OPTIMAL KALMAN FILTER INTEGRATION OF A GLOBAL POSITIONING SYSTEM RECEIVER AND AN LN-94 INERTIAL NAVIGATION SYSTEM

THESIS

James Lawrence Hirning
Air Force Institute of Technology/ENG

AFIT/GE/ENG/90S-02

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY
AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/GE/ENG/90S-02

OPTIMAL KALMAN FILTER INTEGRATION OF A GLOBAL POSITIONING SYSTEM RECEIVER AND AN LN-94 INERTIAL NAVIGATION SYSTEM

THESIS

James Lawrence Hirning
Air Force Institute of Technology/ENG

AFIT/GE/ENG/90S-02

AFIT/GE/ENG/90S-02

OPTIMAL KALMAN FILTER INTEGRATION OF A GLOBAL POSITIONING SYSTEM RECEIVER AND AN LN-94 INERTIAL NAVIGATION SYSTEM

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
In Partial Fulfillment of the
Requirements for the Degree of
Master of Science in Electrical Engineering

James Lawrence Hirning, B.S.E.E.
Air Force Institute of Technology/ENG

September, 1990

Preface

The integration of GPS and INS allows for more precise position and velocity information than is available with either system alone. This thesis investigates the possible performance improvement with GPS/INS integration. This research demonstrates that improvement, and shows attempts to provide verification with empirical data.

The most important tool in this research is the Mutimode Simulation for Optimal Filter Evaluation (MSOFE) software package developed by the Avionics Laboratory at Wright-Patterson AFB, OH. This tool was the major resource used to verify the truth models, predict filter performance, and implement the filter with empirical data.

I would like to thank Lt Col Lewantowicz for his constant push to obtain better results. His help in analyzing the data and results was instrumental to my understanding the problems I was facing. I would also like to thank my thesis committee members, Dr. Maybeck and Capt Paschall, for their help with Kalman filtering concepts and their help in obtaining a better look at the problems being encountered.

The students in the 1989-1990 navigation sequence receive my thanks for forcing me to understand the concepts being used and the many uses to which MSOFE can be put. My thanks go out to Capt Britt Snodgrass and Capt Gregory Johnson for their help with MSOFE and PROFGEN, and Capt Barbara Niblett for her support and help in proofreading and other tasks necessary to the completion of this document.

Finally, I would like to thank my family whose constant support throughout this effort was essential to its completion.

James Lawrence Hirning

Table of Contents

List of Figures

List of Tables

Abstract

This research develops and attempts to implement a Kalman filter integration of a Phase III Global Positioning System (GPS) five-channel receiver and an LN-94 Inertial Navigation System (INS). GPS provides highly accurate position and velocity information in low dynamic environments. An INS provides position and velocity information with lower accuracy over long periods of time, but it is highly responsive in dynamic maneuvers or at high frequencies. The INS has the added advantage of requiring no signals external to the vehicle to function. The integration of these two systems provides more precise information under a wider variety of situations.

A truth model for the INS is verified. A GPS error model is developed and combined with the INS model to provide GPS-aided-INS navigation. This model is used to predict baseline performance of a full-ordered filter. Attempts are made to utilize the filter with empirical data. The data is analyzed, and suggestions are made about ways to account for the errors in evidence. Results to date are presented and analyzed.

OPTIMAL KALMAN FILTER INTEGRATION OF A GLOBAL POSITIONING SYSTEM RECEIVER AND AN LN-94 INERTIAL NAVIGATION SYSTEM

I. Introduction

1.1 Background

The Global Positioning System (GPS) is a navigation system which provides highly accurate position and velocity estimates to the user (5:144). This system consists of three segments: space, control, and user. The space segment consists of 24 satellites in six orbital planes which receive information from the control segment and transmit satellite orbital information to the user segment. The control segment monitors satellites and performs updates when necessary. The user equipment receives signals from at least four different satellites and computes user position and velocity which are provided to the user. For GPS to maintain lock on the signals, even in a jamming environment, requires that the bandwidth of the receiver be quite narrow. However, GPS performance is degraded during high dynamic maneuvers due to the narrow bandwidth (6:2).

An Inertial Navigation System (INS) provides information about user position and velocity without external measurements. The INS can provide the same type of information as GPS without external signals but with a lesser degree of long term accuracy. Also, the vertical channel in the INS is unstable. Errors in the INS gyroscopes and accelerometers cause a degradation of the unit's performance. The errors grow slowly with time. As a result, the INS can provide highly accurate

position and velocity data for short periods of time. This high frequency response allows the INS to continue providing information in highly dynamic environments (6:3).

The integration of these two systems provides many advantages. The first of these advantages is to maintain high accuracy position and velocity information during all phases of flight. Other potential benefits of this integration include an increase in anti-jamming performance of the GPS User Equipment (GPSUE), aiding the reacquisition of satellite signals after interruptions, and an increase in ability to track satellites either with adaptive tracking techniques or with steerable beam antennas (5:144-145).

1.2 Research Objective

The objective of this research is to integrate an LN-94 INS and a GPS receiver using a single Kalman filter. A Kalman filter is a computer algorithm which estimates the errors in the systems from which it receives measurement inputs. The results of this integration are to be used as a truth model baseline from which to compare a two-Kalman-filter integration of the LN-94 INS and GPS receiver and reduced order filters based on the truth model. The two-filter integration scheme is dicussed in Section 1.3.

1.3 Current GPS/INS Integration Techniques

Two methods of integrating GPS and INS have been suggested. One proposed method uses one Kalman filter internal to the GPSUE and a second Kalman filter containing the INS and GPS filtered output (6:1). This configuration is illustrated in Figure 1.1 This approach has received much study because of current configuration of GPSUE for Air Force aircraft. The second method proposes to integrate the systems optimally through the use of a single Kalman filter (5:145). This approach uses the raw pseudorange information and is generally accepted as the analytically better way

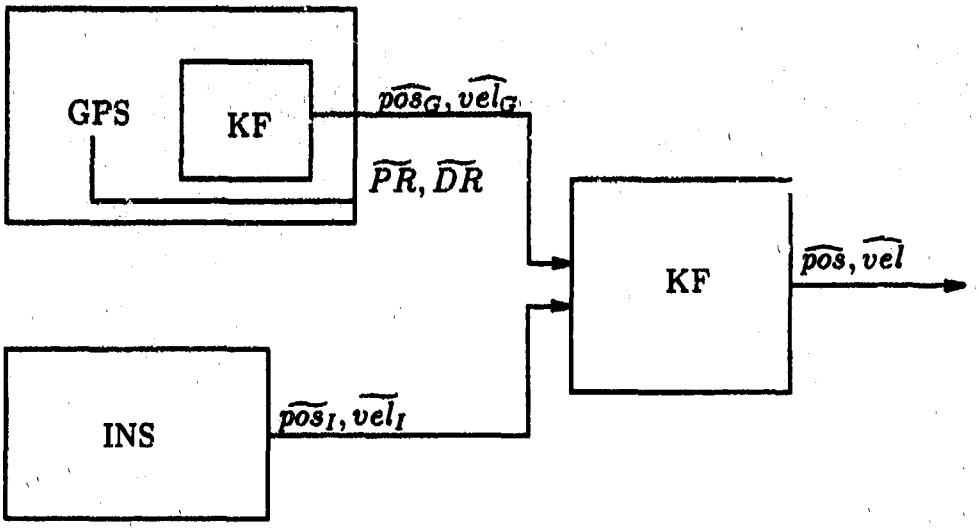


Figure 1.1. Two Filter GPS/INS Integration Scheme

to integrate the two systems. All information shared between equipment onboard Air Force aircraft is transmitted over the military standard 1553 digital bus (MIL-STD-1553). Phase III GPS receivers are not programmed to provide the necessary information (raw pseudorange and delta-range) on the MIL-STD-1553 bus. Hence, the necessity to study the first method of integrating INS and GPS equipment.

1.2.1 ***Two Filter Implementation*** Plessey Avionics, and others, have already implemented the two-filter configuration (8:117). Plessey Avionics actually used three different integrating filters, each for a different dynamic mode of operation. They implemented a number of corrections in all of their filters, including a correction for the lever arm from the antenna to the GPSUE (8:120-121). The main intent of their research was to explore improvement in anti-jamming performance of the GPS receiver using INS aiding. Under highly dynamic maneuvers, the position error growth was nearly zero even with only two satellites being tracked (8:122-123).

One major drawback to the two-filter implementation is a degree of instability during highly dynamic maneuvers (6:1). This instability is attributed to two major sources: the vertical channel instability of the INS and the lack of correlation

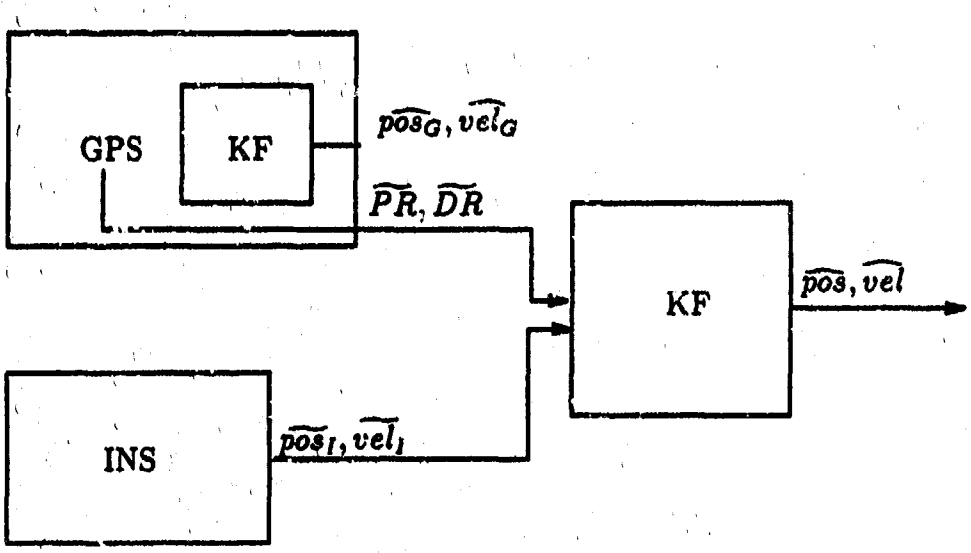


Figure 1.2. External Single Kalman Filter GPS/INS Integration Scheme

information between the two systems. The INS has an inherent vertical channel instability and must be aided by altitude information typically provided by a barometric altimeter, but this device does not totally remove the instability. Also, the information shared by the two Kalman filters may be highly correlated. With the separation of the GPS and INS models, information about this correlation is lost. This contributes to the instability of the overall system (6:4).

1.3.2 *Single Filter Implementation* Cox discusses an approach for the implementation of the single filter configuration (5:145-146). This configuration is shown in Figure 1.2. Utilizing Cox's approach, Texas Instruments has implemented a similar system using their own GPS receiver and INS (23:1). They also used a lever arm correction in their system. Their integration of the two systems provided an improvement in the accuracy of position and velocity data. The position error was maintained below ten meters and the estimated velocity was within 50 millimeters per second of the actual velocity. A test of the receiver was performed to guide an aircraft in a terrain following maneuver. The data from this test also showed a marked improvement over either GPS or INS alone (23:8).

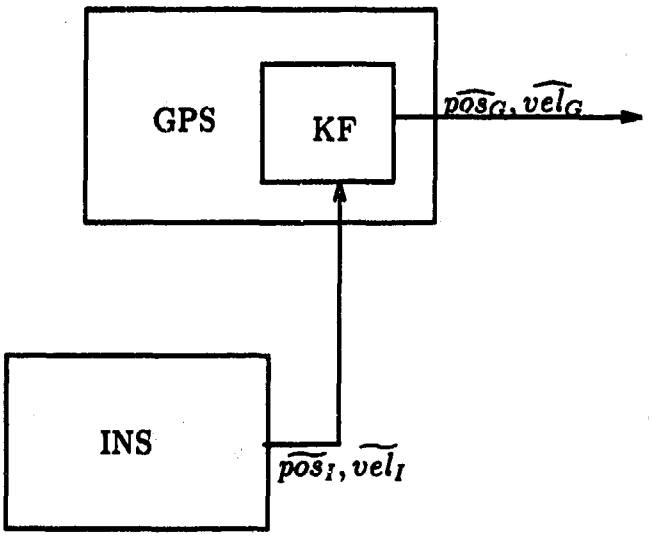


Figure 1.3. Internal Single Kalman Filter GPS/INS Integration Scheme

A second approach to the single filter configuration was implemented by The Charles Stark Draper Laboratory, Inc. utilizing a Rockwell-Collins Phase II GPS receiver (22:124). Here, the internal filter of the GPSUE was used to process all the information provided by the GPS receiver and the INS. This implementation is illustrated in Figure 1.3. The GPSUE filter has very crude approximations in its filter models to allow for its use with a number of different INS units. With a Phase II receiver, it is possible to change the noise strength models to model a particular INS more accurately (22:124). Draper Labs used many different noise strengths to tune the filter for their INS. With the proper tuning, the position error was reduced by 29 percent, and a reduction in velocity error by a factor of five was obtained (22:129).

The single filter configuration of the GPS and INS integration is attractive but is difficult to implement due to current equipment configurations. The GPS does not have the ability to output the necessary range and range-rate information onto the military standard 1553 (MIL-STD-1553) Data Bus. This problem severely limits the options available to perform the integration. If the internal filter is to be used, it would have to be tuned for each different type of Air Force INS. This problem is not insurmountable, but it may be difficult to overcome because of the number of

different INS units currently in use.

1.4 Specific Research Objectives

This research emphasizes the development and implementation of a single Kalman filter method for integrating GPS and INS. Key milestones are developing and validating the LN-94 truth model, combining the GPS truth model with the LN-94 model, developing a Kalman filter for the total system, and implementing and validating the Kalman filter on the Sun workstation. Specifics of the research objectives are given below.

1.4.1 LN-94 Truth Model The LN-94 truth model is developed using information provided by Litton (14) and is validated using the Multimode Simulation for Optimal Filter Evaluation (MSOFE) software developed at the Air Force Wright Aeronautical Laboratories (AFWAL) (4). Results of a covariance propagation simulation are compared to the results from a Monte Carlo analysis performed by Litton.

1.4.2 GPS Truth Model The GPS truth model is added to the LN-94 truth model on MSOFE. The GPS model was developed by Capt Joseph K. Solomon (20). This truth model has not been validated against a known standard. Hence, it is expected that some tuning will be necessary when this model is used as a filter with empirical data. The two truth models are combined, and an MSOFE covariance analysis is performed to predict performance of an operational system.

1.4.3 Kalman Filter Development and Implementation The Kalman filter is developed on a VAX 8650, and it is implemented on the same VAX 8650 and on a Sun workstation. A GPS receiver and LN-94 INS are optimally integrated using a MIL-STD-1553 data bus and an RS-422 port installed on the Sun. The 1553 data bus is used primarily for receiving information from the INS. The RS-422 port is utilized to obtain data from the GPS receiver. This requires writing software to

allow the Sun to collect the data through the RS-422 port.

1.4.4 Kalman Filter Validation Validation of the Kalman filter is performed by operating the filter on the Sun workstation using empirical data. The results are compared to the MSOFE performance prediction.

1.5 Summary

A brief discussion is provided about GPS and INS. Different methods of integration of GPS and INS as implemented by three different groups are described. Then, the specific objectives for this research are discussed.

1.6 Overview

1.6.1 Chapter 2 Chapter 2 provides the theory necessary to this research. Reference frames, coordinate transformations, communication with equipment, and Kalman filtering are among the topics presented.

1.6.2 Chapter 3 Chapter 3 discusses the development and verification of the truth model used for the INS Kalman filter. The results of two different navigation simulations are compared to information provided by Litton Guidance and Control Systems.

1.6.3 Chapter 4 Chapter 4 considers the design of the GPS truth model and its integration with the INS truth model. A baseline simulation with which to compare filter performance with empirical data is presented.

1.6.4 Chapter 5 Chapter 5 delves into empirical data collection and filter operation with empirical data. Problems with filter operation and attempts to correct for unexpected errors are presented.

1.6.5 *Chapter 6* Chapter 6 presents conclusions and recommendations. Conclusions from the information presented and recommendations for further study are discussed.

II. Theory

2.1 Nomenclature Conventions

Different type styles are used throughout the text to identify scalars, vectors, and matrices. Normal and italics text are used for scalar variables (i.e. X_n). Vectors are identified with lowercase boldface text (i.e. $\mathbf{x}$). Uppercase boldface is utilized for matrices (i.e. $\mathbf{C}_b^t$). A superscript on a boldface character ($\mathbf{x}^n$) normally denotes the reference frame in which that variable is expressed. However, a superscript T can be used to mean the transpose of a vector or matrix, or the t may refer to the true reference frame, described later. To differentiate between these two superscripts, on full vectors or matrices, a boldface lowercase superscript ($\mathbf{x}^t$) indicates a reference frame, and a normal or italics uppercase superscript ($\mathbf{x}^T$) indicates the vector or matrix transpose.

2.2 Reference Frames

The LN-94 INS works in six different reference frames: earth-centered earth-fixed (ECEF), navigation, true, computer, platform, and body. The GPSUE uses two frames of reference: earth-centered earth-fixed and navigation. In order to model the two systems properly, it is important to understand the reference frames used.

2.2.1 Earth-Centered Earth-Fixed Frame The earth-centered earth-fixed frame has its origin at the center of the earth. The Litton ECEF frame (X_l, Y_l, Z_l) is different from the ECEF frame utilized by GPSUE (X_g, Y_g, Z_g). These frames are illustrated in Figure 2.1. In the Litton frame, the Y_l axis points along the earth's spin axis. Litton's Z_l axis points towards the intersection of the Prime (Greenwich) Meridian and the equator. The X_l axis in Litton's ECEF frame is rotated 90 degrees east from the Z_l axis along the equatorial plane to complete a right-handed coordinate system. The ECEF frame used by GPSUE has its axes

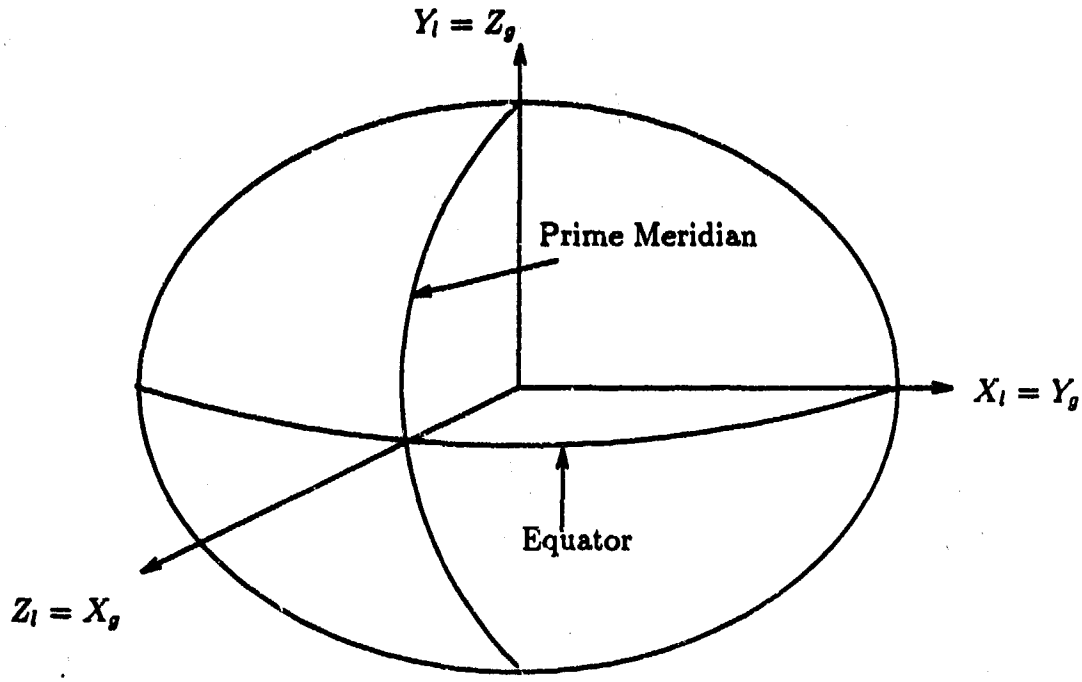


Figure 2.1. Litton and GPS ECEF Frames.

aligned with those of the Litton ECEF frame. However, the GPSUE ECEF frame has its X_g axis aligned with the Z_l axis. The Y_g axis is aligned with the X_l axis, and the Z_g axis is aligned with Litton's Y_l axis.

2.2.2 *Navigation Frame* The navigation frames (X_n, Y_n, Z_n) used by the two systems are the same. The axes are oriented to implement the East, North, Up frame. In this frame, the X_n axis points east, the Y_n axis points north, and the Z_n axis points in the direction of the local vertical. For the GPSUE, the origin of the frame is located at the antenna. The origin of the INS's navigation frame is defined within the INS.

2.2.3 *True Frame* The true frame (X_t, Y_t, Z_t) is a wander azimuth reference frame. The true frame is rotated counterclockwise about the Z_n axis by a wander angle α_t which varies with time and aircraft position. When $\alpha_t = 0$, the true frame is aligned with the navigation frame.

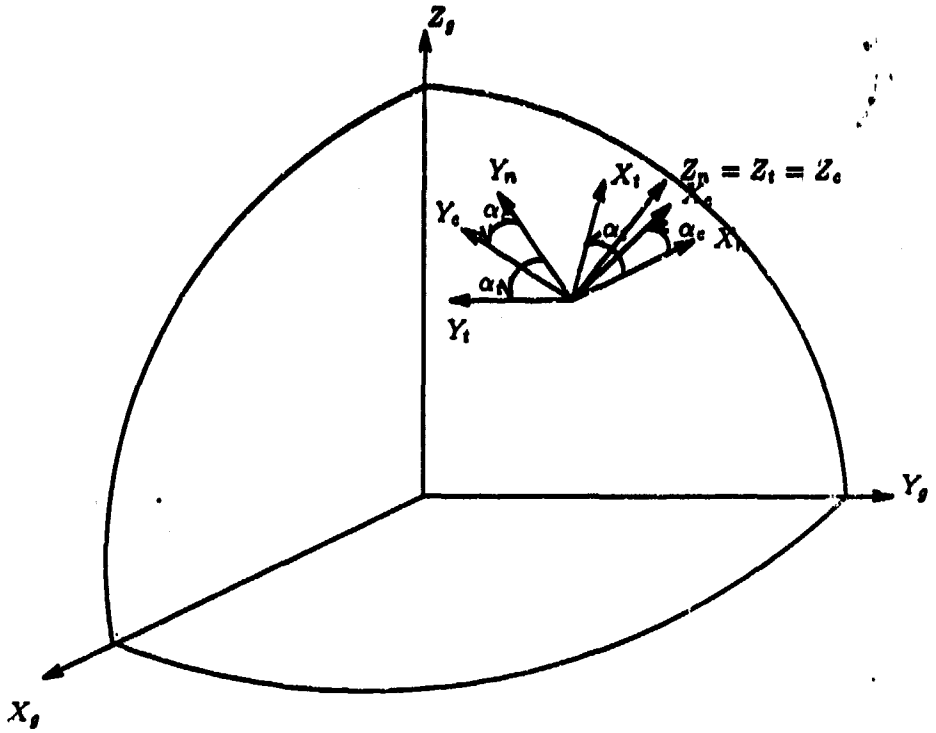


Figure 2.2. Navigation, True, and Computer Frames.

2.2.4 *Computer Frame* The computer frame (X_c, Y_c, Z_c) is a wander azimuth frame also and is the frame which the computer actually implements. The computer frame is rotated by an angle of α_c about the Z_n axis. When $\alpha_c = 0$, the computer frame is aligned with the navigation frame. The computer wander angle, α_c, differs from α_t because the computer frame is the frame which the INS is actually implementing; whereas, the true frame is the frame which the INS is attempting to implement. With no errors, the computer and true frames are aligned, and α_c is the same as α_t. The navigation, true, and computer frames are shown in Figure 2.2.

2.2.5 *Platform Frame* The platform frame has its origin at the INS and is misaligned from the true frame by three small attitude error angles ($\delta\phi_x$, $\delta\phi_y$, $\delta\phi_z$). When the attitude error angles are all zero, the platform is aligned with the true frame.

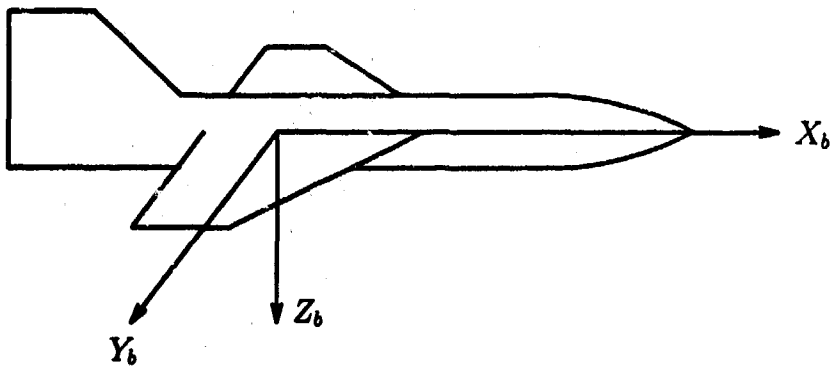


Figure 2.3. Body Frame.

2.2.6 *Body Frame* The body frame (X_b, Y_b, Z_b) has its origin at the aircraft center of mass. The X_b axis is parallel to the fuselage of the aircraft and points towards the nose of the aircraft. The Y_b axis is parallel to the right wing of the aircraft. The Z_b axis points through the floor of the aircraft to complete the right handed coordinate system. The body frame is illustrated in Figure 2.3.

2.3 Coordinate Transformations

Coordinate transformations are required throughout the INS and GPS error model. Calculations with vectors require that all the vectors be expressed in the same frame. Since not all vectors are expressed in the same frame, it is necessary to transform them into a common frame.

2.3.1 *Vector Representations* Vectors can be expressed in any of the reference frames described earlier. Vector notations used in this text and their associated frames are shown below.

