IMPROVING INS/GPS INTGRATION FOR MOBILE ROBOTICS APPLICATIONS

A THESIS IN MECHATRONICS

Presented to the faculty of the American University of Sharjah
Collage of Engineering
in partial fulfillment of
the requirements for the degree of

MASTER OF SCIENCE

by

TARIQ SALMAN ABUHASHIM
B.S. 2003

Sharjah, UAE
April 2008

IMPROVING INS/GPS INTGRATION FOR MOBILE ROBOTICS APPLICATIONS

Tariq Salman AbuHashim, Candidate for Master of Science in Mechatronics Engineering

American University of Sharjah, 2008

ABSTRACT

As unmanned systems become more and more important, reliability and integrity issues become definite, specially when being implemented with low-cost (or sometimes are referred to as commercial-of-the shelf or COTS) sensors while being designed to operate in remote, hazardous and harsh environments. As a result, fault (and failure) detection and identification (FDI) is a must, and it is a crucial requirement in designing unmanned vehicles. In this thesis, integrity is defined as the ability of the system to provide reliable navigation information, to monitor the health of the aids, to detect abnormalities in their behavior, and to survive once a failure in one of its components (whether they are sensors, actuators, mathematical models, and computations) occurs. On the other hand, reliability is component dependent. A navigation system is reliable as its most unreliable component. Therefore, integrity implies reliability while reliability not necessarily implies integrity.

This thesis, mainly, discusses the issue of implementing a low-cost inertial navigation system, aided with satellite navigation system. In doing so, a fault detection and identification scheme must be involved and the performance of all the system components must be verified. The FDI system should take into account types of failures commonly occur, guarantee that all faults will be detected, assist design specifications and respond as fast as possible to faults. On the other hand, it should take into account the complexity of the implementation and its robustness in the presence of mismodelling. Innovation-based techniques, in particular the χ^2 SCT, offer tradeoffs between complexity and performance and detect a large set of failures. However, they are sensitive to filter tuning and have no fault identification ability. On the other hand, the model-based approaches, in particular the multiple model adaptive estimation (MMAE), have an outstanding decision making ability and are insensitive to filter tuning. However, they require a priori knowledge on the system and failure model and are computationally expensive. The integration of both techniques can enhance the FDI performance of both systems. In this thesis a sequential FDI algorithm is proposed. This algorithm employs an innovation-based technique for fault detection and a model-based technique for identification.

The performance of the χ^2-MMAE sequential algorithm is simulated and tested on actual IMU and GPS data. Results showed that the sequential algorithm has a comparable identification ability as the MMAE algorithm with a substantial reduction in computational requirements, since the filters bank was only allowed to operate on segments of time where faults were detected. On the other hand, unlike the MMAE algorithm where the performance of the filter was affected during no-fault conditions, the sequential scheme guaranteed the consistency of the estimator in all of its modes of operation and didn't affect its performance during normal no-fault modes of operation.

Contents

List of Figures

List of Tables

Nomenclature

Notation

$\mathbf{C}_A^B$	Direction cosine matrix transforming quantities from A frame to B frame
ω_{AB}^B	Angular rate of B frame relative to A frame expressed with components in B frame
$\mathbf{b}^A$	Vector $\mathbf{b}$ with components in A frame
$\mathbf{\Omega}^A = \left[\omega^A \times\right]$	Skew symmetric matrix with components of ω in A frame

Axis System (Reference Frames), Angles and Transformations

i	Inertial reference frame
e	Earth-fixed reference frame
n	Navigation reference frame
b	Body reference frame
m	Mechanisation frame
$O_{x_i y_i z_i}$	Inertial axis system
$O_{x_e y_e z_e}$	Earth-fixed axis system
$O_{x_n y_n z_n}$	Navigation axis system
$O_{x_b y_b z_b}$	Body axis system
h	Altitude
φ	Latitude
λ	Longitude
ϕ	Roll angle
θ	Pitch angle
ψ	Yaw angle

Earth Quantities (WGS-84)

g_n	Normal gravitational acceleration ($\varphi = 45^o$)
a	Equatorial radius of the Earth (semimajor axis) = 6378137.000 m
b	polar radius of the Earth (semiminor axis) = 6356752.3142 m
Ω	Earth turn rate with respect to i frame = 7.292116×10^{-5} rad/s
$\boldsymbol{g}_l$	Local gravity column matrix
f	Flattening (ellipticity) = 1/298.257223563 (0.00335281066474)

e	Major eccentricity of Earth = 0.0818191908426
μ	Earth's gravitational constant = $3986005 \times 10^8 m^3/s^2$
M	Mass of Earth (including the atmosphere) = 5.9733328×10^{24} Kg

Dynamic Quantities

V_n	Normal gravitational acceleration ($\varphi = 45^o$)
$\boldsymbol{V}_e^n$	Kinematic velocity expressed in n frame
ω_{nb}^b	Angular rate of b frame relative to n frame expressed in b frame
ω_{en}^n	Angular rate of n frame relative to e frame expressed in n frame
ω_{ib}^b	Angular rate of b frame relative to n frame expressed in b frame
$\boldsymbol{f}^e$	Specific force in b frame
$\boldsymbol{f}^b$	Specific force in n frame
v_N, v_E, v_D	The north, east and down components of $\boldsymbol{V}_e^n$
f_N, f_E, f_D	The north, east and down components of $\boldsymbol{f}_e^n$

Subscripts

j, k, l	Indexes for high speed computer cycle (j-cycle), moderate computer speed cycle (k-cycle) and low computer speed cycle (l-cycle) respectively
N, E, D	North, East and Down components of n frame vector

Symbols

$(k\|k-1)$	Used to denote a quantity at time k immediately before the measurement update
$(k\|k)$	Used to denote a quantity at time k immediately after the measurement update
$z(k-1)$	A growing length measurement history vector consisting of all measurement vectors from 0 through $k-1$
$P(x)$	Probability density function of x
Z^k	Set of k observations

Abbreviations

ACFR	Australian Center for Field Robotics
AI	Artificial Intelligence
CMA	Covariance Matching Approach
COTS	Commercial-Of-The-Shelf
CR	Confidence Region
DCM	Direction Cosine MAtrix

DGPS	Differential Global Positioning System
DoD	Department of Defence
DoF	Degrees of Freedom
ECEF	Earth-Centered Earth-Fixed
ECI	Earth-Centered Inertial
EKF	EKF
FDI	Fault Detection and Identification
FMEA	Failure Modes and Effective Analysis
GA	Genetic Algorithms
GDOP	Geometric Dilution of Precision
GLONASS	Global Navigation Satellite System
GLR	Generalized Likelihood Ratio
GPS	Global Positioning System
GRMMAE	Generalized Residual Multiple Model Adaptive Estimation
IMM	Interactive Multiple Model
IMU	Inertial Measurement Unit
INS	Inertial Navigation System
IODC	Issue Of Date Clock
IODE	Issue Of Date Ephemeris
KF	Kalman Filter
LKF	Linear Kalman Filter
LOS	Line of Site
MBKF	Multiple Bank Kalman Filters
MMAE	Multiple Model Adaptive Estimation
MMSA	Minimal Mean Squared Error
MTBF	Mean Time Between Failure
NA	Navigation Algorithm
NASA	National Aeronautics and Space Administration
NED	North-East-Down
NEES	Normalized (state) Estimation Error Squared
NIS	Normalized Innovation Squared
NN	Neural Networks
PDF	Probability Distribution Function
PPS	Precision positioning System
PRN	pseudorandom noise
PSO	Particle Swarm Optimization
SA	Selective Availability
SCT	Statistical Consistency Test
SDINS	Strapdown Inertial Navigation System

SLAM	Simultaneous Localization and Mapping
SPRT	Sequential Probability Ratio Test
SPS	Standard Positioning System
SV	Satellite Vehicle
TOW	Time Of Week
UAV	Unmanned Aerial Vehicle
UGV	Unmanned Ground Vehicle
UKF	Unscented Kalman Filter
UT	Unscented Transformation

Glossary of Terms

Conning: The cyclic motion of one axis due to rotational motion of the other two axes (Sukkarieh, S., 2000). Such motion can result due to oscillatory motion such as vibration being undetected by the inertial navigation system.

Dead reckoning: Types of navigation systems which rely on the continuous updating of the position data derived from inputs of velocity components or speed and heading generated from a known start position. Inertial navigation systems are considered as sort of dead reckoning systems.

Disturbance: An unknown and uncontrolled input acting on a system (Gustafsson, F., 2000).

Ellipsoid: A three-dimensional shape formed by rotating an ellipse about its minor axis.

Ergodicity: A process is considered ergodic if all of its statistical parameters, mean, variance, and so on, can be determined from arbitrary chosen member functions. A sampled function is considered ergodic if its time-average statistics equal the ensemble averages (Grewal, M. S., & Andrews, A. P., 2001).

Error: Deviation between a measured or computed value of an output variable and the true, specified or theoretically correct value (Gustafsson, F., 2000).

Failure: Permanent interruption of a system ability to perform a required function under specified operating conditions (Gustafsson, F., 2000).

Fault: Unpermitted deviation of at least one characteristic property of parameter of the system from acceptable / ususal / standard condition (Gustafsson, F., 2000).

Fault detection: Determination of faults present in a system and time of detection (Gustafsson, F., 2000).

Fault isolation: Determination of kind, location and time of detection of fault. Follows fault detection (Gustafsson, F., 2000).

Fault identification: Determination of the size and time-variant behavior of a fault. Follows fault isolation (Gustafsson, F., 2000).

Fault diagnosis: Determination of kind, size, location and time of fault. Follows fault detection and includes fault isolation and identification (Gustafsson, F., 2000).

Monitoring: A continuous real time task of determining the conditions of a physical system, by recording information recognizing and indicating anomalies of the behavior (Gustafsson, F., 2000).

Perturbation: An input acting on a system which results in a temporary departure from current state (Gustafsson, F., 2000).

Residuals: Processed measurements. Kalman filter residuals, which are the differences between state estimates predictions and the measurements predictions are called innovations. They can be used as fault indicators, based on deviation between measurements and model-equation-based computations.

Sagnac Effect: When computations for the satellite position are made in an ECEF coordinate system, and during the propagation time of the satellite vehicle signal transmission, a clock of the surface of the Earth will experience a finite rotation with respect to an ECI coordinate system (Kaplan, E. D., & Hegarty,C. J., 2006)

Sculling: A combination of linear and angular oscillatory motions of equal frequency in orthogonal axes.

Symptom: Change of an observable quantity from normal behavior (Gustafsson, F., 2000).

Time-Invariant System: A system is time-invariant if a time shift in the input result in a corresponding time shift in the output. The output of a time-invariant system depends on time differences and not on absolute values of time (Stremler, F. G., 1990).

Acknowledgements

Sincere thanks go to Professor Mohammad-Ameen Al jarrah for his guidance, support and for always believing in me. For the nice trip to Australia, which left a big effect on my research.

Thanks also goes to Assistant Professor Mamoun Abdel-Hafez for his enthusiasm and for also pushing me for more.

To all the guys in the Mechatronics Center, Mohammad Hasan, the best GTA ever, Amer, the most cooperating GTA, Leith, for always asking, I hope you will make a great IMU, Mahmood Geith, the president of a huge number of communities, Ali, thank you for the "IT Crowd" days. For all my colleagues in the Mechatronics.

To our UAE Army officers, Captain Fisal, thanks for always calling and for the nice rides in the "S500", the "future" Captain Hamad, thanks for your friendship, nice rides in the "Camry", long will I remember the nights we spent working in the Lab. and thanks for "Naruto", the two hard drives and the hard times in Musaffah.

Special thanks to Associate Professor Salah Sukkarieh for guiding, pushing, advising and hospitality.

For the people of the ACFR, Sharon, Ben, Jason, Tim, Mitch, and the guy from Korea, I can't spell his name.

Thanks go to the ILL people at AUS-Library, for your non-ending support.

To may work colleagues in Abu Dhabi Police, Mahmood, Khalid, Salloom, Yusif, Nabeil, the "CODE3" guy, thanks for your Emails and calls, the policemen of traffic and patrols, thanks for the nice days and laughter.

To my best friends, Abed-Salam, the descendant of the great Al-Sharief family, thanks for the "Mugs and Beans", Khalid Ford, my old friend, for the days in Amman and Abu Dhabi, for sharing the headaches and the "hot-dogs".

To my parents, brother and sisters, whom my words can't describe.

To Knowledge.

Chapter 1

Introduction

1.1 Objectives

The main objective of this thesis is to design a low-cost inertial navigation system (INS) and integrate it with the Global Positioning System (GPS), while achieving the overall system integrity. In doing so, this thesis aims to:

- Understand the nature of inertial navigation, what is the information it can provide and how accurate they are. Therefore, a detailed understanding of inertial sensors characteristics is required and their effects on the mechanisation process should be investigated. It is also important to look into the inertial navigation computations and look for algorithms that can achieve reliability in the mechanisation process.
- Understand the Global Positioning System (GPS) principles and to learn how to read and process its observables.
- Implement an INS/GPS integrated navigation system using the Kalman filter. This requires a detailed understanding of the Kalman filter equations and to learn about filter consistency and tuning. The filter can take different forms, such as loosely-coupled and tightly-coupled. Therefore, it is essential to understand different filter implementations and what information they can provide.
- Address the issue of navigation system integrity. The fact that, for a navigation system to achieve integrity it should have a fault detection and identification

(FDI) (or simply diagnosis) algorithm. Therefore, different FDI algorithms should be investigated.

- Test the reliability of the developed algorithms with actual inertial and GPS data. This is considered as a first step towards real time automation where the navigation algorithm will be implemented to provide navigation data to an unmanned areal vehicle (UAV) control system.

1.2 The Navigation Problem

Unmanned mobile robots (such as unmanned vehicles) are usually designed to accomplish a certain task with minimum or no-human interaction. Being autonomous, the navigation of unmanned robots takes a number of forms, which are represented by the venn diagram of Figure 1.1. In autonomous navigation missions, it is essential for the unmanned vehicle to localize itself. The focus of this thesis falls in the localization domain. In order for the robot to localize itself, it should estimate (at least) its position and velocity. Position and velocity are relative quantities; they have to be measured with respect to a reference. However, acceleration is an absolute quantity. This motivated the utilization of accelerometers in autonomous vehicle implementations. Integrating the accelerometers output yields the velocity. Integrating the velocity once yields the position. In other words, inertial navigation is a form of dead reckoning.

In order to remove undesired components from accelerometers measurements, the vehicle orientation should be estimated. Gyroscopes, which measures vehicle rotational velocity, were integrated with accelerometers to form an Inertial Measurement Unit (IMU). In there beginnings, IMUs were implemented in *gimballed* configuration (King, A., D., 1998), where gyros were used as the sensing elements in null-seeking servos, with the output of each gyro connected to a servo-motor driving the appropriate gimbal, keeping the platform, on which accelerometers are mounted, aligned

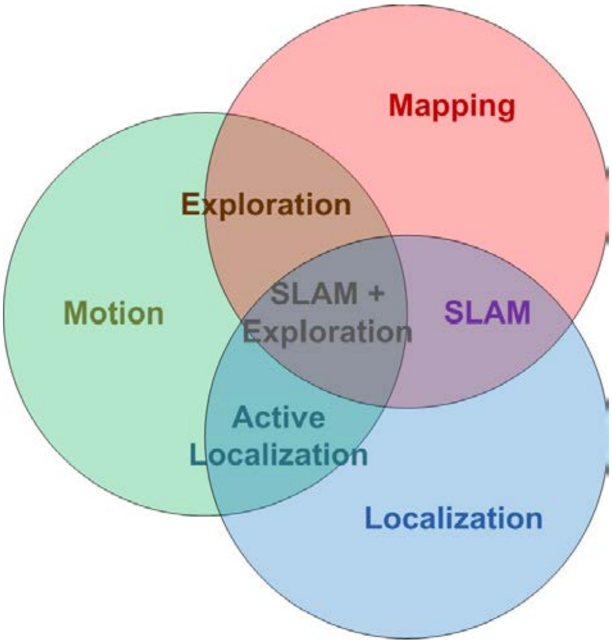


Figure 1.1: The navigation problem

in the inertial frame coordinates. Gimballed IMUs can be very accurate and reliable. However, they are expensive, mechanically very complex, expensive to maintain and extensive calibration has to be done. Strapdown Inertial Navigation Systems (SDINS) overcome the disadvantages of the gimballed ones, but on the cost of computational complexity. The gimbals were removed, and inertial sensors were *strapped down* onto the mounting frame of the vehicle. Gyroscopes are used to measure the vehicle rotational rates to estimate the pointing direction of accelerometers axes. Mechanisation equations are utilized to serve this purpose and remove undesired components from accelerometers measurements. In other words, while gimballed INS are *mechanically* gimballed, strapdown INS are *mathematically* gimballed.

The integration of inertial systems errors cause there estimates to drift over time. External aiding systems are integrated with inertial ones to bind these errors. However, aiding systems might have their own weaknesses. For this reason, aiding systems should be selected in such a way that they and the aided ones are natural complements. The Global Positioning System (GPS) is the most popular aid for inertial systems.

Being implemented for commercial purposes, INS/GPS integrated systems employ low-cost sensors. With such sensors, failures become more probable and the system reliability degrades. To achieve integrity and therefore reliability, the system health should be monitored. Therefore, fault detection and identification becomes an essential requirements in mobile robotics implementations represented by figure 1.1.

1.3 Navigation System Integrity

High integrity navigation system design doesn't have a particular procedure. It varies from design to design. However, the general design may look like figure 1.2. The block diagram highlights the major components of the navigation system. Design specifications are the motive behind this design. Typical design specifications may include the maximum expected frequency of maneuvering and the required accuracy of the system. The former determines the cutoff frequency of the required filter, and therefore determines the maximum bandwidth of the sensors utilized, while the later determines the maximum value of estimated covariance matrix, and therefore the accuracy of the sensors employed. Therefore, sensors are selected such that they agree with design specifications. One major step towards a high integrity design is the determination of employed sensors errors characteristics, such as biases, misalignments, scale-factor error, and random-walk through sensor calibration or manufacturer drawsheets. This is a critical step in the design, since unexpected errors can rise from mismodelled sensor parameters.

A system is reliable as it's most unreliable component (Scheding, S., 1997), and even the most reliable component can fail. Therefore, the design should contain redundant information. This information can be provided by other sensors, such redundancy is referred to as *hardware redundancy*, or by a mathematical model, which is referred to as *analytical redundancy*. In fact, in systems where redundancy is em-

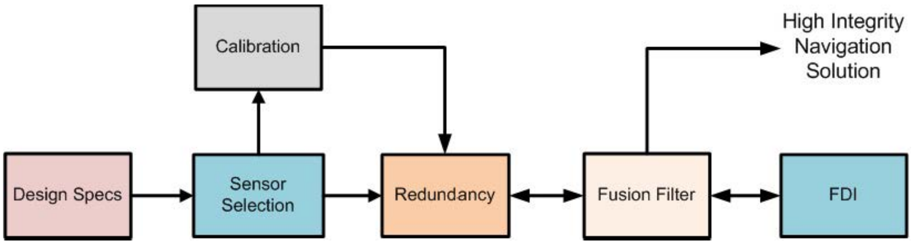


Figure 1.2: General steps towards a high integrity navigation system design

ployed, it becomes (usually) impossible to distinguish between the actual source of information and the redundant one. Redundancy is a must in systems with low-cost components. In fact, redundancy can be an essential requirement to achieve design specifications, if the previously selected sensors *alone* did not satisfy one of the design requirements. One example of such situations may rise if the previously selected sensors did not satisfy the condition of required filter cutoff frequency and, therefore, the redundant system should contain sensors such that the frequency response of the actual sensors and the redundant ones are *complementary* (Scheding, S., 1997), and redundancy (and therefore design specifications) is said to be achieved by the integration of *natural complements*.

The integration of actual information and redundant ones is considered by the fusion filter. It estimated states of interest and provides required to calibrate sensors under consideration. This online calibration compensates for low frequency faults due to sensors biases and drifts. The fault detection and identification scheme extracts information from the filter and utilizes them to generate residual, the residuals are then used by the FDI system to detect faults. Once a fault is detected, the FDI identifies the fault parameters. Based on its decision, the FDI system provides the filter with information on how to do the fusion process. The FDI scheme should consider the following:

- Since residuals are the bases of fault detection, the FDI scheme should guarantee that all faults will be transmitted to the residuals. The frequency domain

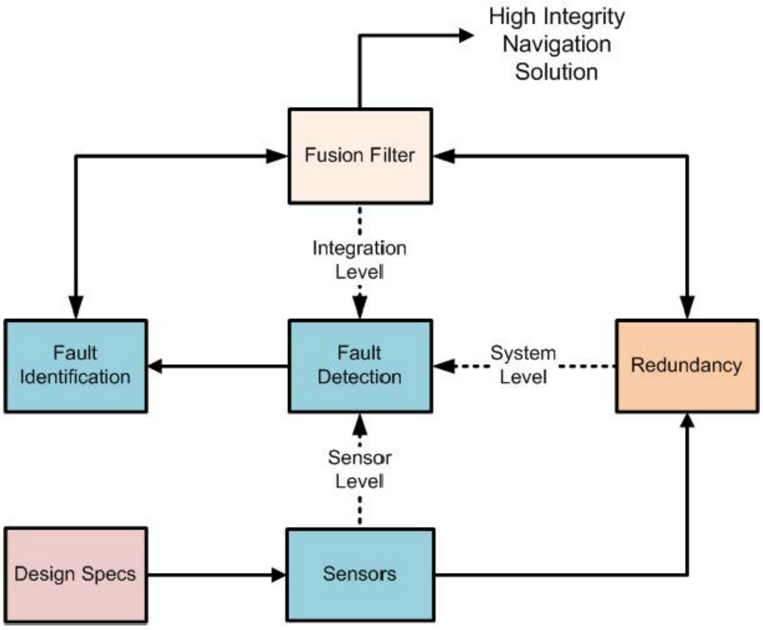


Figure 1.3: Different level of FDI implementations

may be utilized to guarantee faults detectability, and therefore, sensors are re-selected such that their frequency domain become identical. Such scheme is referred to as frequency redundancy (Scheding, S., 1997).

- Fault diagnosis may be implemented in different system levels, as shown in figure 1.3. This guarantees that all faults will be detected and identified more efficiently.

1.4 Analytical Redundancy or Hardware Redundancy?

Fault detection is cost effective method that is based on data consistency. Such consistency requires redundant measurements to generate residuals on which the fault detection process relies. Two types of redundant implementations exist, analytical redundancy and hardware redundancy. *Hardware redundancy* is based on the utilization of multiple aiding sensors, such as Global Positioning System (GPS) receivers, wheel encoders, doppler radars and laser range finders, where the dimensions of the measurements vector becomes larger than that of the state estimate vector. Redundant sensors may be identical, in term of their physical structure and quantities they measure, or similar, by providing identical measurement quantities but operating by different physical principles. Although, the utilization of similar sensors may imply hardware redundancy, it is more common for similar redundant sensors to have identical frequency responses. Therefore, the utilization of similar sensors that have identical frequency characteristics is referred to as *frequency redundancy* (Scheding, S., 1997). In other words, hardware redundancy implies identical hardware. However, frequency redundancy implies identical frequency ranges.

Analytical redundancy is based on mathematical modelling of the system dynamics, such as the utilization of the mechanization equations of the inertial navigation system (INS) or the dynamic model of the vehicle driven by control commands. It is the basis for residuals generation (Chow, E., Y. & Willsky, A. S., 1984). Forms of analytical redundancy include direct redundancy, which makes direct relations (algebraically related) between instantaneous outputs of sensors, and temporal redundancy, which makes indirect relations based on histories between sensor outputs and actuator inputs. Analytical redundancy in more reliable and cost effective than hardware redundancy. In this thesis, analytical redundancy is provided through the utilization of inertial navigation mechanisation equations. It is a self contained system with high short term accuracy and immunity to jamming and interference.

1.5 Thesis Contributions

There are two main contributions in this thesis, and they are as follows:

- **Development of INS/GPS navigation filter that achieves integrity**. This includes a detailed analysis on the Kalman filter operation, including filter consistency, tuning and integrity. It also includes investigation of different

Kalman filter implementations. For the filter to be robust, fault detection and identification techniques are applied to detect and diagnose failures. Simulations as well as actual tests are included to verify the performance of the designed algorithms.

- **Implementing a sequential fault detection and identification scheme**. Sequential FDI is a new topic, never been addressed before. It aims to integrate two different FDI schemes where they compensate for each other weaknesses. The resulted scheme proved to have better performance than both methods with the advantage of the reduction in overall system complexity.

1.6 Thesis Structure

The thesis structure is as follows:

Chapter 2 develops the required background for this thesis to achieve it's objectives. The chapter reviews the Kalman filter equations and discusses it's most critical issues from practical point of view. The Unscented Kalman filter is also discussed as the most accurate and least expensive state of art in non-linear filtering. The chapter then discusses adaptive techniques for Kalman filter tuning. These techniques include the covariance matching approach (CMA), the multiple mmodel adaptive estimation (MMAE), and artificial intelligent (AI) techniques.

Chapter 3 discusses inertial navigation systems (INS) and sets the basis required to develop an accurate and reliable inertial navigation algorithm. Mechanization equations are discussed in both the local geographic or the NED and the ECEF coordinate systems. However, it emphasizes more on the NED implementation since the attitude of vehicle is more physically intuitive in the NED coordinates than in the ECEF coordinates. Also, the NED coordinates separates the unstable vertical axis from the more stable horizontal axes and provides a more intuitive schemes for analyzing INS errors than the ECEF coordinates. The chapter then moves to discuss inertial sensor errors. It derives expressions for the general error equations. Finally, the INS mechanisation is verified via simulation. First, for the case of accurate and error free sensors. Then, when errors are introduced to inertial sensors.

Chapter 4 considers the issue of aiding an inertial navigation system using the global positioning satellite navigation system. It discusses various aspects of the GPS including principles, orbital parameters, calculating satellites and receiver position using satellites ephemeris data. Two different navigation filter schemes are also discussed, namely, the loosely-coupled and the tightly-coupled. Simulation and real system performance are represented to verify the performance of the filter.

Chapter 5 looks at how a navigation system can achieve integrity. It is stated that for the navigation system to achieve integrity it should have a fault detection and identification scheme. The chapter provides a definition of fault, looks at different fault classifications, and surveys various fault detection and identification schemes. A new sequential fault diagnosis scheme is presented. This scheme employs an innovation-based technique for fault detection and a model-based technique for fault identification. The resulted sequential scheme achieves better fault detection and identification than both systems. Finally, simulated as well as actual FDI performance are presented to verify its performance.

Finally, **Chapter 6** provides a conclusion for this thesis along with suggestions for future work.

Chapter 2

Literature Review

2.1 Introduction

Literature is abundant with studies on Inertial Navigation Systems (INS), Global Positioning Systems (GPS), Estimation theory and Kalman Filtering (KF) and Fault Detection and Identification in Dynamic Systems. This chapter provides the background required for this thesis. However, some topics will be left for relevant chapters. In particular this chapter addresses all aspects related to Kalman filtering in both linear and nonlinear formats. It provides a detailed discussion on filter tuning, Innovation consistency test and initialization. Before conclusions, this chapter provides an advanced survey on the adaptive techniques applied to Kalman filtering.

The other thesis related topics, such as inertial navigation, GPS data processing, Kalman filter implementations and fault detection and identification (or shortly, fault diagnosis) techniques, are reviewed in their related chapters.

2.2 Estimation and The Kalman Filter

The Kalman filter is addressed by a huge number of books and theses, of which the most interesting are (Maybeck, P. S., 1979), (Welck, G. & Bishop, G., 1997), (Durrant-Whyte, H. F., 2001), (Grewal, M. S., & Andrews, A. P., 2001), (Sukkarieh, S., 2000).

2.2.1 The Linear Discrete-Time Kalman Filter

Assume that the system under consideration can be described by a simple linear, discrete-time state transition equation of the form

$$\mathbf{x}(k) = \mathbf{F}(k)\mathbf{x}(k-1) + \mathbf{B}(k)\mathbf{u}(k) + \mathbf{G}(k)\mathbf{w}(k) \tag{2.1}$$

where $\mathbf{x}(k-1)$ is the previous estimated state at time $k-1$, $\mathbf{u}(k)$ is a control input vector, $\mathbf{w}(k)$ is an additive process (motion) noise, $\mathbf{B}(k)$ and $\mathbf{G}(k)$ are input and noise transition matrices, $\mathbf{F}(k)$ is the state transition matrix, and $\mathbf{x}(k)$ is the predicted state at the next time step k. Observations of the state of this system made at time k are assumed also to follow a simple linear equation of the form

$$\mathbf{z}(k) = \mathbf{H}(k)\mathbf{x}(k) + \mathbf{v}(k) \tag{2.2}$$

where $\mathbf{z}(k)$ is the observation made at time k, $\mathbf{x}(k)$ is the state at time k, $\mathbf{H}(k)$ is the observation matrix (or model), and $\mathbf{v}(k)$ is an additive observation noise.

Both ,$\mathbf{w}(k)$ and $\mathbf{v}(k)$ are assumed to be Gaussian, uncorrelated and zero mean white noises

$$E\left[\mathbf{w}(k)\right] = E\left[\mathbf{v}(k)\right] = 0, \quad \forall k. \tag{2.3}$$

with corresponding covariances

$$E\left[\mathbf{w}(i)\mathbf{w}^T(j)\right] = \begin{cases} \mathbf{Q}(k) & i = j = k \\ \mathbf{0} & i \neq j \end{cases} \tag{2.4}$$

and

$$E\left[\mathbf{v}(i)\mathbf{v}^T(j)\right] = \begin{cases} \mathbf{R}(k) & i = j = k \\ \mathbf{0} & i \neq j \end{cases} \tag{2.5}$$

and since they are uncorrelated

$$E\left[\mathbf{w}(i)\mathbf{v}^T(j)\right] = \mathbf{0}, \quad \forall i, j. \tag{2.6}$$

The Kalman filter is a set of predictor-corrector equations that is optimal in the sense that it provides the Minimal Mean Squared Error (MMSE) and minimizes the estimated error covariance. It estimates the states at time k given all observations up to time k, $\hat{\mathbf{x}}(k|k)$.

In this subsection, a brief discussion of the filter equation is included. A more detailed derivation of the Kalman filter equation is listed in appendix C. The algorithm starts by predicting an estimate of the state at time k given only information up to time $k-1$, which is called prediction, and is written as $\hat{\mathbf{x}}(k|k-1)$.

$$\hat{\mathbf{x}}(k|k-1) = \mathbf{F}(k)\hat{\mathbf{x}}(k-1|k-1) + \mathbf{B}(k)\mathbf{u}(k) \tag{2.7}$$

Where $\hat{\mathbf{x}}(k-1|k-1)$ is the state estimate at time $k-1$ given all information up to time $k-1$.

The prediction uncertainty is represented by the covariance matrix $\mathbf{P}(k|k-1)$, such that

$$\mathbf{P}(k|k-1) = \mathbf{F}(k)\mathbf{P}(k-1|k-1)\mathbf{F}^T(k) + \mathbf{Q}(k) \tag{2.8}$$

Where $\mathbf{P}(k-1|k-1)$ represents the variance of the error in the estimated states at time k given all observations up to time $k-1$. In a navigation system, equations 2.7 and 2.8 occur each time a sample from the dead reckoning sensor is obtained.

The correction stage starts when observations from external aiding sensors are obtained. The estimated (corrected) states are given by

$$\hat{\mathbf{x}}(k|k) = \hat{\mathbf{x}}(k|k-1) + \mathbf{W}(k)\nu(k) \tag{2.9}$$

Where $\mathbf{W}(k)$ is the Kalman gain and $\nu(k)$ is the innovation vector, given by

$$\nu(k) = \mathbf{z}(k) - \mathbf{H}(k)\hat{\mathbf{x}}(k|k-1) \tag{2.10}$$

Where $\mathbf{z}(k)$ is the observation given *only* at time k and $\mathbf{H}(k)\hat{\mathbf{x}}(k|k-1)$ is the predicted observation at time k. The Kalman gain is what minimizes the mean squared error, and is computed by

$$\mathbf{W}(k) = \mathbf{P}(k|k-1)\mathbf{H}^T(k)\mathbf{S}^{-1}(k) \tag{2.11}$$

Where $\mathbf{S}(k)$ is the innovation covariance, and is computed by

$$\mathbf{S}(k) = \mathbf{H}(k)\mathbf{P}(k|k-1)\mathbf{H}^T(k) + \mathbf{R}(k) \tag{2.12}$$

Due to this correction stage, the covariance matrix is updated as follows

$$\mathbf{P}(k|k) = (\mathbf{I}-\mathbf{W}(k)\mathbf{H}(k))\mathbf{P}(k-1|k-1)(\mathbf{I}-\mathbf{W}(k)\mathbf{H}(k))^T + \mathbf{W}(k)\mathbf{R}(k)\mathbf{W}^T(k) \tag{2.13}$$

By looking into the Kalman filter equations, the following features can be observed:

- The cyclic structure of the filter, where some computations are performed at each time step.
- The covariance loop is independent of both the state estimate loop and the observations. This implies that state covariance and the Kalman gain can be computed off-line prior to state estimate prediction and correction.
- The Kalman filter is a weighted sum of the prediction and observation.
- It provides estimates of states which may not be a part of the observation vector by means of the cross-correlation (i.e. off-diagonal elements of $\mathbf{P}(k)$) between the observed and unobserved states, which makes the Kalman filter operate as an observer (Durrant-Whyte, H. F., 2001).

2.3 Kalman Filter Tuning and Performance Characteristics

In this section a detailed analysis of the Kalman filter performance is introduced. This analysis includes filter consistency and consistency diagnosis, filter initialization and steady state performance, growth of uncertainty and the effect of sensors errors on filter consistency. These concepts are discussed in (Durrant-Whyte, H. F., 2001) and the math behind is represented by (Kreyszig, E., 1993).

A Kalman filter consists of two stages; the prediction stage where the process model provides a predicted state estimate $\hat{\mathbf{x}}(k|k-1)$ with uncertainty of $\mathbf{P}(k|k-1)$, and the correction (update) stage where observations $\mathbf{z}(k)$ (when they become available) are used to enhance estimation uncertainty and produce a corrected state estimate $\hat{\mathbf{x}}(k|k-1)$ with less uncertainty $\mathbf{P}(k|k)$. The whole fusion process is centered around the method by which predictions and observations are weighted. The Kalman filter state estimate is a weighted average of the state prediction and observations.

$$\hat{\mathbf{x}}_{k|k} = \mathbf{K}_k^{(1)}\hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k^{(2)}\mathbf{z}_k \tag{2.14}$$

The gain matrices $\mathbf{K}_k^{(1)}$ and $\mathbf{K}_k^{(2)}$ are derived in such a way that the conditional mean squared estimation error is minimized. The Kalman gain derived to serve this purpose is calculated as in equation 2.11. The derivation of this gain requires an exact knowledge of the process noise covariance matrix $\mathbf{Q}(k)$ and the measurement noise covariance matrix $\mathbf{R}(k)$ representing the true model. However, this information is not available in most of real time systems and the performance of the filter loses its optimality. To implement a successful Kalman filter, it is essential *to tune* for these matrices. The process of tuning is not random, the filter should pass a number of tests to validate its performance. These test are introduced in next section and they aim to prove the consistency of the filter.

Both tuning and fault diagnosis should provide methods for monitoring both stages of the Kalman filter. The conventional may is to compare estimations of both stages with true system values. However, the true states are not usually available for comparison and the only connection between the Kalman filter world of virtual reality and the real world is the observations $\mathbf{z}(k)$. The innovation sequence serves as a tool to validate the observations, check the consistency of prediction and to decide a criteria to fuse both of them.

2.3.1 Filter Consistency

The consistency of static estimators is defined as convergence of the estimator to the true value. For dynamic estimators no convergence of estimates occurs and a filter is called consistent if its estimation errors are compatible with the filter-calculated covariance. For an estimator to be consistent it must satisfy a couple of requirements. They are the unbiasedness and the covariance matching requirements (Durrant-Whyte, H. F., 2001)(Bar-Shalom, Y., Li, X. R., & Kirubarajan, T., 2001).

Definition 2.1

Two-Moment Conditions:

An estimate is said to be consistent if it is unbiased*;*

$$E\left[\mathbf{x}(k) - \hat{\mathbf{x}}(k|k)\right] \triangleq E\left[\delta\mathbf{x}(k|k)\right] = 0$$

and its actual covariance matches the filter-calculated (theoretical) covariance

$$E\left[(\mathbf{x}(k) - \hat{\mathbf{x}}(k|k))(\mathbf{x}(k) - \hat{\mathbf{x}}(k|k))^T|\mathbf{Z}^k\right] \triangleq E\left[\delta\mathbf{x}(k|k)\delta\mathbf{x}^T(k|k)\right] = \mathbf{P}(k|k)$$

Filter gain depends on the covariance, so that consistency is necessary for filter optimality. Theorem 2.1 provides a criteria for testing the consistency of an estimator.

Theorem 2.1

Statistical Tests for Filter Consistency:

Under the hypothesis that a filter is consistent, $\mathbf{x}(k)$ *is the true value of state and* $\hat{\mathbf{x}}(k)$ *is the state estimate, the normalized state estimation error squared (NEES)*

$$p(k) = (\mathbf{x}(k) - \hat{\mathbf{x}}(k))^T P^{-1}(k|k)(\mathbf{x}(k) - \hat{\mathbf{x}}(k))$$

is a χ^2 *distribution in* $n = dim(\mathbf{x}(k))$ *degrees of freedom and*

$$E[p(k)] = n$$

The NEES is utilized to check the consistency of the correction stage. However, in real situations the true state to be estimated is unknown and the difference between the true state and the estimate of the state is replaced by the innovation; that is the difference between the observations and the predicted observations. The innovation sequence should be zero-mean (unbiased), white (uncorrelated) and has covariance $\mathbf{S}(k)$ if the filter is operating correctly (Mehra, R. K., 1970).

Similar to Theorem 2.1, the normalized innovation squared (NIS) is used as a **test of unbiasedness** for the innovation sequence. It is computed according to

$$q(k) = \nu^T(k)\mathbf{S}^{-1}(k)\nu(k) \tag{2.15}$$

under correct filter assumptions, the NIS is χ^2 in m degrees of freedom, where m is the order of the observation vector, and

$$E[q(k)] = m \tag{2.16}$$

This test can be implemented by computing the moving average of the NIS sequence for a number of N samples as follows

$$\bar{q} = \frac{1}{N}\sum_{i=1}^{N} q(i) \tag{2.17}$$

testing the hypothesis that $N\bar{q}$ is a χ^2 random variable in Nm degrees of freedom, a confidence interval (bounding set) $[b_1, b_2]$ is constructed such that

$$P\left\{\bar{q} \in [b_1, b_2] | H_0\right\} = 1 - Q \tag{2.18}$$

represents the (two-sided) percentage probability that a particular observation lies within an ellipsoid (Van Trees, H., L., 1971). This percentage probability is also referred to as the confidence level. Given a confidence level, the confidence interval is calculated as follows

$$\begin{aligned} b_1 &= \frac{\chi^2_{Nm}\left(\frac{Q}{2}\right)}{N} \\ b_2 &= \frac{\chi^2_{Nm}\left(1 - \frac{Q}{2}\right)}{N} \end{aligned} \tag{2.19}$$

where $\chi^2_{Nm}(x)$ is evaluated using tables from appendix F. The average $\bar{q}$ should lie between these boundaries if the filter is consistent (Kreyszig, E., 1993)(Bar-Shalom, Y., Li, X. R., & Kirubarajan, T., 2001). This consistency test may also be implemented in real time by checking that the NIS is χ^2 with confidence level given by the (one-sided) percentage probability such that

$$\nu^T(k)\mathbf{S}^{-1}(k)\nu(k) < \gamma \tag{2.20}$$

where

$$\gamma = \chi^2_m\left(P\left\{\bar{q} < \gamma | H_0\right\}\right) \tag{2.21}$$

The chi-square test of equations 2.15, 2.17, 2.19 and 2.21 is explained as follows, if the innovation sequence is biased, the normalized innovation squared of equation 2.15 exceeds the threshold of 2.21. If the bias is significant, the moving average in 2.17 increases beyond the limits of the two-sided interval $[b_1, b_2]$ determined by 2.19. To estimate ones confidence interval, Chi-squared distribution in appendix F is used. Given a confidence level (say 95%) and the number of degrees-of-freedom (DoF) the Chi-squared distribution table is used to estimate the upper and lower confidence limits and the threshold of equation 2.20.

The next step is the **whiteness test** for the innovation sequence. This is conducted by means of the autocorrelation function of the innovation. The purpose is to test that the innovation is indeed white with

$$E\left[\nu(i)\nu^T(j)\right] = \mathbf{S}(i)\delta_{ij} \tag{2.22}$$

The time-averaged (biased) autocorrelation used to test the whiteness of the innovation sequence is given in (Durrant-Whyte, H. F., 2001) and (Bar-Shalom, Y., Li, X. R., & Kirubarajan, T., 2001) as

Theorem 2.2

Test of Whiteness:

For an innovation sequence $\nu(k)$ to be white with a large enough number of samples N, the time-averaged autocorrelation function given by

$$\rho_l(\tau) = \frac{1}{N-\tau}\sum_{i=0}^{N-\tau-1} \nu^T(i)\nu(i+\tau)$$

where $l = 1, \ldots, m$, is normally distributed with mean zero and variance $\frac{1}{N}$.

Thus, from theorem 2.2, the 95% confidence limits for $\rho_l(\tau)$ are $\pm\frac{1.96}{\sqrt{N}}$. In other words, to test the whiteness of an innovation sequence, plot the values of the statistical autocorrelation function and check the number of times when its values exceeds the 95% confidence limits. If this number is less than 5% of the total, the innovation is white. It is important to remember that this test is based on the assumption of large N. If N is small, other tests may be used (Mehra, R. K., 1970). A filter that does not pass the whiteness test is considered to be operating in the *suboptimal* mode of operation.

The last step in validating filter's model, is to compare the actual innovation covariance, computed by the left side of equation 2.22 with its theoretical value, given by equation 2.12. This can also be accomplished if the innovation sequence is compared with its corresponding estimated covariance.

2.3.2 Filter Tuning

The Kalman gain, obtained by equation 2.11, is based on the fact that the process noise covariance matrix, $\mathbf{Q}_k$, and the observation noise covariance matrix, $\mathbf{R}_k$, represent the true errors in the system. However, in most cases, this in not true and $\mathbf{Q}_k$ and $\mathbf{R}_k$ are unknown. The process of selecting appropriate values of these matrices to achieve the optimal solution, which minimizes the conditional estimated squared error, is referred to as *filter tuning*.

Tuning of the Kalman filter is a very hard process, which is usually left for professionals. It takes much order of magnitude of the time required to implement the filter itself. However, a simple procedure can be applied to tune the filter as close as possible to its optimal performance. The consistency tests discussed in subsection 2.3.1 serves as indicators assisting the tuning process. This process is summarized in Table 2.1. Figure 2.1 represents the relation between tuning parameters ($\mathbf{Q}$ and $\mathbf{R}$) and the levels of correction and correlation of the innovation sequence.

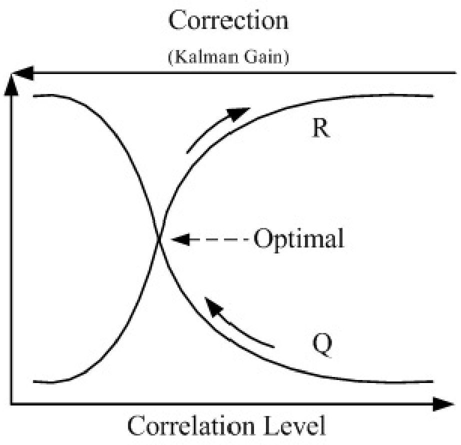


Figure 2.1: Tuning membership function.

These steps results in a filter's model which operates as close as possible to its optimal performance. However, it does not guarantee for it to be a **global optima**. Global optimality of the Kalman filter is an active research area. Artificial Intelligence (AI) algorithms are utilized in different forms to search for the values of process and measurement noise covariances $\mathbf{Q}(k)$ and $\mathbf{R}(k)$ which achieve global optimality. AI algorithms utilized in Kalman filter optimization problem include Fuzzy Logic, Neural Networks (NN) and Genetic Algorithms (GA) are addressed in subsection 2.4.3.

The tuning of the Kalman filter is critical for its performance. Table 2.2 shows the effect of process and measurement noise covariance mismatching. The tuning process starts by fixing one of $\mathbf{Q}(k)$ or $\mathbf{R}(k)$ and varying the other one. It is much more sensitive to changes in $\mathbf{R}(k)$ than to changes in $\mathbf{Q}(k)$. So that, changing $\mathbf{R}(k)$ represents coarse tuning, while changing $\mathbf{Q}(k)$ represents fine tuning. The diagonal elements of $\mathbf{R}(k)$ and $\mathbf{Q}(k)$ depend on the noise levels of the sensors utilized which can be obtained from the manufacturer data sheets or through experimentations (Sukkarieh, S., 2000).

Table 2.1: Steps used as a procedure for Kalman filter tuning process

Step	**Consistency Test**	**Observations**
Adjusting the ratio $\frac{\mathbf{Q}_k}{\mathbf{R}_k}$	Autocorrelation function of the innovation sequence	Regardless of the absolute noise level, a too small ration implies an untrusted observations over predictions and the innovation sequence will be correlated
Setting the noise values of $\mathbf{Q}_k$ and $\mathbf{R}_k$	χ^2 test of the NIS and the moving average NIS	An NIS or a moving average NIS that lies higher than its gating or confidence interval respectively, implies a high levels of $\mathbf{Q}_k$ and $\mathbf{R}_k$
Observing both the innovation sequence and its covariance	The shape of the innovation sequence and its relation to $\mathbf{S}_k$	The innovation sequence must look like white noise (unbiased and zero-mean) and shows no periodic-like behavior. By plotting the innovation sequence with the corresponding standard deviation $\pm\sigma$ (which is the square root of the corresponding diagonal element of $\mathbf{S}_k$) and $\pm 2\sigma$ bounds, at least 95% of the innovation sequence should fall within the $\pm 2\sigma$ bound

2.3.3 Initialization

The first step in implementing a Kalman filter is the selection of starting state estimate vector $\hat{\mathbf{x}}(0|0)$ and its corresponding uncertainty $\mathbf{P}(0|0)$. A low value of $\mathbf{P}(0|0)$ represents high confidence in initial estimate $\hat{\mathbf{x}}(0|0)$, and vise versa. Initial covariance with all zero elements represents complete confidence in initial state, but it results in a very slow converging filter. This is because the calculated Kalman gain depends on state covariance which will be changing, according to equation 2.8, by an amount equals to $\mathbf{Q}(k)$.

The true initial state is a random variable, which is assumed to be normally distributed with mean $\hat{\mathbf{x}}(0|0)$ and variance $\mathbf{P}(0|0)$. The initial state estimate has to be consistent with its initial covariance. Therefore, initial state estimate and covariance

Table 2.2: Effect of mismodeling the process and measurements noise covariances on the operation of the Kalman filter

Mismatch mode	**Consequences**	**Results of validation tests**
Large $\mathbf{Q}_k$	Prediction uncertainty grows according to equation 2.8. When observations become available, predicted uncertainty of inertial solution will be reaching a level where correction will occur regardless of observations accuracy	The innovation will show normal behavior. However, the NIS will be smaller than what would normally be expected and its moving average falls below the lower limit of its 95% confidence interval
Large $\mathbf{R}_k$	Implies an inaccurate observations, which will be less weighted than what would be expected normally	The innovation will show normal behavior. However, the NIS will be smaller than what would normally be expected and its moving average falls below the lower limit of its 95% confidence interval
Small $\mathbf{Q}_k$	Prediction uncertainty grows according to equation 2.8. When observations become available, predicted uncertainty of inertial solution will reach a level which is lower than the true uncertainty in inertial solution, the Kalman gain will be smaller than what is expected and observations will be less weighted than what should be, which results in less corrected errors	Predictions will be more trusted than what should be normally. The innovation sequence will show non-random correlated behavior, which appears in the time-averages autocorrelation test. The NIS will be larger than what would normally be expected and its moving average raises over the upper limit of its 95% confidence interval
Small $\mathbf{R}_k$	Implies accurate observation and fusion occurs regardless of their actual accuracy	The innovation sequence exceeds the computed standard deviation. The NIS will be larger than what would normally be expected and its moving average raises over the upper limit of its 95% confidence interval

matrix should pass the chi-squared test given by

$$\hat{\mathbf{x}}(0|0)^T\mathbf{P}(0|0)^{-1}\hat{\mathbf{x}}(0|0) \leqslant b_2 \tag{2.23}$$

where b_2 is the upper limit of the 95% confidence interval calculated by equation 2.19 using the chi-square distribution (see appendix F) with corresponding number of DoF.

The choice of a reasonably good initial estimates improves convergence and is *essential* in the convergence of the extended Kalman filter (Durrant-Whyte, H. F., 2001). In simulations, the initial state estimate is generated with a mean equal to the true initial state (which is fixed) and with covariance equal to the initial state covariance (which is chosen such that the error is at most 2 times the corresponding standard deviation) (Bar-Shalom, Y., Li, X. R., & Kirubarajan, T., 2001). Another way of doing that, is to reverse the procedure by choosing an initial covariance and then generating the initial estimate according to

$$\hat{\mathbf{x}}(0|0) = \mathbf{x}(0|0) + \zeta\sqrt{\mathbf{P}(0|0)} \tag{2.24}$$

where $\zeta \in [-2, 2]$ and the simplest method to choose an initial covariance is through

$$\mathbf{P}(0|0) = \alpha^2\mathbf{Q}(k) \tag{2.25}$$

and a reasonable start is with $\alpha = 10$. In real time implementation, the simplest initialization method can be done by polynomial fitting of a number of consecutive measurements.

It is a fact that initialization of the Kalman filter algorithms improves its convergence. However, it does not affect its steady state performance. The steady-state performance of the Kalman filter is uniquely determined by the values of process and observation noise covariances, $\mathbf{Q}(k)$ and $\mathbf{R}(k)$.

2.4 Adaptive Kalman Filter Techniques

One limitation of Kalman filter algorithm is the requirement of *precise a priori knowledge of system models and noise properties.* Once the filter is tuned, both of estimated $\mathbf{Q}(k)$ and $\mathbf{R}(k)$ guarantees the consistency of the state estimate and the optimization of the filter. However, due to environmental changes (such as weather changes or clouds, aircraft model variations, sensor failures and GPS signal blockage), $\mathbf{Q}(k)$ and $\mathbf{R}(k)$ no longer represent the true noise condition and the filter loses its optimality. Later is chapter 5 such situations will be referred to as faults. One approach to compensate for environmental changes is by the utilization of what is described as an adaptive implementation of the Kalman filter. By fusing adaptive estimation theory into the Kalman filter equations, the integrated algorithms achieves an online optimization scheme by providing updated estimates of $\mathbf{Q}(k)$ and $\mathbf{R}(k)$ that describes the true (or as close as possible to the truth) process and sensor noise covariances. This section provides an overview of the most popular adaptive implementations of the Kalman filter.

2.4.1 Covariance Matching Approach (CMA)

The basic idea behind CMA (Jwo, D. J., & Chang, S. C., 2007) (Loebis, D., Chudley, J., & Sutton, R., 2003), is to make the *actual* value of the innovation covariance $\hat{\mathbf{S}}(k)$ in consistence with its *theoretical* value $\mathbf{S}(k)$. The innovation sequence is used to compute the actual covariance as follows .

$$\hat{\mathbf{S}}(k) = \frac{1}{M}\sum_{i=i_0}^{N} \nu(i)\nu^T(i) \tag{2.26}$$

where M represents the number of samples in the estimation window, and $i_0 = N - M + 1$ is the first sample inside the estimation window. This actual covariance is used to compute a new estimate of the measurement noise covariance matrix $\hat{\mathbf{R}}$ as follows

$$\hat{\mathbf{R}}(k) = \hat{\mathbf{S}}(k) - \mathbf{H}(k)\mathbf{P}^{-}(k)\mathbf{H}^{T}(k) \tag{2.27}$$

This new estimated value of measurement noise covariance is then used to compute the Kalman gain as in equations 2.10 and 2.12 and to update the state covariance matrix as in equation 2.13. The same approach applies for the process noise covariance matrix $\hat{\mathbf{Q}}$, which is estimated as

$$\hat{\mathbf{Q}}(k) = \frac{1}{M}\sum_{i=i_0}^{N} \Delta x(i)\Delta x^{T}(i) + \mathbf{P}(k) - \mathbf{F}(k)\mathbf{P}(k-1)\mathbf{F}^{T}(k) \tag{2.28}$$

where, $\Delta x(k) = x(k) - \hat{x}^{-}(k)$. This equation can be written in term of estimated innovation covariance $\hat{\mathbf{S}}(k)$ as follows

$$\hat{\mathbf{Q}}(k) \approx \mathbf{W}(k)\hat{\mathbf{S}}(k)\mathbf{W}^{T}(k) \tag{2.29}$$

2.4.2 Multiple Model Adaptive Estimation (MMAE)

The MMAE was initially proposed for GPS carrier-phase integer ambiguity resolution (Henderson, P. E., 2001). Models of equations 2.1 and 2.2 contain an unknown parameter which varies slowly when compared to the states. Assume that the parameter can take on values that are discrete members of a finite set $\{\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_K\}$. MMAE uses a bank of, K, linear Kalman filters (Figure 2.2). Each elemental filter is modelled with a different internal hypothesis based on a single member of the set $\{\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_K\}$. The Kalman filters are provided with the same measurement vector $\mathbf{z}(k)$ and produce its own, hypothesis based, state estimate $\hat{\mathbf{x}}(k)$ and innovation sequence $\nu(k)$. Due to the fact that each elemental filter is based on different internal model, filters estimates and innovations will be different and the magnitude of the innovation provides an indication of how adequately each of the models represent the true operation mode. A hypothesis conditional probability weighting computer makes use of each individual innovation sequence to assign relative probabilities to

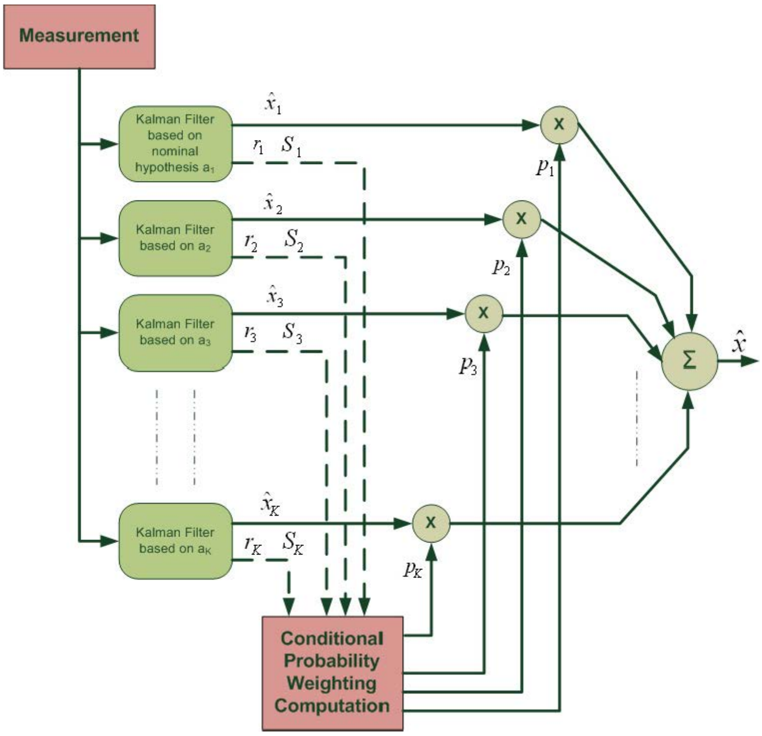

Measurement
Kalman Filter based on nominal hypothesis a1
Kalman Filter based on a2
Kalman Filter based on a3
Kalman Filter based on aK
$\hat{x}_1$
r_1 S_1
$\hat{x}_2$
r_2 S_2
$\hat{x}_3$
r_3 S_3
$\hat{x}_K$
r_K S_K
X
p_1
p_2
p_3
p_K
Σ
$\hat{x}$
Conditional Probability Weighting Computation

Figure 2.2: In MMAE, a bank of Linear Kalman filters is used to detect faults and to provide a state estimate

each filter estimate. The output estimates of all filters are combined optimally using these probabilistic average weights to produce one state estimate. Optimal weights can be used also to estimate the correct parameter $\mathbf{a}_i$, where $i \in \{1, 2, ..., K\}$.

When implemented in fault detection and identification mode, each elemental filter represents a failure model and its corresponding weight is the posteriori probability of that failure. This probability is used as *failure indicator* because it reflects the likelihood that the observed set of measurements correspond to that particular failure model.

