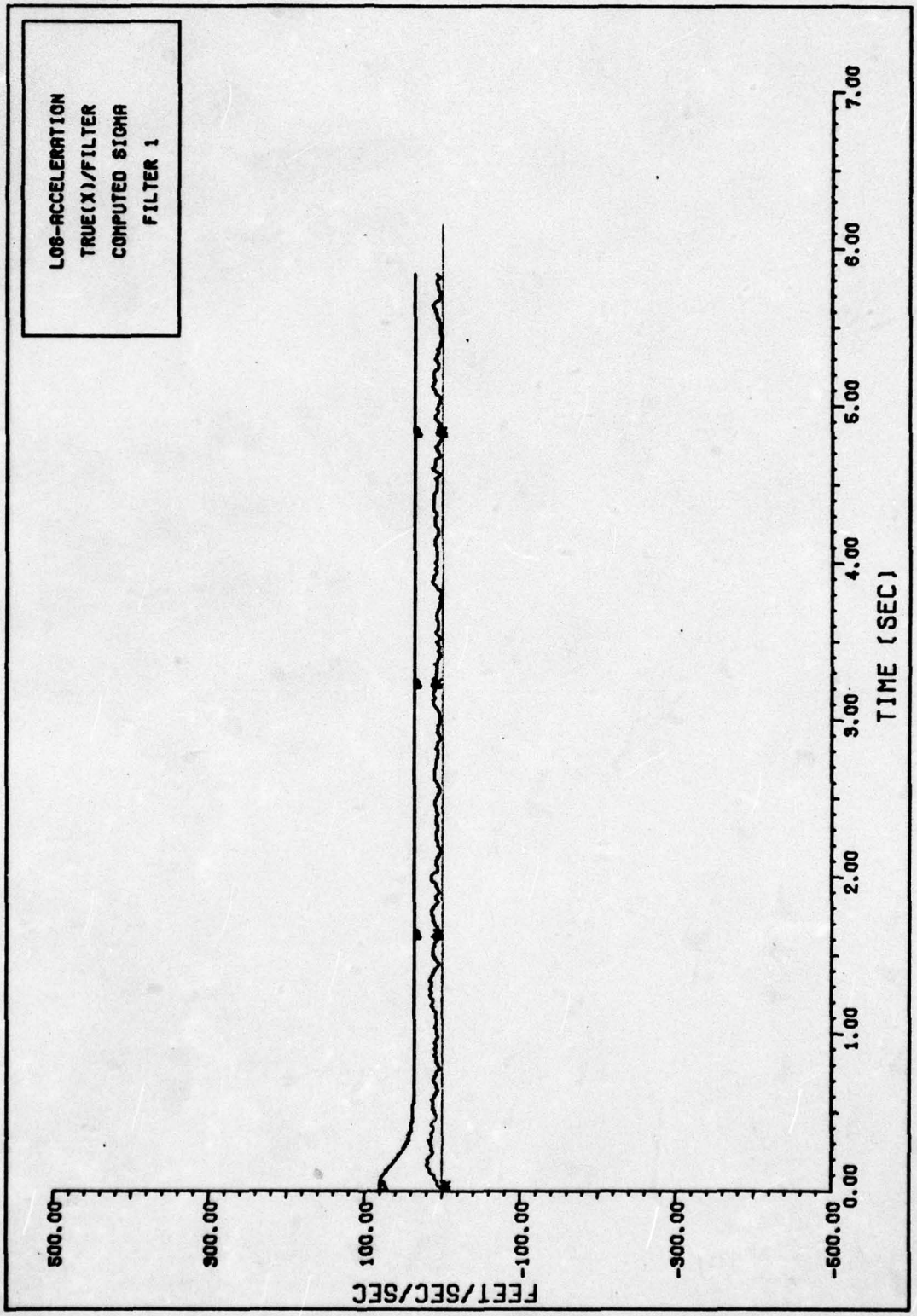


FIGURE 82. LOS-ACCELERATION TUNING FILTER 1

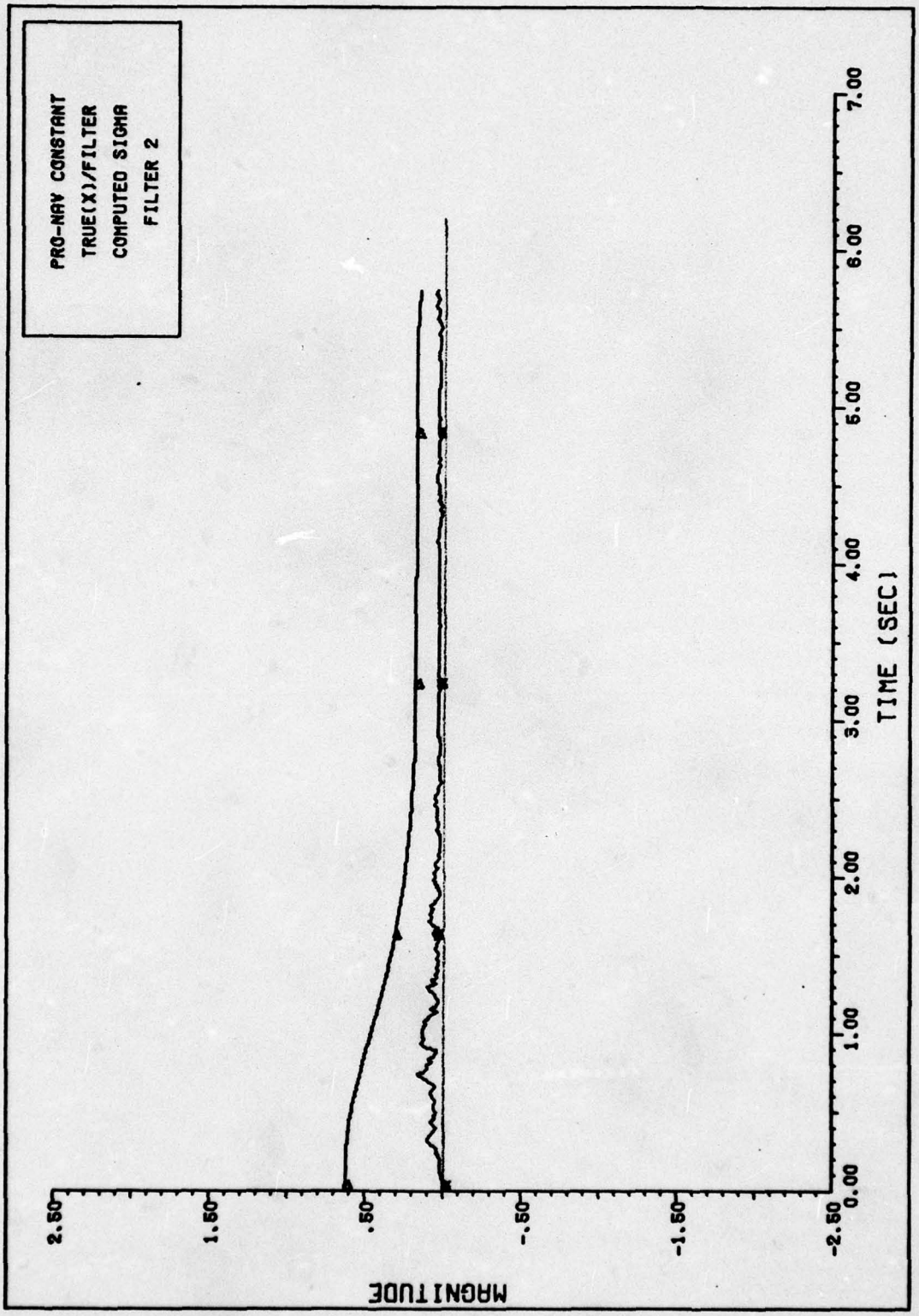


FIGURE 83. PRO-NAV CONSTANT TUNING FILTER 2

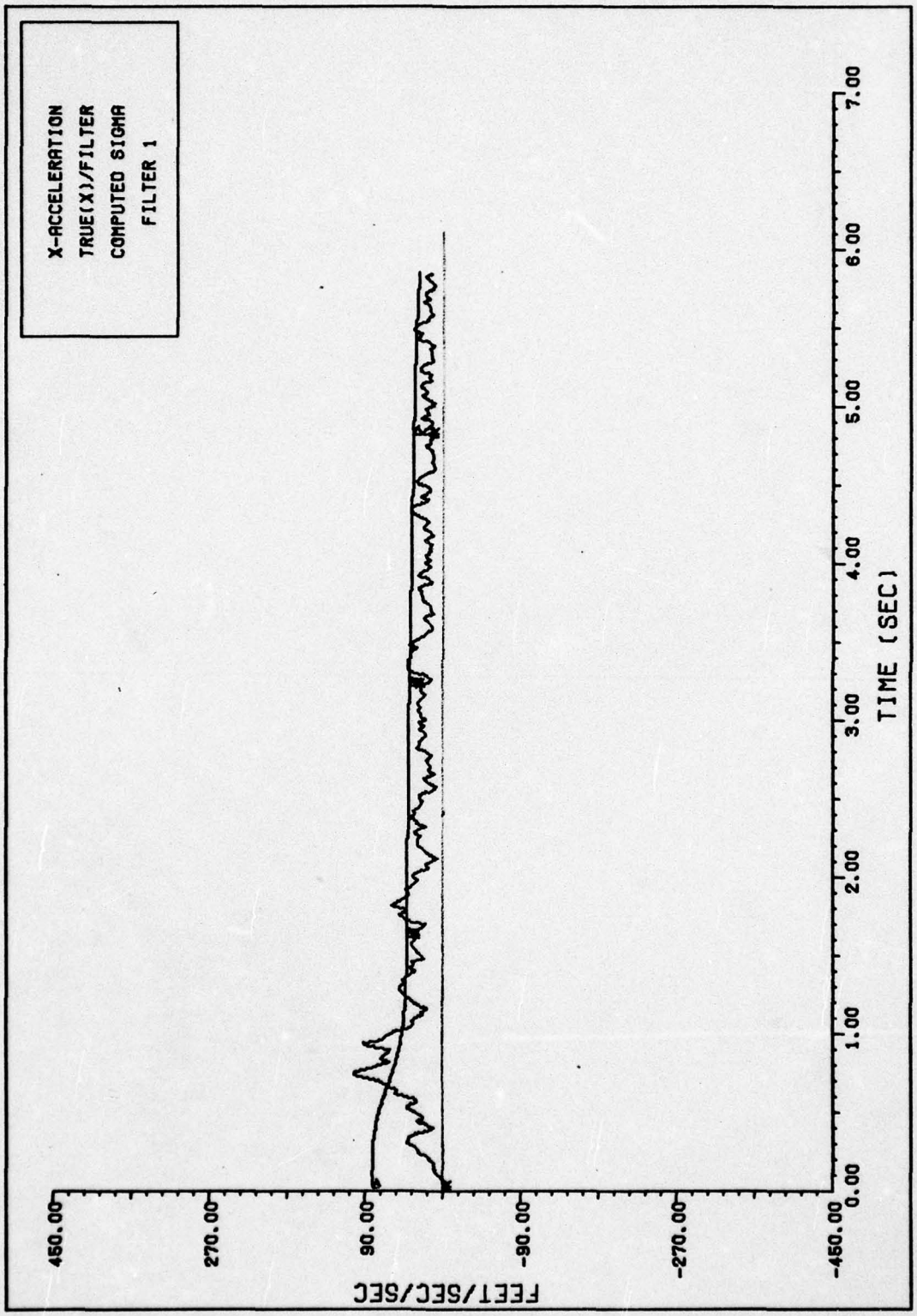


FIGURE 84. X-ACCELERATION TUNING FILTER 1

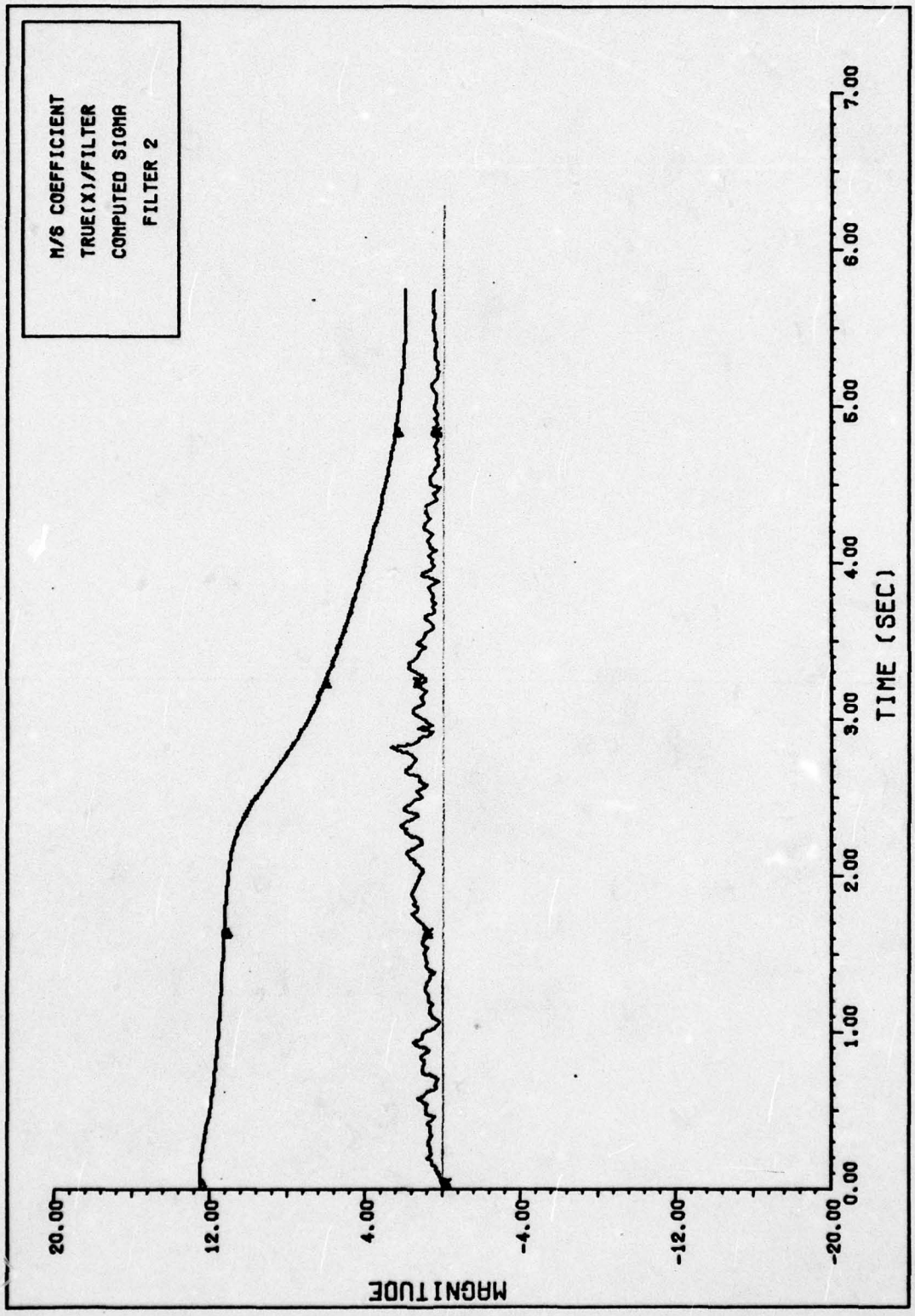


FIGURE 85. M/S COEFFICIENT TUNING FILTER 2

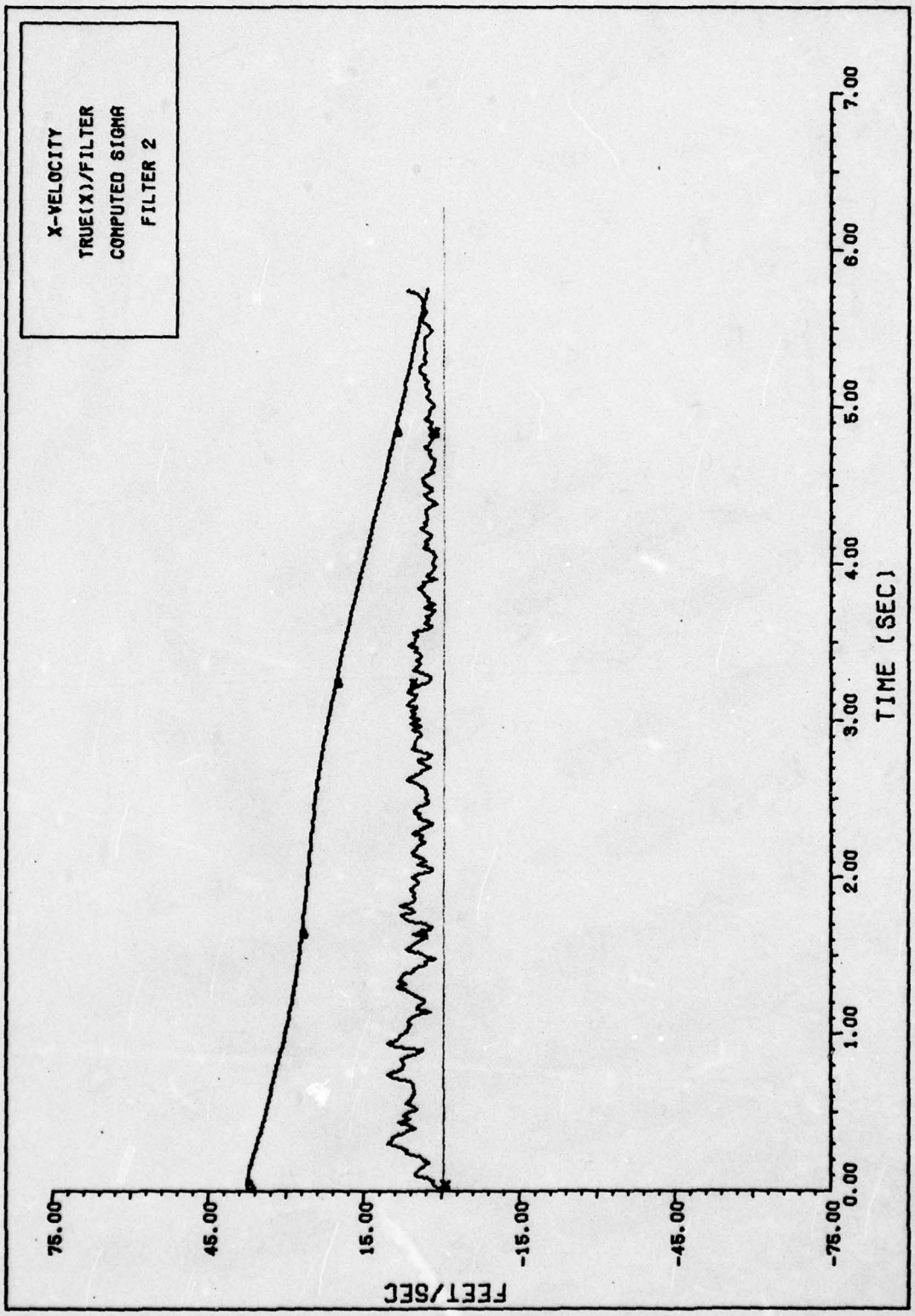


FIGURE 86. X-VELOCITY TUNING FILTER 2

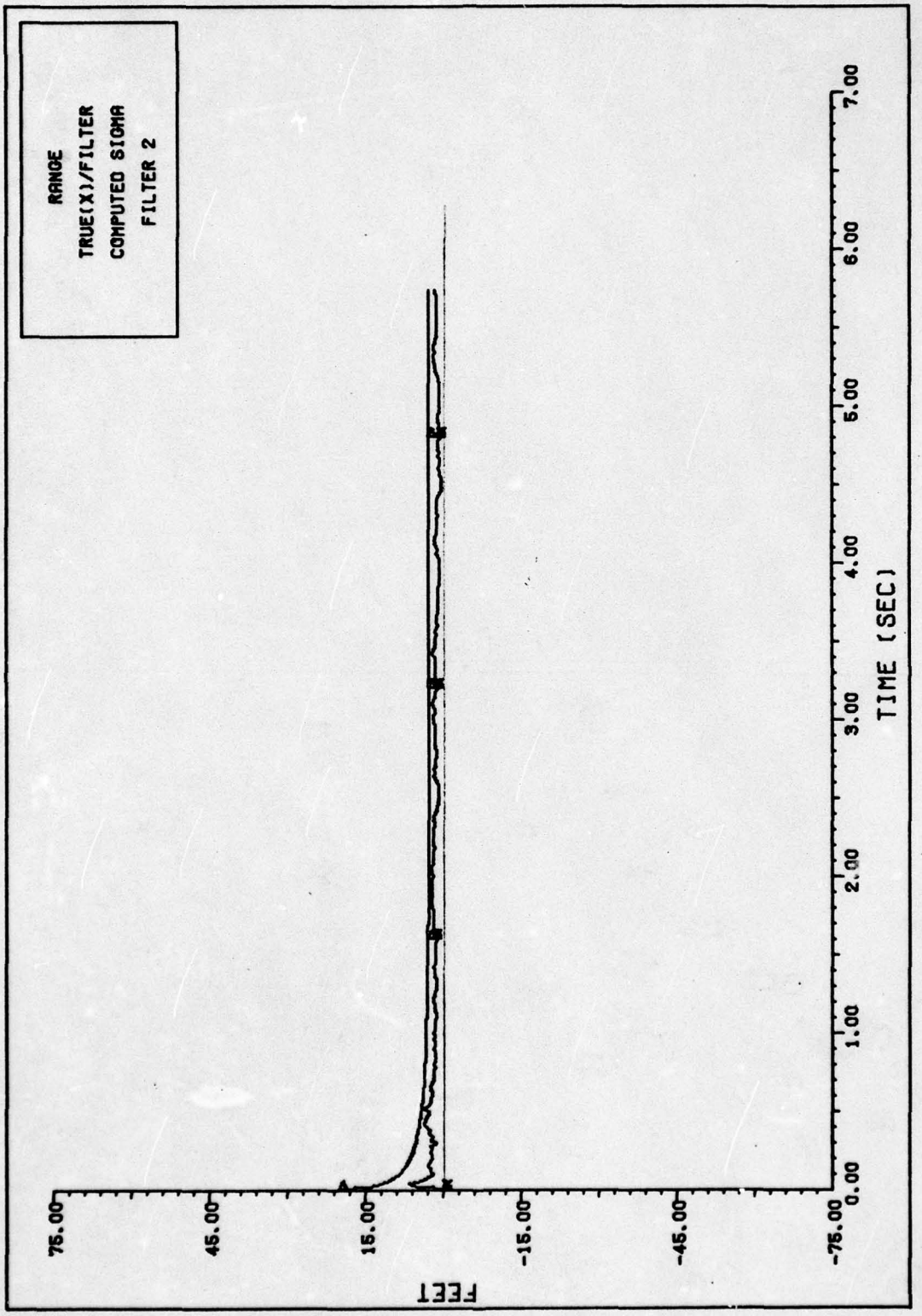
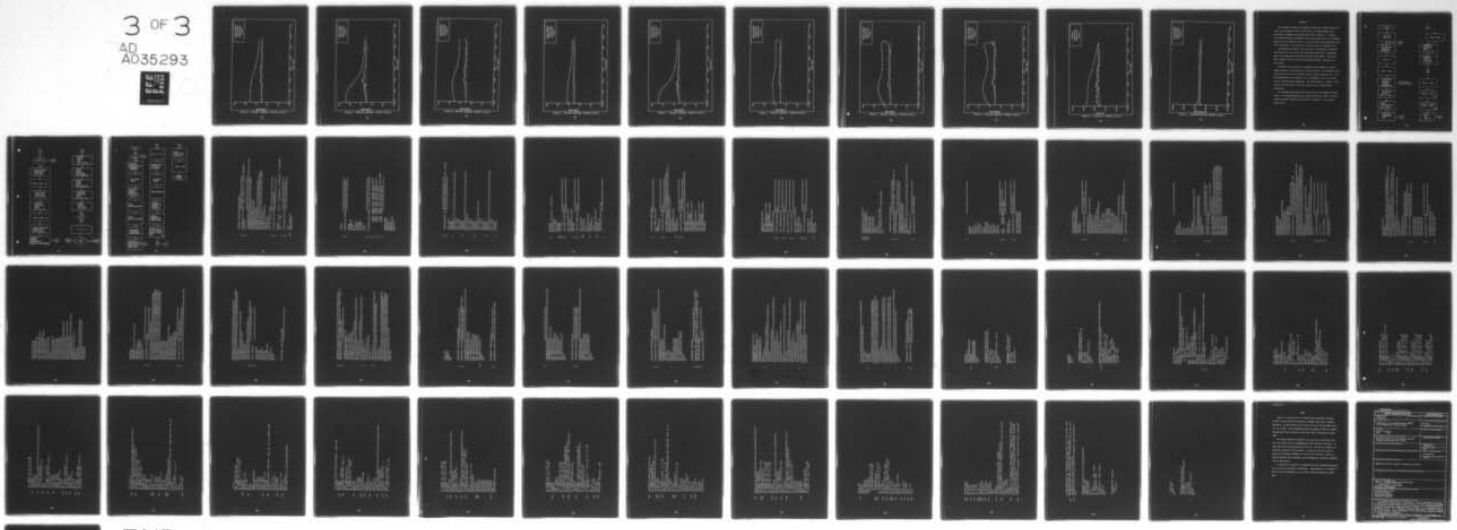


FIGURE 87. RANGE TUNING FILTER 2

AD-A035 293 AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OHIO SCH--ETC F/G 19/5
APPLICATION OF AN EXTENDED KALMAN FILTER TO AN ADVANCED FIRE CO--ETC(U)
DEC 76 R N LUTTER

