NASA Contractor Report 178255

DEVELOPMENT OF A TAKEOFF PERFORMANCE
MONITORING SYSTEM

(NASA-CR-178255) DEVELOPMENT OF A TAKEOFF PERFORMANCE MONITORING SYSTEM Ph.D. Thesis. Contractor Report, Jan. 1984 - Jun. 1985 (Kansas Univ. Center for Research, Inc.) 208 p CSCL 01D G3/06

N87-20264

Unclas 45171

Raghavachari Srivatsan
David R. Downing

THE UNIVERSITY OF KANSAS CENTER FOR RESEARCH, INC.
Lawrence, Kansas

Cooperative Agreement NCC1-79
March 1987

National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23665

PREFACE

The development of the takeoff performance monitoring system described here was carried out under Cooperative Agreement NCC 1-79, between the University of Kansas and NASA Langley Research center. Dr. David R. Downing, of the Department of Aerospace Engineering at the University of Kansas was the Principal Investigator on the project and Raghavachari Srivatsan, served as the Research Assistant. The work was done at NASA Langley Research Center in Hampton, VA.

The work was started under a one year agreement covering the period 15-June-1983 through 14-June-1985, and continued under an extension for the period 15-June-1984 through 14-June-1985. The project was accomplished in two phases. The first phase consisted of conducting a detailed survey of existing flight management systems and a comparison to those that were available on the Transport Systems Research Vehicle (TSRV) Boeing 737 at NASA Langley Research Center. This survey resulted in the identification of three areas needing attention. In phase II one of the identified areas was chosen for detailed development. This report documents the efforts of phase II.

The authors wish to thank Dr. Paul E. Fortin for his help with the administrative aspects at the University of Kansas, and his help with proposals and renewals, and Mr. Wayne H. Bryant, contract monitor at NASA Langley, for his support, availability for discussions, and in the preparation of this document.

Thanks are also due to Mr. William E. Howell and Mr. Ray V. Hood of NASA Langley for their periodic review of work and suggestions. The author also wishes to acknowledge the technical assistance provided by Mr. Richard M.

Hueschen, Mr. Wiiliam T. Bundick of NASA Langley, and the programming assistance provided by Mrs. Jessie C. Yeager, and Mr. Melvin L. Bailey of PRC, and Mr. Carey S. Buttril of TAI. To Ms. Danette Barrett, Mrs. Jessie Yeager, Mrs. Beth Lee, and Ms. Louise Richardson thanks are expressed for their assistance in the preparation of this document.

The authors also wish to thank Mrs. Nancy Hanson, of the University of Kansas - Flight Research Lab, for converting this document from a thesis to a report format and Mr. David B. Middleton, of NASA Langley Research Center, for his considerable help in the genesis of this contractor report.

TABLE OF CONTENTS

LIST OF SYMBOLS

SYMBOL	DEFINITION	DIMENSION
A	Speed of Sound	feet/second
A_0, A_1, A_2, A_3	Acceleration Cubic Polynomial Coefficients	
a	Along Track Acceleration	feet/second^2
b	Wing Span	feet
CAS	Calibrated Airspeed	feet/second
		knots
C_D	Airplane Drag Coefficient	-
C_L	Airplane Lift Coefficient	-
C_m	Airplane Pitching Moment Coefficient	-
$\bar{c}$	Mean Aerodynamic Chord	feet
cg	Center of Gravity Location	-
D	Aerodynamic Drag	lbs
D_{RWY}	Runway Length Used	feet
D_Z	Engine Moment Arm Along Z-axis	feet
EPR	Engine Pressure Ratio	-
{E}, [E]	Euler Transformation Matrix	
F	Force	lbs
F_{nz}	Z Directional Load Factor	

LIST OF SYMBOLS (continued)

SYMBOL	DEFINITION	DIMENSION
Gear	=0 for landing gear retracted	
	=1 for landing gear extended	-
g	Acceleration due to gravity	feet/second^2
$\bar{g}$	Gravitational Acceleration Vector	feet/second^2
H	Height of the airplane (mean sea level)	feet
H_{CG}	Height of the airplane center of gravity	
	above the runway	feet
H_{cg}	Height of the airplane center of gravity	
	above the runway	feet
H_{RWY}	Height of the runway (mean sea level)	feet
H_{RUNWAY}	Height of the runway (mean sea level)	feet
H_r	Height of the runway (mean sea level)	feet
I_{YY_B}	Moment of Inertia about the body Y - axis	slug-feet^2
i_H	Stabilizer incidence	degrees
L	Aerodynamic Lift	lbs
L_G	Landing Gear related Force/Moment	lbs or
		ft-lbs
M	Airplane Pitching Moment	ft-lbs
M	Mach Number	-

LIST OF SYMBOLS (continued)

SYMBOL	DEFINITION	DIMENSION
m	Airplane Mass	slugs
n_Z	Normal Load Factor	g's
p	Roll Rate	deg/second
		rad/second
q	Pitch Rate	deg/second
		rad/second
$\bar{q}$	Dynamic Pressure	$lbs/feet^2$
R	Acceleration Due to Body Rates	$feet/second^2$
R	Radious of the Earth	feet
Rad	Radious of the Earth	feet
r	Yaw Rate	deg/second
		rad/second
S	Wing Area	$feet^2$
T	Engine Thrust related Force/Moment	lbs or
		ft-lbs
T	Iteration Time Step/ Sampling Interval	second
TAS	True Airspeed	feet/second
		knots
THRUST REVERSE	Thrust Reverse Yes/No	
Th	Engine Thrust	lbs

LIST OF SYMBOLS (continued)

SYMBOL	DEFINITION	DIMENSION
T_{total}	Total Temperature	° Kelvin
T_0, T_1, T_2, T_3	Thrust Cubic Polynomial Coefficients	
t_{ramp}	Ramp Time to reach Maximum value	seconds
u	X-axis Velocity	feet/second
u_W	Runway Wind Velocity	feet/second
		knots
V	Velocity	feet/second
V_1	Decision Speed	knots
V_2	Second Segment Climb Speed	knots
V_R	Rotation Speed	knots
$\bar{V}_W$	Wind Vector	feet/second
v	Y-axis Velocity	feet/second
v_G	Ground Track Speed	feet/second
v_T	True Airspeed	feet/second
$v_{T_{rotate}}$	True Airspeed for Rotation	feet/second
W	Airplane Weight	lbs
w	Z-axis Velocity	feet/second
X_B	Aerodynamic Forces along Body X-axis	lbs

LIST OF SYMBOLS (continued)

SYMBOL	DEFINITION	DIMENSION
$X_{B_{TOT}}$	Total Forces Along Body X-axis	lbs
x_T	Engine X-moment arm	feet
Y_B	Aerodynamic Forces along Body Y-axis	lbs
$Y_{B_{TOT}}$	Total Forces Along Body Y-axis	lbs
Z_B	Aerodynamic Forces along Body Z-axis	lbs
$Z_{B_{TOT}}$	Total Forces Along Body Z-axis	lbs
z_T	Engine Z-moment arm	feet
α	Airplane Angle of Attack	degrees
β	Airplane Side Slip Angle	degrees
Γ	Discrete Control Effectiveness Matrix	
γ	Airplane Flight Path Angle	degrees
ΔC_D	Incremental Drag Coefficient	-
ΔC_L	Incremental Lift Coefficient	-
Δs	Computed Distance per Step	feet
ΔT	Iteration Time Step/ Sample Interval	second
Δt	Computed Time Interval	second
$\Delta \mu$	Difference in Friction Coefficient	-
δ_a	Aileron Deflection	degrees
δ_E	Elevator Deflection	degrees

LIST OF SYMBOLS (continued)

SYMBOL	DEFINITION	DIMENSION
δ_e	Elevator Deflection	degrees
δ_F	Flap Deflection	degrees
δ_{FLAPS}	Flap Deflection	degrees
δ_{FSP}	Flight Spoiler Deflection	degrees
δ_{GSP}	Ground Spoiler Deflection	degrees
δ_{GEAR}	Gear Up/Down	-
δ_R	Rudder Deflection	degrees
δ_{SP_L}	Left Spoiler Deflection	degrees
δ_{SP_R}	Right Spoiler Deflection	degrees
δ_{STAB}	Stabilizer Deflection	degrees
δ_{th}	Throttle Position	degrees
θ	Pitch Related	degrees
θ_B	Body Pitch Attitude	degrees
θ_{RWY}	Runway Inclination	degrees
λ	Latitude	degrees
μ	Rolling Friction Coefficient	-
ξ	Discrete Transform Multiplier	
ρ	Air Density	slugs/feet^3

LIST OF SYMBOLS (continued)

SYMBOL	DEFINITION	DIMENSION
τ	Longitude	degrees
Φ	Discrete State Matrix	
ϕ	Bank Angle	degrees
ψ	Yaw Angle	degrees
ψ_{RUNWAY}	Runway Heading	degrees

Superscripts

$\dot{}$	Time derivative
$\hat{}$	Estimated quantity

Subscripts

B	Body Axes
brake	Due to braking
D	Down direction
E	East direction
F1	From the first filter network
FSP	Due to Flight Spoilers
fr	Friction
G	Ground
GSP	Due to Ground Spoilers
left	Left Side
L_B	Body Axis Rolling Moment

LIST OF SYMBOLS (continued)

SYMBOL	DEFINITION	DIMENSION
	Subscripts (continued)	
l_B	Body Axis Rolling Moment	
M	Measured	
M_B	Body Axis Pitching Moment	
max	Maximum value	
m_B	Body Axis Pitching Moment	
N	North direction	
N_B	Body Axis Yawing Moment	
n_B	Body Axis Yawing Moment	
O	Old value	
r	Relative	
right	Right side	
S	Stability Axes	
step	step size	
T	True	
T	Engine Thrust line	
total	Total Force/ Moment	
X_B	Along Body X-axis	
X_S	Along Stability X-axis	
Z_B	Along Body Z-axis	

LIST OF SYMBOLS (continued)

SYMBOL	DEFINITION	DIMENSION
	Subscripts (continued)	
Z_S	Along Stability Z-axis	
μ	For a rolling friction coefficient of μ	

LIST OF TABLES

LIST OF TABLES (continued)

List of Figures

List of Figures (continued)

List of Figures (continued)

List of Figures (continued)

List of Figures (continued)

List of Figures (continued)

List of Figures (continued)

List of Figures (continued)

SUMMARY

An algorithm was developed to monitor the performance of the airplane during the takeoff phase to improve safety in that flight phase. The algorithm is made up of two segments: a pretakeoff segment and a real time segment.

One-time inputs of ambient temperature, pressure, runway wind, airplane gross weight, selected flap and stabilizer setting are utilized by the pretakeoff segment in generating a set of standard acceleration performance data in an offline condition.

The real-time segment, in addition to the above one-time inputs, requires the runway length available for rotation, the runway length available for stopping and an estimated runway rolling friction coefficient. The real-time segment also utilizes instantaneous measurements of throttle position, engine pressure ratios, calibrated airspeed, along track acceleration, and ground speed. The input values and the measured parameters are used in computing engine parameters and airplane acceleration, keeping track of the runway used, runway remaining, and in predicting the runway needed to achieve rotation speed, and the runway needed to stop the airplane. A comparison of measured and predicted values is utilized in detecting performance deficiencies. These comparisons and the runway length computations lead to Go/Abort signals. With the algorithm operating in a command generation mode these signals are transformed into commands to the airplane systems. An important feature of the algorithm is the one-time estimation of the runway rolling friction coefficient early into the takeoff run. The instantaneous measurements and computations of the real-time segment are carried out ten times a second.

The algorithm has been evaluated using a six degree-of-freedom nonlinear airplane simulation as the plant for several design point test cases and two types of engine malfunctions. The algorithm does not cause any false alarms and the error in the predicted runway required to achieve rotate speed is within 5 percent of the runway actually used. The algorithm is not capable of detecting engine failures that do not affect the Engine Pressure Ratio, but even for this type failure the runway predictions are within 5 percent of that used. Engine malfunctions that cause the Engine Pressure Ratio to be affected are identified as engine failures early into the takeoff run.

Sensitivity analysis of the algorithm to errors in one-time inputs indicated a high sensitivity to errors in runway wind inputs. An onboard wind estimator overcomes this sensitivity. The algorithm is also highly sensitive to errors in ambient temperature inputs. The algorithm is capable of adjusting for errors in other one-time inputs as long as these errors do not cause the flight manual recommended rotation speed to be different. Failures of the accelerometers resulting in bias and scale factor errors in their outputs are sensed and compensated for by the algorithm. Bias and scale factor errors in the Engine Pressure Ratio sensors of more than 15 percent of nominal values causes abort signals to be generated. Failures of the ground speed sensor resulting in an unchanged output also results in the generation of abort signals.

CHAPTER 1

INTRODUCTION

The percentage of initiated takeoffs that have resulted in accidents is very small. Looking at this same data in a slightly different manner, however, it is seen that accidents in the takeoff phase account for about 12% of all aircraft related accidents (Reference 1). Looking at it from yet another perspective, fatal accidents in the takeoff phase amount to about 15% of all fatal accidents (Reference 1). Reference 1 also indicates that while the accident rate in the other flight phases has been decreasing in recent years, those in the takeoff phase have remained almost constant. Most of these accidents are related to some kind of performance degradation and a vast majority of these could have been averted had there been some system to monitor the progress of the takeoff roll of the airplane and warn the pilot of such a degradation.

Based on their investigation of accidents in the takeoff flight phase, agencies such as the National Transportation Safety Board (NTSB) and the Civil Aviation Authority of U.K. (CAA) have strongly recommended the development of such a system (Reference 1). The Society of Automotive Engineers (SAE) has felt it important enough to create a committee to prepare a set of standards for the development of such a system.

The concept of takeoff monitoring is nothing new. Almost since the beginning of regulated aviation operation, this phase of flight has been of concern. A history of takeoff monitoring has been given in reference 1. One of the methods proposed was based on checking the speed of the airplane at a predetermined point on the runway (Reference 2). Another simple approach found in the literature is a check on the time required to attain a

specified speed. Other more involved procedures as well as some elaborate ground based systems were proposed and are described in Reference 1.

A typical takeoff run

A typical takeoff run goes as follows. Once the airplane has been configured for the takeoff run, approprite thrust is applied. After brake release, the airplane begins to accelerate. If there is an engine failure before the decision speed, V_1, is reached, the pilot elects to abort the takeoff and stops the airplane. Once past the decision speed, even with an engine failure, the pilot is required by the Federal Aviation Regulations to take the airplane through rotation.

At any point during the takeoff roll, the amount of runway required to achieve rotation speed is a function of the instantaneous speed of the airplane and how well it will accelerate until rotation speed. The instantaneous acceleration of the airplane is given by

$$a = \frac{Th - D - \mu (W - L)}{m} \qquad \text{-- (1.1)}$$

where

a =	acceleration	(feet/sec/sec)
Th=	thrust	(lbs)
D =	drag	(lbs)
μ =	rolling friction coefficient	
W =	weight	(lbs)
L =	lift	(lbs)
m =	mass = W/g	(slugs)

g = gravitational acceleration

(feet/sec/sec)

Figure 1.1 shows the forces acting on the airplane. The thrust varies with the airspeed of the airplane; drag and lift are functions of the square of the airspeed and the friction coefficient depends on the runway condition.

Properties of a good takeoff monitoring system

A takeoff monitoring system, to be useful to the pilot as a decision tool or as a command system needs to continuously evaluate the status of the airplane and detect any performance deficiencies. This system also needs to keep track of the runway used to that instant and the runway available which can be used to achieve rotation speed. The monitor should have the ability to adapt to the prevalent loading and ambient conditions such as weight, temperature, pressure altitude, runway winds and rolling friction coefficient. This monitor can be either a ground based or an aircraft system.

Study Goal

The goal of this study is to develop a takeoff performance monitoring system that meets the following requirements.

* The system must be self contained. By having the entire system on the airplane, its operation becomes airport independent.
* The system must detect performance deficiencies by comparing the airplane's present performance with a nominal performance for the given conditions.

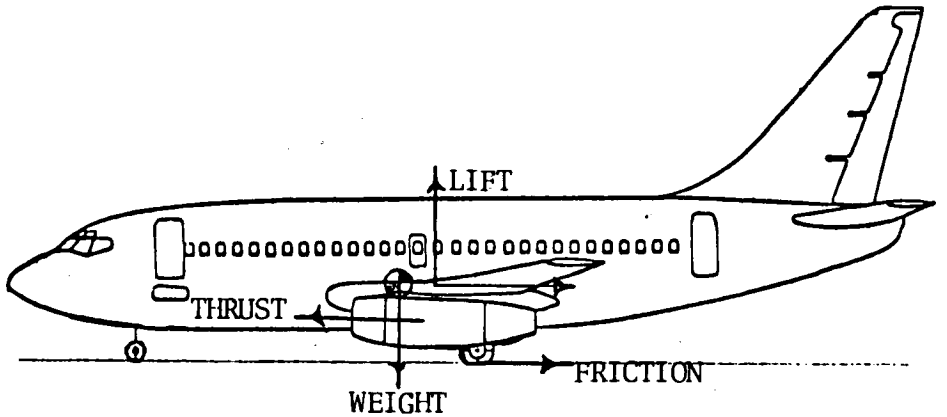

Figure 1.1: The Instantaneous Forces Acting on an Airplane During Takeoff Roll

* The system must keep track of the airplane's position on the runway. By computing runway used, the runway remaining for further action is known.
* The system should have the ability to predict the runway that would be required to achieve rotation speed or to bring the airplane to a complete stop.
* The system should be capable of operating in both an informative mode and in a command generation mode.

Method Options and Difficulties

The nominal performance mentioned above can be obtained in two ways. The first calculates the performance for all possible combinations of ambient conditions (pressure altitude, temperature, runway winds, and rolling friction coefficient), loading conditions (weight and center of gravity), and airplane configuration (flap setting). Also to be included in this list is the possibility of reduced thrust takeoffs. The resulting performance figures could be stored in a computer memory for later lookup and comparison. This is clearly seen to be impossible because of the large number of combinations of the parameters.

The second approach is to have a good model of the aircraft available so that for each takeoff run the scheduled performance can be calculated in real time and thus available for comparisons. The model must include the dynamics of the airplane as well as the engines. Ideally the engine model will include parameters that when checked against measured values will serve as a check on engine health.

Two major problems that must be overcome in the development of the monitor system are the evaluation of the runway rolling friction coefficient

and an estimation of the thrust developed by the engine. As seen from equation 1.1 these two parameters affect the acceleration performance of the airplane significantly. Yet these are two of the most difficult to estimate. No sensor is available to directly measure either the thrust developed by the engines or the rolling friction coefficient. The runway rolling friction coefficient plays an important part in determining the runway needed (Reference 3) but is a very difficult parameter to estimate (Reference 4).

Document Outline

The Takeoff Performance Monitoring System developed during this effort is described in Chapter 2. Chapter 3 gives a brief description of a six degree of freedom simulation model of the Transport Systems Research Vehicle Boeing 737 airplane used to test the algorithm. The normal performance of the algorithm is evaluated in Chapter 4. The test cases include ten combinations of ambient and loading conditions and two types of engine malfunctions. Chapter 5 explores the sensitivity of the algorithm to errors in inputs, and the effects of failures of sensors. Conclusions and recommendations for further work are detailed in Chapter 6.

Appendix A describes the technique used to model a first order lag network in the discrete domain. Appendix B details the development of a second order complementary filter network. Appendix C includes detailed flow charts of the real-time segment of the algorithm.

Details of the project organization, finances, and scheduling are described in Appendix D.

CHAPTER 2

TAKEOFF PERFORMANCE MONITOR ALGORITHM DESCRIPTION

2.1 Introduction

The method used in this study for monitoring the takeoff performance of an airplane is to continuously compare the plane's actual performance with a nominal performance. A prediction of the runway needed to attain rotation speed, the runway required to stop the airplane, and the runway remaining augment this method. In chapter 1, it was discussed that the generation of "nominal performance" data covering all possible situations and the storage of this information for real time reference is impractical. In addition, chapter 1 points out that thrust is not directly measurable on board the airplane.

This chapter discusses a takeoff monitoring algorithm which overcomes the difficulties of generating and storing scheduled performance data covering all possible situations, the measurement of thrust aboard the airplane and the uncertainty associated with the assessment of runway rolling friction coefficient.

2.2 The Takeoff Performance Monitor Algorithm

The takeoff performance monitor algorithm consists of two segments: a pretakeoff segment and a real-time segment. The pretakeoff segment generates the nominal acceleration performance data prior to the start of the takeoff roll. During the takeoff roll, the real-time segment compares the measured acceleration with the scheduled acceleration performance data, predicts the runway length required to either attain rotation speed or to stop the airplane, and based on these generates Go/Abort signals or commands. These segments are described in the following two subsections.

2.2.1 The pretakeoff segment

The airplane's acceleration performance is predicted for two extreme values of rolling friction coefficient: one being a low value (μ=0.005) and the other being a high value (μ=0.040). The algorithm consists of three parts as shown in Figure 2.1 and can be run off-line in the onboard computers or in ground support computers with the results down loaded to the airplane computers.

The first step, using airplane weight, center of gravity, runway pressure altitude, ambient temperature, and selected flap setting obtains from the flight manual the recommended engine pressure ratio for takeoff, the static throttle setting to achieve the engine pressure ratio, decision speed (V_1), and the rotation speed (V_R).

The second step of the pretakeoff algorithm computes the airplane's nominal acceleration and true airspeed performance time histories using an iterative process with a time step of 0.05 second.

In the third step, a least squares cubic polynomial curvefit is performed on the true airspeed-acceleration data computed in the second part. The coefficients obtained from the curvefit are stored for use by the real time segment of the algorithm.

2.2.1.1 Data Requirements:

The pretakeoff segment of the algorithm requires several parameters be input at the start of the calculations. These inputs are the Ambient Conditions, the Loading Data, and the Vehicle Configuration summarized in Table 2.1.

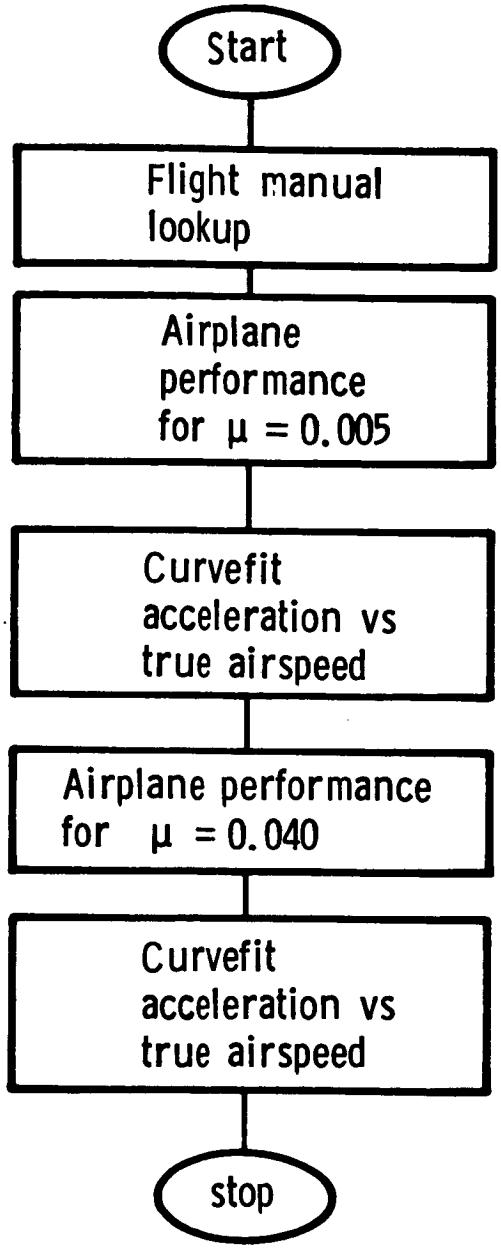

Figure 2.1: The Pretakeoff Segment

Table 2.1: Parameters needed for the Pretakeoff Segment

AMBIENT CONDITIONS
Pressure Altitude
Ambient Temperature
LOADING AND CONFIGURATION INFORMATION
Airplane Weight
Center of Gravity location
Selected Flap Setting

2.2.1.2 Airplane Performance Computation

The airplane nominal acceleration performance is generated using a detailed model of the airplane that includes an aerodynamic model, an engine model and a landing gear dynamics model. These three components are used in an iterative procedure with all variables being initialized to nominal values for the first pass. The procedure is summarized in the next paragraph.

The aerodynamic package is used to compute the lift, drag, and pitching moment coefficients. These coefficients are used to calculate the forces and moments in the stability axis system and are then transformed into the body axis system. These body axis forces and moments are combined with the thrust and landing gear forces and moments to obtain the total forces and moments on the airplane. These forces, combined with the acceleration due

to gravity, result in linear and angular accelerations along the three body axes. These accelerations and velocities are numerically integrated to yield new velocities, linear positions and angular orientations. These new positions and velocities are used in computing new aerodynamic coefficients and the whole process is repeated until rotation speed is achieved. The equations are developed in detail below.

Figure 2.2 details the axis systems used in this development (Reference 5). The center of gravity of the vehicle is the origin for both the body and stability axis systems. The body X-axis (X_B) is along the longitudinal axis of the airplane, with positive direction towards the nose. The body Y-axis (Y_B) is parallel to the wing span and the positive direction is to the right in the top view. The stability Y-axis (Y_S) coincides with the Y_B axis. The positive body Z-axis (Z_B) points downward and is perpendicular to both X_B and Y_B. The stability axis system is another right-handed coordinate axis system obtained by rotations from the body axis system. The stability X-axis (X_S) lies along the projection of the velocity vector onto the X_B-Z_B plane. The positive direction of the stability Z-axis (Z_S) points downward and is at right angles to the X_S axis. The airplane angle of attack (α), the pitch attitude angle (θ), and the flight path angle (γ) are also illustrated in Figure 2.2.

The aerodynamic coefficients, stored as part of the aerodynamic data base for the airplane, are functions of motion variables (α, $\dot{\alpha}$, q, n_z) and control positions (δ_E, δ_F) (Reference 5).

$$C_L = f(\alpha, \dot{\alpha}, q, n_z, \delta_E, \delta_F, i_H, H_{CG}, \text{Gear}) \qquad \text{-- (2.1a)}$$

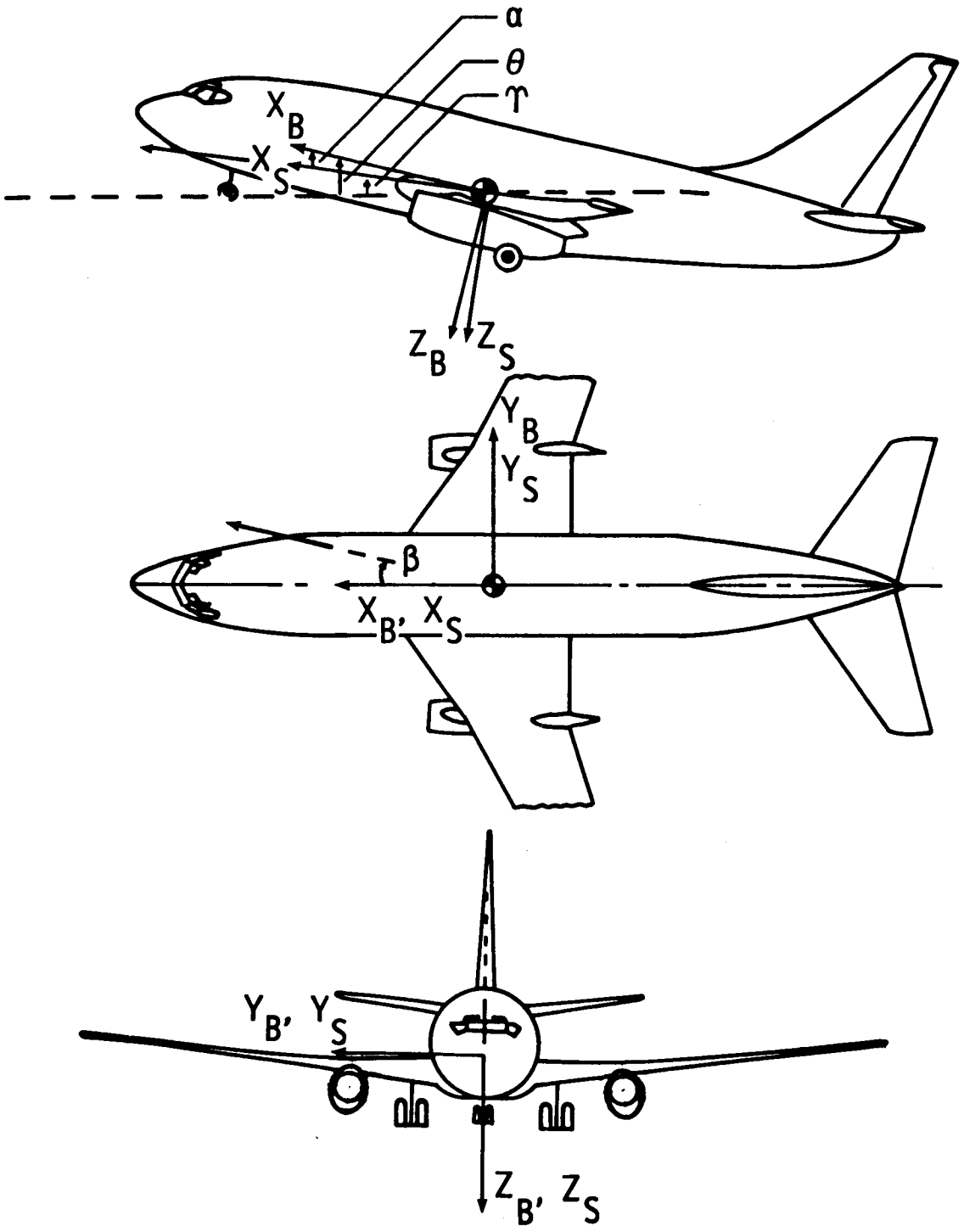

Figure 2.2: The Axes System.

$$C_D = f(C_L, H_{CG}, \text{Gear}) \qquad \text{-- (2.1b)}$$

$$C_m = f(\alpha, q, n_z, \delta_E, \delta_F, i_H, H_{CG}, \text{Gear}) \qquad \text{-- (2.1c)}$$

The angle of attack (α) is initialized to the nominal static pitch attitude. All angular rates and linear velocities are initialized to zero. The stabilizer setting (i_H) is obtained from the flight manual lookup. The height above the runway (H_{CG}) is initialized to the static center of gravity height with normal gear compression. The load factor (n_z) is initialized to unity. The flap deflection (δ_F) is chosen by the pilot and is an input to the flight manual segment of the non-real time algorithm

The dynamic pressure ($\bar{q}$) is calculated as follows:

$$\theta = \theta_B - \theta_{RWY} \qquad \text{-- (2.2)}$$

$$v_T = \sqrt{(u_B + u_W \cos\theta)^2 + (w_B + u_W \sin\theta)^2} \qquad \text{-- (2.3a)}$$

$$\bar{q} = \frac{1}{2} \rho v_T^2 \qquad \text{-- (2.3b)}$$

where

θ_B = pitch attitude angle

θ_{RWY} = runway slope (positive for an up slope)

v_T = true airspeed

u_B = body X-axis inertial velocity

w_B = body Z-axis inertial velocity

u_W = runway wind (head wind is positive)

The aerodynamic coefficients, the computed dynamic pressure, and the airplane geometric constants ($\bar{c}$, S) are used in the following equations to compute the forces along the stability X and Z axes and the pitching moment.

$$F_{X_S} = -\bar{q}\ S\ C_D$$

$$F_{Z_S} = -\bar{q}\ S\ C_L \qquad \text{-- (2.4)}$$

$$M_S = \bar{q}\ S\ \bar{c}\ C_m$$

where $\bar{c}$ = mean aerodynamic chord

S = reference wing area

The two forces and the moment are then transformed into the body axis by

$$F_{X_B} = -\ F_{X_S}\ \text{Cos}\ \alpha - F_{Z_S}\ \text{Sin}\ \alpha$$

$$F_{Z_B} = F_{X_S}\ \text{Sin}\ \alpha + F_{Z_S}\ \text{Cos}\ \alpha \qquad \text{-- (2.5)}$$

$$M_B = M_S$$

Engine thrust (Th), engine pitch attitude (θ_T), and the X and Z engine moment arms (x_T, z_T) are used in computing the engine forces and moments along the body axis, as follows:

$$T_{X_B} = \text{Th}\ \ \text{Cos}\ \theta_T \qquad \text{-- (2.6a)}$$

$$T_{Z_B} = -\ \text{Th}\ \ \text{Sin}\ \theta_T \qquad \text{-- (2.6b)}$$

$$T_{M_B} = -\ T_{Z_B}\ x_T + T_{X_B}\ z_T \qquad \text{-- (2.6c)}$$

where Th = engine thrust, generated by the manufacturer's engine mathematical model

Forces and moments produced by the landing gear (obtained from the manufacturer supplied gear dynamics model) are represented by the variables $L_{G_{X_B}}, L_{G_{Z_B}}, L_{G_{M_B}}$.

The resultant forces acting through the center of gravity along the body X-axis and the Z-axis are obtained by

$$F_{X_{B_{total}}} = F_{X_B} + T_{X_B} + L_{G_{X_B}}$$

$$F_{Z_{B_{total}}} = F_{Z_B} + T_{Z_B} + L_{G_{Z_B}} \qquad \text{-- (2.7a,b)}$$

The resultant moment about the body Y-axis or pitching moment is denoted by

$$M_{B_{total}} = M_B + T_{M_B} + L_{G_{M_B}} \qquad \text{-- (2.7c)}$$

Using these forces, moments, and body X and Z components of gravitational acceleration, the airplane accelerations along the body axes are computed as

$$\dot{u}_B = (F_{X_{B_{total}}}/m) - g \text{ Sin } \theta_B + (r_B v_B - q_B w_B)$$

$$\dot{w}_B = (F_{Z_{B_{total}}}/m) - g \text{ Cos } \theta_B + (q_B u_B - p_B v_B) \qquad \text{-- (2.8)}$$

$$\dot{H}_{CG} = u_B \text{ Sin } \theta - w_B \text{ Cos } \theta$$

In the above equation, the angular rate about the body X-axis (p_B) and that about the body Z-axis (r_B) are assumed to remain unchanged at zero. In addition, the angular rate about the body Y-axis, and the linear velocities

about the body X, Y and Z axis (u_B, v_B, w_B respectively), are initialized to zero.

Combining these speeds and accelerations, the rate of change of angle of attack is calculated as

$$\dot{\alpha} = (u_B \dot{w}_B - w_B \dot{u}_B) / (u_B^2 + w_B^2) \qquad \text{-- (2.9)}$$

The pitching moment combined with the body Y-axis moment of inertia (obtained from the aerodynamic data base as a function of weight), results in pitch acceleration.

$$\dot{q}_B = M_{B_{total}} / I_{YY_B} \qquad \text{-- (2.10)}$$

Rate of change of pitch attitude is written as

$$\dot{\theta}_B = q_B \qquad \text{-- (2.11)}$$

The flight path angle (γ) and the ground speed (v_G) are computed as:

$$\gamma = \theta_B - \alpha \qquad \text{-- (2.12)}$$

$$v_G = \sqrt{(u_B^2 + w_B^2)} \ \text{Cos}\ \gamma \qquad \text{-- (2.13)}$$

The parameters ($\dot{\theta}_B$, $\dot{u}_B$, $\dot{H}_{RWY}$, $\dot{w}_B$, $\dot{q}_B$, $\dot{v}_G$) are integrated using the second order Adam-Bashworth numerical integration scheme (equation 2.14) to obtain new values for (θ_B, u_B, H_{RWY}, w_B, q_B, D_{RWY}).

$$x_{n+1} = x_n + \frac{\Delta T}{2} (3 \dot{x}_n - \dot{x}_{n-1}) \qquad \text{-- (2.14)}$$

The thrust forces used in this model are generated using the manufacturer supplied engine mathematical model. Throttle position, true

airspeed, ambient temperature, ambient pressure, and present engine pressure ratio are inputs to this model. The outputs are engine pressure ratio, engine thrust, compressor and turbine stage rotation speeds, fuel flow rate and exhaust gas temperature. This information is summarized in Figure 2.3.

A nominal throttle movement time history that duplicates typical operational procedures was selected . This involves moving the throttle to an intermediate setting and waiting until the engine pressure ratio attains a prescribed value, and then moving the throttle to the suggested takeoff setting. This throttle movement is replicated by the model.

The commanded throttle position is processed by a throttle servo before being input to the engine mathematical model. This throttle servo is represented by a first order lag with transfer function

$$\frac{\delta_{th}(S)}{\delta_{th_C}(S)} = \frac{a}{S+a} \qquad \text{-- (2.15)}$$

which, after discrete transformation (based on References 6 and 7 and described in Appendix A) is written as

$$\delta_{th}(nT) = \xi\, \delta_{th}[(n-1)T] + (1-\xi)\, \delta_{th_C}(nT) \qquad \text{-- (2.16)}$$

where $\delta_{th}(nT)$ = servo output at time `nT'

$\xi = e^{-aT} = 0.60653$

$\delta_{th_C}(nT)$ = throttle command at time `nT'

T = sampling interval = 0.05 second

The landing gear forces and moments are also generated by a manufacturer supplied mathematical model. The inputs to this model and the outputs from it are illustrated in Figure 2.4.

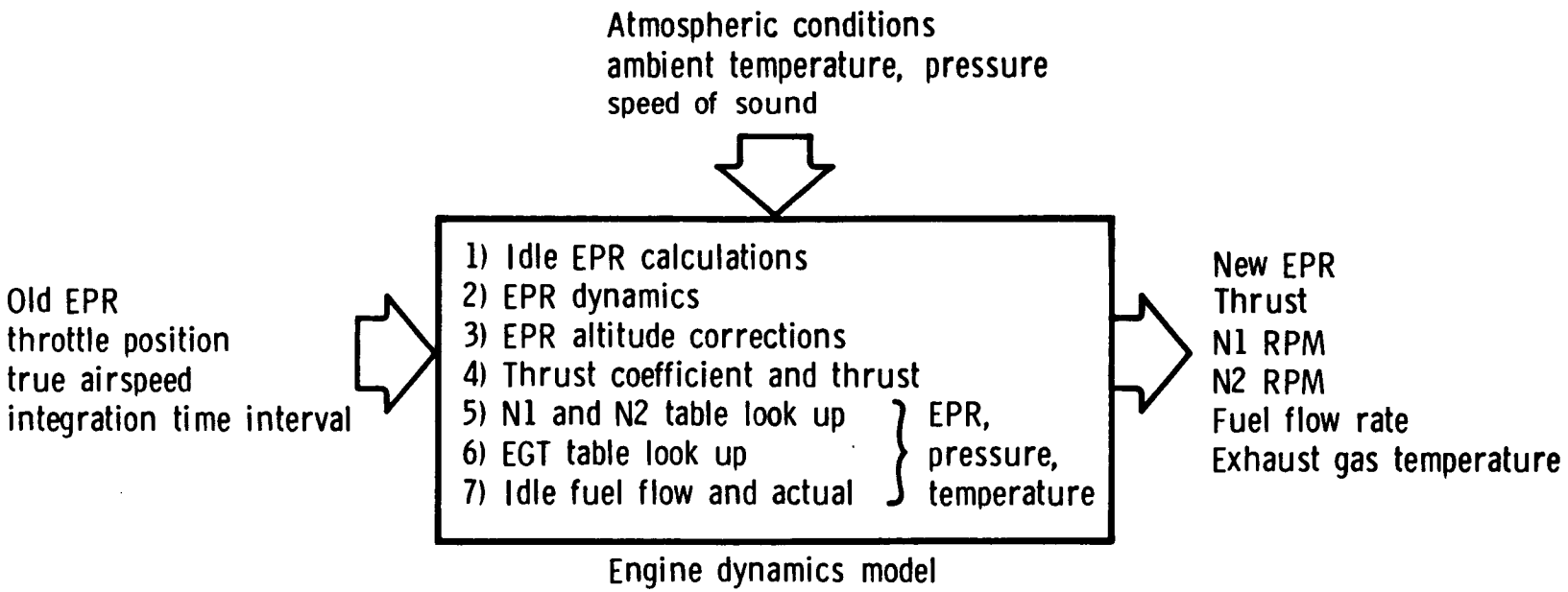

Figure 2.3: Engine Model, Inputs, and Outputs

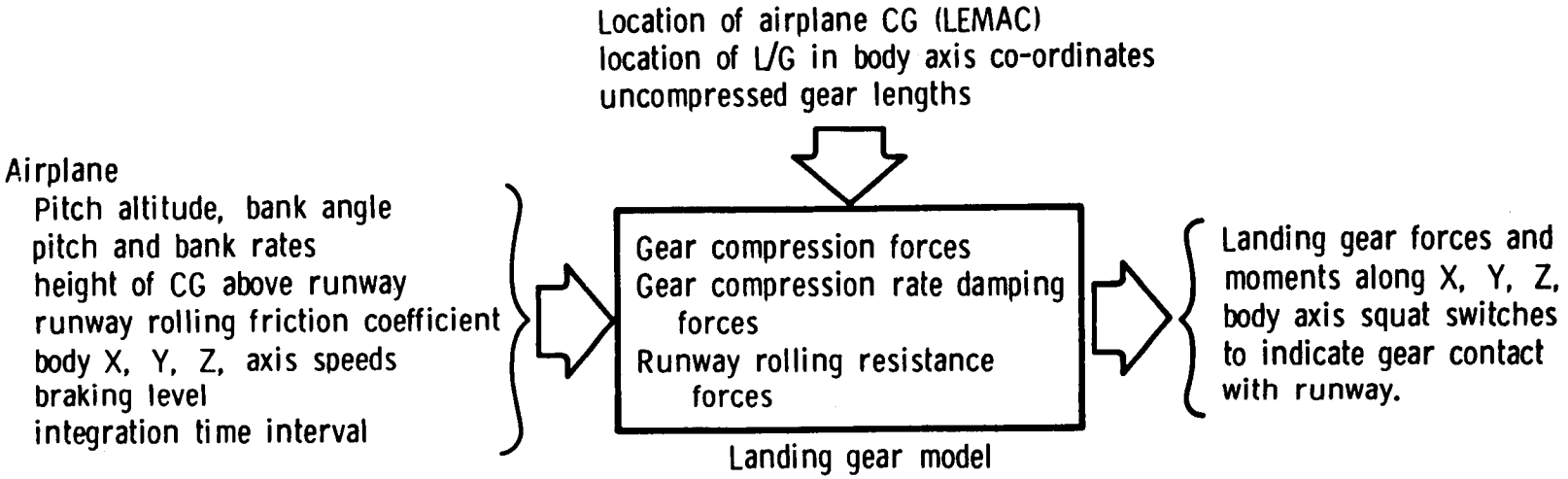

Figure 2.4: Landing Gear Dynamics Model, Inputs, and Outputs

2.2.1.3 Curve Fitting

The next block of the algorithm pretakeoff portion deals with curve fitting the airplane true airspeed ,v_T, (the independent variable) and the along track acceleration ,a, (the dependent variable) to generate a set of coefficients for a "scheduled performance" data set for the takeoff run. The curve fitting process is chosen to minimize the memory requirement to store nominal performance data and to facilitate the process of estimating the runway rolling friction coefficient. A least square error cubic polynomial curvefit method (Reference 8) is utilized to generate

$$a = A_0 + A_1 v_T + A_2 v_T^2 + A_3 v_T^3 \qquad \text{-- (2.17)}$$

This process is carried out twice; once for a low friction coefficient ($\mu = 0.005$) and a second time for a high friction coefficient ($\mu = 0.040$). The low friction coefficient value is chosen to be slightly lower than the nominal lower bound suggested in Reference 9. Similarly the high value is chosen to be higher than the nominal upper bound suggested in the same Reference 9.

Table 2.2 summarizes the flight conditions for a typical takeoff case. Figure 2.5 illustrates the acceleration versus true airspeed curves obtained from the pretakeoff computations for the conditions of Table 2.2. It is seen from the figure that the curve-fit matches the computed curve only after the peak value of acceleration has been attained. This is because the monitor algorithm comes on-line after the dynamics due to the throttle has died out.

Table 2.2 : Flight Conditions for the sample Pretakeoff Calculations

Weight	=	88504	lbs
Center of Gravity	=	19% $\bar{c}$	behind LEMAC
Flap setting	=	5	deg
Pressure Altitude	=	32	feet
Ambient Temperature	=	75	deg F

where LEMAC = leading edge of mean aerodynamic chord

2.2.2 Real Time Segment

The monitor system's real time segment is used 10 times per second in the flight control or navigation computer during the actual takeoff run. This segment performs the following functions.

1) Initially commands the throttle to the required throttle setting for takeoff
2) Monitors the engine in terms of its engine pressure ratio
3) Monitors the performance of the airplane in terms of its acceleration performance
4) Estimates the runway rolling friction coefficient
5) Predicts the runway required to achieve rotation speed
6) Predicts the runway required to stop the airplane and
7) Generates go or abort signals.

It also has the capability to command the application of brakes and reverse thrust. To accomplish these tasks, the algorithm requires a number of aircraft parameter measurements. Prior to use by the system, aircraft sensor data is passed through a filter network. The airplane performance is

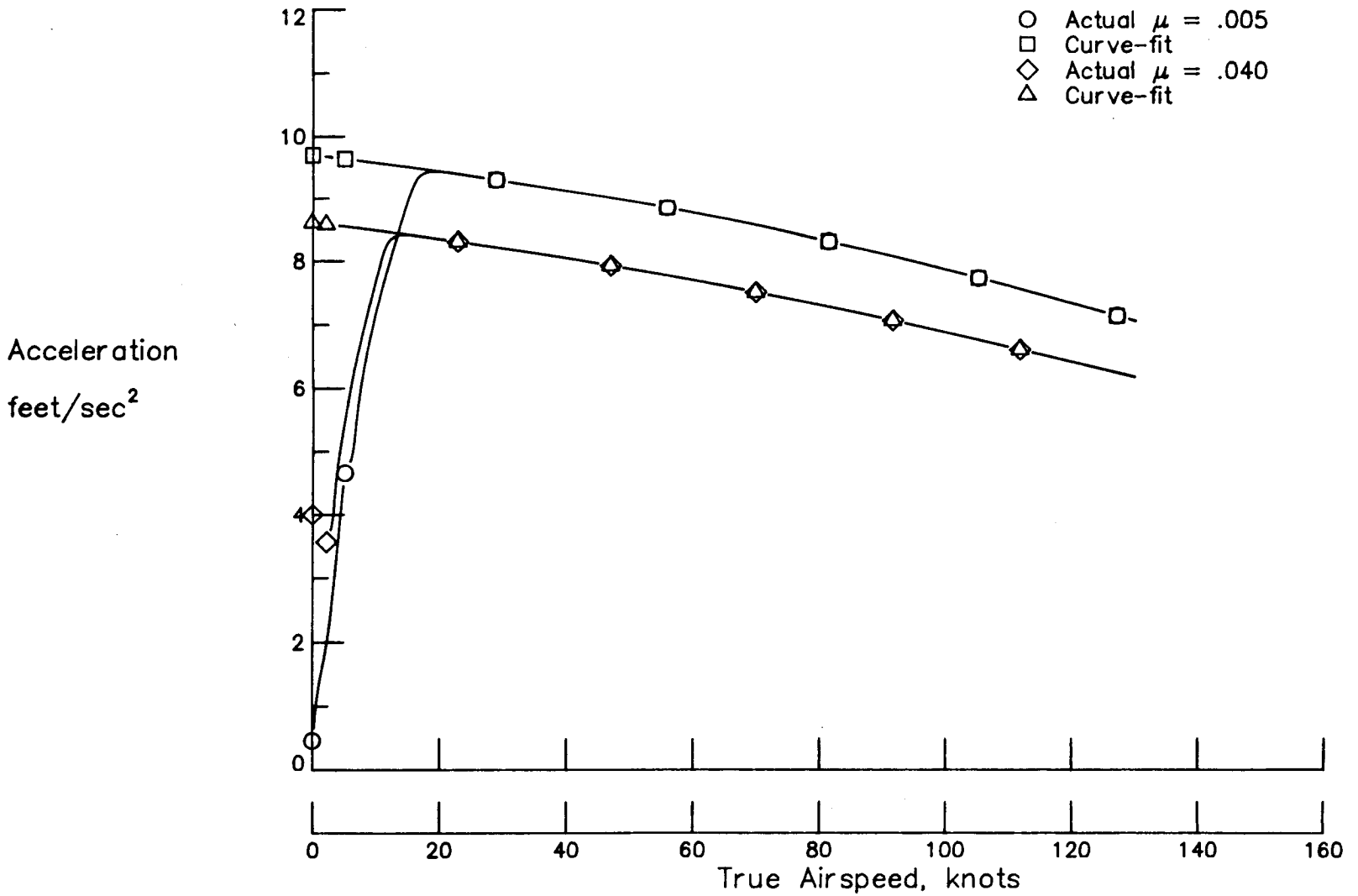

Figure 2.5: Acceleration versus True Airspeed Data from the Pretake-off Segment and the Corresponding Cubic Polynomial Curve Fit.

computed utilizing the filtered sensor data in a point mass formulation for motion along the runway. This computed performance figure is compared with the sensor measured values. The difference is used either to estimate a rolling friction coefficient - a one time operation, or is utilized in the generation of Go/Abort signals. Measured and calculated values are employed during each iteration to predict the runway length required to achieve rotation speed and to bring the airplane to a stop. A comparison of the available and required runway lengths is also used to generate a Go/Abort signal. A functional flow diagram is shown in Figure 2.6 and a detailed programming flow chart of the real time segment in included in Appendix C.

2.2.2.1 Data Requirements

The real-time segment requires several input parameters. Some of these are one-time inputs while others are continuously needed inputs. The one-time inputs include: ambient temperature, ambient pressure, runway winds, airplane weight, stabilizer setting, and flap selection. All of the above values are obtained from the pretakeoff segment. Three additional necessary one-time parameters are runway available for rotation, runway available to stop the airplane, and a nominal rolling friction coefficient.

The parameters needed on a continuous basis are supplied by sensors and include: left and right throttle position, left and right engine pressure ratio, ground speed, along track acceleration, and calibrated airspeed.

The data requirements for the real time segment are summarized in Table 2.3

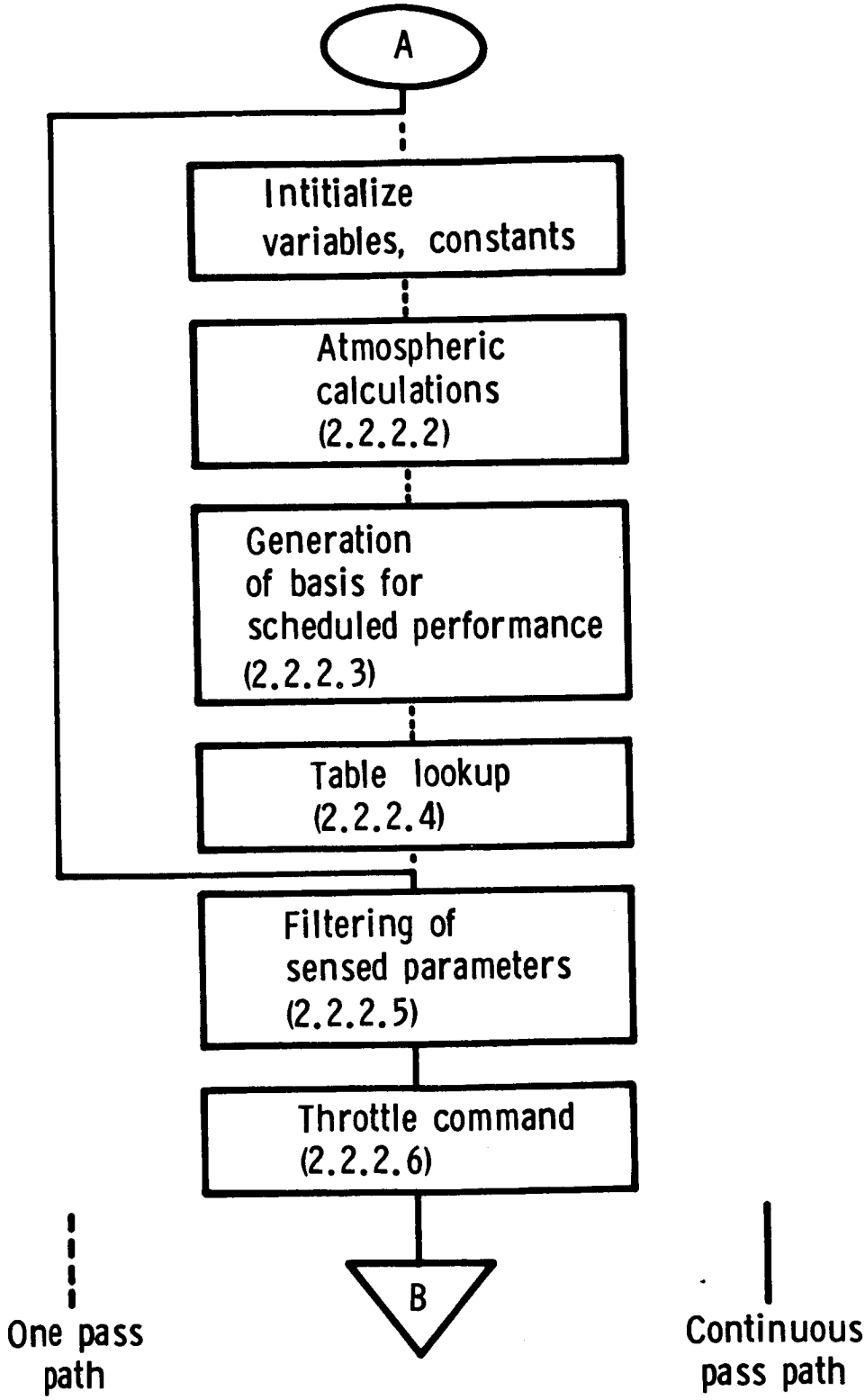

Figure 2.6: The Real-Time Segment.

