NASA CR-165876

NASA Contractor Report 165876

NASA-CR-165876
19820017360

AN AIRCRAFT SENSOR FAULT TOLERANT SYSTEM

Alper K. Caglayan,
Roy E. Lancraft

Bolt Beranek and Newman, Inc.
Cambridge, MA 02238

Contract NAS1-16579
April 1982

National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23665

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

TCV	Terminal Configured Vehicle
MLS	Microwave Landing System
EKF	extended Kalman filter
RSDIMU	dual fail-operational redundant strapdown inertial measurement unit
IAS	indicated airspeed
IMU	inertial measurement unit
A/C	aircraft
DME	Distance Measuring Equipment
ADC	Air Data Computer
det	determinant
ℓn	natural logarithm

LIST OF SYMBOLS

G-frame	ground based Cartesian coordinate system with origin on the runway
r_x, r_y, r_z	A/C position in the G-frame (m)
$\dot{r}_x, \dot{r}_y, \dot{r}_z$	A/C velocity in the G-frame (m/sec)
ϕ, θ, ψ	A/C Euler angles which rotate the body axes into coincidence with the G-frame (deg)
w_x, w_y	horizontal wind components in the G-frame (m/sec)
x	A/C state vector consisting of vehicle position, velocity, Euler angles, and horizontal winds
u	vector composed of accelerometer and rate gyro measurements
b_u	accelerometer and rate gyro measurement bias vector
g	acceleration of gravity (m/sec^2)
τ	sampling period (sec)
u_g	A/C position and velocity increment in a sampling interval due to earth's gravity
A	A/C state transition matrix
B(x)	A/C input transition matrix
A_w	wind dynamics system matrix
T_{GB}	coordinate transformation matrix from the body axes into the G-frame
T_{ER}	transformation matrix relating the body rates to the Euler angle rates
w	white process noise vector representing the integrated effects of accelerometer, rate gyro, and wind process noises on A/C states

V_a	accelerometer measurement noise variance matrix
V_{rg}	rate gyro measurement noise variance matrix
Q_w	wind process noise variance matrix
(x_M, y_M, z_M)	azimuth antenna location in the G-frame (m)
(x_E, y_E, z_E)	elevation antenna location in the G-frame (m)
y_{rn}	MLS range measurement (m)
b_{rn}	MLS range measurement bias (m)
v_{rn}	MLS range measurement noise (m)
y_{az}	MLS azimuth measurement (deg)
b_{az}	MLS azimuth measurement bias (deg)
v_{az}	MLS azimuth measurement noise (deg)
y_{el}	MLS elevation measurement (deg)
b_{el}	MLS elevation measurement bias (deg)
v_{el}	MLS elevation measurement noise (deg)
r_{az}	A/C range from the azimuth antenna (m)
r_{el}	A/C range from the elevation antenna (m)
y_{sp}	IAS measurement (m/sec)
b_{sp}	IAS measurement bias (m/sec)
v_{sp}	IAS measurement noise (m/sec)
y_ϕ, y_θ, y_ψ	IMU roll, pitch, and yaw attitude measurements (deg)
b_ϕ, b_θ, b_ψ	IMU roll, pitch, and yaw measurement biases (deg)
v_ϕ, v_θ, v_ψ	IMU roll, pitch, and yaw measurement noises (deg)

y	measurement vector consisting of MLS, IAS, IMU outputs
b_y	output sensor bias vector
v	measurement noise vector
R	measurement noise variance matrix
$h(x)$	nonlinear transformation relating A/C states to measurements
b	composite bias vector containing input and output sensor biases
$P_b(0)$	bias vector initial condition variance
$P_x(0)$	A/C state initial condition variance
K	no-fail EKF gain
K_o	bias-free EKF gain
K_b	EKF gain partition corresponding to bias estimates
K_x	EKF gain partition corresponding to A/C state estimates
V	bias correction matrix
F	linearized A/C state transition matrix
H	linearized measurement matrix
r	EKF measurement residual vector
r_o	expanded residual vector of the EKF
D	output sensor bias measurement matrix
$P_o(k+1/k)$	bias-free EKF prediction error covariance
$P_o(k/k)$	bias-free EKF estimation error covariance
$P_x(k+1/k)$	EKF state prediction error covariance
$P_b(k+1/k)$	EKF bias prediction error covariance

$P_{xb}(k+1/k)$	EKF state and bias prediction error cross covariance
$P_b(k/k)$	EKF bias estimation error covariance
$\tilde{R}$	EKF innovations covariance
m_i	i'th sensor bias failure magnitude
C_i	measurement matrix relating i'th sensor failure to the residuals of the no-fail EKF
B_i	i'th column of the input transition matrix B
D_i	i'th row of the output sensor bias measurement matrix D
G_i	i'th detector Kalman filter gain
$P_i(k/k)$	i'th detector bias jump estimation error covariance
ℓ	decision residual window length
C_{ij}	cost of accepting i'th hypothesis when j'th hypothesis is true
P_{H_j}	a priori probability of j'th hypothesis being true
Λ_i	likelihood ratio for i'th hypothesis
r_j	residual vector of the j'th detector
$P(H_i \mid Y(K))$	a posteriori probability of i'th hypothesis conditioned on the residual window Y(K)
p,q,r	true roll, pitch, yaw body rates (°/sec)
θ_{xy}, θ_{xz}	misalignment angle which rotates the orthogonal body x-axis into nonorthogonal sensor axis (rad)
p_m, q_m, r_m	misaligned body rates (°/sec)
$\sigma_p, \sigma_q, \sigma_r$	discrete-time roll, pitch and yaw rate measurement noise standard deviations (°/sec)
$\bar{\sigma}_p, \bar{\sigma}_q, \bar{\sigma}_r$	continuous roll, pitch and yaw measurement noise standard deviations (°/sec x $\sqrt{\text{sec}}$)

f_{pqr}	body rate measurement sampling frequency (sec^{-1})
$\sigma_{pb}, \sigma_{qb}, \sigma_{rb}$	standard deviation of the bias uncertainty in the body rate measurements (°/sec)
$\sigma_{sp}, \sigma_{sq}, \sigma_{sr}$	standard deviation of body rate measurement scale factor errors
p_s, q_s, r_s	body rate measurement outputs (°/sec)
KPFB	roll rate gyro bias failure level
KRSF	yaw rate gyro scale factor failure level
KPFN	roll rate gyro noise failure level
b_x, b_y, b_z	body mounted accelerometer bias levels (m/sec^2)
M	maximum number of possible sensor failures

Subscripts

i	i'th failure
b	bias
x	A/C state
xb	A/C state and normal operating bias
°	no-fail

Superscipts

·	time derivative
^	least mean square estimate
-	average of a measurement/replicated version of a measurement matrix
i	i'th replication
'	transpose of a matrix

I. INTRODUCTION*

This study considers the design of a sensor fault tolerant system using analytic redundancy for the TCV (Terminal Configured Vehicle) research aircraft (Boeing 737) in a Microwave Landing System (MLS) environment. The overall objective of the fault tolerant system is to provide reliable estimates for aircraft position, velocity, and attitude in the presence of possible failures in navigation aid instruments and on-board sensors. The estimates, provided by the fault tolerant system, are used by the automated guidance and control system to land the aircraft along a prescribed path. Sensor failures are identified by utilizing the analytic relationship between the various sensor outputs arising from the aircraft equations of motion.

An aircraft sensor fault tolerant system design methodology is developed by formulating the problem in the context of simultaneous state estimation and failure identification in discrete time nonlinear stochastic systems. The resulting sensor fault tolerant system consists of 1) a no-fail estimator which is an extended Kalman filter (EKF) based on the assumption of no failures and

* Use of commercial products or names of manufacturers in this report does not constitute official endorsement of such products or manufacturers, either expressed or implied, by the National Aeronautics and Space Administration.

provides estimates for aircraft state variables and normal operating sensor biases; 2) a bank of detectors which are first order filters for estimating bias jump failures in sensor outputs; 3) likelihood ratio computers; and 4) a decision function which selects the most likely failure mode based on the likelihood ratios.

The operation of the fault tolerant system is as follows: First, the EKF computes estimates for aircraft position, velocity, attitude, horizontal winds, and normal operating sensor biases on the assumption of no sensor failures. The residuals of this EKF drive a bank of detectors each of which estimates a postulated bias jump failure for a given sensor. Then, multiple hypothesis testing procedure is employed to decide whether the EKF is operating with healthy sensors or under one of the hypothesized failed sensor modes. The multiple hypothesis test selects the most likely failure mode based on the likelihood ratios which are computed using the bias jump failure estimates from the detectors. When a failure is declared by the decision logic, the filter-detector structure is reconfigured by throwing out the failed sensor, making the appropriate changes in the no-fail filter and detectors, and initializing the likelihood ratios and a priori probabilities.

The no-fail filter is implemented in a rectangular coordinate system with origin on the runway by using a new separate bias EKF

algorithm which has been obtained by extending the known results for the linear case to nonlinear systems. The body mounted accelerometers and rate gyros form the inputs into the EKF, while MLS range, azimuth, elevation measurements, IAS (indicated airspeed), and IMU (inertial measurement unit) attitude outputs are utilized as measurements by the EKF. If an RSDIMU (dual fail-operational two-degree-of-freedom strapdown inertial measurement unit) is used instead of the IMU, then the RSDIMU accelerometers and rate gyros replace the body mounted accelerometers and rate gyros and the RSDIMU attitude outputs replace the IMU Euler angle measurements. The function of the no-fail filter is similar to that of a navigator coordinatized in a local runway frame of reference. While the navigator equations usually involve open loop integration of the body accelerations in the runway frame with occasional position and velocity fixes, the no-fail EKF in our study performs the position, velocity, and attitude updates on-line in a closed loop feedback system mechanism.

The proposed filter-detector structure is computationally feasible. In contrast to a previous study reported in [22]-[23], where the failure detection system requires the implementation of M+1 high order nonlinear EKF's (where M is the total number of possible sensor failures), the proposed sensor fault tolerant

system design requires a single high order EKF (no-fail filter) and M first order detectors. The state estimation and failure detection performance of the developed sensor fault tolerant system is analyzed by using a nonlinear six-degree-of-freedom simulation of the TCV research aircraft.

The simulation software is essentially an integration of the NASA supplied TCV and RSDIMU computer simulation programs. Aircraft sensor models have been developed and appended into the simulation to provide more realistic normal operating errors. Furthermore, sensor failure models for increased bias, hardover, null, scale factor, ramp, and increased noise type sensor malfunctions have also been assimilated into the software.

Preliminary analysis of the simulation results obtained so far indicates that the no-fail EKF estimation errors compare favorably to those obtained with other types of navigation filters employed in the same MLS environment. Sensor failure detection performance of the fault tolerant system is excellent for the EKF output sensors such as MLS, IAS, IMU measurements, while the failure detection speed for input sensors such as accelerometers and rate gyros has been found to be slower than that of output sensors. A number of technical issues have been identified during the course of the study and will be further investigated for possible improvement of failure detection performance.

The organization of the report is as follows: Chapter 2 describes the development of the proposed aircraft fault tolerant system. Simulation-related work is given in Chapter 3. A discussion of the simulation results is provided in Chapter 4. Chapter 5 contains a summary of the work done along with an outline of issues to be resolved in the second year of the study.

II. FILTER-DETECTOR DEVELOPMENT

In this chapter the analytical structure for an aircraft sensor fault tolerant navigation system will be developed. The resulting system will be shown to consist of a failure-free conditioned (navigation) filter followed by a bank of low-order failure detectors and their companion decision logic. Although the original formulation outlined in our proposal was based on using a linear time-varying model of the aircraft dynamics, it was decided to extend the utility of the resulting system by employing a detector-estimator structure based on nonlinear aircraft point mass equations of motion. This decision necessitated the development of nonlinear filter-detector algorithms analogous to those given in the proposal for the linear case. While the new algorithms are naturally more complex due to the linearizations involved in nonlinear filtering, they have the advantage of being independent of the flight path and the selected trim condition.

The outline of Chapter II is as follows. An overall description of the fault-tolerant system is given in Section 2.1. The aircraft point mass equations of motion and sensor dynamics, on which the filter-detector development is based, is then discussed in Section 2.2. Section 2.3 outlines the operation of the no-fail filter. Failure detector implementation is discussed in Section 2.4, and in Section 2.5, the operation of the failure decision logic is explained.

2.1 Fault Tolerant System Overview

The design problem, in its most generic form, can be stated as follows: Given discrete, redundant measurements of the various on-board sensors and navigation aid instruments on an aircraft, generate estimates for the vehicle states required by the automated guidance and control laws such that reliable estimates are produced in the presence of possible sensor failures. The desired qualities of a fault tolerant system accomplishing these requirements are the following:

- fast detection of different types of failures (i.e, hardover, null, increased inaccuracy, ramp, etc.)
- ability to handle various levels of failures for a given sensor (i.e., hard, mid and soft)
- utilize inherent analytical redundancy
- minimal complexity

With these goals in mind, the aircraft sensor fault detection design problem was formulated in the context of simultaneous state estimation and failure detection in nonlinear discrete time stochastic systems. Figure 1 displays the major components of the resulting filter-detector structure.

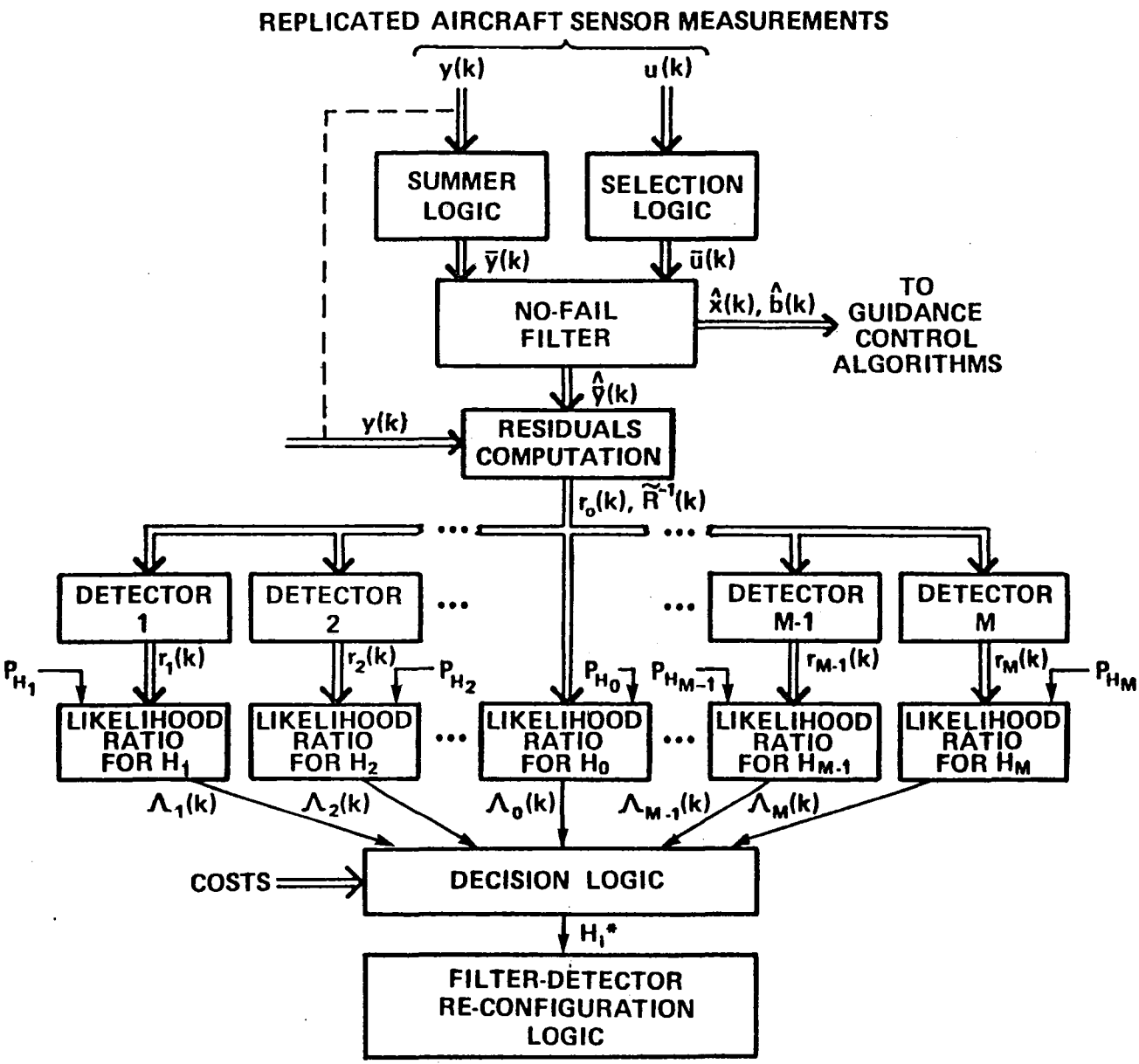


Figure 1. Filter-Detector Structure

As seen in the figure, our filter-detector structure consists of 1) a no-fail estimator which is an extended Kalman (EKF) filter based on the assumption of no failures and provides estimates for A/C state variables and normal operating sensor biases; 2) a bank of detectors which are first order filters for estimating bias jump failures in sensor outputs; 3) likelihood ratio computers; and 4) a decision function which selects the most likely failure mode in the Bayesian sense based on the likelihood ratios. Each of these major components will be discussed in detail in the ensuing sections. In this section, only an overall picture of the operation of the system is discussed.

Notice that only one set of the replicated input sensors and the average of the replicated output sensors enter into the no-fail filter after getting processed in the "selection logic" and "summer logic" blocks. The use of these lower order "generic" inputs and observations serves to reduce the overall complexity of the no-fail filter, without a corresponding loss of generality. The no-fail filter functions essentially as a navigator in this system, estimating the state of the aircraft and the "normal operating" biases on input sensors. However, unlike most navigators, this one continuously filters the navigation aid, IAS, and attitude measurements, so as to constantly correct the propagated state estimates. In addition, it is formulated as a nonlinear extended

Kalman filter so that it is independent of flight path and trim conditions.

Following the no-fail filter in Figure 1 is a functional block which forms an expanded set of residuals (essentially undoing the effects of driving the no-fail filter with averaged measurements). These residuals, in turn, drive a bank of detectors, where each detector is cascaded by a likelihood ratio computer. Currently, each detector is designed to identify a single sensor failure, and compensate the no-fail residuals so as to approximately remove the effects of the hypothesized sensor failure. The detectors are essentially first order Kalman filters each of which estimate a hypothesized bias jump for a given sensor. This simple bias failure model was chosen because it is fairly robust in detecting many other types of failures [24].

Each likelihood ratio computer takes in a residual sequence from the no-fail filter or the detectors, and returns a likelihood ratio which reflects the a posteriori probability of that residual sequence corresponding to the true hypothesis. Likelihood ratios from all the likelihood ratio computers then drive a decision module. Here a Bayes cost function is minimized and the most probable hypothesis along with its failure level estimate and posteriori probability is passed to the filter-detector reconfiguration logic. Depending on the magnitude of the failure

level estimate and the confidence in the selected decision as dictated by its a posteriori probability, the reconfiguration logic either waits for further failure confirmation or deletes the failed sensor, appropriately modifies the filter-detector structure, and initializes the likelihood ratio computers.

2.2 Aircraft Equations of Motion

The estimator-detector algorithms will be developed using aircraft point mass equations of motion mechanized in a ground based, flat earth Cartesian coordinate system (G frame) with its origin located on the runway (See Figure 2). For ease of presentation, it will be assumed that the G-frame is inertial. Since the aircraft position, velocity, and attitude are needed by the automated guidance and control system to land the aircraft along a prescribed path, these variables along with horizontal wind components will be selected as state variables for estimation.

The sensor package considered here includes three body mounted accelerometers, each one aligned along one of the body frame axis [3], a similar set of three body mounted rate gyros, along with an airspeed indicator, and an IMU platform or a RSDIMU. Only the algorithms for the sensor package containing the IMU will be presented. The navigation aid is a ground-based Microwave Landing System (MLS) which transmits position information to aircraft

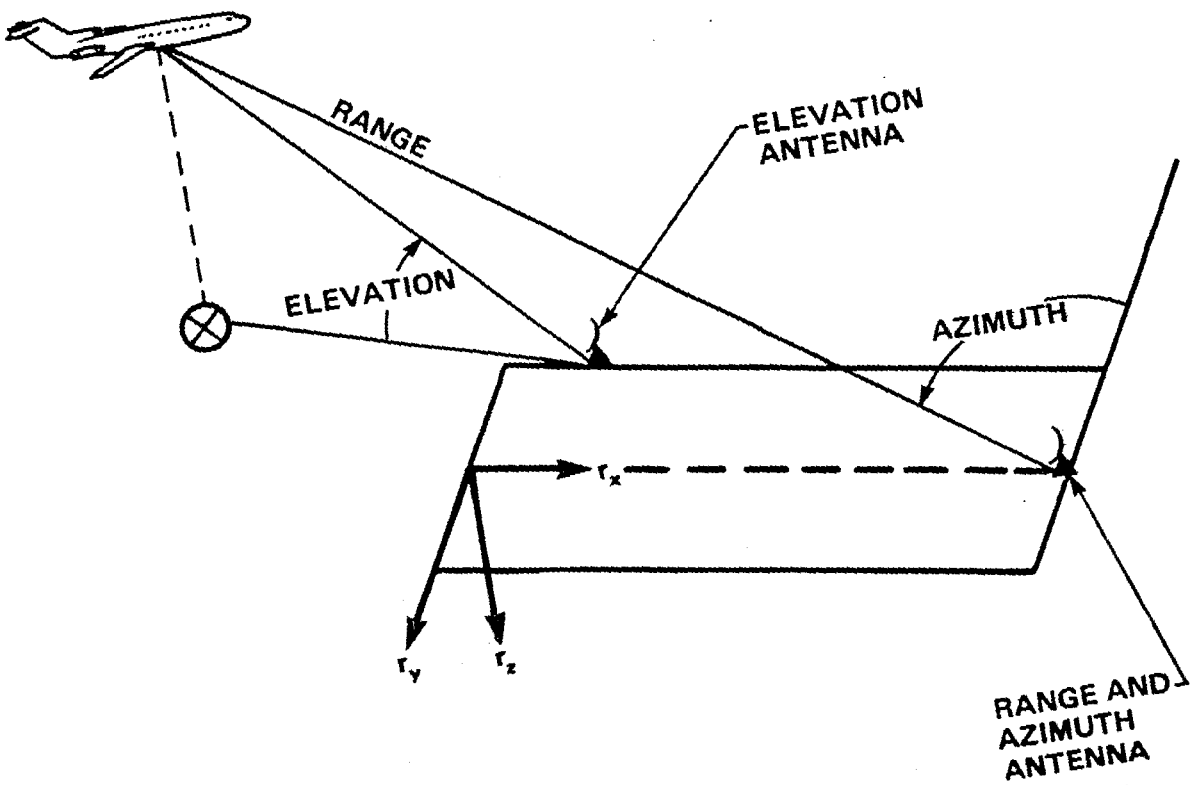


Figure 2. Runway Coordinate System

within its volumetric coverage at discrete time intervals. The MLS consists of a Distance Measuring Equipment (DME) providing aircraft range information, an azimuth antenna co-located with the DME provides the aircraft's angle relative to the runway, and an elevation antenna, located near the glidepath intercept point provides the aircraft with its elevation angle relative to the local horizon.

