Institute of Engineering, Surveying
and Space Geodesy

Integration of GPS and Low Cost INS measurements

By

Christopher Hide BSc

Thesis submitted to the University of Nottingham for
the degree of Doctor of Philosophy
September 2003

ABSTRACT

GPS and Inertial Navigation Systems (INS) are increasingly used for positioning and attitude determination in a wide range of applications. Until recently, the very high cost of the INS components limited their use to high accuracy navigation and geo-referencing applications. Over the last few years, a number of low cost inertial sensors have come on the market. Although they exhibit large errors, GPS measurements can be used to correct the INS and sensor errors to provide high accuracy real-time navigation.

The integration of GPS and INS is usually achieved using a Kalman filter which is a sophisticated mathematical algorithm used to optimise the balance between the measurements from each sensor. The measurement and process noise matrices used in the Kalman filter represent the stochastic properties of each system. Traditionally they are defined *a priori* and remain constant throughout a processing run. In reality, they depend on factors such as vehicle dynamics and environmental conditions.

In this research, three different algorithms are investigated which are able to adapt the stochastic information on-line. These are termed adaptive Kalman filtering algorithms due to their ability to automatically adapt the filter in real time to correspond to the temporal variation of the errors involved.

The algorithms used in this research have been tested with the IESSG's GPS and inertial data simulation software. Field trials using a Crossbow AHRS-DMU-HDX sensor have also been completed in a marine environment and in land based vehicle trials. The use of adaptive Kalman filtering

shows a clear improvement in the on-line estimation of the stochastic properties of the inertial system. It significantly enhances the speed of the dynamic alignment and offers an improvement in navigation accuracy. The use of the low cost IMU in a marine environment demonstrates that a low cost sensor can potentially meet the requirements of navigation and multi-beam sonar geo-referencing applications.

Acknowledgements

The research undertaken for this thesis was conducted at the Institute of Engineering, Surveying and Space Geodesy at the University of Nottingham. I would like to thank my academic supervisors Professor Terry Moore and Dr Martin Smith for their support and guidance throughout this research.

I would like to thank Dr Guohui Hu for his help with the inertial simulation software during his stay at the IESSG. I would also like to thank Mike Pattinson for his help with the GPS processing algorithms, along with Carmen Bonillo and Caroline Noakes for their guidance with the data simulation.

For their contribution to the field work that was undertaken for this thesis, I would like to thank Andy Evans, Sean Ince, Andy Nesbitt and Theo Veneboer. Also I would like to thank Norman Teferle for his help and guidance with Linux and LaTeX. I would also like to thank everybody else at the IESSG who have taken time to help me in various ways during my Ph.D.

Outside of the IESSG, I would also like to thank everyone who helped me from the Royal Naval Hydrographic School HMS Drake during the two days on the survey boat. In particular I would like to thank Rick Reed for his help during, and after the trials.

And finally, and most importantly, I would like to thank my parents for all their support through all of my years of study. Without their support this work would not have been possible.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Inertial Navigation Systems (INS) became widely used in military vehicles and civilian aircraft since the origin of inertial guidance during the second world war. Primarily, the high cost of such systems has restricted the use of INS for other navigation and surveying applications. However, in the last two decades, two significant developments have broadened the use of INS: integration of INS with the Global Positioning System (GPS) and the development of lower cost inertial components.

Combining INS with other navigation systems such as GPS has gained significance due to both systems having complimentary error characteristics. GPS provides consistent accuracy position and velocity provided that there is line-of-sight between the GPS receiver and the orbiting GPS satellites. The latest generation of receivers are capable of providing Real-Time Kinematic (RTK) positioning to centimetre level accuracy by using relative positioning over short baselines (Wang et al., 1999). In order to

achieve such accuracy, integer ambiguity resolution is required in order to resolve the unknown number of integer carrier phase ambiguities between the GPS receiver and satellite.

On the other hand, INS is an autonomous system that does not require measurements to external signals. It provides high accuracy short-term position, velocity and attitude at a high data rate. INS does, however, require accurate knowledge of the initial position and attitude, and the accuracy reduces over time due primarily to the imperfections in the inertial sensors.

Combining GPS and INS measurements can greatly reduce the shortcomings of each standalone system to provide increased reliability and accuracy. The GPS estimates of position and velocity are used to restrict the growth of the INS errors over time and allow estimation of the inertial sensor errors. The INS provides high accuracy short term navigation information at a high data rate which can also be used to aid GPS ambiguity resolution, correct cycle slips and bridge periods when there is no signal reception.

Micro Electro-Mechanical Sensors (MEMS) have emerged in the last 20 years as a low cost method for mass producing small, lightweight accelerometers and gyros (Barbour et al., 1996). MEMS accelerometers have developed primarily from the automotive industry where they are used for triggering safety devices (Yazdi et al., 1998). MEMS accelerometers and gyros have diverse applications in other fields such as virtual reality, robotics, and machine control and monitoring, and are increasingly being used in navigation applications. However, they generally provide only short-term navigation capability due to large sensor errors that require

continuous calibration. GPS measurements can be used to estimate these sensor errors.

The use of lower cost inertial components with GPS has opened up a diverse range of applications. Originally, GPS and INS has been confined to high accuracy navigation and geo-referencing applications due to the high cost of the technology, and also due to government regulation. Examples of geo-referencing imaging sensors in airborne applications can be found in Cramer et al. (2000), Lithopoulos (1998) and Škaloud et al. (1996) and for geo-referencing multi-beam echo sounders in a marine environment in Alkan and Baykal (2001) and Trethewey et al. (1999). Low cost inertial sensors potentially provide the capability to replace higher cost sensors in these types of application.

Now, due to the potential for high accuracy mass-fabricated low cost MEMS sensors, GPS and INS applications have expanded to fill a range of new navigation applications such as personal and vehicular navigation where increased performance and robustness is required over standalone GPS systems. An example of personal navigation using MEMS can be found in Ladetto (2000) and for vehicle navigation in Shin (2001) and Sukkarieh et al. (1999).

1.1.1 Integration methodology

The Kalman filter has emerged as the dominant data fusion method for integrating the two systems as it allows estimation of time-dependent variables termed *states* (Wang et al., 1999). The Kalman filter uses mathematic and stochastic models to estimate the INS error using measurements provided by GPS. The mathematical model comprises of a functional model to relate

the measurements to the states to be estimated, and a dynamic model to describe the way in which the INS errors develop over time. These models are considered to be well defined for GPS and INS systems (Mohamed and Schwarz, 1999; Wang et al., 1999).

The stochastic information describing each system is used by the Kalman filter in order to correctly weight the new measurements to update the states. The stochastic information is obtained using the specification of the GPS and INS systems provided by the manufacturer, or is estimated from empirical testing in a process termed *tuning*. This information is defined *a priori* and generally remains constant throughout processing runs. This has been identified as a deficiency in the conventional Kalman filter algorithm for navigation applications where the stochastic properties of the sensors change over time (Barnes et al., 1998; Collins and Langley, 1999; Mohamed and Schwarz, 1999; Satirapod, 2002; Wang et al., 1999). These errors depend on factors such as vehicle dynamics and environmental conditions and are consequently difficult to estimate.

One example of the time varying nature of low cost INS errors is highlighted during the initialisation and alignment of the INS using GPS updates. Initially, the INS navigation and sensor errors can be large and they vary each time the sensor is switched on. Using a low noise estimate for the INS errors results in a precise, yet most likely biased, estimate which results in a long transition to the correct value. Conversely, a larger noise estimate results in a quicker transition to the correct value but results in a noisier estimate which increases the navigation errors. This highlights the need for adapting the stochastic information used in the filter as the inertial sensor errors are reduced through alignment and the stochastic properties

of the INS change.

In order to reduce the dependence on *a priori* information and allow the filter to adapt on-line, several extensions to the conventional Kalman filtering algorithm exist. These techniques use information from the innovation or residual sequence in the conventional Kalman filtering algorithm to supply additional stochastic information which is used to adapt the *a priori* information. This adaptation of the stochastic properties of the filter is generically termed *adaptive Kalman filtering*. The primary potential benefit of adaptive Kalman filtering is improved performance in terms of reducing the navigation errors of position, velocity and attitude. This includes reducing the time required for INS alignment and resolution of the initial sensor errors.

1.2 Research Aims and Objectives

The aim of the research undertaken in this thesis is to analyse and improve the system performance of GPS integrated with a low cost Inertial Measurement Unit (IMU). The term 'low cost' is used to represent MEMS grade sensors that cost typically less than £5,000 for an IMU assembly. The originality of the research lies in investigating the use of three different adaptive Kalman filtering algorithms for this application area. Only one of the algorithms has been used in GPS and INS integration before, and that research was carried out with a higher grade INS (Mohamed and Schwarz, 1999; Wang et al., 1999). Novel research is also carried out into the use of GPS and low cost INS in a marine environment as a potential replacement for higher cost systems.

The broad aims of this research are therefore summarised as follows:

- Undertake research into improving the system performance of integrated GPS and low cost INS.
- Develop and investigate the performance of an integrated GPS and low cost INS for marine applications.

More specifically, the objectives of this research are:

- To investigate the performance of conventional Kalman filtering for GPS and low-cost INS.
- To propose an alternative to the conventional Kalman filter algorithm that automatically adapts to the temporal variation of the low cost INS errors.
- To undertake simulated data and practical testing of the proposed adaptive Kalman filtering algorithms.

The research undertaken in this thesis demonstrates that adaptive Kalman filtering algorithms can be used to improve the stochastic information used in the filter. This has the effect of reducing the dependence of the initial stochastic information used in the filter which is often difficult to estimate. It results in significant reduction in the time taken to align the INS and resolve the initial sensor errors. Furthermore, it can also result in a reduction in the navigation errors.

1.3 Research methodology

The following research methodology is carried out in this thesis to meet the aims and objectives outlined in the previous section:

- Review existing integration algorithms for GPS and low cost INS.
- Identify alternative algorithms to the conventional Kalman filter which are able to adapt the stochastic information used by the filter depending on the new measurements.
- Develop integration software that provides the capabilities of current integration algorithms and implement the adaptive algorithms.
- Use data simulation to investigate the performance of the adaptive algorithms (this includes contributing to the development of the INS simulation algorithms)
- Undertake practical testing to investigate the performance of GPS and low cost INS for marine applications by comparing the results to a higher grade reference system.
- Undertake practical testing of the adaptive Kalman filtering algorithms in a marine and land-based vehicle environment and use this to validate the results obtained using simulation.

1.4 Thesis Outline

Chapter 2 describes the general principle of operation for low cost inertial sensor assemblies. The essential principles of mechanising the raw acceleration and turn rate measurements to obtain estimates of position, velocity and attitude are described.

The Global Positing System is described in Chapter 3. This provides an overview of the fundamental principles of GPS with particular attention

given to the error sources that corrupt the GPS signals. The chapter also describes existing stochastic estimation methods for GPS observations.

Chapter 4 describes the principles of Kalman filtering and its application to GPS and INS integration. An overview of the conventional Kalman filter algorithm is given and the methods that exist to adapt the stochastic properties of the Kalman filter are described. The way in which the Kalman filter can be used for integrating the two systems is then examined and an overview of the development of the integration software used for this thesis is given.

The data simulation software for simulating GPS and inertial measurements developed at the University of Nottingham is described in Chapter 5 with specific reference given to the contribution provided by the author.

Chapter 6 describes the results obtained using measurement simulation with the conventional Kalman filtering algorithm. This section shows the characteristics of the Kalman filter including the effect that the process noise and initial covariance have on the time to alignment, and also the properties of the innovation and residual sequence. This analysis motivates the use of adaptive Kalman filtering algorithms to provide improved system performance over the conventional algorithm.

Chapter 7 examines the use of the adaptive algorithms using the software described in Chapters 4 and 5. Chapter 7 uses a simulated marine trial to examine the adaptive techniques for alignment of the INS and resolution of the inertial sensor errors. Each adaptive method is described and its performance examined. The true inertial sensor errors from the data simulation are used to analyse the time required to resolve the sensor errors and align the INS.

Chapter 8 applies the algorithms to real data collected from a low cost IMU in a marine trial on a small survey boat in Plymouth. The data collection method is described, and the performance from the conventional Kalman filter is examined. In view of the results obtained, attitude estimation from dual antenna GPS receivers is used to improve the yaw estimation of the low cost trial. The performance of the adaptive Kalman filtering algorithms is then examined and compared to the results obtained using simulation. A second trial is also undertaken to assess the performance of the adaptive Kalman filtering algorithms in a land based vehicle trial.

Chapter 9 concludes the thesis with a summary of the results achieved during this research. Ideas for extending the research are also presented.

Chapter 2

Strapdown Inertial Navigation

2.1 Introduction

Strapdown inertial navigation is a Dead Reckoning (DR) form of navigation. This means that navigation is achieved by measuring direction and displacement from an initial point and orientation. This is achieved through measuring acceleration and turn rates in three orthogonal directions with accelerometers and gyros. The sensor assembly of accelerometers and gyros is termed an Inertial Measurement Unit (IMU).

The objective of this chapter is to cover the fundamental concepts of inertial navigation with the mechanisation of raw measurements along sensor axes to estimates of position, velocity and attitude. In particular, reference is given to the way in which GPS position and velocity updates can be used with knowledge of the inertial error models to reduce the build up of inertial errors over time. The chapter begins with an overview of current inertial sensor technology.

2.2 Low Cost Inertial Sensor Technology

The lowest cost inertial sensors for navigation applications are provided by Micro-machined Electromechanical Sensor (MEMS) technology.

2.2.1 MEMS sensors

MEMS gyros use the Coriolis theorem in order to measure rotation rate. Figure 2.1 shows the principle of Coriolis acceleration. Consider an observer sitting on the x-axis of the coordinate system shown, with the coordinate system rotating about the z-axis at the rotation rate, Ω. If a particle travels along the y-axis at a velocity v, an acceleration will be apparent to the observer perpendicular to the y-axis with magnitude $a = 2v \times \Omega$. This is the principle that is used in MEMS gyros.

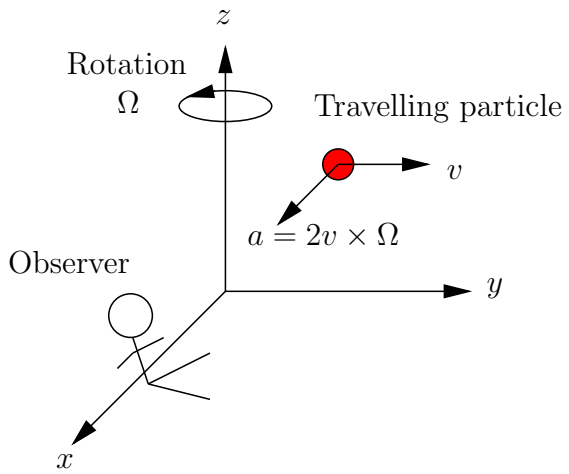


Figure 2.1: Coriolis force (Yazdi et al., 1998)

Low cost inertial sensors contain a vibrating silicon MEMS structure. When the gyro is rotated, this results in a Coriolis acceleration perpendicular to the input axis and proportional to the input rotation. The deflection can be measured to derive the angular rate measurement. The advantage of this type of sensor is that it contains no rotating parts. Consequently,

the sensor can be miniaturised and batch fabricated using micro-machining techniques resulting in a sensor that is small and low power (Yazdi et al., 1998).

MEMS accelerometers consist of a proof mass that is suspended by compliant beams. The deflection of the proof mass is measured under acceleration either by measuring the displacement of the proof mass (open loop), or more typically by measuring the force required to maintain its position (closed loop). This again results in a sensor with no moving parts that can be miniaturised and batch fabricated.

2.2.2 Performance

Table 2.1 compares the dominant error sources of low cost MEMS IMUs (often termed automotive grade due to the original applications for which these sensors were developed) to the higher grade ring laser and fibre optic-based gyro systems. The table uses the Crossbow AHRS-DMU-HDX as an example of a low cost MEMS sensor assembly. The Crossbow unit is available for civilian applications and costs approximately £4,000. The sensor contains additional electronics and a magnetometer to allow standalone attitude operation.

The standalone performance of low cost IMUs for navigation is limited by the large sensor errors. For example, the table shows that the Crossbow has a bias uncertainty of 1°/s compared to the high performance Ring Laser Gyro with a drift of approximately 0.1°/hr. For standalone inertial navigation, the manufacturer's specification of the Litton LN90-100 is 0.8nmi/hr (Northrup Gruman., 2001a) which allows such a system to be used for commercial and military aircraft navigation applications. However, the

Grade	Navigation	Tactical	Low Cost
Example	Litton LTN90-100	Litton LN200	Crossbow AHRS-DMU-HDX
Dimensions (cm)	-	8.9×8.9×8.5	7.62×9.53×10.41
Cost (approx)	>£100k	≈£20k	≈£4k
Gyro	Ring Laser	Fibre Optic	MEMS
Bias (°/h)	0.005-0.1	1-10	<3600
Scale Factor	5ppm	100ppm	<1%
Noise ($°/h/\sqrt{Hz}$)	0.002	0.04-0.1	<0.85
Accelerometer	Silicon	Silicon	Silicon
Bias	$50\mu g$	$200\mu g - 1mg$	$< \pm 30mg$
Scale Factor (ppm)	50	300	$< \pm 1\%$
Noise	$50\mu g$ (1σ)	$50\mu g/\sqrt{Hz}$	<0.15 m/s/$\sqrt{Hr}$

Table 2.1: Strapdown IMU sensor error characteristics
Sources Mohamed (1999); Northrup Gruman. (2001b); Crossbow Technology Inc. (2000)

cost of high accuracy IMUs such as this is approximately 10 times the cost of a low or medium cost INS (Schwarz, 1999). The inertial sensor errors of the Crossbow IMU mean that navigation errors can reach kilometres in minutes. Consequently, standalone navigation for long periods of time is not possible with such sensor technology.

2.3 Coordinate frames and rotations

The mechanisation of the inertial measurements from the gyros and accelerometers into estimates of position, velocity and attitude requires the definition of a number of coordinate frames.

2.3.1 Inertial frame

The inertial coordinate frame is geocentric (has origin at the Earth's centre of mass) and is non-rotating with respect to the fixed stars. This means the coordinate system is non-accelerating. The z-axis coincides with the mean rotation axis of the Earth, and the x-axis points toward the mean vernal equinox. The y-axis completes the right-handed coordinate system. All measurements in an inertial system are measured with respect to the inertial frame.

2.3.2 Earth fixed frame

The Earth fixed frame is geocentric and rotates relative to the inertial frame about the mean rotation axis of the Earth (z-axis). The x-axis points toward the intersection of the Greenwich Meridian and the Equator. The y-axis completes the right handed coordinate system. The approximation to the Earth's rotation rate about the z-axis is given as,

$$\omega_{ie} = 2\pi \frac{1 + 365.25}{365.25 \times 24 \times 60 \times 60} \approx 7.292 \times 10^{-5} rad/s \qquad (2.1)$$

The navigation errors caused by using the mean axis of rotation and the Earth rate approximation are considered to be negligible (Farrell and Barth, 1999). In addition to the gyros detecting the Earth rotation, the rotation of the Earth will result in the accelerometers detecting a Coriolis acceleration.

2.3.3 Navigation frame

The navigation frame (sometimes termed the Geographic frame) is defined with respect to a reference ellipsoid. The navigation frame origin coincides with the origin of the body frame (§2.3.4). The z-axis is defined as pointing

down the ellipsoidal normal, the x-axis points to ellipsoidal north and the y-axis points to ellipsoidal east. When the origin of the navigation frame moves with respect to the Earth frame, the navigation frame rotates with respect to the Earth frame at the transport rate, ω_{en} (§2.5.2).

2.3.3.1 Earth to navigation frame rotation

Measurements from other navigation systems such as GPS are often measured in the Earth frame. Measurements in the navigation frame are converted to measurements in the Earth fixed frame by two rotations. The first rotation is about the Earth's z-axis to align the y-axis with the east axis of the navigation frame. The second rotation is about the new y-axis to align the z-axis with the new down axis. This results in the rotation matrix,

$$C_n^e = \begin{pmatrix} -\sin\lambda\cos\phi & -\sin\phi & -\cos\lambda\cos\phi \\ -\sin\lambda\sin\phi & \cos\phi & -\cos\lambda\sin\phi \\ \cos\phi & 0 & -\sin\phi \end{pmatrix} \tag{2.2}$$

The resultant matrix is orthogonal so that measurements can be converted from the Earth frame to the navigation frame using the transpose,

$$C_e^n = (C_n^e)^{-1} = (C_n^e)^T \tag{2.3}$$

2.3.4 Body and vehicle frames

The instrument axes in an IMU are aligned to the body frame. In most applications it is the position and attitude of the vehicle in which the sensor is rigidly mounted that is of interest. In these instances, the body frame is rigidly fixed to the vehicle frame. The x-axis of the vehicle frame points to

the front of the vehicle, the z-axis points to the bottom of the vehicle and the y-axis completes the right-handed coordinate system.

