1986

GPS integrity: failure detection by autonomous means in a flight environment

Patrick Yung Chih Hwang

Iowa State University

Follow this and additional works at: https://lib.dr.iastate.edu/rtd

Part of the Electrical and Electronics Commons

Recommended Citation
Hwang, Patrick Yung Chih, "GPS integrity: failure detection by autonomous means in a flight environment " (1986). Retrospective Theses and Dissertations. 8256.
https://lib.dr.iastate.edu/rtd/8256

This Dissertation is brought to you for free and open access by the Iowa State University Capstones, Theses and Dissertations at Iowa State University Digital Repository. It has been accepted for inclusion in Retrospective Theses and Dissertations by an authorized administrator of Iowa State University Digital Repository. For more information, please contact digirep@iastate.edu.
INFORMATION TO USERS

While the most advanced technology has been used to photograph and reproduce this manuscript, the quality of the reproduction is heavily dependent upon the quality of the material submitted. For example:

- Manuscript pages may have indistinct print. In such cases, the best available copy has been filmed.

- Manuscripts may not always be complete. In such cases, a note will indicate that it is not possible to obtain missing pages.

- Copyrighted material may have been removed from the manuscript. In such cases, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, and charts) are photographed by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each oversize page is also filmed as one exposure and is available, for an additional charge, as a standard 35mm slide or as a 17”x 23” black and white photographic print.

Most photographs reproduce acceptably on positive microfilm or microfiche but lack the clarity on xerographic copies made from the microfilm. For an additional charge, 35mm slides of 6”x 9” black and white photographic prints are available for any photographs or illustrations that cannot be reproduced satisfactorily by xerography.
Hwang, Patrick Yung Chih

GPS INTEGRITY: FAILURE DETECTION BY AUTONOMOUS MEANS IN A FLIGHT ENVIRONMENT

Iowa State University

University Microfilms International

300 N. Zeen Road, Ann Arbor, MI 48106
PLEASE NOTE:

In all cases this material has been filmed in the best possible way from the available copy. Problems encountered with this document have been identified here with a check mark ✓.

1. Glossy photographs or pages 
2. Colored illustrations, paper or print 
3. Photographs with dark background 
4. Illustrations are poor copy 
5. Pages with black marks, not original copy 
6. Print shows through as there is text on both sides of page 
7. Indistinct, broken or small print on several pages ✓
8. Print exceeds margin requirements 
9. Tightly bound copy with print lost in spine 
10. Computer printout pages with indistinct print 
11. Page(s) lacking when material received, and not available from school or author.
12. Page(s) seem to be missing in numbering only as text follows.
13. Two pages numbered. Text follows.
14. Curling and wrinkled pages 
15. Dissertation contains pages with print at a slant, filmed as received ✓
16. Other

University Microfilms International
GPS integrity: Failure detection by autonomous means in a flight environment

by

Patrick Yung Chih Hwang

A Dissertation Submitted to the Graduate Faculty in Partial Fulfillment of the Requirements for the Degree of DOCTOR OF PHILOSOPHY

Department: Electrical Engineering and Computer Engineering

Major: Electrical Engineering

Approved:

Signature was redacted for privacy.

In Charge of Major Work

Signature was redacted for privacy.

For the Major Department

Signature was redacted for privacy.

For the Graduate College

Iowa State University
Ames, Iowa

1986
TABLE OF CONTENTS

I. INTRODUCTION
   A. The Failure Detection Problem 1
   B. Application to GPS Integrity in Aircraft Navigation 4
   C. Scope of Study 6

II. BACKGROUND MATERIAL
   A. Global Positioning System
      1. General description 7
      2. Performance and selective availability 10
      3. Extra-satellite coverage and GDOP 11
   B. Kalman Filtering
      1. The discrete Kalman filter 13
      2. The GPS measurement model and the extended Kalman filter 18
      3. Random process models 20
      4. Discrete-time dynamical process 23
   C. Magill Adaptive Scheme
      1. Parallel filters 31
      2. Likelihood functions and posterior probabilities 33
      3. Multiple hypothesis testing 37
   D. Brief Survey of Failure Detection Techniques
      1. Voting systems 40
      2. Failure-sensitive filters 40
      3. Multiple hypothesis 41
      4. Innovations-based detection systems 42

III. STATISTICS OF FAILURE DETECTION
   A. Effects of an Additive Deterministic Failure on the State Estimate 44
   B. Effects of a Failure on the Measurements Residuals 52
B. Corrective Action Upon Failure Recognition 128
C. Estimation of Computational Requirements 129

IX. CONCLUSIONS 131
A. Summary of Findings 131
B. Recommendations for Future Research 132

X. REFERENCES 134

XI. ACKNOWLEDGEMENTS 138

XII. APPENDIX 139
A. Uncorrelated Measurement Residuals Affected By Failure 139
B. Computer Program Listings 142
I. INTRODUCTION

In aviation, civil or military, it is certainly a prime requisite to be able to detect navigation system failures quickly and alert the flight crew accordingly. There has been much interest in the use of the much-vaunted Global Positioning System (GPS) for navigation in civil aviation although many concerns regarding its integrity are still in need of resolution. This study is part of a larger effort to determine if GPS navigation possesses enough redundancy to realize that objective. The following subsections will provide introductory concepts to failure detection, the role of failure detection in the autonomous GPS navigation problem in civil aviation, as well as provide the scope of this study.

A. The Failure Detection Problem

The subject of failure detection is by no means a new one. An excellent reference can be found in a survey paper by A. S. Willsky [1], and an especially comprehensive bibliographic listing is provided in a paper by T. Kerr [2]. To place the role that failure detection plays in estimation theory into proper perspective, it can be considered as one of a few extensions of the regular filtering problem that have branched out and developed on their own in the more recent past. Other extensions include the parameter identification problem which is concerned with estimating the parameters of the process model, and that of model identification which deals with questions of robustness and model type uncertainties. Unlike these "broader"
issues, the problem of failure detection restricts itself primarily to uncertainties that corrupt the intrinsic measurements.

To illustrate the idea of failure detection in the context of navigation, consider the lines-of-position problem in a planar navigation setting. Each line-of-position (LOP) represents the locus of points that the user's position might lie as seen by one of the reference sources. To solve the two-dimensional positioning problem, two of these range measurements are needed. This is the minimal system. When an error creeps into one of the measurements, the user obtains an erroneous solution (Figure 1.1) but will remain oblivious to this occurrence.

In Figure 1.2, with an additional measurement, a gross disagreement among the three LOP's is a sure indication that a failure is present in one of the measurements. Still, there is not enough information to identify which of
Figure 1.2. Lines-of-Position from 3 measurements

the three is errant. With yet a fourth measurement in this case (Figure 1.3), we finally have sufficient redundancy to carry out both failure detection and identification. Here, the majority rules; the set of three measurements that agrees on the solution automatically isolates the remaining guilty component.

Figure 1.3. Lines of Position from 4 measurements
This example provides a simple mechanism to illustrate the idea of failure detection and identification. Yet, in the GPS situation, the problem is immensely complicated by a host of other factors. One of these is the due to the inclusion of the clock bias error into the measurements. These time-corrupted measurements are commonly called pseudoranges. In the ordinary positioning problem using pseudoranges, the solution is not at all obvious from graphical procedures as is the case above with range measurements. Along those lines, the solution to the problem of failure detection is likewise nontrivial.

B. Application to GPS Integrity in Aircraft Navigation

The overall objective of this study is to determine if there is enough redundancy in the Global Positioning System to detect the occurrence of a satellite failure, particularly, subtle ones that could deceptively be very harmful to the unsuspecting user. As a result, the specification relating to speed of detection will be of paramount importance in this assessment.

Although it is imperative to detect the failure as soon as possible after it occurs, as with most systems there is usually a margin of grace to encroach on before the situation becomes unduly hazardous. There are two flight environments that must be considered separately because of the differences in their tolerance for errors. In the enroute or cruise situation where the aircraft is in level flight at high altitudes, the accuracy level might be as large as 1000 meters, while it could be as small as 100 meters for the final approach phase prior to landing.
There are many different failure modes that may possibly be encountered in the GPS system. Those that affect the quality of the signal transmission, which would also include the complete loss of signal, are occasionally referred to as hard or catastrophic failures. The detection problem for these cases is, for the most part, trivial. The more subtle or soft failures are largely caused by a degradation in the accuracy of the information disseminated by the satellite. Although there may be a variety of sources for those types of failures, the one that commands the most attention is the satellite clock, a key "participant" in the efficient performance of GPS navigation. One possible failure mode associated with the clock is a sudden shift in the oscillator frequency. This has the effect of adding a ramp to the pseudorange measurement originating from that particular satellite. Additionally, some failures may put the clock into a nonstationary drift (random walk) as well.

The question of redundancy is an important issue. This study attempts to approach the failure detection problem with as little help as possible outside the usual set of four satellite measurements normally available for navigation. Our ultimate goal, of course, is to be able to achieve autonomous navigation using a regular GPS receiver with failure detection capabilities all available to a flight crew in the cockpit of an aircraft. On the other hand, where needed in failure detection, many forms of aiding can be enlisted. The most available is perhaps the barometric altimeter that provides reliable altitude information and is standard equipment in any aircraft. On the other hand, the least accessible and most exotic is a proposed separate monitoring network which would comprise ground stations and a communications
satellite [3]. A number of possibilities lie in between. For instance, a simple
cost-effective alternative would be to use measuring instruments, such as the
altimeter or inexpensive accelerometers, integrated into a suitable detection
scheme. Another proposal calls for placing one or more GPS-like satellite into
going synchronous orbits for the explicit purpose of increasing the number of
available satellites over the Continental United States.

There are numerous options available and still time enough to explore
the possibilities before GPS goes into full operation.

C. Scope of Study

The objective of this project was to develop a Kalman filter-based
detection scheme suitable for detecting "soft" failures. Demonstration of its
performance was to be accomplished by Monte Carlo computer methods
simulating an aircraft navigating in flight with GPS satellites in view. This
scheme was then tested out in different scenarios using different aiding
sources where necessary. An 8-state Kalman filter was used originally,
although an 11-state filter was later developed to handle a higher dynamical
environment. The incorporation of doppler (range rate) measurements was
investigated. Time did not permit inclusion of studying the effects of
selective availability and it remains as one of the topics for future efforts.
II. BACKGROUND MATERIAL

Before getting down to the central research issue, a number of topics have been included in this chapter on background material. These topics, in themselves, are too diverse and wide a scope to cover appreciably within the limitations of this thesis. Hence, only scant treatment can be given for the most part although care was taken to include the most pertinent issues within the confines of the subject.

A. Global Positioning System

The most elaborate navigation system thus far conceived, widely known as GPS, is scheduled to be put into place by the end of this decade. It will end up serving the needs of many, both in the civil as well as the military communities. Sometimes referred to under another acronym as NavSTAR (Navigation Satellite Timing And Ranging), this system has been under development, under the auspices of the Department of Defense (DoD), for the past ten years or more.

1. General description

The navigation system is designed to provide global coverage to users possessing the capability of receiving and decoding the transmissions that will passively emanate from a constellation of special-purpose satellites. Already, a host of exciting applications have been spawned from the promise of the high-precision performance that has been deemed extractable from
this system, as evidenced by the amount of research effort dedicated thus far to it. The three segments that make up GPS, namely, space, control, and user, will now be described.

The space segment of GPS will consist of 18 satellites in six 12-hour orbits with three spare satellites judiciously strewn into orbits that will provide effective backup support. These orbits are inclined at 55° to the equatorial plane to spread the coverage over the entire world (Figure 2.1).

The navigation transmissions are 1500-bit messages containing satellite ephemerides and health status information. These data are then biphase modulated onto two kinds of pseudonoise codes: One, called the C/A code, is easily decodeable; the other, known as the P-code, is virtually impossible to extract without prior knowledge of its true form and is reserved for military
users. The coded information is then ultimately modulated onto a carrier. The carrier, in turn, is made available at two different frequencies: 1575.42 (L₁) and 1227.6 (L₂) MHz. The navigation solution is based on measurements of the range from the user to the satellites. At least four such satellite measurements are needed to solve for the user's position in three dimensions plus the time difference (bias) between the receiver and the system clocks. It is the precision of the atomic clocks on board the satellites that makes possible generation of the highly stable signals which constitute the key to the high accuracies associated with GPS navigation.

The control segment of GPS is responsible for monitoring the exact whereabouts of the satellites. This makes it possible for the system, i.e., the satellites themselves, to ensure that only reliable ephemeris information is disseminated to the user on the ground [5]. A network of ground tracking stations are engaged to do this by solving the navigation problem in reverse. The new information obtained is then used to update the data banks of all the GPS satellites periodically.

Although originally intended for military use, GPS will also accessible to the civil community at a lower level of performance. As a result, the user segment in the GPS triad will actually comprise of anyone equipped to "tap" into the GPS satellite network. There has been a tremendous amount of development done on so-called C/A-code receivers which will be relied almost entirely upon by civil users. Of late, the push has been in the direction towards making receivers that can be marketed as consumer products. Speculation about sub-$1600 receivers as well as receivers packaged as hand-held products are becoming common at navigation forums.
Still, it is unlikely that such products will materialize at least until the system is well in place.

There are two basic types of receivers. The multichannel receiver is capable of tracking all available satellites (usually just four or five) simultaneously because each channel in the receiver is dedicated to tracking just one satellite. The sequencing receiver, on the other hand, uses just one channel and, thus, has to "visit" the satellites one at a time in cyclical fashion. The cycling rate typically used can be from one to several seconds, depending on the design. A faster version of the sequential type is known as the multiplex receiver which trades off signal-to-noise ratio for increased rate of update. The cycle rate here is usually more than 50 Hz. With the eventual decline in hardware costs, the justifications for the multichannel receiver are soon becoming more and more dominant. An excellent tutorial comparing the different types of GPS receivers may be found in [6].

2. Performance and selective availability

In order to obtain the finest accuracy possible in the navigation mode, a user must have access to the P-code. With it, one has a higher frequency code modulation to time on and hence get better resolution in measuring the pseudorange. (The frequency of the binary P-code, or chipping rate, is 10.23 Megabits per second (Mbps), while that of the C/A code is only 1.023 Mbps.) Also, the P-code allows access to the L2 carrier frequency in addition to the regular L1 receivable by C/A code users. Using the measurements at the two different frequencies, ionospheric refraction errors can, to a large extent, be removed [7].
Absolute positioning accuracies of under ten meters have well been established in this mode. Knowledge of the P-code, however, will not be a privilege extended to the civil community in general. Thus, with only the C/A code to go on, the civil user can, at best, with just the \( L_1 \) frequency and lower chipping rate of the code to go on, expect accuracies nominally in tens of meters. Even so, this represents a significant improvement over older navigation systems. Also, the accuracies are ever increasing with better technology built into newer receivers.

Owing to the remarkable performance demonstrable on present-day C/A code receivers, the DoD has announced plans to degrade the accuracies available to the civil community by artificially introducing more errors into the system. This concept is known formally as selective availability [8]. However, very few technical details about its nature have been made known to the public so far, other than to say that the civil community will be allowed an accuracy of about 100 meters, 2\( \text{drms} \) [9].

3. Extra-satellite coverage and GDOP

As pointed out before, at least four satellites must be tracked in order to obtain a stable navigation solution. Ideally, there should be five or more satellites in the visible sky at all times. The extra satellites would not only provide redundancy in the event of a satellite outage, but the user could then also afford to pick out the combination of four satellites that would yield the best results.

The geometry of the locations of the four satellites used for navigating is a crucial consideration (Figure 2.2). The figure-of-merit generally quoted is
a. Good geometry (low GDOP)         b. Poor geometry (high GDOP)

Figure 2.2. A pictorial comparison of the relative degrees of "good" and "poor" satellite geometries

a measure called the geometric dilution of precision or GDOP [10]. Due to the constantly changing satellite geometries, there will exist periods of high GDOP, for any given four satellites, which yields a near singular condition in the algebraic navigation solution. These conditions must, if possible, be avoided by changing satellites combinations. With an 18-satellite constellation, there will be cases at certain geographical locations on the surface of the earth where only four satellites will be available over short periods of time, and thus occasional periods of high GDOP will exist.
B. Kalman Filtering

The concept of filtering in the context of signal theory is simply one of extracting the desired information that makes up the signal content from the unwanted noise that it has been immersed in. In the filtering problem involving random processes in general, the lines that demarcate what is called the "desired signal" and "unwanted noise" are less clear; the signal itself may be noise-like in nature. The ideas of filtering that grew out of the works of N. Wiener and, later, R. E. Kalman, address this topic of stochastic estimation with the objective of optimizing the error between the estimated and the actual observations or output. This type of problem has come to be known as linear minimum-mean-square-error filtering.

The remarkable achievement of Kalman, in making Wiener's original problem numerically tractable in the early 60's, gave rise to what is widely known today as the Kalman filter [11]. This is a tool that has been responsible for making possible, many advances in the fields of aerospace and navigation over the past two decades [12,13]. And yet, it is often a misunderstood subject that can sometimes make it be both overly revered and shunned at the same time. Many good reference texts have been written on this topic and may be consulted for tutorial purposes [14,15,16]; only a brief account of relevant issues will be discussed in the following subsections.

1. The discrete Kalman filter

The Kalman filter is a recursive algorithm that processes noisy measurements that are linearly related to the states of the system and yields
estimates of these states with their associated covariance structure. The Kalman filter model is based on state-space formulations which originated in modern control theory. The process dynamics can be written as a set of first-order differential equations:

\[ \dot{x} = Ax + Bu \]  

where

- \( x \) = the n-tuple state vector;
- \( A \) = n x n dynamical matrix;
- \( B \) = input matrix;
- \( u \) = vector of white-noise inputs.

Deterministic control inputs may also be added if desired but were explicitly left out here without any loss of generality. The discrete-time solution to this dynamical equation is simply:

\[ x_{k+1} = \Phi_k x_k + w_k \]  

where

- \( \Phi_k \) = state transition matrix that translates the states at \( t_k \) to \( t_{k+1} \) through the natural dynamics of the system;
- \( w_k \) = zero-mean vector sequence of Gaussian-distributed numbers uncorrelated in time with known covariance of \( Q_k \).

In stochastic process theory, the first-order difference nature of eq. 2.2 is also known as the vector Markov process. The Markov process is one whose current states functionally depend only on the states at the previous time
The measurement model is a linear algebraic relation of the form:

\[ \mathbf{z}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{v}_k \] (2.3)

where

- \( \mathbf{z}_k \) = measurement vector at \( t_k \);
- \( \mathbf{H}_k \) = linear connection measurement matrix, time varying in general;
- \( \mathbf{v}_k \) = zero-mean vector sequence of Gaussian-distributed numbers with known covariance \( \mathbf{R}_k \), uncorrelated in time and independent of \( \mathbf{w}_k \).

The Kalman filter is very much model dependent. The critical parameters that "describe" the Kalman filter model are \( \Phi, Q, H, \) and \( R \) from eqs. 2.2 and 2.3. The filter is predicated on a probabilistic structure that must be a fairly reasonable representation of the true stochastic dynamics of the process. If not, the possibility of divergence might exist \([11]\). It is quite robust though with respect to some variations in its parameters, in particular, the covariance terms.

After obtaining the model, i.e. its parameters, the implementation of the filter is virtually routine. The filter algorithm is given in eqs. 2.4-2.8:

\[
\begin{align*}
\mathbf{K}_k &= \mathbf{P}_k^{-1} \mathbf{H}_k^T [ \mathbf{H}_k \mathbf{P}_k^{-1} \mathbf{H}_k^T + \mathbf{R}_k]^{-1} \\
\hat{\mathbf{x}}_k^+ &= \hat{\mathbf{x}}_k^- + \mathbf{K}_k [ \mathbf{z}_k - \mathbf{H}_k \hat{\mathbf{x}}_k^- ] \\
\mathbf{P}_k^+ &= [1 - \mathbf{K}_k \mathbf{H}_k] \mathbf{P}_k^- \\
\hat{\mathbf{x}}_{k+1}^- &= \Phi \mathbf{K}_k \hat{\mathbf{x}}_k^- \\
\mathbf{P}_{k+1}^- &= \Phi \mathbf{P}_k^- \Phi^T + \mathbf{Q}_k
\end{align*}
\] (2.4-2.8)
Figure 2.3. A flow diagram for the Kalman filter algorithm
The algorithm is also shown as a flow chart in Figure 2.3 depicting the dichotomy of the covariance computations and those of the state estimation, the two parts linked only by the gain $K$. Initial conditions of the filter need to be established for the state vector estimate $\hat{x}_0$ and its associated covariance $P_0$. These two quantities represent the statistical mean and variability of the initial distribution of the state vector estimate, itself a random variable. If the statistics involved are Gaussian, these two parameters are sufficient to completely describe the initial or prior assumed distribution.

For notational purposes, the circumflex ($\hat{\cdot}$) is used to denote an estimated quantity (implying that it is a random variable), while the tilde ($\tilde{\cdot}$) represents a residual quantity: true minus estimated. The superscripted minus sign denotes a quantity associated with a prior distribution while the superscripted plus sign denotes one associated with a posterior distribution. The notion of prior and posterior distributions is one that arises from Bayesian statistics [18], and the Kalman filter is very much a Bayesian filter once Gaussian statistics are assumed.

In the filter algorithm, upon each recursion, the *a priori* state vector estimate undergoes an update which is based on new information extracted from the current measurements. The measurement residual $(z_k - \hat{x}_k)$ is weighted by the Kalman gain, $K_k$, and used to adjust the prior state estimate to yield the *a posteriori* state estimate. At the end of the recursion, this updated state estimate is projected ahead, through known system dynamics, to the next time step where it will begin the next recursion as the *a priori* estimate once again. The sequence of measurement residuals $\hat{x}_{0:K}$ is sometimes known as the innovations sequence, and the information it
contains greatly influences the outcome of the state estimates. In an optimal filter, the sequence of residuals is unbiased and uncorrelated. It will be seen later on, too, that some failure detection schemes make their decisions based on the statistics of the innovations sequence [1].

2. The GPS measurement model and the extended Kalman filter

The geometric relationships in the GPS measurement model which link the pseudoranges \( \psi \) from the various satellites to the user's location \( X \) in some Cartesian coordinate frame are nonlinear in nature due to the trigonometric functions involved. The model can be linearized by Taylor series approximation in the following way [14]:

\[
\psi - \mathcal{H}(X) = \mathcal{H}(X^*) + \left[ \frac{\partial \mathcal{H}}{\partial X}(X^*) \right] (X - X^*) \Delta X
\]

To begin with, some nominal value \( X^* \) is used as a close approximation of \( X \). The "measurement" in this linearized form of eq. 2.10 is now the residual term, \( \psi - \mathcal{H}(X^*) \), that results as a difference between the true and predicted pseudorange measurements. Here also, the state vector is not \( X \) but \( \Delta X \), an incremental element referenced to \( X^* \). The geometric relationship of this incremental model can be seen in Figure 2.4. It is clear from this picture that the linear connection matrix \( \left[ \frac{\partial \mathcal{H}}{\partial X}(X - X^*) \right] \) is comprised of rows of unit directional vectors pointing from each satellite to the user's position.
With the measurement model in this form, either the linearized or the extended versions of the Kalman filter may be used. In the former, a nominal state trajectory assumed ahead of time is used for $X^*$; while in the extended variety, $X^*$ is based on the most current state estimate made by the filter \cite{14,19}. In this research project, the extended Kalman filter was used.

The geodetic coordinate frame of reference used as the GPS standard is what is called the World Geodetic System of 1972 or, more commonly, as WGS-72. This reference system is earth-fixed and earth-centered. For our purposes of depicting a simulated picture of aircraft flight over a localized region on the surface of the earth, a locally-level frame of reference is much more suitable to deal with. Here, we would have three orthogonal axes respectively pointing east, north, and vertically up from some local reference.
Figure 2.5. The locally-level x-y-z coordinate frame of reference

point (Figure 2.5). The linear transformation between WGS-72 and this locally-level frame is quite straightforward and will not be given here.

3. Random process models

The process model developed for the Kalman filter incorporates the dynamical behavior of the state variables involved. In GPS navigation, the problem of absolute positioning requires solving for the basic state variables, namely, the three components of position. When the receiver is in motion, however, additional state variables are usually needed. Assuming autonomous receiver operation, as is normally the case, the dynamics of the receiver clock errors must also be modeled. Since the topic is closely related to aircraft navigation, process models relevant only to flight dynamics will be considered. The clock error model will also be discussed.

The dynamical behavior of an aircraft varies drastically in different phases of a flight. For instance, when it is cruising at high altitudes for the
much of the journey, the aircraft maintains a nearly straightline path trajectory and encounters little acceleration. On the other hand, from the time it makes its initial descent through the final approach until the landing is completed, the aircraft undergoes a series of maneuvers which can give rise to sizeable accelerations.