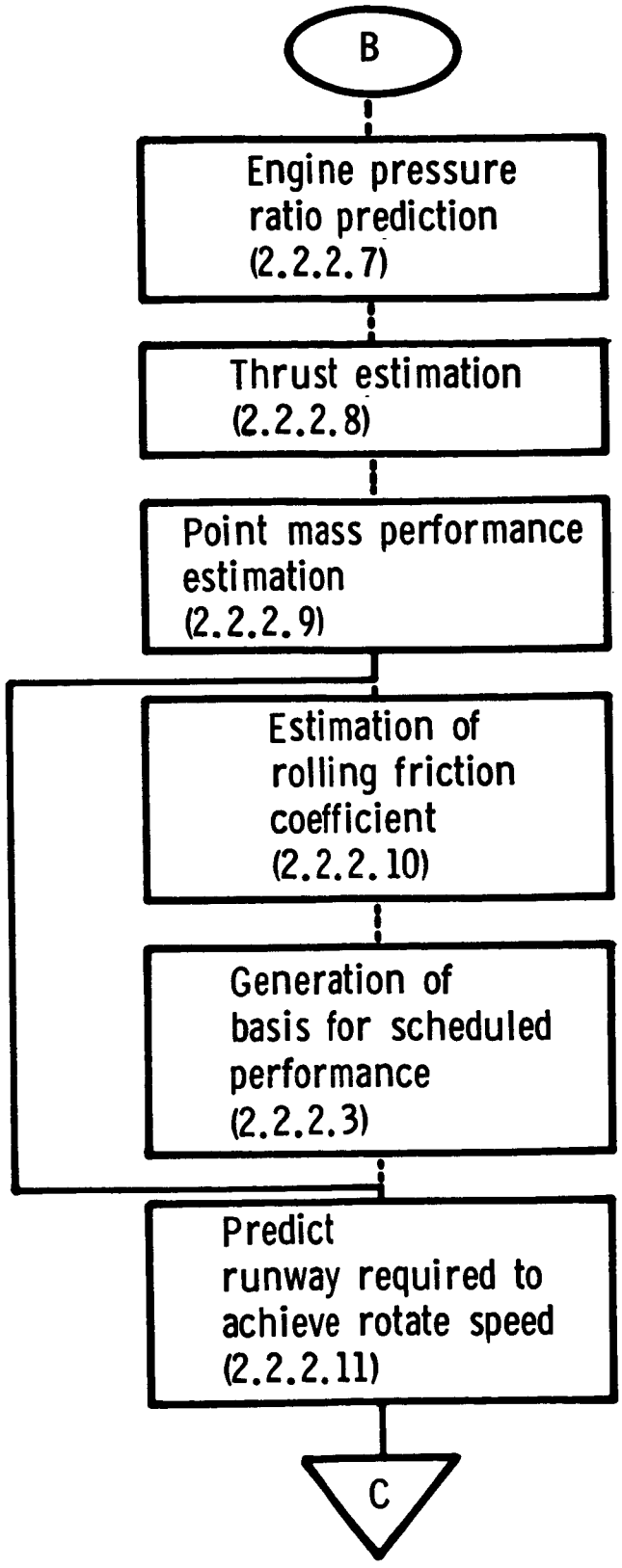

Figure 2.6: Continued

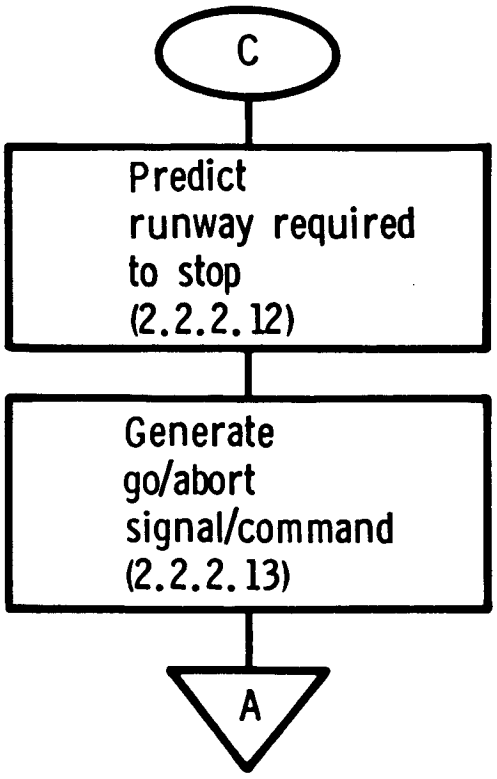

Figure 2.6: Continued.

Table 2.3: Parameters needed for the Real Time Segment

ONE-TIME INPUTS

Ambient Temperature
Ambient Pressure
Runway Wind
Weight
Flap Selection
Stabilizer Setting

} Obtained from the Pretakeoff segment

Runway Available for Rotation

Runway available for Stopping

Nominal Rolling Friction Coefficient

NEEDED CONTINUOUSLY

Left & Right Throttle position

Left & Right Engine Pressure Ratio

Ground Speed

Along Track Acceleration

Calibrated Airspeed

2.2.2.2 Atmospheric Calculations

In this block, the pressure altitude and ambient temperature inputs are used with ideal gas laws to calculate air density, temperature and pressure ratios, and the speed of sound. This is a one-time operation and is used during the first pass through the real-time segment.

2.2.2.3 Generation of Scheduled Performance Basis

In the pretakeoff segment a cubic curve fit was performed to obtain acceleration as a function of true airspeed as given by equation 2.17. This was done for two rolling friction coefficients. In the present section a set of coefficients is obtained for the nominal rolling friction coefficient by linear interpolation as indicated below:

$$A_{0_\mu} = \frac{A_{0_{\mu_1}} (\mu_2 - \mu) + A_{0_{\mu_2}} (\mu - \mu_1)}{(\mu_2 - \mu_1)} \qquad \text{-- (2.18a)}$$

$$A_{1_\mu} = \frac{A_{1_{\mu_1}} (\mu_2 - \mu) + A_{1_{\mu_2}} (\mu - \mu_1)}{(\mu_2 - \mu_1)} \qquad \text{-- (2.18b)}$$

$$A_{2_\mu} = \frac{A_{2_{\mu_1}} (\mu_2 - \mu) + A_{2_{\mu_2}} (\mu - \mu_1)}{(\mu_2 - \mu_1)} \qquad \text{-- (2.18c)}$$

$$A_{3_\mu} = \frac{A_{3_{\mu_1}} (\mu_2 - \mu) + A_{3_{\mu_2}} (\mu - \mu_1)}{(\mu_2 - \mu_1)} \qquad \text{-- (2.18d)}$$

where

μ_1 = low rolling friction coefficient

μ_2 = high rolling friction coefficient

μ = present estimate rolling friction coefficient

The basis for scheduled acceleration performance becomes

$$a = A_{0_\mu} + A_{1_\mu} v_T + A_{2_\mu} v_T^2 + A_{3_\mu} v_T^3 \qquad \text{-- (2.19)}$$

This procedure is carried out one time at the beginning of the real-time segment using the pilot input friction coefficient for μ, and a second time with the algorithm estimated friction coefficient (section 2.2.2.10).

2.2.2.4 Table Lookup

Three table lookups are performed in this block. In the first table lookup, a stabilizer setting (i_H) is obtained as a function of the center of gravity position and could be displayed in the cockpit. Also the nominal lift and drag coefficients (C_L, C_D respectively), increments in lift and drag coefficients with full flight spoiler deflections ($\Delta C_{L_{FSP}}$, $\Delta C_{D_{FSP}}$), and increments in lift and drag coefficients with full ground spoiler deflections ($\Delta C_{L_{GSP}}$, $\Delta C_{D_{GSP}}$), are obtained as functions of flap deflection. The ground effect is accounted for by assuming a nominal gear compression.

The second table lookup is a repeat of the flight manual lookup carried out during the pretakeoff segment. Using airplane weight, runway pressure altitude, ambient temperature, and selected flap setting, the decision speed (V_1), and the rotation speed (V_R) are obtained from the flight manual.

The third table lookup identifies the recommended takeoff engine pressure ratio (EPR) setting and the corresponding throttle setting, and the throttle setting necessary to achieve the intermediate EPR value (for the nominal throttle movement of section 2.2.1.1). All of these are functions of ambient temperature and pressure.

2.2.2.5 Filtering of sensed parameters

The measured acceleration and ground speed values are processed through a second order complementary filter to estimate the bias present in the acceleration signal. The details of this complementary filter implementation are described in Appendix B (Reference 10). The final equations are

$$\underline{x}_{n+1} = \Phi\, \underline{x}_n + \Gamma\, \underline{u}_n \qquad \text{-- (2.20a)}$$

where

$$\underline{x} = \begin{bmatrix} x(1) \\ x(2) \end{bmatrix}$$

$$\underline{u} = \begin{bmatrix} v_G \\ a_M \end{bmatrix}$$

$$\Phi = \begin{bmatrix} 0.8584 & 0.0928 \\ -0.0464 & 0.9976 \end{bmatrix}$$

$$\Gamma = \begin{bmatrix} 0.1416 & 0.0928 \\ 0.0464 & -0.0024 \end{bmatrix}$$

$$\hat{v}_G = x(1) \qquad \text{-- (2.20b)}$$

$$a_{F1} = a_M + x(2) \qquad \text{-- (2.20c)}$$

During the first pass through the real time segment, x(1) is initialized to the measured ground speed and x(2) is initialized to zero. The filter

output for both ground speed and acceleration is the measured value itself.A new set of values is then calculated for the x's. During subsequent passes the x values used are from the previous pass and after use they are updated.

The acceleration output from the complementary filter above, and the measured values of EPR left and right, calibrated airspeed (CAS), and true airspeed (TAS) are passed through a first order lag filter ,to remove noise, as depicted below.

$$
\begin{aligned}
a &= \xi\, a_0 + (1 - \xi)\, a_{F1} \\
EPR_{\substack{left \\ right}} &= \xi\, EPR_{0_{\substack{left \\ right}}} + (1 - \xi)\, EPR_{M_{\substack{left \\ right}}} \\
CAS &= \xi\, CAS_0 + (1 - \xi)\, CAS_M \qquad \text{-- (2.21)} \\
TAS &= \xi\, TAS_0 + (1 - \xi)\, TAS_M
\end{aligned}
$$

with $\xi = 0.7304026$

where the subscript "O" stands for old and the subscript "M" stands for measured sensor data.

During the first pass

$$a_0 = a_{F1}$$

and the other old values are set equal to the measured values. For subsequent passes the old value is the filter output from the previous pass.

2.2.2.6 Throttle Command

This throttle command is identical to the throttle command system in the pretakeoff segment. This involves moving the throttle to an intermediate setting and waiting until the engine pressure ratio attains a prescribed value, and then moving the throttle to the suggested takeoff setting.

2.2.2.7 Engine Pressure Ratio Prediction

An empirical model of the steady state behavior of the Engine Pressure Ratio (EPR) is extracted from the manufacturer supplied engine model and is used in the real time segment to predict what the EPR should be. The transients caused by throttle movement and engine spool up are neglected in this model.

$$\hat{EPR}_{\substack{left \\ right}} = f(\delta_{th_{\substack{left \\ right}}}, T_{total}) \qquad \text{-- (2.22)}$$

where

$\hat{EPR}_{\substack{left \\ right}}$ = left & right estimated EPR

$\delta_{th_{\substack{left \\ right}}}$ = left & right measured throttle position

T_{total} = total (stagnation) temperature

2.2.2.8 Thrust Estimation

In a fashion similar to the EPR computations, thrust calculation is performed with an empirical relation extracted from the manufacturer supplied engine model. Thrust estimation is based on measured EPR and computed Mach number $(\hat{M})$

$$\hat{T}_{\substack{left \\ right}} = f(EPR_{\substack{left \\ right}}, \hat{M}) \qquad \text{-- (2.23)}$$

2.2.2.9 Point Mass Performance Estimation

The developments of this section are based on Reference 9.

True airspeed is the sum of wind speed and filtered ground speed.

$$\hat{v}_T = v_G + u_W \qquad \text{-- (2.24)}$$

The estimated true airspeed is used in estimating a Mach number, and with the density computed by the atmospheric calculations, a dynamic pressure is calculated. Using this dynamic pressure, and nominal lift and drag coefficients from the table lookup, the lift and drag forces are estimated. A friction force is estimated using the estimated rolling friction coefficient (pilot input value until the algorithm estimates it as described in section 2.2.2.10 and after that the algorithm estimated value), weight of the airplane (assumed to remain constant), and the estimated lift force. Using this information an acceleration is estimated. An estimate of the ground speed is obtained by performing a rectangular integration on the estimated acceleration. An estimate of the runway used is obtained by performing a rectangular integration using the filtered ground speed. The equations pertinent to these estimations are as follows:

$$\hat{M} = \hat{v}_T / A \qquad \text{-- (2.25a)}$$

where A = Speed of sound

$$\bar{q} = \frac{1}{2} \rho \hat{v}_T^2 \qquad \text{-- (2.25b)}$$

$$\hat{L} = \bar{q} S C_L \qquad \text{-- (2.25c)}$$

$$\hat{D} = \bar{q} S C_D \qquad \text{-- (2.25d)}$$

$$\hat{F}_{fr} = \mu (W - \hat{L}) \qquad \text{-- (2.25e)}$$

$$\hat{a} = (\hat{T}_{left} + \hat{T}_{right} - \hat{D} - \hat{F}_{fr}) / m \qquad \text{-- (2.25f)}$$

$$\hat{v}_G = \hat{v}_{G_0} + \Delta T \hat{a}_0 \qquad \text{-- (2.25g)}$$

$$D_{RWY} = D_{RWY} + \Delta T v_G \qquad \text{-- (2.25h)}$$

with $\Delta T = 0.1$ second

2.2.2.10 Estimation of rolling friction coefficient

As discussed earlier, rolling friction coefficient is not easily assessed. In the real time segment, the difference between the estimated and measured acceleration is used to estimate a rolling friction coefficient. This is done as follows. First, thrust is represented as a cubic in true airspeed

$$T = T_0 + T_1 v_T + T_2 v_T^2 + T_3 v_T^3 \qquad \text{-- (2.26)}$$

Then using equations (2.25b) through (2.25e) and (2.26), the acceleration equation (2.26f) can be rewritten as

$$a = g((T_0 - \mu W) + T_1 v_T + (T_2 - \tfrac{1}{2}\rho S C_D + \tfrac{1}{2}\mu\rho S C_L) v_T^2 + T_3 v_T^3) / W \qquad \text{-- (2.27)}$$

At any given true airspeed for two given runway friction coefficients the two acceleration expressions are represented as

$$\hat{a} = g((T_0 - \hat{\mu}_1 W) + T_1 v_T + (T_2 - \tfrac{1}{2}\rho S C_D + \tfrac{1}{2}\hat{\mu}_1 \rho S C_L) v_T^2 + T_3 v_T^3) / W \qquad \text{-- (2.28)}$$

$$a = g((T_0 - \hat{\mu}_2 W) + T_1 v_T + (T_2 - \tfrac{1}{2}\rho S C_D + \tfrac{1}{2}\hat{\mu}_2 \rho S C_L) v_T^2 + T_3 v_T^3) / W$$

Subtracting a from $\hat{a}$

$$\hat{a} - a = g(W - \tfrac{1}{2} \rho S C_L v_T^2) (\hat{\mu}_2 - \hat{\mu}_1) / W$$

or solving the above equation for the difference between the friction coefficients

$$\Delta\hat{\mu} = \hat{\mu}_2 - \hat{\mu}_1 = (\hat{a} - a) / (g (W - \tfrac{1}{2} \rho S C_L v_T^2) / W) \qquad \text{-- (2.29)}$$

where $\hat{\mu}_2$ = estimate of the actual runway friction coefficient

$\hat{\mu}_1$ = assumed friction coefficient

$\Delta\hat{\mu}$ = estimated amount by which the friction coefficient has to be adjusted

Thus the actual runway friction coefficient can be estimated as

$$\hat{\mu}_2 = \hat{\mu}_1 + \Delta\hat{\mu} \qquad \text{-- (2.30)}$$

The above procedure is a one time operation carried out early into the takeoff run. Immediately after this process the basis for scheduled performance (section 2.2.2.3) is repeated with $\hat{\mu}_2$ as the present estimate of the friction coefficient.

2.2.2.11 Prediction of the Runway Required

The runway required to achieve rotation speed is computed by a numerical integration scheme. The difference between the present speed and the rotation speed (true airspeed for rotation) is divided into ten equal velocity increments.

$$v_{step} = (v_{T_{rotation}} - \hat{v}_T) / 10 \qquad \text{-- (2.31a)}$$

The speed at the center of the first interval is

$$v = \hat{v}_T + v_{step} / 2 \qquad \text{-- (2.31b)}$$

The variation of acceleration with speed is accounted for by using the scheduled performance equation (equation 2.19) with the v calculated above (equation 2.31b).

$$a = A_0 + A_1 v + A_2 v^2 + A_3 v^3 \qquad \text{-- (2.31c)}$$

The time for each speed interval is given by

$$\Delta t = v_{step} / a \qquad \text{-- (2.31d)}$$

and the distance needed for each interval can be approximated as

$$\Delta s = \Delta t \, (v - u_W)$$

$$= v_{step} \, (v - u_W) / a \qquad \text{-- (2.31e)}$$

The center speed of the next interval is computed by adding v_{step} to the present center speed. This whole process is repeated ten times to cover the entire speed range. The sum of all the Δs's gives a prediction of the runway required to achieve rotation speed. A detailed flow chart of this mechanization is included in Appendix C.

2.2.2.12 Prediction of Stopping Distance

To calculate stopping distance, the takeoff monitor system simulates the effect of a series of commands to the airplane in order to determine the distance required to stop the vehicle if these actions were actually taken. No commands are actually passed to the airplane to bring the vehicle to a stop.

Computation of stopping distance is based on the following assumptions.

1) The flight spoilers are commanded through servos modelled as a first order lag.

2) The ground spoilers are commanded through servos modelled as a first order lag.

3) With full braking the rolling friction coefficient is increased by a constant amount over the prevalent value.

4) Maximum wheel braking is achieved in a ramp fashion per given time period.

5) Thrust is assumed to vary linearly with throttle position (based on present thrust and throttle position).

6) The changes in lift and drag coefficients produced by flight and ground spoilers are assumed to vary linearly with deflection.

The variation of lift and drag coefficients per unit deflection of the flight and ground spoilers is computed.

$$\frac{\Delta C_{L_{FSP}}}{\delta_{FSP}} = \frac{\Delta C_{L_{FSP_{max}}}}{\delta_{FSP_{max}}} \qquad \text{-- (2.32a)}$$

$$\frac{\Delta C_{D_{FSP}}}{\delta_{FSP}} = \frac{\Delta C_{D_{FSP_{max}}}}{\delta_{FSP_{max}}} \qquad \text{-- (2.32b)}$$

$$\frac{\Delta C_{L_{GSP}}}{\delta_{GSP}} = \frac{\Delta C_{L_{GSP_{max}}}}{\delta_{GSP_{max}}} \qquad \text{-- (2.32c)}$$

$$\frac{\Delta C_{D_{GSP}}}{\delta_{GSP}} = \frac{\Delta C_{D_{GSP_{max}}}}{\delta_{GSP_{max}}} \qquad \text{-- (2.32d)}$$

where subscript "FSP" stands for Flight Spoilers

subscript "GSP" stands for Ground Spoilers

The variation of thrust per unit deflection of the throttle are calculated.

$$\frac{T}{\delta_{th}} = 2 * \frac{\hat{T}_{left_{present}} + \hat{T}_{right_{present}} - \hat{T}_{left_{idle}} - \hat{T}_{right_{idle}}}{\delta_{th_{left}} + \delta_{th_{right}}}$$

-- (2.32e)

where $\hat{T}_{\substack{left \\ right}_{idle}}$ = estimated thrust for zero throttle

Next the rate of increase of braking is computed.

$$\frac{\Delta\mu_{brake}}{t} = \frac{\Delta\mu_{brake_{max}}}{t_{ramp}} \qquad \text{-- (2.32f)}$$

where $\Delta\mu_{brake_{max}}$ = maximum increment in rolling friction coefficient from full braking

= 0.45

and t_{ramp} = ramp time for full braking

= 0.6 second

A throttle command to zero, and flight and ground spoilers being commanded to their full deflections are simulated. These simulated movements are slaved through their respective servos.

$$\delta_{th_{new}} = \xi_{th}\ \delta_{th_{old}} \qquad \text{-- (2.33a)}$$

$$\delta_{GSP_{new}} = \xi_{GSP}\ \delta_{GSP_{old}} + (1 - \xi_{GSP})\ \delta_{GSP_{max}} \qquad \text{-- (2.33b)}$$

$$\delta_{GSP_{new}} = \xi_{GSP}\ \delta_{GSP_{old}} + (1 - \xi_{GSP})\ \delta_{GSP_{max}} \qquad \text{-- (2.33c)}$$

with $\xi_{th} = 0.36788$

$\xi_{FSP} = 0.36788$

$\xi_{GSP} = 0.13534$

Based on the simulated throttle position and the assumed linear variation of thrust with throttle position, a simulated engine thrust is computed.

$$Th = \frac{T}{\delta_{th}} \delta_{th} \qquad \text{-- (2.34)}$$

Using the simulated flight and ground spoiler positions and the nominal values for the lift and drag coefficients, simulated lift and drag coefficients are computed.

$$\Delta C_{L_{FSP}} = \frac{\Delta C_{L_{FSP}}}{\delta_{FSP}} \delta_{FSP_{new}} \qquad \text{-- (2.35a)}$$

$$\Delta C_{D_{FSP}} = \frac{\Delta C_{D_{FSP}}}{\delta_{FSP}} \delta_{FSP_{new}} \qquad \text{-- (2.35b)}$$

$$\Delta C_{L_{GSP}} = \frac{\Delta C_{L_{GSP}}}{\delta_{GSP}} \delta_{GSP_{new}} \qquad \text{-- (2.35c)}$$

$$\Delta C_{D_{GSP}} = \frac{\Delta C_{D_{GSP}}}{\delta_{GSP}} \delta_{GSP_{new}} \qquad \text{-- (2.35d)}$$

$$C_L = C_{L_{nominal}} + \Delta C_{L_{FSP}} + \Delta C_{L_{GSP}} \qquad \text{-- (2.36a)}$$

$$C_D = C_{D_{nominal}} + \Delta C_{D_{FSP}} + \Delta C_{D_{GSP}} \qquad \text{-- (2.36b)}$$

The increase in rolling friction coefficient is computed as a product of the braking rate and the time elapsed (initialized to zero for the first pass).

$$\Delta\mu_{brake} = \frac{\Delta\mu_{brake}}{t} t$$

$$\text{if } \Delta\mu_{brake} > \Delta\mu_{brake_{max}} \text{ then } \Delta\mu_{brake} = \Delta\mu_{brake_{max}} \qquad \text{-- (2.36c)}$$

The net rolling friction coefficient is computed as the sum of the unbraked rolling friction coefficient and the increase due to braking, calculated above.

$$\mu = \mu_{nominal} + \Delta\mu_{brake} \qquad \text{-- (2.36d)}$$

$$t = t + \Delta t \qquad \text{-- (2.37)}$$

where Δt = integration time step

These lift and drag coefficients, the net rolling friction coefficient, and the weight of the airplane are used in a point mass performance calculation similar to that in section 2.2.2.9. The acceleration is integrated over a time step to determine the change in ground speed. The time elapsed is incremented by the time step. New positions for the throttles, and flight and ground spoilers are computed and the performance calculations repeated. This process is carried out until the ground speed reaches zero. The total distance used in bringing the airplane to a stop in this simulated stopping is the stopping distance. Appendix C includes a program flow chart for the computation of the stopping distance.

2.2.2.13 Generation of Go/Abort Signal/Command

The engine pressure ratio (EPR) is used as a check on engine health. Sufficient time is allowed for the transients caused by throttle movement to die out. After this, the measured and predicted EPRs are compared. If they are different by more than a preselected limit then an engine failure flag is set.

$$\left| \frac{EPR_{left} - \hat{EPR}_{left}}{EPR_{left}} \right| > EPR_{error\ limit} \Rightarrow E.\ Fail_{left} \qquad \text{-- (2.38a)}$$

$$\left| \frac{EPR_{right} - \hat{EPR}_{right}}{EPR_{right}} \right| > EPR_{error\ limit} \Rightarrow E.\ Fail_{right} \qquad \text{-- (2.38b)}$$

with $EPR_{error\ limit} = 0.15$

After the instant at which the rolling friction coefficient is estimated, any difference between the measured and the predicted accelerations causes a performance failure flag to be set.

$$\left| \frac{a - \hat{a}}{a} \right| > a_{error\ limit} \Rightarrow PER.\ Fail \qquad \text{-- (2.38c)}$$

with $a_{error\ limit} = 0.15$

The following conditions result in a Go signal/command:

1) No engine failure flag or performance failure flag is set and the runway available is greater than the runway required to attain rotation speed.

2) Only one engine failure flag is set and the runway remaining is less than that required for stopping the airplane.

3) Performance failure flag is set without either engine failure flag being set and there is insufficient runway length for stopping.

The following conditions result in a Abort signal/command:

1) Runway length available for achieving rotation speed is less than that required.

2) Both the engine failure flags are set.

3) One engine failure flag is set and there is sufficient runway length available for stopping.

4) Performance failure flag is set and sufficient runway length is available for stopping.

CHAPTER 3

SIMULATION MODEL

3.1 Introduction

The previous chapters dealt with the development of an algorithm and the data requirements for that algorithm. Before the algorithm can be implemented on an actual airplane, it has to be validated in a test environment as close to the real operational environment as possible. To accomplish this initial testing and validation of the algorithm, a good model of the plant (in this case the airplane in ground roll) is required. This chapter describes the model used in this testing and development. The parameters computed by the model are treated as the actual quantities. Section 3.2 describes the model used and outlines its main components.

3.2 The Model

The plant model proposed for use in testing the algorithm is the six degree of freedom nonlinear batch simulation model of the Transport Systems Research Vehicle (TSRV) B-737 (Reference 11) which is available on the computer system at NASA Langley Research Center. This batch simulation is implemented to run at the rate of 20 iterations to a second. This gives an iteration time step of 0.05 second.

The batch simulation uses a combination of inertial and wind axes systems in all its computations. Figure 3.1 shows an outline of this batch simulation. The aircraft variables are first initialized as needed for the particular run. The airplane is then trimmed for a steady state flight condition. Once the airplane has achieved trim for the given steady state flight condition, it is then put in the operate mode. In this mode, the

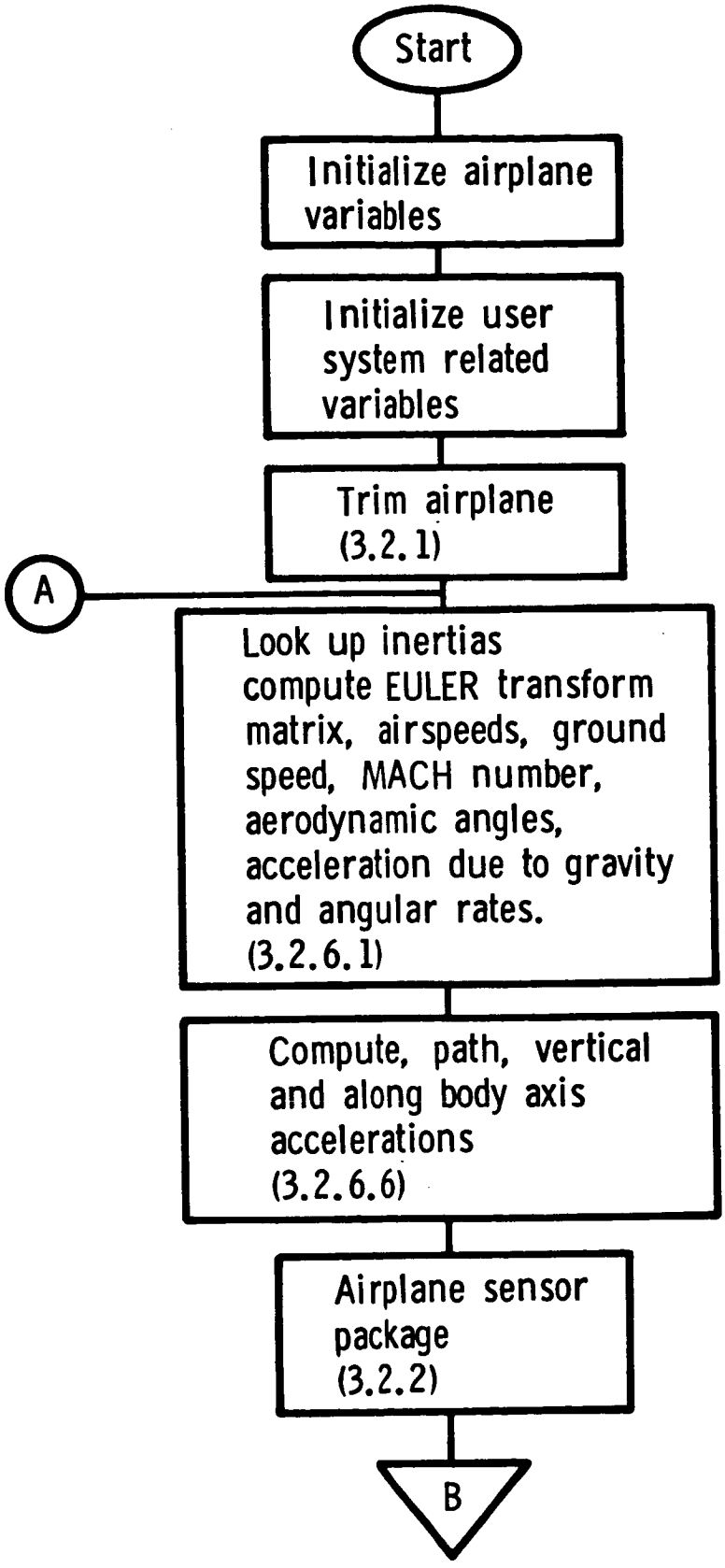

Figure 3.1: Flow Diagram for the Six Degree of Freedom TSRV B-737 Batch Simulation.

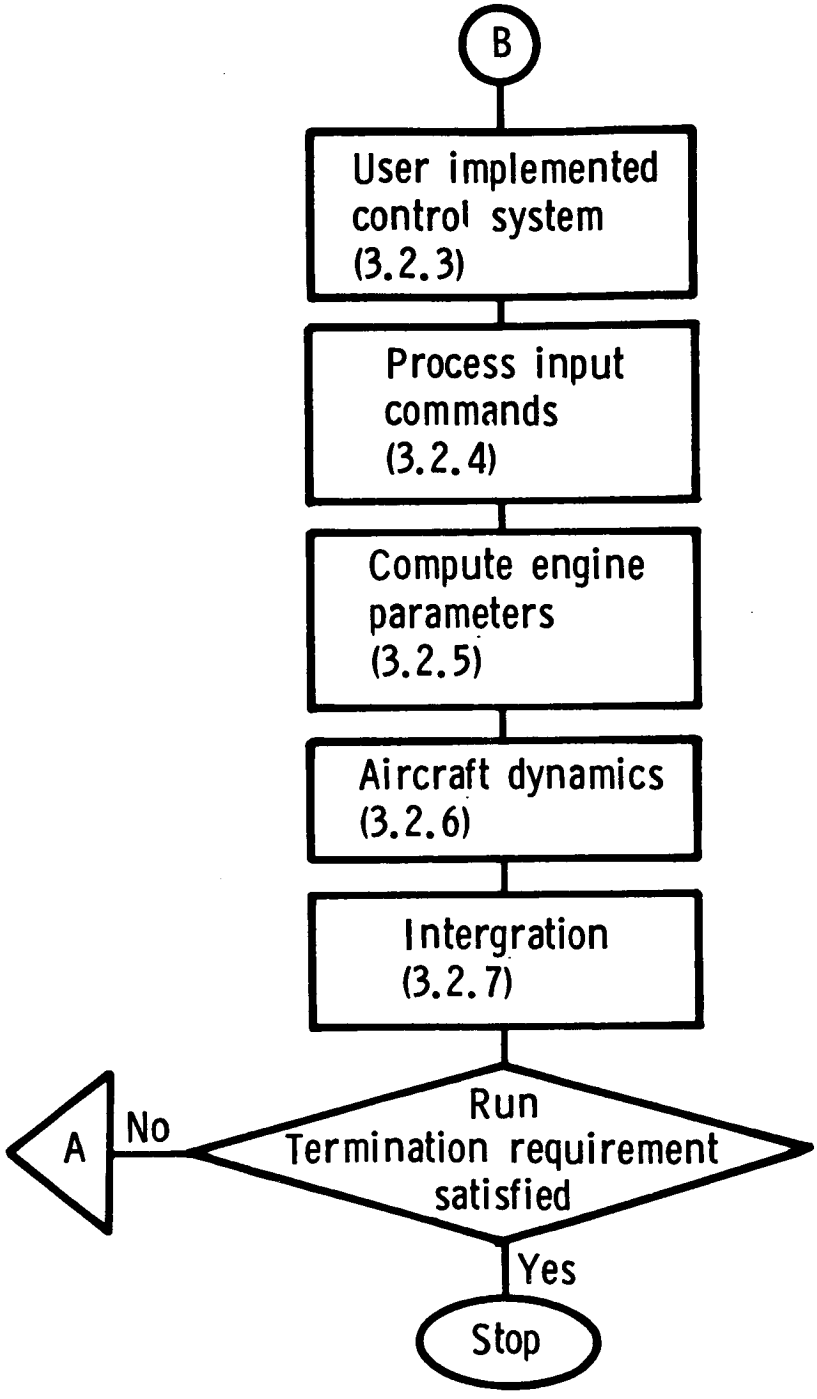

Figure 3.1: Continued

control systems implemented by the user are brought on line. These systems become effective in perturbing the airplane from the trimmed steady state by generating new control inputs. The states of the airplane, control inputs, and constants pertinent to the simulation run serve as inputs to the aircraft dynamics module. The computations in the aircraft dynamics module result in new values for the state derivatives. These state derivatives are then numerically integrated to obtain new states. Control then passes back to the module which contains the user generated control laws. This procedure continues until the simulation is halted because of some parameter having reached a desired value. Each of the modules shown in Figure 3.1 is discussed below.

3.2.1 Trimming the Airplane

Airplane trimming is accomplished using a modified secant iteration method which solves a nonlinear system of equations of the form

$$f_i(x) = 0 \qquad i=1, \quad ,n \qquad \text{-- (3.1)}$$

where $x = (x_1, \quad ,x_n)$ = the independent trim variables

with f_i is a set of n airplane force and moment equations

The trimming process is accomplished by iteratively computing a new n-dimensional vector of independent trim variables satisfying an n-dimensional vector of dependent trim constraints to within a preselected tolerance level.

3.2.2 Aircraft Sensor Package

This module is utilized to simulate real sensors mounted on an actual aircraft with all their attendant noise and bias values. A pseudo random number generator is utilized to superimpose zero mean Gaussian noise signals

with any chosen standard deviations on any of the sensed parameters. This module also includes the capability of adding a constant bias value to the measured parameters. Table 3.1 summarizes the noise characteristics used in this study and represents a set of typical sensor noise characteristics.

Table 3.1 :Noise and Bias Characteristics

All noises are Gaussian with standard deviations as indicated below.

PARAMETER	SIGMA	BIAS
Along track Acceleration (ft/s^2)	0.32	0.32
Pressure Altitude	0.0	0.0
Calibrated Airspeed (kts)	2.0	0.0
True Airspeed (fps)	2.0	4.0
Throttle Position (deg)	0.2	-0.4
Engine Pressure Ratio	0.01	0.02
Engine N_1 RPM	0.01	0.02
Exhaust Gas Temperature (°C)	0.01	0.02
Fuel Flow Rate (lb/hr)	0.01	0.02

where SIGMA -- Standard Deviation

BIAS -- Constant Bias value

3.2.3 User Implemented Control Systems

This is the module where the control system designed by the user will be implemented for testing. This is the module where the Takeoff Performance Monitoring Algorithm developed in the previous chapter will be implemented.

3.2.4 Process Input Commands

The commands generated by the user implemented control system are passed through their respective servos to generate actual positions of control surfaces, and throttles. Any inner loop control systems that are required on the airplane, such as a yaw damper, pitch control system, or a roll control system are included in this module. Any measured inputs to such control systems are affected by the sensor package noises of section 3.2.2.

3.2.5 Compute Engine related Parameters

Figure 3.2 shows the engine module block diagram. The throttle position input to this module is converted into a cross shaft angle. This cross shaft angle is also a function of the difference in the throttle positions between the present and previous iterations. The cross shaft angle position along with the engine inlet stagnation temperature and the Mach number are utilized in computing a commanded engine pressure ratio (EPR). In this process an idle EPR value is computed as a function of a bleed valve controlled by the Mach number and altitude. This bleed valve position (open or closed) is also a function of the EPR value from the previous iteration and whether or not the engine is going through start up. The commanded EPR goes through an EPR dynamics loop which is controlled by the rate at which the EPR is commanded to change. The resulting EPR directly determines the thrust developed by the engine, the fuel flow rate, the compressor stage and

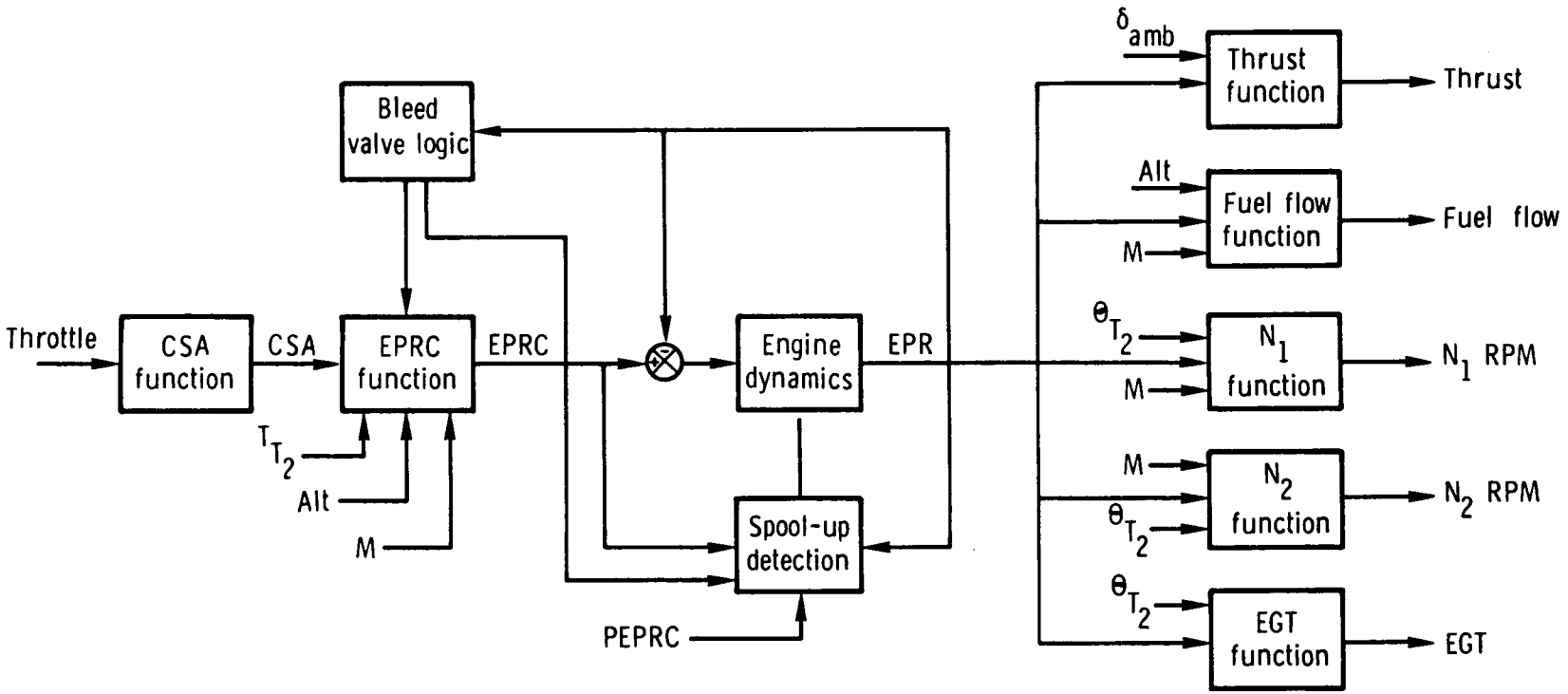

Figure 3.2: Block Diagram of the Engine Model

SYMBOL	DEFINITION
CSA	FUEL CONTROL CROSS SHAFT ANGLE
EPRC	ENGINE PRESSURE RATIO COMMAND
T_{T_2}	ENGINE INLET STAGNATION TEMPERATURE
Alt	PRESSURE ALTITUDE
M	FREE STREAM MACH NUMBER
PEPRC	PREVIOUS ITERATION ENGINE PRESSURE RATIO COMMAND
EPR	ENGINE PRESSURE RATIO
δ_{amb}	AMBIENT PRESSURE RATIO
Θ_{T_2}	ENGINE INLET STAGNATION TEMPERATURE RATIO

Figure 3.2: Continued

turbine stage revolution rates (N_1 and N_2 respectively), and the exhaust gas temperature.

3.2.6 Aircraft Dynamics

Figure 3.3 shows the detailed block diagram representation of the aircraft dynamics module used in the simulation. The aircraft dynamics model implemented in this simulation follows the developments of Reference 12. The inertial velocities of the airplane are transformed to the three airplane body axes. The velocities relative to the airmass, the angle of attack, sideslip angle, true airspeed, calibrated airspeed, Mach number, and the stability axis body angular rates are computed. The airplane stability derivatives are obtained from the aerodynamic data base and the forces and moments along the stability axis are computed. These forces and moments are then transformed into the airplane body axis. The forces and moments generated by the landing gear and the engines along the body axis are computed. The sum of the body axis forces and moments along with the components of the acceleration due to gravity gives the net body axis linear and angular accelerations. The present states, the resultant forces and moments, along with the Euler transformations are used in computing the derivatives of the states:

$$\dot{\underline{x}} = F (\underline{x} , \underline{u}) \qquad \text{-- (3.1)}$$

where $\underline{x}$ = State vector

$\underline{u}$ = Controls

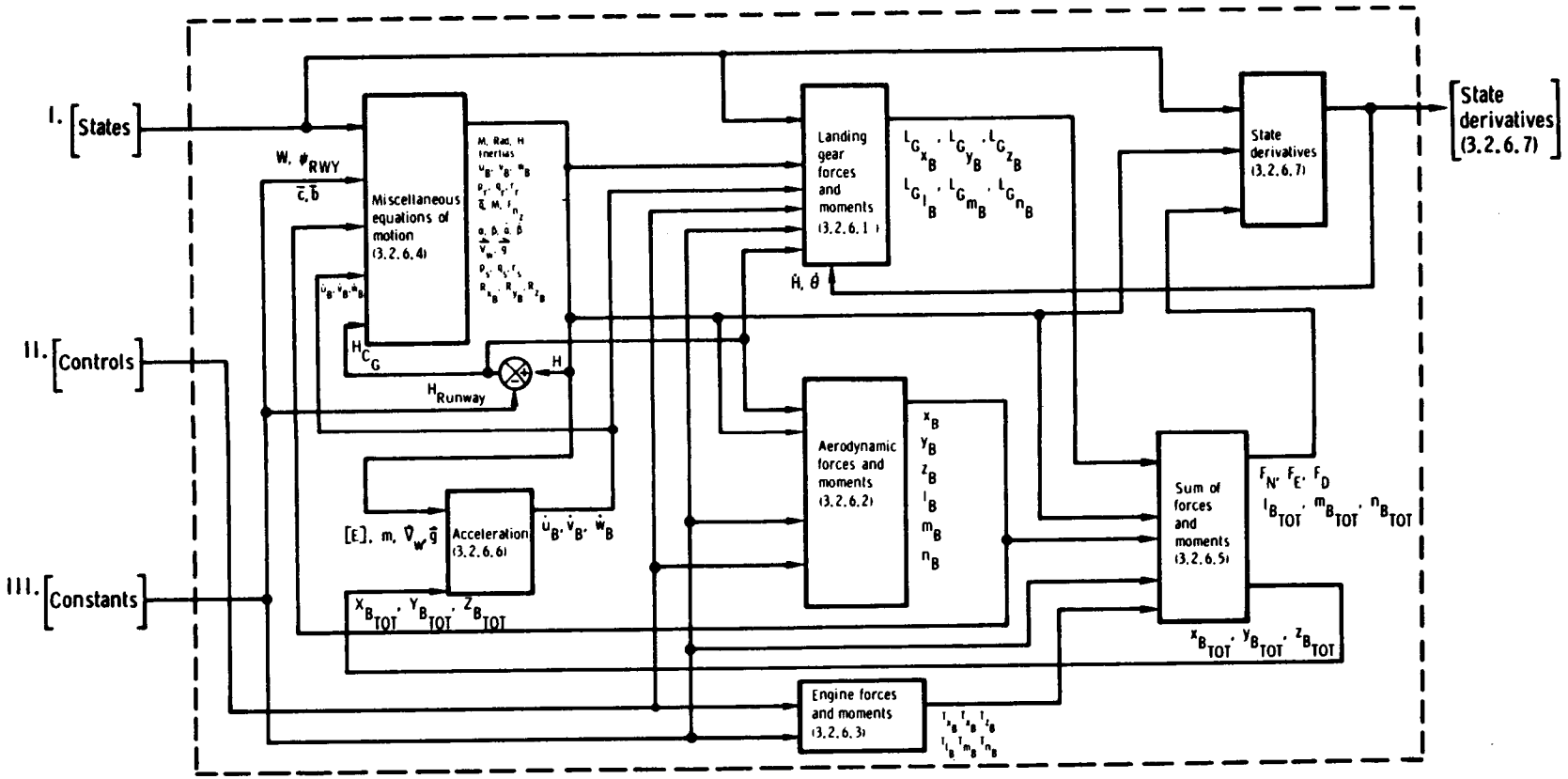

Figure 3.3: Aircraft Dynamics Module.

The state vector is made up of the following parameters:

λ	Airplane Latitude	
τ	Airplane Longitude	
H	Altitude of the Airplane Center of Gravity	
V_N	Velocity North	
V_E	Velocity East	
V_D	Velocity Down	
p_B	Body Axis Roll Rate	
q_B	Body Axis Pitch Rate	
r_B	Body Axis Yaw Rate	
Ψ	Yaw Angle	
Θ	Pitch Angle	Euler Angles
Φ	Bank Angle	

The controls vector consists of

δ_{FLAPS}	Flap Deflection
δ_{STAB}	Stabilizer Deflection
δ_{GEAR}	Gear Up or Down
δ_a	Aileron Deflection
δ_R	Rudder Deflection
δ_e	Elevator Deflection
δ_{SP_R}	Right Spoiler Deflection
δ_{SP_L}	Left Spoiler Deflection

THRUST REVERSE	Yes or No

A brief description of each of the segments of the aircraft dynamics module is given in the following subsections.

3.2.6.1 Landing Gear Forces and Moments

The landing gear forces and moments segment computes the forces and moments along the body axis. These forces ($L_{G_{X_B}}$, $L_{G_{Y_B}}$, $L_{G_{Z_B}}$) and moments ($L_{G_{L_B}}$, $L_{G_{M_B}}$, $L_{G_{N_B}}$) are based on the pitch attitude and bank angles, pitch and roll rates, body axis velocities, computed gear spring forces (based on computed oleo compression), and gear damping forces (based on gear compression forces). The rolling friction forces and the braking forces generated at the gears are also computed in this segment. These forces are transformed into body axis forces and moments.

3.2.6.2 Aerodynamic Forces and Moments

The aerodynamic forces and moments acting on the airplane are computed in this segment. This is accomplished in three steps. First the values for the aerodynamic derivatives (C_L, C_D, C_Y, C_m, C_l, C_n) are computed by utilizing a table lookup based on the aerodynamic angles, proximity to the ground, control surface deflections, and Mach number. Using the dynamic pressure, and airplane constants (mean aerodynamic chord, reference wing area, and wing span), the forces and moments along the stability axis are calculated. These stability axis forces and moments are then transformed to the body axis.

3.2.6.3 Engine Forces and Moments

The engine thrust forces (generated from the engine model) are combined with the engine orientation in the body axis system to calculate the forces and moments produced by the engine along the body axis.

3.2.6.4 Miscellaneous Equations of Motion

This segment includes equations of motion not covered in any of the other segments. The Euler transformation matrix computation, airplane mass, table lookup for moments and products of inertia, sines and cosines of the longitude and latitude, airplane velocities relative to the airmass in the inertial frame and their transformations to the body axis system, computations of the aerodynamic angles (α, β), true and calibrated airspeeds, dynamic pressure, Mach number, body angular rates along the stability axis and their nondimensional equivalents, components of the acceleration due to gravity along the body axis, and acceleration due to body rates are computed in this segment.

3.2.6.5 Sum of Forces and Moments

The forces and moments generated from the different segments described above are added in this segment to generate the net body axis forces and moments acting on the airplane. The total body axis forces are also transformed into the inertial coordinate frame relative to the airmass.

3.2.6.6 Accelerations

The accelerations caused by the total forces acting on the airplane (section 3.2.6.5) are computed using the force-mass relationship. The net accelerations in the body frame are computed as a sum of force generated accelerations, the components of the acceleration due to gravity and the accelerations resulting from the body angular rates (section 3.2.6.4). The

along track, cross track, and vertical accelerations are computed as functions of inertial velocities and accelerations.

3.2.6.7 State Derivatives

The state derivatives are computed using the properties of the inertial and wind axis systems. These are based on the total forces and body angular rates (section 3.2.6.5), the mass of the airplane and the gravitational forces (section 3.2.6.4), and the states themselves.

3.2.7 Integration

The state derivatives are integrated using the Adam-Bashworth numerical integration scheme of equation (2.14). The fuel flow rate along with the iteration time step is used in computing the amount of fuel used and hence a new weight for the airplane.

3.3 Model limitations

The model does not allow the airplane to be trimmed at zero equivalent speed. This necessitates the takeoff run to start at some small non-zero airspeed such as true airspeed of 0.5 knot. Similarly, the model can not be brought to a complete stop on abort. The model does not provide for varying the runway friction force from zero until the limiting frictional force value, and hence if an adequate forward thrust is not provided, the frictional force will cause the airplane to move backwards.

The simulation model can not handle sloped runways.

CHAPTER 4

TAKEOFF MONITOR ALGORITHM EVALUATION

4.1 Introduction

A takeoff performance monitoring algorithm was described in chapter 2. Chapter 3 described a plant with which to evaluate this algorithm. This chapter describes the evaluation of the algorithm using the Transport Systems Research Vehicle (TSRV) B-737 model. Section 4.2 describes the setup of the batch simulation for this evaluation. Section 4.3 presents normal takeoff test cases and two performance degradation cases are considered in section 4.4.

4.2 The Batch Simulation Setup

The batch simulation of the takeoff performance monitoring algorithm is a two part process. The first consists of running the monitor system pretakeoff portion and culminates with the storing of nominal performance data. At the completion of the first part, the batch simulation of the TSRV B-737 begins with the input of ambient conditions and airplane loading information from the same input file as the first part.

The batch simulation is updated every 0.05 second (20 times a second) while the takeoff performance monitoring algorithm is executed ten times each second. This is accomplished by calling the algorithm every other iteration. The simulation is stopped after the calibrated airspeed from the batch simulation model of the airplane reaches the flight manual recommended rotation speed for the given airplane loading.

4.3 Normal Takeoff Test Cases

To demonstrate the algorithm operation ten cases are presented. Each of these ten cases represents one possible combination of ambient and airplane

loading condition and are listed in Table 4.1. An actual airplane, for these cases, would have gone through takeoff roll and successfully rotated off the runway after reaching the flight manual recommended rotation speed.

Table 4.1: Normal Takeoff Test Cases

CASE NO	PRESSURE ALTITUDE (feet)	AMBIENT TEMP. (° F)	RUNWAY WINDS (knots)	WEIGHT (lbs.)	RUNWAY FRICTION COEFF. (-)
I	32.	75.	0.	88504.	.015
II	0.	75.	0.	88504.	.015
III	100.	75.	0.	88504.	.015
IV	32.	0.	0.	88504.	.015
V	32.	100.	0.	88504.	.015
VI	32.	75.	10.	88504.	.015
VII	32.	75.	20.	88504.	.015
VIII	32.	75.	0.	88504.	.025
IX	32.	75.	0.	88504.	.007
X	32.	75.	0.	98000.	.015

Figures 4.1 shows the time histories obtained from the simulation run for case I, assuming perfect aircraft sensors. Figure 4.2 shows the time histories for the same case with realistic sensor noises as detailed in Table 3.1.

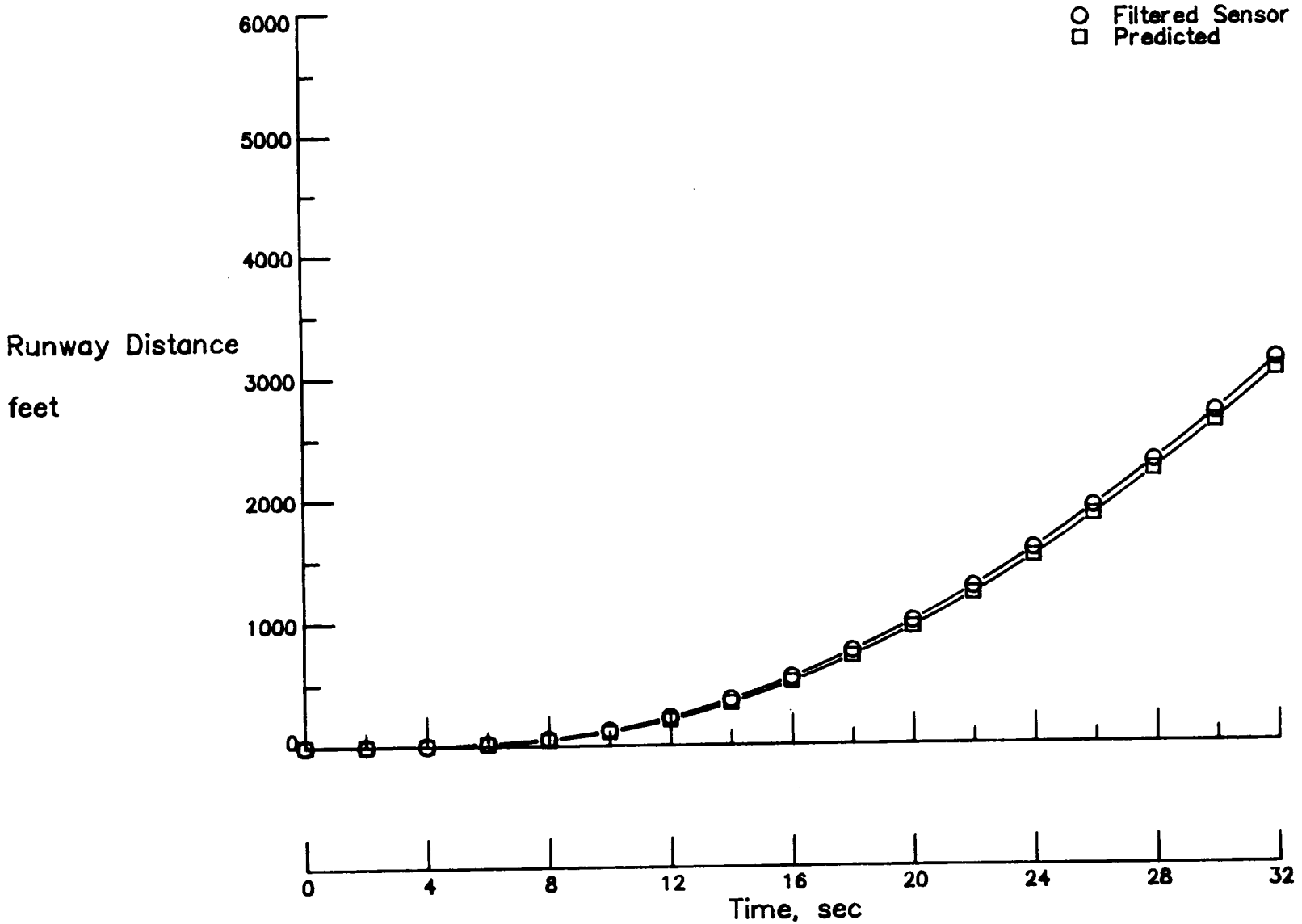

Figure 4.1a: Runway Used Time History for Case I With Perfect Sensors.