Body mounted accelerometers and rate gyro measurements form the inputs into the dynamics, while MLS, IMU and IAS comprise the system dynamics outputs. In the following, unsubscripted variables will imply coordinatization in the G-frame. The aircraft state will be given by $x=[r_x, r_y, r_z, \dot{r}_x, \dot{r}_y, \dot{r}_z, \phi, \Theta, \psi, w_x, w_y]'$ where (r_x, r_y, r_z) and $(\dot{r}_x, \dot{r}_y, \dot{r}_z)$ are the A/C position and velocity with respect to the runway frame, (ϕ, Θ, ψ) are the vehicle Euler angles [4], and (w_x, w_y) are the horizontal wind components. Transforming the specific force [3] measured by the body mounted accelerometers into the G-frame, and integrating this expression along with the differential equations for the Euler angles over a sampling interval of τ seconds [5], the following nonlinear discrete-time stochastic difference equation describing the aircraft dynamics are obtained:

$$x(k+1)=Ax(k) + B(x(k))[u(k)-b_u(k)] + u_g + w(k) \qquad (2.1)$$

where the six dimensional vectors u and b_u are composed of accelerometer and rate gyro measurements, and their associated biases, respectively. $u_g = [0,0,\tau^2 g/2,0,0,\tau g,0,0,0,0,0]'$ where g is the earth's gravitational constant. The vector u_g represents the incremental effect of the earth's constant gravitational force on the system state. The matrices A and B are defined by

$$A = \begin{bmatrix} I & \tau I & 0 & 0 \\ 0 & I & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & 0 & 0 & e^{A_w \tau} \end{bmatrix} \qquad B(x(k)) = \begin{bmatrix} \tau^2/2 & T_{GB}(x(k)) & 0 \\ \tau & T_{GB}(x(k)) & 0 \\ 0 & & \tau T_{ER}(x(k)) \\ 0 & & 0 \end{bmatrix} \tag{2.2}$$

A_w is the 2x2 system matrix associated with the wind dynamics. The 3x3 matrix T_{GB} is the transformation from the body axes into the G frame [4] given by:

$$T_{GB} = \begin{bmatrix} c\theta c\psi & s\phi s\theta c\psi - c\phi s\psi & c\phi s\theta c\psi + s\phi s\psi \\ c\theta s\psi & s\phi s\theta s\psi + c\phi c\psi & c\phi s\theta s\psi - s\phi c\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{bmatrix} \tag{2.3}$$

where ϕ, Θ, ψ are the Euler angles and c,s, and t are abbreviations for cosine, sine and tangent functions, respectively. T_{ER} is the 3x3 matrix relating the body rates to the Euler angles [4] defined by:

$$T_{ER} = \begin{bmatrix} 1 & t\theta s\phi & t\theta c\phi \\ 0 & c\phi & -s\phi \\ 0 & s\phi sc\theta & c\phi sc\theta \end{bmatrix} \tag{2.4}$$

The variance, Q, of the white process noise, w(k), is given by

$$Q = \begin{bmatrix} \frac{\tau^3}{3} T_{GB} V_a T_{GB}' & \frac{\tau^2}{2} T_{GB} V_a T_{GB}' & 0 & 0 \\ \frac{\tau^2}{2} T_{GB} V_a T_{GB}' & \tau T_{GB} V_a T_{GB}' & 0 & 0 \\ 0 & 0 & \tau T_{ER} V_{rg} T_{ER}' & 0 \\ 0 & 0 & 0 & \int_0^\tau e^{A_w s} Q_w e^{A_w' s} ds \end{bmatrix} \tag{2.5}$$

where V_a and V_{rg} are the measurement noise variances for the accelerometers and rate gyros, and Q_w is the process noise variance associated with the wind dynamics.

Note that the system matrix A is constant. However, both the process noise variance, Q(k), and the system input matrix, B, is state dependent due to the nonlinear state dependent transformation T_{GB} and T_{ER}. Now let us consider the measurement equations for the system described by eqs. 2.1-2.5. Let (x_M, y_M, z_M) and (x_E, y_E, z_E) be the azimuth and elevation antenna locations in the runway frame. Then, for the MLS range (y_{rn}), azimuth (y_{az}), and elevation ($y_{e\ell}$) measurements are defined by:

$$y_{rn} = r_{az} + b_{rn} + v_{rn} \tag{2.6}$$

$$y_{az}=\sin^{-1}[(-r_y+y_M)/r_{az}]+b_{az} + v_{az} \tag{2.7}$$

$$y_{el}=\sin^{-1}[(-r_z+z_E)/r_{el}]+b_{el} + v_{el} \tag{2.8}$$

where (b_{rn},b_{az},b_{el}) and (v_{rn},v_{az},v_{el}) are biases and measurement noises associated with the MLS and r_{az},r_{el} are the aircraft range from the azimuth and elevation antennas given by:

$$r_{az} =\sqrt{(r_x-x_M)^2+(r_y-y_M)^2+(r_z-z_M)^2} \tag{2.9}$$

$$r_{el} =\sqrt{(r_x-x_E)^2+(r_y-y_E)^2+(r_z-z_E)^2} \tag{2.10}$$

Now consider the air data computer (ADC) outputs. In general, the ADC would provide indicated airspeed, angle of attack and sideslip angle. In our problem, sideslip measurement is not available. In the software provided by NASA, the airspeed measurement is converted into two pseudo horizontal wind measurements by assuming zero angle of attack, sideslip and pitch. While the assumptions on the angle of attack and pitch can be relaxed, we will not use the same approach due to the following reasons. First, the pseudo wind measurement noises are correlated and state dependent. Secondly, the pseudo wind measurement failures would be dependent on each other, resulting in a complex failure detection scheme.

Assuming a zero angle of attack, the airspeed indicator output, y_{sp}, would then be a noisy version of the aircraft velocity with respect to the atmosphere given by:

$$y_{sp} = \sqrt{(\dot{r}_x - w_x)^2 + (\dot{r}_y - w_y)^2 + \dot{r}_z^2} + b_{sp} + v_{sp} \qquad (2.11)$$

where (w_x, w_y) are the horizontal wind components and b_{sp} and v_{sp} are the IAS normal operating bias and white measurement noise. If the angle of attack measurement is available, then eq.(2.11) would be appropriately modified.

The IMU platform provides the Euler angle outputs. These roll (y_ϕ), pitch (y_θ), and yaw (y_ψ) angle measurements are modelled via

$$y_\phi = \phi + b_\phi + v_\phi \qquad (2.12)$$

$$y_\theta = \theta + b_\theta + v_\theta \qquad (2.13)$$

$$y_\psi = \psi + b_\psi + v_\psi \qquad (2.14)$$

where $(b_\phi, b_\theta, b_\psi)$ and $(v_\phi, v_\theta, v_\psi)$ are the biases and white measurement noises associated with platform outputs. Defining the measurement vector, $y' = [y_{rn} y_{az} y_{el} y_{sp} y_\phi y_\theta y_\psi]'$, the system dynamics output becomes

$$y(k+1) = h(x(k+1)) + b_y + v(k+1) \qquad (2.15)$$

where b_y is the output sensor bias vector defined by $b'_y = [b_{rn} b_{az} b_{el} b_{sp} b_\phi b_\theta b_\psi]'$ and v is the measurement noise vector defined by $v' = [v_{rn} v_{az} v_{el} v_{sp} v_\phi v_\theta v_\psi]'$. The nonlinear measurement function h(x) is defined by eqs. 2.6-2.14. In the next section, the no-fail filter which estimates the state variables and the normal operating biases of the stochastic nonlinear dynamic system described above will be discussed.

2.3 No-Fail Filter

The no-fail filter, to be described in this section, is an extended Kalman filter which estimates the aircraft runway position and velocity attitude and horizontal winds along with the normal operating biases of its inputs and measurements. The estimator uses either RSDIMU body rates, or a set of body mounted accelerometer and rate gyro measurements as its inputs as discussed in the previous section. In the case of replicated inputs, redundant accelerometer and rate gyro sensors are kept as standby equipment. MLS range, azimuth and elevation sensors and the IAS provide the measurements into the filter. If desired, IMU platform outputs, or RSDIMU computed attitudes, can also be included in the measurement set. For the case of hardware redundant measurements, the no-fail filter uses an average of the replicated sensor outputs as its measurement. In this way, filter size is kept to a minimum, without loss of generality. The no-fail filter also estimates the normal operating biases of any specified subset of the sensor complement.

The possibility of using one of the two filters NASA supplied for the no-fail filter was investigated. Of these filters, the first, a set of three decoupled complementary filters, was discarded because: 1) filter performance under a subsistent MLS measurement dropout was severely degraded due to the way the pseudo

MLS measurements were computed, and 2) any sensor failure resulted in coupling the three complementary filters together. It was also decided not to use the second filter which was a Kalman filter based on the approximate error dynamics arising from the aircraft point mass equations of motion. This decision was due to the complexity involved in deriving the equations for the propagation of sensor errors through the Kalman filter-navigation structure utilized.

In the process of obtaining the EKF used in our study, we have extended the separate bias estimation algorithms for linear systems to nonlinear systems via the extended Kalman filter framework. Our extension yields a numerical decomposition procedure for obtaining the extended Kalman filter gains. At each sampling instant, the algorithm sequentially computes: 1) a bias-free gain; 2) bias correction matrix; 3) bias gain; and 4) correction to the bias-free gain. Background material on separate bias estimation algorithms can be found in Appendix B.

To understand the operation of the no-fail filter, a signal flow diagram is shown in Figure 3. The filter can be sectioned into two portions, each of which resembles a linear discrete Kalman filter, except for the nonlinear evaluations of B and h, and the manner in which K_x and K_b will be determined. The lower portion of the diagram can be viewed as a normal-operating bias filter. Its

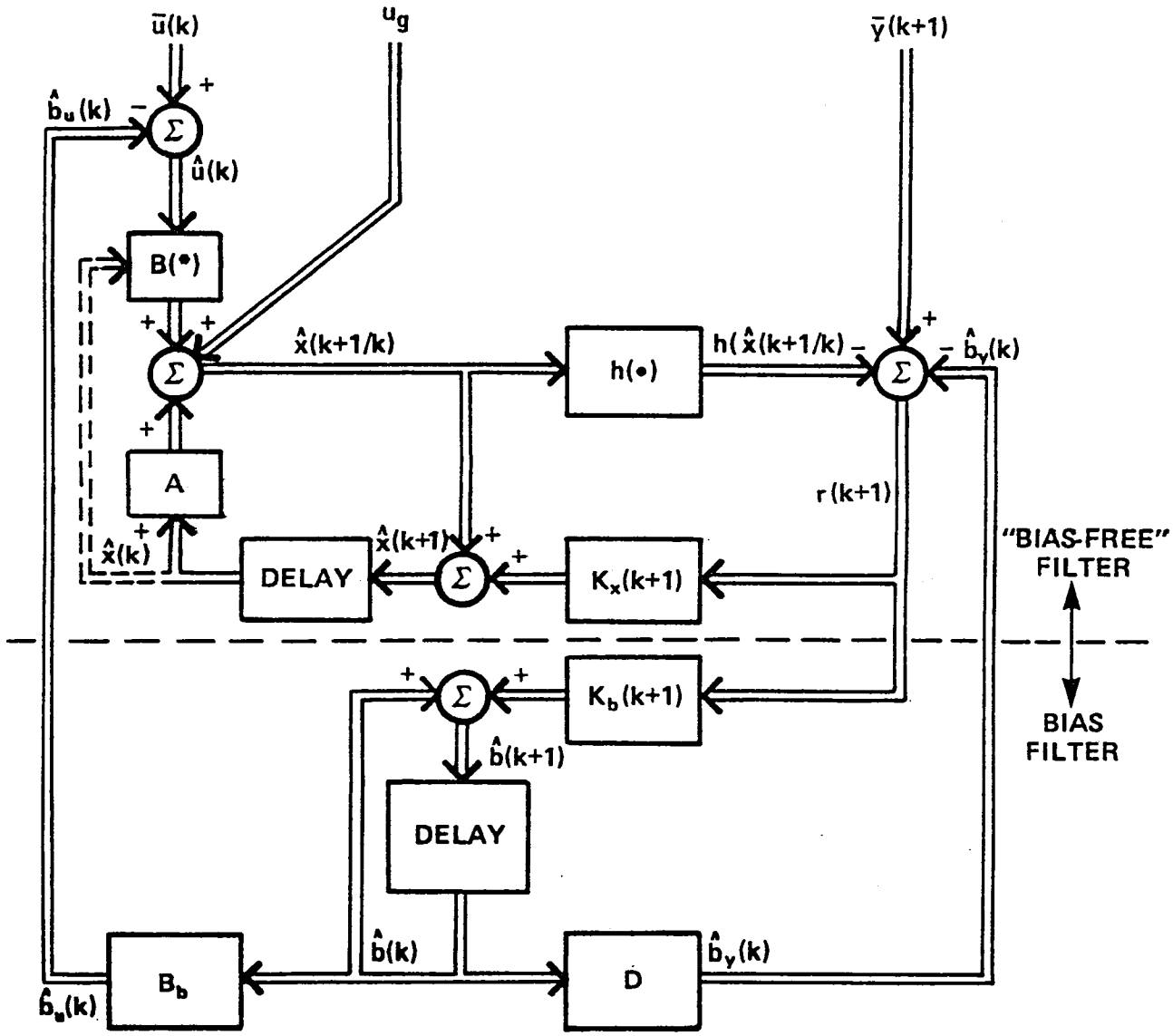


Figure 3. No-Fail Filter Block Diagram

input is the residual sequence r(k+1) from the upper portion, and its outputs are the estimates $\hat{b}_y(k+1)$ and $\hat{b}_u(k+1)$. The upper portion, labelled "bias-free" filter, accepts $\bar{u}(k)$ and u_g as its inputs (where u_g is the incremental effect of earth gravity on the filter state), and $\bar{y}(k+1)$ as its measurement vector. The output of the "bias-free" filter is the residual sequence r(k+1), and the state estimate, $\hat{x}(k+1)$. While the diagram clearly shows the conceptual decoupling of the "bias-free" and bias filters, it also exposes the direct coupling that exists between the two portions. For example, notice how the input and output bias estimates feed back to effect the state estimator, and how the current value of the propagated state estimate, $\hat{x}(k+1/k)$, affects the residual sequence (and therefore the bias estimator) through the nonlinear evaluation of h(.). Not shown in the block diagram is the implicit functional dependence of the Kalman gains K_x, K_b on several quantities to be defined shortly. To implement this filter one need only define the functional form of B(.) and h(.) along with an algorithm for determining the Kalman filter gains.

The computational aspects of the EKF algorithm will now be derived for the system dynamics described by eqs. 2.1-2.15. First combine the input and output bias vectors of the previous section to form a composite bias vector b as $b'=[b_u', b_y']$. The system state and bias initial conditions are assumed to be zero mean Gaussian

random variables with variances $P_x(0)$ and $P_b(0)$, respectively. In addition, it is assumed that the measurement noise $\{v(k), k=1,2,\ldots\}$ is a zero mean, white Gaussian sequence with variance $R(k)$. Furthermore, the plant state and bias initial conditions, measurement and process noise sequences are all assumed to be mutually uncorrelated. In Appendix B, it is shown that the EKF* equations for the nonlinear system dynamics described by eqs. 2.1-2.15 will be given by

$$\hat{x}(k+1)=A\hat{x}(k)+B(\hat{x}(k))\hat{u}(k)+u_g+K(k+1,\hat{x}(k+1/k))r(k+1) \quad (2.16)$$

$$\hat{b}(k+1)=\hat{b}(k)+K_b(k+1,\hat{x}(k+1/k))r(k+1) \quad (2.17)$$

where the innovations sequence of the no-fail filter, $r(k+1)$, is given by:

$$r(k+1) = \bar{y}(k+1) - h(\hat{x}(k+1/k)) - D\hat{b}(k) \quad (2.17a)$$

and the bias compensated input vector, $\hat{u}(k)$, is given by:

$$\hat{u}(k) = u(k) - B_b\hat{b}(k) \quad (2.17b)$$

Note that $D\hat{b}(k) = \hat{b}_y(k)$ and $B_b\hat{b}(k) = \hat{b}_u(k)$; therefore, these matrices are defined as $D = [0\ I]$, $B_b = [I\ 0]$ if all input and output biases are estimated. The filter gain partition, K_x, is defined by:

---

* Reference [6] contains the derivation for the case when the state transition matrix is also nonlinear.

$$K_x(k+1,\hat{x}(k+1/k)) = K_o(k+1,\hat{x}(k+1/k)) + V(k+1,\hat{x}(k+1/k))K_b(k+1,\hat{x}(k+1/k)) \quad (2.17c)$$

and $K_o(k+1,\hat{x}(k+1/k)), V(k+1,\hat{x}(k+1,k))$ and $K_b(k+1,\hat{x}(k+1/k))$ are computed sequentially using the linearized quantities:

$$F(\hat{x}(k),\hat{u}(k)) = A + \frac{\partial B(x(k))u(k)}{\partial x}\Bigg|_{\hat{x}(k),\hat{u}(k)} \quad (2.18)$$

$$H(\hat{x}(k+1/k) = \frac{\partial h(x(k+1))}{\partial x}\Bigg|_{\hat{x}(k+1/k)} \quad (2.19)$$

The expressions for the above partials are given in Appendix A. Using the terminology associated with the linear case, the bias-free filter gain will then be computed by:

$$K_o(k+1,\hat{x}(k+1/k)) = P_o(k+1/k) \quad H'(\hat{x}(k+1/k)[H(\hat{x}(k+1/k) P_o(k+1/k)H'(\hat{x}(k+1/k)+R(k+1)]^{-1} \quad (2.20)$$

and the bias-free prediction error covariance will be given by

$$P_o(k+1/k)=F(\hat{x}(k),\hat{u}(k))P_o(k/k)F'(\hat{x}(k)\hat{u}(k)) +Q(\hat{x}(k),k) \quad (2.21)$$

where eq.(2.21) is initialized with $P_o(0/0)=P_x(0)$ and the bias-free filtering error covariance will be computed by

$$P_o(k+1/k+1)=[I-K_o(k+1,\hat{x}(k+1/k))H(\hat{x}(k+1/k))]P_o(k+1/k) [I-K_o(k+1,\hat{x}(k+1/k))H(\hat{x}(k+1/k)]' +K_o(k+1,\hat{x}(k+1/k))R(k+1)K_o'(k+1,\hat{x}(k+1/k)) \quad (2.22)$$

Note, the bias-free filter gain and the prediction error covariance are computed using H evaluated at $\hat{x}(k+1/k)$ and F evaluated at $\hat{x}(k)$ as opposed to the linear case where one would evaluate them about the bias-free estimates.

The bias-correction matrix V is computed recursively as

$$V(k+1,\hat{x}(k+1/k)) = [I-K_o(k+1,\hat{x}(k+1/k))H(\hat{x}(k+1/k))]F(\hat{x}(k),\hat{u}(k)) V(k,\hat{x}(k/k-1)) + B(\hat{x}(k))-K_o(k+1,\hat{x}(k+1/k)[H(\hat{x}(k+1/k))B(\hat{x}(k))+D] \quad (2.23)$$

where the matrix C is defined by

$$C(k+1,\hat{x}(k+1/k)) = H(\hat{x}(k+1/k[F(\hat{x}(k),\hat{u}(k))V(k,\hat{x}(k/k-1))+B(\hat{x}(k))]+D \quad (2.24)$$

The bias filter gain is expressed as

$$K_b(k+1,\hat{x}(k+1/k)) = P_b(k+1/k+1)[H(\hat{x}(k+1/k))V(k,\hat{x}(k/k-1))+D]'R(k+1)^{-1} \quad (2.25)$$

The bias filter error covariance, P_b, can be computed recursively by propagating the information matrix as

$$P_b^{-1}(k+1/k+1) = P_b^{-1}(k/k)+C'(k+1,\hat{x}(k+1/k))\tilde{R}^{-1}(k+1)C(k+1,\hat{x}(k+1/k)) \quad (2.26)$$

where

$$\tilde{R}^{-1}(k+1) = [H'(\hat{x}(k+1/k))P_o(k+1/k)H(\hat{x}(k+1/k)+R(k+1)]^{-1} \quad (2.26a)$$

The state estimation error covariance $P_x(k+1/k)$, bias estimation error covariance $P_b(k+1/k)$, and cross covariance of state and bias $P_{xb}(k+1/k)$ together define the prediction error covariance for the composite no-fail filter. They are defined by [7],[14]:

$$P_x(k+1/k)=P_o(k+1/k)+[F(\hat{x}(k),\hat{u}(k))V(k,\hat{x}(k/k-1))+B(\hat{x}(k))]P_b(k)$$
$$[F(\hat{x}(k),\hat{u}(k))V(k,\hat{x}(k/k-1))+B(\hat{x}(k))]' \qquad (2.27)$$

$$P_{xb}(k+1/k)=[F(\hat{x}(k),\hat{u}(k))V(k,\hat{x}(k/k-1))+B(\hat{x}(k))]P_b(k) \qquad (2.28)$$

$$P_b(k+1/k)=P_b(k) \qquad (2.29)$$

where $P_o(k+1/k)$ is the prediction error covariance associated with the bias-free computations given by eq.(2.22), the bias correction matrix V is defined by eq. (2.23), and the bias error covariance is computed using eq. (2.26).

Figure 4 summarizes the computational operations involved in this realization of the fail-free filter. The computational sequence given above for obtaining the filter gains provides a numerical decomposition for the extended Kalman filter gain computations and has the following advantages. First, numerical accuracy will be improved due to the lower order matrices involved in the numerical decomposition. Moreover, finite variance for the plant state initial conditions and infinite uncertainty in the a priori bias estimates can be handled easily within this framework.

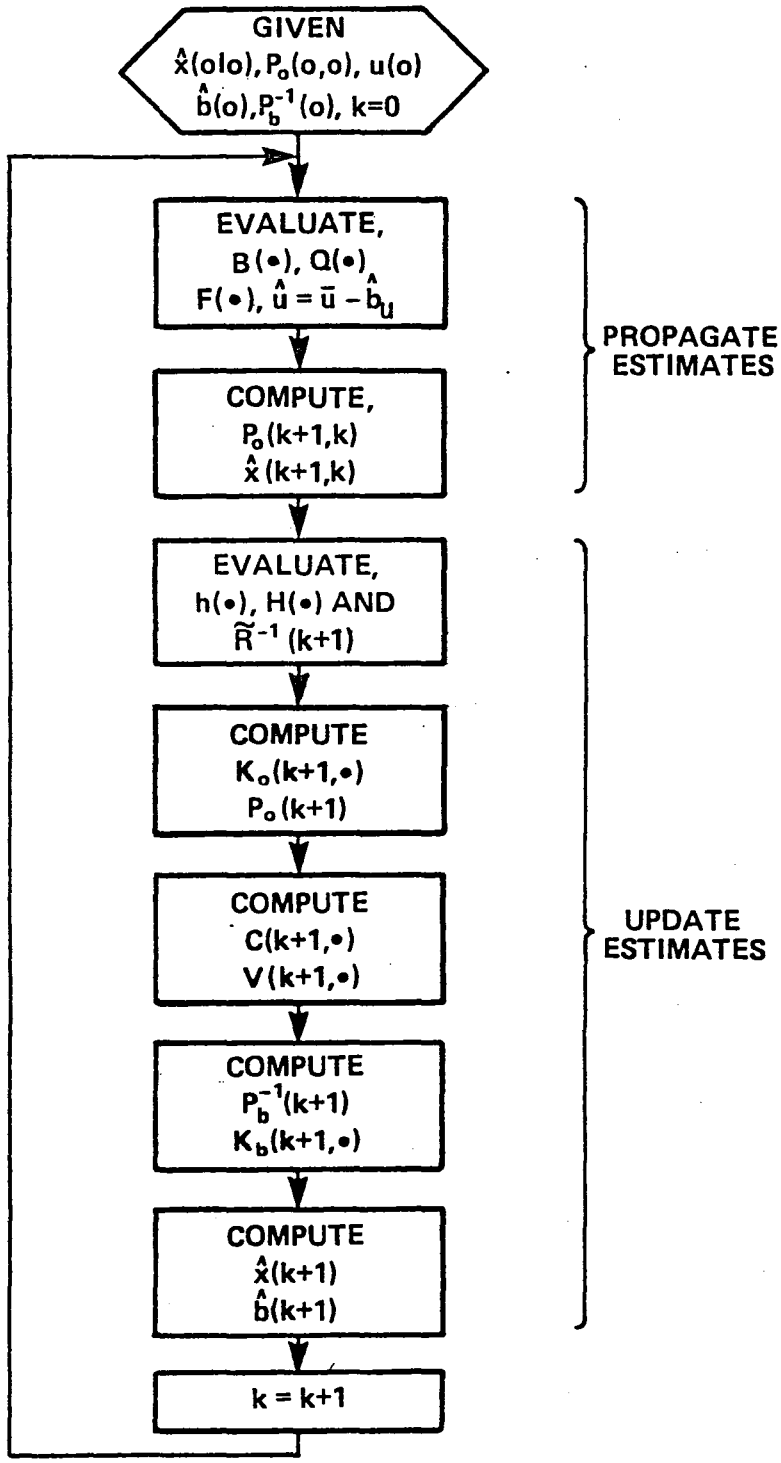


Figure 4. No-Fail Filter Computational Flow Diagram

In addition, the flexibility introduced by the new filter structure can be used to meet other desired design goals. For instance, bias-free and bias gain computations can be performed at different speeds to reduce computational costs.

In the next section, the operation of the detectors, which are driven by the expanded innovations of the fail-free filter described above, will be discussed.

2.4 Detector Implementation

In this section, the blocks in Figure 1 labeled, "residual computation" and "detectors" will be examined in detail. From an input-output point of view it is seen, from Figure 1, that the residual computation block receives as inputs, the "raw" sensor measurements (replicated) and the "no-fail" filter's best estimate of the (unreplicated) measurements. It gives as its output an expanded residual and inverse of the innovations covariance for these expanded residuals.* These expanded residuals, in turn, drive a bank of low order detectors, where each detector delivers a failure corrected residual to the likelihood ratio computers. The function of a detector therefore is to track the occurrence and

* Remember that the no-fail filter receives only the averaged measurements; therefore, the residuals for the individual sensors have to be further computed in order to be able to distinguish failures in like sensors.

level of a sensor failure and compensate the residual sequence of the no-fail filter such that its effects are removed from the residuals.