Figure 2.6. Transfer function block diagram for each position error in x-y-z

A transfer function model such as that shown in Figure 2.6 for one of three dimensions in free space, represents a second-order differential equation (eq. 2.1), thus accounting for the position and velocity states only.

\[
\begin{bmatrix}
\dot{x}_1 \\
\dot{x}_2
\end{bmatrix} =
\begin{bmatrix}
0 & 1 \\
0 & 0
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2
\end{bmatrix} +
\begin{bmatrix}
0 \\
1
\end{bmatrix}u
\]

(2.11)

\[y = \begin{bmatrix}
1 & 0
\end{bmatrix}\begin{bmatrix}
x_1 \\
x_2
\end{bmatrix}\]

(2.12)

where

- \(x_1\) - position in one dimension;
- \(x_2\) - velocity in the same dimension;
- \(u\) - white noise.
In a low-dynamics environment such as the cruising or enroute flight situation, this model is an appropriate choice for the stochastic process involved. Although higher random dynamics may be encountered on occasions of upper atmospheric turbulence, the numbers chosen here reflect the more typical conditions.

In the landing approach situation, the presence of modest accelerations can be tolerated to a certain extent by increasing the magnitude of the white noise input. However, the efficiency and appropriateness of the model eventually degenerates when the size of the white noise gets too large. A more effective solution is to include an acceleration state, thereby raising the complexity of the dynamical process to third order for each dimension. The transfer function model depicted in Figure 2.7 shows such a third-order

![Diagram](https://example.com/diagram.png)

Figure 2.7. Position error model for high dynamics

low-dynamics model with a scalar Markov acceleration state added to the input. The time constant of the Markov component $1/\beta$, was chosen to be 10 seconds and its variance $\sigma^2$, to be $(.2g)^2$, $g$ being the gravitational acceleration constant.
The stochastic nature of precision clock standards have been studied extensively over the years [20]. One fairly well-accepted model uses two state variables to represent the clock bias and the clock drift rate, the latter being the derivative of the former. The transfer function model is then quite similar to the low-dynamics model presented earlier. The only difference, as seen in Figure 2.8, is the presence of another white noise input into the second integrator. The output of the transfer function represents the clock bias, which combines an integrated random walk process with a regular random walk. The only parameters specifiable here pertain to the spectral amplitudes of the noise inputs. For a high-quality crystal clock, the spectral amplitudes of $w_1$ and $w_2$ are $2.70 \times 10^{-23}$ and $9.0 \times 10^{-22}$. The parameters for a lower-quality clock might be $7.5 \times 10^{-20}$ and $4.715 \times 10^{-20}$ [20].

4. Discrete-time dynamical process

Consolidating the various one-dimensional dynamical models presented in the previous subsection, we would have either an 8-state or an 11-state
in the previous subsection, we would have either an 8-state or an 11-state process depending on whether the acceleration states for the high-dynamical model were included or not. Irrespective of that, for conciseness of documentation, we shall be concerned with showing only with the 11-state model, of which the 8-state version is a subset.

The state variables $x_1, x_4, x_7$ are assigned to be position components, $x_2, x_5, x_8$ as velocity components, and $x_3, x_6, x_9$ as acceleration components. The clock bias and drift components are represented by states $x_{10}$ and $x_{11}$, respectively.

The process model, in partitioned form, is then:

\[
\begin{bmatrix}
    x_1 \\
    x_2 \\
    x_3 \\
    x_4 \\
    x_5 \\
    x_6 \\
    x_7 \\
    x_8 \\
    x_9 \\
    x_{10} \\
    x_{11}\end{bmatrix} = \Phi \begin{bmatrix}
    x_1 \\
    x_2 \\
    x_3 \\
    x_4 \\
    x_5 \\
    x_6 \\
    x_7 \\
    x_8 \\
    x_9 \\
    x_{10} \\
    x_{11}\end{bmatrix} + \begin{bmatrix}
    w_1 \\
    w_2 \\
    w_3 \\
    w_4 \\
    w_5 \\
    w_6 \\
    w_7 \\
    w_8 \\
    w_9 \\
    w_{10} \\
    w_{11}\end{bmatrix}
\]

(2.13)
where (with $\Delta t$ as the sampling interval)

$$
\Phi_x = \begin{bmatrix}
1 & \Delta t & \beta^2(\Delta t/\beta - 1 + e^{-\Delta t/\beta}) \\
0 & 1 & \beta(1 - e^{-\Delta t/\beta}) \\
0 & 0 & e^{-\Delta t/\beta}
\end{bmatrix}
$$

and

$$
\Phi_c = \begin{bmatrix}
1 & \Delta t \\
0 & 1
\end{bmatrix}
$$

The covariance structure of the random sequence $w = [w_1, w_2, \ldots, w_1]^T$ is given by:

$$
Q = E[ww^T] = \begin{bmatrix}
q_1 & 0 & 0 & 0 \\
0 & q_2 & 0 & 0 \\
0 & 0 & q_3 & 0 \\
0 & 0 & 0 & q_4
\end{bmatrix}
$$

where

$$
q_4 = \begin{bmatrix}
S_0 \Delta t + S_3 \Delta t^3 & S_0 \Delta t^2 \\
S_3 \Delta t^2 & S_0 \Delta t
\end{bmatrix}
$$
\[ \mathbf{q}_1 - \mathbf{q}_2 - \mathbf{q}_3 = \begin{bmatrix} q_{11} & q_{12} & q_{13} \\ q_{12} & q_{22} & q_{23} \\ q_{13} & q_{23} & q_{33} \end{bmatrix} \]

and

\[
q_{11} = S_1 \frac{\Delta t^3}{3} + S_2 \left[ \frac{\Delta t^3}{3 \beta^2} - \frac{\Delta t^2}{\beta^3} + \frac{2 \Delta t}{\beta^4} - \frac{1}{\beta^5} + \frac{\Delta t e^{-\beta \Delta t}}{\beta^3} + \frac{4 e^{-\beta \Delta t}}{\beta^5} \right]
\]

\[
q_{12} = S_1 \frac{\Delta t^2}{2} + S_2 \left[ \frac{\Delta t^2}{2 \beta^2} - \frac{2 \Delta t}{\beta^3} + \frac{\Delta t e^{-\beta \Delta t}}{\beta^4} - \frac{3 e^{-\beta \Delta t}}{\beta^5} \right]
\]

\[
q_{13} = S_2 \left[ \frac{\Delta t}{\beta^2} - \frac{2}{\beta^3} + \frac{\Delta t e^{-\beta \Delta t}}{\beta^4} + \frac{2 e^{-\beta \Delta t}}{\beta^5} \right]
\]

\[
q_{22} = S_1 \Delta t + S_2 \left[ \frac{\Delta t}{\beta^2} - \frac{2}{\beta^3} + \frac{\Delta t e^{-\beta \Delta t}}{\beta^4} + \frac{2 e^{-\beta \Delta t}}{\beta^5} \right]
\]

\[
q_{23} = S_2 \left[ \frac{1}{\beta^2} - \frac{\Delta t e^{-\beta \Delta t}}{\beta} - \frac{e^{-\beta \Delta t}}{\beta^2} \right]
\]

\[
q_{33} = S_2 \Delta t e^{-\beta \Delta t}
\]

\[ S_1 \text{ and } S_2 \text{ are the spectral amplitudes of the white noise inputs 1 and 2 for this model (from Figure 2.7), and } 1/\beta \text{ is the time constant for the Markov acceleration state. The measurement model is:} \]
where \( h_{x1}, h_{y1}, h_{z1} \) are components of the unit directional vector pointed at the \( i^{th} \) satellite and \( c \) is the speed of light constant. The large matrices in eqs. 2.13 and 2.14 are quite sparse and, as a result, some computational savings can be achieved by using this property.

This model was tested in a Monte Carlo simulation and the estimator error (difference between the state estimate and the true state) is shown for the 3 position components in Figure 2.9. The low-dynamics model described was used. The specifics of the dynamical simulation will be discussed later in Chapter VI. The dashed lines in the three plots of Figure 2.9 show the 2\( \sigma \) confidence intervals within which the estimates remain over 95 percent of the time. Due to the fact that the GPS satellites are located, for the most part, in the region of sky overhead from an observer's vantage point, it is easy to
Figure 2.9a. The x-position state estimate error (2σ-interval-of-variation shown in dotted lines) for a low-dynamics model in an enroute situation.
Figure 2.9b. The y-position state estimate error (20-interval-of-variation shown in dotted lines) for a low-dynamics model in an enroute situation.
Figure 2.9c. The z-position state estimate error (2σ-interval-of-variation shown in dotted lines) for a low-dynamics model in an enroute situation.
see that the error in estimating the vertical position component would expectedly be higher than for the other two components lying in the plane of the ground (refer to Figure 2.5). This is verified in the results of Figure 2.9 where the 2σ bound in the z-position error is noticeably larger than in either the x- or the y-component.

C. Magill Adaptive Scheme

The efficient operation of a Kalman filter is strongly dependent on the accuracy of its model and the associated process parameters. When dealing with physical systems, exact knowledge of these parameters is often unavailable. One reason for this may be that the system is itself nonstationary over long periods of time. In such situations, although a Kalman filter may have modeled the system accurately over shorter operational time spans, the repeatability of its performance over a longer course of time may degrade as the system gradually changes. What then is needed is a filter that is self-learning or adaptive in nature. There have been many studies on adaptive schemes based on the Kalman filter [1, 2], but the one we will be concerned with is that due to D. T. Magill [21].

1. Parallel Filters

The Magill scheme is comprised of the implementation of a bank of parallel Kalman filters, each modeled around a particular realization of the uncertain parameter in question. For apparent reasons, this is sometimes known in control systems literature as the Multiple Model Estimation
Algorithm (MMEA). In our discussion throughout, the concepts of the uncertain parameter and the uncertain model are used interchangeably, commonly denoted by $\alpha$. Although, in some instances, it is a particular parameter that needs to be identified, in general, however, different models may be accommodated in the Magill scheme.

The filters in this parallel bank, however many there are, all operate on the same measurement sequence $z_k^n$ (see Figure 2.10). Each Kalman filter then produces some state estimate that is based on a certain model that the filter is associated with. At the same time, the conditional *a posteriori* (or posterior) probability for each filter is computed to provide some indication as to the closeness of each of the models to the true process. The probability associated with the "correct" model should statistically converge to unity in the limit as the number of measurements processed increase. When this

![Figure 2.10. The Magill adaptive filter](image-url)
happens, the posterior probabilities associated with all other models go to zero, because the sum of all these probabilities should total up to unity.

Presumably, the filter with the model that exactly represents the true process is operating optimally. This fact is then properly accounted for in its posterior probability. However, since it is necessary that the filters implemented need be discrete and finite in number, the restrictions imposed on the practicality of this scheme are obvious. The number of filters implemented is the crucial issue here and this forces a tradeoff in the "search" for that uncertain parameter between the allowable range and the resolution of discretization. Hence, one can at best hope to pick out a model that is only a close approximation of the true process. In the adaptive problem, the model ultimately selected may be less relevant than the optimal estimate obtained while the parameter identification or model resolution is still in progress. The Magill adaptive filter uses the conditional probabilities as weights to blend the output of each Kalman filter together into an optimal estimate.

2. Likelihood functions and posterior probabilities

The derivation of Magill's formulations is well documented elsewhere [14, 22] and will not be repeated here. We will focus our discussion instead on the computation of the posterior probabilities. The \textit{a posteriori} probability \( p(\alpha|Z) \), is a conditional probability that the uncertainty \( \alpha \) is the true parameter or model, given the observed measurement sequence \( Z \). This quantity can be obtained through a term with the conditioning reversed, i.e., the \textit{a priori} (or prior) conditional probability \( p(Z|\alpha) \).
Assuming that the measurements are all Gaussian-distributed, the joint
distribution of the measurement sequence is then

\[
f(Z_n|\alpha) = (2\pi)^{-n/2} |C_n(\alpha)|^{-1/2} \exp\left(-\frac{1}{2}(Z_n - \mu_n)^T C_n(\alpha)^{-1} (Z_n - \mu_n)\right)
\]

(2.15)

where

\[
C_n(\alpha) = E [(Z_n - \mu_n)^T (Z_n - \mu_n)]
\]
\[
\mu_n = E [Z_n|\alpha]
\]
\[
Z_n = [z_0, z_1, ..., z_n]^T; \quad (z_i \text{ is the measurement vector at time step } t_i).
\]

Here, the covariance matrix \( C_n \) is a function of the unknown parameter \( \alpha \). In
general, we can also have \( \mu_n \), or both \( C_n \) and \( \mu_n \), being functionally
dependent on \( \alpha \). The joint density \( f(Z_n|\alpha) \) can also be regarded as a likelihood
function \( L(\cdot) \) (eq. 2.16) though with a subtle difference in interpretation.

\[
L(\alpha) = f(Z_n|\alpha)
\]

(2.16)

The following graphical depictions should illustrate the point. Figure
2.11 shows the usual interpretation of eq. 2.15 in one dimension. Given a
fixed model, the graphical plot depicts the relative probabilities of obtaining
observations \( z_a \) and \( z_b \). On the other hand, say we are considering two
models, \( \alpha_1 \) and \( \alpha_2 \), and have available, an observation \( z_a \). In this case, the
Figure 2.11. The probability density function

The likelihood function provides information about the relative probabilities that such an observation resulted from one or the other model (see Figure 2.12).

In the Magill formulation, the likelihood function is adapted for the Kalman filter setting by considering the joint density of the set of measurements:
where the measurement residual $\mathbf{z}_n = (\mathbf{z}_n - \hat{\mathbf{z}}_n)$ is zero-mean Gaussian. Note that $\hat{\mathbf{z}}_n$ are the optimal estimates output by the Kalman filter and $E[\hat{\mathbf{z}}_n] = \mathbf{z}_n$.

Assuming optimum conditions where the measurement residuals are uncorrelated, the covariance matrix $\mathbf{C}_n$ is a block diagonal matrix with the entries $\mathbf{V}_i = (\mathbf{H}_i \mathbf{P}_i \mathbf{H}_i^T + \mathbf{R}_i)$ for $i=0,...,n$ obtained from the Kalman filter (see eq. 2.4). Given this fact, eq. 2.17 can be written as

$$f(\mathbf{z}_n | \alpha) = (2\pi)^{-n/2} |\mathbf{C}_n(\alpha)|^{-1/2} \exp(-\frac{1}{2} \mathbf{z}_n^T \mathbf{C}_n(\alpha)^{-1} \mathbf{z}_n)$$

or equivalently, in a recursive form,

$$f(\mathbf{z}_n | \alpha) = [(2\pi)^{-1/2} |\mathbf{V}_0|^{-1/2} \exp(-\frac{1}{2} \mathbf{z}_0^T \mathbf{V}_0^{-1} \mathbf{z}_0)] \times
[(2\pi)^{-1/2} |\mathbf{V}_1|^{-1/2} \exp(-\frac{1}{2} \mathbf{z}_1^T \mathbf{V}_1^{-1} \mathbf{z}_1)] \times
\cdots \times
[(2\pi)^{-1/2} |\mathbf{V}_n|^{-1/2} \exp(-\frac{1}{2} \mathbf{z}_n^T \mathbf{V}_n^{-1} \mathbf{z}_n)]$$

or equivalently, in a recursive form,

$$f(\mathbf{z}_n | \alpha) = [(2\pi)^{-1/2} |\mathbf{V}_n|^{-1/2} \exp(-\frac{1}{2} \mathbf{z}_n^T \mathbf{V}_n^{-1} \mathbf{z}_n)] \times f(\mathbf{z}_{n-1} | \alpha)$$

with the uniform initial condition $f(\mathbf{z}_0 | \alpha) = 1$, unless some information about its prior distribution is available.

Now, the posterior probability $p(\alpha | \mathbf{z}_n)$ can be obtained via the Bayes relationship

$$p(\alpha | \mathbf{z}_n) = f(\mathbf{z}_n | \alpha) \cdot f(\alpha) / p(\mathbf{z}_n)$$

or

$$p(\alpha | \mathbf{z}_n) = f(\mathbf{z}_n | \alpha) \cdot f(\alpha) / (\int \mathbf{f}(\mathbf{z}_n | \alpha) \cdot f(\alpha) \, d\alpha)$$

(2.20)
or, if the $\alpha$-space is discrete,

$$p(\alpha|Z_n) = f(Z_n|\alpha) \cdot f(\alpha)/\{\sum_{\alpha} f(Z_n|\alpha) \cdot f(\alpha)\}$$

(2.21)

A simplification to eq. 2.21 (and eq. 2.20) can be made by letting the unconditional distribution of $\alpha$ be uniform. Hence, with unity $f(\alpha)$, eq. 2.21 becomes

$$p(\alpha|Z_n) = f(Z_n|\alpha)/\{\sum_{\alpha} f(Z_n|\alpha)\}$$

(2.22)

In eqs. 2.18 and 2.19, the functional dependence on $\alpha$ was never explicitly shown to notationally leave open the issue of what aspect of the model $\alpha$ actually represents. In his original paper, Magill was interested in the identification of any one of the Kalman filter parameters. This then meant that the functional dependence on $\alpha$ would "proliferate" into the covariance structure and hence, $V_j$. Conversely, in problems where the unknown parameter appears linearly only in the measurement equation, the covariance structure is independent of $\alpha$ and a significant simplification in the implementation is available [23,24]. The failure detection problem falls into this category, and so this matter will be explored further on in Chapter IV.

3. Multiple hypothesis testing

In identifying the model that suits an unknown process best, the Magill adaptive scheme is actually playing the role of a multiple hypothesis tester [24]. In the context of Figure 2.12, the different filters implemented are, in this case, a variety of models with each hypothesized as the true process.
The failure detection and identification problem is primarily concerned with deciding if a failure has indeed occurred, and when it does, the type of failure model that correctly represents it. The *a posteriori* probabilities not only provide the means to make a decision after a predetermined amount of data is processed, they also indicate the level of confidence one can place in the decision itself. For example, when a decision is made to choose the model $\alpha_1$ say, the level of confidence that this is the "correct" model is much higher if the associated posterior probability $p(\alpha=\alpha_1|Z)$ is .99999 than if it were .80.

In sequential processing of real-time data, it is often important to venture upon a decision as soon as it is feasible to make one. If one of the $p(\alpha|Z)$'s surpasses some preset threshold even before all the available data have been processed, this may also be a suitable juncture to terminate the problem and generate a decision appropriately.

The numerical implementation of eqs. 2.19 and 2.22 is quite straightforward. When the measurement residuals of likelihood functions that belong to nonoptimal models accumulate to sizeable values though, the exponentiation operation will yield extremely small numbers. This may cause problems depending on the underflow threshold dictated by the computer used [25]. A way to protect eq. 2.19 from possible distortion of the information it carries in recursion is to look at the log-likelihood version instead:

$$\log f(Z_n|\alpha) = -\frac{1}{2} \log |(2\pi)^{|V_n|}| - \frac{1}{2} \mathbf{g}_n^T \mathbf{V}_n^{-1} \mathbf{g}_n + \log f(Z_{n-1}|\alpha)$$

(2.23)

Eq. 2.22 can be rewritten with an arbitrary scaling factor $A_n$ to enable the
likelihood function from eq. 2.23 to comfortably fit the allowable dynamic range available:

\[ p(\alpha|Z_n) = \frac{A_n f(Z_n|\alpha)}{\sum_\alpha A_n f(Z_n|\alpha)} \]  

(2.24)

The adaptation of the Magill scheme for multiple hypothesis testing has been successfully demonstrated before in other settings [23], and will be pursued here in its application to the failure detection problem.

D. Brief Survey of Failure Detection Techniques

In virtually any complex system, there is always a need to ensure that if the system fails while it is in operation, this information is accessible to all who rely on it. The degree of importance of the role of failure detection, however, may vary from system to system which surely influences the sophistication afforded in its design. There have been many studies done in applied systems settings where there is critical need for failure detection. These have ranged from flight control systems on high-performance aircrafts [26] to nuclear power plants [27]. A comprehensive survey by A. S. Willsky in 1976 on failure detections techniques provides an excellent reference for an overview on the subject [1]. Our discussions will be segmented into various categories of failure detection techniques whose boundaries are, by no means, hard and fast. This classification is based on the Willsky survey paper.
1. Voting systems

Voting schemes are often found in systems where there is a high level of parallel hardware redundancy. In a typical conceptual voting scheme there are at least three identical measurement instruments. The logic to detect and isolate an errant instrument is quite simple: if two or more instruments agree with each other but disagree with one in particular, the latter is immediately identified as bad.

The idea here is very similar to that described earlier in Chapter I using lines of position in a navigation problem. Voting schemes are relatively easy to implement and are particularly good at detecting hard or catastrophic failures. However, they require a good deal of hardware redundancy and do not perform as well, in the presence of noise, when dealing with subtle failures.

2. Failure-sensitive filters

Optimal filters are generally designed to perform well under conditions where there are no modeling errors. When unmodeled aspects of the process become substantial, the possibility of divergence increases. This is because the filter becomes "too confident" in its estimates of the state that it becomes too oblivious to information in the new measurements. Any changes to the optimal model is reacted to in a sluggish manner by this type of filter. To counteract this, there are some techniques that "tinker" with the filter computationally to make it respond to changes faster than it normally would. These usually involve manipulating the noise covariances or the gains of the filter. The use of these types of filters in failure detection may then be an
indirect one; the failure detection decision is based on sudden or uncharacteristic changes in the state estimates.

A more direct approach to failure detection using more sensitive filters is to actually model the failure states and include them in with the system states. One notable technique which uses this approach is due to Kerr [28] where the estimates of the augmented states are compared with nominal values for a no-failure situation. The decision rule in this scheme is based on a procedure known as the two-confidence interval overlap which attempts to determine if the deviation of those estimates from the nominal are statistically significant.

The drawback frequently attributed to these types of filters is the compromise in their performance during normal no-failure conditions that must be contended with. This is a direct result of incorporating the additional failure detection measures into the intrinsic filtering system.

3. Multiple hypothesis

The multiple hypothesis category basically consists of schemes that implement various hypothesized failure models, literally, in a parallel bank of filters. This very same concept is sometimes used in adaptive filtering problems. Decision rules in such schemes are often based on computed posterior probabilities associated with each hypothesis conditioned on the measurement information. An example included in this category by Willsky is the binary hypotheses test called the Sequential Probability Ratio Test (SPRT), an idea that was originated by Wald [29] in experimental statistics. The decision rule dictated by this test provides two thresholds for the test
statistic which, in this case, is the log-likelihood ratio. One threshold bounds the region which corresponds to the acceptance of one of the hypotheses while the second bounds the acceptance region for the other. The middle ground between the thresholds is the region of inconclusion which prolongs the test to at least the next step when the procedure is repeated. This is a powerful test and lends well to analysis. It was adapted to the context of failure detection by Newbold and Ho [30].

4. Innovations-based detection systems

The innovations sequence is made up of measurement residuals, the difference between the actual measurements and what was expected to be measured. The monitoring of this sequence and the analysis of the underlying statistics associated with it form the basis of many detection schemes [1].

The chi-squared detection scheme is predicated on, as the name suggests, chi-squared statistics. When the Gaussian-distributed residual is squared and divided by its variance, the resulting random variable has a chi-squared distribution. The variances of the residuals are computable through the assumption of a probabilistic model for the stochastic process. As these terms are summed (equivalent to being averaged), the statistic formed also has a chi-squared distribution, with the number of degrees of freedom being related to the terms in the summation. A sliding window is generally used in the sum. This statistic is then sequentially analyzed to determine if its sample mean is zero within a certain confidence level. If not so, then a failure is deemed to have occurred and the alarm is raised. It is, however,
not capable of isolating the failure although there are ways it can be adapted to do so.

The simplicity of the chi-squared test, of course, has its drawbacks in performance. This led, in part, to the motivation for the development of the Generalized Likelihood Ratio (GLR), a newer technique which has been adopted widely in many applications today [31,32]. The GLR attempts to isolate different failures by using the knowledge of the different effects such failures have on the innovations. The GLR, as a fundamental statistical concept, tests the hypothesis of no-failure (say \( \theta = \theta_0 \)) against failure (\( \theta \neq \theta_0 \)) with the likelihood ratio statistic [33]. Since its first adaptation to failure detection problems before the mid '70s, many computational variants have been devised to cope with some of its original procedural shortcomings.

The Magill-based detection scheme that will be discussed later in Chapter IV can be placed in either the multiple hypothesis or the innovations-based category.
III. STATISTICS OF FAILURE DETECTION

In the Kalman filter algorithm as given in eqs. 2.4-2.8, the state estimate $\hat{x}_k$, a vector in general at the time $t_k$, is a random quantity that has a mean (expectation) of $x_k$, the state at $t_k$, and a corresponding covariance of $P_k$. If the stochastic processes involved are Gaussian, an assumption we will usually adopt, and if the initial distribution of the state estimate $\hat{x}_0$ is Gaussian, all subsequent estimates $\hat{x}_k$, $k>0$ will retain the Gaussian property. Note here also that the probability distribution of $\hat{x}_k$ is completely specified by $x_k$ and $P_k$.