UNCLASSIFIED GE/EE/76-31 NL

3 OF 3
AD
A035293



END

DATE
FILMED
3-77

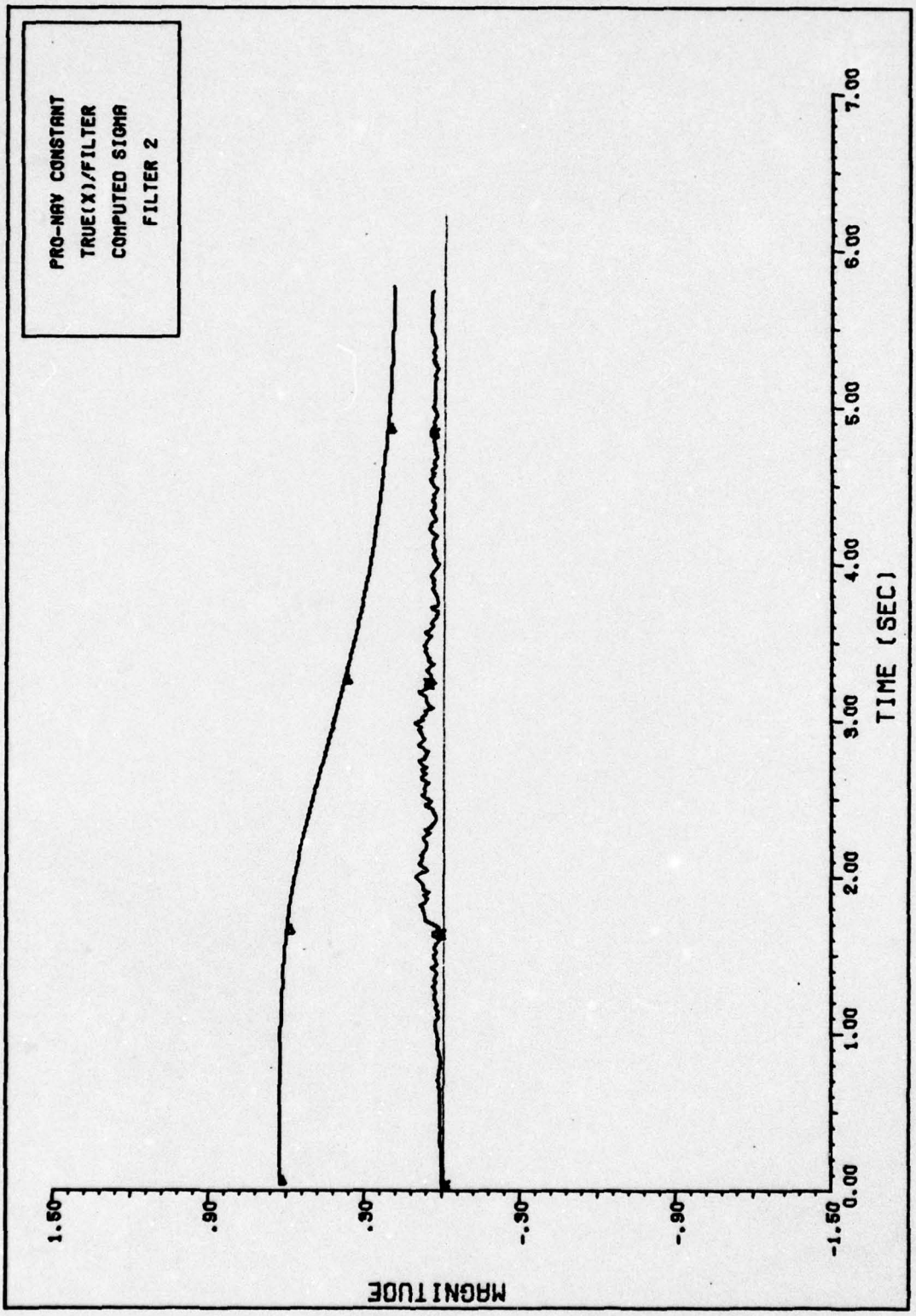


FIGURE 88. PRO-NAV CONSTANT TUNING FILTER 2

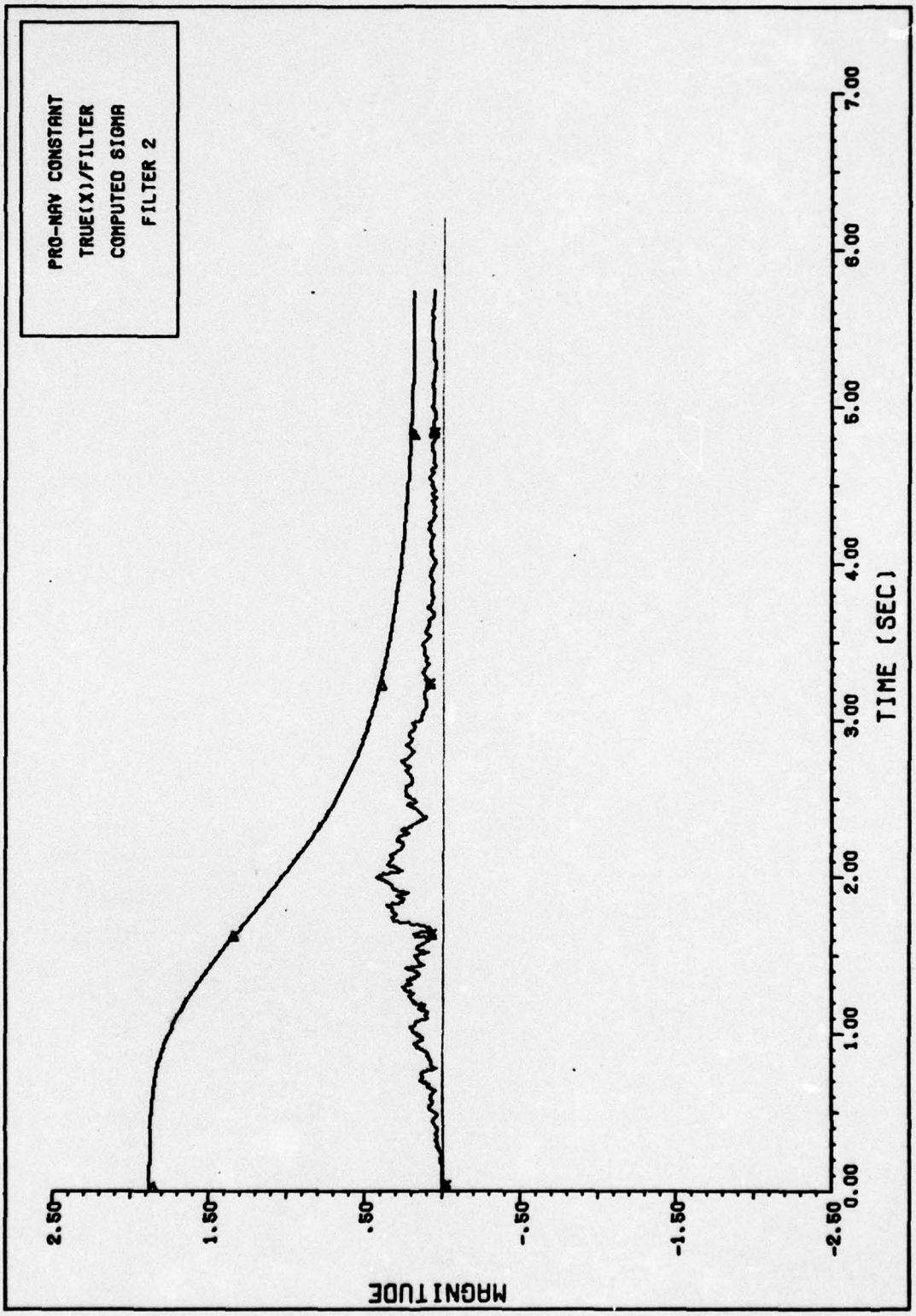


FIGURE 89. PRO-NAV CONSTANT TUNING FILTER 2

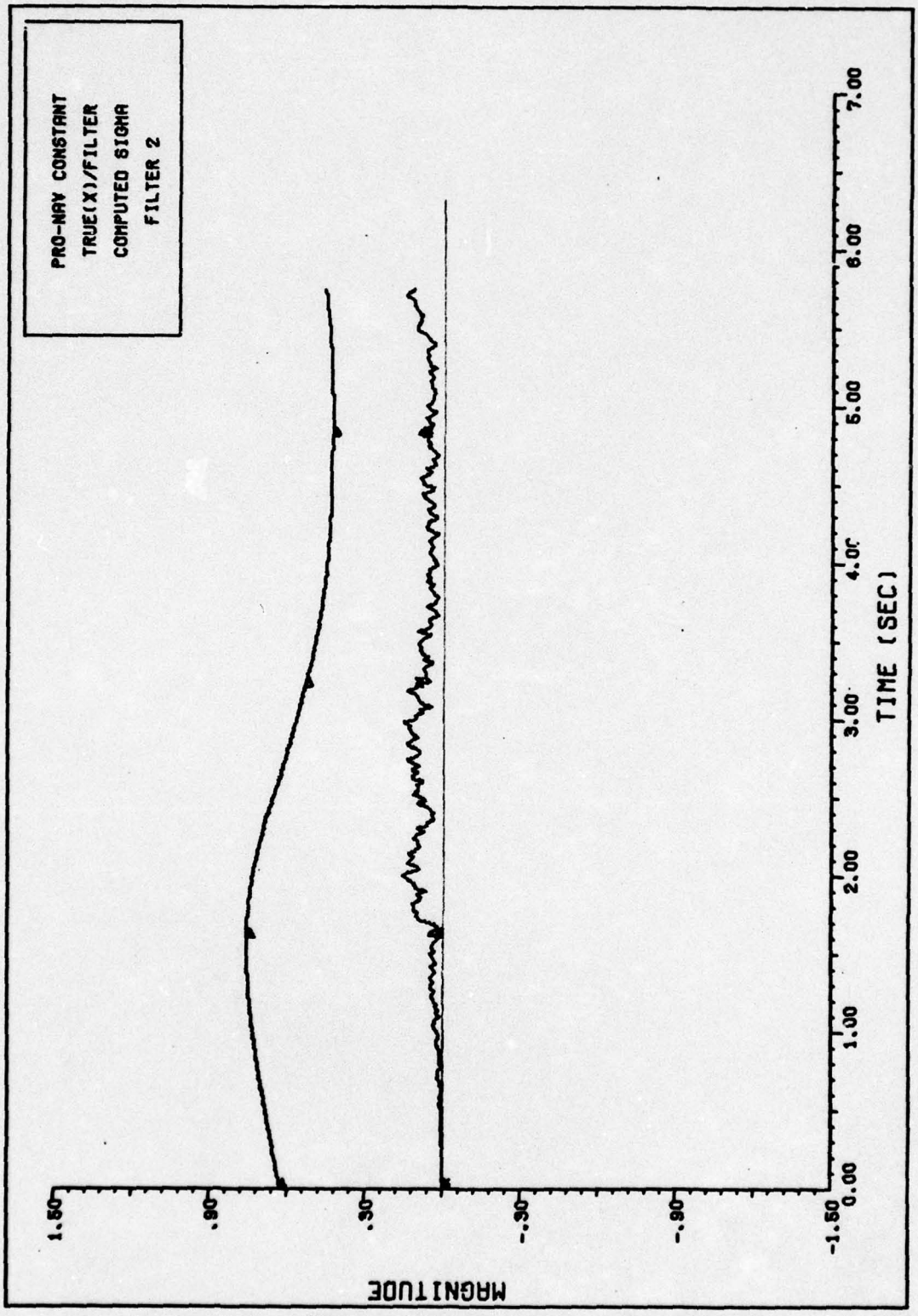


FIGURE 90. PRO-NAV CONSTANT TUNING FILTER 2

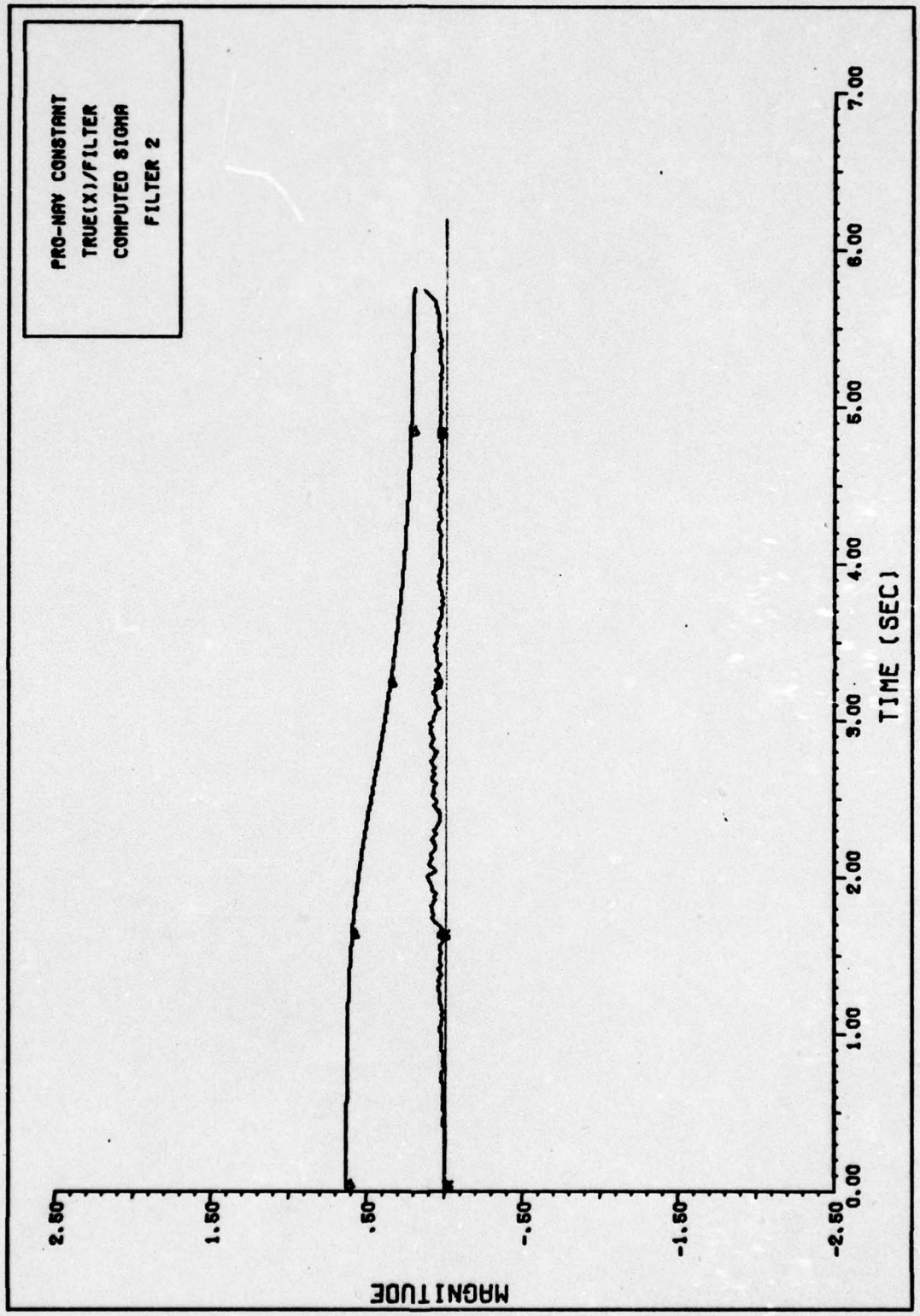


FIGURE 91. PRO-NAV CONSTANT TUNING FILTER 2

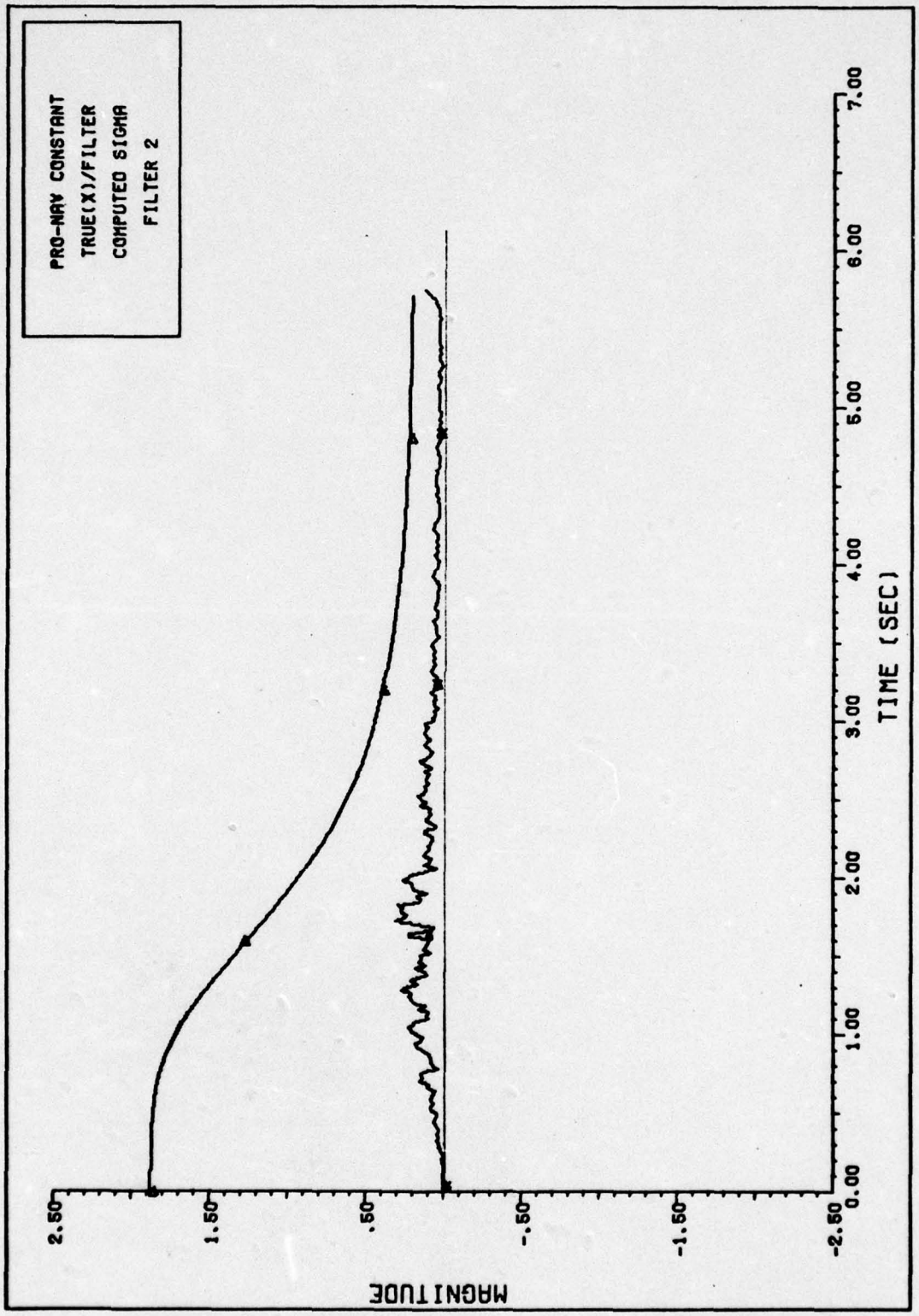


FIGURE 92. PRO-NAV CONSTANT TUNING FILTER 2

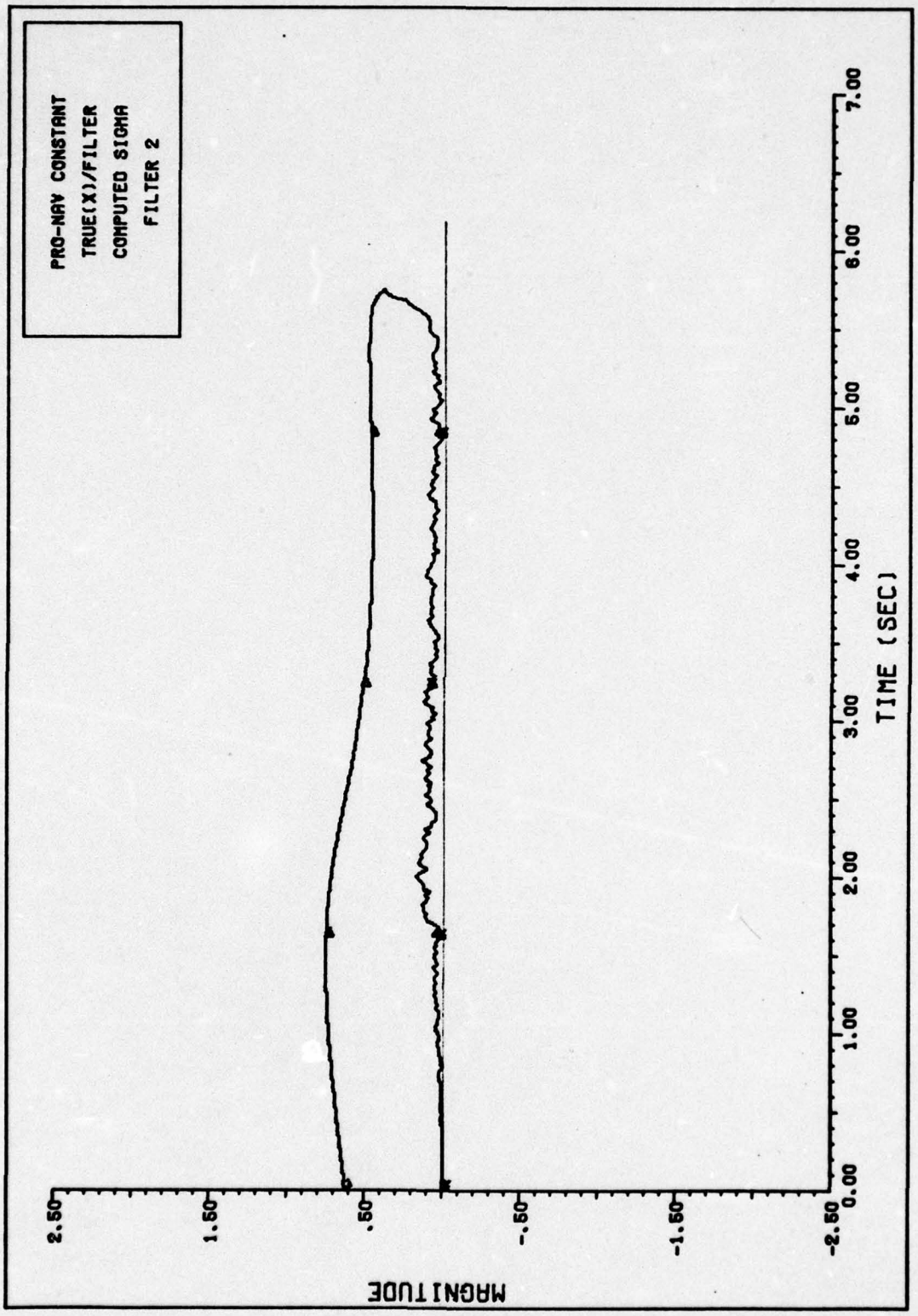


FIGURE 93. PRO-NAV CONSTANT TUNING FILTER 2

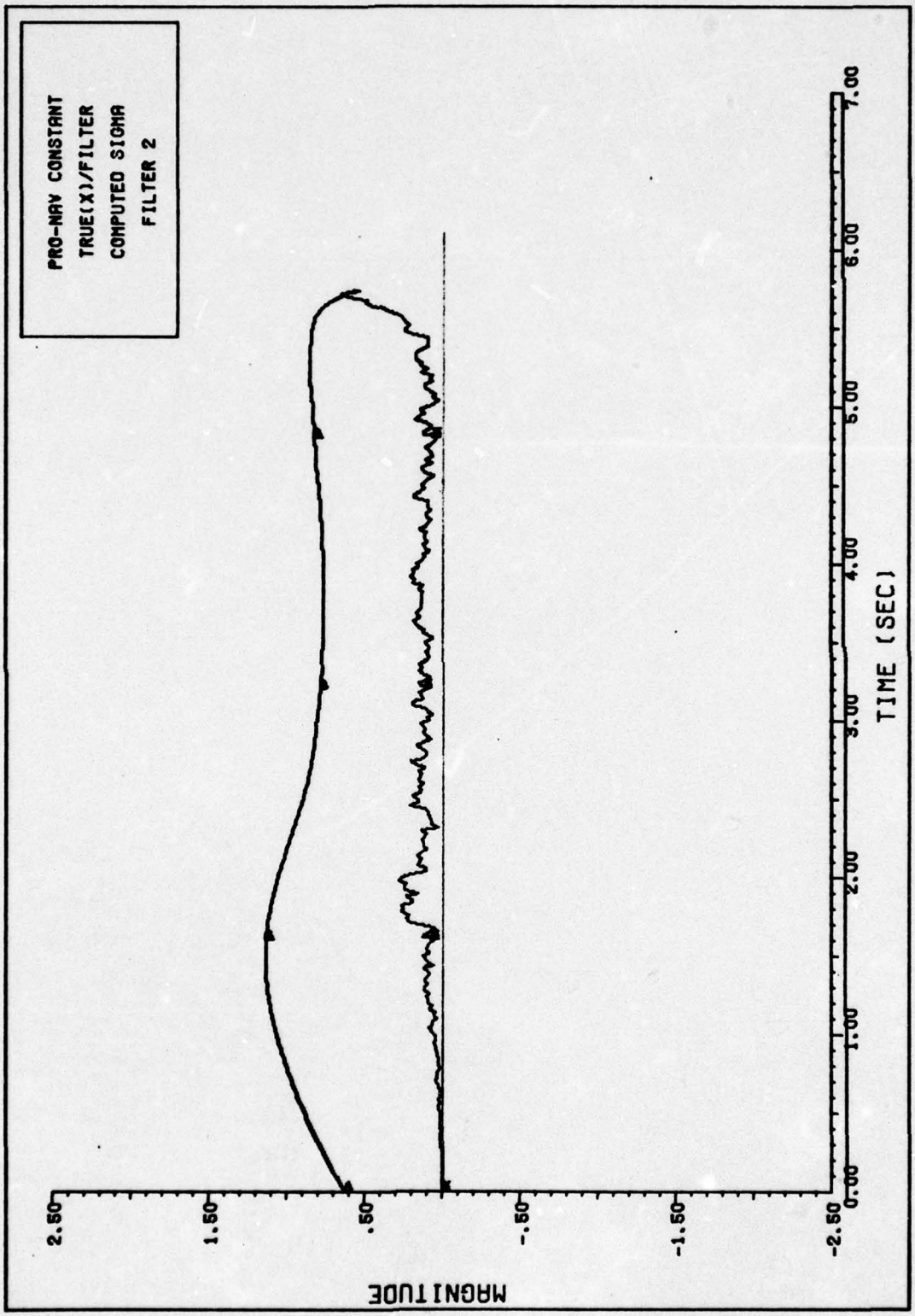


FIGURE 94. PRO-NAV CONSTANT TUNING FILTER 2

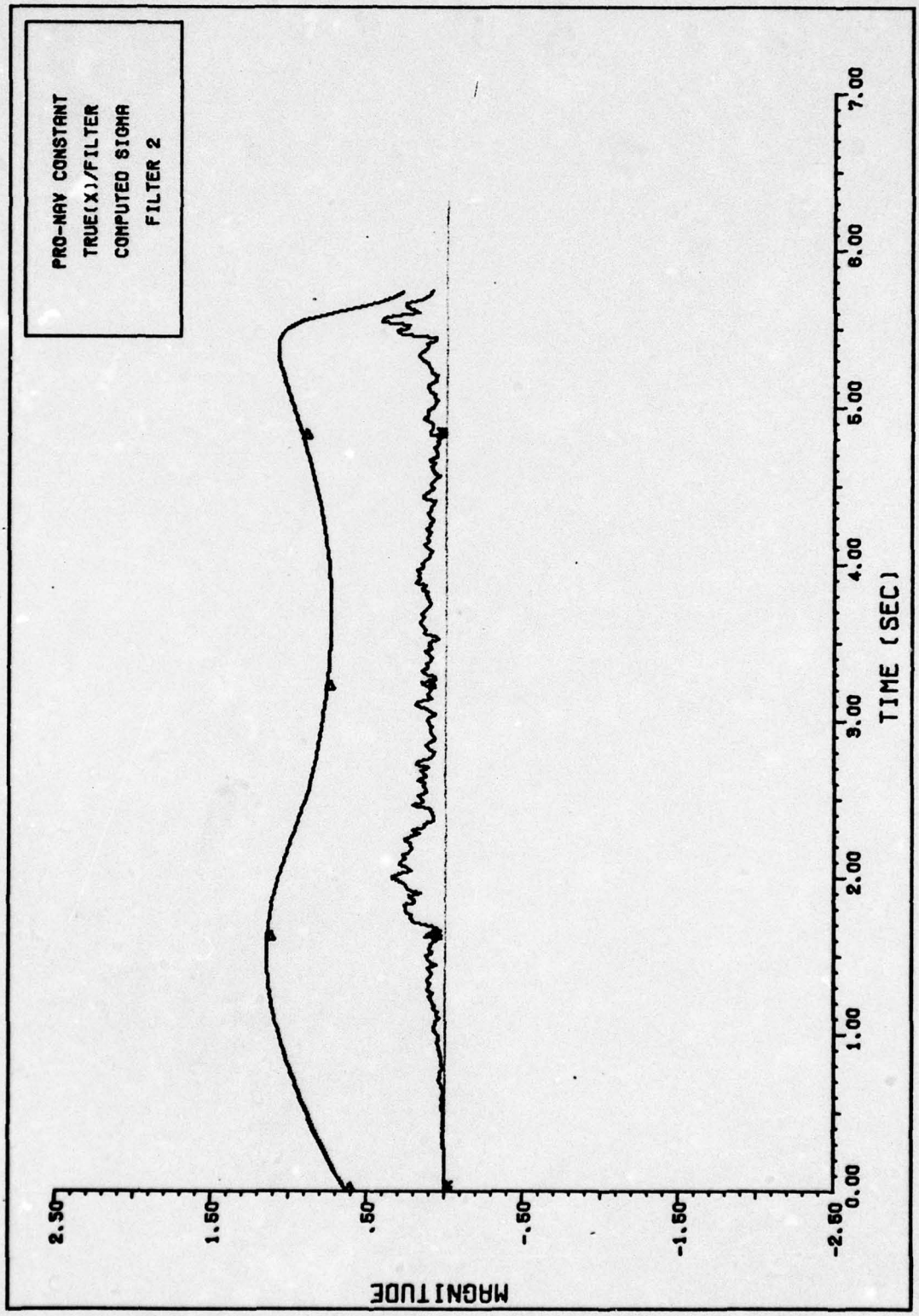


FIGURE 95. PRO-NAV CONSTANT TUNING FILTER 2

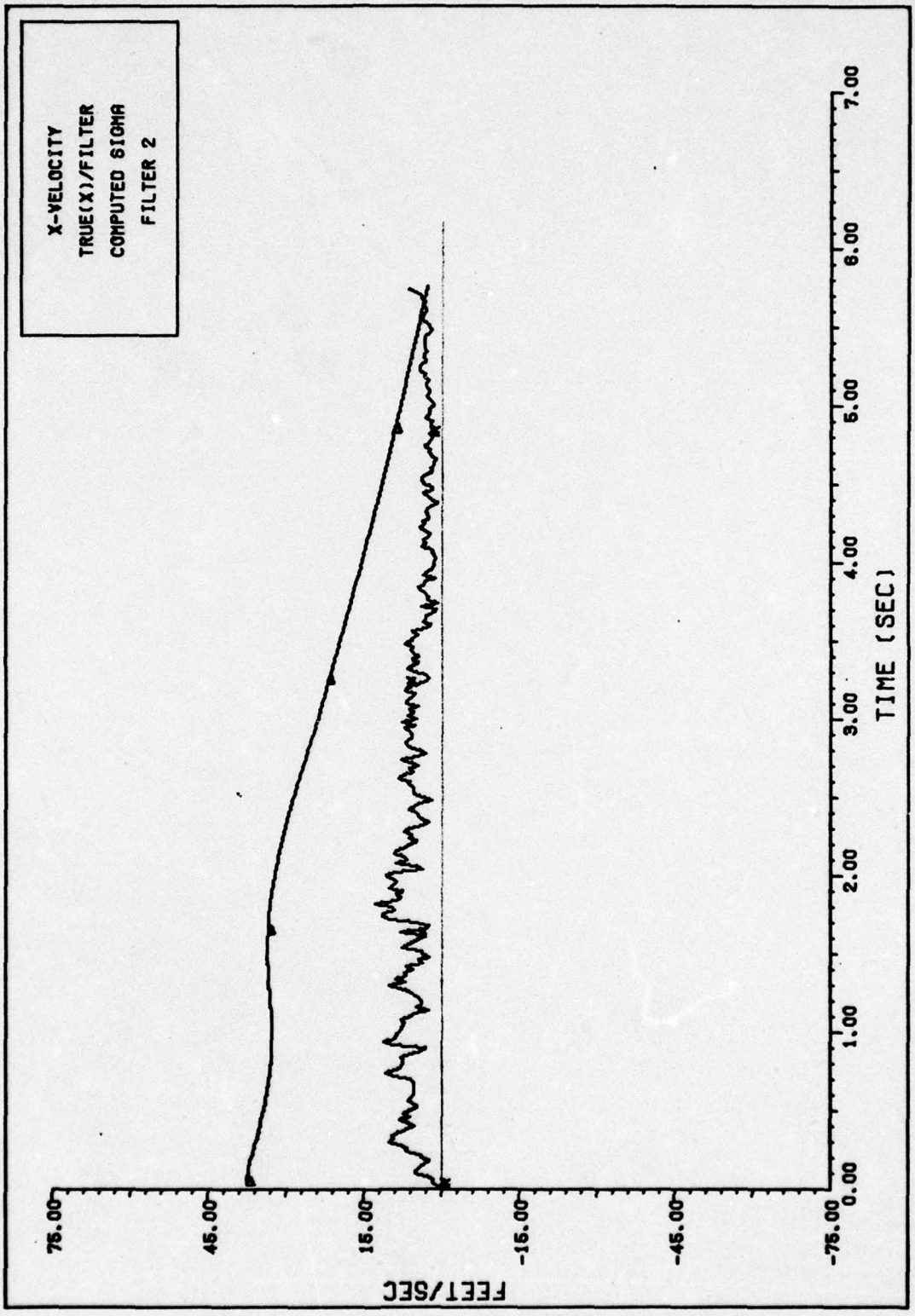


FIGURE 96. X-VELOCITY TUNING FILTER 2

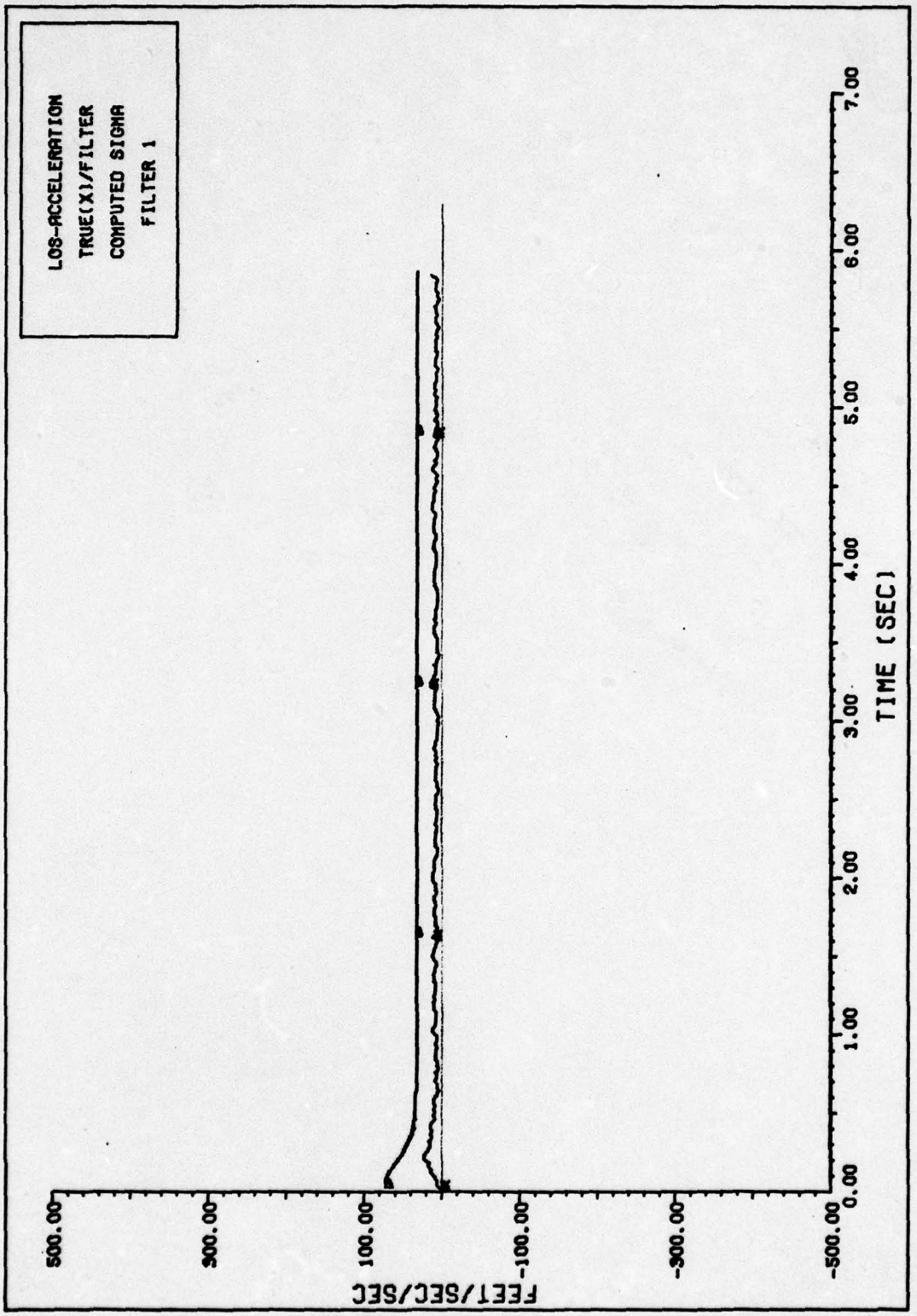


FIGURE 97. LOS-ACCELERATION TUNING FILTER 1

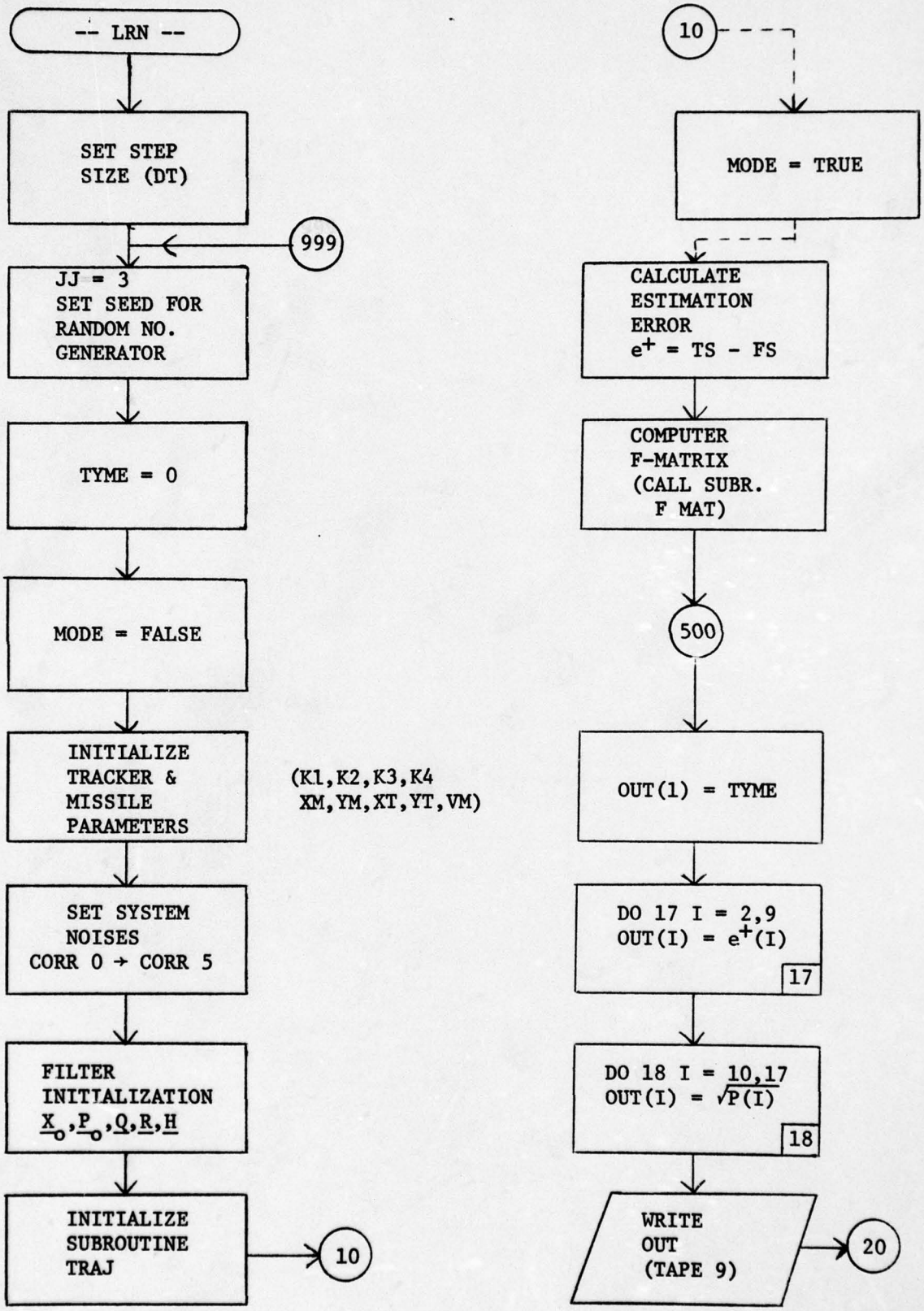
Appendix C

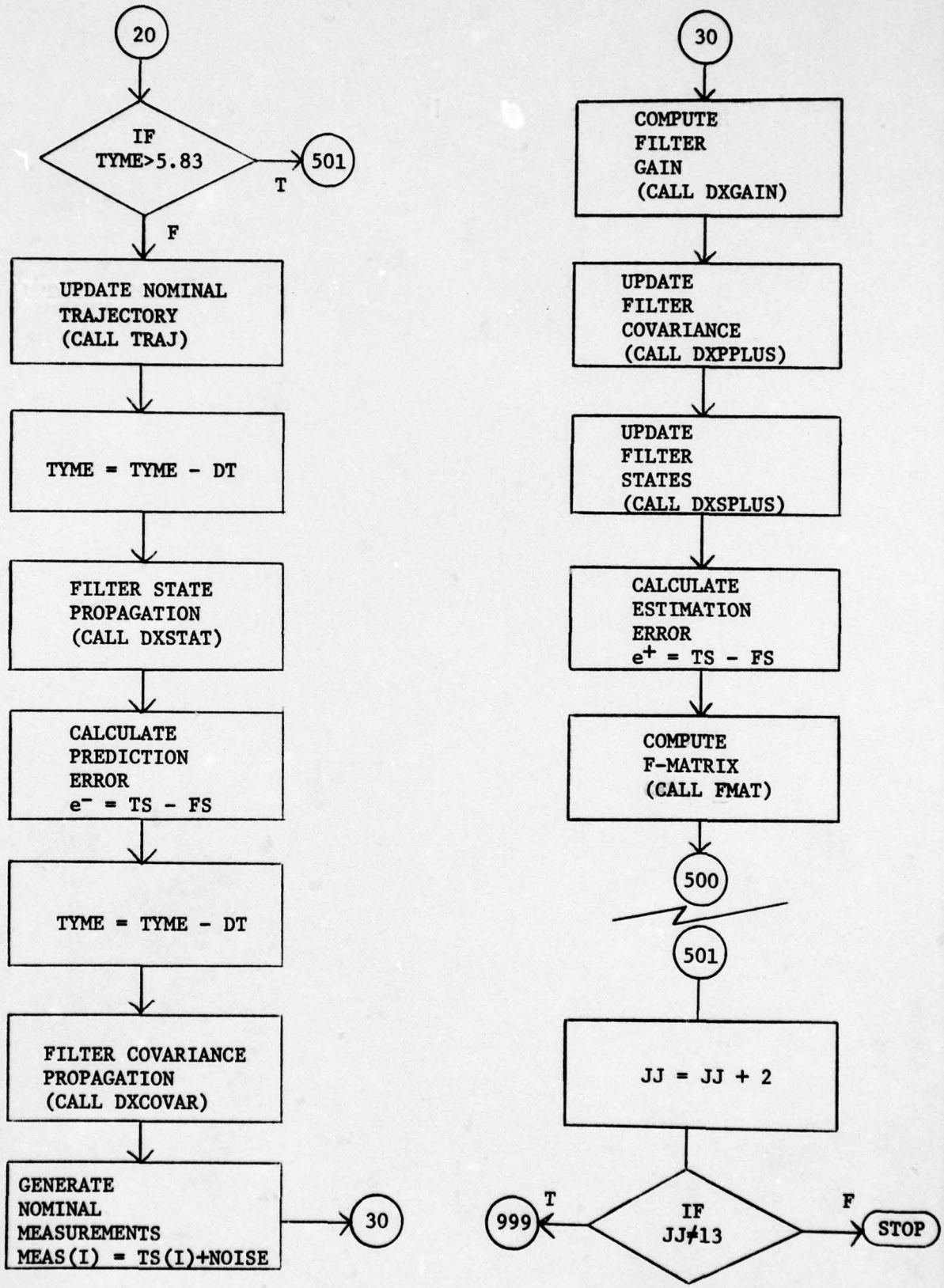