The detectors to be defined in this section will be conditioned on the occurrence of a single sensor failure. Essentially, they estimate the level of a bias jump in a sensor output, which is hypothesized to occur at a time k_0 Furthermore, detectors operate over a "window" of the expanded residuals, with the initial failure level estimates and uncertainties reset at the beginning of each residual window. The start of a new window determines the hypothesized time of failure k_0, and the maximum length of the window determines the time to wait before initiating a new hypothesis. In the case considered (single sensor failures), the total number of detectors is equal to the sum of the number of input sensors and the number of output (replicated ones included) sensors. For instance, in the case of dual sensor failure redundancy there would be twenty of these first order detectors: three for the body mounted accelerometers, three for the body mounted rate gyros, six for the MLS measurements, two for the IAS outputs, and six for the IMU measurements.

In discussing these issues, it is convenient to define sensor <u>type</u> to be the generic type of the sensor measurement of interest, such as MLS azimuth, or body p gyro output, whereas sensor

replication refers to the particular replication of interest (i.e., second replication of MLS range). The replication will be noted by a superscript in the text (i.e., y^1_{az} = first replication of MLS azimuth).

The residual computation block of Figure 1 has a straightforward function; it serves to generate the residual sequence and innovations covariance that would have emerged from the no-fail filter if the individual sensor measurements had been presented to it, rather than the averaged measurements. The residuals are formed as follows (eliminating the time index for clarity):

$$\begin{aligned} r^1_{az} &= y^1_{az} - \hat{y}_{az} \\ r^2_{az} &= y^2_{az} - \hat{y}_{az} \end{aligned} \tag{2.30}$$

$$\begin{aligned} &\cdot \\ &\cdot \\ r^1_{\psi} &= y^1_{\psi} - \hat{y}_{\psi} \\ r^2_{\psi} &= y^2_{\psi} - \hat{y}_{\psi} \end{aligned} \tag{2.31}$$

defining

$$y^i = [y^i_{rn}, y^i_{az}, y^i_{e\ell}, y^i_{IAS}, y^i_{\phi}, y^i_{\theta}, y^i_{\psi}]' \tag{2.32}$$

the expanded innovations can be written as:

$$r_o(k+1) = \begin{bmatrix} y^1 - \hat{y} \\ y^2 - \hat{y} \end{bmatrix} = \begin{bmatrix} y^1_{(k+1)} - h(\hat{x}(k+1/k) - D\hat{b}_y(k) \\ y^2_{(k+1)} - h(\hat{x}(k+1/k) - D\hat{b}_v(k) \end{bmatrix} \tag{2.33}$$

The innovations variance of the expanded, R(k+1), is found by straightforward substitutions to eq. (2.26a) as:

$$\tilde{R}(k+1) = \begin{bmatrix} H(k+1) & D \\ H(k+1) & D \end{bmatrix} \begin{bmatrix} P_x(k+1/k) & P_{xb}(k+1/k) \\ P'_{xb}(k+1/k) & P_b(k+1/k) \end{bmatrix} \begin{bmatrix} H(k+1) & D \\ H(k+1) & D \end{bmatrix}' + \begin{bmatrix} R & 0 \\ 0 & R \end{bmatrix} \tag{2.34}$$

$$= [\bar{H}(k+1) \ \bar{D}] \begin{bmatrix} P_x(k+1/k) & P_{xb}(k+1/k) \\ P'_{xb}(k+1k) & P_b(k+1/k) \end{bmatrix} [\bar{H}(k+1) \ \bar{D}]' + \bar{R}(k+1) \tag{2.34a}$$

where R is the measurement noise covariance for each set of replicated measurements. Eqs. 2.30-2.34 completely define the function of the residual computation block.

The function of the detectors is clearly two-fold. First, the detectors must keep track of how each hypothesized sensor failure, occurring at time k_o, propagates through the system dynamics to effect the expanded residual sequence of the no-fail filter, as well as track its direct effects on the residuals. Secondly, it must estimate the level of the sensor failure so as to eliminate its effect on the residuals. The ensuing outline for the detectors will define these functions in that order. A typical (say, i'th) input sensor detector estimates a postulated bias jump in the i'th input at time k_o so that the i'th input sensor detector design is based on the following modification of the system dynamics given by eq.(2.1):

$$x(k+1) = Ax(k)+B(x(k))[u(k)-b_u]+B_i(x(k))m_i(k)+u_g+w(k) \quad (2.35)$$

$$m_i(k+1)=m_i(k) \text{ with } m_i(k_o)=m_i \text{ and } m_i(k)=0 \text{ for } k<k_o \quad (2.36)$$

where $B_i(x(k))$ is the i'th column of the input matrix $B(x(k))$ and m_i is the failed bias jump magnitude of the i'th sensor to be estimated. On the other hand, the detector for the i'th output sensor failure is based on the following modification of the measurement equation given by eq. (2.15):

$$y(k+1)=h(x(k+1))+b_y+D_i m_i(k)+v(k+1) \quad (2.37)$$

$$m_i(k+1)=m_i(k) \text{ with } m_i(k_o)=m_i \text{ and } m_i(k)=0 \text{ for } k<k_o \quad (2.38)$$

where m_i is the failed bias jump magnitude for the i'th output sensor and D_i is a column vector with unity entry at the i'th row and zeroes elsewhere. It is assumed that the failed bias jump magnitudes are unknown nonrandom variables. As mentioned previously, the residuals of the no-fail filter serve as measurements to the detectors. In Appendix C, it is shown, under suitable assumptions, that the residual of the no-fail filter, as defined by eq. (2.17a) in the case of i'th failure hypothesis will be given by:

$$r_o(k+1)=C_i(k+1,\hat{x}(k+1/k))m_i+\tilde{r}(k+1) \quad (2.39)$$

where $\tilde{r}(k)$ is the innovations of the no-fail filter under the no-fail hypothesis. Therefore, $\tilde{r}(k)$ is a zero mean white noise sequence with variance $\tilde{R}(k+1)$ defined by eq. (2.34). Referring back to eq. (2.39), $\tilde{r}(k+1)$ would then be the measurement noise in the i'th detector model and the measurement matrix $C_i(k+1,\hat{x}(k+1/k))$ would be given by (see Appendix C for the derivation):

$$C_i(k+1,\hat{x}(k+1/k)) = [\bar{H}(\hat{x}(k+1/k))\ \bar{D}] \begin{bmatrix} F_i(\hat{x}(k),\hat{u}(k)) & B(\hat{x}(k)) \\ 0 & I \end{bmatrix}$$

$$\begin{bmatrix} V_{ix}(k,\ \hat{x}(k/k-1)) \\ V_{ib}(k,\ \hat{x}(k/k-1)) \end{bmatrix} + [\bar{H}(\hat{x}(k+1/k))\ \bar{D}] \begin{bmatrix} B_i(x(k)) \\ 0 \end{bmatrix} + D_i \qquad (2.40)$$

Notice, C_i consists of two portions: the left most matrix product shows how the failure propagates through the dynamics to affect the residuals; the middle product depicts the direct effects of input failures, and the right most matrix illustrates the direct effect of output failures. Furthermore, $B_i(\hat{x}(k))$ is zero in the case of output sensor failures and, D_i is zero in the case of input sensor failures. The matrix $F_i(\hat{x}(k),\hat{u}(k))$ is defined by:

$$F_i(\hat{x}(k),\hat{u}(k)) = F(\hat{x}(k),\hat{u}(k)) - \left.\frac{\partial B_i(x(k))m_i}{\partial x}\right|_{\hat{x}(k),m_i(k)} \qquad (2.41)$$

where $F(\hat{x}(k),\hat{u}(k))$ is given by eq. (2.16). Note, for output failures $F_i = F$ since those failures do not enter through the input weighting matrix B.

The matrix $\begin{bmatrix} V_{ix} \\ V_{ib} \end{bmatrix}$ is analogous to the bias correction matrix given by eq.(2.23) and represents the propagation of a sensor failure, occurring at time k_o, through the no-fail filter dynamics. It is computed using the following recursive relationship:

$$\begin{bmatrix} V_{ix}(k+1) \\ V_{ib}(k+1) \end{bmatrix} = \left\{ I - \begin{bmatrix} Kx(k+1) \\ Kb(k+1) \end{bmatrix} [\bar{H}\ \bar{D}] \right\} \begin{bmatrix} F_i & B \\ O & I \end{bmatrix} \begin{bmatrix} V_{ix}(k) \\ V_{ib}(k) \end{bmatrix} + \begin{bmatrix} B_i \\ O \end{bmatrix} - \begin{bmatrix} K_x(k+1) \\ K_b(k+1) \end{bmatrix} \left\{ [\bar{H}\ \bar{D}] \begin{bmatrix} B_i \\ O \end{bmatrix} + D_i \right\} \tag{2.42}$$

The state dependency of variables above has been suppressed for ease of presentation. The gains K_x and K_b are given by eqs.(2.17c) and (2.25). Note that eq.(2.42) is similar to the bias correction matrix recursive relation given by eq.(2.23). In fact, the formula above can be obtained by replacing K_o by K_x, H by [H D], F by $\begin{bmatrix} F_i & B \\ O & I \end{bmatrix}$, B by B_i, and D by D_i in eq.(2.23). This is to be expected since $\begin{bmatrix} V_{ix} \\ V_{ib} \end{bmatrix}$ represents the effect of a sensor bias failure on the composite no-fail filter and V(k+1) represents the effect of a normal operating bias on the bias free portion of the fail free filter. The postulated sensor failure's effect on both state and normal operating bias estimates are thus computed.

Summarizing, the i'th detector design is based on the observation model described by eq. (2.39) and (2.40) and constant

failure dynamics. The development up to this point has assumed the value of m_i is known. In reality, m_i is nonrandom, but unknown. Therefore, one must continuously estimate its value.

The i'th detector estimate, $\hat{m}_i(k)$, of the i'th sensor failure jump, $m_i(k)$, can be computed by the following first order linear Kalman filter for the case of output sensor failures, and by a first order approximate nonlinear filter in the case of input sensor failures:

$$\hat{m}_i(k+1)=\hat{m}_i(k)+G_i(k+1,\hat{x}(k+1/k))[r(k+1)-C_i(k+1,\hat{x}(k+1/k))\hat{m}_i(k)] \quad (2.43)$$

where the detector estimate $\hat{m}_i(k)$ is initialized at the start of a residual window with $\hat{m}_i(k_o)=0$. Figure 5 shows the simple operation of this filter. The detector gain is computed by:

$$G_i(k+1,\hat{x}(k+1/k))=P_i(k+1/k+1)C_i'(k+1,\hat{x}(k+1/k))\tilde{R}^{-1}(k+1) \quad (2.44)$$

where $P_i(k+1/k+1)$ is the error covariance of the i'th detector bias jump estimate. The information matrix, $P_i^{-1}(k/k)$, of the i'th detector is propagated recursively through:

$$P_i^{-1}(k+1/k+1)=P_i^{-1}(k/k)+C_i'(k+1,\hat{x}(k+1/k))\tilde{R}^{-1}(k+1)C_i(k+1,\hat{x}(k+1/k)) \quad (2.45)$$

with

$$P_i^{-1}(k_o/k_o)=0$$

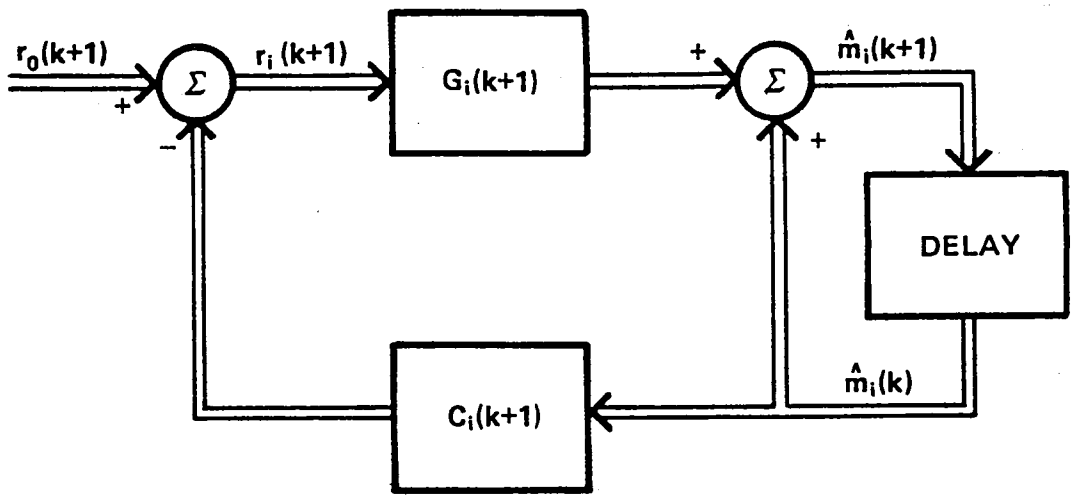


Figure 5. Block Diagram for i'th Detector

That is, the failure bias jump at time k_o is assumed to be a zero mean random variable with infinite covariance (or equivalently, zero information). In the case of output sensor failures, the detector implementation described by eqs. (2.43)-(2.45) above is an exact linear Kalman filter for the hypothesized failure model specified by eqs. (2.37)-(2.39). In the case of input sensor failures, the detector becomes an approximate nonlinear filter due

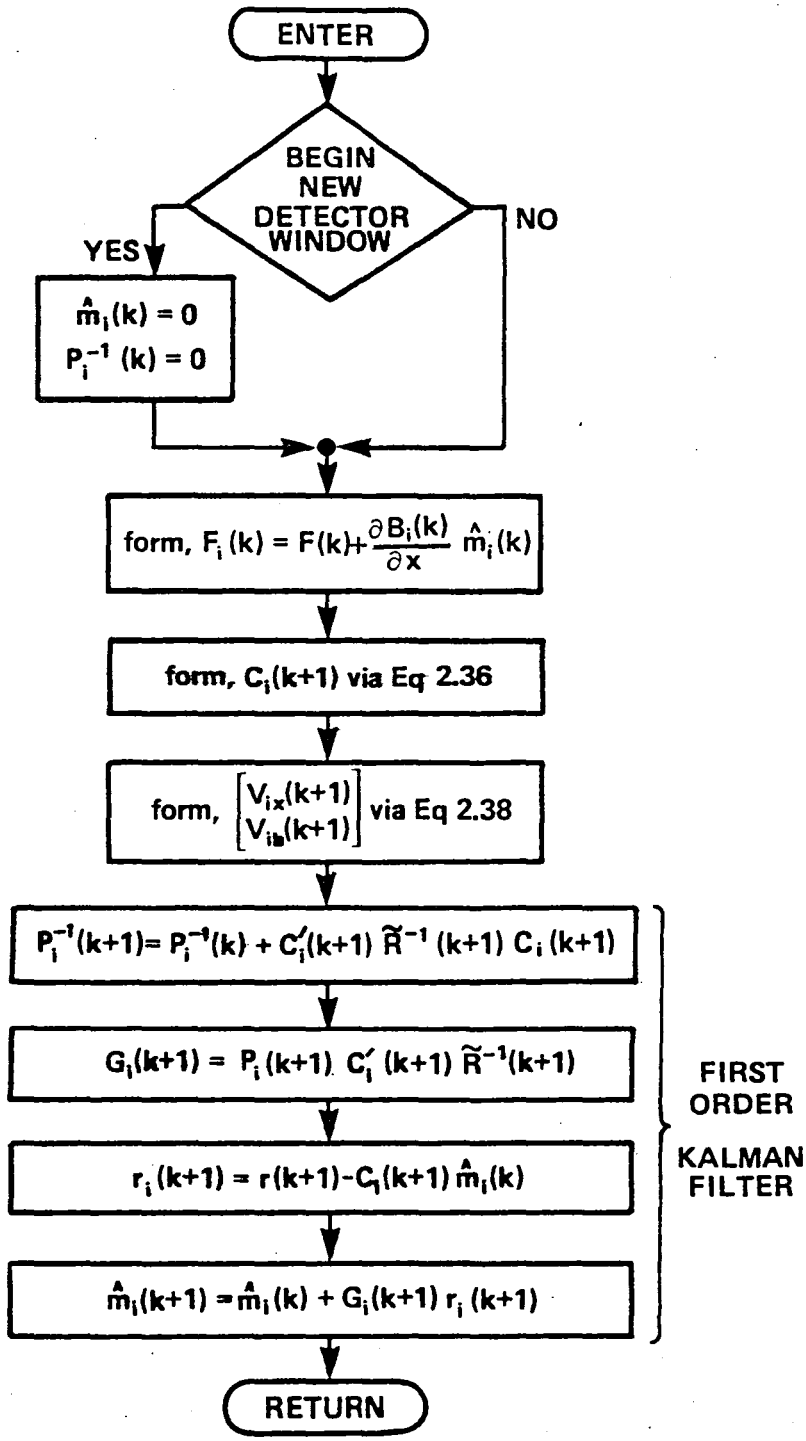


Figure 6. Computational Flow Diagram for i'th Detector

to the dependence of F_i in eq.(2.41) on the failure bias magnitude, m_i, where $\hat{m}_i(k)$ is used in the evaluation of $F_i(\hat{x}(k),\hat{u}(k))$. Figure 6 summarizes the computational operations required to implement each detector, and clearly shows the function of the detectors.

In summary, the detector block consists of a bank of first order estimators driven by the expanded innovations of the no-fail filter. Each detector corresponds to a different sensor failure hypothesis, and, corresponding to each detector, there is an associated residual data window length. The bias jump magnitude for a given sensor failure, hypothesized to happen at the start of the residual window, is estimated by the detector corresponding to that sensor. The residuals of the detectors along with the residual of the no-fail filter are used in the decision block which is discussed in the next section.

2.5 Decision Logic

As seen in Figure 2, the failure compensated residuals from each of the sensor failure detectors along with the expanded innovations sequence of the no-fail filter are used in deciding the most likely failure mode. To arrive at this decision, M-ary hypothesis testing, based on a decision residual window, is utilized. The problem is viewed as follows: Given M sensor failure models, formulate the following M+1 hypotheses:

$$H_o: \quad r_o(k)=r(k) \qquad ,k=k_d,k_d+1,\ldots,k_d+\ell$$

$$H_i: \quad r_o(k)=\tilde{r}(k)+C_i(k,\hat{x}(k/k-1))m_i \qquad ,i=1,2,\ldots,M \tag{2.46}$$

where $r_o(k)$ is the actual expanded innovation sequence of the no-fail filter, $\tilde{r}(k)$ is the innovations sequence of the no-fail filter under no-fail conditions, and ℓ is the length of the decision residual data window on which the M-ary hypothesis test is based. Recall from the previous section, that $\tilde{r}(k)$ is a zero mean white noise sequence with variance $\tilde{R}(k)$ defined by eq. (2.34). The length of the decision residual window is, in general, different from the estimation residual data windows described in the previous section. In summary, an M-ary hypothesis test will be used to decide whether the no-fail filter is operating under no failures (hypothesis H_o), or under the i'th sensor bias jump failure (hypothesis H_i).

The M-ary hypothesis test chosen minimizes [20] the Bayes risk, β , given by:

$$\beta = \sum_{i=0}^{M} \sum_{j=0}^{M} C_{ij} P_{H_j} \int_{Y_i} p(Y(K) \,|\, H_j) dY \tag{2.47}$$

where

C_{ij}: cost of accepting H_i when H_j is true

P_{H_j}: a priori probability of hypothesis H_j being true

Y_i: decision region in the observation space such that hypothesis H_i is selected if $Y(K)$ is in region Y_i.

$Y(K)$: $[r(k_d), r(k_d+1), \ldots, r(K);\ k_d \leq K \leq k_d+\ell]$ observations (in our case, innovations sequence of the no-fail filter) on which the test is based.

$p(Y(K)|H_j)$: probability density of the no-fail filter residuals conditioned on the j'th hypothesis

Bayes risk is a weighted cost of making incorrect decisions. As shown in [20], the Bayes risk is minimized by choosing hypothesis H_j corresponding to the smallest of M+1 possible β_i's defined by:

$$\beta_i(r(K)) = \sum_{j=0}^{M} P_{H_j} C_{ij}\, p\,(Y(K)|H_j) \qquad i=0,1,\ldots,M \tag{2.48}$$

In our case, the conditional probability density above (assuming a Gaussian distribution) will be given by:

$$p\,(Y(K)|H_j) = \left(\prod_{k=k_d}^{K} \frac{1}{(2\pi)^{m/2}(\det\ (\tilde{R}(k))^{\frac{1}{2}}} \right) \Lambda_j(r(K)) \tag{2.49}$$

with $\Lambda_j(r(K))$ defined by

$$\Lambda_j(r(K)) = P_{H_j} \exp\{-1/2 \sum_{k=k_d}^{K} r_j'(k)\tilde{R}^{-1}(k) r_j(k)\} \tag{2.50}$$

where $r_j(k)$ is the j'th detector residual sequence (given in (2.43)) defined by:

$$r_j(k+1) = r_o(k+1) - C_j(k+1, \hat{x}(k+1/k))\hat{m}_j(k) \tag{2.47}$$

Since the denominator term in (2.49) is not "i" or "j" dependent and positive, an equivalent decision would be to choose H_i corresponding to the smallest

$$\beta_i^*(r(K)) = \sum_{j=0}^{M} C_{ij}\Lambda_j(r(K)) \tag{2.52}$$

In computer implementation, Λ_i's are usually scaled by $\frac{1}{\Lambda_o}$ so that they become likelihood ratios. If the costs associated with making wrong decisions are all equal and those of making correct decision are zero (i.e., $C_{ij}=1$ for $i \neq j$ and $C_{ii}=0$), then the optimal Bayesian decision would be to choose H_i corresponding to the smallest one of the M+1 α_i's given by:

$$\alpha_i(r(K)) = +1/2 \sum_{k=k_d}^{K} r_i'(k)\, \tilde{R}^{-1}(k) r_i(k) - \ell n P_{H_i} \tag{2.53}$$

In this case the decision rule is equivalent to choosing hypothesis H_i corresponding to the largest a posteriori probability:

$$P(H_i | Y(K)) = \frac{\Lambda_i(r(K))}{\sum_{j=0}^{M} \Lambda_j(r(K))} \tag{2.54}$$

where $P(H_i|Y(K))$ is the a posteriori probability that H_i is the true hypothesis conditioned on the residual window Y(K).

III. SIMULATION DESCRIPTION

The performance of the filter-detector algorithm has been analyzed by using a six-degree-of-freedom simulation of the TCV Boeing 737 aircraft along with the dual fail-operational two-degree-of-freedom strapdown inertial measurement unit (RSDIMU) software model. These NASA supplied simulation computer programs were first converted from CDC FORTRAN Extended 4 into DEC FORTRAN-10 in order to run them on the BBN computer system. In single precision (36 bit), the aircraft simulation exhibited significant ground track, cross track and glideslope errors. The numerical problems were traced down to the algorithms used in computing waypoints and guidance parameters in earth centered inertial frame. The numerical inaccuracies were alleviated by converting the whole program into double precision (64 bit). The aircraft simulation and the RSDIMU programs were then integrated to permit the use of the RSDIMU as an aircraft sensor. Later, new sensor models were developed and appended into the simulation to provide more realistic normal operating errors. Finally, sensor failure models for increased bias, hardover, null, scale factor, ramp and increased noise type sensor malfunctions were assimilated into the software.

In this chapter, the aircraft and sensor dynamics simulation model will be discussed. Section 3.1 contains a discussion of the

sensor models. Sensor failure models are discussed in Section 3.2.

3.1 Sensor Models

During the early phase of the contract, sensor models for the aircraft sensor complement were developed in order to provide realistic normal operating sensor errors. These models have been integrated into the NASA supplied TCV B-737 aircraft simulation. The generalized sensor models take in true values of measured variables from the aircraft simulation and put out sensor outputs which account for misalignment, measurement noise, bias drift, normal scale factor errors and limits. The model for the rate gyros is described below to outline the features of the modified sensor models.

1. Rate Gyro Misalignment: The true body rates (p,q,r) expressed in the orthogonal body axes are first transformed into the nonorthogonal measurement axes via a small angle transformation to account for rate gyro misalignments:

$$\begin{bmatrix} p_m \\ q_m \\ r_m \end{bmatrix} = \begin{bmatrix} 1 & -\theta_{yz} & \theta_{zy} \\ \theta_{xz} & 1 & -\theta_{zx} \\ -\theta_{xy} & \theta_{yx} & 1 \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \qquad (3.1)$$

where (p_m, q_m, r_m) are the body rates after the misalignment and $(\theta_{xy}, \theta_{yx}, \theta_{xz}, \theta_{zx}, \theta_{yz}, \theta_{zy})$ are six small independent misalignment angles expressed in radians. In our simulation, the misalignment

angles are randomized by specifying the expected standard deviation of the misalignment angles. For instance, rate gyro misalignment angles in the following simulation runs are randomized with 0.03^{o}, as seen in Table I.

