12-1995

A Comparison of Loose and Tight GPS/INS Integration Using Real INS and GPS Data

Warren H. Nuibe

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/GE/ENG/95D-19

A COMPARISON OF LOOSE AND TIGHT GPS/INS INTEGRATION USING REAL INS AND GPS DATA

THESIS

Warren H. Nuibe
Captain, USAF

AFIT/GE/ENG/95D-19

Approved for public release; distribution unlimited

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U. S. Government.

AFIT/GE/ENG/95D-19

A COMPARISON OF LOOSE AND TIGHT GPS/INS INTEGRATION USING REAL INS AND GPS DATA

THESIS

Presented to the Faculty of the Graduate 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

Warren H. Nuibe, B.S. Electrical Engineering
Captain, USAF

December 1995

Approved for public release; distribution unlimited

Acknowledgements

My deep appreciation goes to the members of the AFIT Guidance and Control Class of 1995, my classmates. Their companionship and help made the whole AFIT experience enjoyable.

I would like to express a special thanks to my thesis advisor, Lt Col Robert Riggins, whose enthusiasm made the research pleasurable, and whose knowledge and advice made it possible. I also would like to thank my committee members, Dr. Peter S. Maybeck and Captain Ron Delap, for proof-reading and evaluating my work.

Warren H. Nuibe

Table of Contents

Page

List of Figures

Figure Page

Figure Page

List of Tables

AFIT/GE/ENG/95D-19

Abstract

An extended Kalman filter (EKF) is used to combine the information obtained from a Global Positioning System (GPS) receiver and an Inertial Navigation System (INS) to provide a navigation solution. This research compares the results of a tightly-coupled GPS/INS integrated system with a loosely-coupled integrated system, using real world data. A fair comparison is accomplished by using the same sets of data, and keeping the integration structures as close as possible. Both integrations are feedforward and have the same error states in the navigation Kalman filters. Differences between the two, such as navigation solutions and tuning values, are shown in the research.

A COMPARISON OF LOOSE AND TIGHT GPS/INS INTEGRATION USING REAL INS AND GPS DATA

I. Introduction

Currently within the United States Air Force (USAF) arsenal, multiple navigation tools exist to help the pilot navigate his plane. There are the inertial navigation system (INS) and the Global Positioning System (GPS) to name just two. Historically, navigation tools have evolved from dedicated, single function, mechanical sensor systems to sensor systems which have become quite sophisticated in their function and accuracy. The earlier sensors have been developed, refined, and added to the aircraft as stand-alone devices which provided crew members the increased ability to perform their mission more effectively. In more recently developed sensors, functional outputs from other sensors are added to enhance their performance. An example of this would be the integration of the INS and the GPS. This research will focus on two methods of integrating the inertial navigation system and the Global Positioning System.

The goal of this thesis is to compare a loosely-coupled GPS/INS integration with a tightly-coupled integration. The loosely-coupled configuration is based on the current USAF F-16 fighter aircraft GPS/INS integration. This thesis also builds on the natural progression in the study of integrated navigation systems at the Air Force Institute of Technology (AFIT) [6, 8, 10, 20, 29]. Most AFIT research in this field has been done with computer simulations; only a few have used actual data from hardware [6, 10]. This research uses actual data from hardware in a stationary, post-processing environment. The equipment used is a Litton LN-93 inertial navigation system [11] and a Navstar XR5-M6 GPS receiver citenavstar. The sponsor for this research is the Avionics Directorate of the Wright Laboratory, Wright-Patterson AFB, OH.

1.1 Background

This section provides the background necessary to understand this thesis. It begins with the progression of the research accomplished at AFIT. It then presents the elementary components necessary for the proposed GPS/INS integration study. This section ends with an explanation of GPS/INS integration.

1.1.1 AFIT Research Progression. The integration of navigation systems and the Global Positioning System improves navigation accuracy beyond what is attainable by either alone. Simulations at AFIT and the Avionics Directorate of the Wright Laboratory, Wright-Patterson AFB have shown just that. These simulations included combining GPS and an inertial navigation system in tightly-coupled configurations with various reduced order Kalman filters and various truth models. AFIT has also simulated failures in the truth models and tested failure detection and recovery algorithms. In general, these simulations worked very well and provided organizations like AFIT and the Avionics Directorate invaluable insights into GPS/INS integration issues.

AFIT has researched GPS and INS integration issues such as modeling and fault tolerance, primarily through simulation. Previous theses have attempted to analyze and verify integration designs using real data. In 1990 Capt James Hirning attempted to integrate a Collins 3A GPS receiver with a Litton LN-93 INS, using real data citeHirning. This attempt failed, primarily because the raw measurements and ephemeris data were not easily obtained from the GPS receiver [10]. In 1994 Capt Curtis Evans successfully integrated data from the Navstar XR-4PC and XR-5PC GPS receivers with the LN-93 [6]. Capt Evans's integration was done in a stationary case; the LN-93 power requirements do not allow for easy mobility using the resources available at AFIT. The work proposed for this thesis will continue using real data in the integration. At the same time, the current GPS/INS tight versus loose integration controversy in the United States Department of Defense will be researched.

Current USAF airborne platforms use the GPS and INS integrated in a loosely-coupled fashion (see Section 3.1.1). The inherent reason is loose integration is more convenient to implement in the aircraft than tight integration. A GPS receiver with a Kalman filter (containing a generic INS model) can output position, velocity, and time onto the military standard 1553 data bus, whereas the raw GPS measurements of pseudo-range and delta-range cannot be passed to the 1553 data bus because of physical and security limitations imposed on the bus.

On the other hand, tightly integrated filters are theoretically correct and optimal. Measurements in tight integration are not corrupted by processing and therefore can theoretically contain measurement noise that is white, Gaussian, and zero-mean as the Kalman filter assumes it to be. Measurements to the navigation filter in loose integration are first processed by the generic GPS filter, and therefore do not have errors that can be characterized as white, Gaussian, and zero-mean; thus violating the Kalman filter assumption (see Section 2.2). Although these non-white noise measurements can be modeled with shaping filters, this is not normally done since it will increase the number of filter states. Hence, the loose integration, by design, is mismodelling the time correlations and/or cross-correlations of noises. However, in the real world the Kalman filter algorithm assumptions are not always completely met, so the question of how much more accurate the tight integration is over the loose integration still remains.

1.1.2 Inertial Navigation System. An inertial navigation system (INS) utilizes the inertial properties of accelerometers and gyroscopes mounted onboard a vehicle to execute the navigation function. With appropriate initialization, an inertial navigation system is capable of continuous determination of vehicle position and velocity without the use of external radiation or optical information. However, the errors in the gyroscopes and accelerometers degrade the inertial navigation system's performance. The errors grow slowly but unboundedly over time. As a result, the INS can provide accurate position and velocity data for short periods of time. This high frequency response allows the INS to continue providing reliable information in highly dynamic environments [1].

1.1.3 Global Positioning System. The Global Positioning System (GPS) provides three-dimensional position and velocity information to users anywhere in the world. The GPS consists of a space segment, a control segment, and a user segment. The space segment consists of 24 satellites in six orbital planes. The satellites receive information from the control segment and transmit satellite orbital information to the user segment. The satellite constellation is arranged so that the user has at least four satellites visible anywhere in the world at all times, with the exception of brief outages in a few remote areas. The satellite transmits positioning information modulated with two codes: C/A-code (Clear or Coarse/Acquisition), and the higher accuracy P-code (Precise). When the P-code is encrypted it is called Y-code. The Y-code prevents everyone except the US military and allies from using the P-code [2] . The control segment consists of a master control station and five monitor stations around the world. The monitor stations track all satellites, collect data from each, and relay this information to the master control station for processing. This processing involves the computation of satellite ephemerides and satellite clock corrections. These corrections are transmitted to the satellites and accuracy is maintained. The user segment consists of anyone with an antenna and GPS receiver. The user equipment receives signals from at least four different satellites and computes position and velocity, which are provided to the user.

The pseudo-range is the primary GPS measurement to be used in the Kalman filtering algorithm. Pseudo-range is the true range plus all the measurement errors, the user clock bias being the largest error. The receiver computes pseudo-range as the time shift required to line up a replica of the code generated in the receiver with the received code from the satellite, multiplied by the speed of light. The time shift is the difference between the time of signal reception (measured in the receiver time frame) and the time of emission (measured in the satellite time frame). The difference in the receiver and satellite time frames comprise the user clock bias minus the satellite clock bias [2].

1.1.4 Kalman Filter. A Kalman filter is an optimal recursive data processing (estimation) algorithm. The Kalman filter combines all available measurement data, and with the prior knowledge about the system and measuring devices, produces an estimate of the desired variables in such a manner that the mean squared error is minimized statistically [16]. The conventional Kalman filter is based upon linear system models. For this research, the INS and GPS measurements are combined in an optimal manner providing an estimate of navigation parameters. The navigation equations, however, are nonlinear, so the extended Kalman filter is used. The basic idea of the extended Kalman filter is to relinearize about each estimate once it has been computed. In this manner, it is possible to maintain the validity of the assumption that deviations from the reference trajectory are small enough to allow linear perturbation techniques to be employed with adequate results [17]. Hence, the extended Kalman filter uses the statistical characteristics of the errors in both the GPS and the INS to determine the optimal combination of information.

1.1.5 GPS/INS Integration. The approaches to integrating Global Positioning System and inertial navigation systems implement the Kalman filter either in the direct (total state space) or the indirect (error state space) formulation, and in a feedforward or feedback mechanization. In the direct method, the total states such as vehicle position and velocity are among the state variables in the filter, placing the Kalman filter in the INS loop. Being in the INS control loop, the filter would have to sustain accurate awareness of vehicle angular motion and at the same time suppress noisy and erroneous data. In terms of high dynamics, the filter would need a very fast sample rate, on the order of 100 Hz, and would have to perform all computations within this short sample period; therefore, the direct method is not practical for highly dynamic in-flight use [16].

For the indirect method, the errors in the INS and GPS comprise the estimated variables in the Kalman filter, and the measurement presented to the filter is the difference between the INS and GPS data. The INS follows the high frequency motions of the vehicle very accurately, and there is no need to model these dynamics explicitly in the filter. Instead, the inertial system *error* dynamics are modeled, which are relatively well developed, well behaved, and low frequency. Because the filter is now out of the INS loop and the error dynamics are low frequency, the filter sample rate can be much lower than that of the direct filter case [16].

The indirect feedback configuration compares the INS and GPS data and uses the result to estimate the errors in the inertial system. The estimated errors are fed back into the INS to correct it. In this scheme, the inertial errors are not allowed to grow unchecked, and the adequacy of a linear model is enhanced. However the INS is dependent on the Kalman filter estimates.

The indirect feedforward mechanization also compares the INS and GPS data, and uses the result to estimate the errors in the inertial system, but the estimated errors are then subtracted from the inertial data (external to the INS itself) to obtain the optimal estimates of position, velocity, and attitude. The inertial system is unaware of the existence of the filter or the GPS data, so if either should fail, the unaltered INS information would still be available [16].

1.1.5.1 Cascaded Filter Approach. The cascaded filter approach to GPS/INS integration is so named because its two Kalman filters are arranged in series. The output of the first filter is the input to the second filter (filter-driving-filter). The outputs of the first filter have time-correlated noise and noise correlated with the measured states and, since the Kalman filter expects uncorrelated measurement noise, the measurements to the second filter could lead to filter stability problems. When the measurement noises are correlated the Kalman filter becomes over-confident, thus putting too much weight on its system model, and too little weight on the measurement inputs. Although modelling colored noise can be done with shaping filters this is generally not done. Compensation for the mismodelling is accomplished by processing the first filter's outputs less frequently than they are available, such that the time correlation between the second filter's measurement error inputs are sufficiently reduced. This places restrictions on the measurement processing rate [4].

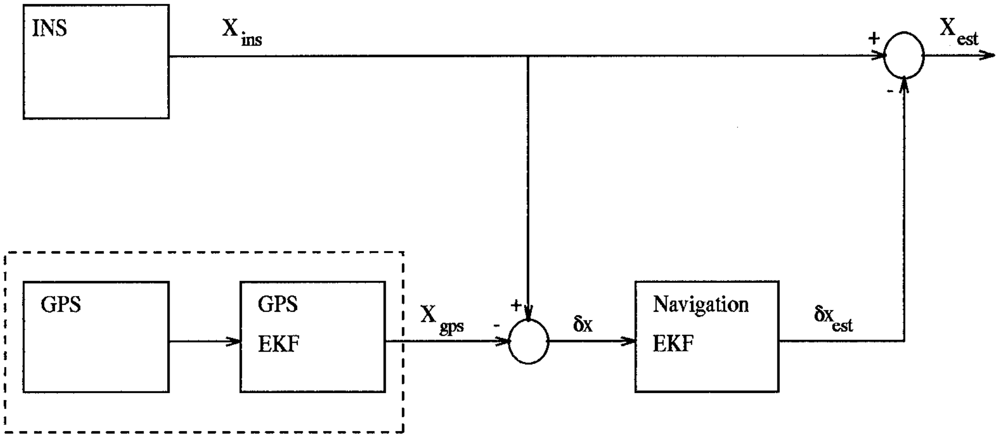

Figure 1.1 Indirect Feedforward Cascaded Filter Integration

Cascaded approaches were the earlier stages of integration, due to the desire to integrate already existing stand alone devices. Thus, in general, the cascaded approach uses the dedicated GPS Kalman filter to produce a position and velocity solution (see Figure 1.1). This GPS solution along with the INS position and velocity solution was the input to the second navigation filter, which provided an estimated navigation solution more accurate than the INS or GPS solution alone. If the GPS receiver has fewer than four satellites in view, the GPS filter outputs degrade and

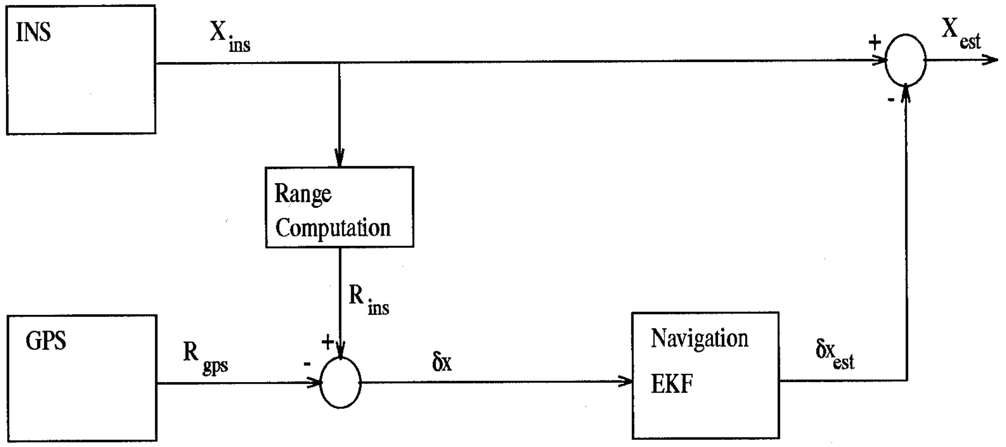

Figure 1.2 Indirect Feedforward Centralized Filter Integration

closely track the rapidly growing INS errors. To avoid even greater filter instability under these conditions the navigation filter is designed to disregard inputs from the GPS receiver when fewer than four satellites are available [13].

1.1.5.2 Centralized Filter Approach. The centralized filter approach to GPS/INS integration utilizes a single Kalman filter. Since only one filter is used, the instability problem with the filter driving filter configuration is nonexistent. The GPS receiver raw data, pseudo-range and pseudo-range rate, are combined with the INS data and used as measurements in the Kalman filter for estimating the error in the inertial system (see Figure 1.2). In this sense, the GPS data can continue to be used in the navigation solution when fewer than four satellites are received [3]. Further advantages can be attained if the GPS receiver is embedded into the INS. First, there will be no need for costly TEMPEST secure communication links when the classified GPS Y-code is being processed. Also, the collocation of the GPS and INS simplifies the interface and communication permitting a tighter control over data timing and latency. The collocated GPS and INS can transfer data via direct memory access and thus the delays usually associated with input/output between the INS and GPS receiver are eliminated [27]. The centralized approach leads to the tightly-coupled integration design used in this thesis.

1.2 Literature Review

This section presents the tight versus loose integration views of four technical papers published in 1989-1994 [4,14,23,30]. These papers analyze a tight and loose GPS/INS integration and recommend which is better. As was discussed in the previous section, there are many ways to set up a tight and loose GPS/INS integration; each paper has its own variation, however the basic concept for the tight integration being to use a single Kalman filter and the loose integration to use two Kalman filters in cascade is the same.

Wei, in his 1990 paper [30], compares the single filter GPS/INS integration with the cascaded filter scheme in which the first filter is a local GPS-only filter. The test data was collected using a road vehicle as the moving base for the INS and GPS. His conclusion is that the accuracy of a sub-optimal cascaded filter does not deteriorate from the centralized filter accuracy. He recommends that the decision for the use of one integration over the other should be made on the basis of computational efficiency, where the cascaded integration is more advantageous, and operationally simpler for the intended application.

Dayton's paper [4] compares a tightly-coupled system, in which the INS is aided by the GPS and the GPS receiver is aided by the INS, with a cascaded system in which the first filter uses measurements from the INS as well as the GPS. In both cases the final error estimates are fed back to the INS, and the INS outputs are used for navigation. Dayton's conclusion is that the tightly-coupled integration "can lead to superior navigation performance due to the tighter integration which allows the system to take full advantage of all available measurements." The cascaded filter, he mentions, was simpler and easier to implement, but the filter driving filter problem places restrictions on measurement processing rate.

Lewantowicz's paper on deep integration [14], analyses a single Kalman filter GPS/INS integration design in simulations. His results are that a "single Kalman filter receiving pseudo-range and delta-range measurements and modelling significant

GPS and barometric altimeter measurement errors can perform significantly better in actual flight than the current cascaded approach." The current cascaded approach refers to the early 1990's GPS/INS implementations used in the U.S. Air Force aircrafts.

Schwarz's 1994 paper [23], compares an embedded GPS (within the INS box) approach with a centralized filter to an aided approach in which the GPS has its own filter which feeds to the second, GPS/INS integration filter. The embedded approach uses velocity feedback from the integrated system to the GPS tracking loop, whereas the cascaded approach has none. The end results of this study showed that the overall performance of the two designs are basically of the same quality.

These papers show the current controversy between tight and loose integration. Loosely-coupled integration is computationally efficient and could be easier to implement into existing airframes, but it is a sub-optimal design. The tightly-coupled integration can achieve more optimality, but the increase in optimality may not be worth the increased computational load and implementation difficulties. These papers as a whole seem to say that the tightly-coupled design can achieve better performance, provided all available measurements are used and the measurement errors are modelled accurately. However, from a practical standpoint, all errors in the real world cannot be accurately modelled, and the number of states needed to model a significant amount of the errors would greatly increase filter computation time. Thus, the true optimality claim of the tightly-coupled integration cannot be achieved in the real world.