This Appendix contains the computer listing and an abbreviated flowchart of the program utilized in the study. The main program, LRN, initializes, propagates and updates the filter estimates. It computes the prediction errors, estimation errors, and square roots of the diagonal elements of the covariance matrix, and prints these values along with time onto a local file. The local file is then stored on a permanent file.

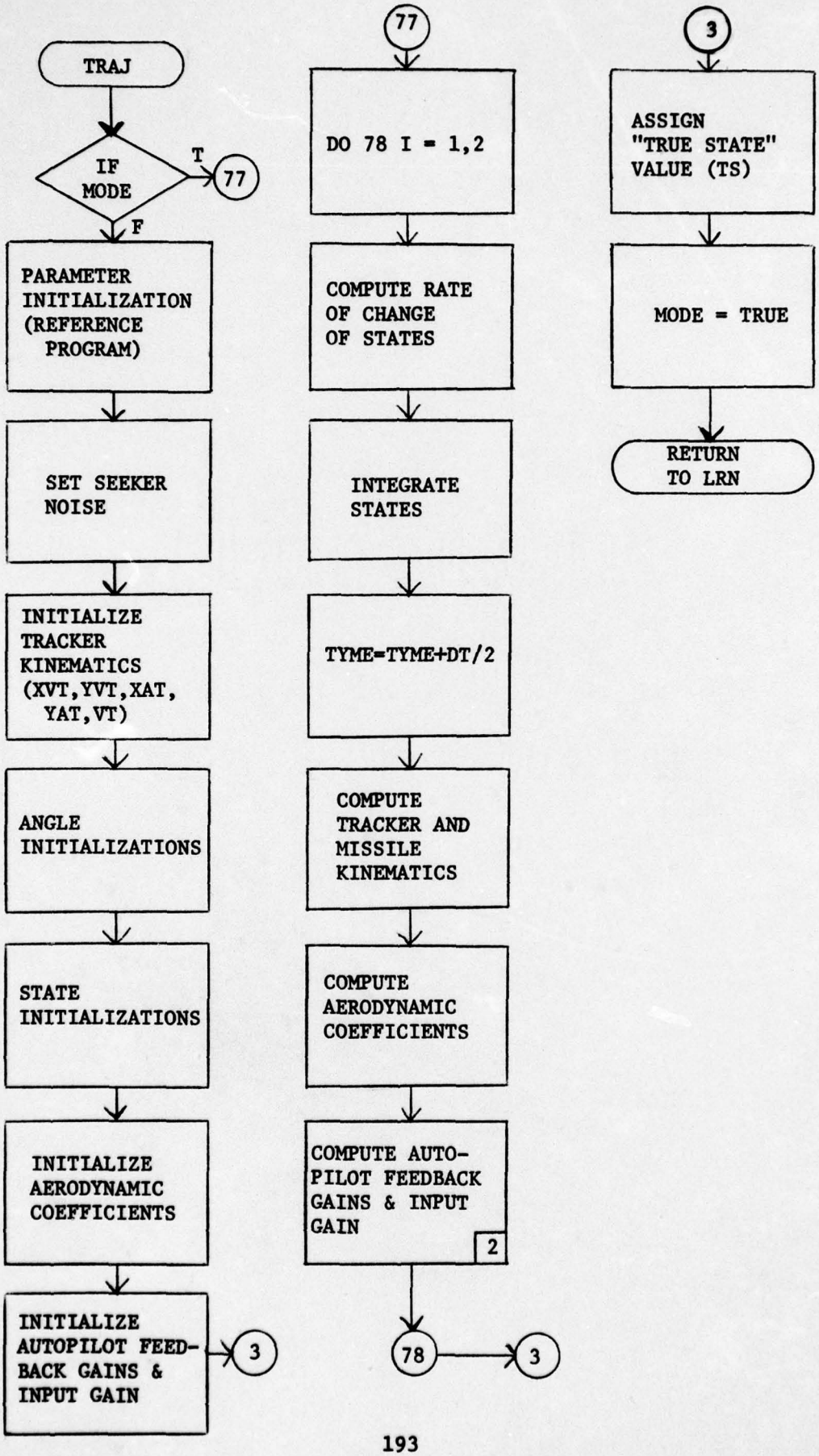
The nominal measurements for the filter are generated by combining the value of the true states (from Subroutine TRAJ) with a measurement noise of the appropriate strength (from Subroutine NOISE). Subroutine TRAJ, together with the noises from Subroutine NOISE, constitute the truth model.

The Monte Carlo analysis is accomplished by "seeding" the random number generator, noise generator, Subroutine NOISE. The "seeding" allows repeatability of the noise sequence over any single simulation run. The seed, designated in the program as JJ, is changed for each of the five runs for a given filter simulation. The same sequence of "seeds" is then used for the five runs of the other filter (over the same nominal trajectory).