2. Measurement Noise: The misaligned quantities are corrupted by zero mean white Gaussian noise

$$\begin{aligned} p_n &= p_m + \sigma_p \times v_p \\ q_n &= q_m + \sigma_q \times v_q \\ r_n &= r_m + \sigma_r \times v_r \end{aligned} \tag{3.2}$$

where (v_p, v_q, v_r) are zero mean white Gaussian sequences with unit variance and $(\sigma_p, \sigma_q, \sigma_r)$ are the standard deviation of the measurement noise in the sampled signal given by

$$\begin{aligned} \sigma_p &= \bar{\sigma}_p \sqrt{f_{pqr}} \\ \sigma_q &= \bar{\sigma}_q \sqrt{f_{pqr}} \\ \sigma_r &= \bar{\sigma}_r \sqrt{f_{pqr}} \end{aligned} \tag{3.3}$$

where $(\bar{\sigma}_p, \bar{\sigma}_q, \bar{\sigma}_r)$ are the standard deviations of the measurement noise in the continuous (p,q,r) measurements and (f_{pqr}) is the associated sampling frequency.

3. Bias Drift: The bias drift is associated with calibration errors and electronic components of a sensor package and is modeled by

Table I. Normal Operating Sensor Model Parameters

Sensor	Units	Misalignment Angle Std. Dev. deg	Noise*	Bias*	Scale Factor St.Dev. %
Roll Rate Gyro	deg/sec	0.03	.02	0.0	.01
Pitch Rate Gyro	deg/sec	0.03	.02	0.0	.01
Yaw Rate Gyro	deg/sec	0.03	.02	0.0	.01
Body Mounted Accelerometer	m/sec^2	0.03	.01	0.1	.25
Body Mounted Accelerometer	m/sec^2	0.03	0.1	-0.05	.25
Body Mounted Accelerometer	m/sec^2	0.03	0.1	0.07	.25
Airspeed Indicator	m/sec	--	--	--	2
MLS Azimuth	deg	--	.05	.037	--
MLS Elevation	deg	--	.05	.03	--
MLS Range	m	--	4.57	30.5	--
IMU Roll Attitude	deg	--	0.27	0.23	--
IMU Pitch Attitude	deg	--	0.23	-0.19	--
IMU Yaw Attitude	deg	--	0.23	0.15	--

$$
\begin{aligned}
p_b &= p_n + \sigma_{pb} \times v_{pb} \\
q_b &= q_n + \sigma_{qb} \times v_{qb} \\
r_b &= r_n + \sigma_{rb} \times v_{rb}
\end{aligned}
\tag{3.4}
$$

where $(\sigma_{pb},\sigma_{qb},\sigma_{rb})$ are the expected standard deviation of biases in the (p,q,r) measurements and (v_{pb},v_{qb},v_{rb}) are zero mean Gaussian random variables with unit variance. That is, the bias level will vary from one run to another, but stay constant during each run. On the other hand, the provisions for providing deterministic instead of random biases are also included in the program.

4. Normal Scale Factor Error: Normal scale factor errors are simulated according to:

$$
\begin{aligned}
p_s &= (1.+ .01 \times \sigma_{sp} \times v_{sp}) \times p_b \\
q_s &= (1.+ .01 \times \sigma_{sq} \times v_{sq} \times q_b \\
r_s &= (1.+ .01 \times \sigma_{sr} \times v_{sr}) \times r_b
\end{aligned}
\tag{3.5}
$$

where $(\sigma_{sp},\sigma_{sq},\sigma_{sr})$ are the expected standard deviation of the scale factor errors in the (p,q,r) measurements. As before the Gaussian random variables (v_{sp},v_{sq},v_{sr}) provide randomized scale factor errors. Similarly, deterministic scale factors can also be introduced in the program. As can be seen in Table I, rate gyro scale factor error standard deviation is 0.01%.

5. Limits: The computed quantities (p_s,q_s,r_s) are compared with the sensor limits and set to the appropriate limit when these thresholds are exceeded.

The other sensor models are essentially similar. The parameter values used for the models are given in Table 1. The measurement noise of indicated airspeed (IAS) is multiplicative. MLS measurement noises can be specified as either a white or time-correlated Gaussian sequence. MLS sensor models also simulate data dropout and filter errors associated with the digital scan mechanism of the microwave landing system. Further details on this model can be found in [1], [18].

Two inertial measurement unit models have also been included in the simulation. The first one, IMU, is a platform whose Euler angle outputs are utilized as measurements in the program. The second one, RSDIMU, is a redundant strapdown package consisting of four two-degree-of-freedom gyros and eight linear accelerometers in a semioctahedron configuration. The RSDIMU model simulates quantization errors in addition to additive noise, bias, scale factor errors. A detailed description of the RSDIMU model can be found in [2].

Sensor models were generalized to accommodate redundant sensor configurations. All sensors can now be dual or triple redundant

with the exception of the RSDIMU which has its own internal hardware replication. The software has been written so that any selected baseline sensor configuration can be handled.

3.2 Sensor Failure Models

The following failure modes have been incorporated into the software for each sensor:

1. Increased Bias Failure: For instance, in the case of roll rate bias failure, the normal operating bias of p rate gyro is bypassed at time of failure for p gyro bias and the standard deviation of the bias level is increased by

$$\bar{\sigma}_{pb} = \text{KPFB} \times \sigma_{pb} \tag{3.6}$$

where KPFB is an integer to be specified. That is, each failure is specified by a time of failure and a ratio by which the failure's standard deviation exceeds the expected normal bias standard deviation.

2. Hardover Failure: Hardover failures correspond to a completely nonoperational sensor status and they are modelled by setting the failed sensor output to its limit. For instance, q rate gyro hardover failure is simulated by setting the pitch rate gyro output to its positive or negative limit at time of failure.

3. Null Failure: Null failures also correspond to an unusable sensor failure mode and they are modelled by zeroing out the failed sensor output at time of failure.

4. Scale Factor Failure: Scale factor failures correspond to a severely degraded scale factor error. For instance, in the case of yaw rate gyro scale factor failure, the standard deviation of the expected scale factor error is increased to

$$\bar{\sigma}_{sr} = \sigma_{sr} \times KRSF \tag{3.7}$$

at time of failure. The integer KRSF specifies the ratio by which the expected failure level exceeds the standard deviation of the normal scale factor error.

5. Increased Noise Failure: These failures are introduced by increasing the sensor measurement noise at time of failure. For example, roll rate gyro noise failure is simulated by setting

$$\bar{\sigma}_{p} = KPFN \times \sigma_{p} \tag{3.8}$$

where KPFN is the specified noise failure level.

6. Ramp Failure: Provisions for introducing ramp failures have also been included. The modelling of these failures is similar to that of bias failures.

Any or all of the above failure modes can be superimposed on any of the sensor models detailed in the previous section. Currently, the time of failure is deterministic (user specified at run time) so that the performance of single simulation runs can be compared to one other. However, randomized specification of failure times will eventually be programmed so that Monte Carlo averaging of the results can be done in an automated fashion.

IV. DISCUSSION OF RESULTS

In this chapter, estimation and failure detection performance of the developed aircraft sensor fault tolerant system will be discussed by examining some typical simulation runs. Section 4.1 is concerned with aircraft state and bias estimation performance when no failures occur. Whereas Section 4.2 investigates the filter performance under failures. Additionally, Section 4.2 is concerned with failure detection performance.

4.1 System Performance - No Failures

The simulation runs start at the point of transition to MLS coverage. The no-fail filter described in Section 2.3 is used to estimate aircraft position, velocity, attitude, and horizontal winds. In these runs, the body mounted accelerometers and MLS range measurement were selected for inclusion in the bias estimator. Therefore, the no-fail filter also provides sensor bias estimates for the body mounted accelerometers and MLS range measurement.

The six-degree-of-freedom simulation was run approximately 120 seconds at 20 Hz. Figures 7 and 8 show the A/C ground track and bank angle profile. Body rate time histories for roll, pitch, and yaw are given in Figure 9. Altitude profile exhibits essentially a constant sink rate. These figures indicate the landing path and transient maneuvers encountered during approach.

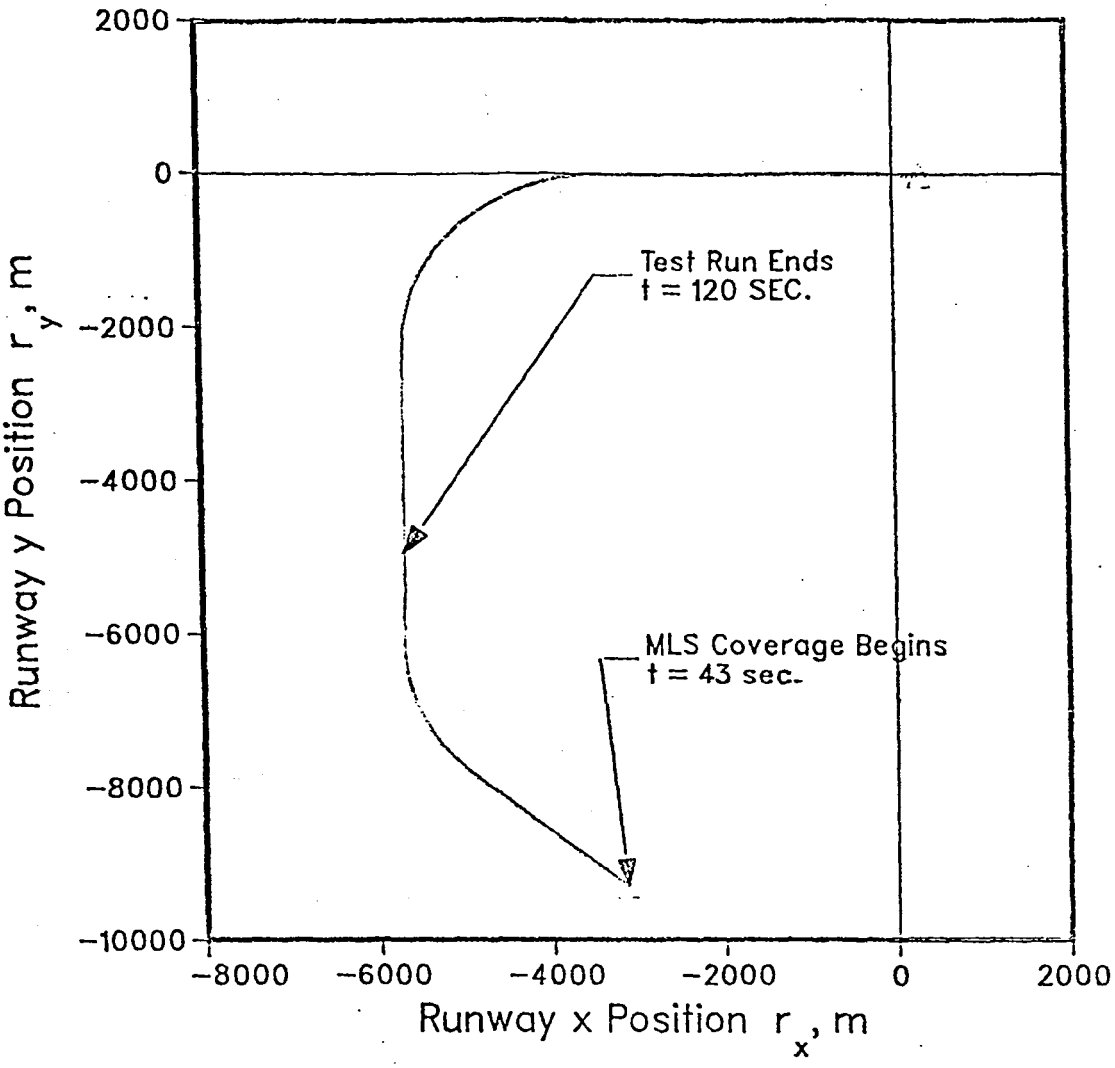


Figure 7: Aircraft Ground Track

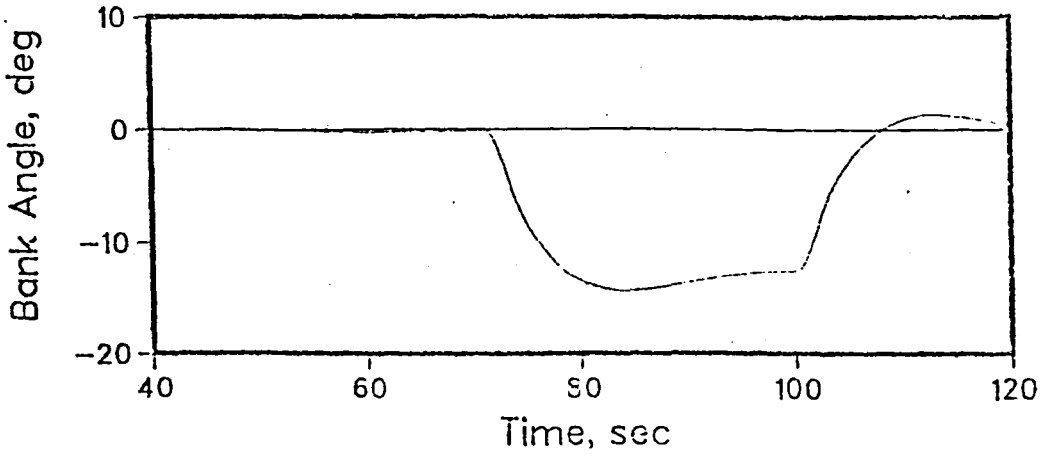
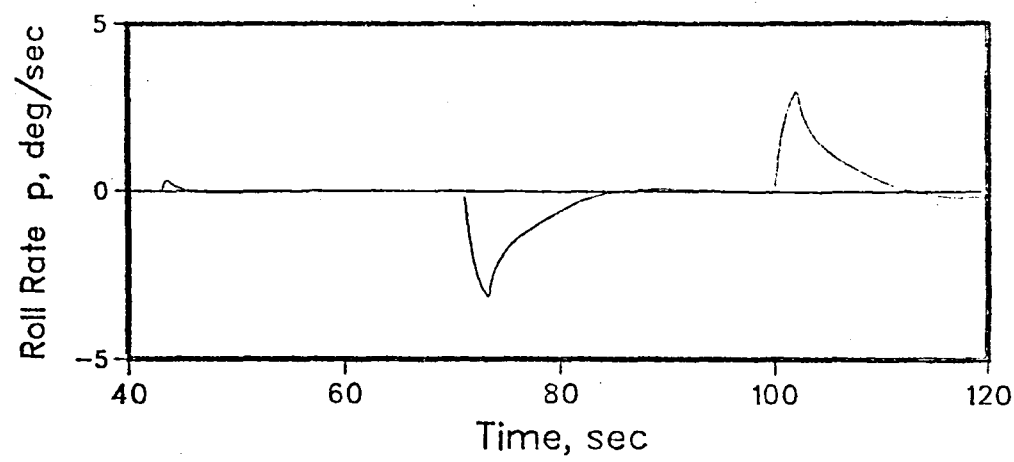
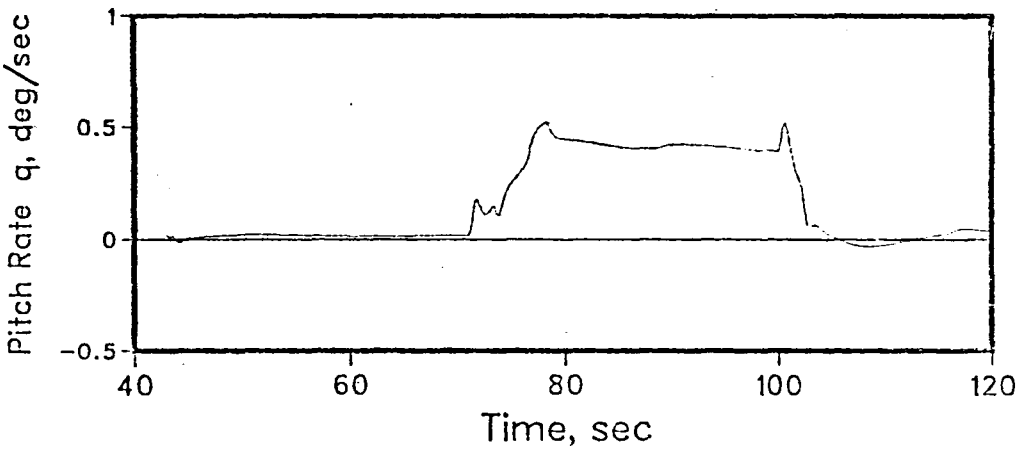


Figure 8: Bank Angle Profile





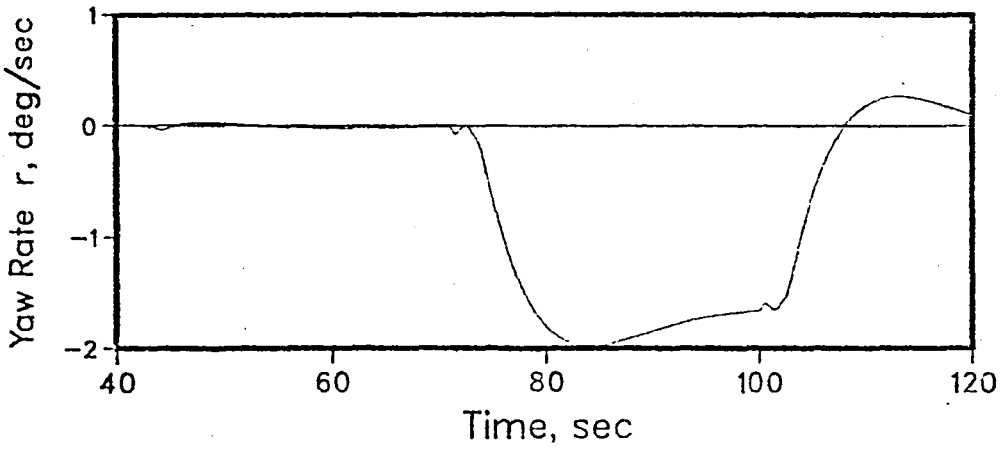


Figure 9: Body Angular Rate Profiles

Table II shows the values used for filter initialization. Initial uncertainty variances, $P_x(0)$ and $P_b(0)$ are both diagonal. Accelerometer biases are denoted by (b_x, b_y, b_z). Actual measurement bias levels are given in Table I. Table III contains the measurement noise standard deviation values used in the filter computations. Actual measurement signals generated in the simulation, however, also contain misalignment, normal operating bias, and scale factor errors depicted in Table I. Moreover, actual IAS noise is multiplicative. All measurement noises are uncorrelated with each other. The horizontal wind dynamics used in the filter were first order filtered noise with a time constant of 1000 seconds.

Figures 10 and 11 show the state estimation performance limit of the baseline EKF in which all sensor biases are set to zero. That is, position and attitude errors in Figures 10 and 11 are the lower limits which can be attained (for the particular run) with the selected normal operating error parameters (misalignment, noise, scale factor, etc.) even if the sensor biases can be perfectly identified. The detectors were also run with this baseline EKF and there were no false alarms.

Notice that although both position and attitude estimation errors are quite low, attitude errors display regular, nonrandom patterns. These errors can be traced to integration errors in the

Table II. Filter Initialization

Variable	Init. Est. Error	Init. Uncert.(St.Dev.)	Units
r_x	-21.34	36.58	m
r_y	13.72	36.58	m
r_z	- 4.57	27.43	m
$\dot{r}_x$	0.61	2.29	m/sec
$\dot{r}_y$	1.52	4.57	m/sec
$\dot{r}_z$	- 0.31	1.07	m/sec
ϕ	0.02	0.3	deg
θ	- 0.11	0.3	deg
ψ	- 0.2	0.64	deg
w_x	- 0.09	0.6	m/sec
w_y	0.2	0.6	m/sec
b_x	0.1	0.3	m/sec^2
b_y	- 0.05	0.3	m/sec^2
b_z	.07	0.3	m/sec^2
b_{rn}	30.5	91.44	m
b_ϕ	0.23	0.8	deg
b_θ	- 0.19	0.8	deg
b_ψ	0.15	2.4	deg

Table III. Filter Process and Measurement Noise Levels

Measurement	St. Dev.	Units
x accelerometer	0.1	m/sec^2
y accelerometer	0.1	m/sec^2
z accelerometer	0.1	m/sec^2
p rate gyro	0.02	deg/sec
q rate gyro	0.02	deg/sec
r rate gyro	0.02	deg/sec
x wind	0.6	m/sec
y wind	0.3	m/sec
y_{az}	0.05	deg
y_{el}	0.05	deg
y_{rn}	4.57	m
y_{sp}	0.5	m/sec
y_{ϕ}	.23	deg
y_{θ}	.23	deg
y_{ψ}	.23	deg