As mentioned earlier in subsection 2.3.1, the innovation sequence is Gaussian with zero mean and variance $\mathbf{S}(k)$. Therefore, the Gaussian density function is used to describe the likelihood functions. The objective of MMAE is to get the joint conditional probability density function (PDF) for the state vector $\mathbf{x}(k)$ and parameter set $\mathbf{a}$ conditioned on the realized measurement history $\mathbf{Z}^k$ which is written, according to Baye's theorem (Bertsekas, D. P., & Tsitsiklis, J. N., 2000), as

$$f_{\mathbf{x}(k),\mathbf{a}|\mathbf{Z}(k)}(\xi,\alpha|\mathbf{Z}(k)=\mathbf{Z}^k) = f_{\mathbf{x}(k)|\mathbf{a},\mathbf{Z}(k)}(\xi|\mathbf{a}=\alpha,\mathbf{Z}(k)=\mathbf{Z}^k)f_{\mathbf{a}|\mathbf{Z}(k)}(\alpha|\mathbf{Z}(k)=\mathbf{Z}^k) \tag{2.30}$$

The first term on the right side of equation 2.30 is the conditional density of having state estimate $\mathbf{x}(k)$ given Kalman filter model of $\mathbf{a}$ and measurements history $\mathbf{Z}^k$, which is analogous to the standard conditional density used in Kalman filtering. Therefore, it is Gaussian with mean $\hat{\mathbf{x}}(k|k)$ and variance $\mathbf{P}(k|k)$. The second expression of equation 2.30 is the probability density of hypothesis $\mathbf{a}$ conditioned on observation history up to $\mathbf{Z}^k$, which also Gaussian. It can be expressed as in (Ormsby, C. D., 2003)

$$p_i(k) \equiv \mathit{prob}\left\{\mathbf{a}=\mathbf{a}_i|\mathbf{Z}(k)=\mathbf{Z}^k\right\} \tag{2.31}$$

$$p_i(k) = \frac{f_{\mathbf{z(k)}|\mathbf{a},\mathbf{Z}(k-1)}(\mathbf{z(k)}|\mathbf{a}=\mathbf{a}_i,\mathbf{Z}(k-1)=\mathbf{Z}^{k-1})p_i(k-1)}{\sum_{j=1}^{K} f_{\mathbf{z(k)}|\mathbf{a},\mathbf{Z}(k-1)}(\mathbf{z(k)}|\mathbf{a}=\mathbf{a}_j,\mathbf{Z}(k-1)=\mathbf{Z}^{k-1})p_j(k-1)} \tag{2.32}$$

where

$$\sum_{i=1}^{K} p_i(k) = 1 \tag{2.33}$$

$$p_i(0) = \frac{1}{K} \quad \forall i \tag{2.34}$$

The conditional density function in equation 5.27 is the conditional density of the measurements $\mathbf{z}$ at time k given the Kalman filter model of $\mathbf{a} \in \{\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_K\}$ and the measurements history $\mathbf{Z}^{k-1}$ represented as

$$f_{\mathbf{z(k)}|\mathbf{a},\mathbf{Z}(k-1)}(\mathbf{z(k)}|\mathbf{a}=\mathbf{a}_i,\mathbf{Z}(k-1)=\mathbf{Z}^{k-1}) = \beta_i(k)\exp\left\{-\alpha\nu_i^T(k)\mathbf{S}_i^{-1}(k)\nu_i(k)\right\} \tag{2.35}$$

$$\beta_i(k) = \frac{1}{(2\pi)^{\frac{m}{2}}\left|\mathbf{S}_i(k)\right|^{\frac{1}{2}}} \tag{2.36}$$

where $|\mathbf{S}_i(k)|$ is the determinant of the innovation covariance matrix, m is the dimension of the measurement vector $\mathbf{z}(k)$ and α is a scaling penalty ($\frac{1}{2}$ mostly).

The optimal state estimate is derived as (Ormsby, C. D., 2003)

$$\hat{\mathbf{x}}(k|k) = \sum_{i=1}^{K} \hat{\mathbf{x}}_i(k|k) p_i(k) \tag{2.37}$$

Therefore, the optimal state estimate of a MMAE is the weighted sum of the elemental filter estimates, weighted by the probability that the hypothesis of that particular filter in correct and based on the current measurements.

Equation 2.35 provides a powerful tool for fault identification and isolation. As indicated earlier in subsection 2.3.1, the innovation alone is not enough to indicate the performance of the filter. The innovation covariance provides a valuable information of how accurate the innovation sequence behaves (that is for a filter to be consistent, the innovation covariance should match the error in the innovation) and this motivates the use of the normalized innovation squared computed by equation 2.15 in equation 2.35. For the elemental filter that provides the correct hypothesis, the value of $q(k)$ is the smallest among all other elemental filters. It's probability $p_i(k)$ becomes the largest and it's contribution in the overall state estimation, given by 2.37, is increased. Contributions from other elemental filters are *isolated* based on how far their hypothesis are from the correct (highest weighted) one.

The covariance of this estimate is also derived in (Ormsby, C. D., 2003) and is given by

$$\begin{aligned} \mathbf{P}(k|k) &= E\left\{ \left[\mathbf{x}(k) - \hat{\mathbf{x}}(k)\right]\left[\mathbf{x}(k) - \hat{\mathbf{x}}(k)\right]^T | \mathbf{Z}(k) = \mathbf{Z}^k \right\} \\ &= \sum_{i=1}^{K} p_i(k) \left\{ \mathbf{P}_i(k|k) + \left[\hat{\mathbf{x}}_i(k) - \hat{\mathbf{x}}(k)\right]\left[\hat{\mathbf{x}}_i(k) - \hat{\mathbf{x}}(k)\right]^T \right\} \end{aligned} \tag{2.38}$$

Equation 2.38 indicates that the uncertainty in MMAE state estimation depends on the overall state estimate $\hat{\mathbf{x}}(k)$, each elemental filter estimate $\hat{\mathbf{x}}_i(k)$, each elemental filter uncertainty $\mathbf{P}_i(k)$, and the likelihood $p_i(k)$ that each elemental filter provides the correct state estimate provided a particular hypothesis. The MMAE state estimate error covariance is not required for an online MMAE.

The optimal hypothesis parameter $\hat{\mathbf{a}}(k)$ along with its covariance $\mathbf{P}_{\mathbf{a}}(k)$ may also be computed (Ormsby, C. D., 2003)

$$\hat{\mathbf{a}}(k) = \sum_{i=1}^{K} \mathbf{a}_i(k) p_i(k) \tag{2.39}$$

$$\mathbf{P}_{\mathbf{a}}(k) = \sum_{i=1}^{K} p_i(k) \left\{ \left[\mathbf{a}_i(k) - \hat{\mathbf{a}}^T(k)\right] \left[\mathbf{a}_i(k) - \hat{\mathbf{a}}^T(k)\right]^T \right\} \tag{2.40}$$

In (Maybeck, P. S. & Hanlon, P. D., 1995) performance enhancements were proposed. They include stripping the β term of equation 2.35, scalar penalty increase, probability smoothing, increased residual propagation, Kalman filter tuning and bounded conditional probabilities. Only the last three, which are residuals propagation, filter tuning and bounded probabilities, are utilized in this work, particulary in the implementation of the sequential algorithm discussed in chapter 5.

The number of elemental filters (hypothesis) in the bank and the size of the bank depend on the number of expected failures and the number of hypothesis parameters. For example, if 5 failure modes are expected with each hypothesis formed by varying 2 parameters, then 32 elemental filters are required. To avoid the potentially large number of elemental filters needed for an MMAE bank, a moving bank MMAE is implemented (Maybeck, P. S. & Vasquez J. R., 1999), (Maybeck, P. S. & Vasquez J. R., 2004). In the moving-bank implementation, the number of elemental filters in the bank is fixed and predetermined by the designer. In the previous example, one might choose to build the system with 3 filters per parameter which reduces the number of filters in the bank to 9.

2.4.3 Artificial Intelligence Algorithms

Artificial intelligent algorithms utilized for Kalman filter adaptive tuning include fuzzy logic, Genetic Algorithms (GA), and Neural Networks. In implementations where fuzzy logic is used, such as (Loebis, D., Sutton, R., Chudley, J., & Naeem, W., 2004) and (Oshman, Y., & Shaviv, I., 2000), the covariance matching approach (CMA) is utilized to estimate a value for the sensor noise covariance $\mathbf{R}$, this value is then compared to a membership function and is adjusted such that the theoretical value becomes equal to the actual value.

Genetic Algorithms are optimization algorithms based on the mechanics of natural selection and natural genetics. In (Oshman, Y., & Shaviv, I., 2000) Genetic Algorithms (GA) were used for filter tuning. In doing so, different hypotheses of covariance matrices $\mathbf{Q}$ or $\mathbf{R}$ are selected as first generation individuals. Statistical Consistency Tests (SCT), such as the χ^2, are then utilized to form fitness functions used to evaluate and join individuals to form the second generation. This process involves three different operations, namely, reproduction, crossover, and mutation. The next generation is created after applying all three operators. The same procedure is applied to following generations and inconsistent individuals are removed, the remaining consistent hypotheses form individuals of new population that are joint again via the three operators to form the following generation. This process is repeated until the most consistent (survived) filter hypothesis is selected as the optimal filer. GAs are ideal for optimization problem on hand because they are very efficient when a large search space is involved and they work well with stochastic object function. Algorithms similar to GAs employed in optimization of the tuning process include particle swarm optimization (PSO) (Jwo, D. J., & Chang, S. C., 2007).

Neural networks can be trained to map vehicle dynamics with corresponding Kalman filter states, at the same rate of measurements update. After the output of the network meets a predetermined similarity threshold, it can be used to correct INS measurements and estimate the value of the sensor noise measurement even with no GPS measurements available. Neural networks are rarely employed in filter tuning. In fact thy are mainly employed to bridge GPS outages and replace the GPS for short periods of time. Therefore, neural networks are mainly utilized in fault detection and identification applications, similar to that discussed in chapter 5. Examples of neural networks applications in Kalman filter can be found in (Wang, J. J., Wang, J., Sinclair, D., & Watts, L., 2006), (Wang, J. J., Wang, J., Sinclair, D., & Watts, L., 1997), (Chiang, K. W., Noureldin, A., & El-Sheimy, N., 2004), and (Semeniuk, L., & Noureldin, A., 2006).

2.5 Conclusions

This chapter has developed the required basic background that is essential to achieve this research goals. In doing so, this chapter has:

- Listed the Kalman filter equations and discussed critical implementations issues. Discussed the Unscented Kalman Filter (UKF) as being the state of art in sensor fusion for navigation using the non-linear model for both the propagation of mean and covariance. Although the UKF is not utilized in this work, it is a hot subject for future work.

- Developed a stochastic framework for loosely coupled INS/GPS navigation. It has done that by discussing the topics of filter consistency and tuning.

- Provided a detailed methodology for a simplified tuning procedure based on the fact that under normal conditions the innovations should be zero mean, white, uncorrelated and the corresponding moving average of the normalized innovations squared (NIS) is χ^2 distributed with a given order.

- Discussed the topics of filer initialization and steady state performance. It was indicated that filter initialization affects the convergence time of the filter, while the steady state performance depends mainly on the noise covariances, **Q** and **R**.

- Introduced the basics of adaptive Kalman filter implementations as sophisticated and autonomous methods for filter tuning. These methods included the covariance matching approach (CMA), the multiple model adaptive estimation (MMAE), and artificial intelligent (AI) algorithms.

In fact, this chapter has built the essential knowledge required for next chapters and necessary to achieve our goal, that is to implement an autonomous INS/GPS navigation system that achieves integrity.

Chapter 3

Strapdown Inertial Navigation

3.1 Introduction

Before Newton formulated his mechanics (Halliday, D., Resnick, R., & Walker, J., 1997), it was thought that the *natural state* of a body is when it was at rest and that while a force was needed to move the body, keeping it moving at constant velocity required maintaining the effect of the force. Otherwise, it would *naturally* stop. Inertia is the natural tendency of bodies to maintain constant translational and rotational velocity, unless disturbed by forces or torques, respectively. An inertial reference frame is a coordinate frame that is neither rotating nor translating and, therefore, at which Newton's laws of motion are valid. Inertial sensors, such as accelerometers and gyroscopes, measure acceleration and rotational rate, respectively, with respect to the inertial reference frame. Accelerations measured by three (not-necessary orthogonal) accelerometers, fixed such that their sensitive axes are oriented with one along vehicle's longitudinal axis, one transverse to the vehicle longitudinal axis, and one vertical (Kelly, A, 1994), are integrated once to compute vehicle's velocity. Then, the computed velocity is then integrated to compute vehicle's position. Gyroscopes are utilized for coordinate system transformations, attitude representation, and removal of undesired components such as the gravity component measured by the accelerometers and misalignments. The *mechanization* equations developed in this chapter, along with their error equations, are employed to free measured accelerations from undesired components and calculate vehicle's position, velocity and attitude. These equation are derived in details in appendix C. Figure 3.1 shows the mechanization process utilized to implement a strapdown inertial navigation system (SDINS).

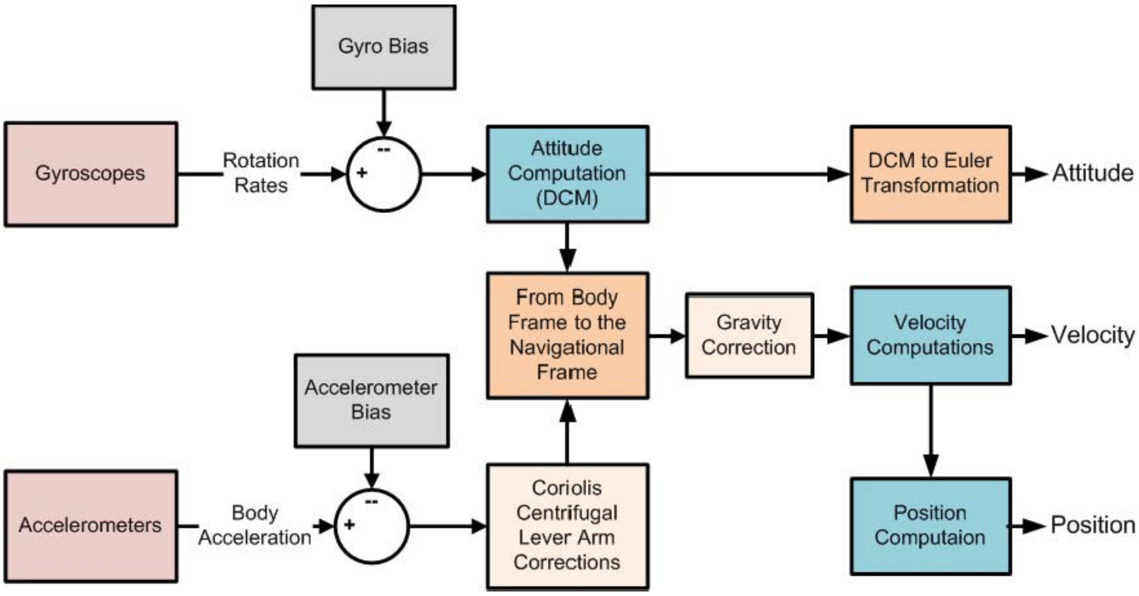


Figure 3.1: Typical Inertial Navigation System (INS) using the Direction Cosine Matrix (DCM) approach utilized in this thesis

In this chapter, SDINS aspects that are most related to our objectives are discussed. Mechanization equations and computational algorithms are discussed in section 3.2. Error models, representations and characteristics are presented in section 3.3. Finally, performance verification and conclusions are listed in section 3.5.

3.2 Computation Algorithms

The transformation of body frame accelerations into the mechanization frame and the removal of the gravitational component are the main purpose of attitude computations. To calculate the velocity and position of the vehicle, accelerations are integrated one to compute the velocity and then once again to compute the position. This section introduces the mechanization equations implemented for velocity and position calculation (Titterton, D., & Weston, W., 2004),(Savage, P. G., 2006),(George, M. G., 2007),(Ahmed, M. S., & Cuk, D. V., 2005). Two different mechanization frames are considered, the navigational frame and the ECEF frame mechanizations. However, more attention is paid on navigational frame mechanization, since it is the frame considered in this work. An up-to-date strapdown mathematical frame is utilized, where velocity / position equations are constructed using a Jordan-like attitude updating approach (Savage, P. G., 2006). It generates an exact solution when provided with exact position / velocity translation vector inputs.

3.2.1 Navigation Frame Mechanization

Attitude Computations:

Attitude computation is the most critical part in the whole INS mechanization process, specially in high dynamic applications. The ability of the strapdown algorithm to keep track of body attitude accurately determines its performance (Titterton, D., & Weston, W., 2004). The conventional approach for attitude computation is to compute the direction cosine matrix, relating the vehicle body reference frame to the reference coordinate system. When navigating with respect to the local geographic frame, it is required to solve the matrix differential equation of the form

$$\dot{\mathbf{C}}_b^n = \mathbf{C}_b^n \boldsymbol{\Omega}_{ib}^b - \boldsymbol{\Omega}_{in}^n \mathbf{C}_b^n \tag{3.1}$$

The first term of equation 3.1, $\mathbf{C}_b^n \boldsymbol{\Omega}_{ib}^b$, is required to update the DCM for vehicle body motion. It is a function of the body rates sensed by the strapdown gyroscopes. While the second term, $-\boldsymbol{\Omega}_{in}^n \mathbf{C}_b^n$, takes into account updating the DCM for navigation frame rotations. This includes the rotation of the navigational frame with respect to the earth frame, which is referred to as the transport rate, and the Earth's rate of rotation. The computation of equation 3.1 is carried out using two steps of direction cosine rotations, the first takes care of body rotations and the second takes care of local navigation frame rotations. Since local navigation frame rotates at much lower rates than the body rotations, it can be implemented at the lower rate l computer cycle (Ahmed, M. S., & Cuk, D. V., 2005),(Titterton, D., & Weston, W., 2004). The two rotation are described by equations 3.2 and 3.3

$$\mathbf{C}_{b(k+1)}^{n(l)} = \mathbf{C}_{b(k)}^{n(l)} \mathbf{C}_{b(k+1)}^{b(k)} \tag{3.2}$$

$$\mathbf{C}_{b(k+1)}^{n(l+1)} = \mathbf{C}_{n(l)}^{n(l+1)} \mathbf{C}_{b(k+1)}^{n(l)} \tag{3.3}$$

where

$\mathbf{C}_{b(k+1)}^{b(k)}$ the DCM which transforms vectors from body frame coordinates at the $k+1$ cycle to body frame coordinates at the k cycle

$\mathbf{C}_{b(k)}^{n(l)}$ the DCM which transforms vectors from body frame coordinates at the k cycle to navigation frame coordinates at the l cycle

$\mathbf{C}_{b(k+1)}^{n(l)}$ the DCM which transforms vectors from body frame coordinates at the $k+1$ cycle to navigation frame coordinates at the l cycle

$\mathbf{C}_{n(l)}^{n(l+1)}$ the DCM which transforms vectors from navigation frame coordinates at the l cycle to navigation frame coordinates at the $l+1$ cycle

$\mathbf{C}_{b(k+1)}^{n(l+1)}$ the DCM which transforms vectors from body frame coordinates at the $k+1$ cycle to navigation frame coordinates at the $l+1$ cycle

In equations 3.2 and 3.3, the terms $\mathbf{C}_{b(k+1)}^{b(k)}$ and $\mathbf{C}_{n(l)}^{n(l+1)}$ update the DCM for the rotations of body and navigation frames respectively. To compute $\mathbf{C}_{b(k+1)}^{b(k)}$, equation 3.4 is used (Titterton, D., & Weston, W., 2004)

$$\mathbf{C}_{b(k+1)}^{b(k)} = \mathbf{I} + f_1(\bar{\sigma})\,[\sigma\times] + f_2(\bar{\sigma})\,[\sigma\times]^2 \tag{3.4}$$

$$f_1(\bar{\sigma}) = \frac{\sin\bar{\sigma}}{\bar{\sigma}} \tag{3.5}$$

$$f_2(\bar{\sigma}) = \frac{1-\cos\bar{\sigma}}{\bar{\sigma}^2} \tag{3.6}$$

$$\sigma = \int_{t_k}^{t_{k+1}} \omega_{ib}^{b} dt = \alpha_{k+1} + \delta\alpha_{k+1} \tag{3.7}$$

where σ is a rotation vector with direction and magnitude such that a rotation of the body frame about σ, through an angle equal to the magnitude of σ and assuming that that direction of the rate-vector ω_{ib}^{b} is unchanging, will rotate the body frame from its orientation at the computer cycle k to its position at the computer cycle $k+1$. This angular vector is computed by equations 3.8-3.10

$$\sigma = \alpha_{k+1} + \delta\alpha_{k+1} \tag{3.8}$$

$$\alpha = \int_{t_k}^{t} \omega_{ib}^{b} dt \tag{3.9}$$

$$\delta\alpha_{k+1} = \int_{t_k}^{t_{k+1}} \alpha \times \omega_{ib}^{b} dt \tag{3.10}$$

Equations 3.7-3.10 can be rewritten as in 3.11

$$\begin{aligned} \sigma_x &= \omega_{ibx}^{b}(t_{k+1} - t_k) + \frac{1}{2}\omega_{ibx}^{b\,2}(t_{k+1} - t_k)^2 \\ \sigma_y &= \omega_{iby}^{b}(t_{k+1} - t_k) + \frac{1}{2}\omega_{iby}^{b\,2}(t_{k+1} - t_k)^2 \\ \sigma_z &= \omega_{ibz}^{b}(t_{k+1} - t_k) + \frac{1}{2}\omega_{ibz}^{b\,2}(t_{k+1} - t_k)^2 \end{aligned} \tag{3.11}$$

where $(\sigma_x, \sigma_y, \sigma_z)$ are the components of σ and $\bar{\sigma}$ is its magnitude, which is given in equation 3.12

$$\bar{\sigma} = \sqrt{\sigma_x^2 + \sigma_y^2 + \sigma_z^2} \tag{3.12}$$

Finally, the skew matrix $[\sigma\times]$ in 3.4 is computed by equation 3.13

$$[\sigma\times] = \begin{pmatrix} 0 & -\sigma_z & \sigma_y \\ \sigma_z & 0 & -\sigma_x \\ -\sigma_y & \sigma_x & 0 \end{pmatrix} \tag{3.13}$$

Similar algorithm is used to compute $\mathbf{C}_{n(l)}^{n(l+1)}$. Equation 3.4 is replaced by equation 3.14

$$\mathbf{C}_{n(l)}^{n(l+1)} = \mathbf{I} + \frac{\sin\bar{\theta}}{\bar{\theta}}\,[\theta\times] + \frac{1-\cos\bar{\theta}}{\bar{\theta}^2}\,[\theta\times]^2 \tag{3.14}$$

where θ is an angle vector with direction and magnitude such that a rotation of the navigation frame about θ, through and angle equal to the magnitude of θ, will rotate the navigation frame from its orientation at the computer cycle l to its position at the computer cycle $l+1$. θ is computed by 3.15

$$\theta = \int_{t_l}^{t_{l+1}} \omega_{in}^{n} dt \tag{3.15}$$

An alternative approach carries the computation of both rotations at the moderate $k - cycle$ computer rate. In this approach equation 3.1 is replace by the differential equation given in 3.16

$$\dot{\mathbf{C}}_b^n = \mathbf{C}_b^n \mathbf{\Omega}_{nb}^b \tag{3.16}$$

where $\mathbf{\Omega}_{nb}^b$ is symmetric skew matrix of ω_{nb}^b, and ω_{nb}^b is the rotation rate of the body frame with respect to the navigation frame measured in the body frame, which is given by

$$\omega_{nb}^b = \omega_{ib}^b - \mathbf{C}_n^b \left(\omega_{ie}^n + \omega_{en}^n\right) \tag{3.17}$$

To solve equation 3.16, equations 3.4 - 3.13 are used. However, the rotation rate ω_{ib}^b must be replaced by ω_{nb}^b which is the relative angular rate of the b frame relative to n frame and it is given by

$$\omega_{nb}^b = \begin{pmatrix} \dot{\phi} - \dot{\psi}\sin\theta \\ \dot{\theta}\cos\phi + \dot{\psi}\cos\theta\sin\phi \\ \dot{\psi}\cos\theta\cos\phi - \dot{\theta}\sin\phi \end{pmatrix} \tag{3.18}$$

Velocity Computations:

The navigation equation expressed in the n frame is derived in appendix C and it is as follows

$$\dot{\boldsymbol{V}}_e^n = \boldsymbol{C}_b^n \boldsymbol{f}^b - \left[2\mathbf{\Omega_{ie}^n} + \mathbf{\Omega_{en}^n}\right] \boldsymbol{V}_e^n + \boldsymbol{g}_l^n \tag{3.19}$$

Where, $\boldsymbol{V}_e^n$ represents velocity with respect to the Earth expressed in the local geographic frame.

$$\boldsymbol{V}_e^n = \begin{pmatrix} \upsilon_N & \upsilon_E & \upsilon_D \end{pmatrix}^T \tag{3.20}$$

$\boldsymbol{f}^b$ is the specific force vector as measured by three orthogonal accelerometers and resolved into the local geographic reference frame by $\boldsymbol{C}_b^n$.

$$\boldsymbol{f}^n = \boldsymbol{C}_b^n \boldsymbol{f}^b = \begin{pmatrix} f_N & f_E & f_D \end{pmatrix}^T \tag{3.21}$$

ω_{en}^n is the transport rate; the turn rate of the navigation frame w.r.t the ECEF frame, and it is calculated by equations 3.22 and 3.23

$$\omega_{en}^n = \begin{pmatrix} \dot{\lambda}\cos\varphi \\ -\dot{\varphi} \\ -\dot{\lambda}\sin\varphi \end{pmatrix} \tag{3.22}$$

$$\begin{aligned} \dot{\varphi} &= \frac{1}{(R_m + h)} v_N \\ \dot{\lambda} &= \frac{1}{(R_n + h)\cos\varphi} v_E \\ \dot{h} &= -V_D \end{aligned} \tag{3.23}$$

R_m and R_n are the meridian radius and curvature and prime (transverse) radius of curvature respectively. With these two parameters, the Earth is modelled as a reference ellipsoid which approximates more closely to the true geometry. They are computed by equations

$$R_m = \frac{a(1 - e^2)}{(1 - e^2\sin^2\varphi)^{3/2}} \tag{3.24}$$

$$R_n = \frac{a}{(1 - e^2\sin^2\varphi)^{1/2}} \tag{3.25}$$

where, e is the major eccentricity of Earth ($= 0.0818191908426$) and a is the equatorial radius of the Earth or semimajor axis ($= 6378137.000$ m). ω_{ie}^n is the turn rate of the earth expressed in the navigation frame

$$\omega_{ie}^{n} = \boldsymbol{C}_{e}^{n}\omega_{ie}^{e} = \begin{pmatrix} -\sin\varphi\cos\lambda & -\sin\varphi\sin\lambda & \cos\varphi \\ -\sin\lambda & \cos\lambda & 0 \\ -\cos\varphi\cos\lambda & -\cos\varphi\sin\lambda & -\sin\varphi \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ \Omega \end{pmatrix} = \begin{pmatrix} \Omega\cos\varphi \\ 0 \\ -\Omega\sin\varphi \end{pmatrix} \tag{3.26}$$

and Ω is the Earth turn rate with respect to i frame ($= 7.292115167 \times 10^{-5}$ rad/s). The term $\boldsymbol{g}_{l}^{n}$ combines accelerations due to gravity and centripetal acceleration. It is given by equation 3.28

$$\boldsymbol{g}_{l}^{n} = \boldsymbol{g}^{n} - \omega_{ie}^{n} \times \left(\omega_{ie}^{n} \times \boldsymbol{r}_{veh}^{ned}\right) \tag{3.27}$$

$$= \boldsymbol{g}^{n} - \frac{\Omega^{2}\left(R_{n} + h\right)}{2} \begin{pmatrix} \sin 2\varphi \\ 0 \\ 1 + \cos 2\varphi \end{pmatrix} \tag{3.28}$$

$\boldsymbol{g}^{n}$ represents the Earth's gravitational field which varies with changes in the position of the navigation system on the Earth or its height above the its surface. An international model for the variation of gravity with latitude and height is given by equations 3.29-3.30

$$\boldsymbol{g}^{n} = \begin{pmatrix} 0 & 0 & g(h) \end{pmatrix}^{T} \tag{3.29}$$

$$g(h) = \frac{g(0)}{\left(1 + h/a\right)^{2}} \tag{3.30}$$

$$g(0) = 9.780318\left(1 + 5.3024 \times 10^{-3}\sin^{2}\varphi - 5.9 \times 10^{-6}\sin^{2}2\varphi\right) m/s^{2} \tag{3.31}$$

For more details on Earth's shape modelling and local gravity derivation refer to appendix C.

Discretization: Equation 3.19 can be discretized in different forms. A simple and straight-forward method, which utilizes rectangular integrations, is implemented in equation 3.32.

$$\begin{aligned}\boldsymbol{V}^n(k+1) =& \ \boldsymbol{V}^n(k) + \Delta\boldsymbol{V}^n - \left[\left(\boldsymbol{\Omega}^n_{en} + 2\boldsymbol{\Omega}^n_{ie}\right)\boldsymbol{V}^n(k) - \boldsymbol{g}^n_l\right](t_{k+1} - t_k) \\ =& \ \boldsymbol{V}^n(k) + \boldsymbol{C}^n_b\boldsymbol{f}^b - \left[\left(\boldsymbol{\Omega}^n_{en} + 2\boldsymbol{\Omega}^n_{ie}\right)\boldsymbol{V}^n(k) - \boldsymbol{g}^n_l\right](t_{k+1} - t_k)\end{aligned} \tag{3.32}$$

However, the unified mathematical framework proposed by (Savage, P. G., 2006) is implemented instead. This algorithm is represented by equations 3.33-3.37

$$\boldsymbol{V}^n(k+1) = \ \boldsymbol{V}^n(k) + \boldsymbol{C}^n_b(k)\Delta\boldsymbol{V}^n_{sf}(k+1) + \Delta\boldsymbol{V}^n_g(k+1) \tag{3.33}$$

where $\Delta\boldsymbol{V}^n_{sf}(k+1)$ is the change in velocity due to specific force acceleration

$$\Delta\boldsymbol{V}^n_{sf}(k+1) = \left[\mathbf{I} + f_2(\bar{\sigma})\left[\sigma\times\right] + f_3(\bar{\sigma})\left[\sigma\times\right]^2\right]\nu \tag{3.34}$$

Where $f_1(\bar{\sigma})$ and $f_2(\bar{\sigma})$ are as in 3.5 and 3.6, respectively, and

$$f_3(\bar{\sigma}) = \frac{1}{\bar{\sigma}^2}\left(1 - f_1(\bar{\sigma})\right) \tag{3.35}$$

$$\nu = \int_{t_k}^{t_{k+1}} \boldsymbol{f}^b.dt \tag{3.36}$$

$\Delta\boldsymbol{V}^n_g(k+1)$ is the change in velocity due to gravity

$$\Delta\boldsymbol{V}^n_g(k+1) = \int_{t_k}^{t_{k+1}} \boldsymbol{g}^n_l.dt \tag{3.37}$$

Position Computations:

Once the vehicle's velocity is calculated, it's position can also be calculated simply by the following differential equation

$$\dot{\boldsymbol{P}}^n_e = \boldsymbol{V}^n_e \tag{3.38}$$

Discretization: The unified mathematical framework of (Savage, P. G., 2006) is extended to position computation as follows

$$\boldsymbol{P}^n(k+1) = \boldsymbol{P}^n(k) + \boldsymbol{V}^n(k)\Delta t_k + \boldsymbol{C}_b^n(k)\Delta\boldsymbol{P}_{sf}^n(k+1) + \Delta\boldsymbol{P}_g^n(k+1) \tag{3.39}$$

Where, $\Delta\boldsymbol{P}_{sf}^n(k+1)$ is the change in position due to specific force acceleration

$$\Delta\boldsymbol{P}_{sf}^n(k+1) = \left[\mathbf{I} + 2f_3(\bar{\sigma})\,[\sigma\times] + 2f_4(\bar{\sigma})\,[\sigma\times]^2\right]\mathbf{S}_\nu \tag{3.40}$$

Where $f_2(\bar{\sigma})$ and $f_3(\bar{\sigma})$ are as in 3.6 and 3.35, respectively, and

$$f_4(\bar{\sigma}) = \frac{1}{\bar{\sigma}^2}\left(\frac{1}{2} - f_2(\bar{\sigma})\right) \tag{3.41}$$

$$\mathbf{S}_\nu = \int_{t_k}^{t_{k+1}} \nu.dt \tag{3.42}$$

$\Delta\boldsymbol{P}_g^n(k+1)$ is the change in position due to gravity

$$\Delta\boldsymbol{P}_g^n(k+1) = \int_{t_k}^{t_{k+1}} \Delta\boldsymbol{V}_g^n(k+1).dt \tag{3.43}$$

The algorithms used for attitude, velocity and position computations in equations 3.4, 3.33 and 3.39, respectively, hold for the assumption of constant angular rate ω_{ib}^b and constant specific force $\mathbf{f}^b$. For this case of constant inputs, the algorithm input is the direct integral of angular rate/specific force vector components provided by gyroscopes and accelerometers. However, for general motion (non-constant inputs) the algorithms should be modified. Integrated angular rates and specific forces are replaced by rotation and velocity/position translation vectors. For more details on the unified approach in case of general motion refer to (Savage, P. G., 2006).

The computed position $\boldsymbol{P}^n(k+1)$ of equation 3.39 is the position of the vehicle in the local geographic frame. To convert this position into a different coordinate frame, a reference is needed. It is conventional to select the starting point as reference

$$\boldsymbol{r}_{ref}^{llh} = \begin{pmatrix} \varphi & \lambda & h \end{pmatrix}_{ref}^{T} \tag{3.44}$$

where (φ, λ, h) are the geodetic latitude, longitude and altitude respectively. The reference position is then converted to ECEF coordinate frame.

$$\boldsymbol{r}_{ref}^{ecef} = \begin{pmatrix} (R_n + h)\cos\varphi\cos\lambda \\ (R_n + h)\cos\varphi\sin\lambda \\ (R_m + h)\sin\varphi \end{pmatrix} \tag{3.45}$$

where R_m and R_n are as in equations 3.24 and 3.25 respectively. Vehicle displacement is then converted into the ECEF coordinate frame using equation 3.47

$$\boldsymbol{D}_{veh}^{ecef} = \boldsymbol{C}_n^e \boldsymbol{r}_{veh}^{ned} \tag{3.46}$$

$$\boldsymbol{r}_{veh}^{ecef} = \boldsymbol{D}_{veh}^{ecef} + \boldsymbol{r}_{ref}^{ecef} \tag{3.47}$$

where $\boldsymbol{C}_n^e$ is computed by taking the transpose of $\boldsymbol{C}_e^n$ computed using φ, λ and h of the reference $\boldsymbol{r}_{ref}^{llh}$ (appendix A). The position of the vehicle in the ECEF frame is simply the summation of the reference position in the ECEF frame $\boldsymbol{r}_{ref}^{ecef}$ and individual displacement of the vehicle in the ECEF frame $\boldsymbol{D}_{veh}^{ecef}$. To convert this position into the geodetic coordinate, equation 3.45 should be reversed. Unfortunately it cannot be reversed. However, it can be solved numerically for example using the iterative

Newton-Raphson Method (Chapra, S., C., & Canale, R., P.,).

$$\boldsymbol{r}_{veh}^{llh} = \begin{pmatrix} h & \varphi & \lambda \end{pmatrix}_{veh}^{T} \tag{3.48}$$

$$h = \sqrt{r_x^2 + r_y^2 + r_z^2} - a \tag{3.49}$$

$$\varphi = \sin^{-1}\left(\frac{r_z}{a+h}\right) \tag{3.50}$$

$$\lambda = \sin^{-1}\left(\frac{r_y}{(a+h)\cos\varphi}\right) \tag{3.51}$$

3.2.2 Earth-Centered Earth-Fixed Frame Mechanization

An Earth frame mechanisation is often used for autonomous vehicle navigation in applications where a ground stations is used to command the vehicle. In brief, ECEF mechanization is derived using appendix C and similar to the navigation frame mechanization it is summarized as follows

Attitude Computations:

Attitude computation in the ECEF frame is similar to that in the local geographic frame except that an additional transformation to the ECEF frame is required. This transformation is done using the following equation

$$\boldsymbol{C}_b^e = \boldsymbol{C}_n^e \boldsymbol{C}_b^n$$

where $\boldsymbol{C}_n^e$ was given given in equation 3.26

Velocity Computations:

$$\dot{\boldsymbol{V}}_e^e = \boldsymbol{C}_b^e \boldsymbol{f}^b - (\omega_{ie}^e + \omega_{ie}^e) \times \boldsymbol{V}_e^e + \boldsymbol{g}_l^e \tag{3.52}$$

$$= \boldsymbol{C}_b^e \boldsymbol{f}^b - 2\,(\omega_{ie}^e \times \boldsymbol{V}_e^e) + \boldsymbol{g}_l^e \tag{3.53}$$

$$= \boldsymbol{C}_b^e \boldsymbol{f}^b - 2\boldsymbol{\Omega}_{ie}^e \boldsymbol{V}_e^e + \boldsymbol{g}_l^e \tag{3.54}$$

and

$$\omega_{ie}^{e} = \begin{pmatrix} 0 \\ 0 \\ \Omega \end{pmatrix}$$

The term $\boldsymbol{g}_l^e$ combines accelerations due to gravity and centripetal acceleration. It is given by equation 3.56

$$\boldsymbol{g}_l^e = \boldsymbol{g}^e - \omega_{ie}^e \times \left(\omega_{ie}^e \times \boldsymbol{r}_{veh}^{ecef}\right) \tag{3.55}$$

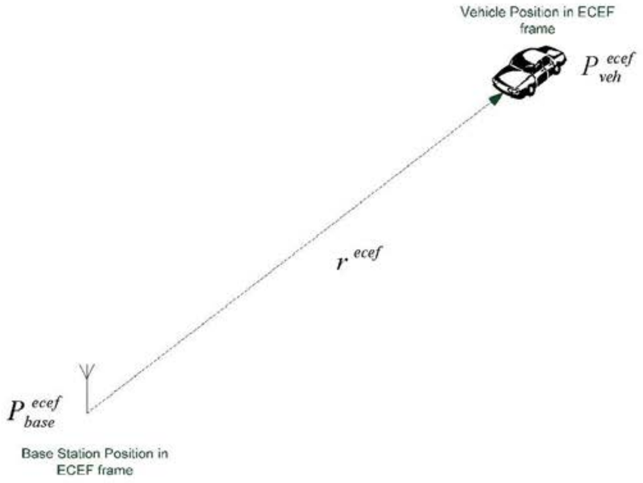
$$= \boldsymbol{g}^e - \Omega^2 \left(R_0 + h\right) \begin{pmatrix} \cos\varphi\cos\lambda \\ \cos\varphi\sin\lambda \\ 0 \end{pmatrix} \tag{3.56}$$

$$\boldsymbol{g}^e = \boldsymbol{g}^n \begin{pmatrix} -\cos\varphi\cos\lambda \\ -\cos\varphi\sin\lambda \\ -\sin\varphi \end{pmatrix} \tag{3.57}$$

The transformation of vectors from the ECEF frame into the local geographic frame is summarized is Figure 3.2 and is done by reversing the procedure described earlier to transform vectors from the local geographic frame into the ECEF frame.

3.3 Error Analysis

We aim to provide an online calibration for the IMU and the INS. The Kalman filter is used to model the error dynamics. The state transition matrix $\mathbf{F}(k)$ in 2.1 models the transition of errors with time and the noise transition matrix $\mathbf{G}(k)$ models the contribution of the noise in the prediction process. The purpose of this section is to explain the nature of errors in the INS estimates and derive a mathematical model for both transition matrices, $\mathbf{F}(k)$ and $\mathbf{G}(k)$.


Vehicle Position in ECEF frame
P^{ecef}_{veh}
r^{ecef}
P^{ecef}_{base}
Base Station Position in ECEF frame

Calculations

$$r^{ecef} = P^{ecef}_{veh} - P^{ecef}_{base}$$

$$P^{ecef}_{veh} = \begin{pmatrix} x_{veh} & y_{veh} & z_{veh} \end{pmatrix}^T$$

$$P^{ecef}_{base} = \begin{pmatrix} x_{base} & y_{base} & z_{base} \end{pmatrix}^T$$

$$P^{llh}_{base} = \begin{pmatrix} \varphi & \lambda & h \end{pmatrix}^T$$

$$C^n_e = \begin{pmatrix} -\sin\varphi\cos\lambda & -\sin\varphi\sin\lambda & \cos\varphi \\ -\sin\lambda & \cos\lambda & 0 \\ -\cos\varphi\cos\lambda & -\cos\varphi\sin\lambda & -\sin\varphi \end{pmatrix}$$

$$r^{ned} = C^n_e r^{ecef}$$

$$P^{ned}_{veh} = r^{ned}$$

$$P^{ned}_{veh} = \begin{pmatrix} n_{veh} & e_{veh} & d_{veh} \end{pmatrix}^T$$

$$P^{ned}_{base} = \begin{pmatrix} 0 & 0 & 0 \end{pmatrix}^T$$

Figure 3.2: Steps to transform components from the ECEF frame over into the local geographic frame

3.3.1 Sensor Errors

For simulation purposes, a generalized sensor error models can be represented as follows (Titterton, D., & Weston, W., 2004)

$$\begin{pmatrix} \delta\omega_x \\ \delta\omega_y \\ \delta\omega_z \end{pmatrix} = \mathbf{B}_G + \mathbf{B}_g \begin{pmatrix} a_x \\ a_y \\ a_z \end{pmatrix} + \mathbf{B}_{ae} \begin{pmatrix} a_y a_z \\ a_z a_x \\ a_x a_y \end{pmatrix} + \mathbf{B}_{ai} \begin{pmatrix} \omega_y \omega_z \\ \omega_z \omega_x \\ \omega_x \omega_y \end{pmatrix} + \mathbf{S}_G \begin{pmatrix} \omega_x \\ \omega_y \\ \omega_z \end{pmatrix} + \mathbf{M}_G \begin{pmatrix} \omega_x \\ \omega_y \\ \omega_z \end{pmatrix} + \mathbf{w}_G \tag{3.58}$$

$$\begin{pmatrix} \delta f_x \\ \delta f_y \\ \delta f_z \end{pmatrix} = \mathbf{B}_A + \mathbf{B}_v \begin{pmatrix} a_y a_z \\ a_z a_x \\ a_x a_y \end{pmatrix} + \mathbf{S}_A \begin{pmatrix} a_x \\ a_y \\ a_z \end{pmatrix} + \mathbf{M}_A \begin{pmatrix} a_x \\ a_y \\ a_z \end{pmatrix} + \mathbf{w}_A \tag{3.59}$$

where

$\delta\omega_x, \delta\omega_y, \delta\omega_z$ represent errors in x, y, and z gyroscopes

$\delta f_x, \delta f_y, \delta f_z$ represent errors in x, y, and z accelerometers

$\mathbf{B}_G$ 3 × 1 vector representing gyroscopes residual fixed biases

$\mathbf{B}_g$ 3 × 3 matrix representing gyroscopes g-dependent bias coefficients

$\mathbf{B}_{ae}$ 3 × 3 matrix representing gyroscopes anisoelastic coefficients

$\mathbf{B}_{ai}$ 3 × 3 matrix representing gyroscopes anisoinertial coefficients

$\mathbf{S}_G$ 3 × 3 diagonal matrix representing gyroscopes scale-factor errors

$\mathbf{M}_G$ 3 × 3 skew matrix representing gyroscopes misalignments and cross-coupling errors

$\mathbf{w}_G$ 3 × 1 vector representing gyroscopes random bias errors

$\mathbf{B}_A$ 3 × 1 vector representing accelerometers residual fixed biases

$\mathbf{B}_v$ 3 × 3 matrix representing accelerometers vibro-pendulous error coefficients

$\mathbf{S}_A$ 3 × 3 diagonal matrix representing accelerometers scale-factor errors

$\mathbf{M}_A$ 3×3 skew matrix representing accelerometers misalignments and cross-coupling errors

$\mathbf{w}_A$ 3 × 1 vector representing accelerometers random bias errors

However, terms such as anisoelastic, anisoinertial, and vibro-pendolous errors mainly affect mechanical inertial sensors, and, therefore, they are neglected. It is important also to understand the nature of employed sensors, since such understanding may simplify sensors error modelling. For example, if optical sensors such as ring laser and fiber optic gyros were utilized, then all acceleration dependent error terms is gyros measurements are usually insignificant and then can be neglected.

A spectral density - based sensor errors model that can be used in discrete-time computer simulations is derived in (Crassidis, J., L., 2005) and is represented as follows

$$\mathbf{b}_{k+1} = \mathbf{b}_k + \sigma_u \Delta t^{1/2} \mathbf{N}_u \tag{3.60}$$

$$\hat{\mathbf{m}}_{k+1} = \mathbf{m}_{k+1} + \frac{1}{2}\left[\mathbf{b}_{k+1} + \mathbf{b}_k\right] + \left[\frac{\sigma_v^2}{\Delta t} + \frac{\sigma_u^2 \Delta t}{12}\right]^{1/2} \mathbf{N}_v \tag{3.61}$$

where

$\mathbf{b}_k$ represents inertial sensor bias at time k

$\mathbf{m}_k$ represents noise free inertial sensor measurements at time k

$\hat{\mathbf{m}}_k$ represents simulated inertial sensor measurements at time k

σ_u and σ_v represent rate random walk and angle random walk respectively

$\mathbf{N}_u$ and $\mathbf{N}_v$ represent zero mean normally distributed random variables with variances of one

$\Delta t = t_{k+1} - t_k$ is the IMU sample time

This model neglects scale factor, non-linearity and misalignment errors. In fact, most IMU manufacturers build an embedded system that can compensate for the significant part of these errors such that they become negligible. On last observation on equations 3.60 and 3.61 is that inertial measurements noise and bias estimates are independent of vehicle's position, velocity, and attitude and depends mainly on internal sensors characteristics. Therefore, they are unaffected by external noise such jamming or interference. These are the main advantages of using inertial sensors as dead reckoning systems.

Figure 3.3: The "Crossbow" IMU400CD-100 inertial measurement unit (IMU) utilized in this thesis.

For more practical simulation environment, errors may be extracted from actual sensor measurement. This can be done by fixing the IMU on flat surface and taking sensors measurements for a relatively long period of time. Therefore, in this case sensor measurements contains nothing but real-time sensor errors. Figures 3.4 and 3.5 represent measurements of a stationary "Crossbow" IMU, shown in Figure 3.3, for a period of 7 hours, while the mean and variance of sensors bias are in table 3.1.

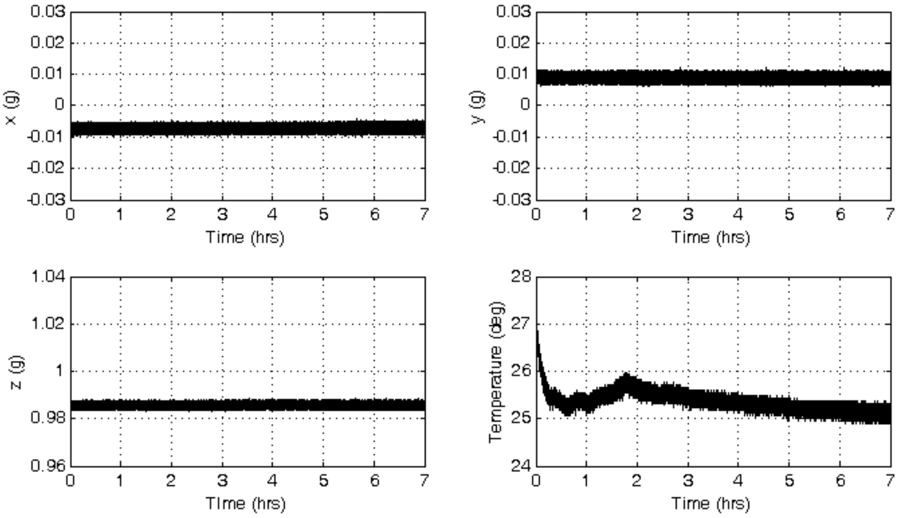


Figure 3.4: Acceleration measurements of three orthogonal accelerometers in "Crossbow" inertial unit. The noise measurements depicts a high bias stability for a period of 7 hours, in spite of the 2 degrees variation in it's internal components temperature.

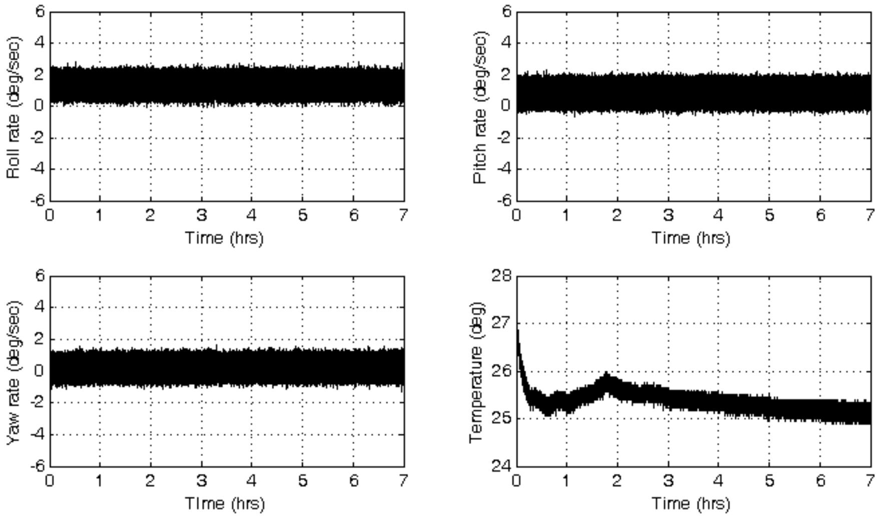


Figure 3.5: Rotation rates measurements of three orthogonal gyroscopes in "Crossbow" inertial unit. The noise measurements depicts a high bias stability for a period of 7 hours, in spite of the 2 degrees variation in it's internal components temperature. However, the gyroscopes show a higher noise level than that of the accelerometers, which is considered as a major source of error in the mechanization process.

Table 3.1: Mean and variance of ”Crossbow” IMU components

Sensor	**Exp. Mean**	**Datasheet Mean**	**Variance**
x-accelerometer	$-7.1mg$		3.1358×10^{-7}
y-accelerometer	$8.9mg$	$< \pm 12mg$	3.7967×10^{-7}
z-accelerometer	$-14.1mg$		3.3861×10^{-7}
Roll rate gyro	$1.3190 deg/sec$		0.0859
Pitch rate gyro	$0.8125 deg/sec$	$< \pm 1 deg/sec$	0.0921
Yaw rate gyro	$0.2096 deg/sec$		0.0792

Other than sensor errors, strapdown inertial navigation systems suffer from a number of errors. In fact, because these errors are either motion dependent or growing with time, inertial systems are unreliable even for relatively short periods. These errors are summarized as follows:

- Initial alignment: Initial attitude errors result into misalignments in the initial direction cosine matrix. During mechanization, all coordinate transformation will have misalignment and the removal of gravity component will cause drifting of the state estimates.

- Maneuvering-dependent: High levels of acceleration and rapid turn rates excite error within the SDINS such as scale factor errors and cross-coupling. Compared to IMU sample rates, higher frequency maneuvers give rise to computational errors within the navigation system.

- Vibration-dependent: Since inertial sensors are fixed to vehicle surface, vibrations results into conning and sculling. This oscillatory motion will be detected by accelerometers. However, gyros have smaller bandwidth and therefore unable to detect small amplitude oscillations. This will result into inaccurate estimation of the component of the gravity and will be inaccurately evaluated as apparent vehicle motion.

- Computational: Attitude computation is the most critical in INS mechanization. The transformations matrices in equation 3.4 and similarly 3.14 contains coefficients ($f_1(\bar{\sigma})$ and $f_2(\bar{\sigma})$ in equation 3.4) that are approximated once implemented in real time. The order of the attitude algorithm depends on the number of Taylor series expansion coefficients included in the computations. Figure 3.6 represents the relationship between the order of the algorithm, the drift rate and the relative magnitude between the sampling rate and the continuous rotation rate of the vehicle. The drift rate is a function of both the sampling time and the continuous rate of rotation. This relation is derived in (Titterton, D., & Weston, W., 2004) and is represented as

$$D_{dc} = \Delta f \left(\bar{\sigma} f_1(\bar{\sigma}) \cos \bar{\sigma} - \sin \bar{\sigma} + \bar{\sigma}^2 f_2(\bar{\sigma}) \sin \bar{\sigma} \right) \tag{3.62}$$

where Δf is the sampling frequency

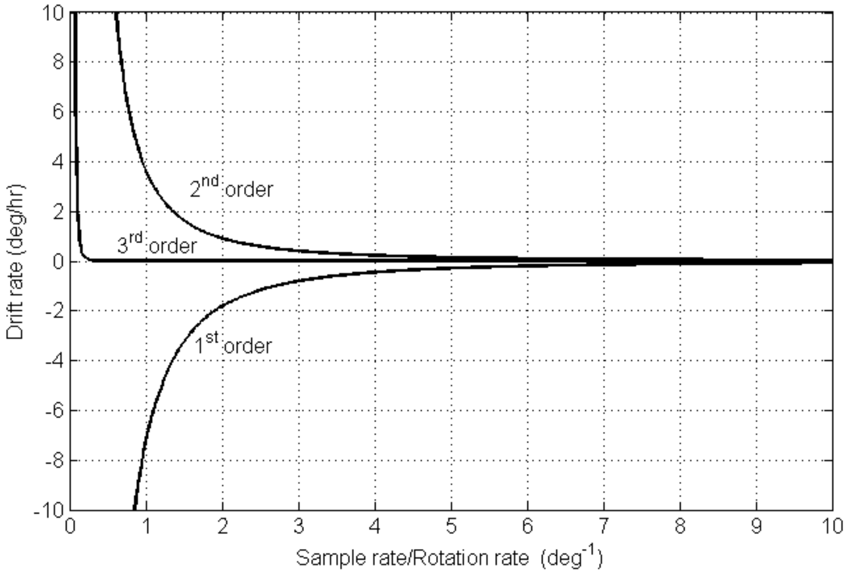


Figure 3.6: Attitude computation drift rates in $^o/hr$ caused by sampling a continuous rotation rates of $20^o/sec$. The drift errors caused by reduced order computational algorithm are kept small as long as the sampling rate is higher magnitude than vehicle's continuous rotation rate. It is also clear that the drift rate is significantly improved by increasing the order of the algorithm rather than by increasing the sampling rate.

As sensors measurements are integrated with time, errors will be integrated too and the inertial solution will drift over time resulting into low-frequency faults in the navigation system (Sukkarieh, S., Nebot, E. M., & Durrant-Whyte, H. F., 1999). In the next section, position, velocity and attitude error equations are derived and the state space model of inertial system state estimates are listed. The effect of sensor measurements errors on system states will be also discussed.

3.3.2 Modelling of Error States

Error models are developed by perturbing the nominal differential equation whose output yields the INS output states (Titterton, D., & Weston, W., 2004)(Sukkarieh, S., 2000)(Kong, X., 2000) (Kim, J., 2004).

Using The Local Geographic Reference Frame as The Computer Frame:

As described in section 3.2.1, the orientation of the vehicle is expressed in terms of the direction cosine matrix $\boldsymbol{C}_b^n$. Due to the misalignment of the reference frame stored in the INS computer, the estimated DCM, denoted by $\tilde{\boldsymbol{C}}_b^n$, may be written in terms of the true direction cosine matrix, $\boldsymbol{C}_b^n$, and attitude errors by taking the transpose of the matrix in A.5 and replacing the angles (ϕ, θ, ψ) by their misalignments (ψ_x, ψ_y, ψ_z):

$$\begin{aligned}\tilde{\boldsymbol{C}}_b^n &= \begin{pmatrix} 1 & \delta\psi_z & -\delta\psi_y \\ \delta\psi_x\delta\psi_y - \delta\psi_z & \delta\psi_x\delta\psi_y\delta\psi_z + 1 & \delta\psi_x \\ \delta\psi_y + \delta\psi_x\delta\psi_z & \delta\psi_y\delta\psi_z - \delta\psi_x & 1 \end{pmatrix} \boldsymbol{C}_b^n \\ &= \begin{pmatrix} 1 & \delta\psi_z & -\delta\psi_y \\ -\delta\psi_z & 1 & \delta\psi_x \\ \delta\psi_y & -\delta\psi_x & 1 \end{pmatrix} \boldsymbol{C}_b^n \\ &= [\mathbf{I} - [\delta\psi \times]]\,\boldsymbol{C}_b^n \end{aligned} \tag{3.63}$$

where $\sin\delta\theta \approx \theta$, $\cos\delta\theta \approx 1$, the product of error terms is zero and $[\delta\psi\times]$ is a symmetrical skew matrix of attitude errors, which is written as:

$$[\delta\psi\times] = \begin{pmatrix} 0 & -\delta\psi_z & \delta\psi_y \\ \delta\psi_z & 0 & -\delta\psi_x \\ -\delta\psi_y & \delta\psi_x & 0 \end{pmatrix} \tag{3.64}$$

Rearranging equation 3.64 for $[\delta\psi\times]$

$$[\delta\psi\times] = \mathbf{I} - \tilde{\boldsymbol{C}}_b^n \boldsymbol{C}_b^{nT} \tag{3.65}$$

Differentiating equation 3.65 yields

$$[\delta\dot{\psi}\times] = -\dot{\tilde{\boldsymbol{C}}}_b^n \boldsymbol{C}_b^{nT} - \tilde{\boldsymbol{C}}_b^n \dot{\boldsymbol{C}}_b^{nT} \tag{3.66}$$

Similar to equation 3.1, the differential of $\tilde{\boldsymbol{C}}_b^n$ is written as

$$\dot{\tilde{\boldsymbol{C}}}_b^n = \tilde{\boldsymbol{C}}_b^n \tilde{\boldsymbol{\Omega}}_{ib}^b - \tilde{\boldsymbol{\Omega}}_{in}^n \tilde{\boldsymbol{C}}_b^n \tag{3.67}$$

Substituting equations 3.1 and 3.67 into 3.66, using equation 3.63 and arranging yields (Titterton, D., & Weston, W., 2004), which is derived in appendix C

$$\delta\dot{\psi} \approx -\omega_{in}^n \times \delta\psi + \delta\omega_{in}^n - \boldsymbol{C}_b^n \delta\omega_{ib}^b \tag{3.68}$$

where

$$\delta\psi = \begin{pmatrix} \delta\psi_x & \delta\psi_y & \delta\psi_z \end{pmatrix}^T \tag{3.69}$$

is the misalignment vector.