A. Effects of an Additive Deterministic Failure on the State Estimate

On the introduction of a "deterministic" failure term $c_k$, that is additive to the actual measurement,

$$z_k = H_k x_k + c_k + v_k$$  \hspace{1cm} (3.1)

the Kalman filter property of preserving Gaussian statistics in its recursive structure is also retained. This lends a benefit to the analysis of the effects resulting from the failure. The term "deterministic" does not suggest that the failure has to be known, just that it is one that is not related to any of the state variables belonging to the random process model of the Kalman filter.
Under optimal conditions with no failure present, if the a priori estimate has a Gaussian distribution

\[ \mathbf{x}_k^- \sim \mathcal{N}(\mathbf{x}_k, \mathbf{P}_k^-) \] (3.2)

then,

\[ \mathbf{x}_k^+ \sim \mathcal{N}(\mathbf{x}_k, \mathbf{P}_k^+) \] (3.3)

and

\[ \mathbf{x}_{k+1}^- \sim \mathcal{N}(\mathbf{x}_{k+1}, \mathbf{P}_{k+1}^-) \]

However, if there is a failure term present in the measurement as in eq. 3.1, then if we start with eq. 3.2, the state estimate update becomes

\[ \mathbf{x}_k^+ \sim \mathcal{N}(\mathbf{x}_k + \mathbf{K}_k \mathbf{e}_k, \mathbf{P}_k^+) \] (3.4)

Through the transition matrix \( \phi_k \), \( \mathbf{x}_k^+ \) is projected ahead to the next time step to give

\[ \mathbf{x}_{k+1}^- \sim \mathcal{N}(\mathbf{x}_{k+1} + \phi_k \mathbf{K}_k \mathbf{e}_k, \mathbf{P}_{k+1}^-) \] (3.5)

In general,

\[ \mathbf{x}_k^+ \sim \mathcal{N}(\mathbf{x}_k + (1 - K_k H_k) \mathbf{x}_k^- + \epsilon_k \mathbf{K}_k \mathbf{e}_k, \mathbf{P}_k^+) \] (3.6)
where $\xi_k$ is a vector of the same dimensions as the measurement model with at most one nontrivial unit entry corresponding to a single failure; the other entries are zeros. The initial condition for the recursive eq. 3.7 is $\lambda_0^- = 0$. If we let

$$\lambda_k^+ = [1 - K_k H_k] \lambda_k^- + \xi_k K_k e_k$$  \hspace{1cm} (3.7)

we then have

$$\hat{x}_k^+ \sim N \left( x_k + \lambda_k^+, P_k^+ \right)$$  \hspace{1cm} (3.8)

and,

$$\hat{x}_{k+1}^- \sim N ( x_{k+1}^- + \lambda_{k+1}^-, P_{k+1}^- )$$  \hspace{1cm} (3.9)

where

$$\lambda_{k+1}^- = \phi_k \lambda_k^+$$  \hspace{1cm} (3.10)

The eqs. 3.7 and 3.10 provide the recursion to compute the expectation of the state estimate, whose posterior distribution after processing the measurement at time step $t_k$ is given by eq. 3.8 under the Gaussian assumption. In general, eq. 3.8 gives the parameters which correspond to the first (mean) and second (covariance) moments of the underlying distribution. Note though that within the realm of Gaussian statistics, the covariance structure of the state estimate (eqs. 3.8 and 3.9) remains the same as for the optimal case when the failure was absent (see eq. 3.3).
Figure 3.1a. The x-position state estimate error in the presence of a 3 m/s ramp (2σ-interval-of-variation shown in dotted lines) for a low-dynamics model in an enroute situation.
Figure 3.1b. The y-position state estimate error in the presence of a 3 m/s ramp (2σ-interval-of-variation shown in dotted lines) for a low-dynamics model in an enroute situation.
Figure 3.1c. The z-position state estimate error in the presence of a 3m/s ramp (2σ-interval-of-variation shown in dotted lines) for a low-dynamics model in an enroute situation
Figure 3.2. The magnitude of the position error due strictly to a 3 m/s ramp for a low-dynamics model in an enroute situation.
Figure 3.3. The magnitude of the position error due strictly to a 3 m/s ramp for a high-dynamics model in a nonprecision approach situation.
The above formulations were made for the purpose of analyzing how a simulated failure affects the estimator. To see this, let us revert to the simulation which involved an aircraft navigating in a cruise environment with GPS (refer to Figure 2.9 in Section B.4 of Chapter II). The plots of Figure 3.1 are of position estimate errors obtained under the same conditions that yielded the results of Figure 2.9, except that here, a 3 m/s ramp failure is present in one of the measurements processed by the steady-state Kalman filter. This deterministic ramp starts out with an initial value of zero at the start of our processing time interval. As before, the dashed lines represent the 2σ bound of the estimates and show a marked departure from the horizontal zero axis as the failure magnitude progressively grows. Due to the peculiarities of the measurement geometry involved in the simulation, it is apparent in Figure 3.1 that the failure affects the three position components differently.

Another interpretation of the error due solely to the failure is given in Figure 3.2 which is a plot of the magnitude of this error vector. This plot assumes zero random measurement error. Figure 3.3 is a plot of the same parameter shown in Figure 3.2, except that the simulation here is done in the nonprecision approach environment where a high-dynamics Kalman filter model was used instead.

B. Effects of a Failure on the Measurement Residuals

Since the measurement residuals play a key part in the determination of the occurrence of a failure, it is important to get an understanding of their
statistics in the presence of one. From eq. 3.9, we have

\[ \hat{x}_k^- \sim N(x_k + \lambda_k^-, P_k^-) \]  

(3.11)

Hence,

\[ (x_k - \hat{x}_k^-) \sim N(-\lambda_k^-, P_k^-) \]  

(3.12)

The measurement residual is given by

\[ \xi_k = z_k - \hat{x}_k^- - H_k (x_k - \hat{x}_k^-) + v_k + \eta_{ij} \]  

(3.13)

Under Gaussian assumptions,

\[ \xi_k \sim N(\epsilon_k - H_k \lambda_k^-, V_k) \]  

(3.14)

Recall that

\[ V_k = H_k P_k^- H_k^T + R_k \]

It had been previously mentioned that for an optimal Kalman filter, the sequence of measurement residuals produced is uncorrelated. Here, the same remains true under optimal conditions even with the presence of a deterministic failure in the measurements processed. In other words, with \( \xi_k \) as given in eq. 3.14, we would still have:
The full details of this result can be found in the Appendix.

C. The Chi-Squared Test on the Residuals

One simple statistical check that can be used on the measurement residuals is the chi-squared test. In the absence of any failure, the Gaussian-based distribution of the measurement residual (from eq. 3.14) is given by

$$\mathbf{z}_k \sim \mathcal{N}(0, \mathbf{V}_k)$$  \hspace{1cm} (3.16)

If we form the statistic

$$l_k = \sum_{j=k-N+1}^{k} \mathbf{z}_j^T \mathbf{V}_j^{-1} \mathbf{z}_j$$  \hspace{1cm} (3.17)

we then have a chi-squared random variable with \(N_p\) degrees of freedom [34], \(p\) being the dimensionality of the measurement vector. The decision threshold for the detection of a failure, determined with the aid of chi-squared statistical tables, is quite simply

$$l_k > \chi^2_{\alpha}(N_p) \Rightarrow \text{failure present}$$
$$l_k \leq \chi^2_{\alpha}(N_p) \Rightarrow \text{no failure present}$$  \hspace{1cm} (3.18)
for a detection probability rate of \( \alpha \). The parameter \( N \) represents the averaging window length. There have been detection schemes devised using this particular concept.

D. Maximum Likelihood Estimation of Parametrized Residuals

In the parametrized form, the measurement residual is a function of some parameter set. This function can be nonlinear in general, although, we will consider here for simplicity, a linear function of a single parameter \( u \). A scalar measurement residual at time \( t_k \) can then be written as:

\[
\mathbf{z} = [w_1 \ w_2] [u] \\
[1]
\]  

(3.19)

The statistic formed in eq. 3.17 is actually the log-likelihood function predicated on Gaussian statistics, but without a minus sign. This statistic, involving the parametrized measurement residuals (scalar in this case though not necessarily so in general) appropriately weighted by their computed variances, can be rewritten as a recursive equation:

\[
[u \ 1] [\lambda_{11} \lambda_{12}] [u] - [u \ 1] [\lambda_{11} \lambda_{12}] [u] + [u \ 1] [w_{1k} \ w_{2k}] [u] \\
[\lambda_{12} \lambda_{22}]_{k+1} [1] [\lambda_{12} \lambda_{22}]_k [1] [w_{2k}] [1]
\]  

(3.20)
Due to the minus sign, the likelihood function is maximized when the quantity on the left-hand side of eq. 3.20 is minimized with respect to the parameter $u$.

$$
\frac{d}{du}(\lambda_{11}u^2 + 2\lambda_{12}u + \lambda_{22}) = 2\lambda_{11}u + 2\lambda_{12}
$$

By setting eq. 3.21 equal to zero, we have $u = \lambda_{11}^{-1}\lambda_{12}$, the optimized value being the maximum likelihood estimate of $u$. Predicated on Gaussian statistics, $u$ consequently has a Gaussian distribution with a variance of $\lambda_{11}$. The mean of this distribution is equal to the actual value of the parameter $u$. 
IV. THE DETECTION AND IDENTIFICATION MODEL

The problem of failure detection and identification of a failed GPS satellite can be treated quite simply as a problem in hypothesis testing and statistical decision. Since it is not unreasonable to regard the occurrence of satellite failures as being rare, we can assume then that the probability of multiple failures occurring at the same time is negligible. The detection problem from here on will be predicated on this basic assumption of single failures only.

In the context of hypothesis testing, the hypothesis (sample) space should contain several segments, each one assigned to a possible failure source. There can be as many of these as there are measurement sources capable of breaking down. For the purpose of discussion, we will specifically consider only four hypothesis segments, each corresponding to one of the four GPS satellites needed for the navigation process. To maintain consistent notation, these hypotheses are designated as $\alpha$ in the following assignments:

- Hypothesis $\alpha_1$: Measurement $z_1$ from Satellite #1 is "bad";
- Hypothesis $\alpha_2$: Measurement $z_2$ from Satellite #2 is "bad";
- Hypothesis $\alpha_3$: Measurement $z_3$ from Satellite #3 is "bad";
- Hypothesis $\alpha_4$: Measurement $z_4$ from Satellite #4 is "bad";
- Hypothesis $\alpha_5$: No failure present.

Bear in mind also that these segments are, in reality, composite hypotheses which constitute a myriad of possible failure modes of varying
Figure 4.1. A multiple hypothesis setting for failure detection
magnitudes for each source. Examples of failure modes include various signal types like a bias, a ramp, or a random walk, while the magnitudes refer to the size of the parameters that describe these failures. A ramp, for instance, requires two parameters: a bias (or time of onset) and a slope. In addition to the segments identified with each possible failure source, a null hypothesis is included to cover the "no failure" situation as well. Figure 4.1 shows the basic block diagram for the multiple hypothesis concept.

In the following sections, three different models, all based on the Magill adaptive scheme, will be formulated and compared.

A. The Two-Slack-Variable Model

This model is called the Two-Slack-Variable Model because the two parameters which describe a ramp, the slope and bias, are incorporated as additional state variables into the Kalman filter model. The process model of eq. 2.13 then becomes

\[
\begin{bmatrix}
\Phi_x & 0 & 0 & 0 & 0 & 0 \\
0 & \Phi_x & 0 & 0 & 0 & 0 \\
0 & 0 & \Phi_x & 0 & 0 & 0 \\
0 & 0 & 0 & \Phi_c & 0 & 0 \\
\end{bmatrix}
\begin{bmatrix}
x \\
y \\
\beta_{k+1} \\
\end{bmatrix}
- 
\begin{bmatrix}
x \\
y \\
\beta_k \\
\end{bmatrix}
+ 
\begin{bmatrix}
w \\
0 \\
0 \\
\end{bmatrix}
\]

(\(x\) is an 11-tuple vector)
It is in the measurement model (see eq. 2.14) that things differ for the various source hypotheses. In eq. 4.2, the two slack variables are "connected" to whichever measurement the hypothesis is associated with.

\[
\begin{bmatrix}
    z_1 \\
    z_2 \\
    z_3 \\
    z_4
\end{bmatrix} =
\begin{bmatrix}
    h_{x1} & 0 & 0 & h_{y1} & 0 & 0 & h_{z1} & 0 & 0 & c & 0 & \mathbf{1}(1)\tau & \mathbf{1}(1) \\
    h_{x2} & 0 & 0 & h_{y2} & 0 & 0 & h_{z2} & 0 & 0 & c & 0 & \mathbf{1}(2)\tau & \mathbf{1}(2) \\
    h_{x3} & 0 & 0 & h_{y3} & 0 & 0 & h_{z3} & 0 & 0 & c & 0 & \mathbf{1}(3)\tau & \mathbf{1}(3) \\
    h_{x4} & 0 & 0 & h_{y4} & 0 & 0 & h_{z4} & 0 & 0 & c & 0 & \mathbf{1}(4)\tau & \mathbf{1}(4)
\end{bmatrix}
\begin{bmatrix}
    x_k \\
    \nu_k
\end{bmatrix} + \mathbf{v}_k
\]

where the unit operator \( \mathbf{1}(1) \) is given by

\[
\mathbf{1}(i) =
\begin{cases}
    1 & \text{if the hypothesis } a_i \\
    0 & \text{otherwise}
\end{cases}
\]

and the time elapsed variable associated with the ramp is \( \tau = k\Delta t \), \( \Delta t \) being the sampling interval.

As a result, the source hypotheses are separate individual Kalman filters each with a process model corresponding to eq. 4.1, and the appropriate measurement model from eq. 4.2. In the null hypothesis, the slack variables are not connected to any of the measurements. Consequently, the Kalman filter corresponding to this null hypothesis does not need to add on the additional states. The process and measurement models for this Kalman filter are then just those given by eqs. 2.13 and 2.14.
In this model, since the $H$-matrix of eq. 4.2 is different for the various hypotheses, the recursive covariance computations must be kept separately for the different hypotheses. Fortunately, there are only a few hypotheses that need to be implemented. As a result, the covariance of the measurement residuals here are, in general, different for the various Kalman filters. The likelihood and posterior probability computations of the Magill scheme are as given in eqs. 2.19 (or 2.23) and 2.22.

The initial conditions for the slope and bias slack variables may be chosen accordingly. In the absence of any prior knowledge regarding these failure parameters, they should have expected values of zero. The initial setting of the uncertainty in the slope variable $\sigma_y$ is nominally a few meters per second while that of the bias variable $\sigma_b$ can be set at a few tens of meters.

B. The Random Walk Model

This model is very similar to the Two-Slack-Variable Model in its structure whereby an additional state variable is used to describe the failure. Here, as opposed to assuming a deterministic ramp failure, we use the random walk which is a stochastic process that is nonstationary because its variance grows linearly with time. At least over the time spans considered, the growth in the variance would seem to go on indefinitely.

Not unlike the form of the Two-Slack-Variable Model described by eqs. 4.1 and 4.2, the process and measurement recursive equations of the Random Walk Model are given by eqs. 4.3 and 4.4 respectively.
\[
\begin{pmatrix}
\Phi_x 0 0 0 \\
0 0 0 0 \\
-0 0 \Phi_x 0 \\
0 0 0 \Phi\sigma \\
\end{pmatrix} \begin{pmatrix}
x \\
\omega_{k+1} \\
\end{pmatrix} + \begin{pmatrix}
w \\
0 \\
\end{pmatrix} \quad \text{; (x is an 11-tuple)}
\]

\[
\begin{pmatrix}
z_1 \\
z_2 \\
z_3 \\
z_4 \\
\end{pmatrix} = \begin{pmatrix}
h_{x1} 0 0 h_{y1} 0 0 h_{z1} 0 0 c 0 \\
h_{x2} 0 0 h_{y2} 0 0 h_{z2} 0 0 c 0 \\
h_{x3} 0 0 h_{y3} 0 0 h_{z3} 0 0 c 0 \\
h_{x4} 0 0 h_{y4} 0 0 h_{z4} 0 0 c 0 \\
\end{pmatrix} \begin{pmatrix}
1(1) \\
1(2) \\
1(3) \\
1(4) \\
\end{pmatrix} + \begin{pmatrix}
v_k \\
\omega_k \\
\end{pmatrix}
\]

where the unit operator \(1(i)\), again, is given by

\[
1(i) = \begin{cases} 
1 & \text{if the hypothesis } \alpha-i \\
0 & \text{otherwise.}
\end{cases}
\]

Here again, the initial conditions for the augmented random walk state variable \(\omega\) are set according to any knowledge about the prior distribution of initial estimate for this variable.

C. The Disjoint Hypotheses Model

In both the previous models, the null hypothesis is a special condition of any and all of the other failure source hypotheses. This condition exists for each Kalman filter when the augmented set of state or states lie in a region
Figure 4.2. Hypotheses of the failure signature corrupting the measurements
near zero. A more proper segmentation of the hypothesis space is achieved with the Disjoint Hypotheses Model.

To introduce the concepts of this model, consider the matched filtering idea incorporated in the block diagram of Figure 4.2. Each filter in the parallel bank works on the same measurement information less the "signature" of the failure signal that is hypothesized. The filter that comes closest to "matching" the actual failure signature will end up processing the "adjusted" measurement sequence that best approximates the uncorrupted signal itself. This information works its way into measurement residuals which, when appropriately weighted with the associated variance $V_k$, completes the likelihood computation. Since the failure affects only the measurement in an additive fashion, the $V_k$'s are free of the parametrization of the failure. In other words, these variances are common to all the Kalman filters, irrespective even of the source the failure originates from.

Figure 4.3. Discrete parameter hypothesis space segmented into composite source hypotheses
From hereon, we shall restrict our attention in the model to failure only in the form of a deterministic ramp. Hence, let us suppose that the discrete elemental "dots" of Figure 4.3 each represent a Kalman filter assigned to a point in the two-parameter space of the ramp failure signature. Each segment in this figure represents a hypothesis of which measurement source has failed. These segments will be termed (failure) source hypotheses and carry the same assignments as the $\alpha$-hypotheses defined earlier. The elements that constitute these segments, on the other hand, represent hypotheses of the failure slope and bias parameters. These elements will, on occasion, be known as (failure) parameter hypotheses.

In Figure 4.3, the elements that are very close to the origin designate a failure that is of negligible size since the slope and bias parameters are nearly zero. These elements from all four source hypotheses can then be grouped into another composite segment representing the null hypothesis. The boundaries of this "null" set of elements are, of course, quite arbitrary.

Within each segment, however, the two-dimensional space is continuous. As a result, a collection of parameter hypotheses that are needed to cover the entire segment exactly would be mathematically uncountable and infinite in number. At this point then, we need to part with the discrete notion that introduced the matched filtering idea, and proceed to the idea of a continuous implementation of the failure hypothesis. If the failure parameters, $\gamma$ (slope) and $\beta$ (bias), are continuous variables, we can then treat the 4-tuple observed measurement vector $z$ as a linear function of those parameters. Thus we have for the $\alpha_1$ hypothesis:
The matrix $B$ shown in eq. 4.3 corresponds to the Kalman filter that hypothesizes (the source of) $Z_1$ to be "bad". In a similar way, for the $i^{th}$ source hypothesis ($i=1, 2, 3, 4$), the first and second columns of the $i^{th}$ row of its $B$ matrix will be nontrivial. Although shown as a vector, the measurement set can also be treated as a series of measurements processed sequentially, provide $R_k$ is diagonal. The functional dependence on the failure parameters of the corrupted measurement in eq. 4.5 also affects the state estimation algorithm in the same way. The measurement residual computation is given by:

$$\begin{align*}
\mathbf{z} &= (\mathbf{v} - \mathbf{z})(\mathbf{u} - \mathbf{h}^T) \\
&= -B\mathbf{u} - \begin{bmatrix} h_1 & h_2 & h_3 \end{bmatrix} \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} \begin{bmatrix} \gamma \\ \beta \\ 1 \end{bmatrix} \\
&= -D\mathbf{u}
\end{align*}$$

where

$$\begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix} \begin{bmatrix} \gamma \\ \beta \\ 1 \end{bmatrix} = C\mathbf{u}$$
The state update equation of the Kalman filter then becomes:

\[ \hat{\mathbf{x}}^* = C \mathbf{u} + K \mathbf{z} \]

\[ - (C + K(B - H C)) \mathbf{u} \]  

(4.7)

Upon projection to the next time step, the new *a priori* state estimate is given by:

\[ \hat{\mathbf{x}}^- = \phi \hat{\mathbf{x}}^* \]

\[ - \phi [C + K(B - H C)] \mathbf{u} \]  

(4.8)

The time indices have been left out of the above equations to reduce the notational clutter. We can now put eqs. 4.6, 4.7 and 4.8 together to extract the recursive form, this time with appropriate time indexing, of the coefficient matrix C:

\[ C_{k+1} = \phi_k \left[ C_k + K_k (B_k - H_k \cdot C_k) \right] \]

\[ - (\phi_k - K_k H_k) C_k + K_k B_k \]  

(4.9)

In the Kalman filter recursive loop, all the pertinent information about the state estimates belonging the different failure hypotheses is compactly carried through in the coefficient matrix C. In other words, we do not need to implement any more than just one Kalman filter to handle all the parameter hypotheses in the measurements from one source.
While the continuous hypothesis space is an appropriate setting for the failure parameters, the failure source hypotheses must still be kept discrete because of differences in the $B$ matrix. Fortunately, with a 4-measurement model, we need only four source hypotheses or, equivalently, four Kalman filters conforming to a parametrized algorithm given in part by eqs. 4.7 and 4.8.

On the other hand, all covariance-related computations, including that of the Kalman gain $K$ and the residual variance $V$, are unaffected by this parametrized formulation. As mentioned previously, these quantities are common to and independent of the parameter as well as the source hypotheses. In short, the four parallel Kalman filters needed to represent the source hypotheses are all served by the same gain and error covariance computations.

The log-likelihood function computations similar to eq. 2.23 can also be rewritten as:

$$
\log L_k = A_k - \frac{1}{2} \mathbf{z}_k^T V_k^{-1} \mathbf{z}_k + \log L_{k-1}
$$

$$
= A_k - \frac{1}{2} \mathbf{u}^T D_k V_k^{-1} D_k \mathbf{u} + \log L_{k-1}
$$

$$
= A_k - \frac{1}{2} \mathbf{u}^T W_k \mathbf{u} + \log L_{k-1}
$$

(4.10)

where

$$
A_k = - \frac{1}{2} \log [(2\pi) |V_k|]
$$

Because of the dependence of $D_k$ on the source hypotheses $\alpha_i$ (i=1,2,3,4), the likelihood functions must be kept track of separately for each $\alpha$. The $V_k$'s, however, are common to all the source hypotheses and so $A_k$ in eq. 4.10 plays
no part in the posterior probability computations of eq. 4.16. The likelihood and log-likelihood function in eq. 4.10 can be written more compactly as

\[
L_k = L_{k-1} \exp \left( -\frac{1}{2} \mathbf{u}^T \mathbf{W}_k \mathbf{u} \right) \\
= \exp \left( -\frac{1}{2} \mathbf{u}^T \Lambda_k \mathbf{u} \right)
\]

(4.11)

where \( \Lambda_k \) additively accumulates information from the \( \mathbf{W}_k \) matrices. This is more evident from the logarithmic version of eq. 4.11:

\[
\log L_k = \log L_{k-1} - \frac{1}{2} \mathbf{u}^T \mathbf{W}_k \mathbf{u} - \frac{1}{2} \mathbf{u}^T \mathbf{W}_{k-1} \mathbf{u} - \ldots - \frac{1}{2} \mathbf{u}^T \mathbf{W}_0 \mathbf{u} \\
= -\frac{1}{2} \mathbf{u}^T \Lambda_k \mathbf{u}
\]

(4.12)

Of all the hypotheses considered, only the source hypotheses are of any importance to us. Since these are, in effect, composite hypotheses made up of the elemental parameter types, the latter must be "grouped" accordingly before computing the posterior probability associated with the former. This so-called "grouping" corresponds to a summation operation as in eq. 2.22. However, since the parameter hypotheses are in a continuous domain, the equivalent operation is integration. As with the discrete example mentioned above, a null hypothesis can be established here in the continuous domain by carving out a region arbitrarily close to the origin for all four source hypothesis segments. One of the four identical source segments is depicted in Figure 4.4 showing the null region denoted as \( H_i \) as being disjoint from the \( H_i \) region representing the source hypothesis \( \alpha_i \) (\( H_i \) and \( H_i \) are complementary
Figure 4.4. The continuous parameter hypothesis space for the $i^{th}$ source hypothesis

sets). Hence, the equivalent likelihood function for the composite source hypothesis $\alpha_i$ is:

$$L(\alpha) = \left| \sum _{i=1}^{4} L(\gamma, \beta) \right| ; \ \alpha = \alpha_1, ..., \alpha_4$$  \hspace{1cm} (4.13)

$$L(\alpha_0) = L'(\alpha_1) + L'(\alpha_2) + L'(\alpha_3) + L'(\alpha_4)$$  \hspace{1cm} (4.14)

where

$$L'(\alpha_i) = \sum _{j=1}^{4} \alpha_{ij}^* L_{ij}(\gamma, \beta) ; \ i=1,2,3,4$$  \hspace{1cm} (4.15)

and $L_{ij}(\gamma, \beta)$ is obtained from eq. 4.11 with a slight change in notation - the subscripted index refers here to the source hypothesis and not to the time step; and the parameters $\gamma, \beta$ are shown explicitly instead of $u$. The posterior probabilities associated with each of the source hypotheses and the null hypothesis is then given by:
At the point in time when a decision is to be made, the source hypothesis \( \alpha_i \) corresponding to the largest \( P(\alpha_i) \) should be selected. Note that the denominator of eq. 4.16 normalizes the posterior probabilities to add to one. A more detailed description of the implementation of this model will be given in the following chapter.