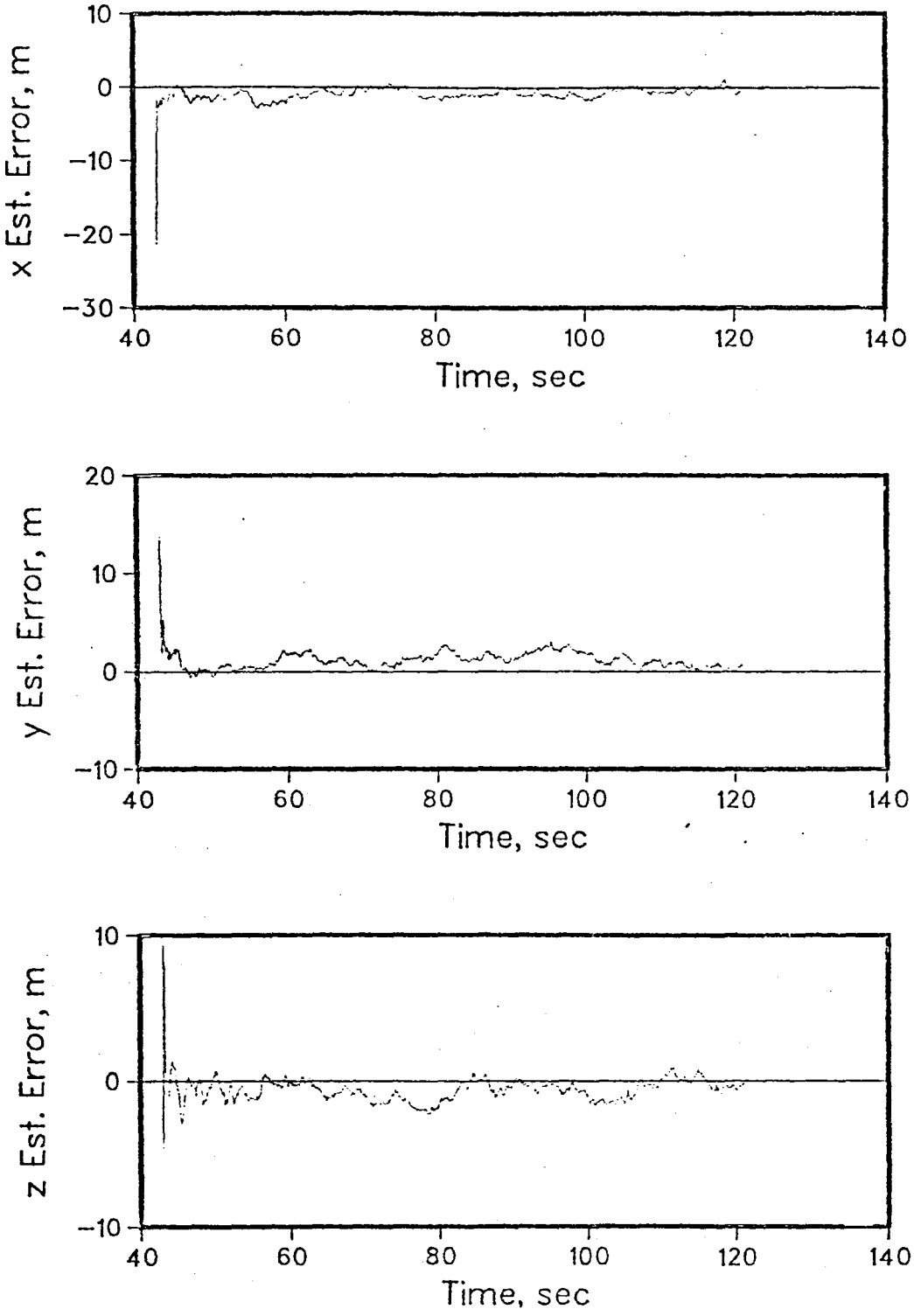


Figure 10: Position Estimation Errors – Baseline

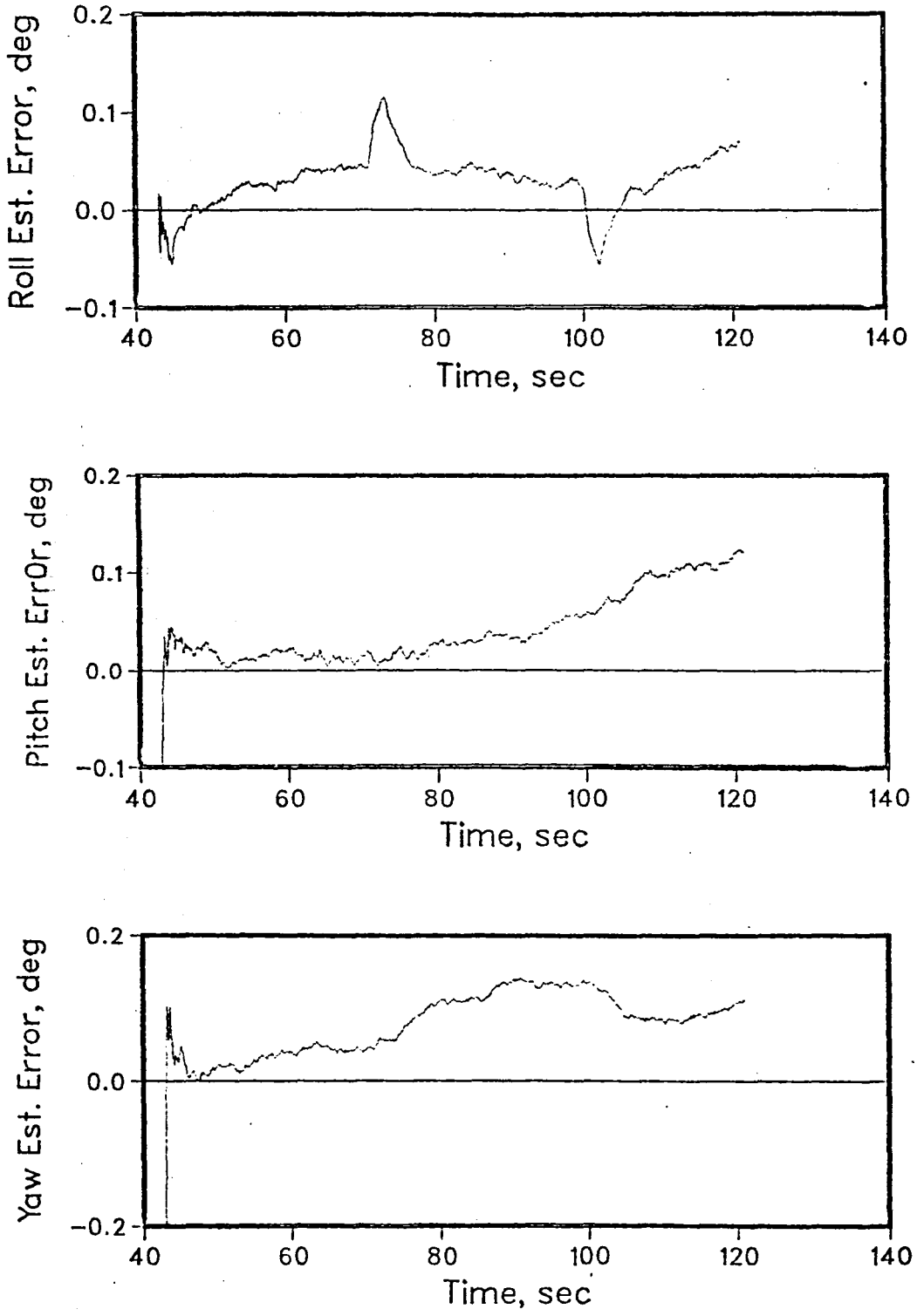


Figure 11: Attitude Estimation Errors – Baseline

mechanization of the no-fail filter. This is due to the fact that the no-fail filter is integrating the body rates too slowly. In fact, if one compares Figures 9 and 11, it will be clear that the estimation errors are correlated with rate gyro transients. The position errors are less effected by this problem since the acceleration signals do not have sharp transients and because high quality MLS measurements are filtered into the position estimate.

All of the simulation runs correspond to a dual redundant sensor configuration. As discussed in Section 2.4, there are 21 possible sensor failure modes in this case since redundant input sensors (body mounted accelerometers and rate gyros) are kept as standby equipment. Table IV describes the 21 hypotheses corresponding to various sensor failures.

In all of the following runs, the a priori probability of the no-fail hypothesis was set to 0.9999, while a priori probabilities for the remaining hypotheses, corresponding to various sensor failures, were all equal to each other. The Bayesian costs in Section 2.4 were selected such that there were no costs associated with making correct decisions and costs of making incorrect decisions were the same. Therefore, the chosen hypothesis maximizes the a posteriori probability.

Table IV. Failure Modes for Dual Redundant Sensor Configuration

Hypothesis	Failed Sensor
H_1	a_x accelerometer
H_2	a_y accelerometer
H_3	a_z accelerometer
H_4	p rate gyro
H_5	q rate gyro
H_6	r rate gyro
H_7	first MLS azimuth
H_8	first MLS elevation
H_9	first MLS range
H_{10}	first IAS
H_{11}	first IMU roll attitude
H_{12}	first IMU pitch attitude
H_{13}	first IMU yaw attitude
H_{14}	second MLS azimuth
H_{15}	second MLS elevation
H_{16}	second MLS range
H_{17}	second IAS
H_{18}	second IMU roll attitude
H_{19}	second IMU pitch attitude
H_{20}	second IMU yaw attitude
H_{21}	none that are used by the no-fail filter

The next set of Figures, 12-14, show the state estimation and the normal operating bias identification performance of the no-fail filter. As expected, the normal operating biases degrade the estimation performance of the no-fail filter. It is also seen that A/C velocity and position estimates improve as the accelerometer bias estimates converge. Furthermore, the steady-state position errors can be directly attributable to the azimuth and elevation bias levels. However, these errors will decrease as the range diminishes, thereby eliminating any need to estimate small biases on MLS azimuth and elevation. Note that the only coupling that exists between the Euler angles and the runway position and velocity is through the input partials defined in eq.(2.18). Furthermore, the strongest coupling is to the heading angle. This coupling produces a slightly larger reduction in the heading angle uncertainty compared to bank and pitch angle variances. The simulation results also suggest that the Euler angles can be computed separately without substantially lowering estimation performance, thus minimizing overall filter complexity.

Convergence performance of the accelerometer bias estimates can be seen in Figure 14. Relatively slower rate of convergence for the z-accelerometer bias estimate compared to the other two is due to the fact that the IAS essentially measures the velocity along the x and y axes. Therefore, z-accelerometer input needs to

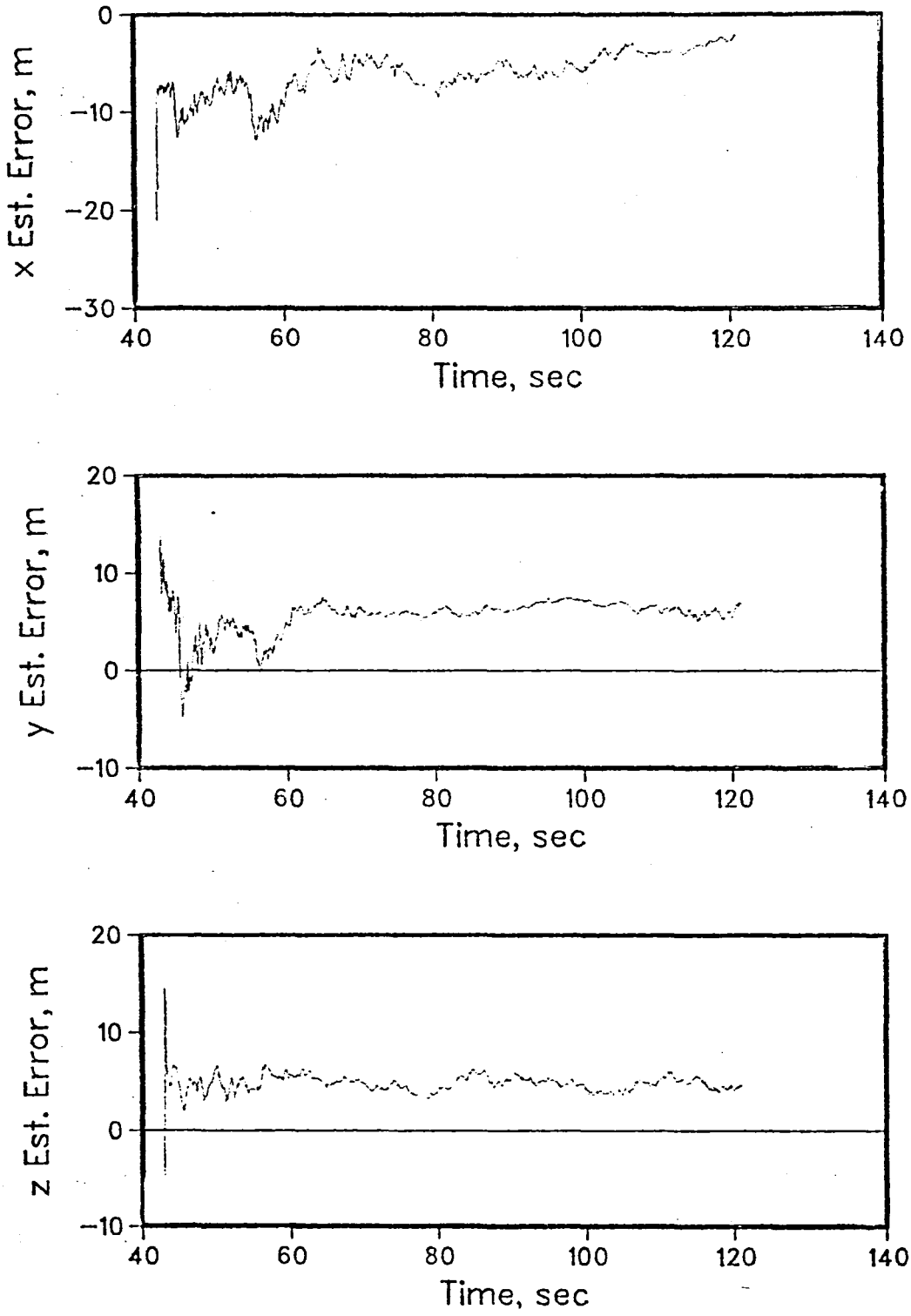
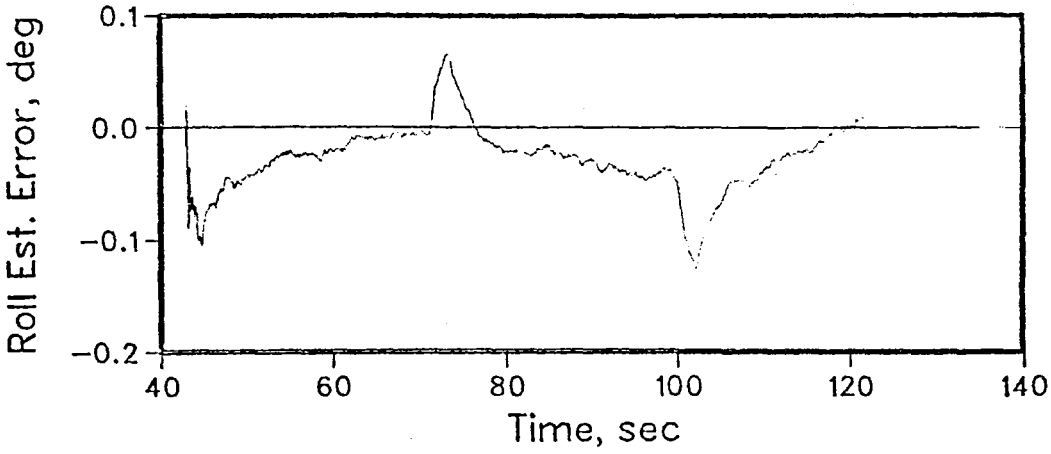
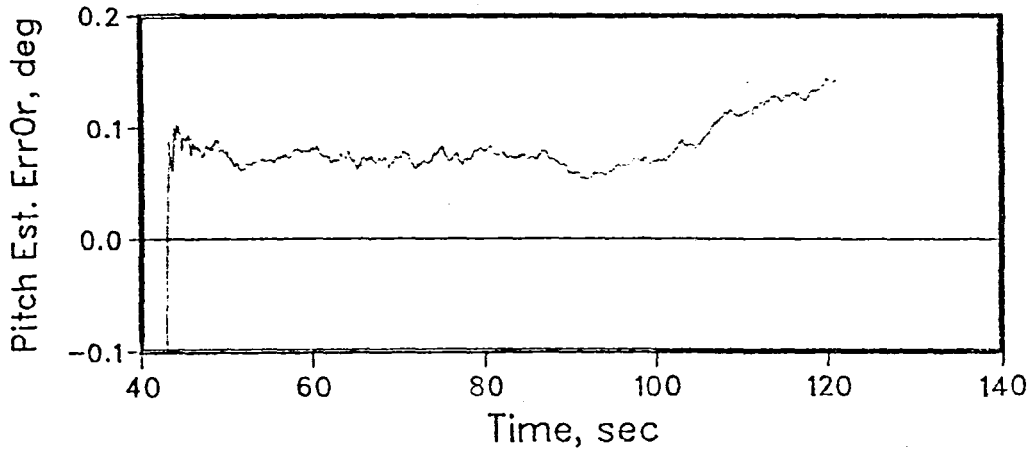


Figure 12: Position Est. Errors – Failure Free





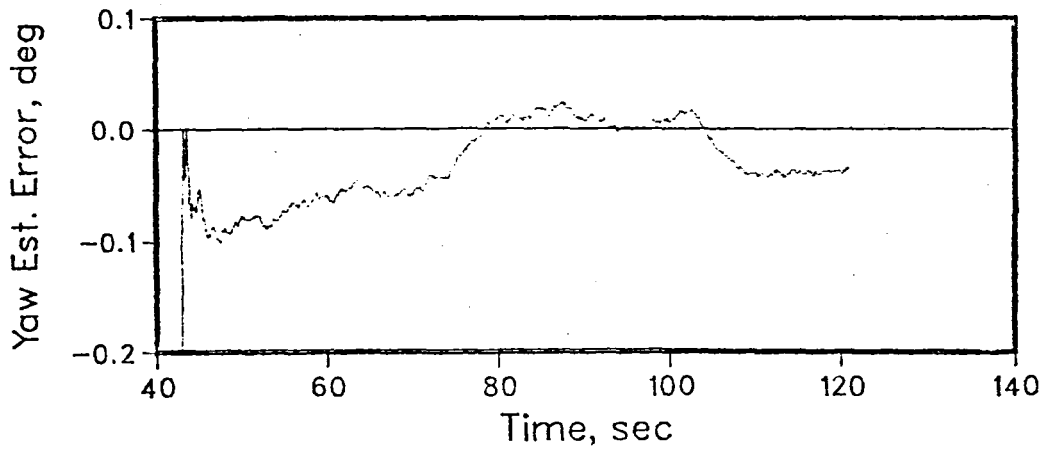
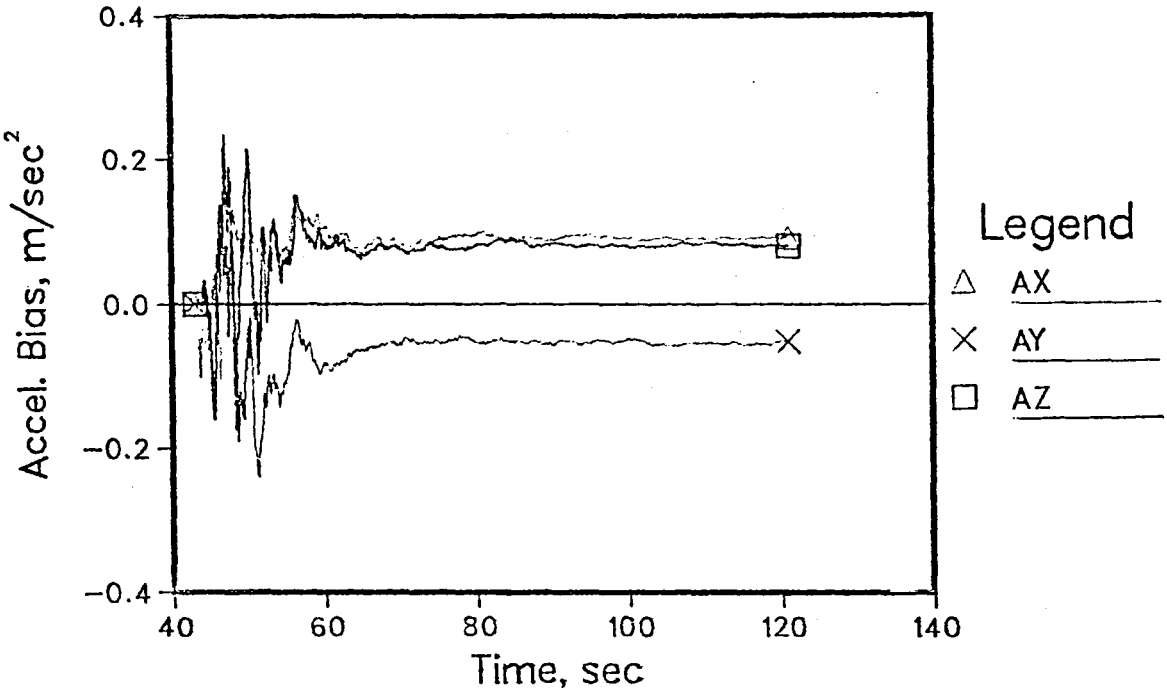


Figure 13: Attitude Est. Errors – Failure Free



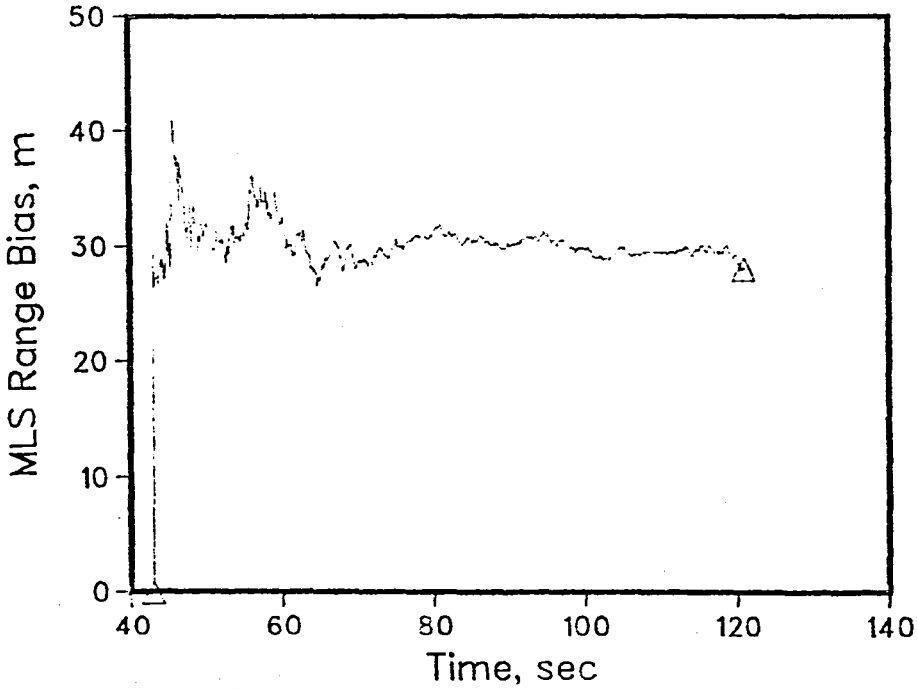


Figure 14: Normal Operating Bias Est. – Failure Free

be integrated twice for a measurement (mostly MLS elevation) comparison. The good convergence performance of the accelerometer bias estimates are typical of the results from other runs with different bias levels. The lower bound of the uncertainty in bias estimates (not shown) compares favorably to the actual estimation error.

Further note that the accelerometer bias estimates need not exactly converge to the values given in Table I due to misalignment and scale factor parameters. For instance, the z-acceleration is nearly constant in these runs so that the scale factor error associated with the z-accelerometer produces a different bias level for the z-accelerometer. Depending on the misalignment angles, the z-acceleration would also produce an additional constant bias in the x and y accelerometers. It was also observed that horizontal wind estimation performance strongly effects the settling time for the accelerometer bias estimates.

The MLS range bias estimation error is depicted in Figure 14. The relatively sharp initial change in the estimate compared to the input bias estimates was typical of the other runs in which elevation and azimuth biases were also estimated. This behavior is due to the fact that there is less filtering of the measurement residuals in obtaining the output bias estimates.

From our experience with several runs using different bias levels, initial conditions, and initial uncertainties, we have found the numerical stability of the no-fail EKF implementation to be as good as the L-D factorization formulation [21] of the EKF for the augmented system. The overall performance of the filter is typical of the EKF applications with the largest estimation errors occurring during startup and abrupt dynamic changes (in our case, during A/C maneuvers in which the linearity assumptions are most severely violated). The following tradeoffs are involved in the identification of bias parameters:

- Rate of convergence for bias parameter estimates can be accelerated by increasing the initial bias uncertainties at the expense of large transients.

- Too large a value for the initial bias uncertainty, especially for the input biases, can severely degrade estimation performance by producing large bias filter gains. A good rule of thumb is to use approximately 3-5 times the standard deviation of the expected bias level.

4.2 System Performance - Failures

In this section, some typical filter-detector simulation runs will be presented to analyze the failure detection capability of the proposed sensor fault tolerant system. Figures 15-17 show the

ultimate failure detection performance of the system. In these runs, the baseline EKF (with all sensor biases set to zero) was used to drive the detectors. As can be seen from Figures 15 and 16, the state estimation performance under failures is essentially the same as the baseline under no failures with slight expected degradation at the failure times.