The flowchart and listing that follow are for the Unknown Parameter Filter. The listing contains the integration routine, INTEE. This subroutine was made available by Capt Eric K. Lindbery of the AF Flight Dynamics Lab.


-- LRN --
SET STEP SIZE (DT)
999
JJ = 3
SET SEED FOR RANDOM NO. GENERATOR
TYME = 0
MODE = FALSE
INITIALIZE TRACKER & MISSILE PARAMETERS
(K1,K2,K3,K4 XM,YM,XT,YT,VM)
SET SYSTEM NOISES
CORR 0 → CORR 5
FILTER INITIALIZATION
$\underline{X}_o, \underline{P}_o, \underline{Q}, \underline{R}, \underline{H}$
INITIALIZE SUBROUTINE TRAJ
10
10
MODE = TRUE
CALCULATE ESTIMATION ERROR
$e^+ = TS - FS$
COMPUTER F-MATRIX (CALL SUBR. F MAT)
500
OUT(1) = TYME
DO 17 I = 2,9
$OUT(I) = e^+(I)$
17
DO 18 I = 10,17
$OUT(I) = \sqrt{P(I)}$
18
WRITE OUT (TAPE 9)
20


20
IF
TYME>5.83
T
501
F
UPDATE NOMINAL
TRAJECTORY
(CALL TRAJ)
TYME = TYME - DT
FILTER STATE
PROPAGATION
(CALL DXSTAT)
CALCULATE
PREDICTION
ERROR
e⁻ = TS - FS
TYME = TYME - DT
FILTER COVARIANCE
PROPAGATION
(CALL DXCOVAR)
GENERATE
NOMINAL
MEASUREMENTS
MEAS(I) = TS(I)+NOISE
30
30
COMPUTE
FILTER
GAIN
(CALL DXGAIN)
UPDATE
FILTER
COVARIANCE
(CALL DXPPLUS)
UPDATE
FILTER
STATES
(CALL DXSPLUS)
CALCULATE
ESTIMATION
ERROR
e⁺ = TS - FS
COMPUTE
F-MATRIX
(CALL FMAT)
500
501
JJ = JJ + 2
IF
JJ≠13
T
999
F
STOP