1.3 Problem Definition

The research conducted under this thesis will analyze tightly-coupled and loosely-coupled GPS/INS Kalman filter integration schemes using real world data in a stationary environment. This research will use extended Kalman filters with an indirect implementation. The loose integration will consist of two extended Kalman filters cascaded in a feedforward design, as configured in the USAF F-16 fighter aircraft navigation integration. The first filter, referred to as the GPS filter, will incorporate a generic reduced order INS model, a 2-state GPS model, and a single state for the barometric altimeter. The second filter, referred to as the navigation filter, will consist of the LN-93-specific INS model. For this research to obtain meaningful results, an "apples versus apples" comparison between the two integration techniques is essential. Therefore, the tight integration is also feedforward, and its single extended Kalman filter will consist of the LN-93-specific INS model used by the loosely-coupled navigation filter, the 2-state GPS model and the single state barometric altimeter model.

1.4 Scope

This research will concentrate on the comparison of loose and tight GPS/INS integration using actual hardware and data collection from a stationary platform. The current equipment available to this research does not make it feasible for a mobile environment. Post processing techniques will be used; all data will be collected before any integration scheme is applied. The scope of this research is limited by time.

1.5 Assumptions

All theses are limited by the assumptions made, and no research can be adequately evaluated unless these assumptions are clearly defined. This section outlines the assumptions that have been made in this thesis. The numerous assumptions often made with simulations are not needed with the use of real data.

1. The INS platform is assumed to be stabilized with a barometric altimeter, which is the commonly used method for the LN-93 [5]. The simulated output of a barometric altimeter was sent to the LN-93 over the 1553 data bus to stabilize the vertical channel during data collection.

2. The comparison between tight and loose integration will be done with a ten-run Monte Carlo analysis. While a larger run size for the Monte Carlo analysis is preferable, this number of runs was selected due to computer and software limitations.

3. Ergodicity is assumed. The statistics of each run do not change over time. This is necessary using real data and hardware; it allows each consecutive run to be averaged.

1.6 Plan of Attack

This research is divided into two basic components: Data collection of the Litton LN-93 INS and the Navstar XR5-M6 GPS receiver, and the integration and comparison of loose and tight configurations.

1.6.1 GPS/INS Data Collection. The first task is to set up the INS and GPS receiver for data collection. Collecting data from the GPS receiver is easily implemented with the use of a PC computer. All GPS data collection protocol is accomplished with the Navstar Data Monitor software package [21]. The LN-93 INS data collection is also done with a PC computer using the SPEPTRE protocol citeSPEPTRE. Since using a stationary platform and post-processing is used, it is possible to collect data from the GPS receiver and the INS independently without any timing problems. Ten runs of data will be collected for each, the INS and GPS, to provide a ten-run Monte Carlo analysis.

1.6.2 Integration Comparison. The INS and GPS data are integrated using the collection of MATLAB [15] m-files called MATSOFE citematsofe (MATLAB Multimode Simulation for Optimal Filter Evaluation). The routines used by MATSOFE are directly patterned after an established USAF software package, Multimode Simulation for Optimal Filter Evaluation (MSOFE), used to develop and test Kalman filter algorithms [9]. The MATSOFE routines are configured for the loose and the tight GPS/INS integration. The MATSOFE routines are modified to accept the real INS and GPS data where it normally uses simulated data. The same data sets are used as measurements to both the tight and loose integrations. Since the true position and velocity is known the error can be properly defined. Thus, the comparison of the loose and tight integration is focused on the accuracy and fidelity in the position and velocity estimates of each.

1.7 Overview of Thesis

Chapter II provides the theory necessary for this research. The extended Kalman filter, Kalman filter tuning, and algorithms for calculating GPS satellite ECEF position are among the topics presented. The theoretical advantages and disadvantages between the two are also presented. Chapter III describes in detail the loose and tight integrations as used in this thesis, along with the filter models for each. This chapter also provides a simplistic, small order example of a tight and loose integration to give insights that may be clouded by higher order models. Chapter IV presents the results of this study. In Chapter V, conclusions from the information presented and recommendations for further study are discussed.

II. Theory

2.1 Overview

This section presents the theory necessary to accomplish this research. The basic theory and equations associated with the extended Kalman filter are presented, along with a brief discussion of filter tuning. A more rigorous development of many of the Kalman filter subjects can be found in [16–18]. The theory then moves to the method of determining GPS satellite ECEF (earth centered, earth fixed) positions. The GPS satellite ECEF position is needed to determine the INS's range to satellite, which is compared to the GPS receiver's measured pseudo-range. This section concludes with the theoretical advantages and disadvantages of the cascaded and centralized integration techniques.

2.2 Extended Kalman Filter Equations

A Kalman filter is an optimal recursive data processing algorithm that can be shown to be optimal with respect to virtually any criterion that makes sense, given several underlying assumptions. These assumptions are that the system can be described through a linear or linearized model and in which the system and measurement noises are white and Gaussian [16]. One of the Kalman filter's aspects of optimality is that it incorporates all information that can be provided to it. The Kalman filter processes all available measurements, regardless of their precision, to estimate the current value of the variables of interest. The Kalman filter makes these estimates with use of

1. Knowledge of the system and measurement device dynamics;
2. The statistical description of the system noises; and
3. Any available information about initial conditions of the variables of interest.

The GPS receiver and INS error state models consist of sets of linearized state space differential equations and nonlinear measurement equations. These nonlinearities prevent the use of the standard Kalman filter; thus the extended Kalman filter (EKF) is needed. The fundamental idea of the EKF is to relinearize about each estimate once it has been computed [17]. In this manner, provided that deviations from the reference trajectory are small enough, linear perturbation techniques can be employed with adequate results. The subsequent derivation and many of the following equations are taken from Maybeck [17].

The extended Kalman filter can be summarized as follows. Let the nonlinear system of interest be described by the dynamics model

$$\dot{\mathbf{x}}(t) = \mathbf{f}[\mathbf{x}(t), \mathbf{u}(\mathbf{t}), t] + \mathbf{G}(t)\mathbf{w}(t) \tag{2.1}$$

where $\mathbf{x}(t_0)$ is modeled as a (Gaussian) random vector with mean $\hat{\mathbf{x}}_0$ and covariance $\mathbf{P}_0$. $\mathbf{f}[\mathbf{x}(t), \mathbf{u}(t), t]$ is the state dynamics vector which is, in general, a nonlinear function of the state vector $\mathbf{x}(t)$ and time t, and the control input $\mathbf{u}(t)$ (assumed to be zero in this research). $\mathbf{G}(t)$ is a noise distribution matrix which, for this research, is an identity matrix without loss of generality. The vector $\mathbf{w}(t)$ is a white Gaussian noise vector having the statistics of zero-mean:

$$E\{\mathbf{w}(t)\} = \mathbf{0} \tag{2.2}$$

and noise strength:

$$E\{\mathbf{w}(t)\mathbf{w}^T(t+\tau)\} = \mathbf{Q}(t)\delta(\tau) \tag{2.3}$$

where $\delta(\tau)$ is the Dirac delta function.

The Kalman filter incorporates sampled-data measurement information from external measuring devices. The discrete-time measurements are modeled as:

$$\mathbf{z}(t_i) = \mathbf{h}[\mathbf{x}(t_i), t_i] + \mathbf{v}(t_i) \tag{2.4}$$

where $\mathbf{z}(t_i)$ is the measurement available at time t_i, and $\mathbf{h}$ is a known vector which is a function of the state and time. The vector $\mathbf{v}(t_i)$ is a white Gaussian noise sequence having the statistics of zero-mean:

$$E\{\mathbf{v}(t_i)\} = \mathbf{0} \tag{2.5}$$

and noise covariance:

$$E\left\{\mathbf{v}(t_i)\mathbf{v}^T(t_j)\right\} = \begin{cases} \mathbf{R}(t_i) & t_i = t_j \\ \mathbf{0} & t_i \neq t_j \end{cases} \tag{2.6}$$

Recalling the basic assumption of the conventional Kalman filter that the system is linear, the nonlinear Equations (2.1) and (2.4) must be linearized. The following derivation is the linearization of these two equations using the linearization method described in [17]:

1. Assume that a nominal state trajectory, $\mathbf{x}_n(t)$, may be generated for all time of concern, starting from the initial condition $\mathbf{x}_n(t_0) = \mathbf{x}_{n0}$ and satisfying the deterministic differential equation:

$$\dot{\mathbf{x}}_n(t) = \mathbf{f}[\mathbf{x}_n, \mathbf{u}(t), t] \tag{2.7}$$

where $\mathbf{f}[\cdot,\cdot,\cdot]$ is specified in Equation (2.1), and $\mathbf{u}(t) = \mathbf{0}$.

2. The sequence of nominal measurements associated with the nominal state trajectory is given by:

$$\mathbf{z}_n(t_i) = \mathbf{h}[\mathbf{x}_n(t_i), t_i] \tag{2.8}$$

3. The perturbation of the state from the assumed nominal trajectory is the difference of Equation (2.7) end Equation (2.1):

$$\dot{\mathbf{x}}(t) - \dot{\mathbf{x}}_n = \mathbf{f}[\mathbf{x}(t), \mathbf{u}(t), t] - \mathbf{f}[\mathbf{x}_n(t), \mathbf{u}(t), t] + \mathbf{G}(t)\mathbf{w}(t) \tag{2.9}$$

4. The equation above may be approximated to first order by a Taylor series expansion:

$$\delta\dot{\mathbf{x}}(t) = \mathbf{F}[t; \mathbf{x}_n(t)]\delta\mathbf{x}(t) + \mathbf{G}(t)\mathbf{w}(t) \tag{2.10}$$

   where $\delta\mathbf{x}(t)$ represents a first-order approximation to the process $[\mathbf{x}(t) - \mathbf{x}_n(t)]$, and $\mathbf{F}[t; \mathbf{x}_n(t)]$ is the matrix of partial derivatives of $\mathbf{f}$ with respect to its first argument, evaluated along the nominal trajectory:

$$\mathbf{F}[t; \mathbf{x}_n(t)] = \left.\frac{\partial \mathbf{f}[\mathbf{x}(t), t]}{\partial \mathbf{x}}\right|_{\mathbf{x}=\mathbf{x}_n(t)} \tag{2.11}$$

5. In a similar manner, the measurement perturbation equation can be derived and is expressed as:

$$\delta\mathbf{z}(t_i) = \mathbf{H}[t_i; \mathbf{x}_n(t_i)]\delta\mathbf{x}(t_i) + \mathbf{v}(t_i) \tag{2.12}$$

   where the matrix $\mathbf{H}$ is defined as:

$$\mathbf{H}[t_i; \mathbf{x}_n(t_i)] = \left.\frac{\partial \mathbf{h}[\mathbf{x}, t_i]}{\partial \mathbf{x}}\right|_{\mathbf{x}=\mathbf{x}_n(t_i)} \tag{2.13}$$

The nonlinear dynamics and measurement equations have been linearized to form perturbation or error state equations. This linearization process allows for the application of a linearized Kalman filter for the system described by Equations (2.10) and (2.12). The output of the filter would be the estimate of $\delta\mathbf{x}(t)$, denoted as $\hat{\delta\mathbf{x}}(t)$.

An estimate of the total state can be computed using:

$$\hat{\mathbf{x}}(t) = \mathbf{x}_n(t) + \delta\hat{\mathbf{x}}(t) \tag{2.14}$$

The expression above for the linearized Kalman filter is computationally advantageous compared to an "optimal" nonlinear filter. Even appropriate higher-order nonlinear filters include higher order terms from the Taylor series expansion of **f** and **h**, thus imposing a severe computational disadvantage. However, unacceptable errors will result with the linearized Kalman filter if the "true" and nominal trajectories differ significantly. To avoid this problem, the extended Kalman filter is used. The EKF relinearizes about newly declared nominals at each sample time, to enhance the linearization process. This relinearization of the states about the new nominal trajectory ensures that the deviations from the nominal trajectory will remain small. This validates the earlier assumption and allows for linear perturbation techniques to be employed with adequate results.

The extended Kalman filter equations are summarized below. The state estimate and covariance are propagated from time t_i to the next sample time t_{i+1} by integrating the following equations:

$$\dot{\hat{\mathbf{x}}}(t/t_i) = \mathbf{f}[\hat{\mathbf{x}}(t/t_i), \mathbf{u}(t), t] \tag{2.15}$$

$$\dot{\mathbf{P}}(t/t_i) = \mathbf{F}[t; \hat{\mathbf{x}}(t/t_i)]\mathbf{P}(t/t_i) + \mathbf{P}(t/t_i)\mathbf{F}^T[t; \hat{\mathbf{x}}(t/t_i)] + \mathbf{G}(t)\mathbf{Q}(t)\mathbf{G}^T(t) \tag{2.16}$$

where the notation (t/t_i) stands for "at time, t, based on measurements up through time t_i," and where:

$$\mathbf{F}[t; \mathbf{x}(t/t_i)] = \left.\frac{\partial \mathbf{f}[\mathbf{x}, t]}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}=\hat{\mathbf{x}}(t/t_i)} \tag{2.17}$$

and the initial conditions are:

$$\hat{\mathbf{x}}(t_i/t_i) = \hat{\mathbf{x}}(t_i^+) \tag{2.18}$$

$$\mathbf{P}(t_i/t_i) = \mathbf{P}(t_i^+) \tag{2.19}$$

where the superscript, +, indicates the value at a time after the incorporation of a measurement.

With the incorporation of the measurement, $\mathbf{z}_i$, the EKF measurement update equations are:

$$\mathbf{K}(t_i) = \mathbf{P}(t_i^-)\mathbf{H}^T[t_i;\hat{\mathbf{x}}(t_i^-)]\{\mathbf{H}[t_i;\hat{\mathbf{x}}(t_i^-)]\mathbf{P}(t_i^-)\mathbf{H}^T[t_i;\hat{\mathbf{x}}(t_i^-)] + \mathbf{R}(t_i)\}^{-1} \tag{2.20}$$

$$\hat{\mathbf{x}}(t_i^+) = \hat{\mathbf{x}}(t_i^-) + \mathbf{K}(t_i)\left\{\mathbf{z}_i - \mathbf{h}[\hat{\mathbf{x}}(t_i^-), t_i]\right\} \tag{2.21}$$

$$\mathbf{P}(t_i^+) = \mathbf{P}(t_i^-) - \mathbf{K}(t_i)\mathbf{H}[t_i;\hat{\mathbf{x}}(t_i^-)]\mathbf{P}(t_i^-) \tag{2.22}$$

where $\mathbf{H}[t;\hat{\mathbf{x}}(t_i^-)]$ is defined in Equation (2.13), and the superscript, −, indicates a value at a time just before incorporation of a measurement.

2.3 Kalman Filter Tuning

The objective of filter tuning is to achieve the best possible estimation performance from a filter of specified structural form (totally specified except for $\mathbf{P}_0$ and the time histories of $\mathbf{Q}$ and $\mathbf{R}$). These tunable matrices, $\mathbf{Q}$ and $\mathbf{R}$, not only account for actual noises and disturbances in the physical system, but also are a means of declaring how adequately the assumed model represents the "real world" system. The less accurate the model, the stronger the noise strengths should be set. In tuning the filter, the $\mathbf{P}_0$ matrix is a determining factor in the initial transient performance of the filter, whereas the $\mathbf{Q}$ and $\mathbf{R}$ histories dictate the longer term or "steady state" performance and time duration of transients [16].

The process noise strength, $\mathbf{Q}$, and measurement noise covariance, $\mathbf{R}$, must be appropriately tuned for the EKF to track the INS errors accurately. Increasing $\mathbf{Q}$ would indicate either stronger noise driving the dynamics or increased uncertainty in the adequacy of the model itself. This will increase the rate of growth of the $\mathbf{P}(t)$

Table 2.1 Ephemeris Representation Parameters

M_0	Mean anomaly at reference time
Δn	Mean motion difference from computed value
e	Eccentricity
$\sqrt{A}$	Square root of the semi-major axis
Ω_0	Right Ascension at reference time
i_0	Inclination angle at reference time
ω	Argument of perigee
$\dot{\Omega}$	Rate of right ascension
C_{uc}	Amplitude of the cosine harmonic correction term to the argument of latitude
C_{us}	Amplitude of the sine harmonic correction term to the argument of latitude
C_{rc}	Amplitude of the cosine harmonic correction term to the orbit radius
C_{rs}	Amplitude of the sine harmonic correction term to the orbit radius
C_{ic}	Amplitude of the cosine harmonic correction term to the angle of inclination
C_{is}	Amplitude of the sine harmonic correction term to the angle of inclination
t_0	Ephemeris reference time
AODW	Age of Data Word

elements between measurement times and also of the steady state values of $\mathbf{P}(\mathbf{t}_\mathrm{i}^-)$ and $\mathbf{P}(\mathbf{t}_\mathrm{i}^+)$, resulting in the measurements being weighted more heavily. Increasing $\mathbf{R}$ would indicate that the measurements are subjected to a stronger corruptive noise or that the measurement model is less dependable, and so the measurements should be weighted less by the filter [16].

2.4 GPS Satellite Positioning Determination

The indirect feedforward GPS/INS integrations (see Figures 1.1 and 1.2) showed the need for range computations to obtain the INS-predicted range to each GPS satellite. This range computation requires the ECEF position of the GPS satellites. This section will present the algorithm used to compute the ECEF positions using GPS satellite ephemeris data.