The misalignment vector is employed in equations 3.63 to calculated the updated DCM as follows

$$\begin{aligned} \boldsymbol{C}_b^n &= [\mathbf{I} - [\delta\psi\times]]^{-1}\tilde{\boldsymbol{C}}_b^n \\ &= [\mathbf{I} + [\delta\psi\times]]\tilde{\boldsymbol{C}}_b^n \end{aligned} \tag{3.70}$$

Similarly, velocity error equation may be estimated by perturbation of the nominal equation of 3.19. The resulted error equation is written as

$$\delta\dot{\mathbf{v}}_e^n = [\mathbf{f}^n\times]\delta\psi + \boldsymbol{C}_b^n\delta\mathbf{f}^b - (2\omega_{ie}^n + \omega_{en}^n) \times \delta\mathbf{v}_e^n - (2\delta\omega_{ie}^n + \delta\omega_{en}^n) \times \mathbf{v}_e^n - \delta\mathbf{g}_l^n \tag{3.71}$$

where

$$\delta\dot{\mathbf{v}}_e^n = \dot{\tilde{\mathbf{v}}}_e^n - \dot{\mathbf{v}}_e^n \tag{3.72}$$

$$\delta\mathbf{f}^b = \tilde{\mathbf{f}}^b - \mathbf{f}^b \tag{3.73}$$

$$\delta\mathbf{v}_e^n = \tilde{\mathbf{v}}_e^n - \mathbf{v}_e^n \tag{3.74}$$

$$\delta\omega_{ie}^n = \tilde{\omega}_{ie}^n - \omega_{ie}^n \tag{3.75}$$

$$\delta\omega_{en}^n = \tilde{\omega}_{en}^n - \omega_{en}^n \tag{3.76}$$

$$\delta\mathbf{g}_l^n = \tilde{\mathbf{g}}_l^n - \mathbf{g}_l^n \tag{3.77}$$

If errors in Coriolis and gravity terms are ignored, then equation 3.71 can be written as

$$\delta\dot{\mathbf{v}}_e^n = [\mathbf{f}^n\times]\delta\psi + \boldsymbol{C}_b^n\delta\mathbf{f}^b \tag{3.78}$$

Finally, perturbation of equation 3.38 results in position error equation of

$$\delta\dot{\mathbf{p}}_e^n = \delta\mathbf{v}_e^n \tag{3.79}$$

Thus, the propagation of velocity and position errors are functions of the specific force transformed over to the local geographic reference frame, $\mathbf{f}^n$, the misalignment of the frame, $\delta\psi$, and inaccuracies in the measurements of the accelerometers, $\delta\mathbf{f}^b$, transformed over to the navigation frame via $\boldsymbol{C}_b^n$.

Perturbation of The ECEF Frame Using The True Frame Approach:

Similar to equation 3.63, the misalignment is represented in the skew matrix form $[\delta\phi\times]$ as

$$\tilde{\boldsymbol{C}}_b^e = [\mathbf{I} - [\delta\phi\times]]\,\boldsymbol{C}_b^e \tag{3.80}$$

rearranging equation 3.80 yields

$$[\delta\phi\times] = \mathbf{I} - \tilde{\boldsymbol{C}}_b^e \boldsymbol{C}_b^{eT} \tag{3.81}$$

and differentiating gives

$$[\delta\dot{\phi}\times] = -\tilde{\boldsymbol{C}}_b^e \dot{\boldsymbol{C}}_b^{eT} - \dot{\tilde{\boldsymbol{C}}}_b^e \boldsymbol{C}_b^{eT} \tag{3.82}$$

The propagation of the the true , $\boldsymbol{C}_b^e$, and evaluated, $\tilde{\boldsymbol{C}}_b^e$, transformation matrices is as follows

$$\dot{\boldsymbol{C}}_b^e = \boldsymbol{C}_b^e \boldsymbol{\Omega}_{be}^e \tag{3.83}$$

$$\dot{\tilde{\boldsymbol{C}}}_b^e = \tilde{\boldsymbol{C}}_b^e \boldsymbol{\Omega}_{be}^e \tag{3.84}$$

Substituting into equation 3.82 gives

$$\begin{aligned}[\delta\dot{\phi}\times] &= -\tilde{\boldsymbol{C}}_b^e \,[\boldsymbol{C}_b^e \boldsymbol{\Omega}_{be}^e]^T - \left[\tilde{\boldsymbol{C}}_b^e \boldsymbol{\Omega}_{be}^e\right] \boldsymbol{C}_b^{eT} \\ &= -\tilde{\boldsymbol{C}}_b^e \left[\tilde{\boldsymbol{\Omega}}_{be}^e - \boldsymbol{\Omega}_{be}^e\right] \boldsymbol{C}_b^{eT}\end{aligned} \tag{3.85}$$

Perturbation of the rotation update matrix gives

$$\delta\boldsymbol{\Omega}_{be}^e = \tilde{\boldsymbol{\Omega}}_{be}^e - \boldsymbol{\Omega}_{be}^e \tag{3.86}$$

therefore

$$[\delta\dot{\phi}\times] = -\tilde{\boldsymbol{C}}_b^e \delta\boldsymbol{\Omega}_{be}^e \boldsymbol{C}_b^{eT} \tag{3.87}$$

or rearranging into

$$\tilde{\boldsymbol{C}}_b^e = \frac{-[\delta\dot{\phi}\times]}{\delta\boldsymbol{\Omega}_{be}^e \boldsymbol{C}_b^{eT}} \tag{3.88}$$

substituting equation 3.88 into equation 3.80 yields

$$\begin{aligned} [\delta\dot{\phi}\times] &= -[\mathbf{I} - [\delta\phi\times]]\,\boldsymbol{C}_b^e \delta\boldsymbol{\Omega}_{be}^e \boldsymbol{C}_b^{eT} \\ &= -\boldsymbol{C}_b^e \delta\boldsymbol{\Omega}_{be}^e \boldsymbol{C}_b^{eT} + [\delta\phi\times]\,\boldsymbol{C}_b^e \delta\boldsymbol{\Omega}_{be}^e \boldsymbol{C}_b^{eT} \end{aligned} \tag{3.89}$$

ignoring the product of the terms $\delta\phi\times$ and $\delta\boldsymbol{\Omega}_{be}^e$, then the skew matrix of attitude error becomes

$$\begin{aligned} [\delta\dot{\phi}\times] &= -\boldsymbol{C}_b^e \delta\boldsymbol{\Omega}_{be}^e \boldsymbol{C}_b^{eT} \\ &= -\boldsymbol{C}_b^e \delta\boldsymbol{\Omega}_{be}^e \boldsymbol{C}_e^b \\ &= -\left[\boldsymbol{C}_b^e \delta\omega_{be}^e \times\right] \end{aligned} \tag{3.90}$$

Gyros measure the total rotation of the body frame with respect to the inertial frame transformed into the body frame, ω_{ib}^b. This includes the rotation of the body with respect to the earth, ω_{be}^b, plus the the rotation of the earth with respect to the inertial frame, ω_{ie}^e transformed over to the body frame. Then,

$$\omega_{be}^b = \omega_{ib}^b - \boldsymbol{C}_e^b \omega_{ie}^e \tag{3.91}$$

Given that the rotation rate of the earth is known (i.e. $\delta\omega_{ie}^e = 0$), the error form of this equation becomes

$$\delta\omega_{be}^b = \delta\omega_{ib}^b \tag{3.92}$$

substituting equation 3.92 into equation 3.90 yields

$$[\delta\dot{\phi}\times] = -\boldsymbol{C}_b^e \left[\delta\omega_{ib}^b \times\right]$$

or

$$\delta\dot{\phi} = -\boldsymbol{C}_b^e \delta\omega_{ib}^b \tag{3.93}$$

Thus, the propagation of attitude errors in the Earth frame equals errors in gyros measurements transformed over to the earth frame via $\boldsymbol{C}_b^e$. Therefore, since attitude computation is critical in inertial navigation mechanization, then *gyros are the most critical components* in inertial measurement units (IMU). Equation 3.93 proves that attitude error computation through perturbation in the ECEF frame as the true frame is independent of the position (Sukkarieh, S., 2000).

From the Earth frame velocity equation given in 3.54, the evaluated velocity is written as

$$\dot{\tilde{\boldsymbol{v}}}_e^e = \tilde{\boldsymbol{C}}_b^e \tilde{\boldsymbol{f}}^b - [2\tilde{\omega}_{ie}^e \times \tilde{\boldsymbol{v}}_e^e] + \tilde{\boldsymbol{g}}_l^e \tag{3.94}$$

by perturbation and using equation 3.80

$$\begin{aligned} \delta\dot{\boldsymbol{v}}_e^e &= \dot{\tilde{\boldsymbol{v}}}_e^e - \tilde{\boldsymbol{v}}_e^e \\ &= \left[\tilde{\boldsymbol{C}}_b^e \tilde{\boldsymbol{f}}^b - \boldsymbol{C}_b^e \boldsymbol{f}^b\right] - [2\tilde{\omega}_{ie}^e \times \tilde{\boldsymbol{v}}_e^e - 2\omega_{ie}^e \times \boldsymbol{v}_e^e] + [\tilde{\boldsymbol{g}}_l^e - \boldsymbol{g}_l^e] \\ &= \boldsymbol{C}_b^e \left[\tilde{\boldsymbol{f}}^b - \boldsymbol{f}^b\right] - [\delta\phi\times]\,\boldsymbol{C}_b^e \tilde{\boldsymbol{f}}^b - \delta\left[2\omega_{ie}^e \times \boldsymbol{v}_e^e\right] + \delta\boldsymbol{g}_l^e \end{aligned} \tag{3.95}$$

Ignoring Coriolis and gravity error terms results in

$$\begin{aligned} \delta\dot{\boldsymbol{v}}_e^e &= \boldsymbol{C}_b^e \delta\boldsymbol{f}^b - [\delta\phi\times]\,\boldsymbol{C}_b^e \tilde{\boldsymbol{f}}^b \\ &= \boldsymbol{C}_b^e \delta\boldsymbol{f}^b - [\delta\phi\times]\tilde{\boldsymbol{f}}^e \end{aligned} \tag{3.96}$$

then

$$\delta\dot{\boldsymbol{v}}_e^e = [\tilde{\boldsymbol{f}}^e \times]\delta\phi + \boldsymbol{C}_b^e \delta\boldsymbol{f}^b \tag{3.97}$$

That is, the propagation of velocity errors in the ECEF frame is a function of the acceleration error in the ECEF frame due to the misalignment $[\tilde{\boldsymbol{f}}^e \times]\delta\phi$, and the errors in acceleration measurements transformed over to the ECEF frame. The position error equation is simply given by

$$\delta \dot{\mathbf{p}}_e^e = \delta \mathbf{v}_e^e \tag{3.98}$$

Equations 3.93 and 3.97 describes attitude and velocity errors propagation respectively. The only terms that need to be determined are the errors in accelerometers measurements $\delta \boldsymbol{f}^b$ and the errors in gyros measurements $\delta \omega_{ib}^b$. These two terms are experimentally evaluated and on-line updated during the integration process as described in subsections 3.3.1 and 4.4.1.

Since perturbation of the nominal navigation states is considered, the resulted error equations become linear and a linear filter is employed. On the other hand, a generalized non-linear error model for low-cost INS integration can be found in (Giroux, R., Gourdeau, R., & Landry, R., J., 2004b) and (Giroux, R., Gourdeau, R., & Landry, R., J., 2004a). In such cases, a non-linear filter such the extended of the unscented Kalman filters must be employed. However, in practice, it is more common in cases where non-linear filter is desired to use the actual system estimates as filter states instead of estimates errors.

3.4 Performance Verification

In this section, the accuracy of the implemented INS mechanization is verified. This is an important step in the design of INS/GPS navigation systems, since the accuracy of the implementation is vulnerable to computational errors which in turn can affect the whole accuracy of the implementation. Figures 3.7, 3.8 and 3.9, shows errors in INS mechanisation when sensors are assumed clean and the performance of these sensors resembles expensive sensors performance. Figure 3.10 shows the simulated trajectory when the "CrossBow" IMU extracted noise of Figures 3.4 and 3.5 are applied to the simulated accelerations and rotational rates. This Figure depicts the fact that, when low -cost sensors are employed, the unaided navigation system in unreliable even for short periods.

3.5 Chapter Summary

The focus of this chapter was on the development on an inertial navigation system based on low-cost inertial sensors. In doing so this chapter has

- Developed the inertial navigation equations in both the local geographical frame and the EFEF frame.
- Discussed the implementation of the mechanization equations using a unified mathematical frame work, where computational errors in one state computation, namely, attitude, velocity and position are independent of each other.
- Discussed errors commonly associated with low-cost inertial sensors and developed the general error equations. Moreover, it discussed issues relating attitude computations, and it was shown that there is a trade off between the order of the attitude algorithm and the sampling rate of the IMU. It was stated that:
    - The bandwidth of the gyros employed is critical in SDINS design. If vibration exceeds the bandwidth of the gyros, it wont be detected and it will be considered as translational motion resulting into errors in calculated velocity and position. However, vibration absorbers may be employed to attenuate for frequencies above the cutoff of the gyros.
    - The ration of sampling rate to maximum expected continuous rotation rate of the vehicle should be higher than 1.
    - Attitude computation error can be efficiently reduced by increasing the order of the attitude algorithm.
- Developed the navigation error equations in the local geographical frame. These equations as considered as the process model in the Kalman filter implementation.

Finally, this chapter provided the necessary background and methods to develop an appropriate low-cost inertial navigation system.

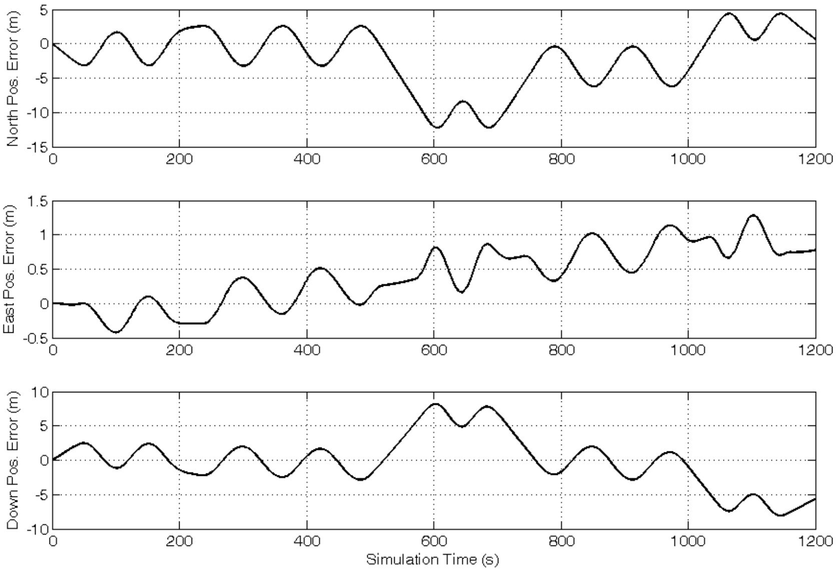


Figure 3.7: Error in INS computed position with bias and noise free inertial sensors. In such environments, errors are due to INS mechanization.

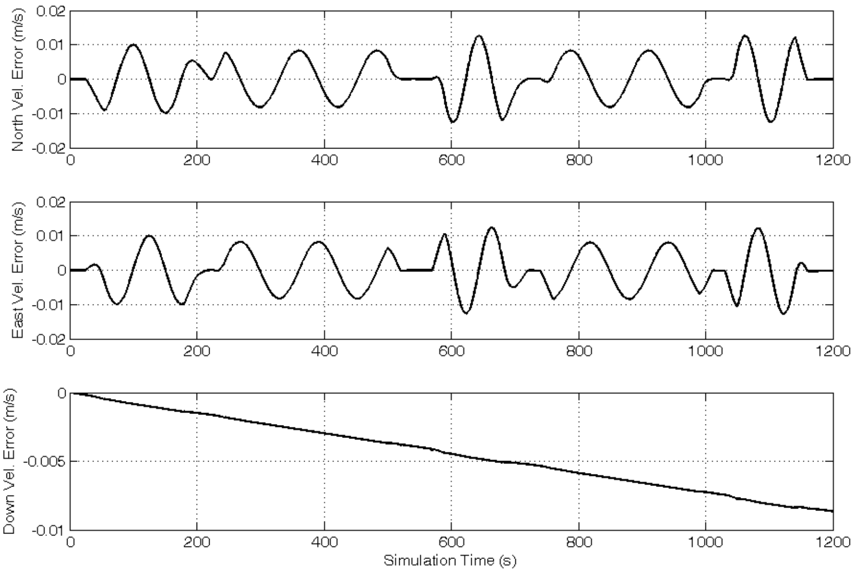


Figure 3.8: Error in INS computed velocity with bias and noise free inertial sensors.

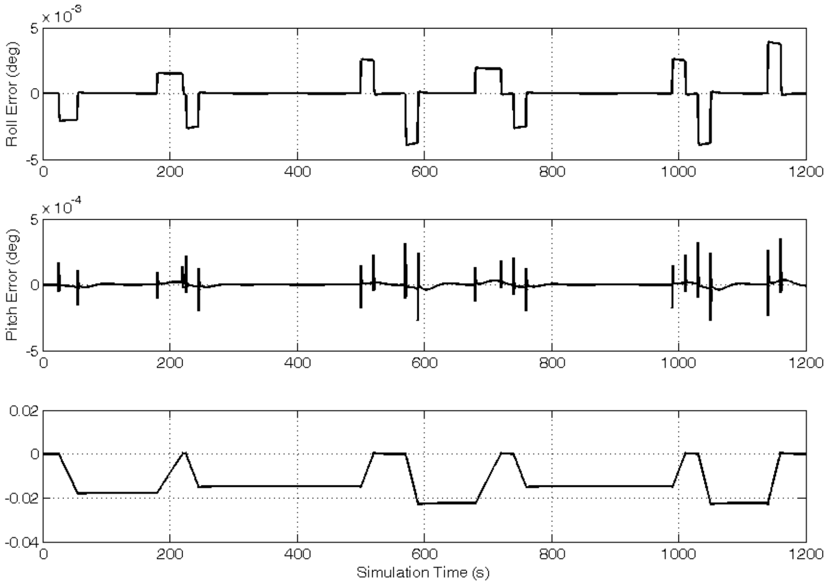


Figure 3.9: Error in INS computed attitude with bias and noise free inertial sensors.

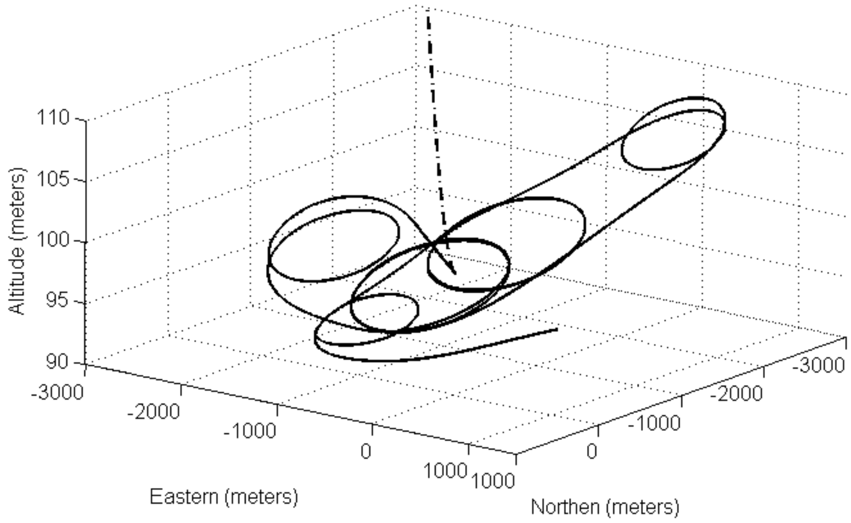


Figure 3.10: Once bias and noise are applied to inertial measurements, the IMU (dashed line) becomes unreliable for navigation even for a very short period of time compared to the noise free case (solid line).

Chapter 4

GPS Aided Inertial Navigation Systems

4.1 Introduction

Early spaced-based radio-navigation aids (or simply Satellite navigation systems), namely the US Navy Navigation Satellite System which was referred to as TRANSIT and the Russian Tsikada system, provided a two-dimensional high accuracy positioning service since 1964. While industry has used the Starfix system since 1986. One Limitation of such systems is its very low update rates (10 to 15 minutes) which made them acceptable for use on low-dynamic platforms, such as shipboard navigation. Although, It had been developed by the Government of the United States, namely the Department of Defence, since 1969, it was in 1994 when the Global Positioning System (GPS) was declared ready for aviation use. Other satellite navigation systems include the European Union GALILEO satellite system, which is expected to be operational in 2008, the Russian GLONASS system, The Chinese BeiDou system and the next generation INMARSAT (Tsui, J. B.Y., 2000),(Kaplan, E. D., & Hegarty,C. J., 2006),(Kelly, A, 1994)

SDINSs, discussed in the previous chapter, provides an inertial-based localization. However, they grift with time. Biases and drifts are low frequency faults in inertial navigation systems. The GPS provides another localization solution. However, it is relatively unreliable. The GPS signal can be in error due to multipath and change in satellite geometry. It requires a line-of-sight (LOS) communication link to communicate efficiently with the receiver. In fact, these are sources of high frequency faults in GPS fixes. In fact, the IMU and the GPS are the natural complements of each other. SDINS low frequency faults are corrected by GPS low frequency fixes (1Hz for the "Novatel" receiver). On the other hand, high frequency faults in the GPS fixes are corrected by the high frequency estimates of the SDINS.

In this chapter, fundamentals of the GPS and data processing are first discussed in section 4.3. Loosely and tightly coupled Kalman filter configurations are discussed in section 4.4. Finally, simulated results and real tests are represented in sections 4.5 and 4.6, respectively.

4.2 GPS Observables

For absolute state estimates, the C/A code pseudoranges and the Doppler frequency information (also known as delta ranges) from all available satellites in view are used to form the measurement equations. The precision of the solution is affected by the number of satellites in view, geometry of the available satellites referred to as Geometric Dilution of Precision (GDOP), and the precision in range determination.

The peudorange measurement from satellite to receiver is represented as (Abdel-Hafez, M., 2003)

$$\rho_r^{sv} = \mathbf{r}_r^{sv} - c\left[\Delta t_{sv} - \Delta t_u - \Delta t_{trop}^{sv} - \Delta t_{ion}^{sv} - \Delta t_{mp}^{sv}\right] + \epsilon_{eph}^{sv} + \eta_U^{sv} \tag{4.1}$$

where

ρ_r^{sv}	represents measured pseudorange between the satellite vehicle (SV) and user (U)
Δt_{sv}	represents the satellite clock bias (in seconds)
Δt_u	represents the receiver clock bias (in seconds)
Δt_{trop}	represents tropospheric delay (in seconds)
Δt_{ion}	represents ionospheric delay (in seconds)
Δt_{mp}^{sv}	represents multipath delay

ϵ_{eph}^{sv} represents ephemeris error (in meters)

η_U^{sv} represents satellite and receiver noise and hardware delays (in meters)

c represents speed of light

$\mathbf{r}_r^{sv}$ represents the geometric range between satellite (SV) and receiver (U), given by

$$\mathbf{r}_r^{sv} = \sqrt{(x_{sv} - x_u)^2 + (y_{sv} - y_u)^2 + (z_{sv} - z_u)^2} \tag{4.2}$$

where

x_{sv}, y_{sv}, z_{sv} represent the satellite (SV) position in the ECEF coordinate system and predicted with the ephemeris information (in meters)

x_u, y_u, z_u represent the receiver position in the ECEF coordinate system and predicted by the INS (in seconds)

The corresponding measurements prediction matrix $\mathbf{H}$ is obtained by linearizing around the nominal user position x_u, y_u, z_u. With n satellites in the view, $\mathbf{H}$ is formulated as follows

$$\mathbf{r}_1 = \sqrt{(x_1 - x_u)^2 + (y_1 - y_u)^2 + (z_1 - z_u)^2} + C_b \tag{4.3}$$

$$\mathbf{r}_2 = \sqrt{(x_2 - x_u)^2 + (y_2 - y_u)^2 + (z_2 - z_u)^2} + C_b \tag{4.4}$$

$$\mathbf{r}_3 = \sqrt{(x_3 - x_u)^2 + (y_3 - y_u)^2 + (z_3 - z_u)^2} + C_b \tag{4.5}$$

$$\mathbf{r}_n = \sqrt{(x_n - x_u)^2 + (y_n - y_u)^2 + (z_n - z_u)^2} + C_b \tag{4.6}$$

where $C_b = c\Delta t_u$

$$\begin{aligned}\mathbf{H} &= \begin{pmatrix} \frac{\partial r_1}{\partial x_u} & \frac{\partial r_1}{\partial y_u} & \frac{\partial r_1}{\partial z_u} & \frac{\partial r_1}{\partial C_b} \\ \frac{\partial \ r_2}{\partial x_u} & \frac{\partial r_2}{\partial y_u} & \frac{\partial r_2}{\partial z_u} & \frac{\partial r_2}{\partial C_b} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial r_n}{\partial x_u} & \frac{\partial r_n}{\partial y_u} & \frac{\partial r_n}{\partial z_u} & \frac{\partial r_n}{\partial C_b} \end{pmatrix} \\ &= \begin{pmatrix} \frac{x_u - x_1}{\mathbf{r}_r^{sv}} & \frac{y_u - y_1}{\mathbf{r}_r^{sv}} & \frac{z_u - z_1}{\mathbf{r}_r^{sv}} & 1 \\ \frac{x_u - x_2}{\mathbf{r}_r^{sv}} & \frac{y_u - y_2}{\mathbf{r}_r^{sv}} & \frac{z_u - z_2}{\mathbf{r}_r^{sv}} & 1 \\ \vdots & \vdots & \vdots & \vdots \\ \frac{x_u - x_n}{\mathbf{r}_r^{sv}} & \frac{y_u - y_n}{\mathbf{r}_r^{sv}} & \frac{z_u - z_n}{\mathbf{r}_r^{sv}} & 1 \end{pmatrix}\end{aligned} \tag{4.7}$$

The difference between predicted and measured (after the removal of all the error terms) is then used to compute the error in the position and clock bias of the receiver using least squares

$$\boldsymbol{\Delta}\mathbf{x} = \left(\mathbf{H}^T\mathbf{H}\right)^{-1}\mathbf{H}\boldsymbol{\Delta}\rho \tag{4.8}$$

or the Kalman filter as in appendix E (George, M. G., & Sukkarieh, S., 2005)(George, M. G., & Sukkarieh, S., 2007)

4.3 GPS Basic Concepts

The GPS satellite constellation consists of 24 satellites arranged in 6 orbital planes with 4 satellites per plane. Navigation and range data are transmitted on two different frequencies, L1 (1575.42 MHz) and L2 (1227.60 MHz). The initial purpose of this dual frequency transmission is to eliminate the ionospheric errors introduced by the propagation of the GPS satellite signal through the atmosphere. Each satellite generates two different codes; a short code referred to as Coarse / Acquisition or C/A code and a long code referred to as the precision or P-code. The P-code provides the Precision Positioning System (PPS) and is encrypted by a Y-code to form a P(Y) code. In order to receive the P(Y) code, one must have the classified Y-code. The Y-code code is only available for US authorized military and government users. The P(Y) code is a 6.1871×10^{12} bit long code transmitted at 10.23 Mbps (10.23 MHz) on both the L1 and L2 frequencies. It repeats every one week and forms a segment of a master code of approximately 2.35×10^{14} bits master code. Each satellite is assigned a segment.

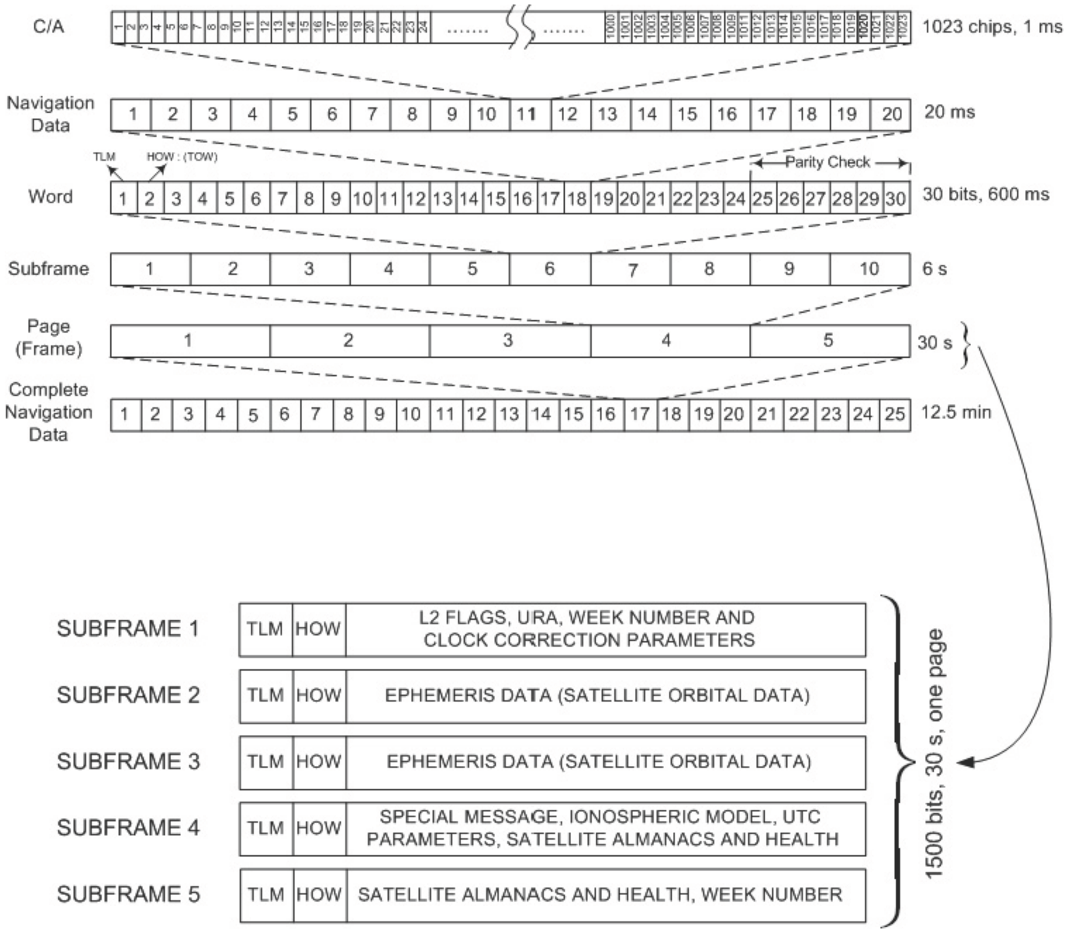


Figure 4.1: GPS data format

The C/A code is a 1023-bit code transmitted at 1.023 Mbps (1.023 MHz) on the L1 frequency only. It repeats every 1ms and carries the Standard Positioning System (SPS) intended for the civil community.

Figure 4.1 (Tsui, J. B.Y., 2000) shows the GPS data format (Tsui, J. B.Y., 2000) (Kayton, M., & Fried, W. R., 1997). The first row shows a C/A code. The second one shows a navigation data bit with rate of 50 Hz; it is 20 ms long and contains 20 C/A codes. The third row represents a word that is 600 ms long and contains 30 data bits. Ten words make a subframe that is 6 seconds long and contains 300 bits. The fifth row shows a page (or frame) containing the GPS navigation message. It spans for 30 seconds and consists of five subframes. The complete data set is 12.5 minutes long. It contains 25 pages and is referred to as superframe. Subframe 1 of each navigation frame contains Issue of date clock (IODC), GPS system week number, GPS system Time-Of-the Week (TOW) and clock correction parameters. Subframes 2 and 3 contain Issue of date ephemeris (IODE) and ephemeris data. Subframes 4 and 5 contain Almanac; that is, the data describe the long-term parameters, and ionospheric correction model. Subframes 1, 2 and 3 repeats every 30 seconds. Subframes 4 and 5 contain $\frac{1}{25}$ of almanac data, so that a complete data message requires the transmission of 25 full 1500-bit frames, or 12.5 minutes.

4.3.1 GPS System Error Budget

Orbital (Ephemeris) Error

A satellite orbit can be described by six Keplerian parameters, of which three are orientational and the other three are dimensional, The six parameters are shown in (Fig. 4.2) and are as follows:

1. **Orientational parameters**
   - Inclination of the orbital plane with equatorial plane, i.
   - The argument of the perigee, ω.
   - Geocentric longitude of the ascending line of nodes (or argument of latitude), Ω.

2. **Dimensional parameters**

    - The semimajor axis of the ellipse, a.
    - The eccentricity of the orbit, e.
    - The time of the perigee passage or the true anomaly, v.

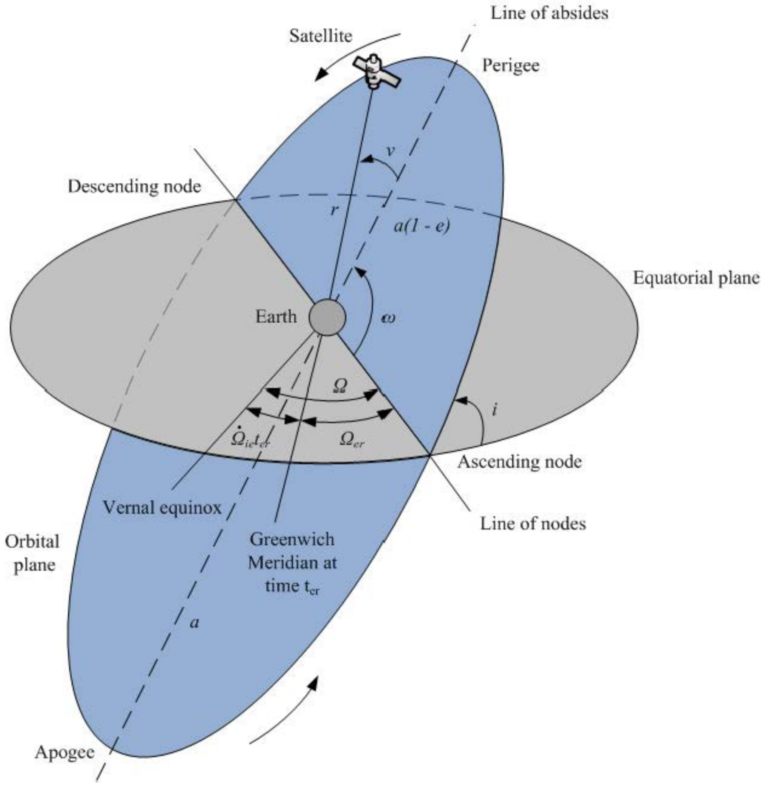


Figure 4.2: Satellite orbit

Perturbations, such as variations in the Earth's gravitational field between the perigee and the apogee and gravitational attraction of the moon, cause the orbital plane to vary from its elliptical path. The navigation message contains the ephemeris data which consists of parameters required to correct for these perturbations, Table 4.1 summarizes the ephemeris parameters while the detailed description of these parameters and their processing can be found in (GPS Standards, 1995) and a friendly representation can be found in (Tsui, J. B.Y., 2000) and (Kayton, M., & Fried, W.

R., 1997). These corrections are generated using a curve fit of the control segment's best prediction of each satellite's position at the time of upload.

Ephemeris are used to calculate satellites positions. Given coarse time of a satellite signal transmission (GPS timing and time calculation is explained in appendix D), Table 4.2 lists the sequence of calculating the satellite position *at time of transmission*. This position is still need to be corrected for different effect such as receiver clock bias, sagnac effect (effect of the Earth rotation), atmospheric delays, and the satellite clock bias. These effects will be discussed in next subsections and later the complete process will be summarized by flow charts.

As shown in (Fig. 4.3), ephemeris errors are generally smallest in the radial direction, which provides the ranging measurements. The components of ephemeris errors in the along-track and the cross-track directions are much larger. However, they do not affect ranging information. Selective availability (SA), which is an error deliberately added by the US Department of Defence, works by corrupting the ephemeris data and increase the radial error component. On the other hand, dithering of satellites clock is the dominant of SA implementation. Errors in ephemeris data will result in satellite position error in the range of 1-6 meters and in the order of 0.8 meters in pseudorange measurements (Kaplan, E. D., & Hegarty,C. J., 2006).

Clock Errors

GPS satellites and receivers contain internal clocks that control GPS system timing operations. Typical satellite and receiver clocks are shown in (Fig. 4.4(a)) and (Fig. 4.4(b)), respectively. However, these clocks drift with time resulting in a degradation in the navigational solution. The navigation message from the satellite includes parameters describing the satellite's clock bias and drift. They are predicted by the

Table 4.1: Ephemeris parameters

Parameter	**Description**	**Unit**
M_0	Mean anomaly at reference time	semicircle
Δn	Mean motion difference from computed value	semicircle/s
e	Orbital eccentricity	dimensionless
$\sqrt{A}$	Square root of the semimajor axis	meters$^{\frac{1}{2}}$
Ω_e	Longitude of ascending node of orbit plane at weekly epoch	semicircle
i_0	Inclination angle at reference time	semicircle
ω	Argument of perigee	semicircle
$\dot{\Omega}$	Rate of right ascending	semicircle/s
$IDOT$	Rate of inclination angle	semicircle/s
C_{uc}	Amplitude of the cosine harmonic correction to argument of latitude	radians
C_{us}	Amplitude of the sine harmonic correction to argument of latitude	radians
C_{rc}	Amplitude of the cosine harmonic correction to the orbit radius	meters
C_{rs}	Amplitude of the sine harmonic correction to the orbit radius	meters
C_{ic}	Amplitude of the cosine harmonic correction to angle of inclination	radians
C_{is}	Amplitude of the sine harmonic correction to angle of inclination	radians
t_{0e}	Reference time at ephemeris computations	seconds
$IODE$	Issue of data - ephemeris	dimensionless

control segment. However, perturbations in satellite position and instability in its clock causes this predictions to be in error. Since the range between the satellite and the user is a function of time, Satellite clock error can result in position error in the range of 2 meters (Kelly, A, 1994). Satellite clock bias is given by

$$\Delta t_{SV} = a_{f_0} + a_{f_1}(t - t_{oc}) + a_{f_2}(t - t_{oc})^2 + \Delta t_r - t_{GD} \tag{4.9}$$

Where a_{f_0} is the satellite clock bias in seconds, a_{f_1} is the clock drift in seconds per seconds, a_{f_2} is frequency drift (aging) in seconds per seconds squared, Δt_r is the relativistic error correction term, t_{GD} and t_{oc} are the group delay differential and the

Table 4.2: Calculating satellite position from ephemeris

Step	**Equation**	**Description of Computation**
1.	$t_k = t_c - t_{0e}$	Time from ephemeris reference epoch t_{0e}
2.	$n_0 = \sqrt{\frac{\mu}{A^3}}$	Computed mean motion
3.	$n = n_0 + \Delta n$	Corrected mean motion
4.	$M_k = M_0 + nt_k$	Mean anomaly at time t_k
5.	$M_k = E_k - e \sin E_k$	Iterate to solve Kepler's equation for mean anomaly at time t_k
6.	$v_1 = cos^{-1}\left(\frac{\cos E_k - e}{1 - e^2 \cos E_k}\right)$	True anomaly from cosine at time t_k
7.	$v_2 = sin^{-1}\left(\frac{\sqrt{1-e^2} \sin E_k}{1 - e \cos E_k}\right)$	True anomaly from sine at time t_k
8.	$v_k = v1 sign(v2)$	True anomaly at time t_k
9.	$\phi_k = v_k + \omega$	Argument of latitude at time t_k
10.	$\delta\phi_k = C_{us} \sin 2\phi_k + C_{uc} \cos 2\phi_k$	Second harmonic correction to argument of latitude at time t_k
11.	$\delta r_k = C_{rs} \sin 2\phi_k + C_{rc} \cos 2\phi_k$	Second harmonic correction to orbit radius at time t_k
12.	$\delta i_k = C_{is} \sin 2\phi_k + C_{ic} \cos 2\phi_k$	Second harmonic correction to inclination angle at time t_k
13.	$u_k = \phi_k + \delta\phi_k$	Corrected argument of latitude at time t_k
14.	$r_k = A(1 - e \sin E_k) + \delta r_k$	Corrected orbit radius at time t_k
15.	$i_k = i_0 + \delta i_k + IDOT \cdot t_k$	Corrected inclination angle at time t_k
16.	$\acute{x}_k = r_k \cos u_k$	x coordinate of satellite position in orbital plane at time t_k
17.	$\acute{y}_k = r_k \sin u_k$	y coordinate of satellite position in orbital plane at time t_k
18.	$\Omega_{er} = \Omega_e + \dot{\Omega} \cdot t_k - \dot{\Omega}_{ie} \cdot t_c$	Corrected longitude of ascending node at time t_k and accounting for Earth's rotation
19.	$x_k = \acute{x}_k \cos \Omega_{er} - \acute{y}_k \cos i_k \sin \Omega_{er}$	ECEF x coordinate of satellite position at time t_k
20.	$y_k = \acute{x}_k \sin \Omega_{er} - \acute{y}_k \cos i_k \cos \Omega_{er}$	ECEF y coordinate of satellite position at time t_k
21.	$x_k = \acute{y}_k \sin i_k$	ECEF z coordinate of satellite position at time t_k

GPS epoch time obtained from the ephemeris data.

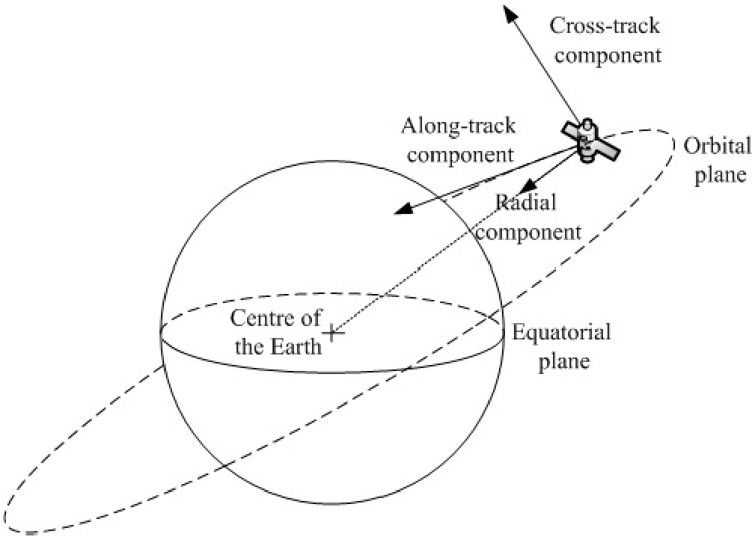
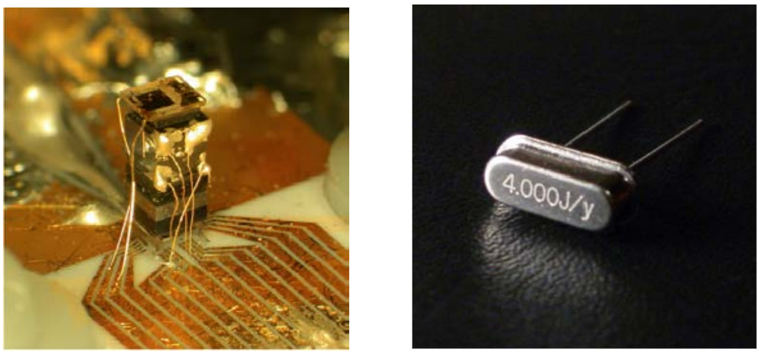


Figure 4.3: Components of ephemeris errors



(a) Atomic clock

(b) Quartz clock

Figure 4.4: Atomic clock located in GPS satellites and Quartz clock located in receivers (http://www.wikipedia.org).

Atmospheric Errors

Atmospheric delay can be classified into two types: non-dispersive associated with the troposphere and dispersive associated with the ionosphere. The two layers affect the propagation of the satellite signal transmission in different ways. The troposphere extends nominally from 8 to 40 km above the surface of the Earth. The characteristics of the troposphere is affected by changes in its temperature, humidity and pressure. Water and other atmospheric constituents cause changes in the index of refraction.

On the other hand, the ionosphere is above 50 km from the surface of the Earth. It consists of ionized air. Changes in the ionization level also affect the refractive indices of ionospheric layers, which in turn affect the propagation of the satellite transmitted signal.

The total atmospheric delay is given by:

$$\Delta t_a = \Delta t_{trop} \pm \Delta t_{ion}(f) \tag{4.10}$$

Where f is the frequency of the carrier of the satellite transmitted signal. The (+) sign is used for code pseudorange, while the (-) sign is used for the phase range (Farrell, J. A., & Barth, M., 1998).

Tropospheric delay is modelled in different ways (Farrell, J. A., & Barth, M., 1998)(Kayton, M., & Fried, W. R., 1997)(Tsui, J. B.Y., 2000)(Kaplan, E. D., & Hegarty,C. J., 2006). It is normally modelled as having two components, a dry one and a wet one. The wet components exists in at lower levels of the troposphere which has higher water-vapor contents than the higher levels, where the dry component exists. The wet components is difficult to be modelled and accounts for approximately 10% of total tropospheric delay. An example of troposphere delay modelling is the Chao model.

$$\Delta\rho_{dry} = 2.276 \times 10^{-5} P \tag{4.11}$$

$$F_{dry} = \frac{1}{\sin(E) + \frac{0.00143}{\tan(E)+0.0445}} \tag{4.12}$$

$$\Delta\rho_{wet} = 4.70 \times 10^{2} \frac{e^{1.23}}{T^2} + 1.705 \times 10^{6} \alpha \frac{e^{1.46}}{T^3} \tag{4.13}$$

$$F_{wet} = \frac{1}{\sin(E) + \frac{0.00035}{\tan(E)+0.017}} \tag{4.14}$$

$$\Delta\rho_{trop} = \Delta\rho_{dry} F_{dry} + \Delta\rho_{wet} F_{wet} \tag{4.15}$$

Where $\Delta\rho$ is in meters. The Chao model requires additional components to be measured, such as the atmospheric pressure P, the temperature in Kelvin T and the partial pressure of water vapor e. This makes this model expensive.

Another simplified models are, for example, the Magnavox and the Collins models, are as follows

$$\Delta\rho_M = \frac{2.208}{\sin(E)}(e^{\frac{-h_r}{6900}} - e^{\frac{-h_s}{6900}}) \tag{4.16}$$

$$\Delta\rho_C = \frac{2.4225}{0.026 + \sin(E)} e^{\frac{-h_r}{7492.8}} \tag{4.17}$$

Where h_r and h_s are the receiver and satellite altitude in meters, respectively.

A more simplified model is as follows

$$\Delta\rho_{trop} = \frac{2.47}{0.0121 + \sin(E)} \tag{4.18}$$

It is important to know whether the GPS receiver used is compensating for tropospheric delay or not before starting to implement the correction algorithm. Typical tropospheric delays may reach 30 meters for satellites at low elevation angles (E).

Ionospheric delay is frequency dependent. Dual frequency receivers can eliminate ionospheric delays by using dual frequency ranging measurements as follows

$$\Delta\rho_{L_1} = \frac{f_{L2}^2}{f_{L2}^2 - f_{L1}^2}(\rho_{L_1} - \rho_{L_2}) \tag{4.19}$$

where

$\Delta\rho_{L_1}$ is the pseudorange measurements ionospheric error on L1.

f_{L1} is the L1 frequency = 1575.42 MHz.

f_{L2} is the L2 frequency = 1227.60 MHz.

ρ_{L_1} is the measured pseudorange on the L1 frequency.

ρ_{L_2} is the measured pseudorange on the L2 frequency.

However, when single frequency receivers are utilized, Ionospheric errors becomes time dependent and a different models is implemented (GPS Standards, 1995).

Relativistic Errors

Due to the elliptical shape of satellites orbits and the fact that a satellite is closer to the Earth at perigee and further at apogee, the satellite velocity is not constant (higher at the perigee and slower at the apogee). This will cause the satellite clock to run slower and then faster respectively. The relativistic error correction term Δt_r in equation D.4 is applied to compensate for this satellite clock variation.

Another relativistic error is the error introduced to the calculated satellite positions by the rotational rate of the Earth, termed as the Sagnac effect. This effect appears when the ECEF coordinate system is utilized, since this frame is rotating with respect to the ECI reference frame and due to this rotation, the actual position of satellites change while the satellite signal is travelling a way towards the Earth. The Sagnac effect can result into position error on the order of 30 meters (Kaplan, E. D., & Hegarty,C. J., 2006).

Multipath Errors

Among all previously mentioned GPS measurement errors, multipath is on of the most significant errors. Multipath is the reception of reflected or diffracted replicas of the desired signal. It varies significantly in magnitude depending on the environment within which the receiver is located, satellite elevation angle, receiver signal processing, antenna gain pattern, and signal characteristics. In land vehicle applications multipath can result from near buildings, trees and other reflecting surfaces. However, in Unmanned Arial Vehicle (UAV) applications, the only source of multipath is the reflection by the aircraft body surfaces. Multipath mitigation techniques include antenna siting, removing or modification of reflective structures in the vicinity of the antenna, replacing the antenna closer or farther (based on the environment) to the ground. Some of GPS antennas are designed to attenuate multipath. Moreover, some of GPS receivers have an embedded multipath processing (GPS Standards, 1995).

In INS/GPS integrated systems, multipath has a significant effect on filter's decision. In places where multipath occurs, the measurement noise covariance matrix **R** is no longer represent the noise level in GPS measurements and filter decision, represented by the magnitude of the Kalman gain, is no longer reflect the uncertainty in the measurements.

4.3.2 Experiments - Ephemeris Data Processing / Calculating Satellites Positions

The flow chart of figure 4.5 shows the process of calculating satellite and receiver positions from ephemeris broadcasts, while the flow chart of figure 4.6 magnifies the step of calculating the user position using any kind of estimators (i.e. Kalman filter or least-squares) and the correction of satellite positions.

Data were taken for the Australian Center for Field Robotics (ACFR) data base. This set of data includes pseudo-ranges measurements, ephemeris parameters and receivers estimates of receiver position and clock bias. These estimates are useful to check the accuracy of the computation. Figure 4.7 shows projections of calculated satellites positions on the Earth's surface, while figures 4.8 and 4.9 shows estimated clock bias and errors in estimated position and clock bias respectively. Finally, Figure 4.10 shows trajectories of the estimated position and calculated by previously described algorithm and the position computed, internally, by the GPS receiver embedded software.