2.3.4.1 Body to navigation frame rotation

The body frame is rotated to align with the navigation frame using a series of rotations about the x, y and z-axes. The angular rotations about each axis are known as Euler angles and are termed roll, φ, pitch, θ, and yaw, ψ, respectively. The order of rotation is important as a different order can result in a different rotation. The three sequential rotations form the Direction Cosine Matrix (DCM). The DCM from the navigation frame to the body frame is defined as,

$$\begin{aligned} C_n^b &= C_3 C_2 C_1 \\ &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\psi & \sin\psi \\ 0 & -\sin\psi & \cos\psi \end{pmatrix} \begin{pmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{pmatrix} \begin{pmatrix} \cos\varphi & \sin\varphi & 0 \\ -\sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{pmatrix} \end{aligned} \tag{2.4}$$

Because the DCM is orthogonal, the rotation from the body to the navigation frame is easily calculated from its transpose,

$$C_b^n = (C_n^b)^{-1} = (C_n^b)^T = C_1^T C_2^T C_3^T \tag{2.5}$$

2.3.5 Quaternions

An alternative to computing rotations using the DCM is the quaternion (Titterton and Weston, 1997). Quaternions use the principle that the transformation between coordinate systems can be achieved by a rotation $\|\mu\|$ about a three-element vector $\mu = [\ \mu_x \quad \mu_y \quad \mu_z\ ]^T$ (ibid.). The quaternion

is defined as,

$$\begin{aligned} q = & \quad \begin{bmatrix} q_1 & q_2 & q_3 & q_4 \end{bmatrix}^T \\ = & \quad \begin{pmatrix} \mu \left(\frac{1}{\|\mu\|} \sin \left(\frac{\|\mu\|}{2} \right) \right) \\ \cos \left(\frac{\|\mu\|}{2} \right) \end{pmatrix} \end{aligned} \tag{2.6}$$

The update equation for the quaternion vector is obtained by integrating the equation,

$$\dot{q} = \Omega q \tag{2.7}$$

where Ω is the 4×4 skew symmetric form of the vector of gyro measurements. The update of the quaternion vector is performed using discrete measurement updates with the equation,

$$q(i+1) = [\alpha I + \beta \Upsilon] q(i) \tag{2.8}$$

where Υ is the 4×4 skew symmetric form of the integrated angular measurements and,

$$\alpha = \cos \left(\frac{\|\upsilon\|}{2} \right) \tag{2.9}$$

$$\beta = \frac{1}{\|\upsilon\|} \sin \left(\frac{\|\upsilon\|}{2} \right) \tag{2.10}$$

where υ is the vector of integrated angle measurements.

The DCM can be written directly in terms of elements of the quaternion and similarly, the quaternion can be formed directly using elements of the DCM. Normalisation of the quaternion vector may be required when the square of the elements does not sum to unity due to computer rounding errors. The quaternion can be normalised using an equation such as,

$$q_{norm} = q / \sqrt{q^T q} \tag{2.11}$$

Care must be taken when using normalisation such as this as the error can be spread between elements (Titterton and Weston, 1997).

2.4 Gravity compensation

The measurements of acceleration due to system movements and the acceleration due to gravity cannot be distinguished because of relativistic theory (Chen, 1992). A gravity model is therefore required to calculate the movement of the sensor.

One of the reasons for selecting the navigation frame for mechanising the inertial measurements is that the gravity model is simpler to define (Škaloud, 1995). The gravity vector is formed using the equation,

$$g^n = \begin{bmatrix} 0 \\ 0 \\ \gamma(\lambda, h) \end{bmatrix} + \begin{bmatrix} \zeta_g \\ -\eta_g \\ \delta_g \end{bmatrix} \tag{2.12}$$

where ζ_g,η_g and δ_g are gravity anomaly terms and,

$$\gamma(\lambda, h) = a_1(1 + a_2 \sin^2 \lambda + a_3 \sin^4 \lambda) + (a_4 + a_5 sin^2 \lambda)h + a_6 h^2 \tag{2.13}$$

with the coefficients $a_1, \ldots, a_6$ for the WGS-84 coordinate system defined in Shin (2001).

2.5 Navigation frame initialisation and mechanisation

The navigation frame mechanisation approach is described in texts such as Britting (1971), Titterton and Weston (1997) and Farrell and Barth (1999). The navigation frame is used in this thesis, however other coordinate frames can be used for the mechanisation process. Using the transformation given in §2.3.3.1, measurements can be easily converted between the Earth

fixed frame and the navigation frame when combining inertial navigation measurements with GPS measurements.

2.5.1 Initialisation and alignment

The iterative nature of the navigation mechanisation means that the initial navigation quantities of position, velocity and attitude must be provided through some form of initialisation process. The term *initialisation* is usually given to the process of determining the initial position and velocity of the sensor, whereas the term *alignment* is given to the process of determining the initial attitude parameters.

In an integrated system, position and velocity can be initialised using GPS. Alignment of the body frame to the navigation frame is obtained in a static environment using analytical formulae which use known gravity and Earth rate measurements. The Earth rate measurement is required in order to obtain the initial yaw estimate. However, for low cost gyros, the Earth rate is below the instrument sensitivity therefore the yaw must be initialised using an external aiding source, for example from a compass or from a multi-antenna GPS system. Static alignment is obtained in two stages: an initial coarse alignment followed by a fine alignment. For situations where either a static alignment is impossible or impractical, the alignment is performed using a dynamic alignment.

2.5.1.1 Coarse alignment

The coarse alignment is the process of using analytical formulae to directly compute an approximate attitude estimate from the raw inertial sensor outputs. The raw accelerometer measurements are used to obtain an

estimate of the initial roll and pitch angles by comparing the vector of body frame measurements, $f^b = [\ f_x \quad f_y \quad f_z\ ]^T$ to the known gravity vector. By analysis of the equation,

$$g^n = C_b^n f^b \tag{2.14}$$

the initial pitch and roll estimates can be obtained using,

$$\varphi = \text{atan2}(-f_y, f_x) \tag{2.15}$$

$$\theta = \text{atan2}(f_x, \sqrt{f_y^2 + f_z^2}) \tag{2.16}$$

This method is only valid for static initialisation where there are no external forces acting on the sensors, and the accuracy of the alignment is directly dependent on the sensor errors of the accelerometers. Coarse alignment using the gyro measurements is not considered here due to the Earth rate being below the instrument sensitivity for low cost sensors.

2.5.1.2 Fine alignment

The fine alignment is the process of refining the attitude obtained from the coarse alignment. The raw inertial measurements are combined using Zero Velocity Update Points (ZUPT) with knowledge of the way that the inertial errors develop over time (see §2.6.1). This is usually achieved using a Kalman filter (Chapter 4.2) using the observations,

$$\delta\omega^n = \omega^n - \hat{\omega}^n \tag{2.17}$$

$$\delta f^n = f^n - \hat{f}^n \tag{2.18}$$

where ω^n and f^n are the known Earth rotation and gravity vectors and $\hat{}$ signifies the measurement obtained from the inertial sensor. The use of the fine alignment for low cost sensors is again restricted due to the sensitivity of the gyros.

2.5.1.3 Dynamic alignment

The previous initialisation techniques require the inertial sensor to remain static in order that the inertial sensors are only measuring the Earth's rotation and acceleration due to gravity. The low cost sensors are unable to sense the Earth's rotation, and the large inertial sensor errors result in inaccurate estimates of the initial attitude. Consequently, a dynamic alignment is used in order to estimate the navigation errors and also the inertial sensor errors. This is usually achieved using a Kalman filter using external estimates of position and velocity from a system such as GPS. The dynamic alignment is also used after a static initialisation to reduce the drift of the inertial sensor over time. The dynamic alignment is the focus of the subsequent chapters in this thesis.

2.5.2 Navigation frame mechanisation

The navigation frame mechanisation is shown in Figure 2.2. The iterative procedure begins with the raw gyro and accelerometer measurements (ω_{ib}^{b} and f^{b} respectively) being sampled at discrete intervals. If information about the sensor errors is available either from laboratory tests or an on-line calibration, the corrections are applied to the raw sensor outputs. The gyro measurements measure the total sensor rotation in the inertial frame which is compensated for the Earth's rotation, ω_{ie}^{n}, and the navigation frame transport rate, ω_{en}^{n}. This gives the turn rate of the body frame with respect to the navigation frame referenced in the body frame,

$$\omega_{nb}^{b} = \omega_{ib}^{b} - C_{n}^{b}\left(\omega_{ie}^{n} + \omega_{en}^{n}\right) \tag{2.19}$$

where,

$$\omega_{ie}^{n} = \begin{bmatrix} \omega_{ie}\cos\lambda & 0 & -\omega_{ie}\sin\lambda \end{bmatrix}^{T} \tag{2.20}$$

$$\omega_{en}^{n} = \begin{bmatrix} \frac{v_E}{R_\lambda + h} & \frac{-v_N}{R_\phi + h} & \frac{v_E \tan\lambda}{R_\lambda + h} \end{bmatrix}^{T} \tag{2.21}$$

and v is the velocity in the navigation frame, h is the ellipsoidal height and R is the latitudinal (λ) or longitudinal (ϕ) radius of the Earth.

The compensated gyro output is used to calculate the updated attitude by using the methods described in §2.3.5. The updated attitude is then used to resolve the specific force measurement from the body frame to the navigation frame. The force in the navigation frame is then compensated for gravity and Coriolis acceleration using,

$$\dot{v}^{n} = f^{n} - (2\omega_{ie}^{n} + \omega_{en}^{n}) \times v^{n} + g^{n} \tag{2.22}$$

Equation 2.22 is integrated to give velocity in the navigation frame and hence longitudinal and latitudinal rates. A second integration yields position. The process then repeats for the next inertial measurements. The measurements from the previous iteration are then used in the correction terms for the next iteration.

It should also be noted that in Equation 2.21, the tangent function will cause problems at extreme latitudes. Since the navigation frame has its axes aligned with North and East directions, at extreme latitudes a small movement will result in a large rotation being necessary to maintain the orientation of the navigation frame (Salychev, 1998). In these instances, the Wander frame is used to overcome this problem by rotating the North axis by an angle termed the wander angle (ibid.).

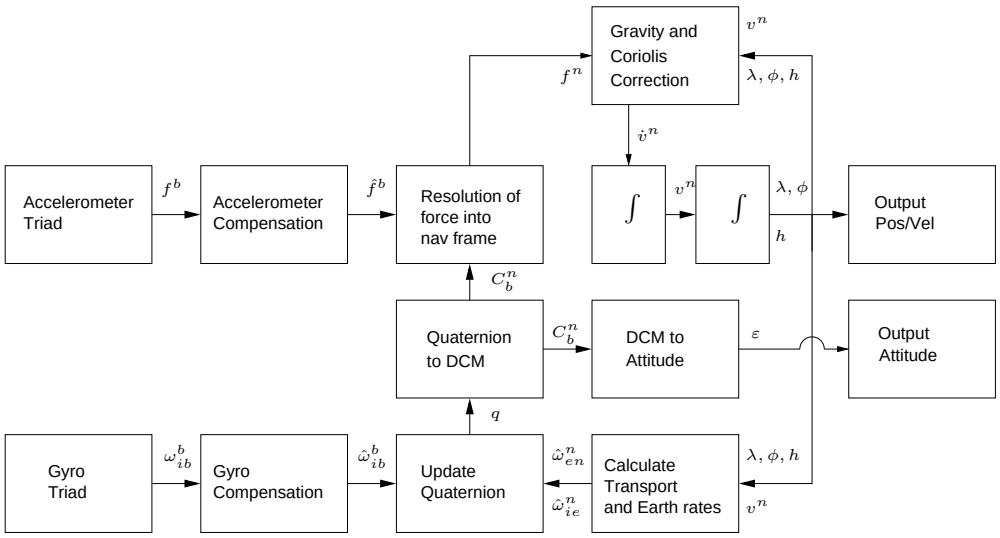


Figure 2.2: Inertial navigation frame mechanisation algorithm

2.6 Inertial navigation error estimation

The INS errors can be estimated using position and velocity updates from GPS. This is accomplished by using knowledge of the way the inertial errors develop over time. This section describes the INS error models.

2.6.1 Navigation frame error model

Linear INS error models are obtained through perturbation analysis in texts such as Britting (1971); Farrell and Barth (1999); Rogers (2000). The following equations describe the position, velocity and attitude dynamics of the INS errors in the navigation frame,

$$\delta\dot{p}^n = \delta v^n \tag{2.23}$$

$$\begin{aligned}\delta\dot{v}^n = {} & f^n \times \varepsilon^n - (2\omega_{ie}^n + \omega_{en}^n) \times \delta v^n + \\ & v^n \times (2\delta\omega_{ie}^n + \delta\omega_{en}^n) + \delta\gamma^n + C_b^n \delta f_b\end{aligned} \tag{2.24}$$

$$\dot{\varepsilon}^n = -\omega_{in}^n \times \varepsilon^n + \delta\omega_{in}^n - C_b^n \delta\omega_{ib}^b \tag{2.25}$$

where δp, δv and ε are the vectors of position, velocity and attitude errors, $\delta\gamma$ is the gravity vector error and $\times$ is the cross product operator. The error equations are used to form the dynamics matrix for estimating the inertial errors using position and velocity updates from GPS (see Chapter 4).

2.6.2 Attitude error dynamics

When using GPS updates, the inertial position and velocity errors are directly estimated using the GPS estimates of these errors. However, in the absence of external attitude observations, the INS attitude error is only observable through the velocity error dynamics. From Equation 2.24, the velocity error due to attitude misalignment in the navigation frame develops through the following system of equations,

$$\delta\dot{v}_N = -f_D\varepsilon_E + f_E\varepsilon_D \tag{2.26}$$

$$\delta\dot{v}_E = f_D\varepsilon_N - f_N\varepsilon_D \tag{2.27}$$

$$\delta\dot{v}_D = -f_E\varepsilon_N + f_N\varepsilon_E \tag{2.28}$$

where the subscripts N, E and D, denote North, East and Down. It is observed from this equation that the attitude misalignments about the north and east axes result in velocity errors through the north, east and down force terms. The force in the down direction is always large due to the force due to gravity, therefore the north and east axis misalignments are constantly observable through velocity updates. Equations 2.26 and 2.27 show that the misalignment about the down axis is only observable through the horizontal acceleration terms, f_N and f_E. The misalignment about the down axis is therefore only observable through velocity aiding when the horizontal acceleration is non-zero.

The attitude misalignments in the navigation frame are related to the roll, pitch and yaw misalignments through the equations,

$$\delta\varphi = \varepsilon_E \frac{\sin\psi}{\cos\theta} + \varepsilon_N \frac{\cos\psi}{\cos\theta} \tag{2.29}$$

$$\delta\theta = \varepsilon_E \cos\psi - \varepsilon_N \sin\psi \tag{2.30}$$

$$\delta\psi = \varepsilon_E \sin\psi \tan\theta + \varepsilon_N \cos\psi \tan\theta + \varepsilon_D \tag{2.31}$$

These equations demonstrate that the roll and pitch errors are dependent on the misalignments about the east and north axes. The yaw error mainly depends on the misalignment about the down axis. Therefore, in order to aid alignment in an integrated system, the host vehicle is usually required to perform a series of manoeuvres to induce horizontal acceleration. This is usually achieved by performing a series of figure of eight turns.

2.6.3 Sensor error estimation

The previous section described the dynamics of the INS navigation errors. Equations 2.24 and 2.25 contain the terms, $C_b^n \delta f_b$, and, $C_b^n \delta w_{ib}^b$, which are the dynamics of the inertial sensor bias errors. The error characteristics of the inertial sensors are considered in detail in Chapter 5.

It is also possible to estimate other inertial sensor errors given knowledge of the way the errors effect the navigation errors. The dynamics of the main inertial error sources are given by the equation (Wolf et al., 1997; Farrell and Barth, 1999),

$$\dot{\varepsilon}^n = C_b^n \delta b_G + \Delta T C_b^n \delta t_G + C_b^n \Omega_b \delta s_G + C_b^n \Psi_b \delta m_G \tag{2.32}$$

$$\delta\dot{v}^n = C_b^n \delta b_A + \Delta T C_b^n \delta t_A + C_b^n F_b \delta s_A + C_b^n E_b \delta m_A \tag{2.33}$$

where the A and G subscripts denote the accelerometer and gyro errors, and

b, t, s, m represent the sensor bias, temperature dependent bias, scale factor and axis misalignment errors respectively. ΔT is the difference between the temperature output and the last calibration temperature of the IMU. Ω_b and F_b are diagonal matrices with the gyro and accelerometer outputs forming the diagonal elements. Ψ_b and E_b are 3×6 matrices formed using,

$$\begin{pmatrix} -x_2 & x_3 & 0 & 0 & 0 & 0 \\ 0 & 0 & x_1 & -x_3 & 0 & 0 \\ 0 & 0 & 0 & 0 & -x_1 & x_2 \end{pmatrix} \tag{2.34}$$

where x is the accelerometer or gyro output respectively.

Chapter 3

Global Positioning System

3.1 Introduction

GPS provides a method for directly obtaining instantaneous position and velocity estimates using passive range measurements from satellites. GPS measurements are integrated with inertial measurements to prevent the INS performance from deteriorating over time. This chapter examines the essential principles of GPS and, in particular, the error sources in the GPS observations. Further attention is also given to the measurement noise estimation for GPS observations.

General GPS concepts are well documented in papers and textbooks, for example, Bingley and Roberts (1998), Hofmann-Wellenhof et al. (1997), Kaplan (1996), Leick (1995) and Xu (2003). For further details of other aspects of GPS, the reader is referred to the aforementioned sources.

3.2 General description

The NAVigation System with Timing And Ranging (NAVSTAR) Global Positioning System (GPS) is a 24 hour, all-weather, passive, satellite positioning system. It provides high accuracy, instantaneous position, velocity and time information to global users.

GPS satellites transmit two L-band (microwave) frequencies termed L1 (1575.42MHz) and L2 (1227.6MHz) derived from the fundamental GPS frequency of 10.23MHz. This results in L1 and L2 carrier wavelengths of approximately 19cm and 24cm respectively. Two codes are modulated onto the carriers characterised by a Pseudo-random Noise (PRN) sequence. The Coarse/Acquisition (C/A) code is modulated onto the L1 carrier only and has a corresponding wavelength of approximately 300m. The C/A code is currently the only signal intentionally available to the civil community (Shaw et al., 2000). The Precise (P) code is modulated onto both L1 and L2 signals and has a wavelength of 30m. In order to restrict the P-code to authorised users, the P-code is encrypted to the Y-code. This is termed Anti Spoofing (A-S).

In addition to the two PRN codes, a navigation message is transmitted which contains the satellite ephemeris, satellite clock coefficients, satellite health information and ionospheric modelling coefficients. Almanac and Hand-Over-Word are also transmitted to assist the receiver in locating the satellites and identifying which segment of the P-code to search.

The GPS control segment consists of ground-based tracking and control stations. These stations monitor and control the satellites which includes determining the satellite orbits and uploading the navigation messages.

3.3 GPS observables

GPS provides three fundamental types of observation which are termed the pseudorange observation, the carrier phase observation and the Doppler observation.

3.3.1 Pseudorange Observation

The pseudorange observation is formed by matching the received code from a GPS satellite with a receiver's internally generated code. The time difference between the transmitted and received signal is scaled by the speed of light to obtain a distance measurement between receiver and satellite. The actual range measurement, ρ, between a satellite, s, and receiver, r, is written as,

$$\rho = \sqrt{(x_s - x_r)^2 + (y_s - y_r)^2 + (z_s - z_r)^2} \tag{3.1}$$

where x, y and z are the satellite and receiver Cartesian coordinates.