The GPS satellite ephemeris data (see Table 2.1) contains the parameters which describe the satellite orbit for a one-hour interval of time. The ephemeris data is in the form of Keplerian parameters, which are used to determine the satellite's

Keplerian orbit. Kepler's equation citeGPSHNDBK is as follows:

$$E(t) = M(t) + esin[E(t)] \tag{2.23}$$

where:

$$\begin{aligned} E(t) &= \text{Eccentric anomaly} \\ M(t) &= \text{Mean anomaly} \\ e &= \text{Eccentricity of the orbit} \end{aligned}$$

Solving for $E(t)$ is impractical in any way except approximately because the exact solution, for $e \leq 0.663$, is

$$E(t) = M(t) + 2\sum_{k=1}^{\infty} \frac{1}{k!} J_k(ke) sin[kM(t)] \tag{2.24}$$

where J_k are Bessel functions of the first kind of order k [28]. In this thesis, the solution was found using successive substitutions to solve Kepler's equation. The equations used to solve for the true anomaly [28], $v(t)$, are

$$sin[v(t)] = \frac{\sqrt{1-e^2} sin[E(t)]}{1 - ecos[E(t)]} \tag{2.25}$$

$$cos[v(t)] = \frac{cos[E(t)] - e}{1 - ecos[E(t)]} \tag{2.26}$$

The ECEF positions of the GPS satellites are found by first solving for the mean motion, n_0, using the semi-major axis of the orbit, A, and the WGS-84 value of the earth's universal gravitational parameter $\mu = 3.986005 \times 10^{14}$ meters/second:

$$n_0 = \sqrt{\frac{\mu}{A^3}} \tag{2.27}$$

The corrected mean motion is then determined by

$$n = n_0 + \Delta n \tag{2.28}$$

where Δn is available in the ephemeris. The time since reference epoch is computed from the difference in actual time and the ephemeris reference time:

$$t_k = t - t_0 \tag{2.29}$$

The mean anomaly at time t_k, the time since reference time t_0, is then found by

$$M_k = M_0 + nt_k \tag{2.30}$$

where M_0 is the mean anomaly at the reference time. Once the mean anomaly is obtained, Kepler's equation can be iteratively solved with the following equation:

$$E_{k+1} = M_k + esin[E_k] \tag{2.31}$$

where the initial eccentric anomaly, E_0, is set to M_0. The true anomaly, v_k, is then calculated from Equations (2.25) and (2.26). Using the true anomaly, v_k, the argument of latitude u_k, radius r_k, and inclination i_k can be determined [2]:

$$u_k = \omega + v_k + C_{us}sin2(\omega + v_k) + C_{uc}cos2(\omega + v_k) \tag{2.32}$$

$$r_k = A[1 - ecos(E_k)] + C_{rc}cos2(\omega + v_k) + C_{rs}sin2(\omega + v_k) \tag{2.33}$$

$$i_k = i_0 + C_{ic}cos2(\omega + v_k) + C_{is}sin2(\omega + v_k) \tag{2.34}$$

where:

C_{uc}, C_{us} = Argument of latitude correction coefficients

C_{rc}, C_{rs} = Orbital radius correction coefficients

C_{ic}, C_{is} = Inclination correction coefficients

ω = Argument of perigee

which are available in the ephemeris. The latitude and radius are then used to determine the satellite's position in the orbital plane:

$$x'_k = r_k cos(u_k) \tag{2.35}$$

$$y'_k = r_k sin(u_k) \tag{2.36}$$

The corrected longitude of the ascending node is found from [2]:

$$\lambda_k = \Omega_0 + (\dot{\Omega} - \omega_e)t_k - \omega_e t_0 \tag{2.37}$$

where Ω_0 and $\dot{\Omega}$ are defined in Table 2.1, and $\omega_e = 7.292115 \times 10^{-5}$ radians/second, is the WGS-84 value of the earth's rotation rate. Using the orbital plane positions, the ECEF positions of the satellites can be computed using:

$$x_k = x'_k cos(\lambda_k) - y'_k cos(i_k) sin(\lambda_k) \tag{2.38}$$

$$y_k = x'_k sin(\lambda_k) + y'_k cos(i_k) cos(\lambda_k) \tag{2.39}$$

$$z_k = y'_k sin(i_k) \tag{2.40}$$

The ECEF frame used in the above equations has the x-axis direction in the true equatorial plane in the direction of the Greenwich meridian and the z-axis along the true earth spin axis, positive in the northern hemisphere, as seen in Figure 2.1.

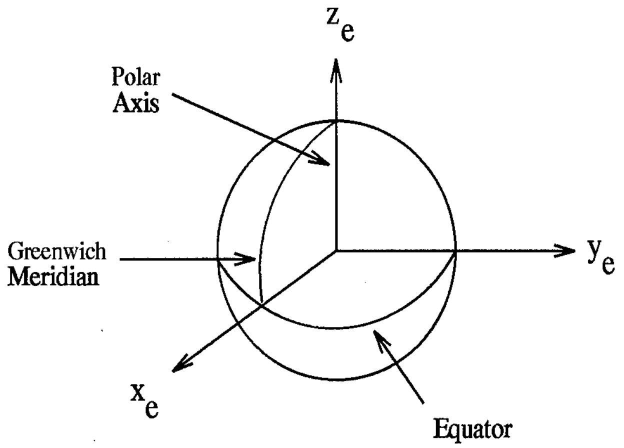

Figure 2.1 Earth Centered Earth Fixed (ECEF) Coordinates

2.5 GPS/INS Integration Theory

This section discusses the theoretical advantages and disadvantages between the centralized and cascaded filter integration techniques. Recall from Section 1.1.5.2, that the centralized filter utilizes all the information at the same time. Thus all the states for the entire system are defined in one global state vector with a corresponding global description of the process noise. The cascaded filter is a two-stage data processing technique. In the first stage, the local filter processes its own data to yield the best possible local estimate. The second stage, the master filter, fuses the local estimates, yielding the best global estimate.

The big advantage the centralized filter has over the cascaded filter is that it can provide the “optimal” solution, in the sense of accuracy. The price for this optimal solution is the high computational load. The cascaded filter, on the other hand, has a reduced computational load but also a suboptimal solution.

Computational load is measured by the number of operations required for one time propagation and one measurement update. Letting n be the dimension of the state vector, s be the dimension of the dynamic driving noise, and m be the dimension of the measurements, then the number of operations can be computed as follows [16]:

Conventional Kalman Filter:

$$\begin{aligned} Adds &\longrightarrow \frac{1}{6}(9n^3 + 3n^2(3m + s - 1) + n(15m + 3s - 6)) \\ Multiplies &\longrightarrow \frac{1}{6}(9n^3 + 3n^2(3m + s + 3) + n(27m + 9s)) \\ Divides &\longrightarrow m \end{aligned}$$

Kalman Filter with U-D Factorization:

$$\begin{aligned} Adds &\longrightarrow \frac{1}{6}(9n^3 + 3n^2(3m + 2s + 2) + 3n(3m + 1)) \\ Multiplies &\longrightarrow \frac{1}{6}(9n^3 + 3n^2(3m + 2s + 7) + 3n(m + 4s - 4) - 6s) \\ Divides &\longrightarrow n(m + 1) - 1 \end{aligned}$$

A comparison of the computational loading between the tight and loose integration used in this thesis is in Chapter 4.

From the practical, real world implementation standpoint, the centralized filter design would be more troublesome to employ. Aircrafts did not initially have GPS technology but did already have inertial navigation systems. The equipment (INS and navigation computer) currently in the aircraft would have to be taken out and redesigned. Thus, the cascaded approach was the simplistic method to incorporate GPS systems into the existing aircrafts. Also, the basic Kalman filter assumptions are usually violated, so it is not clear how "optimal" the centralized filter is when using real world data.

2.6 Chapter Summary

This chapter reviewed the extended Kalman filter theory and the required filter tuning. The EKF is the integration algorithm used in combining the GPS pseudorange and the INS navigation solution. The chapter then presented the equations used to compute the GPS satellite ECEF positions. The chapter concluded with a discussion of the GPS/INS integration theory.

III. Design Methodology and Error Models

This chapter first describes the structure of the tight and loose GPS/INS integration. The chapter then describes the INS, GPS and measurement models used within the extended Kalman filters, and closes with an example using a simple order integration problem.

3.1 GPS/INS Integration Technique

This research compares the loosely-coupled and the tightly-coupled GPS/INS integration techniques. There are many ways to integrate the GPS and INS in loosely-coupled and tightly-coupled configurations. The particular loose GPS/INS integration used in this research is based on the current USAF F-16 GPS/INS integration. To keep a fair comparison, the tight integration maintains the same feedforward configuration and the same filter model states as in the loose. The tightly-coupled configuration is not currently used.

Both integrations use the same INS and GPS. The INS used is a Litton LN-93, and the GPS receiver is a Navstar XR5-M6. The LN-93 is a strapdown INS, with three accelerometers and three ring laser gyroscopes, and has a specification of 0.8 nautical mile per hour drift rate [5]. The XR5-M6 is a six-channel receiver, capable of providing raw pseudorange data.

3.1.1 Loosely-Coupled GPS/INS Integration. The loosely-coupled integration used in this research is based on the current F-16 GPS/INS integration [12]. The F-16 uses a feedback loop which is eliminated for this research. Feedback is not used so that post-processing of the data can be done. Had feedback been used, the integration would have to process in real time for the INS to produce the next navigation solution. Future AFIT research will use feedback when a real-time mobile GPS/INS integration lab is obtained. The loosely-coupled integration used in

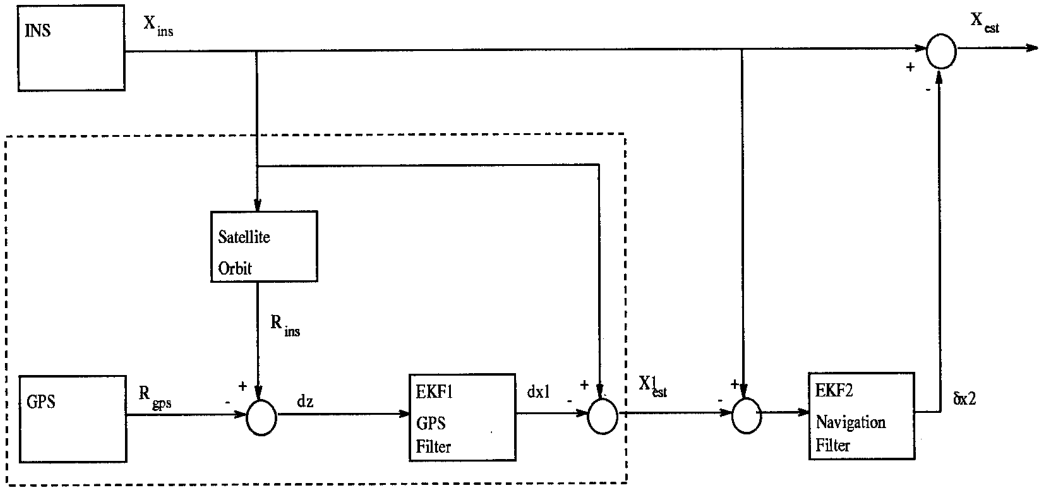

Figure 3.1 Loosely-Coupled GPS/INS Integration

this research is shown in Figure 3.1. The dashed line indicates the GPS equipment (enclosed in one box) added to the existing aircraft navigation system.

The outputs from the INS, X_{ins}, is used with GPS satellite orbit information to calculate R_{ins}, which is the range from the GPS satellite to the INS-indicated position. The GPS satellite orbits are obtained from the ephemeris data being transmitted from each GPS satellite. The GPS receiver's pseudo-range, R_{gps}, is subtracted from the INS range, R_{ins}, and used as measurement input to the first extended Kalman filter, $EKF1$, located in the GPS receiver, referred to as the GPS filter. This filter makes a measurement update at each second, and models a generic INS and GPS, which is the culprit with respect to performance. The output of the filter is an estimate of the generic INS errors, $dx1$, which is subtracted from the actual INS outputs X_{ins} to obtain the first estimate of the INS position and velocity, $X1_{est}$. This estimate is subtracted from the original INS output and used as measurement input to the second extended Kalman filter, $EKF2$, located in the fire control computer and referred to as the navigation filter. This filter has the Litton LN-93-specific INS model and is updated every 10 seconds, but still propagates every second to provide a best estimate navigation solution at the one Hertz rate. The

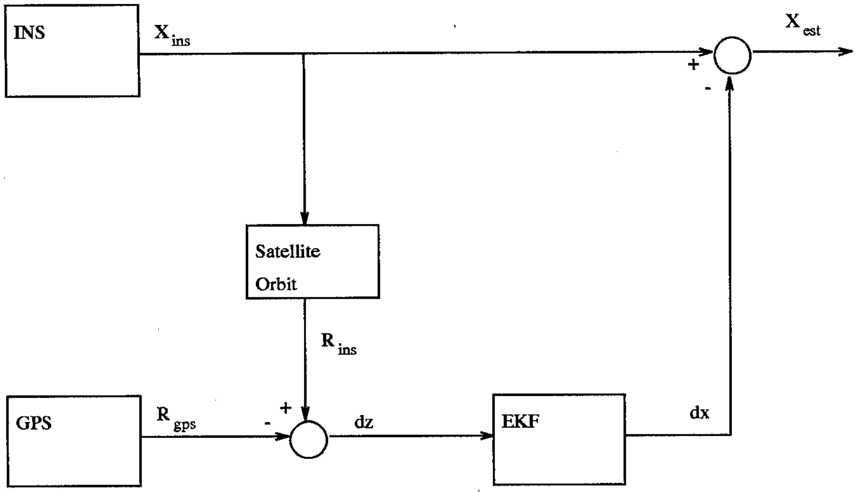

Figure 3.2 Tightly-Coupled GPS/INS Integration

output of the filter is a better estimate of the INS errors which is subtracted from the INS output to obtain the overall estimate of the vehicle position and velocity.

To reduce the filter-driving-filter problem (unmodelled time correlated noise and/or correlated with the measured states), the second extended Kalman filter is updated every 10 seconds whereas the first extended Kalman filter is updated every second. Based on current literature [4,13] and the update rate the F-16's GPS/INS integration uses, the 10-second sample period seemed to be reasonable to prevent filter stability problems. This delay allows the time correlation between measurements to be sufficiently reduced to adequately satisfy the Kalman filter assumption.

3.1.2 Tightly-Coupled GPS/INS Integration. The tightly-coupled integration used in this research is a feedforward configuration consisting of a single extended Kalman filter using the same Litton LN-93-INS-specific states as in the second filter of the loosely-coupled integration, plus the two GPS states. The tightly-coupled integration is shown in Figure 3.2.

The measurements into the extended Kalman filter, EKF, are generated in the same way as for the GPS filter in the loosely-coupled integration. Since the tightly-coupled integration filter models the LN-93-specific INS, the output of the filter is already the best estimate of the errors. These estimates are subtracted from the INS outputs to give the estimated position and velocity. The tightly-coupled extended Kalman filter is updated every second.

3.2 Filter Error Models

The filter models described in this section are based on the F-16 GPS/INS integration filter models, which are derived from the 93-state Litton model. This section begins with a description of Litton 93-state model [5]. The section then describes the models used in the loosely-coupled and tightly-coupled integration. In these error models, the states are referenced against an (X,Y,Z) earth-fixed orthogonal coordinate system, where Y is along the spin axis of the earth and (Z,X) lie on the equatorial plane with Z passing through the Greenwich meridian.

3.2.1 The 93-State LN-93 Error Model. The Litton 93-state model was derived as a truth model for the LN-93 inertial navigation unit. These 93 error states are broken down into six categories as follows:

$$\delta\mathbf{x} = \left[\delta\mathbf{x}_1^T \delta\mathbf{x}_2^T \delta\mathbf{x}_3^T \delta\mathbf{x}_4^T \delta\mathbf{x}_5^T \delta\mathbf{x}_6^T\right]^T \tag{3.1}$$

where:

$\delta\mathbf{x_1}$ contains the first 13 states, which are position, velocity, attitude, and vertical channel errors. These states are classified as "general" errors corresponding to standard Pinson error model [25] states and states associated with barometric altimeter aiding of the vertical channel.

$\delta\mathbf{x_2}$ represents the correlated errors and "trend" states and are modeled as first-order Markov processes in the system truth model. This category is composed of the 16 gyro, accelerometer and baro-altimeter errors.

$\delta\mathbf{x_3}$ consists of gyro bias errors, which are modeled as random constants.

$\delta\mathbf{x_4}$ is also modeled as random constants and is made up of the accelerometer bias errors.

$\delta \mathbf{x}_5$ is a set of first order Markov processes and is composed of the six accelerometer and gyro initial thermal transients.

$\delta \mathbf{x}_6$ is composed of the 18 gyro compliance error states. These states are modeled as biases in the system truth model.

The 93-state Litton model state space differential equation is given by:

$$\begin{bmatrix} \delta\dot{\mathbf{x}}_1 \\ \delta\dot{\mathbf{x}}_2 \\ \delta\dot{\mathbf{x}}_3 \\ \delta\dot{\mathbf{x}}_4 \\ \delta\dot{\mathbf{x}}_5 \\ \delta\dot{\mathbf{x}}_6 \end{bmatrix} = \begin{bmatrix} \mathbf{F}_{11} & \mathbf{F}_{12} & \mathbf{F}_{13} & \mathbf{F}_{14} & \mathbf{F}_{15} & \mathbf{F}_{16} \\ \mathbf{0} & \mathbf{F}_{22} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{F}_{55} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \delta\mathbf{x}_1 \\ \delta\mathbf{x}_2 \\ \delta\mathbf{x}_3 \\ \delta\mathbf{x}_4 \\ \delta\mathbf{x}_5 \\ \delta\mathbf{x}_6 \end{bmatrix} + \begin{bmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix} \tag{3.2}$$

This information was taken from the Litton LN-93 Error Budget [11]. This model is the most detailed model available for the LN-93 as well as the LN-93 inertial navigation units.

3.2.2 Loosely-Coupled Error Model. As stated previously, the loose integration requires two filters, each with a separate model. The GPS filter model consists of 12 states (see Table 3.1). The first nine states are the standard Pinson error states to model any generic INS. The tenth state is the barometric state used for vertical channel stabilization. The last two states are the GPS states used for modelling the largest GPS errors, user clock bias and clock drift.

The second filter, the navigation filter, is used to model a specific INS, the LN-93. This filter consists of 25 states as shown in Table 3.2. The states are the standard Pinson error states, the barometric altimeter error state, nine gyroscope error states, and six accelerometer error states. These states are similar to what

Table 3.1 12-State Filter, Generic INS

State Symbol	Definition	LN-93 State	Loose State
$\delta\Theta_X$	X component of vector angle from true to computer frame	1	1
$\delta\Theta_Y$	Y component of vector angle from true to computer frame	2	2
$\delta\Theta_Z$	Z component of vector angle from true to computer frame	3	3
ϕ_X	X component of vector angle from true to platform frame	4	4
ϕ_Y	Y component of vector angle from true to platform frame	5	5
ϕ_Z	Z component of vector angle from true to platform frame	6	6
δV_X	X component of error in computer velocity	7	7
δV_Y	Y component of error in computer velocity	8	8
δV_Z	Z component of error in computer velocity	9	9
δh_C	Barometer correlated bias noise error	23	10
δR_{uclk}	GPS user clock bias	N/A	11
δD_{uclk}	GPS user clock drift	N/A	12

the F-16 uses in its navigation filter, except the vertical states 3, 9, and 10 are added. These vertical states are needed for an adequate model in the single filter of the tightly-coupled integration, but are not necessary for the second filter of the loosely-coupled integration [22]. However, to keep a fair comparison, these states are included in the navigation filter of the loose integration.

3.2.3 Tightly-Coupled Error Model. The tightly-coupled integration has only one filter modelling the LN-93-specific INS and the GPS. The LN-93 states are the same as in the navigation filter of the loose integration. The two GPS states are the same as the GPS states in the first filter of the loose integration. Together the LN-93 states and the GPS states create the 27-state tightly-coupled Kalman filter error model (see Table 3.3).