4.4 Filter Structure

Once both GPS observations and INS estimates are ready, they become inputs to the Kalman filter. GPS/INS integration techniques varies with respect to filter's structure and the integration scheme. The filter can be constructed in loosely or tightly coupled format, while the integration can take the form of direct or indirect feedback. In this

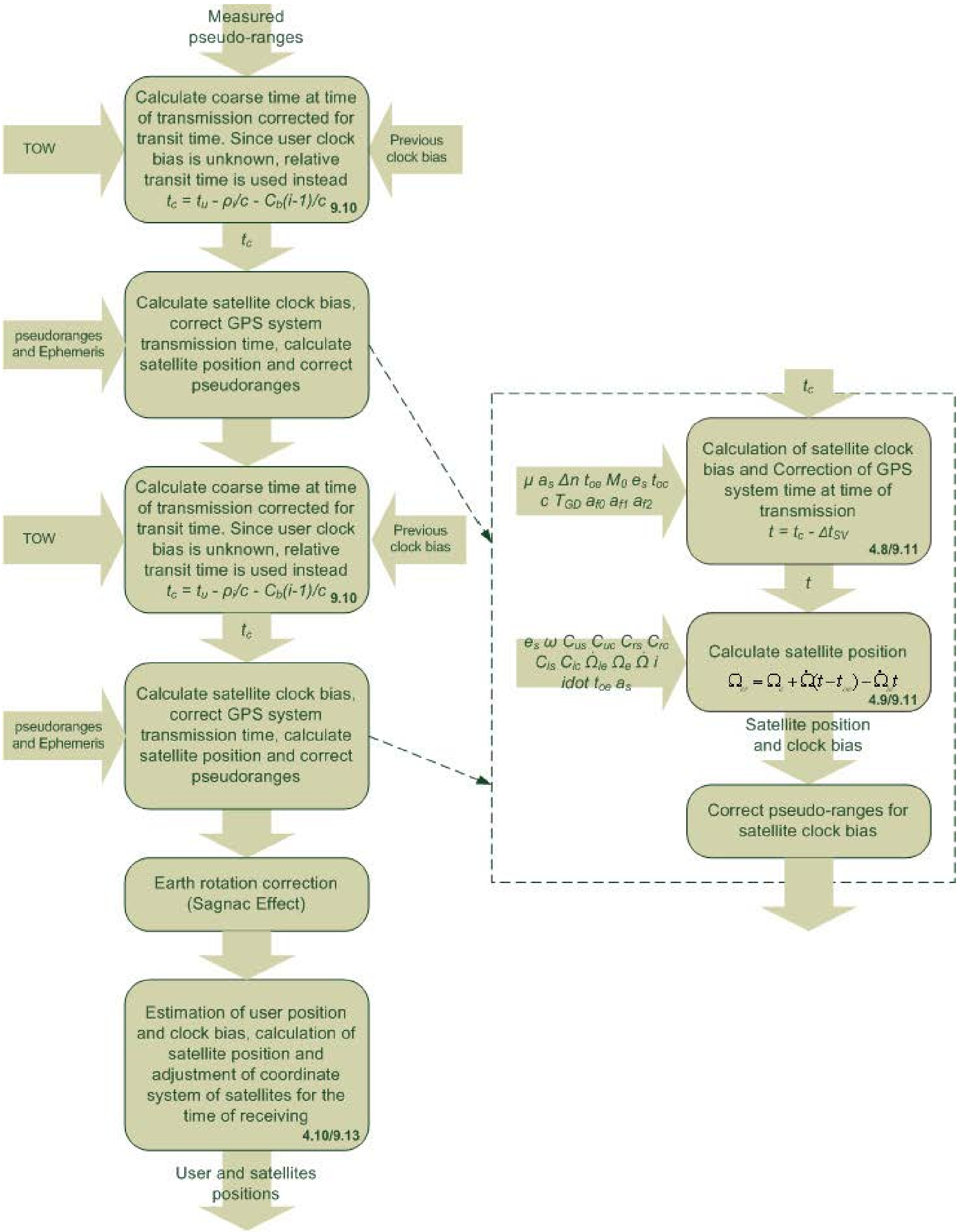

Measured pseudo-ranges
TOW
Calculate coarse time at time of transmission corrected for transit time. Since user clock bias is unknown, relative transit time is used instead
$t_c = t_u - \rho_i/c - C_b(i-1)/c$ 9.10
Previous clock bias
t_c
pseudoranges and Ephemeris
Calculate satellite clock bias, correct GPS system transmission time, calculate satellite position and correct pseudoranges
TOW
Calculate coarse time at time of transmission corrected for transit time. Since user clock bias is unknown, relative transit time is used instead
$t_c = t_u - \rho_i/c - C_b(i-1)/c$ 9.10
Previous clock bias
t_c
pseudoranges and Ephemeris
Calculate satellite clock bias, correct GPS system transmission time, calculate satellite position and correct pseudoranges
Earth rotation correction (Sagnac Effect)
Estimation of user position and clock bias, calculation of satellite position and adjustment of coordinate system of satellites for the time of receiving
4.10/9.13
User and satellites positions
t_c
$\mu\ a_s\ \Delta n\ t_{oe}\ M_0\ e_s\ t_{oc}$
$c\ T_{GD}\ a_{f0}\ a_{f1}\ a_{f2}$
Calculation of satellite clock bias and Correction of GPS system time at time of transmission
$t = t_c - \Delta t_{SV}$
4.8/9.11
t
$e_s\ \omega\ C_{us}\ C_{uc}\ C_{rs}\ C_{rc}$
$C_{is}\ C_{ic}\ \dot{\Omega}_{ie}\ \Omega_e\ \dot{\Omega}\ i$
$idot\ t_{oe}\ a_s$
Calculate satellite position
$\Omega_{er} = \Omega_e + \dot{\Omega}(t - t_{oe}) - \dot{\Omega}_{ie} t$
4.9/9.11
Satellite position and clock bias
Correct pseudo-ranges for satellite clock bias

Figure 4.5: Satellites positions were calculated through ephemeris data processing while receiver position and clock bias were estimated by a least-squares estimator

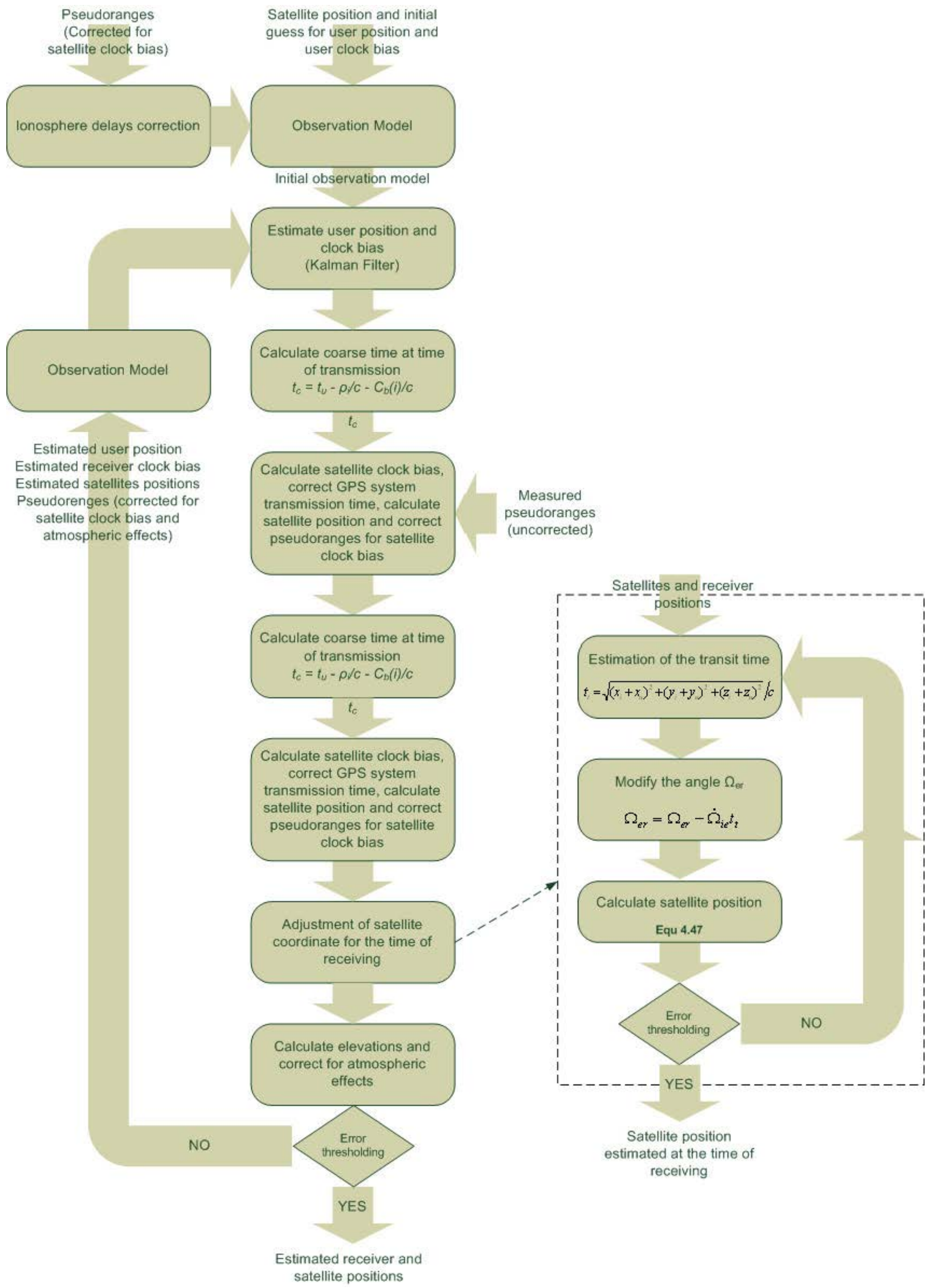

Pseudoranges (Corrected for satellite clock bias)
Satellite position and initial guess for user position and user clock bias
Ionosphere delays correction
Observation Model
Initial observation model
Estimate user position and clock bias (Kalman Filter)
Observation Model
Calculate coarse time at time of transmission $t_c = t_u - \rho/c - C_b(i)/c$
t_c
Estimated user position
Estimated receiver clock bias
Estimated satellites positions
Pseudorenges (corrected for satellite clock bias and atmospheric effects)
Calculate satellite clock bias, correct GPS system transmission time, calculate satellite position and correct pseudoranges for satellite clock bias
Measured pseudoranges (uncorrected)
Calculate coarse time at time of transmission $t_c = t_u - \rho/c - C_b(i)/c$
t_c
Calculate satellite clock bias, correct GPS system transmission time, calculate satellite position and correct pseudoranges for satellite clock bias
Adjustment of satellite coordinate for the time of receiving
Calculate elevations and correct for atmospheric effects
NO
Error thresholding
YES
Estimated receiver and satellite positions
Satellites and receiver positions
Estimation of the transit time
$t_t = \sqrt{(x_s + x_u)^2 + (y_s + y_u)^2 + (z_s + z_u)^2}/c$
Modify the angle Ω_{er}
$\Omega_{er} = \Omega_{er} - \dot{\Omega}_{ie} t_t$
Calculate satellite position
Equ 4.47
Error thresholding
NO
YES
Satellite position estimated at the time of receiving

Figure 4.6: Iteration of estimated receiver position and clock bias and calculated satellites positions

section, the filter structure is discussed. The state space representations of both techniques are listed in appendix E.

4.4.1 Loosely and Tightly Coupled Configurations

In the loosely coupled configuration or decentralized filtering, the navigation solution estimated from the GPS receiver is blinded with the solution estimated from the INS. This configuration is sometimes referred to as cascaded configuration, since position and velocity measurements provided by the GPS receiver are in the first place estimates of a first stage filtering that is implemented within the receiver. This cascaded configuration results in degradation in accuracy when the number of viewed satellites drops bellow four or the satellites geometry becomes poor (Weiss, J., D. & Kee, D., S., 2001)(George, M. G., & Sukkarieh, S., 2005)(George, M. G., 2007).

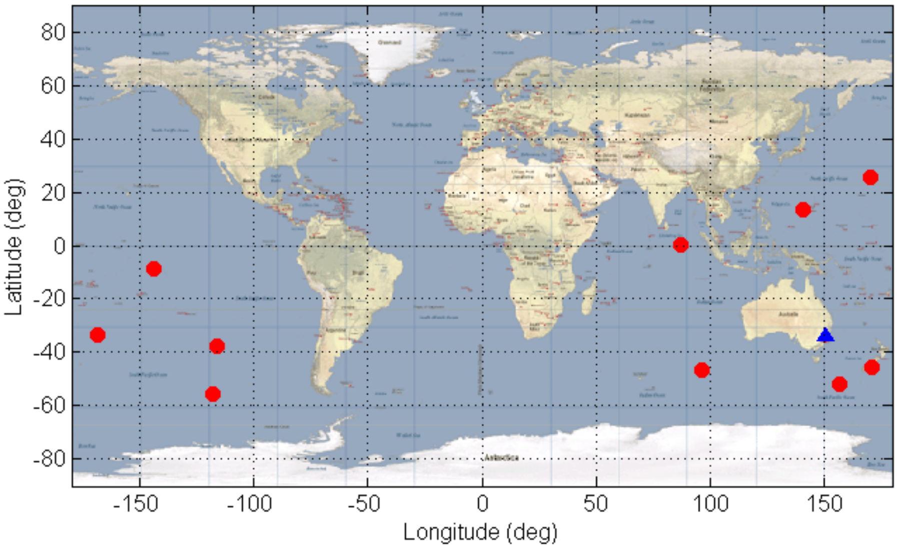


Figure 4.7: Calculated satellites positions (red dots) and estimated receiver position (blue triangle)

The tightly coupled configuration reduces the influence of the number of satellites and their geometry on the integration performance by operating directly on the

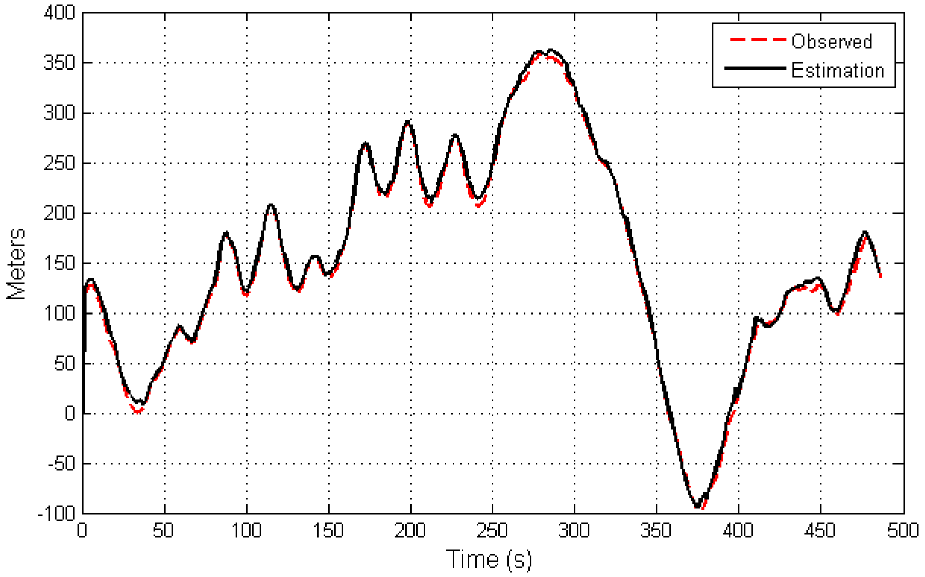


Figure 4.8: Estimated receiver clock bias (solid line) and receivers calculated clock bias (dashed line)

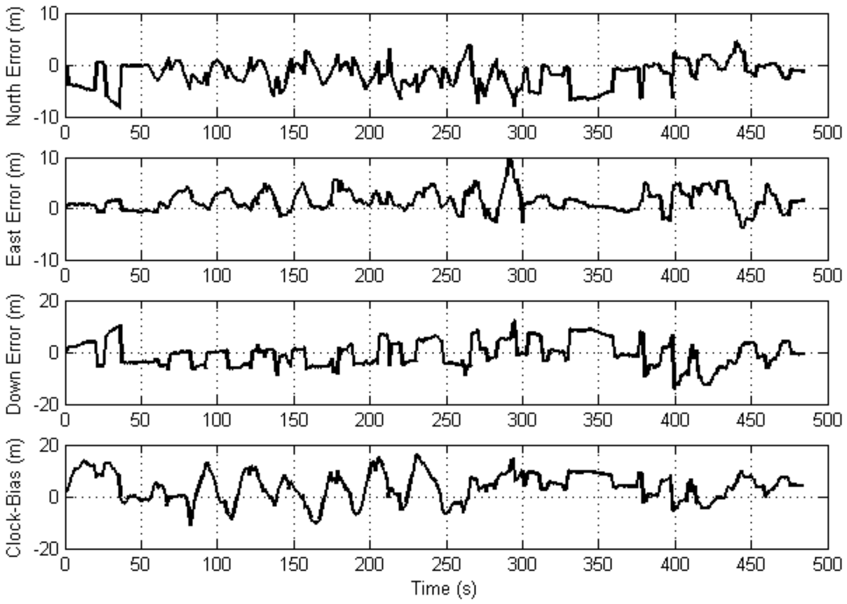


Figure 4.9: Differences between estimated receiver position and clock bias and the calculated ones by the GPS receiver

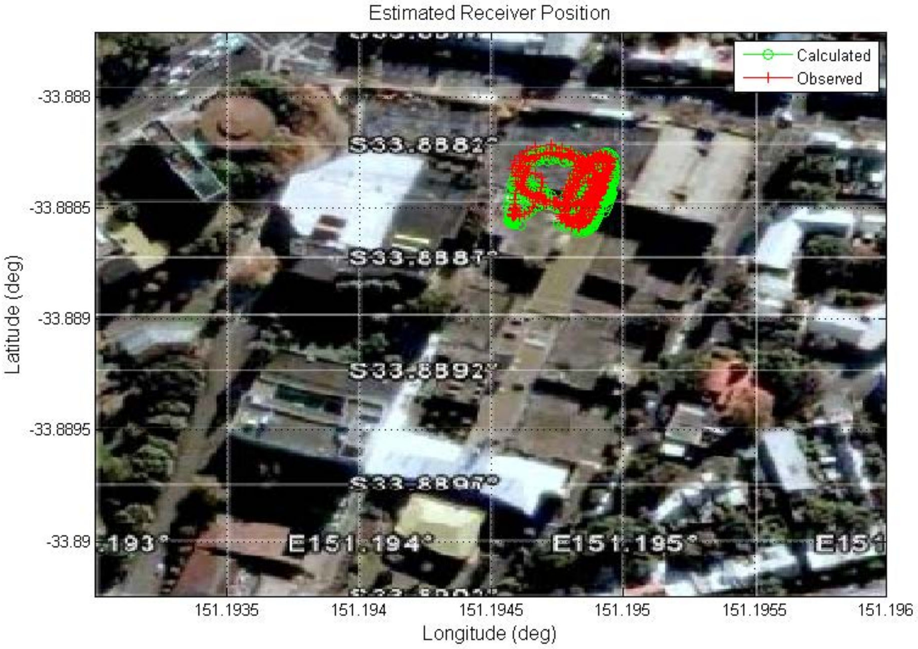


Figure 4.10: Estimated and measured receiver position

raw measurement of ranges (and delta ranges) between the receiver and GPS satellites, and blend these measurements with INS estimates to form innovations used in the update stage. A tightly coupled configuration has only one central Kalman filter, so that it is referred to as centralized filtering. (Fig. 4.11) shows both configurations in term of observations and how innovations are calculated, while Table 4.3 shows how innovations are formulated for both loosely and tightly coupled integration techniques. It also compares between both linear and nonlinear filter structures. The detailed state space representation of both techniques is tabulated in appendix E.

For our purpose, only loosely-coupled configuration is considered and lever arm is to be considered insignificant in small size vehicles. For more details on lever arm compensation and level arm error state estimation refer to (Abdel-Hafez, M., 2003) and (Hong, S., Lee, M., H., Chun, H., H., Kwon, S., H. & Speyer, J., L., 2006), for tightly coupled implementation using the quaternion approach refer to (Abdel-Hafez, M., 2003) and for the DCM approach refer to (George, M. G., 2007)(George, M. G., & Sukkarieh, S., 2005).

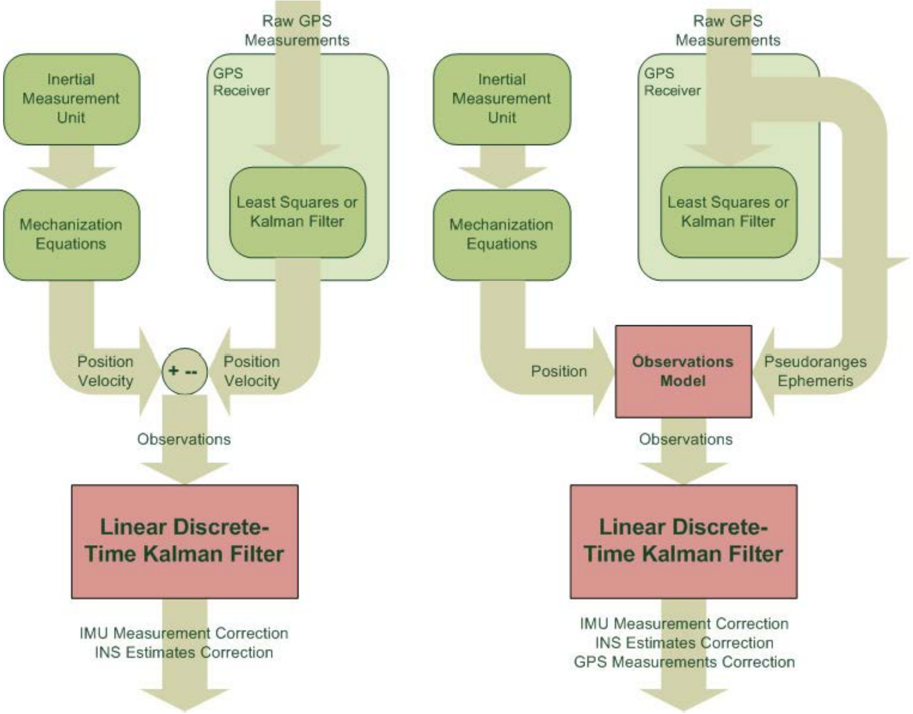


Figure 4.11: The major difference between loosely and tightly coupled configuration is in term of how the innovation is computed

Table 4.3: Different Kalman filter implementations

	Loosely-coupled	Tightly-coupled
Linear	$\delta\hat{\mathbf{x}}_{kf}^{m} = \delta\mathbf{x}_{ins}^{m} + \mathbf{W}[\mathbf{z} - \mathbf{H}\delta\mathbf{x}_{ins}^{m}]$ $\mathbf{z} = \begin{pmatrix} \mathbf{r}_{gps}^{m} - \mathbf{r}_{ins}^{m} \\ \mathbf{v}_{gps}^{m} - \mathbf{v}_{ins}^{m} \end{pmatrix}$	$\delta\hat{\mathbf{x}}_{kf}^{ecef} = \delta\mathbf{x}_{ins}^{ecef} + \mathbf{W}[\delta\rho^{ecef} - \mathbf{H}\delta\mathbf{x}_{ins}^{ecef}]$ $\delta\rho^{ecef} = \rho_{meas}^{ecef} - \rho_{pred}^{ecef}$
Non-linear	$\hat{\mathbf{x}}_{kf}^{m} = \mathbf{x}_{ins}^{m} + \mathbf{W}[\mathbf{z} - h(\mathbf{x}_{ins}^{m})]$ $\mathbf{z} = \begin{pmatrix} \mathbf{r}_{gps}^{m} \\ \mathbf{v}_{gps}^{m} \end{pmatrix}$	$\hat{\mathbf{x}}_{kf}^{ecef} = \mathbf{x}_{ins}^{ecef} + \mathbf{W}[\rho_{meas}^{ecef} - \rho_{pred}^{ecef}]$ $\rho_{pred}^{ecef} = \lvert\mathbf{r}_{sat}^{ecef} - \mathbf{r}_{ins}^{ecef}\rvert$

4.4.2 Discretization

The Jacobian matrix $\mathbf{J}_f(k)$ at time t_k in discrete time is calculated as

$$\begin{aligned}\mathbf{J}_{\mathbf{F}}(k) &= \frac{\partial \mathbf{F}_k(x_k)}{\partial x_k} \\ &\approx \mathbf{I} + \frac{\partial \mathbf{F}_k(t_k, x_{t_k})}{\partial x_k} dt \\ &= \mathbf{I} + \mathbf{J}_c(t_k) dt\end{aligned} \tag{4.20}$$

where $\mathbf{J}_c(t_k)$ is the continuous time jacobian matrix at time t_k. It follows from equation 4.20 that, the discrete time transition matrix is computed by

$$\mathbf{F}(k) = \mathbf{I} + \mathbf{F}dt + \frac{(\mathbf{F}dt)^2}{2!} + \cdots \tag{4.21}$$

While the discrete-time process noise covariance matrix is computed by (Sukkarieh, S., 2000)

$$\mathbf{Q}(k) = \frac{1}{2}\left[\mathbf{F}(k)\mathbf{G}(k)\mathbf{Q}_c(k)\mathbf{G}(k)^T\mathbf{F}(k)^T + \mathbf{G}(k)\mathbf{Q}_c(k)\mathbf{G}(k)^T\right] dt \tag{4.22}$$

4.5 Performance Verification - Simulations

This section presents a series of simulated results to verify the performance of the navigation filter. The performance of the estimator is analyzed in terms of filter tuning and limited satellite availability. The simulated trajectory and sensor readings were generated by the Aerosondi model in MATLAB. The first part of this section deals with simulation environment. It highlights aircraft control commands used to generate the path and sensors noise characteristics. Next, filter tuning results are presented to emphasize the discussion. Finally, a GPS outage is simulated.

4.5.1 Simulation Environment

In this simulation, the "Crossbow" IMU is considered. Spectral noise, given by equations 3.60 and 3.61, is used to simulate the IMU. As shown earlier is section 3.3.1, this noise model contains constant bias component, random noise component and random walk. Figure 4.12 shows the bank angle control command applied to simulate the trajectory of figure 4.13.

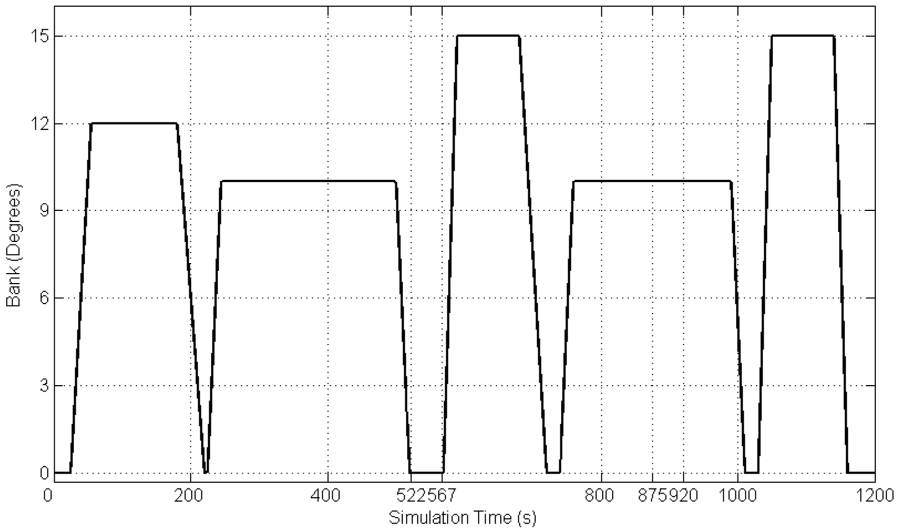


Figure 4.12: The bank angle command applied to simulate the trajectory. The simulated trajectory undergoes five rotations all to the right.

The trajectory contains a series of rotations at constant altitude as represented by figure 4.14. The rotations are all to the right, which resembles a typical trajectory that occurs in typical UAV missions. In subsection 4.5.3 limited satellite availability is considered for two different scenarios. First, while travelling in straight lime path between 522 – 567 seconds. Then, while steady turn between 875 – 920 second. The motivation behind the selection of such maneuvers is to highlight the rule of uncertainty growth in the mitigation of GPS failures. This rule will be discussed later in chapter 5. Table 4.4 shows parameters used for filter initialization and starting the tuning process.

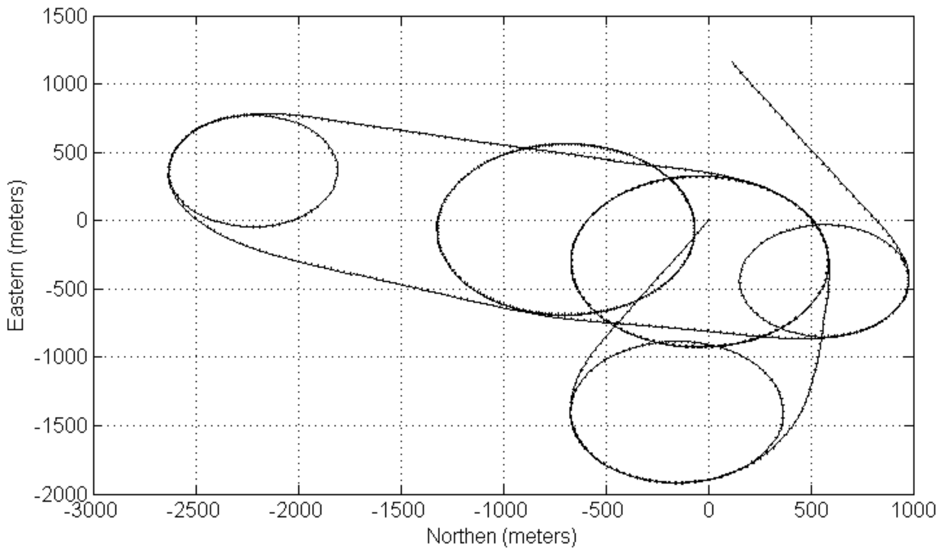


Figure 4.13: The simulated trajectory developed for algorithm validation. The dashed line represent the truth, the dotted line represents the GPS position, and the solid one represents the INS/GPS solution.

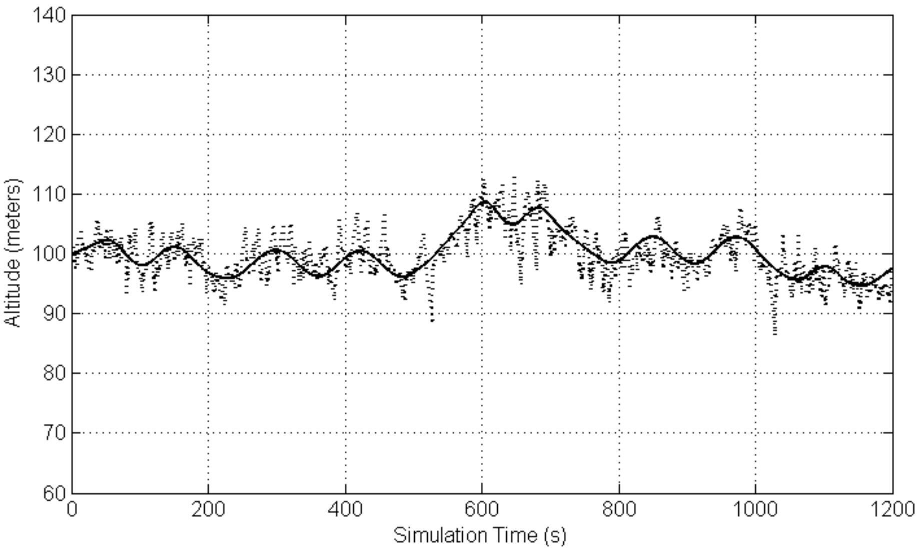


Figure 4.14: Vehicle altitude where the solid line represent the truth, the dashed line represents the INS/GPS solution.

4.5.2 Simulation - Filter Tuning

As indicated earlier in subsection 2.3.1, innovation whiteness and unbiasedness tests are key considerations in the validation of the Kalman filter model. Figure 4.15 shows the NIS (or χ^2) consistency test. The moving average is within the 95% confidence limits, depicting the fact that the computed innovation covariance is in consistence with position and velocity innovations. This may also be proved by just looking into the innovation and its corresponding covariance. Figure 4.16 and 4.17 represent position and velocity innovation respectively. To test for whiteness, the autocorrelations of the position and velocity innovations are constructed in figures 4.18 and 4.19, respectively. The two figures depict the fact that the innovation sequences are uncorrelated. The three listed tests prove the fact that under normal (no-failure) mode, the innovation sequence is zero-mean, white and has covariance of $\mathbf{S}(k)$.

Table 4.4: Simulated Kalman filter parameters

Parameter	**Value**	**Unit**	**Description**
$\sigma_{\mathbf{f}^b}$	$(0.1, 0.1, 0.1)$	$m/s/\sqrt{hr}$	Accelerometers random walk
σ_{ω^b}	$(2.0, 2.0, 2.0)$	$deg/\sqrt{hr}$	Rate gyros random walk
$\sigma_{\Delta\mathbf{f}^b}$	$(0.56, 0.62, 0.58)$	mg	Accelerometers short term bias stability
$\sigma_{\Delta\omega^b}$	$(0.29, 0.30, 0.28)$	deg/s	Rate gyros short term bias stability
$\sigma_{\mathbf{r}_0}$	$(0.5, 0.5, 0.5)$	m	Initial position uncertainty
$\sigma_{\mathbf{v}_0}$	$(0.5, 0.5, 0.5)$	m/s	Initial velocity uncertainty
$\sigma_{\mathbf{\Psi}_0}$	$(0.01, 0.01, 0.01)$	deg	Initial attitude uncertainty
$\sigma_{\mathbf{r}_{gps}}$	$(2.0, 2.0, 3.0)$	m	GPS receiver position observations error
$\sigma_{\mathbf{v}_{gps}}$	$(1.0, 1.0, 2.0)$	m/s	GPS receiver velocity observations error

Figures 4.20, 4.21 and 4.22 show position, velocity and attitude errors respectively. The uncertainty in estimated position and velocity errors represent the state accurately, while this is not the case for attitude errors where observability (Abdel-Hafez, M., F., 2006), which depends on vehicle maneuvers, affects the update of

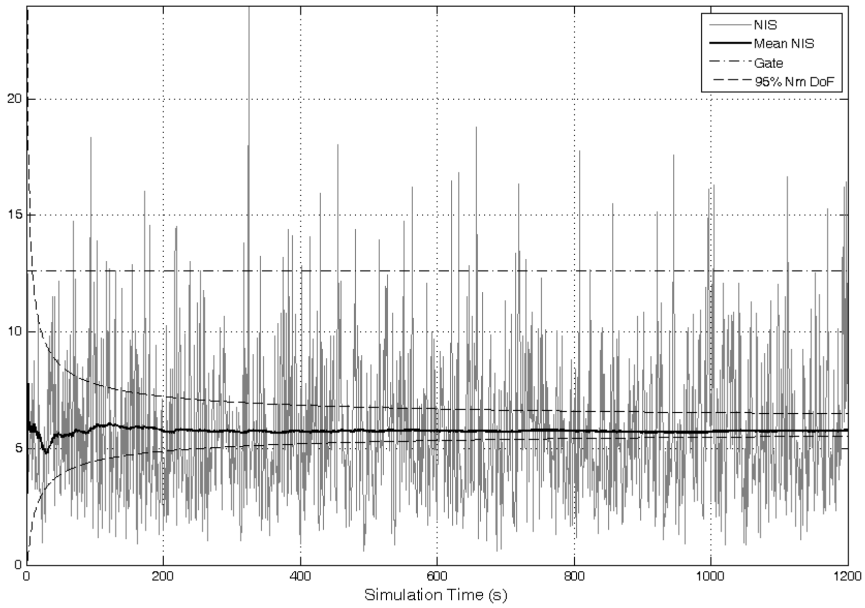


Figure 4.15: The χ^2 test proves the consistency of the innovation sequence with its covariance; the moving average is within the 95% confidence interval

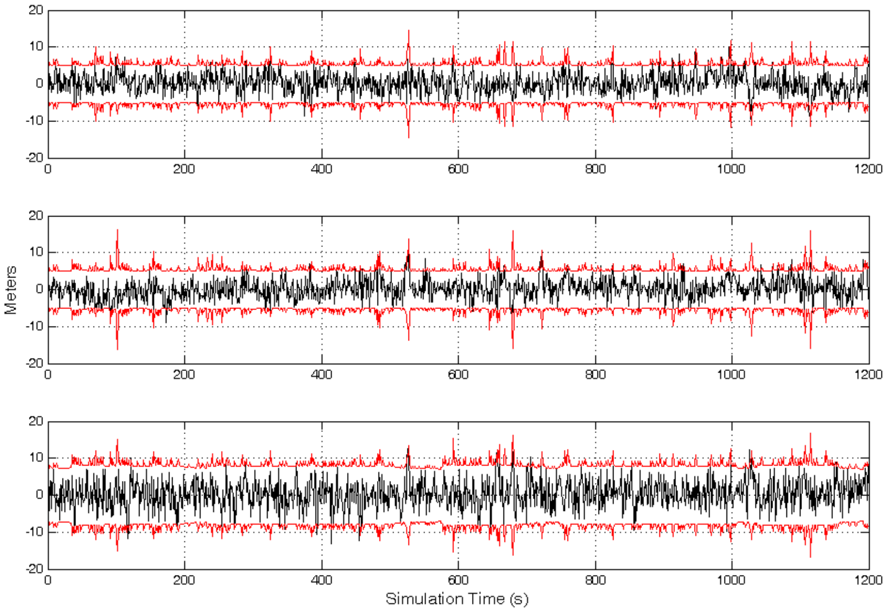


Figure 4.16: Position innovations along with its 2σ uncertainty

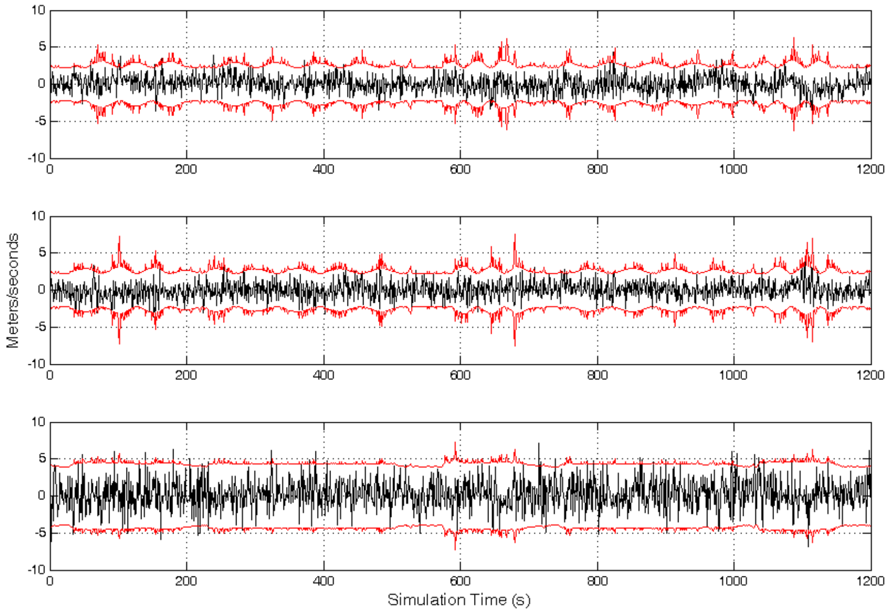


Figure 4.17: Velocity innovations along with its 2σ uncertainty

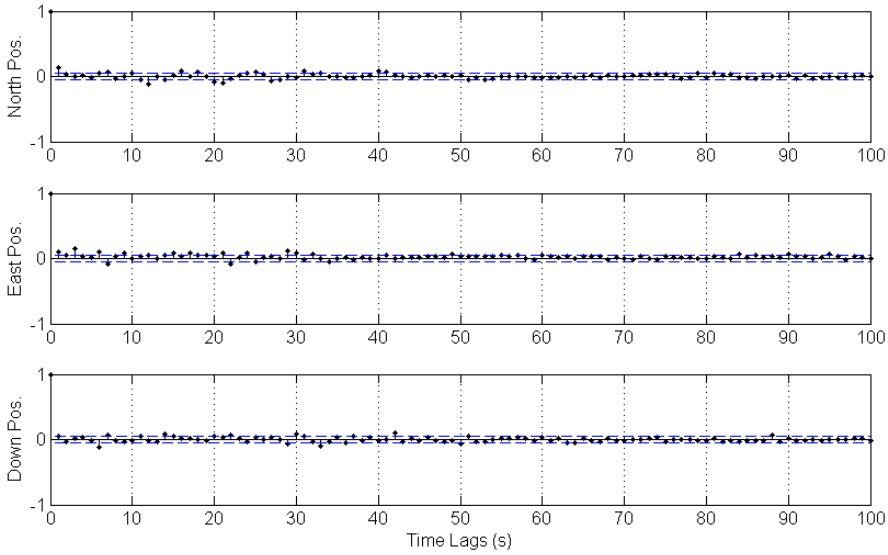


Figure 4.18: Since less than 5% of the position innovations autocorrelation function components lie outside the confidence line, the position innovations are considered uncorrelated and the filter is optimal.

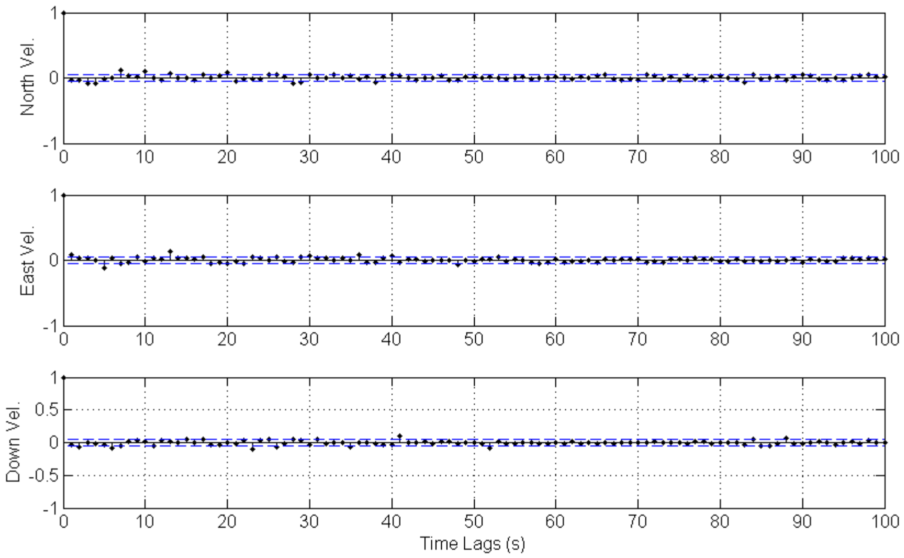


Figure 4.19: Similar to the position innovations, less than 5% of the velocity innovations autocorrelation function components lie outside the confidence line. The velocity innovations are considered uncorrelated and the filter is optimal.

attitude uncertainty specially while the simulated aircraft turns with constant bank angle or travels in straight line with no change in accelerations. Moreover, the growth of position and velocity uncertainty shows some short spikes. This is due to jumps in GPS fixes being detected. The detection of such jumps was implemented using the χ^2 gating function. The implementation of such a technique will be discussed in detail in chapter 5.

Finally, to show the GPS/INS integration corrects INS errors, figures 4.23,4.24 and 4.25 show position, velocity and attitude errors both when stand alone IMU navigation is considered and when fusing IMU predictions with GPS observations.

4.5.3 Simulation - Effect of Limited Satellites Availability

In this subsection, a GPS outage is simulated to show its effect on the Kalman filter estimates. As indicated earlier, the outage is simulated during two periods of 45 seconds. First, while travelling in straight line and then while steady turn. Figures

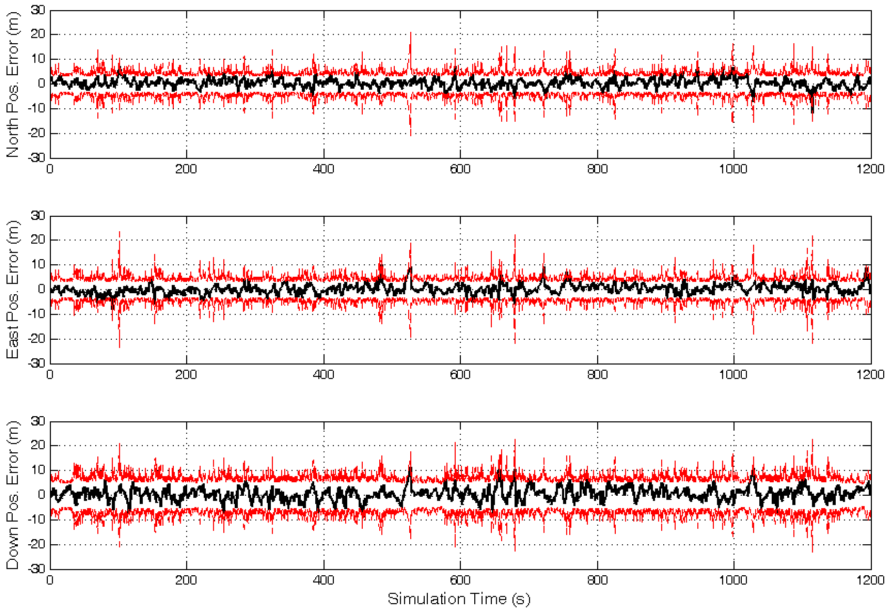


Figure 4.20: Simulated INS/GPS position errors along with its 3σ uncertainty

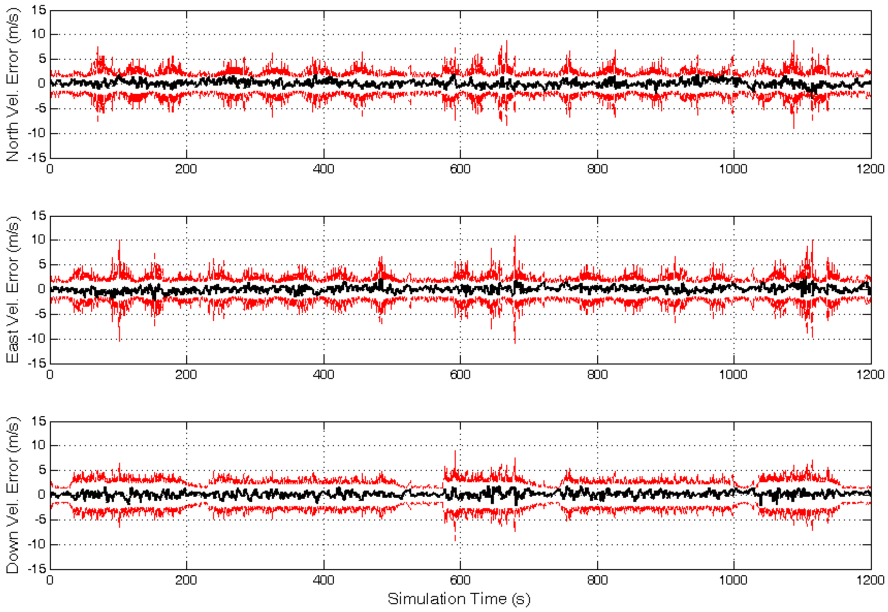


Figure 4.21: Simulated INS/GPS velocity errors along with its 3σ uncertainty

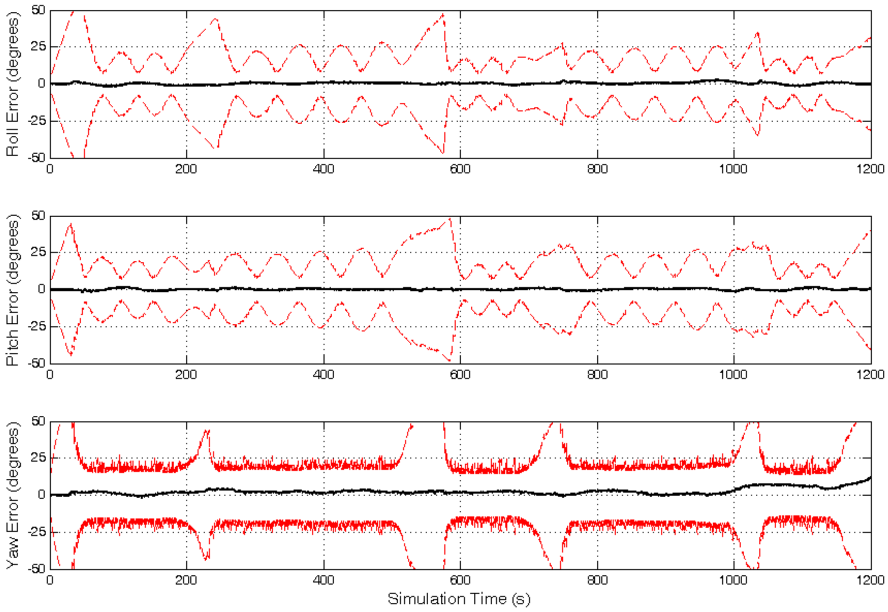


Figure 4.22: Simulated INS/GPS attitude errors along with its 3σ uncertainty

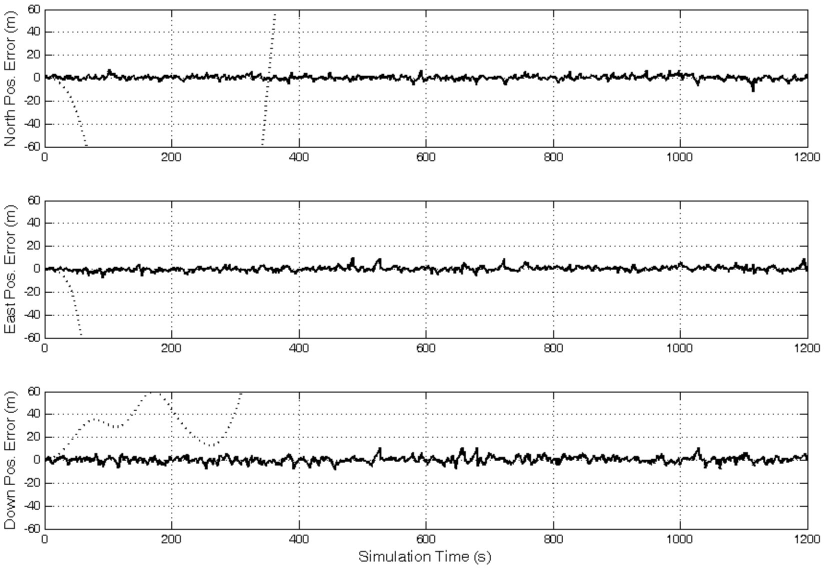


Figure 4.23: Position errors when INS is operating alone (dotted line) versus when aided by GPS fixes (solid line).

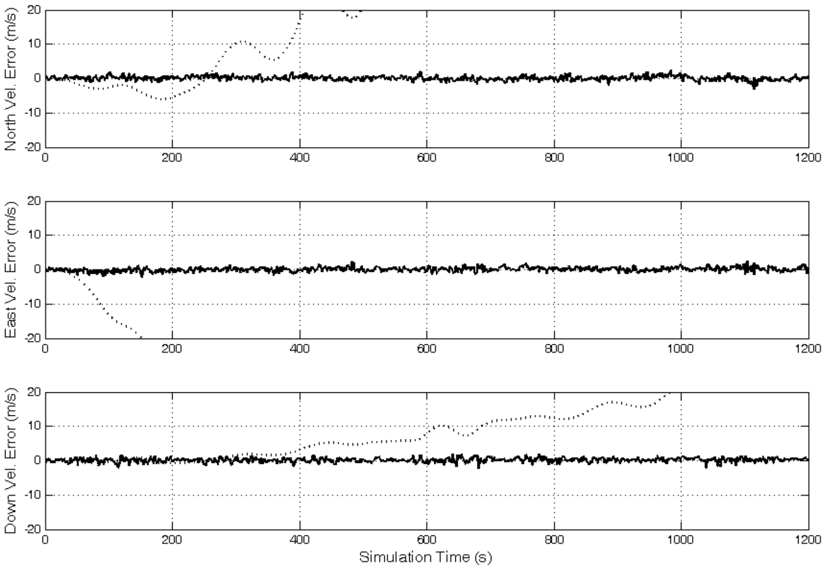


Figure 4.24: Velocity errors when INS is operating alone (dotted line) versus when aided by GPS fixes (solid line).

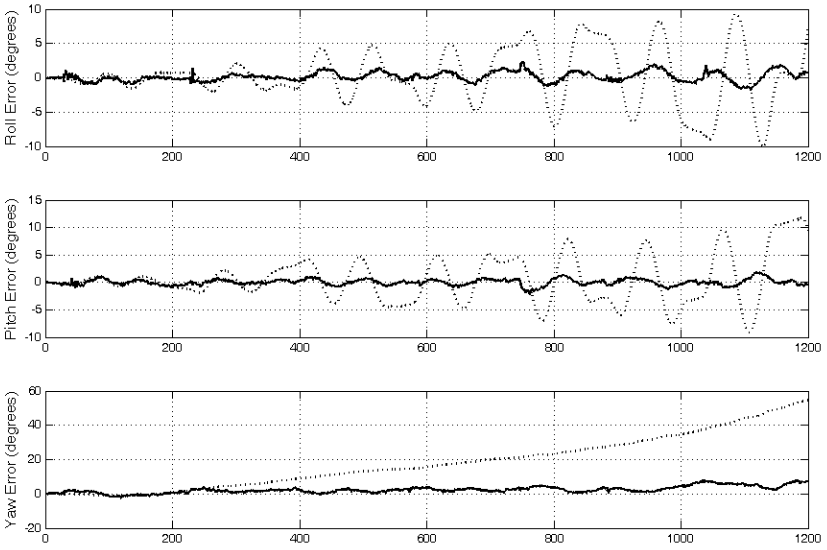


Figure 4.25: Attitude errors when INS is operating alone (dotted line) versus when aided by GPS fixes (solid line).

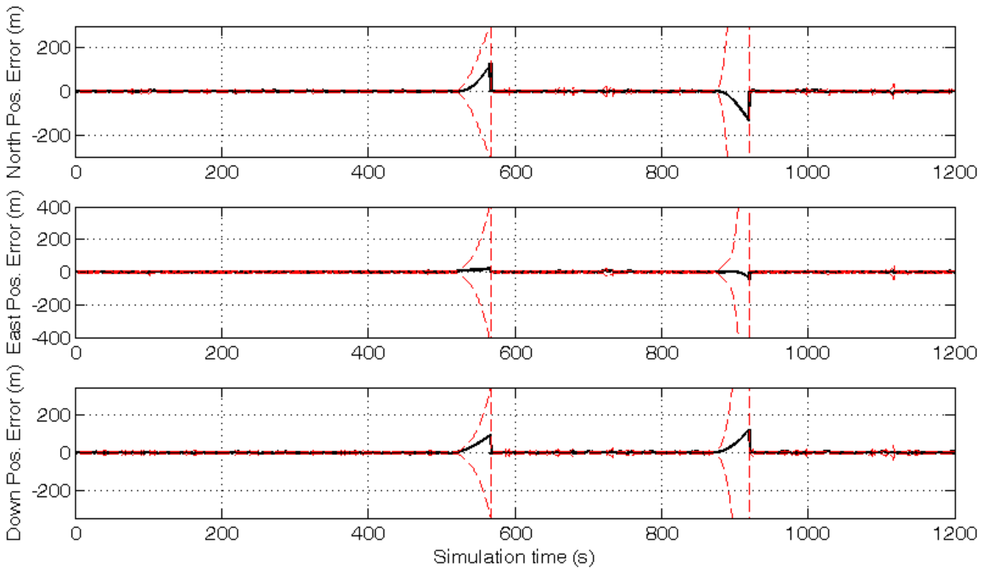


Figure 4.26: Position errors of the loosely-coupled filter showing two intervals of GPS outage. The simulated aircraft underwent the first low-availability between 522-567 seconds while travelling in straight line, and then between 875-920 while steady turn. During GPS outages, and disregarding the vehicle maneuvering, the uncertainty of the error explodes and the estimator is on INS alone. However, this effect is bigger in the case while turning.

4.26, 4.27, and 4.28 show position, velocity and attitude errors, respectively.

4.6 Performance Verification - Land Vehicle Test

After simulation has been used to validate the performance of the algorithm, it is used in this section to process actual IMU-GPS data. Data was logged using a MATLAB code running on a laptop. The inertial measurement unit utilized in the test was the ”Crossbow” IMU shown in figure 4.29, with the internal inertial sensors shown in figures 4.31 and 4.32, while the GPS receiver used is the ”Novatel Smart Antenna”. This receiver is rugged and accurate, it has a less than 5 meters position accuracy and less than 0.05 m/s velocity accuracy. In the following subsections, two actual tests results are represented. The utility and the platform are shown in figures 4.34 and 4.33, respectively. As indicated earlier in subsection A.2, when DCM approach

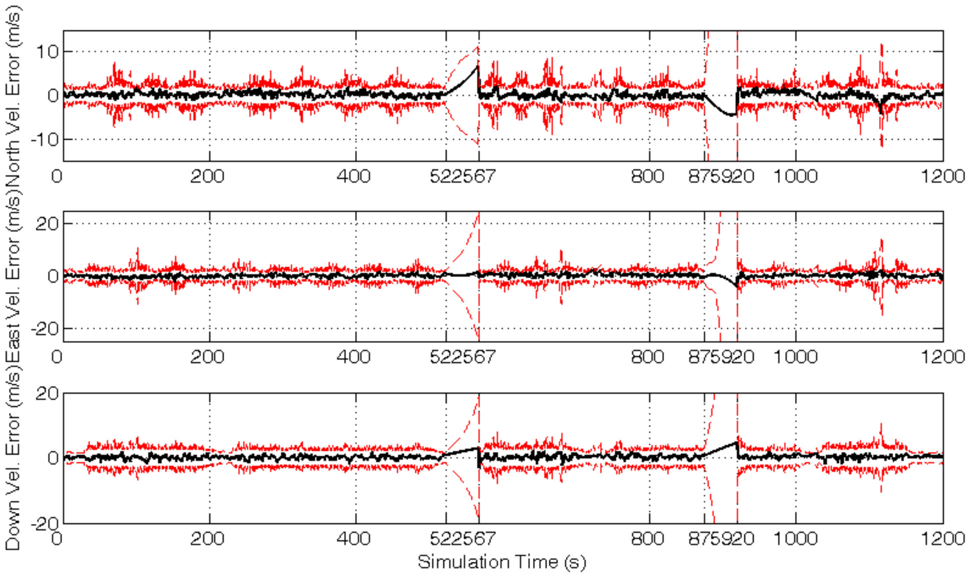


Figure 4.27: Velocity errors show similar behavior to that of position errors due to GPS outage.

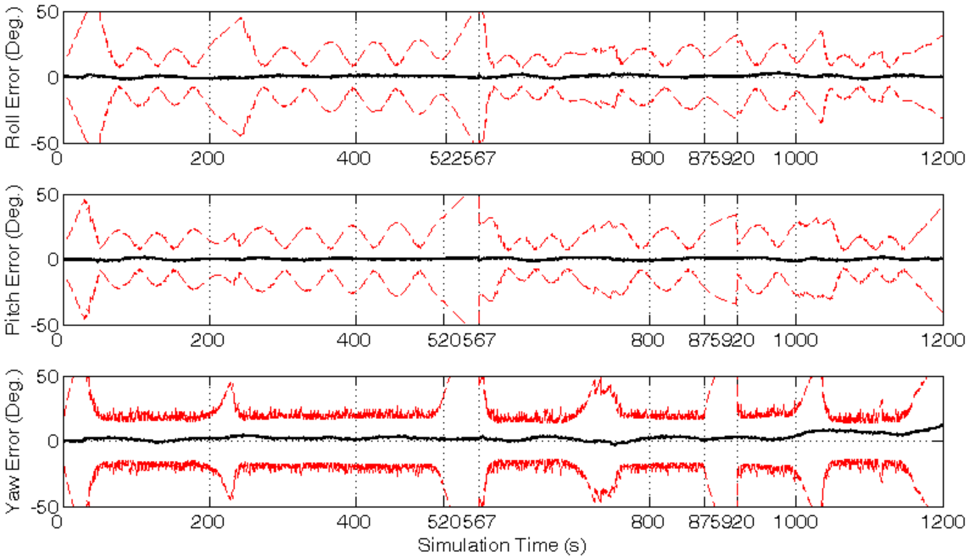


Figure 4.28: Attitude errors show similar behavior due to that of both position and attitude errors due to GPS outage.

was discussed, the initial orientation of the vehicle is required to calculate the initial DCM. Therefore, a digital compass is used to get the initial orientation (Nebot, E. M., & Durrant-Whyte, H. F., 1999).

At the beginning of the test, the components are allowed to worm up for few seconds. IMU measurements are used to calculate the values of inertial sensors biases (Nebot, E. M., & Durrant-Whyte, H. F., 1999), while GPS receiver position and velocity measurements are used to initialize the SDINS algorithm.

Figure 4.29: The "Crossbow" IMU400CD-100 utilized in this thesis.

Figure 4.30: The Novatel "Smart Antenna" GPS utilized in this thesis. The position measurement accuracy of this rec

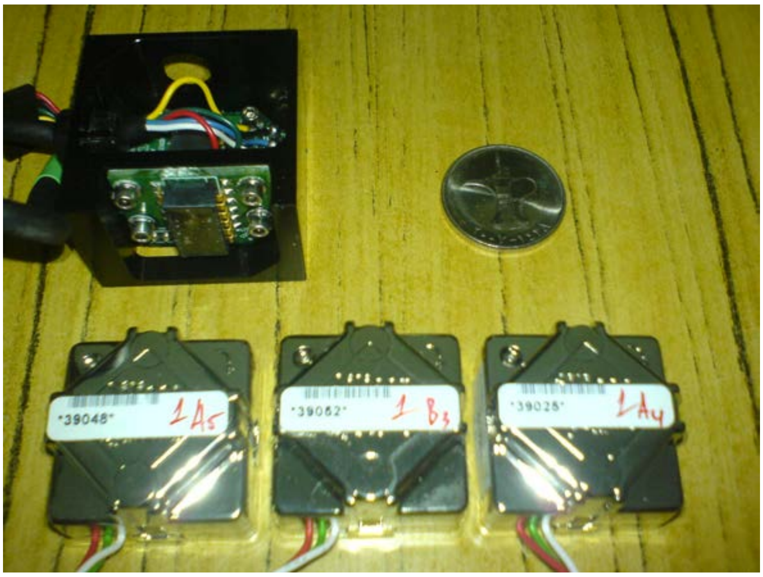


Figure 4.31: Sensors utilized in this thesis. In the upper left corner appear two dual-axis accelerometers (±4g each) and below appear three single-axis gyroscopes (±100°/h each).