The pseudorange observation is formed from the true range measurement corrupted by a number of error sources which are considered in the following sections. The measured pseudorange observation in metres, p, is written in terms of these error sources using the equation,

$$p = \rho + c(dt - dT) + d_{ion} + d_{trop} + d\rho + e_\rho \tag{3.2}$$

where,

ρ	is the true receiver to satellite range (m)
c	is the speed of light (m/s)
dt	is the satellite clock error (s)
dT	is the receiver clock error (s)

d_{ion}	is error caused by ionospheric delay (m)
d_{trop}	is error caused by tropospheric delay (m)
$d\rho$	is the satellite orbit error (m)
e_ρ	is the measurement noise including multipath (m)

The pseudorange observation is obtained from the C/A code or P-code measurements and is used directly to obtain an estimate of position. Pseudoranges are also required for initialising ambiguity searches for carrier based positioning when initial coordinates are unavailable or are of low accuracy.

3.3.2 Carrier Phase Observable

The carrier phase observation is obtained by stripping the code from the received signal (Langley, 1997). Modern GPS receivers are capable of measuring the carrier observation to better than 0.01 cycles which equates to approximately millimetre measurement accuracy. The GPS receiver measures the fractional phase and will measure the accumulated integer number of wavelengths since signal lock-on. The total number of integer wavelengths at lock-on is unknown and is referred to as the integer ambiguity. The technique of resolving the integer ambiguity with no requirement for the receiver to be static is called On-The-Fly GPS (OTF) . For this thesis, ambiguity resolution is achieved using the LAMBDA method (see §4.5.1.1). The carrier observable in metres is written as,

$$\Phi = \rho + c(dt - dT) - d_{ion} + d_{trop} + d\rho + e_\Phi + \lambda N \tag{3.3}$$

where,

Φ is the carrier observation (m)

e_Φ is the carrier phase measurement noise including multipath (m)

λ is the carrier wavelength (m)

N is the unknown integer ambiguity

The integer ambiguity term remains constant while the GPS receiver Phase Lock Loop (PLL) is maintained. Loss of lock may occur due to atmospheric effects, multipath, high vehicle dynamics or physical signal obstructions such as bridges and buildings. A loss of lock results in a reinitialisation of the integer ambiguity and is termed a cycle slip. Examples of cycle slip detection and correction algorithms using GPS observables can be found in Roberts (1997) and Bisnath (2000).

3.3.3 Doppler Observation

The Doppler observation is the instantaneous rate of change of carrier phase which is determined by the frequency change of the carrier signal. This frequency shift is caused by relative motion of the receiver and satellite. The range rate between satellite, s, and receiver, r is computed using,

$$\dot{\Phi} = \frac{(x_s - x_r)(\dot{x}_s - \dot{x}_r) + (y_s - y_r)(\dot{y}_s - \dot{y}_r) + (z_s - z_r)(\dot{z}_s - \dot{z}_r)}{\sqrt{(x_s - x_r)^2 + (y_s - y_r)^2 + (z_s - z_r)^2}} \tag{3.4}$$

The Doppler observation is obtained from the PLL in different ways depending on the receiver technology. For example the Ashtech G12 averages the PLL using phase measurements from the previous half second (Cannon et al., 1997). This results in a measurement with a quarter second time offset which will introduce a timing error into the observation. Consequently techniques have been constructed to obtain Doppler measurements by post processing carrier observations using central difference

approximations, curve fitting, or frequency based techniques. For more details see Szarmes et al. (1997) and Bruton (2000).

The error equation for the Doppler observation is expressed in terms of the rate of change of the error sources in Equation 3.3. Consequently, the integer ambiguity term is removed which gives,

$$\dot{\Phi} = \dot{\rho} + c(d\dot{t} - d\dot{T}) - \dot{d}_{ion} + \dot{d}_{trop} + d\dot{\rho} + e_{\dot{\Phi}} \tag{3.5}$$

3.4 Error sources

The following sections examine the error sources that corrupt GPS observations along with the mitigation techniques that can be implemented.

3.4.1 Satellite errors

3.4.1.1 Satellite clock errors

Although GPS satellites use high quality caesium or rubidium atomic oscillators, there still remain unavoidable time drifts from GPS time (Rizos, 1999). The control segment monitors each satellite's drift and updates the navigation message with three polynomial coefficients which can be used to reduce the clock error. Since each observation made to each satellite is contaminated by the same clock error, the drift can be completely removed by differencing observations (§3.5).

3.4.1.2 Satellite Orbit

The satellite ephemeris information is provided from either the broadcast ephemeris in the satellite message or from a precise ephemeris. The

broadcast ephemeris is accurate to approximately 5 metres (Hofmann-Wellenhof et al., 1997). Precise orbits are available from agencies such as the International GPS Service (IGS) on the world wide web. The IGS provide three levels of accuracy: predicted, rapid and final. The predicted orbits are available in real time and are accurate to approximately 25cm (IGS, 2002). The rapid and final orbits are available with a latency of 17 hours and 13 days respectively, and are accurate to 5cm for the rapid orbits and better than 5cm for the final orbits (ibid.).

The satellite position is considered fixed when solving for receiver position resulting in direct propagation of satellite position error into the receiver position. The satellite position error is largely removed through differencing observations for short baselines. The errors are elevation and azimuth dependent and increase for longer baselines. Further details can be found in Evans (2001).

3.4.2 Propagation errors

3.4.2.1 Ionosphere

The ionosphere is a dispersive medium with respect to the GPS microwave signal (Hofmann-Wellenhof et al., 1997). This results in a frequency dependent non-linear dispersion of the GPS signal as it passes through the ionosphere. The ionosphere results in a delayed code range and an advanced carrier phase which is reflected in Equations 3.2 and 3.3.

The magnitude of the ionospheric delay is dependent on the Total Electron Content (TEC) along the signal path. The temporal TEC varies over a number of well known cycles including the 11 year sun spot cycle,

a seasonal cycle and a diurnal cycle. Many models exist to attempt to remove the ionospheric delay such as the Klobuchar model which is a function of time and latitude. Klobuchar coefficients are transmitted in the navigation message and accommodate approximately 50% of the total ionospheric effect at mid-latitudes (Farrell and Barth, 1999).

Dual frequency observations can be used to form ionospherically free combinations of observables due to the frequency dependent dispersion of the GPS signal. For more details of ionospherically free combinations of measurements, the reader is referred to Chao (1996).

3.4.2.2 Troposphere

The troposphere is non-dispersive at radio frequencies resulting in equal delay errors for L1 and L2 carrier and code signals. This means that dual frequency linear combinations of measurements cannot be used to remove the tropospheric delay error. The troposphere is composed of a wet component and a dry component.

The dry component is relatively well modelled and accounts for approximately 90% of the total tropospheric delay (Hofmann-Wellenhof et al., 1997). The temperature, pressure and humidity along the propagation path is required to accurately model the delay but such measurements are not always easily obtainable. Standard atmospheric models can be used such as the Magnet model which uses station latitude, height and Julian day. Surface pressure and temperature measurements can bring the residual error down to within 2-5% of the total dry delay (Grewal et al., 2001).

The wet component is more difficult to model because of spatial variations in the water vapour content. For short baselines, it is considered to

be removed by differencing measurements.

3.4.2.3 Multipath

Multipath is the effect caused by GPS satellite signals reaching the GPS antenna via one or more indirect path. This is caused by objects near the antenna such as buildings or the surrounding ground reflecting the satellite signals. This results in a distorted correlation of the received signal measurement with the receiver generated signal.

Several multipath reduction methods exist to utilise the effect of signal propagation geometry (Weill, 1997). For static GPS receivers, multipath error exhibits periodic behaviour due to the repeat period of the satellite orbit. This can allow multipath effects to be reduced with time series analysis for long occupation periods. For kinematic applications, multipath effects are more random due to the change in the geometry of surrounding reflective surfaces. Special antennas such as ground-plane or choke ring antennas can be used to shield the antenna from signals arriving beneath the antenna. Careful placing of the GPS antenna away from reflective surfaces may also be possible in static and kinematic environments.

Recent advances in GPS receiver correlator technology use real-time signal processing to mitigate multipath (Weill, 1997). However, these techniques primarily relate to GPS code observations (ibid.).

3.4.3 Receiver errors

3.4.3.1 Receiver clock errors

GPS receivers use relatively inexpensive quartz crystal oscillators which are less stable than satellite clocks. Typically, the clock bias is treated as an additional unknown in the estimation procedure. Therefore observations are required to at least four satellites. However, the offset between receiver clock time and GPS time equally contaminates all measurements made by a single receiver. Therefore, by differencing measurements between satellites, the receiver clock bias can be completely removed.

3.4.3.2 Receiver Measurement Noise

Receiver Measurement Noise is caused by the passing of the GPS signal through the electrical components of the receiver, antenna and cables. One method to determine the level of GPS receiver noise is to perform a zero-baseline test where two receivers are connected to a single antenna using a splitter. Differencing measurements between receivers results in the combined receiver noise for a receiver pair. For the Ashtech Z-XII receivers used in this thesis, Bona and Tiberius (2000) found the standard deviation to be 0.2mm for the undifferenced L1 carrier noise.

The zero-baseline method eliminates preamplifier, sky, and ground noises which dominate the receiver system noise (Langley, 1997). Short (10m) baseline tests were also performed by Bona and Tiberius to provide an estimate of the system measurement noise. For the same Ashtech Z-XII receiver, the system measurement noise of 0.5mm was obtained for the L1 frequency (ibid.). Estimation of the measurement noise error is required

for optimal filtering with other navigation systems. This is considered in §3.6.

3.4.3.3 Phase centre variation

Measurements obtained by a GPS receiver are taken to the electrical Antenna Phase Centre (APC). The APC is vertically and horizontally offset from the reference point of the antenna and is elevation and frequency dependent. APC errors are reduced by differencing measurements taken using the same antenna types which are aligned in the same direction over short baselines. APC models provide frequency dependent offsets for different elevation angles which can be used to reduce APC errors when using, for example, different antenna types or longer baselines.

3.5 Differencing GPS observations

Differencing GPS observations is used in order to reduce or remove some of the errors described in the previous sections. Differencing observations works on the principle that GPS errors are highly correlated over short baselines. A static base receiver is located at a known coordinate and observes measurements from the same satellites as a second, roving receiver. This method is known as Differential GPS (DGPS).

3.5.1 Single difference

The single difference observable is calculated by differencing observations between either receivers or satellites. Differencing observations between satellites removes the receiver clock bias whereas differencing between re-

ceivers removes the satellite bias and reduces the satellite orbit error.

Atmospheric propagation effects are reduced through differencing given the assumption that the differenced signals travel through a similar part of the atmosphere. This means that an increase in the distance between the roving receiver and the base receiver results in an increase in position error.

Differencing observations results in an increase in measurement noise due to combining the observations. Differencing does not remove multipath since multipath is both site and satellite dependent. Furthermore, differencing measurements between receivers results in additional economic and logistical factors.

3.5.2 Double difference

The double difference is formed by differencing observations between receivers and satellites. The double difference observation is convenient as both the satellite and receiver clock offsets are removed in addition to the reduction of atmospheric propagation errors. The double difference does however result in an increase in measurement noise compared to the single differenced observation.

The double difference error equations for pseudorange, carrier and Doppler observations are written respectively as,

$$\Delta\nabla p = \Delta\nabla\rho + \Delta\nabla d_{ion} + \Delta\nabla d_{trop} + \Delta\nabla d\rho + e_{\Delta\nabla\rho} \quad (3.6)$$

$$\Delta\nabla\Phi = \Delta\nabla\rho - \Delta\nabla d_{ion} + \Delta\nabla d_{trop} + \Delta\nabla d\rho + e_{\Delta\nabla\Phi} + \Delta\nabla\lambda N \quad (3.7)$$

$$\Delta\nabla\dot{\Phi} = \Delta\nabla\dot{\rho} - \Delta\nabla\dot{d}_{ion} + \Delta\nabla\dot{d}_{trop} + \Delta\nabla d\dot{\rho} + e_{\Delta\nabla\dot{\Phi}} \quad (3.8)$$

where $\Delta\nabla$ is the double difference operator and $\Delta\nabla\lambda N$ is the double

difference ambiguity term.

For carrier measurements the (differenced) ambiguity term still remains for both the single and double difference observations. The ambiguity term can be removed by further differencing the measurements between epochs resulting in the triple difference observation. The double difference observation is used extensively in this thesis for processing GPS measurements.

3.6 GPS Measurement Noise Estimation

This section considers the problem of estimating the stochastic properties of the GPS observations, i.e. constructing a fully populated variance-covariance matrix. The stochastic model is required for optimal filtering of GPS measurements and needs to take into consideration the inherent observation correlations (Kim and Langley, 2001).

Temporal and spatial observation correlations caused by factors such as signal propagation errors and orbit errors are reduced through differencing measurements. Unmodelled temporal and spatial correlations will therefore remain in the observations. Algebraic correlations introduced through differencing the measurements are considered in the following section. Techniques to estimate the random observation errors due to factors such as measurement noise and temporally and spatially uncorrelated multipath are considered in §3.6.2.

3.6.1 Algebraic Correlations

The measurement noise matrix is often considered to be a diagonal matrix with the off-diagonal elements (correlations) set to zero (Wang et al., 1999).

The differenced observations used in GPS are however inherently correlated due to measurements being formed using the same GPS observations.

The software used in this thesis (see §4.5) uses double difference observations in the Kalman filter. The double difference observations are formed by differencing the single difference observations to a satellite, j, with the single difference observations of a reference satellite, j_0. The criteria for selecting the reference satellite is based on the highest satellite elevation angle and whether a satellite has experienced cycle slips. The single difference observations are formed for each satellite by differencing the ranges for each satellite between the base and roving receiver. Because each range is measured independently, the single difference observations are assumed to be uncorrelated, with the associated matrix of measurement errors being diagonal with the diagonal elements formed from the sum of the variances for each range.

To account for the algebraic correlations in the double difference measurement noise matrix, the following equation is considered which converts the single difference observations to double differences,

$$\Delta\nabla\Phi = J.\Delta\Phi \tag{3.9}$$

$$= \begin{bmatrix} 1 & 0 & \dots & 0 & -1 \\ 0 & 1 & \dots & 0 & -1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & 1 & -1 \end{bmatrix} \begin{bmatrix} \Delta\Phi_1 \\ \Delta\Phi_2 \\ \vdots \\ \Delta\Phi_{j-1} \\ \Delta\Phi_{j0} \end{bmatrix} \tag{3.10}$$

where $\Delta\Phi_n$ is the nth single difference observation between receivers for j observations and $\Delta\Phi_{j_0}$ is the single difference observation to the reference satellite.

Gauss' propagation of errors law is applied to the single difference covariance matrix using the relationship in Equation 3.9 to form the double difference measurement noise matrix,

$$R_k = \begin{bmatrix} \sigma^2_{\Delta\Phi_1} + \sigma^2_{\Delta\Phi_{j0}} & \sigma^2_{\Delta\Phi_{j0}} & \cdots & \sigma^2_{\Delta\Phi_{j0}} \\ \sigma^2_{\Delta\Phi_{j0}} & \sigma^2_{\Delta\Phi_2} + \sigma^2_{\Delta\Phi_{j0}} & \cdots & \sigma^2_{\Delta\Phi_{j0}} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma^2_{\Delta\Phi_{j0}} & \sigma^2_{\Delta\Phi_{j0}} & \cdots & \sigma^2_{\Delta\Phi_{j-1}} + \sigma^2_{\Delta\Phi_{j0}} \end{bmatrix} \tag{3.11}$$

where $\sigma^2_{\Delta\Phi_j}$ is the variance of the jth single difference observation. The variance of the single difference is the sum of the variances of the two independent range measurements. Therefore, if the measurement errors for each range are considered equal, the above matrix simplifies to a matrix with 4's on the diagonal and 2's on the off-diagonals, multiplied by the measurement variance.

Equation 3.11 accounts for the algebraic correlations that exist between measurements but other unmodelled spatial and temporal correlations may exist that are not accounted for (Wang et al., 1999). Spatial and temporal correlations may occur because of, for example, errors in satellite orbits, antenna phase centre or tropospheric errors (Schwieger, 2001; Borre and Tiberius, 2000).

3.6.2 Current Estimation Methods

As considered in the previous section, one approach for estimating the measurement noise is to assume that each of the observations have equal weight. This assumption can result in noisy satellite measurements corrupting the state estimates. Two main techniques for correctly weighting the measurement noise have been developed to date (Collins and Langley,

1999). These are elevation dependent models and models that use the signal to noise ratio output from the GPS receiver. These are discussed in the following sections.

3.6.2.1 Satellite Elevation Angle

One form of weighting the observations is to use the satellite elevation angle as an approximation to measurement noise as in, for example Satirapod and Wang (2000) and Collins and Langley (1999). One equation to predict the measurement error using the satellite elevation is to use an exponential function such as in Xin-Xiang (1996),

$$y = a_0 + a_1 e^{\frac{-x}{x_0}} \tag{3.12}$$

where y is the carrier measurement noise and a_0, a_1 and x_0 are empirically derived constants for different receiver and antenna configurations.

While empirical results indicate that elevation angle can be used to estimate measurement noise, it is the impact of the possible error sources that increases for lower elevation satellites (Wieser, 2001). For example, multipath is likely to increase with lower elevation, as will the noise caused by atmospheric propagation. Therefore the elevation angle dependence varies with the particular kinematic situation (Kim and Langley, 2001). Furthermore, elevation dependence caused by the antenna gain pattern will vary due to the physical orientation of the antenna (ibid.).

An alternative solution for reducing the effect of elevation dependence is to use a higher elevation cut-off angle so that noisier observations are not used. This may not be desirable because it reduces the number of observations available which consequently reduces the geometrical strength

of the GPS observations.

3.6.2.2 Signal-to-noise Ratio

Signal to noise ratio (SNR) has been identified as a potentially useful estimate of measurement noise in, for example, Collins and Langley (1999), Satirapod and Wang (2000) and Hartinger and Brunner (1998). SNR is often represented by the carrier to noise power density ratio C/N_0 with units in dB-Hz. SNR data is available from most GPS receivers, although conversion factors from SNR to C/N_0 are not always readily available from GPS receiver manufacturers. SNR is also used in algorithms that attempt to estimate the signal multipath, for example Comp and Axelrad (1998).

The following equation is given in Langley (1997) to estimate the variance, $\sigma^2_{L1,L2}$ of each satellite range measurement using the C/N_0,

$$\sigma^2_{L1,L2} = S_{L1,L2}.10^{\frac{-C/N_0}{10}} \tag{3.13}$$

where $S_{L1,L2}$ consists of the carrier loop noise bandwidth for L1 or L2, and a conversion term from cycles^2 to m^2.

The C/N_0 model determines how well the GPS receiver tracking loops can track the signal which largely reflects the error in the GPS observation (Langley, 1997). The C/N_0 model is therefore potentially more applicable to modelling the measurement error because it uses direct estimates of observation noise from the GPS receiver.

Chapter 4

GPS and INS integration using Kalman Filtering

4.1 Introduction

The Kalman filter is the commonly used method for integrating GPS and INS measurements, for example in Howell and Tang (1994), Schmidt (1989) and Wolf et al. (1997). Other integration algorithms such as neural networks can be used but are not considered further in this thesis (See, for example, Chiang et al. (2003)). This chapter describes the fundamental principles of the Conventional Kalman Filter (CKF) and the way in which it is used for integrating GPS and INS measurements.

Adaptive Kalman filtering techniques are introduced in Section 4.3 to describe the methods which can be used to adapt the process and measurement noise information used in the filter. The adaptive techniques of Covariance Scaling, Adaptive Kalman Filtering and Multiple Model Adaptive Estimation are described.

Having examined conventional and adaptive Kalman filtering algorithms, the way in which the Kalman filter algorithm can be implemented for GPS and INS integration is considered. Here the centralised and decentralised filter structures are examined and consideration is given to factors such as aiding ambiguity resolution and cycle slip detection. Finally, the development of the integration software used in this thesis is described.