Table 3.2 25-State Filter, LN-93 Specific

State Symbol	Definition	LN-93 State	Loose State
$\delta\Theta_X$	X component of vector angle from true to computer frame	1	1
$\delta\Theta_Y$	Y component of vector angle from true to computer frame	2	2
$\delta\Theta_Z$	Z component of vector angle from true to computer frame	3	3
ϕ_X	X component of vector angle from true to platform frame	4	4
ϕ_Y	Y component of vector angle from true to platform frame	5	5
ϕ_Z	Z component of vector angle from true to platform frame	6	6
δV_X	X component of error in computer velocity	7	7
δV_Y	Y component of error in computer velocity	8	8
δV_Z	Z component of error in computer velocity	9	9
δh_C	Barometer correlated bias noise error	23	10
b_x	X-component of gyro drift repeatability	30	11
b_y	Y-component of gyro drift repeatability	31	12
b_z	Z-component of gyro drift repeatability	32	13
χ_1	X-gyro misalignments about Y axis	36	14
χ_2	Y-gyro misalignments about X axis	37	15
χ_3	Z-gyro misalignments about X axis	38	16
ν_1	X-gyro misalignments about Z axis	39	17
ν_2	Y-gyro misalignments about Z axis	40	18
ν_3	Z-gyro misalignments about Y axis	41	29
Δb_x	X-component of accelerometer bias repeatability	48	20
Δb_y	Y-component of accelerometer bias repeatability	49	21
Δb_z	Z-component of accelerometer bias repeatability	50	22
S_{A_x}	X-component of accelerometer and velocity quantizer scale factor error	51	23
S_{A_y}	Y-component of accelerometer and velocity quantizer scale factor error	52	24
S_{A_z}	Z-component of accelerometer and velocity quantizer scale factor error	53	25

Table 3.3 27-State Filter, LN-93 Specific

State Symbol	Definition	LN-93 State	Tight State
$\delta\Theta_X$	X component of vector angle from true to computer frame	1	1
$\delta\Theta_Y$	Y component of vector angle from true to computer frame	2	2
$\delta\Theta_Z$	Z component of vector angle from true to computer frame	3	3
ϕ_X	X component of vector angle from true to platform frame	4	4
ϕ_Y	Y component of vector angle from true to platform frame	5	5
ϕ_Z	Z component of vector angle from true to platform frame	6	6
δV_X	X component of error in computer velocity	7	7
δV_Y	Y component of error in computer velocity	8	8
δV_Z	Z component of error in computer velocity	9	9
δh_C	Barometer correlated bias noise error	23	10
δR_{uclk}	GPS user clock bias	N/A	11
δD_{uclk}	GPS user clock drift	N/A	12
b_x	X-component of gyro drift repeatability	30	13
b_y	Y-component of gyro drift repeatability	31	14
b_z	Z-component of gyro drift repeatability	32	15
χ_1	X-gyro misalignments about Y axis	36	16
χ_2	Y-gyro misalignments about X axis	37	17
χ_3	Z-gyro misalignments about X axis	38	18
ν_1	X-gyro misalignments about Z axis	39	19
ν_2	Y-gyro misalignments about Z axis	40	20
ν_3	Z-gyro misalignments about Y axis	41	21
Δb_x	X-component of accelerometer bias repeatability	48	22
Δb_y	Y-component of accelerometer bias repeatability	49	23
Δb_z	Z-component of accelerometer bias repeatability	50	24
S_{A_x}	X-component of accelerometer and velocity quantizer scale factor error	51	25
S_{A_y}	Y-component of accelerometer and velocity quantizer scale factor error	52	26
S_{A_z}	Z-component of accelerometer and velocity quantizer scale factor error	53	27

3.3 Measurement Models

The measurement model for the GPS filter of the loose integration and the only filter of the tight integration are the same. The measurements are made available every second. The measurement for the navigation filter of the loose integration is provided every ten seconds.

The measurement model for the tight integration filter, and for the first filter of the loose integration, are pseudorange difference measurements. The number of measurements depends upon the number of satellites being tracked. The XR5-M6 GPS receiver limits this number to a maximum of six. All received satellites are used in the measurement. The pseudorange measurements received by the GPS receiver are differenced with the INS-computed pseudorange to produce a difference measurement:

$$\delta z = R_{INS} - R_{GPS} \tag{3.3}$$

The pseudorange, R_{GPS}, is the sum of the true range from the user to the satellite plus the errors.

$$R_{GPS} = R_t + \delta R_{cl} + \delta R_{trop} + \delta R_{ion} + \delta R_{uclk} - \delta R_{sclk} - v \tag{3.4}$$

where:

R_{GPS} = GPS pseudorange measurement, from satellite to user

R_t = true range, from satellite to user

δR_{cl} = range error due to code loop error

δR_{trop} = range error due to tropospheric delay

δR_{ion} = range error due to ionospheric delay

δR_{uclk} = range error due to user clock

δR_{sclk} = range error due to satellite clock

v = zero-mean white Gaussian measurement noise

The INS-computed pseudorange measurement is found by differencing the satellite ECEF position from the ECEF position provided by the INS:

$$R_{INS} = |\mathbf{X}_u - \mathbf{X}_s| = \left\| \begin{bmatrix} x_u \\ y_u \\ z_u \end{bmatrix}^e - \begin{bmatrix} x_s \\ y_s \\ z_s \end{bmatrix}^e \right\| \tag{3.5}$$

The above equation can be written as:

$$R_{INS} = \sqrt{(x_u - x_s)^2 + (y_u - y_s)^2 + (z_u - z_s)^2} \tag{3.6}$$

The measurements used for the navigation filter of the loose integration are the position error estimates from the GPS filter. These error estimates are the first three states of the GPS filter. Again, the loose integration's navigation filter processes measurements at the 0.1 Hertz rate, but still provides a navigation solution every second.

3.4 Simple, Low-Order Integration Example

This section presents a low-order integration example. This example will simplify the tight and loose integrations, and provide insights for this research. The example is simulated in Matlab's [15] software extension, Simulink [24].

3.4.1 Setup. The loose and tight integration structure for this example are the same as used for this research (see Figures 3.3 and 3.4). This example however is for the one-dimensional case. The INS is modelled by two integrators, with acceleration and noise inputs. The INS noise is modelled as a first-order Markov process, the output of a first order lag, driven by zero-mean white Gaussian noise. The GPS pseudorange is modelled by a constant plus a white discrete-time process with a normal distribution. The first Kalman filter in the loosely-coupled integration is a two-state filter modelling the two states of the INS. The second Kalman filter is a three-state filter: two states to model the INS integrators, and a third state to model the INS colored noise. The Kalman filter in the tightly-coupled integration is the same as the second filter in the loose integration. The Kalman filter in the tight integration is updated every second. Likewise, the first Kalman filter in the loose is updated at the one second rate; the second filter is updated at the 10 second rate, as is done for the actual hardware integration in this research. A detailed description of this example with the Simulink code can be found in Appendix A.

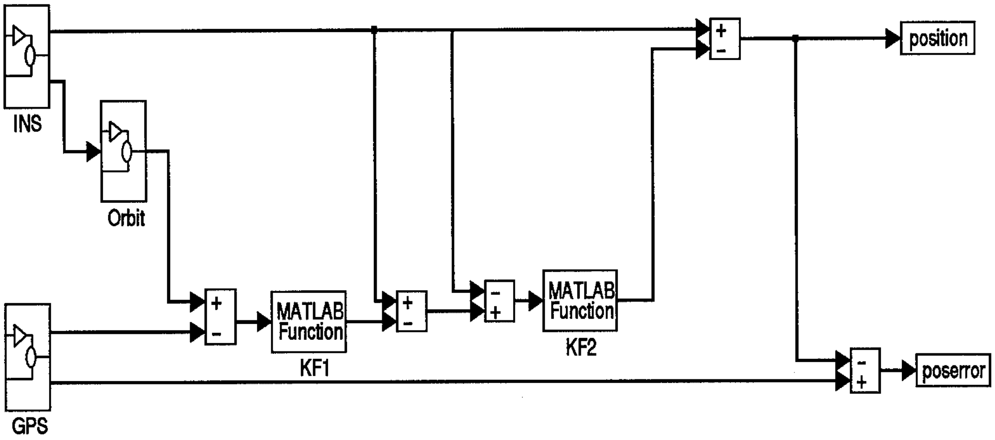

Figure 3.3 Loosely-Coupled Integration Example

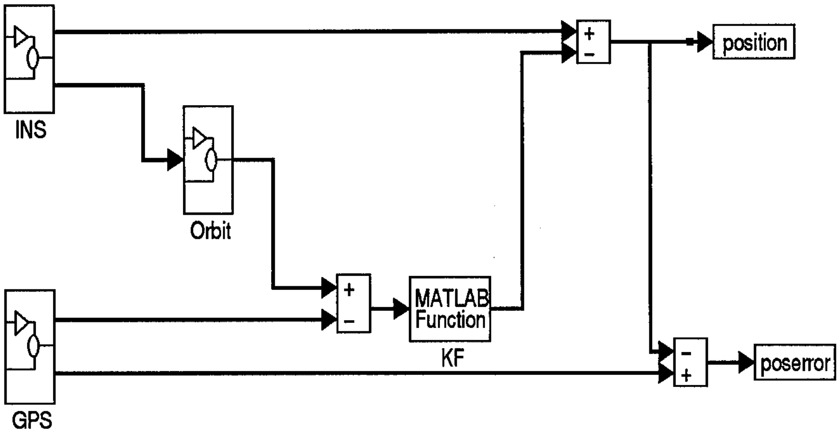

Figure 3.4 Tightly-Coupled Integration Example

3.4.2 Results. The results are from 10 Monte Carlo runs, each consisting of 100 seconds in length (see Figures 3.5 and 3.6). In the plots, the dash-dot line represents the mean filter error, the solid line is the actual mean error ± one sigma (one standard deviation), and the dashed line is the filter-predicted zero mean error ± one sigma. The Kalman filters in both the tight and loose were optimally tuned to reduce position errors. Figure 3.5 shows conservative tuning; however, a less conservative tuning degraded the actual estimates. Analysis of 15 Monte Carlo runs did not provide any further information. Also, run lengths of 1000 seconds presented the same results except the plots were difficult to read.

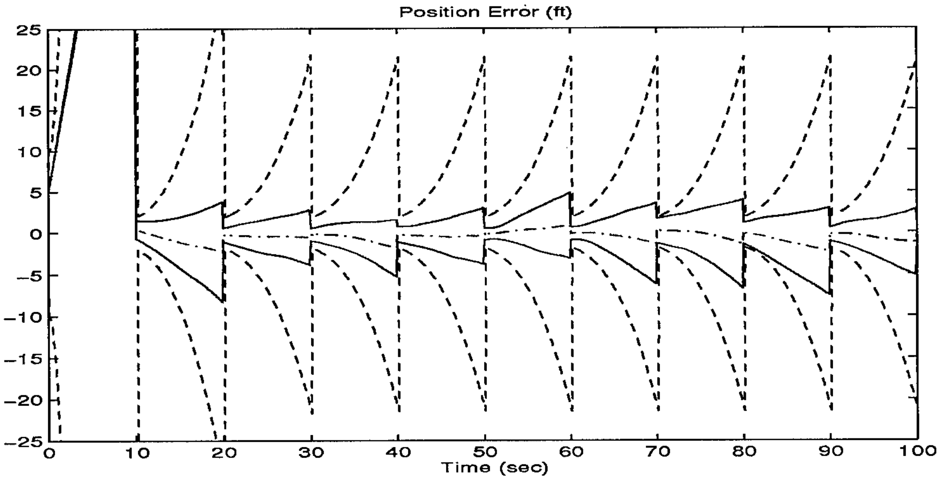

Figure 3.5 Loosely-Coupled Example Position Error

In an effort to compare the two plots, an ad hoc time average is taken. This allows for each plot to be reduced to two meaningful scalars that can be compared. For each plot, the mean value and first sigma is averaged across the length of the runs to provide a time averaged mean error and time averaged standard deviation. For these plots the time average was taken from the 10th through the 100th second mark to allow for the initial transients to die off. The results, in Table 3.4, show the tight integration outperforms the loose integration. Examining the plots, it can be

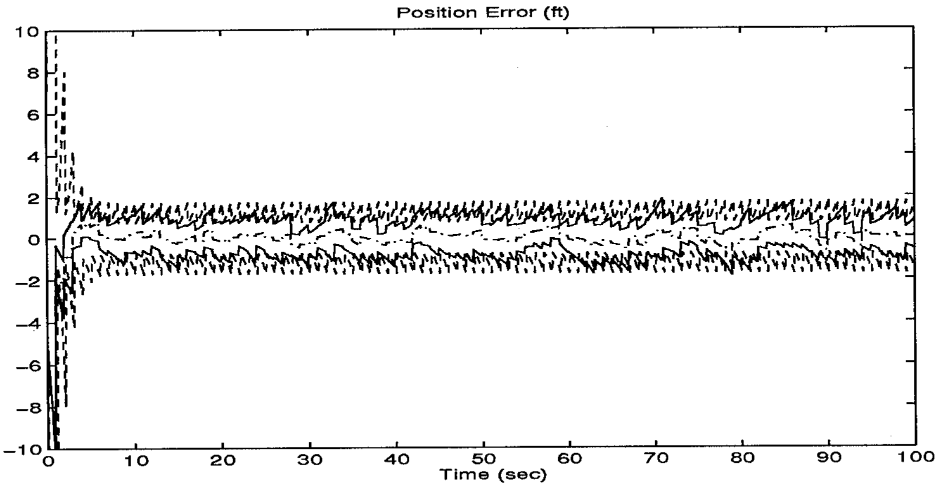

Figure 3.6 Tightly-Coupled Example Position Error

seen the loose integration has a degraded performance mostly due to the propagation cycle being 10 seconds long instead of 1 second, as in the case of the tight integration.

From this simple example, there is an inherent disadvantage for the loosely-coupled integration. However, in the simplicity of this example the tightly-coupled integration is capable of modelling all the characteristics of the INS and GPS to include the noise inputs, which is impossible to do with real world data. Furthermore, the extended Kalman filter must be used in real world applications and is not an optimal filter. Thus, for the real world, the tight integration can never be truly optimal and shows position performance comparable to a loosely-coupled integration to within a few feet.

Table 3.4 Time Averaged Errors

Integration	Mean Error (ft)	One Sigma (ft)
Tight	0.0031	0.8340
Loose	-0.4895	2.4626

3.5 Chapter Summary

This chapter presented the designs used in this research. The integration configurations were detailed. The GPS and INS error models used in the extended Kalman filters were presented, as well as the filter measurement models. The chapter concluded with a simple, low-order example providing some insights for the research. With the integration designs and models described the results can be presented.

IV. Results and Analysis

This chapter presents the results of the GPS/INS integration research. It begins with a section on using real data in the integration instead of simulated data. The chapter then analyzes the INS and the GPS position errors without any aiding or integration. The results of the loose and the tight integrations are then presented, followed by the comparison of the two. The chapter finishes with a section on other comparison issues.

4.1 Integration with Real Data

This section is included because of the various insights that were discovered during this research. Most AFIT research [8, 20, 29] has been done using simulated data with truth models. This is especially significant for the GPS data. The GPS data collected from an actual GPS receiver is not as smooth and refined as the simulated data. For example, the GPS receiver could be tracking a satellite intermittently, or the receiver might be giving bad pseudorange data for a satellite for a few update cycles. In fact, a change in the order of the satellites being tracked and down-loaded to the integration software causes problems. Thus, significant error checking and/or massaging (keeping the satellite data in the same order) of the GPS receiver's data is essential to a satisfactory GPS/INS integration.

The extent of the raw GPS data processing needed, of course, depends on the GPS receiver and how the data is down-loaded for use by the actual GPS/INS integration. Although this research was done in a post-processing environment, the massaging of raw GPS data would be feasible in a real-time situation. The raw GPS data preprocessing needed for this research included a check for bad pseudoranges, and a routine to keep the order of the satellite data the same as the previous sample (to the extent possible). The use of bad pseudoranges in the GPS/INS integration caused large spikes in the navigation solution, sometimes to the extent of not recovering. A change in the order of the satellites also caused spikes in the solution.

Although the integration algorithms had "bad data" checking routines, certain types of corrupt data slipped through and caused wild perturbations in the integration's performance (see Section 4.4.1). To keep the performance comparison between the tight and loose integration simple, all the GPS and INS data sets were free of corruption. Each run of INS data was examined independently so as to prevent any anomalies. Likewise, each run of GPS data was examined independently using a least squares algorithm to check for any corruptions. If a run of either INS or GPS data was found to be corrupted, it was thrown out and a new run of data was collected.

4.2 Data Collection

Ten sets of each the INS and the GPS data was collected. Collection was taken over an approximate 25 minute period, but the analysis covered only the first 20 minutes to ensure full time coverage of data. Since this research is for the stationary case, the GPS and INS data need not, and was not, taken simultaneously. The GPS and INS data was, however, arbitrarily paired with each other and used in each of the loosely-coupled and tightly-coupled integrations with the same pairs.

4.2.1 LN-93 Data. Each set of the LN-93 INS data was collected after an initial alignment phase had been completed. The LN-93 has three alignment modes: gyrocompass alignment, stored heading alignment, and in-flight alignment. The alignment used in this research was the gyrocompass alignment mode. The average position error and its standard deviation of all ten sets of INS data can be

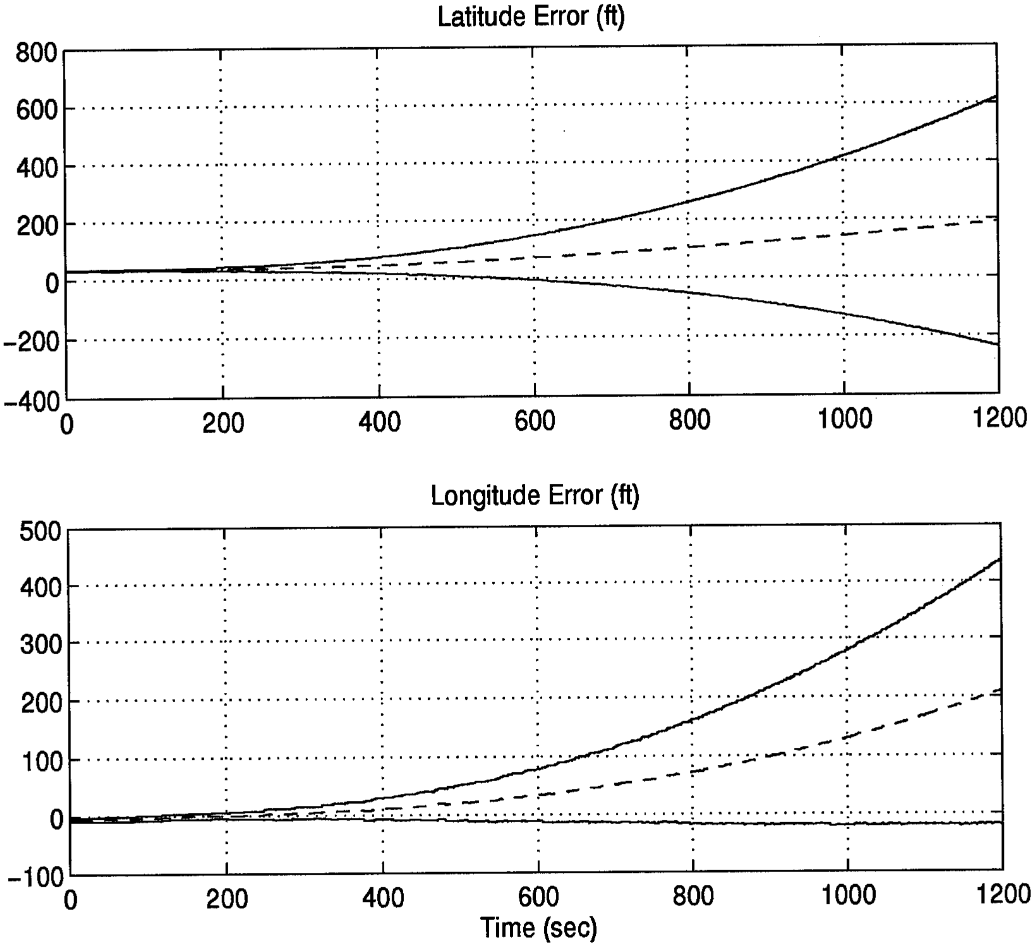

Figure 4.1 LN-93 Position Errors

seen in Figure 4.1. These plots were generated using the same algorithms used in the GPS/INS integration algorithm, and reflect exactly what the INS output with no integration or corrections involved. In these plots the dashed line represents the mean error, and the solid line is the mean error ± one sigma. By extrapolating these plots, it can be seen that the errors are well within the LN-93's 0.8 nm/hr drift rate specification.