In these runs, a sequence of IMU roll, MLS azimuth and accelerometer failures are simulated. For the preliminary results, only bias failures were considered. First IMU roll measurement is failed with a bias jump of 1^{o} at 78 seconds. As seen from the first graph in Figure 17, hypothesis 21 (H_{21}) is selected by the decision logic up to 75.15 seconds. Recall (from Table IV) that H_{21} signifies the decision that none of the measurements currently used by the no-fail filter have failed. At 78.15 seconds, first IMU roll attitude failure is identified by selecting hypothesis H_{11}. The faulty sensor (in this case, the entire first IMU) is rejected so that the no-fail filter starts using the attitude measurements from the second IMU instead of the average of the two IMU's. After the removal of the failed instrument, the decision logic reverts to correctly selecting H_{21}. Second MLS azimuth measurement fails with a bias jump of $.4^{o}$ at 90 seconds. This failure is detected at 90.25 seconds as indicated by the selection of hypothesis 14. Thus, second MLS azimuth sensor is removed from

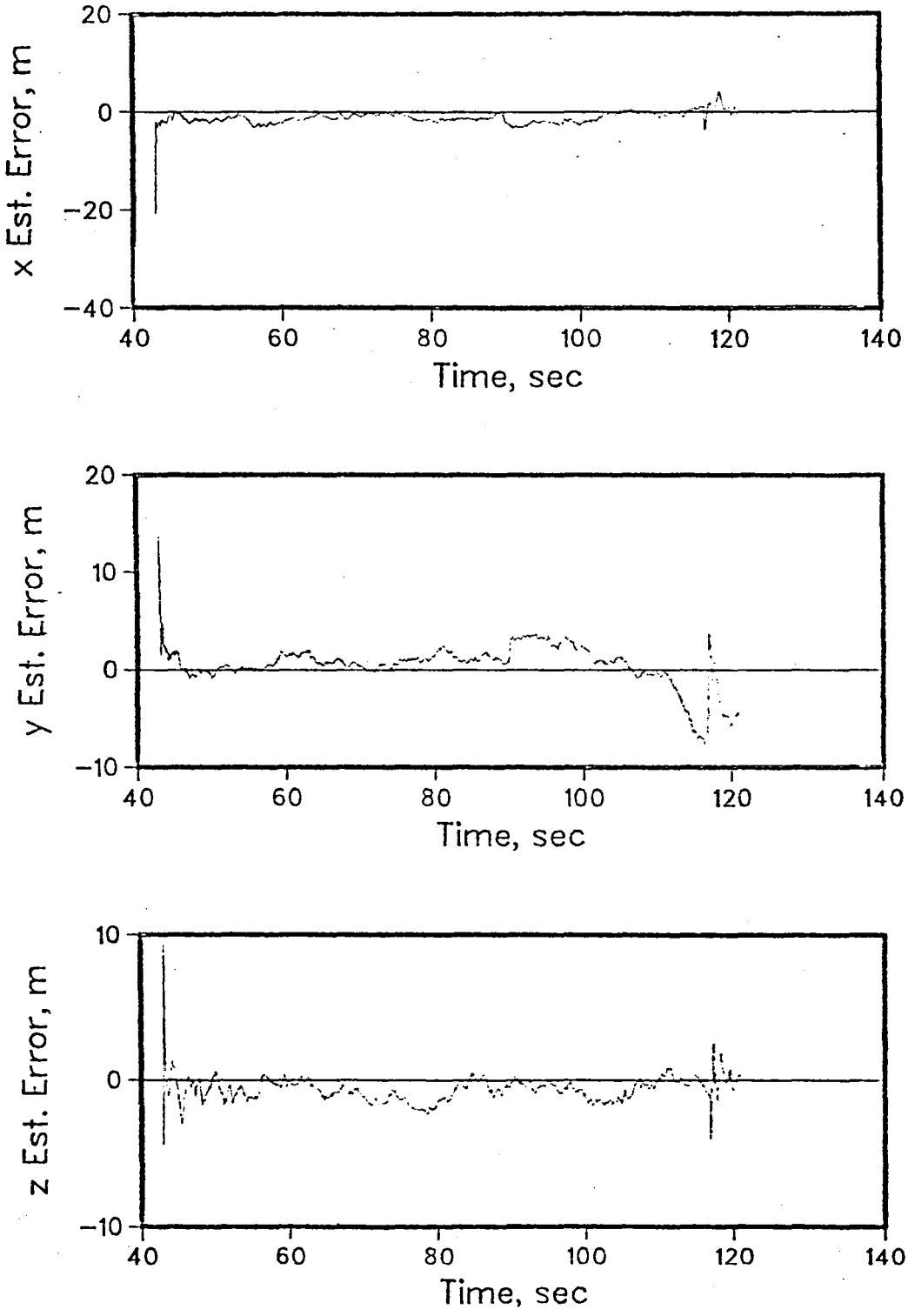
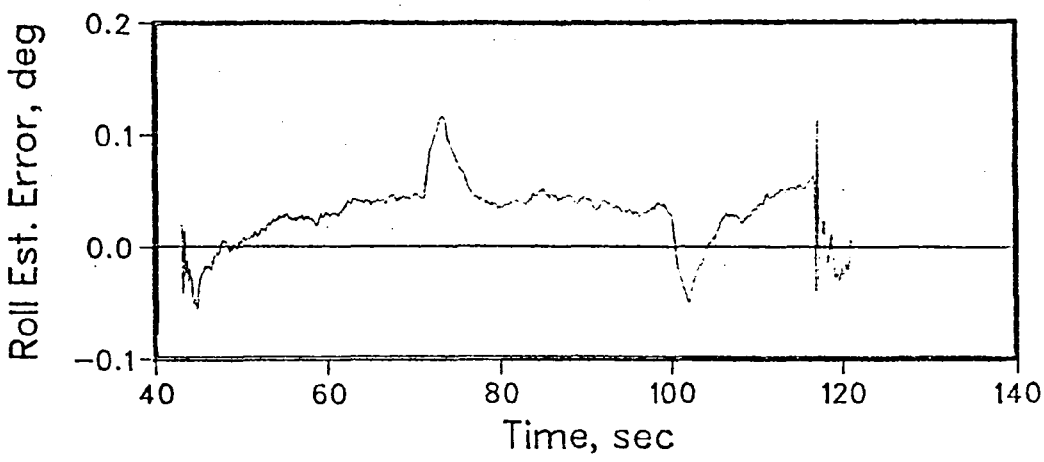
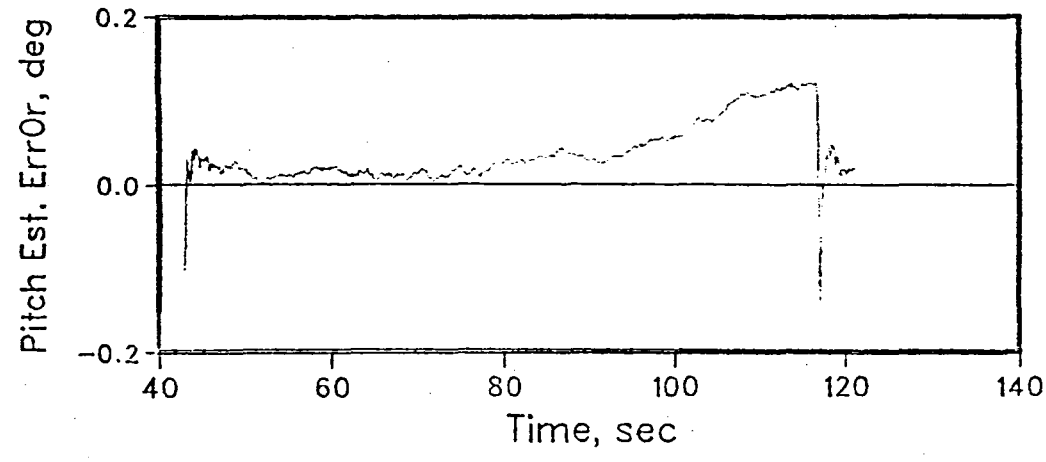


Figure 15: Position Est. Errors – Baseline with Failures





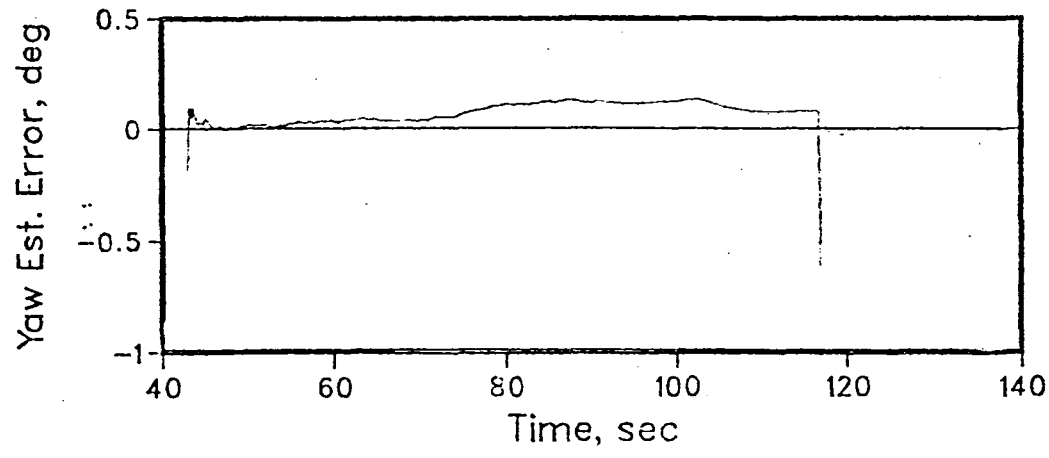


Figure 16: Attitude Est. Errors – Baseline with Failures

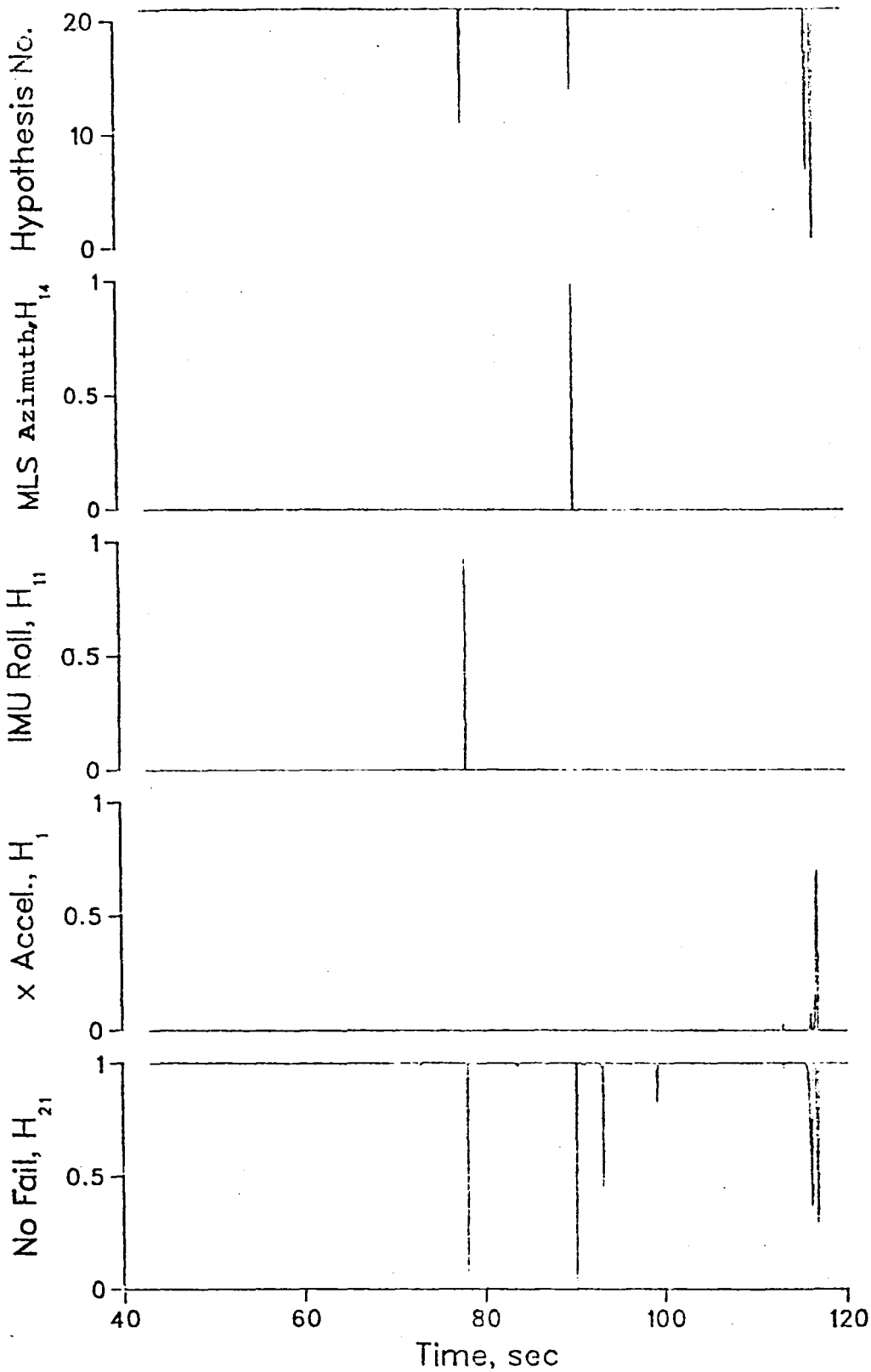


Figure 17: Selected Hypothesis and A Posteriori Time Histories - Baseline with Failures

the measurement set and the decision logic returns to correctly selecting H_{21}. Finally, the first x-accelerometer fails with a bias jump of $1m/sec^2$ at 110 seconds. This failure is detected at 116.85 seconds as indicated by the selection of H_1. At this instant, the first x-accelerometer measurement is rejected and replaced by the second x-accelerometer in the no-fail filter so that the decision logic reverts to correctly selecting H_{21} after the replacement of the faulty sensor.

Figure 17 also shows the a posteriori probabilities for hypotheses H_{14},H_{11}, H_1 and H_{21}. These probability time histories demonstrate that our decision logic is essentially equivalent to choosing the hypothesis with the largest posteriori probability. For instance, the posteriori probability for H_{11} corresponding to the second IMU roll attitude failure spikes to approximately 0.95 at 78.15 seconds while the no-fail posteriori probability simultaneously decreases to 0.05.

As can be seen from Figure 17, the detection of output sensor failures is much faster than that of input sensor failures. This is to be expected since the input sensor failures have to propagate through the no-fail filter dynamics in order to get detected. Furthermore, soft input failures would naturally take more time for detection compared to hard input failures.

The transient effects of filter-detector reconfiguration can also be seen in Figures 15-16. Notice at 117 seconds when an x accelerometer failure is detected, the errors oscillate and are driven back to near zero levels very quickly. Since by design, failures corrupt the no-fail filter estimates, it is important that the filter is able to recover from the integrated effects of these failures once they have been detected.

The next set of Figures, 18-20, show the failure detection performance of the sensor fault tolerant system with the normal operating bias filter in operation. In these runs, all sensor biases are present (see Table I). However, only MLS range and accelerometer biases are selected for identification. Again, the state estimation performance is similar to the case obtained without failures as seen in Figures 18-19. The same sequence of IMU roll, MLS azimuth and x-accelerometer failures have been simulated. Inclusion of the bias filter does not affect the detection of IMU roll and MLS azimuth failures. As can be seen in Figure 21, these failures are identified at the same instants as before. However, the soft failure in the x-accelerometer did not get identified during the course of the simulation.

As seen in Figure 21, the inclusion of the bias filter does increase the false alarm rate due to these phenomena: First, there is an interdependence between the normal operating bias filter and

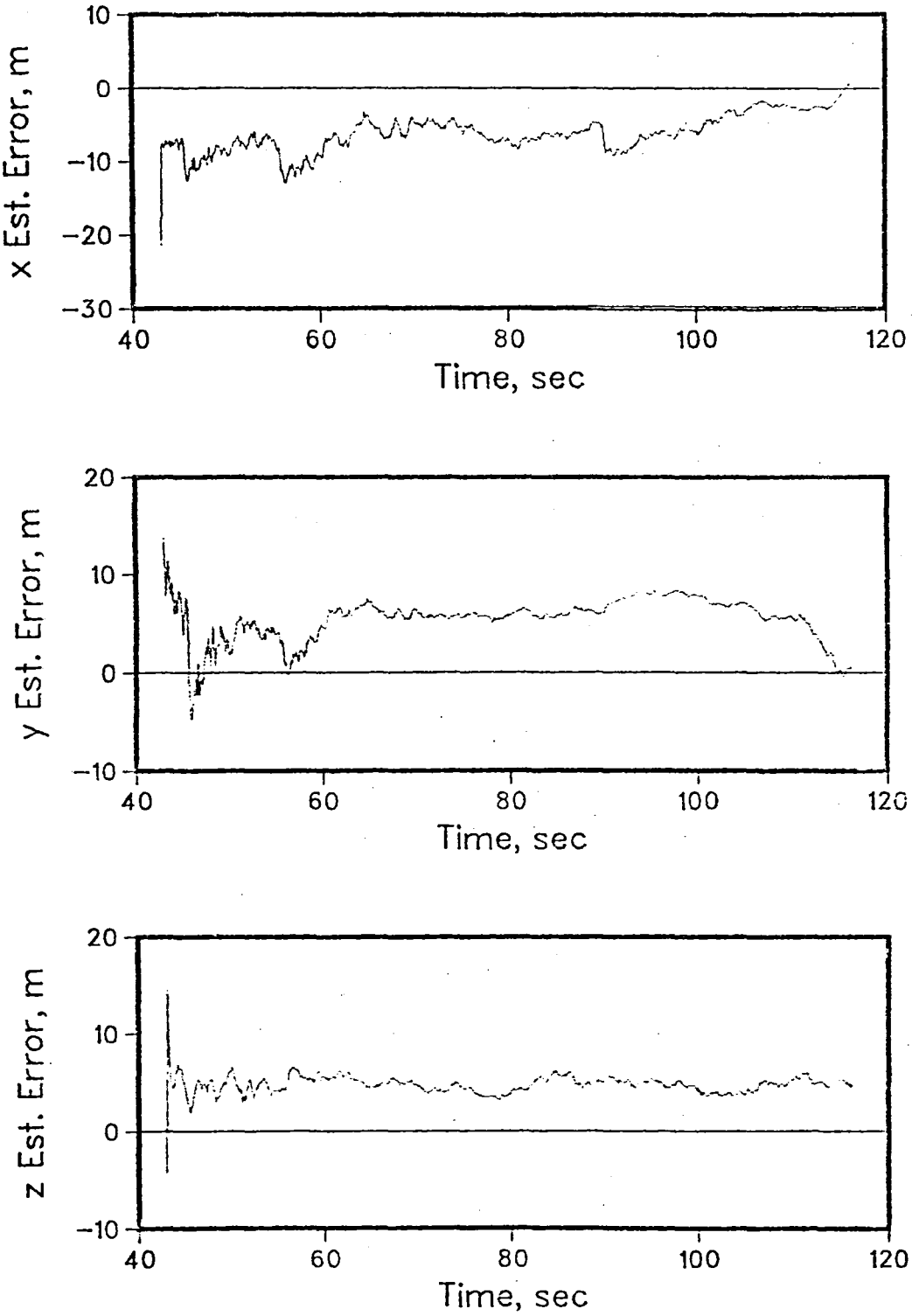
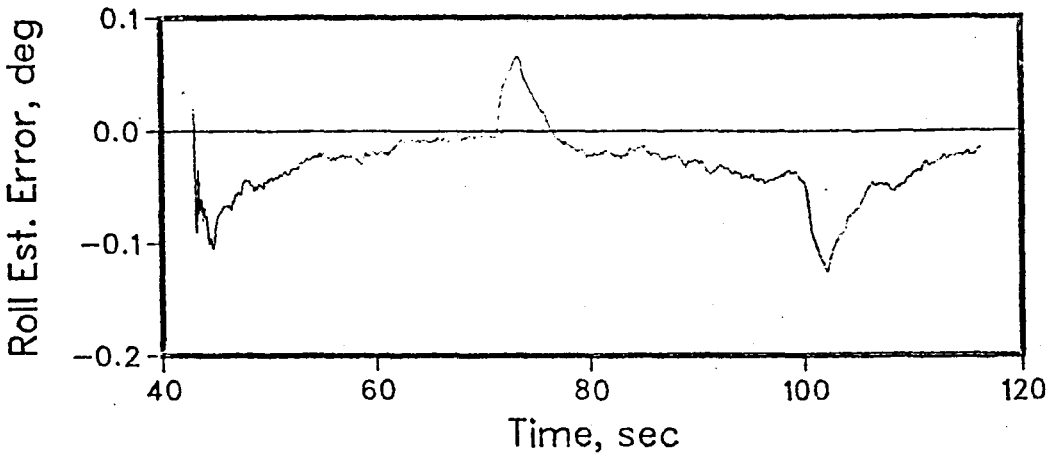
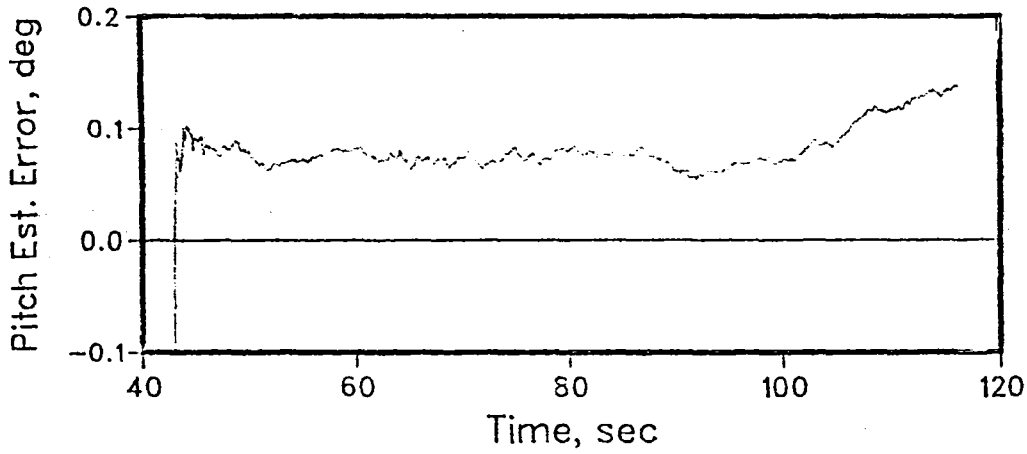


Figure 18: Position Estimation Errors – Failures





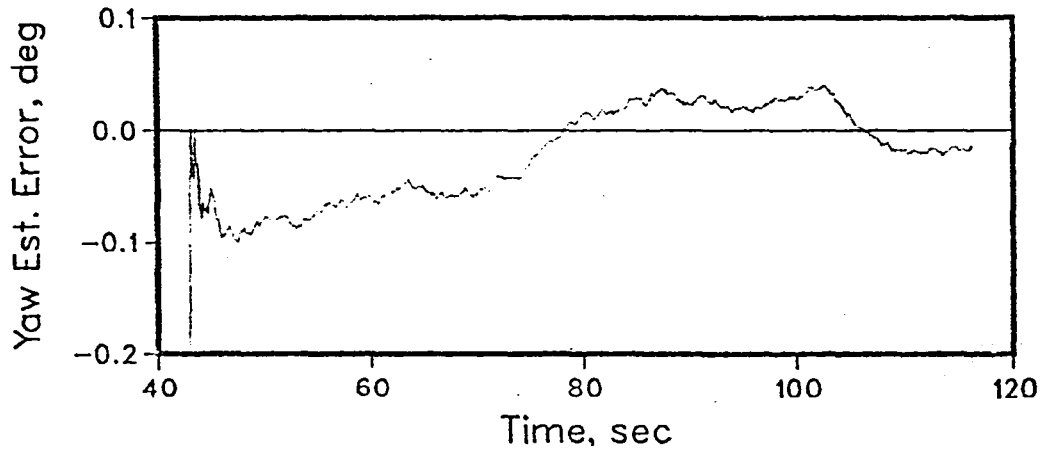
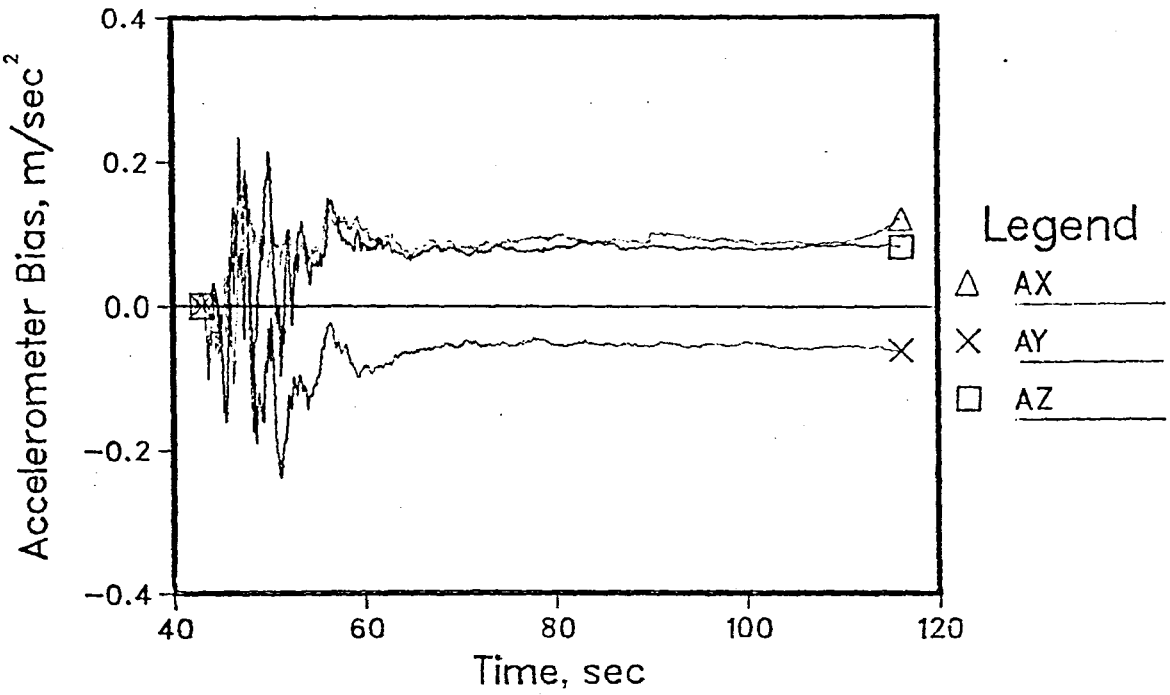