4.2 Kalman filtering

The Kalman filter was first introduced by Kalman (1960). The Kalman filter is a recursive algorithm which filters measurements using knowledge of the system dynamics and the statistical properties of the system measurement errors. The filter uses the measurements to compute the minimum error estimate of a given number of states which are linearly related to the measurements. The states are composed of a set of quantities that are sufficient to model the motion of a system (Tsakiri, 1995). An integrated GPS/INS filter typically estimates the error of nine navigation states, three each for attitude, velocity and position. Sometimes four states are modelled for attitude error when using the quaternion representation. The states modelled for the software used in this thesis are,

$$x = [\ \partial\lambda \quad \partial\phi \quad \partial h \quad \partial v_N \quad \partial v_E \quad \partial v_D \quad \partial\varphi \quad \partial\theta \quad \partial\psi\ ]^T \tag{4.1}$$

Additional states can be modelled in the Kalman filter to compensate for sensor specific errors. For a GPS/INS system, additional error states such as sensor bias, scale factor, temperature dependent bias and axis non-linearity can be included in the estimation process.

The Kalman filter is well suited to navigation applications which contain states that are time-variant. Often redundant measurements are available, and the Kalman filter appropriately weights the measurements according to the stochastic information provided. Another benefit of the Kalman filter is its ability to operate in real-time. The full derivation of the Kalman filter algorithm and further information on Kalman filtering is provided in texts such as Brown and Hwang (1997), Cross (1994), Grewal and Andrews (1993), Gelb (1974) and Maybeck (1979).

4.2.1 Dynamic Model

The dynamic model describes the change in the parameters of the state vector with respect to time. The continuous time representation of the dynamic model is given as,

$$\dot{x} = Fx + Gu \tag{4.2}$$

where,

x	is the system state vector
F	is a dynamics matrix
G	maps the disturbing forces to the states
u	is the vector of disturbing forces

The discrete form of Equation 4.2 is given as,

$$x_{k+1} = \Phi x_k + w_k \tag{4.3}$$

where Φ is termed the state transition matrix which relates the state vector from the epoch k to $k+1$. The system noise, w_k, is the vector of disturbing

forces on the system. This is assumed to be a zero mean and uncorrelated sequence. The covariance of the system noise is defined as,

$$E[w_i w_j^T] = \begin{cases} Q_k & i = j \\ 0 & i \neq j \end{cases} \tag{4.4}$$

The matrix Q_k is termed the process noise matrix which in the GPS/INS integration filter represents the estimates of the INS errors. The process noise is approximated using the approximation (Shin, 2001),

$$Q_k \approx GQG^T \Delta t \tag{4.5}$$

where $\Delta t = t_{k+1} - t_k$ is the sampling interval and Q is the spectral density matrix. Q is a diagonal matrix formed from the standard deviations of the driving noise of the system (i.e. the accelerometer and gyro measurements).

The dynamic model, F, is formed using the error dynamics of the inertial system (§2.6.1), and the state transition matrix is formed from the dynamic model using the approximation,

$$\Phi \approx e^{F\Delta t} \tag{4.6}$$

$$= I + F\Delta t + \frac{(F\Delta t)^2}{2!} + \ldots \tag{4.7}$$

4.2.2 Measurement Model

The measurement model in the Kalman filter describes the geometrical or physical relationship of the measurements to the filter states through the linear model. The discrete measurement model at the epoch k is described as,

$$z_k = H_k x_k + v_k \tag{4.8}$$

where,

z_k	is the vector of measurements at the epoch k
H_k	is the measurement model coefficient matrix (often termed the design matrix) which defines the linear relationship between the states and the measurements
v_k	is the measurement residual vector

The measurement noise is assumed to be a zero mean, uncorrelated random sequence with covariance,

$$E[v_i v_j^T] = \begin{cases} R_k & i = j \\ 0 & i \neq j \end{cases} \tag{4.9}$$

For the GPS/INS integration filter, the measurement noise is formed using the estimates of the GPS errors using a model such as those considered in §3.6. A further assumption for the Kalman filter is that the process and measurement noise are unrelated,

$$E[w_i v_j^T] = 0 \ \forall \ i, j \tag{4.10}$$

4.2.2.1 Linearisation and the Extended Kalman Filter

The Kalman filter is a linear estimation technique. It is observed from Equation 3.1 that the raw GPS observations are non-linear with respect to the required states, therefore the measurements need to be linearised. This can be achieved in two ways. Firstly, the measurements can be linearised about a nominal trajectory which does not depend on the measurement data, this is termed *linearisation* (Brown and Hwang, 1997; Farrell and Barth, 1999). Alternatively, the measurements can be linearised about the predicted trajectory from the Kalman filter in what is termed an *extended Kalman filter* (ibid.). In order for the linear assumption to remain valid and

the filter to remain stable, the difference between the predicted estimate and the measurement must remain small. For a low cost INS, the extended Kalman filter implementation is used to reduce the build up of the inertial errors which ensures that the difference between the estimated trajectory from the INS and the GPS does remain small.

The measurement equation for the double difference GPS observations is formed using,

$$z_k = \begin{pmatrix} \Delta\nabla\rho_{observed} - \Delta\nabla\rho_{predicted} \\ \Delta\nabla\dot{\rho}_{observed} - \Delta\nabla\dot{\rho}_{predicted} \end{pmatrix} \tag{4.11}$$

The measurement model is formed from the partial derivatives of the range measurements to each satellite with respect to the navigation states. For the states defined by Equation 4.1, the design matrix relating n double difference range and range rate observations takes the form,

$$H = \begin{pmatrix} \frac{\partial\Delta\nabla\rho_1}{\partial X} & \frac{\partial\Delta\nabla\rho_1}{\partial Y} & \frac{\partial\Delta\nabla\rho_1}{\partial Z} & 0 & 0 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial\Delta\nabla\rho_n}{\partial X} & \frac{\partial\Delta\nabla\rho_n}{\partial Y} & \frac{\partial\Delta\nabla\rho_n}{\partial Z} & 0 & 0 & 0 & 0 & \dots & 0 \\ \frac{\partial\Delta\nabla\dot{\rho}_1}{\partial X} & \frac{\partial\Delta\nabla\dot{\rho}_1}{\partial Y} & \frac{\partial\Delta\nabla\dot{\rho}_1}{\partial Z} & \frac{\partial\Delta\nabla\dot{\rho}_1}{\partial \dot{X}} & \frac{\partial\Delta\nabla\dot{\rho}_1}{\partial \dot{Y}} & \frac{\partial\Delta\nabla\dot{\rho}_1}{\partial \dot{Z}} & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial\Delta\nabla\dot{\rho}_n}{\partial X} & \frac{\partial\Delta\nabla\dot{\rho}_n}{\partial Y} & \frac{\partial\Delta\nabla\dot{\rho}_n}{\partial Z} & \frac{\partial\Delta\nabla\dot{\rho}_n}{\partial \dot{X}} & \frac{\partial\Delta\nabla\dot{\rho}_n}{\partial \dot{Y}} & \frac{\partial\Delta\nabla\dot{\rho}_n}{\partial \dot{Z}} & 0 & \dots & 0 \end{pmatrix} \tag{4.12}$$

where the coordinate partials are formed by,

$$\frac{\partial\Delta\nabla\rho}{\partial x} = \frac{\partial\Delta\nabla\dot{\rho}_i}{\partial\dot{x}} = -\frac{x_s^i - x}{\rho_i} + \frac{x_s^b - x}{\rho_b} \tag{4.13}$$

and,

$$\frac{\partial\Delta\nabla\dot{\rho}}{\partial x} = -\frac{\dot{x}_s^i - \dot{x}}{\rho_i} + \frac{\dot{x}_s^b - \dot{x}}{\rho_b} + \tag{4.14}$$

$$\frac{x_s^i - x}{\rho_i^3}[(x_s^i - x)(\dot{x}_s^i - \dot{x}) + (y_s^i - y)(\dot{y}_s^i - \dot{y}) + (z_s^i - z)(\dot{z}_s^i - \dot{z})] -$$
$$\frac{x_s^r - x}{\rho_r^3}[(x_s^b - x)(\dot{x}_s^b - \dot{x}) + (y_s^b - y)(\dot{y}_s^b - \dot{y}) + (z_s^b - z)(\dot{z}_s^b - \dot{z})]$$

where x_s^i and x_s^b are the coordinates of the satellite i and reference satellite b respectively.

4.2.3 Kalman filter algorithm

The Kalman filter is an iterative filtering procedure which can be divided into two steps of prediction and measurement update. In order to begin the iterative process the filter needs to be initialised.

4.2.3.1 Initialisation

The Kalman filtering algorithm begins with estimating the initial state estimate $\hat{x}_0^{(-)}$ and the corresponding covariance matrix $P_0^{(-)}$. The initial estimate for the states is provided by an *a priori* estimate of the state which has an associated covariance of,

$$P_0 = E[(x_0 - \hat{x}_0)(x_0 - \hat{x}_0)^T] \tag{4.15}$$

where x_0 is the true value. P_0 is usually formed using an estimate of the variance with the initial correlation between the filter states considered to be zero (Chen, 1992).

4.2.3.2 Prediction

The first step of the Kalman filter iteration is to predict the estimate of the state $\hat{x}_k^{(-)}$ based on the previous best estimate of the state from the previous epoch, $\hat{x}_{k-1}^{(+)}$. This is achieved through using the state transition

matrix Φ,

$$\hat{x}_k^{(-)} = \Phi \hat{x}_{k-1}^{(+)} \tag{4.16}$$

The corresponding predicted covariance matrix defined as $P_k^{(-)} = E[x_k^{(-)} x_k^{(-)T}]$, is also propagated to the next epoch using the equation,

$$P_k^{(-)} = \Phi P_{k-1}^{(+)} \Phi^T + Q_{k-1} \tag{4.17}$$

For the integration software in this thesis, the state prediction (Equation 4.16) is not required. This is because the INS is updated using the Kalman filter error state estimate from the previous epoch in a closed loop, (§4.4.1). Therefore, the predicted state at each epoch is zero for all states. The Kalman filter prediction stage is only used to predict the second moment (covariance) information of the inertial error.

4.2.3.3 Measurement Update

The measurement update phase of the Kalman filter algorithm combines the measurement, z_k, with the predicted estimate of the state. The innovation vector is introduced to the filter as the difference between the measurements and the predicted estimate,

$$v_k^{(-)} = z_k - H_k \hat{x}_k^{(-)} \tag{4.18}$$

The innovation sequence is assumed to be a Gaussian white noise sequence and is used as the basis for the adaptive filtering techniques considered in §4.3. The Kalman gain is the weight between the new innovation and the predicted state and is formed using the equation,

$$K_k = P_k^{(-)} H_k^T (H_k P_k^{(-)} H_k^T + R_k)^{-1} \tag{4.19}$$

The state estimate is then updated using the innovations weighted by the filter gain to give the measurement update filtered estimate,

$$\hat{x}_k^{(+)} = \hat{x}_k^{(-)} + K_k v_k^{(-)} \tag{4.20}$$

The corresponding measurement update covariance matrix is then computed,

$$P_k^{(+)} = (I - K_k H_k) P_k^{(-)} \tag{4.21}$$

Numerical stability problems can occur when using this equation due to finite computer word length. The U-D factorisation method is a mathematically equivalent alternative to the Kalman filter algorithm which has improved numerical behaviour (see Brown and Hwang (1997) or Chen (1992)). A simpler method to improve the numerical stability is to use the Joseph form of the covariance update which has natural symmetry given by,

$$P_k^{(+)} = (I - K_k H_k) P_k^{(-)} (I - K_k H_k)^T + K_k R_k K_k^T \tag{4.22}$$

The Joseph form of the covariance update is used in this thesis.

4.2.3.4 Observability

Observability is the ability to determine a state from a given sequence of measurements. For all of the states to be observable, the matrix,

$$\Xi = \left[\; H^T \;\middle|\; \Phi^T H^T \;\middle|\; (\Phi^T)^2 H^T \;\middle|\; \ldots \;\middle|\; (\Phi^T)^{n-1} H^T \; \right] \tag{4.23}$$

must be of full rank (Gelb, 1974). In, for example, Ford et al. (2001), it is demonstated that for a static INS with unknown gyro bias errors, not all of the states are observable using position updates while the state transition

matrix is constant. If the system moves (hence changing the elements of the state transition matrix), all the states become observable to some degree (ibid.). Therefore, it is important to consider that it is not always possible to determine the states from a given set of measurements.

4.3 Adaptive Kalman filtering methods

One of the assumptions required for the Kalman filter to provide the optimal estimate of the state is that the dynamic model is effectively modelled and that the *a priori* covariance matrices for process and measurement noise contain the correct statistics (Wang et al., 1999). In reality, the measurement noise is not constant due to GPS errors sources such as atmospheric conditions, satellite elevation angle and satellite geometry. Similarly the model weighted by the process noise matrix is likely to vary according to inertial error sources such as changing vehicle dynamics, numerical integration errors and sensor errors.

A similar problem for static systems is experienced in least squares processing. For least squares it is assumed that all variance terms are known *a priori* which is not always the case. Variance Component Estimation (VCE) is an iterative numerical procedure that is used to improve the quality of the variance estimates by applying a scale factor to the errors of each observation type. More information on least squares estimation can be found in Cross (1994) and Rao (1965) and details of VCE can be found in Caspary (1987).

In the following sections, three algorithms are identified for adapting the measurement and process noise matrices used in the Kalman filter.

Collectively these are termed *adaptive Kalman filtering* algorithms. The three algorithms described are covariance prediction, adaptive Kalman filtering and Multiple Model Adaptive Estimation (MMAE). Therefore the distinction should be made between adaptive Kalman filtering (which is a generic term for a Kalman filter where the stochastic information is automatically adapted in the filter) and the adaptive Kalman filter (which is one particular algorithm that achieves this).

4.3.1 Innovation and residual sequences

The innovations and residuals, $v_k^{(-)}$ and $v_k^{(+)}$, respectively, are the key to adaptive estimation in all the adaptive techniques considered in this thesis. The innovation is the difference between the new measurement and the predicted state (Equation 4.18), whereas the residual is the difference between the measurement and the corrected state provided by the Kalman filter,

$$v_k^{(-)} = z_k - H_k \hat{x}_k^{(+)} \tag{4.24}$$

The innovations and residuals therefore contain information about the dynamic and measurement model and the measurement and process noise.

The adaptive Kalman filtering algorithms described in the following sections essentially work on the principle that if a innovation or residual value is small, the filter is working well. Conversely, if the value is large, there is a deficiency in the dynamic model, the measurement model, or the *a priori* stochastic information used in the filter. The well established inertial error model is considered to be sufficiently known (Mohamed, 1999; Wang et al., 1999). Therefore, the innovation or residual sequence is

used for adapting the weight between the measurements and the states (covariance scaling), directly estimating the measurement or process noise matrices (adaptive Kalman filtering), or to identify the correct *a priori* stochastic model (multiple model adaptive estimation). These algorithms are considered in the following sections.

The innovation sequence is assumed to be Gaussian and white (Mehra, 1970). Although the innovation sequence contains information obtained from the previous states, the innovation sequence still provides the most relevant source of information for filter adaptation (Mohamed, 1999). It is also important that blunders (for example cycle slips) are removed from the measurements in order that the error is not absorbed into the model.

4.3.2 Covariance prediction

The simplest way to adapt the weight between the measurements and the system prediction is to apply a scale factor, S_k, to the predicted covariance matrix as given in Hu et al. (2001),

$$P_k^{(-)} = S_k(\Phi P_{k-1}^{(+)}\Phi^T + Q_{k-1}) \tag{4.25}$$

Applying a scale factor $S_k > 1$ will result in the more weight being given to the measurements in the filter whereas $S_k = 1$ results in a return to conventional Kalman filtering.

The scale factor can be defined empirically, for example, the inertial sensor errors are likely to be larger before dynamic alignment. An empirical scale factor can be used to artificially inflate the predicted covariance to give the measurements a higher weighting during alignment. Hu et al. (2001) provide a method for adapting the scale factor determined by the dynamic

and observation model accuracy for GPS measurements. This is achieved by computing the ratio between the new innovation and the average of a previous window length of innovations. When the current innovation is larger than the average, the ratio is used to scale the predicted covariance in order to give more weight to the measurements.

4.3.3 Adaptive Kalman filter

The adaptive Kalman filter estimates for the measurement and process noise matrices are based on the covariance-matching technique proposed by Mehra (1972). The idea is to make the residuals consistent with their theoretical covariance by altering the process or measurement noise (ibid.).

4.3.3.1 Measurement noise covariance estimation

An approximation to the innovation covariance can be obtained using the previous innovation sequence by averaging across a window of N epochs. The covariance estimate is given as,

$$\hat{C}_{v_k^{(-)}} = E[v_k^{(-)} v_k^{(-)^T}] \approx \frac{1}{N} \sum_{j=k-N+1}^{k} v_j^{(-)} v_j^{(-)^T} \tag{4.26}$$

The Kalman filter covariance of the innovation sequence is given as,

$$C_{v_k^{(-)}} = H_k P_k^{(-)} H_k^T + R_k \tag{4.27}$$

The innovation based adaptive estimate of the measurement noise matrix is formed by rearranging Equation 4.27 and substituting Equation 4.26 to give,

$$\hat{R}_k = \frac{1}{N} \sum_{j=k-N+1}^{k} v_j^{(-)} v_j^{(-)^T} - H_k P_k^{(-)} H_k^T \tag{4.28}$$

Mohamed (1999) provides a thorough derivation of the measurement noise covariance matrix based on the maximum likelihood criterion.

Wang (1999) indicates that the innovation based adaptive estimate is very sensitive to linearisation errors in the Extended Kalman filter. As a result, Wang proposed that the measurement residual be used to adapt the measurement noise. The measurement residual based adaptive estimate of R_k is given as (ibid.),

$$\hat{R}_k = \frac{1}{N} \sum_{j=k-N+1}^{k} v_j^{(+)} v_j^{(+)T} + H_k P_k^{(+)} H_k^T \tag{4.29}$$

Wang also noted that Equation 4.29 is always positive definite as it is the sum of two positive definite matrices whereas Equation 4.28 is not always guaranteed to be positive definite. The residual based estimate requires a minimal amount of extra computation over the innovation based estimate since the term $H_k P_k^{(+)} H_k^T$ is not already computed in the CKF algorithm.

These equations show that it is possible to obtain an estimate for the measurement noise covariance using the approximation to the residual covariance matrix and matrices used in the conventional filtering algorithm. Such an adaptation adds a small amount of extra processing to the CKF algorithm which can be considered negligible in the context of the overall data processing algorithm. Another potential benefit of the adaptive measurement noise estimate is that is provides a full matrix containing an estimate of both variance and covariance information.

The innovation covariance approximation uses an averaging window on the last N epochs of the innovation sequence. It is important to use an appropriate window length in order to provide the optimum weighting between filter adaptivity and stability. Statistically, the sequence contained

within the window is considered stationary, i.e. the statistical properties of the process are constant with time. When the process is stationary for the whole dataset, an infinite length window can be used (see Salychev (1998)).

4.3.3.2 Process noise covariance estimation

An adaptive estimate of the process noise covariance matrix is obtained using the maximum likelihood derivation in Mohamed (1999) to give,

$$\hat{Q}_k = \hat{C}_{\Delta x_k} - (\Phi P_{k-1}^{(+)} \Phi^T - P_k^{(+)}) \tag{4.30}$$

where $\hat{C}_{\Delta x_k}$ is the estimated covariance of the state correction sequence,

$$\Delta x_k = x_k^{(+)} - x_k^{(-)} \tag{4.31}$$

A windowing function on the state correction sequence can be used to form an approximation to the covariance in the same way that the innovation covariance approximation was formed to give,

$$\hat{C}_{\Delta x_k} \approx \frac{1}{N} \sum_{j=k-N+1}^{k} \Delta x_j \Delta x_j{}^T \tag{4.32}$$

Using the substitution $\Delta x_k = K_k v_k^{(-)}$, an approximation to the state correction covariance can also be formed using the innovation sequence by,

$$\hat{C}_{\Delta x_k} \approx K_k \hat{C}_{v_k^{(-)}} K_k^T \tag{4.33}$$

Equation 4.30 can be considered to be constructed of two parts. The first part is the estimated covariance of the state correction. When the estimated covariance is large, it indicates that the predicted state is of low quality. Conversely, if the estimated covariance is small, it indicates that the predicted state is a good estimate of the state at each epoch. The second part of Equation 4.30 is the Kalman filter estimate of the difference

of the *a posteriori* covariances between epochs. When the difference in the *a posteriori* covariances is small, the approximation, $\hat{Q}_k \approx \hat{C}_{\Delta x_k}$, can be made (Mohamed and Schwarz, 1999).