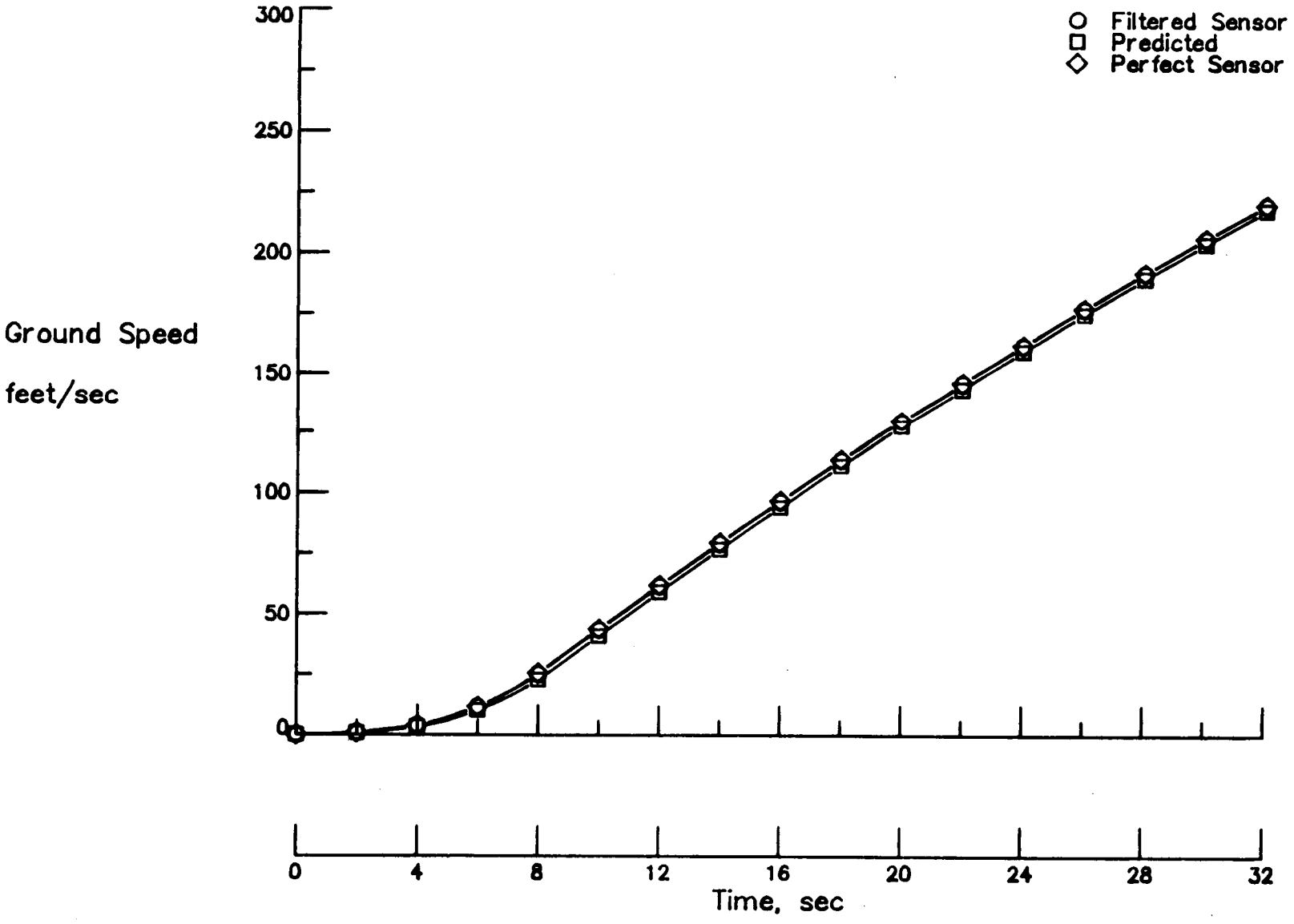

Figure 4.1b: Ground Speed Time Histories for Case I With Perfect Sensors.

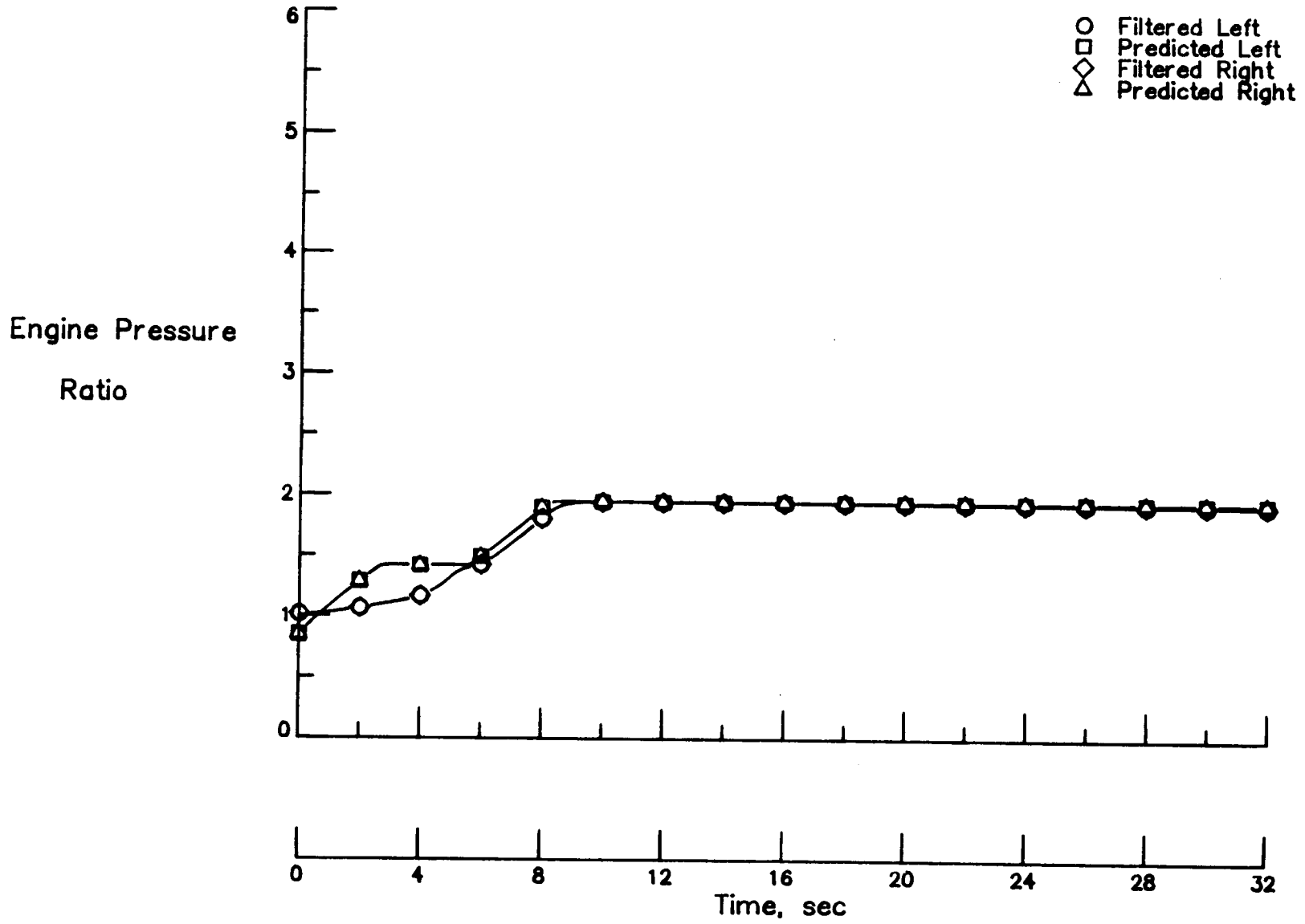

Figure 4.1c: Engine Pressure Ratio Time Histories for Case I With Perfect Sensors.

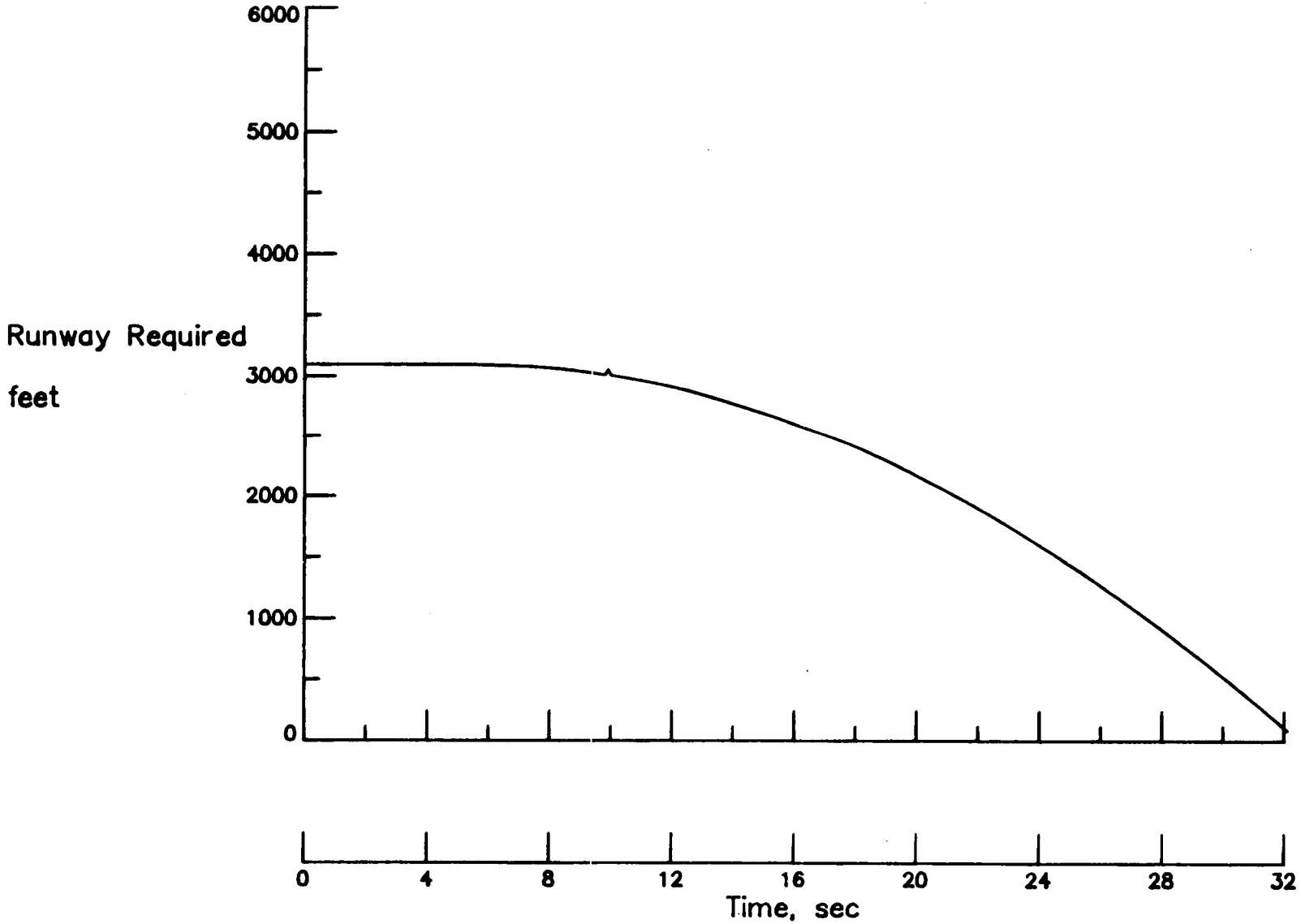

Figure 4.1d: Predicted Runway Required Time History for Case I With Perfect Sensors.

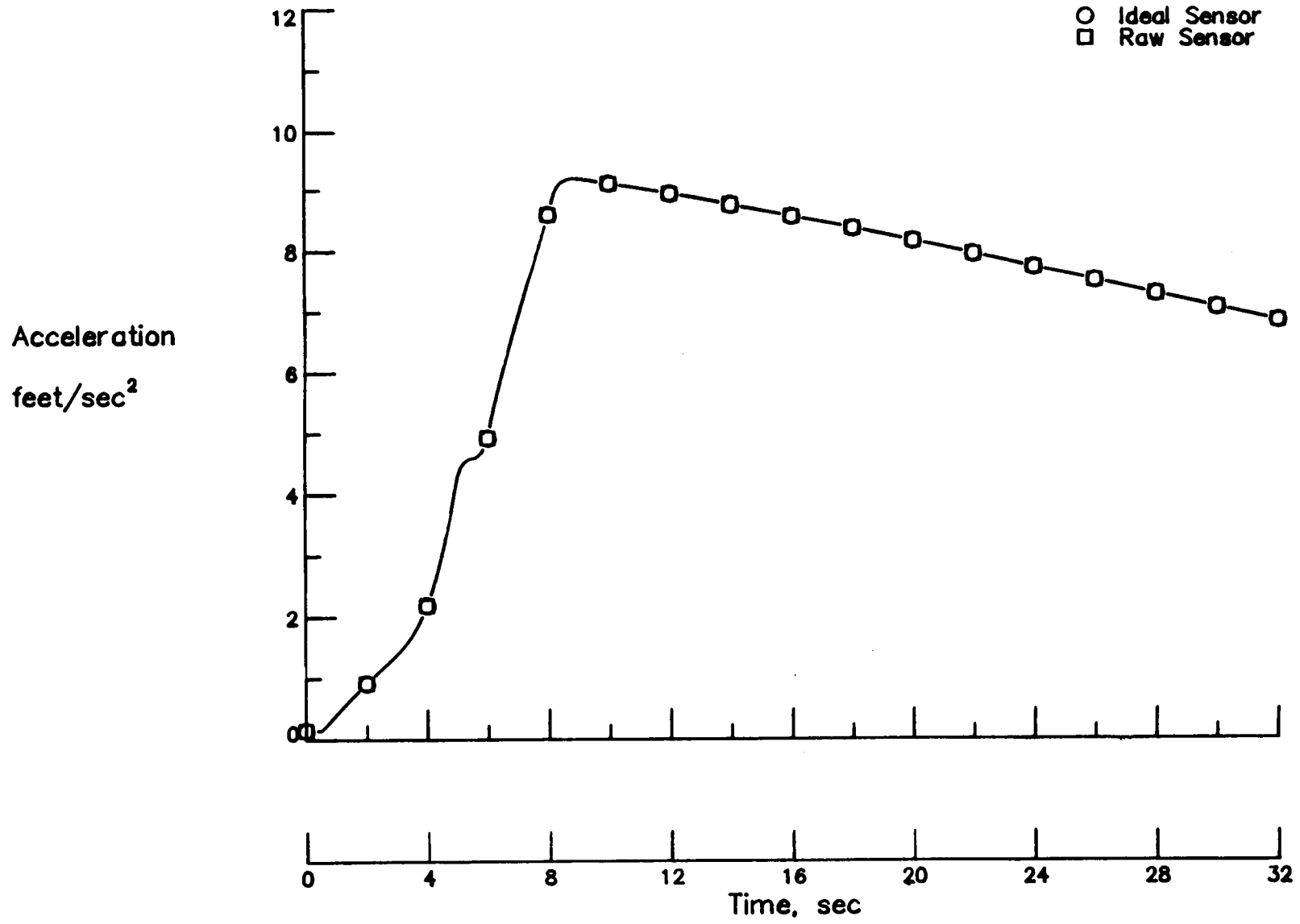

Figure 4.1e: Raw Sensor and Ideal Sensor Acceleration for Case I With Perfect Sensors.

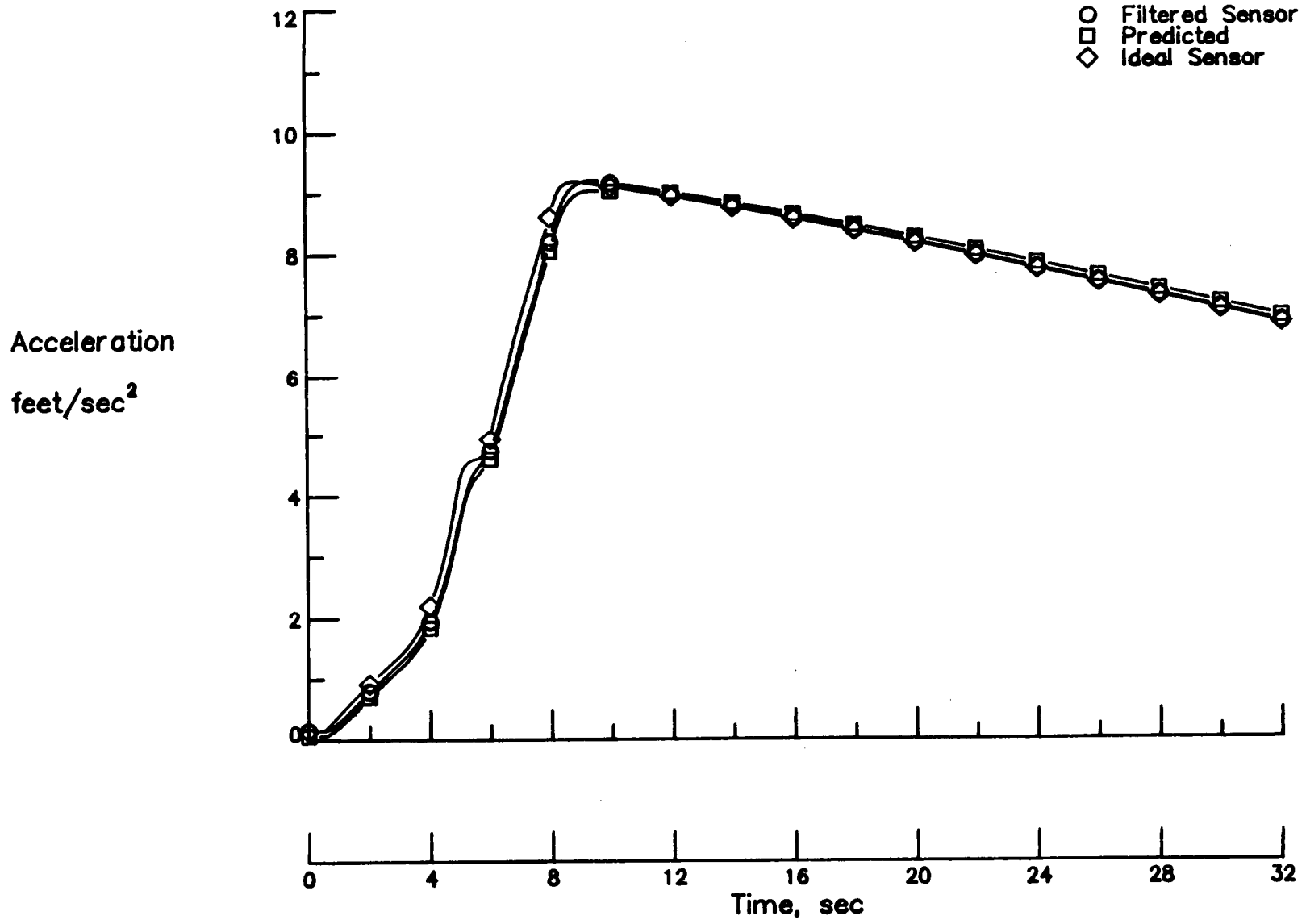

Figure 4.1f: Acceleration Time Histories for Case I With Perfect Sensors.

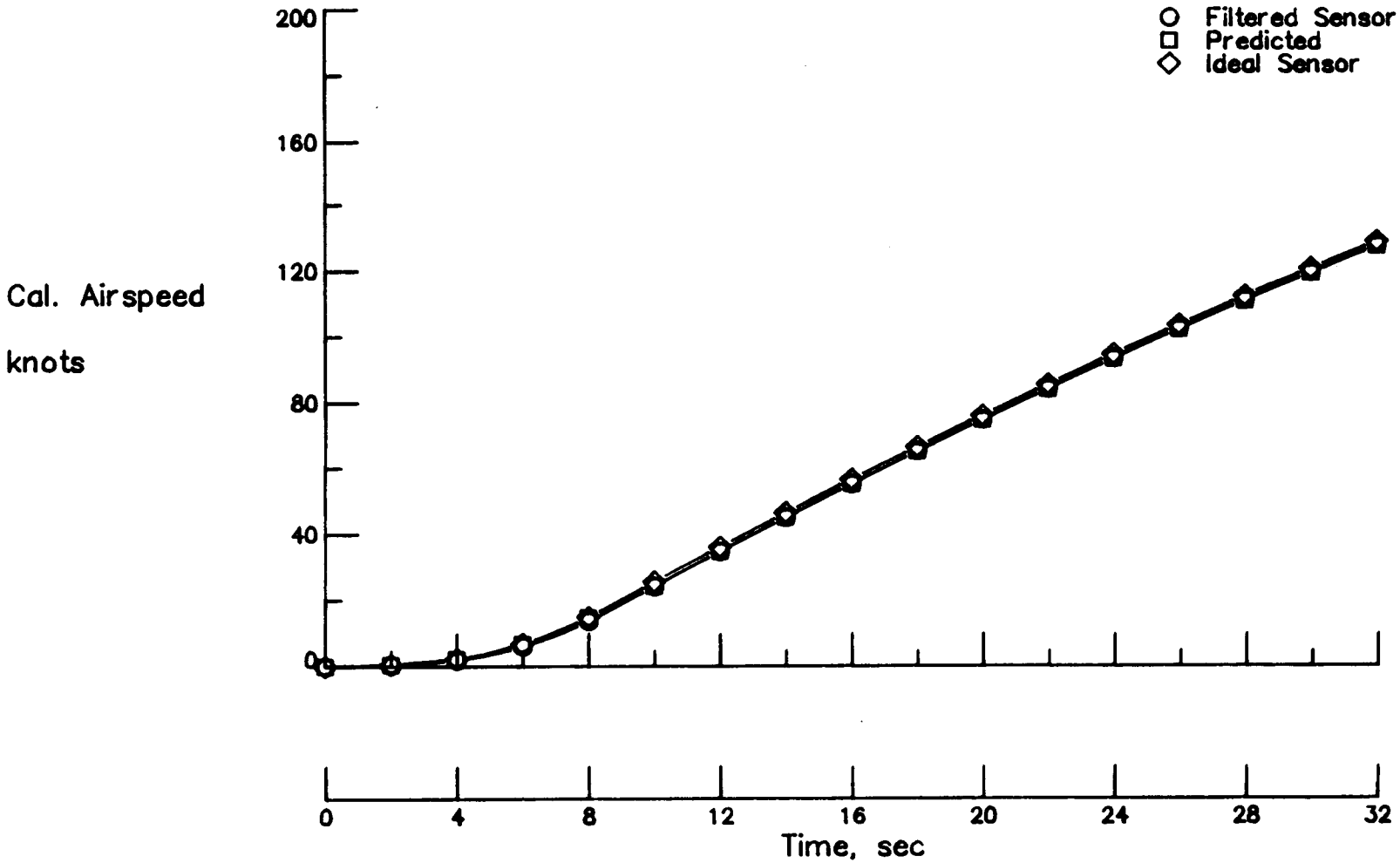

Figure 4.1g: Calibrated Airspeed Time Histories for Case I With Perfect Sensors.

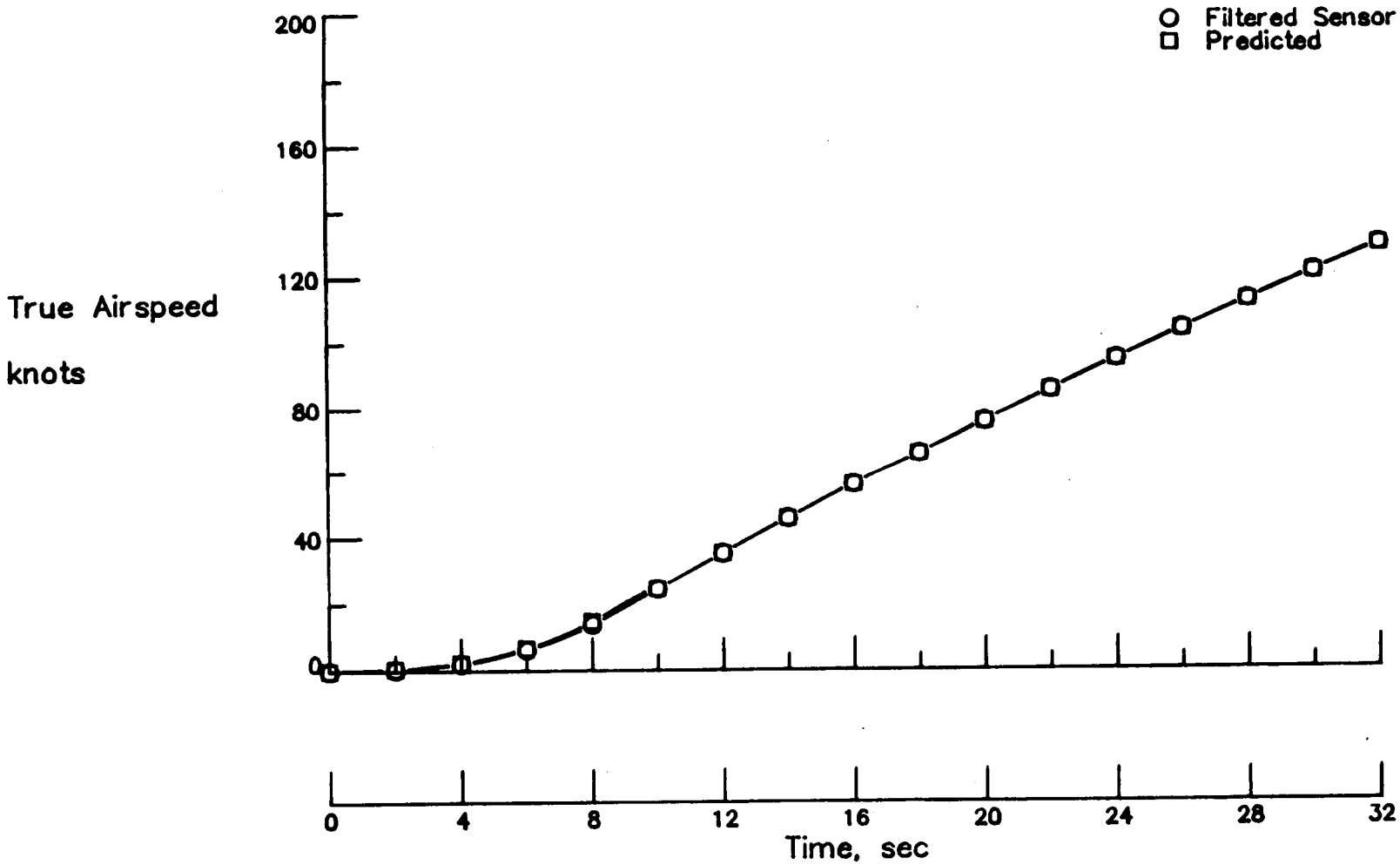

Figure 4.1h: True Airspeed Time Histories for Case I With Perfect Sensors.

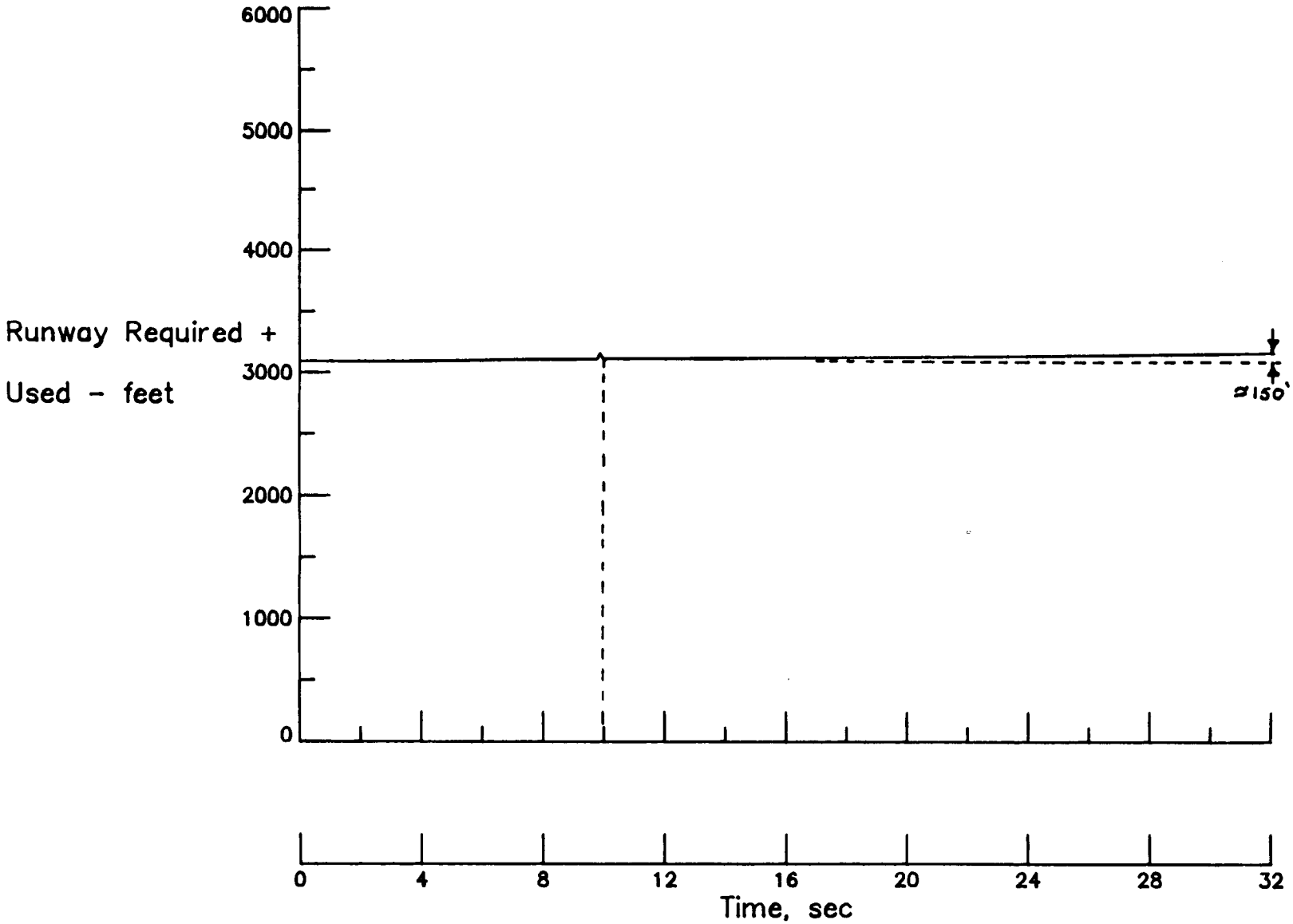

Figure 4.1i: Measure of Goodness of Prediction from Algorithm for Case I With Perfect Sensors.

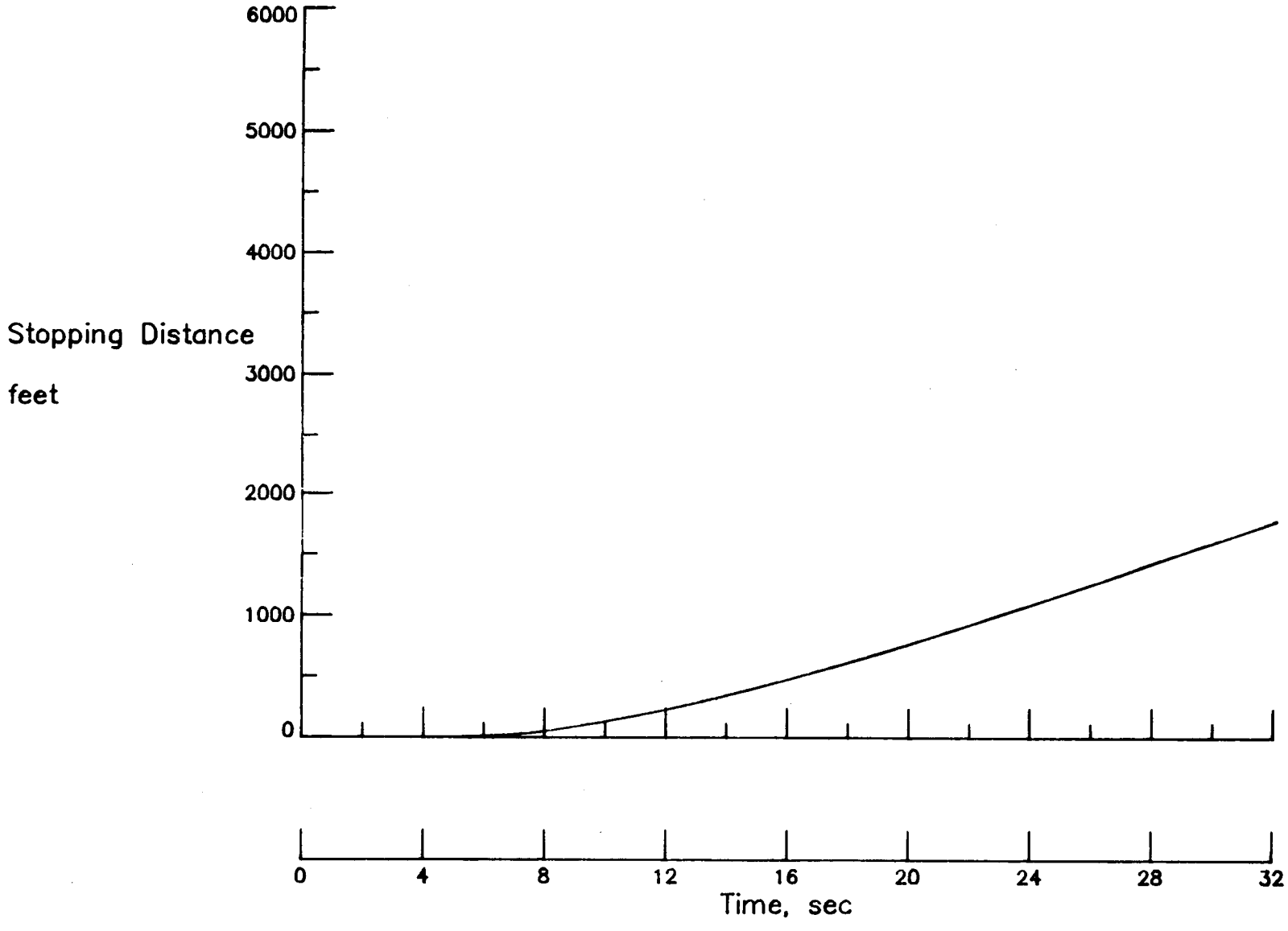

Figure 4.1j: Stopping Distance Time History for Case I With Perfect Sensors.

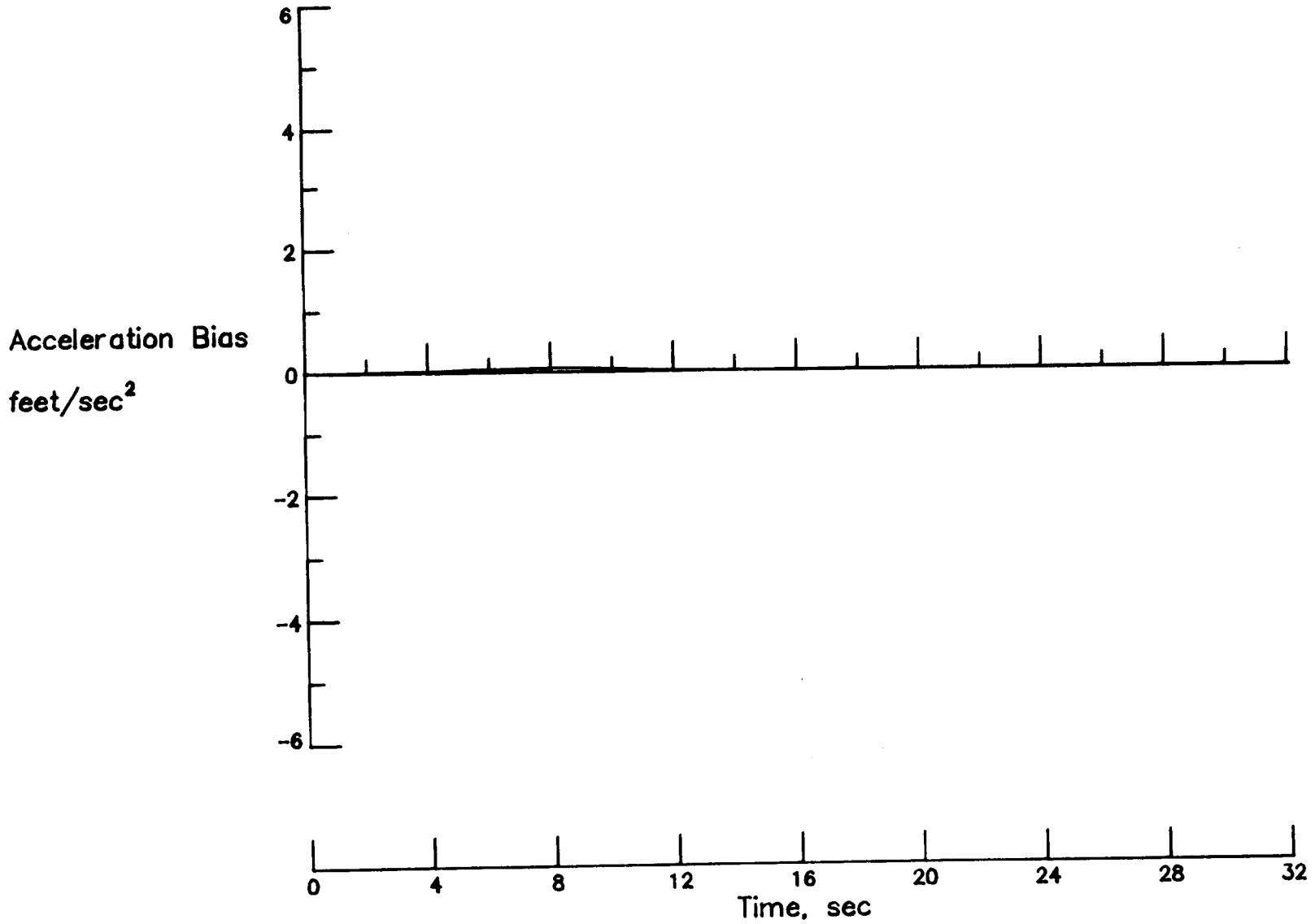

Figure 4.1k: Negative of Estimated Acceleration Bias for Case I With Perfect Sensors.

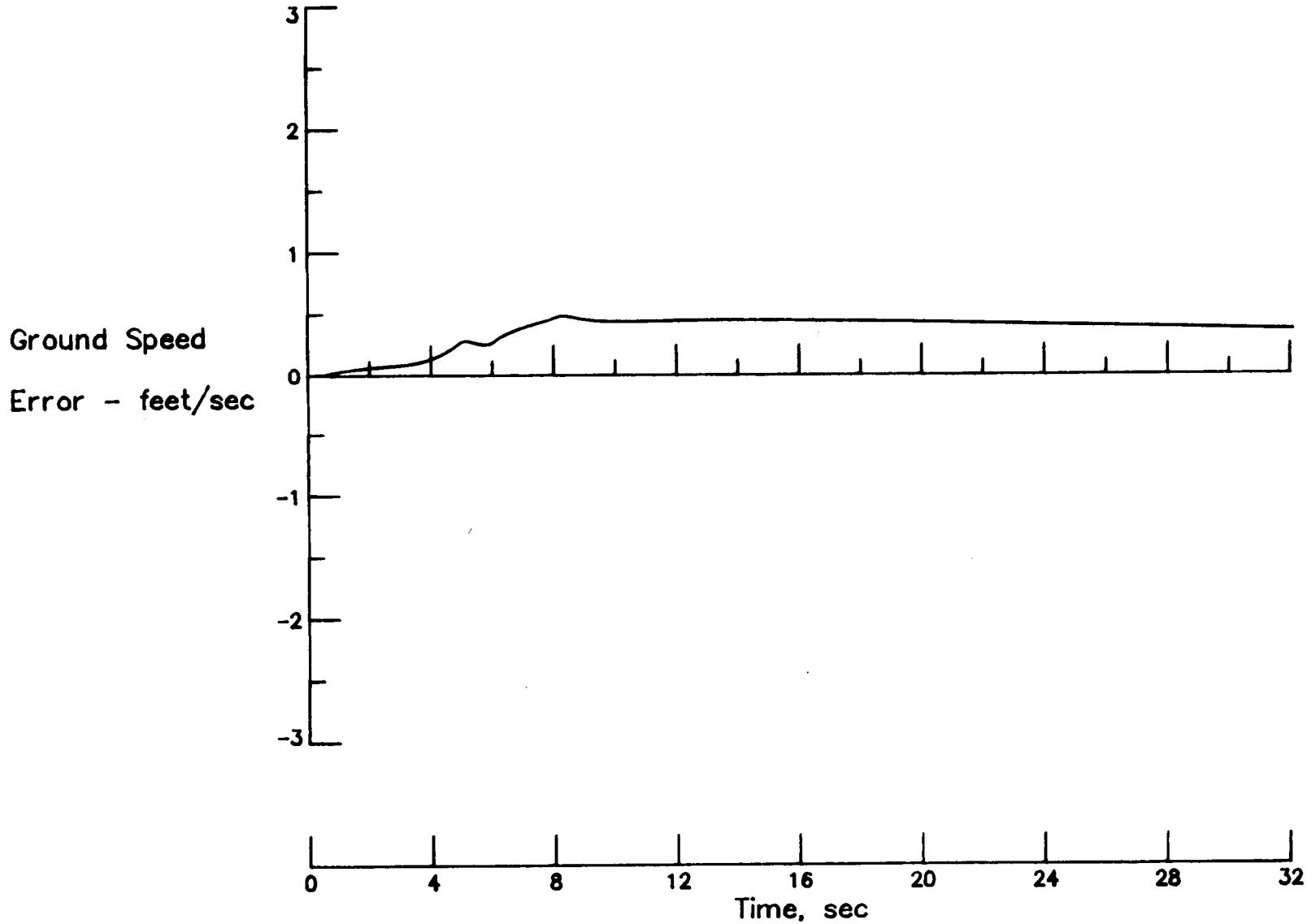

Figure 4.11: Estimated Error in Ground Speed for Case I With Perfect Sensors.

In Figure 4.1a "runway used" computed in the simulation model and the algorithm predicted "runway used" numbers are plotted against time. Towards the end of the takeoff run, the simulation model computed runway used is about a hundred feet more than the algorithm predicted runway used.

Figure 4.1b shows ground speeds from three sources plotted against time:(1) the sensor output from the simulation model after filtering; (2) the algorithm predicted value; and (3) the ideal sensor output from the simulation model. The filtered sensor plot and the ideal sensor plot are seen to be indistinguishable, whereas the predicted ground speed is seen to lag behind the other two values. This is caused by the first order lag filter that initializes the predicted ground speed to the measured value during the first pass through the algorithm.

Figure 4.1c illustrates the Engine Pressure Ratio (EPR) time histories. This figure shows that there is practically no difference between the measured and predicted engine pressure ratios after 10 seconds. Prior to this time the steady state engine model used in the algorithm is not expected to follow the output from the simulation which accounts for the transients caused by throttle movement. This is acceptable, since the algorithm does not perform an engine health check until after ten seconds into the takeoff run.

The algorithm predicted runway length required to achieve rotation speed is plotted against time in Figure 4.1d. At every instant the runway length requirement is predicted as a function of the filtered measured ground speed, the scheduled acceleration performance, and the ground speed required for rotation. For the present case, the initial runway requirement starts at about 3200 feet and goes down to zero as the airplane speed increases.

Figure 4.1e shows the difference between the ideal sensor value and the actual sensor value of acceleration. As this run assumes perfect sensors, the two curves are on top of each other. Plotted in Figure 4.1f are the filtered sensor output, predicted value, and the ideal acceleration. Even though no sensor noise is included in this run, the effect of the filtering can be clearly seen from the acceleration plots. Below about 10 seconds, the three values of acceleration are seen to be distinct. In this interval the ideal sensor output is the highest at any given instant, followed by the filtered sensor output and the predicted value. At the 10 second point the difference between the measured (filter output) and the predicted value is used to estimate a runway rolling friction coefficient. This causes the filtered value and the predicted value to be identical at that point. After the 10 second point, the filter dynamics does not significantly affect the measured acceleration in that the filtered and the ideal sensor value are nearly the same. The predicted value of acceleration is seen to be slightly on the high side after the ten second point. At about the rotation speed, the algorithm is seen to overpredict the acceleration by about one tenth of a foot / second^2 .

Filtered sensor output, predicted value, and an ideal sensor output of calibrated airspeed are plotted against time in Figure 4.1g. The effect of the first order lag filtering is seen to cause the filtered sensor output to be lower than the ideal sensor value. The predicted value of true airspeed is seen to be almost identical to the filtered sensor output.

The filtered sensor output and the predicted value of true airspeed, plotted in Figure 4.1h, are seen to be identical. The predicted true airspeed value plotted here is used in the aerodynamic computations in the algorithm, but not for any decision making or signal generation.

In Figure 4.1i, the sum of the algorithm predicted runway length requirement and the computed runway used is plotted against time. This sum is a measure of the goodness of the runway length requirement prediction by the algorithm. For an ideal runway length requirement prediction, the sum will be invariant with time. Locating the point on this curve corresponding to a time of 10 seconds and drawing a line parallel to the time axis through this point it is seen that at about the rotation time, the goodness line is above the horizontal line by about a hundred and fifty feet.

Figure 4.1j shows the stopping distance time history obtained from the algorithm.

The next two figures illustrate the estimates of errors obtained from the complementary filter. Figure 4.1k shows the negative of the estimated bias in the acceleration signal and Figure 4.1l shows the estimated error in the ground speed measurement. Because of the ideal sensor assumption in this run the filter estimates almost a zero bias in the acceleration (Figure 4.1k) and almost no error in the ground speed (Figure 4.1l).

Table 4.2 summarizes the results for run I. These results are obtained from the output listings that accompany the plots discussed above. The second column in this table represents the filtered value of the sensed calibrated airspeed. The third column shows the instantaneous value of the algorithm predicted runway required to achieve rotation speed. The positive value in this column shows that the algorithm computed calibrated airspeed has not yet reached the flight manual recommended rotation speed. The fourth column gives the runway requirement predicted by the algorithm immediately after the 10 second adjustment of runway friction coefficient. The fifth column shows the runway used to that instant as computed by the simulation model. The sixth column gives the error in prediction of the

runway length required to achieve rotation speed based on the prediction immediately after the friction coefficient adjustment (column 4) and the actual runway used (column 5). The negative number in this column indicates that the runway used was more than the algorithm's prediction and represents an under prediction by the algorithm. The last column (seventh) shows the updated runway rolling friction coefficient after the 10 second adjustment. The 0.010 value shown in this column represents a 33% change from the 0.015 used in the simulation. This large change is caused by the presence of the filter network in the absence of noise.

Table 4.2: Summary of results for Case I with Perfect Sensors

CASE NO.	MEASURED CAS at rotation (knots)	PREDICTED RUNWAY REQUIRED at rotation (feet)	PREDICTED RUNWAY REQUIRED (feet)	RUNWAY USED overall (feet)	RUNWAY PREDICTION ERROR (feet)	UPDATED FRICTION COEFF. (-)
I	128.0	21.	3119.	3150.	-31.	0.010

Note: The measured Calibrated Airspeed (CAS) is the filtered sensor output

The simulation run is stopped based on a measured calibrated airspeed. However, as the measured value reaches 128 knots (which is the flight manual recommended rotation speed for this case) the actual calibrated airspeed obtained from the simulation model is 129.1 knots. Backtracking through the

C-2

output listing, the simulation model is found to have reached a calibrated airspeed of 128.3 knots at 31.9 seconds into the takeoff run. The runway used (computed by the simulation model) at this point is 3106 feet. Thus the algorithm is seen to have over predicted the runway required by 13 feet. The reason that the algorithm predicts a residual runway required figure at the end is the lag introduced by the filters in the measured ground speed and acceleration.

Figures 4.2a- 4.2l depict the results from the simulation run for case I in the presence of realistic sensor noises. It is seen from Figure 4.2a that the filtered sensor data and the predicted data for ground speed are closer together than in the ideal sensor run (Figure 4.1a). The modification of the sensed data by the filter in the absence of noise causes the difference in Figure 4.1a. The same effect is also observed in the ground speed plots of Figure 4.2b. The EPR curves of Figure 4.2c differ from Figure 4.1c only in the sensor noise effect. The runway length required to achieve rotation speed is seen to go negative in Figure 4.2d. This is the result of the simulation model taking more runway to achieve rotation speed than that predicted by the algorithm. Figure 4.2e shows a comparison of the ideal sensor output and the unfiltered noisy sensor output of the acceleration. Figure 4.2f shows better agreement among the filtered sensor output, the predicted value, and the ideal sensor values of acceleration. Residual effects of the sensor noise, after the filter, are as seen in this figure. Figure 4.2g shows that the filtered sensor output of calibrated airspeed does not lag behind the ideal sensor output as much as in Figure 4.1g. The residual effects of the sensor noise, after filtering, are discernable in this figure also. Figure 4.2h, when compared

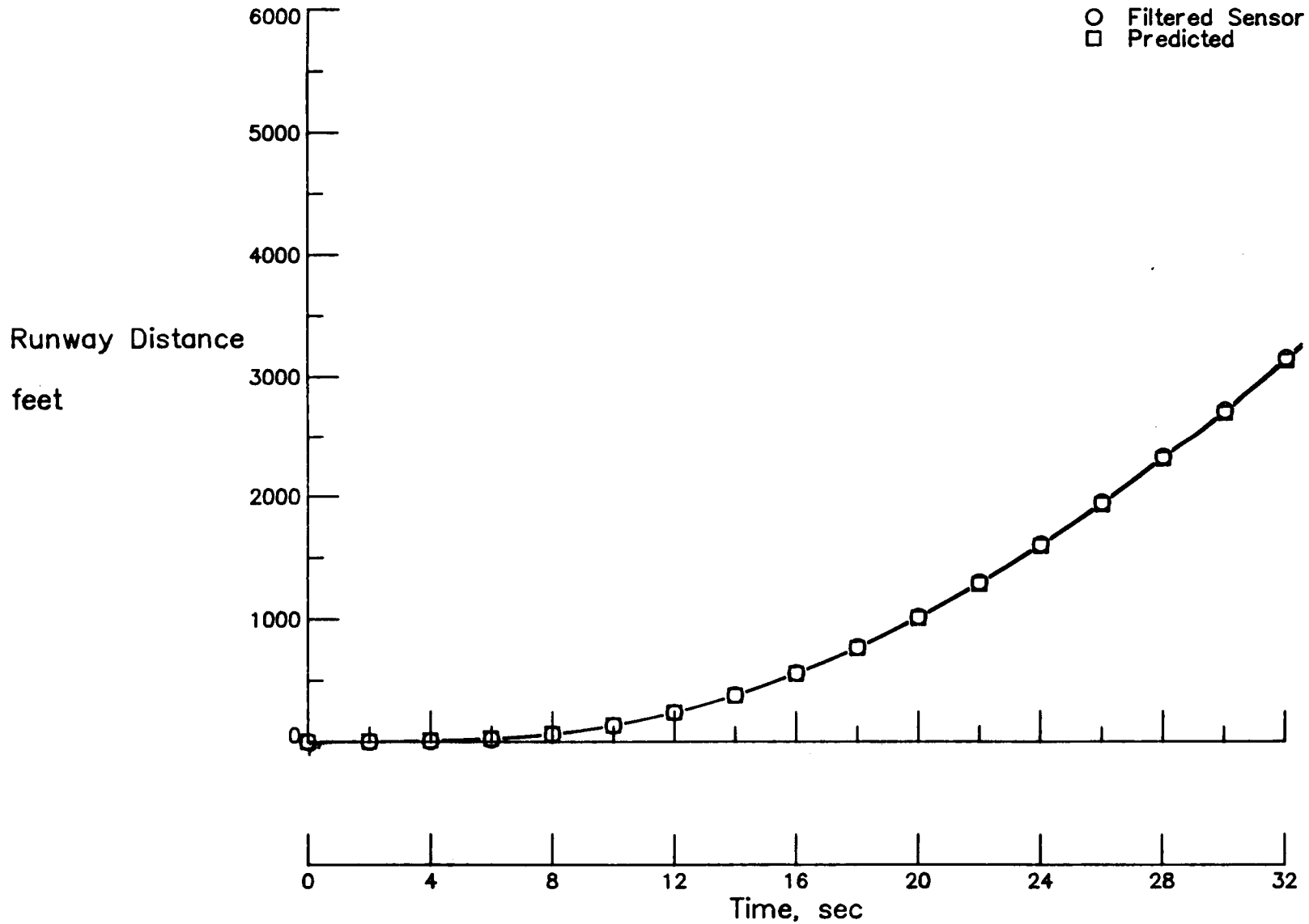

Figure 4.2a: Runway Used Time Histories for Case I.

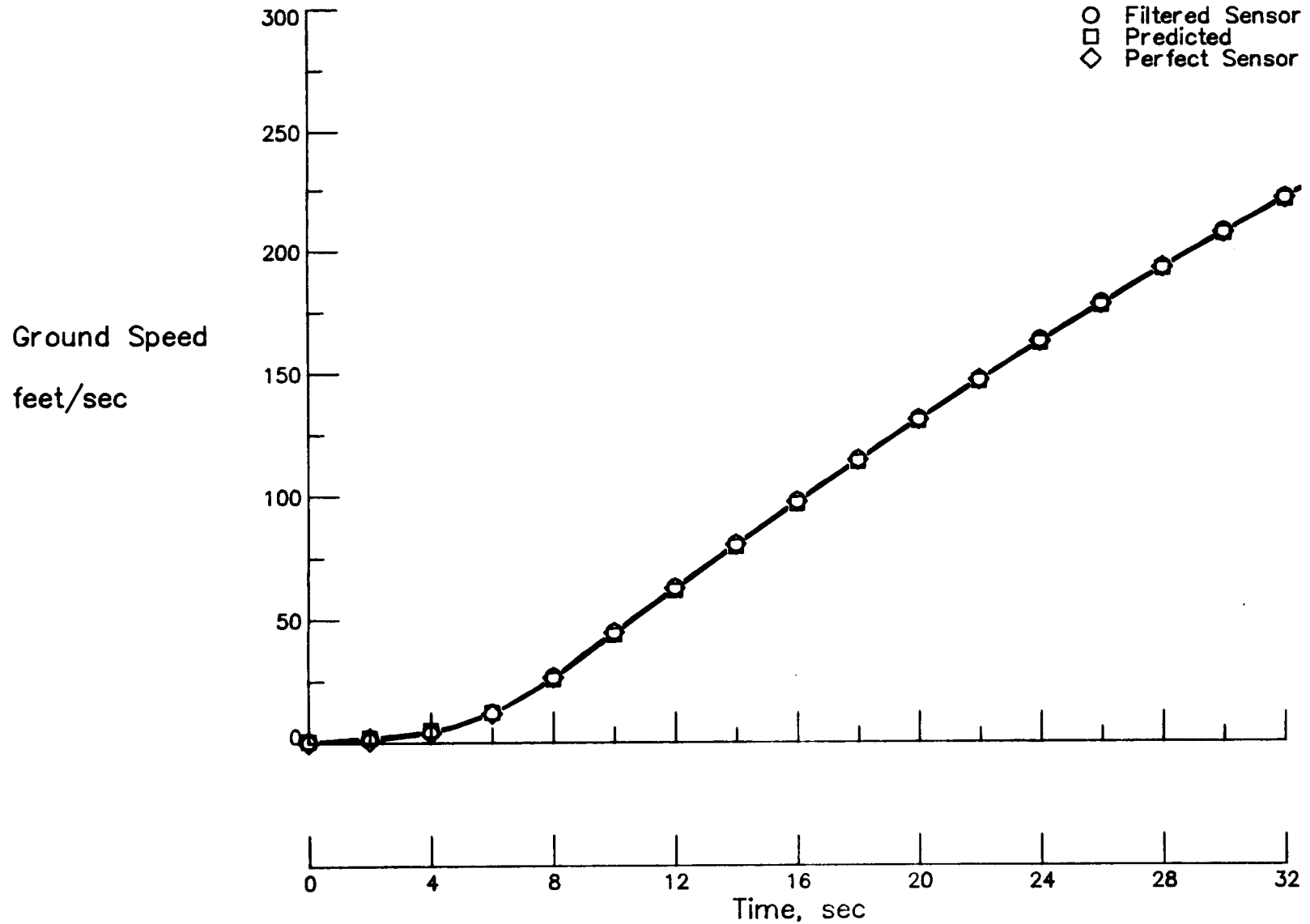

Figure 4.2b: Ground Speed Time Histories for Case I.