TRAJ
IF MODE
T
77
F
PARAMETER INITIALIZATION (REFERENCE PROGRAM)
SET SEEKER NOISE
INITIALIZE TRACKER KINEMATICS (XVT,YVT,XAT, YAT,VT)
ANGLE INITIALIZATIONS
STATE INITIALIZATIONS
INITIALIZE AERODYNAMIC COEFFICIENTS
INITIALIZE AUTOPILOT FEED-BACK GAINS & INPUT GAIN
3
77
DO 78 I = 1,2
COMPUTE RATE OF CHANGE OF STATES
INTEGRATE STATES
TYME=TYME+DT/2
COMPUTE TRACKER AND MISSILE KINEMATICS
COMPUTE AERODYNAMIC COEFFICIENTS
COMPUTE AUTO-PILOT FEEDBACK GAINS & INPUT GAIN
2
78
3
3
ASSIGN "TRUE STATE" VALUE (TS)
MODE = TRUE
RETURN TO LRN

```
      PROGRAM LRN(INPUT,OUTPUT,TAPE9)
C                          ********************************
C     PROGRAM LRN IS A MONTE CARLO ANALYSIS OF A MISSILE TRACKING
C     ALGORITHM
C                          ********************************
      COMMON/SITE/A1(8,8),A2(8,8),A3(8,8),A4(8,3),A5(3,3),A6(3,3),A7(8)
      COMMON/SITE1/K1,K2,K3,K4,W,VM,XT,YT,XM,YM,MODE1
      COMMON/SITE10/XVT,YVT,ATLOS
      COMMON/SITE3/Q(8,8),F(8,8)
      COMMON/SITE4/MEAS(3)
      DIMENSION TS(8),H(3,8),R(3,3),FSD(8),FPD(8,8)
      DIMENSION FS(8),FP(8,8),FSPLUS(8),PREERR(8),ESTERR(8)
      DIMENSION PL1(850),PL2(850),PL3(850),PL4(850),TIM(850)
      DIMENSION OUT(17)
      EXTERNAL DXSTAT,DXCOVAR
      INTEGER C,D
      LOGICAL MODE1
      REAL K1,K2,K3,K4,MEAS
C                          ********************************
C     PROGRAM AND INTEGRATION PARAMETERS
C                          ********************************
      DT=.02
      NSET=1  $ N1=8  $ N2=64  $ DTIN=.0050  $ DTMIN=.002  $ TOLR=.0014
      C=1   $   D=5
C                          ********************************
C     INITIALIZATION FOR MONTE CARLO REPEATABILITY
C                          ********************************
      JJ=3
999   CALL RANSET(JJ)
      MODE1=.FALSE.
```

```
      TYME=0
C                          *********************************
C     TRACKER AND MISSILE INITIALIZATION PARAMATERS
C                          *********************************
      K1=500
      K2=2864.51
      K3=800
      K4=0
      W=.17455
      VM=3051
      XM=-3420
      YM=-9397
      XT=0.
      YT=0.
C                          *********************************
C     INITIALIZTION VALUES FOR SYSTEM AND FILTER NOISES
C
C           CORR0  (ANGLE MEAS. SCINTILLATION NOISE)
C           CORR1  (ANGLE MEAS. THERMAL NOISE)
C           CORR2  (RANGE MEAS. SCINTILLATION NOISE)
C           CORR3  (RANGE MEAS. THERMAL NOISE)
C           CORR4  (RANGE RATE MEAS. SCINTILLATION NOISE)
C           CORR5  (RANGE RATE THERMAL NOISE)
C                          *********************************
      CORR0=.00168
      CORR1=.00125
      CORR2=10
      CORR3=11.7
      CORR4=4.24264
      CORR5=7.
```

```
C                          *********************************
C      INITIALIZTION OF FILTER STATE, COVARIANCE, Q, R, AND H MATRIX
C                          *********************************
       DO 1 I=1,8
       FS(I)=0
1      CONTINUE
       READ*,FS(1),FS(2),FS(3),FS(4),FS(5),FS(6),FS(7),FS(8)
       DO 2 I=1,8
       DO 3 J=1,8
       FP(I,J)=0
3      CONTINUE
2      CONTINUE
       READ*,FP(1,1),FP(2,2),FP(3,3),FP(4,4),FP(5,5),FP(6,6),
      XFP(7,7),FP(8,8)
       DO 4 I=1,8
       DO 5 J=1,8
       R(I,J)=0
5      CONTINUE
4      CONTINUE
       READ*,R(1,1),R(2,2),R(3,3)
       DO 6 I=1,8
       DO 7 J=1,8
       Q(I,J)=0
7      CONTINUE
6      CONTINUE
       READ*,Q(1,1),Q(2,2),Q(3,3),Q(5,5),Q(6,6),Q(7,7),Q(8,8)
       DO 8 I=1,3
       DO 9 J=1,8
       H(I,J)=0
9      CONTINUE
```

```
8       CONTINUE
        H(1,1)=1  $ H(2,4)=1  $ H(3,5)=1
        DO 10 I=1,8
        DO 11 J=1,8
        F(I,J)=0
11      CONTINUE
10      CONTINUE
C                               *********************************
C       TRAJECTORY INITIALIZTION
C                               *********************************
        CALL TRAJ(TYME,DT,TS)
        DO 12 I=1,8
        ESTERR(I)=TS(I)-FS(I)
12      CONTINUE
C                               *********************************
C       FILTER F(PARTIALS) MATRIX
C                               *********************************
        CALL FMAT(FS)
500     OUT(1)=TYME
        DO 17 I=2,9
        J=I-1
        OUT(I)=ESTERR(J)
17      CONTINUE
        DO 18 I=10,17
        J=I-9
        OUT(I)=SQRT(FP(J,J))
18      CONTINUE
        WRITE(9) OUT
        IF(TYME.GT.5.73) GO TO 501
C                               *********************************
```

```
C       TRAJECTORY PROPAGATION
C                               *********************************
        CALL TRAJ(TYME,DT,TS)
        C=C+1
        TYME=TYME-DT
C                               *********************************
C       FILTER STATE PROPAGATION
C                               *********************************
        CALL INTEE(NSET,N1,TYME,FS,DXSTAT,DTIN,DT,DTMIN,TOLR)
        FS(3)=(FS(5)*COS(FS(1))**2+XVT*COS(FS(1))-FS(2)*COS(FS(1)))/
       XSIN(FS(1))+FS(5)*SIN(FS(1))+YVT
        DO 14 I=1,8
        PRESRR(I)=TS(I)-FS(I)
14      CONTINUE
        TYME=TYME-DT
C                               *********************************
C       FILTER COVARIANCE PROPAGATION
C                               *********************************
        CALL INTEE(NSET,N2,TYME,FP,DXCOVAR,DTIN,DT,DTMIN,TOLR)
        X0D=-2*X0
        CALL NOISE(CORR0,0.,CORRUP)
        X0=X0D*DT+CORRUP
        CALL NOISE(CORR1,0.,CORRUP)
        X1=CORRUP
        MEAS(1)=TS(1)+X0+X1
        X2D=-2*X2
        CALL NOISE(CORR2,0.,CORRUP)
        X2=X2D*DT+CORRUP
        CALL NOISE(CORR3,0.,CORRUP)
        X3=CORRUP
```

```
      MEAS(2)=TS(4)+X2+X3
      X4D=-X4
      CALL NOISE(CORR4,0.,CORRUP)
      X4=X4D*DT+CORRUP
      CALL NOISE(CORR5,0.,CORRUP)
      X5=CORRUP
      MEAS(3)=TS(5)+X4+X5
C                          **********************************
C     FILTER GAIN COMPUTATION
C                          **********************************
      CALL DXGAIN(H,FP,R)
C                          **********************************
C     FILTER COVARIANCE UPDATE
C                          **********************************
      CALL DXPPLUS(FP,H)
C                          **********************************
C     FILTER STATE UPDATE
C                          **********************************
      CALL DXSPLUS(FS,FSPLUS)
      DO 15 I=1,8
      FS(I)=FSPLUS(I)
15    CONTINUE
C                          **********************************
C     FILTER F(PARTIALS) MATRIX
C                          **********************************
      CALL FMAT(FS)
      DO 16 I=1,8
      ESTERR(I)=TS(I)-FS(I)
16    CONTINUE
      GO TO 500
```

```
400   FORMAT(2X,F5.2,8(E14.5))
401   FORMAT(7X,8(E14.5))
402   FORMAT(/(5X,8E12.5))
501   PRINT*,"XT=",XT
      PRINT*,"XM=",XM
      PRINT*,"YT=",YT
      PRINT*,"YM=",YM
      PRINT*
      JJ=JJ+2
      IF(JJ.NE.13) GO TO 999
      STOP "MISSILE HAS HIT OR MISSED TARGET"
      END


      SUBROUTINE TRAJ(X,DY,TS)
C                     *********************************
C     SUBROUTINE TRAJ GENERATES THE NOMINAL TRAJECTORIES FOR THE
C     MONTE CARLO ANALYSIS
C                     *********************************
      COMMON/SITE1/K1,K2,K3,K4,W,VM,XT,YT,XM,YM,MODE1
      COMMON/SITE1A/XVT,YVT,ATLOS
      DIMENSION TS(8),Y(14),P(14)
      LOGICAL MODE1
      REAL MACH,LOSR,MQ,MA,MD,LD,LA,K1,K2,K3,K4,KD,LOA,LAM,LAMN,LOA1,
     XMASS
      IF (MODE1)GO TO 77
C                     *********************************
C     INITIALIZTION OF SYSTEM PARAMETERS
```

```
C                          **********************************
      DX=DY/2
      MASS=4
      IYY=50
      SURF=.137
      PNK=4.0
      OMEGA=7.07
      ZETA=.707
      D=.417
      A=1017
      RHO=.001056
      SIGMA=.003
      XKR=.001
      LAMN=.1
      TAU1=.075
      TAU2=0.3
      LAM=50
      G=32.174
C                          **********************************
C     CORR     (SEEKER SCINTILATION/GLINT NOISE)
C     CORR1    (SEEKER THERMAL NOISE)
C                          **********************************
      CORR=.003
      CORR1=.000934
C                          **********************************
C     INITIALIZATION OF TRACKER KINEMATICS
C                          **********************************
      XVT=K1-K2*W*SIN(W*X)
      YVT=K3+K4*W*COS(W*X)
      XAT=-K2*W**2*COS(W*X)
```

```
      YAT=-K4*W**2*SIN(W*X)
      VT=SQRT(XVT**2+YVT**2)
      GAMT=1.57079-ASIN(XVT/VT)
      XTM=XT-XM
      YTM=YT-YM
C                          *********************************
C     ANGLE INITIALIZATION - SEE FIGURE (13)
C                          *********************************
      LOA=ATAN(XTM/YTM)
      RANGE=SQRT((XT-XM)**2+(YT-YM)**2)
      THETA=4.71239-LOA
      THETA0=THETA
      PHIAC0=THETA-GAMT
      PHIAC1=6.2831854-PHIAC0
      PHIM=ASIN((VT/VM)*SIN(PHIAC0))
      GAMM=LOA-PHIM
      PRINT*,"GAMT=",GAMT
      PRINT*,"LOA=",LOA
      PRINT*,"THETA=",THETA
      PRINT*,"PHIAC0=",PHIAC0
      PRINT*,"PHIAC1=",PHIAC1
      PRINT*,"PHIM=",PHIM
      PRINT*,"GAMM=",GAMM
      XVM=SIN(GAMM)*VM
      YVM=COS(GAMM)*VM
      XATLS=COS(THETA)*XAT+SIN(THETA)*YAT
      ATLOS=XATLS
      YATLS=-SIN(THETA)*XAT+COS(THETA)*YAT
C                          *********************************
C     STATE INITIALIZATION
```

```
C                              **********************************
      Y(1)=0
      Y(2)=PHIM
      Y(3)=1/RANGE
      Y(4)=0
      Y(5)=0
      Y(6)=0
      Y(7)=0
      Y(8)=0.
      Y(9)=0
      Y(10)=PHIAC0
      Y(11)=+SIN(PHIAC1)*XATLS-COS(PHIAC1)*YATLS
      Y(12)=0
      Y(13)=COS(THETA)*XVM+SIN(THETA)*YVM
      Y(14)=-SIN(THETA)*XVM+COS(THETA)*YVM
      DP=(RHO/2)*VM**2
      MACH=VM/A
C                              **********************************
C     INITIALIZTION OF AERODYNAMIC COEFFICIENTS AND
C     STABILITY DERIVATIVES
C                              **********************************
      CN=(Y(6)*3*MASS*2)/(RHO*SJRF*VM**2)
      CMQ=-(4943+MACH*(-2563+MACH*2125))/(2.957+MACH*(-3.078+MACH))
      CMA=-(2.292+MACH*(-2.858+MACH*1.824))/(1.646+MACH*(-2.171+MACH))
      CMD=-(8.294+MACH*(-4.894+MACH*1.191))/(3.220+MACH*(-2.760+MACH))
      CNA=(1.250+MACH*(-.5398+MACH*.2834))/(2.393+MACH*(-2.132+MACH))
      CND=(.3116+MACH*(-.2199+MACH*.0665))/(3.781+MACH*(-3.125+MACH))
      CMA=CMA*57.295780
      CMD=CMD*57.295780
      CNA=CNA*57.295780
```

```
      CND=CND*57.295780
      CDO=2/SQRT(MACH)
      CDI=CN**2/CNA
      MQ=(DP*SURF*D**2*CMQ)/(2*IYY*VM)
      MA=(DP*SURF*D*CMA)/IYY
      MD=(DP*SURF*D*CMD)/IYY
      LD=(DP*SURF*CND)/(MASS*VM)
      LA=(DP*SURF*CNA)/(MASS*VM)
C                          ********************************
C     INITIALIZATION OF FEEDBACK COEFFICIENTS AND INPUT GAIN
C                          ********************************
      H1=-LA*(MA-(MD/LD)*LA+OMEGA**2+(2*ZETA*OMEGA+MQ)*(MD/LD+MQ))/(MD*
     X(MA-MD*LA/LD+OMEGA**2)+(MQ+2*ZETA*OMEGA)*(LD*MA-MD*LA))
      H2=(LD*MA*(LA-2*ZETA*OMEGA-MQ)+MD*(LA*(2*ZETA*OMEGA-LA)-OMEGA**2
     X-MA))/(VM*LD*(MD*(MA-MD*LA/LD+OMEGA**2)+(MQ+2*ZETA*OMEGA)*(LD*
     XMA-MD*LA)))
      KD=H1/VM+H2-(1/VM)*((MA+LA*MQ)/(LA*MD-MA*LD))
      CD=2/SQRT(MACH)+CN**2/CNA
      ADRAG=((RHO/(2*MASS))*SURF*CD*VM**2)/G
      VC=XVT*SIN(LOA)+YVT*COS(LOA)-XVM*SIN(LOA)-YVM*COS(LOA)
      GO TO 3
C                          ********************************
C     TRAJECTORY PROPAGATION