$\mathbf{x}^e$ Vector in ECEF frame

$\mathbf{x}^n$ Vector in navigation frame

$\mathbf{x}^t$ Vector in true frame

$\mathbf{x^c}$ Vector in computer frame

$\mathbf{x^p}$ Vector in platform frame

$\mathbf{x^b}$ Vector in body frame

2.3.2 *Attitude Error Angles* The true, computer, and platform frames, implement nearly the same frame. The platform frame, as discussed earlier, is misaligned from the true frame by small attitude error angles ($\delta\phi_x$, $\delta\phi_y$, $\delta\phi_z$). The relationship between these two frames is:

$$\mathbf{x^p} = [\mathbf{I} + \delta\mathbf{\Phi}]\mathbf{x^t} \tag{2.1}$$

where

$$\delta\mathbf{\Phi} = \begin{bmatrix} 0 & \delta\phi_z & -\delta\phi_y \\ -\delta\phi_z & 0 & \delta\phi_x \\ \delta\phi_y & -\delta\phi_x & 0 \end{bmatrix} \tag{2.2}$$

The computer frame is also misaligned from the true frame by small attitude angle errors ($\delta\theta_x$, $\delta\theta_y$, $\delta\theta_z$). The relationship between the computer and true frames is:

$$\mathbf{x^c} = [\mathbf{I} + \delta\mathbf{\Theta}]\mathbf{x^t} \tag{2.3}$$

where

$$\delta\mathbf{\Theta} = \begin{bmatrix} 0 & \delta\theta_z & -\delta\theta_y \\ -\delta\theta_z & 0 & \delta\theta_x \\ \delta\theta_y & -\delta\theta_x & 0 \end{bmatrix} \tag{2.4}$$

2.3.3 *Direction Cosine Matrices* Vectors expressed in the ECEF, navigation, true, computer, or body frame can be transformed to a different frame by use of direction cosine matrices. These matrices allow transformations of vectors between frames which are related to each other through axial rotations. Direction cosine matrices (DCMs) are given in the form $\mathbf{C_n^e}$. The subscript indicates the frame in

which the vector is given, and the superscript indicates the frame into which the vector is transformed. The direction cosine matrices between Litton's ECEF and navigation, navigation and true, and true and body frames are shown below.

$$\mathbf{x}^{\mathbf{l}} = \mathbf{C}^{\mathbf{l}}_{\mathbf{n}}\mathbf{x}^{\mathbf{n}} \tag{2.5}$$

where

$$\mathbf{C}^{\mathbf{l}}_{\mathbf{n}} = \begin{bmatrix} \cos\lambda & -\sin\lambda\sin\Phi & \sin\lambda\cos\Phi \\ 0 & \cos\Phi & \sin\Phi \\ -\sin\lambda & -\cos\lambda\sin\Phi & \cos\lambda\cos\Phi \end{bmatrix} \tag{2.6}$$

and

λ = local terrestrial longitude

Φ = local latitude

The DCM for the transformation from the true to navigation reference frame is:

$$\mathbf{x}^{\mathbf{n}} = \mathbf{C}^{\mathbf{n}}_{\mathbf{t}}\mathbf{x}^{\mathbf{t}} \tag{2.7}$$

where

$$\mathbf{C}^{\mathbf{n}}_{\mathbf{t}} = \begin{bmatrix} \cos\alpha_t & -\sin\alpha_t & 0 \\ \sin\alpha_t & \cos\alpha_t & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.8}$$

The DCM for the true to body frame is obtained from Specification for USAF-15 Inertial Navigation Set (1) and is presented here:

$$\mathbf{x}^{\mathbf{b}} = \mathbf{C}^{\mathbf{b}}_{\mathbf{t}}\mathbf{x}^{\mathbf{t}} \tag{2.9}$$

where

$$C_t^b = \begin{bmatrix} \cos\vartheta\sin\psi & \cos\vartheta\cos\psi & \sin\vartheta \\ \sin\varphi\sin\vartheta\sin\psi + \cos\varphi\cos\psi & \sin\varphi\sin\vartheta\cos\psi - \cos\varphi\sin\psi & -\sin\varphi\cos\vartheta \\ \cos\varphi\sin\vartheta\sin\psi - \sin\varphi\cos\psi & \cos\varphi\sin\vartheta\cos\psi + \sin\varphi\cos\psi & -\cos\varphi\cos\vartheta \end{bmatrix} \tag{2.10}$$

and

$$\begin{aligned} \varphi &= \text{roll} \\ \vartheta &= \text{pitch} \\ \psi &= \text{yaw} \end{aligned}$$

2.4 INS/GPS Integrated Laboratory

The purpose of the integrated laboratory in the Department of Electrical and Computer Engineering at the Air Force Institute of Technology (AFIT) is to allow students to test their filter designs using empirical data. The INS is the Litton LN-94, and the GPSUE being utilized is the Rockwell-Collins Receiver 3A.

2.4.1 Hardware The LN-94 is a strapdown INS. It has three accelerometers and three ring laser gyros mounted on the INS platform, which has no motion relative to the aircraft except for vibrations transmitted by the vibration isolator mount. A gyro and accelerometer along each of the three platform axes measure rotations about and accelerations along the respective bodyframe axes.

The GPS Receiver 3A is a five-channel receiver. Four equations, having the three position components and time as unknowns, are used to determine receiver position. Solving these equations requires range information from four satellites. The satellites transmit signals at two frequencies. The L1 frequency is 1575.42 MHz, and the L2 frequency is 1227.6 MHz (7:2). Four of the receiver's channels are used to receive and decode L1 signals from four separate satellites. The fifth channel may do three things. It may receive signals from a fifth satellite, search for other satellites

to obtain alternative constellations, or receive L2 signals from one of the satellites currently being used by the other channels to help estimate the ionospheric delay.

2.4.2 INS Error Mechanization Equations Terrestrial navigation using inertial sensors involves the measurement of specific forces in three mutually orthogonal axes. In the LN-94 this is accomplished by accelcrometers mounted in such a way as to measure specific force in the body frame. Measurements from the gyros are used by the internal computer to transform the specific force vector to the true frame. The gravity force vector is computed using a gravity field model and is removed from the measurements, and the measurements are compensated for Coriolis and centripetal accelerations. The resulting acceleration vector is integrated with the appropriate initial conditions to obtain earth referenced velocity. The nonlinear velocity vector differential equation has the form (9:13-1):

$$\dot{\mathbf{v}} = \mathbf{a} + \mathbf{c} - \gamma \tag{2.11}$$

where

$\mathbf{a}$ = Measured specific force

$\mathbf{c}$ = Coriolis and centripetal acceleration

γ = Gravity vector

The velocity is transformed to the ECEF frame and is integrated to determine the system's position in terms of latitude (Φ), longitude (λ), wander angle (α_t), and altitude (h). The nonlinear equations have the form:

$$\dot{\Phi} = -\omega^n_{ip_x} \tag{2.12}$$

$$\dot{\lambda} = \omega^n_{ip_y} \sec \Phi \tag{2.13}$$

$$\dot{\alpha_t} = \omega^n_{ip_z} - \omega^n_{ip_y} \tan \Phi \tag{2.14}$$

$$\dot{h} = V^n_z \tag{2.15}$$

The nonlinear Equations (2.11) through (2.15) are combined to form a nonlinear vector differential equation. These equations are supplied for analysis purposes only. They can be expanded in a Taylor series to determine the error characteristics of the system; however, this is not done here. The expansion is performed about a nominal point and truncated to first order. The nominal linearization point generally used in navigation applications is the INS data corrected by Kalman filter estimated errors. The equations for the nominal point are:

$$\hat{\Phi} = \Phi_{ins} + \delta\hat{\Phi} \tag{2.16}$$

$$\hat{\lambda} = \lambda_{ins} + \delta\hat{\lambda} \tag{2.17}$$

$$\hat{h} = h_{ins} + \delta\hat{h} \tag{2.18}$$

$$\hat{\alpha_t} = \alpha_{t_{ins}} + \delta\hat{\alpha_t} \tag{2.19}$$

Engineers at Litton augmented the basic error states with error sources specific to the LN-94 to create a 93-state truth model. This model is presented in Chapter 3, Appendix C, and Litton CDRL No. 1002 (14).

2.4.3 *GPS Error Mechanization Equations* The GPS satellites transmit two codes on the L1 frequncy: coarse acquisition (CA) and precision (P) codes. The coarse acquistion code is utilized by the receiver to obtain lock on the signal. Then, the receiver switches to decoding the P-code signal. The P-code is a pseudo-random signal which repeats every 267 days (18:6). Each satellite sends a different portion

of the code. The receiver determines which satellite has been found by the portion of the code being sent. The signal is time tagged by the satellite, and the range to the satellite (R) is calculated by:

$$R = c\Delta t = c(t_{satellite} - t_{receiver}) \tag{2.20}$$

where

c = vacuum speed of light

This range is termed the pseudorange because it has not been corrected for errors. The main errors in the pseudorange are user clock error, code loop error, and atmospheric and ionospheric delay. The error equations used were developed by Capt Joseph Solomon in a special study for EENG699 at AFIT (20). The development of the error equations is presented later.

2.5 Equipment Communications

Operationally, the filter operates with empirical data. This requires communication with the INS and GPS receiver to obtain the data. Communication with both units is accomplished using the MIL-STD-1553 bus. Additional information needed for this research is obtained from the receiver through the RS-422 instrumentation port.

2.5.1 MIL-STD-1553 Bus The MIL-STD-1553 bus is the only communication link between avionics equipment onboard the aircraft. Communication on the bus is controlled by a bus controller, which issues commands to each remote terminal (RT) to transmit or receive information (12:I-35,I-36).

The data that an RT may receive or transmit is programmed in the units. Each set of data has a subaddress number associated with it that tells the RT what data is contained in the set. Because the LN-94 and GPS receiver are both designated as

remote terminals, only pre-specified types of information may be obtained by use of the MIL-STD-1553 data bus.

Here, the main use of the MIL-STD-1553 is to obtain information from the INS. The information necessary for input to the filter are position, velocity, linear acceleration vectors, attitude and attitude rates. This information is directly available on the MIL-STD-1553 bus.

2.5.2 Instrumentation Port In contrast, none of the information necessary for the GPS filter is available on the MIL-STD-1553. The only method for obtaining the "raw" pseudorange and delta-range data is through the instrumentation port on the GPSUE.

Information available on the instrumentation port has also been predefined in data sets called blocks. Operation of the instrumentation port is much like that of the MIL-STD-1553, but only two devices can be connected using the required RS-422 communication protocol. Intermetrics Incorporated has written a software package called PC Buffer Box which runs on an IBM PC-AT through which it is possible to obtain the information necessary for the filter.

2.6 Kalman Filter Equations

A Kalman filter is utilized to estimate the errors committed by the LN-94 and GPS Receiver 3A. The error states for the INS are those developed by Litton (14). The error states for the GPS receiver were developed by Capt. Solomon, as stated previously (20).

The Kalman filter equations are implemented using a simulation and analysis software package called MSOFE (4) which is discussed later. The form of the differential and measurement update equations used by MSOFE are described here.

The stochastic differential equations have the form:

$$\dot{\delta\mathbf{x}}(t) = \mathbf{F}(t)\delta\mathbf{x}(t) + \mathbf{G}(t)\mathbf{w}(t) \tag{2.21}$$

$$E\{\mathbf{w}(t)\} = \mathbf{0} \tag{2.22}$$

$$E\{\mathbf{w}(t)\mathbf{w}^T(t+\tau)\} = \mathbf{Q}(t)\delta(\tau) \tag{2.23}$$

where the function $E\{\}$ is used to mean the expected value of the argument inside the braces. The states in the INS truth model are error states. Thus, they are referenced in the equations as $\delta\mathbf{x}$. The error states are defined as the difference between the actual and INS measured states.

The error state vector $\delta\mathbf{x}$ and its associated covariance matrix $\mathbf{P}$ are propagated forward in time between measurements by integrating the equations (16:275):

$$\dot{\hat{\delta\mathbf{x}}}(t|t_{i-1}) = \mathbf{F}(t)\hat{\delta\mathbf{x}}(t|t_{i-1}) \tag{2.24}$$

$$\dot{\mathbf{P}}(t|t_{i-1}) = \mathbf{F}(t)\mathbf{P}(t|t_{i-1}) + \mathbf{P}(t|t_{i-1})\mathbf{F}^T(t) + \mathbf{G}(t)\mathbf{Q}(t)\mathbf{G}^T(t) \tag{2.25}$$

The "hat" ($\hat{\delta\mathbf{x}}$) indicates that the quantity is an estimate given by the filter. In addition, the term $\dot{\hat{\delta x}}(t|t_{i-1})$ indicates the estimate of $\dot{\delta\mathbf{x}}$ at time t given knowledge available about $\dot{\hat{\delta x}}(t|t_{i-1})$ through time t_{i-1}. These equations are integrated from time t_0 using the initial conditions:

$$\hat{\delta\mathbf{x}}(0) = \delta\mathbf{x_0} \tag{2.26}$$

$$\mathbf{P}(0) = \mathbf{P_0} \tag{2.27}$$

The INS error states are assumed to be zero mean for lack of *a priori* information. Hence, the vector, $\delta\mathbf{x_0}$, is a zero vector. The elements of the matrix, $\mathbf{P_0}$, are defined in Appendix B, and the values of $\mathbf{P_0}$ indicate the extent to which the values in $\delta\mathbf{x_0}$ are believed to be correct. Updating the state estimates with measurements is performed to improve the quality of the state estimates. Measurements have the

following form (16:205):

$$\delta\mathbf{z}(t_i) = \mathbf{H}(t_i)\delta\mathbf{x}(t_i) + \mathbf{v}(t_i) \tag{2.28}$$

$$E\{\mathbf{v}(t_i)\} = \mathbf{0} \tag{2.29}$$

$$E\{\mathbf{v}(t_i)\mathbf{v}^T(t_j)\} = \begin{cases} \mathbf{R}(t_i) & t_i = t_j \\ \mathbf{0} & t_i \neq t_j \end{cases} \tag{2.30}$$

The measurements are used by the filter to update the state estimates. The measurement update equations are (16:275):

$$\mathbf{K}(t_i) = \mathbf{P}(t_i^-)\mathbf{H}^T(t_i)\left[\mathbf{H}(t_i)\mathbf{P}(t_i^-)\mathbf{H}^T(t_i) + \mathbf{R}(t_i)\right]^{-1} \tag{2.31}$$

$$\hat{\delta\mathbf{x}}(t_i^+) = \hat{\delta\mathbf{x}}(t_i^-) + \mathbf{K}(t_i)\left[\delta\mathbf{z}(t_i) - \mathbf{H}(t_i)\hat{\delta\mathbf{x}}(t_i^-)\right] \tag{2.32}$$

$$\mathbf{P}(t_i^+) = \mathbf{P}(t_i^-) - \mathbf{K}(t_i)\mathbf{H}(t_i)\mathbf{P}(t_i^-) \tag{2.33}$$

The measurement updates are actually performed using the UD covariance factorization update algorithm. This algorithm is discussed by Maybeck (16:392-394). The algorithm actually computes the separate factors $\mathbf{U}$ and $\mathbf{D}$ rather than $\mathbf{P}$, where these matrices are interrelated by:

$$\mathbf{P} = \mathbf{U}\mathbf{D}\mathbf{U}^T \tag{2.34}$$

$\mathbf{U}$ is an upper triangular matrix, and $\mathbf{D}$ is diagonal. The nth columns of $\mathbf{U}$ and $\mathbf{D}$ are calculated by:

$$\begin{aligned} D_{nn} &= P_{nn} \\ U_{in} &= \begin{cases} 1 & i = n \\ P_{in}/D_{nn} & i = n-1, n-2, \ldots, 1 \end{cases} \end{aligned} \tag{2.35}$$

The remaining columns are calculated for $j = n-1, n-2, \ldots, 1$ by:

$$
\begin{aligned}
D_{jj} &= P_{jj} - \sum_{k=j+1}^{n} D_{kk} U_{jk}^2 \\
U_{ij} &= \begin{cases} [P_{jj} - \sum_{k=j+1}^{n} D_{kk} U_{ik} U_{jk}] / D_{jj} & i = j-1, j-2, \ldots, 1 \\ 1 & i = j \\ 0 & i > j \end{cases}
\end{aligned} \tag{2.36}
$$

After **U** and **D** have been calculated at time t_i^-, just before measurements are incorporated, the update is performed on the matrices. MSOFE performs a scalar update for each measurement which occurs. The equations for a scalar update are:

$$\mathbf{f} = \mathbf{U}^T(t_I^-)\mathbf{H}^T(t_i) \tag{2.37}$$

$$v_j = D_{jj}(t_i) f_i \qquad j = 1, 2, \ldots, n \tag{2.38}$$

$$a_0 = R \tag{2.39}$$

The **U** and **D** matrices are updated for $k = 1, 2, \ldots, n$ by:

$$
\begin{aligned}
a_k &= a_{k-1} + f_k v_k \\
D_{kk}(t_i^+) &= D_{kk}(t_i^-) a_{k-1} / a_k \\
b_k &\leftarrow v_k \\
p_k &= -f_k / a_{k-1}
\end{aligned} \tag{2.40}
$$

For $j = 1, 2, \ldots, (k-1)$, the procedure is continued with:

$$
\begin{aligned}
U_{jk}(t_i^+) &= U_{jk}(t_i^-) + b_j p_k \\
b_j &\leftarrow b_j + U(t - i^-) v_k
\end{aligned} \tag{2.41}
$$

The $\leftarrow$ symbol in Equations (2.40) and (2.41) represent the technique in programs in which a variable can be overwritten with a different value. The new forms of **U**

and **D** are then multiplied using (2.34) to form the updated **P** matrix. The state vector is subsequently updated using the following equations.

$$\mathbf{K}(t_i) = \mathbf{b}/a_n \tag{2.42}$$

$$\hat{\delta\mathbf{x}}(t_i^+) = \hat{\delta\mathbf{x}}(t_i^- +) + \mathbf{K}(t_i)[z_i - \mathbf{H}(t_i)\hat{\delta\mathbf{x}}(t_i^-)] \tag{2.43}$$

This process is repeated for as many scalar measurements as are available at time t_i. With the completion of the update, the state vector and covariance matrix can be propagated to another update time.

2.6.1 *Extended Kalman Filter* The Kalman filter equations shown above require that the dynamics equations be linear. One feature of linear equations is that they contain only elements which are taken to the first power. Nonlinear equations contain squares, cubes, or greater powers of terms in the equation. An example of a linear equation is:

$$\dot{x}_1 = x_1 + x_2 \tag{2.44}$$

whereas, a nonlinear equation may take the following form:

$$\dot{x}_1 = x_1^2 + x_2^3 + x_3 \tag{2.45}$$

When some or all of the dynamics equations are non-linear, it becomes necessary to either linearize the model or utilize an extended Kalman filter.

The linearized Kalman filter takes some assumed or known nominal trajectory and evaluates the derivatives of the dynamics equations at the nominal trajectory to form the dynamics matrix. That is, with a nonlinear set of equations,

$$\dot{\mathbf{x}}(t) = \mathbf{f}[\mathbf{x}(t), \mathbf{u}(t), t] \tag{2.46}$$

and some nominal trajectory, $\mathbf{x}_n(t)$, the linearized dynamics matrix is found by (17:41):

$$\mathbf{F}[t; \mathbf{x}_n(t)] = \frac{\partial \mathbf{f}[\mathbf{x}, \mathbf{u}(t), t]}{\partial \mathbf{x}}|_{\mathbf{x}=\mathbf{x}_n(t)} \tag{2.47}$$

A nonlinear set of update equations are linearized in a similar fashion (17:41).

$$\mathbf{H}[t_i; \mathbf{x}_n(t_i)] = \frac{\partial \mathbf{h}[\mathbf{x}, t_i]}{\partial \mathbf{x}}|_{\mathbf{x}=\mathbf{x}_n(t_i)} \tag{2.48}$$

The extended Kalman filter is basically a linearized Kalman filter which relinearizes the dynamics equations after each update (17:42), and the update matrix, $\mathbf{H}$, is relinearized after each propagation (17:44). An extended Kalman filter is used in this thesis. The nonlinear equations are linearized using a Taylor series expansion and truncating to first order.

For truth model validation, a covariance analysis of the linearized filter associated with the extended Kalman filter is performed rather than a Monte Carlo analysis of the extended Kalman filter. The covariance analysis is used to decrease total simulation time to validate the models. The linearization of the extended filter is used because that is how MSOFE processes the filter in covariance mode.

2.7 Software Tools

As mentioned above, PC Buffer Box (11) is instrumental in obtaining empirical data from the GPSUE. Three other programs are also instrumental to the completion of the work. The programs are PROFGEN, MSOFE, and MATRIX$_x$.

2.7.1 PROFGEN PROFGEN is short for PROFile GENerator. This program allows for the generation of flight profiles designed by the user. The user is able to specify the flight route and maneuvers to be flown, and PROFGEN generates position, attitude, velocity, and acceleration information for the flight. The data output by PROFGEN is usable by MSOFE. More information on PROFGEN is

available in 'PROFGEN - A Computer Program for Generating Flight Profiles' (2).

2.7.2 MSOFE MSOFE stands for Multimode Simulation for Optimal Filter Evaluation. This program allows the user to perform Monte Carlo and covariance evaluations of Kalman filters. The user programs in the dynamics and update models. By using internal or external trajectory information, the program allows the user to test the designed filter against a truth model. The program can be used to validate a truth model, and with small modifications, the user can utilize empirical data for measurements. Information on the use of MSOFE is found in the MSOFE user's manual (4).

2.7.3 MATRIX$_x$ MATRIX$_x$, from Integrated Systems, Inc., is a useful tool for matrix manipulation, control design and analysis, and plotting. The main use to which it is put in this work is plotting. MSOFE is modified to store the required outputs in a MATRIX$_x$ readable file. Then, the data is loaded into MATRIX$_x$ and plotted.

2.8 Summary

This chapter introduced the reference frames used in the study, and their relationships to each other through transformation matrices. Also, a short presentation of the INS/GPS Integrated Lab concept and its development is given. A vital link to real world data is through communication with the physical devices. The methods in which this was accomplished are presented as well. The last part of the chapter deals with the software tools used. The project could not have been performed in a timely manner without the aid of the software tools described.

III. INS Truth Model Design and Verification

Litton Guidance and Control Systems developed a 93-state error model for the LN-93. This model is applicable to the LN-94 because it is the same INS repackaged to fit in the F-15 Eagle fighter aircraft. The error model and the method used to verify this particular implementation of the model are discussed here.

3.1 INS Truth Model

The states in the INS truth model are interrelated through the system dynamics. The dynamics matrix, $\mathbf{F}$, defines the relation between the system states and their derivatives through the following equation.

$$\delta\dot{\mathbf{x}} = \mathbf{F}\delta\mathbf{x} + \mathbf{G}\mathbf{w} \tag{3.1}$$

where $\mathbf{w}$ is "white" Gaussian noise.

3.1.1 States of the Truth Model The state vector used by Litton is partitioned into six subvectors. These subvectors are designated $\delta\mathbf{x}_1, \delta\mathbf{x}_2, \ldots, \delta\mathbf{x}_6$. The first subvector, $\delta\mathbf{x}_1$, contains 13 states which are general errors such as position, velocity, attitude, and vertical channel errors. The $\delta\mathbf{x}_2$ subvector contains 16 states which are the gyro, accelerometer, and barometer time-correlated errors and trends, and subvector $\delta\mathbf{x}_3$ consists of 18 states related to gyro bias errors. The $\delta\mathbf{x}_4$ subvector consists of 22 states which are related to the accelerometer and barometer bias errors. Subvector $\delta\mathbf{x}_5$ contains six states related to the accelerometer and gyro initial thermal transients, and the $\delta\mathbf{x}_6$ subvector consists of 18 gyro compliance errors. All 93 errors are shown in Table 3.1. The line in column two of Table 3.1 separates the correlated errors above the line from the trends below. A definition for each state can be found in Appendix A.

Table 3.1. States in the LN-94 Error Model

General Errors	Correlated Errors and Trends	Gyro Bias Errors	Accelerometer Bias Errors	Accelerometer and Gyro Initial Thermal Transients	Gyro Compliance Terms
$\delta\theta_x$	b_{x_c}	b_x	∇_{b_x}	∇_{x_q}	F_{xyz}
$\delta\theta_y$	b_{y_c}	b_y	∇_{b_y}	∇_{y_q}	F_{xyy}
$\delta\theta_z$	b_{z_c}	b_z	∇_{b_z}	∇_{z_q}	F_{xyx}
$\delta\phi_x$	∇_{x_c}	S_{g_x}	S_{A_x}	b_{x_q}	F_{xxy}
$\delta\phi_y$	∇_{y_c}	S_{g_y}	S_{A_y}	b_{y_q}	F_{xzz}
$\delta\phi_z$	∇_{z_c}	S_{g_z}	S_{A_z}	b_{z_q}	F_{xzx}
δV_x	δg_x	χ_1	S_{QA_x}		F_{yzx}
δV_y	δg_y	χ_2	S_{QA_y}		F_{yzz}
δV_z	δg_z	χ_3	S_{QA_z}		F_{yxy}
δh	δh_c	ν_1	f_{xx}		F_{yxz}
δh_L	b_{x_t}	ν_2	f_{yy}		F_{yxx}
δS_3	b_{y_t}	ν_3	f_{xz}		F_{yxy}
δS_4	b_{z_t}	D_{xxx}	f_{xy}		F_{zxy}
	∇_{x_t}	D_{yyy}	f_{xz}		F_{zxx}
	∇_{y_t}	D_{zzz}	f_{yx}		F_{zxz}
	∇_{z_t}	S_{Qb_x}	f_{yz}		F_{zyx}
		S_{Qb_y}	f_{zx}		F_{zyy}
		S_{Qb_z}	f_{zy}		F_{zyz}
			μ_1		
			μ_2		
			μ_3		
			σ_1		

Litton has supplied 1σ (one standard deviation) values for latitude and longitude errors and the following states: $\delta\phi_x$, $\delta\phi_y$, $\delta\phi_z$, δV_x, δV_y, δV_z, and δh. Hence, outputs of the 1σ values of these states are necessary to verify that the model has been correctly programmed into MSOFE. The $\delta\theta_i$ terms represent the errors between the computer frame and the true frame. These two states can be related to the latitude and longitude errors committed by the INS. The $\delta\phi_i$ terms are the terms of the skew symmetric matrix, described earlier, which relate the computer frame orientation to the true frame. The remaining terms of interest are the velocity error components in the true frame and the altitude error.

3.1.2 *Truth Model Dynamics Matrix* Litton has partitioned the dynamics matrix, **F**, into eight non-zero submatrices which contain all the non-zero elements of the dynamics matrix. The dynamics matrix, in terms of the submatrices is:

$$\mathbf{F} = \begin{bmatrix} \mathbf{F}_{11} & \mathbf{F}_{12} & \mathbf{F}_{13} & \mathbf{F}_{14} & \mathbf{F}_{15} & \mathbf{F}_{16} \\ \mathbf{0} & \mathbf{F}_{22} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{F}_{55} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{3.2}$$

The elements of these submatrices are presented in Appendix C. The time-varying terms used in the matrix elements are explained here.