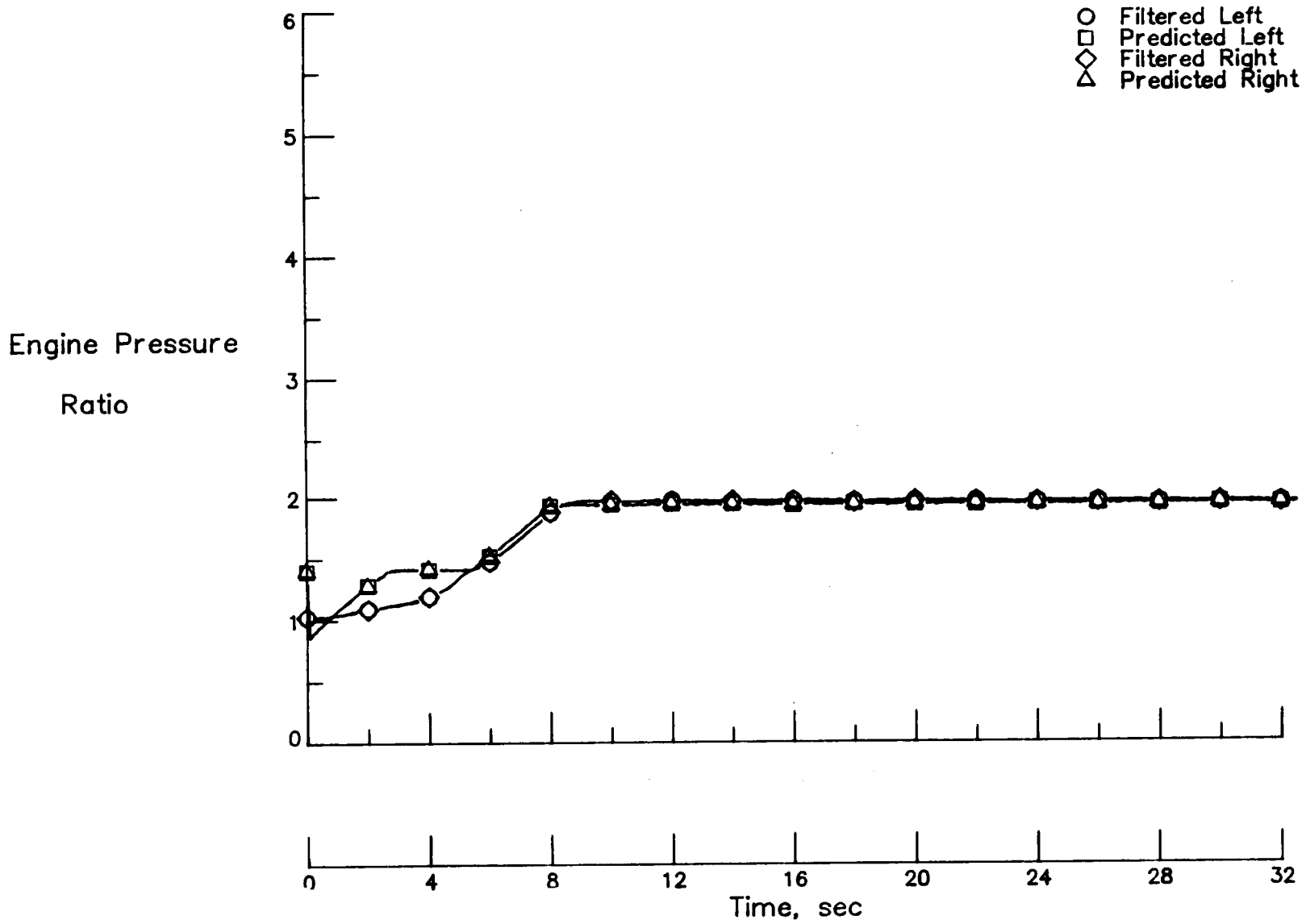

Figure 4.2c: Engine Pressure Ratio Time Histories for Case I.

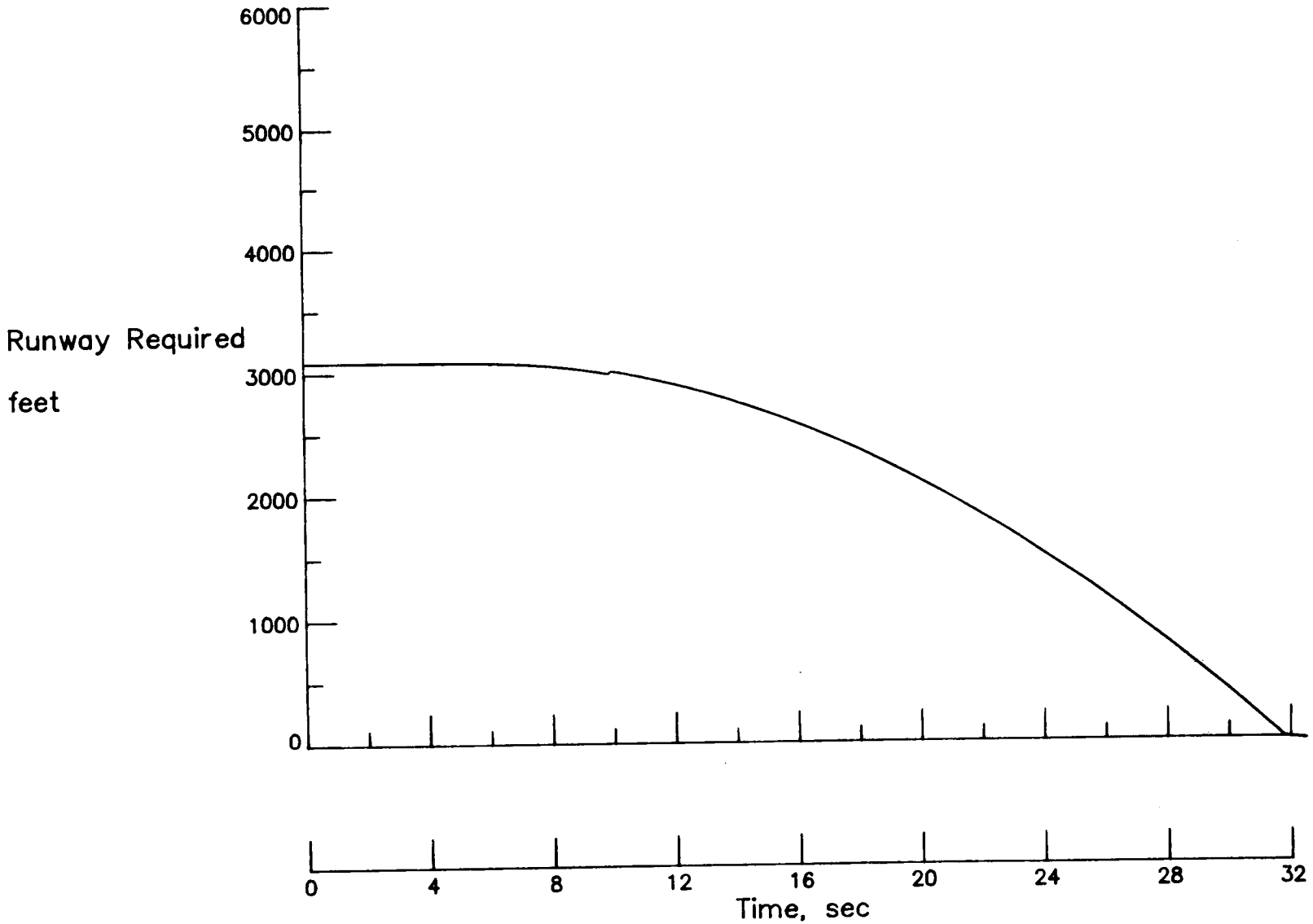

Figure 4.2d: Predicted Runway Required Time History for Case I.

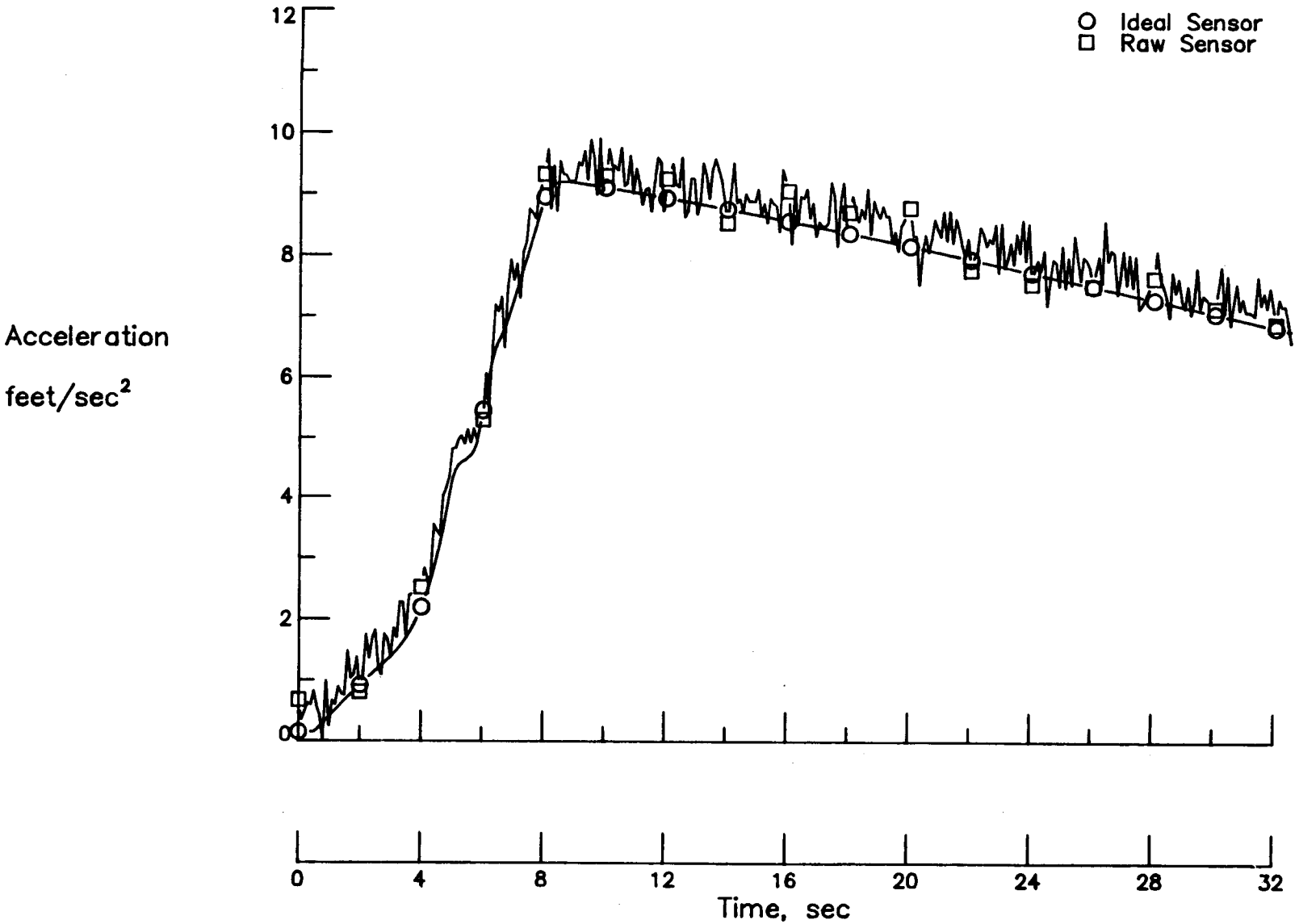

Figure 4.2e: Raw Sensor and Ideal Sensor Acceleration for Case I.

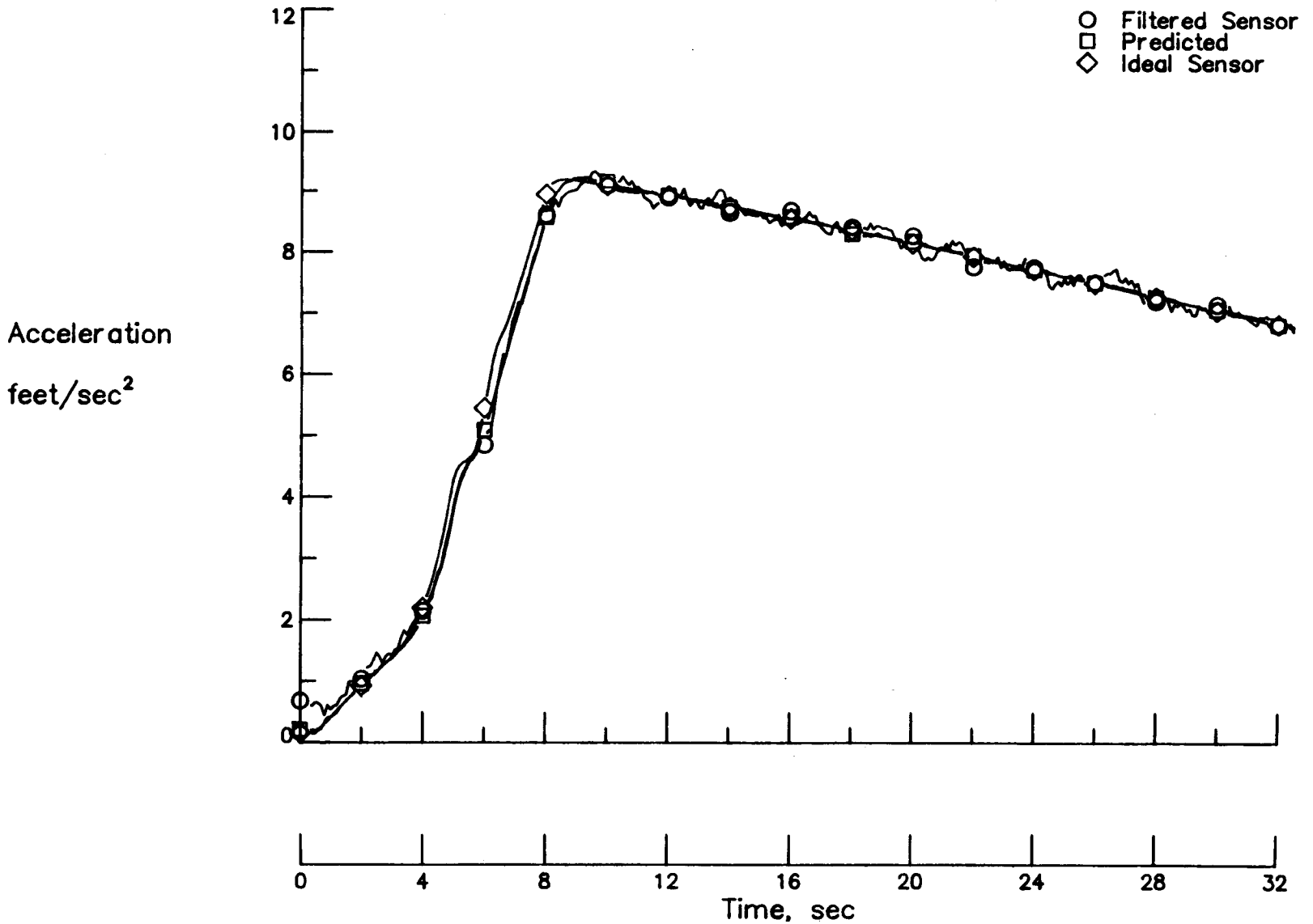

Figure 4.2f: Acceleration Time Histories for Case I.

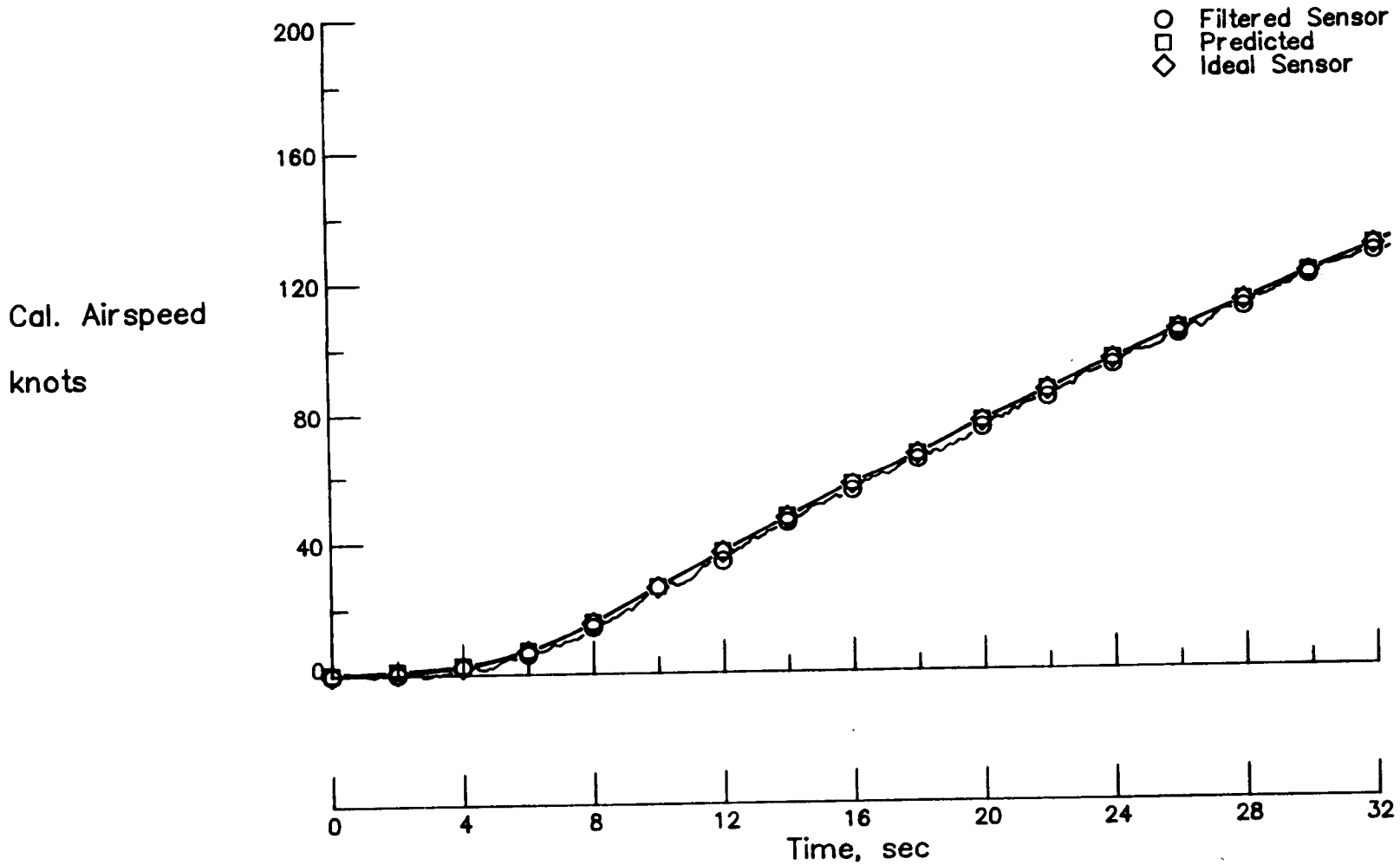

Figure 4.2g: Calibrated Airspeed Time Histories for Case I.

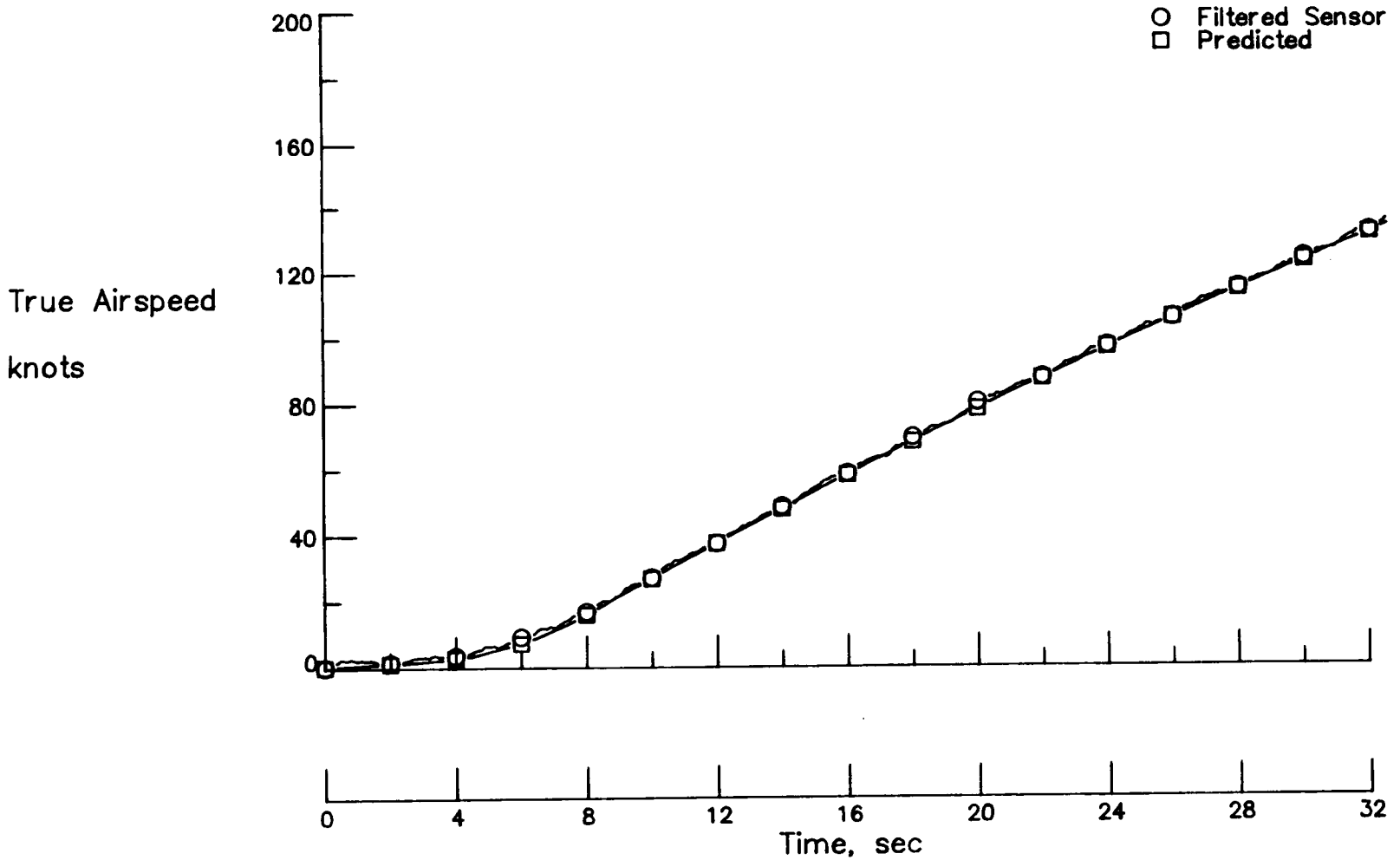

Figure 4.2h: True Airspeed Time Histories for Case I.

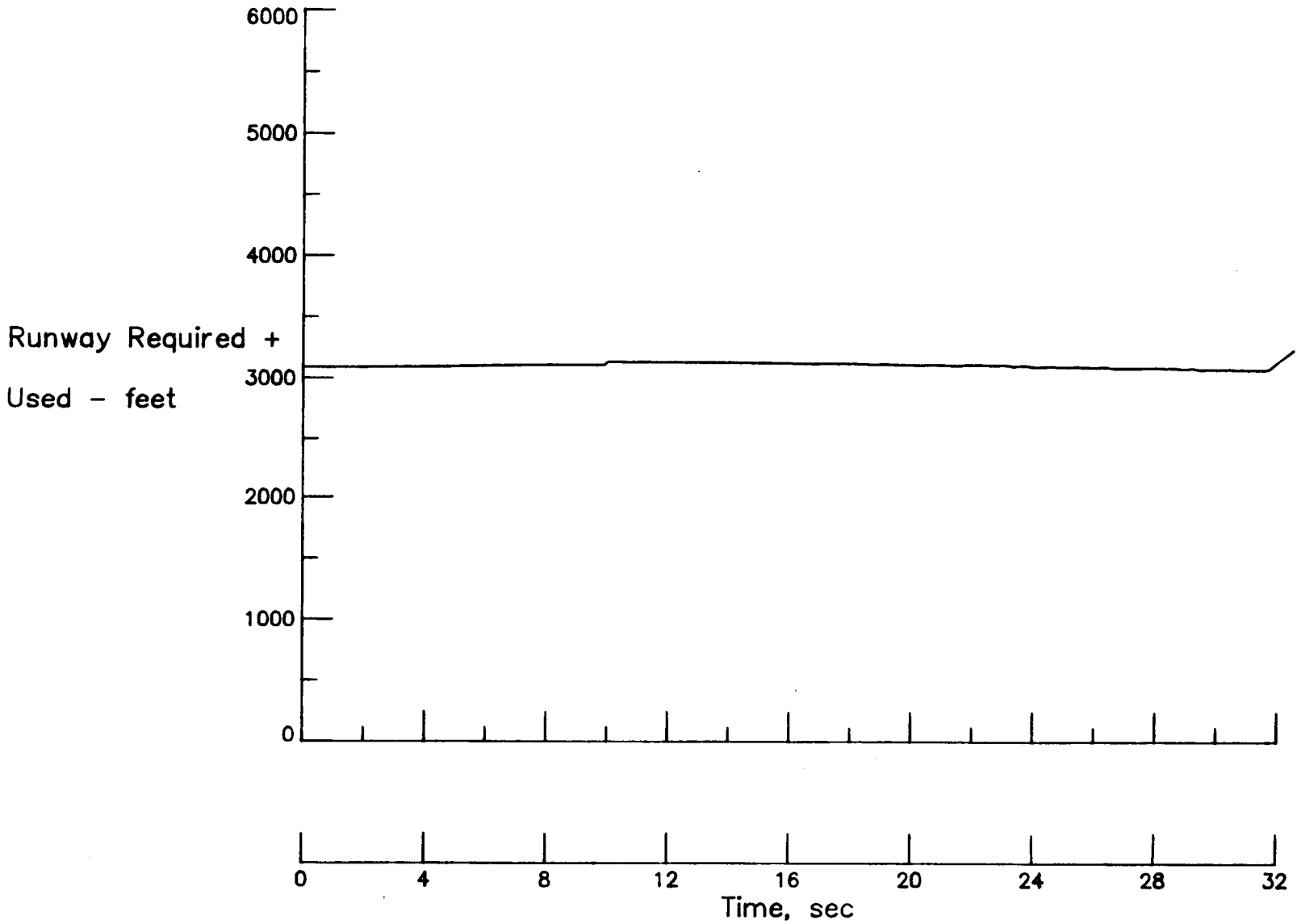

Figure 4.2i: Measure of Goodness of Prediction from Algorithm for Case I.

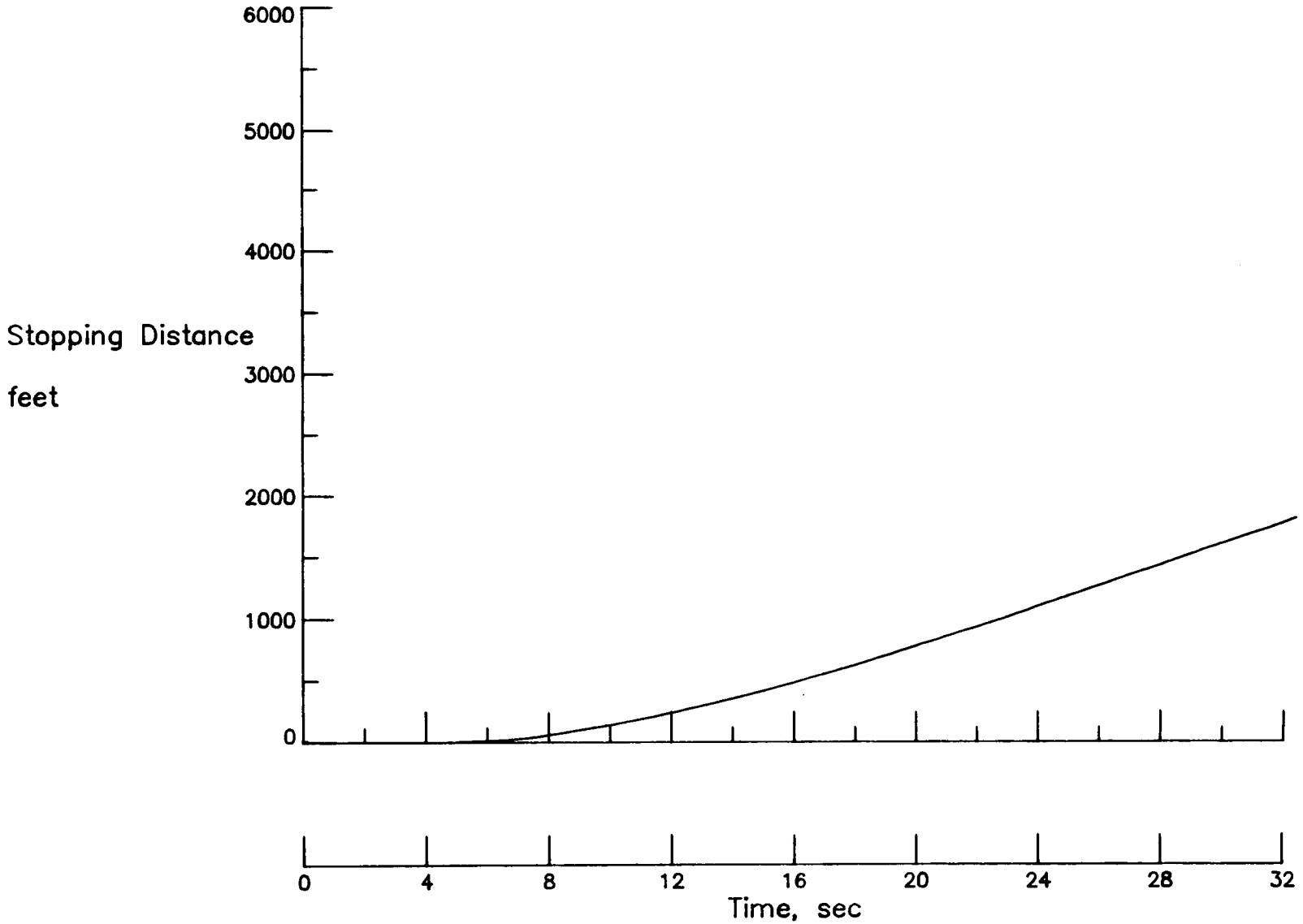

Figure 4.2j: Stopping Distance Time History for Case I.

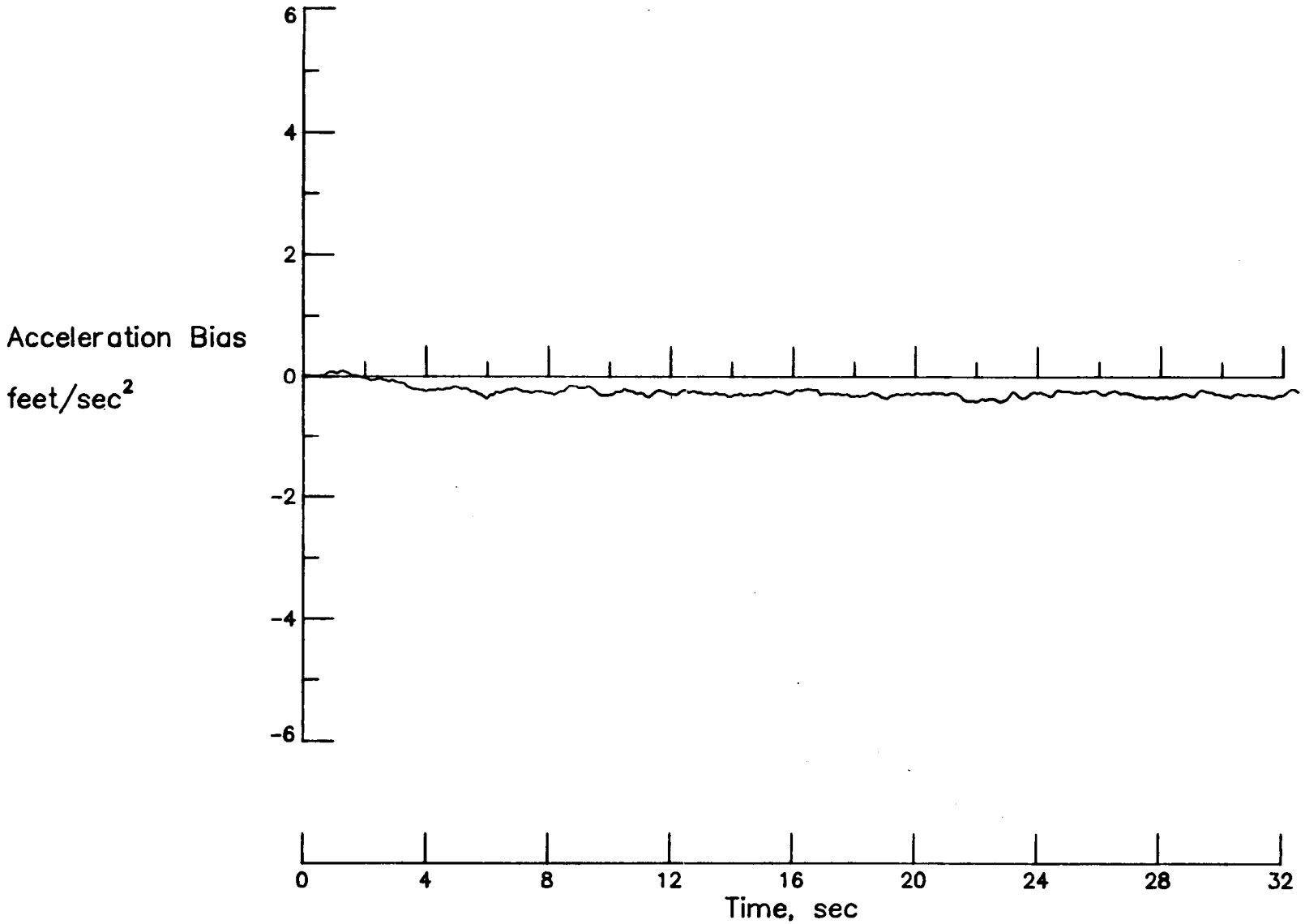

Figure 4.2k: Negative of Estimated Acceleration Bias for Case I.

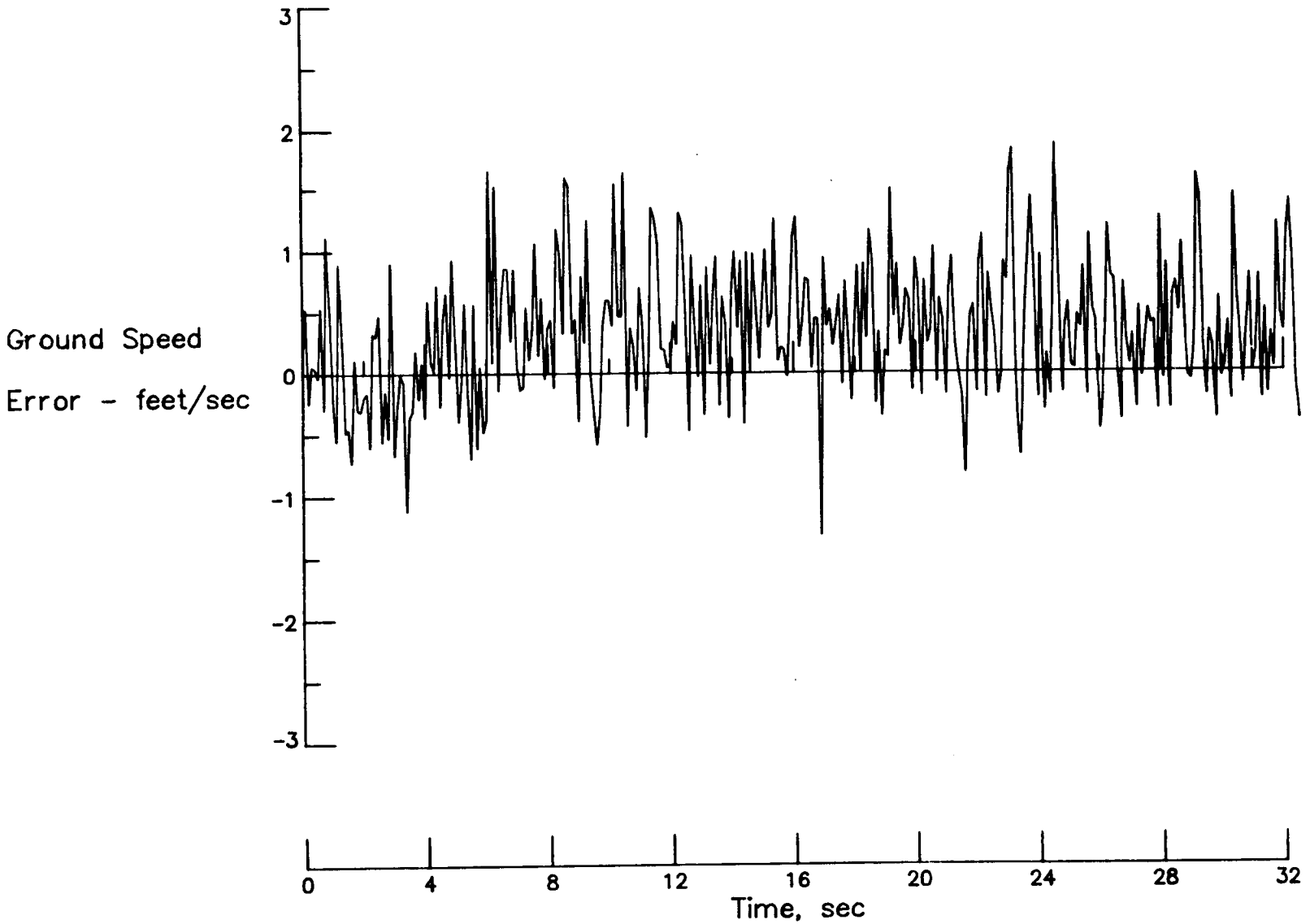

Figure 4.21: Estimated Error in Ground Speed for Case I.

to Figure 4.1h shows the effect of sensor noise. Figure 4.2i shows the sum of the predicted runway length required and the runway used against time. This curve is seen to remain practically horizontal until very near the end, and then slopes upward. This upward sloping is an indication that the model needed more runway to achieve rotation speed than the algorithm had predicted. Figure 4.2j shows a time history of the required runway length to stop the airplane, as predicted by the algorithm. It is seen from Figure 4.2k that the complementary filter estimated a bias error of about .3 feet per second square, which is close to the sensor bias introduced. The effects of the Gaussian noise on the the ground speed, as estimated by the complementary filter is depicted in Figure 4.2l.

Results obtained from the simulation outputs for all ten cases, with sensor noises are summarized in Table 4.3. Looking at the column titled runway prediction error in the above table, the largest absolute error is seen to be 142 feet (case II). For that case, the table indicates that the simulation model needed more runway than was predicted by the algorithm, to achieve rotation speed. As was indicated for the ideal sensor simulation for the conditions of case I, the filter has the effect of introducing a time lag on the measured calibrated airspeed. For the worst error condition in Table 4.3, the actual calibrated airspeed, when the measured (filtered) value reaches 128.3 knots, is 131.3 knots. Again looking through the output for the instant the actual calibrated airspeed exceeds 128 knots, it is seen that the simulation model needed 3088 feet. The algorithm is seen to over predict the runway requirement, that is the algorithm predicted runway requirement for attaining rotation speed is more than what is actually needed. These figures are summarized in Table 4.4.

Table 4.3: Summary of results for Table 4.1 Cases with Noisy Sensors

CASE NO.	MEASURED CAS at rotation (knots)	PREDICTED RUNWAY REQUIRED at rotation (feet)	PREDICTED RUNWAY REQUIRED (feet)	RUNWAY USED overall (feet)	RUNWAY PREDICTION ERROR (feet)	UPDATED FRICTION COEFF. (-)
I	128.1	-21.	3132.	3262.	-130.	0.017
II	128.3	-21.	3124.	3266	-142.	0.017
III	128.6	-19.	3149.	3277.	-128.	0.017
IV	129.1	-12.	2682.	2740.	-58.	0.017
V	129.4	-6.	3625.	3603.	+22.	0.018
VI	129.1	-5.	2690.	2685.	+5.	0.017
VII	128.8	-5.	2289.	2300.	-11.	0.018
VIII	128.2	-8.	3246.	3272.	-26.	0.027
IX	128.1	-10.	3045.	3085.	-40.	0.009
X	138.1	-11.	4142.	4155.	-13.	0.017

Note: The measured Calibrated Airspeed (CAS) is the filtered sensor output

It should be pointed out here that any system implemented on an airplane can only perceive the measured quantity indicated above, and not the actual value calculated in the simulation. This leads to the conclusion that the errors caused by using the measured parameters are unavoidable. Looking at

the error quantity, it is seen that the magnitude is well within 5% of the runway used and hence should be acceptable.

Table 4.4: The Effects of Actual versus Measured Calibrated Airspeed on case II

CASE NO.	MEASURED CAS at ROTATION (knots)	ACTUAL CAS at ROTATION (knots)	PREDICTED RUNWAY REQUIRED (feet)	SIMULATION RUNWAY USED (feet)	ERROR in PREDICTION (feet)
II	128.3	131.3	3124.	3266.	-142.
		128.1	3124.	3088	+36.

4.4 Algorithm Detection of Engine Malfunctions

The ability of the algorithm to detect engine malfunctions and signal deficiencies in the performance of the airplane are illustrated in this section. Two types of engine malfunctions are simulated for this illustration, as listed below:

1) Engine does not develop hand book EPRs

2) Engine does not develop hand book thrust

The section 4.4.1 discusses the first malfunction, and section 4.4.2 deals with the second malfunction.

4.4.1 Engine EPR Failure

The simulation of an engine not developing hand book EPR values can not be simulated directly without affecting the overall simulation run because of the implementation of the engine subroutine. This effect is simulated as follows:

i Compute the correct EPRs.

ii Reduce the EPR by a constant factor.

iii Compute the compressor and turbine stage speeds, engine thrust, and the Exhaust Gas Temperature (EGT) based on the reduced EPRs.

iv Restore the EPRs to their original values for use during the next iteration.

Two types of engine EPR failure are simulated. One case where the EPR developed is 85% of the nominal value and the other case where the EPR developed is 115% of nominal. The first case is directly seen to be a degraded performance case. The second case is also treated as such here because it represents a deviation from nominal and could be interpreted as an indication of impending failure.

Figure 4.3a-4.3e show the results for the run with the EPR reduced to 85% of nominal and the algorithm operating in the signal generation mode. From Figure 4.3a, it is seen that the predicted steady state EPR is higher than than the measured value throughout the run. Comparing this figure to Figure 4.2c , it is seen that the second step climb in EPR is missing. Since the measured EPR value does not reach the recommended intermediate

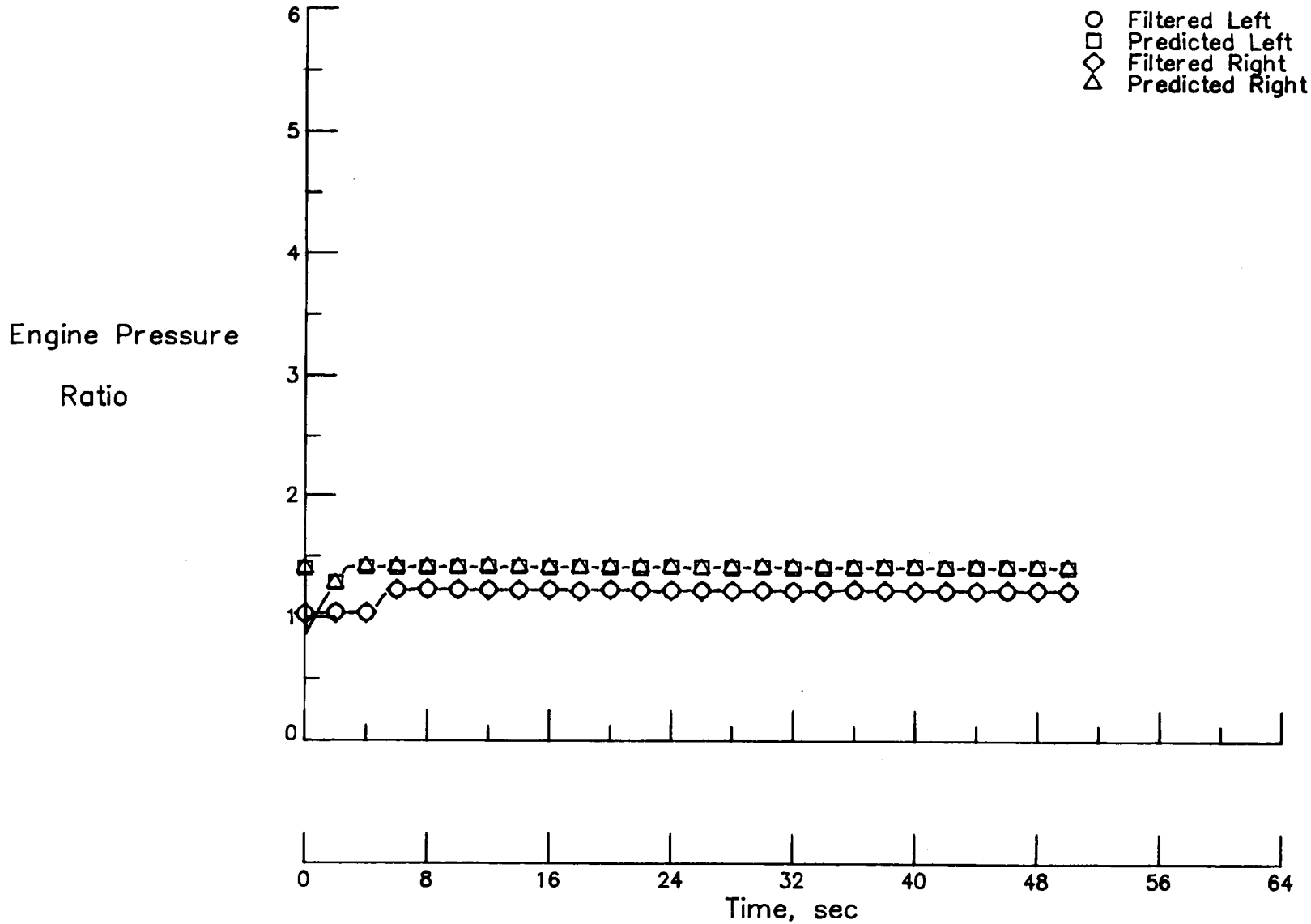

Figure 4.3a: Engine Pressure Ratio Time Histories for the 85% EPR Case.

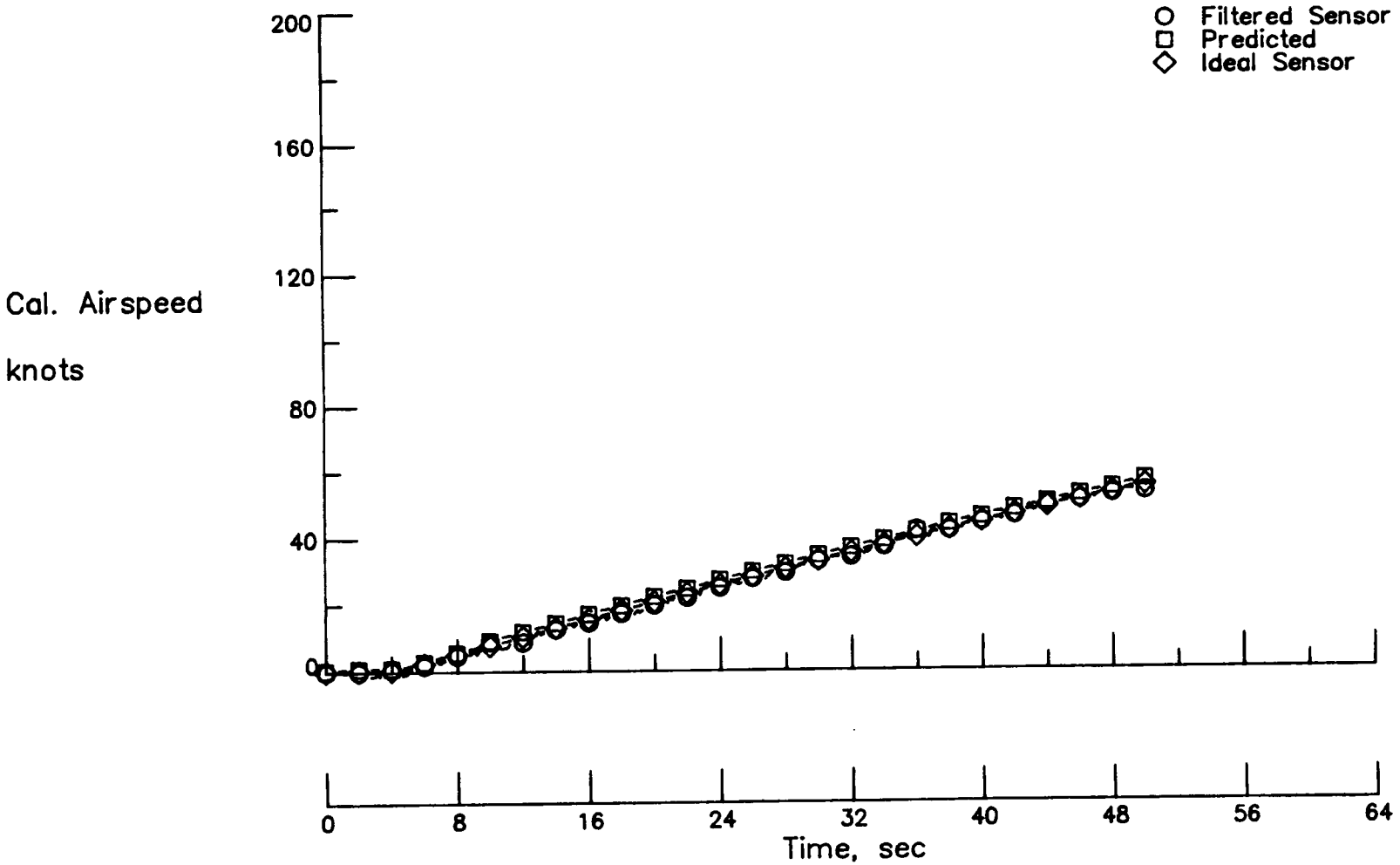

Figure 4.3b: Calibrated Airspeed Time Histories for the 85% EPR Case.

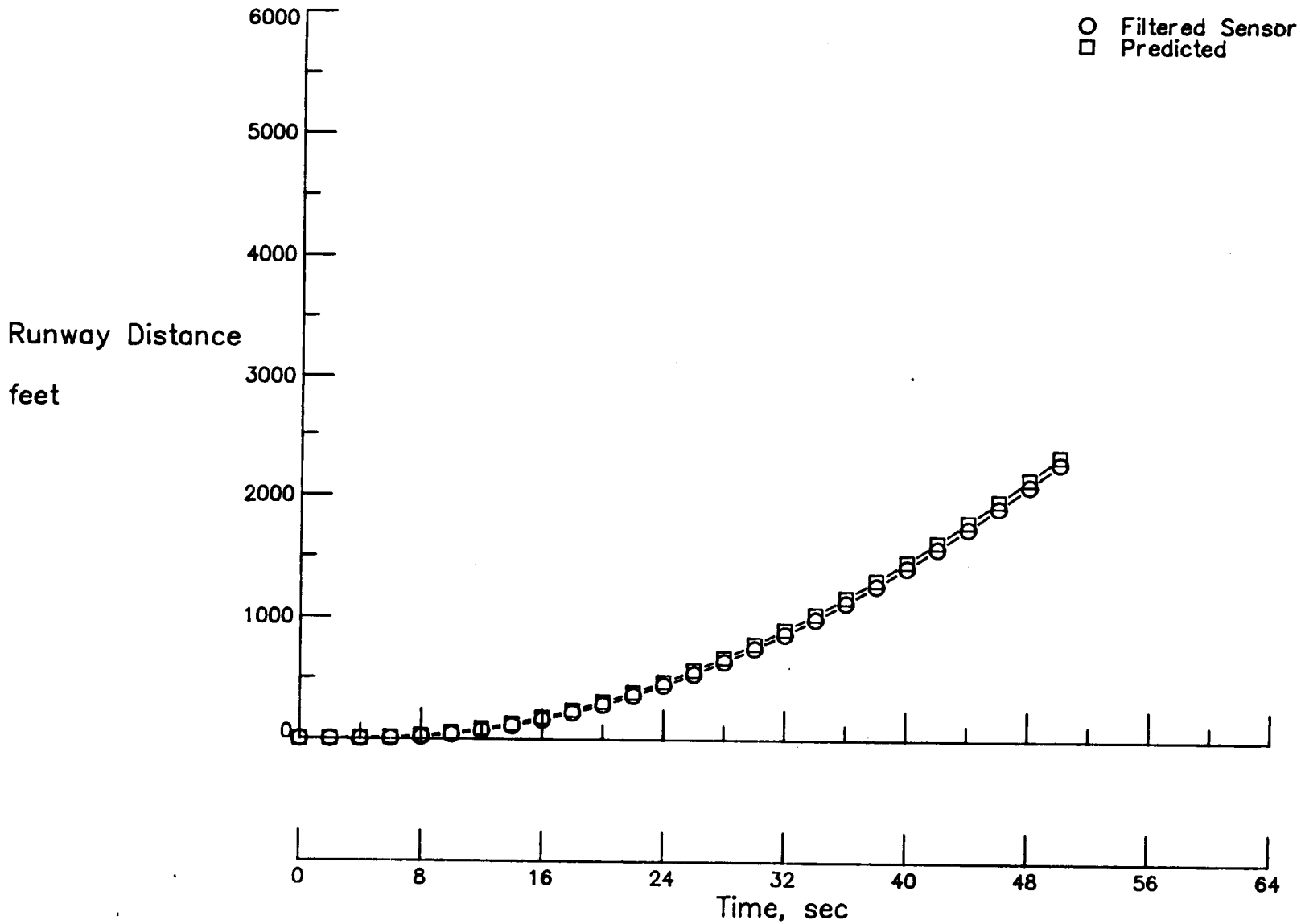

Figure 4.3c: Runway Used Time Histories for the 85% EPR Case.