In a similar way to the adaptive measurement noise estimate, these equations use matrices used in the CKF algorithm with a small amount of extra processing required to compute the approximation to the state correction covariance. Due to the approximations taking place in the estimation of $\hat{Q}$, the estimate is expected to be inferior to the estimation of $\hat{R}$ (Mohamed, 1999). In addition, the estimate is not always guaranteed to be positive definite (Wang et al., 1999). An appropriate window length for the covariance approximation needs to be estimated and Equation 4.30 provides a full process noise matrix which attempts to model any correlations that may exist.

It should be noted that, from the derivation of the equation for $\hat{R}_k$, it is assumed that the process noise matrix is known. Similarly, in the derivation of $\hat{Q}_k$, the correct measurement noise matrix is required. Therefore, if there is a deficiency in the stochastic model that is not being estimated, the error will be incorrectly absorbed into the stochastic model that is being estimated. For example, if the GPS measurement errors are being estimated using Equation 4.28, and the inertial errors are larger than *a priori* errors defined in the process noise matrix, the estimated GPS measurement errors will incorrectly increase. Therefore it is important that the stochastic model not being estimated is correct, and it also indicates that the stability of such algorithms is likely to be reduced. The assumption also means that both the $\hat{R}_k$ and $\hat{Q}_k$ matrices should not be estimated simultaneously.

4.3.4 Multiple Model Adaptive Estimation

Multiple Model Adaptive Estimation (MMAE) was first presented by Magill (1965). The multiple model filter processes a bank of Kalman filters each using a different mathematical or stochastic model. The multiple model filter can be used to identify a single correct state estimate or the state estimate from each filter can be combined.

4.3.4.1 Algorithm

The Kalman filter innovation sequence is a white process with covariance given in Equation 4.27. The Probability Density Function (PDF) for each model is an estimate of the likelihood of a particular model being correct and is a function of the innovation sequence and its corresponding covariance. The PDF for the n^{th} Kalman filter is given as,

$$f_n(z_k) = \frac{1}{\sqrt{(2\pi)^m |C_{v_k^{(-)}}|}} e^{-\frac{1}{2} v_k^{(-)} C^{-1}_{v_k^{(-)}} v_k^{(-)^T}} \tag{4.34}$$

where m is the number of observations. By using Bayes law, the probability, $p_n(k)$, that the n^{th} model is correct at the epoch k is calculated using the recursive equation,

$$p_n(k) = \frac{f_n(z_k).p_n(k-1)}{\sum_{j=1}^{N} f_n(z_k).p_j(k)} \tag{4.35}$$

for N possible models.

The processing methodology of MMAE is described in Figure 4.1. Here it is shown that each of the Kalman filters are running in parallel, each using different process noise, measurement noise or state transition matrix. For each individual filter, n, the weight factor $p_n(k)$ is computed. The state estimate from each filter can then be combined using the computed weight

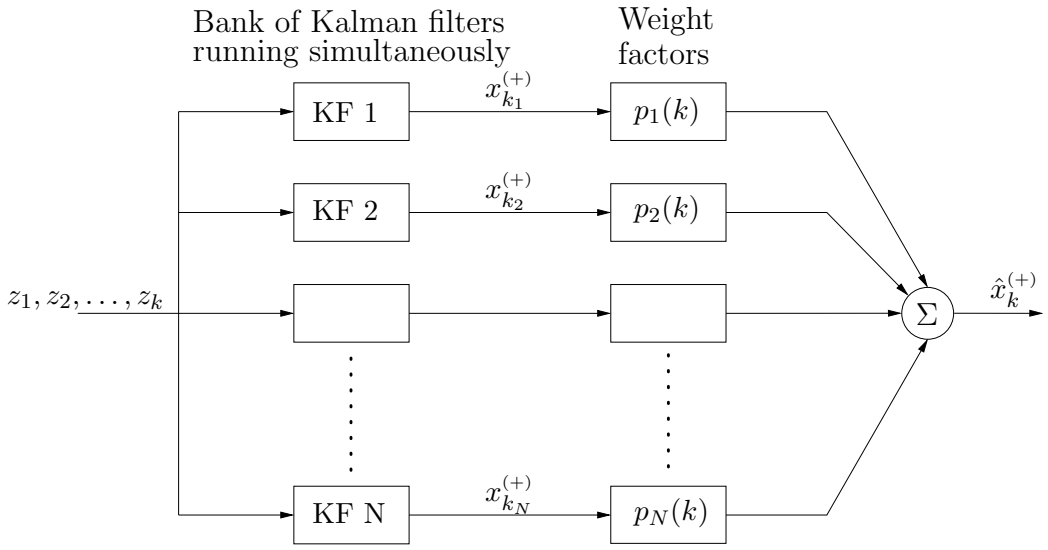


Figure 4.1: Multiple Model Adaptive Estimation (Adapted from Brown and Hwang, 1997)

factors to form the optimal state estimate (Brown and Hwang, 1997),

$$\hat{x}_k^{(+)} = \sum_{j=1}^{N} p_j(k)\hat{x}_{k_j}^{(+)} \tag{4.36}$$

where $\hat{x}_{k_j}^{(+)}$ is the state estimate of the jth Kalman filter. The optimal state estimate covariance is given as,

$$P_k^{(+)} = \sum_{j=1}^{N} p_j(k)\left[P_{k_j}^{(+)} + (\hat{x}_{k_j}^{(+)} - \hat{x}_k^{(+)})(\hat{x}_{k_j}^{(+)} - \hat{x}_k^{(+)})^T\right] \tag{4.37}$$

The process is initialised using the *a priori* probability that a particular model is correct. In most cases the probability is unknown, therefore the probability is initialised by giving each of the models equal weighting.

The MMAE filter can be used in different configurations. The above algorithm describes a filter that will converge to a fixed weighting of each model (Welch and Bishop, 2001). In many instances, it is desirable to force the filter to remain adaptive by using an ad hoc method such as imposing an artificial lower bound on the model probabilities (ibid.). Another method for avoiding model convergence is to directly weight each of the models

according to the probability density function at each epoch using (Welch, 1999),

$$p_n(k) = \frac{f_n(z_k)}{\sum_{j=1}^{N} f_j(z_k)} \tag{4.38}$$

The largest limitation of MMAE is the large computational burden imposed from running simultaneous Kalman filters. However, the continuing advances in computer processor technology mean that it is now feasible to run multiple Kalman filters even for real time navigation systems (see §7.4.4).

To date, applications for MMAE have primarily been for fault detection. For example Hanlon and Maybeck (2000) used multiple Kalman filters to test different failure status models for actuator failure detection. Similarly, MMAE is used in Chen (1992) for fault detection in GPS carrier measurements (i.e. cycle slips).

4.4 Integration architectures

GPS and inertial measurements can be integrated using a Kalman filter in a variety of configurations. This section identifies three levels of integration: decentralised, centralised and full.

4.4.1 Decentralised

The decentralised filter implements two separate Kalman filters and is described in Figure 4.2. Examples of decentralised filters can be found in Sun (1998) and Wolf et al. (1997). The figure divides the integration process into three stages.

1. *Measurement.* Firstly the GPS and IMU data is discretely sampled.

2. *Pre-processing.* The observations for the separate Kalman filters are formed. The GPS filter uses the differenced range and range rate observations. The inertial measurements are mechanised using the algorithms discussed in §2.3.3 to provide position, velocity and attitude measurements.

3. *Filtering.* The filtering stage first processes the GPS observations to provide an estimate of position and velocity. The output is differenced with the INS observations to provide the measurements for the integration filter. The integration filter estimates the error in each of the states and the navigation errors are then fed back to correct the INS. This is termed closed loop filtering and is used to reduce the linearity errors.

The navigation filter measurements are described by,

$$z_k = (\hat{X}_{INS} - \hat{X}_{GPS}) \tag{4.39}$$

$$= (X + \delta X + n_{INS}) - (X + n_{GPS}) \tag{4.40}$$

$$= \delta X + n \tag{4.41}$$

where,

$\hat{X}$	is the estimated navigation parameter from each system
X	is the true navigation parameter
δX	is the inertial navigation parameter error
n	is the random noise

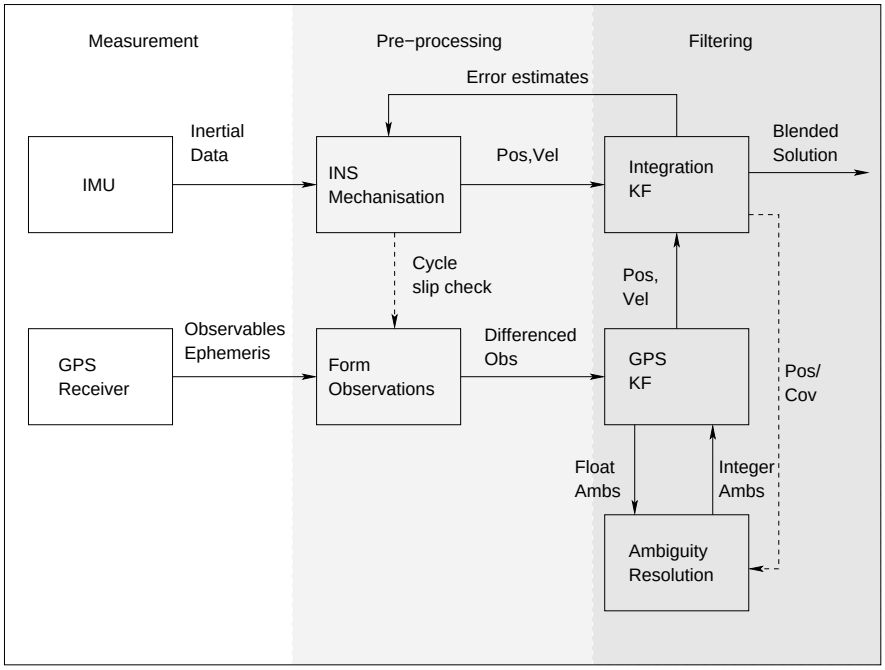


Figure 4.2: Decentralised Filter Structure

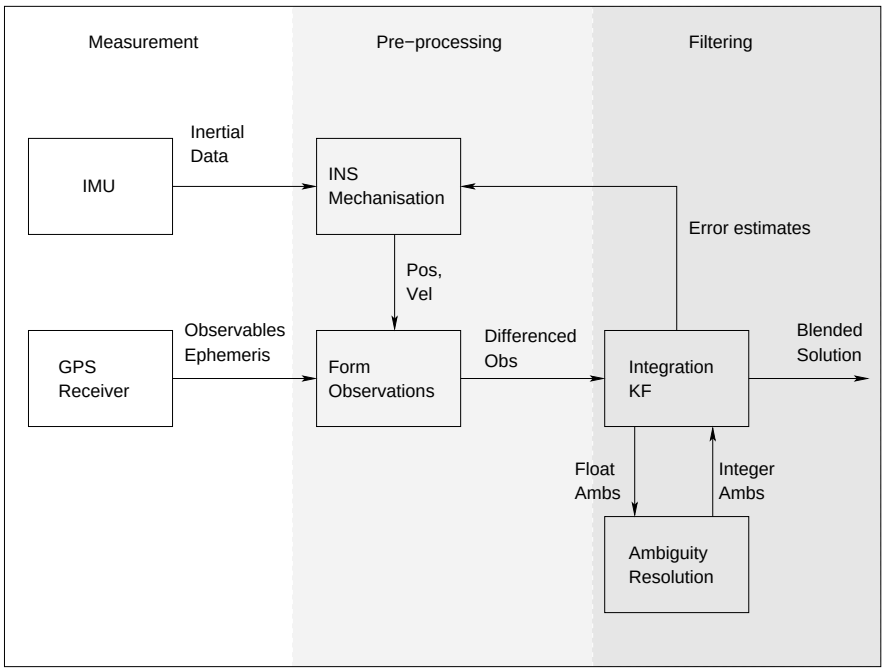


Figure 4.3: Centralised Filter Structure

It should also be noted that the GPS observations are compensated for the lever-arm offset between the origin of the IMU body frame and the GPS phase centre. The previous system of equations shows that the measurement is formed by differencing the GPS and INS navigation parameter which is equivalent to the actual inertial navigation error plus random noise. Using this architecture, the system works as a GPS corrected inertial system.

The two dashed lines represent configurations which form closer coupling of the measurements whilst retaining the decentralised filtering structure. The INS can be used to provide cycle slip detection and correction while the integrated position and covariance information can be used in the ambiguity resolution algorithms. The GPS filter still, however, requires at least four range observations to compute a solution.

The primary advantage of the decentralised filter lies in the ease of implementation. The position and velocity output of a GPS receiver can be used directly in the filter without need to process the raw GPS observations. Moreover, the integration approach is conceptually easier to understand and the computational burden of the integration Kalman filter is minimal. The system is robust in the sense that if one of the systems were to fail, navigation could still be provided by the other sensor.

From a statistical viewpoint, Brown and Hwang (1997) identifies two problems with the decentralised filtering approach. Firstly, the estimation error in the output of the GPS filter is likely to contain time correlations which can be difficult to account for and may contradict the whiteness requirement for the Kalman filter. Secondly, unless the state vector models all of the sub-system errors, measurement information may be lost between

the separate filters.

One method of reducing the measurement correlations is to use the output of the GPS filter less frequently assuming that the error growth of the INS is still small over this period (ibid.). Also, if available, the covariance of the GPS filter can be used to form the measurement noise in the INS filter. Alternatively, the raw measurements can be used in a single filter.

4.4.2 Centralised

The centralised filter architecture is represented by Figure 4.3. Here, the observations are formed using Equation 4.39 where the navigation parameter, X, is substituted for the differenced range and range rate observations. The INS measurements are used to form the equivalent GPS range and range rate observations. The error estimate from the Kalman filter is fed back to correct the INS.

As a consequence of the observations being combined in the single filter, the filter provides the global optimal solution for the system (Brown and Hwang, 1997). The non-linearities in the filter are reduced because the predicted observation is provided using the inertially predicted trajectory. Furthermore, by using the inertial trajectory, GPS observations can still be used in the filter when less than the required 4 satellites are available.

For the centralised filter structure, the inertial position is readily available for cycle slip detection and correction. The time to ambiguity resolution may be reduced due to the improved estimate of receiver position used to form the float solution in the ambiguity search. This improvement will depend on the quality of the inertial sensor and its maintained navigation

performance during GPS outages.

A potential problem is that the centralised filter will fail if the inertial system fails. In the decentralised filter, the navigation solution would still be available from the GPS filter. In practise the filter could be reconfigured to work as a conventional GPS filter to add system robustness. The centralised filter is also more difficult to implement due to the required handling of the raw GPS observations.

4.4.3 Full

The full integration architecture uses the measurements from the inertial system to aid the carrier tracking loops in the GPS receiver. This is sometimes termed deeply coupled filtering. In a conventional receiver, the bandwidth of the carrier tracking loop has to be large enough to maintain lock during large vehicle accelerations (Farrell and Barth, 1999). The acceleration information from the inertial system can be used to reduce the bandwidth of the carrier tracking loop which improves the signal-to-noise ratio of the receiver making it more resistant to interference (ibid.). This implementation is not considered in this thesis because it requires hardware access to the GPS receiver carrier tracking loops which is not available in standard receivers. Examples of using inertial observations to enhance the carrier tracking loop can be found in Kreye et al. (2000) and Gustafson et al. (2000).

4.4.4 Discussion

The previous sections have identified the various configurations in which inertial sensors and GPS can be combined. The terms loosely and tightly coupled are often used instead of the terms decentralised and centralised. The term tightly coupled is used because, when using the centralised filtering structure, the inertial measurements are usually used to aid the processing of the GPS measurements. However, as described in Section 4.4, the decentralised filter can also be used to aid GPS processing. Consequently, in this thesis the terms centralised and decentralised refer to whether the measurements are processed in a single Kalman filter, or two separate Kalman filters. The use of inertial measurements to add robustness to the overall system through providing cycle slip detection and correction, improvement in ambiguity resolution performance and system integrity is termed tight coupling.

4.5 Integration software

The following sections describe the development of the GPS and INS integration software, KinPos^i, which is used throughout this thesis. KinPos^i was developed by the author by combining the IESSG's kinematic GPS software, KinPos, with loosely coupled GPS/INS software also developed at the IESSG.

4.5.1 KinPos

KinPos is the IESSG's post processing kinematic GPS software, originally developed by Dr Wu Chen. KinPos was developed for estimating position

and tropospheric delay at a moving GPS receiver (Dodson et al., 2001). The software is written in FORTRAN and processes L1, L2 or the ionospherically free observable, L0, using double difference observations.

The original GPS only structure of KinPos is described by Figure 4.4. The processing options and filenames are defined in a control file that is read at the initialisation stage. At this point other initial values for processing are defined. KinPos uses SP3 and NOTT2 file formats that are obtained from Rinex format files using the software FILTER and CON2SP3 which are part of the IESSG's GPS Analysis Software (GAS) package. The GPS data is read in and the states are propagated to the next epoch using a choice of random walk, constant velocity or constant acceleration models (see Tsakiri (1995)). The double differenced observed minus predicted observations are then computed. The states are updated with pseudorange and carrier observations sequentially with the updated state from the pseudorange observations forming the position estimate for ambiguity resolution. The updated state estimate is also used as the state prediction for the carrier phase Kalman filter update when the carrier phase ambiguities have been resolved. Doppler observations can be used in both the pseudorange and carrier phase updates.

4.5.1.1 Ambiguity resolution

For ambiguity resolution, the double difference ambiguities are estimated from the real valued double difference float ambiguities using least squares. KinPos uses the Least Squares AMBiguity Decorrelation Adjustment (LAMBDA) from Delft University of Technology, Netherlands (De Jonge and Tiberius, 1996) to search for the integer ambiguities.

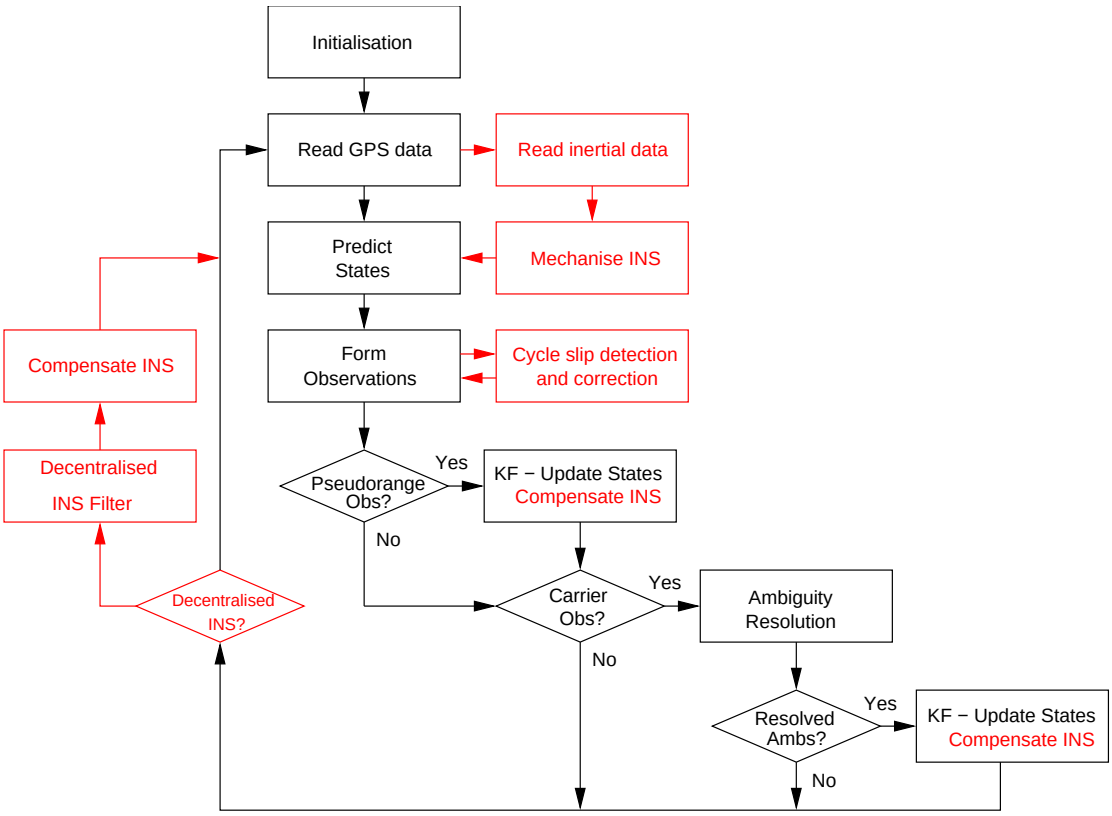


Figure 4.4: KinPos control flow (Red represents KinPos^i modifications)

To estimate the ambiguities, the LAMBDA algorithm first uses a space projection technique to transform the ambiguities and their corresponding covariance to another space that is easier to analyse (Mohamed, 1999). As a result, the satellite to receiver geometry changes and the resulting ambiguities decorrelate (ibid.). In the transformed space, a sequential least squares search is performed which results in a number of possible ambiguity sets. The double differenced float ambiguity is corrupted largely due to residual spatially correlated errors, receiver noise and multipath. Consequently, the norm of the integer ambiguity candidate and the float ambiguity is computed in the transformed space as a measure of the quality of the ambiguity fix, and are used to identify the correct ambiguity set. The prospective ambiguity set is then transformed back to the ambiguity space

using the inverse of the transform.