Three different angular velocities which are used in the dynamics matrices are defined below. The first is the earth rate in the true frame.

$$\begin{bmatrix} \Omega_x \\ \Omega_y \\ \Omega_z \end{bmatrix} = \begin{bmatrix} \Omega C_l^t(2,1) \\ \Omega C_l^t(2,2) \\ \Omega C_l^t(2,3) \end{bmatrix} \tag{3.3}$$

where Ω is the earth sidereal rate, 7.292115x10^{-5} rad/sec. The equation for calculating the craft rate is:

$$\begin{bmatrix} \omega^t_{it_x} \\ \omega^t_{it_y} \\ \omega^t_{it_z} \end{bmatrix} = \begin{bmatrix} -V^t_y C_{RY} + \Omega_x \\ V^t_x C_{RX} + \Omega_y \\ \Omega_z \end{bmatrix} \tag{3.4}$$

The V^t_x and V^t_y terms are the x and y velocity components in the true frame. C_{RX} and C_{RY} are components of the earth inverse spheroid radii of curvature (14:9). They are calculated by:

$$C_{RX} = \frac{1}{a}[1 - \frac{h}{a} - f\{C^{t^2}_l(2,3) - 2C^{t^2}_l(2,1)\} \tag{3.5}$$

$$C_{RY} = \frac{1}{a}[1 - \frac{h}{a} - f\{C^{t^2}_l(2,3) - 2C^{t^2}_l(2,2)\} \tag{3.6}$$

where a is the equatorial radius of the earth, and f is flattening of the meridional ellipse. The third angular velocity is that of the aircraft body with respect to the inertial space. This is calculated by:

$$\begin{bmatrix} \omega^t_{ib_x} \\ \omega^t_{ib_y} \\ \omega^t_{ib_z} \end{bmatrix} = \begin{bmatrix} \omega^t_{it_x} \\ \omega^t_{it_y} \\ \omega^t_{it_z} \end{bmatrix} + \mathbf{C^t_b} \begin{bmatrix} \dot{\varphi} \\ \dot{\vartheta} \\ \dot{\psi} \end{bmatrix} \tag{3.7}$$

where $\dot{\varphi}$, $\dot{\vartheta}$, and $\dot{\psi}$ are the rates of change of roll, pitch, and yaw respectively. This angular velocity is transformed to the body frame by:

$$\omega^{\mathbf{b}}_{\mathbf{ib}} = \mathbf{C^b_t}\omega^{\mathbf{t}}_{\mathbf{ib}} \tag{3.8}$$

where $\mathbf{C^b_t}$ is the matrix inverse of $\mathbf{C^t_b}$.

Specific force values in the true and body frames are terms used in the dynamics matrix. The equation relating the two specific force vectors is:

$$\begin{bmatrix} A_x^t \\ A_y^t \\ A_z^t \end{bmatrix} = \mathbf{C_b^t} \begin{bmatrix} A_x^b \\ A_y^b \\ A_z^b \end{bmatrix} \tag{3.9}$$

In certain elements, an $A_z'^b$ term, the specific force in the z axis of the body frame with gravity removed, is found. This component is found by:

$$A_z'^b = C_b^t(1,3)A_x^t + C_b^t(2,3)A_y^t + C_b^t(3,3)[A_z^t - G] \tag{3.10}$$

where

$$G = g_0[1 - (2.00996)\frac{h}{a} + 5.28659\text{x}10^{-3}\{C_i^l(2,3)\}^2] \tag{3.11}$$

$$g_0 = 32.08744 \text{ ft./sec.}^2 \tag{3.12}$$

The notation $C_b^t(i,j)$ indicates the element in row i, column j of the body to true frame direction cosine matrix.

All nine elements of the body-to-true direction cosine matrix are utilized in the dynamics matrix. To reduce the space required, the notation for indicating each element has been shortened. The element $C_b^t(i,j)$ is indicated in the matrix as C_{ij}.

Four vertical channel variables are used in the dynamics matrix. These constants are used as multipliers of states δh_l, δS_4, and δh_c in the error model which feedback to the vertical velocity and altitude error states. They are calculated by the following algorithm (14).

$$\Delta = |V_z^t| \tag{3.13}$$

$$\Delta_0 = \begin{cases} 30 \text{ fps} & \text{initially} \\ \Delta_0 + 8 & \text{if } \Delta_0 \leq \Delta \\ \Delta_0 - 8 & \text{if } \Delta_0 > \Delta \text{ and } \Delta_0 > 38 \\ 30 \text{ fps} & \text{otherwise} \end{cases} \tag{3.14}$$

$$\lambda_1 = 100[1 + (\frac{\Delta}{\Delta_0})^2] \tag{3.15}$$

$$k_1 = \frac{3}{\lambda_1} \tag{3.16}$$

$$k_2 = \frac{2g_0}{a} + \frac{4}{\lambda_1^2} \tag{3.17}$$

$$k_3 = \frac{2}{\lambda_1^3} \tag{3.18}$$

$$k_4 = \frac{\Delta^2}{\Delta_0^2 + \Delta^2} \tag{3.19}$$

The ten correlated errors in δx_2 and the six accelerometer and gyro initial thermal transients are all first order Markov processes. Their correlation times are presented below.

$$\beta_{b_{[x,y,z]_c}} = \frac{1}{10} \text{ min.}^{-1} \qquad \beta_{\nabla_{[x,y,z]_c}} = \frac{1}{5} \text{ min.}^{-1}$$

$$\beta_{b_{[x,y,z]_q}} = \frac{1}{600} \text{ sec.}^{-1} \qquad \beta_{\nabla_{[x,y,z]_q}} = \frac{1}{60} \text{ sec.}^{-1}$$

$$\beta_{h_c} = \frac{1}{600} \text{ sec.}^{-1} \qquad \beta_{\delta g_{[x,y,z]}} = \frac{|V|}{121522.3} \text{ sec.}^{-1}$$

The subscripts on the correlation time identifiers refer to the particular state to which they correspond (see Appendix A). The bracketed x,y, and z subscripts indicate that there is a separate state for each of the three axes.

The velocity for the gravity error time constant is calculated as:

$$|V| = \sqrt{V_x^2 + V_y^2 + v_z^2}$$

Many of the values in the dynamics matrix are time varying. Thus, the equations presented are programmed into MSOFE as shown so that the dynamics matrix may be time varying.

3.1.3 Model Dynamics Driving Noise As indicated in Equation (3.1), there is some "white" Gaussian noise driving the system. Litton partitioned the **Gw** term into two groupings. The form of the noise term is, using the same partitioning as seen in Equation (3.2):

$$\mathbf{Gw} = \begin{bmatrix} \mathbf{w_1} \\ \mathbf{w_2} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \tag{3.20}$$

It was shown earlier that $E\{\mathbf{w}(t)\} = \mathbf{0}$ and $E\{\mathbf{w}(t)\mathbf{w}^T(t+\tau)\} = \mathbf{Q}(t)\delta(\tau)$. To be consistent with the partitioning of the noise term into two column matrices, the **Q** term is also partitioned into two terms. The form of the matrix used in the model is shown below.

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Q_{11}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q_{22}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{3.21}$$

The elements of $\mathbf{Q_{11}}$ and $\mathbf{Q_{22}}$ are shown in Appendix D. The β terms which appear in the matrices are the same as those defined for the dynamics matrix, as seen on the previous page. The σ terms are displayed below.

$$\begin{aligned} \sigma_{\eta_{b_{[x,y,z]}}} &= .09°/\text{hr.}/\sqrt{\text{Hz}} & \sigma_{\eta_{A_{[x,y,z]}}} &= 10\mu g/\sqrt{\text{Hz}} \\ \sigma_{b_{[x,y,z]_c}} &= .002°/\text{hr.} & \sigma_{\nabla_{[x,y,z]_c}} &= 2\mu g \\ \sigma_{\delta g_{[x,y,z]}} &= 5 \text{ arcsec} & \sigma_{\delta h_c} &= 100 \text{ ft.} \end{aligned}$$

3.2 *Truth Model Verification*

After programming the model into MSOFE, the model is verified. During the process of verification, a number of discrepancies between the equations and matrices provided in the Litton document were discovered. Also, data on calculations of various data and some values of constants were discovered. The discrepancies and omissions were resolved with the help of Mr. Lowell Knudsen (13) and Dr. James Huddle (10) of Litton Guidance and Control Systems. Errata for the truth model document provided by Litton are presented in Appendix F. The Litton document provided five types of navigation simulations to form the basis for comparisons. The ten-hour static navigation with eight-minute gyrocompass alignment and a two-hour fighter navigation with eight-minute gyrocompass alignment are chosen for comparison. The static navigation is chosen because of the ease of implementation on MSOFE, and the fighter profile is chosen because it provides the greatest observability of the error states.

3.2.1 Gyrocompass Alignment The INS undergoes an eight-minute gyrocompass alignment at the beginning of each simulation. The values used for λ, h, V_x, V_y, V_z, A_x, A_y, and A_z are 0 deg., 0 ft., 0 fps, 0 fps, 0 fps, 0 ft./sec.^{2}, 0 ft./sec.^{2}, and 32.08744 ft./sec.^{2} respectively for both flight profiles. The latitude is different for the two flight profiles. For the static navigation, the latitude is 32 deg. 46.6 min. North, and the latitude is 45 deg. North for the fighter profile. Following the discussion in Capt Solomon's master's thesis (21:3-9), the alignment is simulated by updating the horizontal plane velocity errors at a rate of 1/2 Hz. The velocity updates, update rate and the measurement noise variances are provided by Litton engineers as described in Capt Solomon's thesis (21:3-9).

MSOFE requires that the **H** matrix and measurement noise strength (R) values for each update be entered in the program. MSOFE performs the updates in a sequential manner. If two measurements occur at the same time, MSOFE performs

an update with one measurement and then performs a another update with the second measurement. The **H** matrices and measurement noise values used for the alignment are given below.

The values for the update of the 'x' velocity error are:

$$\mathbf{H} = [0\,0\,0\,0\,0\,0\,1\,0\,0 \ldots 0] \tag{3.22}$$

$$R = 0.02 \text{ ft.}^2/\text{sec.}^2 \tag{3.23}$$

The values used for the 'y' velocity error are:

$$\mathbf{H} = [0\,0\,0\,0\,0\,0\,0\,1\,0 \ldots 0] \tag{3.24}$$

$$R = 0.02 \text{ ft.}^2/\text{sec.}^2 \tag{3.25}$$

The alignment simulation is run for an alignment time of eight minutes. Plots of the variances of latitude error, longitude error, and the states $\delta\phi_x$, $\delta\phi_y$, $\delta\phi_z$, δV_x, δV_y, and δV_z are obtained for the covariance analysis simulation and are shown in Figure 3.1.

On the plot of the tilt errors, the north tilt lies on top of the east tilt so that it appears that the east tilt error is not displayed. The east tilt error is actually being displayed, it is merely hidden by the north tilt error. Some processing of the states is performed to obtain output of latitude and longitude errors in feet and to obtain the $\delta\phi_i$ and δV_i terms in the navigation frame. Since a covariance analysis is being performed, the processing is performed on the covariance matrix. The following calculations are used to obtain the latitude and longitude errors in feet:

$$\mathbf{P}_{\delta\theta} = \begin{bmatrix} P(1,1) & P(1,2) & P(1,3) \\ P(2,1) & P(2,2) & P(2,3) \\ P(3,1) & P(3,2) & P(3,3) \end{bmatrix} \tag{3.26}$$

(3.27)

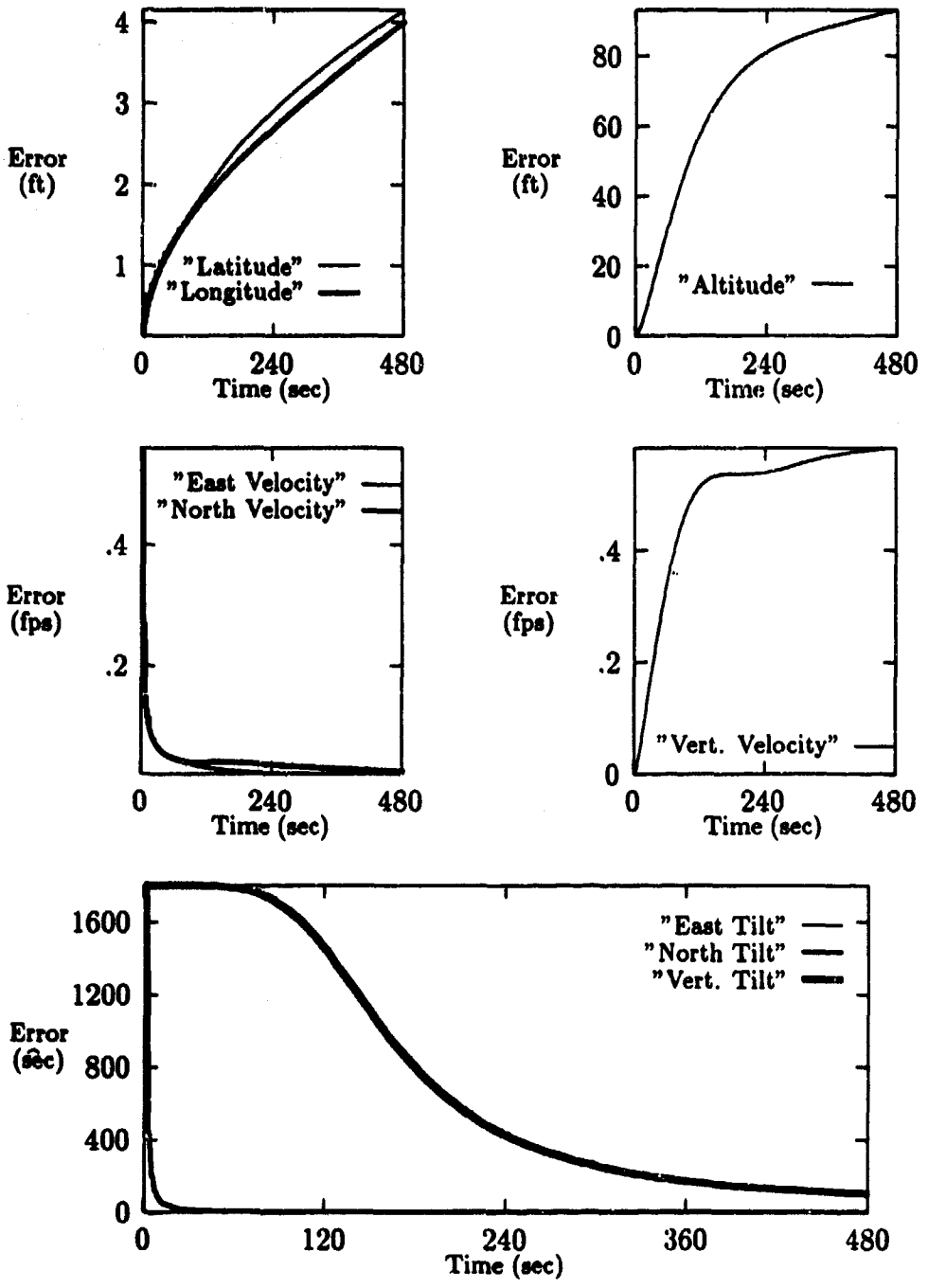


Figure 3.1. Covariance of INS Error States During Alignment Simulation

$$r_e = A(1 - e \sin \hat{L}^2) \tag{3.28}$$

(3.29)

$$\mathbf{T_{Ll}} = \begin{bmatrix} r_e \cos \hat{\alpha_T} & -r_e \sin \hat{\alpha_T} & 0 \\ r_e \sin \hat{\alpha_T} & r_e \cos \hat{\alpha_T} & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.30}$$

(3.31)

$$\mathbf{P_{Ll}} = \mathbf{T_{Ll}} \mathbf{P}_{\delta\theta} \mathbf{T_{Ll}}^T \tag{3.32}$$

The e term in the calculation of r_e , the local distance to earth's center, is the eccentricity of the earth reference ellipse. The equations used for transforming the tilts to the navigation frame are:

$$\mathbf{P}_\phi = \begin{bmatrix} P(4,4) & P(4,5) & P(4,6) \\ P(5,4) & P(5,5) & P(5,6) \\ P(6,4) & P(6,5) & P(6,6) \end{bmatrix} \tag{3.33}$$

(3.34)

$$\mathbf{P_{tilt}} = \mathbf{C_t^n} \mathbf{P}_\phi \mathbf{C_n^t} \tag{3.35}$$

where $\mathbf{C_n^t}$ is the transpose and matrix inverse of $\mathbf{C_t^n}$. A similar procedure, shown below, is performed for the transformation of the velocity error terms:

$$\mathbf{P}_{\delta \mathbf{v_t}} = \begin{bmatrix} P(4,4) & P(4,5) & P(4,6) \\ P(5,4) & P(5,5) & P(5,6) \\ P(6,4) & P(6,5) & P(6,6) \end{bmatrix} \tag{3.36}$$

(3.37)

$$\mathbf{P}_{\delta \mathbf{v_n}} = \mathbf{C_t^n} \mathbf{P}_{\delta \mathbf{v_t}} \mathbf{C_n^t} \tag{3.38}$$

The above transformations are used for the plotting of data from the static navigation and fighter flight simulations.

3.2.2 *Static Navigation Simulation* The static simulation is performed with a position of 32 deg. 46.6 min. North latitude and 0 deg. longitude. MSOFE provides a file at the end of a run which can be used to initialize data at the start of another run. The data available at the end of the alignment simulation for 32 deg. 46.6 min. North latitude is used to initialize the variables at the start of the static navigation.

The variables which are compared to the available Litton information, shown for this simulation and the fighter flight in Appendix F, are plotted in Figure 3.2. These plots are similar to the plots in the Litton CDRL indicating that the model has been correctly implemented. The most significant differences are the size of oscillations of the horizontal velocity and tilt errors. The oscillations are not of the same magnitude, but the same general trend is evident. Also, the end magnitudes are smaller in the covariance simulation than in the Litton document. The differences are attributed to performing a covariance analysis while Litton's plots are the average of ten Monte Carlo simulations.

3.2.3 *Fighter Flight Profile Simultion* The fighter flight begins at a latitude of 45 deg. North. Thus, the initial values of the states and covariance matrix are obtained from the output at the end of the 45 deg. North latitude alignment simulation.

The fighter flight is approximated as closely as possible with the information given using PROFGEN. The resulting flight is shown in Figure 3.3. The major difference between this flight and the Litton profile is that the return flight passes the takeoff point so an abrupt turnaround is performed to complete the flight as close to the takeoff point as possible. The information shown in the Litton document does not show the return flight passing the airfield. However, following the information about times, velocities, and altittudes provided, it is impossible to keep from passing the point of takeoff. The states of interest are plotted in Figure 3.4. These results are

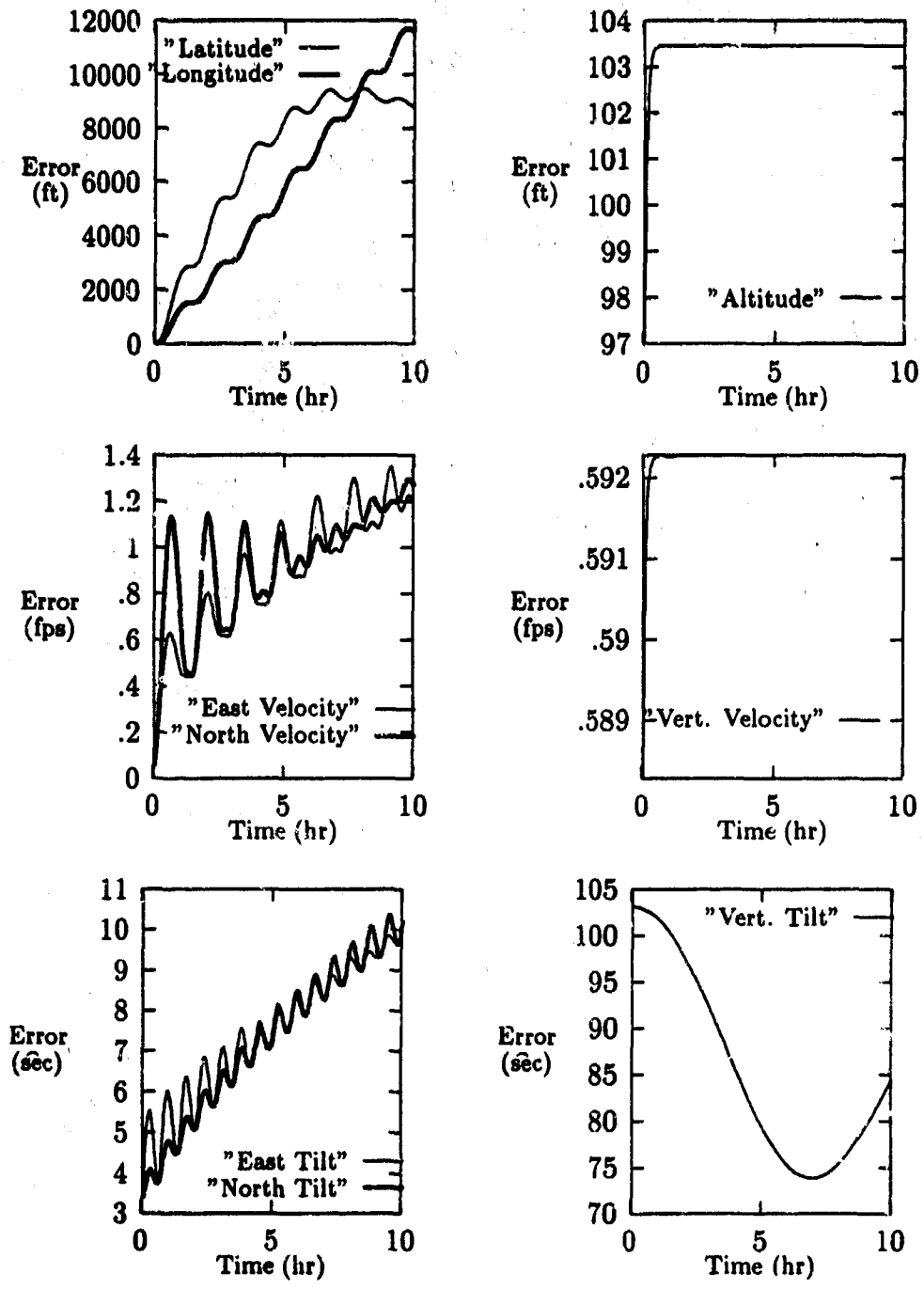


Figure 3.2. 1σ Plots for Static Navigation Simulation

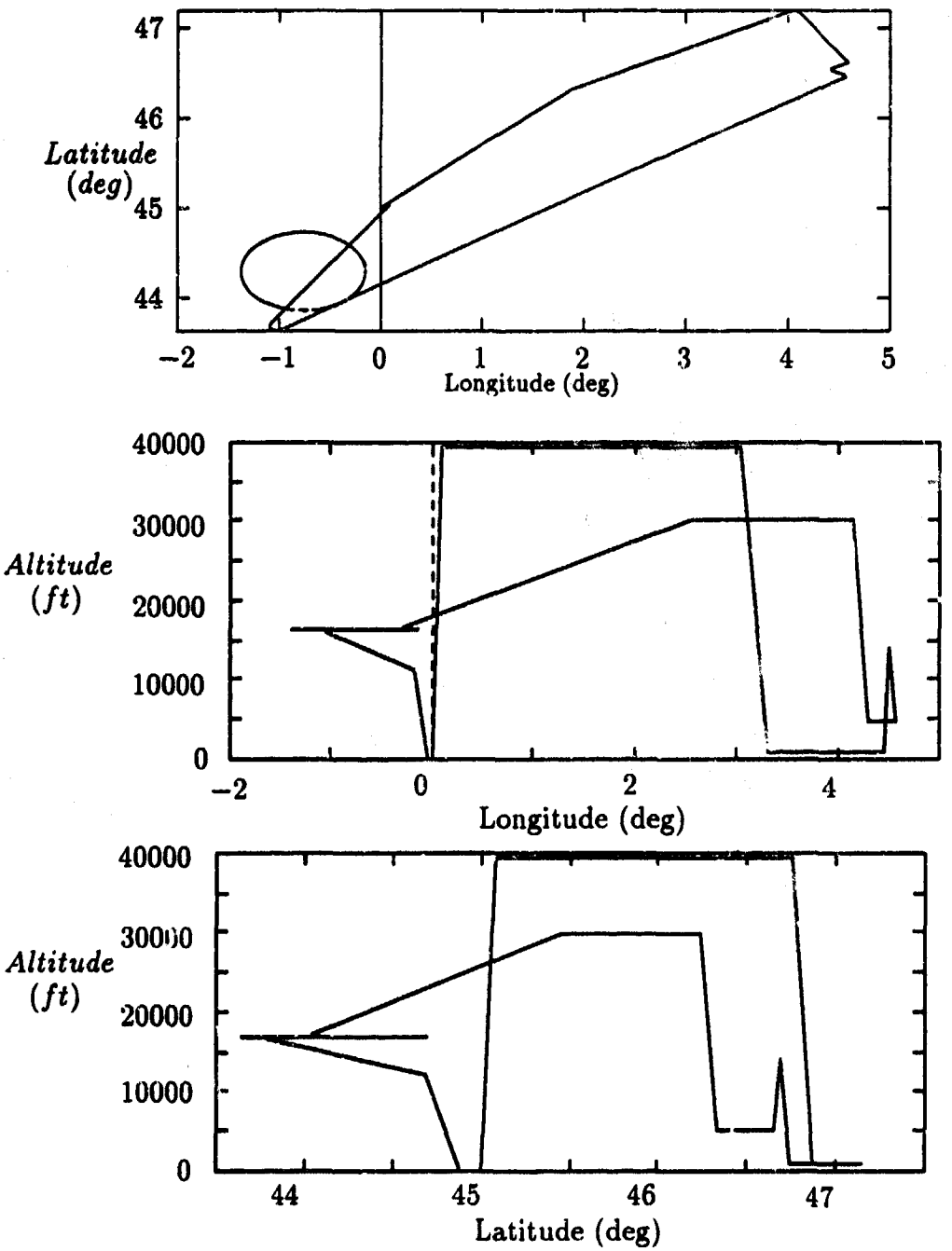


Figure 3.3. Fighter Flight Profile

also similar to the plots provided by Litton in Appenix F. The plots show the same trends, and the horizontal velocity and tilt errors are nearly the same magnitudes as the values shown in the Litton truth model document. This provides additional validation of this implementation of the truth model. Differences are primarily the result of using covariance analysis instead of Monte Carlo simulations as stated earlier. Errors at the end of the flight are the result of the differences in the flight profiles at this point.

3.3 Summary

The truth model as given in the Litton CDRL is programmed into MSOFE. Simulations are performed for an eight-minute alignment, a ten-hour static navigation, and a two-hour fighter flight. A small amount of post processing is necessary to transform the data into the reference system used by Litton. Plots of state variables are compared for the static navigation and fighter flight simulations. The plots are similar in nature. Differences are the result of different types of simulations being performed. Overall, the simulations are quite similar, and the model is judged to be correctly implemented.