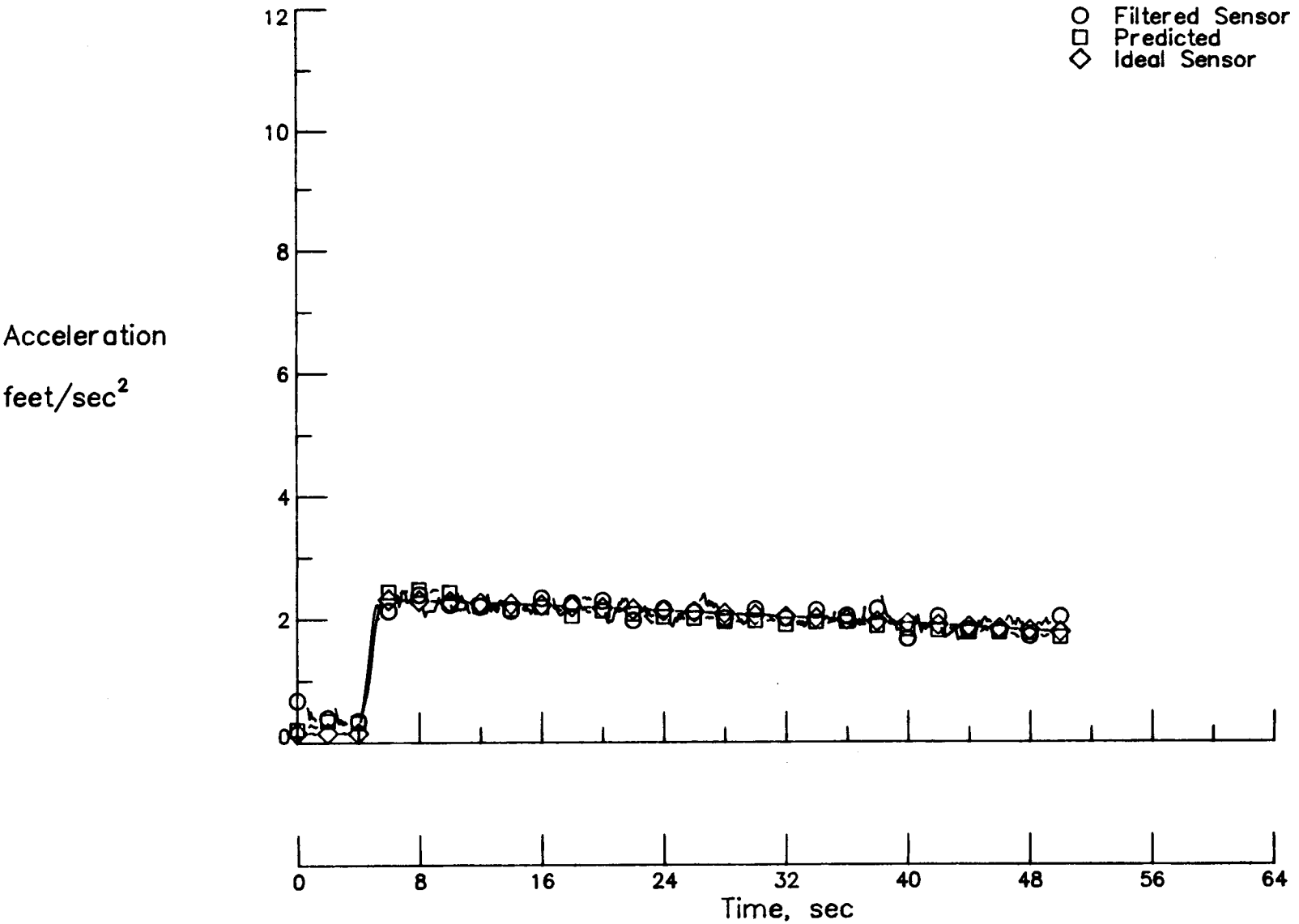

Figure 4.3d: Acceleration Time Histories for the 85% EPR Case.

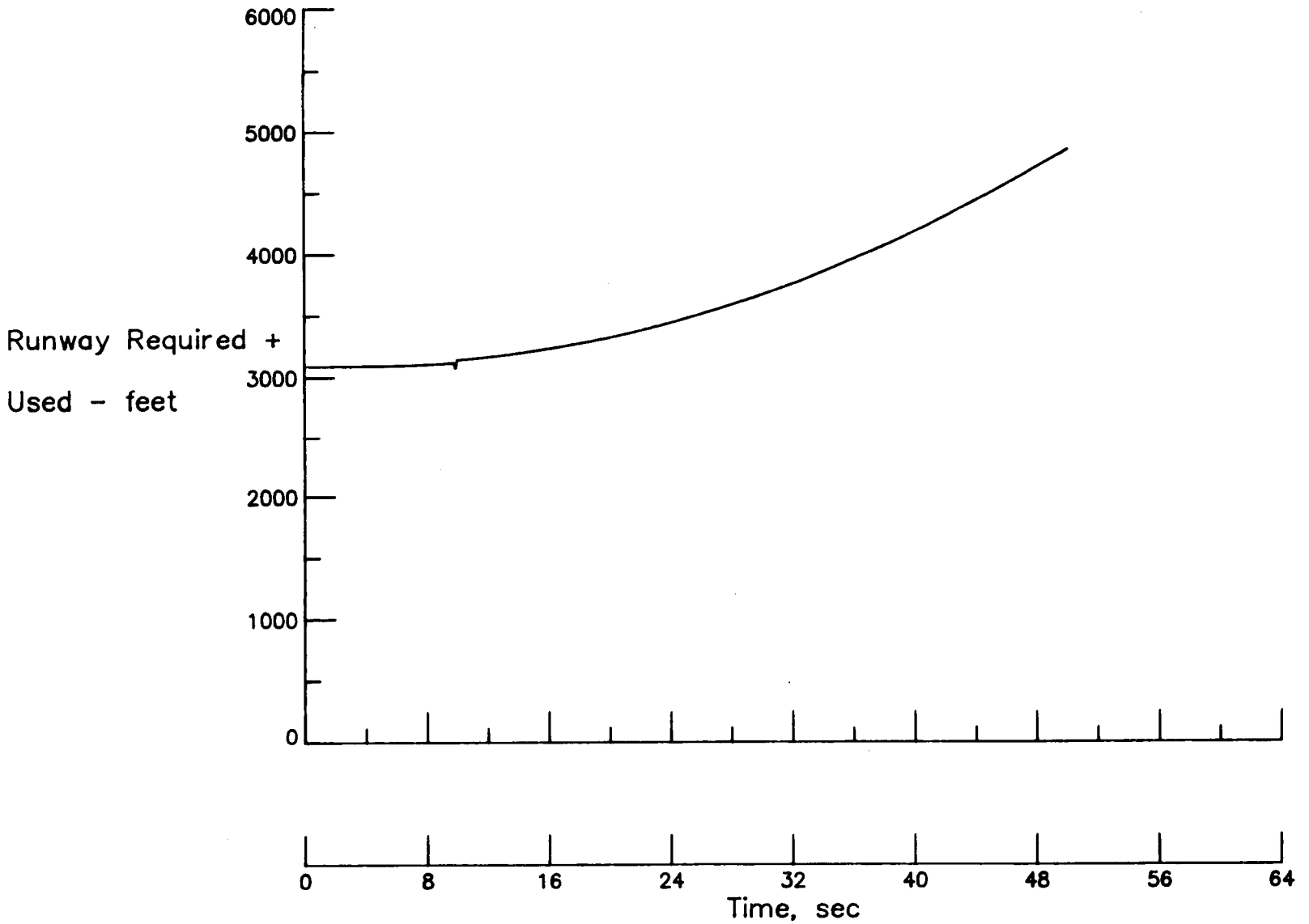

Figure 4.3e: Goodness of Runway Requirement Prediction for the 85% EPR Case.

setting, the throttle is not commanded to the takeoff setting. The calibrated airspeed plot of Figure 4.3b shows a very slow increase to about 45 knots in 50 seconds. Figure 4.3c shows the runway used time history and shows that the airplane has used up about 2500 feet of runway in 50 seconds. Figure 4.3d shows that the measured (and filtered), predicted, and the perfect sensor accelerations are practically identical. Figure 4.3e shows the sum of runway-used-plus-required against time.

The output listings from the simulation run indicate that the algorithm caused the engine failure flags to be set immediately following the 10 second point, and caused an abort signal to be generated. As the abort command sequence was turned off during this run, the simulation model continued its takeoff roll. Within the next few iterations, the performance failure flag was also set (based on the difference between the measured (filtered) and the predicted accelerations). These results are summarized in Table 4.5.

Table 4.5: Summary of Results from the Engine EPR Failure runs

TYPE OF SIMULATED FAILURE	PERFORMANCE FAILURE FLAG	ENGINE FAILURE FLAG	TIME (seconds)
85 % EPR	-	Set	10.1
115% EPR	-	Set	10.1

Figure 4.4a shows the EPR time history for the above run with the abort command generation enabled. It is seen from this figure that the EPR values begin their downward trend after the 10 second mark, at which point the abort command is generated. Figure 4.4b shows the ground speed curves, which after the initial increase until the 10 second point, begin dropping off. The simulation run is stopped before the ground speed actually reaches zero. The acceleration curves of Figure 4.4c are identical to Figure 4.3d until 10 seconds, after which they rapidly go towards zero due to braking and the reduction in engine power.

Figures 4.5a-4.5d show the time histories for the case where the EPR developed is 115% of nominal. The measured EPR values are seen to be higher than the predicted values in Figure 4.5a. Figure 4.5b shows the accelerations obtained from this run. The measured, predicted and the perfect sensor values are seen to start with a non-zero value (2 feet/sec^2). The plot of runway-required-plus-used, shown in Figure 4.5c, is seen to be a curved line as opposed to a horizontal straight line. This shows that the prediction of the runway required is not consistent. Again it needs to be pointed out that the algorithm caused the two engine failure flags to be set immediately after the 10 second point and a abort signal was also generated. These results are also included in Table 4.5.

Figures 4.6a-4.6c depict the time histories for the 115% EPR case with the abort command generation enabled. The EPR curves of Figure 4.6a are identical to those of Figure 4.5a up to the 10 second point. After that time the EPR values (both measured and predicted) begin decreasing. After leveling off between 16 and 18 seconds, the measured values are seen to go up. This is the point where the reverse thrusters come on. Figure 4.6c shows the calibrated airspeed time histories.

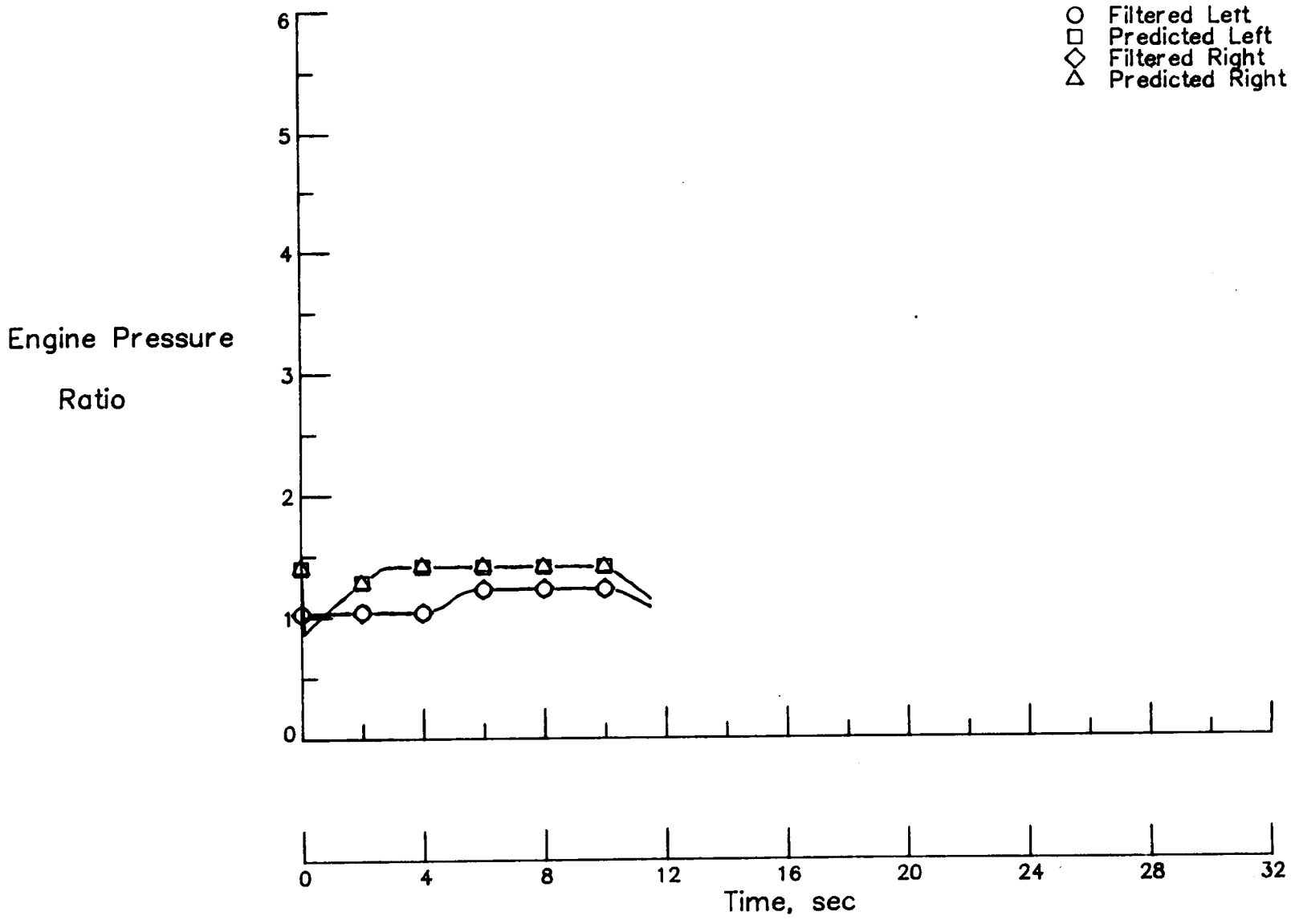

Figure 4.4a: Engine Pressure Ratio Time Histories for 85% EPR Case with Command Generation.

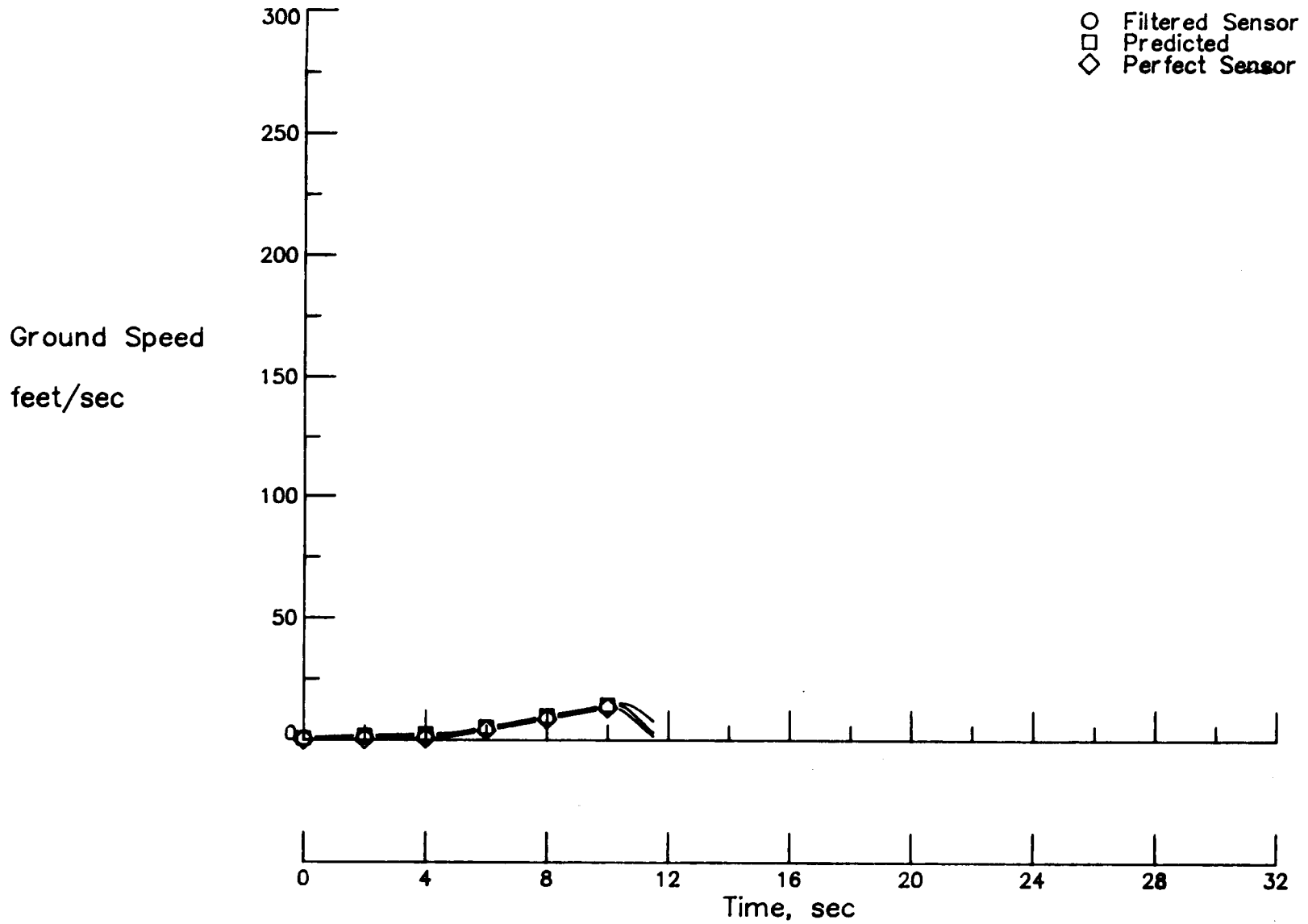

Figure 4.4b: Ground Speed Time Histories for the 85% EPR with Command Generation.

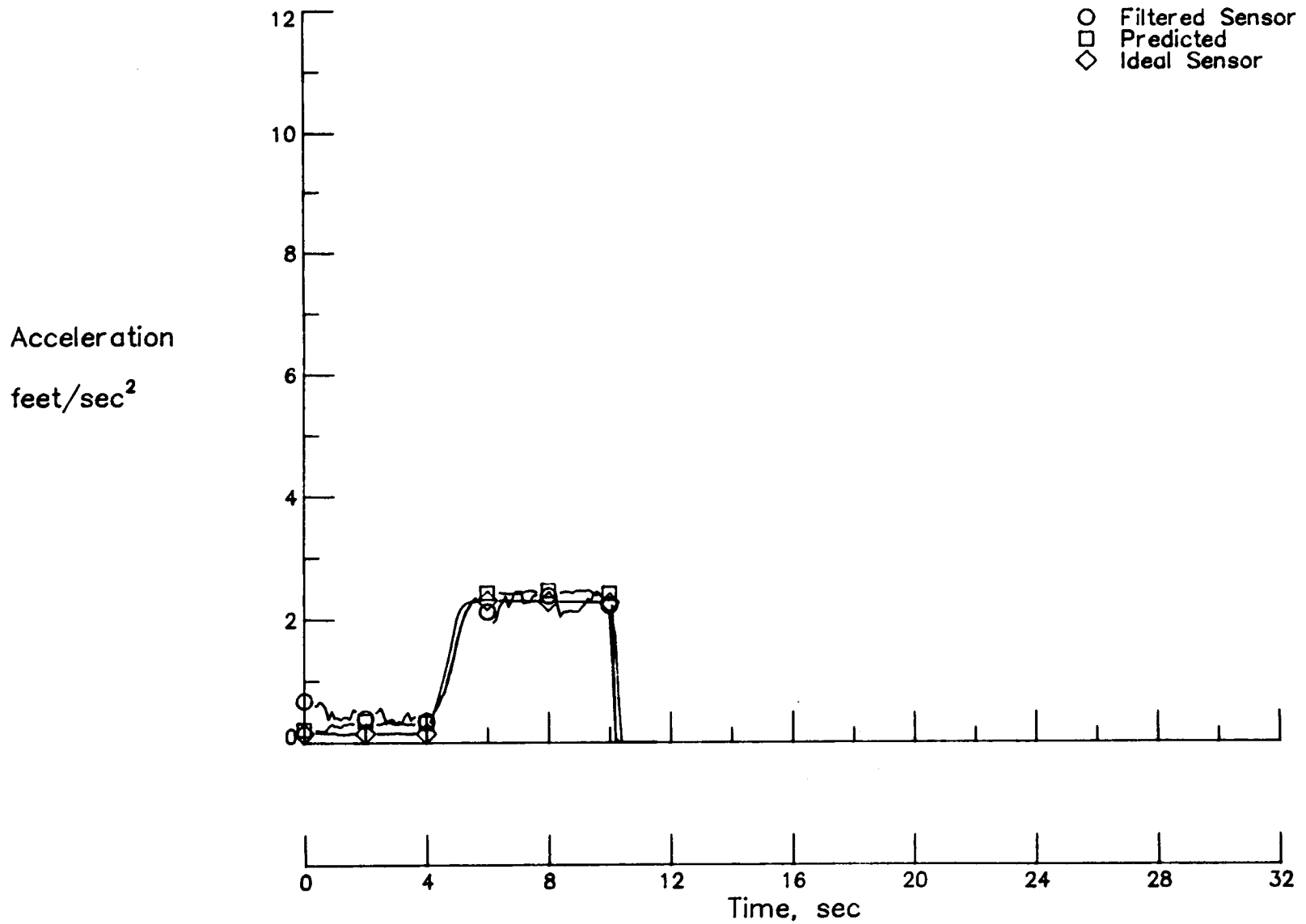

Figure 4.4c: Acceleration Time Histories for the 85% EPR Case with Command Generation.

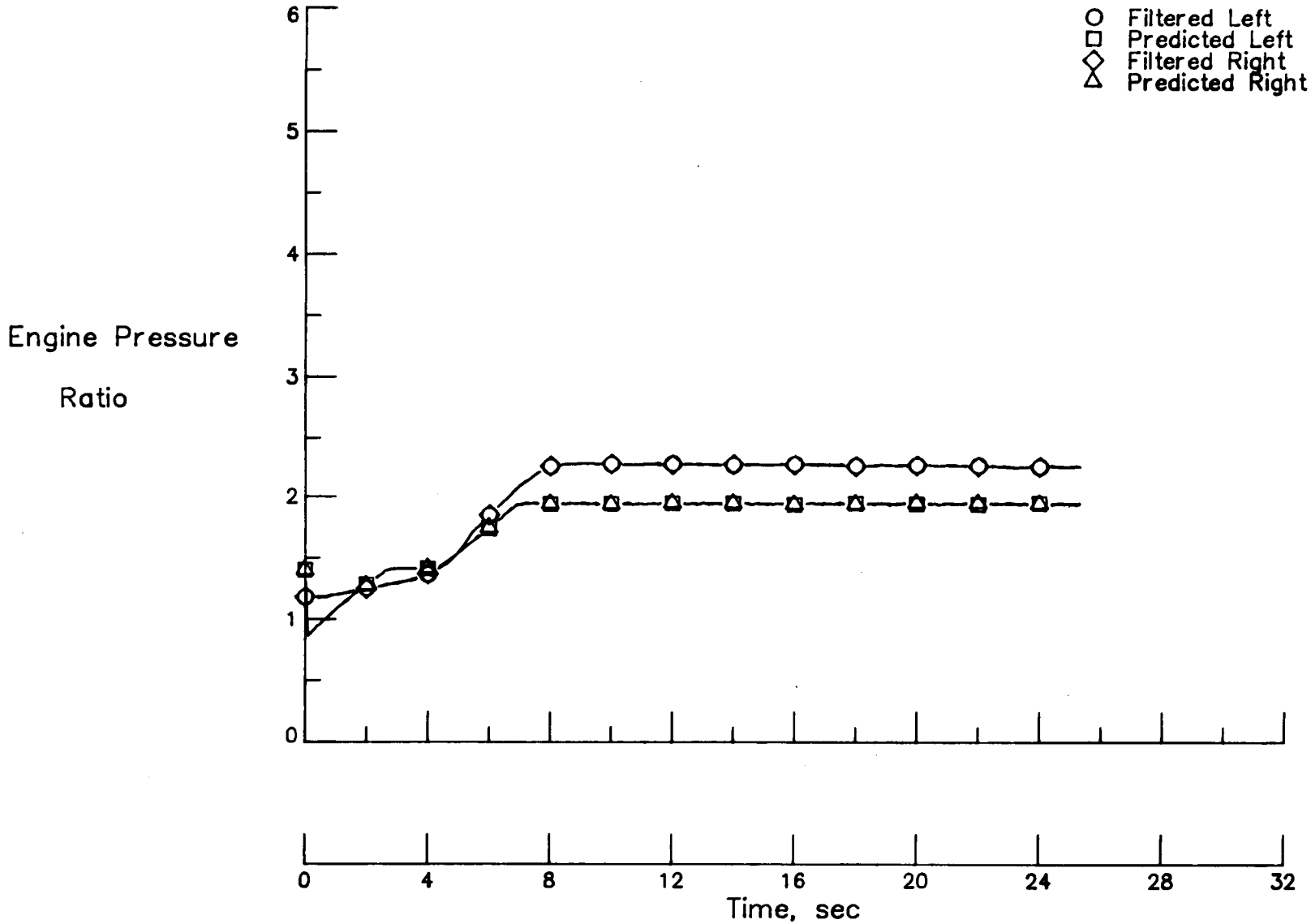

Figure 4.5a: Engine Pressure Ratio Time Histories for the 115% EPR Case.

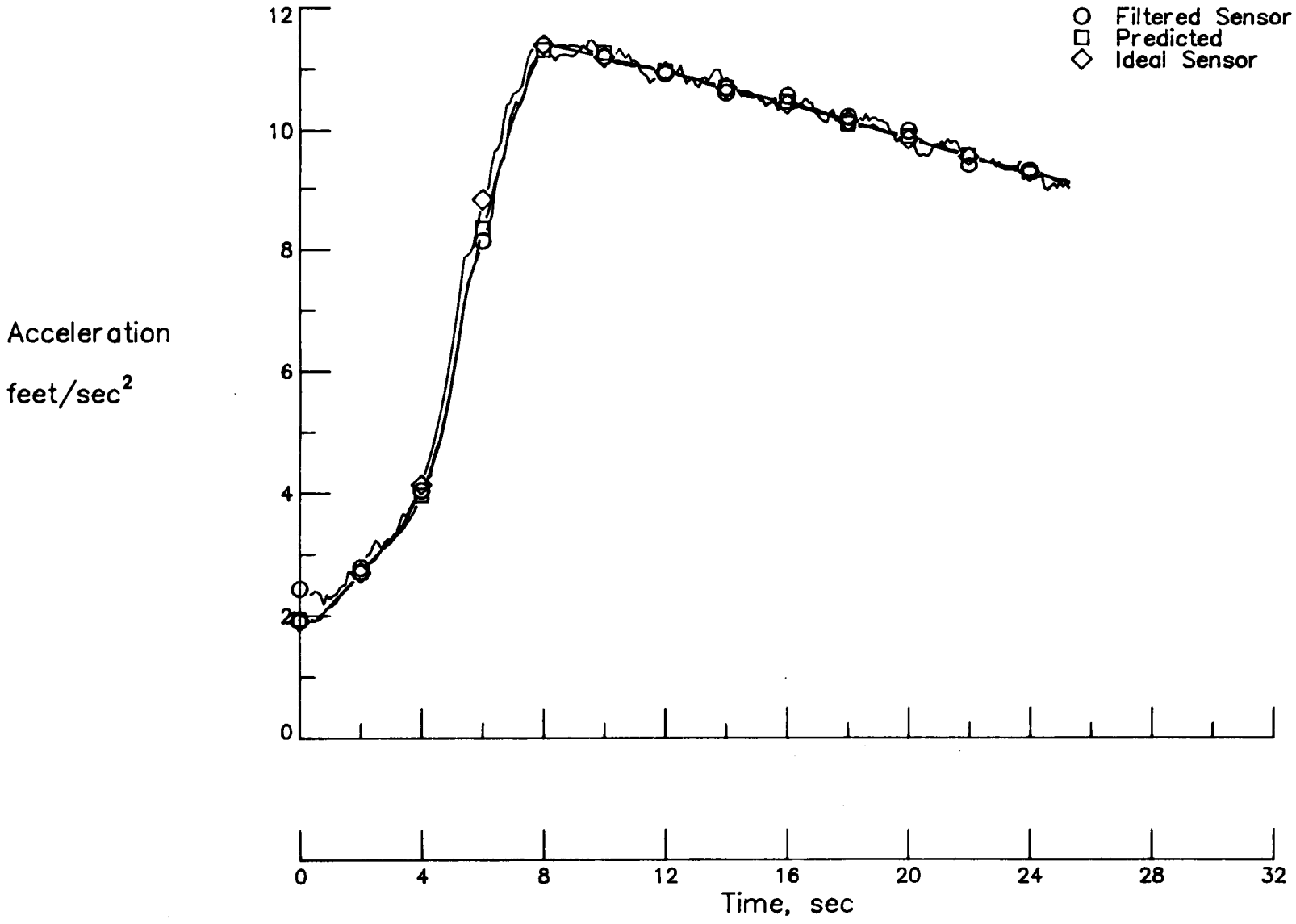

Figure 4.5b: Acceleration Time Histories for the 115% EPR Case.

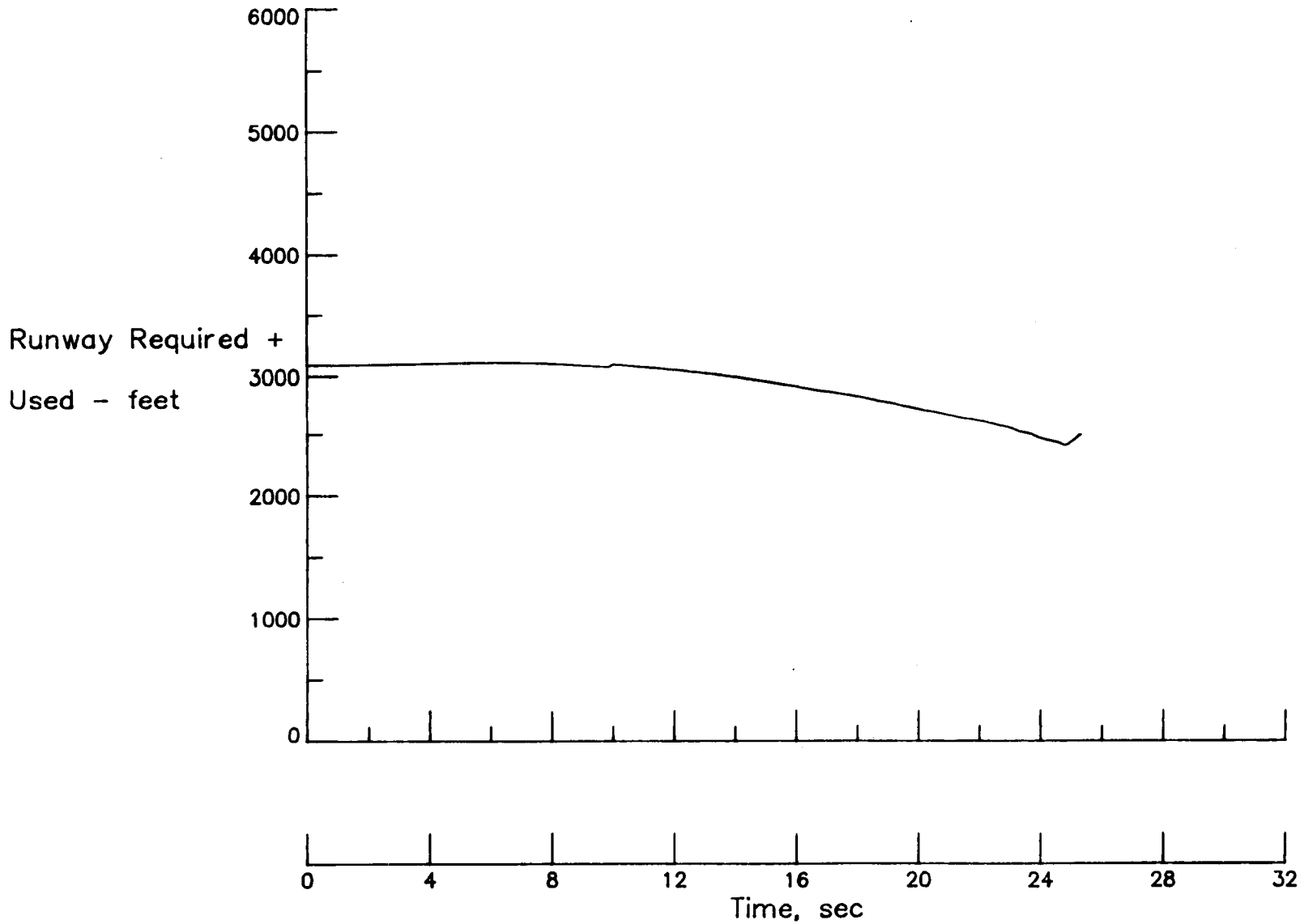

Figure 4.5c: Goodness of Runway Required Prediction for the 115% EPR Case.

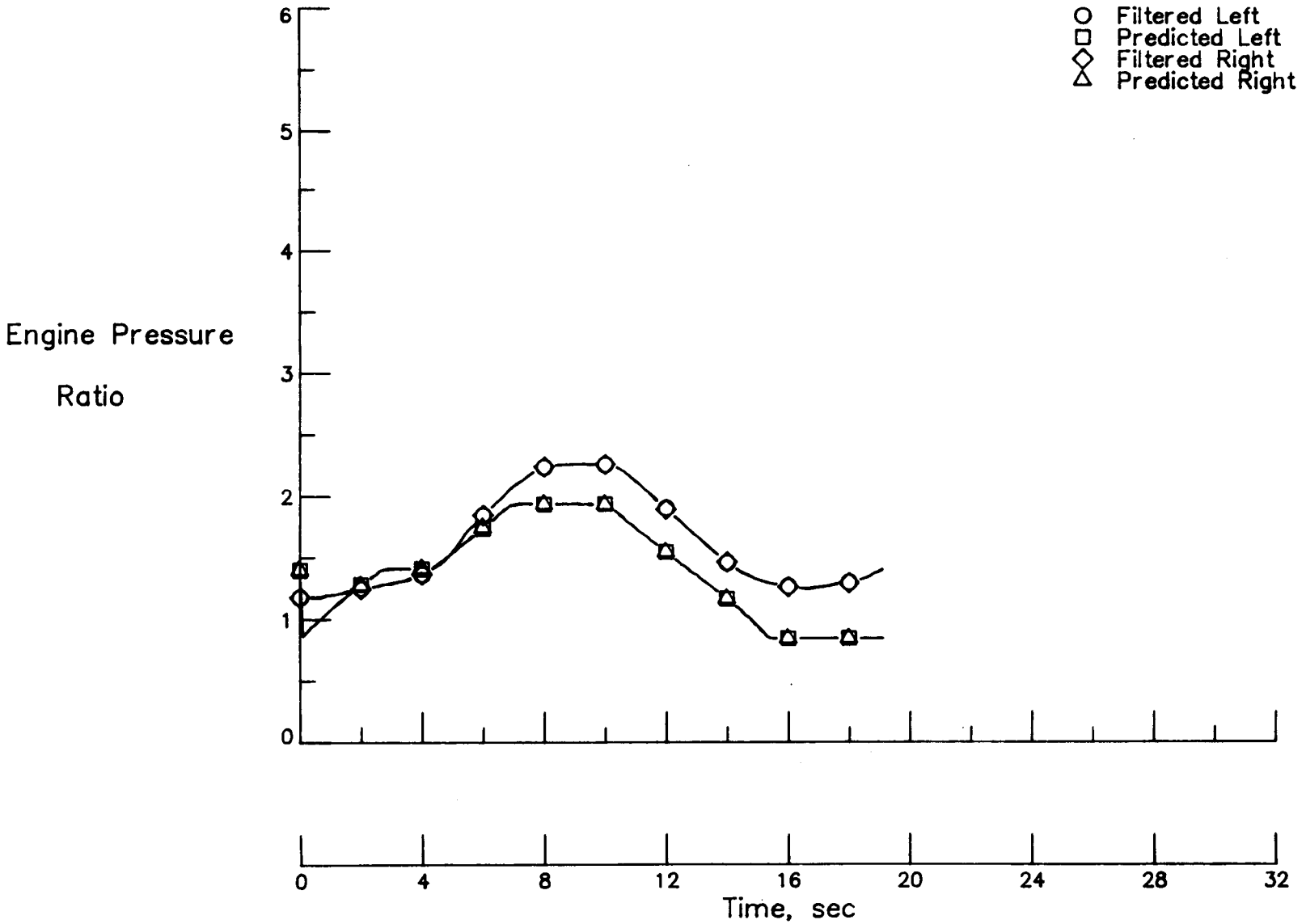

Figure 4.6a: Engine Pressure Ratio Time Histories for the 115% EPR Case With Command Generation.

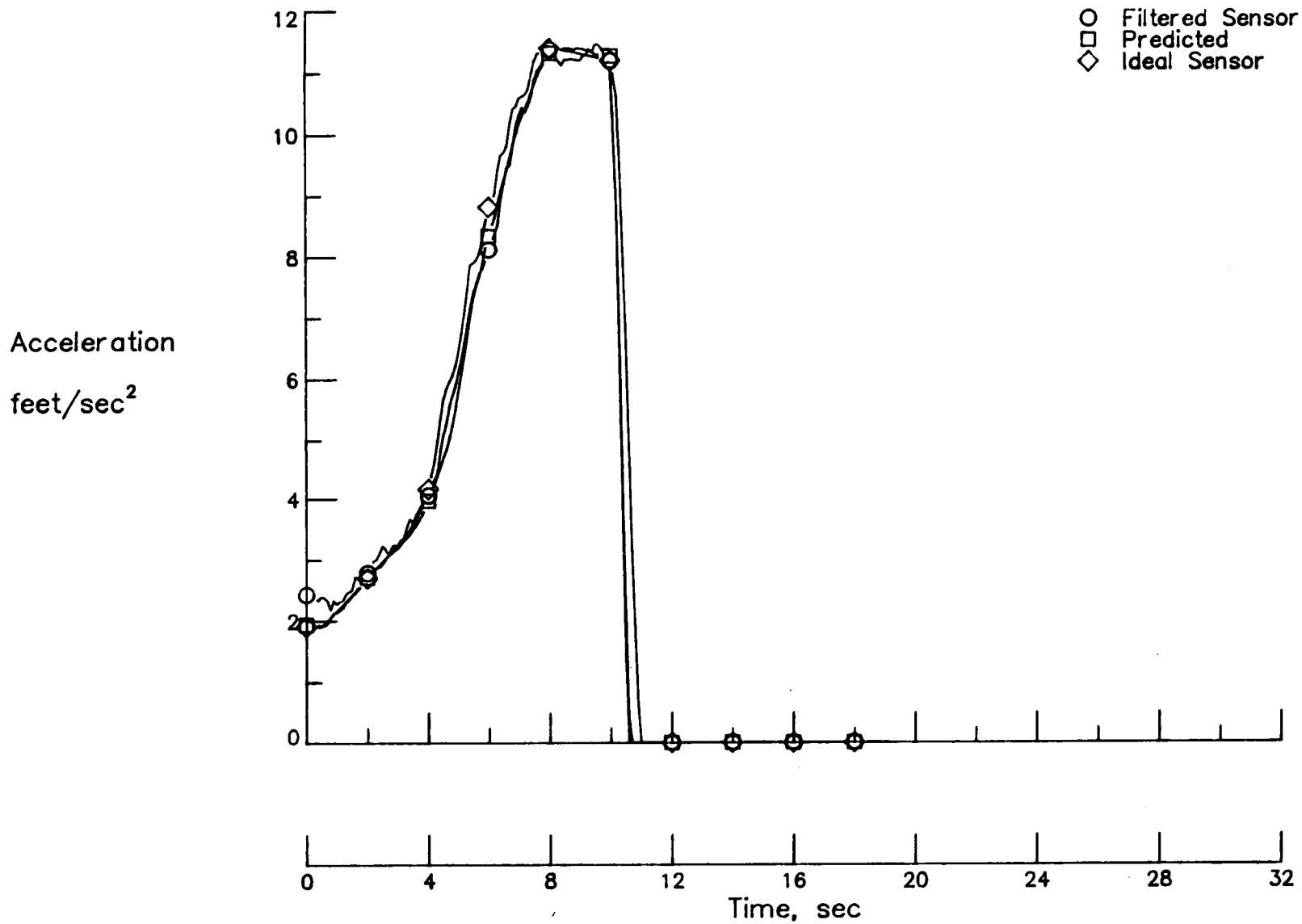

Figure 4.6b: Acceleration Time Histories for the 115% EPR Case With Command Generation.

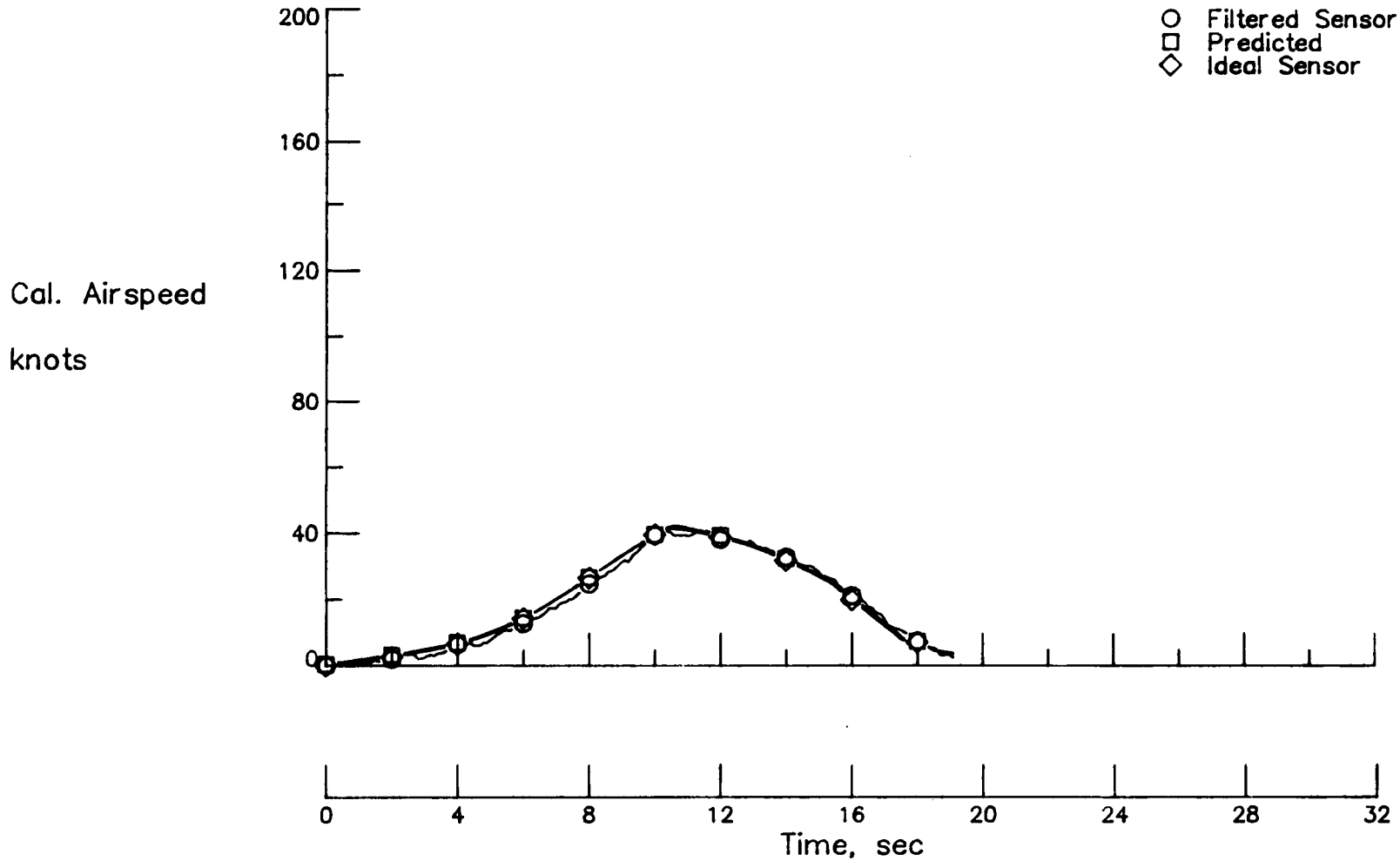

Figure 4.6c: Calibrated Airspeed Time Histories for the 115% EPR Case With Command Generation.

4.4.2 Engine Thrust Failure

Malfunctions in engine thrust are simulated by making a modification to the FORTRAN source code in the simulation model's engine deck. The thrust equation is multiplied by a constant factor to degrade or augment the thrust developed by the engine. Two cases are considered here. In the first case the thrust developed by the engine is forced to be 85% of the nominal value, and in the second case it is forced to be 115% of the nominal value.

Figures 4.7a-4.7c are representative time histories obtained for the 85% of nominal engine thrust case. Figure 4.7a shows the acceleration time histories generated from the simulation run. Before the 10 second point, the predicted acceleration is higher than the measured (filtered) value. At the ten second point, the difference between the measured and predicted accelerations are used to estimate a new runway friction coefficient. After the 10 second point, the measured and predicted values of acceleration are practically coincident, except towards the end of the run where the two values begin to separate. Figure 4.7b plots a time history of the runway required prediction from the algorithm. At the 10 second point, the runway required prediction is seen to go up by about 600 feet. This increase corresponds to the newly estimated higher rolling friction coefficient at that instant. From Figure 4.7c, it is seen that the sum of runway required and the runway used remains constant after the 10 second adjustment of the rolling friction coefficient. This run did not set any engine failure or performance failure flags. The outputs from this run indicate that the 15% reduction in the thrust output from the engine is interpreted as an increase in the rolling friction coefficient from 0.015 to 0.063 . Table 4.6 summarizes the results from this run. It is seen from this table that

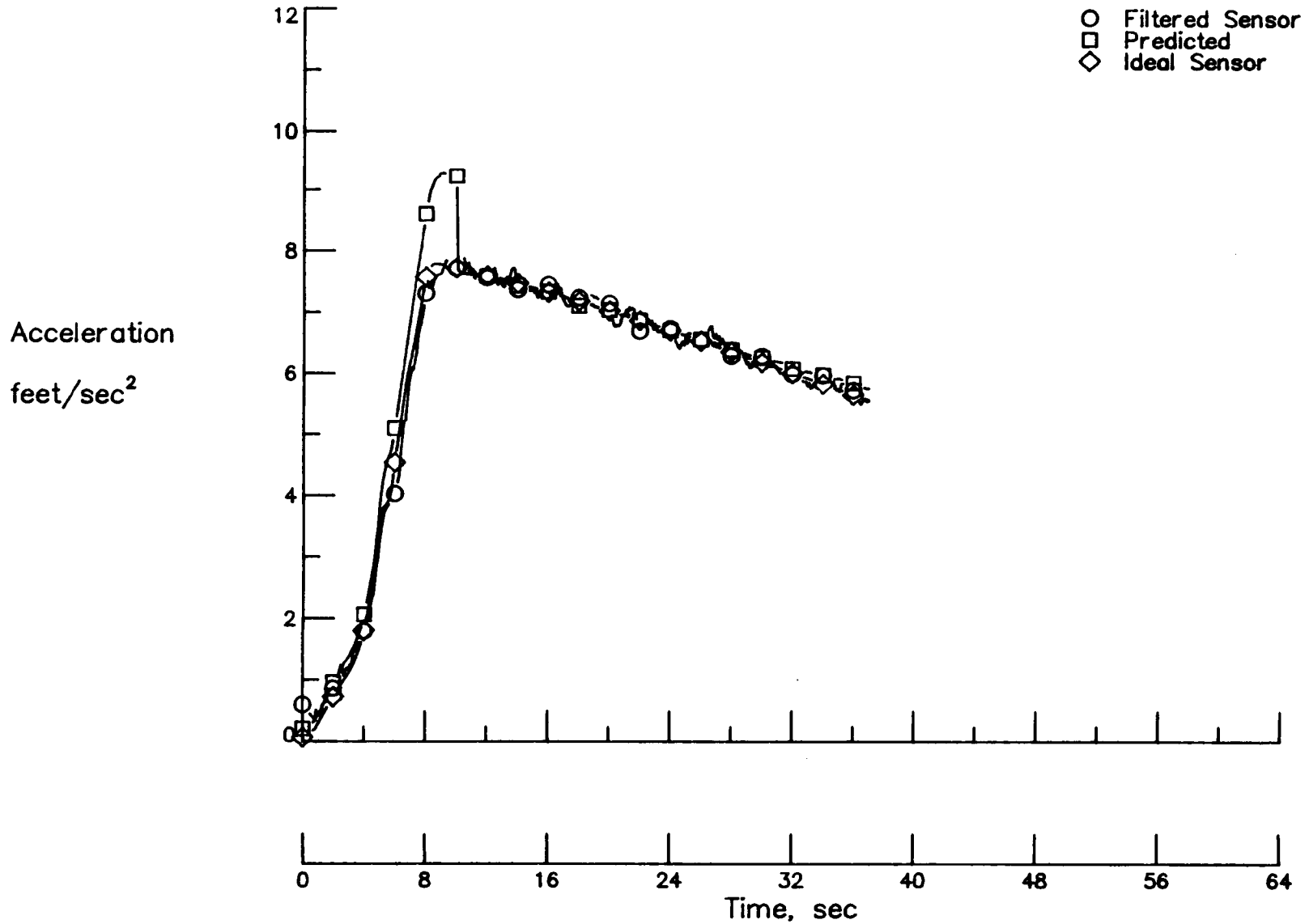

Figure 4.7a: Acceleration Time Histories for the 85% Thrust Case.

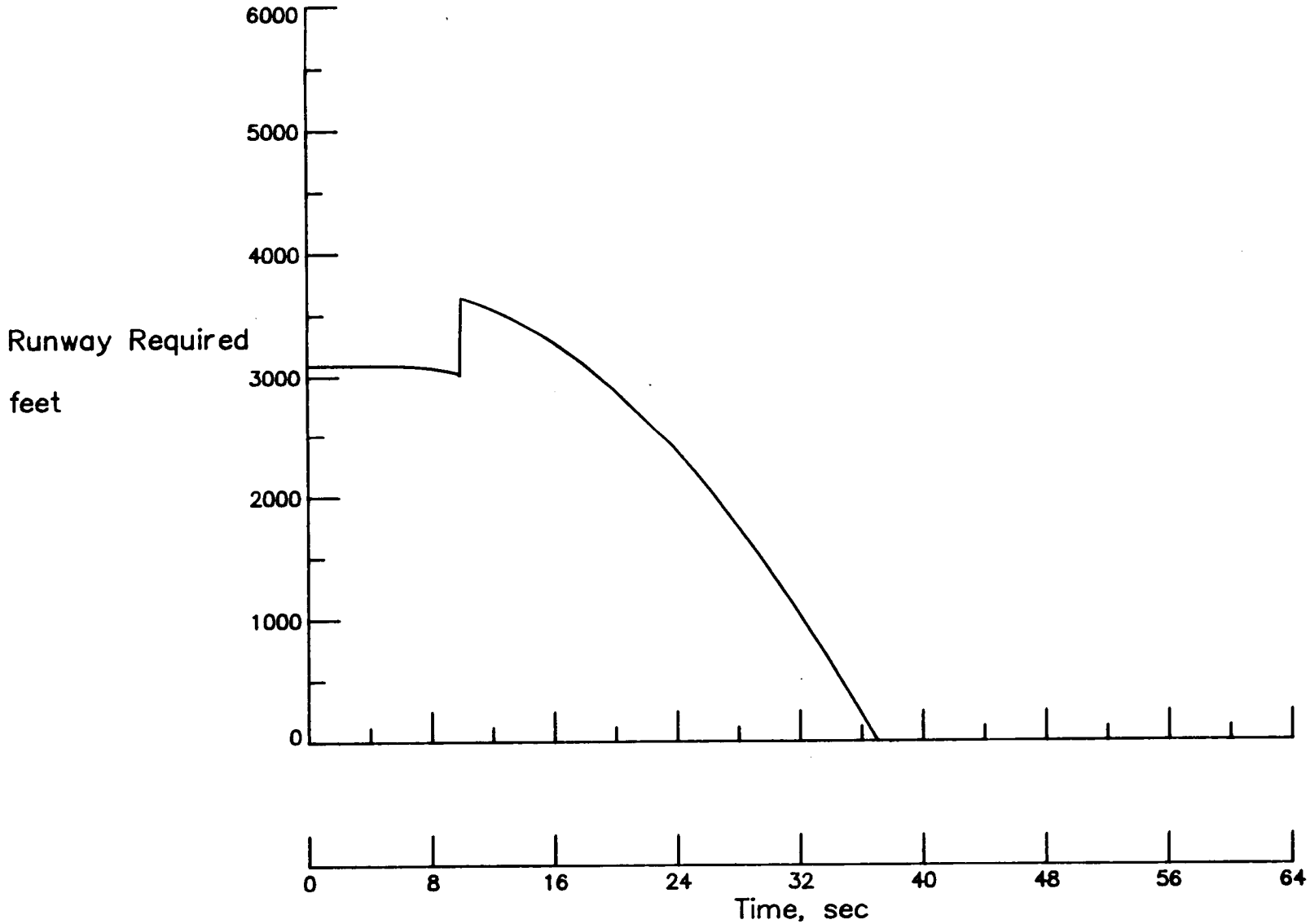

Figure 4.7b: Predicted Runway Required Time History for the 85% Thrust Case.

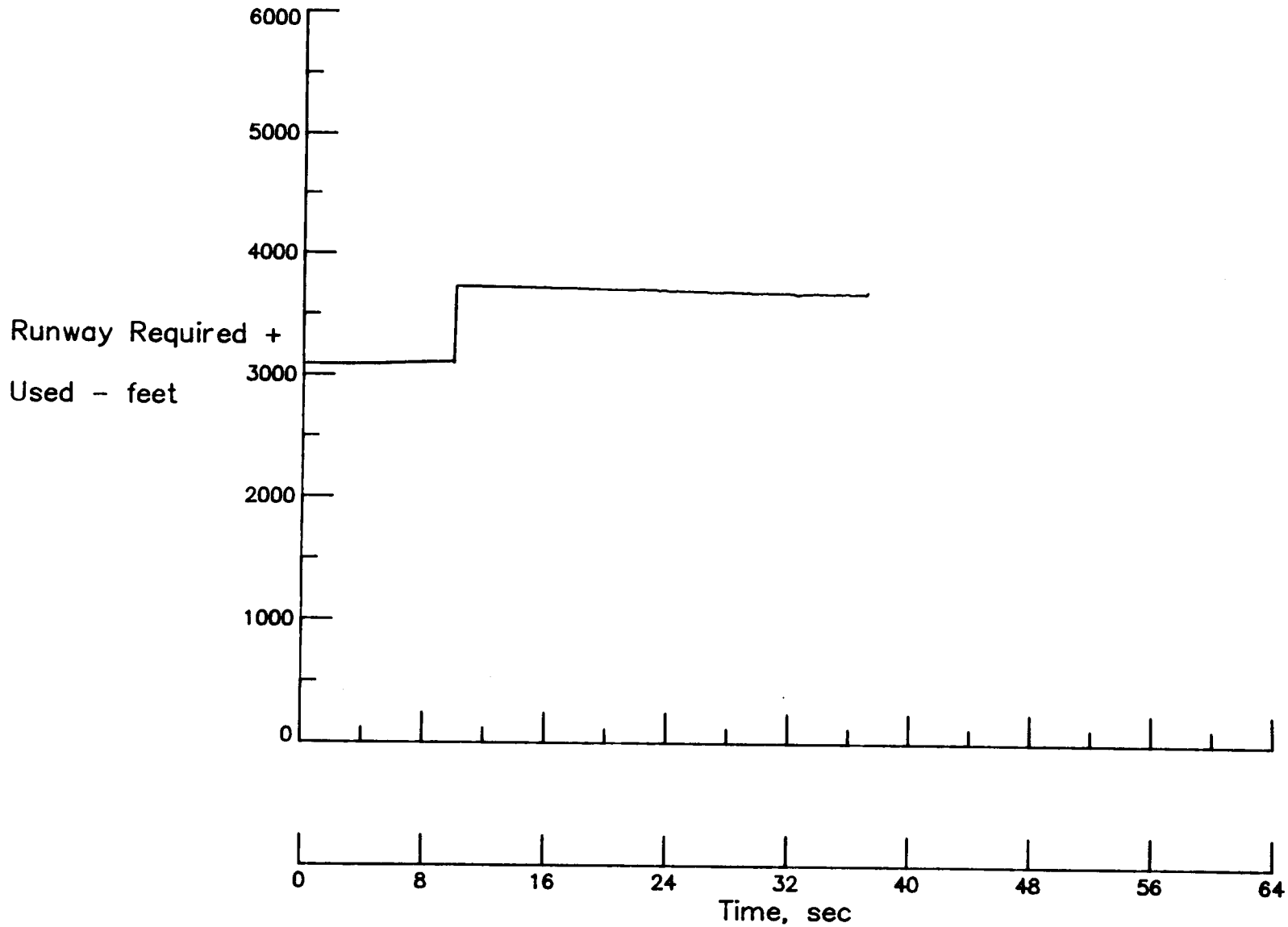

Figure 4.7c: Goodness of Runway Requirement Prediction for the 85% Thrust Case.

although the absolute error in the predicted runway requirement to achieve rotation speed is only 35 feet, relative to the no engine malfunction case (100% Thrust case), the change in the prediction error is 165 feet. This seeming improvement in the performance of the algorithm has resulted because of the 10 second adjustment to friction coefficient. The table indicates that the adjusted friction coefficient for this case is 0.063, which falls outside the nominal range of free rolling friction coefficient values used in this development (0.005 - 0.040).