D. A Comparison of the Magill-based Detection Schemes

The Two-Slack-Variable Model provides a good optimal description of a deterministic ramp failure. However, by augmenting its state vector with two additional “slack” variables, the dimensionality and, hence, the numerical burden of the Kalman filter computations is increased. A perhaps more subtle weakness of this model is that the null hypothesis is a really just a special case, or a subset, of each and every one of the failure hypotheses. Thus the hypotheses space is not disjoint, which means that the parallel filter scheme cannot be defended rigorously in this model.

Also, one has little control in adjusting the sensitivity of the posterior probability calculation corresponding to the null hypothesis. The way to get around this problem is to add the slack variables into the null hypothesis model and “connect” them up to every one of the measurements. By restricting the initial uncertainty of these two variables in the state estimation algorithm, we can obtain some degree of freedom in adjusting the performance of the null hypothesis. The latter however remains a subset of
the failure hypotheses.

The same problems mentioned also apply to the Random Walk Model since it has a very similar structure to the Two-Slack-Variable Model. One of the advantages this model has over the first, though, is that the random walk process requires only one state variable and, hence, demands less of an increase in the dimensionality of the system. Also, the random walk appears to be somewhat more robust when approximating other processes.

Additionally, both models have different covariance structures and must as such implement these separately for however many Kalman filters are needed to represent the hypotheses.

None of these problems are encountered in the conceptually more elaborate Disjoint Hypotheses Model. As the name implies, the hypotheses, including the null, are mutually disjoint. The arbitrary region designated on each source hypothesis segment to make up part of the null hypothesis can be however defined to provide the necessary adjustments for the null hypothesis. Since there is commonality of the gain and covariance calculations among all the hypotheses, a vast amount of computational effort is saved. There is still a sizeable amount of numerical work needed to keep track of the likelihood functions. Regardless, there is ample net savings achieved when compared with the requirements of the other two models.

The three models were chronologically developed in the order as given. Although some earlier work was done with the first two models, the bulk of the results obtained and presented in this report come from studies done with the Disjoint Hypotheses Model. The former are included in this discussion for the sake of completeness.
V. DETAILED DISJOINT HYPOTHESES MODEL

Much of the theoretical foundations for this detection model has already been laid in the previous chapter. Here, we will consider, further aspects of its implementation. The pertinent equations for the likelihood and posterior calculations of eqs. 4.11, 4.13, 4.14, 4.15 and 4.16 will be reiterated below:

\[ L_\alpha = \exp \left( -\frac{1}{2} \mathbf{u}^T \Lambda_\alpha \mathbf{u} \right) \]  
\hspace{2cm} (4.11)

(With the hypothesis index \( \alpha \) used here instead of the time index \( k \).)

\[ L(\alpha) = \int_{H_\alpha} L_\alpha(\gamma, \beta) ; \ \alpha = \alpha_1, ..., \alpha_4 \]  
\hspace{2cm} (4.13)

\[ L(\alpha_0) = L'(\alpha_1) + L'(\alpha_2) + L'(\alpha_3) + L'(\alpha_4) \]  
\hspace{2cm} (4.14)

Where

\[ L'(\alpha_i) = \int_{H_{\alpha_i}} L_{\alpha_i}(\gamma, \beta) ; \ i = 1, 2, 3, 4 \]  
\hspace{2cm} (4.15)

\[ P(\alpha) = L(\alpha) / (\Sigma_{\alpha} L(\alpha)) ; \ \alpha = \alpha_0, ..., \alpha_4 \]  
\hspace{2cm} (4.16)

Equation 4.13 can be written explicitly in terms of multiple integrals, such as:

\[ L(\alpha) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} L_\alpha(\gamma, \beta) \ \mathrm{d}\gamma \ \mathrm{d}\beta \ - \int_{-\delta}^{\delta} \int_{-\epsilon}^{\epsilon} L_\alpha(\gamma, \beta) \ \mathrm{d}\gamma \ \mathrm{d}\beta, \ \alpha = \alpha_1, ..., \alpha_4 \]  
\hspace{2cm} (5.1)

Let us designate the first double integral as \( P_{\text{full}}(\alpha) \) and the second one as
Note that, given the nature of the integrand $L_\alpha(y, \beta)$, $P_{\text{null}}(\alpha)$ is easy to obtain while $P_{\text{null}}(\alpha)$ is not. Let us then first look at the evaluation of the former.

A. Computing the Likelihood Function of a Source Hypothesis

Equation 4.11 represents a function of two independent variables, $y$ and $\beta$ (note that $A_\alpha$ is symmetric). This can also be written, with implicit dependence on $\alpha$, as:

$$L = \exp \left(-\frac{1}{2} \begin{bmatrix} y & \beta & 1 \end{bmatrix} \begin{bmatrix} \lambda_{11} & \lambda_{12} & \lambda_{13} \\ \lambda_{12} & \lambda_{22} & \lambda_{23} \\ \lambda_{13} & \lambda_{23} & \lambda_{33} \end{bmatrix} \begin{bmatrix} y \\ \beta \\ 1 \end{bmatrix} \right)$$

$$= \exp \left(-\frac{1}{2} \begin{bmatrix} y - \mu_1 & \beta - \mu_2 \end{bmatrix} \begin{bmatrix} \lambda_{11} & \lambda_{12} \\ \lambda_{12} & \lambda_{22} \end{bmatrix} \begin{bmatrix} y - \mu_1 \\ \beta - \mu_2 \end{bmatrix} - \frac{1}{2} (\lambda_{33} - \lambda_{33}^*) \right)$$

where

$$\begin{bmatrix} \mu_1 \\ \mu_2 \end{bmatrix} = \begin{bmatrix} \lambda_{11} & \lambda_{12} \\ \lambda_{12} & \lambda_{22} \end{bmatrix}^{-1} \begin{bmatrix} \lambda_{13} \\ \lambda_{23} \end{bmatrix}$$

and

$$\lambda_{33}^* = \lambda_{11} \mu_1^2 + \lambda_{22} \mu_2^2 + 2\lambda_{12} \mu_1 \mu_2$$

To note in passing, the 2-tuple $(\mu_1, \mu_2)$ is what is commonly known as the
maximum likelihood estimate of the principal parameters, $\gamma$ and $\beta$. In this case, the joint distribution of $(\mu_1, \mu_2)$ is bivariate Gaussian with a covariance matrix of

$$
\begin{bmatrix}
\lambda_{11} & \lambda_{12} \\
\lambda_{12} & \lambda_{22}
\end{bmatrix}
$$

The mean of this distribution is dependent on the actual values of $\gamma$ and $\beta$.

Using a familiar mathematical identity, we then have

$$
P_{\text{full}} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} L(\gamma, \beta) \, d\gamma \, d\beta
$$

$$
= \exp\left(-\frac{1}{2}(\lambda_{33} - \lambda_{33}^*)\right) \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \exp\left(-\frac{1}{2} \left[ \begin{array}{c}
\beta - \mu_1 \\
\beta - \mu_2
\end{array} \right] \begin{bmatrix}
\lambda_{11} & \lambda_{12} \\
\lambda_{12} & \lambda_{22}
\end{bmatrix}^{-1} \begin{array}{c}
\beta - \mu_1 \\
\beta - \mu_2
\end{array} \right) \, d\gamma \, d\beta
$$

$$
= 2\pi (\lambda_{11}\lambda_{22} - \lambda_{12}\lambda_{21})^{-1} \exp\left(-\frac{1}{2}(\lambda_{33} - \lambda_{11}\mu_1^2 - \lambda_{22}\mu_2^2 - 2\lambda_{12}\mu_1\mu_2)\right)
$$

(5.5)

As complicated as these equations look, they are far more manageable when it comes down to implementing them in terms of a computational procedure.

B. Computing the Likelihood Function of the Null Hypothesis

As pointed out before, the integrals associated with $P_{\text{null}}$ do not yield explicit solutions and have to be evaluated either numerically using finite elements or by a table lookup scheme. This first alternative can easily use up a tremendous amount of computational resource depending on the accuracy
sought. On the other hand, a table-based scheme is more straightforward although it requires some manipulation and changes in the assumptions of the original formulation. We shall use the latter approach.

In order to be able to formulate a table lookup procedure, the integral problem must meet two conditions:

1. The integrand, which in this case is the likelihood function \( L \), must be factorable into two components, each of which is a function of only one independent variable;
2. The region of integration must have boundaries that are parallel to the orthogonal axes of the variables of integration. As long as the null region is rectangular in shape, this condition is satisfied.

Figure 5.1 shows the contours of the likelihood function projected onto the \( \gamma-\beta \) hypothesis plane.

Figure 5.1 shows the contours of the likelihood function \( L \) for a source hypothesis projected onto the \( \gamma-\beta \) plane. In general, as depicted in the skewed orientation of the ellipses, the likelihood function does not satisfy the first condition above because it contains nontrivial terms that involve both \( \gamma \)
and \( \beta \). Fortunately though, due to the quadratic form of the exponent in \( L \), a linear transformation exists that will decouple the function. A simple linear transformation that will carry out this task is an operation which effects a geometric rotation centered at the extremum point \((\mu_1, \mu_2)\) on the frame of reference (see Figure 5.2). The associated transformation matrix is of the form:

\[
\begin{bmatrix}
  u \\
  v
\end{bmatrix}
= \begin{bmatrix}
  c & -\sqrt{1-c^2} \\
  \sqrt{1-c^2} & c
\end{bmatrix}
\begin{bmatrix}
  \gamma - \mu_1 \\
  \beta - \mu_2
\end{bmatrix}
\]

(5.6)

When the quadratic portion of the exponent in the likelihood function of eq. 5.3 is subjected to this transformation, we have

\[
\begin{bmatrix}
  \gamma - \mu_1 & \beta - \mu_2
\end{bmatrix}
\begin{bmatrix}
  c & -\sqrt{1-c^2} \\
  \sqrt{1-c^2} & c
\end{bmatrix}
\begin{bmatrix}
  \lambda_{11} & \lambda_{12} \\
  \lambda_{12} & \lambda_{22}
\end{bmatrix}
\begin{bmatrix}
  c & \sqrt{1-c^2} \\
  \sqrt{1-c^2} & c
\end{bmatrix}
\begin{bmatrix}
  \gamma - \mu_1 \\
  \beta - \mu_2
\end{bmatrix}
\]

(5.7)

The product of the central matrices must then be equated to a decoupled matrix:

\[
\begin{bmatrix}
  \xi_1 & 0 \\
  0 & \xi_2
\end{bmatrix}
= \begin{bmatrix}
  c & -\sqrt{1-c^2} \\
  \sqrt{1-c^2} & c
\end{bmatrix}
\begin{bmatrix}
  \lambda_{11} & \lambda_{12} \\
  \lambda_{12} & \lambda_{22}
\end{bmatrix}
\begin{bmatrix}
  c & \sqrt{1-c^2} \\
  \sqrt{1-c^2} & c
\end{bmatrix}
\]

(5.8)

The parameter of rotation can then be solved from the relation:

\[
c^4 - c^2 + \lambda_{12}^2 \left( 4 \lambda_{12}^2 + \lambda_{11}^4 + \lambda_{22}^4 - 2 \lambda_{11}^2 \lambda_{22}^2 \right)^{-1} = 0
\]

(5.9)
This satisfies condition #1, but, as can be seen in Figure 5.2, the null region is distorted as a consequence. Now, since the null region was an arbitrarily specified area to begin with, we can justifiably choose to rearrange its borders so that the redefined area is rectangular, hence satisfying condition 2, after the rotational transformation has been carried out. The implications of this choice is minimal since the physical area of this region is quite small. However, because the angle of rotation is different at each step, presumably due to changes in the L matrix, the null region defined in this manner also changes. Even so, in the simulations that have been run to test out this approach in computing the likelihood function associated with the null region, the changes are ever so slight after the first few processing steps.

Having settled on the justification of this approximation, we now consider the evaluation of the likelihood function over this null region:
\[ P_{\text{null}} = \exp\left(-\frac{1}{2} \left( \lambda_{33} - \lambda_{11} \mu_1^2 - \lambda_{22} \mu_2^2 - 2\lambda_{12} \mu_1 \mu_2 \right) \right) \times \]
\[ \int_{u^*-\delta}^{u^*+\delta} \int_{v^*-\epsilon}^{v^*+\epsilon} \exp\left(-\frac{1}{2} \begin{bmatrix} u & v \end{bmatrix} \begin{bmatrix} \xi_1 & 0 \\ 0 & \xi_2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} \right) \, dv \, du \]
\[ - 2\pi (\xi_1 \xi_2)^{-1} \text{Gal}[\xi_1(u^*-\delta), \xi_1(u^*+\delta)] \text{Gal}[\xi_2(v^*+\epsilon), \xi_2(v^*+\epsilon)] \times \]
\[ \exp\left(-\frac{1}{2}(\lambda_{33} - \lambda_{11} \mu_1^2 - \lambda_{22} \mu_2^2 - 2\lambda_{12} \mu_1 \mu_2) \right) \]

(5.10)

with Ga[•, •] denoting the standard Gaussian function (zero mean and unit variance) evaluated over the specified interval. Also in eq. 5.10, \((u^*, v^*)\) is the location of the origin of the \(\gamma-\beta\) reference frame in the transformed \(u-v\) plane given by:

\[
\begin{bmatrix} u^* \\ v^* \end{bmatrix} = \begin{bmatrix} c & -\sqrt{(1-c^2)} \\ \sqrt{(1-c^2)} & c \end{bmatrix} \begin{bmatrix} -\mu_1 \\ -\mu_2 \end{bmatrix}
\]

(5.11)

The calculation of \(P_{\text{null}}\) actually corresponds to that of eq. 4.15 for each source hypothesis. With this, the likelihood function associated with the composite null hypothesis can be obtained as per eq. 4.14:

\[ L(\alpha_0) = P_{\text{null}}(\alpha_1) + P_{\text{null}}(\alpha_2) + P_{\text{null}}(\alpha_3) + P_{\text{null}}(\alpha_4) \]

(5.12)

The likelihood functions associated with the other source hypotheses are, in turn, given by:
\[ L(\alpha_i) = P_{\text{full}}(\alpha_i) - P_{\text{null}}(\alpha_i); \quad i=1,2,3,4. \] (5.13)

The Bayes relation of eq. 4.16 to compute the respective posterior probabilities would normally be the logical step to take next. However, there remains an issue to be addressed in the next section.

C. Weighting the "Null" Likelihood Function

It is clear from Figures 5.1 and 5.2 that the area of the region assigned to the null hypothesis is indeed small compared to the external region belonging to the source hypothesis. If one reverts back to the discrete hypothesis picture discussed previously in Chapter IV, then the disparity between adding up the likelihood measures from "equally spaced Kalman filters" within the null region and doing the same for those outside of that region is quite obvious. What happens then is that the likelihood and the posterior probability associated with the null hypothesis has small values compared to those of the source hypotheses. The latter, all having identical regions, all start on the same footing, so to speak, which is perfectly justified. Surely then, there is need for some form of compensation to balance out the differences caused by region sizes between the null and the source hypotheses.

This compensation cannot just be based on some inverse ratio of the areas because, clearly, the external region is of infinite size. The next progressive step to take would be to weight the locations on the \( \gamma - \beta \) plane differently such that points closer to the origin are weight more than those
further away. A location at infinity must then necessarily be assigned zero weight. The function chosen to dictate this weighting should be uniformly continuous and integrable over the entire $\gamma$-$\beta$ domain. One suitable function which meets these requirements is none other than that of the Gaussian density itself. This function, centered at the origin of the $\gamma$-$\beta$ reference frame, leaves the unspecified scaling parameters $\sigma_\gamma$ and $\sigma_\beta$ as a degree of freedom that can be used to manipulate the performance of the posterior probability of the null hypothesis with respect to those of the source hypotheses:

$$W(\gamma, \beta) = \frac{1}{(2\pi \sigma_\gamma \sigma_\beta)^{1/2}} \exp \left(-\frac{1}{2} \left[ \frac{\gamma^2}{\sigma_\gamma^2} + \frac{\beta^2}{\sigma_\beta^2} \right] \right)$$  \hspace{1cm} (5.14)

Note that this weighting should not be changed in the course of the data processing. It is fixed at the very start and should be used consistently throughout. In that sense, eq. 5.14 plays the role of specifying an initial distribution for the $\gamma$-$\beta$ parameter space. With this weighting distribution then, the ratio of the function integrated within the null region to that obtained outside of it can be calculated \textit{a priori}. This ratio, in turn, forms the basis for whatever compensation it takes, usually a multiplicative factor to rescale $P_{\text{null}}$ to start the null hypothesis off on par with the other hypotheses. Using the likelihood functions computed in eqs. 5.12 and 5.13, the Bayes' relationship of eq. 4.16 then becomes:

$$P(\alpha) = \frac{L(\alpha \mid \alpha_1) + L(\alpha_2) + L(\alpha_3) + L(\alpha_4) + \omega L(\alpha_0))^{-1}}{\text{for } \alpha = \alpha_1, \alpha_2, \alpha_3, \alpha_4}$$  \hspace{1cm} (5.15)
and,

\[ P(\alpha_0) = \omega L(\alpha_0)(L(\alpha_1) + L(\alpha_2) + L(\alpha_3) + L(\alpha_4) + \omega L(\alpha_0))^{-1} \]  

(5.16)

where \( \omega \) is the compensation factor for the diminutive size of the null region.

This admittedly is a somewhat ad hoc procedure. The ultimate choice of the size of the null region and the initial weighting distribution in such an approach can only, at best, be subjectively determined. In all the simulations done for this study, the multiplicative factor mentioned was chosen such that the posterior probability associated with the null hypothesis would start off, in the absence of noise, with a value of .2. The parameters \( \sigma_1 \) and \( \sigma_2 \) were taken to be 3 m/s and 40 m, respectively, and those for the null region were set at \( \delta = 2 \) m/s and \( \epsilon = 2 \) m.

D. The Prior Distribution

Although the "weighting" procedure suggested in the previous section was used exclusively in all the detection schemes implemented, an alternative approach would be to constrain the choice of the prior distribution of the likelihood function such that it conforms to providing the appropriate weighting to the null region. The emphasis here again is to equalize or compensate for the relative small size of the null region, whatever the choice of that may be.

The difference in this newer procedure now over the previous weighting procedure is that the choice of the prior distribution and the initial distribution of the posterior probabilities are no longer independent; the
specification of one, usually the latter, necessarily fixes the other. Using the "weighted" procedure, Figure 5.3 below shows that an independently-chosen prior distribution when weighted over the null region, gives rise to a discontinuous density function with a "concentration of mass" near the mean.

Figure 5.3. The initial distribution as a result of weighting the null region

This is, in effect, the same as choosing a prior distribution with much lower variance to begin with. Figure 5.4 shows that such a distribution is already determined in the choice of the ratio of the shaded areas (or volumes in the case of a bivariate distribution) by the predetermined initial distribution of the posterior probabilities.
E. Choosing the Null Region

There are problems that may arise if the null region chosen is too small. If the distribution of the posterior probabilities is fixed, the size of the null region is then proportional to the variance of the prior distribution needed to compensate for it. Obviously, the problem runs into singular conditions when the null region shrinks to zero. However, even before this occurs, numerical problems may very well materialize. Another factor that must be considered is that, in our simulations, bias errors were not included. In a real situation, of course, every attempt should be made to account for these errors although there can be situations where they may not be modeled completely. In such
cases, a very small null region will greatly increase the sensitivity to those unmodeled bias errors and, consequently, raise the false alarm rate in return. One possible solution to this would be to enlarge the rectangular null region and, in particular, elongate it in the dimension of the bias variate.
VI. THE SIMULATION MODEL

In this study, we are interested in demonstrating the performance of the detection scheme primarily in the flight environment of a civilian aircraft. This objective was realized, in a preliminary way, through simulations of the physical as well as the stochastic aspects of such an environment on a computer. This chapter is devoted to the theoretical and procedural set-up of the simulations.

A. Monte Carlo Simulation

In a Monte Carlo simulation, random processes are generated according to the specified statistics. Naturally, the response of a system to a particular set of these random inputs is just a statistical sample, though, in itself, this sample adheres to some implicit statistical characteristics. The key to an effective Monte Carlo simulation then is clearly ensuring that the random inputs used are statistically typical.

Frequently, the starting point of such simulations involving a sampled-data system is the generation of a time series of uncorrelated numbers, often called a white sequence. There has always been a large variety of random number algorithms available in the mathematical computation literature [35]. Of the many categories of such algorithms, the type that is most amenable to present-day computers and also most practical for analytical-oriented simulations is perhaps the recursive pseudorandom generators. As the name suggests, these generator output pseudorandom
sequences of numbers which eventually recycle themselves. The recycling period, however, can be set to be so large that the recycling is never encountered during the course of the simulation. To start the recursion, an initializing "seed of randomization" is required. This gives the analyst crucial access to the same random sequence should the need arise to duplicate the simulation experiment.

The random number generator chosen for the simulations done in this project is a method called the composite congruential generator [331 and follows the algorithm given here:

\[ x_k = 2^{-31} [(v_k + w_k) \mod 2^{31}] \]  
\[ \text{where} \]
\[ v_k = 65539 \cdot v_{k-1} \mod 2^{32} \]
\[ w_k = 262147 \cdot w_{k-1} \mod 2^{32} \]

The randomization seeds \( v_0 \) and \( w_0 \) are needed to start out the recursive process which yields \( x_k \) (for \( k = 1, 2, \ldots \)) that is uniformly distributed over the interval from 0 to 1. In order to obtain Gaussian-distributed numbers from the \( x_k \) sequence, a suitable transformation is needed. If we pick a pair of numbers \( \{x_{1k}, x_{2k}\} \) sequence, and transformed one of these into a Rayleigh-distributed sample, we would then have specified a point in polar coordinates on a bivariate Gaussian-distributed sample space (see Figure 6.1):

\[ R_k = \sqrt{2 \log(1/x_{1k})} \]  
\[ \theta_k = x_{2k} \]
Figure 6.1. Polar-cartesian coordinate transformation to obtain Gaussian random numbers from Rayleigh- and uniformly-distributed numbers.

Resolving this into two components of orthogonal Cartesian coordinates, we then have the corresponding uncorrelated and unbiased Gaussian-distributed numbers \( g_{1k}, g_{2k} \):

\[
\begin{align*}
g_{1k} &= R_k \cos (2\pi \theta_k) \\
g_{2k} &= R_k \sin (2\pi \theta_k)
\end{align*}
\] (6.4) (6.5)

With this procedure, we now have a way to generate a zero-mean Gaussian white sequence. There is often a need, also, to obtain finite sets of numbers that have a nontrivial correlation structure among the elements of each set but are uncorrelated between any two sets. A sequence of these sets can be thought of as a white vector sequence. If the vectors were 2-tuple, for example, the covariance matrix of each vector itself might then be given by:
The vector \( \mathbf{r} \) can easily be generated from a vector \( \mathbf{w} \) whose entries are taken directly from a white sequence and, as such, are uncorrelated with each other. This is realizable via a linear transformation matrix \( \mathbf{C} \), obtained through the Cholesky Decomposition Theorem:

\[
\begin{bmatrix}
\sigma_1^2 & \rho \sigma_1 \sigma_2 \\
\rho \sigma_1 \sigma_2 & \sigma_2^2
\end{bmatrix} =
\begin{bmatrix}
c_1 & c_2 \\
0 & c_3
\end{bmatrix}
\begin{bmatrix}
c_1 & 0 \\
c_2 & c_3
\end{bmatrix} = \mathbf{C} \mathbf{C}^T
\]

\[\mathbf{r} = \mathbf{C} \mathbf{w}\]

since

\[
E(\mathbf{r}\mathbf{r}^T) = E(\mathbf{C} \mathbf{w} \mathbf{w}^T \mathbf{C}^T) = \mathbf{C} \mathbf{C}^T
\]

The Gaussian-distributed random numbers generated by eqs. 6.2-6.5 have zero means and unity variances. Other Gaussian distributions can then be derived from these normalized random numbers.