Figure 19: Attitude Estimation Errors – Failures



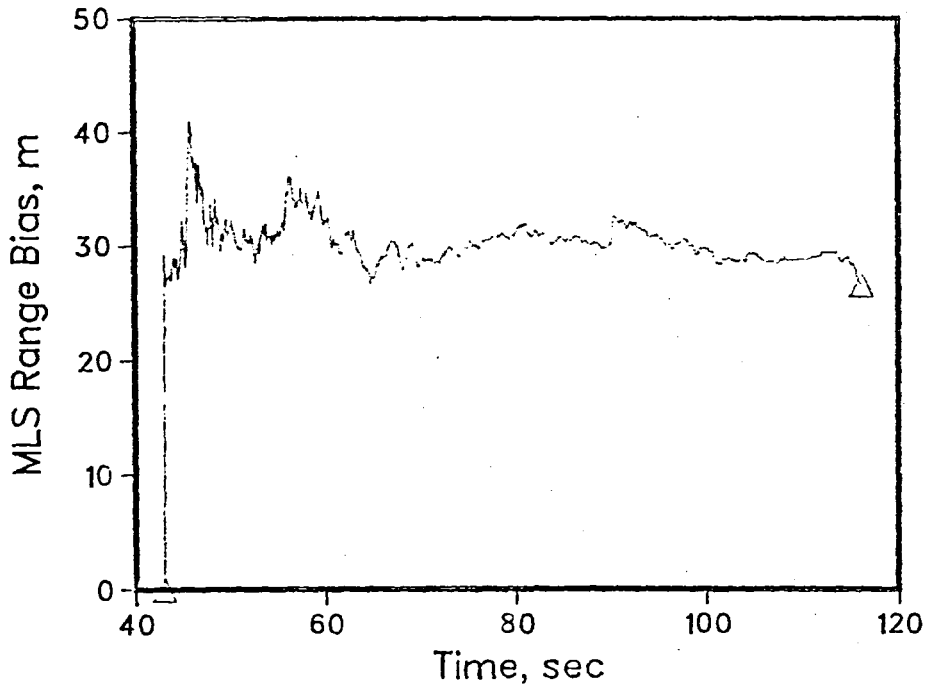


Figure 20: Normal Operating Bias Estimates – Failures

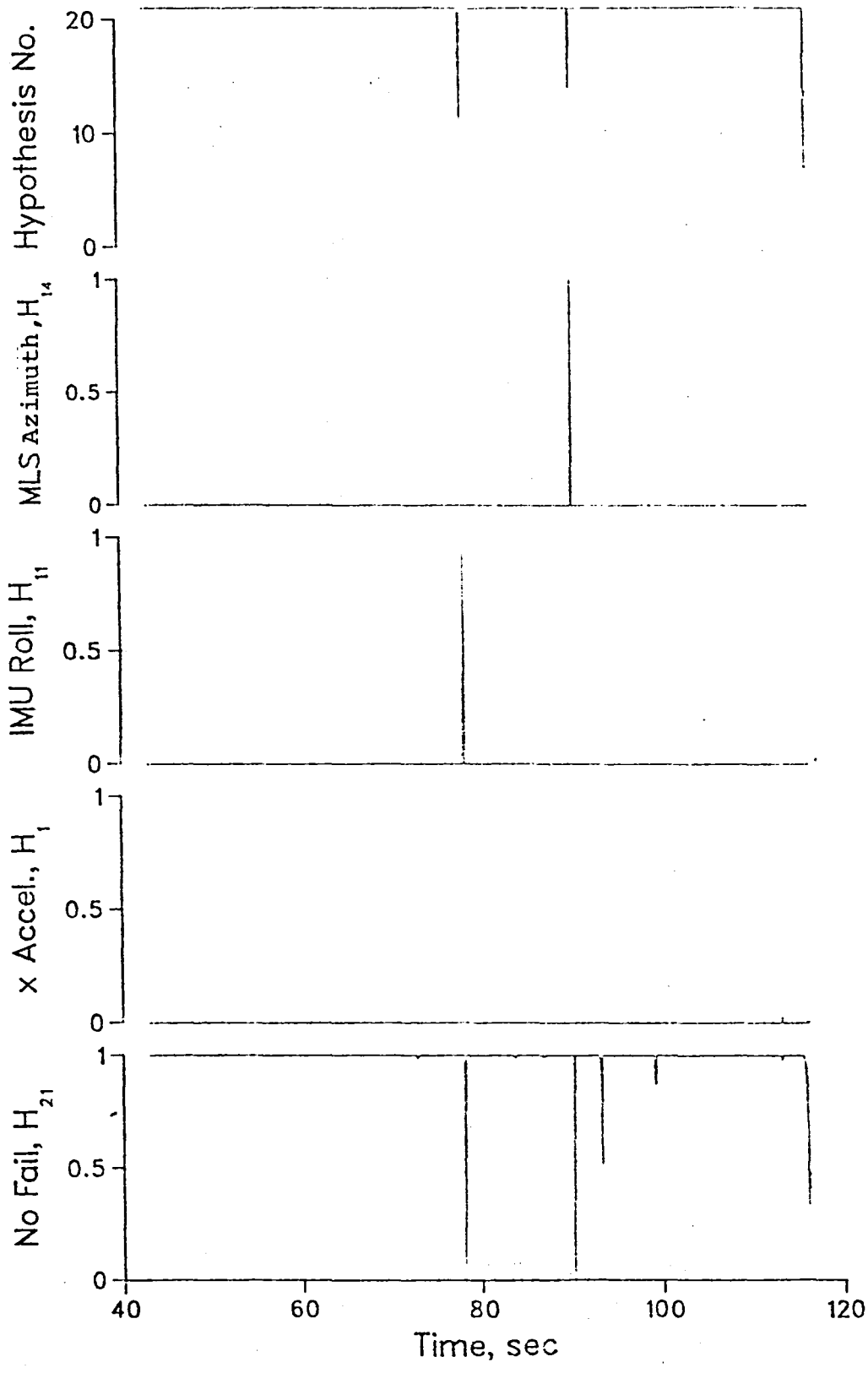
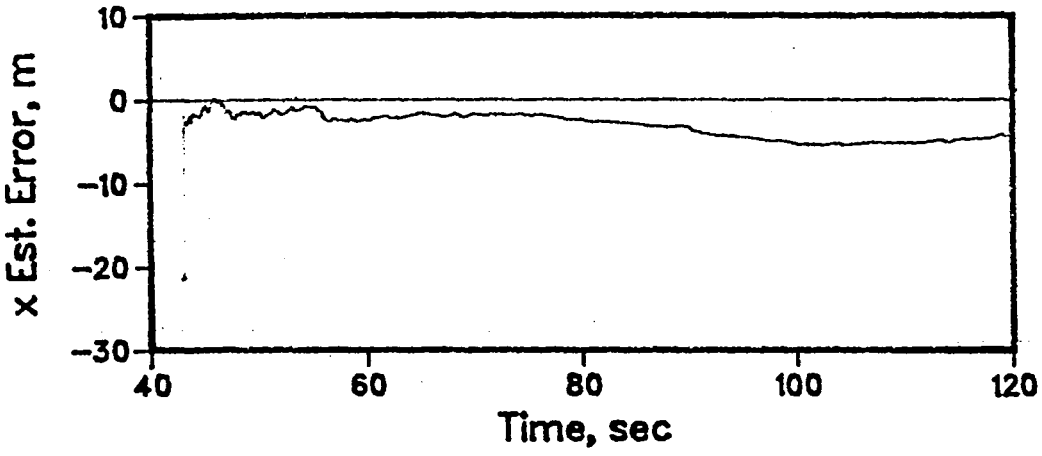


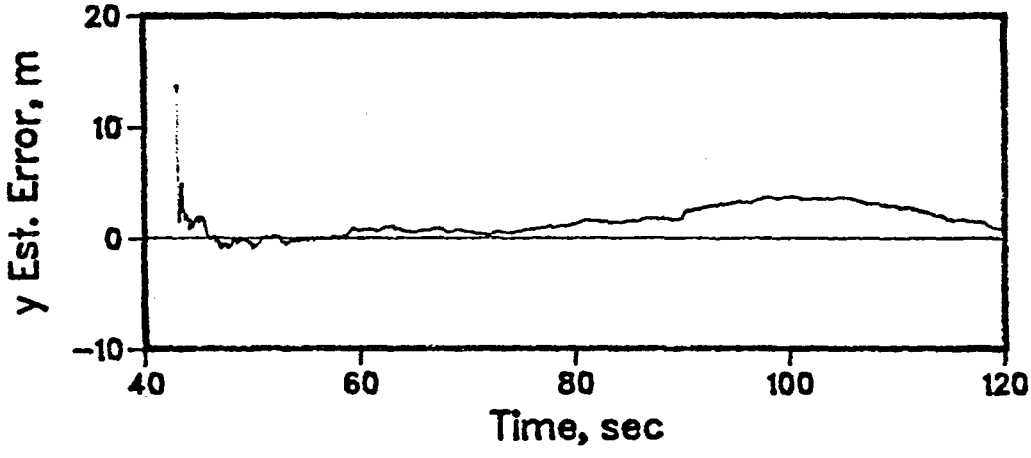
Figure 21: Selected Hypothesis and A Posteriori Probability Time Histories - Failures

those detectors for sensors which are selected for normal operating bias identification. If the normal operating biases are estimated poorly, then the bias estimation error looks like a failure to the corresponding detector. Second, the effect of ignoring normal operating biases essentially produces a similar effect. These issues will be investigated during the second year of the study. In our runs, we have used a priori probability of 0.9999 for the hypothesis H_o to compensate for the expected degradation in the no-fail filter due to sensor biases that are ignored. For instance, another possibility is the use of different Bayesian cost terms instead of the simple case utilized now.

Another possible improvement would be the use of the average of the replicated input sensors (accelerometers and rate gyros) in the no-fail filter, since this would effectively reduce the measurement noise and normal operating bias level in the averaged measurements. Currently, hardware redundant input sensors are kept as standby equipment. However, the current input detectors would need to be modified since they could not distinguish a failure between like inputs in their present form.

Figures 22-23 indicate the failure detection performance of the sensor fault tolerant system with the RSDIMU body accelerations and rates replacing the body mounted accelerometer and rate gyro measurements. A sequence of IMU roll attitude, MLS azimuth and IMU





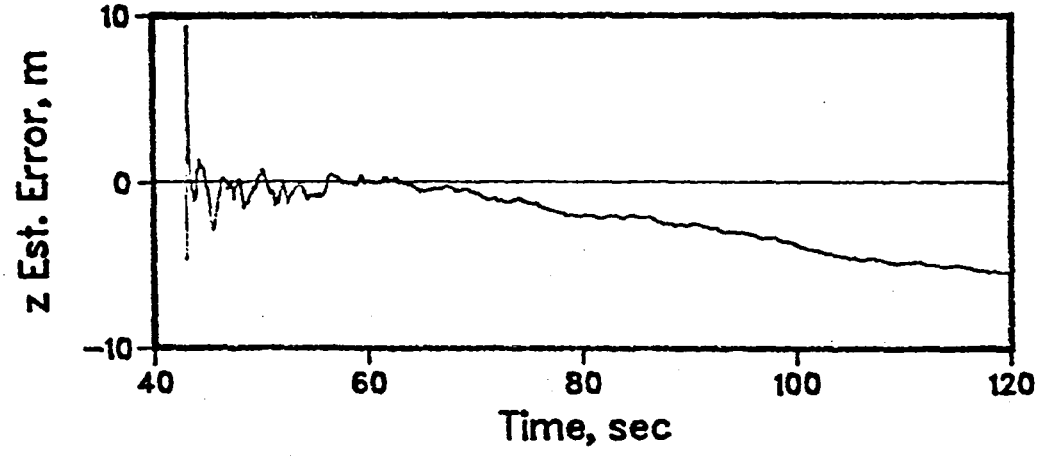
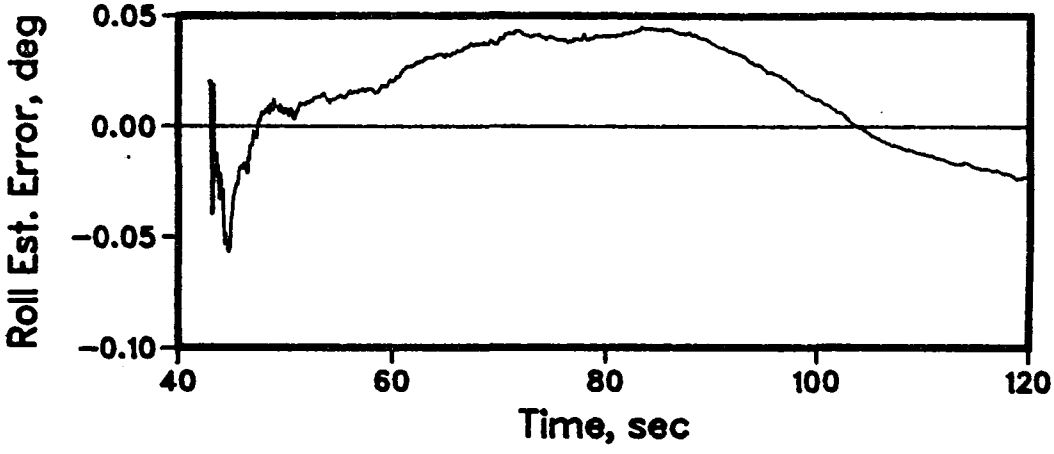
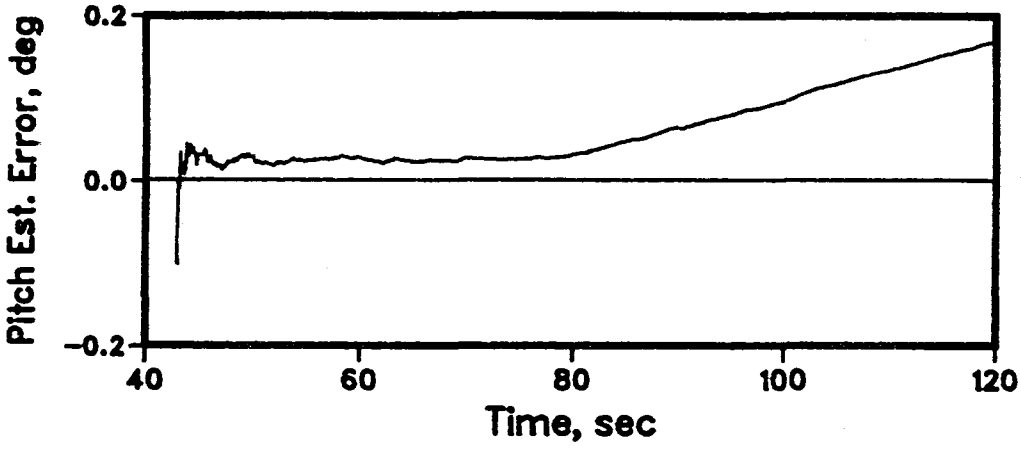


Figure 22: Position Est. Errors — Failures with RSDIMU





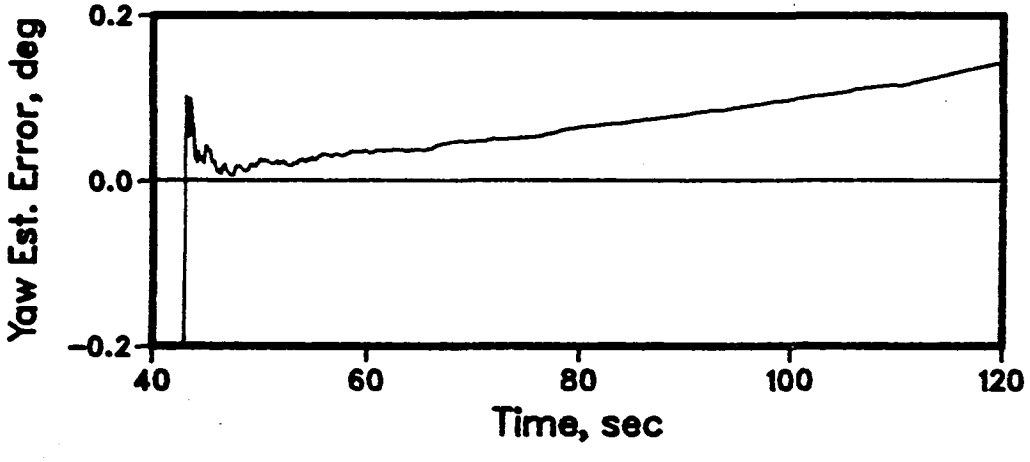


Figure 23. Attitude Est. Errors - Failures with RSDIMU

yaw attitude sensor failures are simulated. The roll attitude of the first IMU fails with a bias jump of 1^{o} at 78 sec. and is quickly identified at 78.15 seconds. The azimuth measurement of the second MLS module fails with a bias jump of .4 deg at 90 sec. and is identified at 90.25 seconds. The speed of identification is the same as the case without the RSDIMU. The yaw attitude failure of the second IMU with a bias jump of 1^{o} at 110 sec. is identified at 110.3 seconds.

As expected (see Figure 22), position estimation performance of the no-fail filter is initially significantly better than the case in which body mounted accelerometers are employed. This is due to the better accuracy of the navigation quality accelerometers and rate gyros in the RSDIMU. The ramp type position estimation errors (e.g., ramp error starting in the z-position estimation error in Figure 22) can be traced to accelerometer impulse type errors due to scale factors.

The failure detection performance (not shown) with colored MLS measurement noise was degraded due to false alarms. This is due mainly to the fact that any time correlation in the no-fail filter residuals looks like a time varying bias jump to the detectors. The sensor fault tolerant system design will be modified in the second year of the study in order to remove the effects of colored MLS noise. The state estimation performance with colored MLS noise was essentially the same as the case with white MLS noise.

Further tuning of the noise parameters will help to improve both the state estimation and failure detection performance. For instance, MLS filter noise is not accounted for in the no-fail filter. The effect of this error, due to digital scanning, would be an effectively higher MLS measurement noise.

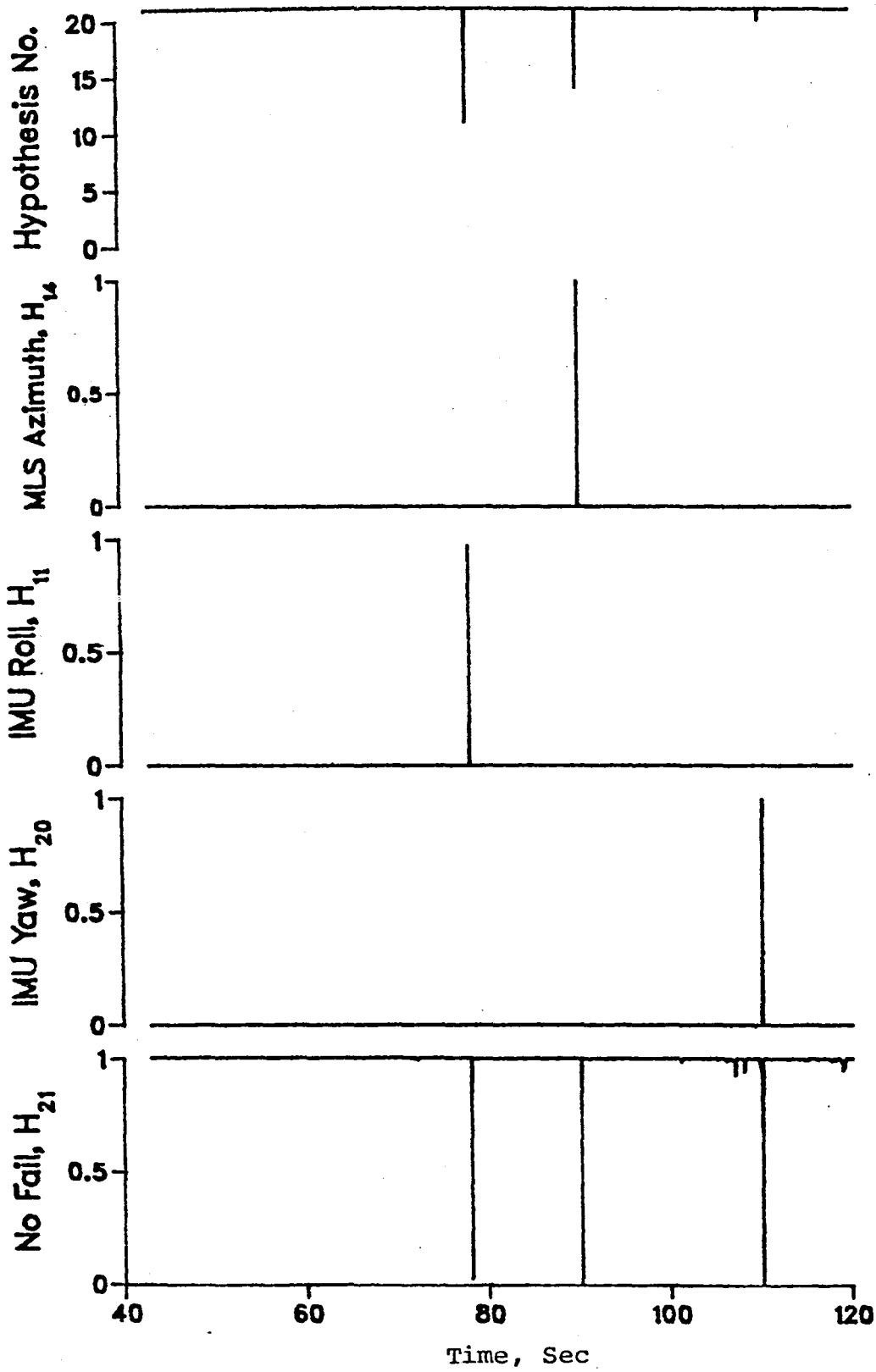


Figure 24: Selected Hypothesis and A Priori Probability Time Histories - Failures with RSDIMU

V. SUMMARY

In the preceding chapters, an aircraft sensor fault tolerant system design methodology has been presented. The design problem is formulated in the context of simultaneous state estimation and failure detection in discrete time nonlinear stochastic systems. The proposed solution involves the implementation of a no-fail extended Kalman filter and a bank of first order detectors. The no-fail EKF computes estimates for aircraft states and normal operating sensor biases on the assumption of no-failures. A new separate bias algorithm has been derived for the EKF implementation by extending the results for the linear case to nonlinear systems. The residuals of the no-fail EKF drive a bank of detectors each of which estimate a postulated sensor bias jump failure for a given sensor. Multiple hypothesis testing procedure is then employed to select the most likely failure mode in the Bayesian sense. When a failure is declared, the filter-detector structure is reconfigured by deleting the faulty sensor.

The state estimation and sensor failure detection performance of the developed aircraft sensor fault tolerant system has been analyzed on the nonlinear six-degree-of-freedom simulation of the TCV research aircraft. In our experience with several runs using different bias levels, initial conditions, and initial uncertainties, we have found the numerical stability of the no-fail

EKF to be as good as the square root algorithm. The overall performance of the filter is typical of the EKF applications with largest estimation errors occurring during startup and aircraft maneuvers. Estimation errors compare favorably to those obtained with other types of navigation filters employed in the same environment. Sensor failure detection performance of the fault tolerant system is excellent for the no-fail EKF output sensors such as MLS, IAS, IMU measurements. The failure detection speed for input sensors such as body mounted accelerometers and rate gyros is slower than that of output sensors. This is to be expected since the input sensor failures have to propagate through the no-fail filter dynamics in order to get detected. During the course of our study we have determined that the following issues should be further investigated for possible improvement of failure detection performance:

- Bias Filter/Detector Interaction: As expected, there is an interdependence between the normal operating bias filter and those detectors for sensors which are selected for normal operating bias identification. The selection of sensors for normal operating bias identification could presumably be decided for the optimization of failure detection performance.

- MLS Colored Noise: As discussed in the previous chapter, when colored MLS measurement noises are assumed to be white in the

filter design, the failure detection performance of the fault tolerant system becomes degraded due to false alarms. Various possible remedies have already been identified and a suitable modification will be incorporated into the design to alleviate this problem.

- Replicated Input Failure Identification: As proposed, the fault tolerant system uses one set of replicated body mounted accelerometers and rate gyros as input. The hardware redundant input sensors are kept as standby equipment. The possibility of using the average of these replicated-input sensors should be investigated for better estimation performance. However, the current input detectors would need to be modified since they could not distinguish a failure between like inputs.

These issues will be resolved in the second year of the study. Furthermore, failure detection performance of the fault tolerant system will be analyzed under different type failures. Tests for healing of a failed sensor will be included. Robustness of the overall system will be analyzed under steady state wind conditions and different landing paths. A sensor configuration design method will be developed so that a sensor complement with a minimum replication of sensors could be selected for a given mission reliability.

REFERENCES

[1] Duff, W.G. and Guarino, C.R., "Refinement and Validation of Two Digital Microwave Landing System (MLS) Theoretic Models", NASA CR-132713, August, 1975.

[2] Morelle, F.R. and Russell, J.G., "Design of a Developmental Dual Fail Operational Redundant Strapped Down Inertial Measurement Unit", NAECON 1980.

[3] Britting, K.R., Inertial Navigation Systems Analysis, John Wiley ' Sons, Inc., New York, 1971.

[4] Etkin, B., Dynamics of Atmospheric Flight, John Wiley ' Sons, Inc., New York, 1972.

[5] Halyo, N. and Caglayan, A.K., "A Separation Theorem for the Stochastic Sampled-Data LQG Problem", International J. of Control, Vol. 23, No. 2, pp. 237-244, February 1976.