C                          ********************************
77    DO 78 I=1,2
C                          ********************************
C     STATE EQUATIONS
C                          ********************************
      P(1)=VT*SIN(Y(10))-VM*SIN(Y(2))
      P(2)=G*Y(6)/VM
```

```
      P(3)=-VC*Y(3)**2
      P(4)=Y(5)
      P(5)=MQ*Y(5)+((G*MA)/(VM*LA))*Y( 6)+(MD-LD*MA/LA)*Y( 7)
      P( 6)=1/G*(VM*LA-H1*LAM*VM*LD)*Y(5)-(LA+H2*LAM*VM*LD)*Y( 6)-1/G*
     XLAM*VM*LD*Y( 7)+LAM*VM*LD*KD*Y( 8)
      P(7)=-H1*LAM*Y(5)-G*H2*LAM*Y( 6)-LAM*Y( 7)+G*LAM*KD*Y( 8)
      P( 8)=+PNK*VC/(G*TAU1*TAU2)*(XKR*Y(4)-Y( 9)-Y(12)-(1+XKR)*Y(1)*
     XY(3))-1/TAU2*Y( 8)
      P( 9)=1/TAU1*(XKR*Y(4)-Y( 9)-Y(12)-(1+XKR)*Y(1)*Y(3))
      P(10)=Y(11)/VT
      P(11)=0
      P(12)=-LAMN*Y(12)
      P(13)=(COS(Y(2))*ADRAG+SIN(Y(2))*Y(5))*G
      P(14)=(+SIN(Y(2))*ADRAG-COS(Y(2))*Y(6))*G
C                          ********************************
C     STATE UPDATES (RECTANGULAR INTEGRATION)
C                          ********************************
      Y(1)=Y(1)+P(1)*DX
      Y(2)=Y(2)+P(2)*DX
      Y(3)=Y(3)+P(3)*DX
      Y(4)=Y(4)+P(4)*DX
      Y(5)=Y(5)+P(5)*DX
      Y(6)=Y(6)+P(6)*DX
C                          ********************************
C     ACCELERATION LIMITER
C                          ********************************
      IF(ABS(Y(5)).LT.15) GO TO 51
      IF(Y(6).LT.0)  Y(6)=-15
      IF(Y(6).GT.0)  Y(6)=15
51    Y(7)=Y(7)+P(7)*DX
```

```
Y(8)=Y(8)+P(8)*DX
Y(9)=Y(9)+P(9)*DX
Y(10)=Y(10)+P(10)*DX
Y(11)=Y(11)+P(11)*DX
CALL NOISE(CORR,0.,CORRUP)
Y(12)=P(12)*DX+CORRUP
CALL NOISE(CORR1,0.,CORRUP)
Y(12)=Y(12)+CORRUP
Y(13)=Y(13)+P(13)*DX
Y(14)=Y(14)+P(14)*DX
X=X+DX
XT=K1*X+K2*COS(W*X)-<2
YT=K3*X+K4*SIN(W*X)
XVT=K1-K2*W*SIN(W*X)
YVT=K3+K4*W*COS(W*X)
XAT=-K2*W**2*COS(W*X)
YAT=-K4*W**2*SIN(W*X)
XATLS=COS(THETA)*XAT+SIN(THETA)*YAT
ATLOS=XAT*COS(THETA0)
YATLS=-SIN(THETA)*XAT+COS(THETA)*YAT
PHIAC1=6.2831854-Y(10)
Y(11)=+SIN(PHIAC1)*XATLS-COS(PHIAC1)*YATLS
VT=SQRT(XVT**2+YVT**2)
VM=SQRT(Y(13)**2+Y(14)**2)
XVMLS=Y(13)
YVMLS=Y(14)
XVM=COS(THETA)*XVMLS-SIN(THETA)*YVMLS
YVM=SIN(THETA)*XVMLS+COS(THETA)*YVMLS
XM=XM+DX*XVM
YM=YM+DX*YVM
```

```
      XTM=XT-XM
      YTM=YT-YM
      RANGE=SQRT((XT-XM)**2+(YT-YM)**2)
      LOA1=LOA
      LOA=ATAN(XTM/YTM)
      THETA0=4.71239-LOA
      QP=(RHO/2)*VM**2
      MACH=VM/A
C                          *********************************
C     AERODYNAMIC COEFFICIENTS
C                          *********************************
      CN=(Y(6)*G*MASS*2)/(RHO*SURF*VM**2)
      CMQ=-(4943+MACH*(-2553+MACH*2125))/(2.957+MACH*(-3.078+MACH))
      CMA=-(2.292+MACH*(-2.858+MACH*1.824))/(1.646+MACH*(-2.171+MACH))
      CMD=-(8.234+MACH*(-4.894+MACH*1.191))/(3.220+MACH*(-2.750+MACH))
      CNA=(1.260+MACH*(-.5398+MACH*.2834))/(2.393+MACH*(-2.132+MACH))
      CND=(.3115+MACH*(-.2199+MACH*.0665))/(3.781+MACH*(-3.126+MACH))
      CMA=CMA*57.295780
      CMD=CMD*57.295780
      CNA=CNA*57.295780
      CND=CND*57.295780
      MQ=(QP*SURF*D**2*CMQ)/(2*IYY*VM)
      MA=(QP*SURF*D*CMA)/IYY
      MD=(QP*SURF*D*CMD)/IYY
      LD=(QP*SURF*CND)/(MASS*VM)
      LA=(QP*SURF*CNA)/(MASS*VM)
C                          *********************************
C     FEEDBACK COEFFICIENTS AND INPUT GAIN
C                          *********************************
      H1=-LA*(MA-(MD/LD)*LA+OMEGA**2+(2*ZETA*OMEGA+MQ)*(MD/LD+MQ))/(MD*
```

```
     X(MA-MD*LA/LD+OMEGA**2)+(MQ+2*ZETA*OMEGA)*(LD*MA-MD*LA))
      H2=(LD*MA*(LA-2*ZETA*OMEGA-MQ)+MD*(LA*(2*ZETA*OMEGA-LA)-OMEGA**2
     X-MA))/(VM*LD*(MD*(MA-MD*LA/LD+OMEGA**2)+(MQ+2*ZETA*OMEGA)*(LD*
     XMA-MD*LA)))
      KD=H1/VM+H2-(1/VM)*((MA+LA*MQ)/(LA*MD-MA*LD))
      CD=2/SQRT(MACH)+CN**2/CNA
      CD0=2/SQRT(MACH)
      CDI=CN**2/CNA
      ADRAG=((RHO/(2*MASS))*SURF*CD*VM**2)/G
78    VC=XVT*SIN(LOA)+YVT*COS(LOA)-XVM*SIN(LOA)-YVM*COS(LOA)
C                          *********************************
C     "TRUE STATES"
C                          *********************************
3     TS(1)=THETA0
      TS(2)=XVM
      TS(3)=YVM
      TS(4)=RANGE
      TS(5)=VC
      TS(6)=PNK
      TS(7)=MASS/SURF
      TS(8)=0.
      MODE1=.TRUE.
      RETURN
      END


      SUBROUTINE DXSTAT(N,T,FS,FSD)
C                          *********************************
```

```
C     SUBROUTINE DXSTAT CONTAINS THE FILTER SYSTEM EQUATIONS AND IS
C     CALLED TO PROPAGATE THE STATES FORWARD IN TIME.  THIS EXAMPLE IS
C     FOR THE UP-FILTER. (SIN(THETA)>.707)
C                        *********************************
      COMMON/SITE1A/XVT,YVT,ATLOS
      DIMENSION FS(8),FSD(8)
      IF(FS(6).LT.3) FS(6)=3.0
      IF(FS(6).GT.6) FS(6)=6.0
      VM=SQRT(FS(2)**2+FS(3)**2)
      IF (ABS(SIN(FS(1))).LT..707)GO TO 1
      FSD(1)=(FS(5)*COS(FS(1)))/(FS(4)*SIN(FS(1)))+(XVT-FS(2))/
     X(FS(4)*SIN(FS(1)))
      IF (ABS(SIN(FS(1))).GT..707)GO TO 2
1     FSD(1)=-(FS(5)*SIN(FS(1)))/(FS(4)*COS(FS(1)))+(FS(3)-YVT)/
     X(FS(4)*COS(FS(1)))
2     TDOT=-FSD(1)
      IF(FS(3).LT.5)GO TO 3
      GAMMF=ATAN(FS(2)/FS(3))
      GO TO 4
3     GAMMF=1.5718
4     FSD(2)=-(.0338*VM**1.5*(1/FS(7))+0.0390*TDOT**2*FS(7)*FS(6)**2
     X*FS(5)**2/VM)*SIN(GAMMF)-FS(5)*FS(5)*TDOT*COS(GAMMF)
      FSD(3)=0.
      FSD(4)=FS(5)
      FSD(5)=TDOT**2*FS(4)+(-(.0338*(1/FS(7))*VM**1.5+0.0390*TDOT**2
     X*FS(7)*FS(6)**2*FS(5)**2/VM)*SIN(GAMMF)-FS(6)*FS(5)*TDOT*
     XCOS(GAMMF))*COS(FS(1))+(-(.0338*(1/FS(7))*VM**1.5+0.0390*TDOT
     X**2*FS(7)*FS(6)**2*FS(5)**2/VM)*COS(GAMMF)+FS(6)*FS(5)*TDOT
     X*SIN(GAMMF))*SIN(FS(1))-ATLOS
      FSD(6)=0.
```

```
      FSD(7)=0.
      FSD(8)=0.
      RETURN
      END


      SUBROUTINE DXCOVAR(N,T,FP,FPD)
C                        *********************************
C     SUBROUTINE DXCOVAR PROPAGATES THE FILTER COVARIANCE MATRIX
C                        *********************************
      COMMON/SITE/A1(8,8),A2(8,8),A3(8,8),A4(8,3),A5(3,3),A6(3,3),A7(8)
      COMMON/SITE3/Q(8,8),F(3,8)
      DIMENSION FP(8,8),FPD(8,8)
      CALL MTRAN(8,8,F,A1)
      CALL MMPY(FP,A1,A2,8,8,8)
      CALL MMPY(F,FP,A1,8,8,8)
      CALL MADD(8,8,A1,A2,A3)
      CALL MADD(8,8,A3,Q,FPD)
402   FORMAT(/(5X,8E12.5))
      RETURN
      END


      SUBROUTINE DXGAIN(H,FP,R)
C                        *********************************
C     SUBROUTINE DXGAIN GENERATES THE EXTENDED KALMAN GAINS
```

```
C                          **********************************
      COMMON/SITE/A1(8,8),A2(8,8),A3(8,8),A4(8,3),A5(3,3),A6(3,3),A7(8)
      COMMON/SITE5/FK(8,3)
      DIMENSION H(3,8),FP(8,8),R(3,3)
      CALL MTRAN(3,8,H,FK)
      CALL MMPY(FP,FK,A4,8,8,3)
      CALL MMPY(H,A4,A5,3,8,3)
      CALL MADD(3,3,A5,R,A5)
      CALL MINV(A5,A5,3)
      CALL MMPY(A4,A5,FK,8,3,3)
      RETURN
      END


      SUBROUTINE DXPPLUS(FP,H)
C                          **********************************
C     SUBROUTINE DXPPLUS UPDATES THE COVARIANCE MATRIX
C                          **********************************
      COMMON/SITE/A1(8,8),A2(8,8),A3(8,8),A4(8,3),A5(3,3),A6(3,3),A7(8)
      COMMON/SITE5/FK(8,3)
      DIMENSION FP(8,8),H(3,8)
      CALL MMPY(FK,H,A1,8,3,8)
      CALL MMPY(A1,FP,A2,8,8,8)
      CALL MSUB(8,8,FP,A2,FP)
      RETURN
      END
```

```
      SUBROUTINE DXSPLUS(FS,FSPLUS)
C                       ********************************
C     SUBROUTINE DXSPLUS UPDATES THE SYSTEM STATES
C                       ********************************
      COMMON/SITE/A1(8,8),A2(8,3),A3(8,8),A4(8,3),A5(3,3),A6(3,3),A7(8)
      COMMON/SITE4/MEAS(3)
      COMMON/SITE5/FK(8,3)
      DIMENSION DIFF(3),FS(8),FSPLUS(8)
      REAL MEAS
      DO 1 I=1,3
      DIFF(I)=0
1     CONTINUE
      DIFF(1)=MEAS(1)-FS(1)
      DIFF(2)=MEAS(2)-FS(4)
      DIFF(3)=MEAS(3)-FS(5)
      CALL MMPY(FK,DIFF,A7,8,3,1)
      CALL MADD(8,3,FS,A7,FSPLUS)
      RETURN
      END



      SUBROUTINE FMAT(FS)
C                       ********************************
C     SUBROUTINE FMAT CONTAINS THE ELEMENTS OF THE LINEARIZED F-MATRIX
C     FMAT IS CALLED IMMEDIATELY AFTER EACH MEASUREMENT UPDATE (NOTE
C     THIS IS NOT STANDARD)
C                       ********************************
      COMMON/SITE10/XVT,YVT,ATLOS
```

```
      COMMON/SITE3/Q(8,8),F(8,8)
      DIMENSION FS(8)
      VM=SQRT(FS(2)**2+FS(3)**2)
      IF(FS(3).LT.5)GO TO 6
      GAMMF=ATAN(FS(2)/FS(3))
      GO TO 7
    6 GAMMF=1.5718
    7 IF (ABS(SIN(FS(1))).LT..707)GO TO 1
      BIGX=(FS(5)*COS(FS(1))+XVT-FS(2))/(FS(4)*SIN(FS(1)))
      BIGX=-BIGX
      F(1,3)=F(1,6)=F(1,7)=F(1,8)=0
      F(1,1)=-FS(5)/(FS(4)*SIN(FS(1))**2)-(XVT-FS(2))*COS(FS(1))/
     X(FS(4)*SIN(FS(1))**2)
      F(1,2)=-1/(FS(4)*SIN(FS(1)))
      F(1,4)=-BIGX/FS(4)
      F(1,5)=COS(FS(1))/(FS(4)*SIN(FS(1)))
      GO TO 2
    1 BIGX=-(FS(5)*SIN(FS(1))+FS(3)-YVT)/(FS(4)*COS(FS(1)))
      BIGX=-BIGX
      F(1,2)=F(1,5)=F(1,7)=F(1,8)=0
      F(1,1)=-FS(5)/(FS(4)*COS(FS(1))**2)+(FS(3)-YVT)*SIN(FS(1))/
     X(FS(4)*COS(FS(1))**2)
      F(1,3)=1/(FS(4)*COS(FS(1)))
      F(1,4)=-BIGX/FS(4)
      F(1,5)=-SIN(FS(1))/(FS(4)*COS(FS(1)))
    2 F(2,1)=F(2,2)=F(2,3)=F(2,4)=F(2,8)=0
      F(2,5)=(+0.0780*BIGX**2*FS(7)*FS(5)*FS(5)**2*SIN(GAMMF))
     X/VM+FS(6)*BIGX*COS(GAMMF)
      F(2,6)=(-0.0780*BIGX**2*FS(7)*FS(6)*FS(5)**2*SIN(GAMMF))
     X/VM-FS(5)*BIGX*COS(GAMMF)
```

```
      F(2,7)=.0338*VM**1.5*SIN(GAMMF)/FS(7)**2-(0.0390*BIGX**2
     X*FS(6)**2*FS(5)**2*SIN(GAMMF))/VM
      DO 3 I=1,8
      F(4,I)=0
3     CONTINUE
      F(4,5)=1
      F(5,1)=-SIN(FS(1))*(-(.0338*VM**1.5/FS(7)+0.0390*BIGX**2
     X*FS(7)*FS(5)**2*FS(6)**2/VM)*SIN(GAMMF)-FS(6)*FS(5)*BIGX*
     XCOS(GAMMF))+COS(FS(1))*(-(.0338*VM**1.5/FS(7)+0.0390*BIGX**2
     X*FS(7)*FS(5)**2*FS(6)**2/VM)*COS(GAMMF)+FS(6)*FS(5)*BIGX
     X*SIN(GAMMF))
      F(5,4)=BIGX**2
      F(5,5)=(0.0780*BIGX**2*FS(7)*FS(5)*FS(6)**2*SIN(GAMMF)/VM)
     X*(COS(FS(1))+SIN(FS(1)))+(FS(6)*BIGX*COS(GAMMF))*(COS(FS(1))
     X-SIN(FS(1)))
      F(5,6)=-(0.0780*BIGX**2*FS(7)*FS(6)*FS(5)**2*SIN(GAMMF)/VM)
     X*(COS(FS(1))+SIN(FS(1)))+(FS(5)*BIGX*COS(GAMMF))*(SIN(FS(1))
     X-COS(FS(1)))
      F(5,7)=(.0338*VM**1.5*SIN(GAMMF)/FS(7)**2-0.0390*BIGX**2*FS(5)**2*
     XFS(6)**2/VM)*SIN(GAMMF)*(COS(FS(1))+SIN(FS(1)))
      RETURN
      END


      SUBROUTINE NOISE(RMSCORR,XMEAN,CORRUP)
C                     ********************************
C     SUBROUTINE NOISE GENERATES THE TRUTH MODEL NOISES
C                     ********************************
```

```
      GAUSS=0.
      DO 10 I=1,12
      GAUSS=GAUSS+RANF(DUM)
10    CONTINUE
      GAUSS=GAUSS-6.+XMEAN
      CORRUP=GAUSS*RMSCORR
      RETURN
      END


       SUBROUTINE MMPY(A,B,C,M,K,N)
      DIMENSION A(M,K),B(K,N),C(M,N)
       DO 10 J=1,N
      DO 10  I=1,M
      C(I,J)= 0.0
      DO 10 L=1,K
10    C(I,J)=C(I,J)+A(I,L) *B(L,J)
      RETURN
      END


      SUBROUTINE MTRAN (K,N,G,GT)
      DIMENSION G(K,N),GT(N,K)
      DO 1 I=1,K
      DO 2 J=1,N
2     GT(J,I)=G(I,J)
```

```
    1   CONTINUE
        RETURN
        END


         SUBROUTINE  MSUB(K,N,A,B,C)
         DIMENSION A(K,N),B(K,N), C(K,N)
        DO 4 I=1,K
        DO 4 J=1,N
        C(I,J)=0.0
    4   C(I,J)= A(I,J)-B(I,J)
        RETURN
        END


        SUBROUTINE INTEE(NSET,N,T,X,DXSUB,DTIN,DTOUT,DTMIN,TOLR)
C*********  REVISON 6D-50N  **********
        DIMENSION X(N)
        COMMON /INCTRL/ INTRP, GOOD
        COMMON/SITE3/D(8,8),F(8,8)
        LOGICAL GOOD
        LOGICAL  ROERR, ROOVR