B. Aircraft Dynamics

An elaborate introduction to the formulation of dynamical models for aircraft flight is already given under the topic of Kalman filtering in Chapter.
II. There are basically two different types of flight environments. In the enroute situation, an aircraft undergoes long durations of near straightline flight under relatively benign conditions. On the other hand, the nonprecision approach encounters acceleration components from both control-induced turns, which are deterministic in nature, and low-altitude random atmospheric turbulence. In the simulation of these two situations, only their stochastic characteristics were varied. Obviously, the dynamics in the latter situation are the "noisier" of the two.

The discrete-time version of the differential equation given in eq. 2.6 used to generate the aircraft trajectories in all simulations is given by:

\[
\begin{bmatrix}
  x_1 \\
  x_2_{k+1}
\end{bmatrix} =
\begin{bmatrix}
  1 & \Delta t \\
  0 & 1
\end{bmatrix}
\begin{bmatrix}
  x_1 \\
  x_2_k
\end{bmatrix} +
\begin{bmatrix}
  w_1 \\
  w_2_k
\end{bmatrix}
\]  

(6.8)

where

- \( x_1 \) - position in one dimension;
- \( x_2 \) - velocity in the same dimension;
- \( w_1, w_2 \) - white sequence with the covariance matrix \( Q \); and

\[
Q = \begin{bmatrix}
S (\Delta t)^3/3 & S (\Delta t)^2/2 \\
S (\Delta t)^2/2 & S \Delta t
\end{bmatrix}
\]

The physical situation behind the simulation involves an aircraft traveling due south at about 180 m/s (approx. 400 mi/h) at an altitude of about 9000 m (nearly 30,000 ft) above the surface of the earth over a northeastern region of the United States.
In the enroute situation, the differential system of eq. 2.11 is driven by a white noise input with a spectral amplitude $S$ of $0.01 \text{ m}^2/\text{s}^3$, and thus yields a position state that "wanders off" as an integrated random walk process by nearly 60 meters over a 100-second time interval.

As for the nonprecision approach scenario, the spectral amplitude $S$ of the white noise input was raised to $1 \text{ m}^2/\text{s}^3$. Under these same stochastic conditions, there was also a simulation case where a deterministic turn was added to the flight path of the aircraft. The turn was initiated at the start of the time period and proceeded until a 90-degree change in direction heading east had been effected. The turn, executed without any change in speed, generated an acceleration vector with a constant magnitude of about $4g$. In the simulation program, this deterministic control was imparted by artificially manipulating the velocity vector of the original no-turn nonprecision approach case.

C. GPS Satellite Geometry

The GPS satellites chosen for this simulation belong to the proposed 18-satellite constellation with three satellites in each of six orbits inclined at $55^\circ$ to the equatorial plane. The layout of these orbits over the surface of the earth can be seen in Figure 2.1. In the simulation models, satellites SV3, SV7, SV10, and SV15 were chosen for the observation model. The location of the origin of the locally-level coordinate frame of reference was fixed at $N39^\circ8'1'', 77^\circ12'49''W$ on the WGS-72 ellipsoid. The GDOP of the set of satellites sighted by a receiver on the aircraft located within the vicinity of the reference origin
is approximately 6.5 in this particular situation. This may perhaps be a higher value than one would like to see in a navigating situation although it is one that is not highly atypical. The sequencing aspect of a single-channel receiver was emulated by processing the satellite measurements in turn one-quarter second apart. A four-satellite complement takes a one-second cycle to complete.

D. Auxiliary Measurements

Studies pertaining to the performance of the detection scheme when aided by additional measurements were also made. Several cases involving a fifth measurement were made, this being from the barometric altimeter. In the simulation model, this measurement is derived from an imaginary fifth satellite placed at the center of curvature of the local earth surface. This "satellite" is actually then located near the center of the earth. The difference between this measurement and the other regular GPS measurements is that there is no clock error present in the former. As such, the barometric altitude when modeled as originating from a satellite is, in fact, a range and not a pseudorange measurement. The noise of the baro-altimeter measurement is set at (30m)².

Another important auxiliary measurement used in certain simulation cases is the doppler or range-rate measurement. This information comes from estimating the instantaneous rate of change of the pseudorange when tracking the GPS signal carrier phase and relating that back to the velocity states in the Kalman filter model. In the simulation model, the extraction of
the doppler information from the pseudoranges measured is bypassed. This measurement was presented directly to the detection processing scheme as a measurement of the velocity states corrupted by noise. The velocity measurement component in one dimension $d$ is then:

$$d = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

where

\begin{align*}
x_1 & \quad \text{position in one dimension;} \\
x_2 & \quad \text{velocity in the same dimension.}
\end{align*}

The size of the noise corrupting this measurement related to the doppler is of course highly dependent on the precision of the phase loops that track the GPS signal carrier.

Although the doppler provides valuable additional information for the system, it must be understood that this does not imply that any quantum leap in the performance can necessarily be expected. Qualitatively, this is because the additional doppler measurements do not provide extra redundancy in the sense that a fifth satellite or inertial aiding would. The doppler measurements coming from the failed satellite are, in effect, themselves "contaminated" by the same failure that corrupts its associated pseudoranges. The former, however, do provide the equivalence of statistical averaging to reduce the noise from the pseudorange measurements. This improvement, however, rests largely on the accuracy of the phase loops that track the carrier signal and, hence, may or may not yield significant overall
benefits especially when the extra data processing required is considered also.

As a final note here, it should also be mentioned that the clock model used was that equivalent to a high-quality crystal reference, the parameters for which was given in Chapter II.
VII. SIMULATION RESULTS

Most of the computer simulations were done on a NAS-9160 mainframe located at the Computation Center of Iowa State University. Some preliminary investigations done prior to the principal part of the research work had been carried out on a Hewlett-Packard 87A personal computer. Results in this report will only include those obtained using the ultimate choice for a detection scheme, the Disjoint Hypotheses Model.

To avoid possible confusion, a clarification regarding the terminology used in some of the simulation cases to be presented is appropriate. Terms such as enroute or cruise and nonprecision approach are used to describe the physical environment that is being simulated. For consistency, the enroute situation connotes an environment where low dynamics is encountered, in contrast with those involved in the nonprecision approach. The terms low-dynamics and high-dynamics models refer, on the other hand, to the model that is being used which may or may not be the appropriate one for the given situation. In our simulations, though, the models used are always close to, if not in fact, matching the dynamical environment that is being simulated.

Most of the probability graphs that follow plot the progression of the posterior probabilities associated with just two relevant hypotheses: the null hypothesis, and the source hypothesis that, if it is present, correctly pinpoints the failure (assigned to Satellite #3 - SV10 in all simulations). The other plot traces have been left out to reduce the clutter in the graphs. There are a few graphs that do include plots for more than the two hypotheses mentioned.
An important thing to note here is that while the convergence of one of the probability plots to unity may indicate that the corresponding hypothesis is true, the convergence of the null hypothesis probability plot to zero, on the other hand, may also mean that it (the null hypothesis) is not the correct choice. This provides the solution to the first part of the problem, namely that of failure detection. Resolving the correct source hypothesis, usually done after the presence of the failure is confirmed, in turn, constitutes a solution to the problem of failure identification.

A. Comparison of Different Measurement Models

These simulations involved three measurement models, one with four, another with five, and a third with six measurements. The comparison is meant to show the differences in the performance of the detection scheme using the various measurement models. In order to deny the user's failure detection system of any benefit it may derive from knowing its own initial position, velocity and clock errors with fair accuracy, the Kalman filter states were initialized with sizeable uncertainties in one set of simulations. One of the simulated measurements, from Satellite #3, was then corrupted with a 1 m/s ramp failure. As is expected, the plots of Figure 7.1 show clearly that the system with four measurements has absolutely little idea about the existence of the failure, let alone identifying it. The five-measurement model of Figure 7.2 does a little better by recognizing the presence of a failure as suggested by the "grounding" of the null hypothesis (uniquely identified by the dotted line), although it too is unable to isolate the failure source. Finally, as is
Figure 7.1. The unsuccessful detection of a failure with 4 measurements in a noisy environment (numeric labels refer to the different hypotheses, of which #3 is the correct one.)
Figure 7.2. The unsuccessful isolation of a failure after detection with 5 measurements in a noisy environment (hypothesis #3 should be the correct choice.)
Figure 7.3. The successful detection and identification of a failure with 6 measurements in a noisy environment (hypothesis #3 is the correct choice.)
consistent with the basic concepts of failure monitoring first introduced in Chapter I, with two redundant measurements in the six-measurement model, both failure detection and identification is clearly achievable as demonstrated in Figure 7.3.

Although the time required to arrive at a decision on both counts may seem excessive, it must be remembered that these simulations do not represent typical conditions. Ordinarily, this problem is posed when the Kalman filter in a GPS receiver system in mid-flight is already in a steady-state condition. The next simulation results presented take this condition into account.

B. The Enroute Flight Environment

When the low-dynamics model is used in this environment, the Kalman filter takes advantage of the relatively "low-noise" stochastic conditions to obtain fairly precise state (position and velocity) estimates. Reiterating the point made in the previous section, when we have reasonably accurate state information, this adds to the redundancy of the failure detection system.

To illustrate this, the case given in Figure 7.4 simulates the use of a low-dynamics model utilizing only four satellite measurements to detect a 3 m/s ramp failure in Satellite #3. The failure is first recognized when the "null" posterior probability given by the dotted line goes to zero within 15 seconds, and the solid line representing the third source hypothesis hits one soon after. A corresponding look at a replication of the same situation without the failure, in Figure 7.5 shows a more gradual convergence of the
Figure 7.4. Detection of a 3 m/s ramp failure in an enroute situation using a low-dynamics model with 4 measurements.
Figure 7.5. No failure present in an enroute situation using a low-dynamics model
Figure 7.5. Detection of a 3 m/s ramp failure in an enroute situation using a low-dynamics model with 5 measurements.
Figure 7.7. Detection of a 1 m/s ramp failure in an enroute situation using a low-dynamics model with 4 measurements
Figure 7.8. Detection of a 1 m/s ramp failure in an enroute situation using a low-dynamics model with 5 measurements
correct hypothesis, the "null" this time. Although this is a more cautious approach to making a decision that no failure is present, it justifiably allows the model greater flexibility in handling even more subtle failures. Such failures may take the form of gentler ramps, perhaps, and these nominally require longer spans of time to resolve. Here, in the case of the no-failure situation, the important thing to note is that the scheme does indeed identify the correct hypothesis, even though it took somewhat longer to do so.

Where a fifth measurement was added in the form of baro-altitude, we see, shown in Figure 7.6, what seems to be a slight improvement over the same situation considered with four satellite measurements in Figure 7.4. The true value of the fifth measurement will be more apparent later on.

The next two cases considered in this rather benign environment involved adding a gentler ramp failure with a 1 m/s slope. The low-dynamics model was again invoked and the graphs of Figures 7.7 and 7.8 show the simulation results for a four- and a five-measurement model respectively. The gentler failure ramp here signifies that we have a smaller "signal" to work with amidst the same noisy conditions as before so its detectability is correspondingly lower. In the four measurement result, the failure was detected but the scheme, even in choosing the correct source hypothesis, was less confident in its choice; $P_3$ tapers off at about .8 over the observed time span. With five measurements, the failure detection does not improve much, although the identification aspect is noticeably better.

In another case considered in this section, the same situation as before was reenacted involving only the four-measurement model. The variation in the simulation again centers on the failure but, this time, a random walk
process is generated. For limited comparative purposes, the size of the random walk was chosen to be the approximate equivalent of a 3 m/s ramp over a 40 second period. The graph of Figure 7.9 would seem to confirm the detectability of this type of failure by the detection scheme.

As further evidence of the robustness of this model to failures other than ramps beginning at t=0, the simulation in Figure 7.10 shows the system response to a step or bias type failure of the order of 50 meters. A condition like this occurs when a sizeable failure is already present at the start of the detection problem, and its prompt detection is then very critical. Here again in the simulation, the low-dynamics model with a four pseudorange measurements were used. The failure was easily detected and identified within five seconds.

The last case in this category involves the use of doppler information to provide measurements for the velocity states of the system. This supplement to the pseudorange measurements helps improve the accuracy of the information from each of the satellites. The degree of improvement depends very much on the noise error of the velocity measurements. When doppler is included, Figures 7.11 and 7.12 show the differences in the improved rates of detection for velocity measurement errors of 4 m/s and 1 m/s respectively. The improvement seen in going from the former case to the latter is quite remarkable. The white measurement error standard deviations used in these two simulations may seem to be rather large. However, it should be remembered that these refer to values associated with individual doppler samples and not the final "smoothed" accuracy.
Figure 7.9. Detection of a random walk failure in an enroute situation using a low-dynamics model with 4 measurements.
Figure 7.10. Detection of a 50m step (bias) failure in an enroute situation using a low-dynamics model with 4 measurements.
Figure 7.11. Detection of a 3m/s ramp failure in an enroute situation using a low-dynamics model with 4 measurements and noisy doppler measurements.
Figure 7.12. Detection of a 3 m/s ramp failure in an enroute situation using a low-dynamics model with 4 measurements and reasonably accurate doppler information.
C. The Nonprecision Approach Flight Environment

Due to the larger magnitudes of the random dynamics of an aircraft in the nonprecision approach environment, we would rightly expect some degradation in the performance of the detection scheme that had exhibited commendable results in the previous section. Of course, the larger point of curiosity is how much degradation will it be and how much of it can be tolerated. The simulations in this section are intended to answer some of these questions.

The first simulations in this category were made with the 8-state Kalman filter model similar to the low-dynamics version described in Chapter II, but differs in having a higher spectral amplitude in its white noise input (1 m²/s³ instead of .01 m²/s³). Figures 7.13 and 7.14 depict results using the above model using four and five measurements respectively to detect a 3 m/s ramp failure. Similarly, the same situations were duplicated for a 1 m/s ramp failure, the results of which appear in Figures 7.15 and 7.16, again, for four and five measurements respectively. When compared to the same four situations simulated in the enroute environment, the results here, not surprisingly, show a marked loss of sensitivity in the detection process.

Although the 8-state model with an increased noise input seems capable of handling the higher dynamical environment of the nonprecision approach, the next simulation shows its inadequacy when its comes to handling fairly sizeable accelerations. This simulation puts the aircraft into a deterministic 90-degree turn that induces a .4g acceleration component which lasts for about 70 seconds. To test the robustness of this 8-state
Figure 7.13. Detection of a 3 m/s ramp failure in a nonprecision approach situation using an 8-state high-dynamics model with 4 measurements.
Figure 7.14. Detection of a 3 m/s ramp failure in a nonprecision approach situation using an 8-state high-dynamics model with 5 measurements.
Figure 7.15. Detection of a 1 m/s ramp failure in a nonprecision approach situation using an 8-state high-dynamics model with 4 measurements.
Figure 7.16. Detection of a 1 m/s ramp failure in a nonprecision approach situation using an 8-state high-dynamics model with 5 measurements.
Figure 7.17. No failure present in a high-dynamical turn maneuver using an 8-state high-dynamics model with 5 measurements.
Figure 7.18. No failure present in a high-dynamical turn maneuver using an 11-state high-dynamics model with 5 measurements
model, no failure was introduced. The dotted line of the null hypothesis should go to one if the proper decision was made. In Figure 7.17, however, although the null posterior probability eventually reaches for unity, a false alarm decision might well have been set off at the 25-second mark. This is clearly not desirable. Going to an 11-state filter model, the same simulation was repeated. Figure 7.18 shows that this model which accounts for the acceleration as a Markov process fares much better in not detecting a failure where there was none. A five-measurement model was used in both models in this comparison. (The fifth measurement was baro-altitude.)

Clearly then, the 11-state model with its increased complexity is the more appropriate model in this situation. The next set of simulations are intended to scrutinize the performance of the 11-state model in detecting a 3 m/s ramp failure under normal high-dynamics conditions in the nonprecision approach environment. As might be expected, Figures 7.19 and 7.20 show some degradation in detectability when compared to the same situations given in Figures 7.13 and 7.14.

Although it may be better, from a practical point of view, to use one standard high-dynamics model to cover both flight environments considered, the performance of such a detection scheme in the enroute environment might be substantially compromised. This model does not taking advantage of the low-dynamical conditions which themselves provide a form of redundancy.
Figure 7.19. Detection of a 3 m/s ramp failure in a nonprecision approach situation using an 11-state high-dynamics model with 4 measurements
Figure 7.20. Detection of a 3m/s ramp failure in a nonprecision approach situation using an 11-state high-dynamics mode with 5 measurements.
D. Summary of Results

The results from all the simulations discussed in the previous sections are organized in Table 1 for easy reference. The determination of whether the figures listed under the time-to-detection and time-to-identification are adequate enough to warn the flight crew of an aircraft of a failure can be an ambiguous proposition. The threshold of safe operation in terms of accurate positioning is eventually breached when a failure corrupts the system long enough. This time duration, however, depends on a multitude of factors, not the least of which is the size of the failure itself. Others include the satellite geometry, the Kalman filter model (process and measurement) used, and the size of unmodeled bias errors. For our purposes here, we will base this threshold value on the effects of a 3 m/s ramp failure. From Figure 3.2, for the enroute situation where a fairly high tolerance of position error is permitted, the threshold should be set at about 100 seconds to be conservative. Putting all the results from the enroute scenario into perspective, it would appear that a failure occurring in such an environment can be detected and identified within the given threshold in all cases with the possible exception of detecting very subtle failures using a four-measurement model.

The same conclusions are not apparent for the nonprecision approach situation. From a similar determination of the error a 3 m/s ramp failure causes an 8-state Kalman filter in a nonprecision approach environment, an interval of about 30 seconds is a reasonable threshold to set. The entries in Table 1 for the nonprecision approach environment suggest that the
Table 1. Summary of failure detection results from different simulated conditions

<table>
<thead>
<tr>
<th>Figure</th>
<th>Flight Environment</th>
<th>Model Dynamics</th>
<th>Measurements</th>
<th>Failure</th>
<th>Time to Detection</th>
<th>Time to Identification</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.4</td>
<td>Enroute</td>
<td>Low</td>
<td>4</td>
<td>3 m/s ramp</td>
<td>14</td>
<td>20</td>
</tr>
<tr>
<td>7.5</td>
<td>Enroute</td>
<td>Low</td>
<td>4</td>
<td>no failure</td>
<td>14</td>
<td>a</td>
</tr>
<tr>
<td>7.6</td>
<td>Enroute</td>
<td>Low</td>
<td>5</td>
<td>3 m/s ramp</td>
<td>14</td>
<td>15</td>
</tr>
<tr>
<td>7.7</td>
<td>Enroute</td>
<td>Low</td>
<td>4</td>
<td>1 m/s ramp</td>
<td>42</td>
<td>a</td>
</tr>
<tr>
<td>7.8</td>
<td>Enroute</td>
<td>Low</td>
<td>5</td>
<td>1 m/s ramp</td>
<td>48</td>
<td>65</td>
</tr>
<tr>
<td>7.9</td>
<td>Enroute</td>
<td>Low</td>
<td>4</td>
<td>random walk</td>
<td>14</td>
<td>15</td>
</tr>
<tr>
<td>7.10</td>
<td>Enroute</td>
<td>Low</td>
<td>4</td>
<td>50 m step</td>
<td>5</td>
<td>6</td>
</tr>
<tr>
<td>7.11</td>
<td>Enroute</td>
<td>Low</td>
<td>4 with doppler (4 m/s error)</td>
<td>3 m/s ramp</td>
<td>14</td>
<td>15</td>
</tr>
<tr>
<td>7.12</td>
<td>Enroute</td>
<td>Low</td>
<td>4 with doppler (1 m/s error)</td>
<td>3 m/s ramp</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>7.13</td>
<td>NP Approach</td>
<td>High(8S)</td>
<td>4</td>
<td>3 m/s ramp</td>
<td>25</td>
<td>a</td>
</tr>
<tr>
<td>7.14</td>
<td>NP Approach</td>
<td>High(8S)</td>
<td>5</td>
<td>3 m/s ramp</td>
<td>15</td>
<td>28</td>
</tr>
<tr>
<td>7.15</td>
<td>NP Approach</td>
<td>High(8S)</td>
<td>4</td>
<td>1 m/s ramp</td>
<td>40</td>
<td>a</td>
</tr>
<tr>
<td>7.16</td>
<td>NP Approach</td>
<td>High(8S)</td>
<td>5</td>
<td>1 m/s ramp</td>
<td>48</td>
<td>110</td>
</tr>
<tr>
<td>7.17</td>
<td>NP Approach</td>
<td>High(8S)</td>
<td>5</td>
<td>no failure</td>
<td>1</td>
<td>b</td>
</tr>
<tr>
<td>7.18</td>
<td>NP Approach</td>
<td>High(11S)</td>
<td>5</td>
<td>no failure</td>
<td>1</td>
<td>a</td>
</tr>
<tr>
<td>7.19</td>
<td>NP Approach</td>
<td>High(11S)</td>
<td>4</td>
<td>3 m/s ramp</td>
<td>25</td>
<td>a</td>
</tr>
<tr>
<td>7.20</td>
<td>NP Approach</td>
<td>High(11S)</td>
<td>5</td>
<td>3 m/s ramp</td>
<td>13</td>
<td>32</td>
</tr>
</tbody>
</table>

*a No complete decision was reached.

*b A false alarm decision was made.
four-measurement model is quite inadequate. Also, to anticipate the incidence of accelerative maneuvers, an 11-state filter model is much more appropriate over the 8-state model, although a slight compromise in performance must be traded off. The fifth measurement in the baro-altimeter contributes valuable redundancy as evidenced by the results obtained in simulations using it. The doppler information also shows promise but is, of course, highly dependent on the error associated with it.
VIII. OTHER IMPLEMENTATION CONSIDERATIONS

The preceding chapters have covered the formulation and analysis of the performance of the Disjoint Hypotheses Model for failure detection. There are indeed more practical issues that need to be addressed before this scheme can be regarded as a viable option for monitoring GPS integrity. In this chapter, we will consider some of these issues.

A. The Observation Window and Decision Procedure

In real time, the decision on the detection of a failure is generally made after a period of time has been spent observing all available measurements. This span of time, either conditionally or unconditionally fixed, will be referred to as the observation (time) window. In generating the time profile of the posterior probabilities of the various hypotheses over this time window, the model makes the implicit assumption that the conditions of the process are stationary or unchanging over this span of time. Thus, whatever the choice of the window size is used to detect a failure, the solution yielded by this scheme is faithful to the model only if the presence or absence of a failure holds true for the entire observation window. In reality, however, the onset of a failure is unknown, and so it is most unlikely that it will occur coincidentally at the very start of an observation window.

In the event that it starts out somewhere in the middle of a window, the detection model, although now suboptimal as far as modeling the failure is concerned, might just be robust enough arrive at a correct decision. Under
these conditions, the model attempts to approximate the failure it sees with a failure signature it is restricted to hypothesize (Figure 8.1). As can be expected, the reliability of this inadequate model degrades quickly as that point of onset of failure occurs further away from the start of the window. When this happens, there is really no guarantee that the scheme can detect a failure during the allotted time interval. If the problem is reset and solved afresh, starting at where the previous window terminated, then the failure should be detectable in this new observation window. The correct decision can then be made but only after having being delayed by a time window. It does not take much though to appreciate that the size of the window must not
be too large. Yet, on the other hand, a window that is too small may not adequately gather enough information to yield a sound decision.

A possible solution is to circumvent such a tradeoff with overlapping windows (Figure 8.2). This alternative calls for solving the detection problem simultaneously over several staggered observation windows. As such, several detection schemes must be run in parallel at any one point in time. This may at first glance seem like quite a staggering proposition. However, when the commonality of the regular Kalman filter computations among the parallel schemes is considered, the extra complexities lie, for the most part, only in the generation of more likelihood functions.
With this structure, the selection of the window size and the stagger interval is at best a subjective determination because of the number of factors involved. In the simulated environments presented here, a reasonable choice of the window size and stagger interval would appear to be about 100 seconds and 10 seconds respectively.