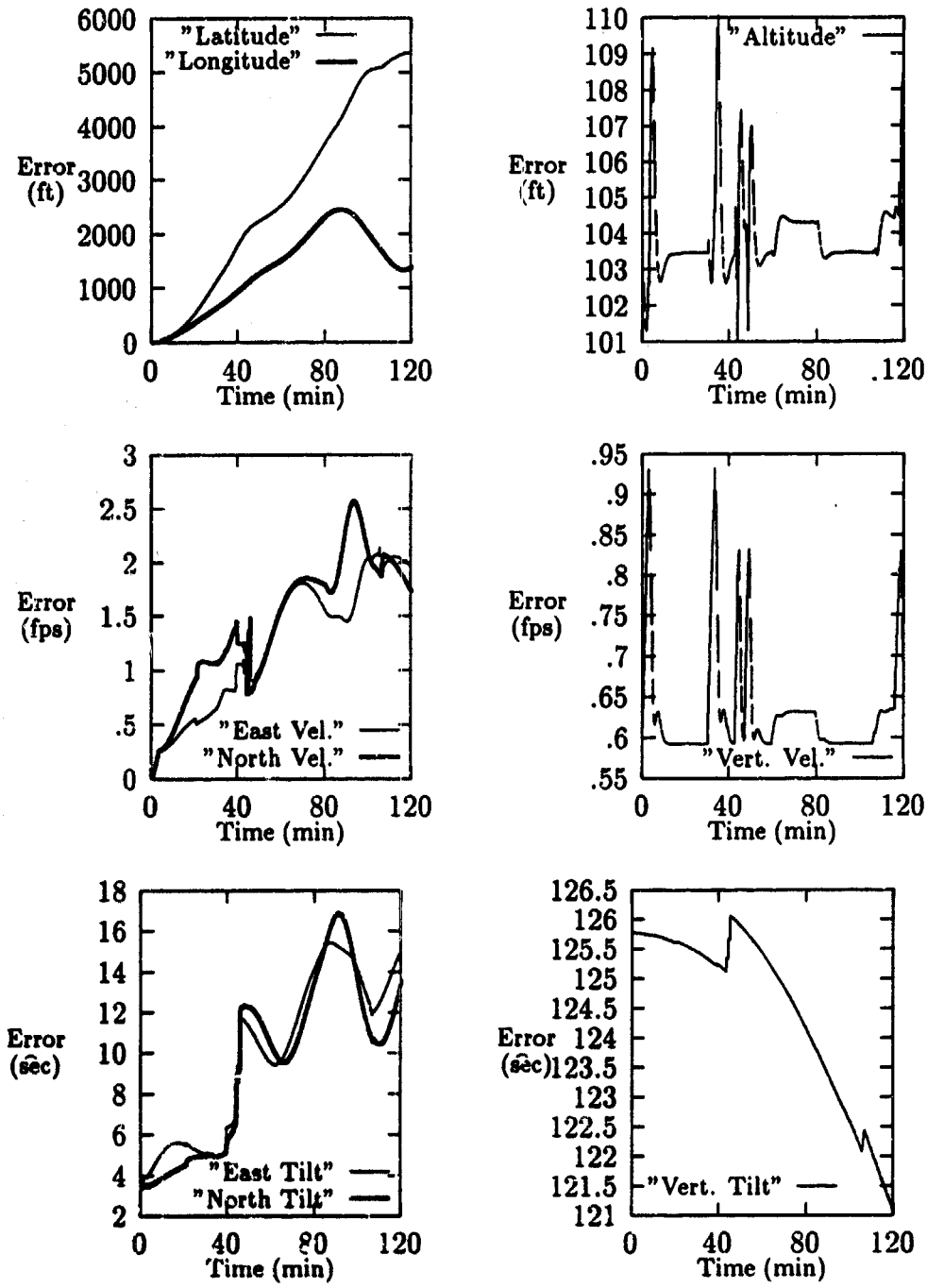


Figure 3.4. 1σ Plots for Fighter Flight Simulation

IV. GPS Truth Model Design, Validation and Integration with INS

The truth model used for the Global Positioning System is based on a model developed by Capt Solomon (20). The truth model contains 22 states. Twenty states are five states repeated four times, once for each satellite being used to calculate the navigation solution. The remaining states pertain to the user clock errors.

4.1 GPS Truth Model States

The GPS truth model contains states for the user clock error, code loop error, atmospheric error, and satellite position error. A brief discussion of each error is given here. The code loop error, atmospheric error, and satellite position errors are repeated four times, once for each satellite. User clock errors are the predominant errors in the pseudorange and delta-range measurements. Thus, it would seem that more states should be used to model the user clock errors than the errors due to the satellites. However, the user clock errors are predominantly a bias and a drift. Hence, only two states are used to model this error. However, the satellite errors are large enough to affect the measurements, and the satellite position errors are not easily classified into one or two states. Hence, five states are used to model the satellite errors.

4.1.1 User Clock Error Two states are used to model the user clock error. These states model the clock error as a bias and a drift. This model is implemented with the following equations.

$$\begin{bmatrix} \dot{\delta x_b} \\ \dot{\delta x_d} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \delta x_b \\ \delta x_d \end{bmatrix} \tag{4.1}$$

The initial conditions used for these errors are:

$$\begin{aligned} \delta\mathbf{x}(t_0) &= \mathbf{0} \\ \mathbf{P}(t_0) &= \begin{bmatrix} 9\text{x}10^{14}\ \text{ft.}^2 & 0 \\ 0 & 9\text{x}10^{10}\ \text{fps}^2 \end{bmatrix} \end{aligned}$$

No driving noise is included in this model. This could cause the Kalman filter gains to go to zero if the same model were used without change as part of the filter design model. Hence, it might be necessary to include some pseudonoise in the filter to prevent this occurrence.

4.1.2 *Atmospheric Error* The atmospheric error is modeled as a first order Markov process with a correlation time of 500 seconds (20:7). This model takes into consideration not only atmospheric delay, but it also includes ionospheric delays which have not been totally removed by the receiver prior to updating with the pseudorange and delta-range measurements. The equations used for this model are shown below.

$$\begin{aligned} \dot{\delta x}_{atm}(t) &= -\frac{1}{500}\delta x_{atm}(t) + w_{atm} \qquad (4.2) \\ E\{w_{atm}(t)\} &= 0 \\ E\{w_{atm}(t)w_{atm}(t+\tau)\} &= 0.004\ \frac{\text{ft.}^2}{\text{sec.}}\ \delta(\tau) \end{aligned}$$

The initial conditions for this model are:

$$\begin{aligned} \delta x_{atm}(t_0) &= 0 \\ P_{atm}(t_0) &= 1\ \text{ft.}^2 \end{aligned}$$

The correlation time and initial covariance values used by Capt Solomon are based on an aircraft flight at an average of 30,000 feet. The laboratory is stationary at approximately 1000 feet; therefore, it may be necessary to decrease these values for

filter implementation with empirical data because there is more atmosphere between the receiver and the satellites. Thus, the atmospheric errors may change at a greater rate than is being modeled. The values of $Q = 0.004$ ft.^{2}/sec. and $P_{atm}(t_0) = 1$ ft.^{2} yield stationary process characteristics.

Atmospheric delay is not only a function of time but is also a function of portion of the sky in which the satellite is located. The signal from a satellite behind a cloud will be delayed a different amount than the signal from a satellite with clear sky between it and the GPS antenna. Thus, separate atmospheric states are used for each satellite because different satellites may be in different portions of the sky.

4.1.3 *Code Loop Error* The phase lock loop in the GPSUE has a bandwidth of approximately 1 radian per second to maintain signal lock even in a jamming environment. Therefore, this error is also modeled with a first order Markov process shaping filter. The equations for implementing this model are:

$$\begin{aligned} \dot{\delta x}_{code}(t) &= -\delta x_{code}(t) + w_{code} \\ E\{w_{code}(t)\} &= 0 \\ E\{w_{code}(t)w_{code}(t+\tau)\} &= 0.5\,\frac{\text{ft.}^2}{\text{sec.}}\delta(\tau) \end{aligned} \tag{4.3}$$

with initial conditions:

$$\begin{aligned} \delta x_{code}(t_0) &= 0 \\ P_{code}(t_0) &= 0.25 \text{ ft.}^2 \end{aligned}$$

The correlation time of this model is one second. The correlation time and the initial covariance of 0.25 ft.^{2} determines the noise strength of 0.5ft.^{2}/sec in order to yield stationary process characteristics.

The five channel receiver in use in this research utilizes a separate correlator board for each satellite being used. Hence, a separate code loop is being used for

each satellite. Thus, a separate code loop error is modeled for each satellite.

4.1.4 *Satellite Position Error* The satellite position error is modeled as a three-dimensional random bias vector. Each state models one component of the satellite's position error in the ECEF frame. Some pseudonoise may be necessary to portray the time varying nature of this error accurately. The equation used to model this error is:

$$\dot{\delta \mathbf{x}}_{pos}(t) = \mathbf{0} \tag{4.4}$$

with initial conditions:

$$\begin{aligned} \delta \mathbf{x}_{pos}(t_0) &= \mathbf{0} \\ \mathbf{P}_{pos}(t_0) &= \begin{bmatrix} 25 & 0 & 0 \\ 0 & 25 & 0 \\ 0 & 0 & 25 \end{bmatrix} \text{ ft.}^2 \end{aligned}$$

4.2 GPS Truth Model Equations

The equations for the GPS error models are programmed into MSOFE. The forms of the dynamics and process noise matrices are presented here.

4.2.1 *GPS Dynamics Matrix* The dynamics matrix for the GPS model is taken directly from the equations for the error states. The dynamics matrix is shown below:

$$\mathbf{F}_{GPS} = \begin{bmatrix} \mathbf{F}_{clk} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{F}_{sv_1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{F}_{sv_2} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{F}_{sv_3} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{F}_{sv_4} \end{bmatrix} \tag{4.5}$$

where

$$\mathbf{F}_{clk} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}$$

$$\mathbf{F}_{sv_i} = \begin{bmatrix} -\frac{1}{500} & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

4.2.2 *GPS Process Noise Matrix* The process noise matrix **Q** is broken into four parts. The form of this matrix is:

$$\mathbf{Q}_{GPS} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \mathbf{Q}_{sv_1} & 0 & 0 & 0 \\ 0 & 0 & 0 & \mathbf{Q}_{sv_2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \mathbf{Q}_{sv_3} & 0 \\ 0 & 0 & 0 & 0 & 0 & \mathbf{Q}_{sv_4} \end{bmatrix} \tag{4.6}$$

where

$$\mathbf{Q}_{sv_i} = \begin{bmatrix} 0.004 & 0 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \frac{\text{ft.}^2}{\text{sec.}}$$

4.3 GPS Truth Model Verification

A one-hour filter run is performed to determine whether or not the equations are entered correctly. The diagonal terms of the covariance matrix are examined to determine if the states are behaving as expected.

The user clock drift and satellite position error variances should remain constant, and the user clock bias should be a line with slope equal to the clock drift. The code loop and atmospheric delays are first order Markov processes. With the initial variances presented earlier, the variances for these two states should be constant. However, in order to display the transient which occurs with different initial variances, the initial variances for the atmospheric and code loop delay states are switched. Hence, their variances should approach their final values within their respective correlation times and remain constant thereafter. The plots for the one hour simulation of the GPS model are shown in Figure 4.1. The models of atmospheric and code loop delays and satellite position errors are the same for all four satellites, and this results in identical variances during the analysis. Hence, the results for just one satellite are shown. While the models for the satellite states are the same for each satellite, the actual values of the errors in the "real" world are different for different satellites. Thus, the separation of the five satellite specific errors is retained. The user clock errors display the expected results as do the atmospheric delay and satellite position errors. The code loop delay appears to be just a straight line; however, the code loop correlation time is one second. Thus, the code loop delay reached its final value before the first sample was taken at ten seconds into the run. Therefore, the models are determined to be programmed correctly.

4.4 GPS Integration with INS

With the INS model verified against a known standard and the GPS model determined to be correctly programmed, the models are combined into a single truth model, and they are programmed as a single Kalman filter. The full-ordered filter is

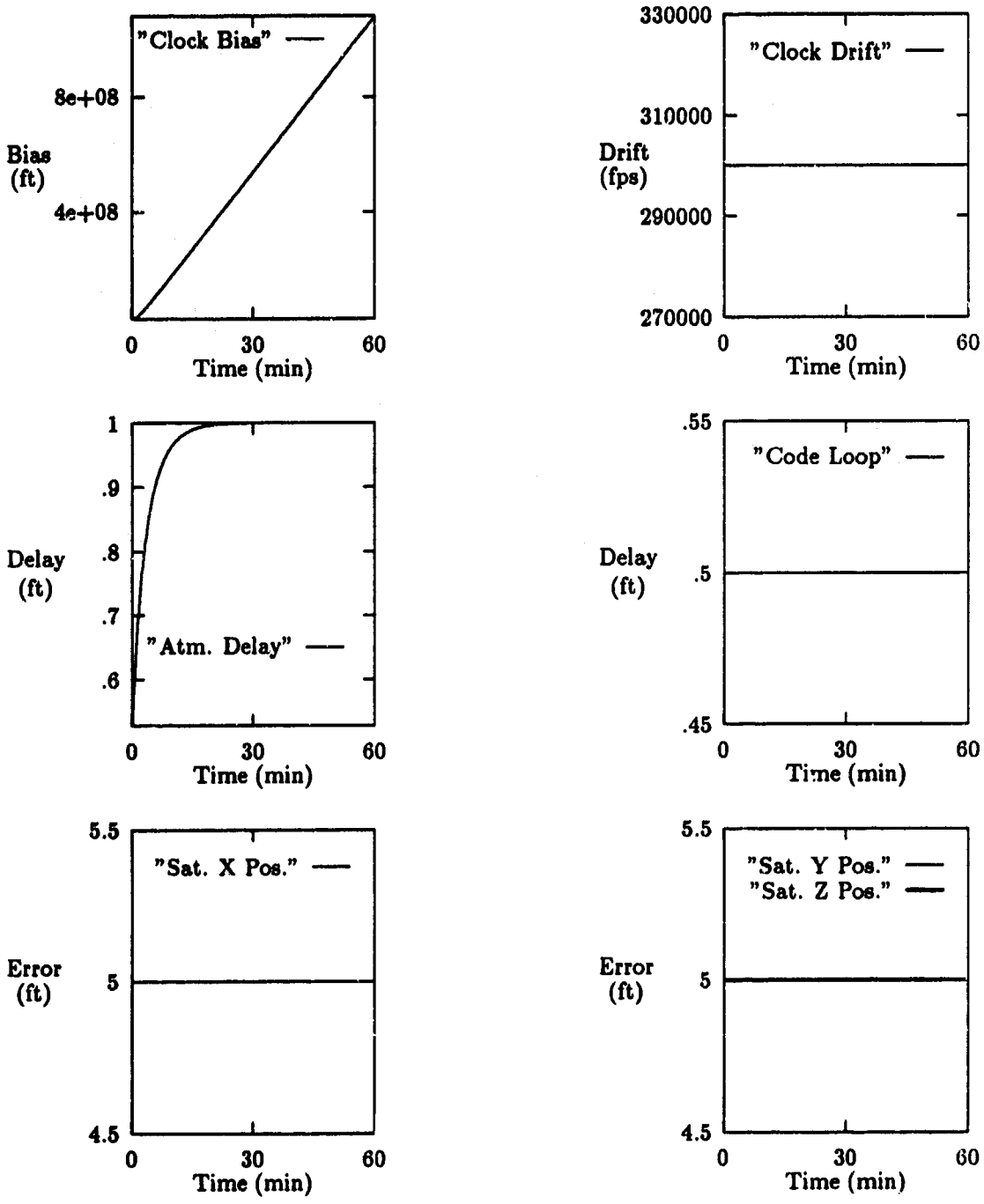


Figure 4.1. One Hour GPSUE Simulation

to be used as a baseline for order reduction in laboratory experiments; so, no order reduction is performed at this point. Hence, the full-ordered filter is being simulated.

4.4.1 *True vs Filter Trajectory Information* The trajectory information for the verification of the models is uncorrupted by errors ("true" trajectory data). This information is either programmed directly into MSOFE, static navigation case, or provided by PROFGEN, fighter flight profile case. For the filter performance prediction, the truth model dynamics matrix is calculated with the "true" trajectory data. However, the filter is provided corrupted trajectory information. The truth model calculations of the INS errors are subtracted from the "true" trajectory to obtain information similar to that provided by an actual INS, and the filter estimation of the errors are added to the information to obtain the filter's estimate of the trajectory data. The calculations to provide the filter trajectory data are shown below. The wander angle is corrupted by:

$$\alpha_{t_{ins}} = \alpha_t - \delta\alpha_t \tag{4.7}$$

where

$$\delta\alpha_t = \delta\theta_z - \delta\lambda \sin\Phi \tag{4.8}$$

The Kalman filter estimate of the corrupted wander angle is:

$$\hat{\alpha}_t = \alpha_{t_{ins}} + \delta\hat{\alpha}_t \tag{4.9}$$

where

$$\delta\hat{\alpha}_t = \delta\hat{\theta}_z - \delta\hat{\lambda} \sin\hat{\Phi} \tag{4.10}$$

The estimate of the position vector, $\hat{\mathbf{x}}$, is determined by:

$$\hat{\mathbf{x}} = \begin{bmatrix} \Phi_{true} - \delta\Phi + \delta\hat{\Phi} \\ \lambda_{true} - \delta\lambda + \delta\hat{\lambda} \\ h_{true} - \delta h + \delta\hat{h} \end{bmatrix} \tag{4.11}$$

where

$$\delta\Phi = \delta\theta_y \sin\alpha_t - \delta\theta_x \cos\alpha_t \tag{4.12}$$

$$\delta\hat{\Phi} = \delta\hat{\theta}_y \sin\hat{\alpha}_t - \delta\hat{\theta}_x \cos\hat{\alpha}_t \tag{4.13}$$

$$\delta\lambda = (\delta\theta_y \cos\alpha_t + \delta\theta_x \sin\alpha_t)\sec\Phi \tag{4.14}$$

$$\delta\hat{\lambda} = (\delta\hat{\theta}_y \cos\hat{\alpha}_t + \delta\hat{\theta}_x \sin\hat{\alpha}_t)\sec\hat{\Phi} \tag{4.15}$$

$$\delta h = \delta h \tag{4.16}$$

$$\delta\hat{h} = \delta\hat{h} \tag{4.17}$$

The velocity vector is also used in the dynamics matrix. The equation used for calculating the velocity vector follows:

$$\hat{\mathbf{v}}^t = \begin{bmatrix} V_{x_{true}} - \delta V_x + \delta\hat{V}_x \\ V_{y_{true}} - \delta V_y + \delta\hat{V}_y \\ V_{z_{true}} - \delta V_z + \delta\hat{V}_z \end{bmatrix} \tag{4.18}$$

where $\delta\hat{V}_x$, $\delta\hat{V}_y$, and $\delta\hat{V}_z$ are filter states. Since specific force errors are not a part of the model, the specific force vector utilized in the filter is obtained from the "true" trajectory data.

4.4.2 GPS Measurement Equations Two types of measurements are used to update the filter. These are pseudorange and delta-range information from the

GPSUE. The form of the filter calculated pseudorange ($\hat{R}_c$) equation is (20):

$$\begin{aligned} \hat{R}_c &= \sqrt{(\hat{X}_{sv} - \hat{X}_r)^2 + (\hat{Y}_{sv} - \hat{Y}_r)^2 + (\hat{Z}_{sv} - \hat{Z}_r)^2} \\ &\quad + \delta\hat{R}_{clk} + \delta\hat{R}_{atm} + \delta\hat{R}_{cl} \end{aligned} \tag{4.19}$$

$$= \hat{R}_t + \delta\hat{R}_{clk} + \delta\hat{R}_{atm} + \delta\hat{R}_{cl} \tag{4.20}$$

The model for the measured pseudorange (R_m) is:

$$R_m = \sqrt{(X_{sv} - X_r)^2 + (Y_{sv} - Y_r)^2 + (Z_{sv} - Z_r)^2} + \delta R_{clk} + \delta R_{atm} + \delta R_{cl} + v \tag{4.21}$$

where v is the measurement noise, and:

$$\begin{aligned} E[v(t_i)] &= 0 \\ E[v(t_i)^2] &= 1026 \text{ ft.}^2 \\ E[v(t_i)v(t_j)] &= 0 \ i \neq j \end{aligned}$$

The value of the measurement noise strength is taken from Martin's paper on GPSUE error models (15:118). The pseudorange calculation provides a measurement equation of the form:

$$z = h(x,t) + v \tag{4.22}$$

The update equation requires that $h(x,t)$ be linearized about the current best estimate of the states. Expanding the pseudorange calculation in a Taylor series and truncating to first order yields:

$$\begin{aligned} \hat{R}_c &= \hat{R}_t + \frac{\hat{X}_{sv} - \hat{X}_r}{\hat{R}_t}\delta\hat{X}_{sv} + \frac{\hat{Y}_{sv} - \hat{Y}_r}{\hat{R}_t}\delta\hat{Y}_{sv} + \frac{\hat{Z}_{sv} - \hat{Z}_r}{\hat{R}_t}\delta\hat{Z}_{sv} \\ &\quad - \frac{\hat{X}_{sv} - \hat{X}_r}{\hat{R}_t}\delta\hat{X}_r - \frac{\hat{Y}_{sv} - \hat{Y}_r}{\hat{R}_t}\delta\hat{Y}_r - \frac{\hat{Z}_{sv} - \hat{Z}_r}{\hat{R}_t}\delta\hat{Z}_r \\ &\quad + \delta\hat{R}_{clk} + \delta\hat{R}_{atm} + \delta\hat{R}_{cl} \end{aligned} \tag{4.23}$$

$$
\begin{aligned}
&= \hat{R}_t + U\widehat{LOS}_x\delta\hat{X}_{sv} + U\widehat{LOS}_y\delta\hat{Y}_{sv} + U\widehat{LOS}_z\delta\hat{Z}_{sv} \\
&\quad -U\widehat{LOS}_x\delta\hat{X}_r - U\widehat{LOS}_y\delta\hat{Y}_r - U\widehat{LOS}_z\delta\hat{Z}_r \\
&\quad +\delta\hat{R}_{clk} + \delta\hat{R}_{atm} + \delta\hat{R}_{el}
\end{aligned} \tag{4.24}
$$

The position errors for the receiver, which are shared with the INS model, are in the true frame, and the measured pseudorange is in the ECEF frame. Therefore, the position errors must be transformed to the ECEF frame for the calculation of the estimated pseudorange. The transformation takes the following form:

$$
\begin{bmatrix} Y_r \\ X_r \\ Z_r \end{bmatrix} = \mathbf{C}_t^l \begin{bmatrix} -(R_e+h)\delta\theta_x \\ (R_e+h)\delta\theta_y \\ \delta h \end{bmatrix} \tag{4.25}
$$

where (R_e+h) is the local distance to the center of the earth at the current position. Note that X_r and Y_r appear to be interchanged. This accounts for the position error states' rotations about the axis rather than distances. The multiplication by (R_e+h) translates the rotations into distances. This yields an **H** array, for satellite number one, of the form:

$$
\begin{aligned}
\mathbf{H} = [& -U\widehat{LOS}_x{}^t \; -U\widehat{LOS}_y{}^t \; -U\widehat{LOS}_z{}^t \; 0 \cdots 0 \\
& 1 \; 0 \; 1 \; 1 \\
& U\widehat{LOS}_x{}^e \; U\widehat{LOS}_y{}^e \; U\widehat{LOS}_z{}^e \; 0 \cdots 0 \;]
\end{aligned} \tag{4.26}
$$

The other three satellite updates would differ from (4.26) in the placement of the last two non-zero terms on the second row and three non-zero terms in the last row of the equation (composed of twelve terms in all). The elements shown as $U\widehat{LOS}_i{}^t$ are meant to include all transformations which take the INS position errors in the

true frame to the proper dimensions in the ECEF frame.

The delta-range (DR) is a measure of the range rate over the past time interval. The delta-range is measured by counting the carrier cycles over a period of time (15:110). Thus, the delta-range is the change in pseudorange over a particular time interval rather than the rate of change of pseudorange. The range rate equation is derived by differentiating the true range (R_t) equation. This results the equation:

$$\dot{R}_t = \frac{(X_{sv} - X_r)(V_{X_{sv}} - V_{X_r})}{R_t} + \frac{(Y_{sv} - Y_r)(V_{Y_{sv}} - V_{Y_r})}{R_t} + \frac{(Z_{sv} - Z_r)(V_{Z_{sv}} - V_{Z_r})}{R_t} \tag{4.27}$$

(4.28)

This derivation assumes the laboratory is not experiencing roll rates or high g maneuvers. Thus, for this specific case, the model for the measured delta-range is approximated by the range rate equation:

$$DR \approx \dot{R}_m \tag{4.29}$$

$$\approx \frac{(X_{sv} - X_r)(V_{X_{sv}} - V_{X_r})}{R_m} + \frac{(Y_{sv} - Y_r)(V_{Y_{sv}} - V_{Y_r})}{R_m} + \frac{(Z_{sv} - Z_r)(V_{Z_{sv}} - V_{Z_r})}{R_m} + \dot{\delta R}_{clk} + v \tag{4.30}$$

where

$$E[v(t_i)] = 0$$

$$E[v(t_i)^2] = 0.16 \text{ ft.}^2$$

$$E[v(t_i)v(t_j)] = 0 \quad i \neq j$$

and $\dot{\delta R}_{clk}$ is the user clock drift (the derivative of the user clock bias). Again, the measurement noise is obtained from Martin's paper (15:118). The equation for filter calculation of the delta-range is similar to that of the measured delta-

range. Substituting $\hat{R}_c$ for R_m and using the filter estimates of the other values in Equation 4.30 yields the filter calculated delta-range.

Expanding $\dot{R}_c$ in a Taylor series, and truncating to first order yields:

$$\begin{aligned}\dot{\hat{R}}_c &= \dot{\hat{R}}_t + \frac{(\hat{\mathbf{V}}_{sv} - \hat{\mathbf{V}}_r)^T}{\hat{R}_t}[\mathbf{I} - \widehat{\mathbf{ULOS}}^T \widehat{\mathbf{ULOS}}](\delta\hat{\mathbf{x}}_{sv} - \delta\hat{\mathbf{x}}_r) \\ &\quad + \widehat{\mathbf{ULOS}}(\delta\hat{\mathbf{v}}_{sv} - \delta\hat{\mathbf{v}}_r) + \delta\dot{\hat{R}}_{clk}\end{aligned} \tag{4.31}$$

The transformation necessary for the pseudorange is also used here to transform the position errors into the proper dimensions. A transformation of the velocity errors to the ECEF frame is also needed. For the velocity errors, the direction cosine matrix to transform vectors in the true frame to the ECEF frame can be used directly. The elements of the update matrix **H** are not shown directly here; however, they can be determined from Equation (4.32).