Table 4.6: Summary of results for the Thrust Failure Runs

CASE	MEASURED CAS at rotation (feet)	PREDICTED RUNWAY REQUIRED overall (feet)	RUNWAY USED (-)	UPDATED FRICTION COEFF. (feet)	RUNWAY PREDICTION ERROR (feet)	CHANGE IN PREDICT. ERROR
85% Thrust	128.4	3738.	3703.	0.063	35.	165.
100% Thrust	128.1	3132.	3262.	-0.028	-130.	0.
115% Thrust	128.4	2689.	2732.	-0.028	-43.	87.

Note: The measured Calibrated Airspeed (CAS) is the filtered sensor output

Figures 4.8a-4.8c summarize the time histories obtained for a run where the thrust developed is 115% of the nominal value. In Figure 4.8a the predicted acceleration is below the measured value until the 10 second point. At the 10 second point the prediction is adjusted upward by the newly estimated low runway friction coefficient. After the 10 second point the measured, predicted, and the ideal sensor accelerations are practically identical. The time history of the algorithm predicted runway requirement is plotted in Figure 4.8b. The shift downward of about 500 feet at the 10 second point is caused by the adjustment to the runway friction coefficient at that point. Figure 4.8c, where the sum of runway required and the runway used is plotted, also shows this jump at the 10 second point. In addition, there is an upward movement of the curve just before rotation speed. As explained before, this is caused by the lags introduced in the measured calibrated airspeed by the filtering process. No engine or performance failure flags are set during this run. Just as for the 85% thrust case, the absolute error performance of the algorithm seems to have improved to a mere -43 feet from a -130 feet for the no engine malfunction case. The 15% increase in engine thrust is interpreted by the algorithm as a decrease in the runway friction coefficient, from a value of 0.015 to -0.028, which is an invalid quantity. Table 4.6 includes the results from this run also.

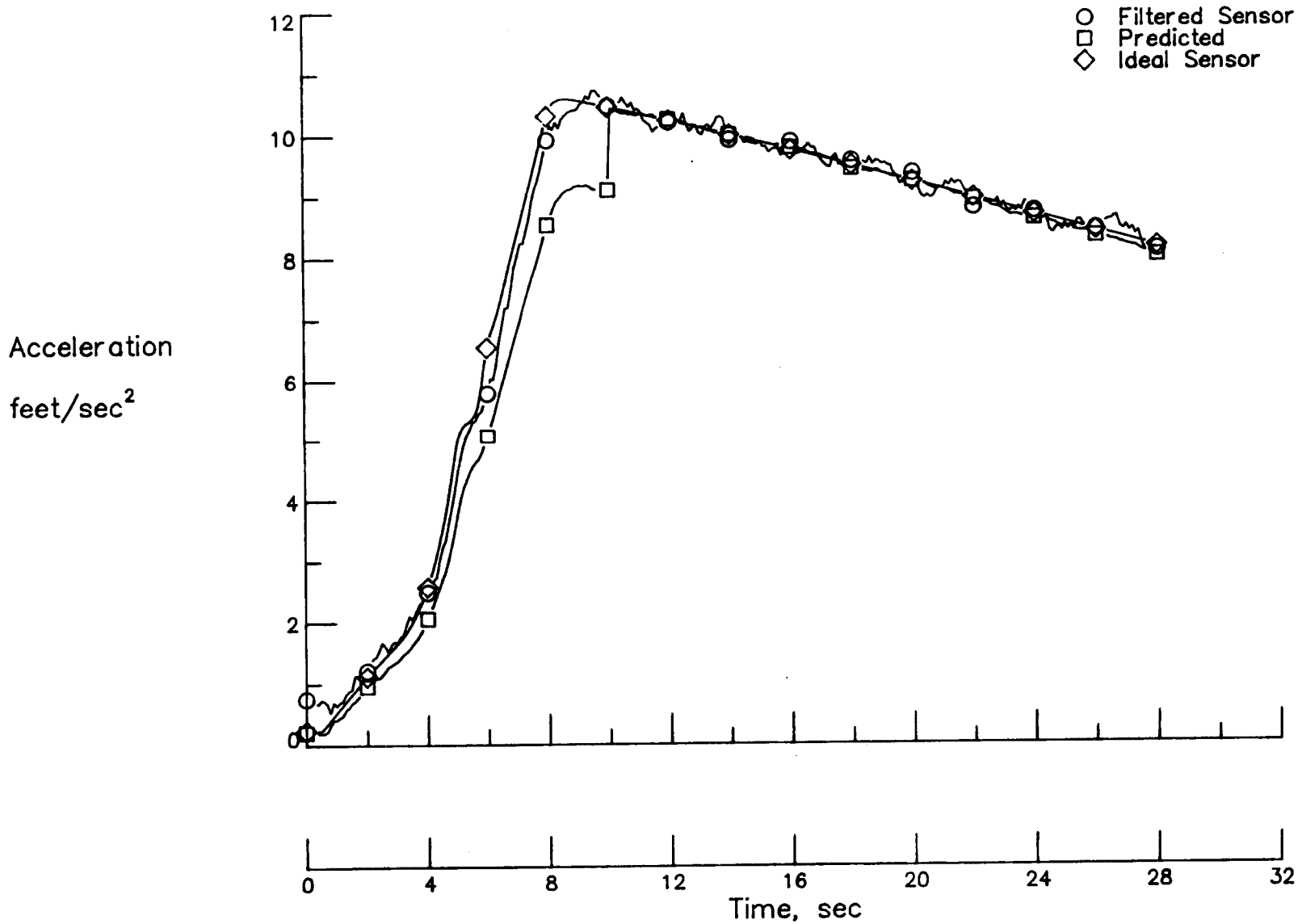

Figure 4.8a: Acceleration Time Histories for the 115% Thrust Case.

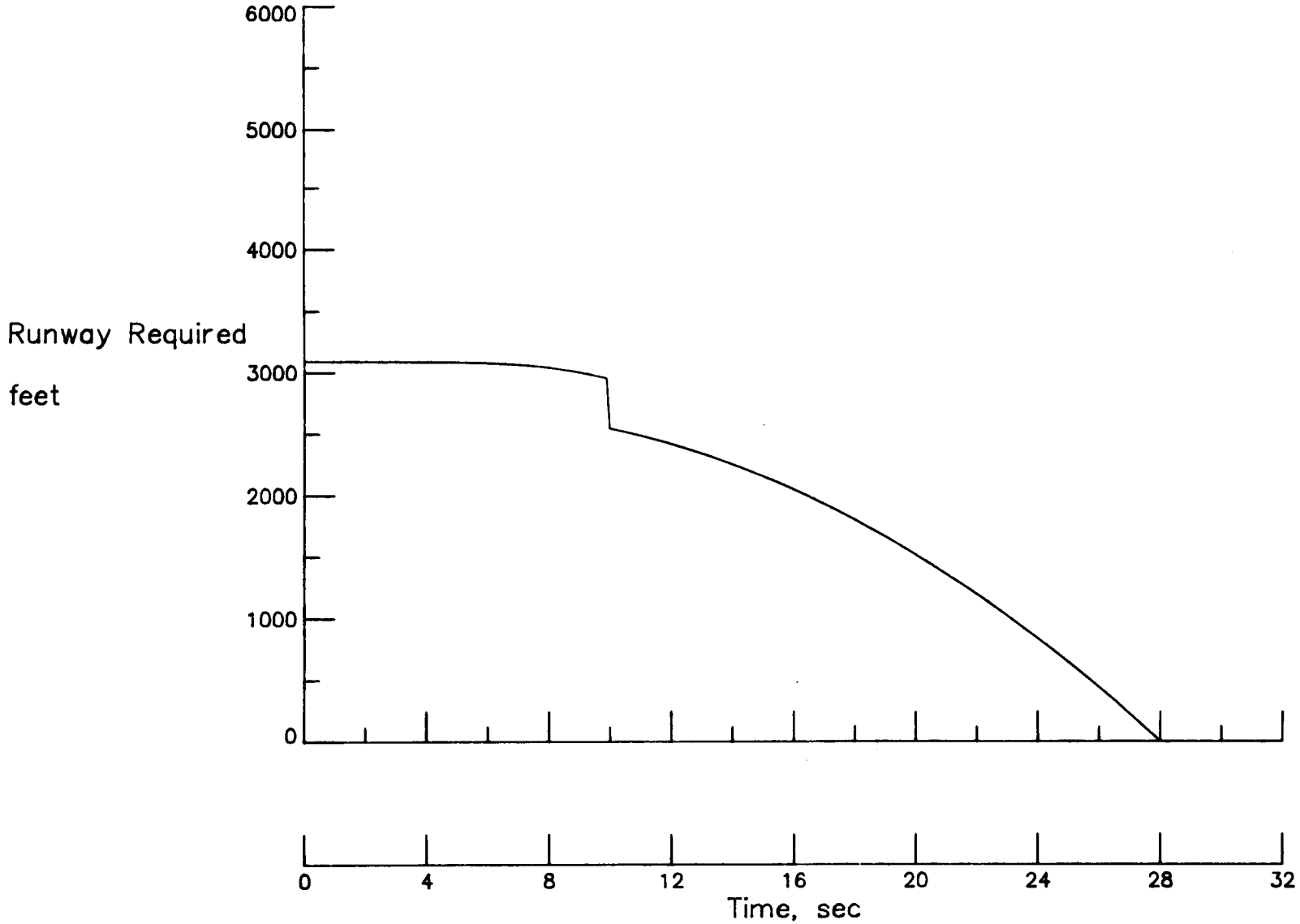

Figure 4.8b: Predicted Runway Required Time History for the 115% Thrust Case.

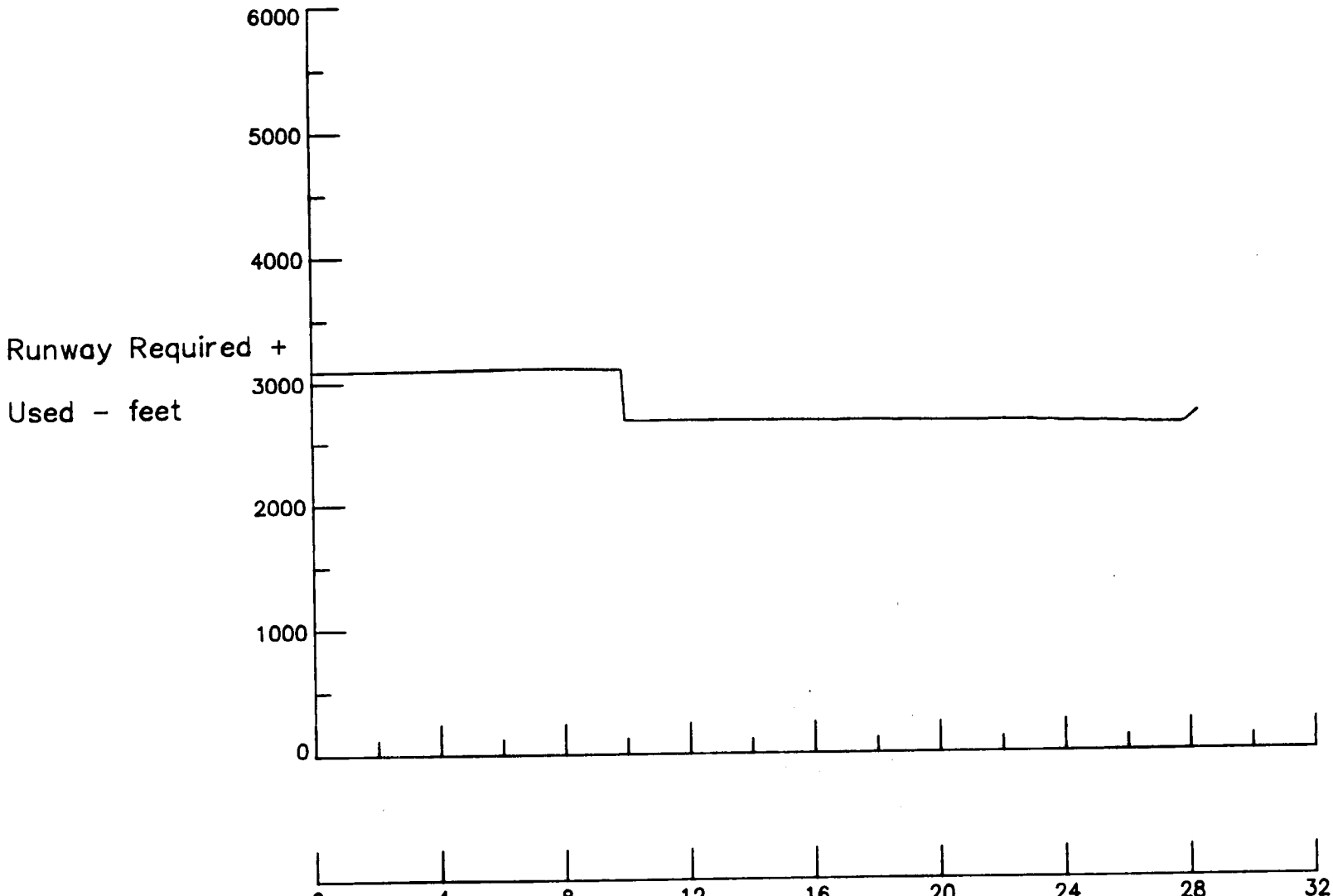

Figure 4.8c: Goodness of Runway Required Prediction for the 115% Thrust Case.

4.5 Summary of Results

The algorithm is seen to function very well for all the 10 combinations of ambient and loading conditions. The algorithm is also able to identify engine failures that affect the Engine Pressure Ratio (EPR). The algorithm, as it is set up, is unable to distinguish between engine performance degradations that do not affect the EPR and discrepancies in the input runway friction coefficient. Looking at the friction coefficient data of Table 4.6 suggests that a check on the adjusted friction coefficient may be used as an additional check on airplane performance anomalies during the takeoff run.

CHAPTER 5

SENSITIVITY AND FAILURE MODE ANALYSIS

5.1 Introduction

The previous chapter dealt with evaluating the Takeoff Performance Monitoring System. The algorithm was shown to work under several different ambient conditions during normal takeoff runs. A few performance degradation cases were also demonstrated. All the simulation runs in that chapter dealt with design point cases, that is, the inputs to the algorithm were the actual ambient conditions. In this chapter, the sensitivity of the algorithm to off-design cases and the effects of sensor failures are explored. The sensitivity analysis is done by forcing selected inputs to the algorithm and the simulation to be different and comparing the algorithm's predictions with the true values generated by the simulation model.

The failure analysis is carried out by causing the sensor outputs from the simulation model to be in error and again comparing the algorithm's predictions with the simulation model performance.

The algorithm can function in a closed loop mode where it generates command inputs to the airplane systems, or it can function in an open loop mode where it generates Go/Abort informative signals for use by the pilot. Both modes of operation are discussed here.

Section 5.2 deals with sensitivity studies and section 5.3 evaluates the effects of different failures. The chapter ends with a summary of results.

All of the parameter variations are for a base line case as listed in Table 5.1.

Table 5.1: Baseline Flight Conditions for Sensitivity and Failure Analysis Runs

PARAMETER	VALUE	UNITS
Pressure Altitude	32.	feet
Ambient Temperature	75.	° F
Runway Winds	0.	knots
Runway Friction Coefficient	0.015	-
Gross Weight	88504.	lbs.
Flap Setting	5.	degrees

5.2 Sensitivity Analysis

As has been noted before, the algorithm consists of a pre-takeoff and a real-time segment, both of which require a set of one-time inputs. In this section the sensitivity of the algorithm to errors in some of these inputs are explored. Such errors could occur in parameters such as runway winds, ambient temperature, airplane weight and the flap setting selected for takeoff. Effects of aerodynamic contamination and a reduction in the frequency of calls to the algorithm are also considered.

5.2.1 Sensitivity to errors in inputs.

This section discusses the sensitivity of the algorithm to errors in the one time inputs.

The first parameter considered is runway winds. The algorithm is forced to use an assumed runway wind condition while the simulation model is run at the true wind speed. This is accomplished with changes to the FORTRAN

source code for the algorithm. The effects of wind speed errors are summarized in Table 5.2. Figure 5.1a shows the calibrated airspeed time histories obtained for a run with an assumed windspeed of 20 knots but an actual wind of 0 . The predicted value in this plot is 20 knots higher than the filtered sensor and the ideal sensor values. This reflects the 20 knot error introduced in the one time inputs.

Table 5.2: Effect of Wind Speed Error

WIND SPEED		PERFORMANCE					
ASSUMED SPEED (knots)	ACTUAL SPEED (knots)	MEASURED CALIBRATED AIRSPEED (knots)	PREDICTED RUNWAY REQUIRED (feet)	RUNWAY USED (feet)	ADJUSTED FRICTION COEFF.	PREDIC--TION ERROR (feet)	CHANGE IN PREDIC--TION ERROR (feet)
10.	0.	128.1	2645.	3260.	0.012	-615.	-620.
10.	10.	129.1	2690.	2685.	0.017	5.	0.
10.	20.	128.7	2742.	2295.	0.022	447.	442.
20.	0.	128.1	2204.	3262.	0.007	-1058.	-1046.
20.	10.	129.1	2241.	2685.	0.012	-444.	-432.
20.	20.	128.7	2283.	2295.	0.017	-12.	0.
20.	30.	128.0	2330.	1902.	0.023	428.	440.

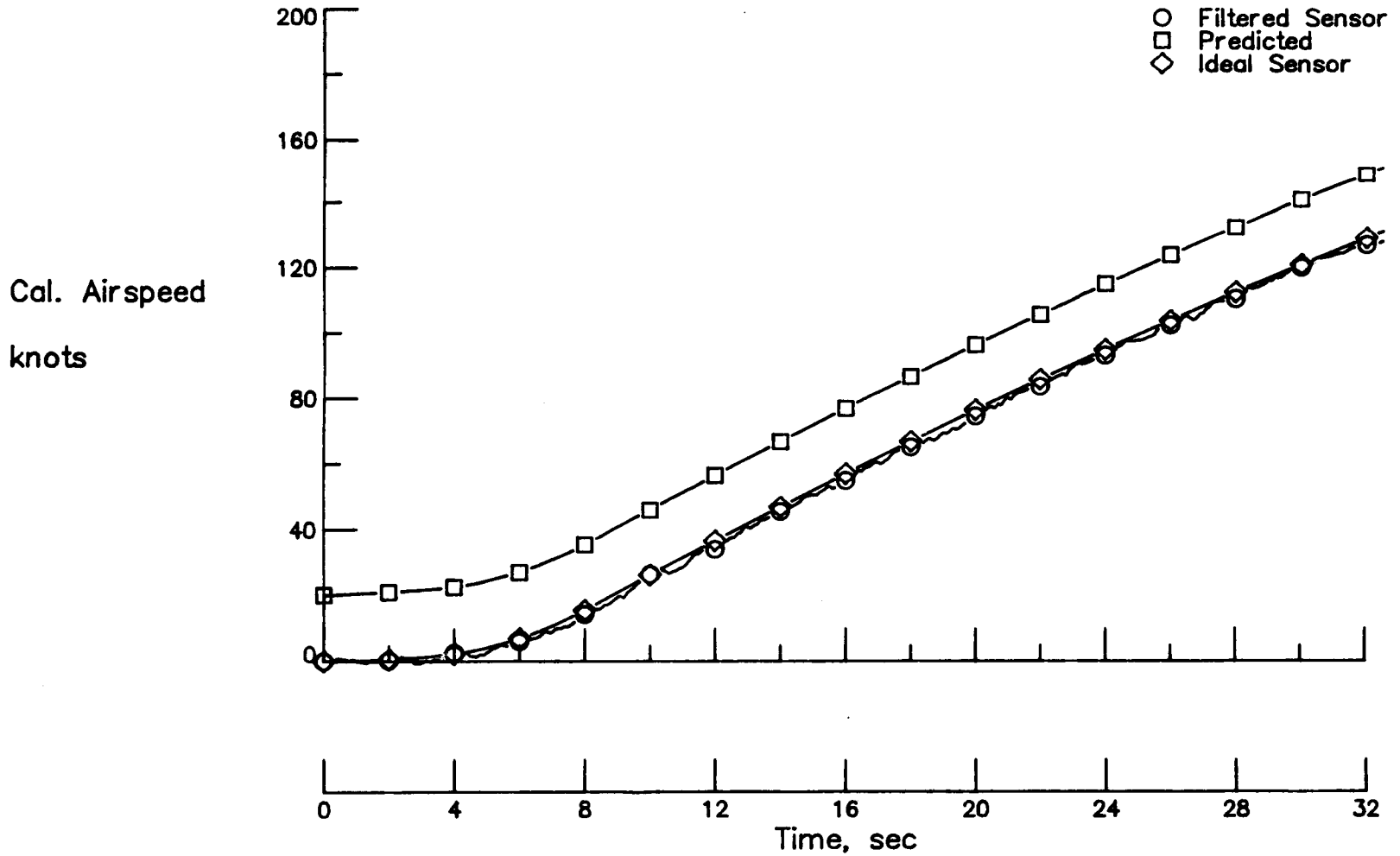

Figure 5.1a: Calibrated Airspeed Time Histories for the Wind Speed Error Case - Assumed u_w = 20 k, Actual u_w = 0.

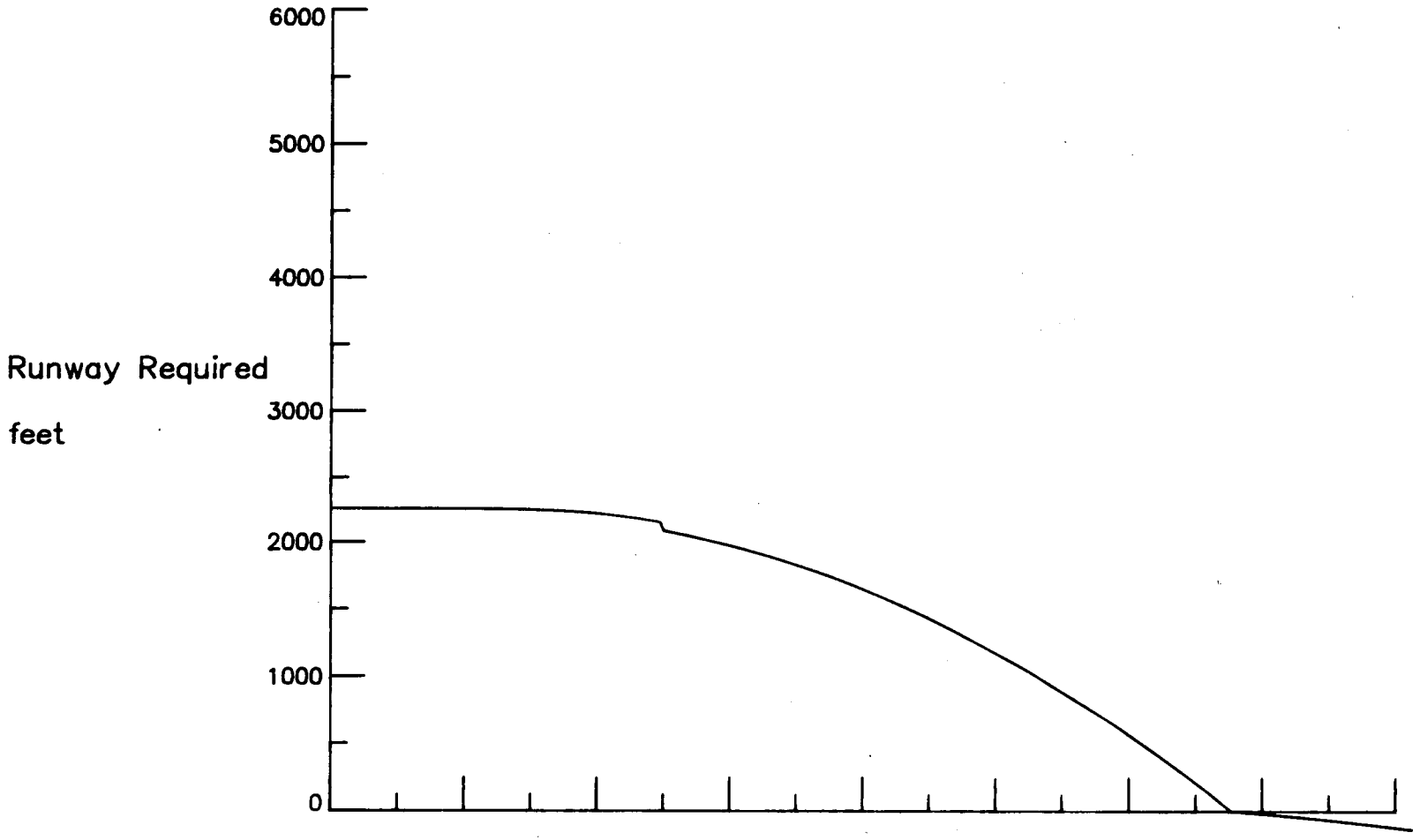

Figure 5.1b: Runway Required Time History for the Wind Speed Error Case - Assumed u_W = 20 k, Actual u_W = 0.

Figure 5.1b shows the runway required prediction from the algorithm. It is seen from this plot that there is a small adjustment at the 10 second point, caused by an adjustment to the runway friction coefficient, after which it steadily decreases to zero. At about the 27 second point, the predicted runway required reaches zero, and after that a negative value is predicted. Because of the initial runway wind input of 20 knots, the algorithm computes a ground speed required to rotation which is 20 knots below the value actually needed, causing the runway required prediction to go negative after the algorithm computed speed has been attained.

Figure 5.2a is the calibrated airspeed time history for a run with an assumed windspeed of 20 knots and an actual windspeed of 25 knots. Here all three calibrated airspeeds start at non-zero values. The predicted value is 5 knots below the measured values, reflecting the 5 knot error in wind speed input. The runway required plot of Figure 5.2b does not reach a value of zero. The algorithm, because of the 5 knot error in wind speed, predicts that a nonzero runway length is required to attain rotation speed.

The above sensitivity analysis indicates that the algorithm is highly sensitive to errors in runway winds. No Abort signal is generated by the algorithm. To explore the effects of an onboard wind estimator a simple runway wind estimator is implemented in the algorithm. At the 9.9 second point, the computed calibrated airspeed is subtracted from the measured filtered calibrated airspeed to form a one-time estimate of the runway wind which is used throughout the rest of the run. Table 5.3 summarizes the results obtained from this run for the runway wind error case which assumes a wind of 20 knots, when in reality there is no wind. It is seen from this table that the error in the predicted runway required is reduced from an underprediction of 1058 feet to an under prediction of 139 feet.

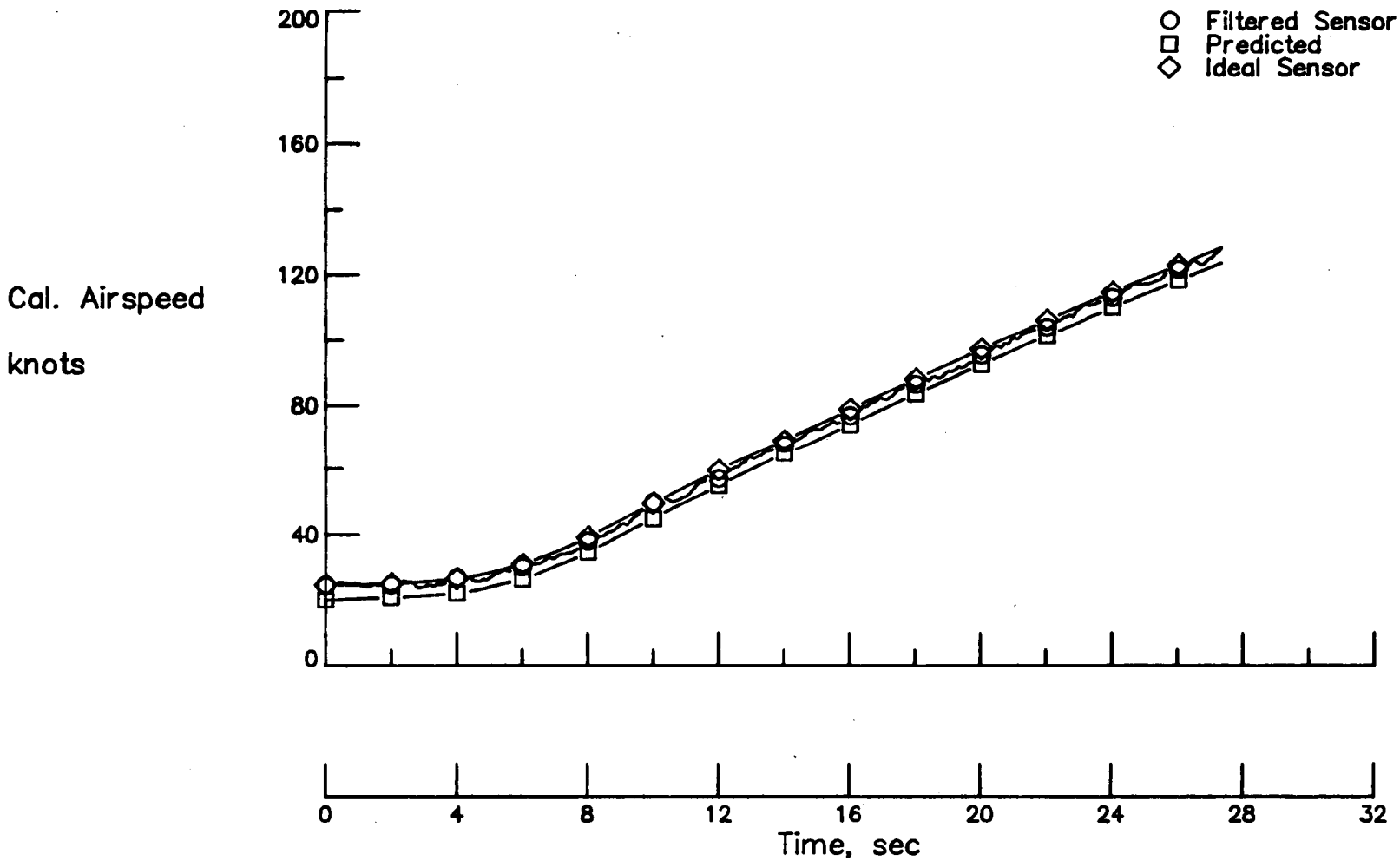

Figure 5.2a: Calibrated Airspeed Time Histories for the Wind Speed Error Case - Assumed u_w = 20 k, Actual u_w = 25 k.

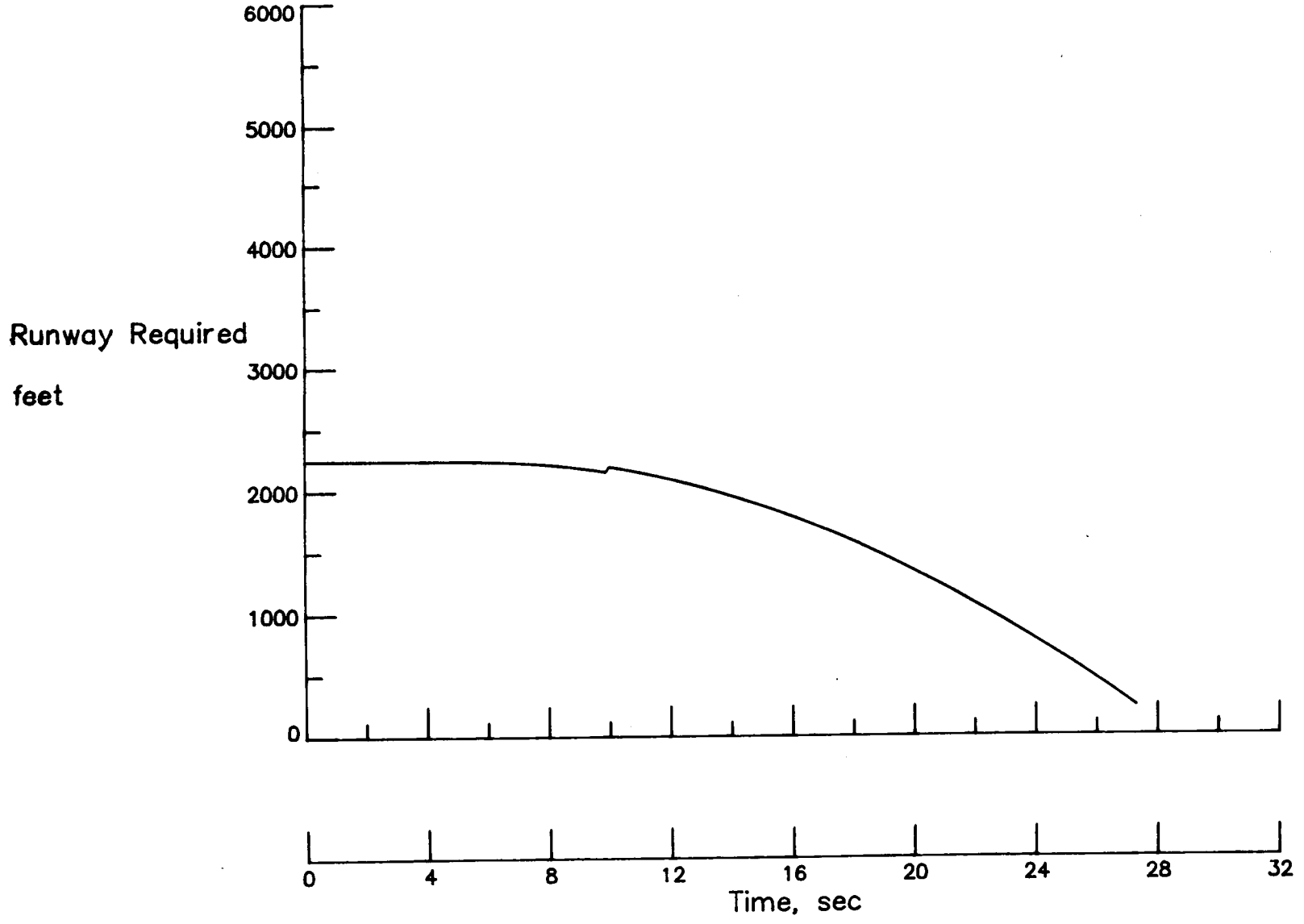

Figure 5.2b: Runway Required Time History for the Wind Speed Error Case - Assumed u_w = 20 k, Actual u_w = 25.

Table 5.3: Effect of Wind Speed Error with a Wind Estimator

WIND SPEED		PERFORMANCE			
ASSUMED SPEED (knots)	ACTUAL SPEED (knots)	MEASURED CALIBRATED AIRSPEED (knots)	PREDICTED RUNWAY REQUIRED (feet)	RUNWAY USED (feet)	PREDICTION ERROR (feet)
20.	0.	128.1	3123.	3262.	-139.

The effect of errors in the ambient temperature inputs is summarized in Table 5.4. Two cases considered, for this analysis, assume an ambient temperature of 50 °F, whereas the actual temperatures are 25 °F and 75 °F respectively. From the table it is seen that the error in the ambient temperature inputs does not cause the estimated friction coefficient to change. For an assumed temperature of 50 °F with an actual temperature of 25 °F the algorithm is seen to overpredict by 347 feet. With the error going in the other direction, it is seen that the algorithm underpredicts by 505 feet. Thus it is seen that the algorithm is sensitive to errors in ambient temperature inputs.

Errors in the gross weight of the airplane are considered next. The two cases assume a gross weight of 88,504 pounds and actual weights of 98,504 and 78,504 pounds respectively. Figure 5.3a shows the acceleration time histories for the run with an actual weight of 98,504 pounds. Due to the assumed weight being 88,504 pounds, the algorithm predicted acceleration is higher than the measured values until the 10 second point. At that time,the

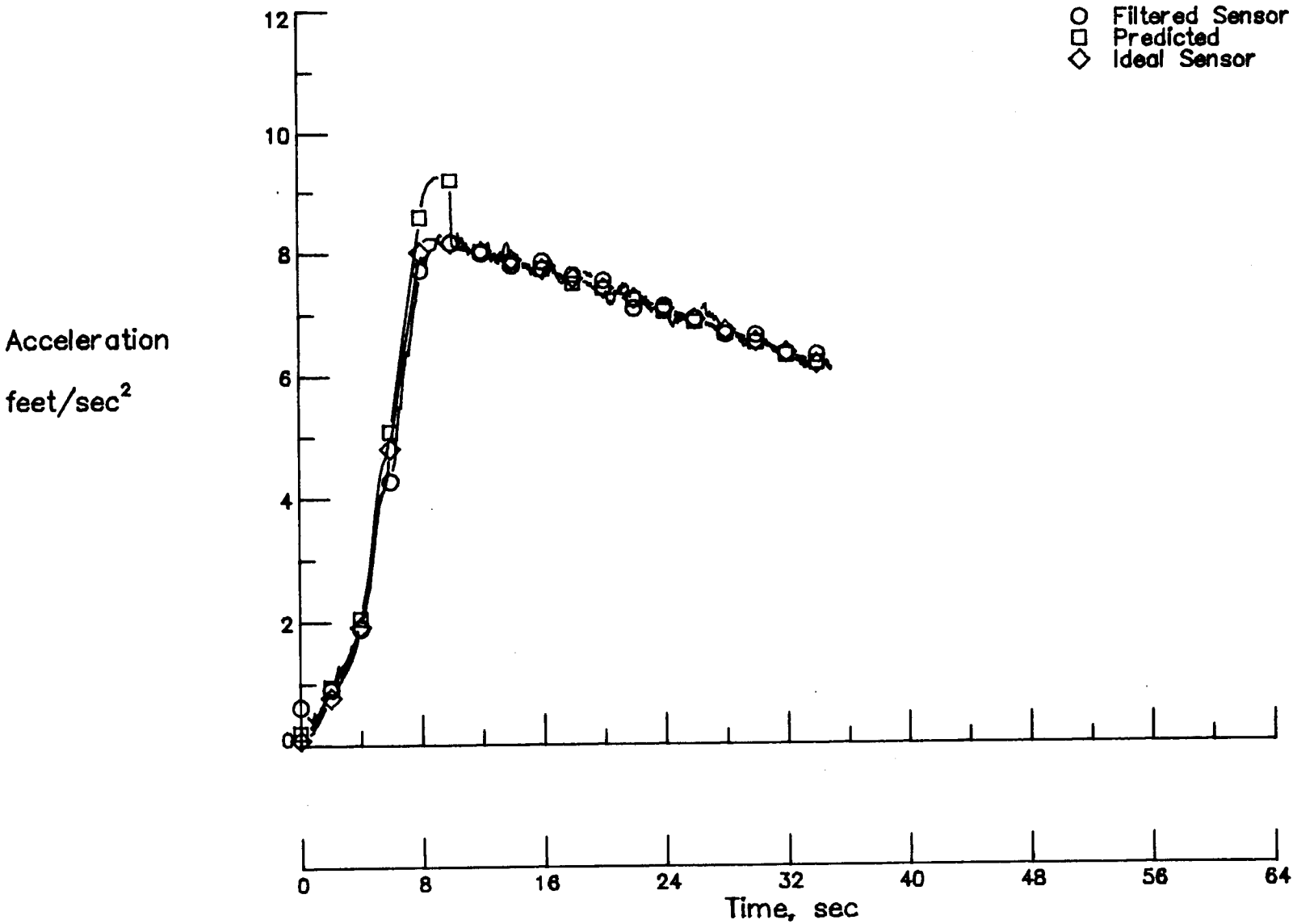

Figure 5.3a: Acceleration Time Histories for the Weight Error Case - Assumed W = 88,504 lbs, Actual W = 98,504.

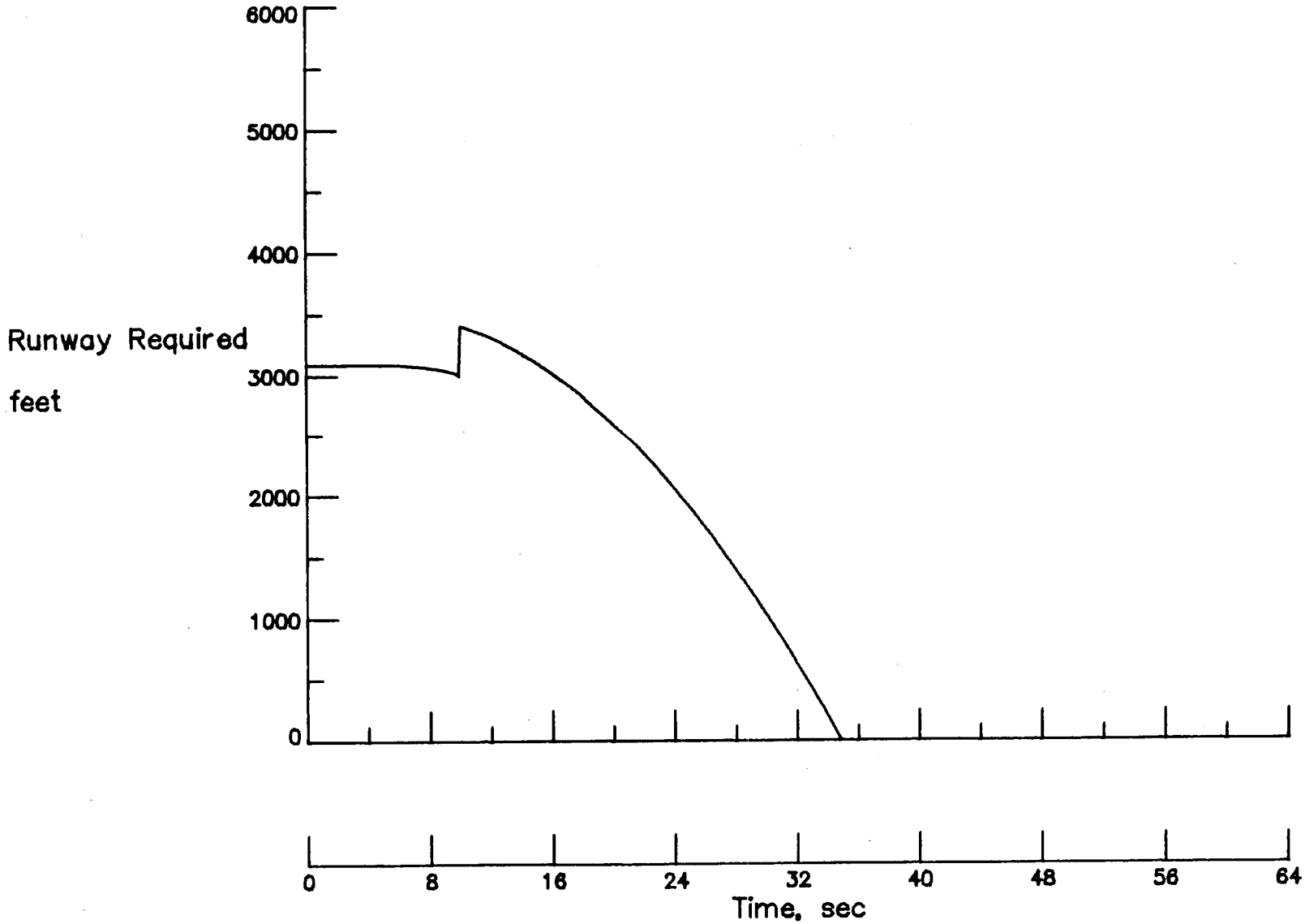

Figure 5.3b: Runway Required Time History for the Weight Error Case - Assumed W = 88,504 lbs, Actual W = 98,504 lbs.

Table 5.4: Effect of Ambient Temperature Errors

TEMPERATURE		PERFORMANCE					
ASSUMED TEMP. (° F)	ACTUAL TEMP. (° F)	MEASURED CALIBRATED AIRSPEED (knots)	PREDICTED RUNWAY REQUIRED (feet)	RUNWAY USED (feet)	ADJUSTED FRICTION COEFF.	PREDIC--TION ERROR (feet)	CHANGE IN PREDIC--TION ERROR (feet)
50.	25.	128.6	2964.	2617.	0.017	347.	387.
50.	50.	128.2	2975.	3015.	0.017	-40.	0.
50.	75.	128.1	2986.	3491.	0.017	-505.	-465.

difference between the predicted and measured accelerations is interpreted as being caused by discrepancies in the assumed runway friction coefficient. A new friction coefficient is estimated based on this difference. This new friction coefficient causes the two acceleration values to be the same. Figure 5.3b illustrates the variation of predicted runway required with time. The runway friction coefficient update at the 10 second point shows up as an increase in the predicted runway requirement of about 400 feet. Another important effect of this error, which does not show up in either the plot or the outputs from the run, is that the rotation speed of 128 knots chosen by the algorithm is based on the 88,504 pound gross weight. The flight manual recommended rotation speed for the 98,504 pound gross weight is 138 knots. The effect of an actual weight of 78,504 pounds is similar but in the opposite direction. Figure 5.4a shows the predicted acceleration

getting adjusted upward at the 10 second point, resulting in a reduction of the estimated friction coefficient. Figure 5.4b shows a corresponding reduction in the predicted runway requirement. In this case, the airplane remains on the runway longer than needed. The flight manual recommended rotation speed for a 78,504 pound airplane is 119 knots, but the algorithm required a speed of 128 knots (rotation speed for a 88,504 pound airplane) to be reached. This results in the airplane remaining on the runway for 387 feet more than needed. Neither case generates Abort signals. Table 5.5 summarizes the results from the simulation outputs.

The next parameter considered is Flap setting. As indicated in Table 5.6, the assumed flap setting is 5 degrees. The two actual flap settings chosen for this run are 1 degree and 10 degrees. For the 1 degree setting the flight manual recommended rotation speed is 133 knots, and for the 15 degree setting it is 123 knots. The simulation run for the first case reaches only 128 knots. For the second run, the measured calibrated airspeed is higher than 123 knots 2828 feet into the takeoff run. But the airplane is kept on the ground until the calibrated airspeed exceeds 128 knots, the recommended rotation speed for 5 degrees flaps. Thus the airplane remains on the ground 437 feet more than required.

5.2.2 Sensitivity to aerodynamic degradation

The sensitivity of the algorithm to contaminants on the airplane body which alter the aerodynamic characteristics are considered here. Ice formation on the wings is one example of such contamination. The effects of such contamination is simulated by reducing the lift coefficient and

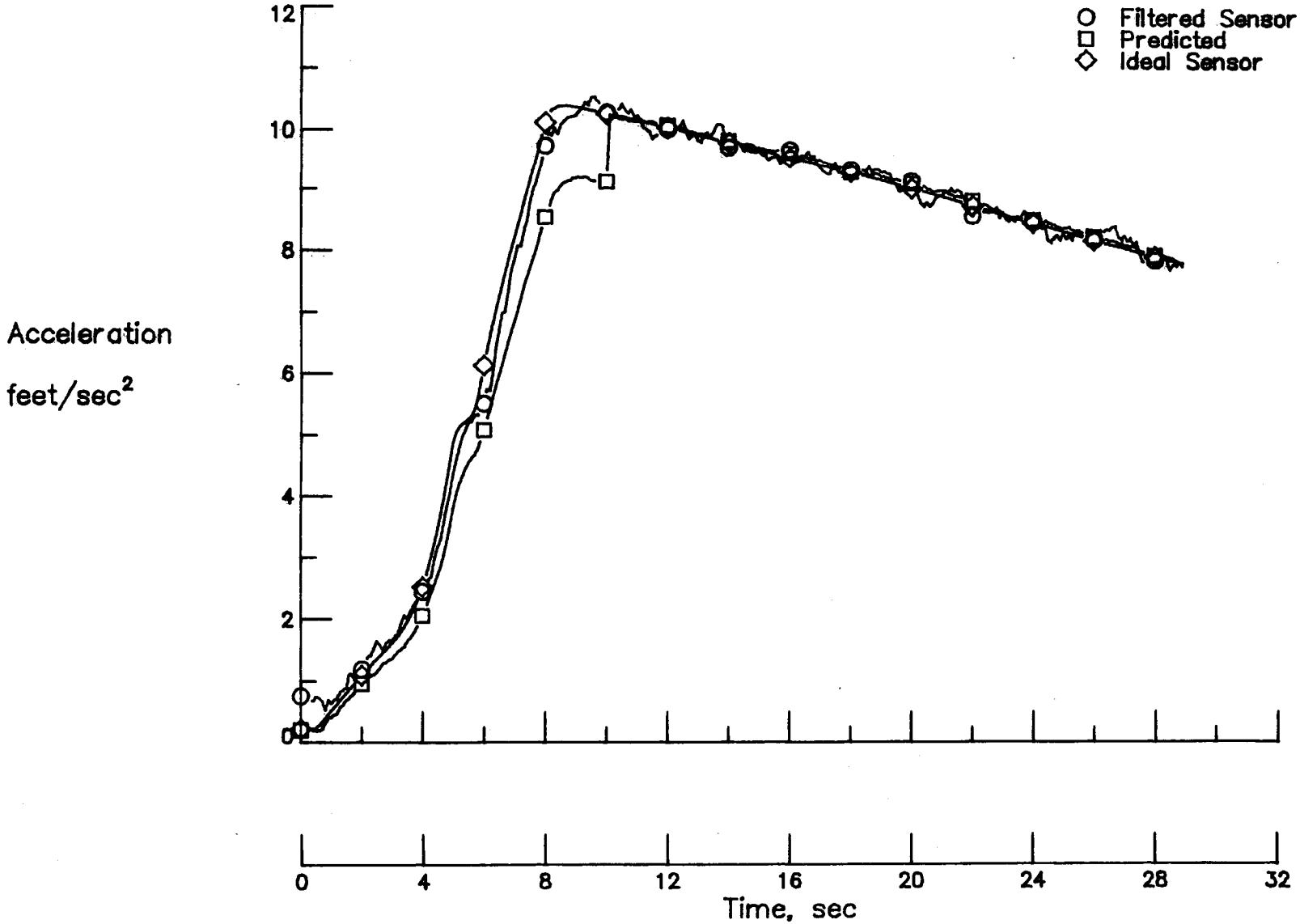

Figure 5.4a: Acceleration Time Histories for the Weight Error Case - Assumed W = 88,504 lbs, Actual W = 78,504 lbs.

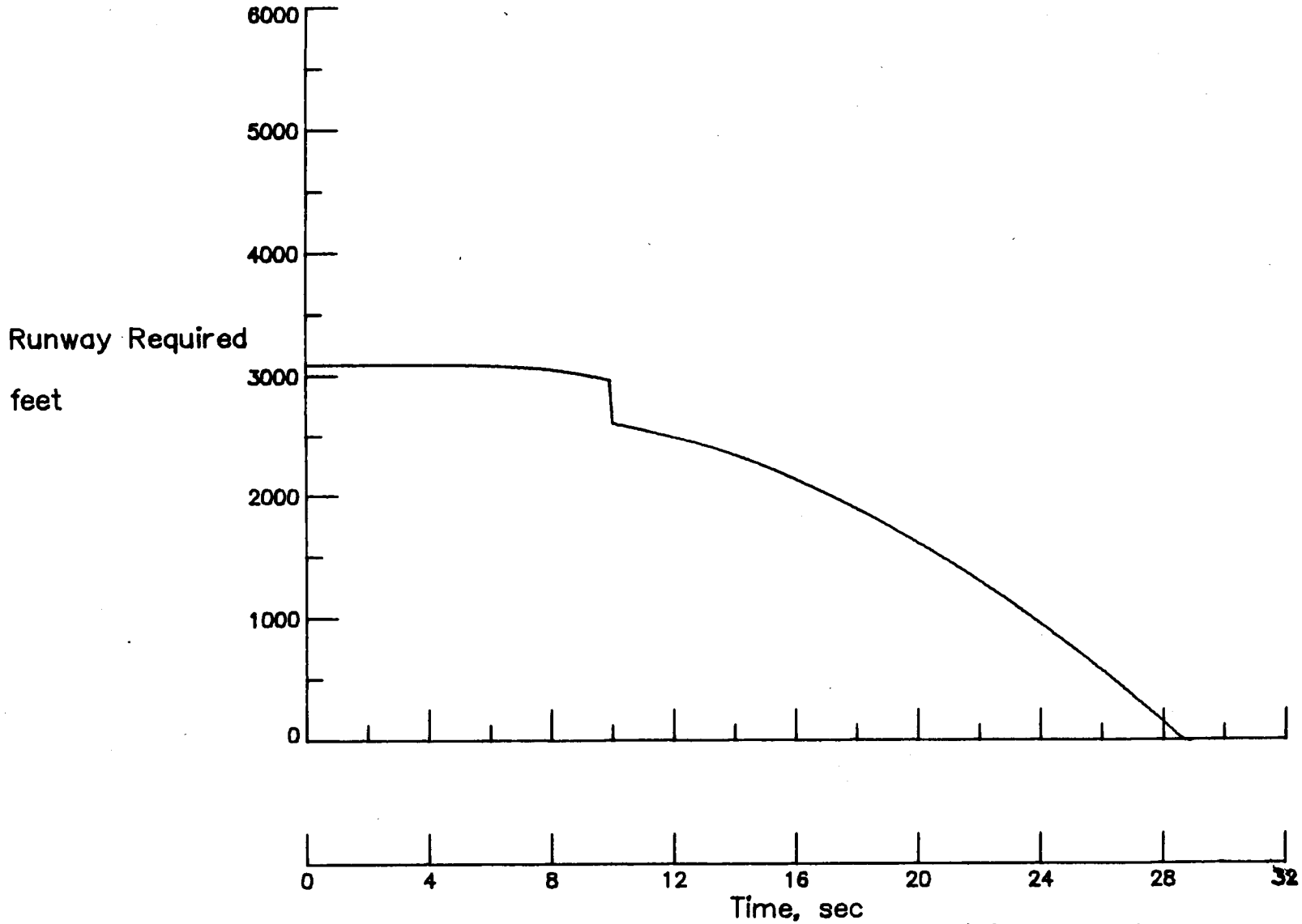

Figure 5.4b: Runway Required Time History for the Weight Error Case - Assumed W = 88,504 lbs, Actual W = 78,504 lbs.

Table 5.5: Effect of Gross Weight Errors

WEIGHT		PERFORMANCE					
ASSUMED WEIGHT (lbs)	ACTUAL WEIGHT (lbs)	MEASURED CALIBRATED AIRSPEED (knots)	PREDICTED RUNWAY REQUIRED (feet)	RUNWAY USED (feet)	ADJUSTED FRICTION COEFF.	PREDIC--TION ERROR (feet)	CHANGE IN PREDIC--TION ERROR (feet)
88504.	78504.	128.1	2747.	2802.	-.021	-55.	75.*
88504.	88504.	128.1	3132.	3262.	0.017	-130.	0.
88504.	98504.	128.5	3510.	3449.	0.048	61.	191.**

* For the actual weight of 78,504 lbs., the flight manual recommended rotation speed is 119 knots. For this run, the output indicates that this speed is reached at 2415 feet of runway used.

** For an actual weight of 98,504 lbs., the flight manual recommended rotation speed is 138 knots.

Table 5.6: Effect of Flap Setting Errors

FLAPS		PERFORMANCE					
ASSUMED SETTING (deg.)	ACTUAL SETTING (deg.)	MEASURED CALIBRATED AIRSPEED (knots)	PREDICTED RUNWAY REQUIRED (feet)	RUNWAY USED (feet)	ADJUSTED FRICTION COEFF.	PREDIC--TION ERROR (feet)	CHANGE IN PREDIC--TION ERROR (feet)
5.	1.	128.7	3123.	3278.	0.017	-155.	-25.*
5.	5.	128.1	3132.	3262.	0.017	-130.	0.
5.	15.	128.5	3121.	3265.	0.017	-144.	-14.**

* For the actual flap setting of 1 °, the flight manual recommended rotation speed is 133 knots.