        LOGICAL     PRNT, PRNTS
        DIMENSION  NRSTP(5)
        EXTERNAL DXSUB
        LOGICAL  EVEN
```

```
      LOGICAL ER1,ER2
      REAL  LOTOL
      LOGICAL  NONRE
      DIMENSION D1(64),D2(64),D3(64),D4(64),D5(64),D6(64),D7(64),D8(64)
      DIMENSION X1(64),X2(64),X3(64),XP(64),X0(64)
           DATA   XP,TP,ERRP/66*1.E+321/
        DATA NMAX/64/
           DATA   NSETP,NHIACS/-13, 0 /
      DATA  NONRE/.FALSE./
      DATA  ICMAX, ICMX2/ 3, 6/
      DATA  RLO,RHI/.015625,100./
      DATA  CRK,CRC/.166666666666667,.416666666666667E-1/
      DATA   PRNT,PRNTS /2(.FALSE.)/
      M=IABS(N)
      IF (M.GE.1 .AND. M.LE.NMAX) GO TO 20
      PRINT 10, N
10    FORMAT(33H SUBROUTINE INTEG HAS ARGUMENT N=,I10,13H,OUT OF RANGE)
      CALL SYSTEM(200,1L )
20    T0=T
      T1= T0
      INTRP = 0
      IF (NSET.LT.0)  PRNTS=.TRUE.
      NSETA=IABS(NSET)
      IF (N.LT.0)  PRNT = .TRUE.
       TOL= ABS(TOLR)
      ETOL= TOL*14.
      LOTOL= ETOL*RLO
      HITOL= ETOL*RHI
       DSTOL= TOL
           TGO= SIGN(DTOUT,DTIN)
```

```
      DTF= ABS(TGO)
      DMIN=ABS(DTMIN)
      ER1=.TRUE.
      ER2=.TRUE.
      TOUT=T0+SIGN(DTF,DTIN)
      DO 22 I=1,5
22    NRSTP(I)= 0
      ISW= 0
      IF (NONRE) GO TO 50
      IF(NSETA.NE.NSETP)  GO TO 50
      IF (H.NE.DTIN) GO TO 50
      IF (T.NE.TP) GO TO 50
      DO 25 I=1,M
      IF (X(I).NE.XP(I)) GO TO 50
25    CONTINUE
      GO TO 346
50    H= DTIN
      NONRE= .FALSE.
      DO 52  I=1,M
      X0(I)= X(I)
      X1(I)= X0(I)
52    CONTINUE
50    CONTINUE
      CALL STEPS (DTF,H ,HITOL,EVEN,NSTEP,RSTEP)
      IF (NSTEP.LT.4) NSTEP=4
      H= ADJH(SIGN(DTF/NSTEP,H),HD,HH)
      ASSIGN 210 TO KS
      ROOVR= .FALSE.
      DO 90 I=1,M
90    X2(I)= X1(I)
```

```
100   CALL DXSUB(M,T1,X2,D8)
      IF (INTRP .EQ. 0)  GO TO  103
      IF (INTRP .LT. -10)  RETURN
      IF (INTRP .LT. 0 .OR. GOOD) GO TO 490
      IF (INTRP .LT. 10) GO TO 472
103   CONTINUE
      GOOD= .F.
105   T2= T1 + 1
      T3= T2 + 4
110   DO 120 I=1,M
120   X2(I)= X1(I) + H*D8(I)
      CALL DXSUB(M,T2,X2,D1)
      IF (INTRP .EQ. 0)  GO TO  123
      IF (INTRP .LT. -10)  RETURN
      IF (INTRP .LT. 10) GO TO 472
      IF (INTRP .LT. 0)  GO TO 480
123   CONTINUE
      DO 130 I=1,M
130   X2(I)= X1(I) + H*D1(I)
      CALL DXSUB(M,T2,X2,D2)
      IF (INTRP .EQ. 0)  GO TO  133
      IF (INTRP .LT. -10)  RETURN
      IF (INTRP .LT. 0)  GO TO 480
      IF (INTRP .LT. 10) GO TO 472
133   CONTINUE
      DO 140 I=1,M
140   X2(I)= X1(I) + H0*D2(I)
      CALL DXSUB(M,T3,X2,D3)
      IF (INTRP .EQ. 0)  GO TO  143
      IF (INTRP .LT. -10)  RETURN
```

```
      IF (INTRP .LT. 0)  GO TO 475
      IF (INTRP .LT. 10) GO TO 472
143   CONTINUE
      COEF=HO*CRK
      DO 150 I=1,M
150   X2(I)= X1(I) + COEF*(D0(I)+D1(I)+D1(I)+D2(I)+D2(I)+D3(I))
      NRSTP(5)= NRSTP(5)+1
      GO TO KS,(210,220,230,240)
210   HO=H
      H=HH
      IF (PRNT)  PRINT 902
902   FORMAT(21H RUNGE-KUTTA POINT A  )
      DO 212 I=1,M
      D4(I)= D0(I)
212   X3(I)= X2(I)
      ERRP= 1.E+321
      ROERR= .FALSE.
      ASSIGN 220 TO KS
      GO TO 105
220   T1= T3
      IF (PRNT)  PRINT 903
903   FORMAT(21H RUNGE-KUTTA POINT B  )
      DO 222 I=1,M
222   X1(I)= X2(I)
      ASSIGN 230 TO KS
      GO TO 100
230   ERR= 0.
      IF (PRNT)  PRINT 904
904   FORMAT(21H RUNGE-KUTTA POINT C  )
      IF (ROOVR) GO TO 233
```

```
      DO 232 I=1,M
232   ERR= AMAX1(ERR,ABS(X3(I)-X2(I))/AMAX1(ABS(X2(I)),1.E-8))
      IF (PRNT)  PRINT 901,  ERR,ETOL, ERRP, HD
901   FORMAT(17H ERR,ETOL,ERRP,H\  ,4G20.11)
      IF (ERR.LT.ETOL) GO TO 2325
      IF (ERR.LT.ERRP) GO TO 2345
      IF (ROERR ) GO TO 238
      ROERR= .TRUE.
      ROERR= .FALSE.
      ERRPP=ERRP
      ERRP = ERR
      GO TO 235
2325  ROERR= .FALSE.
233   T1=T3
      DO 234 I=1,M
      X1(I)= X2(I)
234   D5(I)= D8(I)
      ASSIGN 240 TO KS
      GOOD= .T.
      GO TO 100
2345  ROERR= .FALSE.
235   IF (ABS(H).GE.DMIN) GO TO 235
      IF (ER2.AND.(PRNT.OR.PRNTS)) PRINT 802,  DMIN, ERR, TOLR, T0, HD,
     1  ROERR
      ER2= .FALSE.
      IF (.NOT.ROERR ) GO TO 233
      HH=H
      H=HD
      DO 2355  I=1,M
2355  X1(I)= X3(I)
```

```
      T1= T3
      CALL  DXSUB(M,T1,X1,D8)
      IF (INTRP .EQ. 0)  GO TO 2353
      IF (INTRP .LT. -10)  RETURN
      GO TO 490
2353  CONTINUE
      ASSIGN 230 TO KS
      GO TO 239
236   HD=H
      H=.5*H
      T1=T0
      NSTEP= NSTEP+NSTEP
      ERRP= ERR
      DO 237 I=1,M
      X3(I)= X1(I)
      D8(I)= D4(I)
237   X1(I)= X0(I)
      ASSIGN 220 TO KS
      GO TO 105
238   IF(PRNT.GT.PRNTS)  PRINT 901,  ERR, TOL, T1, HD, ERRPP, ERRP
      HH=HD
      H= HH + HH
      DO 2385  I=1,M
      X1(I)= X0(I)
2385  D8(I)= D4(I)
      T1= T0
      ASSIGN 210 TO KS
239   HD= H + H
      ROOVR= .TRUE.
      CALL STEPS (DTF,M ,MITOL,EVEN,NSTEP,RSTEP)
```

```
      IF (.NOT.EVEN.AND.(PRNT.OR.PRNTS)) PRINT 805
      GO TO 105
240   CONTINUE
      IF (PRNT)  PRINT 905
905   FORMAT(21H RUNGE-KUTTA POINT D  )
      GOOD= .T.
      CALL DXSUB(M,T3,X2,D7)
      IF (INTRP .EQ. 0)  GO TO  243
      IF (INTRP .LT. -10)  RETURN
      GO TO 475
243   CONTINUE
      GOOD= .F.
      DO 242 I=1,M
      X1(I)= X2(I)
242   D6(I)= D8(I)
      H= ADJH(HD,HD,HH)
290   T1=T3
      NHIACS=0
      NSTEP= NSTEP-3
300   CONTINUE
      T1=T1+H
      COEF= CPC*H
      DO 302 I=1,M
302   X2(I)= X1(I) + COEF*(55.*D7(I)-59.*D6(I)+37.*D5(I)-9.*D4(I))
      NRSTP(1)= NRSTP(1)+1
      IF (PRNT)  PRINT 912
912   FORMAT(14H PREDICT STEP  )
      ICOR= 0
310   CALL DXSUB(M,T1,X2,D8)
      IF (INTRP .EQ. 0)  GO TO  313
```

```
      IF (INTRP .LT. -10)  RETURN
      IF (INTRP .LT. 0)  GO TO 490
      IF (INTRP .LT. 10) GO TO 485
313   CONTINUE
      DO 320 I=1,M
320   X3(I)= X1(I) + COEF*(9.*D8(I)+19.*D7(I)-5.*D6(I)+D5(I))
      NRSTP(3)= NRSTP(3)+1
      IF (PRNT)  PRINT 913
913   FORMAT(14H CORRECT STEP  )
      ERR= 0.
      DO 325 I=1,M
325   ERR= AMAX1(ERR,ABS(X2(I)-X3(I))/AMAX1(ABS(X3(I)),1.E-8))
      IF (PRNT)  PRINT 911,  ERR,ETOL,LOTOL, M
911   FORMAT(18H ERR,ETOL,LOTOL,M\  ,4G20.11)
      IF(ERR.GT.ETOL) GO TO 350
      ISW= 0
      IF(ERR.GT.LOTOL) GO TO 330
      NHIACS= NHIACS+1
      GO TO 340
330   NHIACS= 0
340   DO 345 I=1,M
      X1(I)=X3(I)
      D1(I)=D2(I)
      D2(I)=D3(I)
      D3(I)=D4(I)
      D4(I)=D5(I)
      D5(I)=D6(I)
345   D6(I)=D7(I)
      GOOD= .T.
      CALL DXSUB(M,T1,X1,D7)
```

```
      IF (INTRP .EQ. 0)  GO TO  343
      IF (INTRP .LT. -10)  RETURN
      GO TO 490
343   CONTINUE
      GOOD= .F.
      TGO=  TOUT-T1
      DTF= ABS(TGO)
      IF ( DTF.LE.TOL)  GO TO 500
      IF ( TGO*H.LT.0.) GO TO 348
346   IF (NHIACS.LT.2)  GO TO 347
      CALL STEPS (DTF,HD,DSTOL,EVEN,NSTEP,RSTEP)
      IF (PRNT)  PRINT 909,  RSTEP,NSTEP,TOL,TGO ,H
909   FORMAT(24H RSTEP,NSTEP,TOL,TGO,H\  ,020  ,I10,3G20.11)
      IF (.NOT.EVEN)  GO TO 347
      H= ADJH(.5*TGO/NSTEP,HD,HH)
       GO TO 400
347   CONTINUE
      CALL STEPS (DTF,H ,HITOL,EVEN,NSTEP,RSTEP)
      IF (PRNT)  PRINT 909,  RSTEP,NSTEP,TOL,TGO, H
      IF (.NOT.EVEN)  GO TO 348
      H= ADJH(TGO/NSTEP,HD,HH)
       GO TO 300
348   IF (PRNT.OR.PRNTS )  PRINT 804, T1, H, TOUT
      H = TGO*.25
      T0= T1
      DO 349  I=1,4
349   X0(I)= X1(I)
      GO TO 60
350   ICOR= ICOR+1
      IF (PRNT)  PRINT 905,  ICOR
```

```
906   FORMAT(17H SINGLE STEP ICOR ,I5 )
      IF (ICOR.GT.1)  NHIACS=0
      IF (ICOR.LT.ICMAX) GO TO 360
      IF((ICOR.GE.ICMAX).OR.(ERR.GT.HITOL)) GO TO 370
350   DO 362 I=1,M
352   X2(I)= X3(I)
      GO TO 310
370   IF (ABS(HH).GE.DMIN) GO TO 380
      ISW= ISW+1
      IF(ISW.LE.ICMAX)  GO TO 350
      IF (ER1.AND.(PRNT.OR.PRNTS))PRINT 802, DMIN,ERR, TOLR, T1, H
      ER1= .FALSE.
      GO TO 330
380   T1=T1-H
381   H= ADJH(HH,HO,HH)
      IF (NSTEP.LE.1)  GO TO 385
      TO= T1
      DO 382 I=1,M
      XO(I)= X1(I)
382   D8(I)=D7(I)
      ASSIGN 210 TO KS
      ROOVR= .FALSE.
      GO TO 105
385   NSTEP= 4
      GO TO 381
400   COEF= CPC*HO
      ICOR= 0
      T1= T1+HO
      NHIACS= 0
      DO 402 I=1,M
```

```
402   X2(I)= X1(I) + COEF*(55.*D7(I)-59.*D5(I)+37.*D3(I)-9.*D1(I))
      NRSTP(2)= NRSTP(2)+1
      IF (PRNT)  PRINT 915
915   FORMAT(24H PREDICT DOUBLE STEP      )
410   CALL DXSUB(M,T1,X2,D8)
      IF (INTRP .EQ. 0)  GO TO  413
      IF (INTRP .LT. -10)  RETURN
      IF (INTRP .LT. 0)  GO TO 490
      IF (INTRP .LT. 10) GO TO 488
413   CONTINUE
      DO 412 I=1,M
412   X3(I)= X1(I) + COEF*(9.*D8(I)+19.*D7(I)-5.*D5(I)+D3(I))
      NRSTP(4)= NRSTP(4)+1
      IF (PRNT)  PRINT 914
914   FORMAT(24H CORRECT DOUBLE STEP      )
      ERR= 0.
      DO 420 I=1,M
420   ERR= AMAX1(ERR,ABS(X2(I)-X3(I))/AMAX1(ABS(X3(I)),1.E-8))
      IF (PRNT)  PRINT 911,  ERR,ETOL,LOTOL,HD
      IF (ERR.GT.HITOL)    GO TO 470
      IF (ERR.GT.ETOL) GO TO 450
      DO 425 I=1,M
      X1(I)=X3(I)
      D4(I)=D3(I)
      D3(I)=D1(I)
425   D6(I)=D7(I)
      GOOD= .T.
      CALL DXSUB(M,T1,X1,D7)
      IF (INTRP .EQ. 0)  GO TO  433
      IF (INTRP .LT. -10)  RETURN
```

```
      GO TO 490
433   CONTINUE
      GOOD= .F.
      TGO= TOUT-T1
      DTF= ABS(TGO)
      NSTEP= NSTEP-1
        IF(NSTEP.EQ.0)  GO TO 435
         H= ADJH(TGO/NSTEP,HO,HH)
         GO TO 440
435      H=ADJH(HO,HO,HH)
440      IF (DTF.LE.TOL) GO TO 500
      IF (TGO*H.LT.0) GO TO 348
      GO TO 300
450   ICOR= ICOR+1
      IF (PRNT)  PRINT 907,  ICOR
907   FORMAT(17H DOUBLE STEP ICOR ,I5 )
      IF (ICOR.GE.ICMAX)   GO TO 470
450   DO 462 I=1,M
462   X2(I)= X3(I)
      GO TO 410
470   T1=T1-HO
      NSTEP= NSTEP+NSTEP
      GO TO 300
472   T=T0
      DO 473 I=1,M
473   X(I)= X0(I)
      GO TO 498
475   T=T3
      GO TO 482
480   T=T2
```

```
482    DO 483 I=1,M
483    X(I)= X2(I)
       GO TO 498
485    T= T1-H
       GO TO 492
488    T= T1-HD
       GO TO 492
490    T= T1
492    X(I)=X1(I)
498    NONRE= .TRUE.
       GO TO 505
500    DO 502 I=1,M
       XP(I)=X1(I)
502    X(I)=X1(I)
       NSETP= NSETA
       TP=T1
       T=T1
505    CONTINUE
       IF(SIGN(1.,H).EQ.SIGN(1.,DTIN) ) DTIN=H
       IF (PRNTS.OR.PRNT)  PRINT 950, (NRSTP(I),I=1,5)
950    FORMAT(20H NUMBER OF STEPS\ P= ,I4,5H, DP=,I4,4H, C=,I4,5H, DC=,I4
     1,5H, RK=,I4)
       PRNT = .FALSE.
       PRNTS= .FALSE.
       RETURN
801    FORMAT(51H *****ROUND-OFF ERROR BUILD-UP PREVENTED THE ERROR( ,E15
     1.6,35H) FROM BEING REDUCED TO TOLERANCE ( ,E15.6,1H),/ 7H  AT T= ,
     2 E15.6,17H, WITH STEP SIZE\,E15.6,28H, WHERE THE PREV. ERRS WERE
     3,E15.6, 6H THEN ,E15.6)
802    FORMAT(11H *****DMIN= ,E15.6,22H  PREVENTED THE ERROR= ,E15.6,32H
```

```
     1 FROM SATISFYING THE TOLERANCE= ,E15.6,/ 7H  AT T= ,E15.5,26H  WHE
     2RE THE STEP SIZE WAS ,E15.6,15H AND ROERR WAS ,L6)
804   FORMAT(12H ***** AT T= ,E20.11,16H, THE STEP SIZE= ,E20.11,30H COU
     1LD NOT BE USED TO GO TO T= ,E20.11)
805   FORMAT(52H ***** ERROR IN NSTEP AFTER ROUNDOFF ERROR DETECTION )
      END


      SUBROUTINE STEPS(DTF,H,TOL,EVEN,NSTEP,RSTEP)
       LOGICAL  EVEN
      EVEN= .FALSE.
      R=DTF/ABS(H)
      N= R
      D= R-N
      IF (D.LE.TOL)  EVEN= .TRUE.
      IF (.NOT.EVEN) N=N+1
      C= 1.-D
      IF (C.LE.TOL)  EVEN= .TRUE.
      NSTEP= N
      RSTEP= R
      END


      FUNCTION ADJH (H,HD,HH)
         TH= H
         HH= .5*TH
```

```
      HD= TH+T4
     ADJH= TH
    END


    SUBROUTINE MADD(K,N,A,B,C)
     DIMENSION A(K,N), B(K,N),C(K,N)
    DO 3 I=1,K
     DO 3 J=1,N
     C(I,J)=0.0
3   C(I,J)= A(I,J)+B(I,J)
    RETURN
    END
```