To provide stability to the stand alone INS solution it is necessary to provide barometric altitude measurement updates to the filter. The form for the calculated altitude follows:

$$h_c = h_{INS} + \delta\hat{h} \tag{4.32}$$

The measured altitude is computed as the altitude of the navigation laboratory (approximately 964 feet). The form for the measured altitude is:

$$h_m = h_{actual} + v \tag{4.33}$$

where

$$\begin{aligned}E[v(t_i)] &= 0 \\ E[v(t_i)^2] &= 0.0001 \text{ ft.}^2 \\ E[v(t_i)v(t_j)] &= 0 \ i \neq j\end{aligned}$$

This measurement noise may not be correct for an actual altimeter, but it is indicative of the accuracy of the laboratory altitude measurement. The **H** matrix for this update consists of zeros except for the element associated with state x_{10}, δh.

4.4.3 Integrated GPS/INS Simulation The truth model is programmed into the system portion of MSOFE, and the dynamics noise is programmed to be injected into the proper system states. Also, the update equations are programmed into the system and filter parts of MSOFE. With this accomplished, a simulation is performed to predict the filter performance.

The simulation consists of an eight-minute alignment followed by thirty minutes of static navigation. The filter is updated every ten seconds by both pseudorange and delta-range measurements during static navigation, and the altitude measurements are incorporated every two seconds throughout the simulation. Plots are obtained for the measurement residuals, the position and velocity estimation errors, and the satellite state estimation errors and are presented in Figures 4.2-4.7. All channels exhibit the same behavior characteristics; hence, the satellite state estimation errors are shown for one channel. The user clock state estimation errors are not presented because they start out with large magnitudes, quickly drop to zero and stay nearly zero. In the residual plots, the solid lines are the residuals, and the dashed lines are the filter calculated 1σ values for the residual. In the estimation error plots, the solid lines are the error, and the dashed lines are the estimated 1σ bounds on the error. The estimation errors are the difference between the "true" system values and the values for those quantities as estimated by the filter.

The pseudorange residuals, shown in Figure 4.2, are approximately zero mean and stay within a reasonable multiple of the calculated 1σ value. Residual monitoring was utilized with a bound of 10σ, and no measurements were discarded. The plots display a quantization of the residuals, in that all residuals are multiples of four (e.g. 4, 8, -32). This is attributable to two phenomena: single precision variables

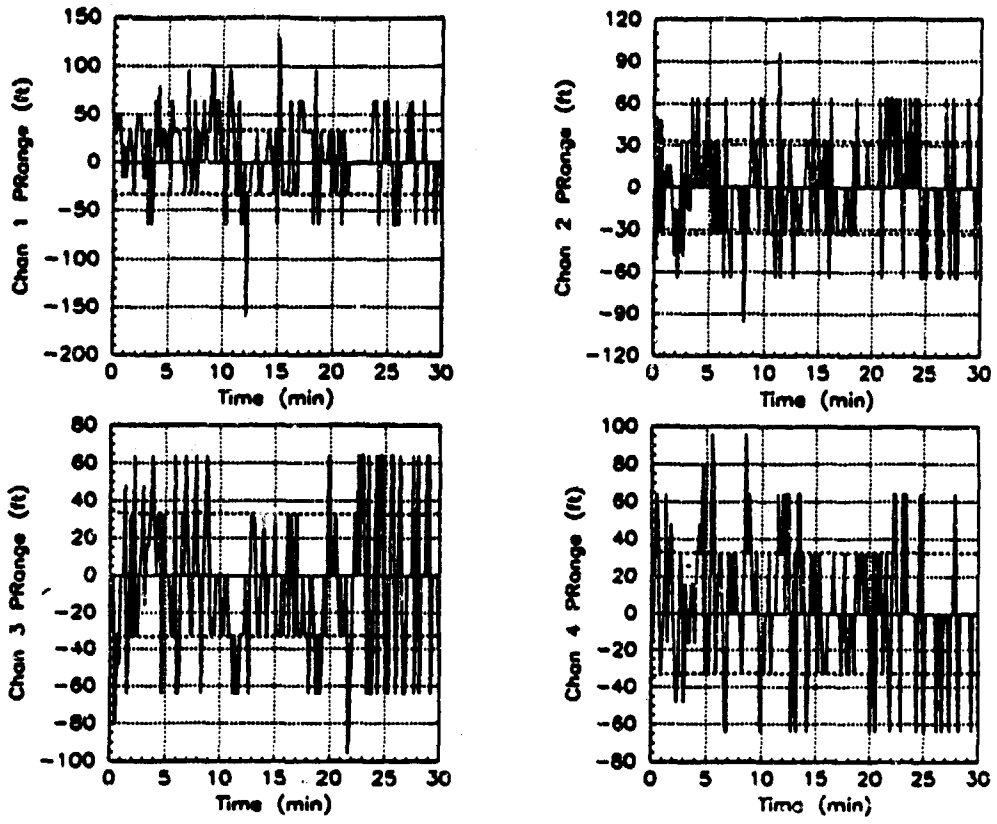


Figure 4.2. Pseudorange Residuals

being used for measurements and residuals being the difference of small numbers. The measurement values for pseudoranges are sufficiently large that, with single precision variables, the precision is increased to the units magnitude. Thus, the least residual is expected to have a value of one. With the way data is stored in a digital computer, as binary numbers, the accuracy of the single precision variables may actually be decreased to the point where the least significant bit actually represents a magnitude of four. This accounts for the residuals appearing quantized as multiples of four. The decision is made to continue in single precision because this does not appear to be affecting the simulation in a negative way.

Delta-range residuals are presented in Figure 4.3. The residuals appear to be nearly zero-mean although some plots tend to be slightly biased. They also remain within the 10σ bound for residual monitoring. However, the delta-range calculations appear to be less accurate than originally anticipated; so, the measurement noise is

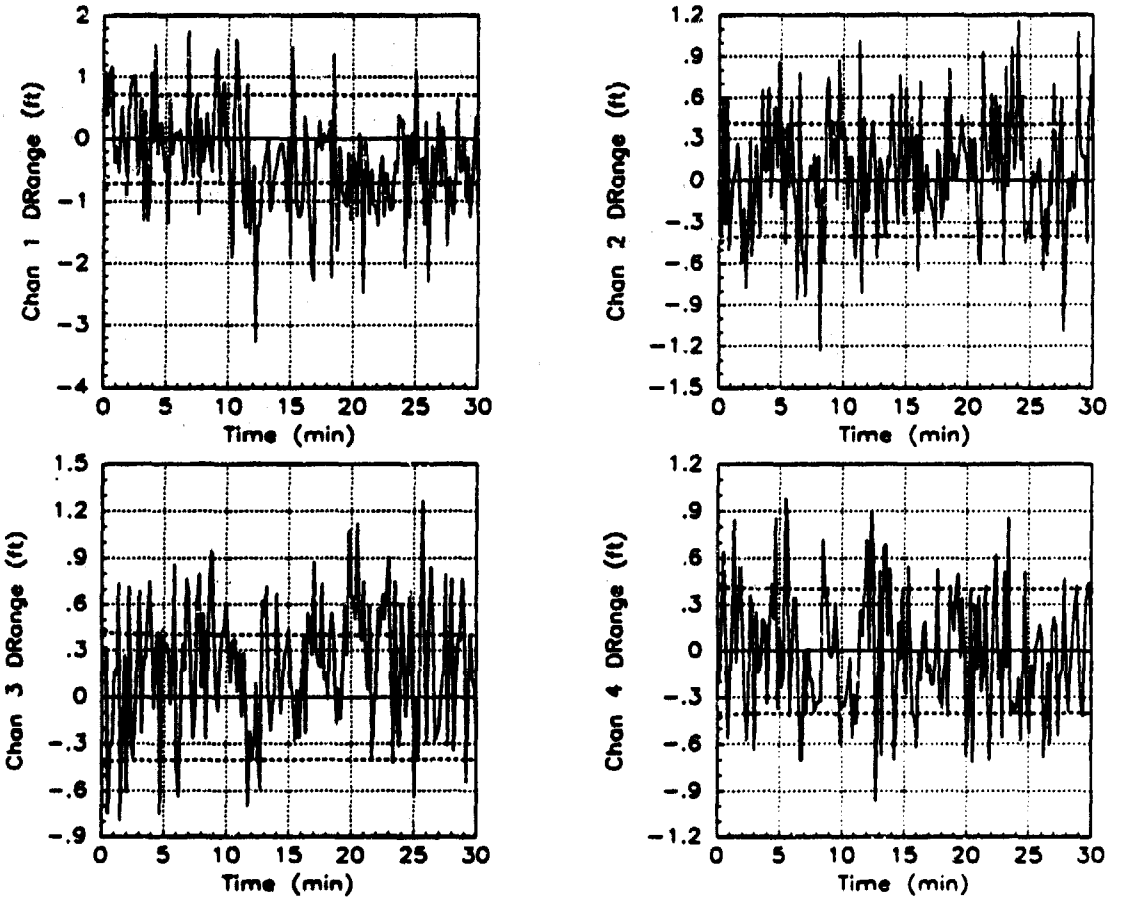
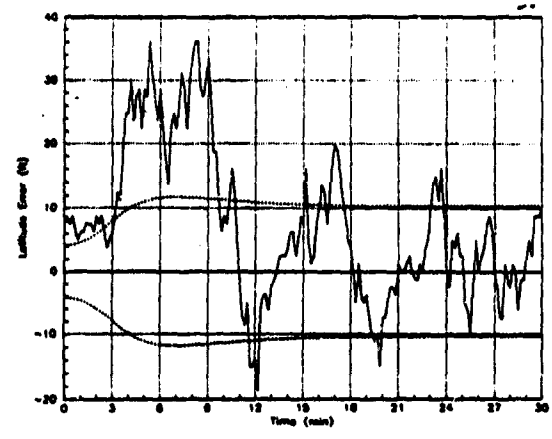
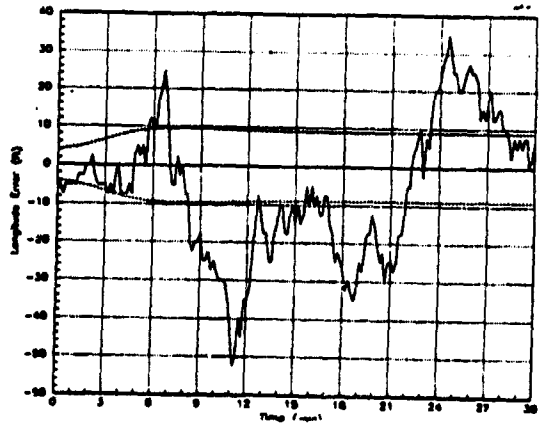


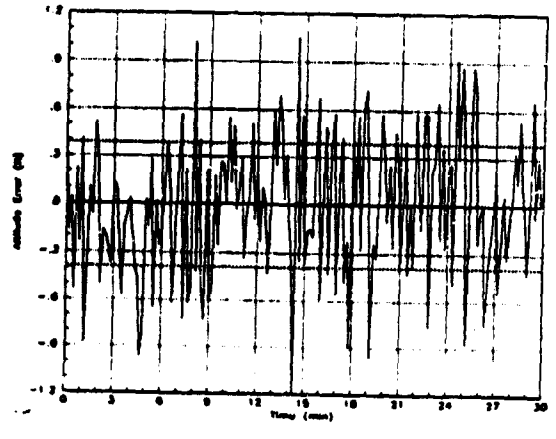
Figure 4.3. Delta-Range Residuals



(a)

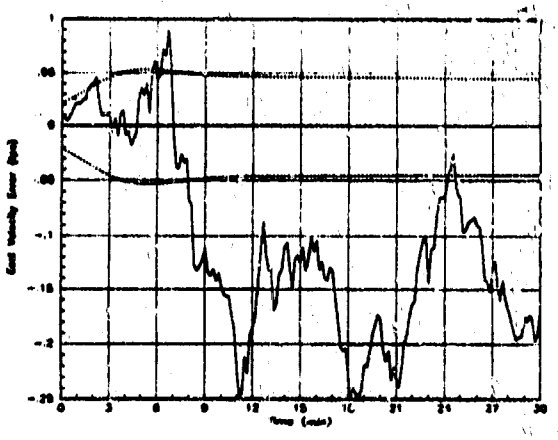


(b)

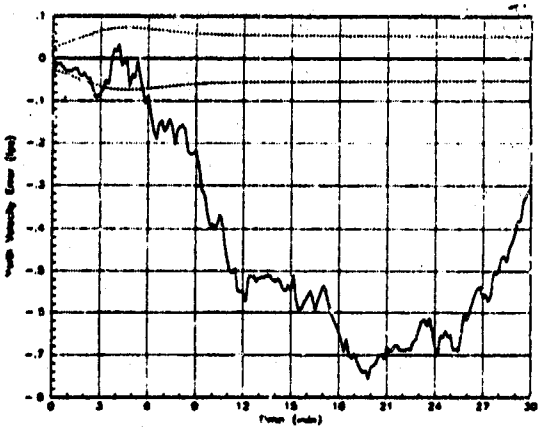


(c)

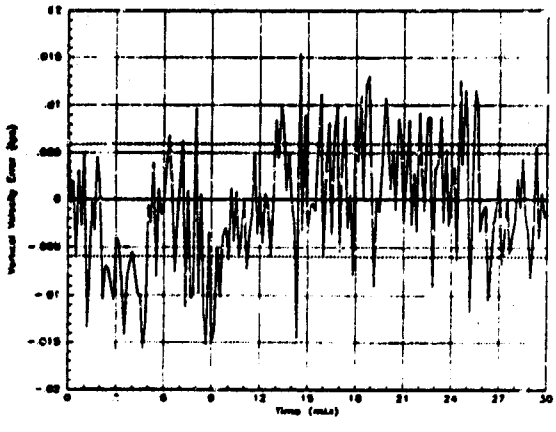
Figure 4.4. User Position Estimation Errors



(a)

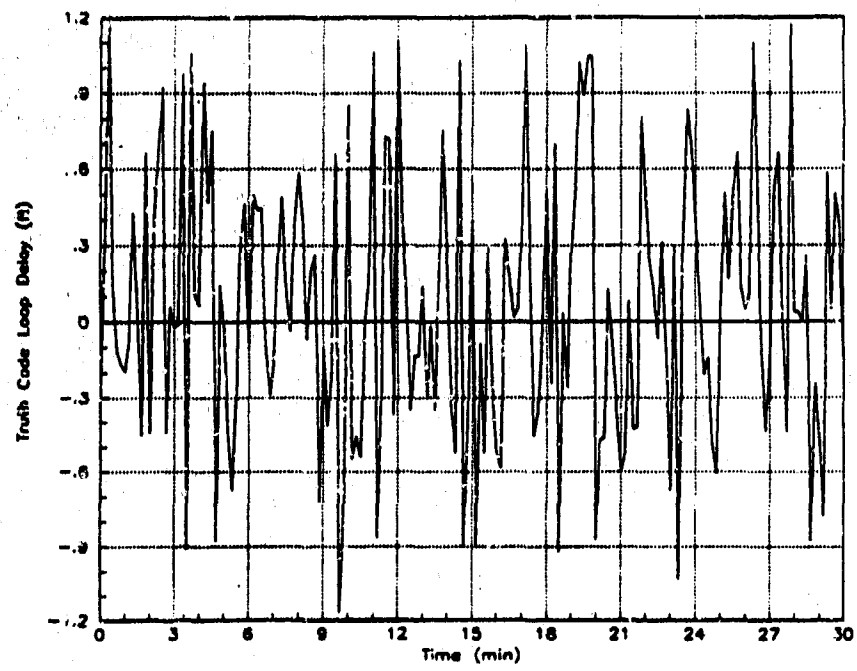


(b)

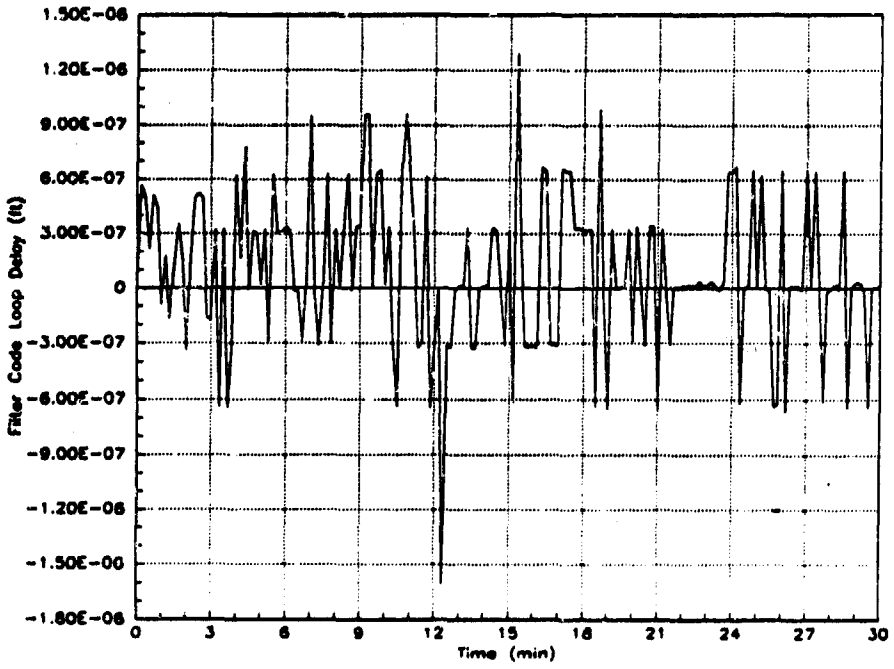


(c)

Figure 4.5. User Velocity Estimation Errors



(a)



(b)

Figure 4.6. Code Loop Delay (a) System (b) Filter

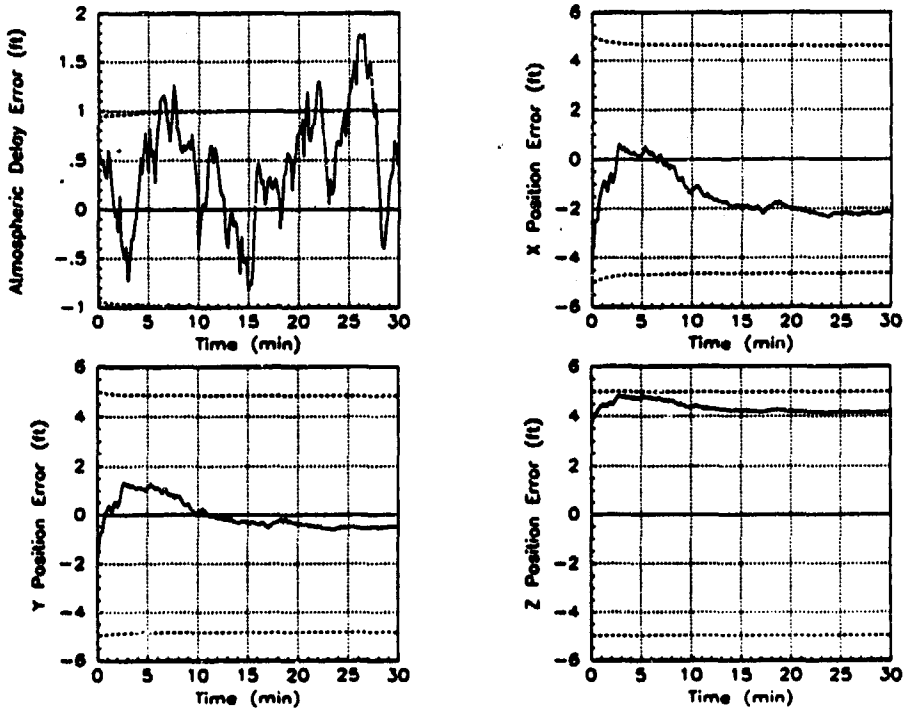


Figure 4.7. Atmospheric and Satellite Position Estimation Errors

increased to 0.5 ft.^{2} for the delta-range update. The slight bias being shown in two of the plots is likely due to having performed only one Monte Carlo simulation. The biases should disappear with more simulations averaged and should be investigated later.

Figure 4.4 presents the user position estimation errors. The longitude error remains within 60 feet of zero and shows no particular bias over the whole time period. The latitude error takes approximately ten minutes before settling about zero error. The altitude error oscillates rapidly about zero as the result of the high accuracy altitude updates. The error being displayed is basically reflecting the residuals from the altitude updates. Some slight biases over small time intervals may be noted in both the latitude and longitude plots. These are again decided to be the result of having performed only one simulation. More simulation runs should be averaged to determine the correctness of this statement.

The horizontal velocity errors, shown in Figure 4.5, exhibit a distinct bias. The vertical velocity error is zero-mean, and again it reflects the high accuracy used for the altitude updates. The biases shown in the horizontal velocities could be the result of having run only one simulation, it could show something wrong with the delta-range updates, or it could show the effect of the high accuracy altitude updates pushing vertical errors into the horizontal plane. These are areas for further study. At this point, the errors are within expected values for the velocity errors and the decision is made to continue.

The code loop delay is plotted with two separate plots because of the disparity in size. The system delay is behaving as a first order Markov process with a one second time constant is expected to behave. However, the filter does a very poor job of estimating the code loop delay. This indicates a couple of possibilities: the state is almost unobservable, the driving noise is too small, or the update rate is insufficient for the filter to get a good estimate of this error. Since the driving noise is the same for the system and filter, the second possibility is not considered. The

other two possibilities are areas for further study. The decision is made to continue as the error contributed by the code loop is not so large that compensation can not be provided in other states.

The remaining four states are shown in Figure 4.7. The atmospheric error displays a slight bias. This could be the result of the poor code loop delay estimation or of having performed only one simulation. The satellite position errors display biases in the X_l and Z_l directions. A bias may be found in any of the three position errors for each channel. This is attributable to using stationary satellites during static navigation. Sufficient observability of the satellite position errors is not available when utilizing one degree of freedom (i.e., line-of-sight vector to satellite is constant). With these small errors, the decision is made to proceed to using empirical data.

4.5 Summary

The GPSUE error model is presented and validated. All states behave as expected. The GPS model is combined with the INS model, and pseudorange, delta-range, and altitude updates are derived. A single Monte Carlo run is performed and the results are analyzed. Some small errors are noted. Most of these are attributed to having performed a single-sample Monte Carlo simulation. Some minor biases are observed but are deemed small enough to justify implementing the filter with empirical data.

V. *Kalman Filter Performance with Empirical Data*

Data collection and interpretation as well as MSOFE modifications are necessary to analyze the performance of the Kalman filter with empirical data. The LN-94 is allowed to align for eight minutes before switching to navigation mode and collecting the data. Following the INS alignment, data is collected from both the GPSUE and the LN-94.

5.1 Obtaining and Interpreting Data

The data collection and interpretation requires many steps and the use of two computer systems, an IBM PC-AT and a Sun 3 Workstation. The personal computer is used to collect data from the GPS receiver through the instrumentation port, and the workstation is used to obtain INS information on the MIL-STD-1553 bus.

5.1.1 GPS Data Collection Data is collected from GPSUE through the instrumentation port using PC Buffer Box software on an IBM PC-AT. PC Buffer Box consists of two pieces of software, the data collection software (BB.EXE) and the post processing software (PP.EXE). Information on this software is found in the PC Buffer Box User Manual (11). BB.EXE is menu driven and very user friendly. Proceeding through the menu, a Phase III receiver 3A is chosen for the receiver type. Using ICD-GPS-215 (3), the decision is made to request block 1022 to obtain the pseudorange and delta-range information and blocks 1026 and 1027 for satellite position information. The data collection is set to occur for twenty minutes, the file to store the data is given a name, and the data collection is started.

In order for the post processor to interpret the data correctly, a block definition file is programmed and given the name BXXXXX.BDF, where XXXXX is a five-digit number corresponding to the block being processed. The block definition file for block 1022 (named B01022.BDF) is shown in Appendix E. The first column is

the name given to each piece of data in the block. The second column is the block number; it is always 1022 for this block. The third column is the word count within the block at which the corresponding data starts. The word length of the data is in column four, and column five indicates the data type: R for floating point, I for integer, and CF for Collins CAPS floating point.

Following the data collection and block definition file setup, PP.EXE is executed to interpret the information. Selecting printer listings and block 1022 using the menus, the GPS time, satellite numbers, measurement type, and measurement value are printed to a file for each channel. The four files thus obtained are transported to the Sun 3 Workstation, and MSOFE is edited to read in the information for the performance of measurement updates.

A similar exercise is repeated to transport the satellite position information to the Sun 3 Workstation and edit MSOFE to read the four files containing the satellite positions. These files actually contain the line-of-sight vectors to the satellites being utilized. Thus, a fifth file is generated to obtain the user position. MSOFE is modified to read this file and add the user position vector at each time to the satellite line-of-sight vectors to obtain satellite positions. Also, MSOFE is modified to ignore any data which is not indicated to be a pseudorange or delta-range measurement. This indication is provided in block 1022 and is included in the files containing the measurements. The main reasons for this are that the receiver does not provide a pseudorange and delta-range update for a channel at any particular time when either fewer than four satellites are visible or the channel is changing to receive information from a different satellite.

5.1.2 *INS Data Collection* Position, velocity, acceleration, wander angle, body attitude, and attitude rates are necessary to calculate the dynamics matrix of the filter. The data is obtained from the LN-94 through the MIL-STD-1553 bus utilizing the Sun 3 Workstation. The BCU/VME-1014 MIL-STD-1553 commu-

nications card made by SCI Technology, Incorporated was installed on the Sun 3 Workstation by Mr. Bruce Clay of Systems Research Laboratory (SRL). Information on the software necessary to do this is found in the BCU-VME-1014 Operations Manual (19). The software written by Mr. Clay allows the workstation to act as a bus controller. In the bus controller mode, the communications card collects all data output on the bus in response to commands sent by it.

To collect the data it is necessary to know the remote terminal number of the device, the subaddress numbers which contain the required data, and the number of thirty-two bit words in each subaddress. The LN-94 is hard-wired to be remote terminal number five. Utilizing FNU-85-1 (1), the decision is made to request subaddress numbers one and sixteen. Altitude information is supplied to the INS using subaddress four. Subaddress one is twenty-two words long, subaddress four is two words in length, and the wordlength of subaddress sixteen is thirty-two (1:257,260-262). The execution blocks are programmed as specified in the BCU-VME-1014 Operations Manual to send the altitude information subaddress and request subaddresses one and sixteen. A parameter block is set to output the messages at a twenty Hertz rate. Mr. Clay's program is modified to save the requested data every ten seconds for twenty minutes. Also, the data is scaled to the proper units, utilizing the information in FNU 85-1, before being saved. The INS is initialized to the laboratory's position and allowed to align for eight minutes. The program to collect the data is begun when the INS is set to the navigation mode.

The program to obtain the data from the LN-94 is written in the C programming language. To utilize the data with MSOFE, the data must be saved in an unformatted FORTRAN readable file and in the same order as the data supplied by PROFGEN for the fighter flight profile. To this end, a FORTRAN program was written which read the formatted data as output by the C program and wrote an unformatted file readable by MSOFE. MSOFE is modified so as not to read header information from the file FLIGHT since no header information is provided in the

empirical INS data.