** For an actual flap setting of 15 °, the flight manual recommended rotation speed is 123 knots. For this run, the output indicates that this speed is reached at 2828 feet of runway used.

increasing the drag coefficient. Two such cases are considered here. The first involves a 10% reduction in the lift coefficient and simultaneous 10% increase in the drag coefficient. The second case considers the effect of a 15% simultaneous change in the lift and drag coefficients. The results are summarized in Table 5.7. The effect of the increase in the drag coefficient

is accommodated but the effect of the reduction in the lift coefficient on the rotation speed of the airplane is not. This effect can be approximated as follows:

$$V_{R_{new}} = V_{R_{original}} * \sqrt{1 /(1- d)} \qquad \text{-- (5.1)}$$

where $V_{R_{new}}$ = rotation speed with contamination

$V_{R_{original}}$ = rotation speed without contamination

d = fractional reduction in lift coefficient

Table 5.7: Effects of Aerodynamic Degradation

DEGRADATION LEVEL (percentage)	PERFORMANCE					
	MEASURED CALIBRATED AIRSPEED (knots)	PREDICTED RUNWAY REQUIRED (feet)	RUNWAY USED (feet)	ADJUSTED FRICTION COEFF.	PREDIC--TION ERROR (feet)	CHANGE IN PREDIC--TION ERROR (feet)
0.	128.1	3132.	3262.	0.017	-130.	0.
10.	128.1	3134.	3274.	0.018	-140.	-10.
15.	128.8	3135.	3291.	0.018	-156.	-26.

5.2.3 Sensitivity to reduced frequency of calls to the algorithm

In this section the effects of calling the takeoff performance monitoring algorithm 5 times a second, instead of the usual 10 times a second, are investigated. Results from this run are summarized in Table 5.8. The algorithm is seen function under this reduced frequency of use also. Because of the increased time between calls, the speed increase between time steps is also higher and hence the excess speed over the required speed is also higher. This time interval increase also causes a difference in the predicted runway requirements, because of the difference in the integration interval. It is observed from Table 5.8 that when the frequency of calls to the algorithm is reduced from 10 calls per second to 5 calls per second, it has an adverse effect on the estimated runway friction coefficient. The estimated value is seen to go from 0.017 to 0.002 with the simulation using a value of 0.015.

Table 5.8: Effects of frequency of calls to the algorithm

FREQUENCY OF CALLS (per second)	PERFORMANCE					
	MEASURED CALIBRATED AIRSPEED (knots)	PREDICTED RUNWAY REQUIRED (feet)	RUNWAY USED (feet)	ADJUSTED FRICTION COEFF.	PREDIC--TION ERROR (feet)	CHANGE IN PREDIC--TION ERROR (feet)
10	128.1	3132.	3262.	0.017	-130.	0.
5	129.5	3078.	3283.	0.002	-205.	-75.

5.3 Failure Analysis

Effects of sensor failures on the algorithm are considered in this section. The failures are simulated for the accelerometer, the engine pressure ratio sensors, and the ground speed sensor.

The effects of accelerometer failures are considered first. Two types of accelerometer failures are considered. The first biases the accelerometer output by a constant value. Figures 5.5a- 5.5d show the time histories obtained from a run with a bias value of +2.32 feet/second^2. The negative bias superimposed on the measured signal is seen as a standoff between the ideal sensor output (noise and bias free signal) and the raw (unprocessed) sensor output. The bias estimate from the complementary filter is shown in Figure 5.5b. This is the negative of the actual bias. It is seen from this figure that the filter takes about 6 seconds to fully estimate the bias present in the signal. The estimated bias quantity from the complementary filter is added to the measured signal to obtain a bias free, but noisy signal. This noisy signal is processed by a first order lag network and the result is plotted in Figure 5.5c as the filtered acceleration. Initially, the filtered acceleration signal and the ideal sensor value differ by the bias value. As the estimate of the bias improves, the filtered acceleration value approaches the ideal sensor value. The predicted acceleration, it is seen, is very close to the ideal sensor value. Filtered ground speed value is the other output from the complementary filter. Figure 5.5d shows a standoff between the filtered and ideal sensor outputs on the one hand and the predicted value on the other. This is caused in the initial few seconds where the filtered acceleration output is in error. Effects of a bias of -1.68 feet/second^2 are similar but

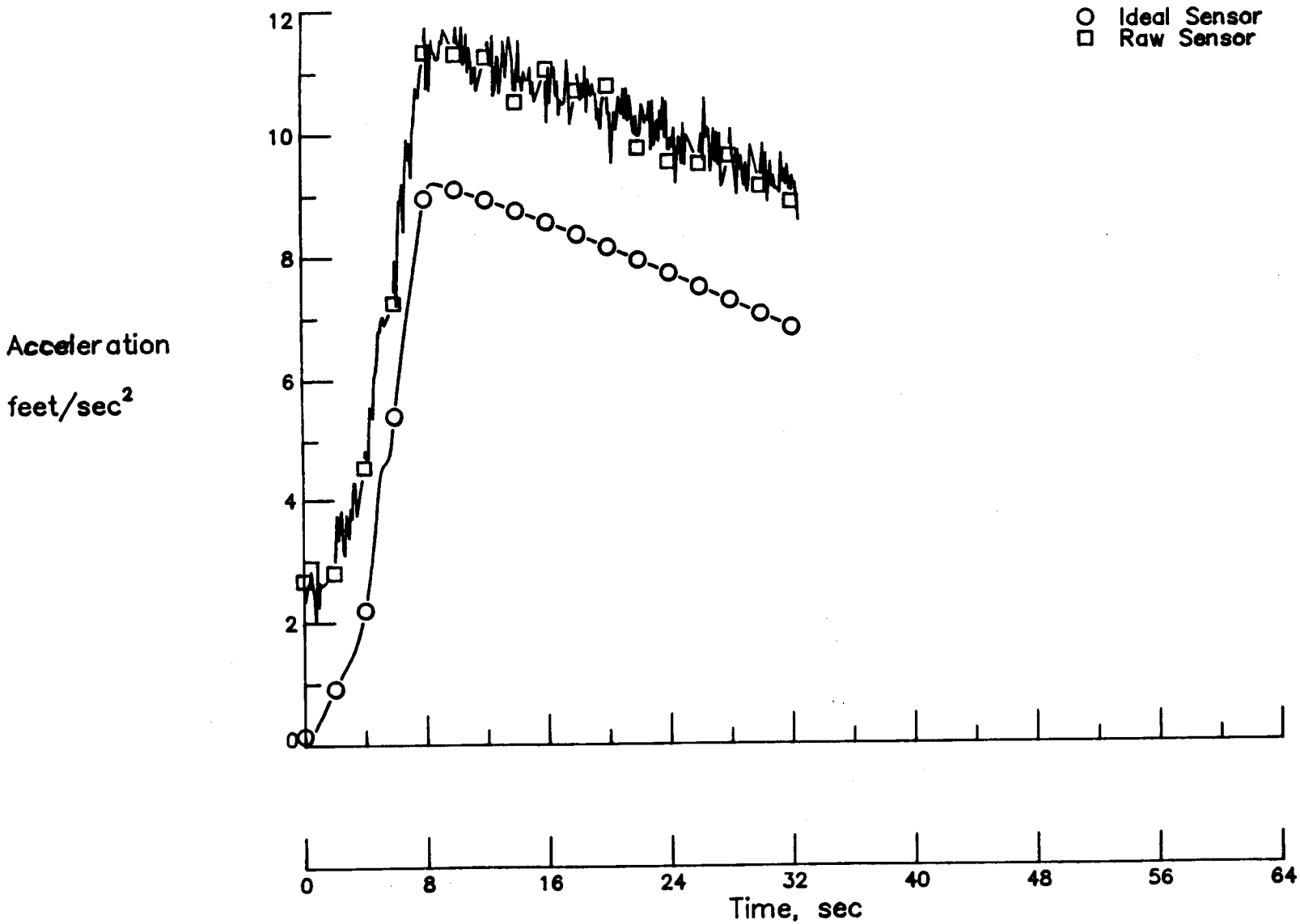

Figure 5.5a: Ideal and Raw Sensor Outputs of Acceleration for an Accelerometer Bias of 2.32 ft/sq. Second.

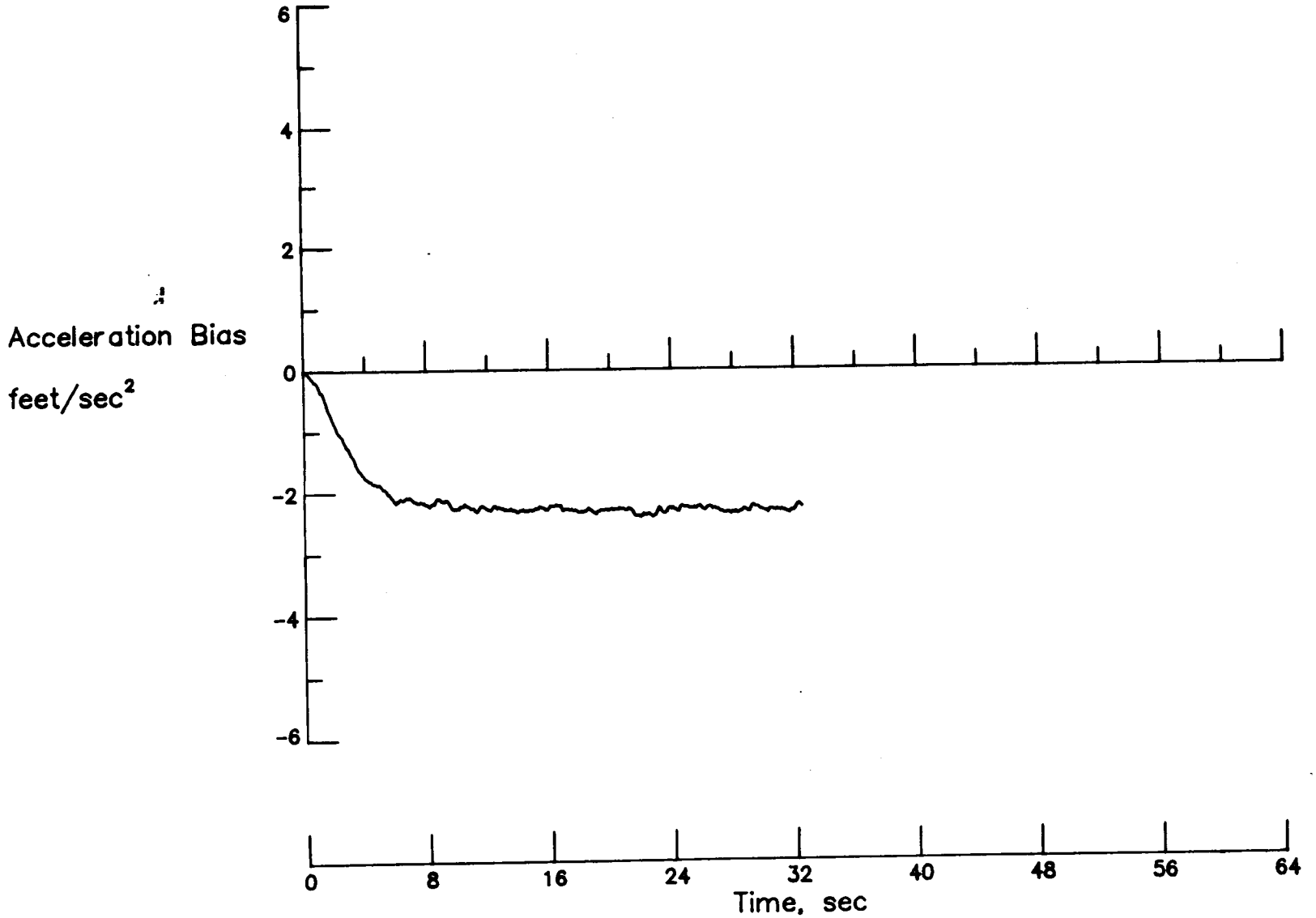

Figure 5.5b: Complementary Filter Estimate of the Negative of Bias for an Accelerometer Bias of 2.32 ft/sq. Second.

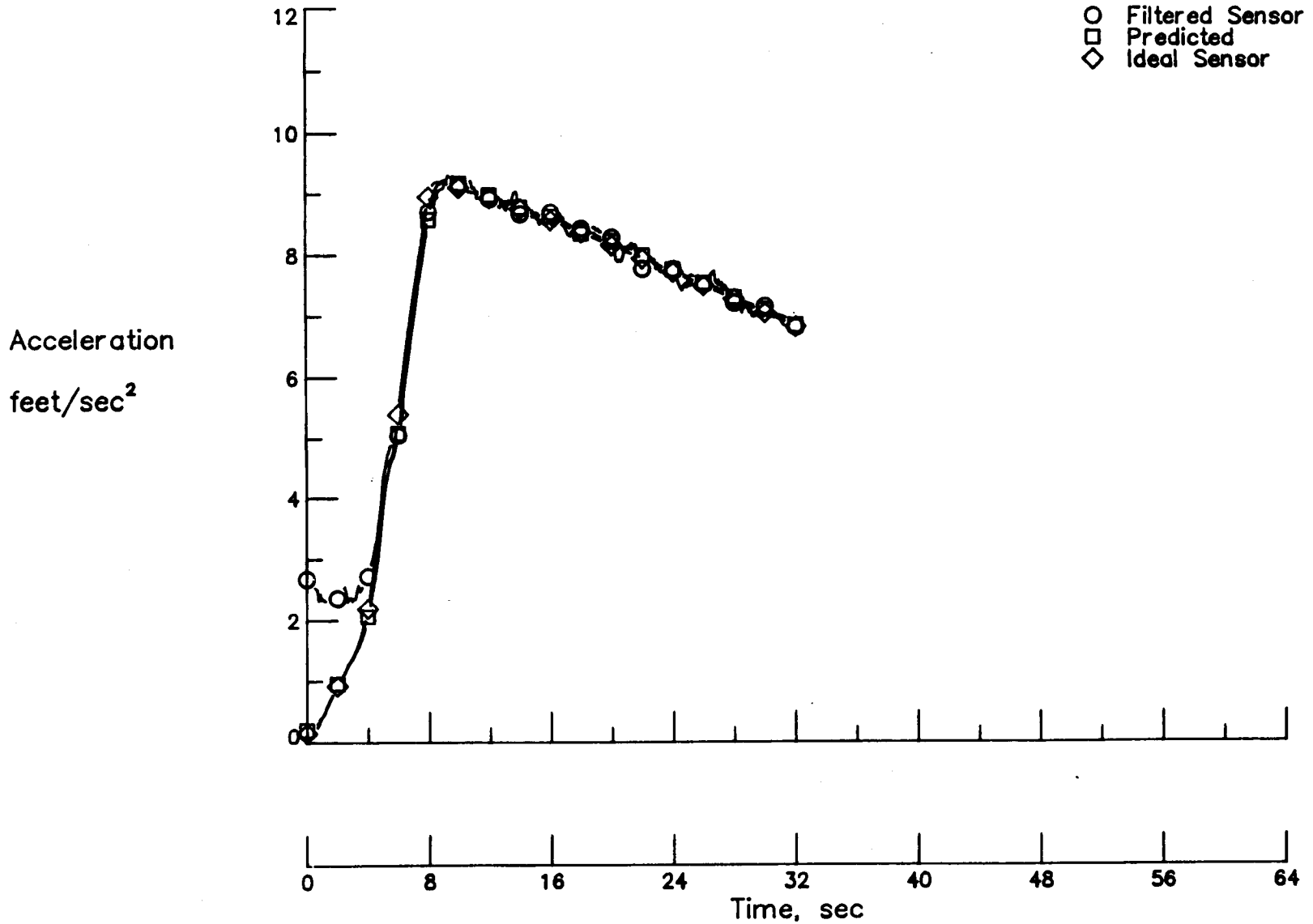

Figure 5.5c: Acceleration Time Histories with an Accelerometer Bias of 2.32 ft/sq. Second.

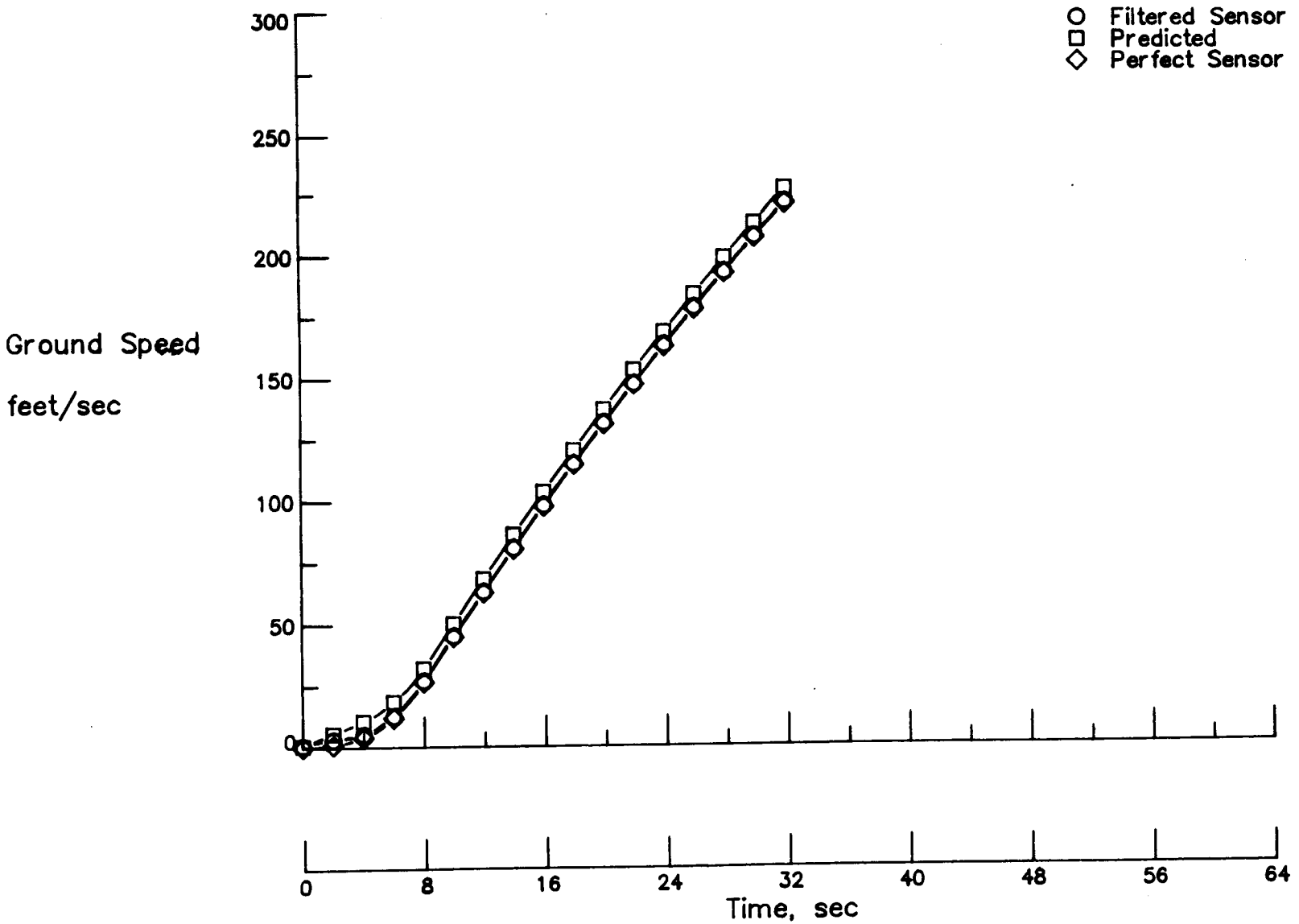

Figure 5.5d: Ground Speed Time Histories for an Accelerometer Bias of 2.32 ft/sq. Second.

in the opposite direction. Results of these two runs are summarized in Table 5.9. It is seen that the algorithm has the capability to estimate such bias values. Abort signals are not generated.

The effect of a scaling failure of the acceleration measurement is considered next. Figures 5.6a - 5.6e are the time histories for a run with a scale factor of 0.85 (reduction of 15%) in the acceleration measurement. The ideal sensor output and the measured (defective) signals are plotted in Figure 5.6a. The bias estimate from the complementary filter is shown in

Table 5.9: Effects of Accelerometer Bias

ACCELEROMETER BIAS	PERFORMANCE					
	MEASURED CALIBRATED AIRSPEED	PREDICTED RUNWAY REQUIRED	RUNWAY USED	ADJUSTED FRICTION COEFF.	PREDIC--TION ERROR	CHANGE IN PREDIC--TION ERROR
($ft./second^2$)	(knots)	(feet)	(feet)		(feet)	(feet)
2.32	128.1	3112.	3262.	0.016	-150.	-20.
0.32	128.1	3132.	3262.	0.017	-130.	0.*
-1.68	128.1	3132.	3262.	0.018	-130.	0.

* Nominal case

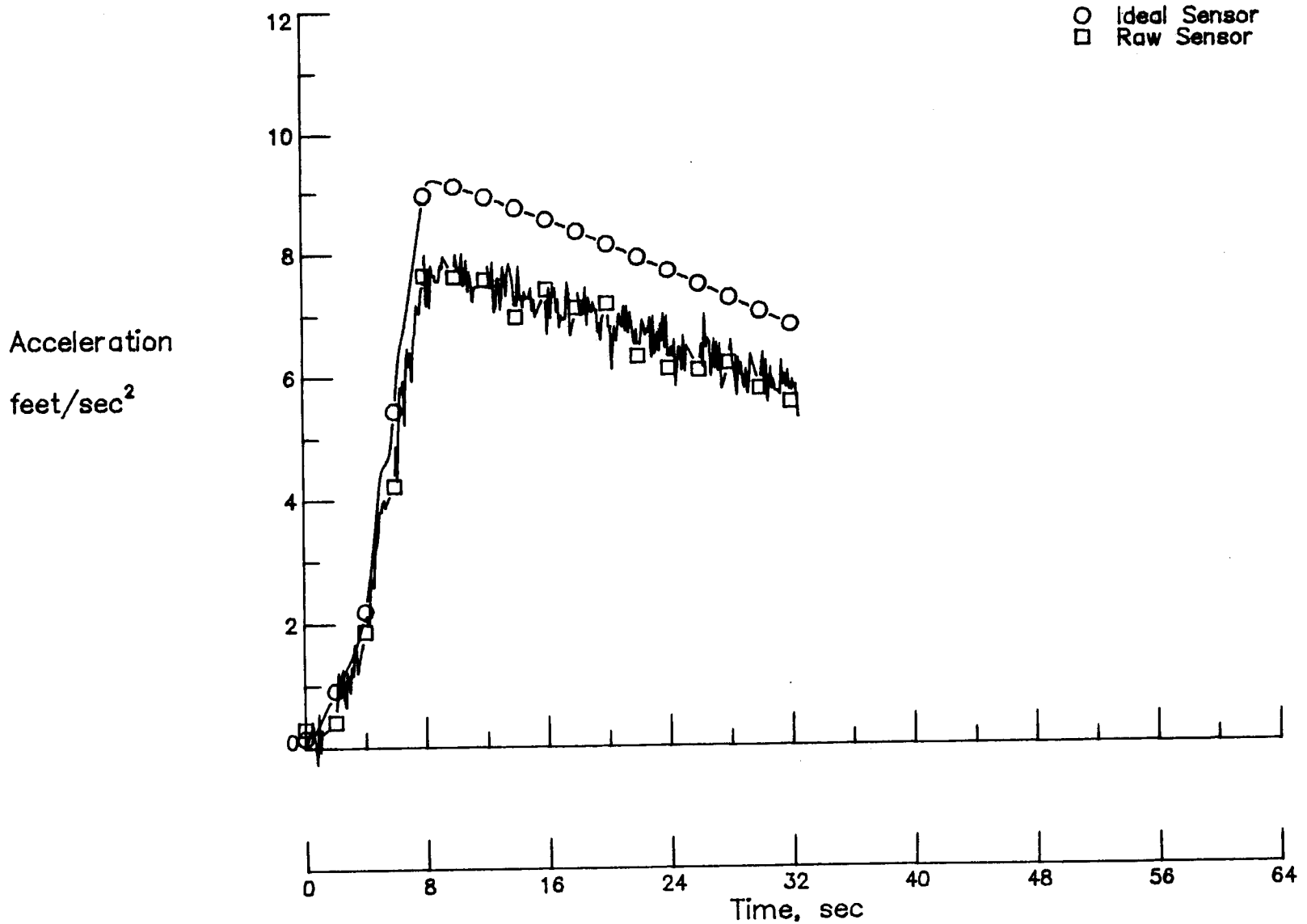

Figure 5.6a: Ideal and Raw Sensor Outputs of Acceleration for an Accelerometer Scale Factor of 85%.

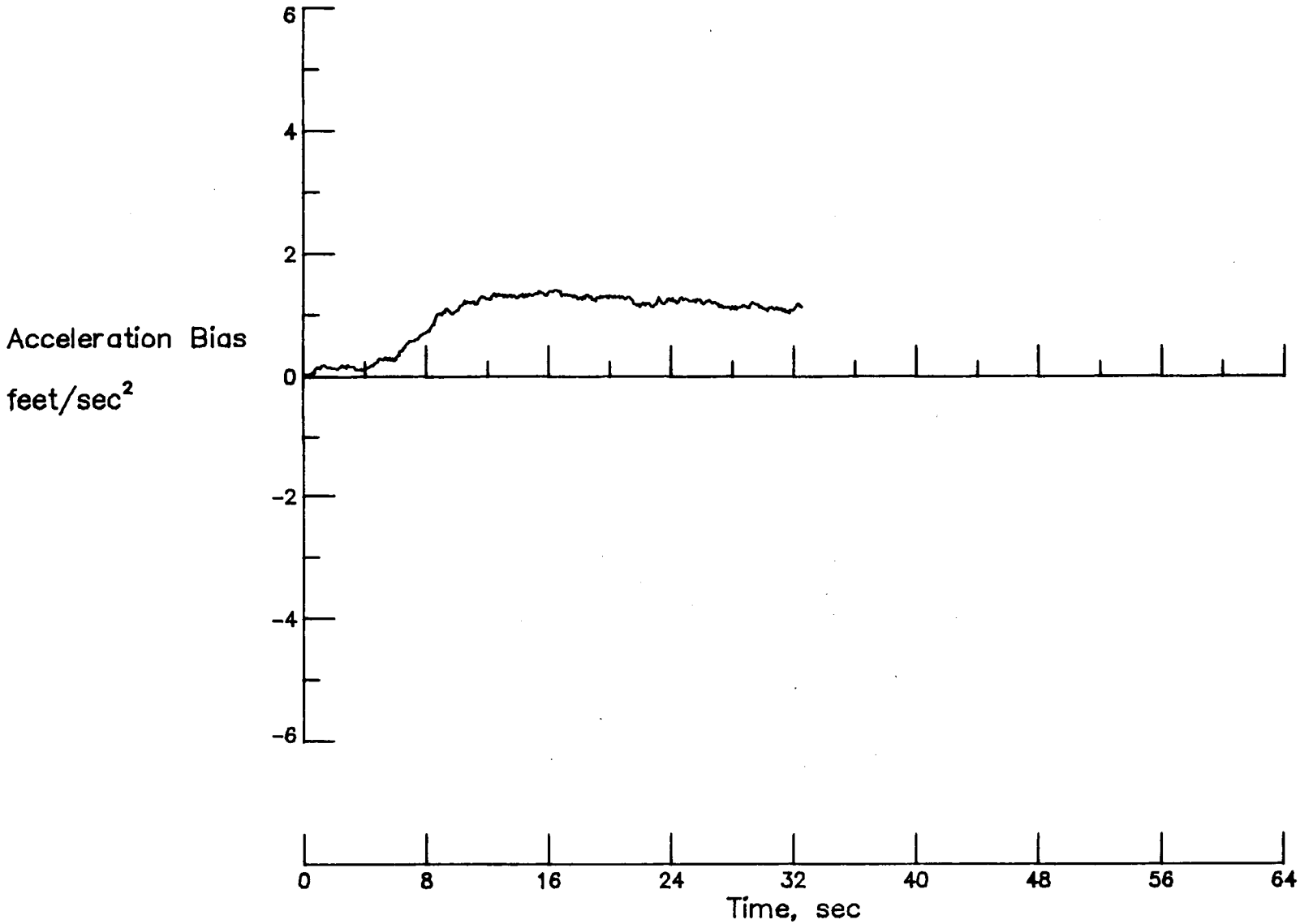

Figure 5.6b: Complementary Filter Estimate of Bias for an Accelerometer Scale Factor of 85%.

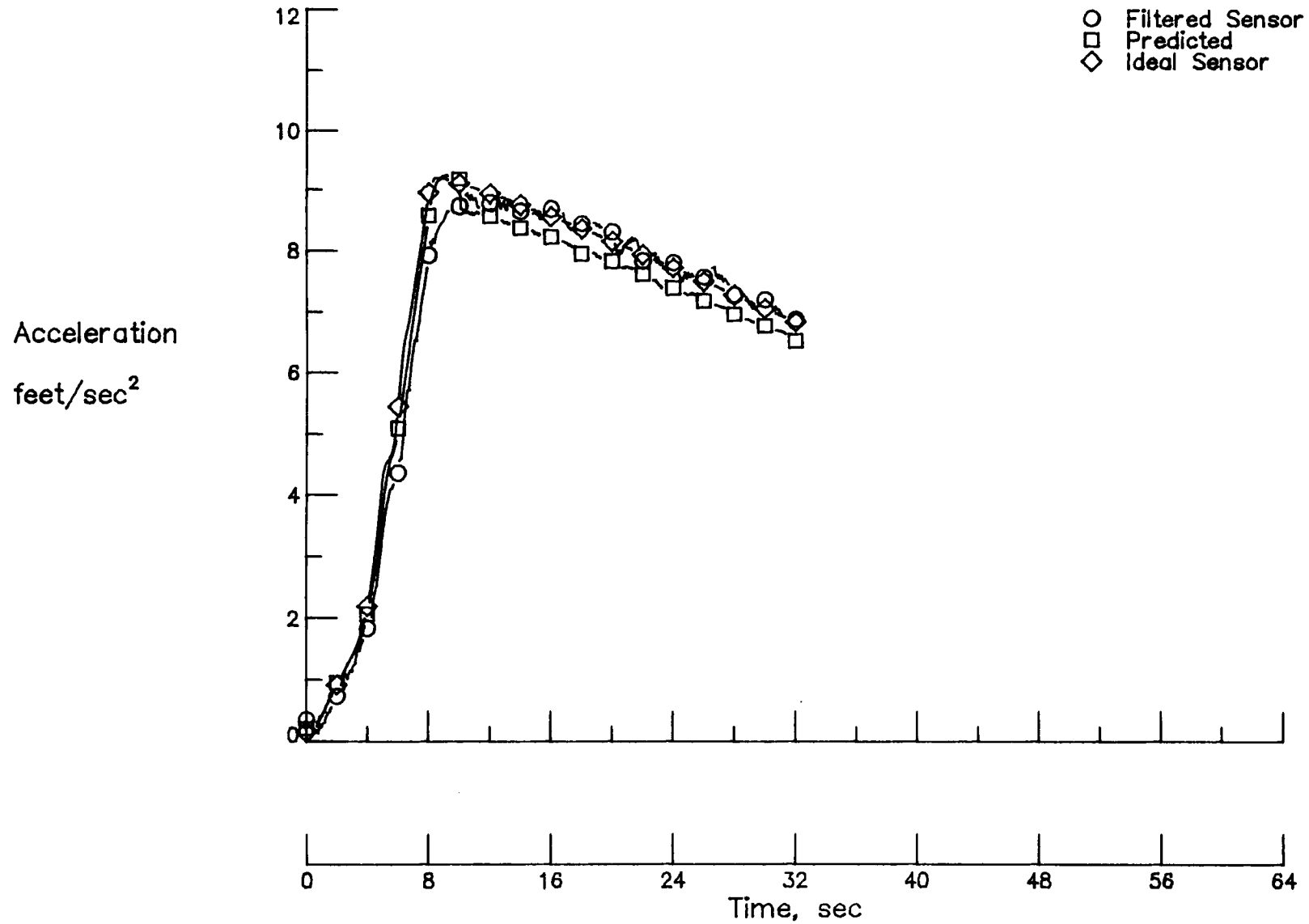

Figure 5.6c: Acceleration Time Histories for an Accelerometer Scale Factor of 85%.

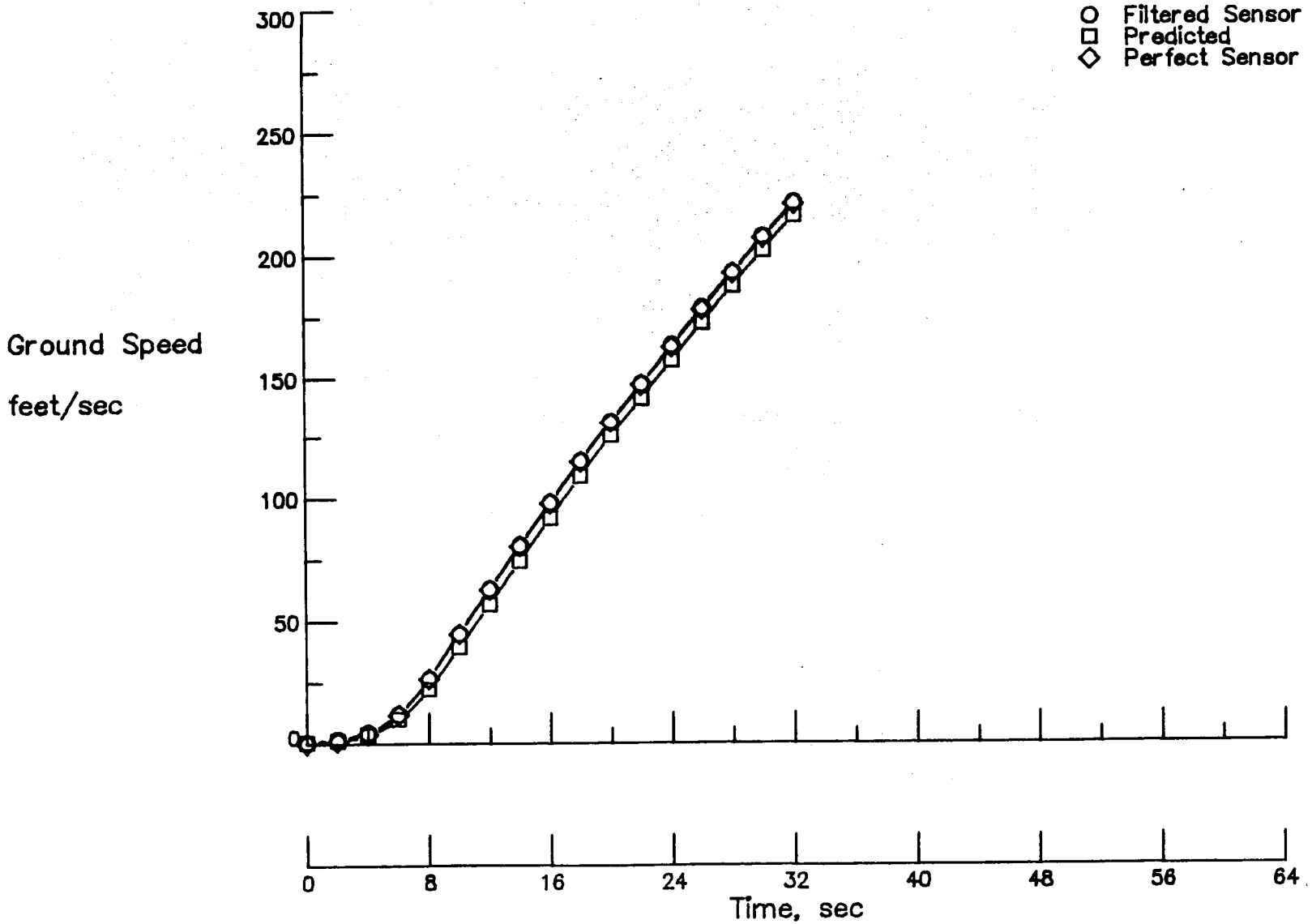

Figure 5.6d: Ground Speed Time Histories for an Accelerometer Scale Factor of 85%.

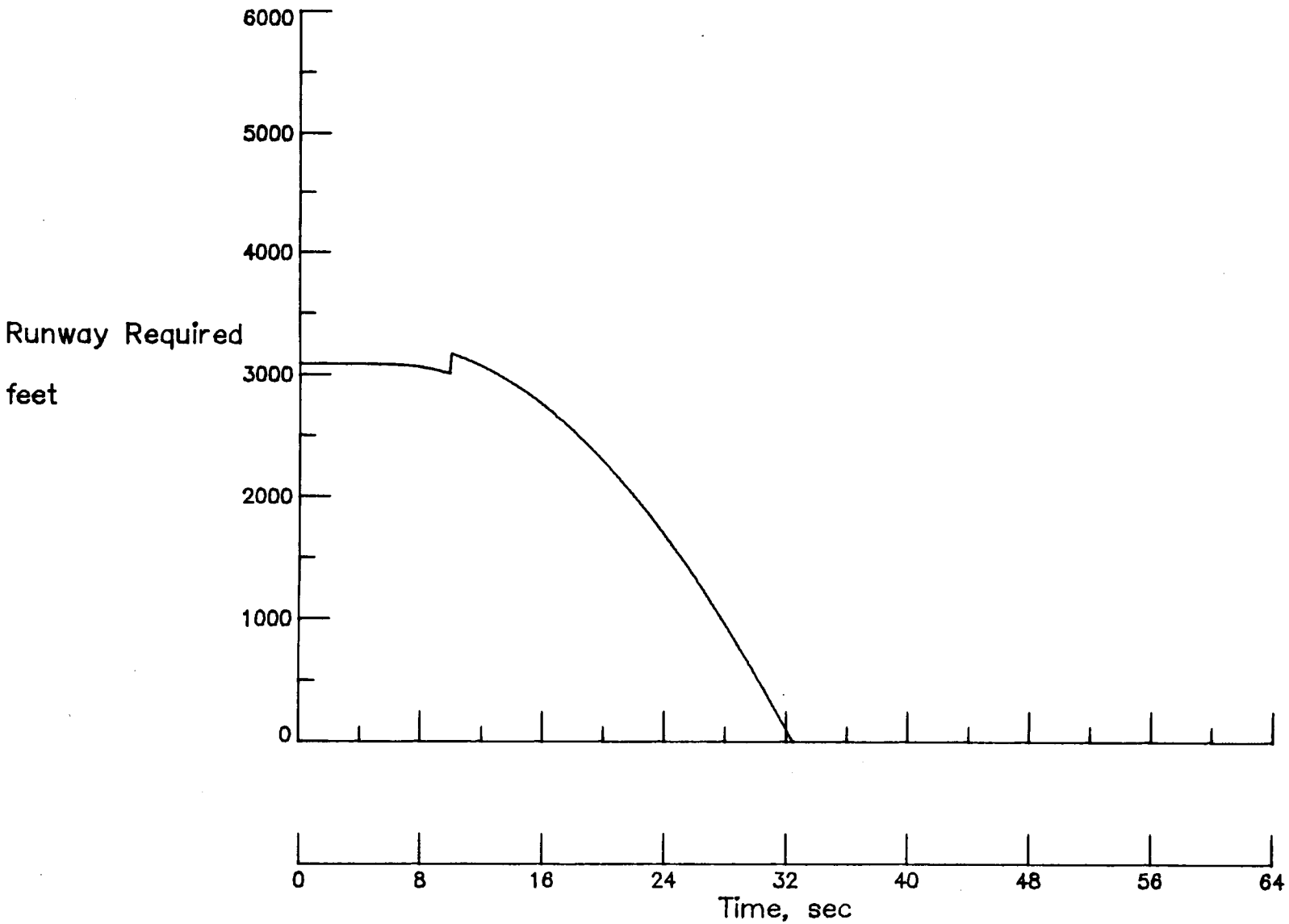

Figure 5.6e: Runway Required Time History for an Accelerometer Scale Factor of 85%.

Figure 5.6b. It is seen from this figure that the complementary filter perceives this scale factor as a bias which changes with time. The bias estimate decreases with time after the 12 second peak. The filtered acceleration, predicted value and the ideal sensor output are plotted in Figure 5.6c. This figure shows the ideal sensor value and the prediction to be very close before the 10 second point. But the filtered value lags the other two. At the 10 second point, the difference between the predicted and the filtered values is used to estimate a new runway friction coefficient. This causes the predicted acceleration value to drop after the 10 second adjustment. In the mean time, the filtered acceleration value increases and comes closer to the ideal sensor output, causing the predicted value to lag behind. This effect is also seen in the ground speed time history plots of Figure 5.6d, where all three values of ground speed start identical, but the predicted output increasingly lags behind the other two. The adjustment in the runway friction coefficient is also clearly seen in the predicted runway required time history plot of Figure 5.6e. The effect of a scale factor error in the opposite direction (115%) produces results in the opposite direction. These results are summarized in Table 5.10. The 115% scale factor error case produces an error in the prediction of runway required of -253 feet. At the beginning, due to the high acceleration (scale factor of 115%) the runway required is predicted to be 3009 feet, but the actual runway used turns out to be 3269 feet. It is seen from Table 5.10 that the estimated friction coefficient after the 10 second point is significantly different from the simulated 0.015, even though within the range of values chosen as acceptable for this study.

Table 5.10: Effects of Accelerometer Scale Factor

ACCELEROMETER SCALE FACTOR (percentage)	PERFORMANCE					
	MEASURED CALIBRATED AIRSPEED (knots)	PREDICTED RUNWAY REQUIRED (feet)	RUNWAY USED (feet)	ADJUSTED FRICTION COEFF.	PREDIC--TION ERROR (feet)	CHANGE IN PREDIC--TION ERROR (feet)
85	128.1	3269.	3262.	0.029	7.	137.
100	128.1	3132.	3262.	0.017	-130.	0.*
115	128.1	3009.	3262.	0.006	-253.	-123.

* Nominal case

Introducing a bias error of -0.3 in the engine pressure ratio measurement (16% of the nominal EPR of 2.0 subtracted from the nominal sensor bias of 0.02) causes an engine failure flag to be set at the 10 second point and hence an abort signal to be generated. If command generation is enabled, full brakes are commanded and the throttle is commanded to full reverse thrust position. An EPR bias of +0.34 (a bias of 0.32 added to the nominal sensor bias of 0.02) also causes identical actions to be taken. Table 5.11 summarizes the results of these two cases.

The other simulated EPR sensor failure pertains to scale factor error caused by the sensor. Two scale factor errors are simulated here. In the first case the sensor reads 15% higher than actual (sensor output is 115% of

Table 5.11: Effect of EPR sensor Bias

EPR BIAS (-)	PERFORMANCE FAILURE FLAG	ENGINE FAILURE FLAG	TIME (seconds)
-0.30	-	Set	10.1
0.02	-	-	- *
+0.34	-	Set	10.1

* Nominal case

actual). In the second case the sensor reads 15% lower than actual (sensor output is 85% of actual). In both cases, the algorithm sets an engine failure flag at 10.1 seconds. Both cases result in the generation of Abort signals. This condition, with the command generation turned on, initiates the abort command procedure. Table 5.12 summarizes these results.

The next sensor considered here is the ground speed sensor. The output from this sensor is forced to remain at a constant level and its effect on the algorithm investigated. The three different values considered for the sensor output are 0, 100, and 250 feet/second. All three cases result in a performance failure flag being set at 10.1 seconds and hence an abort signal being generated. These results are summarized in Table 5.13.

Figures 5.7a - 5.7d are time histories obtained for a ground speed sensor output of 250 feet/second with command generation enabled. The estimate of the ground speed error from the complementary filter is plotted in Figure 5.7a.

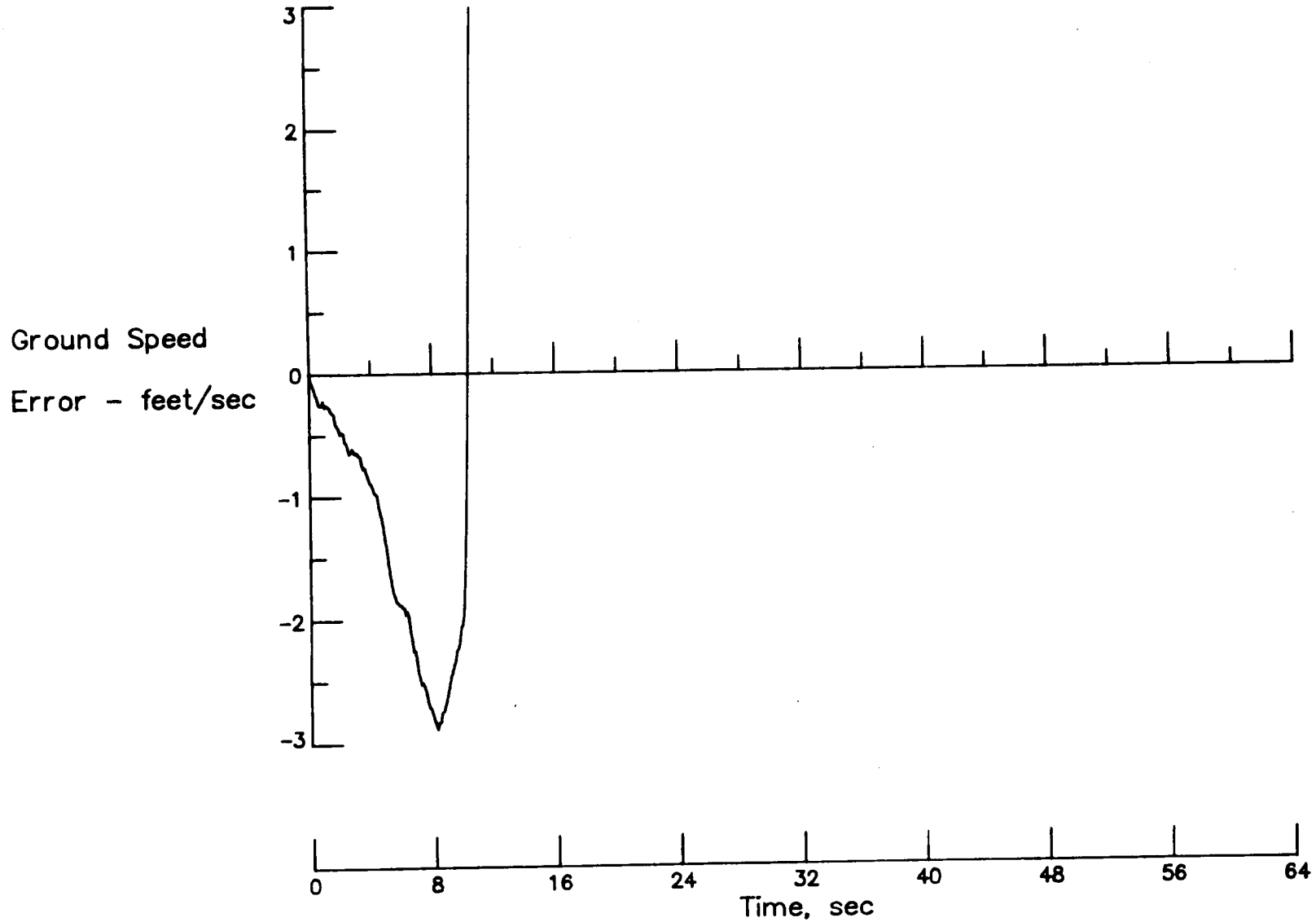

Figure 5.7a: Fstimate of the Ground Speed Error from the Complementary Filter for a Ground Speed Sensor Output of 250 feet/second.

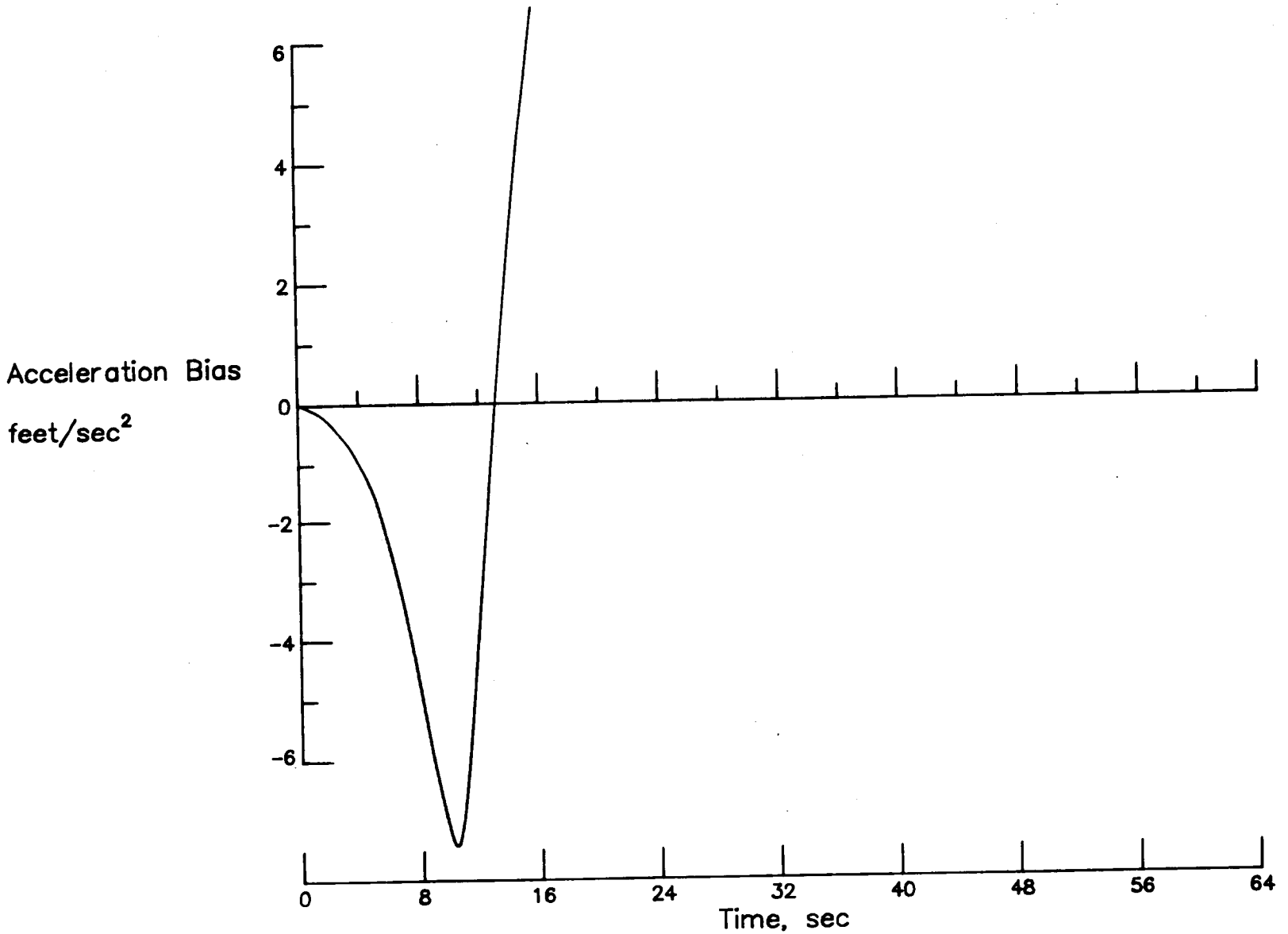

Figure 5.7b: Acceleration Bias Estimated by the Complementary Filter for a Ground Speed Sensor Output of 250 feet/second.

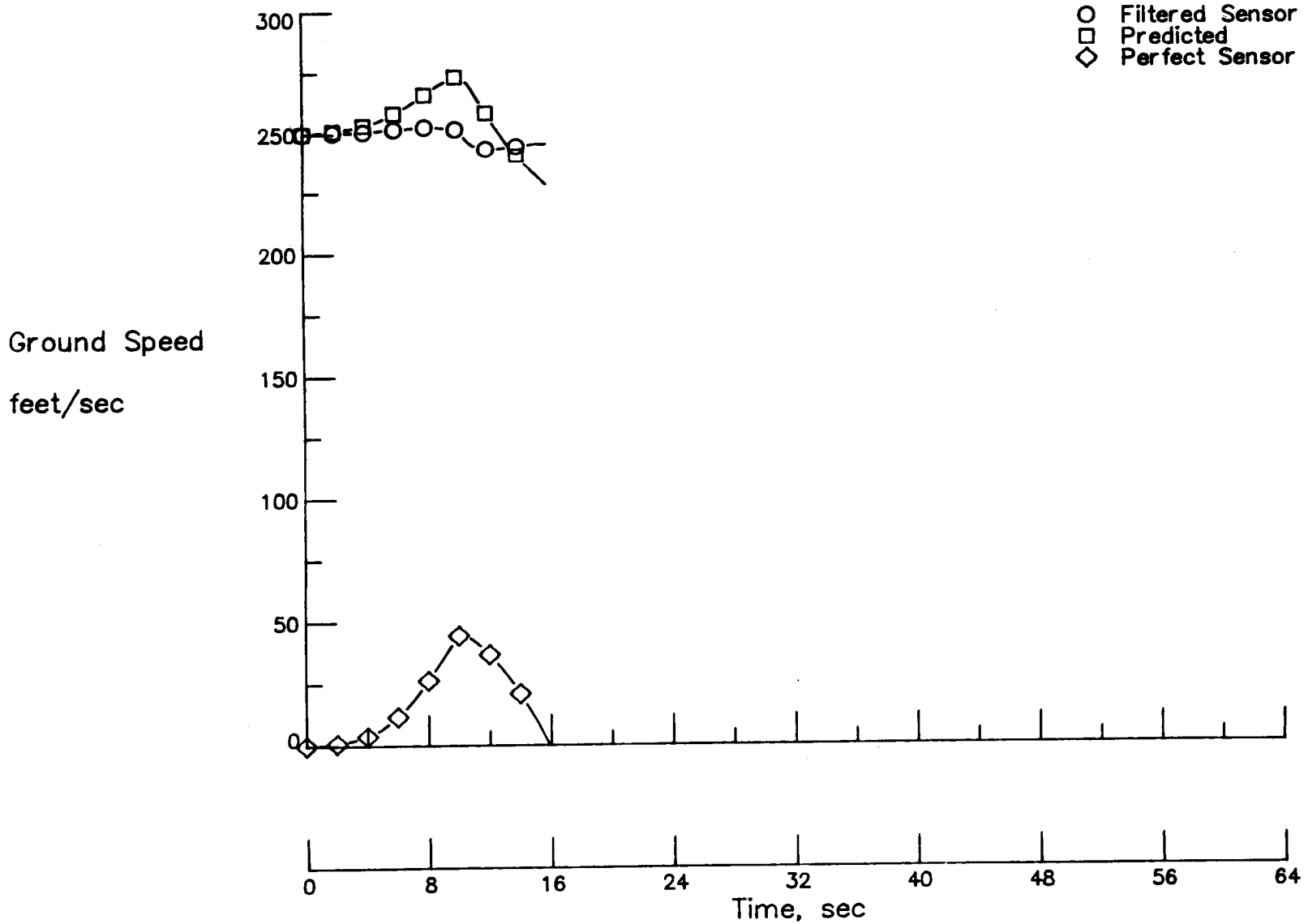

Figure 5.7c: Ground Speed Time Histories for a Ground Speed Sensor Output of 250 feet/second.