The latitude and longitude errors in Figure 4.1 have the normal characteristics of inertial navigation systems. An inertial navigation system has low frequency error characteristics known as the Earth and Foucault rates and the Schuler Frequency oscillations [1]. These low frequency characteristics, along with the gyro drift errors, are the cause for the increasing latitude and longitude errors seen in Figure 4.1. Since the errors continuously grow, it does not make sense to calculate a temporally averaged error, as was done with the example in Chapter 3.

4.2.2 XR5-M6 Data. The GPS data collected was taken over different hours of the day. This provided a variety of satellites received and gave a geometric dilution of precision (GDOP) ranging from 2.25 to 5.12. The GDOP is a measure of the geometrical "strength" of the received GPS satellite configuration. GDOP is computed from the variance of the estimated user position in each axis and in the user time offset [19].

$$GDOP = \sqrt{\sigma_{xx}^2 + \sigma_{yy}^2 + \sigma_{zz}^2 + \sigma_{tt}^2} \tag{4.1}$$

where σ_{xx}, σ_{yy} and σ_{zz} are the variance of the estimated user position in each axis and σ_{tt} is the variance of the estimated user time offset. GDOP changes with time as the satellites travel along their orbits. The value of the GDOP is a multiplier to the measurement accuracy [2].

$$\sigma = GDOP \times \sigma_o \tag{4.2}$$

where σ is the position accuracy and σ_o is the measurement accuracy. Basically, the higher the GDOP value, the worse the position accuracy will be. The average position error and its standard deviation can be seen in Figure 4.2. In these plots the dashed line represents the mean error, and the solid line is the mean error ± one sigma. The temporally averaged error was 246.46 ± 92.76 feet in latitude and 107.99 ± 93.05 feet in longitude. As mentioned earlier, real data is not as smooth

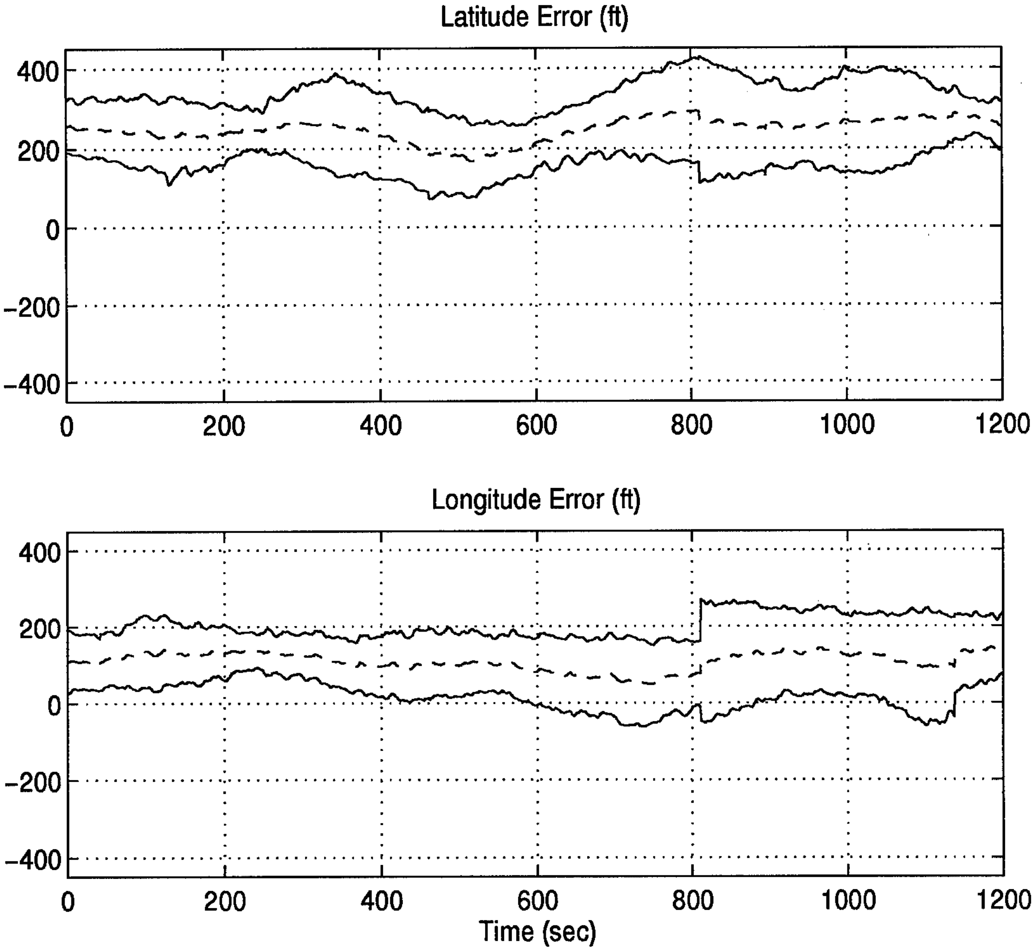

Figure 4.2 XR5-M6 GPS Position Errors

as simulated data. Such a case can be seen as a jump at the 810 second point where one set of GPS data had satellite transitions.

These plots were generated using the same algorithms used in the GPS/INS integration algorithm. This algorithm kept the one-second sample period, performed the same checks for bad pseudorange data, and manipulated the data so that the order of the received satellites was the same or as close to the same order as with the previous set of data. All the GPS data collected, by chance, had at least four satellites under track; had this not been the case, a GPS-only solution would not have been possible. The GPS data also included satellite transitions and satellite drop outs. The data was evaluated for bad pseudorange data and found none. As was mentioned in Section 4.1, any corrupt GPS data was thrown out; in fact, eight of the 19 sets of GPS data collected had a corruption in it.

4.3 Integration Results

This section presents the results of the loose and tight GPS/INS integration using actual hardware and real data. The results presented are position errors in latitude and longitude, and velocity errors along the INS x-axis and y-axis. As stated in the assumptions in Chapter 1, the INS received simulated barometric inputs to the z-axis, so altitude and z-velocity errors are not presented. In all the plots, the dotted line represents the mean filter error, the solid line is the actual mean error ± one sigma (one standard deviation), and the dashed line is the filter-predicted zero mean error ± one sigma.

4.3.1 Filter Tuning. The EKF is very versatile and can be tuned for any situation. The problem is that its performance may be excellent in the environment for which it is tuned, but it may be rather lacking if conditions change, such as for a failure or change in dynamics. In this research the filters were tuned such that the best estimates of position and velocity errors could be attained. It should be noted that the tuning is for this small set of data (10 runs), and for the stationary case. For a more robust filter, tuning should be done in a mobile environment with a large number of Monte Carlo runs (this research was limited due to computer and software limitations).

4.3.2 Tight Integration. The tight integration results are presented in this section. The tight integration position error plots are in Figure 4.3, and the

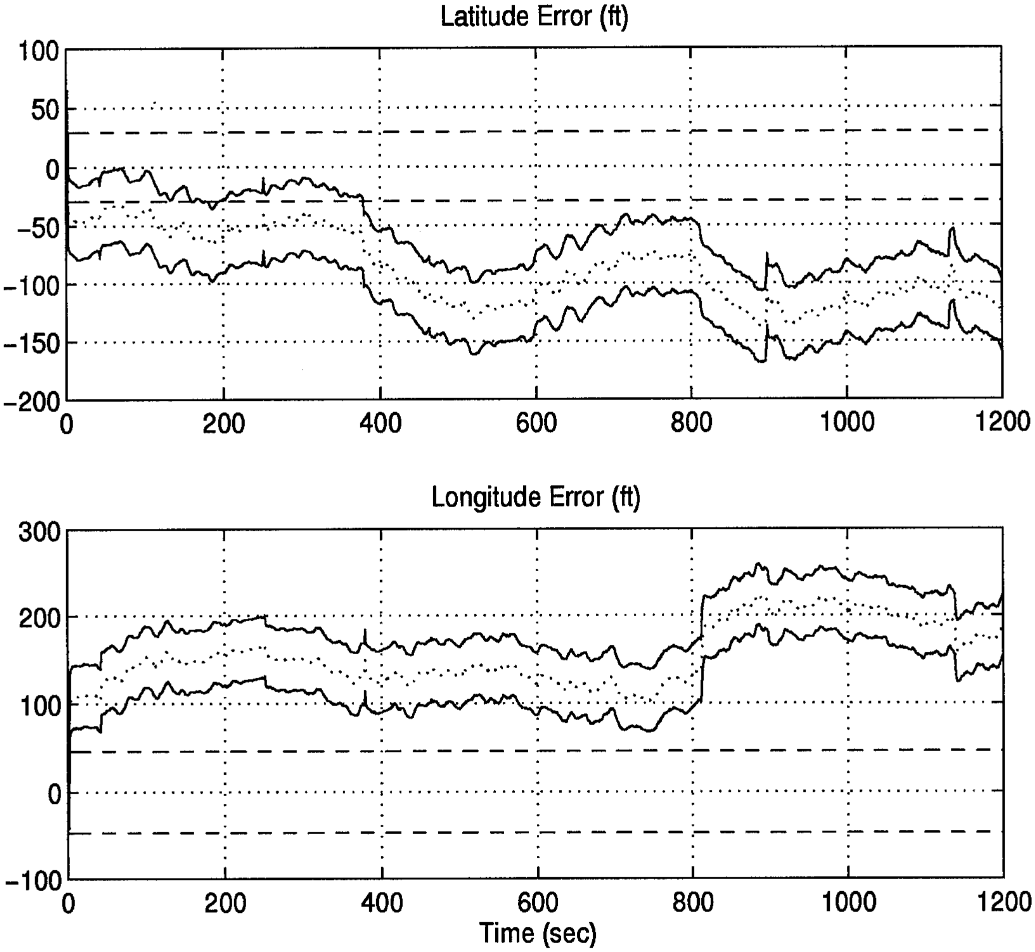

Figure 4.3 Tight Position Errors

velocity error plots are in Figure 4.4. Tuning for the tight integration was easier than for the loose, since there is only one filter. The dynamic driving noise values and measurement noise values are shown in Table B.1 and Table B.2 of Appendix B, respectively. Table B.2 gives the measurement values for the case when only four satellites are used.

In the position error plots, the latitude temporally averaged error is -87.53 ± 31.13 feet. The longitude temporally averaged error is 154.13 ± 35.05 feet. The plots show that the filter and the actual error estimates are stable; the plots are not

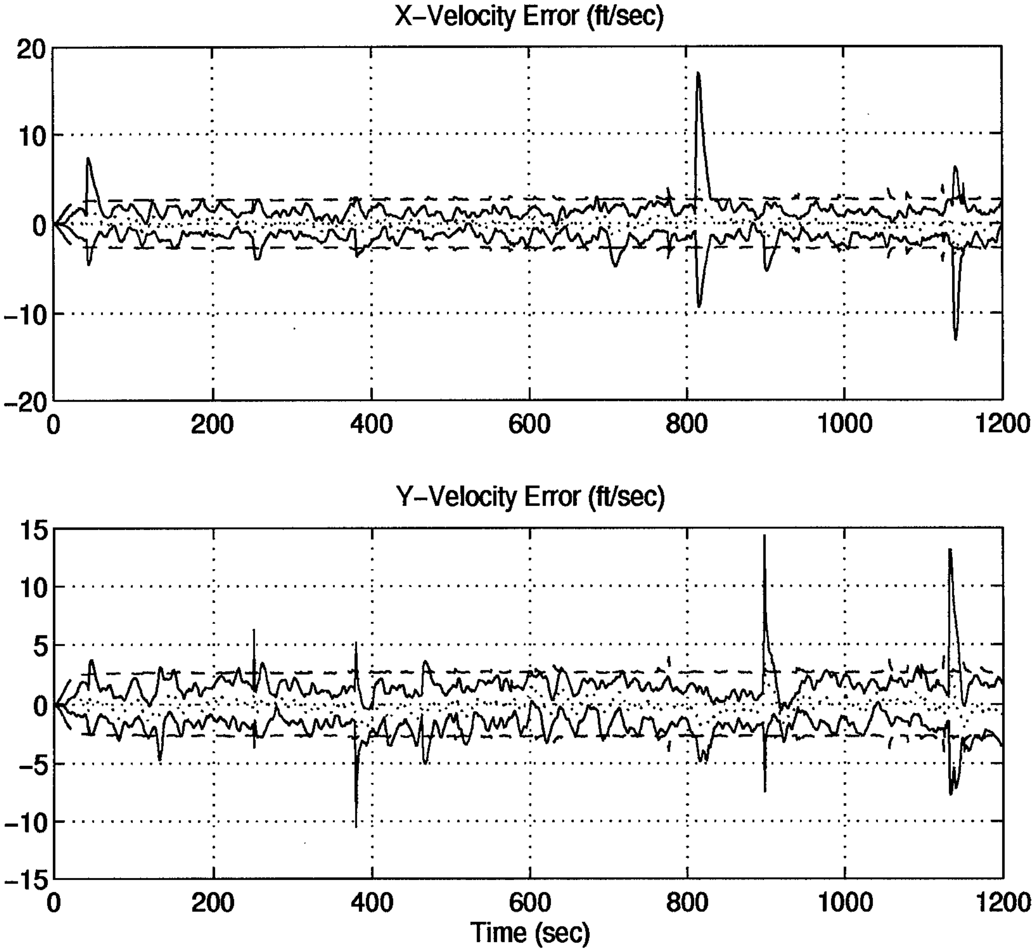

Figure 4.4 Tight Velocity Errors

diverging off as with the position plots of the INS alone, see Figure 4.1. There is, however, an obvious bias in the position error. This is attributed partially to the fact that only 10 Monte Carlo runs were done. Had many more runs been done the bias may be alleviated. The tuning values for the plots were obtained by provided the best position and velocity error estimates.

For comparison purposes, the emphasis is placed on the sigma value of the actual error. A smaller sigma value implies a tighter accuracy. The position sigma values of the tight integration is almost three times smaller than the GPS-alone sigma values. This shows the significance of integration over a GPS system alone.

The velocity error plots also show stability of the filter and of the actual error estimates. The mean values are close to zero. The x-axis velocity temporally averaged error is -0.04233 ± 1.162 feet/second. The y-axis velocity temporally averaged error is -0.1392 ± 1.738 feet/second. The filter is tuned slightly on the conservative side for these velocity components (most of the actual sigma values fall within the filter sigma). The velocity plots are quite accurate, but it should be remembered that this research is for the stationary case, and velocity was not part of the filter measurements. The larger spikes are not extreme and are most likely due to various jumps in the GPS data caused from satellite transitions (notice the spike at the 810 second point where one set of GPS data had a large jump caused by satellite transitions).

4.3.3 Loose Integration. The loose integration position error plots are in Figure 4.5, and the velocity error plots are in Figure 4.6. This integration required the tuning of two filters. In an actual implementation, the first filter (the generic INS filter) would be tuned to give the best performance for all the INS's that would be used with it. For example, the F-16 can be configured with three different INS's, so the generic INS filter would be tuned for all three. In this research, both filters were tuned for best overall results for the single INS used. The dynamic driving noise and measurement noise values, Q and R, that gave the best response are shown in Tables B.3, B.4, B.5, and B.6 of Appendix B.

In the position error plots, the latitude temporally averaged error is -85.48 ± 32.69 feet. The longitude temporally averaged error is 153.49 ± 37.76 feet. These plots show that the filter and the actual error estimates are stable. However, the

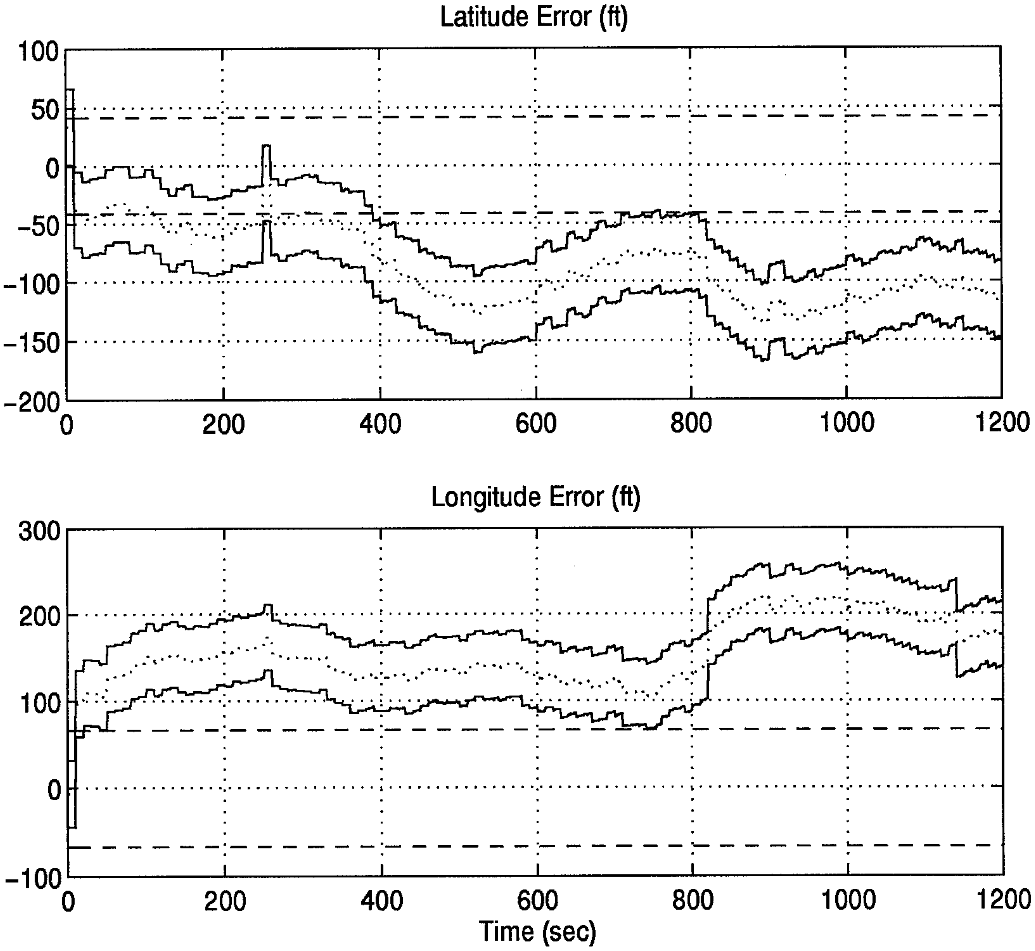

Figure 4.5 Loose Position Errors

bias that was present in the tight integration position error estimates is also seen here. The loose integration position sigma values outperform those of the GPS-only sigma values by about three times. Likewise, the position error is not diverging as the position errors of the INS-alone case does.