The double differenced carrier observations are corrected with the estimated ambiguities and are used to update the states in the Kalman filter. A cycle slip, loss of lock, or change of base satellite results in the reinitialisation of the ambiguity resolution process. Further details of the LAMBDA ambiguity resolution algorithm can be found in De Jonge and Tiberius (1996). More information on ambiguity resolution in KinPos can be found in (Pattinson, 2002).

4.5.2 Inertial Software

The inertial software on which KinPos^i is based is a post processed, loosely coupled 12 state Kalman filter written in the C programming language developed by Dr Hu at the IESSG. The software provides the algorithms described in §2.5.2 for mechanising the raw inertial measurements in the navigation frame. GPS position and velocity is read in from a file and integrated with the inertial observations using the decentralised architecture described in §4.4.1. The original Kalman filter contained 9 navigation states and 3 gyro bias states.

4.5.3 KinPos^i

The above GPS and inertial software was combined and extended by the author to produce KinPos^i. KinPos^i provides both decentralised and centralised integration methods utilising double difference carrier phase GPS measurements. The following sections provide an overview of the modifications made to the original software.

4.5.3.1 Modified Filter Architecture

The inertial software was converted to FORTRAN and combined with the KinPos code. This allows the inertial measurements to be used for cycle slip detection/correction and ambiguity aiding. KinPos^i provides the centralised and decentralised filtering structures described in 4.4. The modifications to accommodate inertial measurement processing are indicated in red in Figure 4.4.

For the decentralised and centralised filtering structures, the inertial processing options are selected from a modified KinPos control file. A control file example for KinPos^i is provided in Appendix B. After each GPS epoch has been read in, the inertial data is read and is recursively mechanised up to the current GPS epoch.

For the centralised filtering structure, the GPS only state transition matrix and process noise matrices are replaced with the inertial error model matrices. The observations for the Kalman filter are formed by using the inertially predicted position and velocity to form range measurements. The observed minus predicted double difference observations are then formed using the inertial ranges and the GPS observations. The filter follows the same control flow as the GPS only filter except the state estimation is used to compensate the INS errors, and the state estimates are reset in a closed loop. Using this methodology, the ambiguity estimation algorithm uses the inertial/GPS observations.

For the decentralised filter, the inertial measurements are again mechanised up to the GPS epoch when the data is read in. This also allows the INS to be used for cycle slip detection and correction if required, and

for ambiguity resolution by forming the inertially predicted ranges from the inertial position and velocity estimates. The conventional GPS control flow is then used and the position and velocity is used to update the INS in a separate Kalman filter.

Cycle slip detection is provided by comparing the inertially predicted change in range estimate to the increment in carrier phase between epochs. If the difference in range increments is larger than a threshold value then a cycle slip is detected. The value of the nearest integer to the difference in range increments corresponds to the inertial estimate of the cycle slip and can be used to correct the carrier observation. The ability to detect and resolve cycle slips is dependent on the specification of the inertial sensor, and the accuracy of the INS alignment. For example, in Figure 8.6, the difference between the predicted position from a low cost IMU and the position from the GPS can be as large as 12cm, even after INS alignment has taken place. Therefore cycle slips of 1 cycle cannot be detected and corrected confidently using such a low cost sensor.

In addition to the cycle slip check, the integrity of the Kalman filter innovation sequence can be checked using the 3-σ test,

$$|v_k^{(-)}|_i \leq 3\sqrt{[H_k P^{(-)} H_k^T + R_k]_{i,i}} \tag{4.42}$$

where i represents the i^{th} vector element and i, i is the i^{th} matrix diagonal element. If the innovation measurement is larger than the predicted 3-σ variance, the measurement is omitted from the Kalman filter update.

4.5.3.2 Inertial navigation improvements

Several improvements were made to the inertial navigation software over the original software described in §4.5.2. The static alignment routines described in §2.5.1 were added, and also the initial alignment values can be specified in the control file. The sensor error model in the Kalman filter was extended to include states for gyro and accelerometer bias, scale factor, temperature dependent bias and misalignments using the error models described in §2.6.3.

4.5.3.3 GPS modifications

Some further modifications were made to the GPS processing algorithms in the original KinPos software by the author. The software was updated to include Doppler measurements in the Kalman filter in order to improve velocity estimation for updating the inertial measurements. The stochastic models described in §3.6 were also added to the software in order to improve the stochastic modelling of the GPS observations.

4.5.3.4 Adaptive Kalman filtering algorithms

KinPos^i was further modified to include the adaptive Kalman filtering methods identified in §4.3. The adaptive Kalman filtering algorithms can be used with both the centralised and decentralised filtering structures. For centralised filtering the adaptive estimate of measurement noise can be used in the ambiguity resolution routine.

Chapter 5

Development of a Navigation System Simulator

5.1 Introduction

A problem always encountered with evaluating the performance of an integrated GPS and INS system is obtaining reference information for kinematic datasets. The problem faced is that integrated GPS and INS systems potentially provide one of the highest accuracy methods for position, velocity and attitude estimation in a dynamic environment.

Possible solutions for providing reference information for position, velocity and attitude include using testing facilities such as the SESSYL test track at Laboratoire Central des Ponts et Chaussés (LCPC) (see LCPC (2003)), or using the parameters obtained from a photogrammetric bundle adjustment when the INS is mounted to a camera in a mapping system, for example in Škaloud et al. (1996). Another method that can be used when using lower cost inertial sensors is to compare the low cost GPS/INS

solution to that obtained using a higher grade GPS/INS system. However, such solutions are expensive and can also be restrictive in aspects such as operating dynamics and strict control over possible error sources.

The Navigation System Simulator (NSS) is a GPS and inertial measurement software simulator. The software was developed by various authors at the IESSG to provide measurement simulation for developing and testing algorithms for the integration of GPS and (principally low cost) INS measurements. The simulator is rare in its ability to simultaneously model GPS and IMU measurements which allows full analysis of the results obtained from an integrated GPS/INS system. This includes being able to use the centralised integration architecture as described in the previous chapter.

Originally, the NSS was developed for GPS only measurement simulation. The inclusion of inertial simulation algorithms was part of the European Space Agency (ESA) Low Cost Navigator (LCN) project. The author's role was to validate the algorithms and models used for simulation. The author also extended some of the models to more accurately model the low cost inertial sensor used in this thesis. This chapter documents the models and algorithms used in the measurement simulation, with particular reference given to the contribution provided by the author to the development of the inertial simulation. The NSS is used later in Chapters 6 and 7 to provide test data with full truth information for the development and testing of new GPS and INS integration algorithms.

5.2 GPS simulation

The GPS data simulator generates pseudorange, carrier and Doppler data consistent with the US DoD GPS system described in Chapter 3. The error sources for the GPS measurements are also described in Chapter 3. The following section gives a brief overview of the models used for the simulation of the GPS measurements. The error models and corresponding model parameters are specified in the control file for the NSS, an example of which is found in Appendix C.

5.2.1 Orbital errors

The ephemeris information for simulation is obtained from an input SP3 file. The ephemeris information obtained from the SP3 file is considered the true position and velocity of the satellite. Satellite orbit errors are introduced by using a different ephemeris file for processing to that used for constructing the simulated data. For example, the IGS precise emphemerides can be used for simulation, whereas the data can be processed using the broadcast ephemeris.

5.2.2 Clocks simulation model

The satellite clock errors can be generated by using the clock error from an SP3 file or the polynomial coefficients from a Rinex file. The NSS also provides the option to create randomly generated polynomial coefficients for each satellite using the polynomial (Chao, 1996),

$$dt = a_0 + a_1(T - T_0) + a_2(T - T_0)^2 \tag{5.1}$$

where a_0, a_1 and a_2 are the randomly generated coefficients, T is the GPS time and T_0 is the reference time for computing the polynomial. A random component of the clock error is simulated using a 1-sigma variance specified in the control file.

The receiver clock errors are also simulated using the drift model in Equation 5.1. The coefficients are randomly generated for each receiver. A random clock error is also modelled with the 1-sigma variance specified in the control file.

5.2.3 Atmospheric simulation models

The NSS simulates atmospheric propagation errors using standard atmospheric models. A number of tropospheric models are available in the simulator, for example Magnet or Saastamoinen. For more information about tropospheric delay models see Baker (1998). The Saastamoinen model is used for all simulation runs in this thesis and is a function of elevation angle, total barometric pressure, absolute temperature, water vapour and geodetic height. The simulator uses preset standard meteorological data for the model.

5.2.4 Multipath simulation model

Multipath is simulated using a single reflection surface with a random component. The random noise component is specified as a 1-sigma noise value in the control file. The reflection surface is randomly initialised for each satellite and remains fixed for the reference station. The reflection surface for the roving receiver is randomly generated at each epoch when

the receiver is moving.

5.2.5 Measurement noise

Receiver measurement noise is simulated for all measurements simulated by the NSS. The NSS simulates white noise with 1-sigma error variance specified in the control file.

5.2.6 Cycle slips

Cycle slips are introduced by adding a random integer number of cycles to the carrier data. Cycle slips can occur on L1, L2 or L1 and L2 frequencies. The rate and magnitude of the cycle slips can be altered and all cycle slips are written to a log file. Cycle slips are not simulated for the datasets in this thesis.

5.2.7 GPS attitude model

GPS attitude can be achieved using multiple roving antennas on a fixed baseline. However, as there is no GPS attitude processing software available at the IESSG, provision for simulating multiple kinematic GPS observations has not been included. GPS attitude observations are instead simulated by adding white noise to the attitude measurements in the trajectory file with zero mean and 1-sigma variance as specified in the control file. The misalignment between GPS and INS attitude observations is considered in §5.4.2.

5.3 Inertial models

The models specific to the generation of the inertial measurements are described in this section. The model types and their corresponding values are specified in the NSS control file.

5.3.1 Inertial sensor error equation

The output from the gyros and accelerometers are considered simultaneously in this section since the error models are largely formed using the same error characteristics. The error model is given for a triad of orthogonally mounted sensors and therefore includes sensor misalignment. The inertial sensor measurement can be written in terms of the vector of the true measurement, l, with error characteristics formed using the equation,

$$\hat{l} = l + b + Sl + Nl + \varepsilon \tag{5.2}$$

where,

$\hat{l}$	is the measurement obtained by the sensor in the body frame
l	is the actual measurement vector
b	is sensor bias vector
S	is the scale factor error matrix
N	is the misalignment matrix
ε	is the vector of sensor noise

The errors given in Equation 5.2 are examined in detail in the following sections.

5.3.2 Bias

The model used in the NSS to model the sensor bias from Equation 5.2 is given as (Dorobantu and Zebhauser, 1999),

$$b = b_0 + b_1 t + b_{c1} + (b_{T1} + b_{T2}\Delta T)\Delta T \tag{5.3}$$

where b_0 is the g-independent bias, b_1 is the bias variation, b_{Tn} is the n^{th} order temperature dependent bias and b_{c1} is a cross-axis sensitivity term. ΔT is the difference between the physical temperature, T, and the calibration temperature of the sensor.

The g-independent bias is the constant bias which is independent of the motion of the sensor. The bias may vary from switch-on to switch-on, but is considered constant while the sensor is in operation.

The bias term includes the first order cross axis sensitivity term, b_{c1}. Cross axis sensitivity is caused by imperfections in the sensor assembly resulting in the sensor incorrectly responding to a force perpendicular to the sensitive axis. The cross axis sensitivity is modelled as a percentage of the force sensed by the other two axes of the sensor triad.

The original bias model was extended by the author to include two further bias terms: bias instability and bias temperature dependence. These terms were added in accordance with the specifications quoted by manufacturers of inertial sensors, and results obtained from laboratory testing. The bias variation term was included since the sensor bias is unlikely to be constant over time. If the sensor bias were to be completely constant, the error would be deterministic and could be removed from the Kalman filter once it is resolved. The bias variation is modelled as a first order Gauss-Markov process with the power spectral density and correlation time

specified in the control file.

The temperature dependence term was included due to the temperature sensitivity experienced by low cost MEMS sensors (Yazdi et al., 1998). Equation 5.3 also contains linear and quadratic terms for the sensor temperature dependence. The difference, ΔT, between the actual temperature and calibration temperature is modelled in the simulator by a polynomial expression. The coefficients for the polynomial are specified in the control file. The temperature dependent component is likely to be repeatable and can be reduced through calibration using a temperature controlled environment, hence the temperature dependent bias is not always modelled.

Figure 5.1 shows an example of the temperature sensitivity for the low cost Crossbow AHRS-DMU-HDX sensor used in Chapter 8. The figure

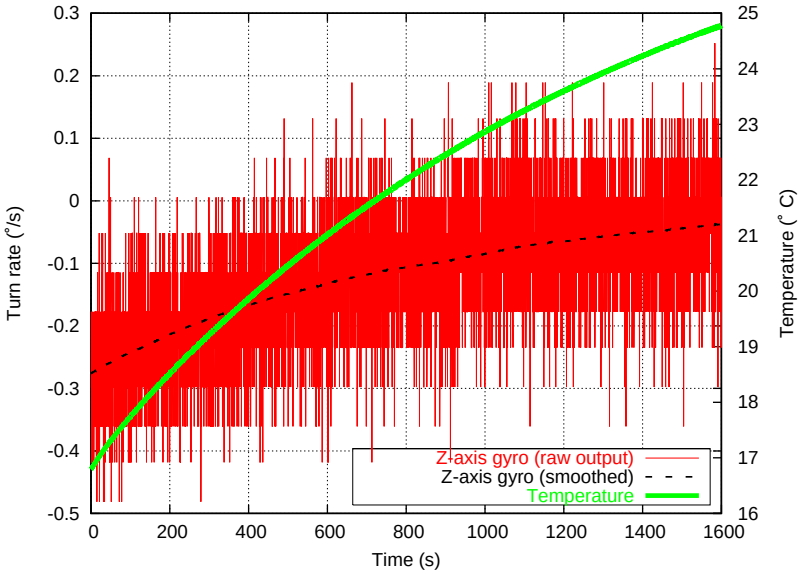


Figure 5.1: Gyro temperature sensitivity

shows the output of one of the gyros while the sensor is stationary and the temperature increases by 8°C. It is clear from the figure that the sensor bias varies by 0.24°/s over the temperature change.

5.3.3 Scale factor

The scale factor is the ratio between the sensor input and sensor output. The scale factor error model in the NSS is represented by the 3×3 matrix (Schwarz and El-Sheimy, 2000),

$$S = S_1 + S_2 l \tag{5.4}$$

where S_n is the matrix of n^{th} order scale factor errors. The scale factor matrix is a diagonal matrix with the errors for the x, y and z axes forming the diagonal elements of S_n. Scale factor calibration can be achieved in the laboratory by incrementing the acceleration or rotation rate and fitting a curve to improve the model (Lawrence, 1999). Small changes in scale factor error for inertial sensors can occur due to temperature changes or because of g-dependent forces (Titterton and Weston, 1997). The scale factors modelled in NSS are constant for each processing run.

5.3.4 Non-orthogonality

The non-orthogonality of the instrument axes is represented by the skew-symmetric matrix, N, in Equation 5.5. The misalignment of the sensor axes is caused by the imperfections in the construction of the sensor assembly. The axis misalignment results in a random error for kinematic applications when the vehicle undergoes frequent manoeuvres. Under constant velocity conditions the error becomes systematic. The elements are formed using the angular misalignment M_{ab} which is the misalignment from the a-axis

to the b-axis.

$$N = \begin{pmatrix} 0 & M_{xz} & -M_{xy} \\ -M_{yz} & 0 & M_{yx} \\ M_{zy} & -M_{zx} & 0 \end{pmatrix} \tag{5.5}$$

5.3.5 Sensor noise and quantisation

Noise in the output of the inertial sensors is caused by factors such as cabling and A-D converter noise, as well as noise caused by the inertial sensors themselves. NSS models sensor noise by adding white noise with standard deviation specified in the control file.

The quantisation error defines the minimum resolution of an inertial measurement and also contributes to the sensor noise. In a physical INS, quantisation error is dependent on the specification of the A-D converter used (bit resolution, full scale range and input gain) and the full output range of the inertial sensor. For example, the smallest resolution of voltage or Least Significant Bit (LSB) of the 16-bit ADAC PCM-5516-16 A-D converter using zero is gain given by,

$$\text{LSB} = \frac{\text{Full scale range}}{\text{Bit resolution}} = \frac{20\text{V}}{2^{16}} \approx 3 \times 10^{-4}\text{V} \tag{5.6}$$

For example, for the Crossbow AHRS-DMU-HDX used in Chapter 8, the conversion from raw voltage output to standard units can be approximated by $1\text{V} \approx 10\text{ms}^{-1}$ and $1\text{V} \approx 36.62°/\text{s}$ (Crossbow Technology Inc., 1999). This results in a quantisation error for the accelerometers and gyros of 0.003ms^{-2} and $0.011°/\text{s}$ respectively when using a 16-bit A-D converter. Measurement resolution can be improved by choosing a sensor which has a smaller operating range and using a high resolution A-D converter.

5.3.6 Saturation

When the measurement exceeds the operating range the sensor becomes saturated. When the sensor is saturated, the maximum output of the sensor is taken as the measurement. This is also modelled in NSS.

5.4 GPS and INS system errors

There are a number of system errors that can be simulated in the combined GPS and inertial data simulator. The system errors are synchronisation error, lever-arm error and GPS to INS attitude misalignment error. These errors were implemented in NSS by the author and are considered in the following sections.

5.4.1 Synchronisation error

In order to combine the inertial measurements with other sensors, the measurements must be synchronised to a common time frame. When GPS measurements are available, the INS measurements are time tagged relative to GPS time. The time tagging method developed by the author for recording the inertial measurements used in this thesis is described at this point in order to motivate the error characteristic to be simulated.

The time tagging method used is a post-processing technique which was developed by the author to provide high accuracy system timing. The INS measurements are sampled by an Analogue to Digital (A-D) converter. The A-D converter used for this thesis is the ADAC PCM-5516-16 which has an on-board user programmable counter (for more information see ADAC Corporation (2002)). The principle used for sensor time tagging

is described in Figure 5.2.

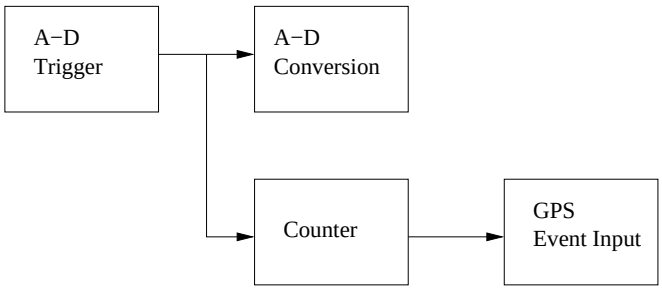


Figure 5.2: Flow diagram for time tagging method

The figure shows that each A-D conversion is triggered by the A-D trigger. The A-D trigger also increments the counter. When the counter reaches a threshold value, the counter is reset and a pulse is sent to the GPS receiver event marker input. The GPS receiver records the time of the pulse in GPS time and stores the event in the internal memory of the receiver. The A-D data is matched to the GPS events when all the data has been recorded. The stability of the A-D oscillator allows pulses to be sent to the GPS receiver at intervals of greater than 1 second. The time of the A-D data recorded between these time-tags can be interpolated between the recorded GPS time-tags.