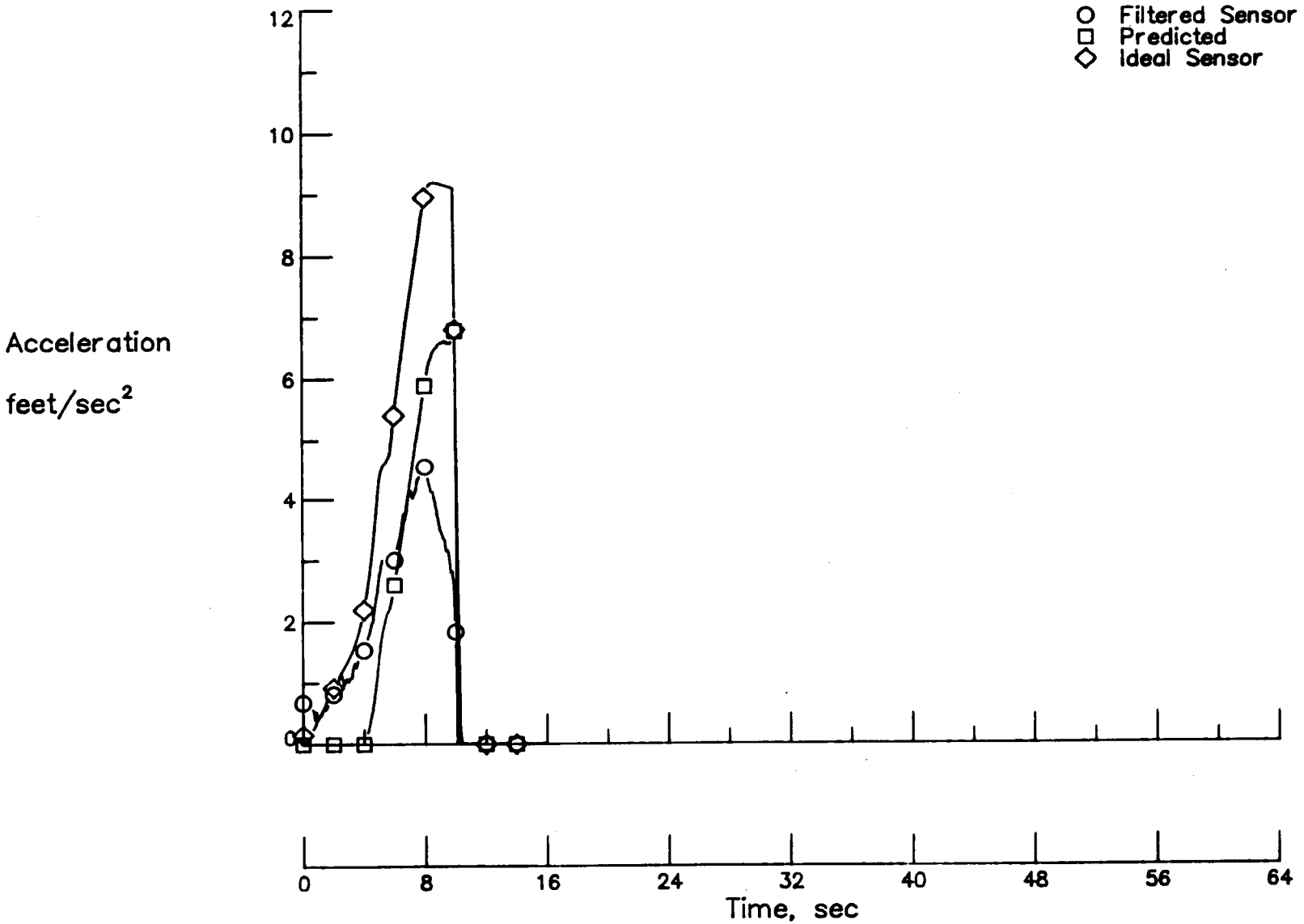

Figure 5.7d: Acceleration Time Histories for a Ground Speed Sensor Output of 250 feet/second.

Table 5.12: Effect of EPR sensor Scale Factor

SCALE FACTOR (%)	PERFORMANCE FAILURE FLAG	ENGINE FAILURE FLAG	TIME (seconds)
115	-	Set	10.1
85	-	Set	10.1

Table 5.13: Effect of Ground Speed Sensor Failure

SENSOR OUTPUT (ft./sec)	PERFORMANCE FAILURE FLAG	ENGINE FAILURE FLAG	TIME (seconds)
0.	Set	-	10.1
100.	Set	-	10.1
250.	Set	-	10.1

The estimate of the bias in the acceleration is plotted in Figure 5.7b. These two figures indicate that the complementary filter is unable to handle this type of error. The resulting Ground Speed time histories are shown in Figure 5.7c. The actual ground speed is depicted by the perfect sensor plot, and the filtered value and the predicted are offset by about 250 feet/second. The acceleration time histories are shown in Figure 5.7d.

5.4 Summary

The sensitivity analysis indicates that the algorithm is highly sensitive to errors in runway wind inputs. This points to a need for an onboard wind estimator.

The algorithm is also found to be sensitive to errors in ambient temperature inputs. An error of about 25 °F in the temperature inputs causes errors of about 400 feet in the prediction of runway requirements to achieve rotation speed.

The algorithm is able to adjust its operations by changing the runway friction coefficient to compensate for input errors in gross weight, and flap setting. This is true as long as the error does not shift the airplane to a different rotation speed category in the flight manual.

The effects of aerodynamic degradation are similar to errors in flap setting inputs.

Cutting down the frequency of calls to the algorithm from 10 times a second to 5 times a second causes large errors in the estimated runway friction coefficient. The estimate is changed from 0.017 to 0.002 with the simulation using a value of 0.015.

The algorithm can function well under failures of the acceleration sensors that cause a bias of up to +/- 2 feet /second^2. The algorithm handles failures in the acceleration sensors that cause scale factor errors in the range of 0.85 to 1.15 by changing the friction coefficient value.

Failures of the EPR sensors that cause bias failures of magnitude 0.5 and scale factor errors of 15% cause abort signals, as this is the only source of engine health check.

Failures of the ground speed sensor that cause its output to be remain unchanged also cause abort signals.

CHAPTER 6

CONCLUSIONS AND RECOMMENDATIONS

6.1 CONCLUSIONS

A Takeoff Performance Monitoring Algorithm has been developed and was tested using the six degree of freedom non-linear batch simulation of the Transport Systems Research Vehicle B-737 on the NASA Langley computer network. The conclusions are clustered into cases that resulted in normal takeoffs and those that resulted in abort signals.

6.1 Normal Takeoffs

Ten cases consisting of different ambient and loading conditions were utilized in testing the algorithm. All of these cases resulted in normal takeoffs. The runway required was predicted to within +/- 150 feet.

The engine malfunction test case affecting only the thrust output did not cause an engine failure flag to be set. Since none of the other conditions for aborting the takeoff run were satisfied, the algorithm generated a Go signal. The algorithm compensated for the difference in thrust level by adjusting the runway friction coefficient to a value outside the nominal range of values used in this study.

The algorithm was found to be very sensitive to errors in the runway wind inputs. A 10 knot error in the runway wind input (assumed wind speed 10 knots; actual 0) caused the error in the predicted runway required to change by -620 feet. A 20 knot error (assumed 20 knot; actually no wind) caused the runway prediction error to change by -1046 feet. The addition of a simple runway wind estimator eliminated this sensitivity. Using the estimator a 20 knot error in the runway wind (assumed value of 20 knots when actual 0) resulted in a change in the prediction error of -127 feet.

The algorithm was also found to be sensitive to errors in ambient temperature inputs. A temperature input 25 °F below the actual value (of 50 °F) caused a change in the prediction error of -465 feet. An input which was 25 °F above the actual value (of 50 °F) caused the prediction error to change by 387 feet.

For a gross weight input of 88,504 lbs., which is 10,000 lbs. greater than the actual, the error in runway prediction changed by 75 feet. This change in the prediction error was 191 feet when the weight input of 88,504 lbs. was 10,000 lbs. less than the actual value. The rotation speed choice was made based on the erroneous input to the algorithm. These errors also resulted in an adjusted friction coefficient which was outside the nominal range of values.

Using a 1 degree flap setting instead of the 5 degrees input to the algorithm caused the error in the predicted runway required to change by -25 feet. Using a 15 degree flap setting, for the same assumed value caused the error to change by -14 feet. Again the rotation speeds chosen were based on the input flap setting.

A 10% reduction in the lift coefficient and a 10% increase in the drag coefficient caused errors in the prediction of runway required to change by -10 feet. A 15% change produced error changes of -26 feet. The rotation speeds here were based on the uncontaminated surfaces.

Utilizing the algorithm 5 times a second instead of 10 times a second caused the error in the prediction of runway requirement to change by -75 feet. The runway friction coefficient estimated at the 10 second point turned out to be less than the lower limit of the range of values used in this development.

An accelerometer bias error of 2 feet/second^2 over nominal caused the runway prediction error to change by -20 feet. A -2 feet/second^2 bias (over nominal) did not change the prediction error.

An accelerometer with an output that is 85% of actual caused a change in the runway requirement prediction error of 137 feet. An accelerometer with a 115% output changed the error by -123 feet. In both cases the friction coefficient was changed, but remained within the chosen range.

6.1.2 Aborts

The engine malfunction test case that affected the engine pressure ratio caused engine failure flags to be set and resulted in Abort signals being generated 10.1 seconds into the takeoff run.

Failure of the engine pressure ratio sensor that results in a bias of +/- 0.32 (over nominal), resulted in the engine failure flags being set at 10.1 seconds, and hence an abort signal.

Failures of the engine pressure ratio sensor resulting in a scaling of 85% and 115% also resulted in the engine failure flags being set at 10.1 seconds and hence an abort signal.

Errors in the ground speed sensor that caused constant outputs of 0, 100, and 250 feet/second resulted in performance failure flags being set at 10.1 seconds and thus in abort signals.

6.2 RECOMMENDATIONS

This implementation of the algorithm does not check the friction coefficient estimated at the 10 second point. It is seen that the friction coefficient is changed substantially in the presence of errors in the input gross weight, and in the engine malfunction test case (the case which does not affect the engine pressure ratio). These two cases result in Go signals

from the algorithm even though the takeoffs should have been aborted. The performance of the algorithm could be improved by checking the estimated friction coefficient against a predetermined reasonable range to flag performance deficiencies.

In the current algorithm the runway rolling friction coefficient is estimated at the 10 second point. This time was chosen to allow for the transients caused by a changing throttle position (as it is being commanded to the takeoff setting) to die out. This estimate of the runway rolling friction coefficient is used throughout the remaining part of the takeoff roll. In reality the rolling friction coefficient varies with speed. The validity of estimating the friction coefficient at the 10 second point needs to be investigated.

In computing the runway required to bring the airplane to a complete stop it has been assumed that with the application of brakes, the friction coefficient increases by a constant amount above the prevalent free rolling friction coefficient. This might not hold true, especially for contaminated runways. There is no data available to correlate the free (unbraked) rolling friction coefficient with what can be achieved with braking, with an antiskid mechanism. A literature search failed to come up with any such information. Such information is important to realistically predict required stopping distances.

Another important point needs to be mentioned here. If this system were used as an advisory system then what time delay factor should be introduced in computing the stopping distances? Additional research is required to determine this value. Additional work is also needed to develop formats and methods of displaying this information to the pilot.

This implementation does not use the flight manual provided decision speed (V_1). The concept of V_1 evolved from an engine failure consideration and needs to be addressed from the present context. Another related item is the way in which the runway requirements are specified in the flight manual. The flight manual tabulates only the balanced field length, that is the runway required to either go through rotation and clear a 35 foot obstacle or abort the takeoff at V_1 and bring the airplane to a complete stop. Only the ground roll part of the balanced field length can be used to achieve rotation speed, whereas the full length could be used to bring the airplane to a complete stop. The full potential of the takeoff performance monitoring algorithm can be realized only if such information is available. The margin of safety to be built into these lengths needs to be investigated and decided upon.

LIST OF REFERENCES

1. Minutes of the First Meeting of the SAE Takeoff Performance Monitoring Ad Hoc Committee of the Aircraft Division of the Aerospace Council: May 15-16, 1984, Washington, DC.

2. Small, Maj. J. T.; Feasibility of Using Longitudinal Acceleration (N_x) for Monitoring Takeoff and Stopping Performance from the Cockpit; Proceedings of the Twenty-Seventh Symposium, Beverly Hills, CA, September 28 - October 1, 1983, Pages 143-154; (A84-16157 05-05).

3. Foxworth, T. G.; Marthinsen, H. F.; Another Look at Landing and Stopping Criteria; AIAA Paper No. 74-956; August, 1974.

4. Horne, W. B.; Yager, T. J.; et al; Preliminary Test Results of the Joint FAA-USAF-NASA Runway Research Program, Part II - Traction Measurements of Several Runways under Wet, Snow Covered, and Dry Conditions with a Douglas DC-9, a Diagonal-Braked Vehicle and a Mu-Meter; NASA TM X-73910; 1977.

5. Roskam, Jan; Airplane Flight Dynamics and Automatic Flight Controls Part-I; Roskam Aviation and Engineering Corp.; 1979.

6. Franklin, G. F.; Powell, J. D.; Digital Control of Dynamic Systems; Addison-Wesley; June 1981.

7. Oppenheim, A. V.; Schafer, R. W.; Digital Signal Processing; Prentice-Hall; 1975.

8. Central Computing Complex Document N-3; Mathematical and Statistical Software at Langley; NASA Langley Research Center; April 1984.

9. Lan, C-T., E.; Roskam, J.; Airplane Aerodynamics and Performance; Roskam Aviation and Engineering Corp.; 1981.

10. Pines, Samuel; Terminal Area Automatic Navigation, Guidance, and Control Research Using the Microwave Landing System (MLS): Part 2- RNAV/MLS Transition Problems for Aircraft; NASA Contractor Report 3511; January 1982.

11. Guenther, A. D.; Martin, D. J.; TCV/User Oriented FORTRAN Program for the B-737 Six DOF Dynamic Model; Report No. SP-170-021; Sperry Support Services; March 1981.

12. Etkin, Bernard; Dynamics of Atmospheric Flight; John Wiley & Sons; Sixth Edition, 1967.

APPENDIX A

DISCRETIZATION OF FIRST ORDER SYSTEMS

A.1 Introduction

This appendix describes approaches taken in modelling a first order system in the discrete domain. Section A.2 describes the rules used in the translation of a servo actuator and also a simple first order recursive filter. The sampling interval is represented by T (also ΔT) and the digital equivalents have frequency response characteristics which are periodic in ω with period $2\pi/T$ and the plots have symmetry about the Nyquist frequency of π/T.

A.2 The First Order Lag

The first order lag is represented in the S plane by

$$H(S) = \frac{a}{S+a} \qquad \text{-- (A1)}$$

The technique used to transform this is a pole-zero mapping technique (Reference 8) with some changes.

1) The pole at $s = -a$ is mapped into a pole at $z = e^{ST} = e^{-aT}$.

2) The zero at $s = \infty$ is mapped into a zero at $z = 0$.

Thus

$$H_{pz}(Z) = \frac{KZ}{(Z - e^{-aT})} \qquad \text{-- (A2)}$$

Choosing $S = 0$ and $z = 1$ to match gains, the gain K is computed to be

$$K = 1 - e^{-aT} \qquad \text{-- (A3)}$$

Hence

$$H_{pz}(Z) = \frac{Z(1 - e^{-aT})}{(Z - e^{-aT})} \qquad \text{-- (A4)}$$

Letting $e^{-aT} = \xi$

$$H_{pz}(Z) = \frac{Z(1 - \xi)}{(Z - \xi)} = \frac{1 - \xi}{1 - Z^{-1}\xi} = \frac{Y(Z)}{X(Z)}$$

That is,

$$Y(Z) = Z^{-1}Y(Z)\ \xi + X(Z)\ (1 - \xi)$$

Using the properties of the Z-transform (Reference 9),

$$y(nT) = \xi\ y[(n - 1)T] + (1 - \xi)\ x(nT) \qquad \text{-- (A5)}$$

A.3 A typical implementation

For a cut off frequency of 5 radians/second (a = 5) and a sampling interval of 0.1 second (T = ΔT = 0.1), the following results:

$$H(S) = \frac{5}{(S+5)}$$

Nyquist Frequency = 31.4 radians/second

Period = 62.8 radians/second

$\xi = 0.6065$

$y(nT) = 0.6065\ y[(n-1)T] + 0.3935\ x(nT)$

Figures A1 depict the magnitude and phase plots for the above example.

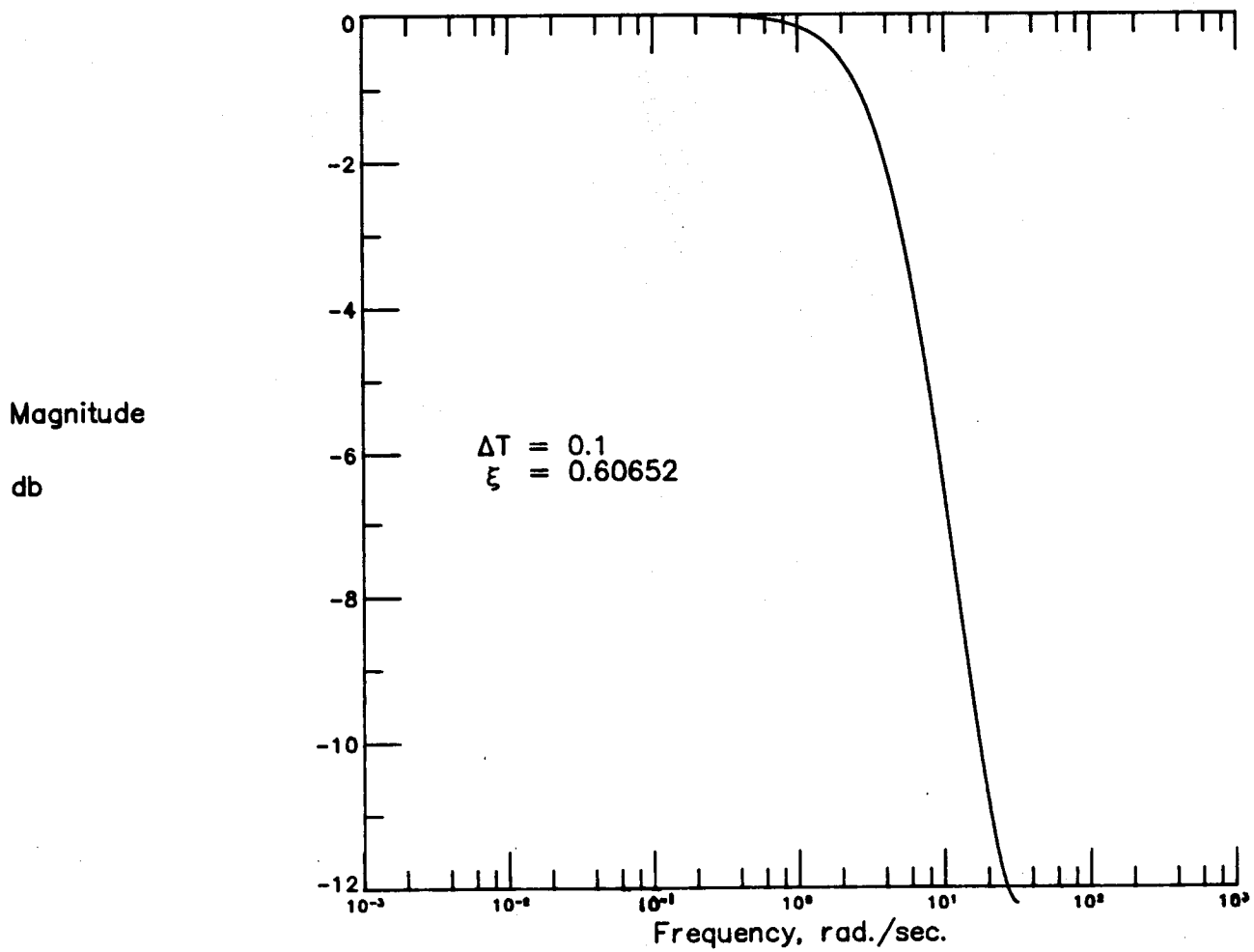

Figure A1: Frequency Response for a First Order Lag.

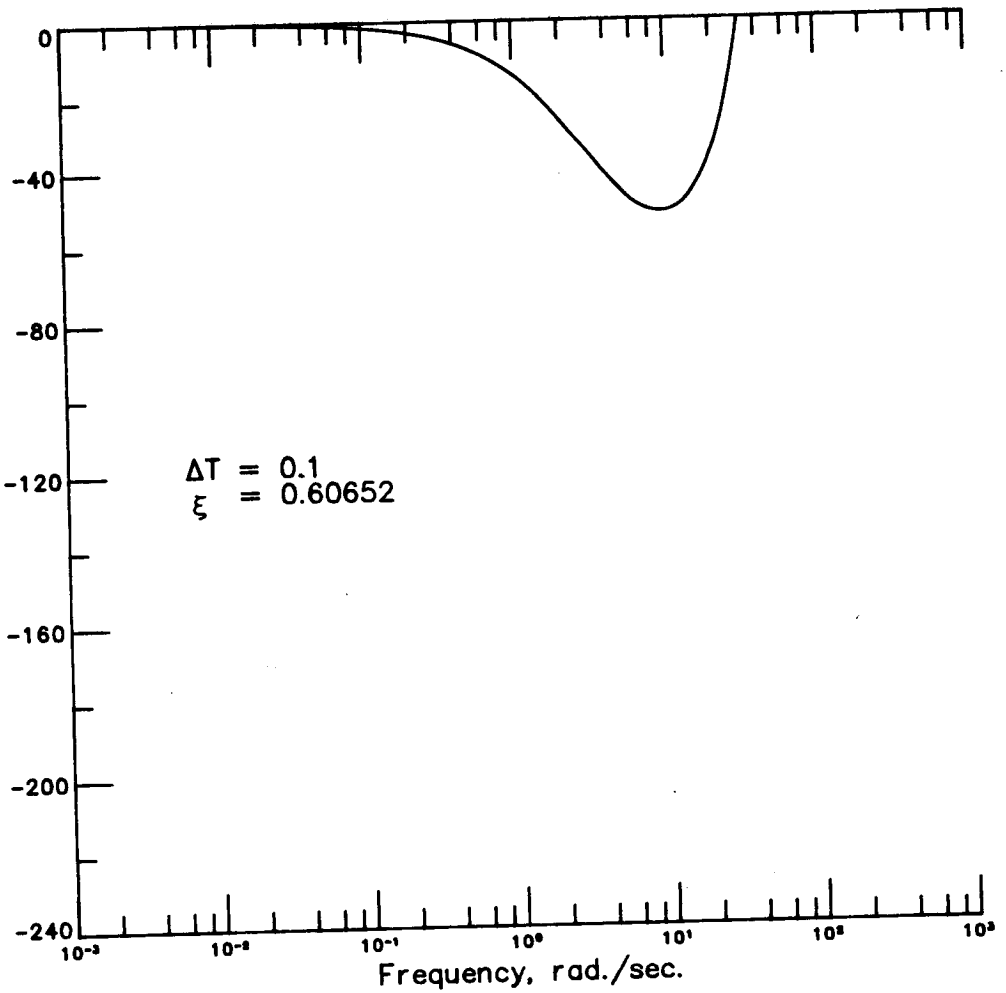

Figure A1: Continued.

APPENDIX B

Formulation of a Second Order Complementary Filter

B.1 Introduction:

This appendix briefly describes the equations involved in the development of a second order complementary filter. The developments of this appendix are based on the methods of Reference 10. The inputs to the filter are the measured along track acceleration of the airplane and its ground speed. The outputs from the filter are the negative of the bias on the acceleration measurement and an estimated ground speed.

B.2 The Formulation:

A block diagram of the second order complementary filter is shown in Figure B.1 . The two states and their derivatives are given by the following set of equations:

$$\dot{x}_1 = K_1(\dot{\bar{x}} - x_1) + \ddot{\hat{x}} \qquad - (B.1)$$

where $\ddot{\hat{x}}$ = estimated acceleration .

$\dot{\bar{x}}$ = measured ground speed

The estimated acceleration in the block diagram is obtained as

$$\ddot{\hat{x}} = \ddot{x} + x_2 \qquad - (B.2)$$

using (B.2) in (B.1),

$$\dot{x}_1 = K_1(\dot{\bar{x}} - x_1) + \ddot{x} + x_2 \qquad - (B.3)$$

The derivative of the second state is given by

$$\dot{x}_2 = K_2(\dot{\bar{x}} - x_1) \qquad - (B.4)$$

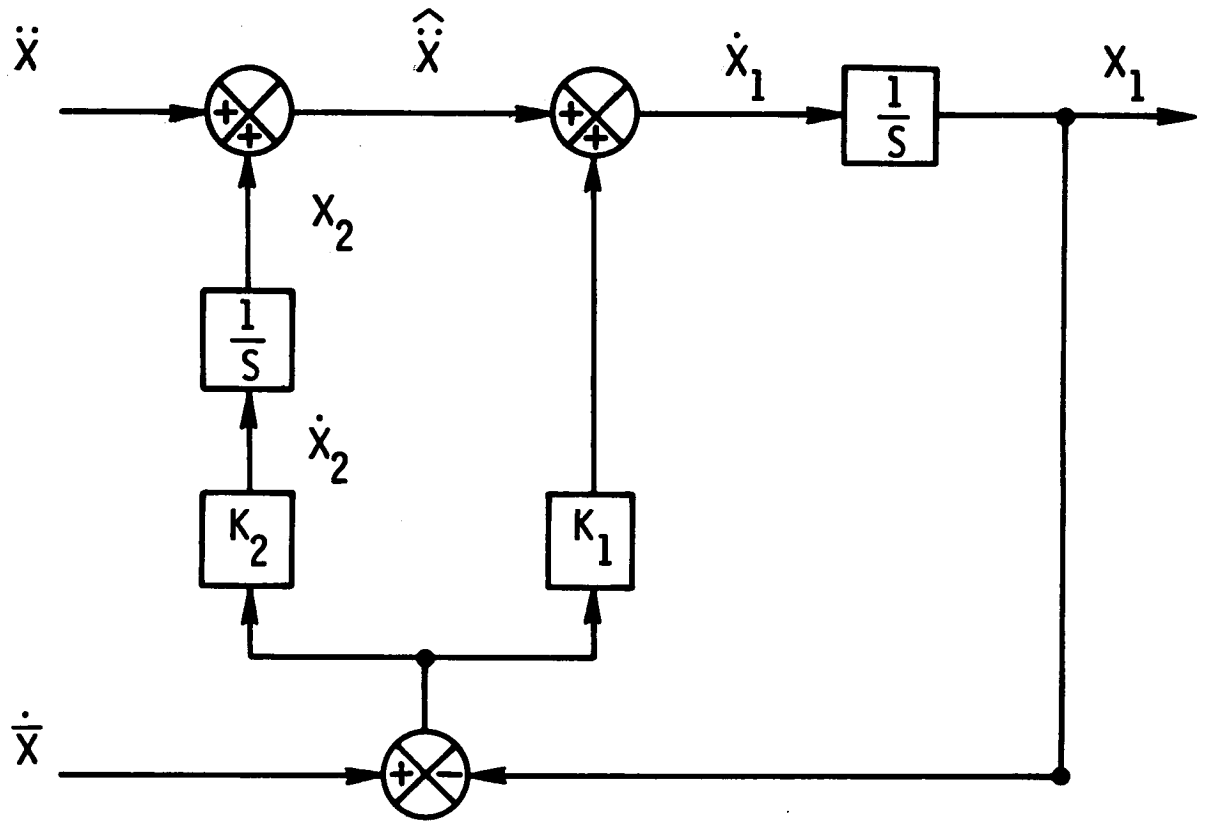

Figure B.1: Block Diagram for the Second Order Complementary Filter.

which can be rewritten as

$$x_1 = \dot{\bar{x}} - \dot{x}_2/K_2 \qquad \text{- (B.5)}$$

Equation (B.5) on differentiation yields

$$\dot{x}_1 = \ddot{\bar{x}} - \ddot{x}_2/K_2 \qquad \text{- (B.6)}$$

Substituting equations (B.5) for x_1, (B.6) for $\dot{x}_1$ in (B.3),

$$\ddot{\bar{x}} - \ddot{x}_2/K_2 = K_1 \, (\dot{x}_2/K_2) + \ddot{x} + x_2$$

Rearranging the terms,

$$\ddot{x}_2 + K_1 \, \dot{x}_2 + K_2 \, x_2 = K_2 \, (\ddot{\bar{x}} - \ddot{x}) \qquad \text{- (B.7)}$$

Due to the discrete nature of the acceleration measurements, the value remains a constant in the time interval $0<\text{time}<\tau$. If the measurement consists of a bias value added on to the actual acceleration then the right hand side of equation (B.7) is seen to be a constant in each time interval. Thus the characteristic equation of the system is given by

$$\ddot{x}_2 + K_1 \, \dot{x}_2 + K_2 \, x_2 = 0 \qquad \text{- (B.8)}$$

This characteristic polynomial can have either two real roots or a complex conjugate pair of roots. Taking the Laplace transform of (B.8) with zero initial conditions, one obtains

$$X(S)\{S^2 + 2\zeta\omega_n \, S + \omega_n^2 \} = 0 \qquad \text{- (B.9)}$$

Comparing terms one obtains:

$$K_1 = 2 \, \zeta \, \omega_n$$

and $\qquad \text{- (B.10)}$

$$K_2 = \omega_n^2$$

B.3 Implementation:

The equation for the continuous implementation of the filter is given by

$$\dot{\underline{x}} = F\,\underline{x} + G\,\underline{u} \qquad \text{- (B.11)}$$

with

$$\underline{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

$$F = \begin{bmatrix} -K_1 & 1 \\ -K_2 & 0 \end{bmatrix} \qquad \text{- (B.12)}$$

$$G = \begin{bmatrix} K_1 & 1 \\ K_2 & 0 \end{bmatrix}$$

where x_1 = filtered speed

x_2 = filter estimate of the negative of the bias

In discrete notation the filter implementation becomes

$$\underline{x}_{n+1} = \Phi\,\underline{x}_n + \Gamma\,\underline{u}_n \qquad \text{- (B.13)}$$

where $\Phi = e^{FT}$ - (B.14)

$$\Gamma = \int_0^T e^{F\eta}\,d\eta\; G \qquad \text{- (B.15)}$$

and T = sampling time interval.

B.4 A numeric example

For a system with $K_1 = 1.5$ and $K_2 = 0.5$ (this results in a characteristic equation with two real roots):

$$F = \begin{bmatrix} -1.5 & 1.0 \\ -0.5 & 0.0 \end{bmatrix}$$

C-3

$$G = \begin{bmatrix} 1.5 & 1.0 \\ 0.5 & 0.0 \end{bmatrix}$$

Thus for T = 0.1

$$\Phi = \begin{bmatrix} 0.8584 & 0.0928 \\ -0.0464 & 0.9976 \end{bmatrix}$$

$$\Gamma = \begin{bmatrix} 0.1416 & 0.0928 \\ 0.0464 & -0.0024 \end{bmatrix}$$

APPENDIX C

PROGRAMMING FLOW CHARTS

This appendix includes programming flow charts for the real-time segment of the algorithm. Section C.1 depicts the flow chart for the overall real-time segment. Subsection C.1.1 contains the flow chart for predicting the runway required to achieve rotation speed. The flow chart for predicting the stopping distance is included in subsection C.1.2

C.1 Real-Time Segment

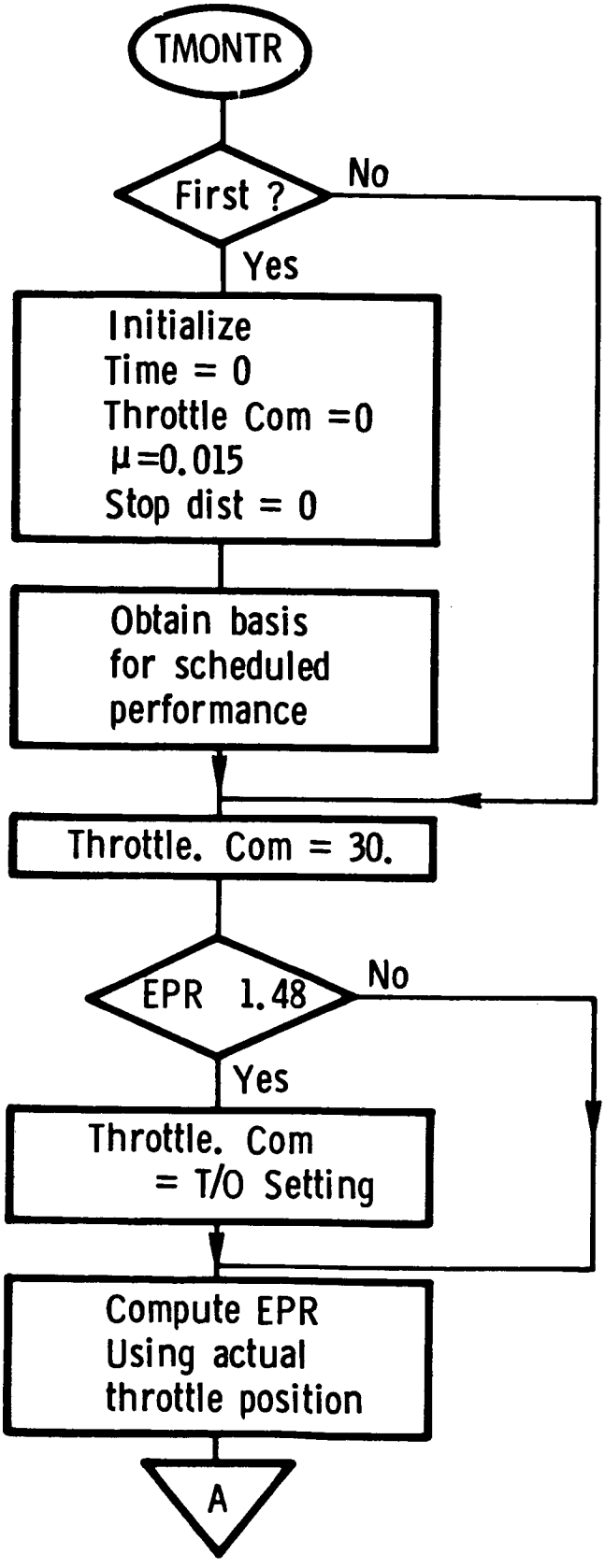

Figure C.1: Programming Flow Chart for the Real-Time Segment

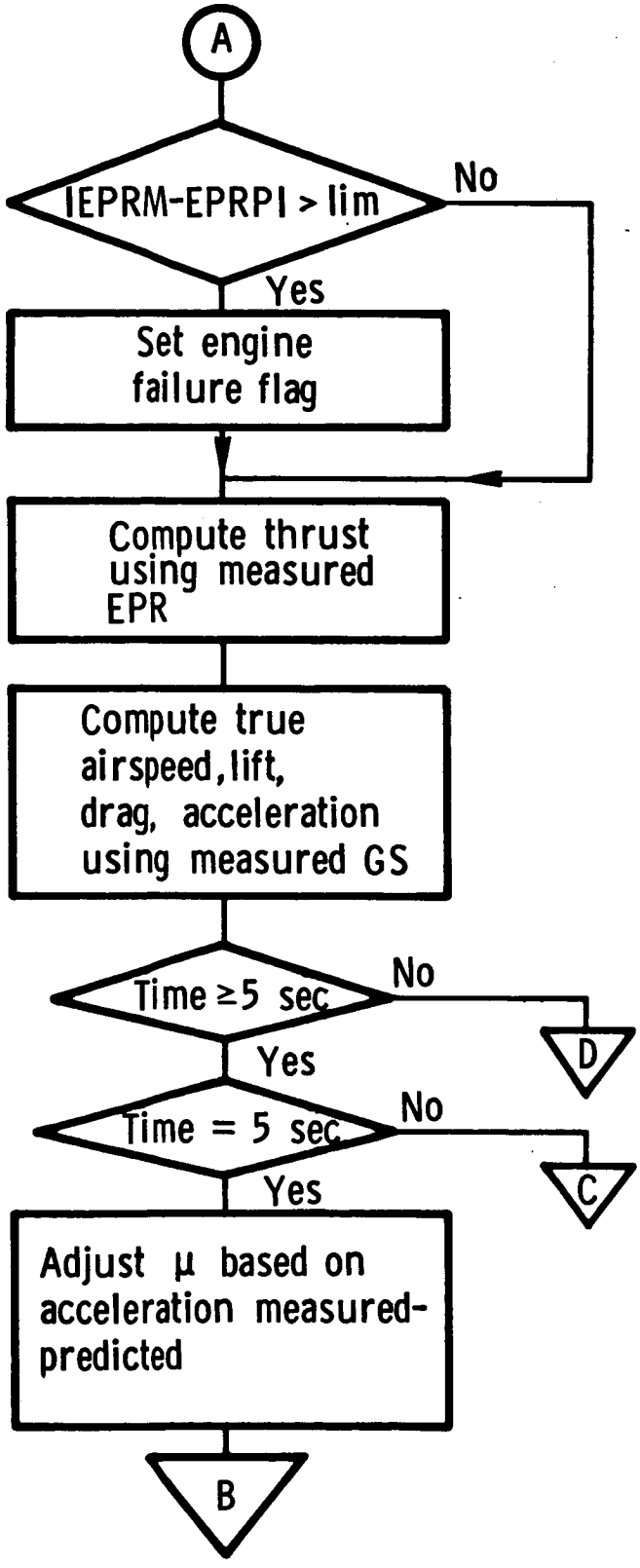

Figure C.1: Continued

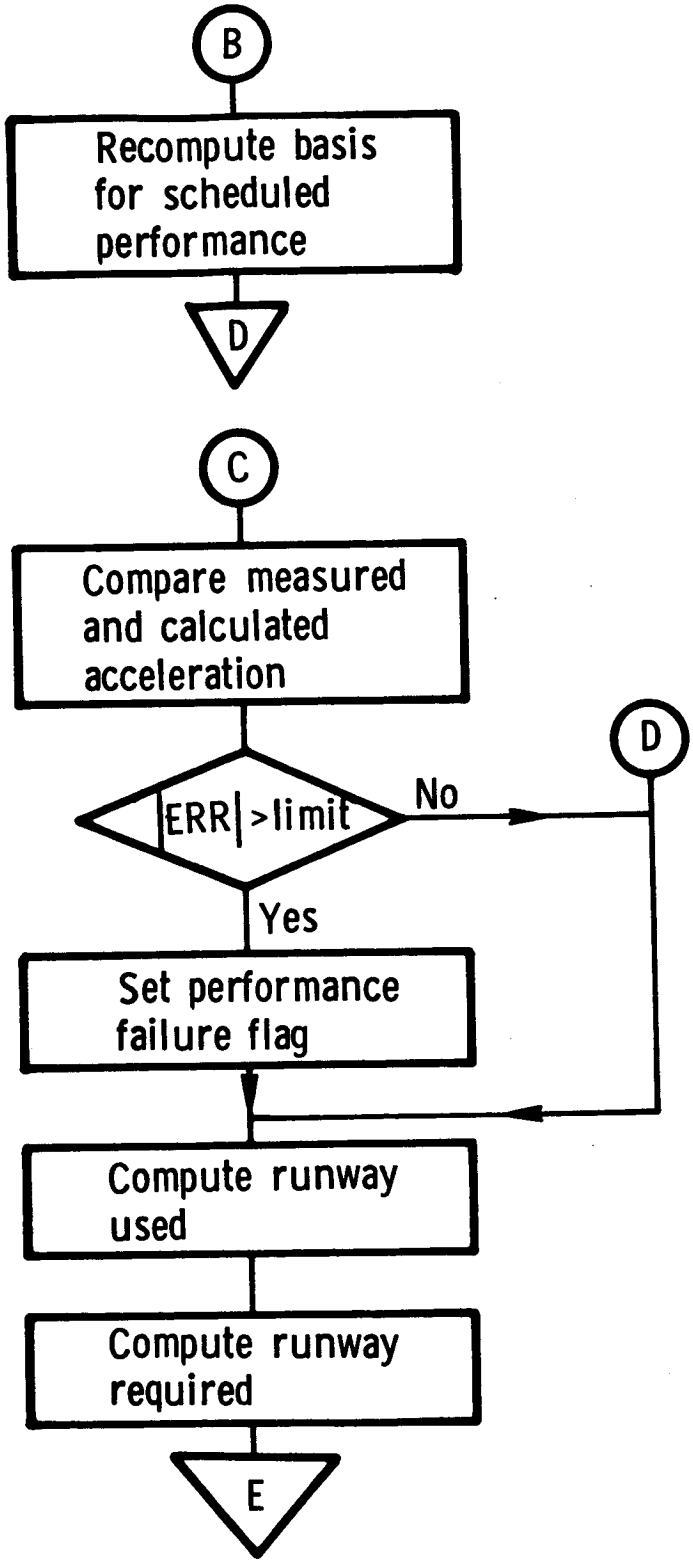

Figure C.1: Continued

C.1.1 Prediction of Runway Required to Achieve Rotation Speed

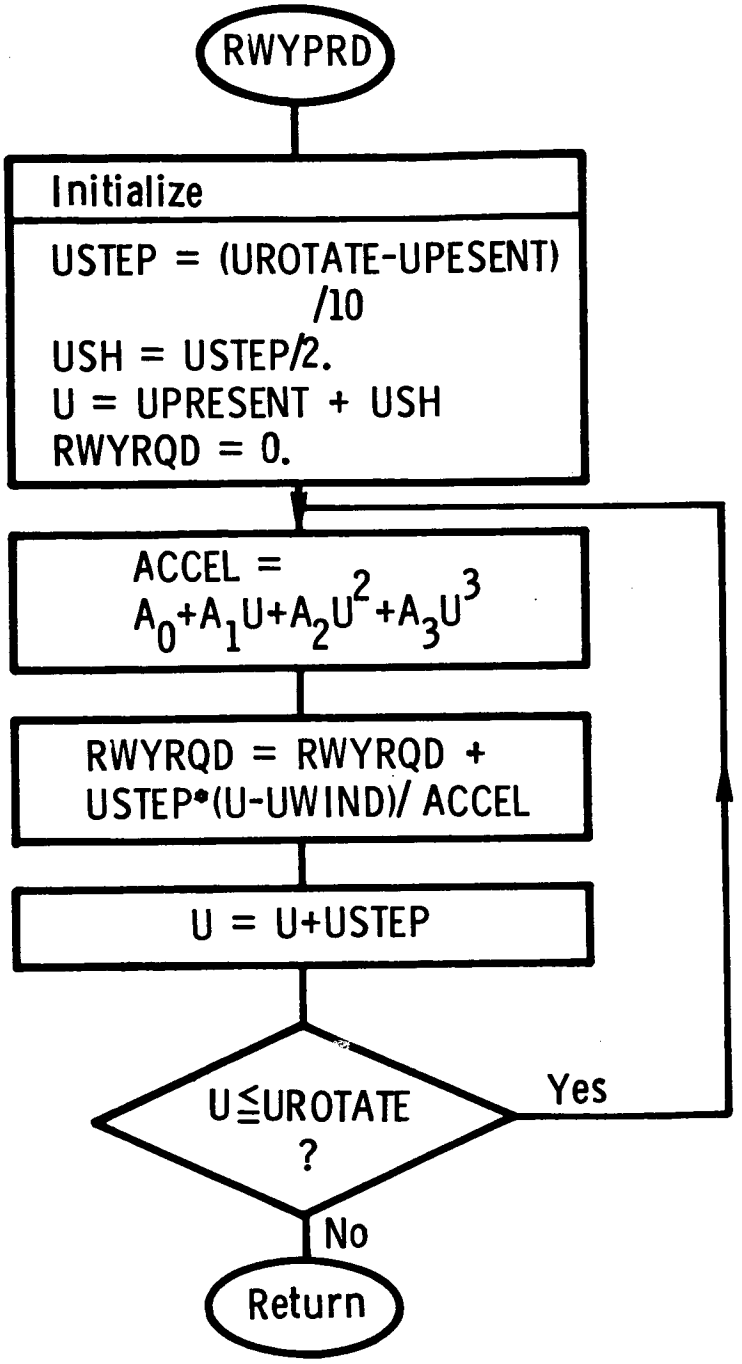

Figure C.2: Programming Flow Chart for Predicting the Runway Needed to Achieve Rotate Speed

C.1.2 Prediction of Stopping Distance

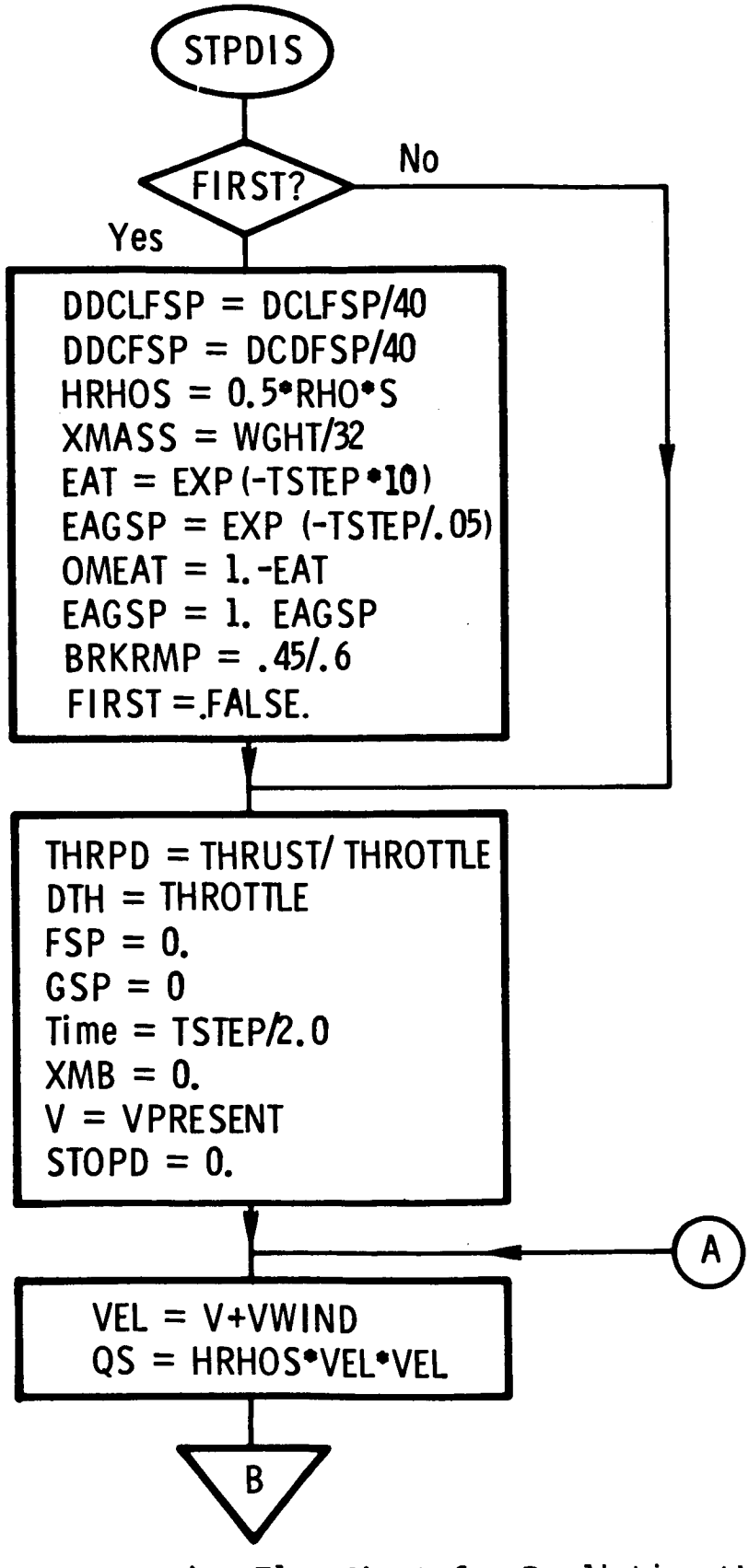

Figure C.3: Programming Flow Chart for Predicting the Stopping Distance

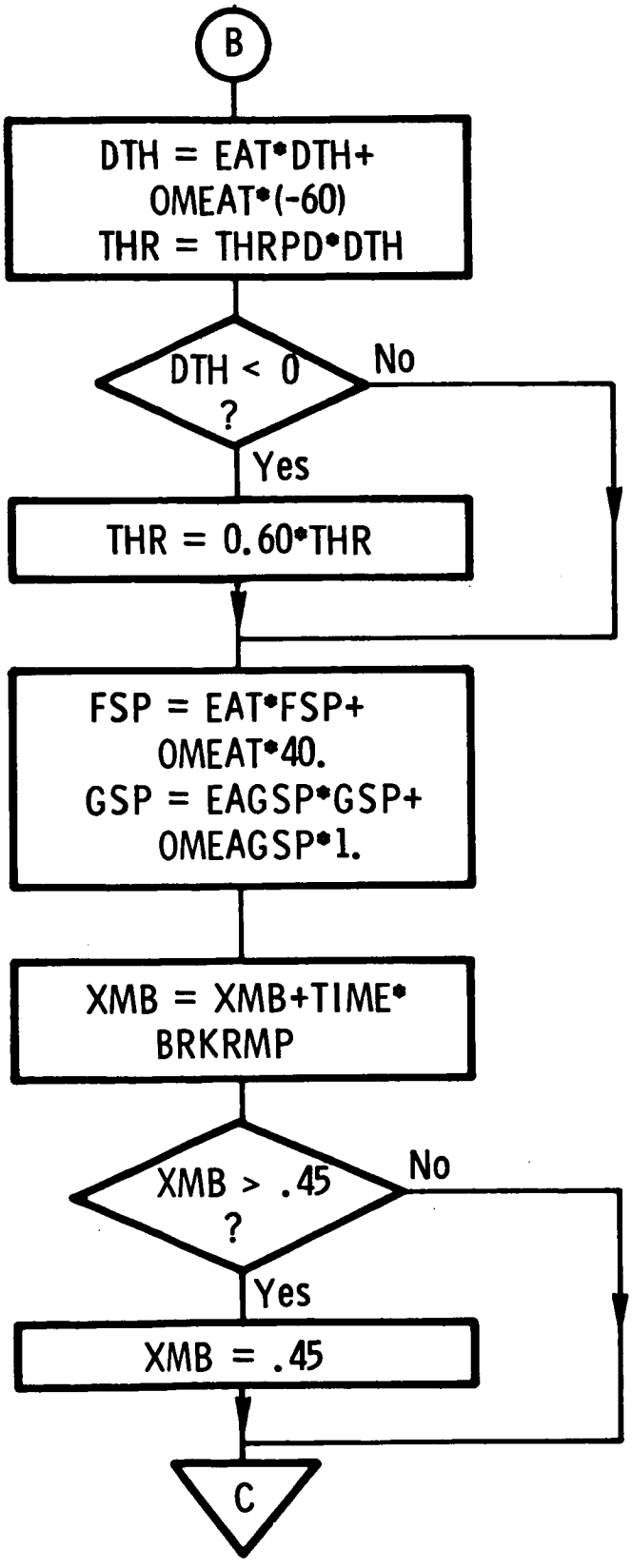

Figure C.3: Continued

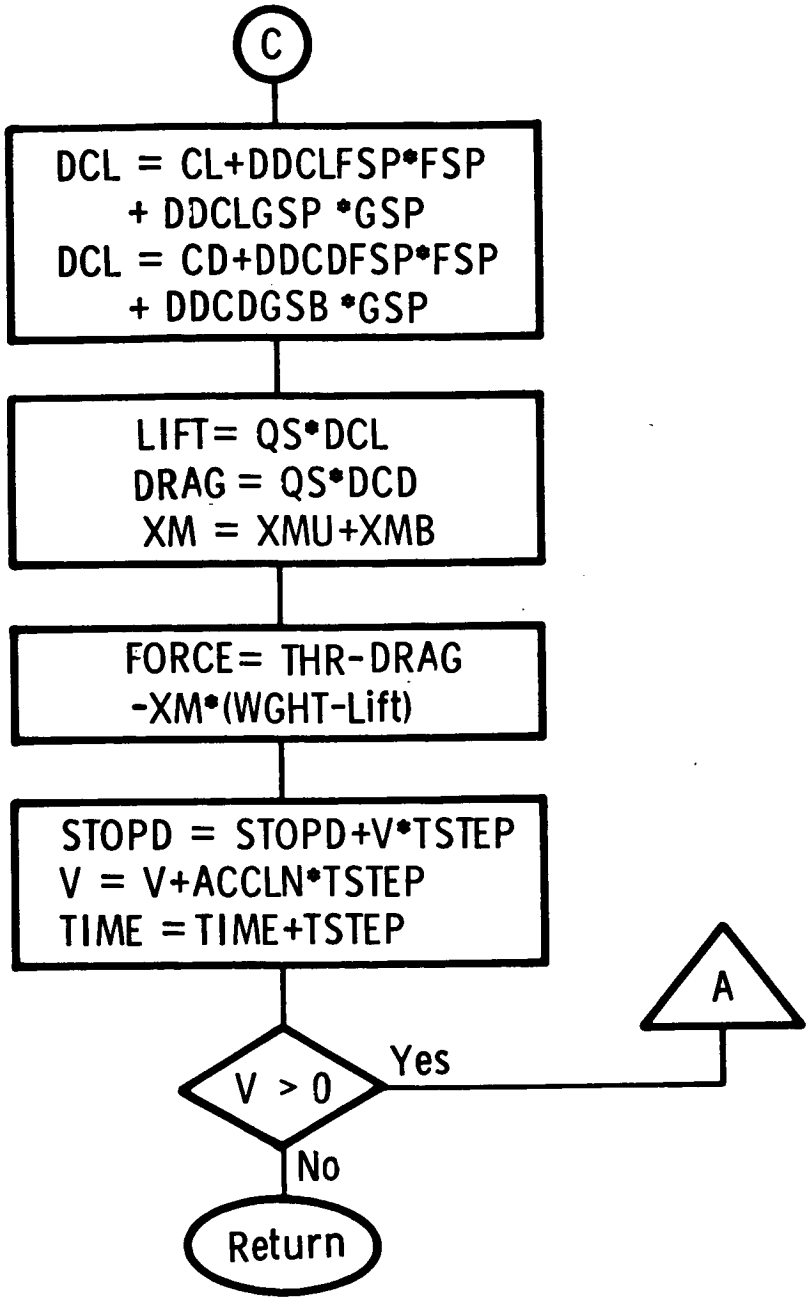

Figure C.3: Continued

Standard Bibliographic Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA CR- 178255		

4. Title and Subtitle	5. Report Date
Development of a Takeoff Performance Monitoring System	March 1987
	6. Performing Organization Code
7. Author(s)	8. Performing Organization Report No.
Raghavachari Srivatsan and David R. Downing	KU-FRL-629-1
9. Performing Organization Name and Address	10. Work Unit No.
The University of Kansas Center for Research, Inc. Lawrence, KS 66045	11. Contract or Grant No. NCC1-79
12. Sponsoring Agency Name and Address	13. Type of Report and Period Covered
National Aeronautics and Space Administration Washington, DC 20546	Contractor Report January 1984 to June 1985
	14. Sponsoring Agency Code 505-66-41-05

15. Supplementary Notes Raghavachari Srivatsan, Project Manager of NCC1-79, performed and documented this work as his doctoral dissertation. David R. Downing, Principal Investigator, was his advisor at the University of Kansas.

16. Abstract

This document discusses the development and testing of a real-time takeoff performance monitoring algorithm. The algorithm is madeup of two segments: a pretakeoff segment and a real-time segment.

One-time inputs of ambient conditions and airplane configuration information are used in the pretakeoff segment to generate scheduled performance data for that takeoff.

The real-time segment uses the scheduled performance data generated in the pretakeoff segment, runway length data, and measured parameters to monitor the performance of the airplane throughout the takeoff roll. Airplane and engine performance deficiencies are detected and annunciated. An important feature of this algorithm is the one-time estimation of the runway rolling friction coefficient.

The algorithm was tested using a six-degrees-of-freedom airplane model in a computer simulation. Results from a series of sensitivity analyses are also included.

17. Key Words (Suggested by Authors(s))		18. Distribution Statement
Algorithm	Safety	Unclassified - Unlimited
Airplane	Performance Monitor	Subject Category 06
Runway	Pretakeoff	
Takeoff	Real-time	
Aborted Takeoff	Friction	
	Acceleration	

19. Security Classif.(of this report)	20. Security Classif.(of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	209	A10

For sale by the National Technical Information Service, Springfield, Virginia 22161

NASA Langley Form 63 (June 1985)