5.2 Kalman Filter Evaluation

With the data collected and interpreted, MSOFE is modified to read the GPS measurement and satellite position files, to utilize the data to perform the measurement updates, and to calculate estimates of satellite velocities at the update times. An eight-minute alignment of the INS is simulated with no GPS updates, followed by filter operation for twenty minutes of GPS pseudorange and delta-range updates and altitude updates.

5.2.1 Filter Operation On the first attempt with empirical data, the filter fails to update with all four satellites. The residual monitoring built into MSOFE accepts the first channel update and rejects the rest. Subsequently, the residuals for all satellites become worse over time.

As a first attempt to identify the problem, the measurement noise covariance in the filter is increased for all measurements. This results in allowing the filter to update with all measurements. The residuals are reduced in magnitude but remain biased on the order of 10^6 feet. This indicates that there is some unaccounted error which affects each satellite's measurements individually.

At this point, the delta-range updates are dropped. The filter can perform adequately without the delta-range updates, and some problems experienced in the filter simulations point out that incorrect delta-range calculations can seriously degrade performance. The delta-range update is thus dropped for the first iteration to eliminate a possible source of additional errors. Also, the simulation time is decreased to five minutes during which four satellites are used for most of the time period. This allows for testing the filter without introducing the increased errors when fewer than four satellites are in use.

Having applied some simple fixes to decrease the modeling error, the true

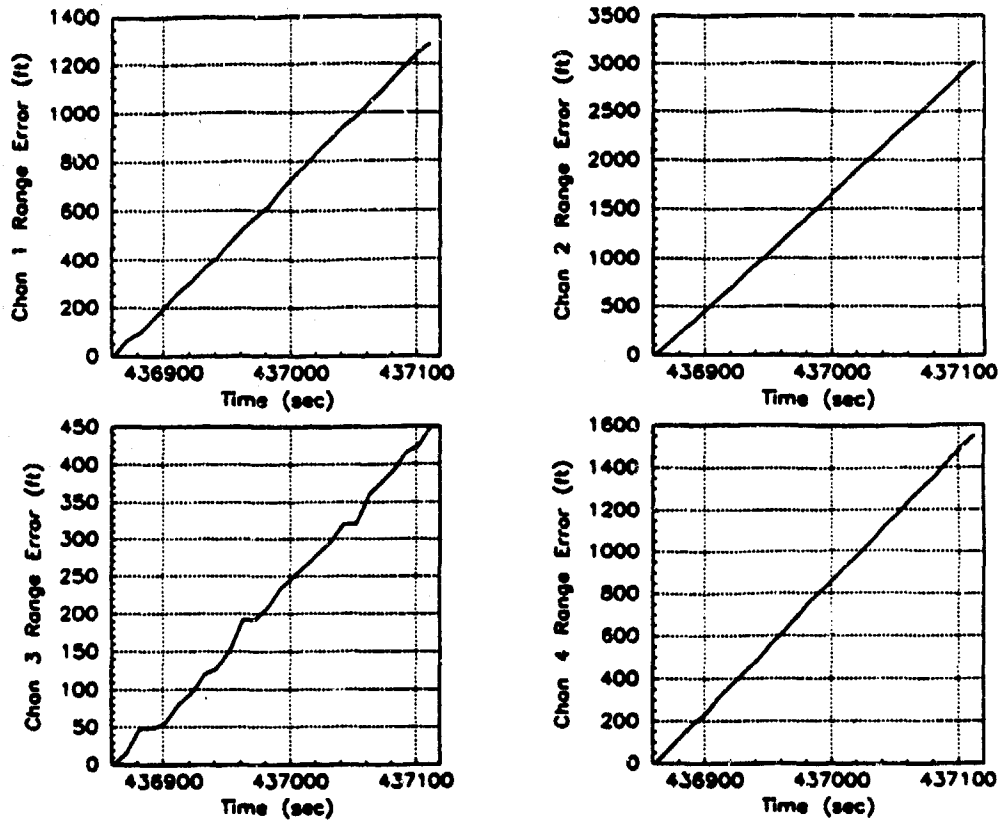


Figure 5.1. Range Drifts Adjusted for Initial Biases

ranges to the satellites are computed and compared to the pseudorange measurements supplied by the receiver. The true range is subtracted from the pseudorange, and the initial biases are subtracted out so that the numbers involved are smaller and provide easier analysis results. The results of this comparison are shown in Figure 5.1. The initial biases for channels one through four are: -77,381,625, -76,523,193, -76.090,289, and -76,518,473 feet, respectively. It is seen that there is a large discrepancy among the biases, even to the millions place in one instance. From Figure 5.1, it is obvious that the biases do not have the same drift either. The plot for channel three also noticably varies in ways which are not attributable to either a bias or a drift.

The errors in evidence in Figure 5.1 are distinctly biases with a drift. The drift is apparent from the errors growing in a straight line. However, the biases and drift rates are different for each satellite. This suggests a phenomenon which is separate

among the satellites. Also, the plots are not perfectly straight lines. Because of the scales used in the plots, channel three displays the slight nonlinearity better than the other channels. This indicates a varying which may be attributable to atmospheric delay. The code loop delay has a one second time constant; thus, it should not be causing changes which evidence themselves after twenty seconds or more. Hence, the error is attributed to atmospheric delay error. Although there is the possibilty that some unmodeled error is causing the variations. The source of this error is a subject for further investigation

The Global Positioning System is currently functioning with selective availability enabled at unannounced times. Selective availability is a method to degrade the accuracy of non-friendly receivers by denying access to the P-code. Selective availability results in larger residuals and less accurate position and velocity estimation than in a full availability situation. This causes the filter performance to be degraded from the predictions and is an unvalidated source of substantial error.

The pseudorange and delta-range measurements and the satellite positions provided by the GPSUE are output with an associated GPS time. This time tagging of the information on the two different busses is provided at one second intervals. It is possible that the time tagging of the measurements is performed differently than the time tagging of the satellite position data. Also, the satellite positions may have been calculated at different times within the one second interval. Time differences as small as a millisecond have a significant effect on the value of the pseudorange. This is another area for further study.

The first attempt to correct the error is to add bias and drift states to each satellite model. The attempt is suggested from the appearance of the error as a bias and drift unique to each satellite and lack of information about any other processes which display this characteristic. This effort reduces the residuals with the measurement noise values retained at the values used in the simulation. However, the residuals increase slowly with time.

The next attempt to correct the problem is to reduce the atmospheric error time constant to 100 seconds. The decrease in the time constant is suggested partially by the errors displayed in plots of the range errors. Also, the length of time before the residuals begin to grow larger appears to be between one and two minutes. The results from this attempt are shown in Figure 5.2. The residuals are slightly better than in previous attempts; however, the position errors tend to ramp away from zero. Also, the residuals still tend to grow with time. Particularly, the residuals in channels one and three display a noticable ramp at the beginning of data processing.

Further attempts to obtain an operational filter are made by varying the atmospheric error time constant, the atmospheric error driving noise strength, and the measurement noise variance. Results from the best of the attempts are presented in Figure 5.3. The atmospheric time constant for this attempt is 50 sec. The measurement noise variance is reduced to 250 ft.^{2}, and the driving noise strength is set to 0.03 ft.^{2}/sec. The residuals remain within a reasonable range around zero for the entire time period. With selective availability being used, residuals with magnitudes around two hundred feet are not unexpected. However, the latitude and longitude errors still ramp away from zero. The error in the position estimation increases at a slower rate than with the atmospheric time constant at 100 sec. This is an indication that the tuning is proceeding in the proper direction; however, the intent is to keep the position errors from growing at all. It may be necessary to separate the ionspheric delay from the atmospheric delay state. Using different time constants for the two phenomena may reduce the growth of the latitude and longitude errors even further. It is evident that more work is needed in analyzing the errors in the pseudoranges.

5.3 Summary

Pseudoranges, delta-ranges, and satellite positions are obtained from the GPS receiver through the instumentation port, and MSOFE is modified to utilize the

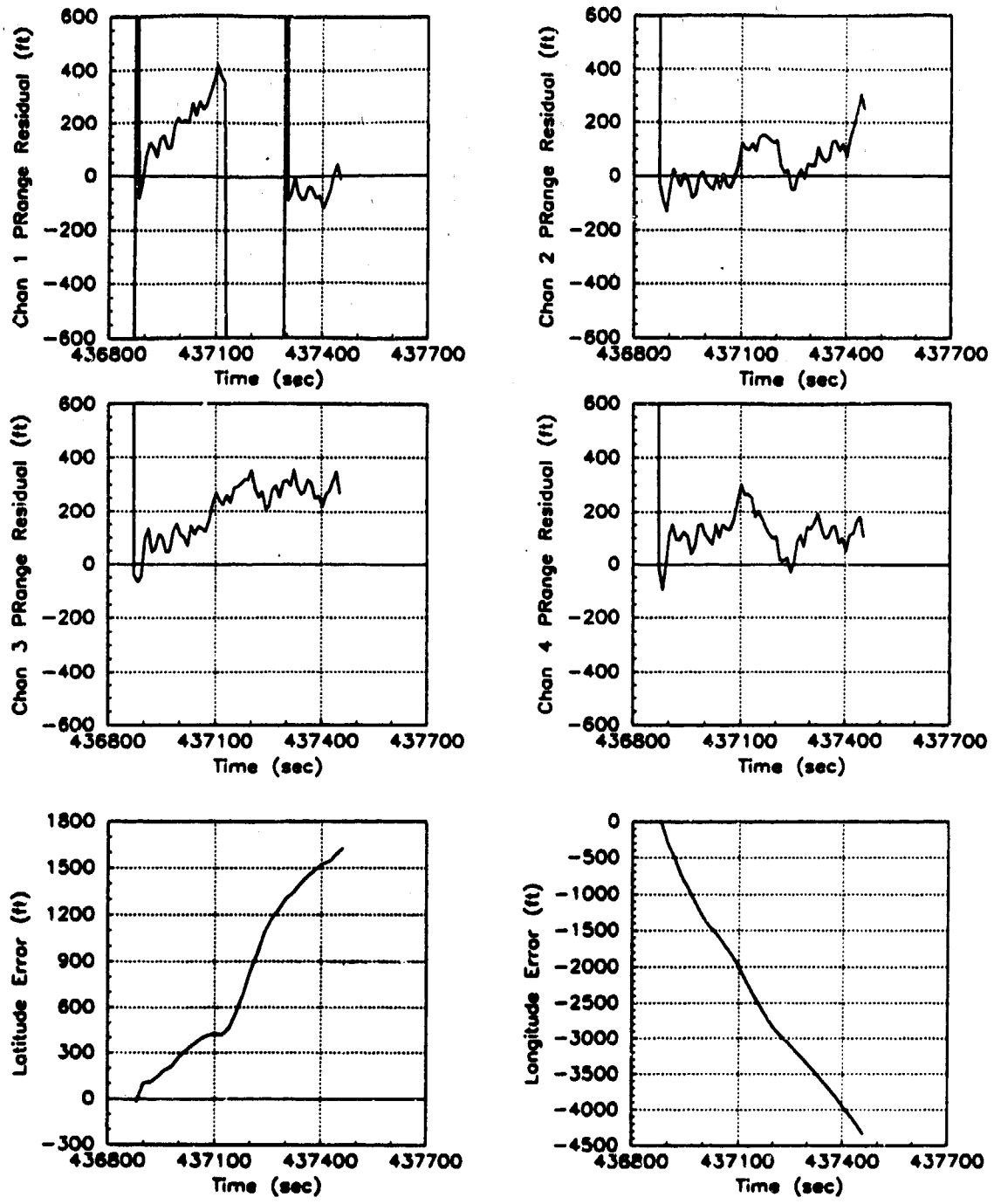


Figure 5.2. Residuals and Position Errors for Atmospheric Time Constant 100 sec.

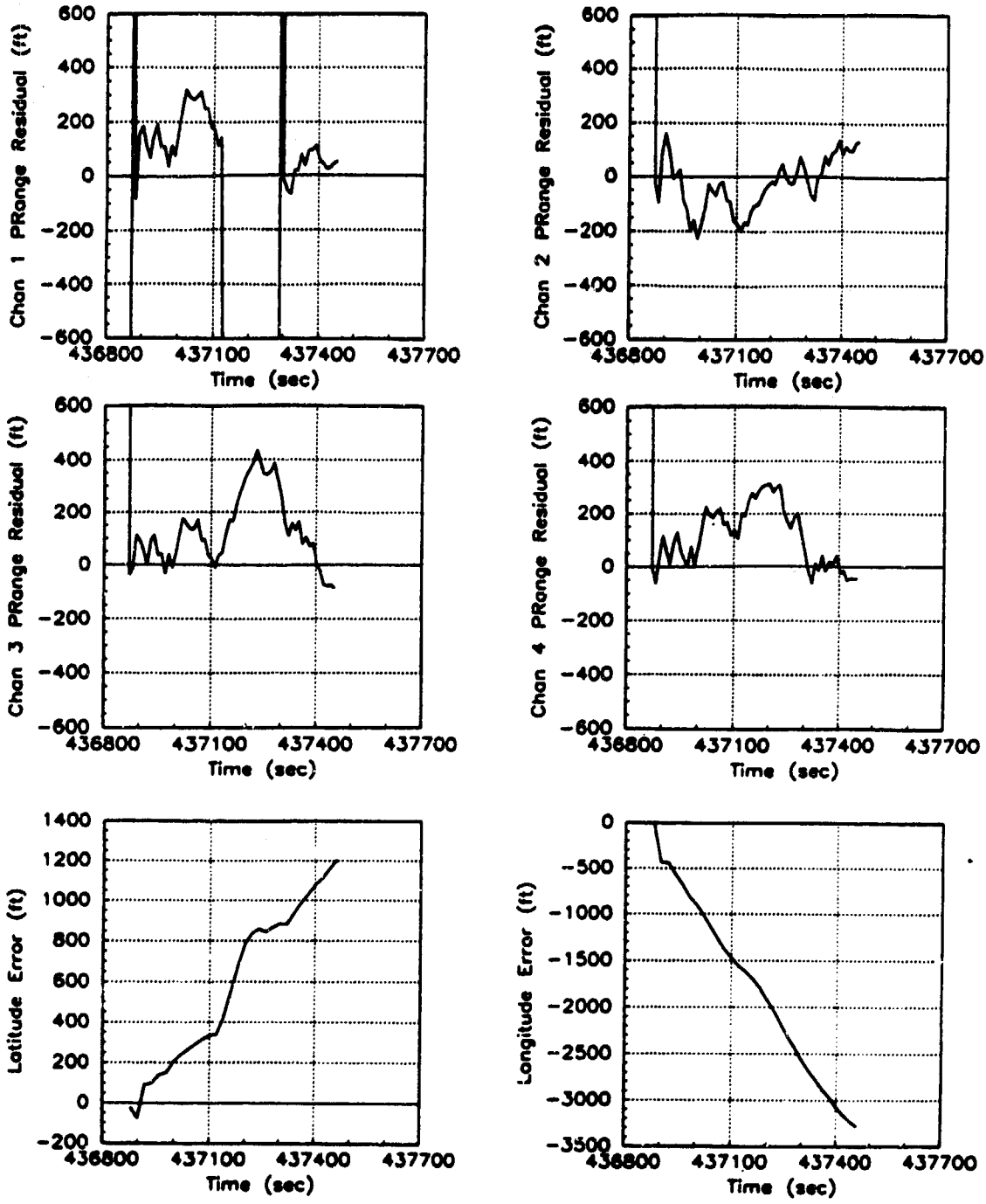


Figure 5.3. Residuals and Position Errors for Atmospheric Time Constant 50 sec.

information. Empirical data is collected from the LN-94 following an eight-minute alignment. Data is obtained from the GPSUE and INS at the same time. The INS data is put in a file in the same form as used by PROFGEN to provide FLIGHT files, so MSOFE only needs to be modified to the extent that no header information is read from the FLIGHT file. The filter is allowed to perform with real data, and errors are noted. The errors are analyzed, and some attempts to correct for the errors are made.

VI. Conclusions and Recommendations

All stated research objectives are attempted; most are successful, some require additional work. Conclusions drawn from the research and recommendations for further research are discussed.

6.1 Conclusions

The overall objective stated in the first chapter is to integrate GPS measurements with INS outputs. The first step is to model the two systems and combine them in a single Kalman filter. The INS model is obtained from Litton, programmed into MSOFE, and validated. With some small anomalies, this is accomplished. The anomalies are attributable to having performed a covariance analysis; whereas, the basis of comparison is the set of sample statistics obtained by averaging over ten Monte Carlo simulations. The INS truth model is determined to be sufficiently correct. The GPS model is a modified version of that developed by Capt Solomon (20). The major difference is the use of a single user clock error rather than separate clock errors for the separate satellites. The GPS model is programmed into MSOFE, and the states behave as predicted. Thus, the GPS model is deemed to be correct.

The next step is to combine the two models and predict the performance of the joint Kalman filter processing empirical data. A simulation of the full state filter running against the truth model is performed to this end. The residuals from the pseudorange and delta-range updates remain within expected bounds, and the position and velocity errors also remain relatively close to zero. Thus, the filter is determined to be correctly implemented. Inability of the GPS model to obtain good estimation of the code loop error indicates that this state may be a candidate for removal when filter reduction is performed in later research. The estimation errors in the satellite position errors indicate a possibility to condense the three satellite

position states into one state along the current line-of-sight when filter reduction is performed.

The final step is to process empirical data through the filter. This step is not completely accomplished. The data is obtained to operate the filter, but filter performance is well below that which is predicted. Analysis of the data indicates that the filter could possibly be tuned more carefully to alleviate the problem, or that the GPS model may need to be changed to account for the errors observed. Some small changes are made to the filter which clean up the residuals reasonably, but the position errors remain unacceptable.

6.2 Recommendations

Some ideas for further research are presented below. These areas will provide useful information with which to obtain better filter performance with the empirical data.

6.2.1 Discover and Fix Problem with Pseudoranges It is possible that the ranges compared with the pseudoranges are being incorrectly calculated. If this is the case, discovering this problem and fixing it should clear up most of the errors being experienced.

6.2.2 Further Analyze the Discrepancies in the Data Other errors are in evidence than just the large bias problem. Analysis and resolution of these discrepancies will serve to improve filter performance further.

6.2.3 Tune the Filter Tuning the filter using the empirical data (corrected for errors if necessary) will provide improved filter performance. The closer the filter is to modeling the "real" world, the better the filter performance will be.

6.2.4 Reevaluate the GPS Error Model A reevaluation of the GPS error model to determine correctness will be useful. If the model is significantly different from what is actually occurring, filter performance will be degraded. This is a possible explanation for some of the problems encountered here.

6.2.5 Check Data Time Tagging If time tagging of the pseudorange and delta-range information is performed differently from the time tagging of the satellite positions, a significant error could result from even a difference in time as small as a millisecond. Closer examination of how the time tagging of the information is performed may reduce some of the errors evident here.

6.2.6 More Filter Simulations One Monte Carlo anaylsis was performed in the combined model prediction. Performing more Monte Carlo runs to validate the analysis in Chapter 4 may yield more insight to the problems encountered.

6.2.7 Horizontal Velocity Errors The horizontal velocity error biases displayed in the filter performance prediction need to be studied. The delta-range and altitude updates are possible sources for this error as stated in Chpater 4. Removing these biases may improve filter performance.

6.2.8 Code Loop Estimation The filter does a very poor job of estimating the code loop error. Further investigation as to the cause of this poor estimation should be considered. Possible sources for this problem are that the code loop delay state is unobservable, and that the driving noise strength on this state is too small.

6.2.9 Satellite Position Errors The filter also does a poor job of estimating the satellite position errors during performance prediction. This is attributed to both the satellites and the GPS unit being stationary during the simulation. More work needs to be done to verify that this is the case. Simulating either satellite or aircraft movement will verify whether or not this is true.

Appendix A. *INS Truth Model State Definitions*

The truth model is composed of 93 states that are all errors committed by the LN-94. The state number used in MSOFE, the state symbol as used in the Litton CDRL (14), and a short definition of each state are given here. The states are shown in tables which are broken into the state subvectors defined previously. (Note: The state symbol here, $\delta\phi_i$, is given by Litton as ϕ_i. The change here is to remove confusion between all the terms which are symbolized by ϕ in the Litton CDRL.)

Table A.1. Definition of INS Truth Model Subvector $\delta\mathbf{x}_1$

State Number	State Symbol	Definition
1	$\delta\theta_x$	X-component of vector angle from true to computer frame
2	$\delta\theta_y$	Y-component of vector angle from true to computer frame
3	$\delta\theta_z$	Z-component of vector angle from true to computer frame
4	$\delta\phi_x$	X-component of vector angle from true to platform frame
5	$\delta\phi_y$	Y-component of vector angle from true to platform frame
6	$\delta\phi_z$	Z-component of vector angle from true to platform frame
7	δV_x	X-component of error in computed velocity
8	δV_y	Y-component of error in computed velocity
9	δV_z	Z-component of error in computed velocity
10	δh	Error in vehicle altitude above reference ellipsiod
11	δh_L	Error in lagged inertial altitude
12	δS_3	Error in vertical channel aiding state
13	δS_4	Error in vertical channel aiding state

Table A.2. Definition of INS Truth Model Subvector $\delta \mathbf{x}_2$

State Number	State Symbol	Definition
14	b_{x_c}	X-component of gyro correlated drift rate
15	b_{y_c}	Y-component of gyro correlated drift rate
16	b_{z_c}	Z-component of gyro correlated drift rate
17	∇_{x_c}	X-component of accelerometer and velocity quantizer correlated noise
18	∇_{y_c}	Y-component of accelerometer and velocity quantizer correlated noise
19	∇_{z_c}	Z-component of accelerometer and velocity quantizer correlated noise
20	δg_x	X-component of gravity vector errors
21	δg_y	Y-component of gravity vector errors
22	δg_z	Z-component of gravity vector errors
23	δh_c	Barometer correlated bias noise error
24	b_{x_t}	X-component of gyro trend
25	b_{y_t}	Y-component of gyro trend
26	b_{z_t}	Z-component of gyro trend
27	∇_{x_t}	X-component of accelerometer trend
28	∇_{y_t}	Y-component of accelerometer trend
29	∇_{z_t}	Z-component of accelerometer trend

Table A.3. Definition of INS Truth Model Subvector $\delta \mathbf{x}_3$

State Number	State Symbol	Definition
30	b_x	X-component of gyro drift rate repeatability
31	b_y	Y-component of gyro drift rate repeatability
32	b_z	Z-component of gyro drift rate repeatability
33	S_{g_x}	X-component of gyro scale factor error
34	S_{g_y}	Y-component of gyro scale factor error
35	S_{g_z}	Z-component of gyro scale factor error
36	χ_1	X gyro misalignment about Y-axis
37	χ_2	Y gyro misalignment about X-axis
38	χ_3	Z gyro misalignment about X-axis
39	ν_1	X gyro misalignment about Z-axis
40	ν_2	Y gyro misalignment about Z-axis
41	ν_3	Z gyro misalignment about Y-axis
42	D_{xxx}	X gyro scale factor non-linearity
43	D_{yyy}	Y gyro scale factor non-linearity
44	D_{zzz}	Z gyro scale factor non-linearity
45	S_{Qb_x}	X gyro scale factor asymmetry error
46	S_{Qb_y}	Y gyro scale factor asymmetry error
47	S_{Qb_z}	Z gyro scale factor asymmetry error

Table A.4. Definition of INS Truth Model Subvector $\delta \mathbf{x}_4$

State Number	State Symbol	Definition
48	∇_{b_x}	X-component of accelerometer bias repeatability
49	∇_{b_y}	Y-component of accelerometer bias repeatability
50	∇_{b_z}	Z-component of accelerometer bias repeatability
51	S_{A_x}	X-component of accelerometer and velocity quantizer scale factor error
52	S_{A_y}	Y-component of accelerometer and velocity quantizer scale factor error
53	S_{A_z}	Z-component of accelerometer and velocity quantizer scale factor error
54	S_{QA_x}	X-component of accelerometer and velocty quantizer scale factor asymmetry
55	S_{QA_y}	Y-component of accelerometer and velocty quantizer scale factor asymmetry
56	S_{QA_z}	Z-component of accelerometer and velocty quantizer scale factor asymmetry
57	f_{xx}	Coefficient of error proportional to square of measured acceleration
58	f_{yy}	Coefficient of error proportional to square of measured acceleration
59	f_{zz}	Coefficient of error proportional to square of measured acceleration
60	f_{xy}	Coefficient of error proportional to products of acceleration along and orthogonal to acceleromter sensitive axis
61	f_{xz}	Coefficient of error proportional to products of acceleration along and orthogonal to acceleromter sensitive axis
62	f_{yx}	Coefficient of error proportional to products of acceleration along and orthogonal to acceleromter sensitive axis
63	f_{yz}	Coefficient of error proportional to products of acceleration along and orthogonal to acceleromter sensitive axis
64	f_{zx}	Coefficient of error proportional to products of acceleration along and orthogonal to acceleromter sensitive axis
65	f_{zy}	Coefficient of error proportional to products of acceleration along and orthogonal to acceleromter sensitive axis
66	μ_1	X accelerometer misalignment about Z-axis
67	μ_2	Y accelerometer misalignment about Z-axis
68	μ_3	Z accelerometer misalignment about Y-axis
69	σ_3	Z-accelerometer misalignment about X-axis

Table A.5. Definition of INS Truth Model Subvector $\delta \mathbf{x}_6$

State Number	State Symbol	Definition
70	∇_{x_q}	X-component of accelerometer bias thermal transient
71	∇_{y_q}	Y-component of accelerometer bias thermal transient
72	∇_{z_q}	Z-component of accelerometer bias thermal transient
73	b_{x_q}	X-component of initial gyro drift rate bias thermal transient
74	b_{y_q}	Y-component of initial gyro drift rate bias thermal transient
75	b_{z_q}	Z-component of initial gyro drift rate bias thermal transient

Table A.6. Definition of INS Truth Model Subvector $\delta \mathbf{x}_6$

State Number	State Symbol	Definition
76	F_{xyz}	X gyro compliance term
77	F_{xyy}	X gyro compliance term
78	F_{xyx}	X gyro compliance term
79	F_{xzy}	X gyro compliance term
80	F_{xzz}	X gyro compliance term
81	F_{xzx}	X gyro compliance term
82	F_{yzx}	Y gyro compliance term
83	F_{yzz}	Y gyro compliance term
84	F_{yzy}	Y gyro compliance term
85	F_{yxz}	Y gyro compliance term
86	F_{yxx}	Y gyro compliance term
87	F_{yxy}	Y gyro compliance term
88	F_{xzy}	Z gyro compliance term
89	F_{zzz}	Z gyro compliance term
90	F_{zzz}	Z gyro compliance term
91	F_{zyx}	Z gyro compliance term
92	F_{zyy}	Z gyro compliance term
93	F_{zyz}	Z gyro compliance term

Appendix B. *Initial INS Covariance Matrix Values*

The covariance matrix prior to alignment is a diagonal matrix where the diagonal terms are the variances of the states. The off-diagonal terms are the covariances between states. Since the covariance terms are all zero, they are not provided in the following tables. The variances are broken into six tables, one table for each state subvector.