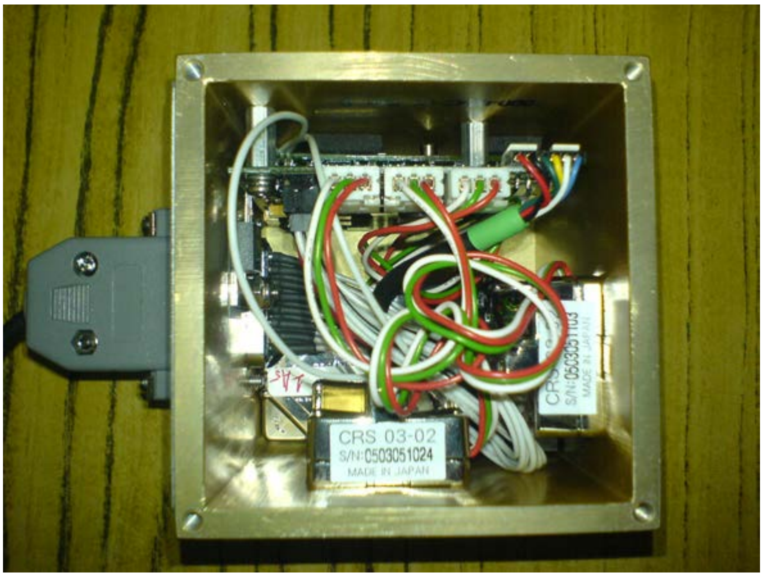


Figure 4.32: Three orthogonal accelerometers and three orthogonal gyroscopes assembly in "Crossbow" IMU utilized in this thesis. It provides measurements at maximum sample rate of 133 sample per second.

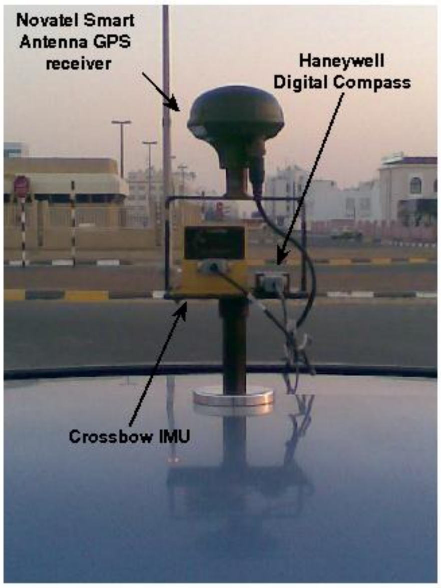


Figure 4.33: The INS/GPS platform used in the field test. It contains a 6DoF IMU, a GPS receiver, and a digital compass. The digital compass is only used for initialization.

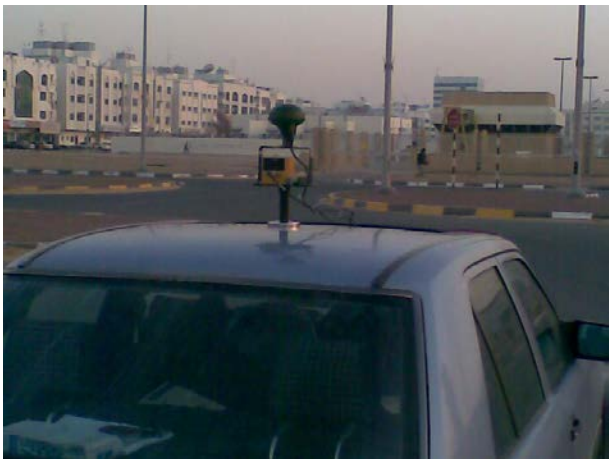


Figure 4.34: The vehicle utilized in the test with the platform mounted on its roof. Since the vehicle is considered as a small size vehicle, then the lever arm between the location of the IMU and the center of the vehicle gravity is neglected.

4.6.1 Test No.1

Figures 4.35 and 4.36 show three accelerometers and three gyroscopes measurements, respectively. Position innovations are shown in figures 4.37 and velocity innovations are in figure 4.38. Figure 4.39 shows the test trajectory represented by GPS observations and INS/GPS estimations. Figure 4.40 highlights the performance of the filter during rotational motion.

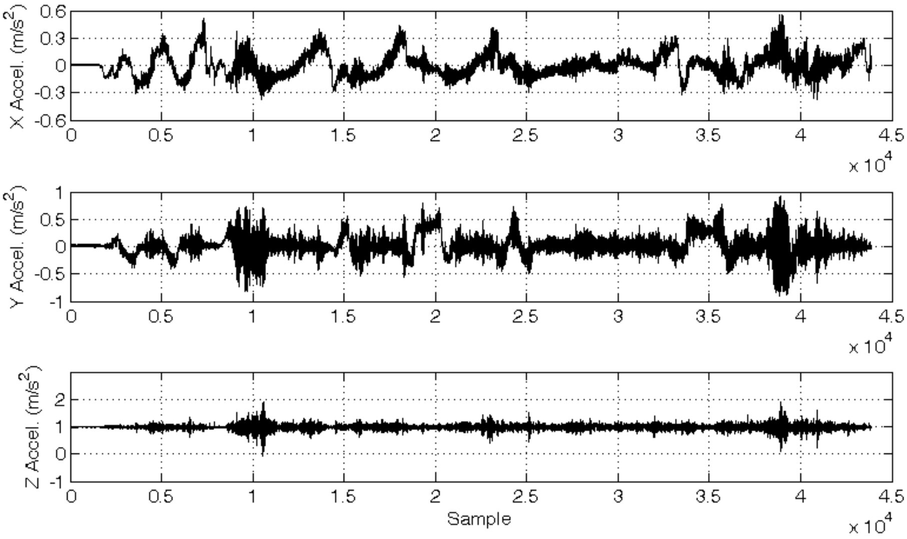


Figure 4.35: Accelerations measurements of three orthogonal accelerometers in the "Crossbow" IMU

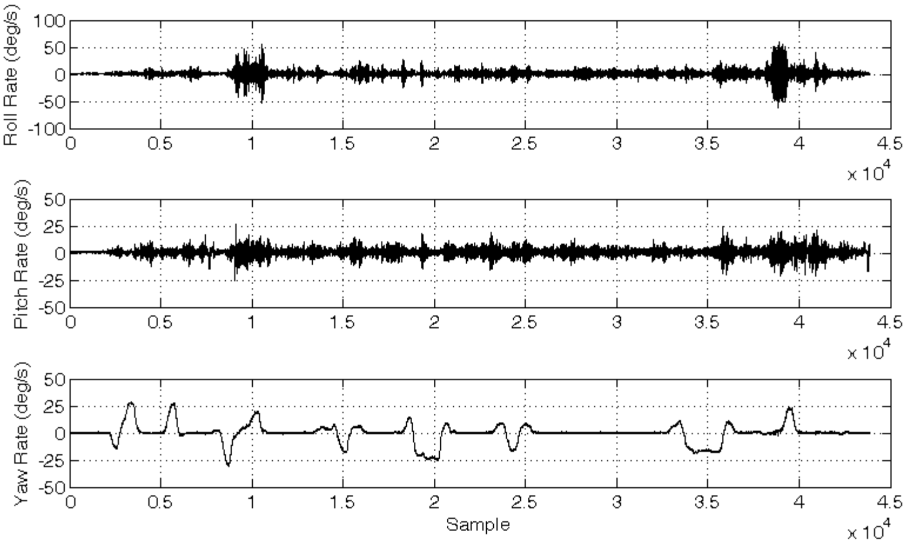


Figure 4.36: Angular rotations measurements of three orthogonal gyros in the ”Crossbow” IMU

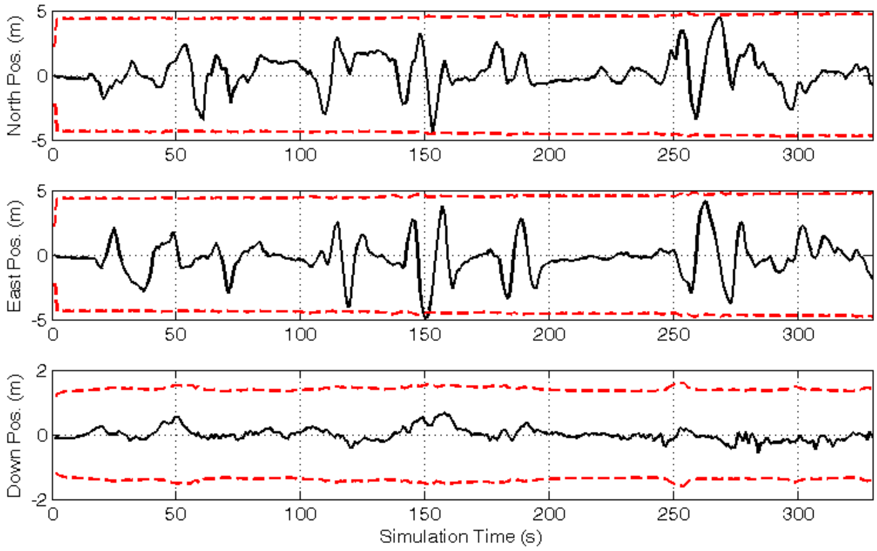


Figure 4.37: Position innovations showing a correlated behavior. This is due to incorrect IMU sampling time, since it was assumed constant, and mistuning in process and observation noise covariance matrices.

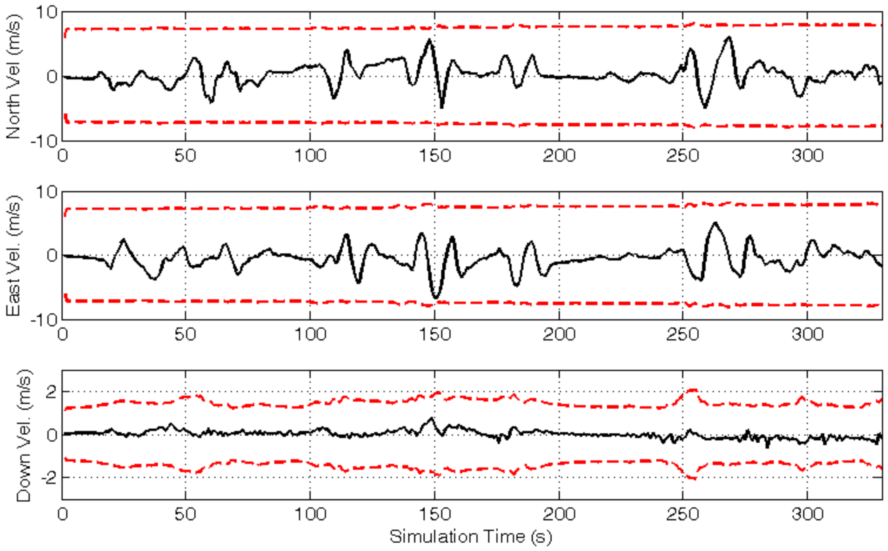


Figure 4.38: Similar to position innovations, velocity innovations show a correlated behavior.

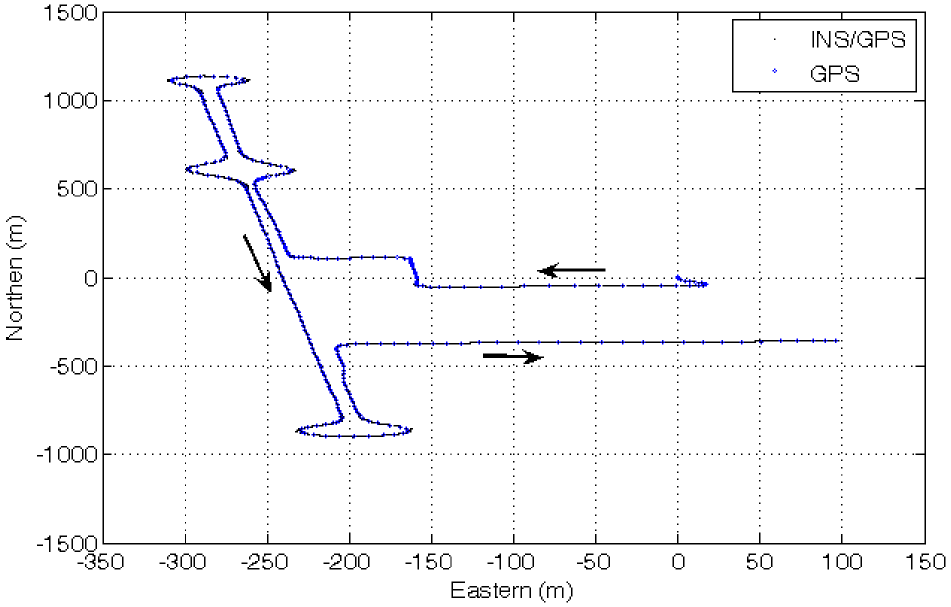


Figure 4.39: Real test trajectory representing GPS position (+) and INS-GPS solution (solid).

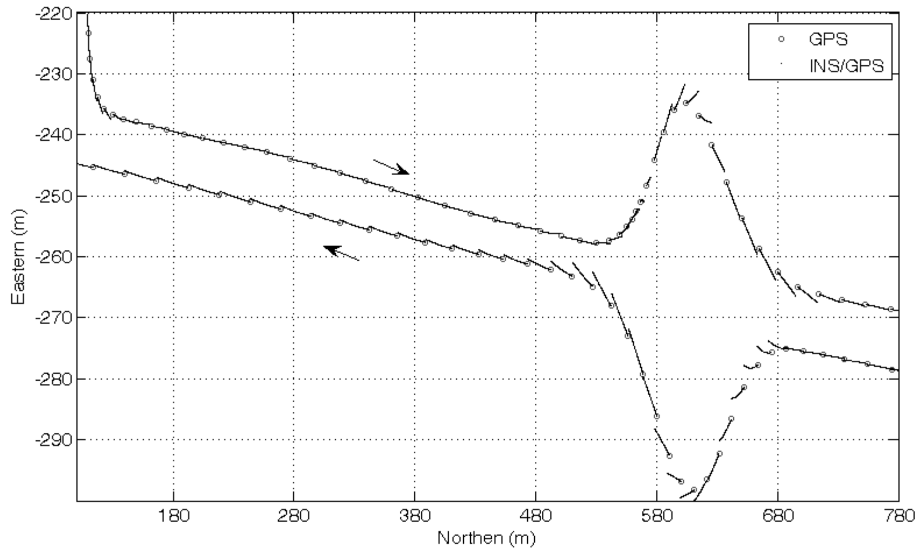


Figure 4.40: The loosely-coupled linear filter shows a poor performance at corners. The fact that, due to the nature of the filter where attitude and bias estimates are zeros during prediction, and are only available once GPS becomes available, the filter estimates deviate and this deviation becomes significant at corners. Moreover, incorrect IMU sample time and mistuning caused incorrect mechanisation which magnified this effect.

4.6.2 Test No.2

In this test, different trajectory was selected. Figures 4.41 and 4.42 show measurements of the three accelerometers and three gyroscopes, respectively. Position and velocity innovations are shown in figures 4.43 and 4.44, respectively. Finally, the tested trajectory is represented in figure 4.45 by the GPS fixes and the INS/GPS navigation filter solution.

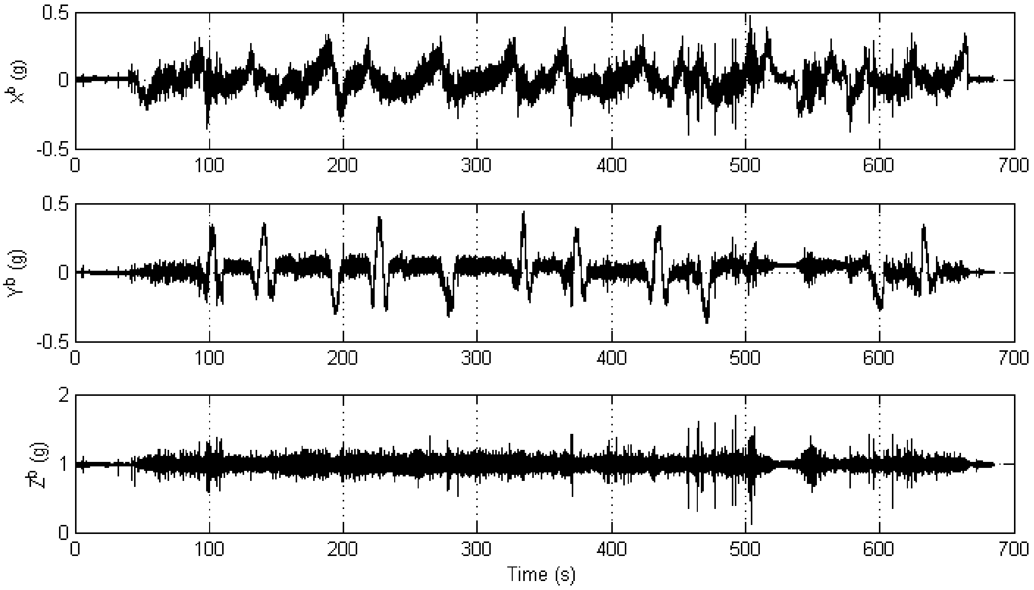


Figure 4.41: Accelerations measurements of three orthogonal accelerometers in the "Crossbow" IMU

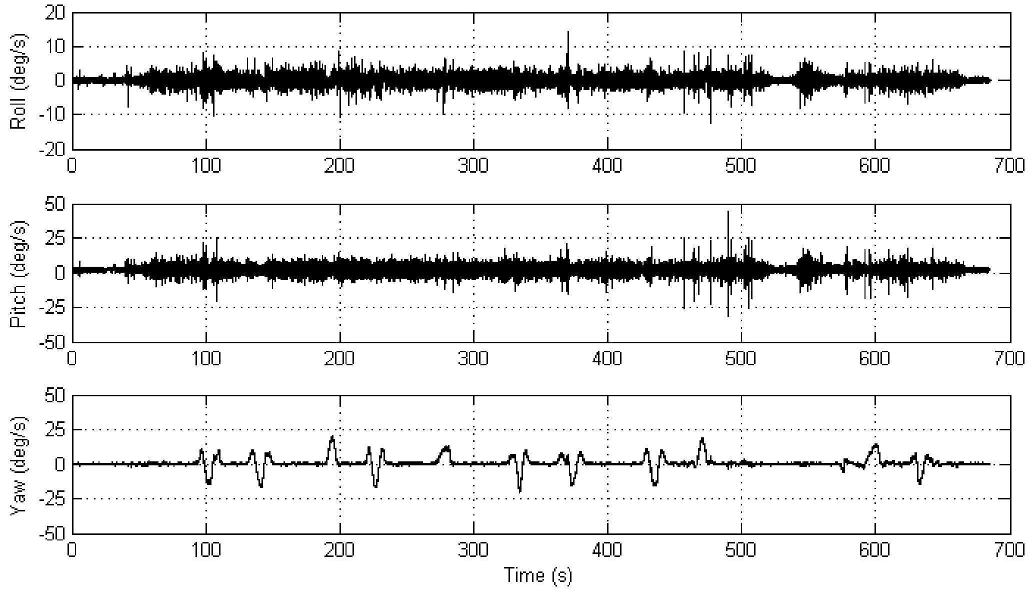


Figure 4.42: Angular rotations measurements of three orthogonal gyros in the "Crossbow" IMU

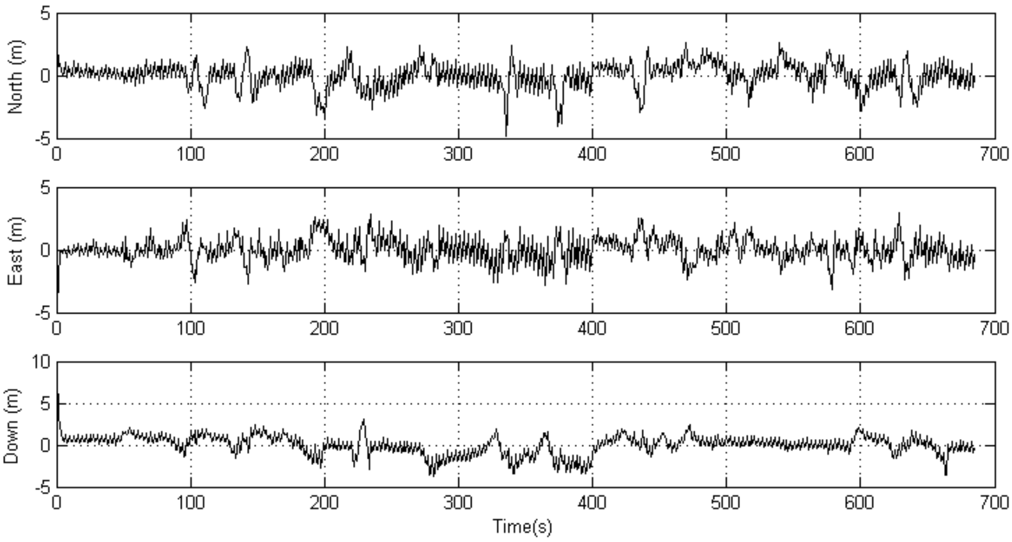


Figure 4.43: Unlike the first test, the position innovations this time resemble white noise behavior. This is because the IMU data was logged using different software and the sample time used in INS mechanisation is accurate.

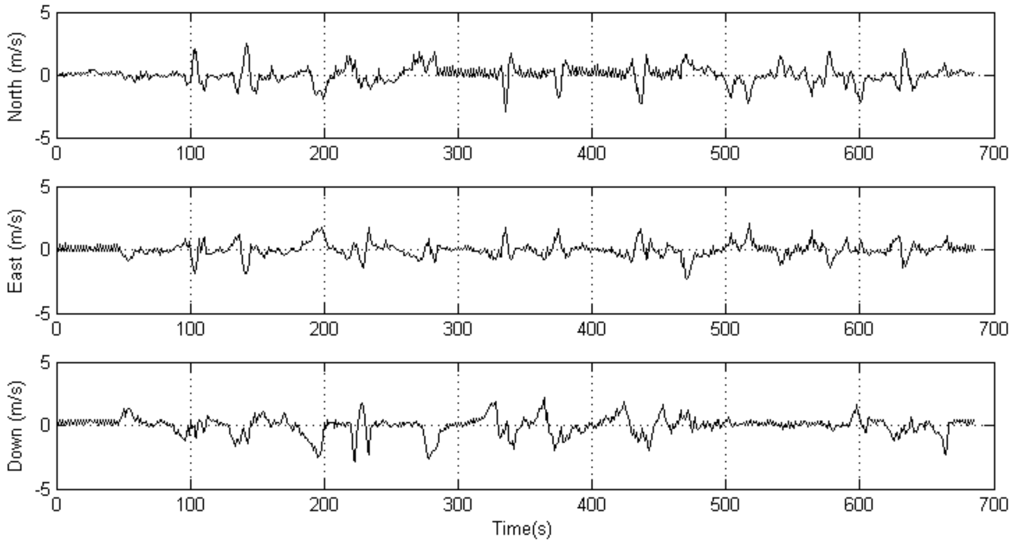


Figure 4.44: Similar to position innovations, velocity innovations resemble white noise.

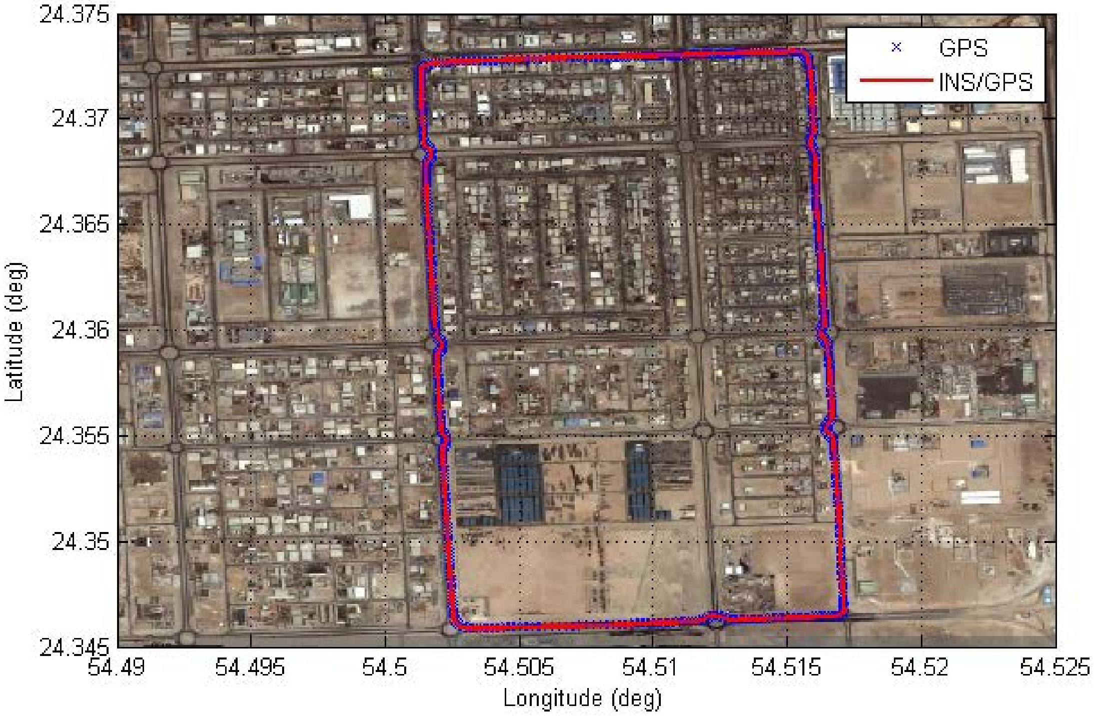


Figure 4.45: Actual test trajectory representing GPS position (+) and INS-GPS solution (solid)

4.6.3 Test No.3

The third test was conducted in Sheikh-Zayed Seaport. The test trajectory is shown on the map in figure 4.46. The filter estimated trajectory is shown in figure 4.47. Position and velocity innovations are shown in figures 4.48 and 4.49, respectively, with corresponding 2σ uncertainty boundaries. Position and velocity autocorrelation functions are represented by figures 4.50 and 4.51, respectively. These figures highlight the operation of the filter, and verify its performance. Finally, figure 4.52 shows the effect of correct filter tuning and IMU sampling time on SDINS mechanisation and Kalman filter operation compared to previous test represented by figure 4.40.

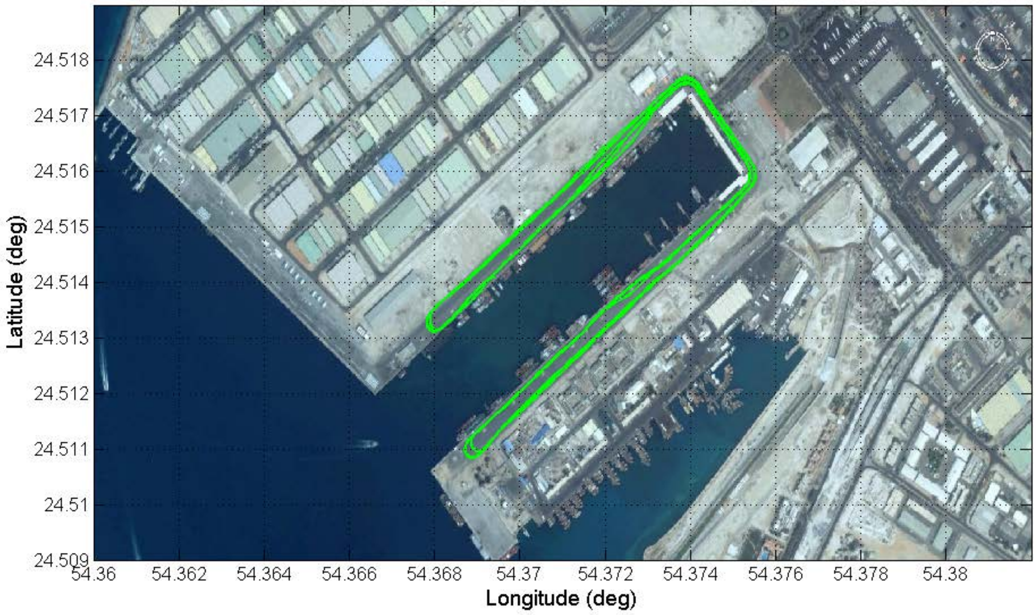


Figure 4.46: Sheikh-Zayed seaport where land vehicle test was conducted. The test trajectory contained three complete loops with approximately 4Km length each.

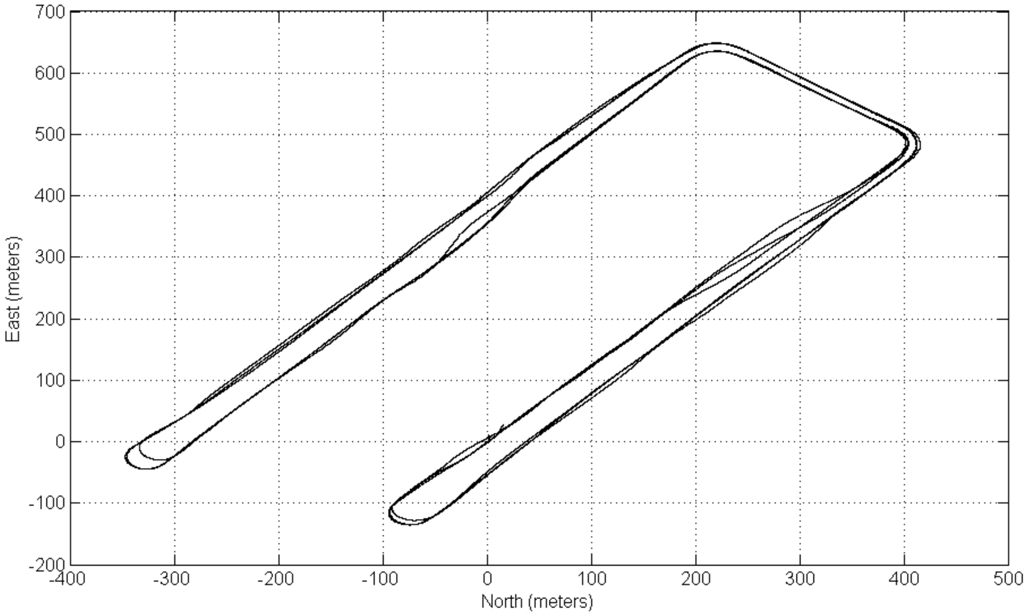


Figure 4.47: Estimated SDINS/GPS vehicle trajectory

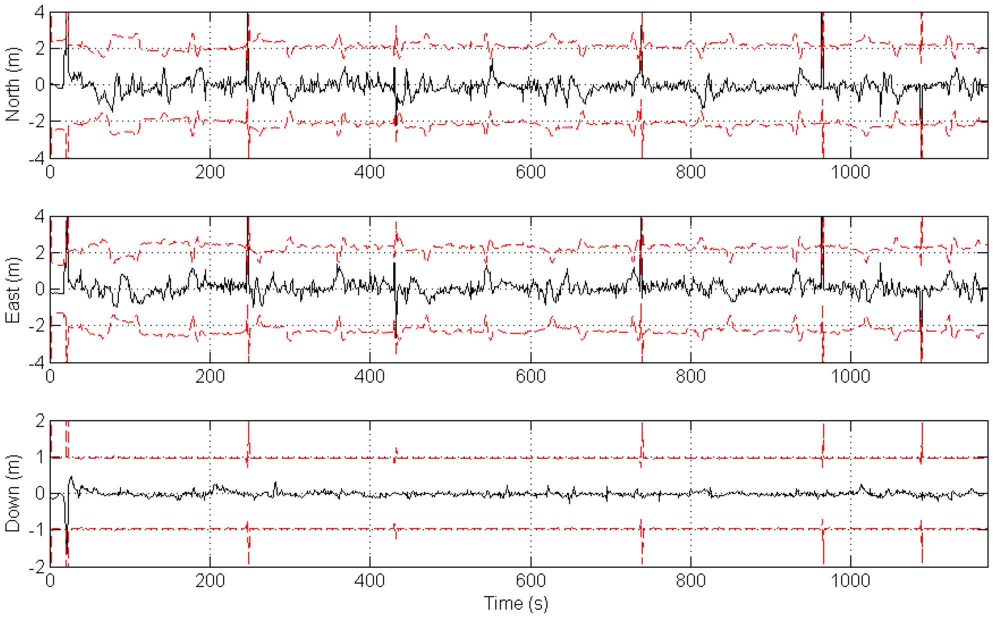


Figure 4.48: Position innovations with 2σ uncertainty boundaries between which 95% of innovations should lie

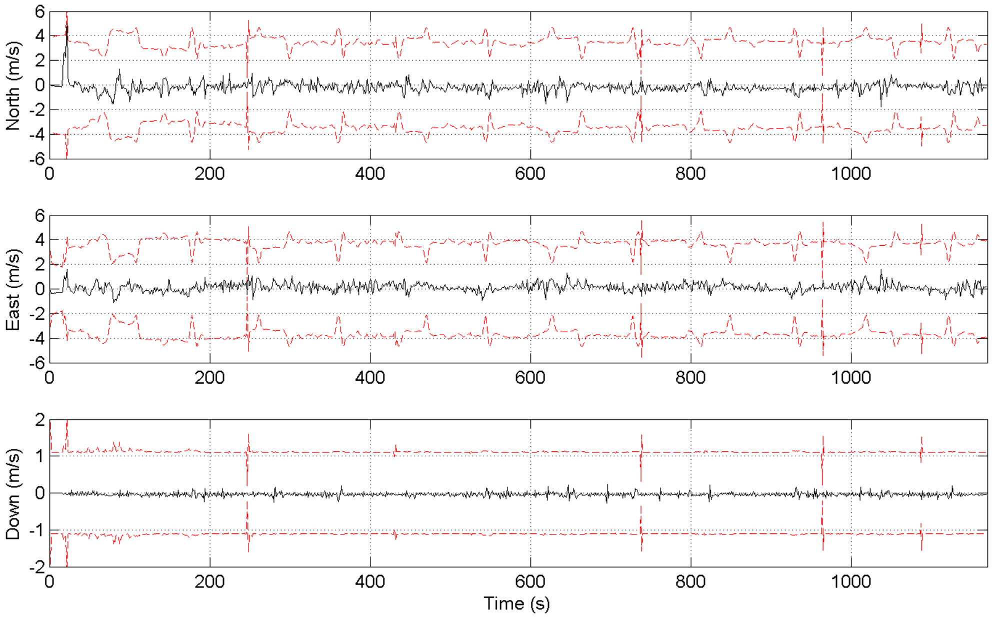


Figure 4.49: Velocity innovations with 2σ uncertainty boundaries between which 95% of innovations should lie

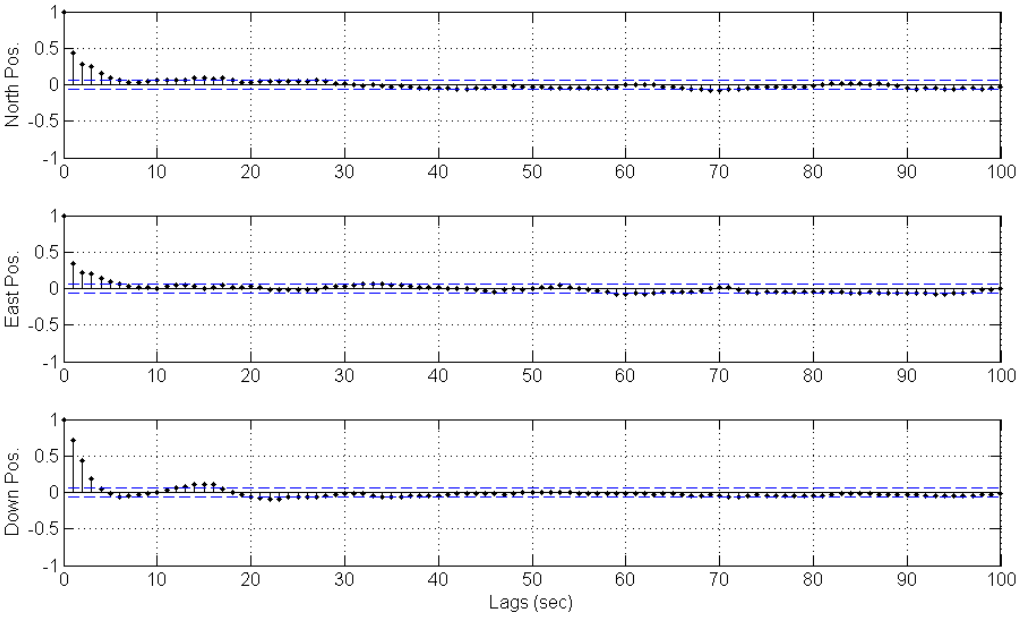


Figure 4.50: Position innovations autocorrelation function with 95% confidence boundaries

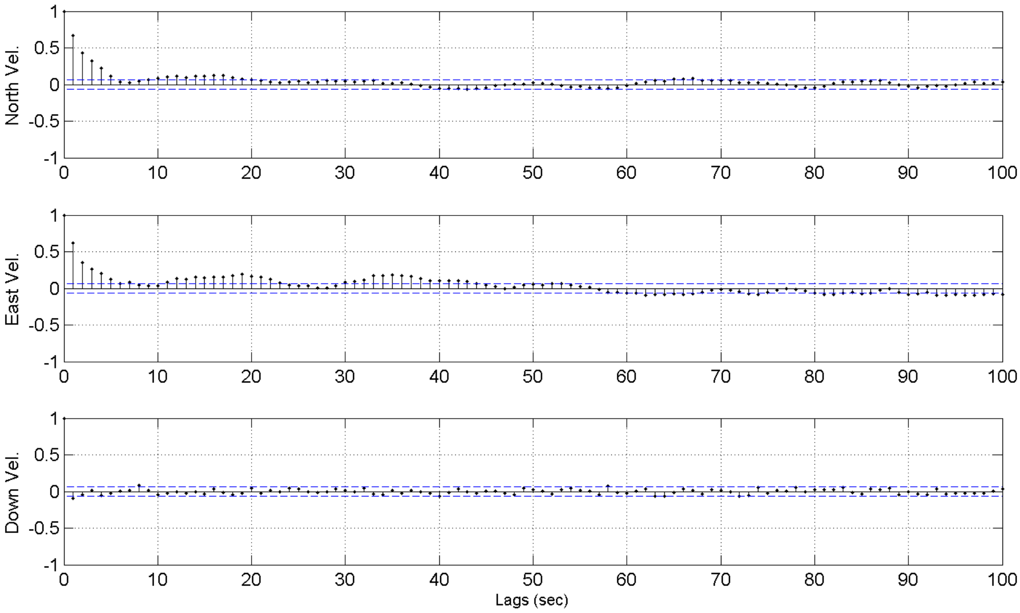


Figure 4.51: Velocity innovations autocorrelation function with 95% confidence boundaries

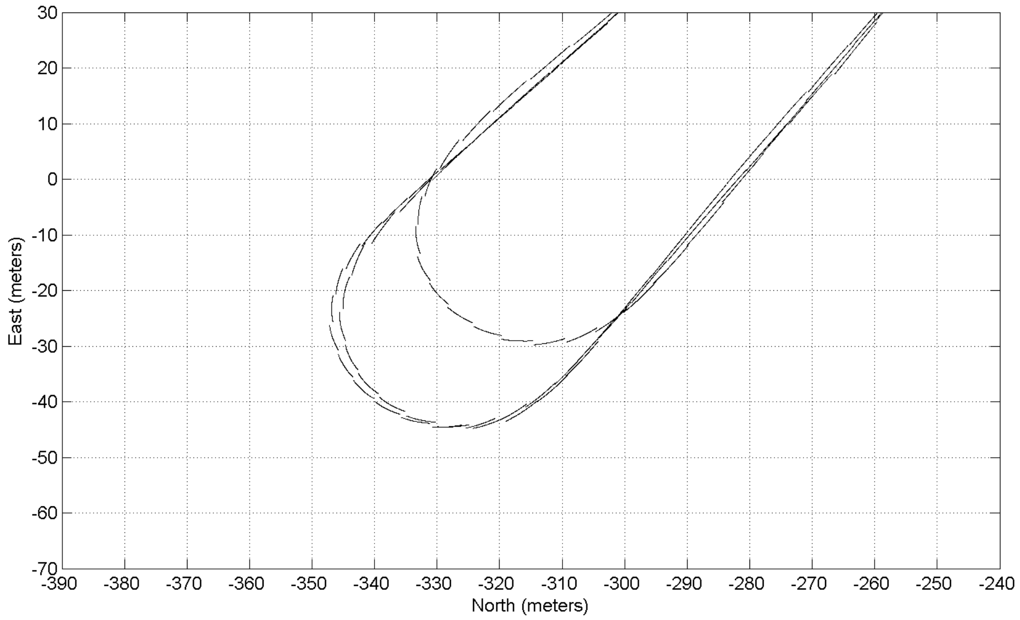


Figure 4.52: Effect of filter tuning and correct sampling time on attitude and bias estimates of the filter and the mechanisation process.

4.7 Chapter Summary

The focus in this chapter was to discuss the implementation of GPS-aided-INS systems. During the discussion the GPS principles were introduced including GPS basics, observables and Error budget. The most common GPS errors were discussed, modelled and corrected for. A model for the GPS pseudorange signal was given in which all significant error sources were compensated for. Basics for GPS satellites positioning including GPS orbital parameters, ephemeris parameters, ephemeris data processing and satellite position calculation were introduced. Algorithms for satellite positioning through ephemeris data processing were explained in two detailed flow charts and then tested using real time GPS receiver pseudoranges and ephemeris data. These topics formed the backbone knowledge required to implement a tightly-coupled navigation filter.

This chapter also introduced the loosely and tightly coupled filters, and discussed the effect of low satellite availability on both of them. It was stated that the tightly-coupled distinguished performance over the loosely-coupled is due to the fact that the tightly-coupled filter operates directly on the GPS row measurements, while the loosely-coupled depends on the GPS receiver internal filter's position and velocity estimates, which are inaccurate at low GPS availability, because the embedded receiver's filter is implemented as a reduced order filter.

Finally, the GPS-aided-INS navigation filter was tested and verified using actual GPS and IMU data sets. This chapter has achieved a major step toward the implementation of a high integrity navigation system. Integrity issues will be discussed in the following chapter, where the implementation loop will be completed.

Chapter 5

Fault Diagnosis

5.1 Introduction

The main purpose of this work is to develop a commercial low-cost autonomous UAV navigation algorithm with a Cost-Of-The-Shelf (COTS) inertial sensors and a GPS receiver. The utilization of such low grade equipments, which implies low reliability, requires the development of a high integrity navigation (localization) loop. Integrity implies the *robustness* of the system against failures and the ability of the the system to *survive* once a failure in one of its components (whether they are sensors, actuators, mathematical models or even computations) occurs. Without integrity the real time implementation of autonomous navigation system will be impossible. Standards are required to define design requirements to achieve integrity. A survey of NASA and military reliability and fault tolerance standards applied to robotics can be found in (Cavallaro, J. R. & Walker, I., D., 1994). These standards include handbooks, parts specifications, and documents describing procedures and programs for failure modes and effect analysis (FMEA) and reports standards.

There are a number of issues that must be considered when designing a fault diagnosis system. Among them the most important are:

- Types of failure modes that can be considered.
- Complexity of the implementation.
- Overall system performance.
- The robustness of the system in the presence of mismodeling.

Sensor characteristics along with its operational environment determine types of expected failures. In (Bhatti, U. I., & Ochieng, W. Y., 2007) failure modes and models of GPS-aided-INS systems were discussed. Modes were divided into groups based on their effect on the integration process and on each individual component of the integrated system. In this thesis, The classification of faults will be discussed in section 5.3

Complexity of the implementation increases the cost of the system in terms of hardware (for systems with hardware redundancy) and software (for systems with analytical redundancy). In fault diagnosis systems, it is more favorable for the detection algorithm to be as simple as possible and for the identification or/and isolation algorithm to be as efficient as possible. Fault detection can be implemented (as will be represented later) as a binary decision whether a fault has occurred or not. On the other hand, efficiency implies that there should be a trade-off between the level of complexity and performance.

The more complex the system model and the more model-dependent the technique, the more tolerated the system to sensitivity issues. The robustness of the system is measured by its resistance to changes due to mismodeling errors. In (Hanlon, P. D., & Maybeck, P. S., 2000), the mismodeling of Kalman filter system matrices was discussed in term of its effect on the mean and covariance of the residuals. Such mismodeling can result from mismodeling of the state transition matrix due to errors in state estimates, mismodeling of system matrices due to reduced order filter implementation or mismodeling of system dynamics and can result into false alarms or faults not to be detected. In (Hanlon, P. D., & Maybeck, P. S., 2000) expressions for the mean and covariance of the Kalman filter residuals in the presence of mismodeling in the input control matrix, output matrix, and state transition were derived. However, the effect of mismodeling in the process noise covariance matrix **Q** and the sensor noise covariance matrix **R** were neglected. This resulted into the conclusion that the conditional covariance matrix of the residuals is not dependent on the Kalman filter model, and it is only dependent on the covariance of the measurements. The motivation behind the assumption that **Q** and **R** represent the true model was to focus on system matrices variations because they commonly occur in failure detection applications considered by (Hanlon, P. D., & Maybeck, P. S., 2000) where MMAE was used. In this thesis, the characterization of the Kalman filter residuals in the presence of mismodeling in **Q** and **R** will be discussed when FDI results are introduced is section 5.5.

The overall performance of the system is measured by its tolerance to false alarms and how fast it detects and diagnoses failures. Therefore, a successful FDI system should have the following characteristics:

- Guarantees that all the modelled and the unmodelled faults will be detected.
- Doesn't affect system performance during no-fault condition. The performance of the system is reflected, as previously discussed in section 2.3.1, in the consistency of the estimator. It is essential always to understand that, filter consistency implies its optimality.
- Assists design specifications, including filter's cutoff frequency (highest frequency maneuvers), the required accuracy (function of the estimated covariance matrix) and minimum cost sensors;
- Responds fast to faults.
- Offers a trade off between complexity and FDI performance.

This chapter discusses mainly the detection and identification of any unexpected behavior of the INS/GPS navigation system. The organization of this chapter is as follows. First, section 5.2 reviews architectures of available FDI systems and surveys it's most common techniques. Next, section 5.3 classifies faults by their nature and modelling. Kalman filter - based fault diagnosis systems are discussed in section 5.4. Finally, performance validation tests are represented to highlight the system performance.

5.2 FDI: Terminology and Structure

In section 2.2 the Kalman filter was formulated. The filter model resembles the true model under four important assumptions (Maybeck, P. S., 1979):

- Predictions and observations are assumed Gaussian variables described by the first and second order statistics (mean and variance).
- The process and observation noise terms are assumed white, uncorrelated with covariances $\mathbf{Q}(k)$ and $\mathbf{R}(k)$, respectively.
- The model is assumed linear (doesn't hold for cases where the unscented filter is employed).
- The system is of stated order.

The system is said to be operating normally, if the previously listed assumptions hold. The mode of operation where these assumptions hold is referred to as *normal mode* or *no-fault* or *no-failure mode* of operation. On the other hand, if one of these assumptions becomes void, then the Kalman filter no longer produces a consistent estimates and the system is said to be in fault. Operation modes where such situations occur are referred to as *fault* or *failure modes* of operation.

The first two assumptions, namely Gaussianness and whiteness, are model dependent. Therefore mismodelling in Kalman filter system matrices makes the filter fail. Moreover, there is no mathematical model is able to describe the real word.

However, the mathematical model should be able to capture major changes in vehicle dynamics. As indicated previously in subsection 3.3.1, where sensor errors are discussed, the frequency of the model (which is, basically, the sampling rate of the IMU) should be higher in order of magnitude than the highest frequency of vehicle maneuvers (cut-off frequency), while neglecting higher frequency dynamics such as vibrations. The fact that, this is the reason why the Kalman filter frequency response should resemble the frequency response of a low-pass filter (Scheding, S., 1997). On the other hand, mismodelling in system matrices **F** and **H** will cause the predictions to be in error. While mismodelling in **Q** and **R** will cause the Kalman gain, and therefore the fusion decision, to be in error.

The third assumption, that is linearity, is desirable because linear systems are easier to manipulate. However, when the system becomes highly non-linear, linearisation by perturbation of the system nominal equations or by calculating the Jacobian about nominal point becomes inadequate and the filter fails. Therefore, non-linear filters such as the unscented filter *must* be employed.

Reduced order filters are usually implemented in systems where the fill order state vector contains redundant information. However, reducing the order of the state vector by removing some of its states, such as removing the biases states, results into degradation in the overall performance of the system. The effect of reduced order state vector when the full vector dimension is required is similar to the effect of mismodelling in the system matrices, since it neglects some of important system dynamics.

5.2.1 FDI System Structure

Figure 5.1 shows the general components of an FDI system. It is composed of two basic stages, residual generation and decision making. Residual generation is responsible for generating the residuals, which carry failures signatures or indicators, utilized by the decision making stage. The decision making stage calculates statistics, based on the residuals, and provides a decision on the system mode of operation. This decision may include states and/or parameters estimation.

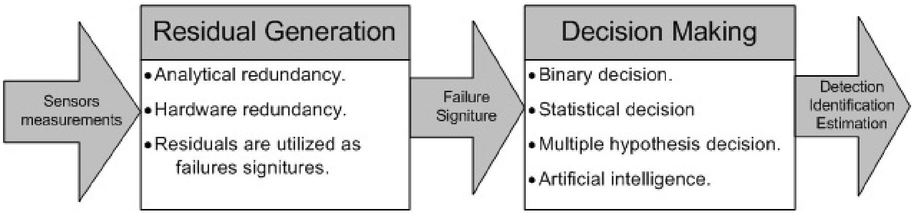


Figure 5.1: General architecture of FDI systems.

FDI techniques were surveyed by (Willsky, A. S., 1976) and (Yang, Q., 2004). In (Willsky, A. S., 1976), Two different designs were introduced; failure sensitive filters, where a complete redesign of the filter is performed to include failures, and failure monitors, where an auxiliary system is designed to monitor the operation of the nominal filter. The later design has an advantage over the former since it does not affect the filter during no-fault mode of operation. In (Yang, Q., 2004), the FDI problem was classified into two main categories based on their operational requirements. Model-based, where a priori knowledge of system dynamics is required, and data-driven, where a history of observations and system estimates are required to build up a data base. Examples of modelled-based approaches include the multiple-hypothesis filters and parity space, while data-driven approaches may include statistical classifiers and artificial intelligence. The Ven diagram in Figure 5.2 demonstrates the two FDI basic stages with their related techniques, while Table 5.1, briefly, describes various FDI techniques (Sukkarieh, S., Nebot, E. M., & Durrant-Whyte, H. F., 1998), (Sukkarieh, S., Nebot, E. M., & Durrant-Whyte, H. F., 1999), (Maybeck, P. S. & Hanlon, P. D., 1995), (Rago, C., Prasanth, R., Mehra, R. K., & Fortenbaugh, R., 1998), (Hashimoto, M., Kawashima, H., Nakagami, T. & Oba, F., 2001), (Hashimoto, M., Kawashima, H., Nakagami, T. & Oba, F., 2003), (Goel, P., Dedeoglu, G., Roumeliotis, S. I. &

Sukhatime, G. S., 2000), (Da, R. & Lin, C., F., 1995), (Willsky, A. S. & Jones, H., L. , 1974), (Willsky, A. S., Chow, E., Y., Gershwin, S., B., Greene, C., S.,, Houpt, P., K. & Kurkjian, A., L., 1980).

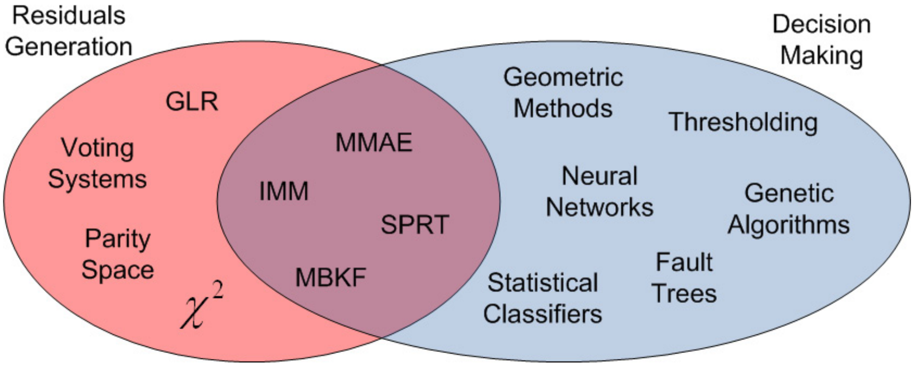


Figure 5.2: Stages of a typical FDI system with different implementations.

5.3 Classification of Faults

Faults classification plays a major rule in understanding the expected types of faults and how the system will behave under faulty conditions. Faults may be classified according to their behavior, models, and effect on the navigation system. In GPS/INS systems, faults may be classified into high frequency faults caused by jumps in the GPS receiver observations and low frequency faults in the IMU (Sukkarieh, S., Nebot, E. M., & Durrant-Whyte, H. F., 1998)(Sukkarieh, S., Nebot, E. M., & Durrant-Whyte, H. F., 1999). High frequency faults are mainly due to multipath and changes in satellites geometry, while low frequency faults may result from inertial sensors biases and the misalignment of the unit. IMU faults are impossible to diagnose during fusion process; bias estimates of the filter are supposed to provide an online compensation of INS biases and drifts. However, fault diagnosis of IMU faults before the fusion process is achievable through inertial sensors calibration (Titterton, D., & Weston, W., 2004)(Nebot, E. M., & Durrant-Whyte, H. F., 1999)(Chatfield, A., B., 1997)(Abdel-Hafez, M., F., 2006), where the IMU is operated for a long enough period while the vehicle is stationary and before it starts moving.

In (Bhatti, U. I., & Ochieng, W. Y., 2007) failure modes were classified into groups with each group representing a single component of a GPS-aided-INS system with MEMS inertial sensors. In this thesis, only sensor (i.e. GPS) faults are considered. Failure modes are divided into two groups, based on their effects on the performance of the GPS, hard failures and soft failures. A hard failure results in total sensor failure (i.e., the sensor reads nothing but noise, stop reading at all or stuck at a constant value) (Roumeliotis, S. I., Sukhatime, G. S. & Bekey G. A., 1998a), (Roumeliotis, S. I., Sukhatime, G. S. & Bekey G. A., 1998b). A soft failure results into degradation in sensor accuracy. Such a failure can occur due to several reasons, such as, incorrect modelling of orbital parameters, jamming, interference, atmospheric errors, multipath and low availability.

When implemented in MMAE mode, hard failures are modelled by zeroing the corresponding row of the output matrix $\mathbf{H}(k)$ (Hanlon, P. D., & Maybeck, P. S., 2000). For example, if failure affects the j^{th} observation of the measurements vector, then the failure mode is modelled by zeroing the j^{th} row of the $\mathbf{H}(k)$. The corresponding j^{th} element of the innovation sequence becomes

$$\nu_j(k) = z_j(k) \tag{5.1}$$

The corresponding elements of the innovation covariance are

$$\mathbf{S}_{j,j}(k) = \mathbf{R}_{j,j}(k) \tag{5.2}$$

$$\mathbf{S}_{j,k}(k) = 0 \quad \forall \quad k \neq j, k = 1, \ldots, m \tag{5.3}$$

$$\mathbf{S}_{k,j}(k) = 0 \quad \forall \quad k \neq j, k = 1, \ldots, m \tag{5.4}$$

Table 5.1: The most Common Fault detection and identification techniques

Method	**Description**
Failure sensitive	Redesign the filter to keep it sensitive to new data. Require a priori knowledge on faults effect on innovations. Optimality is sacrificed.
Voting systems	High degree of hardware redundancy (at lest three). Redundant sensors *must* be exactly alike. Easy to implement (only binary decisions). Provide fast detection of hard failures. Difficulties in detecting soft failures.
Innovation-based	Utilize the residuals (innovations) as fault signature. Employ analytical redundancy to map sensors-to-innovations. Easy to implement. Internal performance indications of the filter. Require no a priori knowledge neither on system dynamics nor on failure modes. Become model dependent in systems with analytical redundancy (such as INS). Detect abrupt-pulsate changes in sensor measurements. Can be implemented in various design levels. Detect an infinite set of faults. Cause no performance degradation during no-failure mode. Sensitive to filter tuning. Fail to detect slowly occurring faults. Offer tradeoffs between performance and complexity.
Multiple-hypothesis	Utilize a bank of filters each with different failure mode. Contain a conditional probability weighting computer. Utilize the Gaussian probability distribution function as fault signature. Provide an optimal state estimate via probabilistic weighting of different hypothesis. Guarantees filter convergence. Insensitive to filter tuning (although, an appropriate pre-tuning enhances bank performance). Covers a wide range of failures, when implemented in moving bank mode. Require a priori knowledge on both system dynamics and failure modes. Difficulties in diagnosis of abrupt-pulsate changes in sensor measurement. Computational complexity. Of all the methods, they are the most complex. Optimality is sacrificed during no-failure mode. Finite transit time to converge to appropriate failure mode. Require identical hypothesis parameters in order to converge to exact fault mode.
Geometric methods	Utilize predictions and updates uncertainties as fault signatures. Overlapping uncertainties indicate no-failure mode. Fail to detect faults when running on prediction for long periods.
Parity space	Utilize analytical redundancy. Check the consistency of the process model, sensor measurements, and command inputs.
Artificial Intelligence	Data-driven techniques. Considered autonomous only in systems with finite/limited failure modes. Require a priori knowledge on expected failures. Require sufficient amount of memory. Difficulties in fault detection. Should be integrated with a fault detection technique.

and its contribution in the normalized innovation squared is

$$q_j(k) = z_j^T(k)\mathbf{R}_{j,j}^{-1}z_j(k) \tag{5.5}$$

which should be minimal since it is consistent, if the filter was properly pre-tuned. Then the hypothesis is correct and will have higher probability.