B. Corrective Action Upon Failure Recognition

After the process of detection and identification, the user needs to take on some form of corrective action. The choice of action is simple when there are five or more measurement sources available. Since only four are actually needed to obtain a stable navigation solution, the bad measurement can be isolated and discarded with no undesirable consequences.

However, if only four are available to begin with, it becomes critical to reacquire another measurement source before the accuracy of the system deteriorates beyond safety limits. This may not always be necessary though. If the parameters of the failure can be estimated with reasonable accuracy, the bad measurement can still be used if the estimated failure signature is subtracted from it. Of course, this adjusted measurement will be "noisier" than it would normally have been, uncorrupted with the failure. Estimates of the failure parameters are available in the likelihood functions generated, namely the maximum likelihood estimates. Corrective action was not one of the major objectives of this investigation, so this will not be pursued further here.
C. Estimation of Computational Requirements

The data processing aspect is a critical consideration due to amount of information that must be handled. However, the processing load is perhaps not quite as unwieldy as it may first seem. The commonality that exists in the parallel structure proves invaluable in providing substantial savings since the bulk of the computational burden usually associated with the Kalman filter lies in the covariance and gain calculations.

Table 2 below summarizes the CPU times required to execute that portion of an algorithm which represents the processing of measurements observed over one second of real time.

Table 2. Comparison of CPU times equivalent to one second of real observation time for different algorithms

<table>
<thead>
<tr>
<th>Model</th>
<th>Measurements</th>
<th>Processing Time (seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low-dynamics</td>
<td>4</td>
<td>.07</td>
</tr>
<tr>
<td>Low-dynamics</td>
<td>5</td>
<td>.10</td>
</tr>
<tr>
<td>High-dynamics*</td>
<td>4</td>
<td>.22</td>
</tr>
<tr>
<td>High-dynamics*</td>
<td>5</td>
<td>.25</td>
</tr>
</tbody>
</table>

* The high-dynamics model was an 11-state process.

Although the actual values mean little since these algorithms were run on a
mainframe, these numbers reflect the relative sizes of the different algorithms used.

The smallest of these algorithms is the low-dynamics model with four measurements. When this algorithm was implemented on a standard IBM-compatible personal computer (Compaq Portable), a second's worth of real time observation data required about 17 seconds to execute on this machine. Obviously, this is too slow for real-time operation. There are, however, much faster microprocessor-based machines with special numeric processors that can improve processing speeds by 20 to 30 times over that of the standard PC. It must also be remembered that further savings can be realistically expected by using more optimized code.

A possible configuration for implementing the failure detection scheme is to have the software written in with the usual Kalman filter navigation algorithms. This would require the development of a customized GPS receiver. A perhaps more practical alternative is to use a regular C/A code GPS receiver and run the failure detection scheme in an external processing module.
IX. CONCLUSIONS

A. Summary of Findings

The results of this study shows that there is much promise in an autonomous failure detection scheme that utilizes fully the complement of resources of the Global Positioning System that is available to a user. Although all the demonstrations here have been in the form of simulations, realistically workable detection models were also generated in the process.

One of the earlier findings made in the comparison of the different models formulated was that the Disjoint Hypotheses Model with its parametrized likelihood functions was perhaps the best of those investigated. It is clearly the most efficient computationally.

In the enroute situation, the prompt detection and identification of a satellite failure has been demonstrated to be achievable even with only four satellite measurements. Of course, any aiding with other forms of measurements will only improve its performance. The results obtained were possible simply because the low dynamics of this flight environment and the clock stability were well exploited. Hence, if some standardized high-dynamics model is used instead for the enroute scenario, the performance is likely to suffer.

As for the nonprecision approach scenario, the detection of a failure seems to be achievable, but the ability to do prompt identification with just four measurements in this situation is more questionable. With aiding from a fifth measurement (baro-altitude in the simulation), the performance was
much improved. Also, an 11-state model that includes acceleration states was shown to be more robust in a high-acceleration turn maneuver than its 8-state counterpart that does not account for acceleration.

We have also seen that accurate doppler information can significantly improve the performance of the detection and the identification process.

B. Recommendations for Future Research

In this study, a suitable detection model was developed and tested out with Monte Carlo computer simulations. Eventually, this scheme will, of course, need to be tested with real-life data. Before this phase of the research is reached, however, there are several topics of interest suitable for further investigation that are worthy of mention here.

One of the concerns of the civil community at the present time is the problem of selective availability. This plan to degrade the accuracy available to C/A code users is bound to have an impact on the failure detection problem. The degree to which the performance of the detection scheme will be degraded in this situation is an important question that needs to be answered.

Another important consideration with respect to the worthiness of the detection scheme presented here is its robustness in unmodeled situations. Such situations might include encountering other types of failure modes other than the ramp, and the presence of unmodeled bias errors in general.

Also, more in-depth analyses can be performed on the statistical characteristics of the detection scheme as related to traditionally criteria such
as false alarm and detection rates.

Although it very much appears that the computational aspects of this scheme is manageable in a real-time situation when handling a reasonable amount of observation data, further characterization of the computational requirements prior to actual implementation is also a worthwhile pursuit.
X. REFERENCES


XI. ACKNOWLEDGEMENTS

This work was supported by the Federal Aviation Administration through the U.S. Department of Transportation's Transportation Systems Center. I would like to thank Dr. Rudolph M. Kalafus of the Center, and Keith D. McDonald of the FAA for their helpful guidance and suggestions offered throughout the course of the project.

To my wonderful parents, brother, and sister in Malaysia, for their constant love and support, and for paving my way to greater possibilities in life, I gratefully dedicate this piece of work.

I wish also to express my deepest appreciation to my mentor and friend, Professor Robert Grover Brown, who has, over the last four years, made a tremendous impact on my life. His guidance, confidence, encouragement, and example has made invaluable contributions to my personal growth throughout a most rewarding graduate career.

I would also like to extend my gratitude to the other members of my Ph.D. committee, Professors John P. Basart, William Q. Meeker, David T. Stephenson, and Allan G. Potter, for their contributions in overseeing my academic curriculum.

Special thanks are due also to Allen Daubendiek and David Garrelts for their gracious help in some benchmark testing with a personal computer.
A. Uncorrelated Measurement Residuals Affected by Failure

In this supplement, it will be shown that even with the presence of an additive deterministic "failure" term in the measurement, the measurement residuals remain uncorrelated.

In general, $\hat{\mathbf{x}}_k^-$ has a Gaussian distribution with a mean equal to $\mathbf{x}_k$ plus a "corruptive" term due to the failure, and a covariance matrix $\mathbf{P}_k^-:$

$$
\hat{\mathbf{x}}_k^- \sim \mathcal{N}(\mathbf{x}_k + \lambda_k, \mathbf{P}_k^-) \quad (A.1)
$$

$$
\Rightarrow \mathbf{x}_k^- \hat{\mathbf{x}}_k^- \sim \mathcal{N}(-\lambda_k, \mathbf{P}_k^-) \quad (A.2)
$$

The measurement residual $\mathbf{z}$ is given by:

$$
\mathbf{z}_k = \mathbf{z}_k + \mathbf{e}_k - \mathbf{H}_k \hat{\mathbf{x}}_k^- - \mathbf{c}_k + \mathbf{H}_k (\mathbf{x}_k^- \hat{\mathbf{x}}_k^-) + \mathbf{v}_k \quad (A.3)
$$

$$
\Rightarrow \mathbf{z}_k \sim \mathcal{N}(\mathbf{c}_k - \mathbf{H}_k \lambda_k, \mathbf{V}_k) \quad (A.4)
$$

The Kalman filter update equation in turn yields $\hat{\mathbf{x}}_k^+:$

$$
\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + \mathbf{K}_k [\mathbf{z}_k + \mathbf{e}_k - \mathbf{H}_k \hat{\mathbf{x}}_k^-]
- \mathbf{K}_k \mathbf{e}_k + \mathbf{K}_k \mathbf{H}_k \mathbf{x}_k + [1 - \mathbf{K}_k \mathbf{H}_k] \hat{\mathbf{x}}_k^- \quad (A.5)
$$
When the state is projected to the next step, we have:

\[
\begin{align*}
\tilde{z}^*_{k+1} &= \phi_k \tilde{z}_k^* \\
\tilde{z}_{k+1} &= N \left( x_{k+1} + \lambda_{k+1}, P_{k+1} \right)
\end{align*}
\]  

(A.7)

(A.8)

where

\[
x_{k+1} = \phi_k x_k + w_k
\]

and

\[
\lambda_{k+1} = \phi_k (\lambda_k + K_k e_k - K_k H_k \lambda_k)
\]

In order to show that the measurement residuals are uncorrelated, we need to show that \( \tilde{z}_k \) and \( \tilde{z}_{k+1} \) are uncorrelated:

\[
\text{Cov}(\tilde{z}_k, \tilde{z}_{k+1}) = \text{Cov}(\tilde{z}_{k+1}, \tilde{z}_k) = E[\tilde{z}_{k+1} \tilde{z}_k^T] - E[\tilde{z}_{k+1}] E[\tilde{z}_k] = 0
\]

(A.9)

\[
E[\tilde{z}_{k+1} \tilde{z}_k^T] = E[(H_{k+1}(\phi_k x_k + w_k) + v_{k+1} + c_{k+1} - H_{k+1} \hat{z}_{k+1}) \tilde{z}_k^T]
\]

\[
- E[(H_{k+1}(\phi_k x_k + w_k) + v_{k+1} + c_{k+1} - H_{k+1} \phi_k (\tilde{z}_k^* + K_k \tilde{z}_k)) \tilde{z}_k^T]
\]

\[
- H_{k+1}(\phi_k E[\tilde{z}_k^T] + H_{k+1} E[w_k \tilde{z}_k^T] + E[v_{k+1} \tilde{z}_k^T] + E[c_{k+1} \tilde{z}_k^T])
\]

\[
- H_{k+1}(\phi_k E[\tilde{z}_{k+1}^T] - H_{k+1} \phi_k K_k E[\tilde{z}_k \tilde{z}_k^T])
\]

(A.9)

\[
E[\tilde{z}_k^T] = [c_k - H_k \lambda_k]^T
\]

(A.10)
\( E \{w_k^T \} - E \{v_k^T \} = 0 \) \hspace{1cm} (A.11)

\[ E \{\xi_k^T - \xi_k^T \} = E \{\xi_k^T (H_k (x_k - \xi_k^T) + \epsilon_k + v_k^T)\} \]
\[ = (E \{\xi_k^T x_k^T - \xi_k^T \} )H_k^T + E \{\xi_k^T \epsilon_k^T\} + E \{\xi_k^T v_k^T\} \]
\[ = [(x_k + \lambda_k) x_k^T - P_k^T - (x_k + \lambda_k)(x_k + \lambda_k)^T] H_k^T + (x_k + \lambda_k) \epsilon_k^T \] \hspace{1cm} (A.12)

\[ E \{\xi_k^T \xi_k^T \} = V_k + [\epsilon_k - H_k \lambda_k] [\epsilon_k - H_k \lambda_k]^T \] \hspace{1cm} (A.13)

Substituting eqs. A.10-A.13 into eq. A.9, we have

\[ E \{\xi_{k-1}^T \xi_k^T \} = H_{k+1} \phi_k x_k [\epsilon_k - H_k \lambda_k]^T + \epsilon_{k+1} [\epsilon_k - H_k \lambda_k]^T \]
\[ - H_{k+1} \phi_k ([(x_k + \lambda_k) x_k^T - P_k^T - (x_k + \lambda_k)(x_k + \lambda_k)^T] H_k^T + (x_k + \lambda_k) \epsilon_k^T) \]
\[ - H_{k+1} \phi_k K_k (V_k + [\epsilon_k - H_k \lambda_k] [\epsilon_k - H_k \lambda_k]^T) \]
\[ - H_{k+1} \phi_k x_k \epsilon_k - H_{k+1} \phi_k x_k \lambda_k^T H_k^T + \epsilon_{k+1} [\epsilon_k - \epsilon_{k+1} \lambda_k^T H_k^T] \]
\[ - H_{k+1} \phi_k x_k \epsilon_k - H_{k+1} \phi_k \lambda_k x_k^T H_k^T - H_{k+1} \phi_k \lambda_k \epsilon_k^T \]
\[ + H_{k+1} \phi_k \lambda_k \epsilon_k^T - H_{k+1} \phi_k \lambda_k^T H_k^T \]
\[ - H_{k+1} \phi_k \lambda_k \epsilon_k^T - H_{k+1} \phi_k \lambda_k \epsilon_k^T \]
\[ = \epsilon_{k+1} [\epsilon_k - H_k \lambda_k]^T \] \hspace{1cm} (A.14)

\[ E \{\xi_{k+1} \} = \epsilon_{k+1} - H_{k+1} \lambda_k \]
\[ = \epsilon_{k+1} - H_{k+1} \phi_k (\lambda_k + K_k \epsilon_k - K_k H_k \lambda_k) \]
\[ 
\begin{align*}
\text{Equation A.14 equals eq. A.15} \Rightarrow \text{Cov} (\mathbf{Z}_{k+1}, \mathbf{Z}_k^T) &= 0 \\
\end{align*}
\]

B. Computer Program Listings

The computer programs given in this supplement were written in the WATFIV (Fortran IV) language which is JCL-based. The first few lines of each program listing are "cards" that belong to the JCL.

The first listing is that of a detection scheme consisting of an 8-state low-dynamics process and a 4-measurement model. Information pertaining to satellite and receiver positions are read in from a data file, SRSLGS (see line 4 of the JCL) in this case, which was prepared previously by a separate dynamical simulation program included last in this chapter.

The next listing is that of a detection scheme consisting of an 11-state high-dynamics process and a 3-measurement model. The dynamical simulation data are read from SRHGS here. Both these data processing programs print out, sequentially, the posterior probabilities associated with the various hypotheses.

The third and final program included here is the dynamical simulation program mentioned above. The random trajectory of an aircraft in flight as
well as the deterministic orbital trajectories of specific set of GPS satellites are simulated with this program, which then stores the position data generated into files such as SRLGS and SRHGS.
PROCESSOR: 8-STATE DISJOINT HYPOTHESES MODEL WITH A 4-PSEUDORANGE MEASUREMENT MODEL

DEFINITION OF VARIABLE NAMES:

STXYZ: X-Y-Z POSITIONS OF SATELLITES IN .25-SEC INCREMENT BLOCKS, EACH REPRESENTING FIVE SATELLITES IN SEQUENCE.

RCVR: X-Y-Z POSITIONS OF RECEIVER IN .25-SEC INCREMENTS

PROB: CONTAINS THE POSTERIOR PROBABILITIES CALCULATED FOR EACH OF THE FIVE HYPOTHESES OVER A 100-SEC SPAN OF TIME.

STM: STATE TRANSITION MATRIX FOR A .25-SEC PROJECTION

Q: COVARIANCE MATRIX OF THE PROCESS NOISE

PPR: STATE COVARIANCE MATRIX (A PRIORI)

H: LINEAR-CONNECTING MEASUREMENT MATRIX

HYPO: COMPOSITE LIKELIHOOD FUNCTION

PH: INTERMEDIATE MATRIX PRODUCT OF "PPR" AND "H"

GAIN: KALMAN GAIN VECTOR

PUP: UPDATED STATE COVARIANCE MATRIX

HX: INTERMEDIATE MATRIX PRODUCT OF "H" AND "CFO"

XLKHD: LIKELIHOOD WEIGHTING MATRIX

CFO: COEFFICIENT MATRIX FOR THE PARAMETER VECTOR

CFI: UPDATED VERSION OF CFO

XPR: UPDATED STATE ESTIMATE

XNOM: NOMINAL VALUE OF THE STATE VECTOR

RESDL: MEASUREMENT RESIDUAL VECTOR

KH: INTERMEDIATE MATRIX PRODUCT OF "GAIN" AND "H"

GAUSS: ARRAY OF VALUES OF THE STANDARD GAUSSIAN DISTRIBUTION FOR TABLE LOOKUP PURPOSES

IMPLICIT REAL*8 (A-H,0-Z)

DIMENSION STXYZ(3,200),RCVR(4,400),PROB(5,100)

DIMENSION STM(8,8),Q(8,8),PPR(8,8),H(4,8),HYPO(5)

DIMENSION PH(8),GAIN(8),PUP(8,8),HX(3),XLKHD(4,3,3)

DIMENSION CFO(4,4,3),CFI(8,3),XPR(8),XNOM(3),RESDL(3)

DIMENSION XL33(4),COPULL(4),COUNULL(4)

COMMON GAUSS(40,10)

READING STXYZ AND RCVR DATA ARRAYS FROM THE FILE SRLGS (SEE JCL)

READ (10) ((STXYZ(I,J),I=1,3),J=1,2000)

READ (10) ((RCVR(I,J),I=1,4),J=1,400)

INITIALIZING VALUES:
C RAMP : INITIAL VALUE OF FAILURE
C DT : TIME STEP
C ISTEPS : TOTAL NUMBER OF PROCESSING STEPS (ONE SECOND OF
OBSERVATION TIME
C IFAIL : FAILURE FLAG
C PCT : INITIAL WEIGHT OF NULL POSTERIOR PROBABILITY
C W01 : DIMENSION OF NULL REGION CORRESPONDING TO SLOPE
C W02 : DIMENSION OF NULL REGION CORRESPONDING TO BIAS
C SIG1 : 1-SIGMA OF PRIOR DISTRIBUTION OF LIKELIHOOD FUNCTION
C CORRESPONDING TO SLOPE
C SIG2 : 1-SIGMA OF PRIOR DISTRIBUTION OF LIKELIHOOD FUNCTION
C CORRESPONDING TO BIAS
C IDYN0 : DYNAMICS FLAG (0-LOW; 1-HIGH)
C ICLOCK : CLOCK QUALITY FLAG (0-POOR; 1-GOOD)
C SX,SY,SZ : SPECTRAL AMPLITUDES OF PROCESS NOISE FOR POS.-VEL.
C SG,SH : SPECTRAL AMPLITUDES OF PROCESS NOISE FOR CLOCK

C RAMP=0.0D0
DT=.25D0
ISTEPS=100
IFAIL=1
PCT=.2D0
W01=1.0D0
W02=1.0D0
SIG1=3.0D0
SIG2=4.0D0
PI=3.141592653589793D0
IDYN0=0
ICLOCK=1
IF (IDYN0.EQ.0) THEN
SX=1.0D-2
SY=1.0D-2
SZ=1.0D-2
ELSE
SX=1.0D0
SY=1.0D0
SZ=1.0D0
ENDIF
IF (ICLOCK.EQ.0) THEN
SG=4.7150-20
SH=7.5D-20
ELSE
SG=9.0D-22
SH=2.70427160590D-23
ENDIF

C READING THE INITIAL VALUES FOR "PPR" AND "XPR" FROM DATA SET
DO 35 I=1,8
DO 35 J=1,7,2
  READ(5,9) PPR(I,J),PPR(I,J+1)
35 CONTINUE
DO 45 I=1,7,2
  READ(5,9) XPR(I),XPR(I+1)
45 CONTINUE

C INITIALIZING THE COEFFICIENT MATRIX "CFO" AND THE LIKELIHOOD
WEIGHTING MATRIX "XLKHD"
DO 10 IFTR=1,4
  DO 10 I=1,8
  DO 10 J=1,8
146

0(J,J)=0.00
IF (J.GT.3) GO TO 10
IF (J.EQ.3) THEN
0(IFTR=1,J)=XPR(I)
ELSE
0(IFTR=1,J)=0.00
ENDIF
10 CONTINUE
DO 15 IFTR=1,4
DO 15 J=1,3
0(IFTR=1,J)=0.00
15 CONTINUE
C
C SETTING THE PRIOR DISTRIBUTION FOR THE LIKELIHOOD FUNCTION
XLKHD(IFTR=1,1)=1.00/SIG1/SIG1
XLKHD(IFTR=2,2)=1.00/SIG2/SIG2
13 CONTINUE
C
C "R" IS THE MEASUREMENT NOISE VARIANCE
R=20.00002
C
C INITIALIZING THE STATE TRANSITION MATRIX
DO 20 I=1,7,2
STM(I,1)=1.00
STM(1,1+1)=1.00
STM(I,1+1)=DT
STM(I+1,1)=0.00
20 CONTINUE
C
C SETTING UP THE STANDARD GAUSSIAN LOOKUP TABLE
DO 25 I=1,40
DO 25 J=1,6,5
READ(5,0) GAUS(I,J),GAUS(I,J+1),GAUS(I,J+2),
    + GAUS(I,J+3),GAUS(I,J+4)
25 CONTINUE
PP1=GAUS((W0N1/SIG1)-GAUS((W0N2/SIG2)
PP2=GAUS((W0N2/SIG2)-GAUS((W0N1/SIG1)
C
C COMPUTING THE WEIGHTING FACTOR FOR LIKELIHOOD FUNCTION
C ASSOCIATED WITH THE NULL REGION
WTNULL=PCT/(1.00-PCT)/PP1/PP2
C
C INITIALIZING THE PROCESS COVARIANCE MATRIX "O"
Q(1,1)=SX0DT003/3.00
Q(2,2)=SX0DT002/2.00
Q(2,1)=0(1,2)
Q(3,3)=SY0DT003/3.00
Q(4,4)=SY0DT002/2.00
Q(4,3)=0(3,4)
Q(5,5)=SZ0DT003/3.00
Q(6,6)=SZ0DT002/2.00
Q(6,5)=0(5,6)
Q(7,7)=SH0DT003/3.00
Q(8,8)=SH0DT002/2.00
Q(8,7)=Q(7,8)
INITIALIZING THE "H" MATRIX
DO 30 J=1,3
H(1,J)=1.4
H(2,J)=2.0
H(3,J)=0.2
30 CONTINUE

SEED1, SEED2 ARE SEEDS OF RANDOMIZATION FOR THE MEASUREMENT NOISE
SEED1=2445695584.0D0
SEED2=2235498392.0D0
BIG31=2.0000031
BIG32=2.0000032

OUTERMOST RECURSIVE TIME LOOP
DO 40 K=1,ISTEPS

SEQUENTIAL MEASUREMENT LOOP
DO 50 ISAT=1,4

GENERATING A GAUSSIAN-DISTRIBUTED RANDOM NOISE TERM
SEED1=DMOD(SEED1*65539.0D0,BIG32)
SEED2=DMOD(SEED2*262147.0D0,BIG32)
G1=4.6566126730-10*DSORT(4.0D0/LOG11.0D0/G1)
SEED1=DMOD(SEED1*65539.0D0,BIG32)
SEED2=DMOD(SEED2*262147.0D0,BIG32)
G2=4.6566126730-10*DSORT(2.0D0/LOG11.0D0/G1)
RN=DSORT(2.0D0/LOG11.0D0/G1)
GNSE=RN*DSORT(G1)

SETTING THE NOMINAL TERMS TO BE EQUAL TO THE BEST POSITION
XNOM(1)=CF0(4.1,13)
XNOM(2)=CF0(4.3,33)
XNOM(3)=CF0(4.5,53)
IS=400*(K-1)+ISAT
IR=400*(K-1)+ISAT

ALTERNATIVE SAMPLING SCHEME:
FAST SAMPLING (4 SIMULTANEOUS MEASUREMENTS .2E SECONDS APART)
IS=500*(K-1)+ISAT
IR=K

THE OBSERVED MEASUREMENT "AMEAS" IS CORRUPTED BY RANDOM NOISE "GNSE" AND THE FAILURE "RAMP" (OPTIONAL)
AMEAS=AMAGO+RCVR(4.1R)*299792500.0D0+20.0D0*GNSE
IF (ISAT.EQ.3.AND.IFAIL.EQ.1) THEN
RAMP=RAMP*3.0D0
AMEAS=AMEAS*RAMP
ENDIF

THE OBSERVED MEASUREMENT "AMEAS" IS CORRUPTED BY RANDOM NOISE "GNSE" AND THE FAILURE "RAMP" (OPTIONAL)
AMEAS=AMAGO+RCVR(4.1R)*299792500.0D0+20.0D0*GNSE
IF (ISAT.EQ.3.AND.IFAIL.EQ.1) THEN
RAMP=RAMP*3.0D0
AMEAS=AMEAS*RAMP
ENDIF
X1=STXYZ(1,IS)-XNOM(1)
Y1=STXYZ(2,IS)-XNOM(2)
Z1=STXYZ(3,IS)-XNOM(3)
AMAG1=DSORT(X1*X1+Y1*Y1+Z1*Z1)
H(ISAT,1)=-(X1/AMAG1)
H(ISAT,3)=-(Y1/AMAG1)
H(ISAT,3)=-(Z1/AMAG1)
DO 70 I=1,8
PH(I)=0.0D0
DO 70 J=1,A
   PH(I)=PH(I)+PPR(I,J)*PH(ISAT,J)
70 CONTINUE
V=0.0
DO 80 I=1,A
   V=V+PH(ISAT,I)*PH(I)
80 CONTINUE
V=V*R