[6] Caglayan, A.K. and Lancraft, R.E., "A Bias Identification and State Estimation Methodology for Nonlinear Systems", to be presented at 6th IFAC Symposium on Identification and System Parameter Estimation, Washington, D.C., June 7-11, 1982.

[7] Friedland, B., "Treatment of Bias in Recursive Filtering", IEEE Trans. on Automatic Control, Vol. AC-14, August 1969.

[8] Friedland, B., "Notes on Separate-Bias Estimation", IEEE Trans. on Automatic Control, Vol. AC-23, No. 4, August 1978.

[9] Tacker, E.C. and Lee, C.C., "Linear Filtering in the Presence of Time-Varying Bias", IEEE Trans. on Automatic Control, Vol. AC-17, December 1971.

[10] Tanaka, A., "Parallel Computation in Linear Discrete Filtering", IEEE Trans. on Automatic Control, Vol. AC-20, August 1975.

[11] Sinha, A.K. and Mahalanabis, A.K., "Modeling Error Compensation in Nonlinear Estimation Problems", IEEE Trans. Systems, Man, and Cybernetics, Vol. AC-18, No. 6, November 1973.

[12] Shreve, R.L. and Hedrick, W.R., "Separating Bias and State Estimates in a Recursive Second-Order Filter", IEEE Trans. Auto. Control, Vol. AC-19, No. 5, October 1974.

[13] Mendel, J.M., "Extension of Friedland's Bias Filtering Technique to a Class of Nonlinear Systems", IEEE Trans. on Automatic Control, Vol. AC-21, April 1976.

[14] Caglayan, A.K., "Necessary and Sufficient Conditions for Detectability of Jumps in Linear Systems", IEEE Trans. on Automatic Control, Vol. AC-25, No. 4, August 1980.

[15] Caglayan, A.K., "Simultaneous Failure Detection and Estimation in Linear Systems", Proc. of 1980 Conf. on Decision and Control, December 10-12, 1980, Albuquerque, NM.

[16] Jazwinski, A.H., Stochastic Processes and Filtering Theory, Academic Press, New York, 1970.

[17] Anderson, B.D.O. and Moore, J.B., Optimal Filtering, Prentice-Hall, Inc., New Jersey, 1979.

[18] Pines, S., Schmidt, S.F., and Mann, F., "Automated Landing, Rollout and Turnoff Using MLS and Magnetic Cable Sensors", NASA CR-2907, Oct. 1977.

[19] Halyo, N., "Development of a Digital Guidance and Control Law for Steep Approach Automatic Landings Using Modern Control Techniques", NASA CR-3704, February 1979.

[20] Van Trees, H.L., Detection Estimation and Modulation Theory, Academic Press, New York 1970.

[21] G.J. Bierman, Factorization Methods for Discrete Sequential Estimation, Academic Press, New York 1977.

[22] Montgomery, R.C. and Caglayan, A.K., "Failure Accommodation in Digital Flight Control Systems by Bayesian Theory", J. of Aircraft, Vol. 13, No. 2, February 1976.

[23] Montgomery, R.C. and Price, D.B., "Failure Accommodation in Digital Flight Control Systems Accounting for Nonlinear Aircraft Dynamics", J. of Aircraft, Vol. 13, No. 2, February 1976.

[24] Willsky, A.S., "A Survey of Design Methods for Failure Detection in Dynamic Systems", Automatica, Vol. 12, 1976.

APPENDIX A

EKF INPUT AND MEASUREMENT PARTIALS

This appendix contains the input and measurement partials which are needed in the implementation of the no-fail filter of Section 2.3.

Input Partials: Let $u=[a_x,a_y,a_z,p,q,r]'$ where (a_x,a_y,a_z) and (p,q,r) are the input variables associated with the body mounted accelerometers and rate gyros. Then, the input partials for the EKF will be given by:

$$\frac{\partial}{\partial x}\,(B(x(k))u) = \begin{bmatrix} \frac{\tau}{2}^2 T_{GBP} \\ \tau\, T_{GBP} \\ \tau\, T_{ERP} \\ 0 \end{bmatrix} \qquad \text{(A.1)}$$

where the matrices T_{GBP} and T_{ERP} are the partials associated with the transformations T_{GB} and T_{ER}. The nonzero elements of T_{GBP} are defined by

$$T_{GBP}(1,7) = c_1 a_y + c_2 a_z \tag{A.2}$$

$$T_{GBP}(1,8) = (-s\theta c\psi)ax + (s\theta c\theta c\psi)a_y + (c\phi c\theta c\psi)a_z \tag{A.3}$$

$$T_{GBP}(1,9) = (-c\theta s\psi)a_x + c_3 a_y + c_x a_z \tag{A.4}$$

$$T_{GBP}(2,7) = -c_x a_y + c_z a_z \tag{A.5}$$

$$T_{GBP}(2,8) = (-s\theta s\psi)a_x + (s\phi c\theta s\psi)a_y + (c\phi c\theta s\psi)a_z \tag{A.6}$$

$$T_{GBP}(2,9) = (c\theta c\psi)a_x - c_2 a_y + c_1 a_z \tag{A.7}$$

$$T_{GBP}(3,7) = (c\phi c\theta)a_y - (s\phi c\theta)a_z \tag{A.8}$$

$$T_{GBP}(3,8) = (-c\theta)a_x - (s\phi s\theta)a_y - (c\phi s\theta)a_z \tag{A.9}$$

where

$$c_1 = c\phi s\theta s\psi + s\phi s\psi \tag{A.10}$$

$$c_2 = -s\phi s\theta c\psi + c\phi s\psi \tag{A.11}$$

$$c_3 = -s\phi s\theta s\psi - c\phi c\psi \tag{A.12}$$

$$c_4 = -c\phi s\theta s\psi + s\phi c\psi \tag{A.13}$$

The nonzero elements of T_{ERP} are defined by:

$$T_{ERP}(1,7) = (c\phi t\theta)q - (s\phi t\theta)r \tag{A.14}$$

$$T_{ERP}(1,8) = (s\phi sc^2\theta)q + (c\phi sc^2\theta)r \tag{A.15}$$

$$T_{ERP}(2,7) = (s\phi)q - (c\phi)r \tag{A.16}$$

$$T_{ERP}(3,7) = (c\phi s\theta)q - (s\phi s\theta)r \tag{A.17}$$

$$T_{ERP}(3,8) = (s\phi sc\theta t\theta)q + (c\phi sc\theta t\theta)r \tag{A.18}$$

Measurement Partials: The nonzero elements of the measurement partial $H(x(k) = \frac{\partial h(x(k)}{\partial x}$ are defined by:

$$H(1,1) = \frac{r_x - x_M}{r_{az}} \tag{A.19}$$

$$H(1,2) = \frac{r_y - y_M}{r_{az}} \tag{A.20}$$

$$H(1,3) = \frac{r_z - z_M}{r_{az}} \tag{A.21}$$

$$H(2,1) = \frac{(r_x - x_M)(r_y - y_M)}{r_{az}^2 \cdot r_{xz}} \tag{A.22}$$

where $r_{xz} = \sqrt{(r_x - x_M)^2 + (r_z - z_M)^2}$

$$H(2,2) = \frac{-r_{xz}}{r_{az}^2} \tag{A.23}$$

$$H(2,3) = \frac{(r_y - y_M)(r_z - z_M)}{r_{az}^2 \cdot r_{xz}} \tag{A.24}$$

$$H(3,1) = \frac{(r_x - x_E)(r_z - z_E)}{r_{e\ell}^2 \cdot r_{xy}} \tag{A.25}$$

where $r_{xy} = \sqrt{(r_x - x_E)^2 + (r_y - y_E)^2}$

$$H(3,2) = \frac{(r_y - y_E)(r_z - z_E)}{r_{e\ell}^2 \cdot r_{xy}} \tag{A.26}$$

$$H(3,3) = \frac{-r_{xy}}{r_{e\ell}^2} \tag{A.27}$$

$$H(4,4) = \frac{\dot{r}_x - w_x}{s} \tag{A.28}$$

where $s = \sqrt{(r_x - w_x)^2 + (\dot{r}_y - w_y)^2 + \dot{r}_z^2}$

$$H(4,5) = \frac{\dot{r}_y - w_y}{s} \tag{A.29}$$

$$H(4,6) = \frac{\dot{r}_z}{s} \tag{A.30}$$

$$H(4,10) = -H(4,4) \tag{A.31}$$

$$H(4,11) = -H(4,5) \tag{A.32}$$

APPENDIX B

DERIVATION OF THE SEPARATE BIAS EKF ALGORITHM

In [7], Friedland has shown that the least mean square state estimator for a linear dynamic system augmented with bias states can be decomposed into three parts: 1) a bias-free state estimator; 2) a bias estimator; and 3) a blender. The bias-free state estimator is designed on the assumption of zero biases. The innovations of this bias-free filter are then used as measurements by the bias estimator. Finally, the bias-correction matrix computed in the blender is used to blend the bias estimators with the bias-free state estimate to obtain the optimum state estimate. Even if this new filter structure is not utilized, these results provide numerically advantageous decomposition procedures for computing the estimator gains for the composite filter corresponding to the system state augmented with the bias parameters.

Several extensions [6]-[15] of the separate bias estimation algorithm have appeared in the literature since [7]. The extension of the separate-bias estimation algorithm to nonlinear systems is, naturally, of practical interest and there has been a number of efforts [11]-[13] in this area. In [11], nonlinear system dynamics with input biases were considered, but the work was limited to

linear observations with no measurement biases. The EKF for the augmented system was implemented by using the decomposition for the prediction error covariance derived in [7]. The authors did not take advantage of the decomposition for the composite filter gain, which involves lower order matrices. Furthermore, it is not clear which estimates are used for linearizations needed in the extended Kalman filter. Separate bias estimation results have also been applied to bias identification using a second order suboptimal filter [12]. This work was concerned with state estimation in nonlinear continuous systems with discrete measurements; and, only sensor biases were considered.

The special case in which the bias enters linearly into the nonlinear system dynamics and observations was treated in [13]. In this study, an algorithm along the lines of the separate-bias estimation approach was proposed for the EKF implementation. This method required the computation of the bias correction matrix twice and thus introduced additional numerical complexity. Furthermore, the proposed estimation algorithm is not an equivalent implementation of the extended Kalman filter for the composite system augmented with the bias states but rather an approximation for it. In fact, it is shown in [6] that a separate bias-free filter structure is not possible for this class of systems.

In [6], discrete-time nonlinear stochastic systems with biases both in inputs and outputs is considered. In this work, the bias model includes input biases entering nonlinearly into the system dynamics so that the class of nonlinear systems considered encompasses the earlier studies. Friedland's separate-bias estimation algorithm is then generalized to the EKF formulation for the class of systems considered. It is shown that a separate bias-filter structure, in contrast to the linear case, is no longer possible. On the other hand, the computations for the extended Kalman filter gains can still be performed using a decomposition analogous to that of the linear filtering problem. This computational procedure for obtaining the filter gains has the following advantages over the EKF algorithms for the augmented systems: First, numerical accuracy is improved due to the lower order matrix operations involved. Secondly, zero a priori information about the bias state initial conditions can be handled by implementing the information filter form of the bias covariance equations while using the standard filtering equations for the system state covariance.

We will now outline the derivation of the separate bias EKF algorithm. Expanding the nonlinearities in (2.1) about the conditional mean, $\hat{x}(k)=E[x(k)|y(1),\ldots y(k)]$, and the nonlinearity in (2.15) about the single stage prediction, $\hat{x}(k+1/k)$, and

retaining only the first order terms in the Taylor series expansion, we obtain

$$x(k+1)=F(\hat{x}(k),\hat{u}(k))x(k)+B_b(\hat{x}(k))b_u(k)+f(k)+w(k) \quad (B.1)$$

$$y(k+1)=H(\hat{x}(k+1/k))x(k+1)+b(k+1)+z(k+1)+v(k+1) \quad (B.2)$$

where the matrices F and H are defind by eqs. (2.18)-(2.19) and in Appendix A and the vectors f(k) and z(k+1) are given by

$$f(k)=-\left.\frac{\partial B}{\partial x}(x)u\right|_{\hat{x}(k),\hat{u}(k)}\hat{x}(k) + B(\hat{x}(k))u(k)+u_g \quad (B.3)$$

$$z(k+1)=h(\hat{x}(k+1/k)-H(\hat{x}(k+1/k)\hat{x}(k+1/k) \quad (B.4)$$

An EKF [16]-[17] of order n+p (where n and p are the number of states and biases, respectively) could be obtained by applying the standard Kalman filtering algorithms to the linear system described by (B.1) and (B.2) while treating f(k) as a known input and z(k) as a known output. Instead, Friedland's bias decomposition algorithm [7]will be applied to the linear system described by (2.1)-(2.15), while treating f(k) and z(k) as known inputs and outputs. Hence, the bias-free state estimate $\hat{x}_o(k)$, which is the conditional mean, $\hat{x}(k)$, obtained with B in B.1, D in B.2, and $P_b(0)$ all set to zero, will be given by

$$\hat{x}_o(k+1)=F(\hat{x}(k),\hat{u}(k))\hat{x}_o(k)+f(k) \quad (B.5)$$
$$+K_o(k+1,\hat{x}(k+1/k))[y(k+1)-H(\hat{x}(k+1/k))\hat{x}_o(k+1/k)-z(k+1)]$$

with the filter gain K_o computed by eqs. (2.20)-(2.22). Note that the bias-free filter gain and the prediction error covariance are computed using H evaluated at $\hat{x}(k+1/k)$ and F evaluated at $\hat{x}(k)$ and $\hat{b}(k)$ as opposed to evaluating them at $\hat{x}_o(k+1/k)$ and $\hat{x}_o(k)$. This is in contrast to the linear case. That is, the bias-free filter gain is a function of the total state estimate $\hat{x}_o(k)$ which is given by

$$\hat{x}(k+1)=\hat{x}_o(k+1)+V(k+1,\hat{x}(k+1/k))\hat{b}(k+1) \tag{B.6}$$

where the bias correction matrix V is computed by using eq. (2.23). We shall now investigate how the bias-free filter given by equations (B.5)-(B.6) can be implemented. Utilizing the facts that

$$\hat{x}_o(k+1/k)=F(\hat{x}(k),\hat{u}(k))\hat{x}_o(k)+f(k) \tag{B.7}$$

$$\hat{x}(k+1/k)=A\hat{x}(k)+B(\hat{x}(k))\hat{u}(k)+u_g \tag{B.8}$$

and substituting these expressions into equations (B.31)-(B.5) employing eq. (B.6), and simplifying, we obtain for the bias-free filter

$$\begin{aligned}\hat{x}_o(k+1)=&A\hat{x}(k)+B(\hat{x}(k))\hat{u}(k)+u_g-F(\hat{x}(k),\hat{u}(k))V(k,\hat{x}(k/k-1))\hat{b}(k)\\&+K_o(k+1,\hat{x}(k+1/k))\{y(k+1)-h(\hat{x}(k+1/k))+[H(\hat{x}(k+1/k)\\&(F(\hat{x}(k),\hat{u}(k))V(k,\hat{x}(k/k-1))+B(\hat{x}(k),k))]\hat{b}(k)\}\end{aligned} \tag{B.9}$$

From the above, it is clear that the bias-free filter is <u>dependent</u> on the bias estimate, whereas it would be independent of

it in the linear case. Even when B is not state dependent (the class of nonlinear systems considered in [13]), a first order expansion about $\hat{x}_o(k+1/k)$ for $h(\hat{x}(k+1/k),k+1)$ would not result in a decoupled bias-free filter. For instance, carrying out the indicated expansion would result in the following expression for the bias-free filter residual:

$$y(k+1)-h(\hat{x}_o(k+1/k)+[H(\hat{x}(k+1/k)-H(\hat{x}_o(k+1/k)][\hat{x}(k+1/k)-\hat{x}_o(k+1/k)] \tag{B.10}$$

Therefore, the bias free filter structure of [13] can be obtained only by assuming that $H(\hat{x}_o(k+1/k)=H(\hat{x}(k+1/k)$. That is, the filter algorithm proposed in [13] is not an equivalent implementation of the composite EKF for the augmented system but rather an approximation to it. Furthermore, a nonlinear state transition model with linear observations, as explored in [12], would still not result in a decoupled bias-free filter. However, the bias-free and bias estimator equations eq. (B.9) and eq. (2.17), can be merged together by substituting them in eq. (B.6) and performing the necessary algebraic simplifications to get eqs. (2.16)-(2.17).

APPENDIX C

DERIVATION OF DETECTOR ALGORITHMS

This appendix contains the derivation of detector algorithms as outlined in Section 2.4. The detector equations are derived by analyzing the residual sequence of the no-fail filter described by eqs. (2.16)-(2.17) under a specific input failure model given by eqs. (2.35)-(2.36), or an output failure model given by eqs. (2.37)-(2.38). First, define the state and normal operating bias estimation errors of the no-fail filter for hypothesis H_i by:

$$\tilde{x}_i(k) = x(k) - \hat{x}_i(k) \tag{C.1}$$

$$\tilde{b}_i(k) = b(k) - \hat{b}_i(k) \tag{C.2}$$

where $\hat{x}_i(k)$ and $\hat{b}_i(k)$ are the state and bias estimates provided by the no-fail filter when hypothesis H_i is true. That is, the estimates $\hat{x}_i(k)$ and $\hat{b}_i(k)$ are computed by eqs. (2.16)-(2.17) with the input and output sensor models given by eqs. (2.35)-(2.36) and (2.37)-(2.38). Expanding the input nonlinearities in eq. (2.35) about $\hat{x}_i(k)$ and the output nonlinear ties in eq. (2.37) about $\hat{x}_i(k+1/k)$ and subtracting the no-fail filter equations from these expressions, we obtain the following estimation error dynamics for the no-fail filter under hypothesis H_i:

$$\begin{bmatrix} I - \begin{bmatrix} K_x H & K_x D \\ K_b H & K_b D \end{bmatrix} \end{bmatrix} \begin{bmatrix} F & B \\ 0 & I \end{bmatrix} = \begin{bmatrix} (I - K_x H) F_i & (I - K_x H) B - K_x D \\ -K_b H F_i & (I - K_b D) - K_b H B \end{bmatrix}$$

$$\begin{bmatrix} \tilde{x}_i(k+1) \\ \tilde{b}_i(k+1) \end{bmatrix} = \begin{bmatrix} I - \begin{bmatrix} K_x(k+1) \\ K_b(k+1) \end{bmatrix} & [H(\hat{x}_i(k+1/k)) \quad D] \end{bmatrix} \begin{bmatrix} F_i(\hat{x}_i(k), \hat{u}(k)) & B(\hat{x}_i(k)) \\ 0 & I \end{bmatrix} \cdot$$

$$\begin{bmatrix} \tilde{x}_i(k) \\ \tilde{b}_i(k) \end{bmatrix} + \left\{ \begin{bmatrix} B_i(\hat{x}_i(k)) \\ 0 \end{bmatrix} - \begin{bmatrix} K_x(k+1) \\ K_b(k+1) \end{bmatrix} \left\{ [H(\hat{x}_i(k+1/k)) \quad D] \begin{bmatrix} B_i(\hat{x}_i(k)) \\ 0 \end{bmatrix} + D_i \right\} \right\} m_i$$

$$+ \begin{bmatrix} I - \begin{bmatrix} K_x(k+1) \\ K_b(k+1) \end{bmatrix} & [H(\hat{x}_i(k+1/k)) D] \end{bmatrix} \begin{bmatrix} w(k) \\ 0 \end{bmatrix} - \begin{bmatrix} K_x(k+1) \\ K_b(k+1) \end{bmatrix} v(k+1) \qquad (C.3)$$

where $F_i(\hat{x}_i(k), \hat{u}(k))$, $H(\hat{x}_i(k+1/k))$ are defined by eqs. (2.41), (2.19), respectively. The filter gains $K_x(k+1, \hat{x}_i(k+1/k))$ and $K_b(k+1, \hat{x}_i(k+1/k))$ are defined by eqs. (2.17c), (2.25). Now define the variables $\tilde{x}_o(k)$ and $\tilde{b}_o(k)$ by eq. (C.3) above with m_i set to be zero. That is,

$$\begin{bmatrix} \tilde{x}_o(k+1) \\ \tilde{b}_o(k+1) \end{bmatrix} = \begin{bmatrix} I - \begin{bmatrix} K_x(k+1) \\ K_b(k+1) \end{bmatrix} & [H(\hat{x}_i(k+1/k)) \quad D] \end{bmatrix} \begin{bmatrix} F_i(\hat{x}_i(k), \hat{u}(k)) & B(\hat{x}_i(k)) \\ 0 & I \end{bmatrix} \cdot$$

$$\begin{bmatrix} \tilde{x}_o(k) \\ \tilde{b}_o(k) \end{bmatrix} + \begin{bmatrix} I - \begin{bmatrix} K_x(k+1) \\ K_b(k+1) \end{bmatrix} & [H(\hat{x}_i(k+1/k)) D] \end{bmatrix} \begin{bmatrix} w(k) \\ 0 \end{bmatrix} - \begin{bmatrix} K_x(k+1) \\ K_b(k+1) \end{bmatrix} v(k+1) \qquad (C.4)$$

Note that all of the above linearizations are made about $\hat{x}_i(k)$ and $\hat{x}_i(k+1/k)$. The reason for defining $\tilde{x}_o(k)$ and $\tilde{b}_o(k)$ is to show that expression C.4 corresponds to the no-fail filter's computed estimation error covariance under hypothesis H_i. Subtracting eq. (C.4) from (C.3), we get:

$$\begin{bmatrix} \tilde{x}_i(k) \\ \tilde{b}_i(k) \end{bmatrix} = \begin{bmatrix} \tilde{x}_o(k) \\ \tilde{b}_o(k) \end{bmatrix} + \begin{bmatrix} V_{ix}(k) \\ V_{ib}(k) \end{bmatrix} m_i \qquad \text{(C.5)}$$

where $\begin{bmatrix} V_{ix}(k) \\ V_{ib}(k) \end{bmatrix}$ is defined by eq. (C.5). Using the relationship above, we obtain for the residual of the no-fail filter under hypothesis H_i:

$$r(k+1)=C_i(k+1,\hat{x}_i(k+1/k))m_i+\tilde{r}(k+1) \qquad \text{(C.6)}$$

where $\tilde{r}(k+1)$ is defined by

$$\tilde{r}(k+1)=H(\hat{x}_i(k+1/k))\tilde{x}_o(k+1)+D\tilde{b}_o(k+1)+v(k+1) \qquad \text{(C.7)}$$

In the case of output failures, the computed innovations statistics of the no-fail filter would be equal to the statistics of $\tilde{r}(k)$ defined above. For input sensor failures, the computed innovations statistics will be an approximation to the statistics of $\tilde{r}(k)$ due to the dependence of F_i in eq. (C.4) on the failure level m_i.

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA CR-165876		

4. Title and Subtitle	5. Report Date
An Aircraft Sensor Fault Tolerant System	April 1982
	6. Performing Organization Code

7. Author(s)	8. Performing Organization Report No.
Alper K. Caglayan and Roy E. Lancraft	4858
	10. Work Unit No.

9. Performing Organization Name and Address	11. Contract or Grant No.
Bolt Beranek and Newman Inc. 10 Moulton Street Cambridge, MA 02238	NAS1-16579
	13. Type of Report and Period Covered
12. Sponsoring Agency Name and Address NASA Langley Research Center Hampton, VA 23665	Contractor Report
	14. Sponsoring Agency Code

15. Supplementary Notes

Langley Technical Monitor: Frederick R. Morrell
Interim Report

16. Abstract

This report describes the design of a sensor fault tolerant system using analytic redundancy for the NASA TCV research aircraft in a Microwave Landing System (MLS) environment. The objective of the fault tolerant system is to provide reliable estimates for the aircraft states in the presence of possible failures in navigation aid instruments and on-board sensors. The developed sensor fault tolerant system consists of 1) a no-fail estimator which is an extended Kalman filter (EKF) based on the assumption of no failures, and provides estimates for aircraft position, velocity, attitude, horizontal winds, and normal operating sensor biases; 2) a bank of detectors, driven by the residuals of the no-fail filter, which are first order filters for estimating bias jump failures in sensor outputs; 3) likelihood ratio computers; and 4) a decision function which selects the most likely failure mode based on the likelihood ratios. Sensor failures are identified by utilizing the analytic relationship between various sensor outputs arising from the aircraft point mass equations of motion. Preliminary simulation results indicate an estimation performance comparable to that of other navigation filters utilized in the same environment and an excellent failure detection performance for the EKF output sensors such as MLS, IAS, and IMU measurements.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Sensor failure detection; Analytic redundancy; Fault tolerant aircraft navigation; Extended Kalman filter; Separate bias estimation algorithm	Unclassified-Unlimited Subject Category - 04

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	112	

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

End of Document