On the other hand, soft failures are modelled by varying the measurement error noise covariance matrix $\mathbf{R}_k$ (White, N. A., Maybeck, P. S., & DeVilbiss, S. L., 1998). All elemental filters will have similar innovation, and the only difference is in $\mathbf{R}(k)$. The hypothesis that satisfies the true condition will have minimum NIS (or closer to one, when smaller values of $\mathbf{R}(k)$ are modelled) and will have higher probability.

5.4 Kalman Filter - Based FDI

This section, studies the effect of sensor faults on the Kalman filter innovations and innovations covariance matrix. Expressions for both are derived and related to the FDI problem. In the second part, multipath is simulated and three different FDI implementations are applied to diagnose that fault.

5.4.1 Formulation of the FDI Problem

The Kalman filter model was represented in chapter 2.2. However, it represented her again to derive an expression for the innovation and the innovation covariance under fault conditions.

If the system is modelled as a linear time-invariant following equations

$$\hat{\mathbf{x}}(k|k-1) = \mathbf{F}(k)\hat{\mathbf{x}}(k-1|k-1) \tag{5.6}$$

$$\hat{\mathbf{z}}_{nf}(k) = \mathbf{H}(k)\hat{\mathbf{x}}_{nf}(k|k-1) \tag{5.7}$$

$$\mathbf{z}_{nf}(k) = \mathbf{H}(k)\mathbf{x}(k) + \mathbf{v}(k) \tag{5.8}$$

$$\mathbf{z}_{f}(k) = \mathbf{H}(k)\mathbf{x}(k) + \mathbf{v}(k) + \mathbf{n}(k) \tag{5.9}$$

$$\mathbf{n}(k) = \sum_{i=1}^{M} \mathbf{L}_i(k)\mu_i(k) \tag{5.10}$$

where

$\mathbf{n}(k)$ represents failure modes which are unknown and arbitrary functions of time and are zero when there is no fault

$(\cdot)_{nf}(k)$ represents vector under no-fault conditions

$(\cdot)_{f}(k)$ represents vector under fault conditions

μ_i models the time-varying amplitude of a fault

$\mathbf{L}_i$ are fault transition functions that map a fault to the corresponding observation which are monic so that $\mu_i \neq 0$ implies $\mathbf{L}_i\mu_i \neq 0$

Therefore, under no-fault conditions, the innovations are defined as

$$\begin{aligned} \nu_{nf}(k) &= \mathbf{z}_{nf}(k) - \hat{\mathbf{z}}_{nf}(k) \\ &= \mathbf{z}_{nf}(k) - \mathbf{H}(k)\hat{\mathbf{x}}_{nf}(k|k-1) \end{aligned} \tag{5.11}$$

On the other hand, under fault conditions

$$\begin{aligned} \nu_{f}(k) &= \mathbf{z}_{f}(k) - \hat{\mathbf{z}}_{f}(k) \\ &= \mathbf{z}_{nf}(k) + \mathbf{n}(k) - \mathbf{H}(k)\hat{\mathbf{x}}_{f}(k|k-1) \end{aligned} \tag{5.12}$$

Assume perturbation of $\hat{\mathbf{x}}_f(k|k-1)$ such that

$$\hat{\mathbf{x}}_f(k|k-1) = \hat{\mathbf{x}}_{nf}(k|k-1) + \delta\hat{\mathbf{x}} \tag{5.13}$$

then, equation 5.12 becomes

$$\begin{aligned} \nu_f(k) &= \mathbf{z}_{nf}(k) - \mathbf{H}(k)\hat{\mathbf{x}}_{nf}(k|k-1) + \mathbf{n}(k) - \mathbf{H}(k)\delta\hat{\mathbf{x}} \\ &= \nu_{nf}(k) + \mathbf{n}(k) - \mathbf{H}(k)\delta\hat{\mathbf{x}} \end{aligned} \tag{5.14}$$

Equation 5.14 implies that, the innovations under fault conditions are not only biased due to the fault vector $\mathbf{n}(k)$, but also accumulates errors over time due to drifts in the state estimates represented by $\delta\hat{\mathbf{x}}$ transformed into the innovations space via $\mathbf{H}(k)$. Therefore, the innovation under fault condition are biased.

The innovation covariance is estimated as follows;

$$\begin{aligned} \mathbf{S}_f(k) &= E\left[\nu_f(k)\nu_f(k)^T\right] \\ &= E\left[\nu_{nf}(k) + \mathbf{n}(k) - \mathbf{H}\delta\hat{\mathbf{x}})(\nu_{nf}(k) + \mathbf{n}(k) - \mathbf{H}\delta\hat{\mathbf{x}})^T\right] \end{aligned} \tag{5.15}$$

since $\nu_{nf}(k)$, $\mathbf{n}(k)$, and $\delta\hat{\mathbf{x}}$ are uncorrelated, equation 5.15 becomes

$$\begin{aligned} \mathbf{S}_f(k) &= E\left[\nu_{nf}(k)\nu_{nf}(k)^T\right] + E\left[\mathbf{n}(k)\mathbf{n}(k)^T\right] + E\left[\mathbf{H}\delta\hat{\mathbf{x}}\delta\hat{\mathbf{x}}^T\mathbf{H}^T\right] \\ &= \mathbf{S}_{nf}(k) + \mathbf{H}\Delta\mathbf{P}(\mathbf{k}|\mathbf{k-1})\mathbf{H}^T + \mathbf{N}(k) \\ &= \mathbf{HP}(k|k-1)\mathbf{H}^T + \left[\mathbf{R}(k) + \mathbf{H}\Delta\mathbf{P}(k|k-1)\mathbf{H}^T + \mathbf{N}(k)\right] \end{aligned} \tag{5.16}$$

where

$\Delta\mathbf{P}(k|k-1)$ is state prediction error covariance due to fault. It represents the error in state estimates uncertainty due to $\delta\hat{\mathbf{x}}$

$\mathbf{N}(k)$ represents the covariance of the fault modes vector $\mathbf{n}(k)$

Equation 5.16 implies that, under fault conditions the sensor noise covariance must be increased so that the innovation and its covariance become consistent. In fact, this explains the methodology by which the MMAE identifies the fault and produces a consistent state estimates.

5.4.2 High Integrity Navigation Design

As indicated earlier in section 5.1, to design a high integrity navigation system the fault detection and identification algorithm should meet the design requirements. These requirements include detectability of all modelled and unmodelled faults, doesn't sacrifice filter's optimality during no-fault, fast response time and offers complexity-performance tradeoff. On the other hand, the detection algorithm should be as simple as possible while having an efficient detection ability. The only detection scheme that satisfies all these requirements is innovation based detection. Therefore, in this thesis the fault detection is based on the innovation sequence and is implemented through the gating of the normalized innovation squared (NIS) given by equation 2.20 and is rewritten here again

$$q(k) = \nu^T(k)\mathbf{S}^{-1}(k)\nu(k) \leq \gamma \tag{5.17}$$

where $\nu(k)$ is the innovation sequence given by equation 2.10, $\mathbf{S}(k)$ is the innovation covariance given in equation 2.12 and γ is a percentage probability determined prior to the fusion process which grantees that a particular observation lies within an ellipsoid (level of confidence). The NIS is a χ^2 distributed in m DoF, where m is the dimension of the measurements vector, and the gating value is selected from the Table in appendix F (12.59 in the case of 3-D position and 3-D velocity). However, this value is sometimes *relaxed* to best suit the environment and the probability region allowed. If the NIS is less than that particular gate and the filter was properly pre-tuned, then the Kalman filter model represented by $\mathbf{Q}(k)$ and $\mathbf{R}(k)$ resembles the true model of the system. On the other hand, if the NIS exceeds the gating threshold, then a fault must have occurred and the fault detection stage sets an alarm.

In spite of its simplicity, the implementation of this fault diagnosis is crucial. The rejection of GPS fixes may result in a continuous deviation of INS estimates which will results in increasing innovations and continuous rejection of GPS fixes, so that the loop will never enter the correction stage. As a result, filter tuning is a critical step in filter implementation. Tuning the Kalman filter, it is important to consider the following scenarios (Sukkarieh, S., Nebot, E. M., & Durrant-Whyte, H. F., 1998), (Sukkarieh, S., Nebot, E. M., & Durrant-Whyte, H. F., 1999):

- During rejection of GPS fixes, the Kalman filter stays in the prediction stage. The uncertainty in INS estimates will grow according to $\mathbf{Q}$. If $\mathbf{Q}$ is set to larger value, states will be updated using the first available GPS fix irrespective to its accuracy. Similar situation will result if $\mathbf{R}$ is set to small value.
- If $\mathbf{Q}$ is set to small value, INS estimates will be less corrected. During the rejection of GPS fixes, due to faults such as multipath, the filter stays in the prediction stage. However, due to small value of $\mathbf{Q}$, the uncertainty grows in smaller rate and there is a risk of the INS missing all GPS fixes. Which in turn results into a diverging solution.
- A small value of $\mathbf{Q}$ and large value of $\mathbf{R}$ imply a more trusted predictions over observations and the innovation will be correlated.
- If the gating function is applied, a small value of $\mathbf{R}$ will result into more rejection of GPS fixes. Once GPS observations are validated, more weight will be applied to the innovation which will result into a larger correction of INS predictions. If the GPS fixes are noisy, then the corrected INS states will be noisy too.

Figure 5.3 shows the membership function between the level of correlation and the selected value of the sensor noise covariance $\mathbf{R}$ when the gating function is employed. Moreover, the gating function is only a detection scheme. It doesn't identify or isolate failures. Therefore, even though a single Kalman filter is enough for fault detection. Fault identification requires a bank of Kalman filters. For the filter bank to span the entire failure space, it should be implemented as a moving bank. In this thesis, the moving-bank MMAE scheme discussed previously in section 2.4.2 is utilized to isolate the fault and form the optimal state estimate along with the required parameter that models the failure.

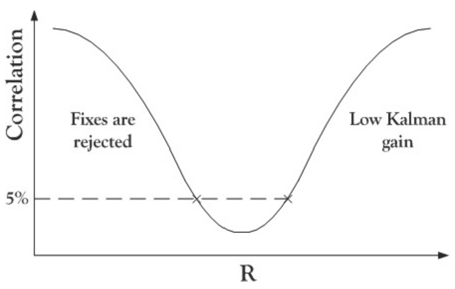


Figure 5.3: Tuning membership function when the χ^2 gate is applied.

As discussed earlier in section 5.2.1, thresholding and MMAE, each of them has its own drawbacks. However, both schemes can be utilized in a sequential manner, where each of them compensates for the other weaknesses. Figure 5.4 indicates the effect of one scheme on the other. Among all of these effects the most important is the reduced computational cost, enhanced no-failure performance and decreased effect of convergence time of the MMAE bank. On the other hand, even though an innovation based approach is utilized, the implementation is no longer sensitive to the tuning process and can detect slowly changing ramp failures.

The implementation of this sequential scheme is represented by the block diagram in Figure 5.5. The FDI system is implemented with *a fixed nominal filter that is independent of the bank*, and a filters bank consisting of K elemental filters implemented as moving bank. The innovation based algorithm monitors the innovations of the nominal filter, once a fault is detected a fault alarm is set and sent to the MMAE algorithm. The MMAE algorithm starts tuning for the appropriate value of sensor noise covariance matrix through the utilization of the Gaussian probability density

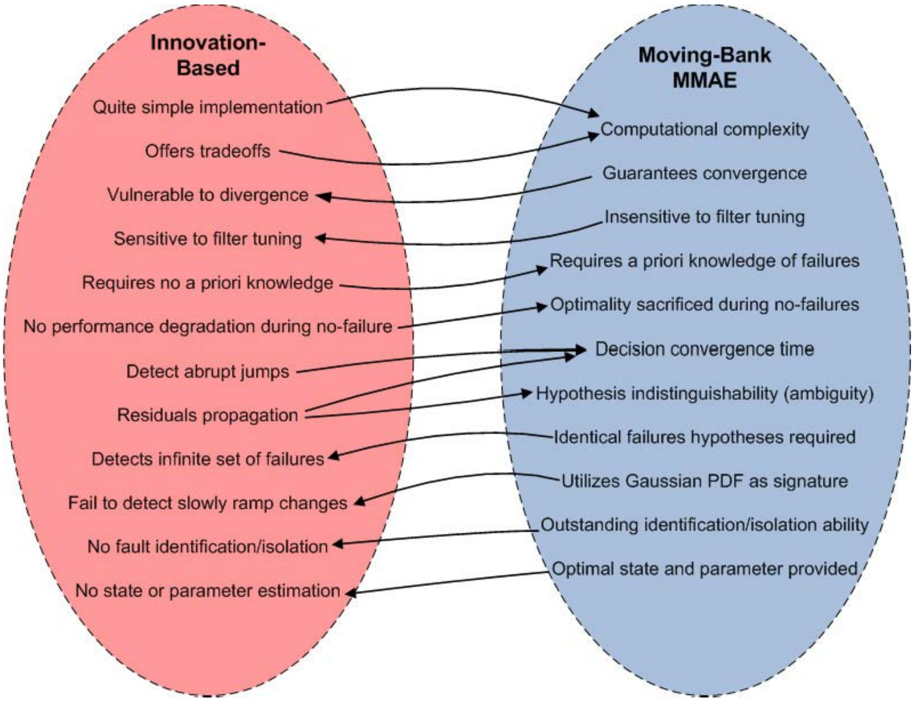

Innovation-Based
Quite simple implementation
Offers tradeoffs
Vulnerable to divergence
Sensitive to filter tuning
Requires no a priori knowledge
No performance degradation during no-failure
Detect abrupt jumps
Residuals propagation
Detects infinite set of failures
Fail to detect slowly ramp changes
No fault identification/isolation
No state or parameter estimation
Moving-Bank MMAE
Computational complexity
Guarantees convergence
Insensitive to filter tuning
Requires a priori knowledge of failures
Optimality sacrificed during no-failures
Decision convergence time
Hypothesis indistinguishability (ambiguity)
Identical failures hypotheses required
Utilizes Gaussian PDF as signature
Outstanding identification/isolation ability
Optimal state and parameter provided

Figure 5.4: FDI design tradeoffs between innovation-based and moving bank MMAE schemes. Arrows directions indicate the compensation effect of one scheme advantages on the other scheme disagvantages.

function, and calculated as follows,

$$\mathbf{a}_{MMAE}(k) = \sum_{i=1}^{K} p_i(k)\mathbf{a}_i \tag{5.18}$$

$$\mathbf{R}_{MMAE}(k) = \mathbf{a}_{MMAE}(k)\mathbf{R}_{MMAE}(k-1) \tag{5.19}$$

where

$p_i(k)$ represents the i_{th} elemental filter hypothesis probability at time k.

a_i represents the i_{th} elemental filter hypothesis parameter.

a_{MMAE} represents the filters bank optimal estimated failure parameter at time k.

$\mathbf{R}_{MMAE}(k)$ represents the filters bank optimal estimated failed sensor noise covariance at time k.

and the bank is centered at $\mathbf{R}_{MMAE}$. However, due to transit time requirements of the bank, the consistency of the bank's estimate must be checked before it's optimal state estimate is calculated. An algorithm similar to the χ^2 gating is implemented at *the FDI system level.* The filter bank residuals utilized for consistency test are generated as follows;

$$q_{MMAE}(k) = \sum_{i=1}^{K} p_i(k) q_i(k) \tag{5.20}$$

where

$q_i(k)$ represents the i_{th} elemental filter calculated NIS.

$q_{MMAE}(k)$ represents the filter bank calculated NIS.

Then, $q_{MMAE}(k)$ is compared with a threshold selected as in equations 2.20 and 2.21. If the filter bank's innovations are consistent with the probabilistic weighted sensor noise covariance $\mathbf{R}_{MMAE}(k)$, the MMAE algorithm continues and the optimal state estimate $\hat{\mathbf{x}}(k|k)$ along with it's uncertainty $\mathbf{P}(k|k)$ are calculated as in equations 2.37 and 2.38, respectively. On the other hand, if this condition is not satisfied, the filter bank is considered inconsistent, so that it doesn't enter neither the prediction stage nor the correction stage and the algorithm's estimates are taken as the predictions of the fixed nominal filter. This process is repeated until one of them, i.e. the fixed nominal or the filters bank, becomes consistent and it's estimates are selected as the optimal solution.

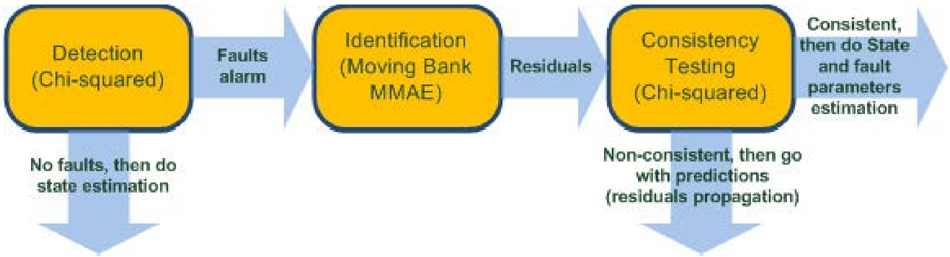

Detection (Chi-squared)
Faults alarm
Identification (Moving Bank MMAE)
Residuals
Consistency Testing (Chi-squared)
Consistent, then do State and fault parameters estimation
No faults, then do state estimation
Non-consistent, then go with predictions (residuals propagation)

Figure 5.5: FDI Methodology

This sequential implementation enhances the overall performance as follows:

- Since both scheme's estimates are consistency checked, optimality is guaranteed during fault and no-fault conditions.
- The filters bank is only allowed to operate on segments of time where a fault has been declared. Therefore, computational complexity is reduced.
- The effect of the filters bank convergence time is eliminated since it is consistency checked.
- The effect of tuning on the fixed nominal filter is eliminated since, as discussed in section 2.4.2, the MMAE is an adaptive technique.
- Slowly changing ramp faults are detected and identified.

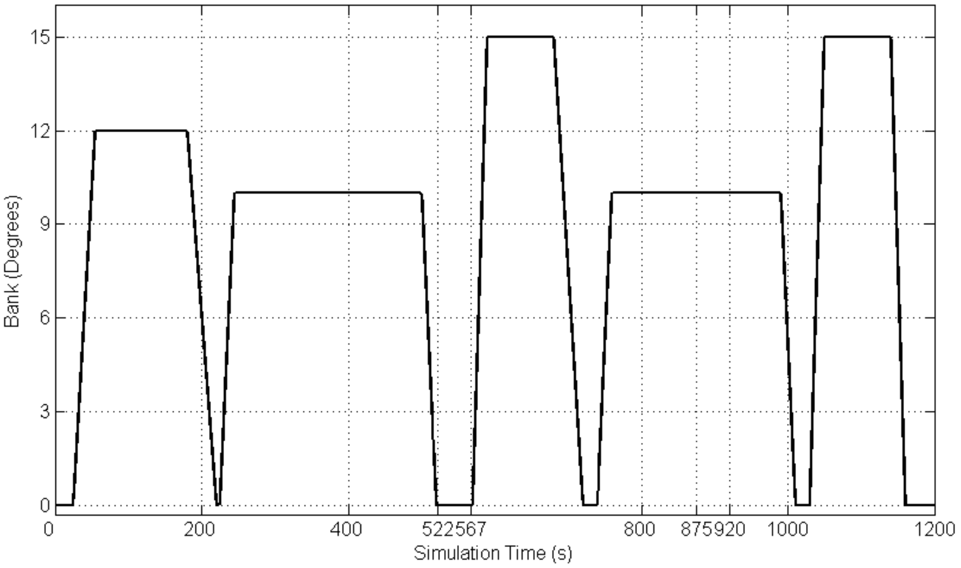


Figure 5.6: The bank angle command applied to simulate the trajectory. The simulated trajectory undergoes five rotations all to the right. The two intervals where faults are simulated lie between 522-567 while travelling in straight line and 875-920 while steady turn.

5.5 Performance Verification - Simulation

In this section, Errors are introduced to GPS observations in two different scenarios. First, while the vehicle in travelling in straight line. Then, while steady turning. Both failures of GPS signal last for 45 seconds and varies in magnitude for both position and velocity observations. For the same trajectory discussed earlier in section 4.5. The command input to the bank angle is shown here again, in Figure 5.6. These errors are composed of multipath and low satellite availability commonly occur in INS/GPS systems. These errors where introduced to the GPS signal as random noise and ramp signal, respectively. Table 5.2 represents the parameters of added noise to simulate failures in GPS signal.

Table 5.2: Simulated failure noise parameters

Parameter	**Straight Line 522-567 sec**	**Steady Turn 875-920 sec**
Position 1σ	10.0 m	12.0 m
Velocity 1σ	5.00 m/s	6.00 m/s
Position slope	0.15 m/s	0.15 m/s
Velocity slope	0.10 m/s/s	0.10 m/s/s

The MMAE is configured with five banks, while the sequential implementations contains one extra fixed nominal filter. The five hypotheses are [100,10,1,0.1,0.01]. Figures 5.7 and 5.8 show the innovations, while 5.9 and 5.10 show the autocorrelation function of the innovations when χ^2-squared FDI is used. The continuous jumps of GPS fixes were detected, and failure of GPS sensor was declared. The rejection of GPS fixes and accumulation of inertial sensor biases (which include zero biases, misalignments, scale factor errors, random errors, random walk and computational errors) cause drifts in the navigational solution and the filter remains in the prediction stage. Once the uncertainty becomes consistent with its innovation, GPS observations are validated and the filter enters the update stage. Figures 5.11 to 5.13 show estimated errors. The gating function shows a poor FDI performance against unexpected navigation system failures. However, designed to reject failures, the filter can be re-tuned to place less weighting on observations. This is accomplished by decreasing the covariances in the process noise matrix $\mathbf{Q}(k)$ which places more trust on INS predictions and increasing covariances in the sensor noise covariance matrix $\mathbf{R}(k)$. This method increases the autocorrelation of the innovation, and the filter is in the *suboptimal* mode of operation (Sukkarieh, S., 2000). Finally, Figure 5.14 shows the χ^2 consistency test. The test shows an in consistent innovations during failures.

Similar failure scenarios are now represented to both the MMAE and sequential algorithms. Figures 5.15 to 5.18, show the innovations and their corresponding autocorrelations. Unlike the innovation-based case, The innovation looks consistent, zero mean and uncorrelated. Similar results are shown in Figures 5.19 to 5.22 for the sequential case, which show similar behavior.

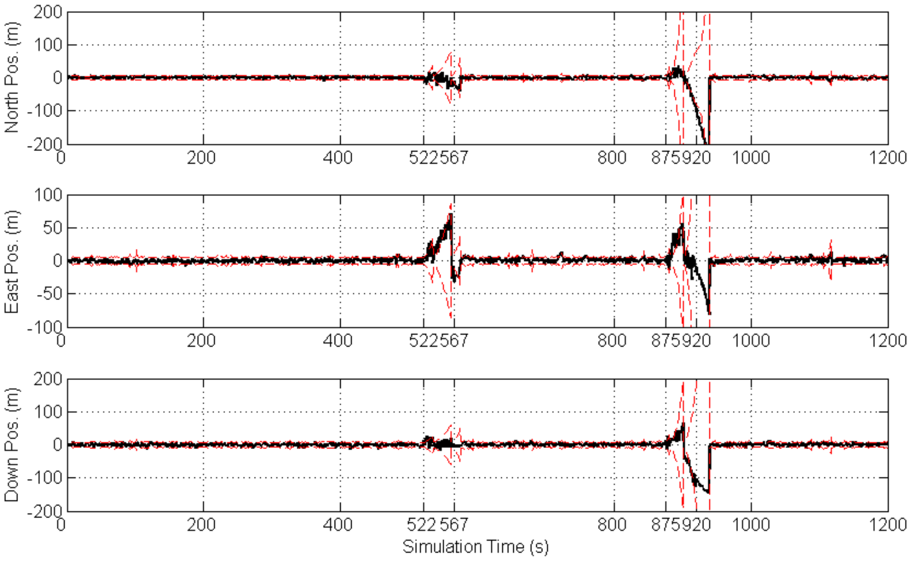


Figure 5.7: Position innovations and its 2σ uncertainty when innovation-based FDI is used to detect and diagnose failures between 522-567 sec (straight line motion) and between 875-920 sec (rotational motion).

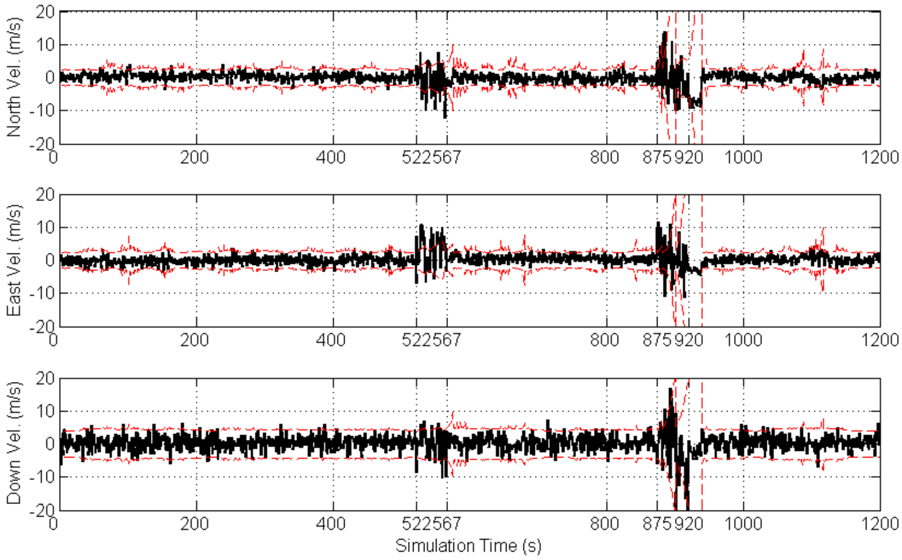


Figure 5.8: Velocity innovations and its 2σ uncertainty when innovation-based FDI is used to detect and diagnose failures between 522-567 sec (straight line motion) and between 875-920 sec (rotational motion).

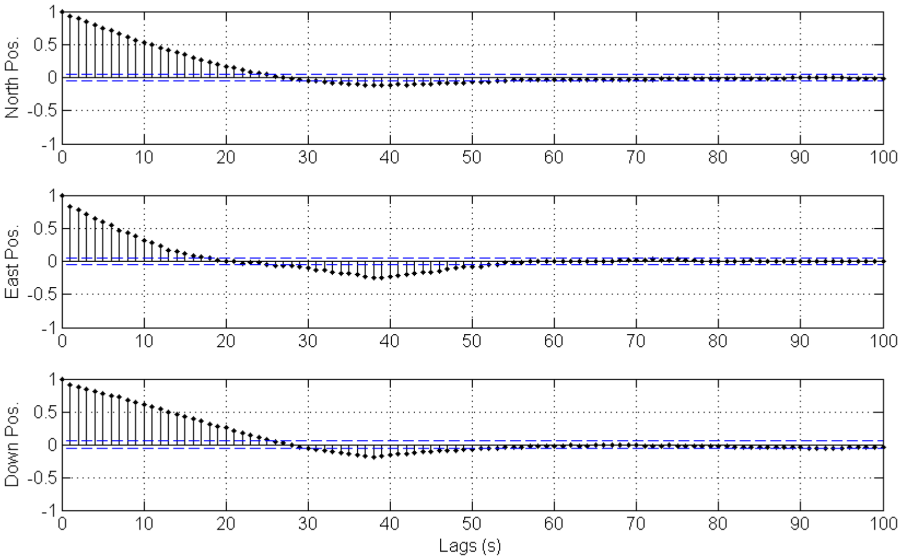


Figure 5.9: Auto-correlation of the position innovation depicts the fact that continuous rejection of GPS position due to mismodelling in its noise covariance increases the correlation of the innovation sequence.

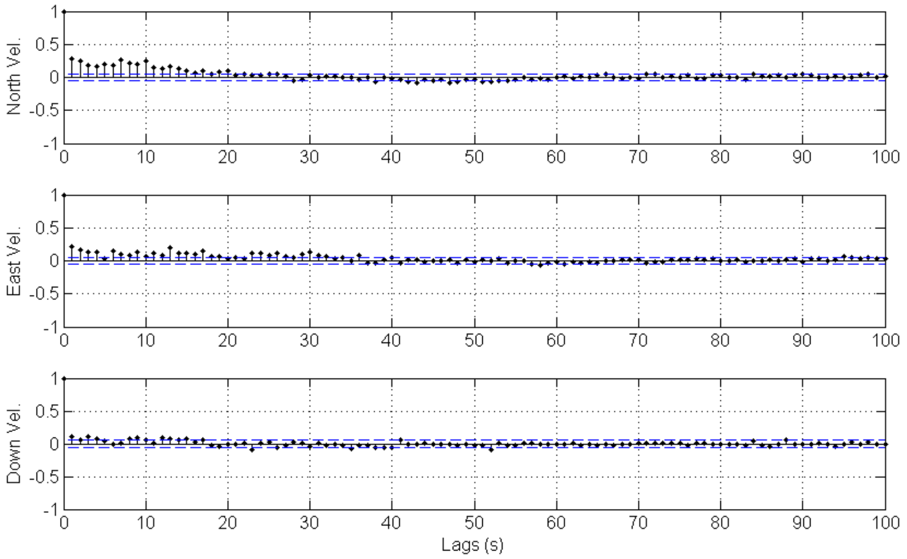


Figure 5.10: Auto-correlation of the velocity innovation depicts the fact that continuous rejection of GPS position due to mismodelling in its noise covariance increases the correlation of the innovation sequence.

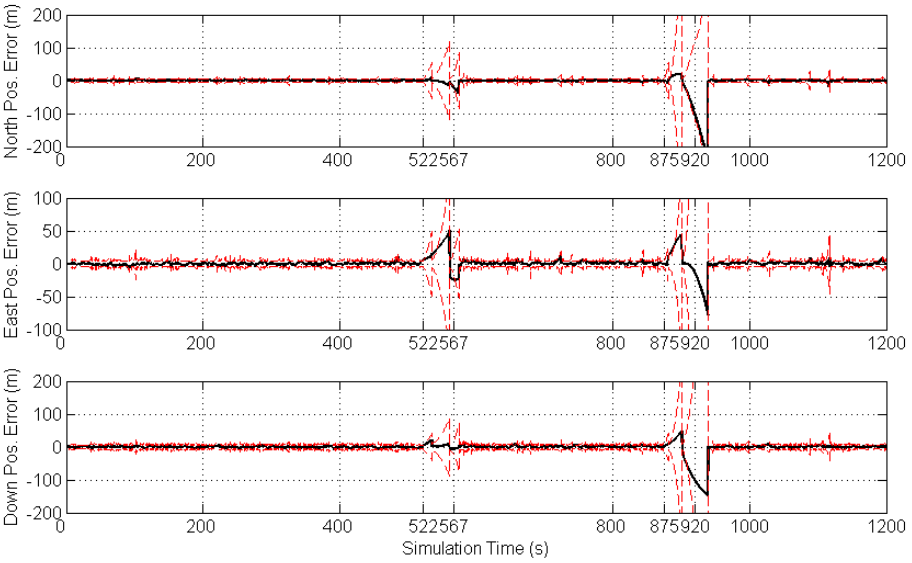


Figure 5.11: Error in estimated position along with its 3σ uncertainty when innovation-based FDI is used to diagnose failures between 522-567 sec (straight line) and 875-920 sec (steady turn).

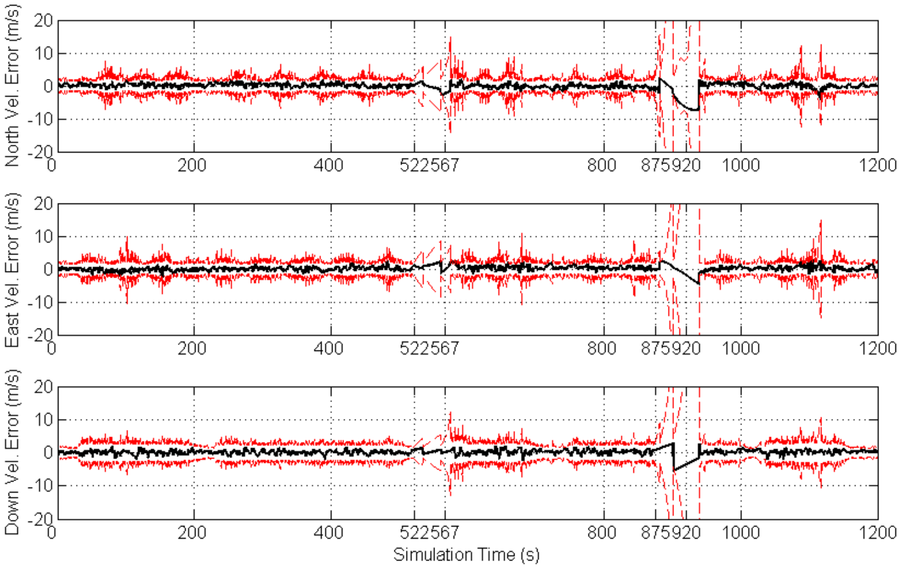


Figure 5.12: Error in estimated velocity along with its 3σ uncertainty when innovation-based FDI is used to diagnose failures between 522-567 sec (straight line) and 875-920 sec (steady turn).

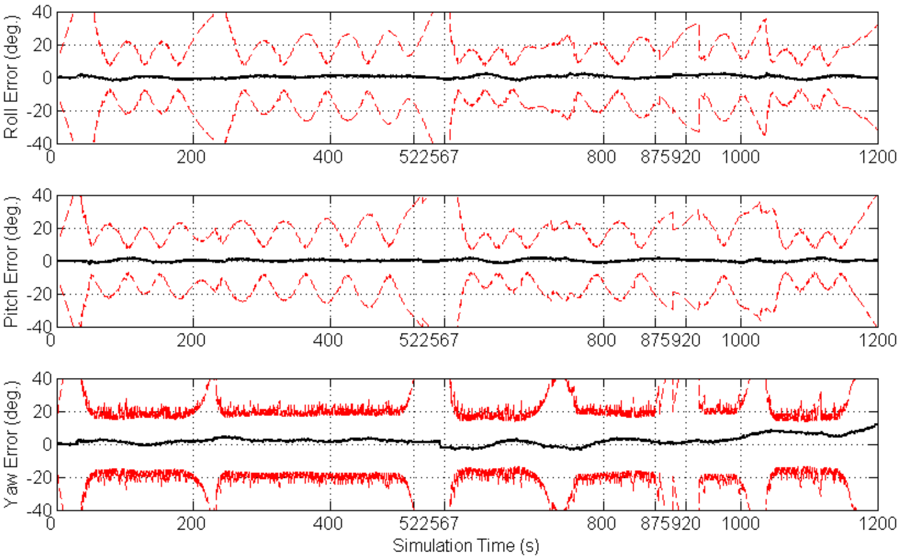


Figure 5.13: Error in estimated attitude along with its 3σ uncertainty when innovation-based FDI is used to diagnose failures between 522-567 sec (straight line) and 875-920 sec (steady turn).

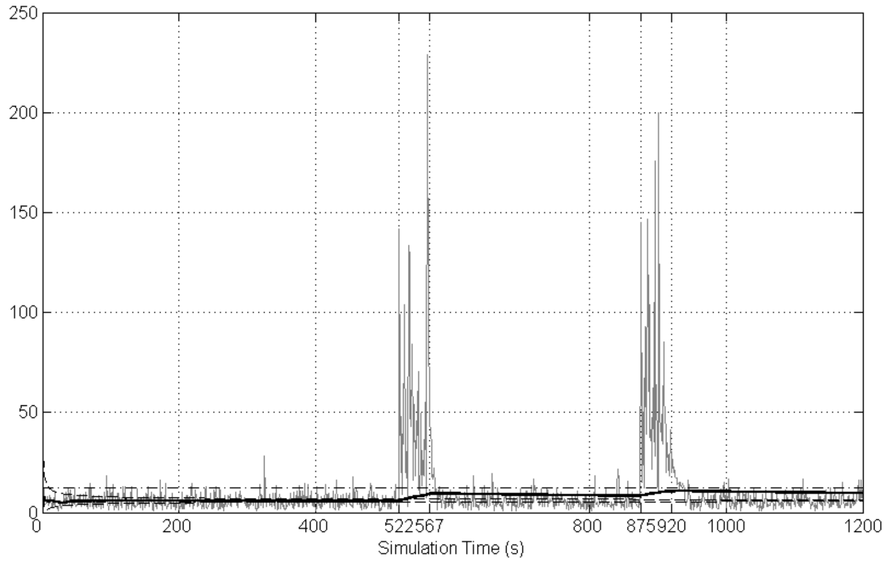


Figure 5.14: The χ^2-test of the innovation sequence when innovation-based FDI is used to detect and diagnose failures between 522-567 sec (straight line motion) and between 875-920 sec (rotational motion).

Figure 5.23 shows the simulated path and the estimated trajectory when the three methods are applied. Figure 5.24 magnifies the segment where failures occurred while travelling in straight line. On the other hand, Figure 5.25 shows the segment where failures occurred while the aircraft was undergoing a rotational motion. For more demoncentration on the performance of the sequential MMAE algorithms, Figure 5.26 shows the percentage of time when the filters bank in the sequential implementation was utilized to identify failures. Figure 5.27 shows the Gaussian conditional probabilities of the five elemental filters, while Figure 5.28 shows the estimated noise covariance $\mathbf{R}_{MMAE}$ for the six positions and velocities observations. Finally, the optimal estimate of the bank's hypothesis in the sequential algorithm is represented in Figure 5.29.

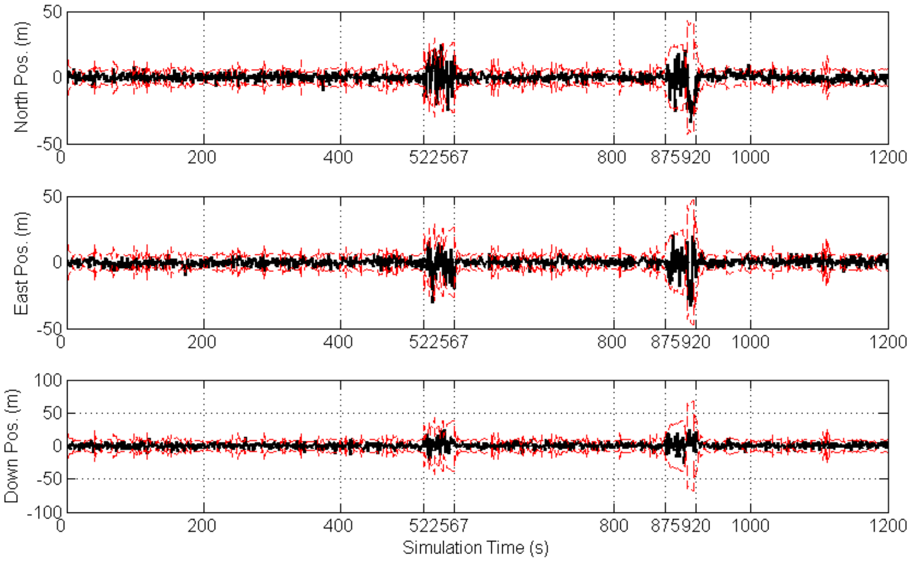


Figure 5.15: Position innovations and its 2σ uncertainty when MMAE is used to detect and diagnose failures between 522-567 sec (straight line motion) and between 875-920 sec (rotational motion).

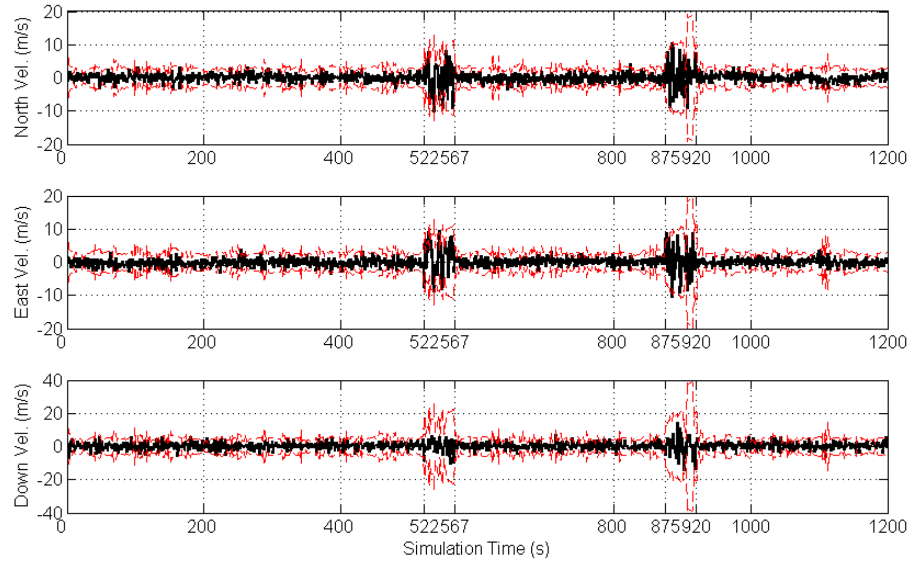


Figure 5.16: Velocity innovations and its 2σ uncertainty when MMAE FDI is used to detect and diagnose failures between 522-567 sec (straight line motion) and between 875-920 sec (rotational motion).

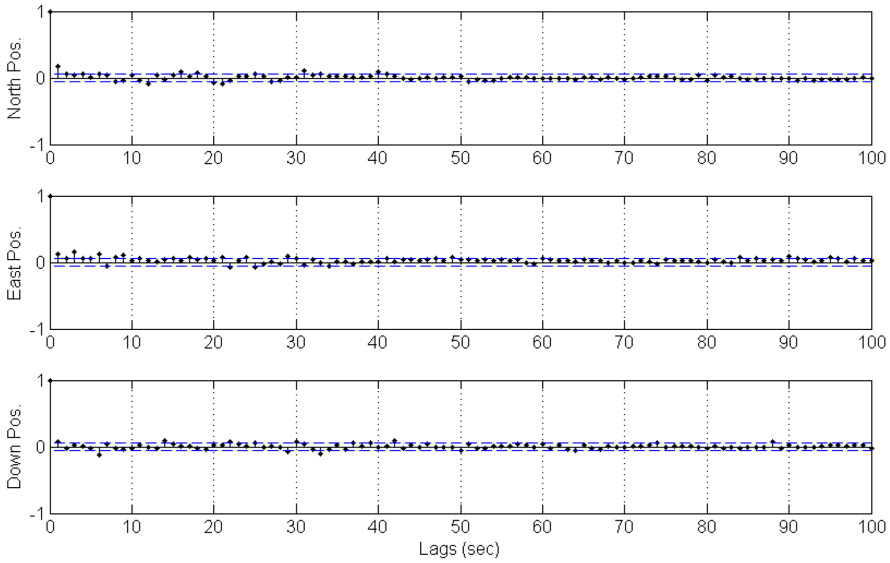


Figure 5.17: Unlike the case of innovation-based FDI, the auto-correlation of the position innovation depicts the fact that the innovations sequence when MMAE is used to reject errors introduced to GPS signal are indeed uncorrelated.

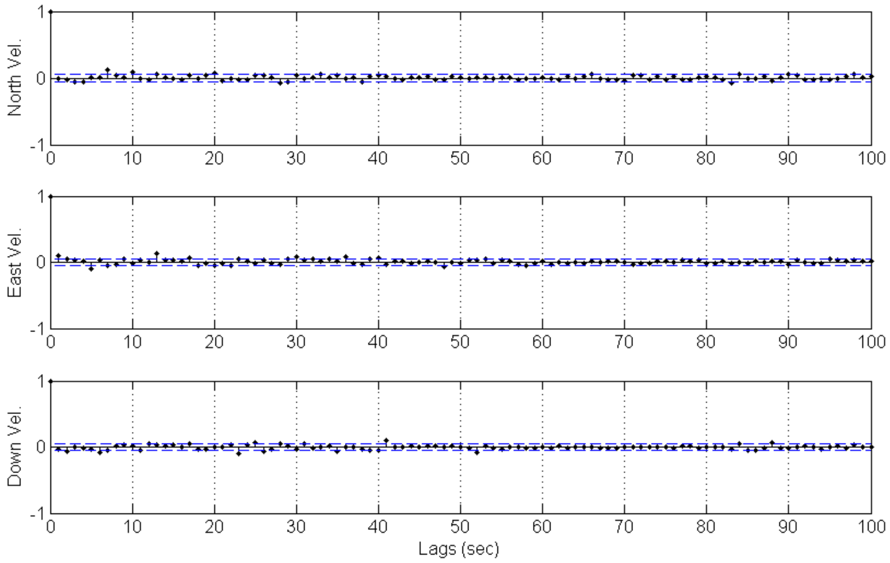


Figure 5.18: Unlike the case of innovation-based FDI, the auto-correlation of the velocity innovation depicts the fact that the innovations sequence when MMAE is used to reject errors introduced to GPS signal are indeed uncorrelated.

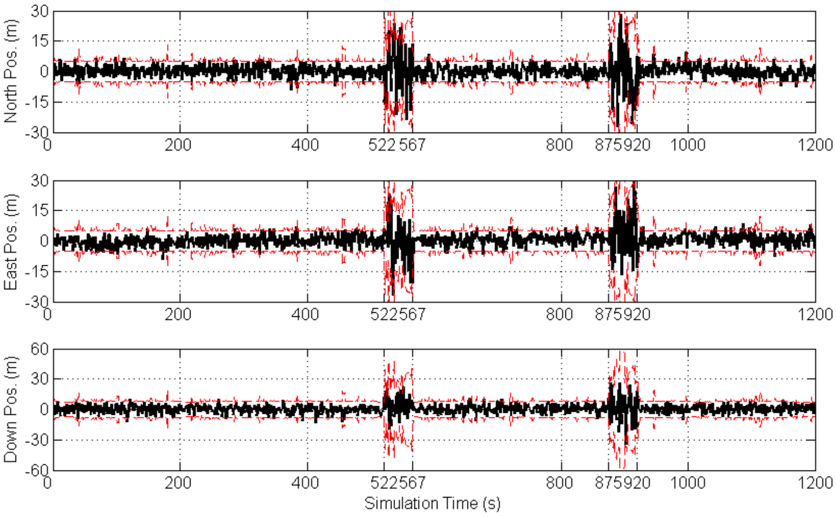


Figure 5.19: Position innovations and its 2σ uncertainty when sequential Chi-MMAE is used to detect and diagnose failures between 522-567 sec (straight line motion) and between 875-920 sec (rotational motion). This depicts the fact that the MMAE has compensated for the Chi-squared inability to detect slow changing ramp failures and applied an online filter tuning.

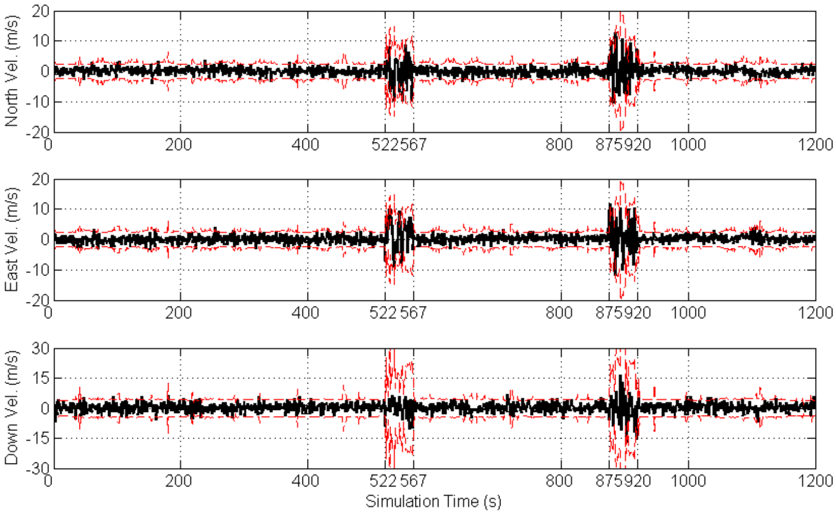


Figure 5.20: Velocity innovations and its 2σ uncertainty when sequential Chi-MMAE is used to detect and diagnose failures between 522-567 sec (straight line motion) and between 875-920 sec (rotational motion).

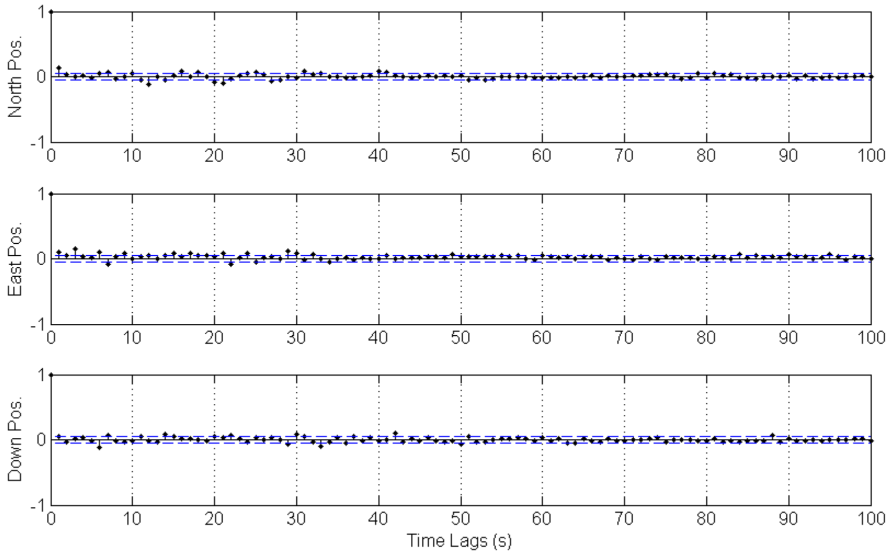


Figure 5.21: Similar to the MMAE case , the auto-correlation of the position innovation depicts the fact that the innovations sequence when the sequential algorithm is used to reject errors introduced to GPS signal are indeed uncorrelated.

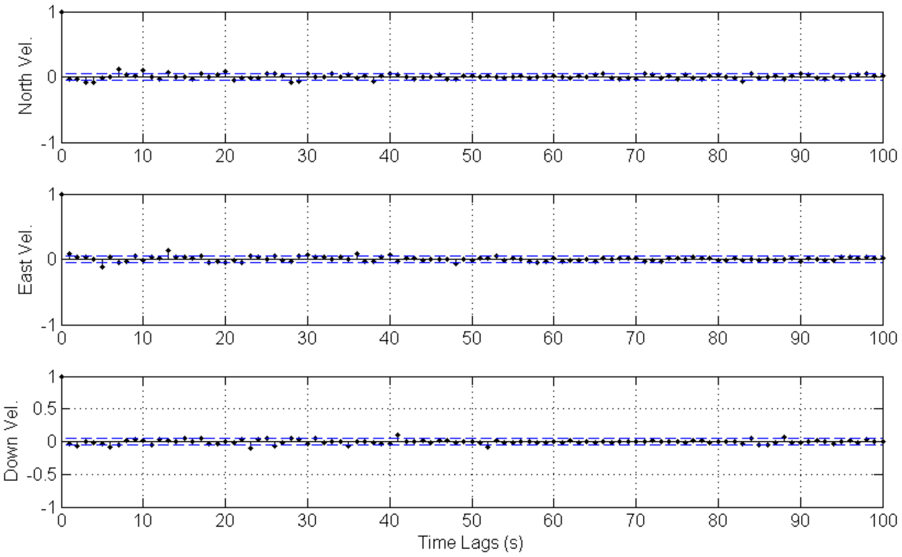


Figure 5.22: Similar to the MMAE case , the auto-correlation of the position innovation depicts the fact that the innovations sequence when the sequential algorithm is used to reject errors introduced to GPS signal are indeed uncorrelated.

5.6 Performance Verification - Actual Test

In this section the FDI performance of the χ^2, MMAE, and their sequential implementation is verified on actual inertial and GPS data. In doing so, random noise is introduced to actual GPS observations. The data used is that of Sheikh-Zayed seaport test presented earlier in section 4.6. In that test, χ^2 SCT-based FDI was implemented in both sensor level and individual observation level. This can be noticed by looking into Figure 4.48 and Figure 4.49 where spikes in the 2σ uncertainty boundaries represent jumps in one or more of the GPS fixes and the whole GPS package was discarded. Figure 5.30, shows filter estimated trajectory and corrupted GPS data. Figure 5.31 shows the resulted trajectory due to corrupted GPS logs with no FDI implemented. Figure 5.32 and Figure 5.33 show the resulted trajectory when the χ^2 SCT is applied when the filter is operated as optimal and suboptimal, respectively. To test the MMAE algorithm, a bank with five elemental filters was implemented with hypotheses and initial probabilities of

$$\mathbf{a} = [100, 10, 1, 0.1, 0.01]$$
$$p(0) = [0.025, 0.20, 0.55, 0.20, 0.025]$$

which are neither too coarse nor too fine. The sequential scheme is implemented with extra independent and nominal filter for fault detection. Figure 5.34 and Figure 5.35 show resulted trajectory when MMAE is used alone and when the sequential scheme is employed. Figure 5.37 compares the performance of both implementations while cornering and during normal mode of operation.

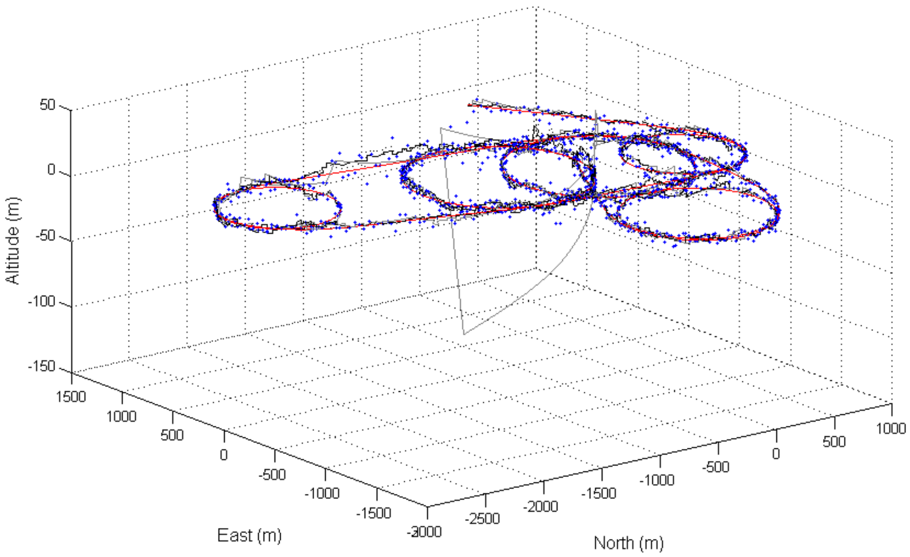


Figure 5.23: 3D view of the simulated trajectory, the GPS fixes (dots), innovations-based algorithm (solid gray), MMAE (dotted black), and sequential Chi-MMAE (solid black).

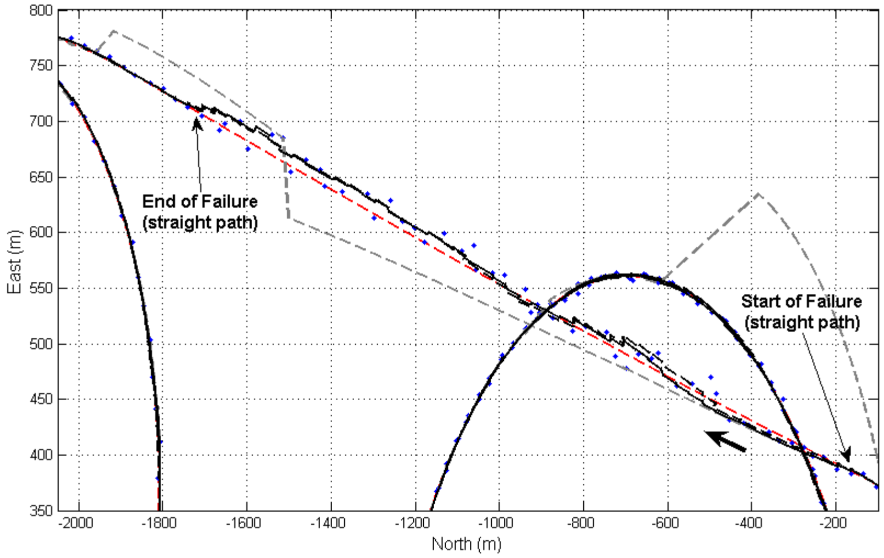


Figure 5.24: The straight line trajectory where failure occurred. The chi-squared algorithm (dashed gray) shows a poor performance compared to the sequential algorithm (solid black), which shows a better result than the MMAE (dashed black).