COMPUTING THE KALMAN GAIN
DO 90 I=1,A
   G(A,I)=PH(I)/V
90 CONTINUE

THE FAILURE SOURCE HYPOTHESES LOOP
DO 60 IFTR=1,A
   IF (IFTR.EQ.ISAT) THEN
      YCF1=0.D0
      YCF2=1.0
   ELSE
      YCF1=0.D0
      YCF2=0.0
   ENDIF
   DO 100 I=1,3
      HX(I)=0.D0
   DO 100 J=1,7,2
      IF (I.EQ.3.AND.J.NE.7) THEN
         XINC=CF0(IFTR,J)*XMOV(J+1)*2
      ELSE
         XINC=CF0(IFTR,J)
      ENDIF
      HX(I)=HX(I)+H(IDAT,J)*XINC
100 CONTINUE

COMPUTING THE MEASUREMENT RESIDUALS
   RESDL(1)=YCF1-HX(1)
   RESDL(2)=YCF2-HX(2)
   RESDL(3)=AMAG1-HX(3)

UPDATING THE COEFFICIENT MATRIX
DO 110 I=1,A
   DO 110 J=1,3
      CF0(I,J)=CF0(IFTR,I,J)+G(A,I)*RESDL(J)
110 CONTINUE

UPDATING THE LIKELIHOOD WEIGHTING MATRIX
DO 150 I=1,3
   DO 150 J=1,3
      XLKHDI(IFTR,I,J)=XLKHDI(IFTR,I,J)+RESOL(I)*RESOL(J)/V
150 CONTINUE

PROJECTING THE COEFFICIENT MATRIX AHEAD
DO 140 I=1,A
   DO 140 J=1,7,2
      CF0(IFTR,J+1)=STM(J,J)*CF0(I,J)
140 CONTINUE
VCF1=0.0
VCF2=0.0

XLKHDI(1,3,3)=0.0
UPDATING THE STATE COVARIANCE MATRIX

DO 130 1=1,N
  DO 130 J=1,N
    PUP(I,J) = PUP(I,J) + STM(I,J) * PUP(I,J+1)
    PUP(I,J+1) = PUP(I,J) + STM(I,J+1) * PUP(I+1,J)
  END

PROJECTING THE STATE COVARIANCE MATRIX AHEAD

DO 50 1=1,2N,2
  DO 50 J=1,2N,2
    PPR(I,J) = PPR(I,J) + STM(I,J) * PPR(I,J+1)
    PPR(I,J+1) = PPR(I,J) + STM(I,J+1) * PPR(I+1,J)
  END

COMPUTING THE COMPOSITE LIKELIHOOD FUNCTIONS

TOTAL=0.D0
  HYPO(5)=0.D0
  DET=XLKHD(IFTR+1,1) + XLKHD(IFTR+2,2) - XLKHD(IFTR+1,2) + 2.D0
  XINV11=XLKHD(IFTR+2,2)/DET
  XINV12=XLKHD(IFTR+1,2)/DET
  XINV22=XLKHD(IFTR+1,1)/DET

XM1,XM2 ARE THE MAXIMUM LIKELIHOOD ESTIMATES OF THE PARAMETERS

XM1=XM110XKHD(IFTR+1,3) + XINV220XLKHD(IFTR+2,3)
XM2=XM110XKHD(IFTR+1,3) + XINV20XLKHD(IFTR+2,3)
XL33=XLKHD(IFTR+1,1) + XM102XKHD(IFTR+2,3) + 2.D0XXM10XM22
C  ROTATIONAL TRANSFORMATION ELEMENTS A11,A22,A12 ARE COMPUTED
A11=DSQRT (C+XLKHD(IFTR,1,1)*C+XLKHD(IFTR,1,2)*C12)
+ C120(XLKHD(IFTR,2,1)*C+XLKHD(IFTR,2,2)*C12))
A22=DSQRT (C+XLKHD(IFTR,1,1)*C12-XLKHD(IFTR,1,2)*C)
+ -C120(XLKHD(IFTR,2,1)*C12-XLKHD(IFTR,2,2)*C)
A12=-C0(XLKHD(IFTR,1,1)*C12-XLKHD(IFTR,1,2)*C)
+ -C120(XLKHD(IFTR,2,1)*C12-XLKHD(IFTR,2,2)*C)
UM1=-C*XM1-C12*XM2
UM2=C12*XM1-C*XM2
C TABLE LOOKUP IS USED HERE TO COMPUTE THE APPROXIMATE INTEGRAL OF
C THE LIKELIHOOD FUNCTION OVER THE NULL REGION
P1=GAUZ((UM1+WDW1)*A11)-GAUZ((UM1+WDW1)*A11)
P2=GAUZ((UM2+WDW2)*A22)-GAUZ((UM2+WDW2)*A22)
CONULL(IFTR)=P1*P2/(A11*A22)
180 CONTINUE
C  TABLE LOOKUP IS USED HERE TO COMPUTE THE APPROXIMATE INTEGRAL OF
C THE LIKELIHOOD FUNCTION OVER THE NULL REGION
C  THE POSTERIOR PROBABILITIES ARE COMPUTED
C  THE POSTERIOR PROBABILITIES FOR EACH HYPOTHESIS ARE PRINTED OUT
C  THIS SUBPROGRAM RETURNS THE CUMULATIVE PROBABILITY FOR A
C  STANDARD GAUSSIAN DISTRIBUTION (INTEGRATED FROM MINUS INF.
C TO Z)
FUNCTION GAUZ(Z)
IMPLICIT REAL08 (A-H,O-D
COMMON GAUSS(40,10)
NEG=0
IF (Z.GT.3.99D0) THEN
GAUZ=1.0D0
STOP
END
RETURN
ENDIF
IF (Z.LT.-3.99D0) THEN
GAUZ=0.00
RETURN
ENDIF
IF (Z.LT.0.00) THEN
Zz=-Z
NEG=1
ELSE
Zz=Z
ENDIF
I1 = INT(ZZ0.11)
I2=INT(ZZ0.22)·1.01
WT=ZZ0.22·DFLOAT(IDINT(ZZ0.112))
C1=GAU55(11+1.12+1)
IF (I2.EQ.0) THEN
C2=GAU55(I1+1.12+1)
ELSE
C2=GAU55(I1+1.12+2)
ENDIF
GAUZ=C1+0.00+MT1+C20WT
IF (NEG.EQ.1) GAUZ=1.00-GAUZ
RETURN
END

C
C THIS SUBROUTINE MULTIPIES "A" AND "B" TO GIVE "C" WHILE
C CHECKING FOR NUMERICAL OVERFLOW AND UNDERFLOW
SUBROUTINO CHEX(A,B,C)
IMPLICIT REALC(A-H.O-Z1
IF (A.NE.0.00) THEN
CA=LOG10(IDABS(A))
ELSE
CA=-1.00
ENDIF
IF (B.NE.0.00) THEN
CB=LOG10.IDABS(B))
ELSE
CB=-1.00
ENDIF
IF (CA+CB.LT.-70.DO) THEN
C=.00
ELSE
C=A*B
ENDIF
RETURN
END
ENTRY
0.47383209988543893D0 0.17530581488510700D0 0.17530581488510700D0 0.17530581488510700D0
-0.13622987562729760D0 0.86654723845978500D0 0.86654723845978500D0 0.86654723845978500D0
-0.3308692549192500D0 0.35097945367168800D0 0.35097945367168800D0 0.35097945367168800D0
-0.1075734911874500D0 0.35097945367168800D0 0.35097945367168800D0 0.35097945367168800D0
0.17530581488510720D0 0.18485460305016950D0 0.18485460305016950D0 0.18485460305016950D0
0.9997311397306770D0 0.6866216973063500D0 0.6866216973063500D0 0.6866216973063500D0
0.10572586138703960D0 0.1078858432341730D0 0.1078858432341730D0 0.1078858432341730D0
0.9997311397306770D0 0.1078858432341730D0 0.1078858432341730D0 0.1078858432341730D0
0.17823826625635480D0 0.17823826625635480D0 0.17823826625635480D0 0.17823826625635480D0
-0.13622987562729760D0 0.9997311397306770D0 0.9997311397306770D0 0.9997311397306770D0
-0.05010827624821560D0 0.28287612770646930D0 0.28287612770646930D0 0.28287612770646930D0
-0.38811163072305760D0 0.25342609785651690D0 0.25342609785651690D0 0.25342609785651690D0
-0.12574628688848390D0 0.39832954519029810D0 0.39832954519029810D0 0.39832954519029810D0
0.17530581488510700D0 0.17530581488510700D0 0.17530581488510700D0 0.17530581488510700D0
0.86654723845978500D0 0.86654723845978500D0 0.86654723845978500D0 0.86654723845978500D0
0.35097945367168800D0 0.35097945367168800D0 0.35097945367168800D0 0.35097945367168800D0
0.35097945367168800D0 0.35097945367168800D0 0.35097945367168800D0 0.35097945367168800D0
0.18485460305016950D0 0.18485460305016950D0 0.18485460305016950D0 0.18485460305016950D0
0.6866216973063500D0 0.6866216973063500D0 0.6866216973063500D0 0.6866216973063500D0
0.1078858432341730D0 0.1078858432341730D0 0.1078858432341730D0 0.1078858432341730D0
0.1078858432341730D0 0.1078858432341730D0 0.1078858432341730D0 0.1078858432341730D0
0.17823826625635480D0 0.17823826625635480D0 0.17823826625635480D0 0.17823826625635480D0
0.9997311397306770D0 0.9997311397306770D0 0.9997311397306770D0 0.9997311397306770D0
0.28287612770646930D0 0.28287612770646930D0 0.28287612770646930D0 0.28287612770646930D0
0.25342609785651690D0 0.25342609785651690D0 0.25342609785651690D0 0.25342609785651690D0
0.39832954519029810D0 0.39832954519029810D0 0.39832954519029810D0 0.39832954519029810D0
PROCESSOR: 11-STATE DISJOINT HYPOTHESES MODEL WITH
A 4-PSUEDORANGE/HARD-ALTITUDE MEASUREMENT MODEL

DEFINITION OF VARIABLE NAMES:

STXYZ ; X-Y-Z POSITIONS OF SATELLITES IN .25-SEC INCREMENT BLOCKS, EACH REPRESENTING FIVE SATELLITES IN SEQUENCE.
RCVR : X-Y-Z POSITIONS OF RECEIVER IN .25-SEC INCREMENTS
PROB : CONTAINS THE POSTERIOR PROBABILITIES CALCULATED FOR EACH OF THE FIVE HYPOTHESES OVER A 100-SEC SPAN OF TIME.
STM : STATE TRANSITION MATRIX FOR A .25-SEC PROJECTION
Q : COVARIANCE MATRIX OF THE PROCESS NOISE
PPR : STATE COVARIANCE MATRIX (A PRIORI)
H : LINEAR-CONNECTING MEASUREMENT MATRIX
HYP0 : COMPOSITE LIKELIHOOD FUNCTION
SIGR : ONE-SIGMA OF THE MEASUREMENT NOISE
PH : INTERMEDIATE MATRIX PRODUCT OF "PPR" AND "H"
GAIN : KALMAN GAIN VECTOR
PUP : UPDATED STATE COVARIANCE MATRIX
HX : INTERMEDIATE MATRIX PRODUCT OF "H" AND "CFO"
XLKHD : LIKELIHOOD WEIGHTING MATRIX
CFO : COEFFICIENT MATRIX FOR THE PARAMETER VECTOR
CI : UPDATED VERSION OF CFO
XPR : UPDATED STATE ESTIMATE
XNM : NOMINAL VALUE OF THE STATE VECTOR
RES_DL : MEASUREMENT RESIDUAL VECTOR
PSTM : INTERMEDIATE MATRIX PRODUCT OF "PUP" AND "STM"
KH : INTERMEDIATE MATRIX PRODUCT OF "GAIN" AND "H"
GAUSS : ARRAY OF VALUES OF THE STANDARD GAUSSIAN DISTRIBUTION FOR TABLE LOOKUP PURPOSES

C IMPLICIT REAL*8 (A-H,O-Z)
DIMENSION STXYZ(3,2000),RCVR(4,400),PPRD(6,100)
DIMENSION STM(12,12),Q(12,12),PPR(12,12),H(5,12),HYP0(6),SIGR(5)
DIMENSION PH(12),GAIN(12),PUP(12,12),HK(3),XLKHD(5,3,3)
DIMENSION CFO(5,12,3),CFO(12,3),XPR(11),XNM(3),RES_DL(3)
DIMENSION XLKHD(5),CFO(5),CONULL(5),PSTM(12,12)
REAL*8 KH(12,12)
COMMON GAUSS(40,10)
C
C READING STXYZ AND RCVR DATA ARRAYS FROM THE FILE "SRHGS" (SEE JCL)
READ (10) ((STXYZ(I,J),I=1,3),J=1,2000)
READ (10) ((RCVR(I,J),I=1,4),J=1,400)
C INITIALIZING VALUES:
C RAMP : INITIAL VALUE OF FAILURE
C DT : TIME STEP
C ISTEPS : TOTAL NUMBER OF PROCESSING STEPS (ONE SECOND OF OBSERVATION TIME
C IFAIL : FAILURE FLAG
C PCT : INITIAL WEIGHT OF NULL POSTERIOR PROBABILITY
C WDW1 : DIMENSION OF NULL REGION CORRESPONDING TO SLOPE
C WDW2 : DIMENSION OF NULL REGION CORRESPONDING TO BIAS
C SIG1 : 1-SIGMA OF PRIOR DISTRIBUTION OF LIKELIHOOD FUNCTION CORRESPONDING TO SLOPE
C SIG2 : 1-SIGMA OF PRIOR DISTRIBUTION OF LIKELIHOOD FUNCTION CORRESPONDING TO BIAS
C IDYN0 : DYNAMICS FLAG (0=LOW; 1=HIGH)
C ICLOCK : CLOCK QUALITY FLAG (0=POOR; 1=GOOD)
C IFAST : SAMPLING RATE FLAG (0=SEQUENTIAL; 1=SIMULTANEOUS)
C A : INVERSE OF THE MARKOV ACCELERATION TIME constant
C SX,SY,SZ : SPECTRAL AMPLITUDES OF PROCESS NOISE FOR POS.-VEL.
C S2X,S2Y,S2Z : SPECTRAL AMPLITUDES OF SECOND PROCESS NOISE INPUT
C SG,SH : SPECTRAL AMPLITUDES OF PROCESS NOISE FOR CLOCK
C
RAMP=0.0D0
DT=.250D0
ISTEPS=1
IFAIL=1
PCT=.1429D0
WDW1=1.0D0
WDW2=1.0D0
SIG1=3.0D0
SIG2=4.0D0
PI=3.14159265358979
IDYN0=1
ICLOCK=1
IFAST=0
A=.100
IF (IDYN0.EQ.0) THEN
   SX=1.0D-2
   SY=1.0D-2
   SZ=1.0D-2
   S2X=0.0D0
   S2Y=0.0D0
   S2Z=0.0D0
ELSE
   SX=1.0D-1
   SY=1.0D-1
   SZ=1.0D-1
   S2X=7.684D0
   S2Y=7.684D0
   S2Z=7.684D0
ENDIF
IF (ICLOCK.EQ.0) THEN
   SG=4.715D-20
   SH=7.5D-20
ELSE
   SG=9.0D-22
   SH=2.7042716059D-23
ENDIF
C C READING THE INITIAL VALUES FOR "PPR" AND "XPR" FROM DATA SET
DO 35 I=1,12
DO 35 J=1,10,3
READ(5,0) PPR(I,J), PPR(I,J+1), PPR(I,J+2)
35 CONTINUE
DO 45 I=1,10,3
IF (I.NE.10) THEN
READ(5,0) XPR(I), XPR(I+1), XPR(I+2)
ELSE
READ(5,0) XPR(I), XPR(I+1)
ENDIF
45 CONTINUE
C
C INITIALIZING THE COEFFICIENT MATRIX "CFO" AND THE LIKELIHOOD
C WEIGHTING MATRIX "XLKHD"
DO 10 I=1,12
DO 10 J=1,12
CFO(I,J)=0.00
DO 10 IFTR=1,5
IF (I.NE.12 .AND. J.LE.3) THEN
IF (J.EQ.3) THEN
CFO(IFTR,I,J)=XPR(I)
ELSE
CFO(IFTR,I,J)=0.00
ENDIF
ENDIF
10 CONTINUE
DO 13 IFTR=1,5
DO 13 1=1,3
DO 13 J=1,3
XLKHD(IFTR,1,J)=0.00
13 CONTINUE
C
C SETTING THE PRIOR DISTRIBUTION FOR THE LIKELIHOOD FUNCTION
XLKHD(1,1,1)=1.00/SIG1/SIG1
XLKHD(1,2,2)=1.00/SIG2/SIG2
13 CONTINUE
C
C SETTING THE MEASUREMENT NOISE SIGMA
SIGR(1)=20.00
SIGR(2)=20.00
SIGR(3)=20.00
SIGR(4)=20.00
SIGR(5)=30.00
C
C INITIALIZING THE STATE TRANSITION MATRIX
DO 20 I=1,10,3
STM(I,1)=1.00
STM(I,1+1)=1.00
STM(I,1+2)=0.00
IF (I.NE.10) THEN
STM(I+1,1)=1.00+(1.00+DEXP(-AODT))/A
STM(I+1,2)=1.00+(1.00+DEXP(-AODT))/A
STM(I+2,1)=DEXP(-AODT)
ELSE
STM(I+1,2)=0.00
STM(I+1,1)=0.00
STM(I+2,1)=0.00
20 CONTINUE
ENDIF
20 CONTINUE
C SETTING UP THE STANDARD GAUSSIAN LOOKUP TABLE
DO 25 I=1,40
   DO 25 J=1,6,5
      READ(I,J) GAUSS(I,J),GAUSS(I,J+1),GAUSS(I,J+2),
         GAUSS(I,J+3),GAUSS(I,J+4)
25 CONTINUE
PP1=GAUZ(WDV1/SIG1)-GAUZ(-WDV1/SIG1)
PP2=GAUZ(WDV2/SIG2)-GAUZ(-WDV2/SIG2)
C COMPUTING THE WEIGHTING FACTOR FOR LIKELIHOOD FUNCTION
C ASSOCIATED WITH THE NULL REGION
WTNULL=PCT/II.DO-PCT J/PP1/PP2
C INITIALIZING THE PROCESS COVARIANCE MATRIX "O"
QC11=DT003/3.DO/A/A-DT003/3.DO/004+4.DO/005
   +DO003/3.DO/006+DO004/3.DO/005
QC12=DT003/2.DO/A/A-2.DO004/3.DO/005
   -2.DO005/3.DO/006+DO004/3.DO/006
QC13=DO003/3.DO/A/A-2.DO004/3.DO/006+2.DO005/3.DO/006
QC21=DO003/2.DO/A/A-2.DO004/3.DO/006+2.DO005/3.DO/006
QC22=DO003/2.DO/A/A-2.DO004/3.DO/006+2.DO005/3.DO/006
QC23=DO003/2.DO/A/A-2.DO004/3.DO/006
QC31=DO003/3.DO/A/A-2.DO004/3.DO/006+2.DO005/3.DO/006
QC32=DO003/2.DO/A/A-2.DO004/3.DO/006+2.DO005/3.DO/006
QC33=DO003/2.DO/A/A-2.DO004/3.DO/006
C INITIALIZING THE "H" MATRIX
C SETTING UP THE STANDARD GAUSSIAN LOOKUP TABLE
DO 25 I=1,40
   DO 25 J=1,6,5
      READ(I,J) GAUSS(I,J),GAUSS(I,J+1),GAUSS(I,J+2),
         GAUSS(I,J+3),GAUSS(I,J+4)
25 CONTINUE
PP1=GAUZ(WDV1/SIG1)-GAUZ(-WDV1/SIG1)
PP2=GAUZ(WDV2/SIG2)-GAUZ(-WDV2/SIG2)
DO 30 I=1,5
   IF (I.EQ.5) THEN
      H(I,10)=290792500.D0
   ELSE
      H(I,10)=0.D0
   ENDIF
   DO 30 J=1,10,3
      H(I,J+1)=0.00
      H(I,J+2)=0.00
   30 CONTINUE

C
C SEE01.SEED2 APE SEEDS OF RANDOMIZATION FOR THE MEASUREMENT NOISE
SEED1=2445955550.D0
SEED2=2235498392.D0
BIG31=2.000031
BIG32=2.000032
C OUTERMOST RECURSIVE TIME LOOP
DO 40 K=1,ISTEPS
C SEQUENTIAL MEASUREMENT LOOP
DO 50 ISAT=1,5

C GENERATING A GAUSSIAN-DISTRIBUTED RANDOM NOISE TERM
SEED1=DMOD(SEED1665539.D0,BIG32)
SEED2=DMOD(SEED2665147.D0,BIG32)
G1=4.656612873D-10*DMOD(SEED1*BIG31)
SEED1=DMOD(SEED1665539.D0,BIG32)
SEED2=DMOD(SEED2665147.D0,BIG32)
G2=4.656612873D-10*DMOD(SEED1*BIG31)
RN=RANDNORM(2.000000)*BIG32
RN=RN*31
G1=31*G1

C SETTING THE NOMINAL TERMS TO BE EQUAL TO THE BEST POSITION
C ESTIMATE AVAILABLE FROM THE FILTER
XNOM(1)=CF0(4.1,3)
XNOM(2)=CF0(4.4,3)
XNOM(3)=CF0(4.7,3)
IF (IFAST.EQ.0) THEN
   JS=30*(K-1)+50*(ISAT-1)+ISAT
   IR=JS*(K-1)+ISAT
ELSE
   JS=50*(K-1)+ISAT
   IR=K
ENDIF
IF (ISAT.EQ.5) THEN
   JS=15-6
   IR=JS-1
ELSE
   JS=15-1
ENDIF
ENDIF

C SETTING UP BAROALTITUDE "RANGE" MEASUREMENT (NOT PSEUDORANGE)
IF (ISAT.EQ.5) THEN
   STX=0.D0
   STY=0.D0
   STZ=-6344000.D0
ELSE
   STX=STX+JS
   STY=STY+JS
ENDIF
C THE OBSERVED MEASUREMENT "AMEAS" IS CORRUPTED BY RANDOM NOISE "GNSE" AND THE FAILURE "RAMP" (OPTIONAL)

IF (ISAT.NE.5) THEN
  AMEAS=AMAG0*RCVR(4,1)*299792500.D0+SIGR(ISAT)*GNSE
ELSE
  AMEAS=AMAG0+SIGR(ISAT)*GNSE
ENDIF

C COMPUTING THE KALMAN GAIN
DO 90 I=1,12
    GAIN(I)=PHI(I)/V
90 CONTINUE

C THE FAILURE SOURCE HYPOTHESES LOOP
DO 60 IFTR=1,E
    IF (IFTR.EQ.ISAT) THEN
        YCF1=1.D0
        YCF2=0.D0
    ELSE
        YCF1=0.D0
        YCF2=0.D0
    ENDIF
    DO 100 I=1,3
        HX(I)=0.D0
    DO 100 J=1,10,3
        IF (J.EQ.3.AND.J.NE.10) THEN
            XINC=CF0(IFTR,I)-XNOM((J+2)/3)
        ELSE
            XINC=CF0(IFTR,I)
        ENDIF
        HX(I)=HX(I)+H(ISAT,J)*XINC
100 CONTINUE
C COMPUTING THE MEASUREMENT RESIDUALS

STZ=STX*Z(3+15)
ENDIF
X0=STX+RCVR(1,1)
Y0=STY+RCVR(2,1)
Z0=STZ+RCVR(3,1)
AMAG0=DSQR(T(X0*X0+Y0*Y0+Z0*Z0))
RESDL(1)=YCF1-HX(1)
RESDL(2)=YCF2-HX(2)
RESDL(3)=AMFAS-AMAG1-HX(3)

C UPDATING THE COEFFICIENT MATRIX
DO 110 I=1,11
   DO 110 J=1,3
      CF(I,J)=CF(IFTR,I,J)*GAIN(I)*RESDL(J)
   CONTINUE

C UPDATING THE LIKELIHOOD WEIGHTING MATRIX
DO 150 I=1,3
   DO 150 J=1,3
      XLKHD(IFTR,I,J)=XLKHD(IFTR,I,J)*RESDL(I)*RESDL(J)*V
   CONTINUE