The velocity error plots are not as good as for the tight integration. The filter sigma diverge to about 1.25×10^4 feet/second. The actual mean error, however, remain close to zero. The actual sigma grows to about one foot/second for the x-velocity and about 0.5 feet/second for the y-velocity. The x-axis velocity tempo-

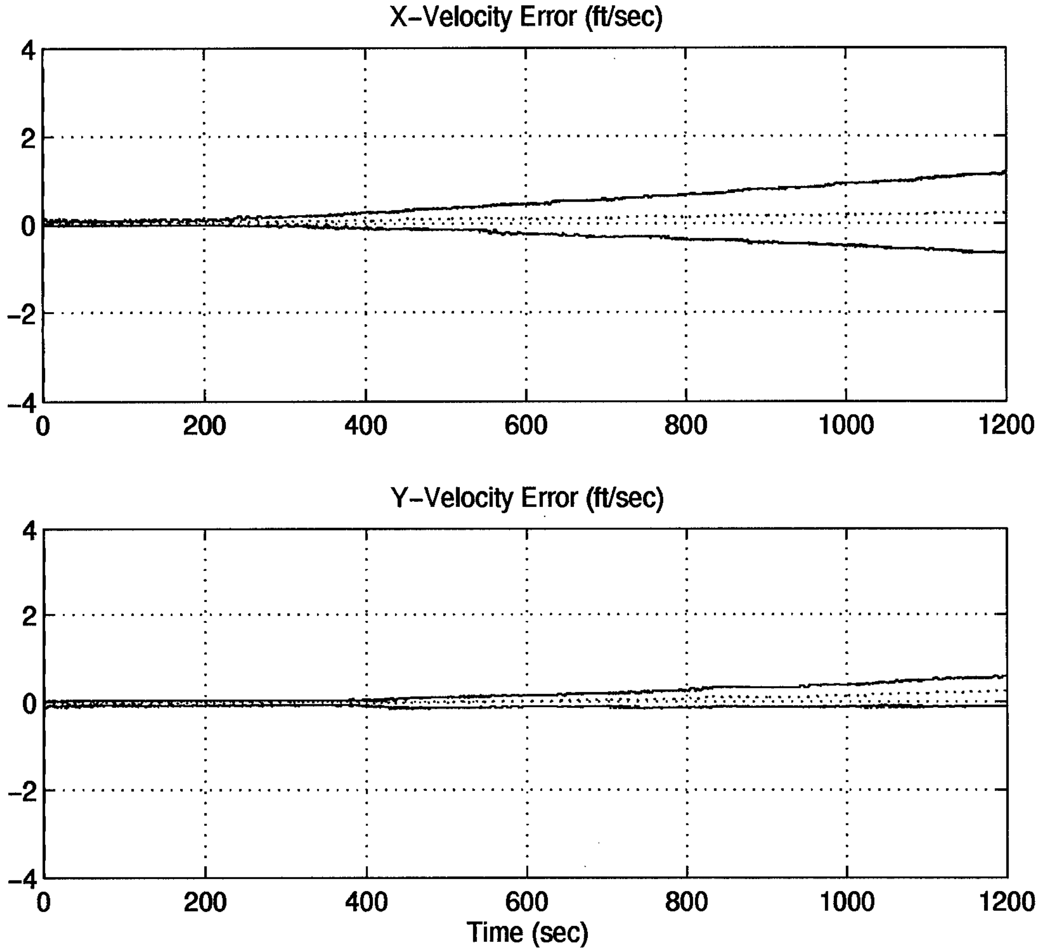

Figure 4.6 Loose Velocity Errors

rally averaged error is 0.1195 ± 0.3704 feet/second. The y-axis velocity temporally averaged error is 0.0411 ± 0.1511 feet/second.

4.3.4 GPS/INS Performance Comparison. The tight and loose GPS/INS integrations are compared for performance in x,y-axis position and velocity in this section. Once again, the integration configurations and filter models were designed to provide an "apples to apples" comparison. To make the comparison easier, the temporal average is taken across the length of the run to provide a scalar value. The results are summarized in Table 4.1.

The tight and loose position error estimate sigma values are very close. The loose velocity error estimates are comparable to the tight, but appear to be slowly growing. A correction for this might be to add velocity measurements to the second filter, using the velocity error estimates from the first filter. These velocity measurements will be cross correlated with the position measurements and should be modelled by the measurement model of the navigation filter. This may keep the loose integration's velocity error estimates bounded and thus be very comparable in performance to the tight integration. This research, with some refinements to the loose integration, supports the idea that although a centralized filter is theoretically optimal, when the filters are implemented in the real world where the theoretical assumptions are violated and the models are not exact, the non-optimal cascaded filter performs just as well.

Table 4.1 Tight vs Loose Time Averaged Errors

	Position (ft)		Velocity (ft/sec)	
Integration	Latitude	Longitude	X-axis	Y-axis
Tight	-87.53 ± 31.13	154.13 ± 35.05	-0.04233 ± 1.162	-0.1392 ± 1.738
Loose	-85.48 ± 32.69	153.49 ± 37.76	0.1195 ± 0.3704	0.0411 ± 0.1511

4.4 Other Comparison Issues

The comparison between tight and loose integration goes beyond position and velocity performance. Other issues for comparison would be the behavior of each integration with corrupt data, and also the computational load for each. This section presents one case of corrupt data and the computational load difference between the two integrations.

4.4.1 Integration With Corrupt GPS Data . All the data used as measurements in the previous section was good, non-corrupt data. The GPS data was screened for corruption or errors before being used in the integration filters. This type of massaging of raw GPS data would be feasible in a real-time situation. However, the amount of bad data checking that could be done is limited, due to the fact that it must be done within the filter update rate (one second for these integrations). Furthermore, one cannot account for all types of erroneous data, thus some bad data may slip by the data checking routines and be used in the filter. Therefore, a comparison of how the loose and tight integrations might behave under bad data would be of interest. This section presents a case where corrupt GPS data is used in each the tight and loose integrations.

The corrupt data used in this section is from actual GPS data taken from the XR5-M6 GPS receiver (see Figure 4.7). These plots were made with a least squares algorithm using only GPS data and only one run. A single run is used to show the failure. The GDOP was around 3.7, which is considered good. The spike in the data reached above 8×10^6 feet in latitude and approached 4×10^6 feet in longitude. The tuning values used in the tight and loose integration filters are the same as for the case when good data is used, as shown in the previous section.

The results of the tight integration are shown in Figures 4.8 and 4.9. Both the position and velocity reflect the corrupt data by also showing a spike in the error estimates. Recall that the Kalman filters are estimating the INS errors, so a spike in the error estimate will drastically affect the overall navigation solution. The latitude error estimate spikes up to 2.5×10^6 feet and, after about 350 seconds of oscillation, recovers to within 500 feet at the 590 second point. The longitude error estimate spikes up above 1.5×10^6 feet and recovers to within 500 feet at the 500 second point,

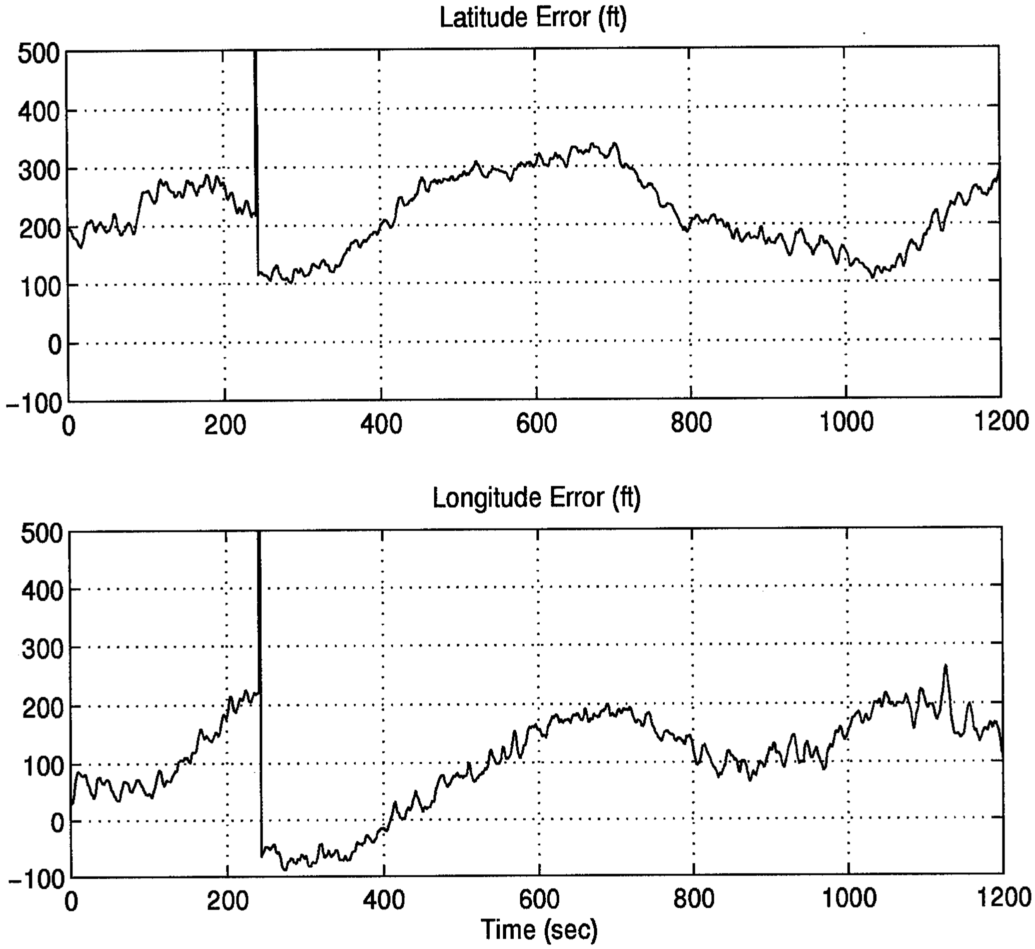

Figure 4.7 XR5-M6 GPS Position Error

after about 260 seconds of oscillation. Both position error estimates return to within 200 feet by the end of the 1200 second run.

The x-axis velocity error estimate spikes up to 2×10^5 feet/second and stabilizes to within 500 feet/second at the 500 second point. The y-axis velocity error estimate spikes up just short of 4×10^5 feet/second and stabilizes to within 500 feet/second at the 490 second point. Both velocity error estimates return to within 50 feet/second by the end of the 1200 second run.

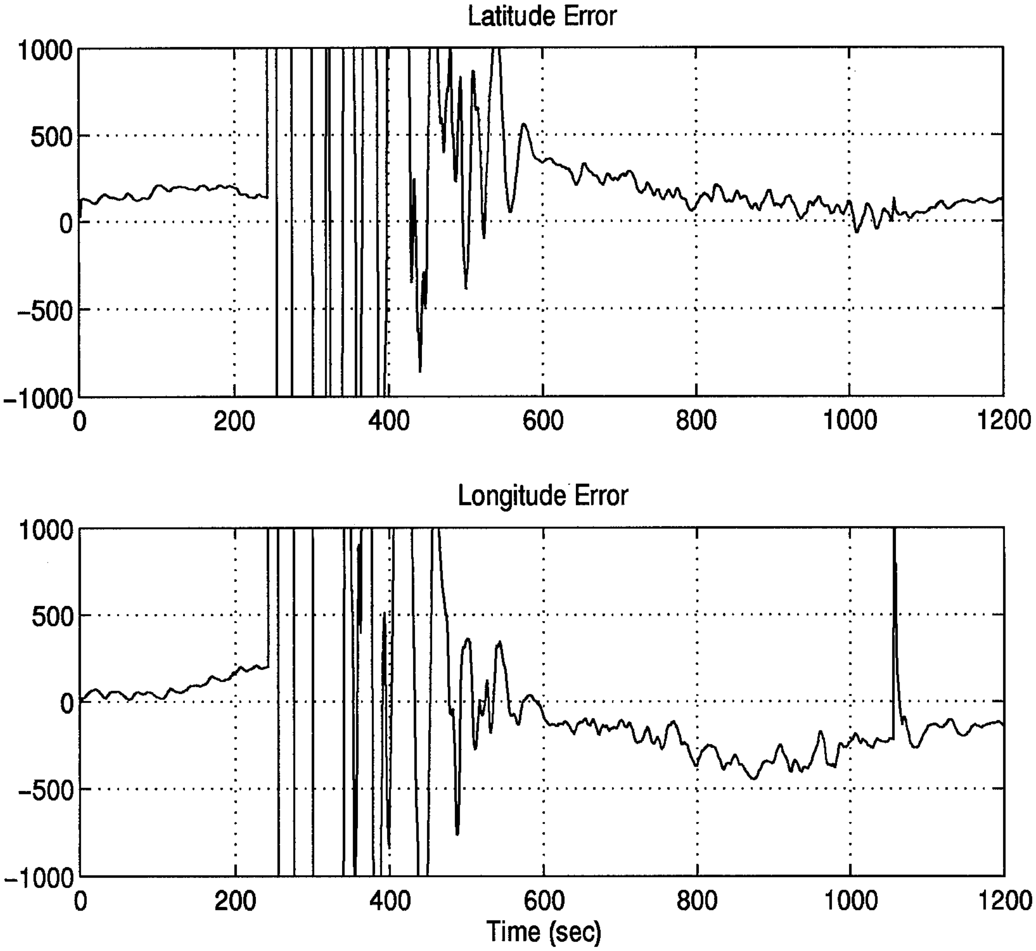

Figure 4.8 Tight Position Errors

The results of the loose integration are shown in Figures 4.10 and 4.11. Once again the corrupt data causes a spike in the loose integration error estimates. The latitude error estimate spikes up to 13 × 10^4 feet, and the longitude error estimate spikes up to 18 × 10^4 feet. These spikes are over an order of magnitude less than that for the tight case. Both the latitude and longitude recover to within 500 feet at about the 300 second point, without any oscillations. By the end of the run both are within 200 feet of error.

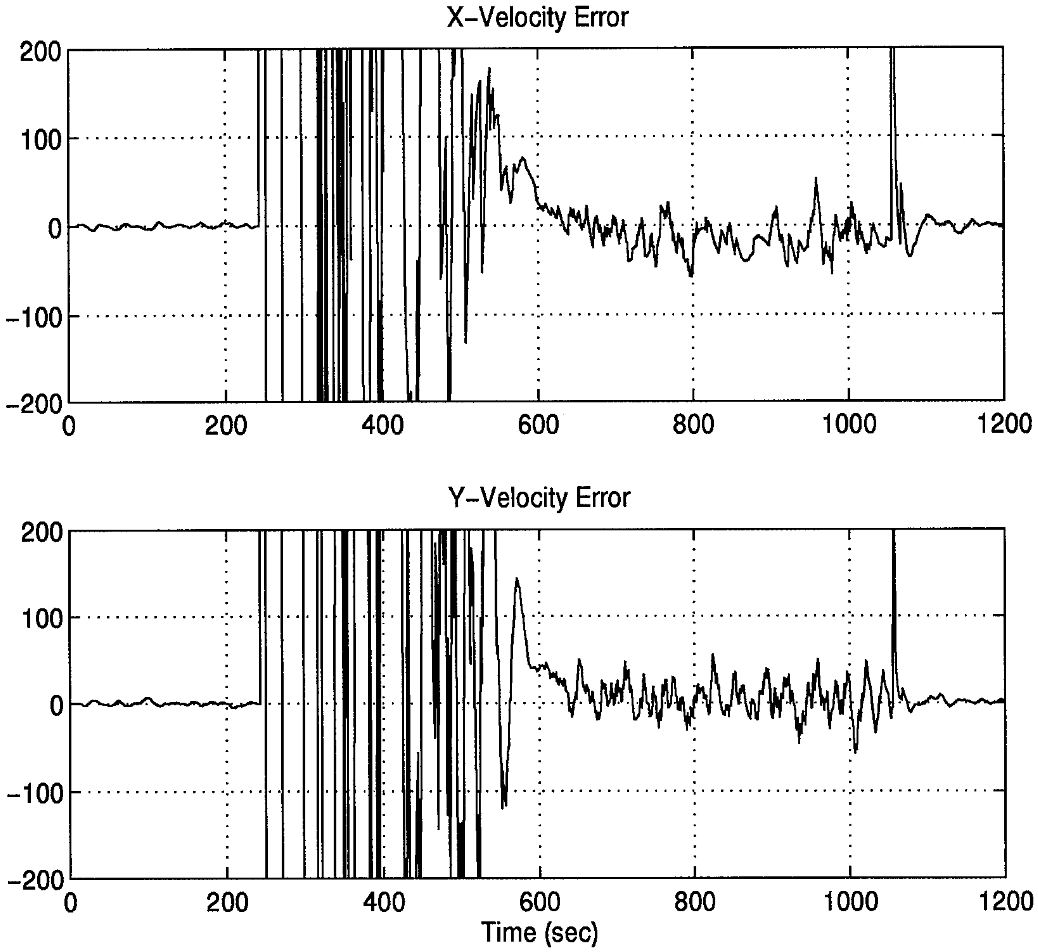

Figure 4.9 Tight Velocity Errors

The x-axis velocity error estimate spikes up only to 320 feet/second and the y-axis velocity error estimates spikes up to 475 feet/second. Both velocities stabilize to within 10 feet/second by the 275 second mark.

It appears that for this corrupt data case, the loose integration behaves in a less dramatic manner than does the tight. The loose error estimates in position do not spike up as high as the tight. Although the actual magnitude of the spike is not so critical, the loose recovers much more quickly and with fewer oscillations. Likewise, the loose error estimates in the x,y-axis velocities do not spike nearly as

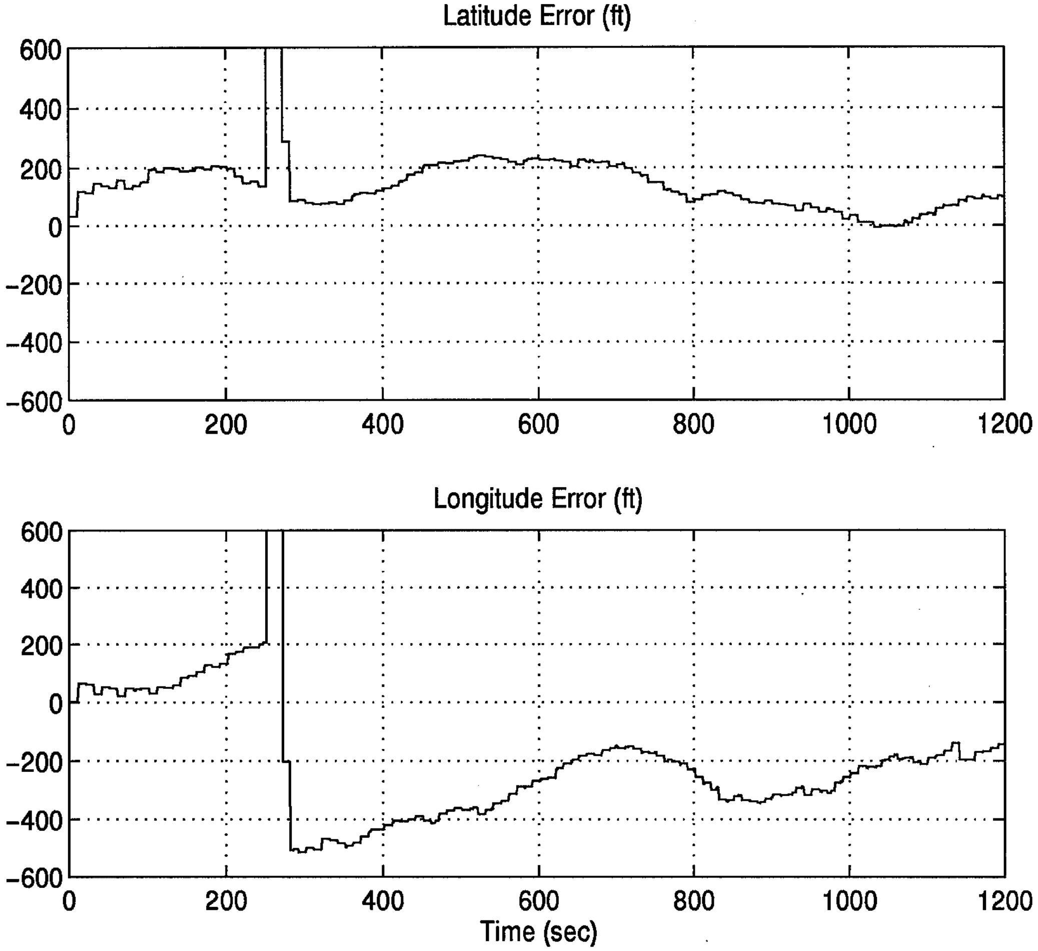

Figure 4.10 Loose Position Errors

highly as in the tight case and also recovers extremely quickly. This implies the loose integration handles this corrupt data case much better than the tight integration. In all, the loose recovered from the corrupt data within 50 seconds for the position and within 30 seconds for the velocities.