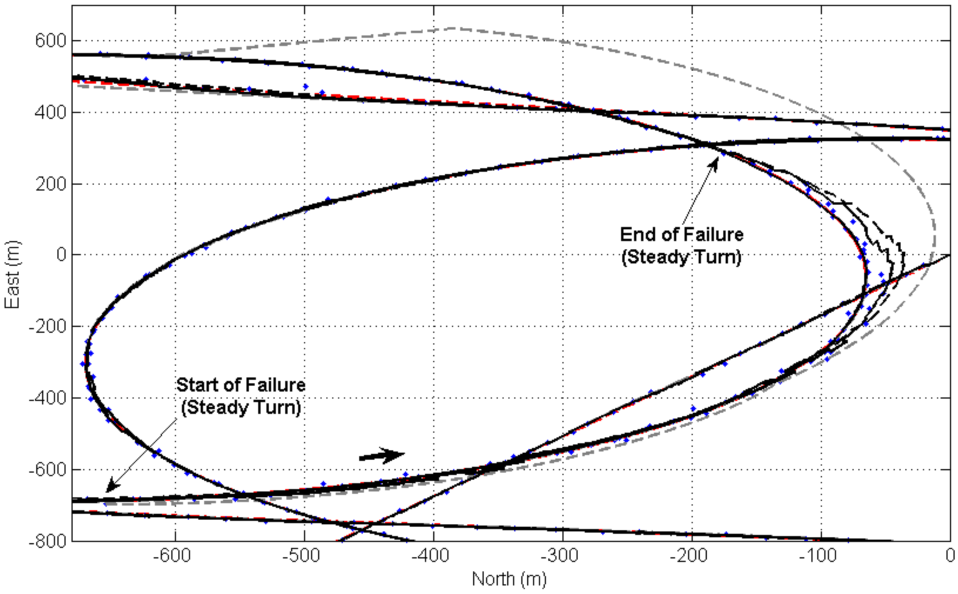


Figure 5.25: Two dimensional vehicle position showing the rotational path segment of the simulated trajectory where failure occurred. The chi-squared algorithm (dashed gray) shows a very poor trajectory compared to the sequential algorithm (solid black) In fact, the chi-squared performance is even worse than that while travelling in straight line. The sequential algorithm shows a better result than the case when MMAE alone was employed (dashed black). However, both algorithms show a degraded performance than that while travelling in straight path specially when the simulated aircraft reached the corner of the turn, this is due to the fact that while turning the attitude errors are larger than that while travelling in straight line. Moreover, for the MMAE to diagnose this failure effectively, it should converge to identical value of noise covariance. In this case, it is obvious that the bank converged to a value higher than the true noise covariance and starts to show a behavior similar to that of the innovation-based.

5.7 Summary and Conclusions

The focus of this chapter has been on achieving integrity of the INS/GPS navigation systems. I discussed mainly sensor faults. It was stated that, for a navigation system to achieve integrity it should guarantee that all the expected and unexpected faults must be detected. The FDI system shouldn't also affect the nominal operation of the filter. It should have fast response time and assist design specifications. The detection algorithm should be as simple and fast as possible, while the identification algorithm should be as efficient as possible. Moreover, the whole FDI algorithms should prove tradeoffs between complexity and performance, such that a more complex system may become desirable in order to achieved a successful identification problem. In order to achieve this goal, this chapter has:

- Provided a detailed definition and identification of the terms faults, failures, no-fault mode and fault mode. These terminologies were discussed from the Kalman filter point of view, such that a mode was considered a failure mode if one of the filter assumptions, namely, Gaussianness, whiteness, linearity and state vector order, become void.
- Discussed the general structure of an FDI system and surveyed related DFI algorithms.
- Provided a classification of failure modes based on their effects and how they are modelled in FDI systems. The first class of failures results into complete operational failure of the sensor. This kind of failures were termed hard failures. The other class of failures result into degradation in sensor's behavior. This kind of failures were termed soft failures.
- Discussed modelling of hard and soft failures in multiple model filters banks, where soft failures are modelled as a change in sensors noise covariance, while hard failures are modelled by zeroing the corresponding row in the measurement transition matrix **H**.
- Provided a formulation for the innovations and the innovations covariance at fault conditions. It was stated that during failure mode of operation, the innovations and biased and have covariance that it larger in order of magnitude than that during no-failure mode of operation.
- Discussed the design of high integrity navigation, and focussed mainly on the

Chi-squared algorithm performance as a detection scheme and on the MMAE algorithm as a complete FDI scheme.

- Proposed a sequential implementation of the chi-squared detection and MMAE identification to reduce computational complexity and provide solutions for most common FDI problems.
- Provided simulated results where faults were introduced to the GPS observations during two different motion trajectories, straight line motion and while steady turn. Both algorithms, MMAE and sequential provided similar performance in the case of straight motion. However, the sequential algorithms showed better performance in rotational motion.

This then provides the necessary background and methods to develop an appropriate and robust GPS-aided-INS navigation systems for autonomous mobile robots.

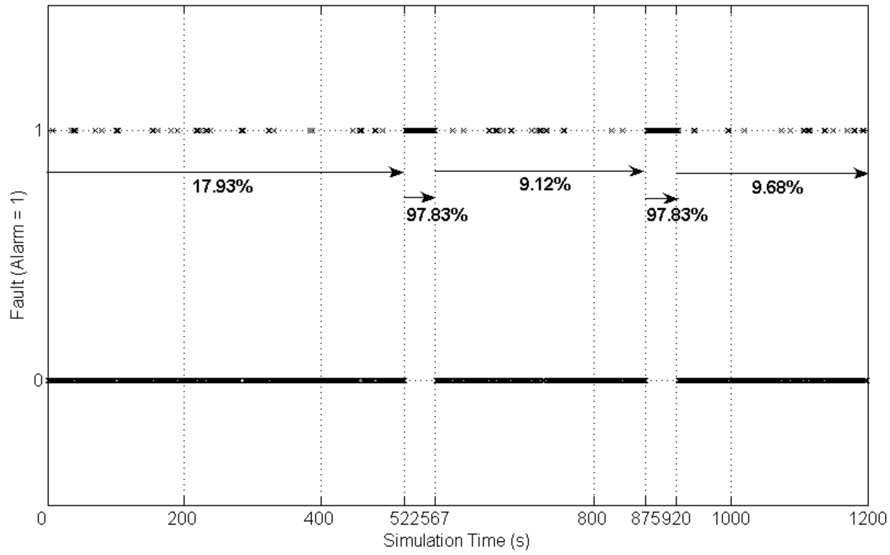


Figure 5.26: As expected from the sequential FDI algorithm, in this failure alarm plot, a 1 represents detection of fault, and the filters bank was only activated during segments of time when a failure was detected. Figures show the percentage of time when the bank was activated.

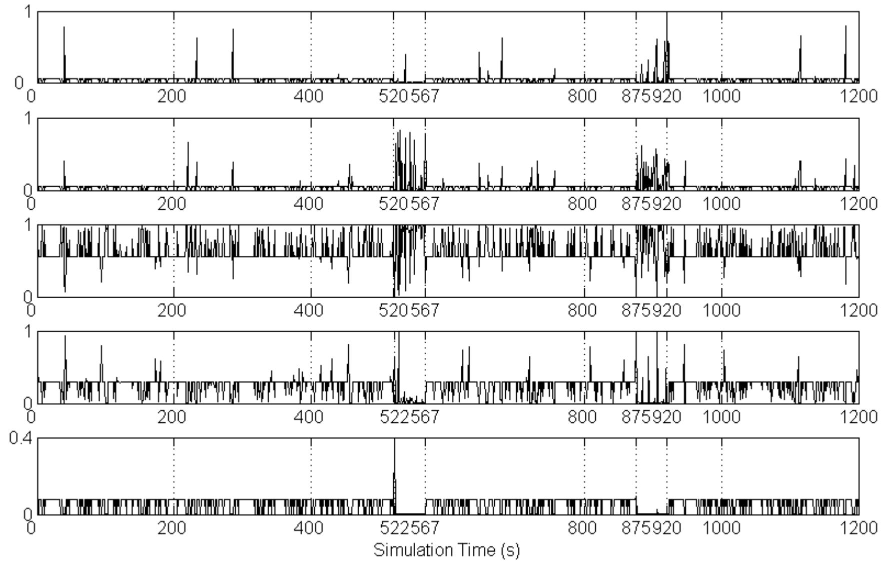


Figure 5.27: Gaussian probabilities of the five elemental filters in the sequential implementation.

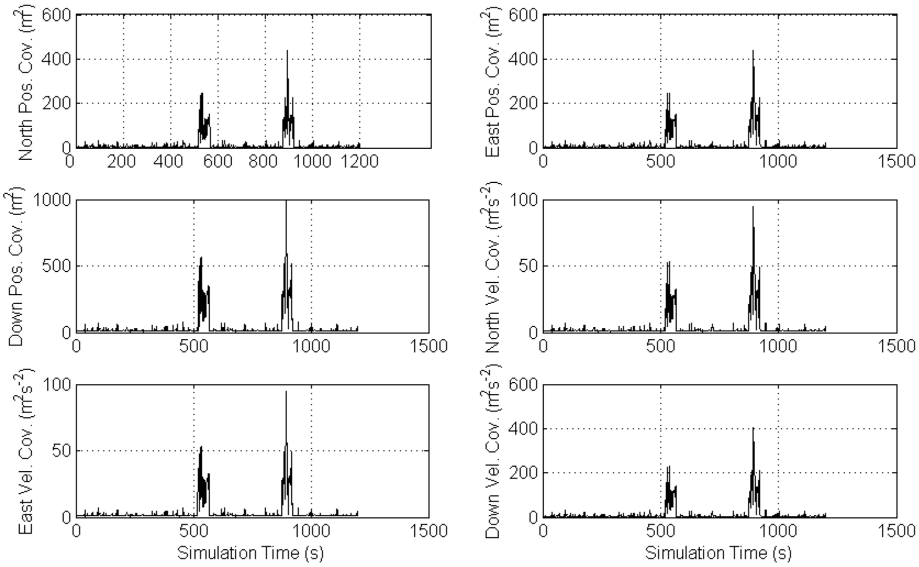


Figure 5.28: The estimated GPS noise covariance during the while simulation represents the fast response of the detection and identification algorithms in the sequential FDI algorithm.

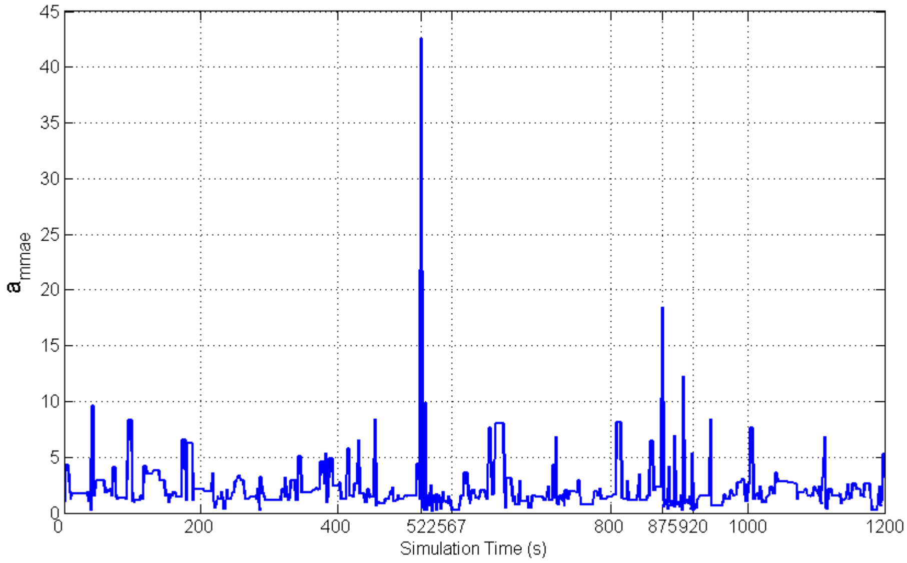


Figure 5.29: The optimal estimated hypothesis a_{MMAE} of the filters bank in the sequential algorithm.

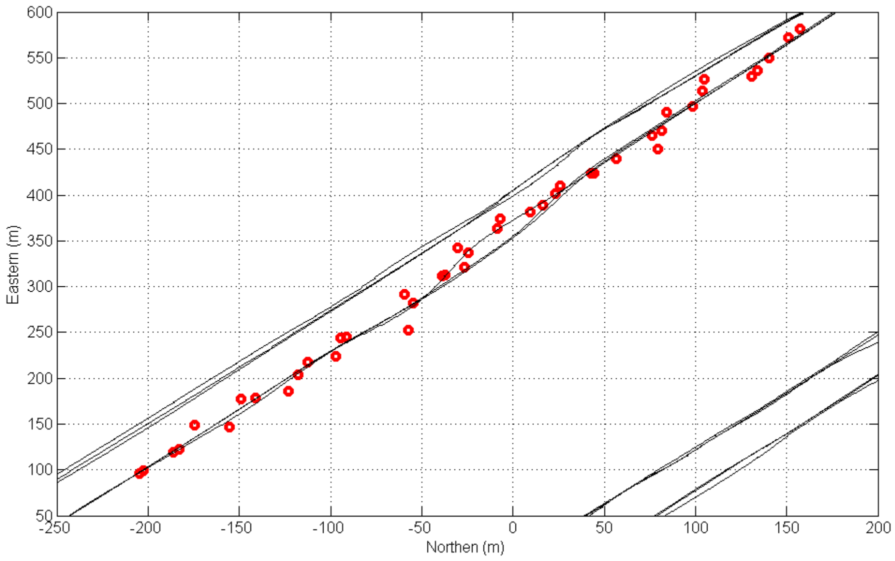


Figure 5.30: GPS data was corrupted by adding noise. Noisy GPS data are represented by circles while the filter estimated trajectory with actual GPS data are represented by the solid line.

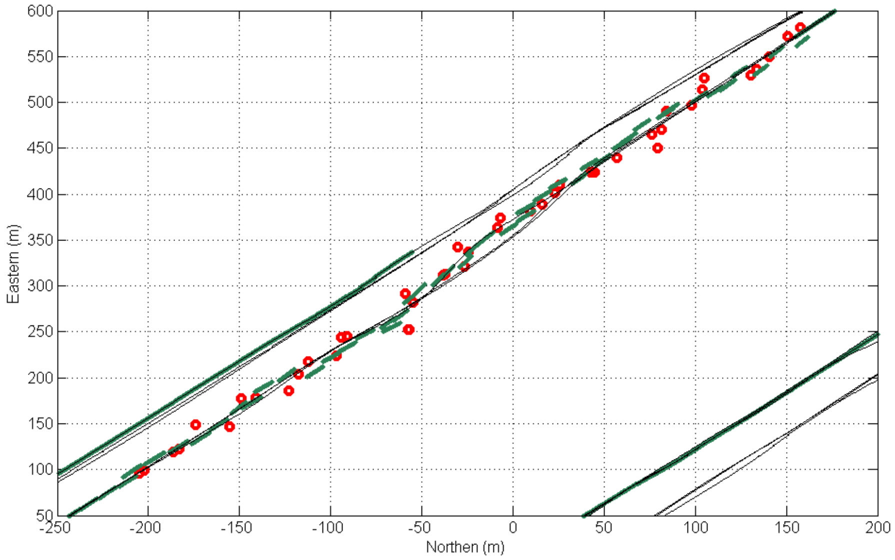


Figure 5.31: With no FDI implemented, the estimated trajectory is jumping to follow the noisy GPS observations.

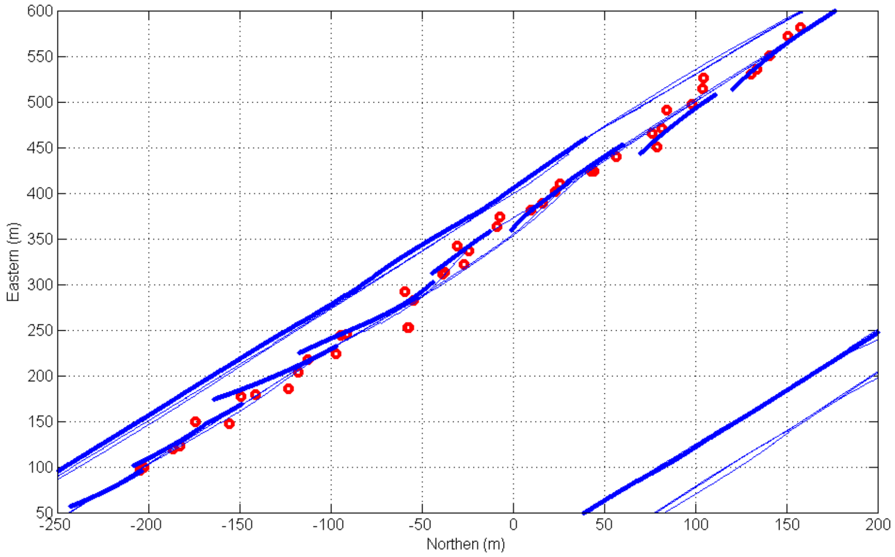


Figure 5.32: With χ^2-based FDI, inconsistent GPS observations are neglected and the filter remains in the prediction stage. This is represented by the drifting lines due to integration of inertial sensors errors. Once uncertainty reaches the level of consistency, observations are validated and the filter enters the update stage. When updated, the position is closer to the error free trajectory than the noisy GPS observations representing a successful tuning.

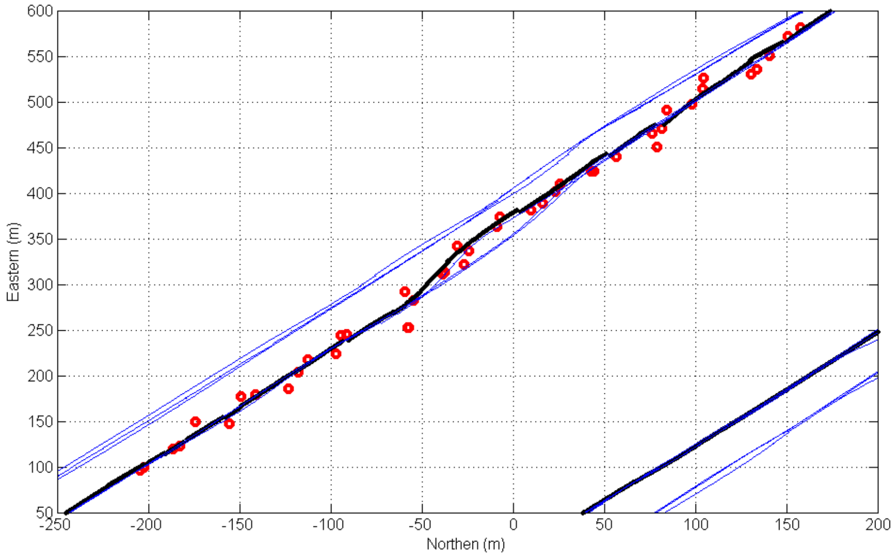


Figure 5.33: One solution to overcome the effect of GPS position observations jumps is to tune the filter as suboptimal. This is done by increasing the value of GPS position uncertainty in $\mathbf{R}(k)$. However, suboptimality affects the performance of the filter in places other than where multipath is detected.

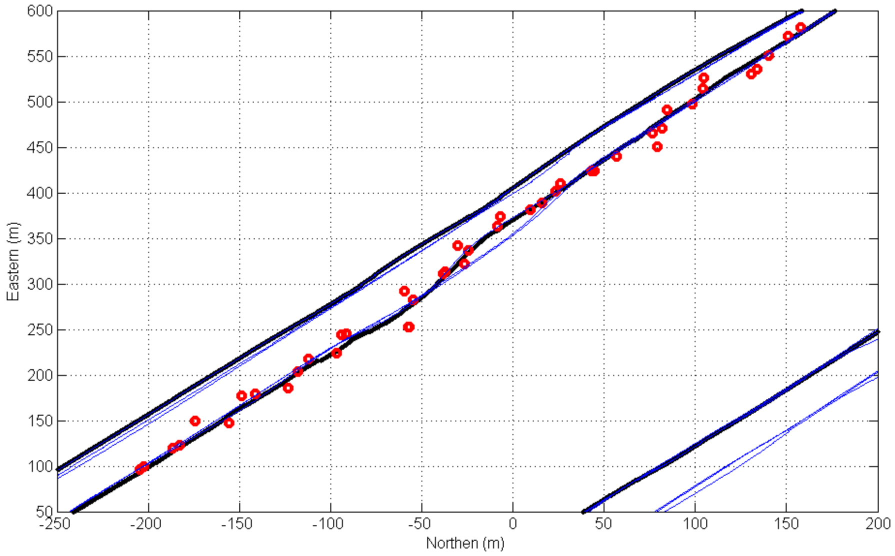


Figure 5.34: Employing MMAE as FDI scheme resulted into a successful fault identification.

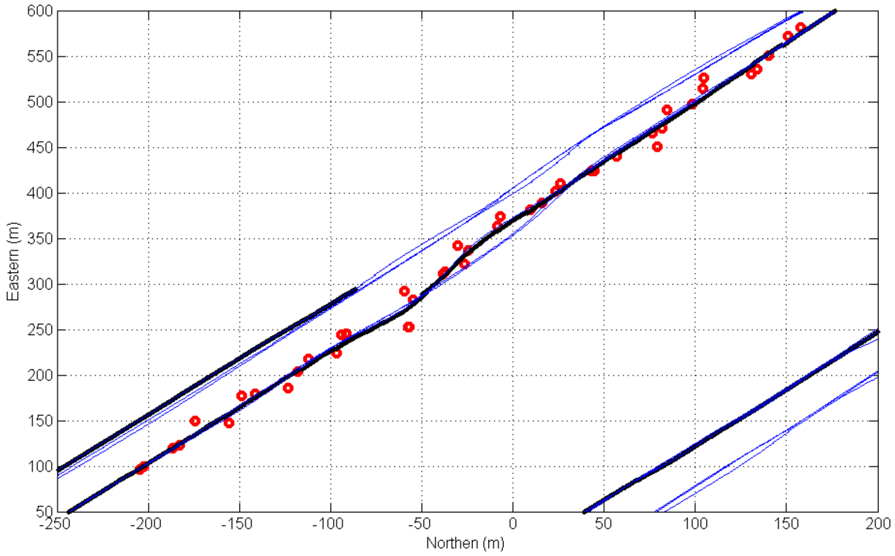


Figure 5.35: The sequential χ^2-MMAE scheme results into similar FDI performance as that of the MMAE approach during GPS failure.

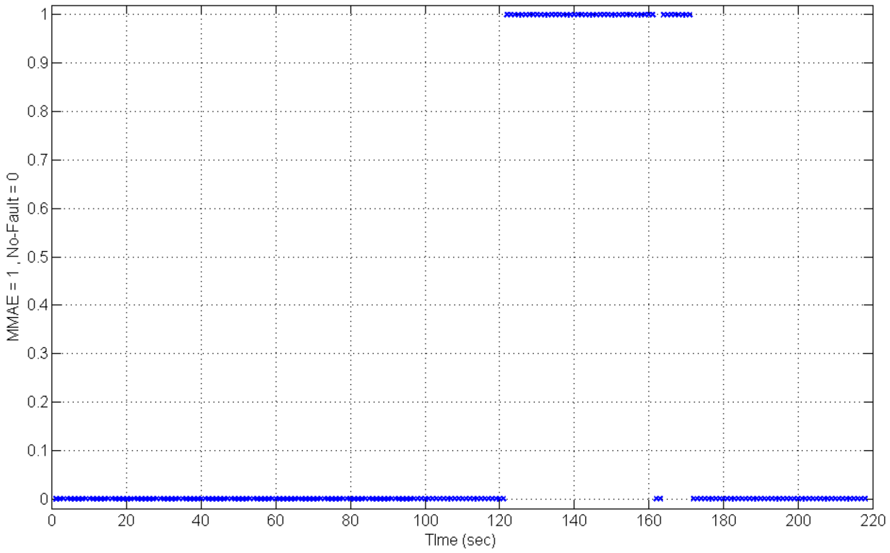


Figure 5.36: When sequential χ^2-MMAE scheme is applied, the filters bank is allowed to operate only on segments of time where faults are detected. In this alarm plot, a 1 represents a fault alarm, while a 0 represents a nominal (no-fault) operation.

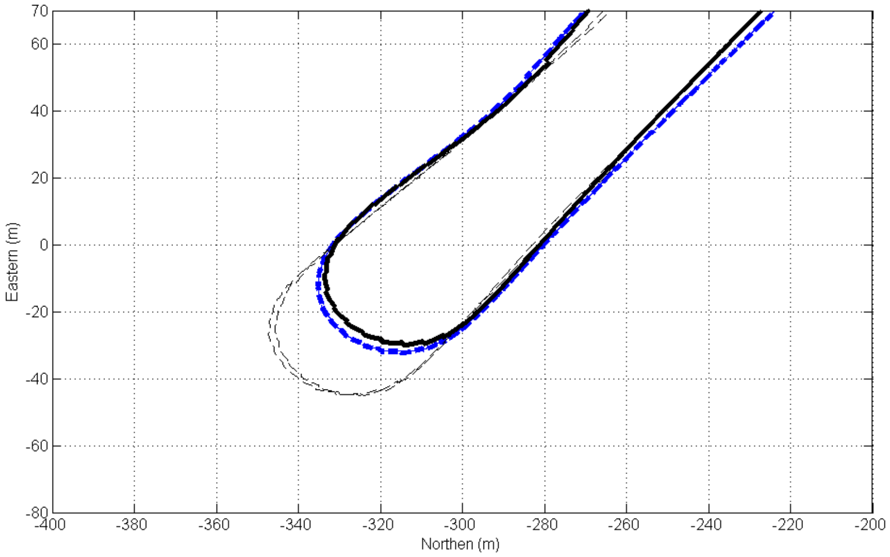


Figure 5.37: Unlike the MMAE FDI case (bold dashed line), the sequential (solid line) implementation did not affect the performance of the filter during no-fault conditions (light dashed line). This is because once the filter is in the nominal case, estimates are provided by the nominal filter which was initially tuned.

Chapter 6

Thesis Conclusions and Future Work

6.1 Conclusions and Summary of Contributions

This thesis has provided a mean for aiding inertial navigation sensors with GPS data. In doing so, the thesis has discussed principles of inertial navigation. It has derived the equations required for the local geographic (NED) and the Earth-Centered Earth-Fixed (ECEF) frames mechanisations. Local geographic frame was chosen for implementation because attitude of vehicle is more physically intuitive in the NED coordinates than in the ECEF coordinates. Also, the NED coordinates separates the unstable vertical axis from the more stable horizontal axes and provides a more intuitive schemes for analyzing INS errors than the ECEF coordinates. Unlike conventional SDINS mechanisations, a unified mathematical frame work has been used for computations, which resulted into significant reduction in mechanisation errors, and the direction cosine matrix approach was selected for attitude representation in the mechanisation process. To achieve inertial system reliability, inertial sensor errors have been analyzed and modelled. The computational algorithms for the mechanisation equations were discussed and were verified on simulated data. A comparison was conducted between the case of error free inertial sensors and when conventional low-cost inertial sensor errors are introduced. Results showed that, when low-cost inertial sensors are employed the SDINS becomes unreliable even for a short period of time. It was also stated that attitude computations are the most critical. The accuracy of attitude computation depends on the bandwidth of inertial sensors employed, mainly the gyroscopes, and the order of the algorithm. Errors in attitude computations can be reduced by increasing the bandwidth of the gyroscopes. However, errors can be reduced more rapidly and effectively by increasing the order of the algorithm. On the other hand, the effect of high frequency undesired motion such as vibrations may be reduced by employing shock absorbers. The general navigation error equations were derived in the NED and ECEF coordinates. These equations describe how uncertainties in one state affects uncertainties in other states, and therefore, lie the rules for a successful filter tuning.

This thesis also discussed The Kalman filter as being the best algorithm for providing navigational solution through the fusion of different solutions based on their accuracy. Different aspects of Kalman filter implementation were discussed. It was stated that for the filter to be operating correctly the innovations should be zero mean, white and uncorrelated. Also, the normalized innovation squared should be χ^2 distributed and its moving average should lie inside a confidence interval, computed from the χ^2 distribution tables with a given level of confidence and degrees of freedom. Based on filter innovations behavior, filter tuning was discussed and a simplified tuning algorithm was implemented. The details of a loosely-coupled and a tightly-coupled INS/GPS data fusion algorithms using the Kalman filter were presented. In doing so, GPS observables, mainly pseudoranges, GPS satellite orbital parameters, ephemeris data processing and calculating satellites and receiver positions were discussed and all algorithms were verified via actual data processing.

A loosely-coupled linear filter was implemented. The state vector included fifteen states (NED positions, NED velocities, Euler angles and accelerometers and gyros biases). The performance of the filter was verified on simulated data before proceeding to actual data that was post-processed. Results represented filter testing for whiteness and unbiasedness, position and velocity innovations and innovations auto-correlations. Results showed the effect of filter tuning on attitude and bias estimates, specially while cornering. The effect of maneuvering on the observability of the attitude states uncertainty was introduced. Simulated maneuvers included right-hand rotations with few straight line motions. Results showed that attitude uncertainty grows significantly while travelling in straight line with no change in vehicle acceleration, while attitude states become observable while banking.

The vulnerability of low-cost INS/GPS systems to failures compromises the consistency of the estimator and therefore the integrity of the navigation system. Therefore, the navigation algorithm should provide a mean of supervising the integration process. While this implies that the filter should be adaptive, the navigation algorithm should detect faults in any of its components, whether it is hardware or software, and take the required reaction to identify and isolate the faulty party. The fault diagnosis system should:

- Guarantee the detectability of all faults. In doing so, the frequency domain can be utilized to make sure that all faults will be transmitted to the innovations.
- Do not affect the performance of the navigation system during no-fault condition.
- Assists design specifications, including highest frequency maneuvers, required accuracy and minimum cost.
- Have short response time.
- Offer tradeoff between complexity and FDI performance.

This motivated the investigation of different approaches to fault detection and identification. Process reliability has been already considered by modelling of inertial sensor errors, designing a reliable mechanisation, including bias error states in the filter estimated state vector (Scheding, S., 1997), designing appropriate maneuvers and by appropriate tuning. Therefore, only aiding sensors faults were discussed. Failures (or faults) were identified as situations where at least one of the Kalman filter assumptions becomes void. These assumptions include; filter states and observations are assumed gaussian random variables, the process and observations noise are assumed white, uncorrelated with known covariance, the system is assumed linear (works with linear filter implementation only), and the state vector of the assumed order.

The fault detection and identification problem was viewed as having two primary stages; residuals generation and decision making. The residual generation stage is responsible for the generation of filter residuals which carry failure signature and therefore to make sure that faults are transmitted to the residuals. On the other hand, decision making is responsible for the detection of faults and making decisions regarding the fusion process. Decision making can take the form of a binary decision whether something went wrong or not, or it may include parameters and/or states estimation. Sensor failures were classified into two categories, soft failures and hard failures. Soft failures results into degradation of sensors accuracy, such failure can occur from multipath, atmospheric errors, jamming/interference and errors in orbital and ephemeris parameters. Soft failures are modelled as increment in the sensor noise covariance matrix. On the other hand, a hard failure results into a huge degradation in the sensor performance, such failure can occur from power failure, physical damage and low satellites availability. Hard failures are modelled by zeroing the corresponding row in the sensor observations transition matrix $\mathbf{H}(k)$. The FDI problem was formulated. It was stated that under fault conditions the innovations are simply the innovation under no fault condition plus a vector representing the fault effect on the innovations. Therefore, under faulty conditions, the innovations are biased. On the other hand, the innovation covariance required to maintain the consistency of the estimator increases during faults.

Finally, the problem of high integrity navigation system design in real-time was discussed, where an innovation-based technique, namely the χ^2, was selected as the detection scheme and the MMAE scheme was selected for fault identification and parameters/states estimation. Both techniques were implemented in a *sequential* form, where the fault identification stage is activated by the detection stage once a fault is detected. When tested on actual inertial and GPS data, such a sequential implementation produced comparable identification ability as the MMAE alone approach and reduced system complexity, where the identification approach is allowed only to operate in segments of time where faults are detected. On the other hand, unlike the MMAE algorithm where the performance of the filter was affected during no-fault conditions, the sequential scheme guaranteed the consistency of the estimator in all of its modes of operation and didn't affect its performance during normal modes of operation. It also provided a mean of checking the consistency of the estimator for both FDI stages.

6.2 Future Work

As a continuation from this work, the future focus should be on the improvement of the estimator performance. Since the linear loosely-coupled filter was used in this thesis and as it was discussed in section 4.6 when real tests were represented, the estimator considers the linearity of the dynamic model as well as the states. Therefore, a nonlinear filter, such as the unscented Kalman filter (UKF) must be used (Julier, S., & Uhlmann, J., K., 1996), (Julier, S., & Uhlmann, J., K., 1997), (Julier, S., Uhlmann, J. & Durrent-Whyte, H., F., 2000), (Lefebvre, T., Bruynincks, H., & De Schutter, J., 2001), (Julier, S., Uhlmann, J. & Durrent-Whyte, H., F., 2002), (Haykin, S., 2001), (Crassidis, J., L., 2005), (Zhang, P., Gu, J., Milios, E., E. & Huynh, P.,). The UKF has better performance than the EKF, because it propagates both the mean and the covariance through the nonlinear model, and has less computational cost than the EKF, because it does not compute the Jacobian. The UKF should be built in the tightly-coupled configuration. As discussed earlier in section 4.4, raw GPS measurements and ephemeris data should be used to form the filter observations. The fact that, both the loosely-coupled and tightly-coupled filters have similar performance when the GPS signal from at least four satellites is available. However, during low-availability of GPS signal, the performance of the tightly-coupled filter overcomes that of the loosely-coupled one. This degradation in the loosely-coupled filter compared to the tightly-coupled one is because the loosely-coupled filter uses the position and velocity estimates of the reduced order filter (because it does not include attitude and biases states) implemented in the GPS receiver.

The observability of the filter states should be thoroughly investigated. As it was represented in section 4.5, unlike position and velocity errors, the uncertainty of attitude errors does not represent the state accurately. In fact, the uncertainty increase while the vehicle is travelling in straight line and constant acceleration, this is more obvious in the heading case, and it is updated once the vehicle is turning. This leads to a conclusion that, even though attitude and bias estimates are not directly observed by the filter, they can become observable by maneuvering. Therefore, as a part of the path planning, the effect of vehicle maneuvering and dynamics on the Kalman filter states observability must be investigated.

Appendix A

Attitude Representations

A transformation matrix is required to transform accelerations sensed by strapdown accelerometers into the desired coordinates frame. This transformation matrix consists of the roll, pitch and yaw angles. Attitude computation is critical in inertial systems; errors in estimated attitude cause significant errors in computations when integrated over time. Algorithms available include Euler representation, direction cosine matrix and the quaternion approach (Sukkarieh, S., 2000)(Titterton, D., & Weston, W., 2004) (George, M. G., 2007)(Abdel-Hafez, M., 2003)(Nebot, E. M., 2005).

Regardless of the type of the attitude representation employed, the result will be identical. However, they differ in term of the computational requirements (which is dimensional dependent) and the amount of errors introduced due to discretisation or simplification of the computations. In this thesis, the direction cosine approach is utilized, since it is commonly used in low-cost inertial navigation systems, offers the best representation, has minimal trigonometric calculations and the least expensive (Sukkarieh, S., 2000).

A.1 Euler Representation

Euler representation is the easiest to understand. Body rates, sensed by strapdown gyroscopes, are used to compute the three Euler angles (roll ϕ, pitch θ and yaw ψ).

Equation A.2 is used to compute Euler angles from sensed body rates

$$\dot{\mathbf{\Psi}} = \mathbf{E}\omega_{ib}^{b} \tag{A.1}$$

$$= \begin{pmatrix} 1 & \frac{\sin\phi\sin\theta}{\cos\theta} & \frac{\cos\phi\sin\theta}{\cos\theta} \\ 0 & \cos\phi & -\sin\phi \\ 0 & \frac{\sin\phi}{\cos\theta} & \frac{\cos\phi}{\cos\theta} \end{pmatrix} \omega_{ib}^{b} \tag{A.2}$$

where

$$\dot{\mathbf{\Psi}} = \begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} \tag{A.3}$$

and $\dot{\phi}, \dot{\theta}$ and $\dot{\psi}$ are Euler rates. Estimated Euler angles are then used to compute the desired Euler transformation matrix in equation A.5. This transformation is derived by three rotation sequence which can be written as three direction cosine matrices.

$$\boldsymbol{C}_b^n = \begin{pmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi & \cos\phi \end{pmatrix} \tag{A.4}$$

$$= \begin{pmatrix} \cos\theta\cos\psi & \sin\phi\sin\theta\cos\psi - \cos\phi\sin\psi & \cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi \\ \cos\theta\sin\psi & \sin\phi\sin\theta\sin\psi + \cos\phi\cos\psi & \cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi \\ -\sin\theta & \sin\phi\cos\theta & \cos\phi\cos\theta \end{pmatrix} \tag{A.5}$$

A.2 The Direction Cosine Matrix

The direction cosine approach attempts to solve the general matrix differential equation

$$\dot{\boldsymbol{C}} = \boldsymbol{C}\boldsymbol{\Omega} \tag{A.6}$$

where $\boldsymbol{\Omega}$ is the skew matrix formed from the elements of the turn rate vector ω, for example

$$\dot{\boldsymbol{C}}_b^n = \boldsymbol{C}_b^n \boldsymbol{\Omega}_{nb}^b \tag{A.7}$$

where

$$\boldsymbol{\Omega}_{nb}^b = \left[\omega_{nb}^b \times\right] \tag{A.8}$$

and $\boldsymbol{C}_b^n$ is the DCM transforming vectors from the body frame coordinates into navigation frame coordinates. The DCM approach procedure is as follows:

- Initial orientation of the vehicle, represented by Euler angles, is estimated by initial alignment of the IMU.
- Initial $\boldsymbol{C}_b^n$ is computed by equation A.5.
- Body rotations, sensed by strapdown gyros, are used to estimate $\boldsymbol{\Omega}_{nb}^b$.
- Equation A.7 is then used to update the DCM.

The direction cosine approach in the attitude algorithm utilized in this work.

A.3 The Quaternion Approach

In the Quaternion approach the transformation from one frame to another is accomplished by a single rotation about a vector $\mathbf{q}$ through an angle q. It contains four parameters; i.e., three to describe the vector and one to describe the magnitude of the angle of rotation. The quaternion may be calculated from elements of the direction cosine matrix as follows

$$q(i) = \begin{pmatrix} \frac{1}{2}\sqrt{1 + \boldsymbol{C}_{b\,11}^n + \boldsymbol{C}_{b\,22}^n + \boldsymbol{C}_{b\,33}^n} \\ \frac{1}{4q(1)}\left(\boldsymbol{C}_{b\,32}^n - \boldsymbol{C}_{b\,23}^n\right) \\ \frac{1}{4q(1)}\left(\boldsymbol{C}_{b\,13}^n - \boldsymbol{C}_{b\,31}^n\right) \\ \frac{1}{4q(1)}\left(\boldsymbol{C}_{b\,21}^n - \boldsymbol{C}_{b\,12}^n\right) \end{pmatrix} \tag{A.9}$$

The discretization of the quaternion approach (Sukkarieh, S., 2000) starts by integrating gyro outputs to determine the change in the angle in the body frame

$$\phi = \int \omega .dt \tag{A.10}$$

$$\phi = \sqrt{\phi_x^2 + \phi_y^2 + \phi_z^2} \tag{A.11}$$

Then, the quaternion operator is formulated as

$$h(i) = \begin{pmatrix} \delta \\ \gamma\phi_x \\ \gamma\phi_y \\ \gamma\phi_z \end{pmatrix} \tag{A.12}$$

where $\gamma = \frac{\sin\frac{\phi}{2}}{2}$ and $\delta = \cos\frac{\phi}{2}$. Finally, the quaternion is updated as follows

$$q(i+1) = \begin{pmatrix} q(i)_1 & -q(i)_2 & -q(i)_3 & -q(i)_4 \\ q(i)_2 & q(i)_1 & -q(i)_4 & q(i)_3 \\ q(i)_3 & q(i)_4 & q(i)_1 & -q(i)_2 \\ q(i)_4 & -q(i)_3 & q(i)_2 & q(i)_1 \end{pmatrix} h(i) \tag{A.13}$$

To obtain accelerations in the desired mechanization frame, the quaternion is used to transform measured accelerations in the body frame into the mechanization frame. If the mechanization frame is selected as the navigation frame, then acceleration data is transformed as follows

$$\begin{aligned} \boldsymbol{f}^n &= \boldsymbol{f}^b \boldsymbol{C}_b^n \\ &= q(i+1)\boldsymbol{f}^{b^T} q^*(i+1) \end{aligned} \tag{A.14}$$

It is less expensive to convert quaternion back into $\boldsymbol{C}_b^n$ than to use the complex formula, then

$$\boldsymbol{C}_b^n = \begin{pmatrix} (q_1^2 + q_2^2 - q_3^2 - q_4^2) & 2\,(q_3 q_2 - q_1 q_4) & 2\,(q_2 q_4 + q_1 q_3) \\ 2\,(q_2 q_3 + q_1 q_4) & (q_1^2 - q_2^2 + q_3^2 - q_4^2) & 2\,(q_{31} q_4 - q_1 q_2) \\ 2\,(q_2 q_4 - q_1 q_3) & 2\,(q_3 q_4 + q_1 q_2) & (q_1^2 - q_2^2 - q_3^2 + q_4^2) \end{pmatrix} \tag{A.15}$$

and the direction cosine approach is applied to transform body accelerations into the desired mechanization frame.

Appendix B

Earth's Shape, Reference Frames and Coordinate Systems

Inertial navigation systems varies with respect to the coordinate system in which navigation equations are mechanized. In this chapter two concept are discussed, reference frames and coordinate systems (Kelly, A, 1994). Reference frames are physics related, for example, the Earth's rotational rate with respect to the inertial frame is represented as Ω_{ie}. In this example the reference frame is the inertial frame. On the other hand, coordinate systems are mathematics related. Their main purpose is visual or reduction of computational complexity (George, M. G., 2007). Back to the previous example, if navigational frame mechanization is desired, then the Earth's rotational rate with respect to the inertial frame must be transformed into the local level frame. This is represented as Ω_{ie}^{n}. In this example, the reference frame is the inertial frame, while the coordinate system is the local level (or navigation) frame.

B.1 Modelling of the Earth's Shape

The surface of the Earth is irregular in shape and can be modelled in different ways (Titterton, D., & Weston, W., 2004). Topographic models, which represent the physical shape of the Earth and the mean level of the oceans, and the Geodetic models, which yields a surface 'the geoid' that is everywhere perpendicular to the local gravity component.

Table B.1: WGS-84 ellipsoid parameters

Parameter	**Value**	**Unit**	**Relations**
Semimajor axis a	6378137	m	
Semiminor axis b	6356752.3142	m	
Eccentricity of the Earth ellipsoid e	0.0818191908426	dimensionless	$e = \sqrt{1 - \frac{b^2}{a^2}}$
Second eccentricity $\acute{e}$	$\sqrt{0.00673949674228}$	dimensionless	$\acute{e} = \sqrt{\frac{a^2}{b^2} - 1}$
Flattening (Ellipticity) f	0.00335281066474	dimensionless	$f = \frac{a-b}{a}$
Earth's angular velocity Ω_{ie}	7.292115×10^{-5}	rads	
Earth's Gravitational Constant μ	$3986004.418 \times 10^{8}$	m^3s^2	
Mass of the Earth M	5.9733328×10^{24}	Kg	
Speed of the Light c	2.99792459×10^{8}	ms	
Pi π	3.1415926535898	dimensionless	

For GPS applications, the standard physical model of the Earth is the World Geodetic System 1984 (WGS-84). In WGS-84 the Earth's shape is ellipsoidal, where cross-sections of the Earth parallel to the equatorial plane are circular and the equatorial cross-section has radius $6,378.137km$ (which is the mean equatorial radius of the Earth). While the cross-sections of the Earth normal to the equatorial plane are ellipsoidal. The semimajor axis, a, coincides with the equatorial diameter of the earth and has the same length as the equatorial radius. While the semiminor axis, b, coincides with the polar diameter of the Earth and has a radius of $6,356.7523142km$. The WGS-84 parameters and relations are summarized in table B.1.

The WSG-84 standards define the parameters of the *ellipsoid*. However, the surface of the Earth is not flat. It contains surfaces with different heights (terrain). The surface of the earth, that is everywhere normal to gravity is referred to as the *geoid*. It corresponds to the global mean sea level. The geodetic hight is the hight above the surface of the ellipsoid. This surface provides no physical reference point for measurement. It is the geoid, i.e. mean sea level, that offers much more convenient

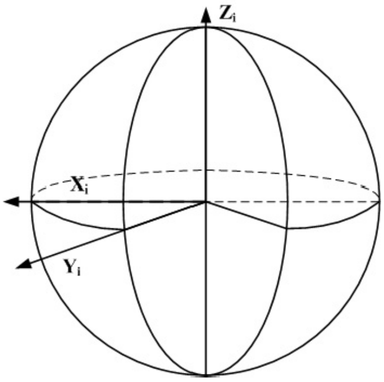


Figure B.1: The ECI reference frame

vertical reference (Titterton, D., & Weston, W., 2004). The measured height above the geoid is referred to as *topographic height*.

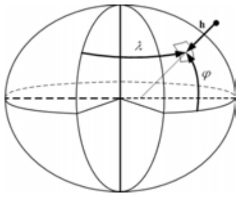
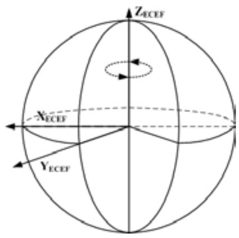
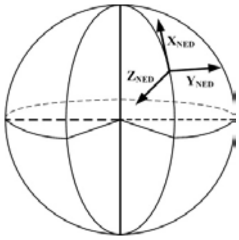
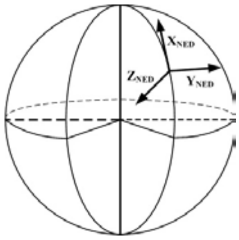
B.2 Earth Centered Inertial (ECI) Reference Frame

The inertial frame can be defined as one in which Newton's lows apply. Such as frame must intuitively be non-accelerating and non-rotating. The most common choice for this frame is the Earth Center Inertial frame (ECI) as shown in Figure **??**. The origin of the frame is at the Earth's geometric center with its z axis pointing up through the geometric north pole, its x axis pointing to the vernal equinox and its y axis completing the orthogonal right-handed set. This frame does not rotate with the Earth.

B.3 Coordinate Systems

The coordinate systems considered in this thesis are Earth stabilized systems, i.e. systems which maintain orientation referenced in variance ways to the Earth. These coordinate systems are represented in table B.2.

Table B.2: Earth stabilized coordinate systems

Coordinate system	**Description**	**Representation**
Geodetic coordinate system (llh)	(1) Fixed w.r.t rotating Earth (2) Origin: Earth's geometric center (3) Geodetic Latitude: angle between the ellipsoid normal vector and its projection into the equatorial plane (4) Geocentric latitude: angle relative to the ellipsoid geometric center (5) longitude: angle relative to Greenwich, England (6) Height: minimum distance between user and reference ellipsoid	
Earth Centered Earth Fixed (ECEF)	(1) Fixed w.r.t rotating Earth (2) Origin: Earth's geometric center (3) z-axis: points up through north pole (4) x-axis: points to zero longitude (5) y-axis: completes the set	
Local Level Coordinate System (ES)	(1) Fixed to the Earth's surface at a defined latitude, longitude and height (2) x-axis: points to the local north (3) y-axis: points to the local east (4) xy-plane: tangential to the Earth's surface (5) z-axis: points to the Earth's center (6) Used for navigation in small areas	
Geographic Coordinate System (NED)	(1) a local level frame defined at the vehicle's current position (2) Moves with the vehicle (3) x-axis: tangent to Earth's surface and points to the north (4) y-axis: tangent to Earth's surface and points to the east (5) z-axis: completes the set and positive to down (6) Most common mechanization system (7) Useful for large missions	

Appendix C

Derivations

This chapter provides derivations of mathematical formulations of the strapdown inertial navigation and Kalman filter algorithms utilized in this thesis. It is considered as a part of its literature survey.

C.1 Strapdown Inertial Navigation System (SDINS) Equations

In this section, equations of strapdown inertial navigation are derived. They include the development of different attitude representations, derivation of velocity and position dynamics, and derivation of the local gravity component. For more detailed discussion on inertial navigation equations refer to (Kelly, A, 1994).

C.1.1 Mechanization Equations

Assume an arbitrary rigid body that is located in the ECEF, as represented by figure C.1. For the time being, ignore the m-frame and assume that the body is really located in the ECEF frame. The ECEF velocity of the body is expressed as

$$\mathbf{v}_e = \left(\frac{d\mathbf{p}}{dt}\right)^e \tag{C.1}$$

The Coriolis theorem is the connection between different frames; it represents velocity in one frame as it would be seen on another frame. By applying Coriolis theorem

$$\left(\frac{d\mathbf{p}}{dt}\right)^i = \mathbf{v}_e + \Omega_{ie} \times \mathbf{p} \tag{C.2}$$

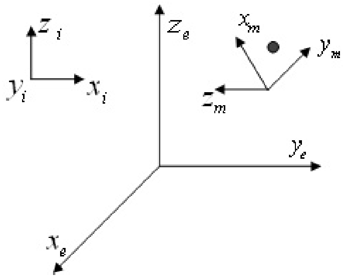


Figure C.1: The position of the assumed body with respect to different coordinate frames.

and by taking the derivative of equation C.2

$$\left(\frac{d^2\mathbf{p}}{dt^2}\right)^i = \left(\frac{\mathbf{v}_e}{dt}\right)^i + \Omega_{ie} \times \left(\frac{d\mathbf{p}}{dt}\right)^i + \dot{\Omega}_{ie} \times \mathbf{p} \tag{C.3}$$

Since the angular acceleration of the Earth's rotation is zero, $\dot{\Omega}_{ie} = 0$. Then, by substituting equation C.2 into equation C.3

$$\begin{aligned}\left(\frac{d^2\mathbf{p}}{dt^2}\right)^i &= \left(\frac{\mathbf{v}_e}{dt}\right)^i + \Omega_{ie} \times (\mathbf{v}_e + \Omega_{ie} \times \mathbf{p}) \\ &= \left(\frac{\mathbf{v}_e}{dt}\right)^i + \Omega_{ie} \times \mathbf{v}_e + \Omega_{ie} \times (\Omega_{ie} \times \mathbf{p})\end{aligned} \tag{C.4}$$

Now, return to figure C.1, assume the m-frame. This is referred to as the *mechanization* frame. By applying Coriolis theorem

$$\left(\frac{d\mathbf{v}_e}{dt}\right)^i = \left(\frac{d\mathbf{v}_e}{dt}\right)^m + \omega_{im} \times \mathbf{v}_e \tag{C.5}$$

substituting into equation C.4 yields

$$\left(\frac{d^2\mathbf{p}}{dt^2}\right)^i = \left(\frac{d\mathbf{v}_e}{dt}\right)^m + (\omega_{im} + \Omega_{ie}) \times \mathbf{v}_e + \Omega_{ie} \times (\Omega_{ie} \times \mathbf{p}) \tag{C.6}$$

$\left(\frac{d^2\mathbf{p}}{dt^2}\right)^i$ is the total inertial acceleration of the body under consideration. This quantity is the sum of the gravitational acceleration and the specific force measured by the inertial sensors.

$$\mathbf{f}^i + \mathbf{g}_m(\mathbf{p}) = \left(\frac{d\mathbf{v}_e}{dt}\right)^m + (\omega_{im} + \Omega_{ie}) \times \mathbf{v}_e + \Omega_{ie} \times (\Omega_{ie} \times \mathbf{p}) \tag{C.7}$$

where

$(\omega_{im} + \Omega_{ie}) \times \mathbf{v}_e$ is the Coriolis acceleration term.

$\Omega_{ie} \times (\Omega_{ie} \times \mathbf{p})$ is the centripetal acceleration term.

$\mathbf{f}^i$ is the specific force acceleration sensed by accelerometers.

$\mathbf{g}(\mathbf{p})$ is the gravitational acceleration component (that is position dependent) in the mechanization frame.

$\left(\frac{d\mathbf{v}_e}{dt}\right)^m$ is the acceleration of the body in the mechanization frame.

ω_{im} is the angular rate of rotation of the mechanization frame with respect to the inertial frame.

Ω_{ie} is the angular rate of rotation of the Earth with respect to the inertial frame.

$\mathbf{v}_e$ is the ground velocity of the body in the ECEF.

$\mathbf{p}$ is the position of the body in the mechanization frame.

Since both the gravitational and centripetal accelerations are position dependent, they are combined to form the local gravity vector

$$\mathbf{g}_l(\mathbf{p}) = \mathbf{g}_m(\mathbf{p}) - \Omega_{ie} \times (\Omega_{ie} \times \mathbf{p}) \tag{C.8}$$

and the navigation equation becomes

$$\left(\frac{d\mathbf{v}_e}{dt}\right)^m = \mathbf{C}_b^i \mathbf{f}^b - (\omega_{im} + \Omega_{ie}) \times \mathbf{v}_e + \mathbf{g}_l(\mathbf{p}) \tag{C.9}$$

As stated by the i upper-script, equation C.9 was derived with respect to the inertial frame and the velocity assumed was with respect to the ECEF. The navigation equation in C.9 can be expressed in different coordinate frames as

$$\left(\frac{d\mathbf{v}_e}{dt}\right)^m = \mathbf{C}_b^m\mathbf{f}^b - (\omega_{em}^m + 2\Omega_{ie}^m) \times \mathbf{v}^m + \mathbf{g}_l^m(\mathbf{p}) \tag{C.10}$$

where ω_{em}^m is the transport rate, which is the angular rate of rotation of the mechanization frame with respect to the ECEF frame, and $\omega_{im}^m = \omega_{ie}^m + \omega_{em}^m$. For example, if the navigation frame is considered as the mechanization frame, then equation C.10 becomes

$$\dot{\mathbf{v}}^n = \mathbf{C}_b^n\mathbf{f}^b - (\omega_{en}^n + 2\omega_{ie}^n) \times \mathbf{v}^n + \mathbf{g}_l^n(\mathbf{p}) \tag{C.11}$$

In the next subsection, an expression for the local gravity component is derived.

C.1.2 General Error Equations

Attitude Equations

Here we derive error equations in attitude computation. The analysis starts by navigation frame case and then generalize it for other coordinate frame systems.
Starting by equation C.12

$$\tilde{\mathbf{C}}_b^n = [\mathbf{I} - [\delta\psi\times]]\mathbf{C}_b^n \tag{C.12}$$

where

$\tilde{\mathbf{C}}_b^n$ is the estimated DCM.

$\mathbf{C}_b^n$ is the true DCM.