Table B.1. Initial Variances for Subvector δx_1

State	1σ Value for Static Navigation	1σ Value for Fighter Flight
1	0 rad	0 rad
2	0 rad	0 rad
3	0 rad	0 rad
4	1800 $\widehat{\text{sec}}$	1800 $\widehat{\text{sec}}$
5	1800 $\widehat{\text{sec}}$	1800 $\widehat{\text{sec}}$
6	1800 $\widehat{\text{sec}}$	1800 $\widehat{\text{sec}}$
7	0 fps	0 fps
8	0 fps	0 fps
9	0 fps	0 fps
10	0 ft	0 ft
11	0 ft	0 ft
12	0 ft	0 ft
13	0 fps^2	0 fps^2

Table B.2. Initial Variances for Subvector $\delta \mathbf{x}_2$

State	1σ Value for Static Navigation	1σ Value for Fighter Flight
14	0°/hr.	0.002°/hr.
15	0°/hr.	0.002°/hr.
16	0°/hr.	0.002°/hr.
17	2 μg	2 μg
18	2 μg	2 μg
19	2 μg	2 μg
20	0 $\widehat{\text{sec}}$	5 $\widehat{\text{sec}}$
21	0 $\widehat{\text{sec}}$	5 $\widehat{\text{sec}}$
22	0 $\widehat{\text{sec}}$	0 $\widehat{\text{sec}}$
23	0 ft.	100 ft.
24	0.0001°/hr./hr.	0.0001°/hr./hr.
25	0.0001°/hr./hr.	0.0001°/hr./hr.
26	0.0001°/hr./hr.	0.0001°/hr./hr.
27	1 μg/hr.	1 μg/hr.
28	1 μg/hr.	1 μg/hr.
29	0 μg/hr.	1 μg/hr.

Table B.3. Initial Variances for Subvector δx_3

State	1σ Value for Static Navigation	1σ Value for Fighter Flight
30	0.003°/hr.	0.003°/hr.
31	0.003°/hr.	0.003°/hr.
32	0.003°/hr.	0.003°/hr.
33	0.0005%	0.0005%
34	0.0005%	0.0005%
35	0.0005%	0.0005%
36	1.5 sec	1.5 sec
37	1.5 sec	1.5 sec
38	0 sec	1.5 sec
39	0 sec	1.5 sec
40	1.5 sec	1.5 sec
41	1.5 sec	1.5 sec
42	$0.5(°/hr.)/(rad/sec.)^2$	$0.5(°/hr.)/(rad/sec.)^2$
43	$0(°/hr.)/(rad/sec.)^2$	$0.5(°/hr.)/(rad/sec.)^2$
44	$0.5(°/hr.)/(rad/sec.)^2$	$0.5(°/hr.)/(rad/sec.)^2$
45	1 ppm	1 ppm
46	0 ppm	1 ppm
47	1 ppm	1 ppm

Table B.4. Initial Variances for Subvector δx_4

State	1σ Value for Static Navigation	1σ Value for Fighter Flight
48	15 μg	15 μg
49	15 μg	15 μg
50	0 μg	15 μg
51	0%	0.012%
52	0%	0.012%
53	0%	0.012%
54	0%	0.0025%
55	0%	0.0025%
56	0%	0.0025%
57	$0\mu g/g^2$	$3\mu g/g^2$
58	$0\mu g/g^2$	$3\mu g/g^2$
59	$0\mu g/g^2$	$3\mu g/g^2$
60	$0\mu g/g^2$	$3\mu g/g^2$
61	$0\mu g/g^2$	$3\mu g/g^2$
62	$0\mu g/g^2$	$3\mu g/g^2$
63	$0\mu g/g^2$	$3\mu g/g^2$
64	$0\mu g/g^2$	$3\mu g/g^2$
65	$0\mu g/g^2$	$3\mu g/g^2$
66	0 $\hat{\text{sec}}$	4 $\hat{\text{sec}}$
67	0 $\hat{\text{sec}}$	4 $\hat{\text{sec}}$
68	0 $\hat{\text{sec}}$	4 $\hat{\text{sec}}$
69	0 $\hat{\text{sec}}$	4 $\hat{\text{sec}}$

Table B.5. Initial Variances for Subvector δx_5

State	1σ Value for Static Navigation	1σ Value for Fighter Flight
70	6μg	6μg
71	6μg	6μg
72	0μg	6μg
73	0.003°/hr.	0.003°/hr.
74	0.003°/hr.	0.003°/hr.
75	0.003°/hr.	0.003°/hr.

Table B.6. Initial Variances for Subvector $\delta \mathbf{x}_6$

State	1σ Value for Static Navigation	1σ Value for Fighter Flight
76	0 sec/g	0.3 sec/g
77	0 sec/g	0.3 sec/g
78	0 sec/g	0.3 sec/g
79	0 sec/g	0.3 sec/g
80	0.3 sec/g	0.3 sec/g
81	0.3 sec/g	0.3 sec/g
82	0.3 sec/g	0.3 sec/g
83	0.3 sec/g	0.3 sec/g
84	0 sec/g	0.3 sec/g
85	0 sec/g	0.3 sec/g
86	0 sec/g	0.3 sec/g
87	0 sec/g	0.3 sec/g
88	0 sec/g	0.3 sec/g
89	0 sec/g	0.3 sec/g
90	0 sec/g	0.3 sec/g
91	0 sec/g	0.3 sec/g
92	0 sec/g	0.3 sec/g
93	0 sec/g	0.3 sec/g

Appendix C. *Elements of the INS Truth Model Dynamics Matrix*

The INS truth model dynamics matrix is a 93 by 93 matrix. This matrix is broken into submatrices which are defined in the text. The non-zero elements of the submatrices are shown here. Each element is referred to by its place in the overall dynamics matrix, not by its position in the submatrix. The elements of the $\mathbf{C}_{\mathbf{s}}^{\mathbf{t}}$, sensor-to-true, matrix are used here as C_{ij} where i is the row and j is the column in the transformation matrix.

Table C.1. Elements of the Dynamics Submatrix $\mathbf{F_{11}}$

Element	Term	Element	Term
(1,3)	$-\rho_y$	(1,8)	$-C_{RY}$
(2,3)	ρ_x	(2,7)	C_{RX}
(3,1)	ρ_y	(3,2)	$-\rho_x$
(4,2)	$-\Omega_z$	(4,3)	Ω_y
(4,5)	ω_{it_z}	(4,6)	$-\omega_{it_y}$
(4,8)	$-C_{RY}$	(5,1)	Ω_z
(5,3)	$-\Omega_x$	(5,4)	$-\omega_{it_z}$
(5,6)	ω_{it_x}	(5,7)	C_{RX}
(6,1)	$-\Omega_y$	(6,2)	Ω_x
(6,4)	ω_{it_y}	(6,5)	$-\omega_{it_x}$
(7,1)	$-2V_y\Omega_y - 2V_z\Omega_z$	(7,2)	$2V_y\Omega_x$
(7,3)	$2V_z\Omega_y$	(7,5)	$-A_z$
(7,6)	A_y	(7,7)	$-V_zC_{RX}$
(7,8)	$2\Omega_z$	(7,9)	$-\rho_y - 2\Omega_y$
(8,1)	$2V_x\Omega_y$	(8,2)	$-2V_x\Omega_x - 2V_z\Omega_z$
(8,3)	$2v_z\Omega_y$	(8,4)	A_z
(8,6)	$-A_x$	(8,7)	$-2\Omega_z$
(8,8)	$-V_zC_{RY}$	(8,9)	$\rho_x + 2\Omega_x$
(9,1)	$2V_x\Omega_z$	(9,2)	$2V_y\Omega_z$
(9,3)	$-2V_y\Omega_y - 2V_x\Omega_x$	(9,4)	$-A_y$
(9,5)	A_x	(9,7)	$\rho_y + 2\Omega_y + V_xC_{RX}$
(9,8)	$-\rho_x - 2\Omega_x + V_yC_{RY}$	(9,10)	$2g_o/a$
(9,11)	$-k_2$	(9,12)	-1
(9,13)	k_2	(10,9)	1
(10,11)	$-k_1$	(10,13)	$k_1 - 1$
(11,10)	1	(11,11)	-1
(12,11)	k_3	(12,13)	$-k_3$
(13,10)	k_4	(13,11)	$-k_4$
(13,13)	$k_4 - 1$		

Table C.2. Elements of the Dynamics Submatrix $\mathbf{F_{12}}$

Element	Term	Element	Term	Element	Term
(4,14)	C_{11}	(4,15)	C_{12}	(4,16)	C_{13}
(4,24)	$C_{11}t$	(4,25)	$C_{12}t$	(4,26)	$C_{13}t$
(5,14)	C_{21}	(5,15)	C_{22}	(5,16)	C_{23}
(5,24)	$C_{21}t$	(5,25)	$C_{22}t$	(5,26)	$C_{23}t$
(6,14)	C_{31}	(6,15)	C_{32}	(6,16)	C_{33}
(6,24)	$C_{31}t$	(6,25)	$C_{32}t$	(6,26)	$C_{33}t$
(7,17)	C_{11}	(7,18)	C_{12}	(7,19)	C_{13}
(7,20)	1	(7,27)	$C_{11}t$	(7,28)	$C_{12}t$
(7,29)	$C_{13}t$	(8,17)	C_{21}	(8,18)	C_{22}
(8,19)	$C_{23}\cdot$	(8,21)	1	(8,27)	$C_{21}t$
(8,28)	$C_{22}t$	(8,29)	$C_{23}t$	(9,17)	C_{31}
(9,18)	C_{32}	(9,19)	C_{33}	(9,22)	1
(9,23)	k_2	(9,27)	$C_{31}t$	(9,28)	$C_{32}t$
(9,29)	$C_{33}t$	(10,23)	k_1	(12,23)	$-k_3$
(13,23)	$k_4/600$				

Table C.3. Elements of the Dynamics Submatrix $\mathbf{F_{13}}$

Element	Term	Element	Term	Element	Term
(4,30)	C_{11}	(4,31)	C_{12}	(4,32)	C_{13}
(4,33)	$C_{11}\omega_{ib_z}$	(4,34)	$C_{12}\omega_{ib_y}$	(4,35)	$C_{13}\omega_{ib_z}$
(4,36)	$C_{11}\omega_{ib_z}$	(4,37)	$-C_{12}\omega_{ib_z}$	(4,38)	$C_{13}\omega_{ib_y}$
(4,39)	$-C_{11}\omega_{ib_y}$	(4,40)	$C_{12}\omega_{ib_z}$	(4,41)	$-C13\omega_{ib_z}$
(4,42)	$C_{11}\omega_{ib_z}^2$	(4,43)	$C_{12}\omega_{ib_y}^2$	(4,44)	$C_{13}\omega_{ib_z}^2$
(4,45)	$0.5C_{11}\lvert\omega_{ib_z}\rvert$	(4,46)	$0.5C_{12}\lvert\omega_{ib_y}\rvert$	(4,47)	$0.5C_{13}\lvert\omega_{ib_z}\rvert$
(5,30)	C_{21}	(5,31)	C_{22}	(5,32)	C_{23}
(5,33)	$C_{21}\omega_{ib_z}$	(5,34)	$C_{22}\omega_{ib_y}$	(5,35)	$C_{23}\omega_{ib_z}$
(5,36)	$C_{21}\omega_{ib_z}$	(5,37)	$-C_{22}\omega_{ib_z}$	(5,38)	$C_{23}\omega_{ib_y}$
(5,39)	$-C_{21}\omega_{ib_y}$	(5,40)	$C_{22}\omega_{ib_z}$	(5,41)	$-C23\omega_{ib_z}$
(5,42)	$C_{21}\omega_{ib_z}^2$	(5,43)	$C_{22}\omega_{ib_y}^2$	(5,44)	$C_{23}\omega_{ib_z}^2$
(5,45)	$0.5C_{21}\lvert\omega_{ib_z}\rvert$	(5,46)	$0.5C_{22}\lvert\omega_{ib_y}\rvert$	(5,47)	$0.5C_{23}\lvert\omega_{ib_z}\rvert$
(6,30)	C_{31}	(6,31)	C_{32}	(6,32)	C_{33}
(6,33)	$C_{31}\omega_{ib_z}$	(6,34)	$C_{32}\omega_{ib_y}$	(6,35)	$C_{33}\omega_{ib_z}$
(6,36)	$C_{31}\omega_{ib_z}$	(6,37)	$-C_{32}\omega_{ib_z}$	(6,38)	$C_{33}\omega_{ib_y}$
(6,39)	$-C_{31}\omega_{ib_y}$	(6,40)	$C_{32}\omega_{ib_z}$	(6,41)	$-C33\omega_{ib_z}$
(6,42)	$C_{31}\omega_{ib_z}^2$	(6,43)	$C_{32}\omega_{ib_y}^2$	(6,44)	$C_{33}\omega_{ib_z}^2$
(6,45)	$0.5C_{31}\lvert\omega_{ib_z}\rvert$	(6,46)	$0.5C_{32}\lvert\omega_{ib_y}\rvert$	(6,47)	$0.5C_{33}\lvert\omega_{ib_z}\rvert$

Table C.4. Elements of the Dynamics Submatrix $\mathbf{F_{14}}$

Element	Term	Element	Term	Element	Term
(7,48)	C_{11}	(7,49)	C_{12}	(7,50)	C_{13}
(7,51)	$C_{11}A_x^B$	(7,52)	$C_{12}A_y^B$	(7,53)	$C_{13}A_z^{B'}$
(7,54)	$C_{11}\lvert A_x^B\rvert$	(7,55)	$C_{12}\lvert A_y^B\rvert$	(7,56)	$C_{13}\lvert A_z^{B'}\rvert$
(7,57)	$C_{11}A_x^{B2}$	(7,58)	$C_{12}A_y^{B2}$	(7,59)	$C_{13}A_z^{B'2}$
(7,60)	$C_{11}A_x^BA_y^B$	(7,61)	$C_{11}A_x^BA_z^B$	(7,62)	$C_{12}A_y^BA_x^B$
(7,63)	$C_{12}A_y^BA_z^B$	(7,64)	$C_{13}A_x^BA_z^B$	(7,65)	$C_{13}A_y^BA_z^B$
(7,66)	$C_{11}A_y^B$	(7,67)	$-C_{12}A_x^B$	(7,68)	$C_{13}A_y^B$
(7,69)	$C_{13}A_x^B$	(8,48)	C_{21}	(8,49)	C_{22}
(8,50)	C_{23}	(8,51)	$C_{21}A_x^B$	(8,52)	$C_{22}A_y^B$
(8,53)	$C_{23}A_z^{B'}$	(8,54)	$C_{21}\lvert A_x^B\rvert$	(8,55)	$C_{22}\lvert A_y^B\rvert$
(8,56)	$C_{23}\lvert A_z^{B'}\rvert$	(8,57)	$C_{21}A_x^{B2}$	(8,58)	$C_{22}A_y^{B2}$
(8,59)	$C_{23}A_z^{B'2}$	(8,60)	$C_{21}A_x^BA_y^B$	(8,61)	$C_{21}A_x^BA_z^B$
(8,62)	$C_{22}A_y^BA_x^B$	(8,63)	$C_{22}A_y^BA_x^B$	(8,64)	$C_{23}A_x^BA_z^B$
(8,65)	$C_{23}A_y^BA_z^B$	(8,66)	$C_{21}A_y^B$	(8,67)	$-C_{22}A_x^B$
(8,68)	$C_{23}A_y^B$	(8,69)	$C_{23}A_x^B$	(9,48)	C_{31}
(9,49)	C_{32}	(9,50)	C_{33}	(9,51)	$C_{31}A_x^B$
(9,52)	$C_{32}A_y^B$	(9,53)	$C_{33}A_z^{B'}$	(9,54)	$C_{31}\lvert A_x^B\rvert$
(9,55)	$C_{32}\lvert A_y^B\rvert$	(9,56)	$C_{33}\lvert A_z^{B'}\rvert$	(9,57)	$C_{31}A_x^{B2}$
(9,58)	$C_{32}A_y^{B2}$	(9,59)	$C_{33}A_z^{B'2}$	(9,60)	$C_{31}A_x^BA_y^B$
(9,61)	$C_{31}A_x^BA_z^B$	(9,62)	$C_{32}A_y^BA_x^B$	(9,63)	$C_{32}A_y^BA_z^B$
(9,64)	$C_{33}A_x^BA_z^B$	(9,65)	$C_{33}A_y^BA_z^B$	(9,66)	$C_{31}A_y^B$
(9,67)	$-C_{32}A_x^B$	(9,68)	$C_{33}A_y^B$	(9,69)	$C_{33}A_x^B$

Table C.5. Elements of the Dynamics Submatrix $\mathbf{F_{15}}$

Element	Term	Element	Term	Element	Term
(4,73)	C_{11}	(4,74)	C_{12}	(4,75)	C_{13}
(5,73)	C_{21}	(5,74)	C_{22}	(5,75)	C_{23}
(6,73)	C_{31}	(6,74)	C_{32}	(6,75)	C_{33}
(7,70)	C_{11}	(7,71)	C_{12}	(7,72)	C_{13}
(8,70)	C_{21}	(8,71)	C_{22}	(8,72)	C_{23}
(9,70)	C_{31}	(9,71)	C_{32}	(9,72)	C_{33}

Table C.6. Elements of the Dynamics Submatrix $\mathbf{F_{16}}$

Element	Term	Element	Term	Element	Term
(4,76)	$C_{11}A_y^B\omega_{ib_z}$	(4,77)	$C_{11}A_y^B\omega_{ib_y}$	(4,78)	$C_{11}A_y^B\omega_{ib_x}$
(4,79)	$C_{11}A_z^B\omega_{ib_y}$	(4,80)	$C_{11}A_z^B\omega_{ib_z}$	(4,81)	$C_{11}A_z^B\omega_{ib_x}$
(4,82)	$C_{12}A_z^B\omega_{ib_z}$	(4,83)	$C_{12}A_z^B\omega_{ib_z}$	(4,84)	$C_{12}A_z^B\omega_{ib_y}$
(4,85)	$C_{12}A_x^B\omega_{ib_z}$	(4,86)	$C_{12}A_x^B\omega_{ib_z}$	(4,87)	$C_{12}A_x^B\omega_{ib_y}$
(4,88)	$C_{13}A_x^B\omega_{ib_y}$	(4,89)	$C_{13}A_x^B\omega_{ib_z}$	(4,90)	$C_{13}A_x^B\omega_{ib_z}$
(4,91)	$C_{13}A_y^B\omega_{ib_z}$	(4,92)	$C_{13}A_y^B\omega_{ib_y}$	(4,93)	$C_{13}A_y^B\omega_{ib_z}$
(5,76)	$C_{21}A_y^B\omega_{ib_z}$	(5,77)	$C_{21}A_y^B\omega_{ib_y}$	(5,78)	$C_{21}A_y^B\omega_{ib_x}$
(5,79)	$C_{21}A_z^B\omega_{ib_y}$	(5,80)	$C_{21}A_z^B\omega_{ib_z}$	(5,81)	$C_{21}A_z^B\omega_{ib_x}$
(5,82)	$C_{22}A_z^B\omega_{ib_z}$	(5,83)	$C_{22}A_z^B\omega_{ib_z}$	(5,84)	$C_{22}A_z^B\omega_{ib_y}$
(5,85)	$C_{22}A_x^B\omega_{ib_z}$	(5,86)	$C_{22}A_x^B\omega_{ib_z}$	(5,87)	$C_{22}A_x^B\omega_{ib_y}$
(5,88)	$C_{23}A_x^B\omega_{ib_y}$	(5,89)	$C_{23}A_x^B\omega_{ib_z}$	(5,90)	$C_{23}A_x^B\omega_{ib_z}$
(5,91)	$C_{23}A_y^B\omega_{ib_z}$	(5,92)	$C_{23}A_y^B\omega_{ib_y}$	(5,93)	$C_{23}A_y^B\omega_{ib_z}$
(6,76)	$C_{31}A_y^B\omega_{ib_z}$	(6,77)	$C_{31}A_y^B\omega_{ib_y}$	(6,78)	$C_{31}A_y^B\omega_{ib_x}$
(6,79)	$C_{31}A_z^B\omega_{ib_y}$	(6,80)	$C_{31}A_z^B\omega_{ib_z}$	(6,81)	$C_{31}A_z^B\omega_{ib_x}$
(6,82)	$C_{32}A_z^B\omega_{ib_z}$	(6,83)	$C_{32}A_z^B\omega_{ib_z}$	(6,84)	$C_{32}A_z^B\omega_{ib_y}$
(6,85)	$C_{32}A_x^B\omega_{ib_z}$	(6,86)	$C_{32}A_x^B\omega_{ib_z}$	(6,87)	$C_{32}A_x^B\omega_{ib_y}$
(6,88)	$C_{33}A_x^B\omega_{ib_y}$	(6,89)	$C_{33}A_x^B\omega_{ib_z}$	(6,90)	$C_{33}A_x^B\omega_{ib_z}$
(6,91)	$C_{33}A_y^B\omega_{ib_z}$	(6,92)	$C_{33}A_y^B\omega_{ib_y}$	(6,93)	$C_{33}A_y^B\omega_{ib_z}$

Table C.7. Elements of the Dynamics Submatrix $\mathbf{F_{22}}$

Element	Term	Element	Term	Element	Term
(14,14)	$-\beta_{b_{x_c}}$	(15,15)	$-\beta_{b_{y_c}}$	(16,16)	$-\beta_{b_{z_c}}$
(17,17)	$-\beta_{\nabla_{x_c}}$	(18,18)	$-\beta_{\nabla_{y_c}}$	(19,19)	$-\beta_{\nabla_{z_c}}$
(20,20)	$-\beta_{\delta g_x}$	(21,21)	$-\beta_{\delta g_y}$	(22,22)	$-\beta_{\delta g_z}$
(23,23)	$-\beta_{\delta h_c}$				

Table C.8. Elements of the Dynamics Submatrix $\mathbf{F_{55}}$

Element	Term	Element	Term	Element	Term
(70,70)	$-\beta_{\nabla_{x_q}}$	(71,71)	$-\beta_{\nabla_{y_q}}$	(72,72)	$-\beta_{\nabla_{z_q}}$
(73,73)	$-\beta_{b_{x_q}}$	(74,74)	$-\beta_{b_{y_q}}$	(75,75)	$-\beta_{b_{z_q}}$

Appendix D. *Elements of the Process Noise Matrix*

The process noise matrix contains mostly zeros. The matrix was subdivided into two submatrices which contain the non-zero elements and a number of zero matrices. Only the non-zero elements of the submatrices are shown here. The elements are referred to by their placement in the overall matrix, not their placement in the submatrices.

Table D.1. Non-zero Elements of Process Noise Submatrix $\mathbf{Q_{11}}$

Element	Term	Element	Term
(4,4)	$\sigma^2_{\eta_{b_x}}$	(5,5)	$\sigma^2_{\eta_{b_y}}$
(6,6)	$\sigma^2_{\eta_{b_z}}$	(7,7)	$\sigma^2_{\eta_{A_x}}$
(8,8)	$\sigma^2_{\eta_{A_y}}$	(9,9)	$\sigma^2_{\eta_{A_z}}$

Table D.2. Non-zero Elements of Process Noise Submatrix $\mathbf{Q_{22}}$

Element	Term	Element	Term
(14,14)	$2\beta_{b_{x_c}}\sigma^2_{b_{x_c}}$	(15,15)	$2\beta_{b_{y_c}}\sigma^2_{b_{y_c}}$
(16,16)	$2\beta_{b_{z_c}}\sigma^2_{b_{z_c}}$	(17,17)	$2\beta_{\nabla_{x_c}}\sigma^2_{\nabla_{x_c}}$
(18,18)	$2\beta_{\nabla_{y_c}}\sigma^2_{\nabla_{y_c}}$	(18,18)	$2\beta_{\nabla_{z_c}}\sigma^2_{\nabla_{z_c}}$
(20,20)	$2\beta_{\delta g_x}\sigma^2_{\delta g_x}$	(21,21)	$2\beta_{\delta g_y}\sigma^2_{\delta g_y}$
(22,22)	$2\beta_{\delta g_z}\sigma^2_{\delta g_z}$	(23,23)	$2\beta_{\delta h_c}\sigma^2_{\delta h_c}$

Appendix E. *Instrumentation Port Block 1022 Definition File*

The block definition file for block 1022 from the GPS receiver's instrumentation port is shown below. Explanations of each column may be found in Chapter 5. The block definition file is necessary for post processing interpretation of data collected with PC Buffer Box.

```
!
!BDF BLOCK 1022 FILTER OBSERVATION BLOCK
!
TIME                1022    1    4    R
MEAS_TIME_TAG       1022    5    3    CF
MEAS_COUNT          1022    8    1    I
MEAS_TYPE_1         1022    9    1    I
MEAS_TYPE_2         1022    10   1    I
MEAS_TYPE_3         1022    11   1    I
MEAS_TYPE_4         1022    12   1    I
MEAS_TYPE_5         1022    13   1    I
MEAS_TYPE_6         1022    14   1    I
MEAS_TYPE_7         1022    15   1    I
MEAS_TYPE_8         1022    16   1    I
MEAS_TYPE_9         1022    17   1    I
MEAS_TYPE_10        1022    18   1    I
MEAS_TYPE_11        1022    19   1    I
PSEUDO_RNG_1        1022    20   3    CF
PSEUDO_RNG_2        1022    23   3    CF
```

PSEUDO_RNG_3	1022	26	3	CF
PSEUDO_RNG_4	1022	29	3	CF
PSEUDO_RNG_5	1022	32	3	CF
DELTA_RNG_1	1022	35	2	CF
DELTA_RNG_2	1022	37	2	CF
DELTA_RNG_3	1022	39	2	CF
DELTA_RNG_4	1022	41	2	CF
DELTA_RNG_5	1022	43	2	CF
ALTITUDE_MEAS	1022	45	2	CF
PRED_PSU_RNG_1	1022	47	3	CF
PRED_PSU_RNG_2	1022	50	3	CF
PRED_PSU_RNG_3	1022	53	3	CF
PRED_PSU_RNG_4	1022	56	3	CF
PRED_PSU_RNG_5	1022	59	3	CF
PRED_DEL_RNG_1	1022	62	2	CF
PRED_DEL_RNG_2	1022	64	2	CF
PRED_DEL_RNG_3	1022	66	2	CF
PRED_DEL_RNG_4	1022	68	2	CF
PRED_DEL_RNG_5	1022	70	2	CF
PRED_DEL_RNG_6	1022	72	2	CF
SV_ID_1	1022	74	1	I
SV_ID_2	1022	75	1	I
SV_ID_3	1022	76	1	I
SV_ID_4	1022	77	1	I
SV_ID_5	1022	78	1	I
SV_ID_6	1022	79	1	I
SV_ID_7	1022	80	1	I
SV_ID_8	1022	81	1	I

SV_ID_9	1022	82	1	I
SV_ID_10	1022	83	1	I
SV_ID_11	1022	84	1	I
CHAN_ID_1	1022	85	1	I
CHAN_ID_2	1022	86	1	I
CHAN_ID_3	1022	87	1	I
CHAN_ID_4	1022	88	1	I
CHAN_ID_5	1022	89	1	I
CHAN_ID_6	1022	90	1	I
CHAN_ID_7	1022	91	1	I
CHAN_ID_8	1022	92	1	I
CHAN_ID_9	1022	93	1	I
CHAN_ID_10	1022	94	1	I
CHAN_ID_11	1022	95	1	I

Appendix F. *Errata for LN-93 INS Truth Model*

The document containing the truth model for the LN-94 contains a number of discrepancies and omissions. This is a list of the errors discovered in the document listed by page and section if multiple sections are on a single page.

page 6

"β_{bh_c}" should be "$\beta_{\delta h_c}$"

"$\beta_{b_{z_c}}$" = 1/10 per min., not the 5 min. correlation time that is shown.