This section demonstrated the response of the tight and loose integration under bad GPS data. It was found that the loose integration behaved gracefully and with a quicker recovery time than the tight integration. If adaptive or self-tuning estimation algorithms had been used, a comparison of the tight and loose filter residuals should

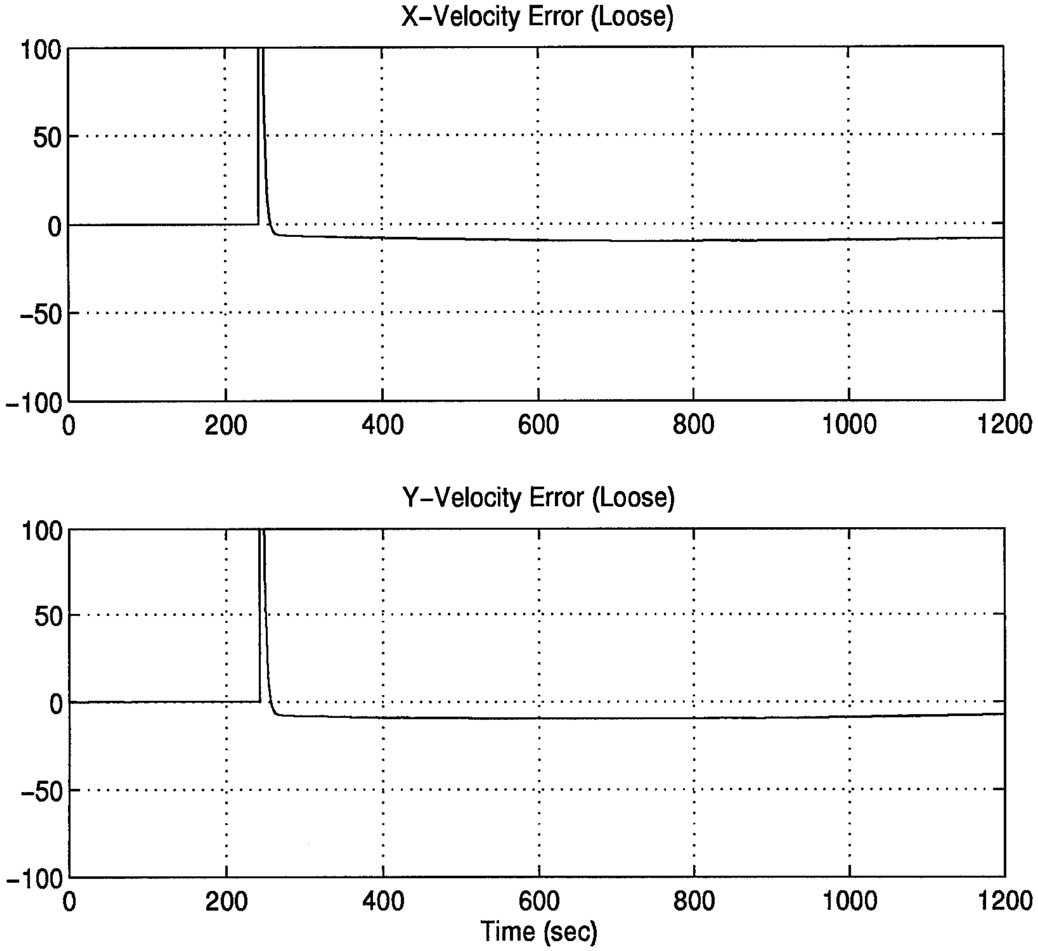

Figure 4.11 Loose Velocity Errors

be examined for the ease of fault detection. The residual provides the information to detect such an error as the bad GPS data in the above case. Once an error is detected the self-tuning estimation algorithms can readjust the noise strengths in the filter's internal model so that the filter is continually "tuned" as well as possible [17]. Further information on adaptive estimation can be found in books on stochastic models and estimation, such as Maybeck (Reference [17]).

4.4.2 Computational Loading. The computational loading is the number of operations required for one time propagation and one measurement update;

see Section 2.5 for the equations used to calculate the number of operations. The integrations used in this thesis are of the conventional Kalman filter type. If these integrations were to be implemented in actual avionics the U-D factorization method would be used for enhanced numerical characteristics; thus the computational loading is computed for both (see Table 4.2). Comparing the computational load of the 27-state filter with the 25-state filter shows how computationally arduous the Kalman filter becomes as the number of states increase.

Table 4.2 Computational Load

		Conventional Kalman Filter			U-D Factorization		
		Adds	Multiplies	Divides	Adds	Multiplies	Divides
Tight	27-State	43,983	46,413	4	54,486	57,564	134
Loose	12-State	4,428	4,968	4	5,406	5,964	59
	25-State	34,225	36,275	3	42,625	45,275	99

The tight integration uses a single 27-state Kalman filter performing a propagate and a measurement update for every navigation solution output it provides. The total number of operations are $90,400$ for the conventional Kalman filter and $112,184$ for the U-D factorization method. These values are computed with four measurements, as if exactly four satellites are received.

The total numbers of operations for the loose integration (both filters) are $79,903$ for the conventional Kalman filter and $99,428$ using the U-D factorization method. These totals are for both filters doing a propagate and an update. However, the loose integration uses a 12-state Kalman filter performing a propagate and a measurement update plus a 25-state Kalman filter performing a propagate cycle for every navigation solution output. The 25-state filter only performs a measurement update at one tenth the rate of the navigation solution output. Therefore, the total number of operations for the loose integration are high estimates. Again, the 12-state filter assumes four measurements, and the 25-state filter uses 3 measurements.

4.5 Chapter Summary

This chapter presented the results of the tight and loose GPS/INS integration study. It discussed the preprocessing of the raw data, and showed plots of the INS and GPS alone. The tight integration results were presented next, followed by the loose integration results. The performance comparison showed very little difference in position accuracy, supporting the adequacy of the loose integration. This showed that in the real world the theoretically optimal tight integration loses its optimality and is comparable to the non-optimal loose integration. Corrupt data was presented to the integrations and the loose dramatically outperformed the tight in its smooth, quick recovery. Finally, the computational loading for each integration was provided and showed the computational burden the tight has over the loose.

V. Conclusions and Recommendations

This chapter presents the conclusions drawn from the results presented in Chapter 4 and recommendations for future AFIT research. The conclusions generalize the results of the loose and tight GPS/INS integration study. The recommendations points out potential problem areas identified in the research, provides suggestions to remedy these shortcomings, and recommends future topics to be included in future AFIT theses.

5.1 Conclusions

The research presented in Chapter 4 provided several interesting conclusions between cascaded and centralized GPS/INS integrations. The integrations and Kalman filters used in the research were designed to provide a fair comparison. Real data from actual hardware was used in the integration to maintain a comparison that is as realistic as possible. All errors in the real world cannot be modelled and thus the tight integration is no longer optimal.

The performance of the tight integration is then compared to the theoretically non-optimal loose integration. The research in Chapter 4 presented plots and scalar values of latitude and longitude position errors, and x,y-axis velocity errors for both integration configurations. The results showed very small differences between the tight and loose integration in position performance, with the tight being slightly more accurate. Although the loose integration's actual errors were comparable to those of the tight, there was a slight divergence. Velocity updates to the second filter of the loose may bound the velocity solution and thus make the loose comparable to the tight. Further comparison issues were then addressed: corrupt data and computation load.

The two integrations were subjected to corrupt GPS data. Both tight and loose responded to the corrupt data with a spike in its position and velocity errors. However, the tight integration went through large oscillations before recovering, whereas the loose integration had no oscillation and recovered rather quickly. A further comparison on error detection via the filter residuals may also provide tradeoffs between the tight and loose integration.

Computer loading for the two integrations were then examined. The loose integration requires the computations for two filters, but these filters are of smaller order than the single filter of the tight integration. The way the calculations are required for the Kalman filter makes the loose integration more desirable with regards to computer loading.

In summary, the tight integration had a very slight advantage over the loose in estimating the position and velocity errors. However, for the corrupt GPS data case used in this research the loose integration provided a smoother and quicker recovery. Finally, as was expected, the loose integration requires significantly less computation than does the tight integration.

5.2 Recommendations

The following section provides this researcher's recommendations for future AFIT research topics and enhancements to GPS/INS integrations.

5.2.1 Preprocessing of GPS Data. Working with real data presented many unexpected GPS/INS integration difficulties. The amount of fluctuations in the GPS data was the biggest surprise. If the idea holds that the smoother the input data to the integration is, the better the outputs will be, then preprocessing of the integration data is desirable. In this research, small amounts of preprocessing of GPS data was carried out. Small items like maintaining the order of the pseudoranges from one measurement to the next, and a rough check for bad pseudoranges, were accomplished. A good criterion for determining a bad pseudorange or corrupt ephemeris data was not established. This is especially difficult in the real-world when no truth data is available for comparison. One way to enhance the GPS/INS integrations might be to investigate the preprocessing of raw data.

5.2.2 Measurement Models. This research used only position as the measurements for all the filters in both the loose and tight integration. Future research should include velocity measurements to the filters. The delta-range data from the GPS can be differenced with the INS velocity with respect to the satellites to provide measurements to the GPS filter of the loose and the only filter of the tight. With regards to the navigation filter of the loose integration, the three velocity error states of the GPS filter can be used as measurements to the navigation filter. The velocity measurements will be cross correlated with the position measurements and should be modelled as so by the measurement model of the navigation filter. This velocity updates may improve on this research's problem with the second filter velocity divergence. In fact, all the INS states of the GPS filter can be used as measurements to the navigation filter with proper cross correlation terms in the measurement model. However, it should be remembered that an increase in the number of measurements increases the computational loading. A study on the navigation solution performance with this increased number of measurements would be of interest.

5.2.3 GPS/INS Integration with Feedback. With the recent AFIT acquisition of a Rockwell inertial measurement unit (IMU), a feedback GPS/INS integration configuration can be done. The inertial measurement unit outputs raw accelerometer and gyroscopic measurements. Since these measurements are not processed inside the box as an INS does, a feedback configuration can be achieved without the problems of having a real-time system. The current problem with real-time systems at AFIT is the lack of speed, since the software programs have to keep up in a real-time scenario. The Rockwell IMU will allow more control of the "INS" in a post-processing environment. First, a computer program must be made to use the raw accelerometer and gyroscopic measurements to output an INS navigation solution. Estimates from a GPS/INS integration can be used as added inputs to the computer program and thus, an INS using feedback in a post-processing environment is created. Research of this type using real data in a GPS/INS feedback configuration will be a first for AFIT.

5.2.4 Mobile GPS/INS Integration. The Rockwell inertial measurement unit is a compact battery operated unit. This finally gives AFIT the opportunity to take its INS/GPS integration on the road. The Rockwell IMU along with its battery and a laptop computer for data collection can be easily secured onto a pallet and taken on the road in a small vehicle. The XR5-M6 Navstar GPS receiver can be fitted with a second laptop computer and placed in the same vehicle with the IMU. An accurate method for timing of the collection of the IMU and GPS data would need to be designed.

5.3 Summary

Hopefully, the results of this research and the recommendations provided will help the AFIT Navigation, Guidance, and Control section in attaining its goal of developing a mobile integrated system. This chapter has presented the conclusions and recommendations from this research.

Appendix A. Simple, Low-Order Integration

This appendix provides a complete breakdown of the simple, low-order integration example in Chapter 3. The example was simulated in Matlab's Simulink software. The integration configurations were done in block diagram format using Simulink's preprogramed blocks with the exception of the Kalman filter blocks. The Kalman filters were programmed as Matlab functions using m-files.

Figures A.1 and A.2 show the loose and tight GPS/INS integration configurations. The INS, Orbit, and GPS blocks are common to both configurations. The INS block is shown in Figure A.3, and consists of two integrators driven by colored noise and an acceleration input. The colored noise is modelled as a first-order Markov process driven by zero-mean white Gaussian noise. The Orbit block is shown in Figure A.4. This block simulates the ephemeris data of a GPS satellites by providing the location of the satellite in this simulation (it is at a constant location), and thus computing the INS range to satellite. The GPS block is shown in Figure A.5 and simulates the pseudorange outputs of a GPS receiver. This block receives the true location of the vehicle throughout the run of the simulation and adds white Gaussian noise to simulate a pseudorange. The Profile block, shown in Figure A.6, reads in the acceleration input and provides the true position of the vehicle.

The Kalman filter blocks are m-files programmed as Matlab functions. The algorithm used is the standard (versus extended) discrete-time Kalman filter algorithm. The tight filter and the first filter in the loose has an update rate of one second. The second fitler in the loose updates every ten seconds.

The Simulink integration block diagrams are run from a shell, script file. This shell provides the Kalman filter initial conditions, vehicle acceleration inputs, random number seeds, number of Monte Carlo runs and run length, and the plotting routines. The shell calls the Simulink diagrams using the 'rk45' command, which is the Runge-Kutta fifth order integration function.

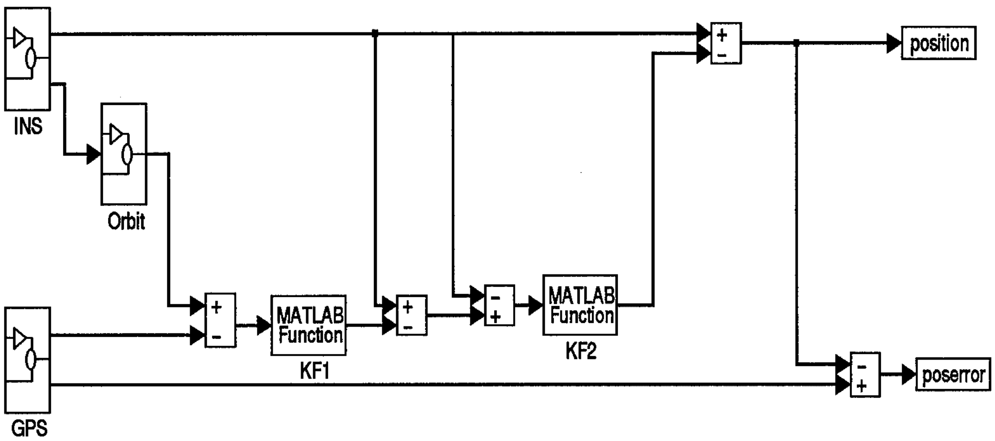

Figure A.1 Loosely-Coupled Integration

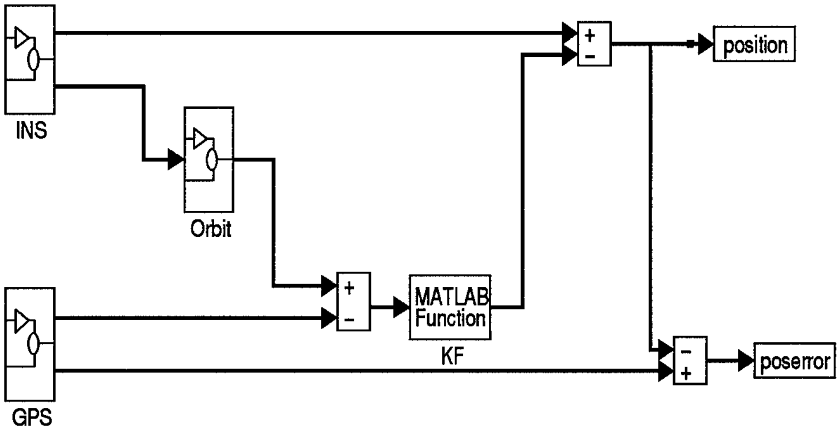

Figure A.2 Tightly-Coupled Integration

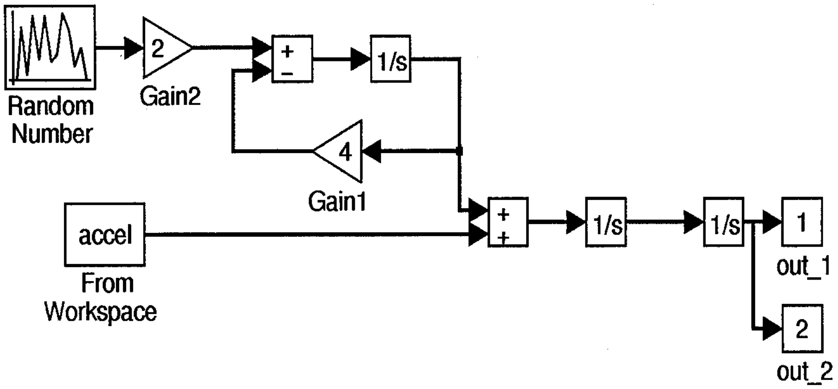

Figure A.3 INS Block

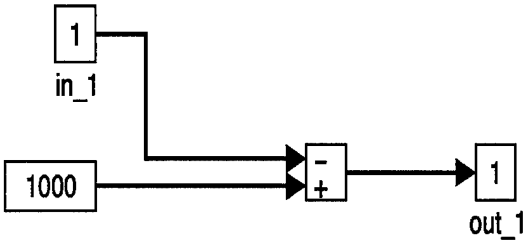

Figure A.4 Orbit Block

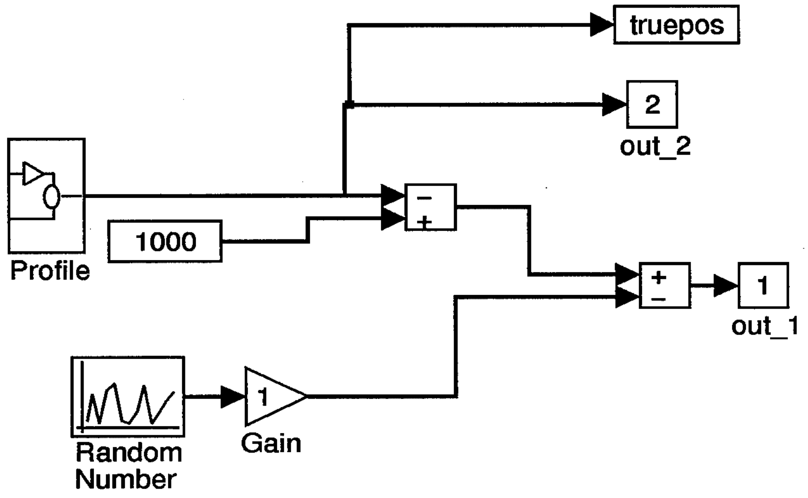

Figure A.5 GPS Block

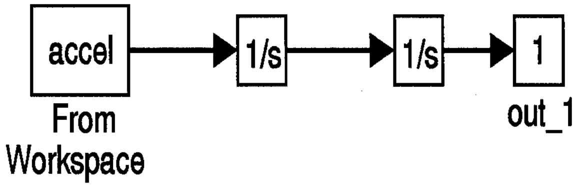

Figure A.6 Profile Block

Appendix B. Filter Tuning Parameters

This appendix provides the dynamics driving noise strength, Q, and measurement variance, R, tuning values used in the tight and loose GPS/INS integrations. The tuning parameters for the 27-state filter of the tight integration is shown in Tables B.1 and B.2. The tuning parameters for the 12-state filter of the loose integration is shown in Tables B.3 and B.4. The tuning parameters for the 25-state filter of the loose integration is shown in Tables B.5 and B.6. The 12-state, and 27-state filter measurement noise in Tables B.4 and B.2 reflect the measurements from four satellites, although the number of measurements may vary depending on the number of satellites received. The number of measurements for the 25-state filter is always three (3 positions).