C PROJECTING THE COEFFICIENT MATRIX AHEAD
IUP=0
IF (ISAT.LE.3.AND.ISAT.EQ.0) IUP=1
IF (ISAT.EQ.3) IUP=1
IF (IUP.EQ.1) THEN
   DO 180 I=1,3
      DO 180 J=1,3
         IF (I.EQ.1) CF(IFTR,J,I)=STMI(J,I)*CF(I,J)
         OR CF(IFTR,J,I)=STMI(J,I)*CF(I,J)
         ELSE
            DO 170 I=1,3
               DO 170 J=1,3
                  CF(IFTR,I,J)=CF(I,J)
               CONTINUE
            CONTINUE
         ENDIF
      CONTINUE
   CONTINUE
ELSE
   DO 120 I=1,12
      DO 120 J=1,12
         KHI(I,J)=-GAIN(I)*H(ISAT,J)
      CONTINUE
   CONTINUE
ENDIF

C UPDATING THE STATE COVARIANCE MATRIX
DO 130 I=1,12
   DO 130 J=1,12
      PUP(I,J)=PUP(I,J)+KHI(I,J)*PUP(J,J)
   CONTINUE
DO 130 JJ=1,12
   PUP(I,J)=PUP(I,J)*PUP(J,J)
CONTINUE
CONTINUE

PROJECTING THE STATE COVARIANCE MATRIX AHEAD

IF (IUP.EQ.1) THEN
  DO 140 I=1,10,3
  DO 140 J=1,10,3
  
  DO 122 II=1,3
  DO 122 JJ=1,3
  PSTM(II+JJ)=PSTM(II+JJ-1)+PUP(II+II-1,JI)*PSTM(JJ+JJ-1)
  ENDIF

  CONTINUE
  DO 124 II=1,3
  DO 124 JJ=1,3
  PPR(I+II-1,II-1)=PUP(I+II-1,II-1)/DET
  ENDIF

  CONTINUE
ENDIF

ELSE
  DO 55 I=1,12
  DO 55 J=1,12
  PPR(I,J)=PUP(I,J)
  ENDIF

  CONTINUE
ENDIF

COMPUTING THE COMPOSITE LIKELIHOOD FUNCTIONS

TOTAL=0.D0
  DO 160 IFTR=1,5
  DET=XLKHD(IFTR,1.1)*XLKHD(IFTR,2.2)-XLKHD(IFTR,1.2)**2
  XINVI1=XLKHD(IFTR,1.2)/DET
  XINVI2=XLKHD(IFTR,1.1)/DET

  XM1,XM2 ARE THE MAXIMUM LIKELIHOOD ESTIMATES OF THE PARAMETERS
  XM1=-XINV1*XLKHD(IFTR,1.1)-XINV2*XLKHD(IFTR,1.2)
  XM2=-XINV1*XLKHD(IFTR,1.2)-XINV2*XLKHD(IFTR,1.1)
  XL33(IFTR)=XLKHD(IFTR,1.1)*XM1**2+XLKHD(IFTR,2.2)*XM2**2
  +2.D0*XLKHD(IFTR,1.2)*XM1*XM2
  C=DSOR(T(1.D0+2.DO*DSOR(T(1.D0+2.DO*GSORT(1.D0-4.DO*CC)))/2.D0))
C ROTATIONAL TRANSFORMATION ELEMENTS A11, A22, A12 ARE COMPUTED
A11 = SQRT(C12 (XLKHD(IFTR,1,11) + XLKHD(IFTR,1,2) * C12)
+ C12 (XLKHD(IFTR,2,11) + XLKHD(IFTR,2,2) * C12))
A22 = SQRT((XLKHD(IFTR,1,11) - XLKHD(IFTR,1,2) * C12)
+ C12 (XLKHD(IFTR,2,11) - XLKHD(IFTR,2,2) * C12))
A12 = -C (XLKHD(IFTR,1,11) - XLKHD(IFTR,1,2) * C12)
= -C (XLKHD(IFTR,2,11) - XLKHD(IFTR,2,2) * C12)

C THE LIKELIHOOD FUNCTION OVER THE NULL REGION
P1 = GAUZ((UM1 + WDW1) / A11)
= GAUZ((UM1 + WDW1) / A11)
P2 = GAUZ((UM2 + WDW2) / A22)
= GAUZ((UM2 + WDW2) / A22)

C CONULL (IFTR) = P1 + P2
= P1 + P2

CONTINUE
CEIL = 0.0
DO 165 IFTR = 1.5
IF (XL33(IFTR) > CEIL) CEIL = XL33(IFTR)
165 CONTINUE
DO 175 IFTR = 1.5
IF (CEIL > 300.0) XL33(IFTR) = XL33(IFTR) - CEIL * 300.0
IF (XL33(IFTR) < -300.0) XL33(IFTR) = -300.0
CALL CHEXCHVPO(IFTR), DEXP(XL33(IFTR)/2.0), PBFULL
CALL CHEXCHVPO(IFTR), DEXP(XL33(IFTR)/2.0), PBNULL
HP0D(IFTR) = PBFULL - PBNULL
TOTAL = TOTAL + HP0D(IFTR)
175 CONTINUE

C POSTERIOR PROBABILITIES ARE COMPUTED
HP0D61 = HP0D61 + WINULL
TOTAL = TOTAL + HP0D61

CONTINUE
DO 190 IFTR = 1.5
CALL CHEXHYPO(IFTR), 1.0 / TOTAL, PROB(IFTR, K)
190 CONTINUE

CONTINUE
C THE POSTERIOR PROBABILITIES FOR EACH HYPOTHESIS ARE PRINTED OUT
DO 200 K = 1, 101, 2
WRITE(6,1) K, PROB(1,K), PROB(2,K), PROB(3,K), PROB(4,K),
+ PROB(5,K), PROB(6,K), PROB(7,K), PROB(8,K), PROB(9,K), PROB(10,K)
200 CONTINUE

FORMAT(14,6F6.3,14,6F6.3)
STOP
END

C THIS SUBPROGRAM RETURNS THE CUMULATIVE PROBABILITY FOR A
C STANDARD GAUSSIAN DISTRIBUTION (INTEGRATED FROM MINUS INF.
C TO Z)
FUNCTION GAUZ(Z)
IMPLICIT REAL*8 (A-H,O-Z)
COMMON GAUSS(40,10)
NEG = 0

1 IF (Z.GT.3.99D0) THEN
GAUZ = 1.00
RETURN
ENDIF

NEG = 1
1 IF (Z.LT.-3.99D0) THEN
GAUZ = 0.00
RETURN
ENDIF

1
IF (Z.LT.0.00) THEN
    ZZ = -Z
    NEG = 1
ELSE
    ZZ = Z
ENDIF

11 = IDINT(ZZ»I.011
12 = IDINT(ZZ»1.021-11»1.01
WT = ZZ>M.02-0FL0ATC1DINT«ZZ«1.02 I)

C1 = GAUSS(11+1.12+1)
IF (I2.EQ.0.9) THEN
    C2 = GAUSS(I1+1.12+2)
ELSE
    C2 = GAUSS(I1 + 1.12 + 2)
ENDIF

GAUZ = C1»«1.00-WT»+C2»WT
IF (NEG.EQ.1) GAUZ = 1.00-GAUZ
RETURN
END

C
C THIS SUBROUTINE MULTIPLES "A" AND "B" TO GIVE "C" WHILE
C CHECKING FOR NUMERICAL OVERFLOW AND UNDERFLOW
SUBROUTINE CHEX (A, B, C)
IMPLICIT REAL*8(A-H.0-Z)

IF (A.NE.0.00) THEN
    CA = 0LOG10(OABS(A))
ELSE
    CA = -1.0D10
ENDIF

IF (B.NE.0.00) THEN
    CB = DL0G10(DABS(B))
ELSE
    CB = -1.0D10
ENDIF

IF (CA+CB.LT.-70.00) THEN
    C = 0.00
ELSE
    C = A*B
ENDIF
RETURN
END

$ENTRY
  0.2390118D 03    0.7278441D 02    0.10291241D 00
  0.2971883D 02    0.69393223D 01    0.5175291D 00
  0.21755583D 02    0.69510969D 01    0.45949510D 00
  0.11379145D 07    0.681975910D 10    0.00000000D 00
  0.7273841D 02    0.35329246D 02    0.20997060D 00
  0.69053918D 01    0.21736136D 01    0.20899706D 00
  0.68865979D 01    0.24567529D 01    0.27416390D 00
-0.16651654D 09    -0.83765558D 12    0.00000000D 00
  0.10291241D 02    0.70701453D 01    0.29516229D 00
  0.47945011D 00    0.19966540D 00    0.23998835D 00
  0.67943381D 00    0.31332215D 00    0.44021614D 01
-0.35471476D 10    -0.27196620D 12    0.00000000D 00
  0.2971883D 02    0.69053918D 01    0.47945011D 00
  0.46431899D 03    0.12612696D 00    0.14986917D 02
-0.14046759D 02    -0.44424732D 01    0.16101616D 00
-0.14345724D 07    -0.11116459D 09    0.00000000D 00
  0.69393223D 01    0.21736136D 01    0.19966540D 00
  0.12612696D 00    0.52218107D 02    0.89449578D 01
<table>
<thead>
<tr>
<th>165</th>
</tr>
</thead>
<tbody>
<tr>
<td>9032000</td>
</tr>
<tr>
<td>9114000</td>
</tr>
<tr>
<td>9192400</td>
</tr>
<tr>
<td>9264700</td>
</tr>
<tr>
<td>9331900</td>
</tr>
<tr>
<td>9394300</td>
</tr>
<tr>
<td>9452000</td>
</tr>
<tr>
<td>9505300</td>
</tr>
<tr>
<td>9554300</td>
</tr>
<tr>
<td>9599400</td>
</tr>
<tr>
<td>9640700</td>
</tr>
<tr>
<td>9676400</td>
</tr>
<tr>
<td>9712800</td>
</tr>
<tr>
<td>9744100</td>
</tr>
<tr>
<td>9772500</td>
</tr>
<tr>
<td>9798200</td>
</tr>
<tr>
<td>9821400</td>
</tr>
<tr>
<td>9842200</td>
</tr>
<tr>
<td>9861000</td>
</tr>
<tr>
<td>9877800</td>
</tr>
<tr>
<td>9892800</td>
</tr>
<tr>
<td>9906100</td>
</tr>
<tr>
<td>9918000</td>
</tr>
<tr>
<td>9928600</td>
</tr>
<tr>
<td>9937900</td>
</tr>
<tr>
<td>9946100</td>
</tr>
<tr>
<td>9953400</td>
</tr>
<tr>
<td>9959800</td>
</tr>
<tr>
<td>9965300</td>
</tr>
<tr>
<td>9970200</td>
</tr>
<tr>
<td>9974400</td>
</tr>
<tr>
<td>9978100</td>
</tr>
<tr>
<td>9981300</td>
</tr>
<tr>
<td>9984100</td>
</tr>
<tr>
<td>9986500</td>
</tr>
<tr>
<td>9988600</td>
</tr>
<tr>
<td>9996300</td>
</tr>
<tr>
<td>99991800</td>
</tr>
<tr>
<td>99993100</td>
</tr>
<tr>
<td>99994200</td>
</tr>
<tr>
<td>99995200</td>
</tr>
<tr>
<td>99996000</td>
</tr>
<tr>
<td>99996600</td>
</tr>
<tr>
<td>99997300</td>
</tr>
<tr>
<td>99997700</td>
</tr>
<tr>
<td>99998100</td>
</tr>
<tr>
<td>99998500</td>
</tr>
<tr>
<td>99998900</td>
</tr>
<tr>
<td>99999100</td>
</tr>
<tr>
<td>99999300</td>
</tr>
<tr>
<td>99999400</td>
</tr>
<tr>
<td>99999500</td>
</tr>
<tr>
<td>99999600</td>
</tr>
<tr>
<td>99999700</td>
</tr>
</tbody>
</table>
DYNAMICAL SIMULATION FOR FIVE SATELLITES AND A MOVING RECEIVER

DEFINITION OF VARIABLE NAMES:

STXYZ : X-Y-Z POSITIONS OF SATELLITES IN 0.25-SEC INCREMENT BLOCKS, EACH REPRESENTING FIVE SATELLITES IN SEQUENCE.

RCVR : X-Y-Z POSITIONS OF RECEIVER IN 0.25-SEC INCREMENTS.

Q11, Q12, Q22 : ELEMENTS OF A DIAGONAL BLOCK OF THE PROCESS COVARIANCE MATRIX "Q".

A11, A12, A22 : ELEMENTS OF A LINEAR CONNECTION MATRIX WHICH SHAPES INDEPENDENT VARIATES INTO CORRELATED ONES ACCORDING.

XP : SYSTEM STATE VECTOR (3=POSITION, 3=VELOCITY, 2=CLOCK).

TARGET: TIME STEP.

IDYNO : DYNAMICS FLAG (0=LOW, 1=HIGH).

ICLOCK : CLOCK QUALITY FLAG (0=POOR, 1=GOOD).

SX, SY, SZ : SPECTRAL AMPLITUDES OF PROCESS NOISE FOR POS.-VEL.

SG, SH : SPECTRAL AMPLITUDES OF PROCESS NOISE FOR CLOCK.

All computations carried out in DOUBLE PRECISION.

IMPLICIT REAL*(A-H, O-Z)

DIMENSION STXYZ(3,4000), RCVR(4, 400)

DIMENSION Q11(4), Q12(4), Q22(4), A11(4), A12(4), A22(4)

DIMENSION XP(8), ALFA(4), THE(4)

DIMENSION TTX(4,4)

DT=0.2500

PI=3.1415926535897930

IDYNO=0

ICLOCK=1

IF (IDYNO.EQ.0) THEN
  SX=1.0, D-2
  SY=1.0, D-2
  SZ=1.0, D-2
ELSE
  SX=1.0, D0
  SY=1.0, D0
  SZ=1.0, D0
ENDIF

IF (ICLOCK.EQ.0) THEN
  SG=0.7150-20
  SH=7.50-20
ELSE
  SG=9.0, D-22
  SH=2.70427160590-23
ENDIF
SIMULATING THE AIRCRAFT DYNAMICS

INITIALIZING THE PROCESS COVARIANCE MATRIX "O"

\[
\begin{align*}
Q_{11}(1) &= 5xO0T03/3.00 \\
Q_{12}(1) &= 5xO0T02/2.00 \\
Q_{22}(1) &= 5xO0T03/3.00 \\
Q_{11}(2) &= 5xO0T03/3.00 \\
Q_{12}(2) &= 5xO0T02/2.00 \\
Q_{22}(2) &= 5xO0T03/3.00 \\
Q_{11}(3) &= 5xO0T03/3.00 \\
Q_{12}(3) &= 5xO0T02/2.00 \\
Q_{22}(3) &= 5xO0T03/3.00 \\
Q_{11}(4) &= 5xO0T03/3.00 \\
Q_{12}(4) &= 5xO0T02/2.00 \\
Q_{22}(4) &= 5xO0T03/3.00
\end{align*}
\]

COMPUTING A11, A12, A22 USING CHOLESKY'S DECOMPOSITION

\[
\begin{align*}
A_{11}(1) &= O11(1) - O12(1)*O22(I) \\
A_{12}(1) &= O12(1)/O22(I)
\end{align*}
\]

INITIALIZING THE STATE VECTOR

\[
\begin{align*}
& \text{DO} 15 \text{ I}=1,7 \\
& \text{READ} (5,9) \text{ XP}(I), \text{ XP}(I+1)
\end{align*}
\]

SEED1, SEED2 ARE SEEDS OF RANDOMIZATION FOR THE NOISE PROCESS

TO GENERATE A GAUSSIAN-DISTRIBUTED RANDOM SEQUENCE

\[
\begin{align*}
& \text{DO} 20 \text{ J}=1,800 \\
& \text{DO} 30 \text{ J}=1,4 \\
& \text{SEED1=DMOD(SEED1+565539.D0*BIG32)} \\
& \text{SEED2=DMOD(SEED2+262147.D0*BIG32)} \\
& \text{SCL1=4.656612783D-10*DMOD(SEED1*SEED2,BIG31)} \\
& \text{SEED1=DMOD(SEED1+565539.D0*BIG32)} \\
& \text{SEED2=DMOD(SEED2+262147.D0*BIG32)} \\
& \text{SCL2=4.656612783D-10*DMOD(SEED1*SEED2,BIG31)} \\
& \text{IF (J.LT.2) GO TO 30} \\
& \text{RN=DSORT(2.D0*DLOC(1.00/SCL1))} \\
& \text{GO=RN*DOSIN(2.D0*PI*SCL2)} \\
& \text{UI=AI1(I)*GO+AI2(J)*GI} \\
& \text{U2=AI2(J)*GI} \\
& \text{ISUB=20*(J-1)+1} \\
\end{align*}
\]

THE VECTOR MARKOV DIFFERENCE EQUATIONS

\[
\begin{align*}
& \text{XP(ISUB)=XP(ISUB)+GO*XP(ISUB+1)+UI} \\
& \text{XP(ISUB+1)=XP(ISUB)+U2} \\
\end{align*}
\]

\[
\begin{align*}
& \text{DO} 30 \text{ J}=1,4 \\
& \text{SEED1=DMOD(SEED1+565539.D0*BIG32)} \\
& \text{SEED2=DMOD(SEED2+262147.D0*BIG32)} \\
& \text{SCL1=4.656612783D-10*DMOD(SEED1*SEED2,BIG31)} \\
& \text{SEED1=DMOD(SEED1+565539.D0*BIG32)} \\
& \text{SEED2=DMOD(SEED2+262147.D0*BIG32)} \\
& \text{SCL2=4.656612783D-10*DMOD(SEED1*SEED2,BIG31)} \\
& \text{IF (J.LT.2) GO TO 30} \\
& \text{RN=DSORT(2.D0*DLOC(1.00/SCL1))} \\
& \text{GO=RN*DOSIN(2.D0*PI*SCL2)} \\
& \text{UI=AI1(I)*GO+AI2(J)*GI} \\
& \text{U2=AI2(J)*GI} \\
& \text{ISUB=20*(J-1)+1} \\
\end{align*}
\]

THE VECTOR MARKOV DIFFERENCE EQUATIONS

\[
\begin{align*}
& \text{XP(ISUB)=XP(ISUB)+GO*XP(ISUB+1)+UI} \\
& \text{XP(ISUB+1)=XP(ISUB)+U2} \\
\end{align*}
\]

IF (J.LT.2) GO TO 20

RCVR(1,1)=XP(1)
RCVR(2,1)=XP(3)
RCVR(3,1)=XP(5)
RCVR(4,1)=XP(7)
SIMULATING THE GPS SATELLITES

C

C INITIAL ANGLES FOR SATELLITES: 1) SV5 2) SV7 3) SV10 4) SV15

C

RADIUS=26661000.00
ALFAO(1)=160.00D0/180.00
ALFAO(2)=80.00D0/180.00
ALFAO(3)=120.00D0/180.00
ALFAO(4)=40.00D0/180.00
THETO(1)=-60.00D0/180.00
THETO(2)=-120.00D0/180.00
THETO(3)=-180.00D0/180.00
THETO(4)=-240.00D0/180.00
GEOX=4.21B0960D70SCOS(100.00D0/180.00)
GEOY=-4.21B0960D70DS1NI100.00D0/180.00
GEOZ=TTX(1,1)*GEOX+TTX(1,2)*GEOY+TTX(1,3)*GEOZ
GEOSTX=TTX(1,1)*GEOX+TTX(1,2)*GEOY+TTX(1,3)*GEOZ
GEOSTY=TTX(1,1)*GEOX+TTX(1,2)*GEOY+TTX(1,3)*GEOZ
GEOSTZ=TTX(1,1)*GEOX+TTX(1,2)*GEOY+TTX(1,3)*GEOZ

DU 40 IT=1,800
DO 40 ISAT=1,5
ISUB=(IT-1)10**ISAT
IF (ISAT.EQ.5) THEN
  STXYZ(1.ISUB)=GEOSTX
  STXYZ(2.ISUB)=GEOSTY
  STXYZ(3.ISUB)=GEOSTZ
ELSE
  TT=DFLOAT(ISAT/12T0.2500
  ALFA=ALFAO(ISAT)+TT/12T0.00D0
  THETO=THETO(ISAT)+TT/239.333333333000D0
  GX=DCOS(ALFA)DCOS(THETO)
  GY=DCOS(ALFA)DCOS(THETO)
  GZ=DCOS(ALFA)DCOS(THETO)
  GX=RA01US*(GX*DS1N(ALFA)DCOS(INTHETO)DS1N(THETO)DS1N(35.00D0/180.00))
  GY=DS1N(ALFA)DCOS(INTHETO)DS1N(THETO)DS1N(35.00D0/180.00))
  GZ=DS1N(ALFA)DCOS(INTHETO)DS1N(THETO)DS1N(35.00D0/180.00))
  STXYZ(1.ISUB)=TTX(I,1)*GX+TTX(I,2)*GY+TTX(I,3)*GZ+TTX(I,4)
  STXYZ(2.ISUB)=TTX(I,1)*GX+TTX(I,2)*GY+TTX(I,3)*GZ+TTX(I,4)
  STXYZ(3.ISUB)=TTX(I,1)*GX+TTX(I,2)*GY+TTX(I,3)*GZ+TTX(I,4)
ENDIF

C WRITING INTO A DATA FILE "SRLGS" (SEE JCL)
WRITE(10) ((STXYZ(1.J),J=1,400)
WRITE(10) ((PCVR(I,J),I=1,4),J=1,800)
STOP
END

C THIS SUBROUTINE COMPUTES THE LINEAR TRANSFORMATION MATRIX NEEDED
C TO CONVERT AN EARTH-CENTERED EARTH- FIXED COORDINATE FRAME SUCH
C AS THE WGS-72 TO A LOCALLY-LEVEL FRAME CENTERED AT A POINT
C 3900 08MM 60SS NORTH LATITUDE AND 77DD 12MM 49SS WEST LONGITUDE

C SUBROUTINE XFM(TTX)
IMPLICIT REAL*8 (A-H.O-Z)
DIMENSION ATX(4,4),BFX(4,4),CTX(4,4),DTX(4,4,4),STX(4,4),XTX(4,4,4)
REAL*8 NE2,NTHETA,NRHO,NPHICtNX,NY,NZ
PI=3.141592653589793D0
NE=6.768658E-3
WES=6.694310^3
NTHETA=(282.0+47.0/60.0+10.712530/3600.0)/0.01/180.0
NPHI=(90.0-10.0+8.0/60.0+1.22940/3600.0)/0.01/180.0
NRHO=6378135.00
HT=137.78600
NPHI=CPI/2.00+NP1
1.
OEN=DSOR(WH00-E20DSINV(NPHIC))/0.02
1.
OEN=DSOR(WH00-E20DSINV(NPHIC))/0.02
NX=(NRHO/DEHT)*DCOS(NPHIC)/OSIN(NTHETA)
NY=(NRHO/DEHT)*DCOS(NPHIC)/OSIN(NTHETA)
NZ=(NRHO/DEHT)*DCOS(NPHIC)/OSIN(NTHETA)
NDT(N1,1)=1.000000020
1.
NDT(N2,2)=1.000000020
1.
NDT(N3,3)=1.000000020
1.
NDT(N4,4)=1.000000020
1.
NDT(N1,3)=1.000000020
1.
NDT(N2,3)=1.000000020
1.
NDT(N3,2)=1.000000020
1.
NDT(N4,2)=1.000000020
1.
NDT(N1,4)=1.000000020
1.
NDT(N2,4)=1.000000020
1.
NX=2.00+NX+DTX(1.2)*NY+DTX(1.3)*NZ+DTX(1.4)
NY=2.00+NX+DTX(2.2)*NY+DTX(2.3)*NZ+DTX(2.4)
NZ=2.00+NX+DTX(3.2)*NY+DTX(3.3)*NZ+DTX(3.4)
NWR=DSOR(WH00-E20W002)
ST=WX/WR
CT=WX/W0
W0Z=WRO/2.00
DO 10 I=1,10
FZ=2.00-WE2)*NZ/W0Z-1.00)
FZ=FZ-2.00+WX)*NZ/W0Z-1.00)
FZ=FZ-2.00+WX)*NZ/W0Z-1.00)
10 CONTINUE
10 CONTINUE
W0=DSOR(WH00-E20W002)
W0=DSOR(WH00-E20W002)
10 CONTINUE
10 CONTINUE
10 CONTINUE
CTX(2,3)=SP
DO 20 J=1,4
CTX(I,J)=0
DO 20 J=1,4
STX(I,J)=CTX(I,J)+CTX(I,2)*CTX(I,1)+CTX(I,3)*CTX(1,2)
20 CONTINUE
DO 30 I=1,4
DO 30 J=1,4
TTX(I,J)=CTX(I,J)+CTX(I,2)*CTX(I,1)+CTX(I,3)*CTX(3,2)
30 CONTINUE
RETURN
END

*ENTRY
-0.794005030100000000 02
-0.923534977285621400 00
-0.9908607800000000 03
0.180226370034060000 01
0.895776403225000000 01
0.895776403225000000 04
0.194507759274000000 08
-0.2627004298556620 00