Considering the described time tagging method, the total timing error t_{error} is formed from the equation,

$$t_{error} = (t_{gps} - t_{conv}) + t_{noise} \tag{5.7}$$

where,

t_{conv} is the time of the A-D conversion

t_{gps} is the time at which the GPS receiver receives the pulse

t_{noise} is the random noise caused by the variation in A-D conversion length, variation in pulse transmission time

and the random noise caused by the resolution of the GPS event marker

The timing error therefore consists of a constant bias with a random component, the magnitude of which is specified in the data simulator control file. For the ADAC PCM-5516-16 card, the A-D conversion takes approximately 5μs (ADAC Corporation, 2002) while the signal transmission time from A-D board to GPS receiver is unknown.

In addition to the model described for timing the A-D conversions, there is also an error caused by the A-D converter sequentially measuring each sensor. Therefore the A-D card actually samples each sensor measurement at a separate time. For example, if the data rate for the IMU measurements is 200Hz, the A-D sample rate is set to 1200Hz when recording six measurements. Therefore the maximum timing error is $1/(S/2)$ where S is the sample rate in Hz. The NSS can also be configured to provide INS measurements using this sequential timing method.

Analysis of the above error sources shows that the timing error is dominated by the sequential A-D conversion and the resolution of the GPS receiver event marker input. The physical navigation error caused by the timing error in GPS and INS integration will also depend on the trajectory characteristics. Timing error will produce no position error for a non-accelerating system. The error will increase proportionally to the system acceleration.

5.4.2 Lever-arm separation and attitude misalignment

In real world applications the phase centre of the GPS antenna and the body frame of the IMU are separated by a lever-arm. When GPS attitude is available, there will also be a misalignment between the GPS derived attitude and the INS attitude.

The position correction for measurements separated by a lever-arm in the navigation frame is computed using the equation,

$$p^n_{GPS} = p^n_{IMU} + C^n_b d^b \tag{5.8}$$

where,

p^n	is the position of the GPS receiver or IMU in the navigation frame
C^n_b	is the DCM from the body frame to the navigation frame
d^b	is the lever-arm separation in the body frame

The velocity correction for measurements separated by a lever-arm in the navigation frame is derived from Equation 5.8 to form,

$$\begin{aligned} v^n_{GPS} &= v^n_{IMU} + \dot{C}^n_b d^b & (5.9)\\ &= v^n_{IMU} + C^n_b \Omega^b_{nb} d^b & (5.10) \end{aligned}$$

where,

v^n	is the GPS or IMU velocity in the navigation frame
Ω^b_{nb}	is the skew-symmetric form of the rotation of the body frame with respect to the navigation frame referenced in the body frame

The misalignment between the GPS attitude and IMU attitude is intro-

duced using the equation,

$$C_n^{b_G} = C_{b_I}^{b_G} C_n^{b_I} \tag{5.11}$$

where b_G and b_I indicate the GPS and IMU body frame respectively. The matrix $C_{b_I}^{b_G}$ is formed using the small misalignment angles between the GPS and IMU frames.

5.5 Environmental errors

In addition to the aforementioned sensor and GPS/INS system errors, environmental error sources exist that degrade the performance of a GPS/INS system. These are vibration and gravity anomalies.

5.5.1 Vibration

Noise in the output of the inertial sensor is also caused by vibration of the sensor, for example from the engines running in a vehicle. Sensor vibration is not strictly an inertial sensor error, but it is considered here due to the error manifesting itself as inertial sensor noise.

Figure 5.3 shows the Fast Fourier Transform (FFT) for the x-axis accelerometer located in a small marine survey vessel with the engines off and the engines running. The data was collected with the vessel moored to a jetty. The movement of the vessel oscillating on the water consists of low frequency movement which from Figure 5.3 is shown as the spike at approximately 0.5Hz. The higher frequency components of the accelerometer output with the engines off can be attributed to noise caused by the internal sensors. The FFT of the x-accelerometer with engines running

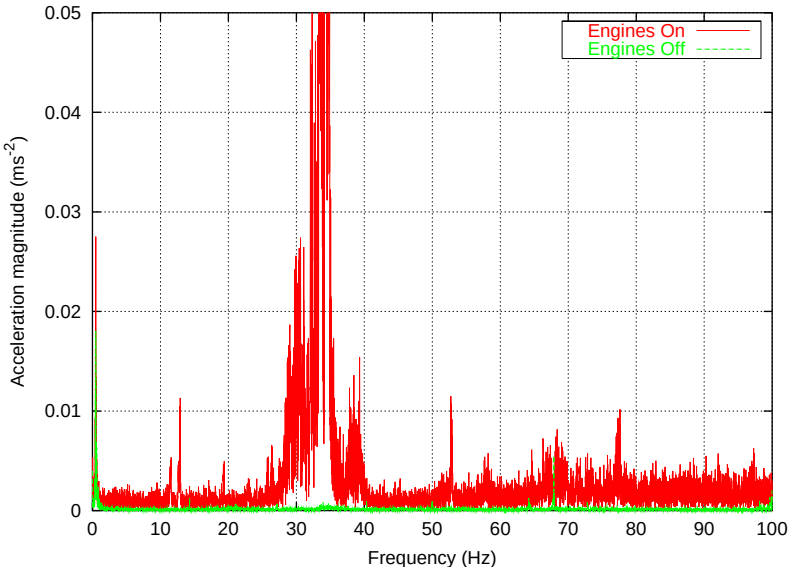


Figure 5.3: FFT for Crossbow x-axis accelerometer

shows the noise at 35Hz caused by vibration due to the engines running. High frequency noise can be removed from the sensor output by using low pass signal filtering. For more information see Škaloud (1995).

The vibration characteristics differ for each sensor axis and depend on the vehicle type and mounting location. The vibration is also likely to vary throughout a dataset due to factors such as engine throttle. This makes sensor vibration very difficult to model. Consequently, noise caused by vibration is only modelled in the simulator by increasing the sensor white noise.

5.5.2 Gravity Anomalies

Section 2.4 described a geodetic model for removing the contribution of gravity to the accelerometer measurements. Equation 2.12 includes three gravity anomaly terms which are location dependent. Because of the small magnitude of the gravity anomalies, they are not modelled for the low cost

sensors.

5.6 Differentiation and interpolation

The original GPS-only data simulator required a trajectory file containing positions and velocities for the kinematic receiver. For the inclusion of inertial data, attitude measurements for each epoch were added to the trajectory file. The simulator provides interpolation of measurements so that data can be simulated at higher data rates than those entered in the trajectory file.

For the GPS-only simulation, the velocity input is used solely for the creation of the Doppler data. The velocity input is the instantaneous velocity of the GPS receiver, an approximation to which is provided by differencing the positions between epochs. For the inertial measurements however, the differentiation of the trajectory input becomes critical. This is because the accelerations and turn rates output by the simulator are integrated recursively by the GPS and INS integration software to calculate velocity and position.

The original version of the data simulator differenced the interpolated velocity inputs from the trajectory file between IMU epochs to calculate the IMU acceleration vector. Everett interpolation was used to interpolate the velocity input which fits a curve to the data, in this case using 12 points (see Agrotis (1984)). This results in two problems. The first is that differencing between epochs is an inaccurate method for numerical differentiation. If an improved method of integration is used to return to velocity, an error will result. This problem also exists in the differentiation

of the attitude in the trajectory file. The second problem occurs when interpolating the differenced velocity in the trajectory file to calculate the acceleration vector which is used for simulating the inertial measurements. This is demonstrated by Figure 5.4. The figure shows four discrete velocity inputs taken from the trajectory file at epochs $i-1$ to $i+2$. The shaded region is the area that needs to be calculated by integration to obtain the correct position due to approximating the velocity by differencing positions in the trajectory file. By fitting a curve to the velocity data points for interpolation, it is observed that integrating the curve at a higher data rate will result in a position error.

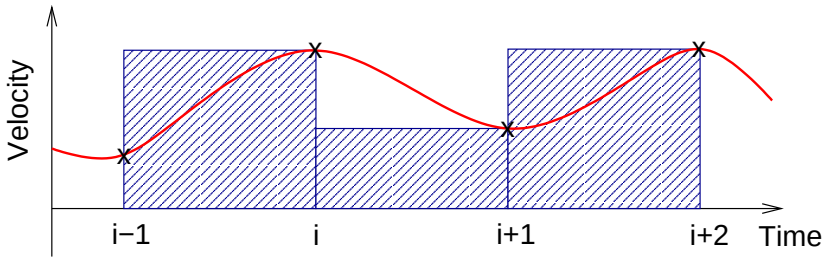


Figure 5.4: Velocity interpolation

A possible solution to the first problem is to use a more accurate differentiation method, for example Simpson's Rule. If the step size between epochs is small, the differentiation/ integration process will only introduce a small error. A possible solution to the second problem would be to simply differentiate the interpolated positions twice to obtain acceleration instead of differentiating the velocity. However, this results in an inaccurate value for acceleration due to the errors introduced through double numerical integration. Instead, this provides the motivation for the spline interpolation technique described in the next section.

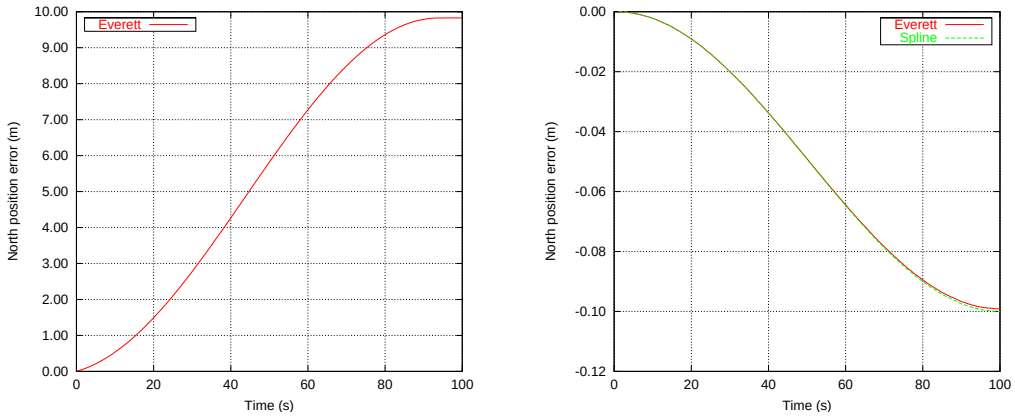
5.6.1 Cubic spline differentiation and interpolation

One method for interpolating the positions from the trajectory file is to use curve fitting. However, different curves that are fitted between each trajectory file position may result in a discontinuous curve at each trajectory input point.

The solution developed by the University of Nottingham based on results supplied by the author is to use cubic splines. Cubic splines fit a series of cubic functions to discrete points with the condition that the slope and curve at each of the adjoining cubic equations is equal (Burden and Faires, 1993). An overview of the method is given in Appendix A. The cubic spline algorithm is used to fit cubic functions to the discrete position and attitude trajectories in the simulator. The cubic functions provide interpolation curves between the position and attitude inputs which can be mathematically differentiated to provide exact velocity, acceleration and attitude rates for simulating the inertial measurements. Therefore no errors are introduced through double differentiation. This fully overcomes the two problems described in the previous section. As a result, the effect of integration error in the inertial system can be evaluated with the data simulator.

Figure 5.5 shows the position error from using the different interpolation/ differentiation methods on error free inertial data. Figure 5.5(a) shows the effect described in Figure 5.4 where the velocity estimates obtained by differencing the positions in the trajectory file are interpolated and used to form the acceleration vector for the inertial measurements. It is observed from the figure that during an acceleration manoeuvre from

zero to 20ms^{-1} in a straight line, this method results in a significant North position error of approximately 10 metres. Figure 5.5(b) shows the error obtained from using cubic spline interpolation and also Everett interpolation using exact instantaneous velocity in the trajectory file. The exact velocity was obtained by differentiating the mathematical equations that were used to model the acceleration manoeuvre.



(a) Differenced velocity with Everett interpolation

(b) Spline Interpolation and Everett interpolation with exact velocity

Figure 5.5: Differentiation/ integration error

The figure shows that the two methods obtain very similar integration errors of 9.91cm and 9.99cm at the end of the manoeuvre for the Everett and Spline interpolation methods respectively. Therefore, particularly for low cost inertial sensors, either Spline integration or Everett interpolation with exact velocity information is suitable for the interpolation of the trajectory input. However, it should be noted that obtaining the exact instantaneous velocity from real trajectory files is not possible (see §5.8).

5.7 Simulation mechanisation

The inputs required for the data simulator are the IMU position, velocity, acceleration, attitude and attitude rate at a user defined epoch separation. The derivatives of position and attitude are calculated a separate program which performs the cubic spline differentiation as described in the previous section. This section describes the reverse mechanisation process used in the simulator to calculate the body frame turn rates and accelerations.

The mechanisation process is described by Figure 5.6. Firstly the Coriolis and Gravity accelerations are added to the vehicle acceleration in the Earth frame. The Cartesian positions are then converted to geodetic positions (for details of this transformation see Leick (1995)). The geodetic position is used to calculate the rotation matrix in Equation 2.3 to convert the Earth frame acceleration to the navigation frame. The attitude vector from the trajectory file is used to calculate the DCM which is used to rotate the force measurement from the navigation frame to the body frame.

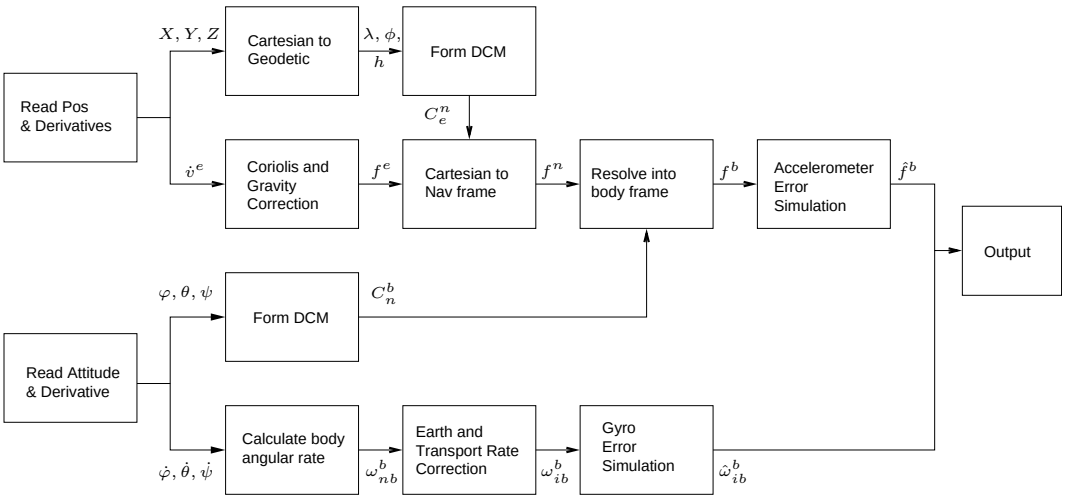


Figure 5.6: Simulator measurement mechanisation

The attitude rates are converted to body frame angular rates using the

equation,

$$\begin{pmatrix} \omega_x \\ \omega_y \\ \omega_z \end{pmatrix} = \begin{pmatrix} 1 & 0 & -\sin(\theta) \\ 0 & \cos(\varphi) & \cos(\theta)\sin(\varphi) \\ 0 & -\sin(\varphi) & \cos(\theta)\cos(\varphi) \end{pmatrix} \begin{pmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} \tag{5.12}$$

The derived quantity is the rotation of the navigation frame with respect to the body frame referenced in the body frame. Equation 2.19 is rearranged to calculate the rotation of the body frame with respect to the inertial frame, which is the quantity measured by the accelerometers. Combining this mechanisation algorithm with exact derivatives of position and attitude measurements removes any errors associated with the mechanisation, apart from those introduced by finite computer precision.

5.8 Construction of kinematic trajectories

As mentioned previously in this chapter, the data simulator requires trajectory data in the form of position, velocity, acceleration, attitude and attitude rate data. The cubic spline interpolation method removes the requirement for entering the derivatives of position and attitude. The epoch separation for the input trajectory can be any value as long as it is constant. If modelling the high frequency motion of a vehicle, the Nyquist theorem means that the data rate must be twice the frequency that needs to be modelled (Chatfield, 2003).

Simple computer programs can be developed to produce reference trajectories. For example, static and constant velocity trajectories are easily formed which may be desirable when trying to assess the performance of a non-accelerating system. Other simple trajectories such as circles or figure-

of-eights can also be formed using mathematical equations. By using mathematical equations, it is also possible to directly derive the velocity which allows the simulator to be used with the Everett interpolation algorithm.

Another approach for forming reference trajectories is to use the trajectory obtained from a GPS or integrated GPS/INS trial. This has the advantage of simulating the actual movement of a vehicle. However, care needs to be taken using OTF GPS for forming trajectory files where loss of lock on the satellite signals can result in jumps in the position data. Furthermore, it is not possible to derive the exact velocity from the position estimates. This means that the cubic spline differentiation and interpolation algorithm must be used.

The marine trial used in the following chapter was obtained from the TSS POS/MV integrated GPS and INS system described in Chapter 8. The GPS/INS data has the benefit that navigation accuracy is not significantly degraded during short GPS outages. Furthermore, the navigation output is provided at a high data rate which allows the high frequency characteristics of the trajectory to be modelled.

Chapter 6

Analysis of CKF alignment using data simulation

6.1 Introduction

The chapter demonstrates the use of the CKF algorithm for integrating GPS and low cost INS measurements obtained using the simulation software described in the previous chapter. The NSS simulation software is used to model a figure-of-eight alignment trajectory in a marine environment. The marine environment is particularly challenging for INS alignment since it is not possible to use static initialisation techniques due to the marine vessel moving on the surface of the water. This chapter is used to show the characteristics of the CKF, validate the simulation algorithms from Chapter 5, and motivate the use of the adaptive Kalman filtering algorithms used in the following chapter.

6.2 Trajectory and trial description

The following examination of the CKF uses the trajectory shown in Figures 6.1 and 6.2. The trajectory was obtained from a TSS Position and Orientation System for Marine Vessels (POS/MV) integrated GPS and INS system installed in a small survey boat owned by the Royal Naval Hydrographic School in Plymouth (see §8.2).

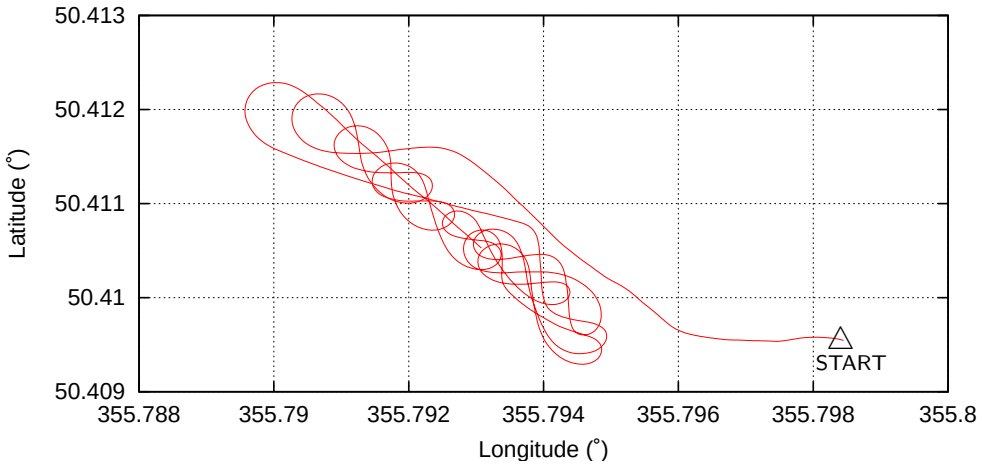


Figure 6.1: Marine alignment trajectory

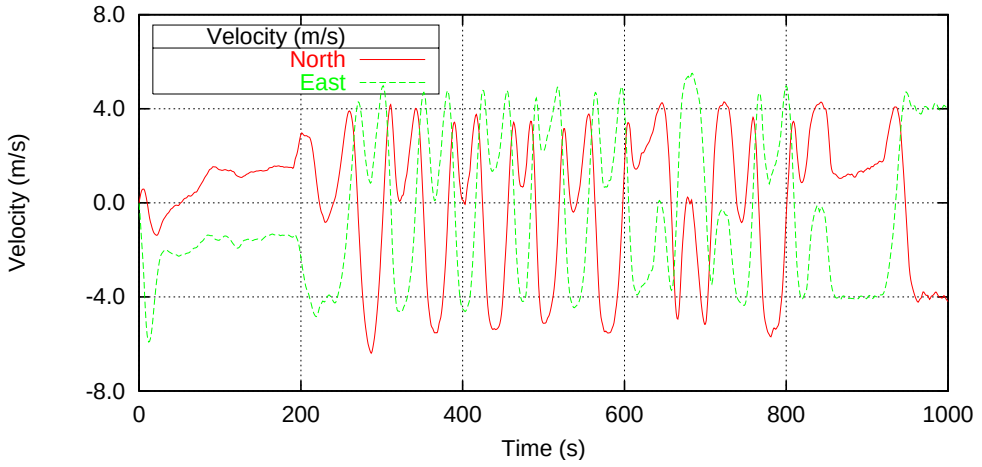


Figure 6.2: Marine alignment velocity

In order to model the high frequency motion of the vessel, the input trajectory was defined for simulation at 10Hz. The vessel undergoes a short traverse for approximately 3 minutes before commencing figure-of-eight alignment manoeuvres that introduce horizontal acceleration. The north and east velocity is shown in Figure 6.2 as an indication of the horizontal acceleration. The horizontal acceleration information is important as it is required for the estimation of some of the states in the filter as described in §2.6.2. The dataset lasts for approximately 17 minutes.