Vita

Robert N. Lutter was born 19 October 1946 in Kenosha, Wisconsin. In 1964, he was graduated from Mary D. Bradford High School, Kenosha, Wisconsin. He then entered the Air Force as an Air Force Academy cadet in June of 1964. After graduating from the Academy in 1968, he attended Undergraduate Pilot Training at Reese AFB, Texas, graduating in August 1969.

His flying experience includes a one year tour in South East Asia and a four year tour as an Undergraduate Pilot Training Instructor. In South East Asia, he was stationed at Chu Lai, Viet Nam as a forward air controller flying the OV-10 aircraft. During his instructor tour at Vance Air Force Base, Oklahoma, he flew the T-38 aircraft, served as a Student Squadron Class Commander, and as Headquarter's Squadron Commander, 71st Air Base Group.

In June 1975, he entered the Graduate Electrical Engineering program at the Air Force Institute of Technology. Upon graduation in December of 1976, he will be assigned to the Air Force Avionics Laboratory, WPAFB, Ohio.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: GE/EE/76-31	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE *(and Subtitle)*: Application of an Extended Kalman Filter to an Advanced Fire Control System	5. TYPE OF REPORT & PERIOD COVERED: MS Thesis
6. PERFORMING ORG. REPORT NUMBER	
7. AUTHOR(s): Robert N. Lutter, Capt USAF	8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Air Force Institute of Technology (AFIT-EN), Wright-Patterson AFB, Ohio 45433	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS	12. REPORT DATE: December 1976
	13. NUMBER OF PAGES: 232
14. MONITORING AGENCY NAME & ADDRESS *(if different from Controlling Office)*	15. SECURITY CLASS. *(of this report)*: Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT *(of this Report)*

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*

18. SUPPLEMENTARY NOTES

Approved for public release; IAW AFR 190-17

Jerral F. Guess, Captain, USAF
Director of Information

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*

Extended Kalman Filter
Air-to-Air Missile
Proportional Guidance
Parameter Estimation

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

An extended Kalman Filter is developed to aid the tracking of an air-to-air missile from a maneuvering target aircraft. The filter exploits knowledge of the dominant dynamic effects acting on a missile that is non-thrusting and utilizing a proportional navigation guidance scheme, i.e. accelerations due to aerodynamic forces. It is designed to provide both dynamic tracking estimates in a local inertial frame and estimates of the proportional navigation constant and another pertinent parameter.

A feasibility analysis of the filter is conducted. Its performance is

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

20. compared to a more conventional filter that utilizes a first order Gauss-Markov random process acceleration model. In addition, an evaluation is made of the filter's capability to recover from large initial errors in state estimates.

The study establishes the feasibility of the modelling approach. The estimates provided by the designed filter are, in general, less sensitive to system measurement noises. The filter performance is trajectory dependent, however, and the requirement for a higher order missile model within the filter system model is established (a zero-order model was used to develop as simple a filter as would provide adequate performance).

The results of the study strongly suggest that the navigation constant can be estimated by the filter. The recovery analysis provides additional insights into the filter's ability to estimate this parameter. It gives a general indication of the effects that varying the initial variance and noise strength (on the navigation constant channel) have on the tuning and recovery characteristics of the navigation constant estimate. A graphic filter analysis is included that portrays the estimation accuracy and recovery characteristics of the filter.