Table B.1 Dynamic Driving Noise Values for 27-State Filter

Element of Q	Definition	Value
$(1,1)$	X component of vector angle from true to computer frame	1×10^{-13}
$(2,2)$	Y component of vector angle from true to computer frame	1×10^{-13}
$(3,3)$	Z component of vector angle from true to computer frame	1×10^{-13}
$(4,4)$	X component of vector angle from true to platform frame	1×10^{2}
$(5,5)$	Y component of vector angle from true to platform frame	1×10^{2}
$(6,6)$	Z component of vector angle from true to platform frame	1×10^{2}
$(7,7)$	X component of error in computer velocity	1.5×10^{2}
$(8,8)$	Y component of error in computer velocity	1.5×10^{2}
$(9,9)$	Z component of error in computer velocity	1.5×10^{2}
$(10,10)$	Barometer correlated bias noise error	1×10^{-6}
$(11,11)$	GPS user clock bias	1×10^{5}
$(12,12)$	GPS user clock drift	5×10^{5}
$(13,13)$	X-component of gyro drift repeatability	3×10^{-3}
$(14,14)$	Y-component of gyro drift repeatability	3×10^{-3}
$(15,15)$	Z-component of gyro drift repeatability	3×10^{-3}
$(16,16)$	X-gyro misalignments about Y axis	5×10^{-6}
$(17,17)$	Y-gyro misalignments about X axis	5×10^{-6}
$(18,18)$	Z-gyro misalignments about X axis	5×10^{-6}
$(19,19)$	X-gyro misalignments about Z axis	5×10^{-6}
$(20,20)$	Y-gyro misalignments about Z axis	5×10^{-6}
$(21,21)$	Z-gyro misalignments about Y axis	5×10^{-6}
$(22,22)$	X-component of accelerometer bias repeatability	2.5×10^{-3}
$(23,23)$	Y-component of accelerometer bias repeatability	2.5×10^{-3}
$(24,24)$	Z-component of accelerometer bias repeatability	2.5×10^{-3}
$(25,25)$	X-component of accelerometer and velocity quantizer scale factor error	144×10^{-6}
$(26,26)$	Y-component of accelerometer and velocity quantizer scale factor error	144×10^{-6}
$(27,27)$	Z-component of accelerometer and velocity quantizer scale factor error	144×10^{-6}

Table B.2 Sensor Measurement Noise Values for 27-State Filter

Element of R	Definition	Value
$(1,1)$	Satellite 1	100
$(2,2)$	Satellite 2	100
$(3,3)$	Satellite 3	100
$(4,4)$	Satellite 4	100

Table B.3 Dynamic Driving Noise Values for 12-State Filter

Element of Q	Definition	Value
$(1,1)$	X component of vector angle from true to computer frame	5×10^{-13}
$(2,2)$	Y component of vector angle from true to computer frame	5×10^{-13}
$(3,3)$	Z component of vector angle from true to computer frame	5×10^{-13}
$(4,4)$	X component of vector angle from true to platform frame	1×10^{5}
$(5,5)$	Y component of vector angle from true to platform frame	1×10^{5}
$(6,6)$	Z component of vector angle from true to platform frame	1×10^{5}
$(7,7)$	X component of error in computer velocity	1.5×10^{5}
$(8,8)$	Y component of error in computer velocity	1.5×10^{5}
$(9,9)$	Z component of error in computer velocity	1.5×10^{5}
$(10,10)$	Barometer correlated bias noise error	1×10^{-6}
$(11,11)$	GPS user clock bias	1.5×10^{-2}
$(12,12)$	GPS user clock drift	1×10^{2}

Table B.4 Sensor Measurement Noise Values for 12-State Filter

Element of R	Definition	Value
$(1,1)$	Satellite 1	50
$(2,2)$	Satellite 2	50
$(3,3)$	Satellite 3	50
$(4,4)$	Satellite 4	50

Table B.5 Dynamic Driving Noise Values for 25-State Filter

Element of Q	Definition	Value
$(1,1)$	X component of vector angle from true to computer frame	1×10^{2}
$(2,2)$	Y component of vector angle from true to computer frame	1×10^{2}
$(3,3)$	Z component of vector angle from true to computer frame	1×10^{2}
$(4,4)$	X component of vector angle from true to platform frame	1×10^{8}
$(5,5)$	Y component of vector angle from true to platform frame	1×10^{8}
$(6,6)$	Z component of vector angle from true to platform frame	1×10^{8}
$(7,7)$	X component of error in computer velocity	1.5×10^{10}
$(8,8)$	Y component of error in computer velocity	1.5×10^{10}
$(9,9)$	Z component of error in computer velocity	1.5×10^{10}
$(10,10)$	Barometer correlated bias noise error	1×10^{3}
$(11,11)$	X-component of gyro drift repeatability	3×10^{-3}
$(12,12)$	Y-component of gyro drift repeatability	3×10^{-3}
$(13,13)$	Z-component of gyro drift repeatability	3×10^{-3}
$(14,14)$	X-gyro misalignments about Y axis	5×10^{-6}
$(15,15)$	Y-gyro misalignments about X axis	5×10^{-6}
$(16,16)$	Z-gyro misalignments about X axis	5×10^{-6}
$(17,17)$	X-gyro misalignments about Z axis	5×10^{-6}
$(18,18)$	Y-gyro misalignments about Z axis	5×10^{-6}
$(19,19)$	Z-gyro misalignments about Y axis	5×10^{-6}
$(20,20)$	X-component of accelerometer bias repeatability	2.5×10^{-3}
$(21,21)$	Y-component of accelerometer bias repeatability	2.5×10^{-3}
$(22,22)$	Z-component of accelerometer bias repeatability	2.5×10^{-3}
$(23,23)$	X-component of accelerometer and velocity quantizer scale factor error	144×10^{-6}
$(24,24)$	Y-component of accelerometer and velocity quantizer scale factor error	144×10^{-6}
$(25,25)$	Z-component of accelerometer and velocity quantizer scale factor error	144×10^{-6}

Table B.6 Sensor Measurement Noise Values for 25-State Filter

Element of R	Definition	Value
$(1,1)$	X component of vector angle from true to computer frame	100
$(2,2)$	Y component of vector angle from true to computer frame	100
$(3,3)$	Z component of vector angle from true to computer frame	100

Bibliography

1. Britting, Kenneth R. *Inertial Navigation Systems Analysis.* New York: Wiley-Interscience, 1971.

2. Canadian GPS Associates. *Guide to GPS Positioning.* ISBN: 0-920-114-73-3, Fredericton, New Brunswick: University of New Brunswick Graphic Services, May 1987.

3. Coffee, John R. "GPS-INS Integration Requirements for Robust NOE Helicopter Navigation," *American Helicopter Society, 47th Annual Forum Proceedings.* 1075-1082. Alexandria, VA, AHS, 1991.

4. Dayton, Ronald B. and Nielson, John T. "A Flight Test Comparison of Two GPS/INS Integration Approaches," *Institute of Navigation Satellite Division, International Technical Meeting.* 269-273. Washington D.C., ION, 1989.

5. Department of the Air Force. *Specification for USAF F-15 Inertial Navigation Set.* FNU 85-1, Revision A. Aeronautical System Division, Wright-Patterson AFB, OH, 1 March 1986.

6. Evans, Curtis D. *The Design and Analysis of Integrated Navigation Systems Using Real INS and GPS Data.* MS Degree Thesis, AFIT/GE/ENG/94D-08. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December 1994 (AD-A289380).

7. Gao, Y. and others. "Comparison and Analysis of Centralized, Decentralized, and Federated Filters," *Navigation: Journal of The Institute of Navigation* Vol. 40, No. 1. 69-86. Alexandria, VA, ION, Spring 1993.

8. Bray, Robert, A. *An Integrated GPS/INS/BARO and Radar Altimeter System for Aircraft Precision Approach Landings.* MS Degree Thesis, AFIT/GE/ENG/94D-13. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December 1994 (AD-A289280).

9. Griffin, Gordon and others. *The Creation and Validation of a Matlab Tool as a Possible Alternative to MSOFE.* EENG-735 Class Project. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, June 1994.

10. Hirning, James L. *Optimal Kalman Filter Integration of a Global Positioning System Receiver and an LN-94 Inertial Navigation System.* MS Degree Thesis, AFIT/GE/ENG/90S-02. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, September 1990 (AD-A227222).

11. Knudsen, L. *Performance Accuracy (Truth Model/Error Budget) Analysis for the LN-93 Inertial Navigation System Inertial Navigation Unit.* Technical Report, Litton Guidance and Control Systems, January 1985. DID No. DI-S-21433 B/T:CDRL No. 1002.

12. General Dynamics, Fort Worth Division. *Computer Program Product Specification for the F-16 Fire Control Computer Operational Flight Program.* Volumes I and II, Block 50 Production Tape 2, Code Ident: 81755, Fort Worth, TX, February 1992.

13. Lewantowicz, Zdzislaw H. "Architectures and GPS/INS Integration: Impact on Mission Accomplishment," *IEEE Position, Location, and Navigation Symposium.* 284-289. New York, IEEE Press, 1992.

14. Lewantowicz, Zdzislaw H. and Paschall, Randall N. "Deep Integration of GPS, INS, SAR, and Other Sensor Information," Final draft. Avionics Directorate, Wright Laboratory, Wright-Patterson AFB, OH, 1994.

15. *MATLAB.* Version 4.1. Computer software. The Math Works, Inc. 24 Prime Park Way, Natick, MA 01760, June 1993.

16. Maybeck, Peter S. *Stochastic Models, Estimation, and Control.* Volume 1. Reprint. Alexandria, VA: Navtech Book & Software Store, 1994.

17. Maybeck, Peter S. *Stochastic Models, Estimation, and Control.* Volume 2. Reprint. Alexandria, VA: Navtech Book & Software Store, 1994.

18. Maybeck, Peter S. *Stochastic Models, Estimation, and Control.* Volume 3. New York: Academic Press, Inc., 1982.

19. Milliken, R. J. and Zoller, C. J. "Principle of Operation of NAVSTAR and System Characteristics," *Global Positioning System*, The Institute of Navigation, Volume I. Alexandria, VA 22314, 1980.

20. Mosle, Willian B. *Detection, Isolation, and Recovery of Failures in an Integrated Navigation System.* MS Degree Thesis, AFIT/GE/ENG/93D-28. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December 1993 (AD-A274056).

21. *Navstar Ltd Power Users Data Monitor Toolkit (XR5 Series).* 212-PL-G1 Issue 1.0. Navstar Systems Ltd. Daventry, Northants NN11 5PJ, England, September 1993.

22. Nuibe, Warren H. and others. *Comparison of a Loose and Tight Integration of a Navigation System.* EENG-735 Class Project. Air Force Institute of Technology, Wright-Patterson AFB OH, 2 June 1995.

23. Schwarz, K. P. and others. "Aided Versus Embedded A Comparison of Two Approaches to GPS/INS Integration," *IEEE Position, Location, and Navigation Symposium.* 314-322. New York, IEEE Press, 1994.

24. *Simulink Dynamic System Simulation Software.* Computer software. The Math Works, Inc. 24 Prime Park Way, Natick, MA 01760, April 1993.

25. Siouris George, M. *Aerospace Avionics Systems A Modern Synthesis.* San Diego: Academic Press, Inc., 1993.

26. *SCE4 (MS/DOS) Application Software for the DTI-1120.* Digital Technology, Inc. Dayton, OH, 1988 (RM-1120-20501).

27. Tazartes, D. A. and J. G. Mark "Integration of GPS Receivers into Existing Inertial Navigation Systems," *Navigation: Journal of the Institute of Navigation.* Vol 34, No. 2, 815 Fifteenth Street, Suite 832, Washington, D.C. 20005, Summer 1988.

28. Van Dierendonck, A.J. and others. "The GPS Navigation Message," *Global Positioning System*, The Institute of Navigation, Volume I. Alexandria, VA 22314, 1980.

29. Vasquez, Juan R. *Detection of Spoofing, Jamming, or Failure of a Global Positioning System (GPS).* MS Degree Thesis, AFIT/GE/ENG/92D-37. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December 1992 (AD-A259023).

30. Wei M. and Schwarz, K. P. "Testing a Decentralized Filter for GPS/INS Integration," *IEEE Position, Location, and Navigation Symposium.* 429-435. New York, IEEE Press, 1990.

Vita

Captain Warren H. Nuibe [redacted] He graduated from Torrance High School in Torrance, California in 1986. He attended the University of California at San Diego, graduating Cum Laude with a Bachelors of Science Degree in Electrical Engineering in December 1990. He received his commission through the Reserve Officer Training Corps and reported for active duty in September 1991. His first assignment was as a Combat Systems Engineer with the Tactical Air Warfare Center at Eglin AFB, Florida. He came to the Air Force Institute of Technology in May 1994, where he was awarded the degree of Master of Science in Electrical Engineering, in December 1995. Captain Nuibe is currently assigned to the Avionics Directorate, Wright-Patterson AFB, OH.

Permanent address: [redacted]

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 0704-0188*

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE December 1995	3. REPORT TYPE AND DATES COVERED Master's Thesis

4. TITLE AND SUBTITLE A COMPARISON OF LOOSE AND TIGHT GPS/INS INTEGRATION USING REAL INS AND GPS DATA	5. FUNDING NUMBERS
6. AUTHOR(S) Warren H. Nuibe Captain, USAF	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Air Force Institute of Technology WPAFB OH 45433-6583	8. PERFORMING ORGANIZATION REPORT NUMBER AFIT/GE/ENG/95D-19
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) WL/AAAI Avionics Directorate Wright-Patterson AFB, OH 45433	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)*

An extended Kalman filter (EKF) is used to combine the information obtained from a Global Positioning System (GPS) receiver and an Inertial Navigation System (INS) to provide a navigation solution. This research compares the results of a tightly-coupled GPS/INS integrated system with a loosely-coupled integrated system, using real world data. A fair comparison is accomplished by using the same sets of data, and keeping the integration structures as close as possible. Both integrations are feedforward and have the same error states in the navigation Kalman filters. Differences between the two, such as navigation solutions and tuning values, are shown in the research.

14. SUBJECT TERMS Global Positioning System, Inertial Navigation System GPS/INS Integration, Kalman Filter	15. NUMBER OF PAGES 88
	16. PRICE CODE

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

GENERAL INSTRUCTIONS FOR COMPLETING SF 298

The Report Documentation Page (RDP) is used in announcing and cataloging reports. It is important that this information be consistent with the rest of the report, particularly the cover and title page. Instructions for filling in each block of the form follow. It is important to ***stay within the lines*** to meet ***optical scanning requirements***.

Block 1. Agency Use Only *(Leave blank)*.

Block 2. Report Date. Full publication date including day, month, and year, if available (e.g. 1 Jan 88). Must cite at least the year.

Block 3. Type of Report and Dates Covered. State whether report is interim, final, etc. If applicable, enter inclusive report dates (e.g. 10 Jun 87 - 30 Jun 88).

Block 4. Title and Subtitle. A title is taken from the part of the report that provides the most meaningful and complete information. When a report is prepared in more than one volume, repeat the primary title, add volume number, and include subtitle for the specific volume. On classified documents enter the title classification in parentheses.

Block 5. Funding Numbers. To include contract and grant numbers; may include program element number(s), project number(s), task number(s), and work unit number(s). Use the following labels:

C	-	Contract	**PR**	-	Project
G	-	Grant	**TA**	-	Task
PE	-	Program Element	**WU**	-	Work Unit Accession No.

Block 6. Author(s). Name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the report. If editor or compiler, this should follow the name(s).

Block 7. Performing Organization Name(s) and Address(es). Self-explanatory.

Block 8. Performing Organization Report Number. Enter the unique alphanumeric report number(s) assigned by the organization performing the report.

Block 9. Sponsoring/Monitoring Agency Name(s) and Address(es). Self-explanatory.

Block 10. Sponsoring/Monitoring Agency Report Number. *(If known)*

Block 11. Supplementary Notes. Enter information not included elsewhere such as: Prepared in cooperation with...; Trans. of...; To be published in.... When a report is revised, include a statement whether the new report supersedes or supplements the older report.

Block 12a. Distribution/Availability Statement. Denotes public availability or limitations. Cite any availability to the public. Enter additional limitations or special markings in all capitals (e.g. NOFORN, REL, ITAR).

DOD - See DoDD 5230.24, "Distribution Statements on Technical Documents."
DOE - See authorities.
NASA - See Handbook NHB 2200.2.
NTIS - Leave blank.

Block 12b. Distribution Code.

DOD - Leave blank.
DOE - Enter DOE distribution categories from the Standard Distribution for Unclassified Scientific and Technical Reports.
NASA - Leave blank.
NTIS - Leave blank.

Block 13. Abstract. Include a brief *(Maximum 200 words)* factual summary of the most significant information contained in the report.

Block 14. Subject Terms. Keywords or phrases identifying major subjects in the report.

Block 15. Number of Pages. Enter the total number of pages.

Block 16. Price Code. Enter appropriate price code *(NTIS only)*.

Blocks 17. - 19. Security Classifications. Self-explanatory. Enter U.S. Security Classification in accordance with U.S. Security Regulations (i.e., UNCLASSIFIED). If form contains classified information, stamp classification on the top and bottom of the page.

Block 20. Limitation of Abstract. This block must be completed to assign a limitation to the abstract. Enter either UL (unlimited) or SAR (same as report). An entry in this block is necessary if the abstract is to be limited. If blank, the abstract is assumed to be unlimited.

★U.S.GPO:1993-0-336-043