$\delta\psi\times$ is the skew symmetric matrix of misalignments.

rearranging and differentiating for $[\delta\psi\times]$ results into

$$[\delta\dot{\psi}\times] = -\dot{\tilde{\mathbf{C}}}_b^n\mathbf{C}_b^{nT} - \tilde{\mathbf{C}}_b^n\dot{\mathbf{C}}_b^{n\,T} \tag{C.13}$$

given that

$$\dot{\mathbf{C}}_b^n = \mathbf{C}_b^n\mathbf{\Omega}_{ib}^b - \mathbf{\Omega}_{in}^n\mathbf{C}_b^n \tag{C.14}$$

taking its transpose results into

$$\dot{\mathbf{C}}_b^{n^T} = -\boldsymbol{\Omega}_{ib}^b \dot{\mathbf{C}}_b^{n^T} - \dot{\mathbf{C}}_b^{n^T} \boldsymbol{\Omega}_{in}^n \tag{C.15}$$

this equation may also be written in term of estimated DCM as

$$\dot{\tilde{\mathbf{C}}}_b^n = \tilde{\mathbf{C}}_b^n \tilde{\boldsymbol{\Omega}}_{ib}^b - \tilde{\boldsymbol{\Omega}}_{in}^n \tilde{\mathbf{C}}_b^n \tag{C.16}$$

using equations C.14 and C.16 to replace $\dot{\tilde{\mathbf{C}}}_b^n$ and $\dot{\mathbf{C}}_b^{n^T}$ in equation C.13 results into

$$\begin{aligned}
[\delta\dot{\psi}\times] &= \left(-\tilde{\mathbf{C}}_b^n \tilde{\boldsymbol{\Omega}}_{ib}^b + \tilde{\boldsymbol{\Omega}}_{in}^n \tilde{\mathbf{C}}_b^n\right) \mathbf{C}_b^{nT} + \tilde{\mathbf{C}}_b^n \left(\boldsymbol{\Omega}_{ib}^b \mathbf{C}_b^{nT} - \mathbf{C}_b^{nT} \boldsymbol{\Omega}_{in}^n\right) \\
&= -\tilde{\mathbf{C}}_b^n \tilde{\boldsymbol{\Omega}}_{ib}^b \mathbf{C}_b^{nT} + \tilde{\boldsymbol{\Omega}}_{in}^n \tilde{\mathbf{C}}_b^n \mathbf{C}_b^{nT} + \tilde{\mathbf{C}}_b^n \boldsymbol{\Omega}_{ib}^b \mathbf{C}_b^{nT} - \tilde{\mathbf{C}}_b^n \mathbf{C}_b^{nT} \boldsymbol{\Omega}_{in}^n \\
&= -\tilde{\mathbf{C}}_b^n \left[\tilde{\boldsymbol{\Omega}}_{ib}^b - \boldsymbol{\Omega}_{ib}^b\right] \mathbf{C}_b^{nT} + \tilde{\boldsymbol{\Omega}}_{in}^n \tilde{\mathbf{C}}_b^n \mathbf{C}_b^{nT} - \tilde{\mathbf{C}}_b^n \mathbf{C}_b^{nT} \boldsymbol{\Omega}_{in}^n
\end{aligned} \tag{C.17}$$

By employing equation C.12, equation C.17 becomes

$$\begin{aligned}
[\delta\dot{\psi}\times] &= -[\mathbf{I} - [\delta\psi\times]]\mathbf{C}_b^n \left[\tilde{\boldsymbol{\Omega}}_{ib}^b - \boldsymbol{\Omega}_{ib}^b\right] \mathbf{C}_b^{nT} + \tilde{\boldsymbol{\Omega}}_{in}^n [\mathbf{I} - [\delta\psi\times]]\mathbf{C}_b^n \mathbf{C}_b^{nT} - [\mathbf{I} - [\delta\psi\times]]\mathbf{C}_b^n \mathbf{C}_b^{nT} \boldsymbol{\Omega}_{in}^n \\
&= -[\mathbf{I} - [\delta\psi\times]]\mathbf{C}_b^n \left[\tilde{\boldsymbol{\Omega}}_{ib}^b - \boldsymbol{\Omega}_{ib}^b\right] \mathbf{C}_b^{nT} + \tilde{\boldsymbol{\Omega}}_{in}^n [\mathbf{I} - [\delta\psi\times]] - [\mathbf{I} - [\delta\psi\times]]\boldsymbol{\Omega}_{in}^n
\end{aligned} \tag{C.18}$$

By applying small angle perturbation given by

$$\delta\boldsymbol{\Omega}_{in}^n = \tilde{\boldsymbol{\Omega}}_{in}^n - \boldsymbol{\Omega}_{in}^n \tag{C.19}$$

$$\delta\boldsymbol{\Omega}_{ib}^b = \tilde{\boldsymbol{\Omega}}_{ib}^b - \boldsymbol{\Omega}_{ib}^b \tag{C.20}$$

equation C.18 becomes

$$\begin{aligned}
[\delta\dot{\psi}\times] &= -[\mathbf{I} - [\delta\psi\times]]\mathbf{C}_b^n \delta\boldsymbol{\Omega}_{ib}^b \mathbf{C}_b^{nT} + \delta\boldsymbol{\Omega}_{in}^n - \tilde{\boldsymbol{\Omega}}_{in}^n [\delta\psi\times] + [\delta\psi\times]\boldsymbol{\Omega}_{in}^n \\
&\approx \delta\boldsymbol{\Omega}_{in}^n + [\delta\psi\times]\boldsymbol{\Omega}_{in}^n - \mathbf{C}_b^n \delta\boldsymbol{\Omega}_{ib}^b \mathbf{C}_b^{nT}
\end{aligned} \tag{C.21}$$

and by doing an element by element comparison, the attitude error equation becomes

$$\delta\dot{\psi} \approx \delta\omega_{in}^{n} - \mathbf{\Omega}_{in}^{n}\delta\psi - \mathbf{C}_{b}^{n}\delta\omega_{ib}^{b} \tag{C.22}$$

Velocity Equations

Starting from equation C.11 and applying perturbation of the nominal state results into

$$\begin{aligned}\delta\dot{\mathbf{v}}^{n} &= \dot{\tilde{\mathbf{v}}}^{n} - \dot{\mathbf{v}}^{n}\\ &= \tilde{\mathbf{C}}_{b}^{n}\tilde{\mathbf{f}}^{b} - \mathbf{C}_{b}^{n}\mathbf{f}^{b} - (\tilde{\omega}_{en}^{n} + 2\tilde{\omega}_{ie}^{n}) \times \tilde{\mathbf{v}}^{n} + (\omega_{en}^{n} + 2\Omega_{ie}^{n}) \times \mathbf{v}^{n} + \tilde{\mathbf{g}}_{l}^{n}(\mathbf{p}) - \mathbf{g}_{l}^{n}(\mathbf{p})\end{aligned} \tag{C.23}$$

substituting for

$$\delta\dot{\mathbf{v}}_{e}^{n} = \dot{\tilde{\mathbf{v}}}_{e}^{n} - \dot{\mathbf{v}}_{e}^{n} \tag{C.24}$$

$$\delta\mathbf{f}^{b} = \tilde{\mathbf{f}}^{b} - \mathbf{f}^{b} \tag{C.25}$$

$$\delta\mathbf{v}_{e}^{n} = \tilde{\mathbf{v}}_{e}^{n} - \mathbf{v}_{e}^{n} \tag{C.26}$$

$$\delta\omega_{ie}^{n} = \tilde{\omega}_{ie}^{n} - \omega_{ie}^{n} \tag{C.27}$$

$$\delta\omega_{en}^{n} = \tilde{\omega}_{en}^{n} - \omega_{en}^{n} \tag{C.28}$$

$$\delta\mathbf{g}_{l}^{n} = \tilde{\mathbf{g}}_{l}^{n} - \mathbf{g}_{l}^{n} \tag{C.29}$$

$$\tilde{\mathbf{C}}_{b}^{n} = [\mathbf{I} - [\delta\phi\times]]\mathbf{C}_{b}^{n} \tag{C.30}$$

gives

$$\delta\dot{\mathbf{v}}_{e}^{n} = [\mathbf{f}^{n}\times]\delta\psi + \boldsymbol{C}_{b}^{n}\delta\mathbf{f}^{b} - (2\omega_{ie}^{n} + \omega_{en}^{n}) \times \delta\mathbf{v}_{e}^{n} - (2\delta\omega_{ie}^{n} + \delta\omega_{en}^{n}) \times \mathbf{v}_{e}^{n} - \delta\mathbf{g}_{l}^{n} \tag{C.31}$$

If errors in Coriolis and gravity terms are ignored, then

$$\delta\dot{\mathbf{v}}_{e}^{n} = [\mathbf{f}^{n}\times]\delta\psi + \boldsymbol{C}_{b}^{n}\delta\mathbf{f}^{b} \tag{C.32}$$

C.1.3 The Local Gravity Component

Accelerometers provide measurements of the difference between the acceleration with respect to the inertial frame and the gravitational acceleration at the location of the navigational system. For a precise removal of the gravitational component, it is necessary to model accurately the local gravitational field. Since navigation frame mechanization is considered in this thesis, derivations will be relative to the navigation frame. However, if other frames mechanizations are considered, then the same procedure should be applied.

Given vehicle position in spherical coordinates; i.e. geodetic latitude (φ for simplicity), longitude (λ) and altitude (h), then the vehicle position in the ECEF is given by

$$\mathbf{r}_{veh}^{ecef} = \begin{matrix} (R_n + h)\cos\varphi\cos\lambda \\ (R_n + h)\cos\varphi\sin\lambda \\ (R_m + h)\sin\varphi \end{matrix} \tag{C.33}$$

Then, the position of the vehicle in the navigation frame is expressed by

$$\begin{aligned} \mathbf{r}_{veh}^{ned} &= \mathbf{C}_e^n \mathbf{r}_{veh}^{ecef} \\ &= \begin{pmatrix} -\sin\varphi\cos\lambda & -\sin\varphi\sin\lambda & \cos\varphi \\ -\sin\lambda & \cos\lambda & 0 \\ -\cos\varphi\cos\lambda & -\cos\varphi\sin\lambda & -\sin\varphi \end{pmatrix} \mathbf{r}_{veh}^{ecef} \\ &= \begin{pmatrix} -(R_n + h)\cos\varphi\sin\varphi + (R_m + h)\cos\varphi\sin\varphi \\ 0 \\ -(R_n + h)\cos^2\varphi - (R_m + h)\sin^2\varphi \end{pmatrix} \\ &= \begin{pmatrix} r_1 \\ 0 \\ r_3 \end{pmatrix} \end{aligned} \tag{C.34}$$

The centripetal acceleration in the navigation frame is calculated by

$$\Omega_{ie}^{n} \times \left(\Omega_{ie}^{n} \times \mathbf{r}_{veh}^{ned}\right) \tag{C.35}$$

where

$$\boldsymbol{\Omega}_{ie}^{n} = \mathbf{C}_{e}^{n}\boldsymbol{\Omega}_{ie}^{e} = \begin{pmatrix} -\sin\varphi\cos\lambda & -\sin\varphi\sin\lambda & \cos\varphi \\ -\sin\lambda & \cos\lambda & 0 \\ -\cos\varphi\cos\lambda & -\cos\varphi\sin\lambda & -\sin\varphi \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ \Omega_{Earth} \end{pmatrix} = \begin{pmatrix} \Omega_{Earth}\cos\varphi \\ 0 \\ -\Omega_{Earth}\sin\varphi \end{pmatrix} \tag{C.36}$$

Therefore;

$$\Omega_{ie}^{n} \times \mathbf{r}_{veh}^{ned} = \begin{pmatrix} 0 \\ r_1\Omega_{Earth}\sin\varphi + r_3\Omega_{Earth}\cos\varphi \\ 0 \end{pmatrix} \tag{C.37}$$

and the centripetal acceleration in the navigation frame becomes

$$\begin{aligned} \Omega_{ie}^{n} \times \left(\Omega_{ie}^{n} \times \mathbf{r}_{veh}^{ned}\right) &= \begin{pmatrix} -r_1\Omega_{Earth}^{2}\sin^{2}\varphi + r_3\Omega_{Earth}^{2}\sin\varphi\cos\varphi \\ 0 \\ -r_1\Omega_{Earth}^{2}\sin\varphi\cos\varphi - r_3\Omega_{Earth}^{2}\cos^{2}\varphi \end{pmatrix} \\ &= \frac{\Omega_{Earth}^{2}(R_n + h)}{2} \begin{pmatrix} \sin 2\varphi \\ 0 \\ 1 + \cos 2\varphi \end{pmatrix} \end{aligned} \tag{C.38}$$

The other component of the local gravity is the gravitational acceleration due to the Earth's gravity. The derivation of this component requires accurate modelling of Earth's gravity.

Given the geodetic latitude φ and altitude h, the local gravity component at the surface of the geoid can be written as

$$\begin{aligned} g(\varphi,h) =& 9.780327(1+5.324\times10^{-3}\sin^2\varphi-5.8\times10^{-6}\sin^2 2\varphi) \\ &-(3.0877\times10^{-6}-4.4\times10^{-9}\sin^2\varphi)h+7.2\times10^{-14}h^2 \quad m/s^2 \end{aligned} \tag{C.39}$$

C.2 The Kalman Filter

If the system model is assumed time invariant, then the steady state Kalman filter model is given by

$$\mathbf{x}(k) = \mathbf{F}(k)\mathbf{x}(k-1) + \mathbf{B}(k)\mathbf{u}(k) + \mathbf{G}(k)\mathbf{w}(k) \tag{C.40}$$

$$\mathbf{z}(k) = \mathbf{H}(k)\mathbf{x}(k) + \mathbf{v}(k) \tag{C.41}$$

$$\tag{C.42}$$

In equation C.40 we assumed that no mismodeling exists between the Kalman filter model and the true system model and so the Kalman filter model represented by matrices $\mathbf{F}(k)$, $\mathbf{B}(k)$, $\mathbf{G}(k)$, and $\mathbf{H}(k)$ represents the true system model.

Kalman filter state estimate propagation equation based on the Kalman filter model is

$$\hat{\mathbf{x}}(k|k-1) = \mathbf{F}(k)\hat{\mathbf{x}}(k-1|k-1) + \mathbf{B}(k)\mathbf{u}(k) \tag{C.43}$$

$$\hat{\mathbf{z}}(k) = \mathbf{H}(k)\hat{\mathbf{x}}(k|k-1) \tag{C.44}$$

C.2.1 Propagation of Means

$$\begin{aligned} \hat{\mathbf{x}}_{k|k-1} &= E\left[\mathbf{x}_k|\mathbf{Z}^{k-1}\right] \\ &= E\left[\mathbf{F}_k\mathbf{x}_{k-1} + \mathbf{B}_k\mathbf{u}_k + \mathbf{G}_k\mathbf{w}_k|\mathbf{Z}^{k-1}\right] \\ &= \mathbf{F}_kE\left[\mathbf{x}_{k-1}|Z^{k-1}\right] + \mathbf{B}_k\mathbf{u}_k + \mathbf{G}_kE\left[\mathbf{w}_k|\mathbf{Z}^{k-1}\right] \\ &= \mathbf{F}_k\hat{\mathbf{x}}_{k-1|k-1} + \mathbf{B}_k\mathbf{u}_k \end{aligned} \tag{C.45}$$

Finding the recursive linear estimator, we assume that the estimate $\hat{\mathbf{x}}_{k|k}$ is a linear sum of the prediction $\hat{\mathbf{x}}_{k|k-1}$ and the observation $\mathbf{z}_k$. That is

$$\hat{\mathbf{x}}_{k|k} = \mathbf{K}_k^{(1)} \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k^{(2)} \mathbf{z}_k \tag{C.46}$$

where $\mathbf{K}_k^{(1)}$ and $\mathbf{K}_k^{(2)}$ are both time varying gain matrices chosen to minimize the conditional mean squared estimation error given by

$$\begin{aligned} L_k &= E\left[\delta \mathbf{x}_{k|k}^T \delta \mathbf{x}_{k|k} | \mathbf{Z}^k\right] \\ &= trace\left[E\left[\delta \mathbf{x}_{k|k}^T \delta \mathbf{x}_{k|k} | \mathbf{Z}^k\right]\right] \\ &= trace[\mathbf{P}_{k|k}] \end{aligned} \tag{C.47}$$

where

$$\delta \mathbf{x}_{k|k} = \hat{\mathbf{x}}_{k|k} - \mathbf{x}_k \tag{C.48}$$

If the estimate $\hat{\mathbf{x}}_{k|k}$ is assumed unbiased, i.e. the conditional expected error is zeros, then by equations C.46 and C.48

$$\begin{aligned} \delta x_{k|k} &= \hat{\mathbf{x}}_{k|k} - \mathbf{x}_k \\ &= \mathbf{K}_k^{(1)} \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k^{(2)} \mathbf{z}_k - \mathbf{x}_k \\ &= \mathbf{K}_k^{(1)} \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k^{(2)} \mathbf{H}_k \mathbf{x}_k + \mathbf{K}_k^{(2)} \mathbf{v}_k - \mathbf{x}_k \\ &= \mathbf{K}_k^{(1)} [\mathbf{x}_k + \delta \mathbf{x}_{k|k-1}] + \mathbf{K}_k^{(2)} \mathbf{H}_k \mathbf{x}_k + \mathbf{K}_k^{(2)} \mathbf{v}_k - \mathbf{x}_k \\ &= [\mathbf{K}_k^{(1)} + \mathbf{K}_k^{(2)} \mathbf{H}_k - 1] \mathbf{x}_k + \mathbf{K}_k^{(1)} \delta \mathbf{x}_{k|k-1} + \mathbf{K}_k^{(2)} \mathbf{v}_k \end{aligned} \tag{C.49}$$

Then

$$E\left[\delta \mathbf{x}_{k|k-1} | \mathbf{Z}^k\right] = \left[\mathbf{K}_k^{(1)} + \mathbf{K}_k^{(2)} \mathbf{H}_k - 1\right] E\left[\mathbf{x}_k | \mathbf{Z}^k\right] = 0 \tag{C.50}$$

so that

$$\mathbf{K}_k^{(1)} = 1 - \mathbf{K}_k^{(2)} \mathbf{H}_k \tag{C.51}$$

Let $\mathbf{W}_k = \mathbf{K}_k^{(2)}$, then by equation C.46

$$\begin{aligned}\hat{\mathbf{x}}_{k|k} &= [1 - \mathbf{W}_k\mathbf{H}_k]\hat{\mathbf{x}}_{k|k-1} + \mathbf{W}_k\mathbf{z}_k \\ &= \hat{\mathbf{x}}_{k|k-1} + \mathbf{W}_k[\mathbf{z}_k - \mathbf{H}_k\hat{\mathbf{x}}_{k|k-1}]\end{aligned} \tag{C.52}$$

C.2.2 Propagation of Variances

$$\begin{aligned}\mathbf{P}_{k|k-1} &= E\left[[\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1}][\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1}]^T|\mathbf{Z}^{k-1}\right] \\ &= E\left[[\mathbf{F}_k(\mathbf{x}_k - \hat{\mathbf{x}}_{k-1|k-1}) + \mathbf{G}_k\mathbf{w}_k][\mathbf{F}_k(\mathbf{x}_k - \hat{\mathbf{x}}_{k-1|k-1}) + \mathbf{G}_k\mathbf{w}_k]^T\right] \\ &=\mathbf{F}_kE[(\mathbf{x}_k - \hat{\mathbf{x}}_{k-1|k-1})(\mathbf{x}_k - \hat{\mathbf{x}}_{k-1|k-1})^T]\mathbf{F}_k^T + \mathbf{G}_kE[\mathbf{w}_k\mathbf{w}_k^T]\mathbf{G}_k^T \\ &\quad + \mathbf{F}_kE[(\mathbf{x}_k - \hat{\mathbf{x}}_{k-1|k-1})\mathbf{w}_k^T]\mathbf{G}_k^T + \mathbf{G}_kE[\mathbf{w}_k(\mathbf{x}_k - \hat{\mathbf{x}}_{k-1|k-1})^T]\mathbf{F}_k^T \\ &= \mathbf{F}_kE[(\mathbf{x}_k - \hat{\mathbf{x}}_{k-1|k-1})(\mathbf{x}_k - \hat{\mathbf{x}}_{k-1|k-1})^T]\mathbf{F}_k^T + \mathbf{G}_kE[\mathbf{w}_k\mathbf{w}_k^T]\mathbf{G}_k^T \\ &= \mathbf{F}_k\mathbf{P}_{k-1|k-1}\mathbf{F}_k^T + \mathbf{G}_k\mathbf{Q}\mathbf{G}_k^T\end{aligned} \tag{C.53}$$

$$\begin{aligned}\mathbf{P}_{x,z} &= E\left[(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1})(\mathbf{z}_k - \hat{\mathbf{z}}_k)^T\right] \\ &= E\left[(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1})\left(\mathbf{H}_k(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1}) + \mathbf{v}_k\right)^T\right] \\ &= E\left[(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1})\right]\mathbf{H}_k^T + E\left[(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1})\mathbf{v}_k^T\right] \\ &= \mathbf{P}_{k|k-1}\mathbf{H}_k^T\end{aligned} \tag{C.54}$$

$$\begin{aligned}\mathbf{S}_k &= E\left[(\mathbf{z}_k - \hat{\mathbf{z}}_k)(\mathbf{z}_k - \hat{\mathbf{z}}_k)^T\right] \\ &= E\left[(\mathbf{H}_k(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1}) + \mathbf{v}_k)(\mathbf{H}_k(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1}) + \mathbf{v}_k)^T\right] \\ &=\mathbf{H}_kE\left[(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1})(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1})^T\right]\mathbf{H}_k^T + E[\mathbf{v}_k\mathbf{v}_k^T] \\ &\quad + \mathbf{H}_kE\left[(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1})\mathbf{v}_k^T\right] + \mathbf{H}_k^T\left[(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1})^T\mathbf{v}_k\right] \\ &= \mathbf{H}_kE\left[(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1})(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1})^T\right]\mathbf{H}_k^T + E[\mathbf{v}_k\mathbf{v}_k^T] \\ &= \mathbf{H}_k\mathbf{P}_{k|k-1}\mathbf{H}_k^T + \mathbf{R}_k\end{aligned} \tag{C.55}$$

The uncertainty in estimate $\hat{\mathbf{x}}_{k|k}$ is described by the covariance matrix $\mathbf{P}_{k|k}$.

This covariance is derived as follows

$$\mathbf{P}_{k|k} = E\left[\delta\mathbf{x}_{k|k}^T \delta\mathbf{x}_{k|k} | \mathbf{Z}^k\right] \tag{C.56}$$

where, from C.52

$$\begin{aligned} \delta\mathbf{x}_{k|k} &= \hat{\mathbf{x}}_{k|k} - \mathbf{x}_k \\ &= (1 - \mathbf{W}_k\mathbf{H}_k)\hat{\mathbf{x}}_{k|k-1} + \mathbf{W}_k\mathbf{z}_k - \mathbf{x}_k \\ &= (1 - \mathbf{W}_k\mathbf{H}_k)\hat{\mathbf{x}}_{k|k-1} + \mathbf{W}_k[\mathbf{H}_k\mathbf{x}_k + \mathbf{v}_k] - \mathbf{x}_k \\ &= (1 - \mathbf{W}_k\mathbf{H}_k)\hat{\mathbf{x}}_{k|k-1} - (1 - \mathbf{W}_k\mathbf{H}_k)\mathbf{x}_k + \mathbf{W}_k\mathbf{v}_k \\ &= (1 - \mathbf{W}_k\mathbf{H}_k)\delta\mathbf{x}_{k|k-1} + \mathbf{W}_k\mathbf{v}_k \end{aligned} \tag{C.57}$$

so that

$$\begin{aligned} \mathbf{P}_{k|k} &= E\left[\delta\mathbf{x}_{k|k}\delta\mathbf{x}_{k|k}^T | \mathbf{Z}^k\right] \\ =&(1 - \mathbf{W}_k\mathbf{H}_k)E\left[\delta\mathbf{x}_{k|k-1}\delta\mathbf{x}_{k|k-1}^T | \mathbf{Z}^k\right](1 - \mathbf{W}_k\mathbf{H}_k)^T \\ &+ \mathbf{W}_k E\left[\mathbf{v}_k\mathbf{v}_k^T\right]\mathbf{W}_k^T \\ &+ 2(1 - \mathbf{W}_k\mathbf{H}_k)E\left[\delta\mathbf{x}_{k|k-1}\mathbf{v}_k^T\right]\mathbf{W}_k^T \\ &= (1 - \mathbf{W}_k\mathbf{H}_k)\mathbf{P}_{k|k-1}(1 - \mathbf{W}_k\mathbf{H}_k)^T + \mathbf{W}_k\mathbf{R}_k\mathbf{W}_k^T \end{aligned} \tag{C.58}$$

C.2.3 The Kalman Gain

The gain matrix $\mathbf{W}_k$ is assumed to minimize the conditional mean squared estimation error given by C.47, that is

$$\frac{\partial L_k}{\mathbf{W}_k} = 0 \tag{C.59}$$

from the fact that for any matrix $\mathbf{A}$ and a symmetric matrix $\mathbf{B}$;

$$\frac{\partial}{\partial\mathbf{A}}\left(trace(\mathbf{A}\mathbf{B}\mathbf{A}^T)\right) = 2\mathbf{A}\mathbf{B} \tag{C.60}$$

equation C.59 becomes

$$\frac{\partial L_k}{\mathbf{W}_k} = -2(1 - \mathbf{W}_k\mathbf{H}_k)\mathbf{P}_{k|k-1}\mathbf{H}_k^T + 2\mathbf{W}_k\mathbf{R}_k = 0 \tag{C.61}$$

and then

$$\mathbf{P}_{k|k-1}\mathbf{H}_k^T + \mathbf{W}_k\mathbf{H}_k\mathbf{P}_{k|k-1}\mathbf{H}_k^T + \mathbf{W}_k\mathbf{R}_k = 0 \tag{C.62}$$

Rearranging provides

$$\mathbf{W}_k = \frac{\mathbf{P}_{k|k-1}\mathbf{H}_k^T}{\mathbf{H}_k\mathbf{P}_{k|k-1}\mathbf{H}_k^T + \mathbf{R}_k} \tag{C.63}$$

or, in more detailed expression

$$\mathbf{W}_k = \frac{(\mathbf{F}_k\mathbf{P}_{k-1|k-1}\mathbf{F}_k^T + \mathbf{G}_k\mathbf{Q}\mathbf{G}_k^T)\mathbf{H}_k^T}{\mathbf{H}_k(\mathbf{F}_k\mathbf{P}_{k-1|k-1}\mathbf{F}_k^T + \mathbf{G}_k\mathbf{Q}\mathbf{G}_k^T)\mathbf{H}_k^T + \mathbf{R}_k} \tag{C.64}$$

Appendix D

GPS Time and Reference Systems

For an accurate processing of GPS row measurements, the GPS satellite time at the time of transmission must be accurately determined. This chapter discusses all aspects related to satellite time determination. For more detailed analysis on satellite time and reference systems refer to (Kaplan, E. D., & Hegarty,C. J., 2006), (Tsui, J. B.Y., 2000), (Kelly, A, 1994).

D.1 Satellite Navigation Time Scales

Time is an issue that requires careful attention in the description of astronomical, physical, and geodic phenomena. The Solar day is determined from subsequent meridian transits of the Sun, which is measured in days of 86400 seconds duration. However, due to orbital motion of the Earth around the Sun, the Sun's right ascension changes by approximately one degree per day which contributes into 4 minutes reduction in day time from the Solar day resulting into what is referred to as the Siderial day (approximately $23^{hr}56^{min}4^{sec}.1$). It is equal to the time between successive meridian passage of the vernal equinox.

The GPS system time is referenced to the Universal Time Coordinate (UTC). The UTC is a time scale that is composed of two time scales, namely, the International Atomic Time (TAI) based on atomic second and the Universal Time 1 (UT1) based on the Earth's rotation with respect to the Sun. An epoch in GPS system time is measured by the number of seconds elapsed since SaturdaySunday midnight and the week number. The epoch time is transmitted in the ephemeris data. GPS weeks are numbered sequentially from week 0, which began on 0 hours (midnight) January 6,

1980. Both the week number, 10-bits, and the time of the week (TOW), 19-bits, form the Z count (Tsui, J. B.Y., 2000).

D.2 The GPS System Time at the Time of Transmission

For accurate calculation of satellite position, the GPS transmission time is essential. However, at start, the actual transmission time is unavailable due to system errors and delays. These errors include satellite clock bias, atmospheric delays and the effect of the Earth's rotation, referred to as 'Sagnac' effect. GPS system timing and delays are represented in figure D.1. The sagnac effect is not represented in this figure, but it will be introduced during the discussion.

where

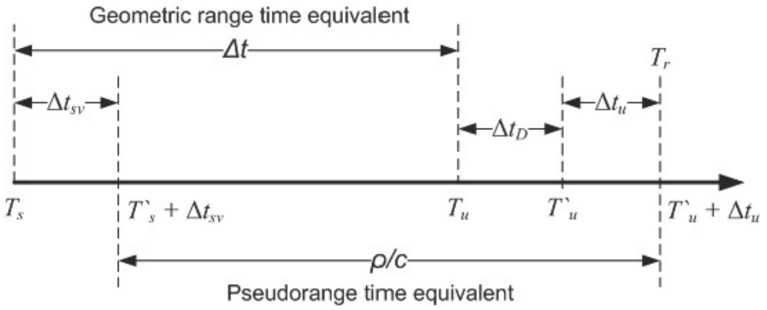


Figure D.1: GPS system timing

T_s is the GPS system time at which signal left the satellite.

T_u is the GPS system time at which signal would have reached the user in the absence of errors.

Δt_D is the satellite link propagation error.

$\acute{T}_u$ is the GPS system time at which the signal reached the receiver with Δt_D

Δt_{sv} is the satellite clock bias.

Δt_u is the receiver clock bias.

ρ is the satellite's measured pseudorange.

c is the speed of light.

r is the geometric range.

Therefore, the geometric range and the measured pesudorange are related as follows

$$\begin{aligned} \frac{\rho}{c} &= (\acute{T}_u + \Delta t_u) - (T_s + \Delta t_{sv}) \\ &= (\acute{T}_u - T_s) + \Delta t_u + \Delta t_D - \Delta t_{sv} \\ &= \frac{r}{c} + \Delta t_u + \Delta t_D - \Delta t_{sv} \end{aligned} \tag{D.1}$$

then

$$r = \rho - c[\Delta t_u + \Delta t_D - \Delta t_{sv}] \tag{D.2}$$

where

$$\Delta t_D = (\Delta t_{Iono} + \Delta t_{Tropo}) + \Delta t_{MP} + \Delta t_{HW} + \Delta t_{Noise} \tag{D.3}$$

is the total satellite link propagation error and

Δt_{Iono}	is the ionospherical delay component.
Δt_{Tropo}	is the tropospherical delay component.
Δt_{MP}	is the delay component due to multipath.
Δt_{HW}	is the hardware delay component.
Δt_{Noise}	is the receiver noise component.

To calculate the positions of GPS satellites, the *GPS time at the time of transmission* is required. However, this time is difficult to be calculated since the satellite clock bias and the receiver clock bias are unknown. The satellite clock bias may be calculated by

$$\Delta t_{SV} = a_{f_0} + a_{f_1}(t - t_{oc}) + a_{f_2}(t - t_{oc})^2 + \Delta t_r - t_{GD} \tag{D.4}$$

where

a_{f_0}	is the satellite clock bias in seconds.
a_{f_1}	is the clock drift in seconds per seconds.
a_{f_2}	is frequency drift (aging) in seconds per seconds squared.
Δt_r	is the relativistic error correction term.
t_{GD}	is the group delay differential obtained from the ephemeris data.

t_{oc} is the epoch time obtained from the ephemeris data.

and

$$\Delta t_r = F e_s \sqrt{a_s} \sin E \tag{D.5}$$

$$F = \frac{-2\sqrt{\mu}}{c^2} \tag{D.6}$$

where

μ is the Earth's universal gravitational parameter.

c is the speed of light.

e_s is the satellite orbit eccentricity.

a_s is the satellite orbit semimajor axis.

E is the eccentric anomaly obtained by solving Kepler's equation.

On the other hand, to solve for E the GPS system time at the time of transmission is also required. To solve this conflict, a course value of this time can be obtained as the *the course GPS system time at time of transmission corrected for transit time* t_c, where

$$t_c = t_u - \frac{\rho_i}{c} \tag{D.7}$$

This time still needs to be corrected for other factors, specified by figure D.1. Therefore, once Kepler's equation is solved for E and the satellite clock bias may be computed and a better estimate of t_c can be obtained by

$$t_c = t_u - \frac{\rho_i}{c} - \Delta t_{SV} \tag{D.8}$$

This time is still a course time, since E was computed using the previous course time. This process is iterated, until no change (based on predetermined threshold value) is detected on the value of the satellite clock bias or on t_c, and a better estimate of the GPS system time at the time of transmission is obtained. Unfortunately, this is still a course time, since the receiver clock bias is unknown. The receiver clock bias wont become available until least squares (or Kalman filter) is applied to solve for position and receiver clock bias errors. Once the receiver clock bias is available the course GPS time at the time of transmission is updated as follows

$$t_c = t_u - \frac{\rho_i}{c} - \Delta t_{SV} - \Delta t_u \tag{D.9}$$

This is also still a course time, since satellite positions were calculated based on the previous course time. The updated course time is then used again to calculate the satellites positions and the new satellites positions are used to get a better estimate of the receiver clock bias. This process is also iterated until no change on the receiver clock bias (based on predetermined threshold value) is detected.

Appendix E

Loosely and Tightly Coupled Configurations

In this chapter, loosely and tightly linear Kalman filter state space formulations are represents. In short, table E.1 summarizes Kalman filter model matrices for both the loosely coupled and tightly coupled linear systems.

where

$$\delta\mathbf{x}(t) = \mathbf{F}(t)\delta\mathbf{x}(t-1) + \mathbf{G}(t)\mathbf{w}(t) \tag{E.1}$$

$$\mathbf{F}_{ins} = \begin{pmatrix} \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & -2\omega_{ie}^{e}\times & (\mathbf{C}_b^e\mathbf{f}^b)\times \\ \mathbf{0} & \mathbf{0} & -\omega_{ie}^{e}\times \end{pmatrix} \tag{E.2}$$

$$\mathbf{F}_{coupled} = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{C}_b^e & \mathbf{0} \\ \mathbf{0} & -\mathbf{C}_b^e \end{pmatrix} \tag{E.3}$$

$$\mathbf{F}_{biases} = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix} \tag{E.4}$$

$$\mathbf{F}_{clock} = \begin{pmatrix} \mathbf{0} \end{pmatrix} \tag{E.5}$$

$$\mathbf{w}(t) = \begin{pmatrix} w_{gyro}(t) \\ w_{accl}(t) \end{pmatrix} \tag{E.6}$$

Table E.1: Different Kalman filter implementations

	Loosely-Coupled-Linear	Tightly-Coupled-Linear
State Vector $\delta\hat{\mathbf{x}}$	$\begin{pmatrix} \delta\mathbf{P}_e & \delta\mathbf{V}_e & \delta\boldsymbol{\Psi} & \delta\mathbf{f}^b & \delta\omega_{ib}^b \end{pmatrix}^T$ 15 × 15	$\begin{pmatrix} \delta\mathbf{P}_e & \delta\mathbf{V}_e & \delta\boldsymbol{\Psi} & \delta\mathbf{c}_b & \delta\mathbf{f}^b & \delta\omega_{ib}^b \end{pmatrix}^T$ 16 × 16
Measurement Vector $\delta\hat{\mathbf{x}}$	$\begin{pmatrix} \mathbf{r}_{gps}^m - \mathbf{r}_{ins}^m \\ \mathbf{v}_{gps}^m - \mathbf{v}_{ins}^m \end{pmatrix}$ 6 × 1	$\left(\rho_{meas}^{ecef} - \rho_{pred}^{ecef}\right)$ m ×1
State Transition $\mathbf{F}$	$\begin{pmatrix} \mathbf{F}_{ins} & \mathbf{F}_{coupled} \\ \mathbf{0} & \mathbf{F}_{biases} \end{pmatrix}$ 15 × 15	$\begin{pmatrix} \mathbf{F}_{ins} & \mathbf{0} & \mathbf{F}_{coupled} \\ \mathbf{0} & \mathbf{F}_{clock} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{F}_{biases} \end{pmatrix}$ 16 × 16
Observation Transition $\mathbf{H}$	$\begin{pmatrix} \mathbf{I}_{6\times6} & \mathbf{0} \end{pmatrix}$ 6 × 15	$\begin{pmatrix} \frac{\partial r_1}{\partial x} & \frac{\partial r_1}{\partial y} & \frac{\partial r_1}{\partial z} & \frac{\partial r_1}{\partial C_b} \\ \frac{\partial\ r_2}{\partial x} & \frac{\partial r_2}{\partial y} & \frac{\partial r_2}{\partial z} & \frac{\partial r_2}{\partial C_b} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial r_n}{\partial x} & \frac{\partial r_n}{\partial y} & \frac{\partial r_n}{\partial z} & \frac{\partial r_n}{\partial C_b} \end{pmatrix}$ $m \times 4$
Initial State Covariance $\mathbf{P}_0$	$\begin{pmatrix} \sigma_{\mathbf{r}}^2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \sigma_{\mathbf{v}}^2 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \sigma_{\boldsymbol{\Psi}}^2 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \sigma_{\Delta\mathbf{f}^b}^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \sigma_{\Delta\omega^b}^2 \end{pmatrix}$ 15 × 15	$\begin{pmatrix} \sigma_{\mathbf{r}}^2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \sigma_{\mathbf{v}}^2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \sigma_{\boldsymbol{\Psi}}^2 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \sigma_{C_b}^2 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \sigma_{\Delta\mathbf{f}^b}^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \sigma_{\Delta\omega^b}^2 \end{pmatrix}$ 16 × 16
Process Noise Covariance $\mathbf{Q}$	$\begin{pmatrix} \sigma_{\mathbf{f}^b}^2 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \sigma_{\omega^b}^2 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \sigma_{\Delta\mathbf{f}^b}^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \sigma_{\Delta\omega^b}^2 \end{pmatrix}$ 15 × 15	$\begin{pmatrix} \sigma_{\mathbf{f}^b}^2 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \sigma_{\omega^b}^2 & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \sigma_{C_b}^2 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \sigma_{\Delta\mathbf{f}^b}^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \sigma_{\Delta\omega^b}^2 \end{pmatrix}$ 16 × 16
Sensors Noise Covariance $\mathbf{R}$	$\begin{pmatrix} \sigma_{\mathbf{r_{gps}}}^2 & \mathbf{0} \\ \mathbf{0} & \sigma_{\mathbf{v}_{gps}}^2 \end{pmatrix}$ 6 × 6	$\begin{pmatrix} \sigma_{\rho_1}^2 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \sigma_{\rho_1}^2 & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \sigma_{\rho_m}^2 \end{pmatrix}$ m × m
Noise Transition $\mathbf{G}$	$\begin{pmatrix} \mathbf{G}_{ins} & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_{biases} \end{pmatrix}$ 15 × 15	$\begin{pmatrix} \mathbf{G}_{ins} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_{clock} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{G}_{biases} \end{pmatrix}$ 16 × 16

$$\mathbf{G}_{ins} = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{C}_b^e & \mathbf{0} \\ \mathbf{0} & -\mathbf{C}_b^e \end{pmatrix} \tag{E.7}$$

$$\mathbf{G}_{biases} = \mathbf{I}_{6\times 6} \tag{E.8}$$

$$\mathbf{G}_{clock} = \mathbf{I} \tag{E.9}$$

$$\delta\mathbf{z}(t) = \mathbf{H}(t)\delta\mathbf{x}(t) + \mathbf{W}(t)\mathbf{v}(t) \tag{E.10}$$

$$\rho_{i,pred}^{ecef} = \sqrt{(X_i - X_u)^2 + (Y_i - Y_u)^2 + (Z_i - Z_u)^2} + C_b \tag{E.11}$$

$$\rho_{i,meas}^{ecef} = \rho_{i,row} - c\Delta t_D + c\Delta t_{SV} \tag{E.12}$$

$$\delta z_i = \rho_{i,meas}^{ecef} - \rho_{i,pred}^{ecef} \tag{E.13}$$

$$\mathbf{i}(k) = \delta\mathbf{z}(k) - \mathbf{H}\delta\mathbf{x}(k) \tag{E.14}$$

Appendix F

Chi-square Distribution

DoF	$F(z)$							
	0.005	0.01	0.025	0.05	0.95	0.975	0.99	0.995
1	0.00	0.00	0.00	0.00	3.84	5.02	6.63	7.88
2	0.01	0.02	0.05	0.10	5.99	7.38	9.21	10.60
3	0.07	0.11	0.22	0.35	7.81	9.35	11.34	12.84
4	0.21	0.30	0.48	0.71	9.49	11.14	13.28	14.86
5	0.41	0.55	0.83	1.15	11.07	12.83	15.09	16.75
6	0.68	0.87	1.24	1.64	12.59	14.45	16.81	18.55
7	0.99	1.24	1.69	2.17	14.07	16.01	18.48	20.28
8	1.34	1.65	2.18	2.73	15.51	17.53	20.09	21.96
9	1.73	2.09	2.70	3.33	16.92	19.02	21.67	23.59
10	2.16	2.56	3.25	3.94	18.31	20.48	23.21	25.19
11	2.60	3.05	3.82	4.57	19.68	21.92	24.73	26.76
12	3.07	3.57	4.40	5.23	21.03	23.34	26.22	28.30
13	3.57	4.11	5.01	5.89	22.36	24.74	27.69	29.82
14	4.07	4.66	5.63	6.57	23.68	26.12	29.14	31.32
15	4.60	5.23	6.26	7.26	25.00	27.49	30.58	32.80
16	5.14	5.81	6.91	7.96	26.30	28.85	32.00	34.27
17	5.70	6.41	7.56	8.67	27.59	30.19	33.41	35.72
18	6.26	7.01	8.23	9.39	28.87	31.53	34.81	37.16
19	6.84	7.63	8.91	10.12	30.14	32.85	36.19	38.58
20	7.43	8.26	9.59	10.85	31.41	34.17	37.57	40.00

DoF	$F(z)$							
	0.005	0.01	0.025	0.05	0.95	0.975	0.99	0.995
21	8.00	8.90	10.3	11.6	32.7	35.5	38.9	41.4
22	8.60	9.50	11.0	12.3	33.9	36.8	40.3	42.8
23	9.30	10.2	11.7	13.1	35.2	38.1	41.6	44.2
24	9.90	10.9	12.4	13.8	36.4	39.4	43.0	45.6
25	10.5	11.5	13.1	14.6	37.7	40.6	44.3	46.9
26	11.2	12.2	13.8	15.4	38.9	41.9	45.6	48.3
27	11.8	12.9	14.6	16.2	40.1	43.2	47.0	49.6
28	12.5	13.6	15.3	16.9	41.3	44.5	48.3	51.0
29	13.1	14.3	16.0	17.7	42.6	45.7	49.6	52.3
30	13.8	15.0	16.8	18.5	43.8	47.0	50.9	53.7
40	20.7	22.2	24.4	26.5	55.8	59.3	63.7	66.8
50	28.0	29.7	32.4	34.8	67.5	71.4	76.2	79.5
60	35.5	37.5	40.5	43.2	79.1	83.3	88.4	92.0
70	43.3	45.4	48.8	51.7	90.5	95.0	100.4	104.2
80	51.2	53.5	57.2	60.4	101.9	106.6	112.3	116.3
90	59.2	61.8	65.6	69.1	113.1	118.1	124.1	128.3
100	67.3	70.1	74.2	77.9	124.3	129.6	135.8	140.2

For DoF > 100

$$\chi^2(m,k) = \tfrac{1}{2}(\sqrt{2m-1}+k)^2$$

where

γ	0.005	0.01	0.025	0.05	0.95	0.975	0.99	0.995
k	-2.58	-2.33	-1.96	-1.64	1.64	1.96	2.33	2.58

References

Abdel-Hafez, M. (2003). *High Integrity GPS/INS Filter for Precise Relatice Navigation.* PhD thesis, School of Engineering, mechanical Engineering Department.

Abdel-Hafez, M., F. (2006). On the Development of an Inertial Navigation Error-Budget System.

Ahmed, M. S., & Cuk, D. V. (2005). Comparison of Different Computtion Methods for Strapdown Inertial Navigation Systems. *Scientific-Technical Review*, *LV*(2), 22–29.

Bar-Shalom, Y., Li, X. R., & Kirubarajan, T. (2001). *Estimation with Applications to Tracking and Navigation : Theory Algorithms and Software.* John Wiley & Sons, Ltd.

Bertsekas, D. P., & Tsitsiklis, J. N. (2000). *Introduction to Probability.* Lecture Notes, M.I.T.

Bhatti, U. I., & Ochieng, W. Y. (2007). Failure Modes and Models for Integrated GPS/INS Systems. *The Journal of Navigation*, *VOL. 60*, pp. 327–348.

Cavallaro, J. R. & Walker, I., D. (1994). A Survey of NASA and Military Standards on Fault Tolerance and Reliability Applied to Robotics. *in Proceedings of AIAA-NASA Conference on Intelligent Robots in Field, Factory, Service, and Space*, pp. 282–286.

Chapra, S., C., & Canale, R., P. *"Numerical Methods for Engineers"* (5th ed.).

Chatfield, A., B. (1997). *Fundamentals of High Accuracy Inertial Navigation.* Progress in Astronautics and Aeronautics Series, V-174. AIAA.

Chiang, K. W., Noureldin, A., & El-Sheimy, N. (2004). A New Weighting Updating Method for INS/GPS Integration Architectures Based on Neural Networks. *Measurement Science and Technology*, *15*, pages 2053–2061.

Chow, E., Y. & Willsky, A. S. (1984). Analytical Redundancy and the Design of Robust Failure Detection System. *IEEE Transactions on Automatic Control*, *VOL. AC-29*(No. 7), pp. 603–614.

Crassidis, J., L. (2005). Sigma-Point Kalman Filtering for Integrated GPS and Inertial Navigation. *in AIAA Guidance, Navigation and Control Conference.*

Da, R. & Lin, C., F. (1995). Sensor Failure Detection with a Bank of Kalman Filters. *in Proceedings of the 1995 American Control Conference*, pp. 1122–1126.

Durrant-Whyte, H. F. (2001). *Introduction to Estimation and the Kalman Filter.* Australian Centre for Field Robotics, The University if Sydney, NSW 2006, Australia.

Farrell, J. A., & Barth, M. (1998). *The Global Positioning System and Inertial Navigation.*

George, M. G. (2007). All-Source Navigation for an Unmanned Aircract. Masters Thesis, School of Engineering, Aerospace, Mechanical and Mechatronic Department, Australian Centre for Field Robotics, The University of Sydney.

George, M. G., & Sukkarieh, S. (2005). Tightly Coupled INS/GPS with Bias Estimation for UAV Applications. *In Proceedings of the Australasian Conference on Robotics & Automation (ACRA) 2005.*

George, M. G., & Sukkarieh, S. (2007). Camera Aided Inertial Navigation in Poor GPS Environment. *In Proceedings 2007 IEEE Aerospace Conference.*

Giroux, R., Gourdeau, R., & Landry, R., J. (2004a). On a Generalized Non-linear Error Model for Low-Cost INS Integration. Part I: Implementation. *in Proceedings IEEE Transactions on Robotics*, *vol. 7*(No. 4), pages 487–493.

Giroux, R., Gourdeau, R., & Landry, R., J. (2004b). On a Generalized Non-linear Error Model for Low-Cost INS Integration. Part I: Theoretical Development. *in Proceedings IEEE Transactions on Robotics*, *vol. 7*(No. 4), pages 487–493.

Goel, P., Dedeoglu, G., Roumeliotis, S. I. & Sukhatime, G. S. (2000). Fault Detection and Identification in a Mobile Robot using Multiple-Model Estimation and Neural Network. *in Proceedings of IEEE Internation Conference on Robotics and Automation*, pp. 2302–2307.

GPS Standards (1995). *Global Positioning System Standard Positioning Service Signal Specification.*

Grewal, M. S., & Andrews, A. P. (2001). *Kalman Filtering, Theory and Practice Using MATLAB* (Second ed.). John Wiley & Sons, Inc.

Gustafsson, F. (2000). *Adaptive Filtering and Change Detection.* John Wiley & Sons, Ltd.

Halliday, D., Resnick, R., & Walker, J. (1997). *Fundamentals of Physics.* John Wiley & Sons, Ltd.

Hanlon, P. D., & Maybeck, P. S. (2000). Characterization of Kalman Filter Residuals in the Presence of Mismodeling. *IEEE Transactions on Aerospace and Electronic Systems*, *VOL.36*(NO.1), pp. 1208–1217.

Hashimoto, M., Kawashima, H., Nakagami, T. & Oba, F. (2001). Sensor Fault Detection and Identification in Dead-Reckoning System of Mobile Robot: Interactive Multiple Model Approach. *in Proceedings of the 2001 IEEE/RSJ International Conference on Intellignet Robots and Systems*, pp. 1321–1326.

Hashimoto, M., Kawashima, H., Nakagami, T. & Oba, F. (2003). A Multi-Model Based Fault Detection and Diagnosis of Internal Sensor for Mobile Robot. *in Proceedings of the 2003 IEEE/RSJ International Conference on Intellignet Robots and Systems*, pp. 3787–3792.

Haykin, S. (Ed.). (2001). *Kalman Filtering and Neural Networks.* John Wiley & Sons, Ltd.

Henderson, P. E. (2001). Development and Testing of a Multiple Filter Approach for Precise DGPS Positioning and Carrier-Phase Ambuguity Resolution. Master's thesis, Faculty of the Graduate, School of Engineering and Management of the Air Force Institute of Technology, Air University.

Hong, S., Lee, M., H., Chun, H., H., Kwon, S., H. & Speyer, J., L. (2006). Experimental Study on the Estimation of Lever Arm in GPS/INS. *in Proceedings of IEEE Transactions on Vehicular Technology*, *vol. 55*(No. 2), pages 431–448.

Julier, S., & Uhlmann, J., K. (1996). A General Method for Approximating Nonlinear Transformations of Probability Distributions.

Julier, S., & Uhlmann, J., K. (1997). A New Extension of the Kalman Filter to Nonlinear Systems.

Julier, S., Uhlmann, J. & Durrent-Whyte, H., F. (2000). A New Method for the Nonlinear Transformation of Means and Covariances in Filters and Estimators. *IEEE Transactions on Automatic Control, Technical Notes and Correspondence*, *vol.45*(No.3), p. 477–482.

Julier, S., Uhlmann, J. & Durrent-Whyte, H., F. (2002). Comment on "A New Method for the Nonlinear Transformation of Means and Covariances in Filters and Estimators". *IEEE Transactions on Automatic Control*, *vol.47*(No.8), p. 1406–1409.

Jwo, D. J., & Chang, S. C. (2007). Application of Optimization for GPS Navigation Kalman Filter Adaptation. *17th IFAC World Congress.*

Kaplan, E. D., & Hegarty,C. J. (2006). *Understanding GPS, Principles and Applications.* Artech House.

Kayton, M., & Fried, W. R. (1997). *Avionics Navigation Systems.*

Kelly, A (1994). Modern Inertial and Satellite Navigation Systems. *Technical Report CMU-RI-TR-94-15.*

Kim, J. (2004). *Autonomous Navigation for Airborn Applications.* PhD thesis, Aerospace, Mechanical and Mechatronic Engineering, Australian Centre for Field Robotics, The University of Sydney.

King, A., D. (1998). Inertial Navigation Forty Years of Evolution. *General Electric Companey Reviews*, *vol. 13*(No. 3), pp. 140–149.

Kong, X. (2000). *Inertial Navigation System Algorithms for Low cost IMU.* PhD thesis, Aerospace, Mechanical and Mechatronic Engineering, Australian Centre for Field Robotics, The University of Sydney.

Kreyszig, E. (1993). *Advanced Engineering Mathematics* (Seventh ed.). John Wiley & Sons, Inc.

Lefebvre, T., Bruynincks, H., & De Schutter, J. (2001). "Kalman Filters for Nonlinear Systems: A Comparison of Performance". *IEEE Transactions on Automatic Control.*

Loebis, D., Chudley, J., & Sutton, R. (2003). A Fuzzy Kalman Filter Optimized Using A Genetic Algorithm for Accurate NAvigation of an Autonomous Underwater Vehicle. In *Proceeding of the 6th IFAC Conference on Manoeuvring and Control of Marine Craft*, (pp. pages 19–24)., Girona, Spain.

Loebis, D., Sutton, R., Chudley, J., & Naeem, W. (2004). Adaptive Tuning of a Kalman Filter via Fuzzy Logic for an Intelligent AUV Navigation System. *Guidance and Control of Underwater Vehicle*, *12*(12), pages 1531–1539.

Maybeck, P. S. (1979). *Stochastic models, estimation, and control, Volume 1.* Academic Press.

Maybeck, P. S. & Hanlon, P. D. (1995). Performance Enhancement of a Multiple Model Adaptive Estimator. *IEEE Transactions on Aerospace and Electronic Systems*, *VOL.31*(NO.4), pp. 1240–1254.

Maybeck, P. S. & Vasquez J. R. (1999). Density Lgorithm Based Moving-Bank MMAE. *Proceedings of the* 38_{th} *Conference on Decision & Control*, *VOL.* (NO.), pp. 4117–4122.

Maybeck, P. S. & Vasquez J. R. (2004). Enhanced Motion and Sizing of Bank in Moving-Bank MMAE. *IEEE Transactions on Aerospace and Electronic Systems*, *VOL.40*(NO.3), pp. 770–779.

Mehra, R. K. (1970). On the Identification of Variances and Adaptive Kalman Filtering. *IEEE Transactions on Automatic Control*, *AC-15*(2), pp.175–184.

Nebot, E. M. (2005). *Navigation System Design.* Australian Centre for Field Robotics, The University of Sydney NSW 2006, Australia: Centre of Excellence for Autonomous Systems.

Nebot, E. M., & Durrant-Whyte, H. F. (1999). Initial Calibration and Alignment of Low Cost Inertial NAvigation Units for Land Vehicle Applications. *Journal of Robotics Systems*, *VOL. 16*(NO. 2), pp. 81–92.

Ormsby, C. D. (2003). *Generalized Residual Multiple Model Adaptive Estimation of Parameters and States.* Dissertation, Faculty of the Graduate, School of Engineering and Management of the Air Force Institute of Technology, Air University.

Oshman, Y., & Shaviv, I. (2000). Optimal Tuning of a Kalman Filter Using Genetic Algorithms. *AIAA Guidance, Navigation and Control Conference and Exhibit.*

Rago, C., Prasanth, R., Mehra, R. K., & Fortenbaugh, R. (1998). Failure Detection and Identification and Fault Tolerant Control using the IMM-KF with Applications to the Eagle-Eye UAV.

Roumeliotis, S. I., Sukhatime, G. S. & Bekey G. A. (1998a). Fault Detection and Identification in a Mobile Robot using Multiple-Model Estimation. *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 2223–2228.

Roumeliotis, S. I., Sukhatime, G. S. & Bekey G. A. (1998b). Sensor Fault Detection and Identification in a Mobile Robot. *Proceedings of IEEE/RSJ International Conference on Intelligent Robotics and Systems*, pp. 1383–1388.

Savage, P. G. (2006). "A Unified Mathematical Framework for Strapdown Algorithm Design". *Jornal of Guidance and Dynamics*, *Vol.29*(No.2), Pages(237–249).

Scheding, S. (1997). *High Integrity Navigation.* PhD thesis, Australian Centre for Field Robotics, Mechanical and Mechatronic Engineering, The University of Sydney.

Semeniuk, L., & Noureldin, A. (2006). Bridging GPS Outages Using Neural Network Estimates of INS Position and Velocity Errors. *Measurement Science and Technology*, *17*, pages 2783–2798.

Stremler, F. G. (1990). *Introduction to Communication Systems* (Third ed.). Addison Wesley.

Sukkarieh, S. (2000). *Low Cost, High Integrity, Aided Inertial Navigation Systems for Autonomous Land Vehicles.* PhD thesis, Mechanical and Mechatronic Engineering, Australian Centre for Field Robotics, The University of Sydney.

Sukkarieh, S., Nebot, E. M., & Durrant-Whyte, H. F. (1998). Achieving Integrity in an INS/GPS Navigation Loop for Autonomous Land Vehicle Applications. *IEEE International Conference on Robotics and Automation*, pp. 3437–3442.

Sukkarieh, S., Nebot, E. M., & Durrant-Whyte, H. F. (1999). A High Integrity IMU/GPS Navigation Loop for Autonomous Land Vehicle Applications. *IEEE Transactions on Robotics and Automation*, *VOL. 15*(NO. 3), pp. 572–578.

Titterton, D., & Weston, W. (2004). *Strapdown Inertial Navigation Technology* (2nd ed.). Number 5 in IEE Radar, Sonar and Navigation series 17. The Institution of Electrical Engineers (IEE).

Tsui, J. B.Y. (2000). *"Fundamentals of Global Positioning System Receivers: A Software Approach"*.

Van Trees, H., L. (1971). *Detection, Estimation, and Modulation Theory, Part I: Detection, Estimation, and Linear Modulation Theory.* John Wiley & Sons, Ltd.

Wang, J. J., Wang, J., Sinclair, D., & Watts, L. (1997). A Neural Network and Kalman Filter Hybrid Approach for GPS/INS Integration.

Wang, J. J., Wang, J., Sinclair, D., & Watts, L. (2006). Designing a Neural Network for GPS/INS/PL Integration. *International Global Navigation Satellite Systems Society.*

Weiss, J., D. & Kee, D., S. (2001). A Direct Performance Comparison Between Loosely Coupled and Tightly Coupled GPS/INS Integration Techniques. *IEEE Sensors Jornal*, *vol.*(No.), pages 537–544.

Welck, G. & Bishop, G. (1997). *An Introduction to the Kalman Filter.* Department of Computer Science, University of North Carolina at Chapel Hill.

White, N. A., Maybeck, P. S., & DeVilbiss, S. L. (1998). Detection of Interference/Jamming and Spoofing in a DGPS-Aided Inertial System. *IEEE Transactions on Aerospace and Electronic Systems*, *VOL.34*(NO.4), pp. 1208–1217.

Willsky, A. S. (1976). A Survay of Design Methods for Failure Detection in Dynamics Systems. *Automatica*, *VOL.12*, pp.601–611.

Willsky, A. S. & Jones, H., L. (1974). A Generalized Likelihood Ratio Approach to State Estimation in Linear Systems Subject to Abrupt Changes. *Proceedings of the 1974 IEEE Conference on Decision & Control, including the 13th Symposium on Adaptive Processes.*

Willsky, A. S., Chow, E., Y., Gershwin, S., B., Greene, C., S.,, Houpt, P., K. & Kurkjian, A., L. (1980). Dynamic Model-Based Techniques for the Detection of Incidents on Freeways. *IEEE Transactions on Automatic Control*, *VOL. AC-25*, pp. 347–360.

Yang, Q. (2004). *Model-Based and Data Driven Fault Diagnosis Methods with Applications to Process Monitoring.* PhD thesis, Elctrical Engineering Department, Case Western Reserve University.

Zhang, P., Gu, J., Milios, E., E. & Huynh, P. "Navigation with IMU/GPS/Digital Compass with Unscented Kalman Filter". *Proceedings of the IEEE Internatinal Conference on Mechatronics & Automation.*