"$\beta_{\nabla_{z_c}}$" = 1/5 per min., not the 10 min. correlation time that is shown.

"$\beta_{b_{z_q}}$" = 1/600 per sec., not the 60 sec. correlation time that is shown.

page 7

Section 3.1

"$\delta\theta_x =$ " should be "$\delta\dot{\theta}_x =$ "

"$\delta\theta_y =$ " should be "$\delta\dot{\theta}_y =$ "

"$\delta\theta_z =$ " should be "$\delta\dot{\theta}_z =$ "

Section 3.2

"$\phi_x =$ " should be "$\dot{\phi}_x =$ "

in the $\dot{\phi}_x$ equation, "$\omega_x\phi_y$" should be "$\omega_x\phi_y$"

"$\phi_y =$ " should be "$\dot{\phi}_y =$ "

"$\phi_z =$ " should be "$\dot{\phi}_z =$ "

Section 3.3

"$\delta V_x =$ " should be "$\delta \dot{V}_x =$ "

"$\delta V_y =$ " should be "$\delta \dot{V}_y =$ "

page 8

To make the equations reflect the fact that Litton used $\delta h_B = \delta h_c$

in the $\delta \dot{V}_x$ equation, "$k_2 \delta h_B$" should be "$k_2 \delta h_c$"

in the $\delta \dot{h}$ equation, "$k_1 \delta h_B$" should be "$k_1 \delta h_c$"

in the $\delta \dot{S}_3$ equation, " $- k_3 \delta h_B$" should be " $- k_3 \delta h_c$"

in the $\delta \dot{S}_4$ equation, " $- k_4 \delta h_B$" should be " $- k_4 \delta \dot{h}_c$"

$\delta \dot{h}_B = \delta \dot{h}_c$

Also,

in the $\delta \dot{S}_4$ equation, "$k_4 \delta h_L$" should be "$k_4 \delta \dot{h}_L$"

page 9

in the C_{RX} equation, " $- 2C_{Yy}^2$" should be " $- 2C_{Yx}^2$"

"$C_{Xx} =$ " should be "$\dot{C}_{Xx} =$ "

"$C_{Xy} =$ " should be "$\dot{C}_{Xy} =$ "

"$C_{Xz} =$ " should be "$\dot{C}_{Xz} =$ "

"$C_{Yx} =$ " should be "$\dot{C}_{Yx} =$ "

"$C_{Yy} =$ " should be "$\dot{C}_{Yy} =$ "

"$C_{Yz} =$ " should be "$\dot{C}_{Yz} =$ "

page 10

$C_{Xx}(0) = S\lambda(0)C\Phi(0)$

page 18

" $= (6\mu g)C^{t/60\text{sec}}$" should be " $= (6\mu g)e^{t/60\text{sec}}$"

page 21

$\delta\dot{S}_4 = (k_4 - 1)\delta S_4 + k_4\delta\dot{h}_L - k_4\delta\dot{h}_c$. Omit rest of line.

page 22

element (6,2), should be "C_{32}"

page 23

element (5,12), " $- C_{23}\omega_x$"

page 24

element (7,6), "$C_{13}A'_x$"

element (8,6), "$C_{23}A'_x$"

element (8,12), "$C_{23}A'^2_x$"

element (9,12), "$C_{33}A'^2_x$"

element (7,19), "$C_{11}A_y$"

element (8,19), "$C_{21}A_y$"

element (9,19), "$C_{31}A_y$"

element (7,20), " $- C_{12}A_x$"

element (8,20), " $- C_{23}A_x$"

element (9,20), " $-C_{32}A_x$"

page 26

heading of column 10 should be "$F_{y_{x_z}}$"

heading of column 11 should be "$F_{y_{x_x}}$"

element (4,11), "$C_{12}A_x\omega_x$"

element (5,11), "$C_{22}A_x\omega_x$"

element (6,11), "$C_{32}A_x\omega_x$"

page 36

gyro compliance errors are in arcseconds/g not seconds/g

Other errors and discrepancies:

It is never stated that $\sigma_{\delta h_c} = 100$ ft. for the simulation.

The following are some notes about items which may escape attention in the truth model document because of their placement in the text.

Notes:

ω's and A's in the ϵ and ∇ equations, on pages 14 and 17 respectively, are referenced in the body frame.

Different sets of initial errors and driving noises are used for different navigation simulations.

Only four of the six accelerometer misalignment errors are used in the truth model.

Appendix G. *Litton Error Plots for Static Navigation and Flight*

The error plots provided by Litton for comparison of navigation simulations are presented here. The errors for the static navigation case are presented in the first two figures, followed by the errors for the fighter flight profile.

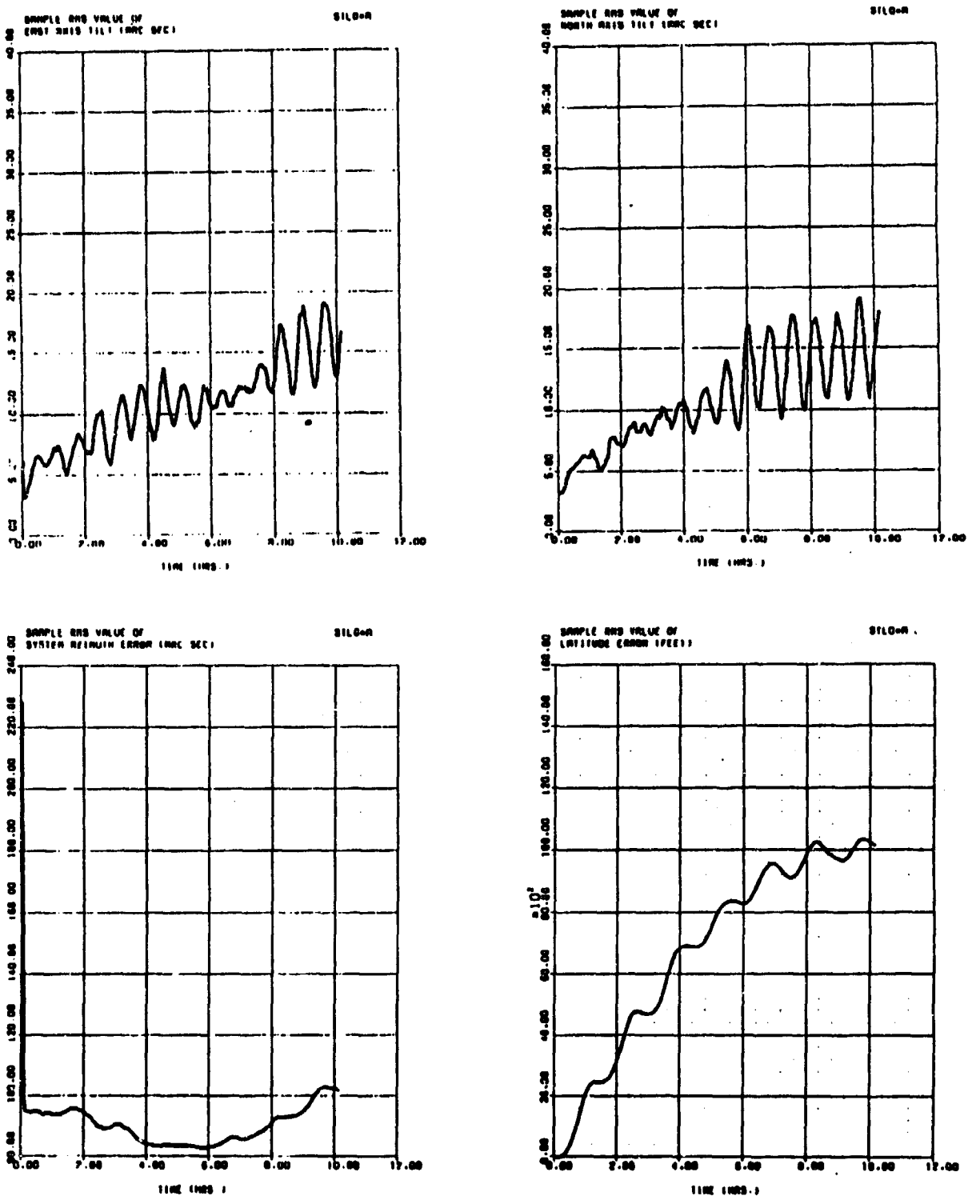


Figure G.1. Litton Static Navigation Errors

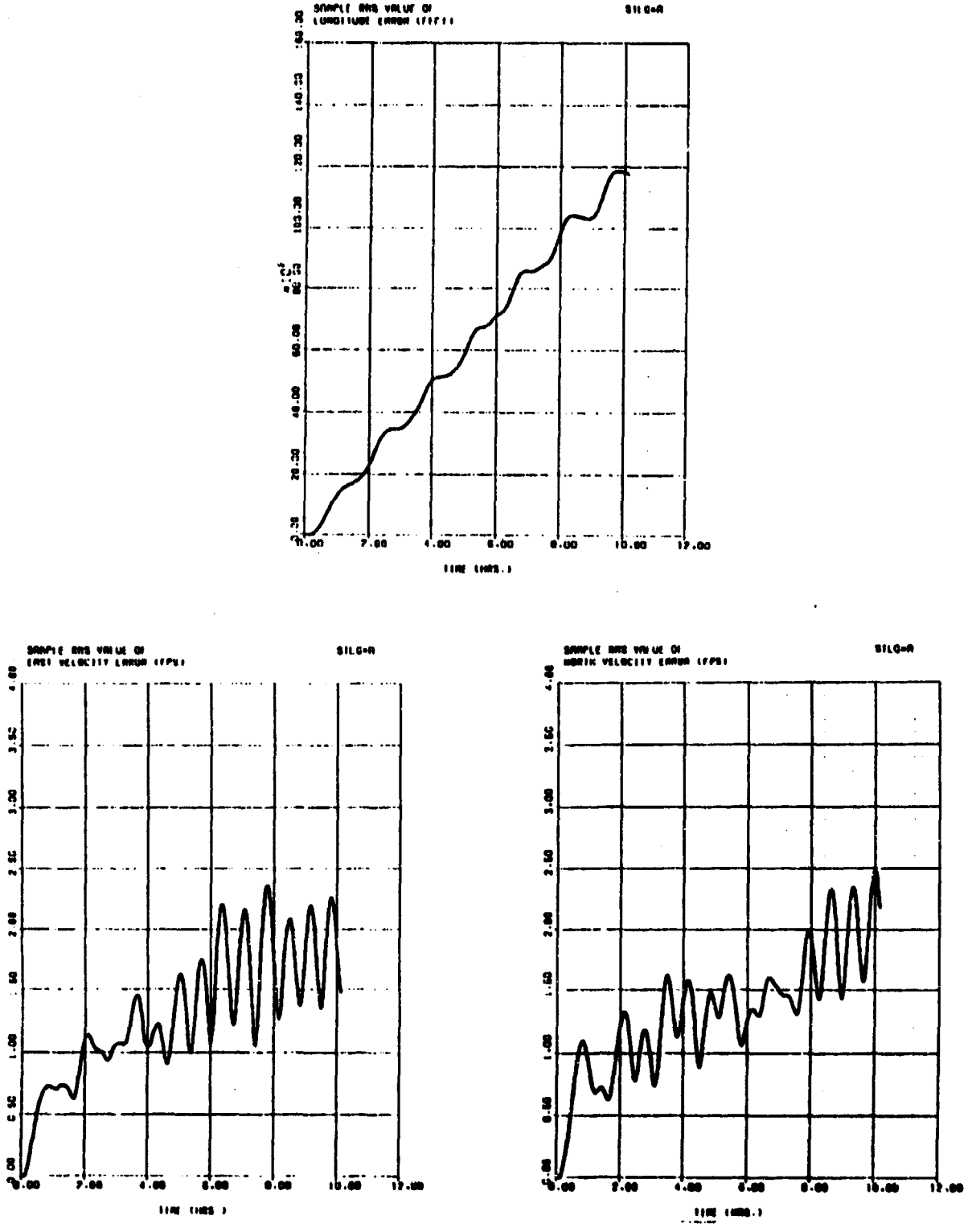


Figure G.2. Static Navigation Errors

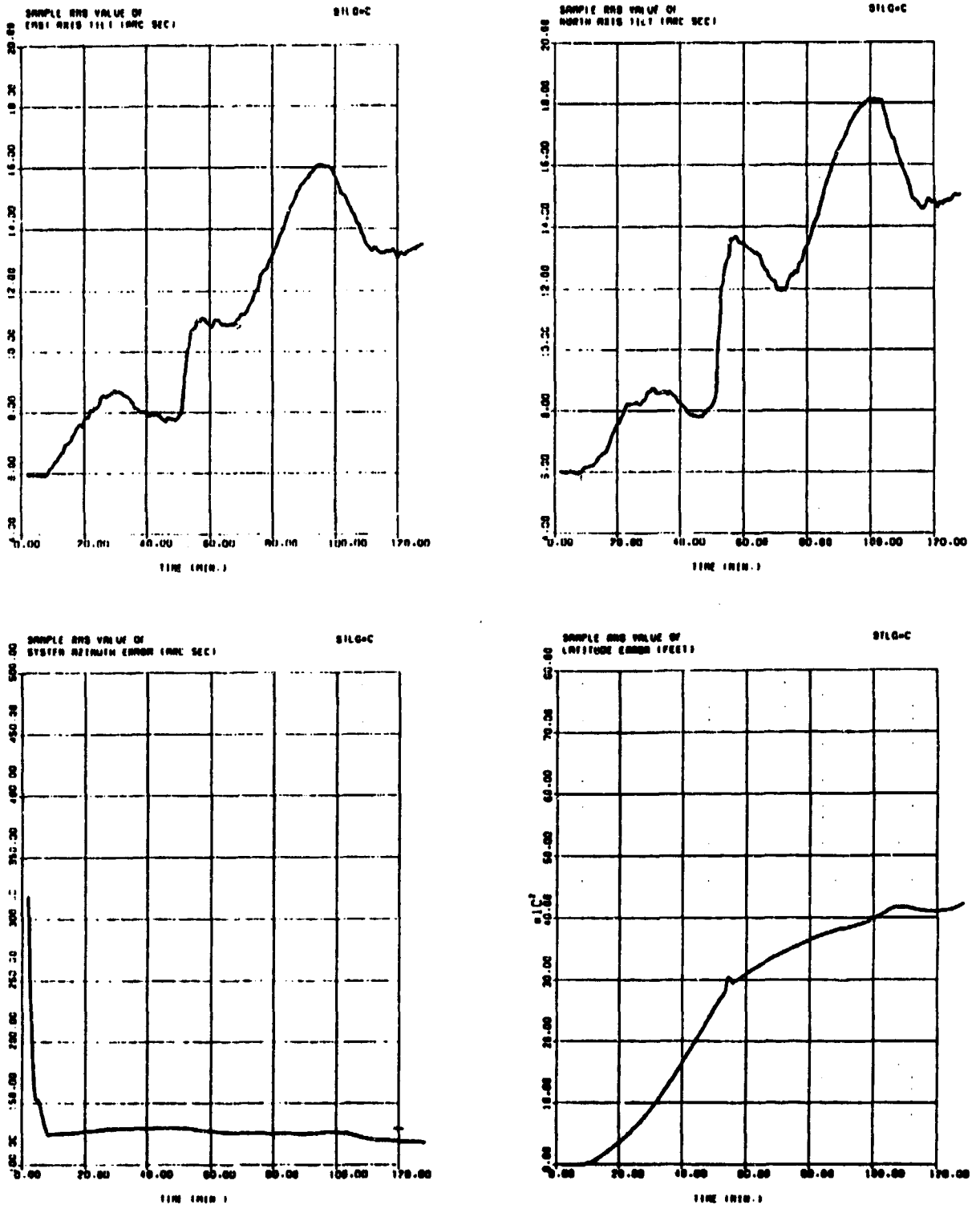


Figure G.3. Litton Fighter Flight Errors

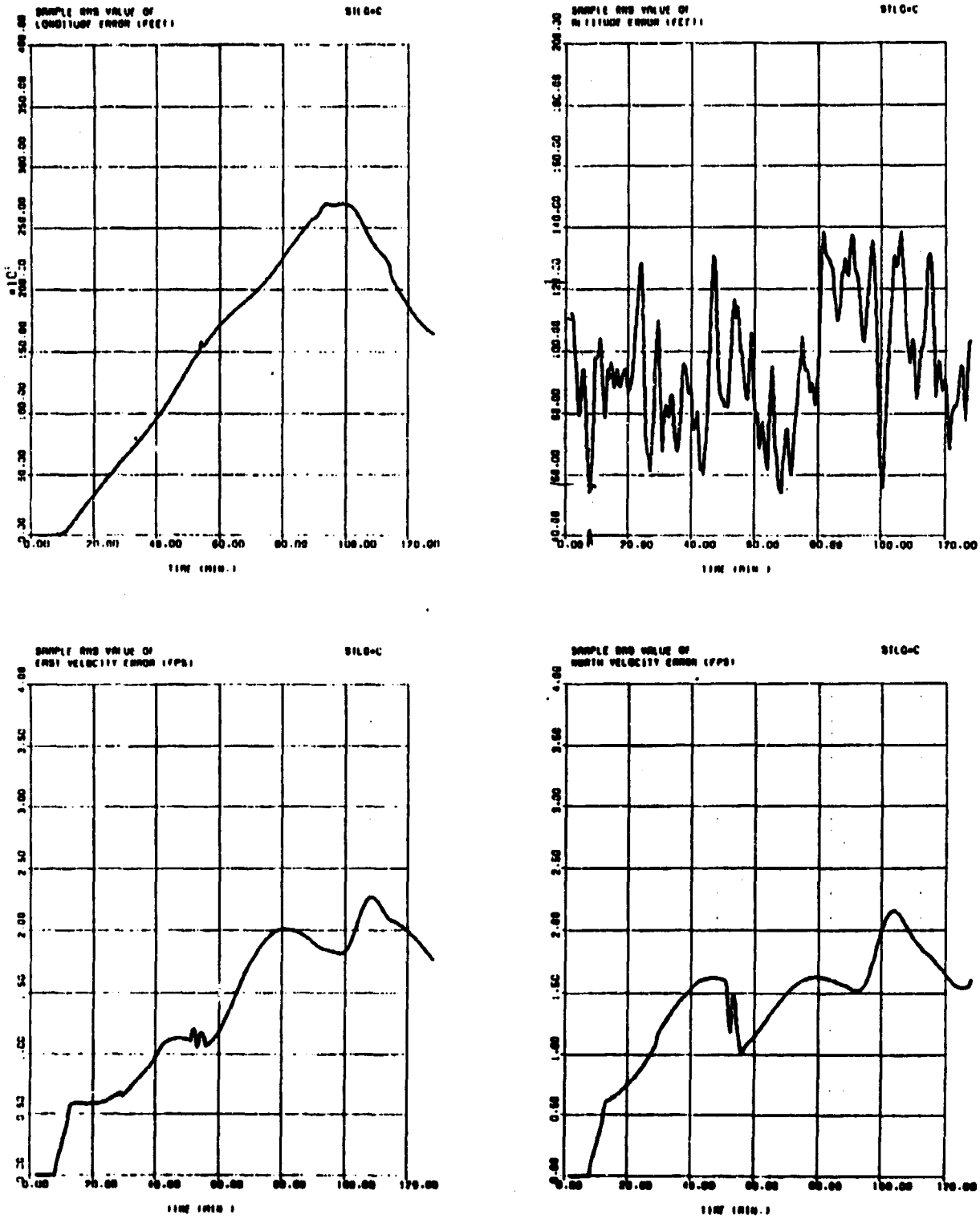


Figure G.4. Fighter Flight Errors

Bibliography

1. Aeronautical Systems Division. *Specification for USAF F-15 Inertial Navigation Set*. Technical Report FNU 85-1. March 1986.
2. Air Force Avionics Laboratory. *PROFGEN - A Computer Program for Generating Flight Profiles*. Technical Report AFAL-TR-76247. November 1976.
3. ARINC Research Corporation. *Navstar Global Positioning System Instrumentation Port*. Technical Report ICD-GPS-215. March 1987.
4. Avionics Laboratory. *Users' Manual for a Multimode Simulation for Optimal Filter Evaluation (MSOFE)*. Operators Manual AFWAL-TR-88-1138. April 1990.
5. Cox Jr., Duncan B. "Integration of GPS with Inertial Navigation Systems." In Janiczek, P. M., editor, *Global Positioning System*, Volume 1, 815 Fifteenth Street, Suite 832, Washington, D.C. 20005: The Institute of Navigation, 1978.
6. Cunningham, Joseph R. and Zdzislaw H. Lewantowicz. "Dynamic Interaction of Separate INS/GPS Kalman Filters (Filter-Driving-Filter Dynamics)." Presented at The Institute of Navigation GPS Conference, 1988.
7. Deputy for Space Navigation Systems Navstar Global Positioning System Joint Program Office. *GPS Navstar User's Overview*. Technical Report YEE-82-009A. September 1984.
8. Ellis, Terry and others. "Practical Integration of a Digital GPS Receiver with IN Systems," *The Institute of Navigation, Proceedings of the 44th Annual Meeting*, pages 117–123 (1989).
9. Huddle, James R. "Inertial Navigation System Error Model Considerations in Kalman Filter Applications," *Advances in the Techniques and Technology of the Application of Nonlinear Filters and Kalman Filters*, *AGARD-AG-256*:13–1:13–9 (1982).
10. Huddle, James R. Telephone Interviews, Litton Guidance and Control Systems, September - December 1989.
11. Intermetrics, Inc. *PC Buffer Box User Manual*. Operators Manual. July 1982.
12. Jarecki, Herb and Steve Muth. *MIL-STD-1553 Designer's Guide*. 105 Wilbur Place, Bohemia, NY 11716: ILC Data Device Corporation, 1982.
13. Knudsen, Lowell. Telephone Interviews, Litton Guidance and Control Systems, June - September 1989.
14. Litton Guidance and Control Systems. *Performance Accuracy (Truth Model/Error Budget) Analysis for the LN-93 Inertial Navigation System Inertial Navigation Unit*. CDRL No. 1002. 1985.

15. Martin, E. H. "GPS User Equipment Error Models." In Janiczek, P.M., editor, *Global Positioning System*, Volume 1, 815 Fifteenth Street, Suite 832, Washington, D,C, 20005: The Institute of Navigation, 1980.

16. Maybeck, Peter S. *Stochastic Models, Estimation, and Control, Volume 1*. 1250 Sixth Avenue, San Diego, CA: Academic Press, Inc., 1979.

17. Maybeck, Peter S. *Stochastic Models, Estimation, and Control, Volume 2*. 1250 Sixth Avenue, San Diego, CA: Academic Press, Inc., 1982.

18. Milliken, R. J. and C. J. Zoller. "Principle of Operation of NAVSTAR and System Characteristics." In Janiczek, P. M., editor, *Global Positioning System*, Volume 1, 815 Fifteenth Street, Suite 832, Washington, D,C, 20005: The Institute of Navigation, 1980.

19. SCI Technology, Inc. *BCU/VME-1014 Operations Manual*. Operators Manual Revision A. March 1987.

20. Solomon, Joseph K. "The Design of a Kalman Filter for CIRIS Aided with GPS Pseudo-range Measurements." EENG 699, Special Study, 1989.

21. Solomon, Joseph K. *Development of the Extended Kalman Filter for the Advanced Completely Integrated reference Instrumentation System (CIRIS)*. MS thesis, AFIT/GE/ENG/89M-8, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, March 1989.

22. Soltz, Arnold J. and others. "An Option for Mechanizing Integrated GPS/INS Solutions." In *The Institute of Navigation, Proceedings of the 44th Annual Meeting*, pages 124–129, 1988.

23. Teasley, Stewart P. "Flight Test Results of an Integrated GPS and Strapdown Inertial System." Presented at the IEEE Symposium on Position, Location and Navigation, 1988.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	September 1990	MS Thesis

4. TITLE AND SUBTITLE
OPTIMAL KALMAN FILTER INTEGRATION OF A GLOBAL POSITIONING SYSTEM RECEIVER AND AN LN-94 INERTIAL NAVIGATION SYSTEM

5. FUNDING NUMBERS

6. AUTHOR(S)
James L. Hirning

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Air Force Institute of Technology
Wright-Patterson AFB, OH 45433

8. PERFORMING ORGANIZATION REPORT NUMBER
AFIT/GE/ENG/90S-02

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
6585 Test Group
Holloman AFB, NM 88330-5000

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

This research develops and attempts to implement a Kalman filter integration of a Phase III Global Positioning System (GPS) five-channel receiver and an LN-94 Inertial Navigation System (INS). The GPS provides highly accurate position and velocity information in low dynamic environments. An INS provides position and velocity information with lower accuracy over long periods of time, but it is highly responsive in dynamic maneuvers or at high frequencies. The INS has the added advantage of requiring no signals external to the vehicle to function. The integration of these two systems provides more precise information under a wider variety of situations.

A truth model for the INS is verified. A GPS error model is developed and combined with the INS model to provide GPS-aided-INS navigation. This model is used to predict baseline performance of a full-ordered filter. Attempts are made to utilize the filter with empirical data. The data is analyzed, and suggestions are made about ways to account for the errors in evidence. Results to date are presented and analyzed. Keywords: Theses, JLB

14. SUBJECT TERMS	15. NUMBER OF PAGES
Global Positioning System, Inertial Navigation System, GPS/INS Integration, GPS-aided-INS	120
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	Unlimited