The following trials simulate a sensor with the performance characteristics of a low cost IMU. The errors simulated correspond to the Crossbow AHRS-DMU-HDX sensor described in Table 6.1. The IMU measurements are simulated at 100Hz and the temperature of the sensor remains constant throughout the dataset (the temperature is assumed constant to simplify the analysis). The sequential sampling method described in §5.4.1 is used to simulate the synchronisation error. Dual frequency carrier phase and pseudorange GPS measurements to 8 satellites from a static base and single roving GPS antenna are simulated. The baseline separation between the base and roving antennas is less than 3km for the entire processing run. The L1 frequency carrier phase and Doppler data is used to obtain position and velocity and the GPS ambiguities are fixed for the entire dataset. A 1 metre lever arm offset in each axis from IMU to GPS antenna is also defined. The initial pitch, roll and yaw misalignment is 1.6°, 0.6° and 2° respectively which corresponds to the level error of the vessel resting on the water and an initial heading error which would be obtained from, for example, an on-board compass.

The INS measurements are integrated in this chapter using the de-

Error source	(Units)	Gyro	(Units)	Accel
Saturation	(°/s)	100	(ms^{-2})	40
Quantisation	(°/s)	0.011	(ms^{-2})	0.003
Bias	(°/s)	1	(ms^{-2})	0.294
Bias Variation	(°/s/$\sqrt{Hz}$)	1×10^{-5}	($ms^{-2}/\sqrt{Hz}$)	1×10^{-5}
Correlation time	(s)	100	(s)	60
Cross-axis sensitivity	(-)	-	(%)	< 1
Scale Factor	(%)	<1.0	(%)	<1.0
Scale Factor Non-linearity	(% FS)	<1.0	(% FS)	<1.0
1st Order Temp Sensitivity	(°/s/°C)	0.017	(ms^{-2}/°C)	0.002
Noise	(°/s/$\sqrt{Hz}$)	0.005	($ms^{-2}/\sqrt{Hz}$)	0.1
Axis Misalignment	(mrad)	3	(mrad)	3

Table 6.1: Specification for Crossbow AHRS-DMU-HDX

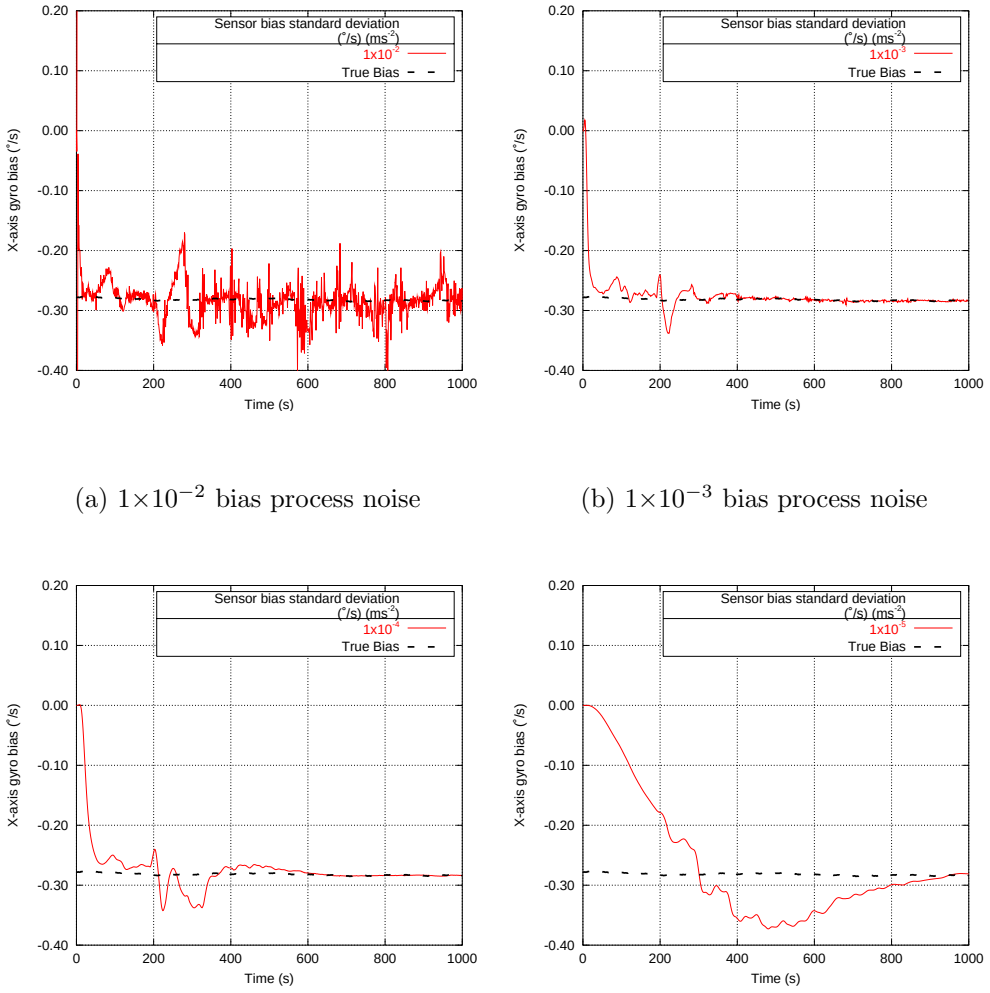
centralised filter architecture with the centralised filter only being used when explicitly mentioned. This is because the large initial INS biases cause large initial navigation errors which can degrade ambiguity resolution when the initial inertial sensor error estimates have not been obtained. The gyro and accelerometer biases are estimated in the filter whereas the scale factor and misalignment errors are not estimated and are therefore considered random errors due to the movement of the boat. Both position and velocity measurements from the GPS filter are used as measurements in the decentralised INS filter.

6.3 INS alignment using the CKF

For marine applications it is rarely possible to estimate the inertial sensor errors using the static initialisation techniques described in §2.5.1 because of the movement of the vessel on the water. Low cost sensors that experience large switch-on to switch-on variation can be estimated using dynamic alignment. The purpose of the dynamic alignment is to initialise the sensor error estimates, and align the INS to the navigation frame. Navigation performance of the INS is therefore reduced before alignment and it is desirable to obtain full INS alignment as rapidly as possible. Furthermore, if large inertial sensor errors remain uncompensated, the filter may become unstable due to the large build up of the inertial navigation errors. The following section performs an analysis using the sensor bias and attitude alignment to show the time that is required for alignment.

Figure 6.3 shows the estimate of the x-axis gyro bias from the CKF in KinPos^i using different *a priori* sensor bias process noise estimates. Figure 6.4 shows the corresponding attitude estimate for two of these cases. The same sensor bias process noise estimates are used for both the accelerometer bias error (with units ms^{-2}) and gyro bias error (with units $^\circ$/s). Units for the process noise bias states are therefore omitted in the following analysis.

It is clearly observed from Figure 6.3 that the larger process noise values of 1×10^{-2} to 1×10^{-4} result in a more rapid transition to the correct bias estimate. The bias estimate of 1×10^{-2} results in the most rapid transition in approximately 20 seconds. However, the bias estimate is noisy and deviates during the processing run by up to approximately 0.15°/s. This is



(a) 1×10^{-2} bias process noise

(b) 1×10^{-3} bias process noise

(c) 1×10^{-4} bias process noise

(d) 1×10^{-5} bias process noise

Figure 6.3: CKF x-axis gyro bias estimate

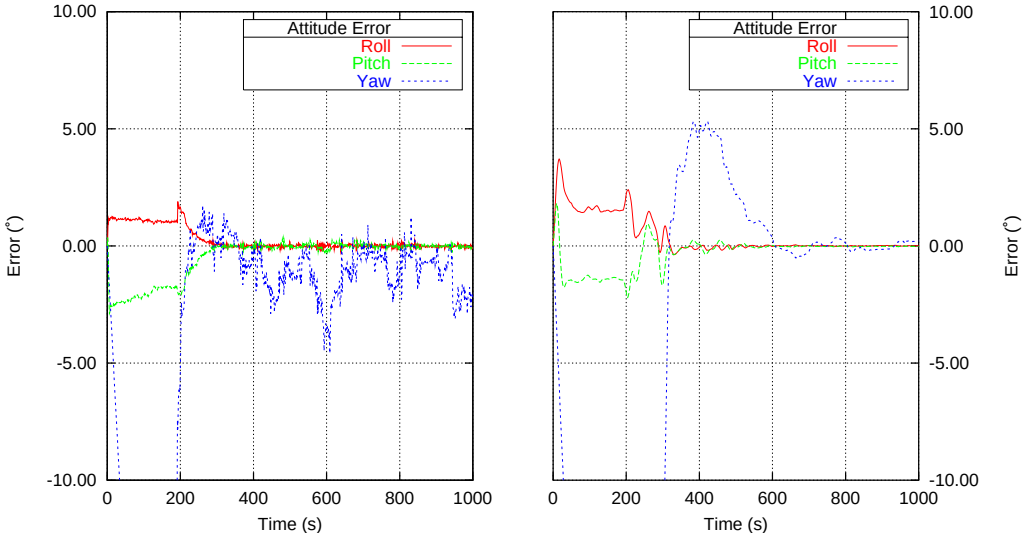
because too much weight from the innovation sequence is given to the bias estimate. The points at which there appears to be a systematic deviation from the true bias value are points where there are high dynamic manoeuvres in the vessel's trajectory (see the corresponding vessel velocity in Figure 6.2). For approximately the first 200 seconds of the dataset, the trajectory does not undergo significant horizontal acceleration in order for all of the error states to become observable. Weakly observable states such

as yaw error, z-axis gyro bias and x and y-axis accelerometer bias states are not fully resolved after 200 seconds and this affects the bias estimation of the x-axis gyro shown. This is examined further in the following section. Once all states are resolved, the 1×10^{-2} sensor bias estimate provides a noisy estimate of the x-axis gyro bias.

The bias process noise standard deviation estimates of 1×10^{-3} and 1×10^{-4} result in similar performances, with the larger process noise resulting in a slightly quicker transition to the correct estimate. Once again, the bias estimate temporarily deteriorates after 200 seconds due to the dynamic manoeuvre, however after this point a good estimate of the bias is maintained. It is observed from the 1×10^{-5} process noise estimate that the smaller process noise results in a very slow transition to the correct bias estimate. The estimate still has not fully converged after approximately 16 minutes.

Figure 6.4 shows the corresponding attitude alignment for two of the cases in used Figure 6.3. Figure 6.4(a) corresponds to the bias standard deviation of 1×10^{-2}. This figure shows that after approximately 220 seconds the attitude error state has converged, however, the resultant attitude estimate is noisy. The attitude error after the initial 220 second alignment period is 0.07°, 0.09° and 1.54° RMS for roll, pitch and yaw respectively. The attitude alignment for the bias standard deviation of 1×10^{-4} in Figure 6.4(b) results in an improved navigation performance after the alignment period with corresponding RMS errors of 0.01°, 0.02° and 0.21°. However, the alignment takes approximately 600 seconds which is over twice alignment time of the larger bias estimate.

Figures 6.3 and 6.4 demonstrate that there is a requirement to adapt the



(a) Bias process noise 1×10^{-2} (b) Bias process noise 1×10^{-4}

Figure 6.4: CKF Attitude alignment

stochastic information used in the CKF during alignment. The estimation of the stochastic properties of the filter is a challenging task as the inertial errors vary from switch-on to switch-on and the estimation of the sensor errors depends on factors such as vehicle manoeuvres. This problem is particularly prevalent in the marine environment due to the inability to perform any form of static initialisation, but is also an issue in all other operating environments where a rapid alignment is required.

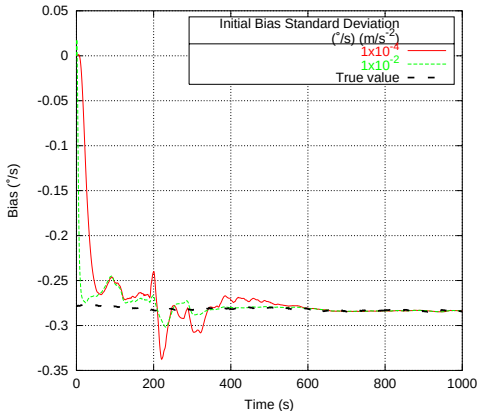
6.3.1 Dependence on initial covariance

This section examines the effect of the initial covariance, P_0, on the alignment of the INS. The initial covariance is formed using *a priori* estimates of the error states. For the states which are directly observable (i.e. position and velocity) the dependence on the initial covariance is minimal since

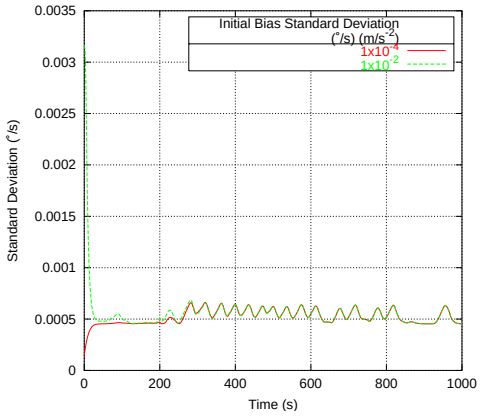
the observations are used directly to estimate the states. For the weakly observable states, in particular, yaw misalignment, z-axis gyro and x and y-axis accelerometer states, the initial covariance has a greater influence on the state estimate as shown in the following analysis.

Figure 6.5 shows the x and z-axis gyro bias alignment and corresponding updated state covariance, $P_k^{(+)}$, using two different initial bias state covariance estimates in the Kalman filter. The two cases are formed using a small initial covariance where the initial covariance is set equal to the process noise, and an empirical estimate where the initial covariance is estimated using the truth from the data simulator and several processing runs (the standard deviation used is 1×10^{-2}). In both cases the initial covariance matrix, P_0, is a diagonal matrix. The navigation state process noise elements remain the same as those used in the previous section and the bias process noise is set to 1×10^{-4} for both the gyros and accelerometers. This value is used for the bias states as it provided the best navigation performance once the system is aligned.

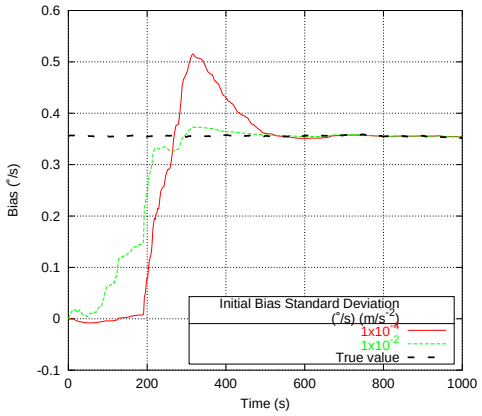
The Figures 6.5(a) and 6.5(b) show the x-axis gyro bias estimate and corresponding covariance. It is observed from Figure 6.5(a) that the larger initial covariance results in a more rapid convergence to the correct bias value due to more weight from the measurements being applied to the states. The x-axis gyro bias is estimated to within 0.02°/s in the initial 20 seconds but the estimate is not maintained. The bias does not converge to 0.01°/s until approximately 350 seconds due to other bias estimates not being fully resolved. The state covariance estimates for both initial covariance matrices in Figure 6.5(b) demonstrate that the standard deviations converge to within $5\times10^{-6\circ}$/s after 20 seconds. This is due to the gyro



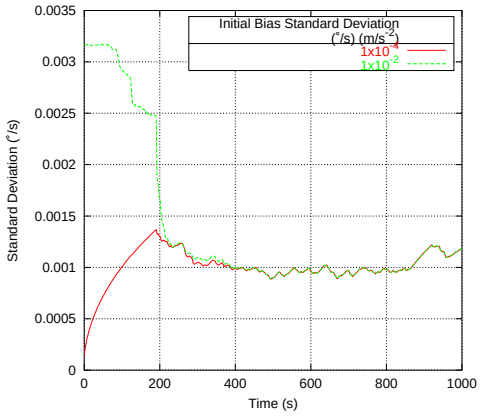
(a) x-axis gyro bias



(b) x-axis gyro bias covariance



(c) z-axis gyro bias



(d) z-axis gyro bias covariance

Figure 6.5: CKF x and z-axis gyro bias alignment for different initial covariances

bias state being observable through the acceleration in the down direction which is always large due to the force due to gravity. The state converges because the same process noise estimates are used in both cases.

The Figures 6.5(c) and 6.5(d) show the z-axis gyro bias estimate which is weakly observable and requires the presence of horizontal accelerations in order for it to be estimated. It is observed from the z-axis bias estimate

that the larger initial covariance again results in a quicker convergence to the true value. For the smaller initial covariance, the bias estimate remains relatively constant during the first 200 seconds where there is low horizontal acceleration. However, the larger initial covariance results in the z-axis gyro bias state being updated from the small horizontal acceleration manoeuvres due to more weight being applied from the measurements through the filter gain. This results in the z-axis gyro bias estimate being improved during this time.

It is observed from the corresponding z-axis gyro bias covariance that the initial covariance is not updated as rapidly as the x-axis gyro bias covariance due to the weak observability. At 200 seconds, the estimates converge due to the presence of the large dynamics manoeuvre. This indicates that the initial covariance for the z-axis gyro is more critical than other states with higher observability. As a result of the improved estimate of the initial covariance, the z-axis gyro bias estimate becomes aligned to within $0.01^\circ/s$ within 385 seconds, compared to 490 seconds for the lower initial covariance. Furthermore, the improved bias estimation also results in improved navigation performance during alignment and at 453 seconds, the yaw misalignment is obtained to within 0.5°.

While the larger empirically derived initial covariance estimates provide improved results, the values were defined using empirical knowledge of the system using multiple processing runs. Furthermore, defining an initial covariance that is too large can result in a significant decrease in alignment performance due to giving too much weight to the measurements. For example, the initial gyro bias error specification from Table 6.1 is $1^\circ/s$. If this value is used as the initial covariance for the gyro bias, the Kalman

filter overestimates the gyro bias resulting in bias estimates of up to 10°/s during alignment. Because the initial bias is unknown and is different for each sensor, the initial bias covariance cannot be reliably estimated. When the initial covariance information is of low quality, the process noise information needs to be adapted in order to identify the correct state covariance. Therefore, it is desirable to estimate the stochastic information on-line at each epoch by another means such as the adaptive techniques investigated in the following chapter.

6.3.2 Innovation and residual sequence

Section 4.3.1 identified that an indication of the system performance can be obtained from the innovation and residual sequences. Figure 6.6 shows the innovation and residual sequence for the north velocity error from the previous section using the bias process noise of 1×10^{-4} and a small initial covariance. The figure also shows the corresponding north velocity error and the yaw misalignment.

It is observed from the figure that the innovation and residual sequences during alignment contain significant biases during the initial alignment period of approximately 600 seconds. The biases in the innovation and residual sequences reflect the trend of the error for the north velocity. This is expected since the GPS observations provide a relatively unbiased estimate of the receiver velocity, and the inertial errors are significantly larger than the GPS errors during alignment. The residual sequence reflects this error more accurately since the residual sequence is the difference between the INS and the GPS estimate after the Kalman filter update. The difference between the innovation and residual sequences is the error

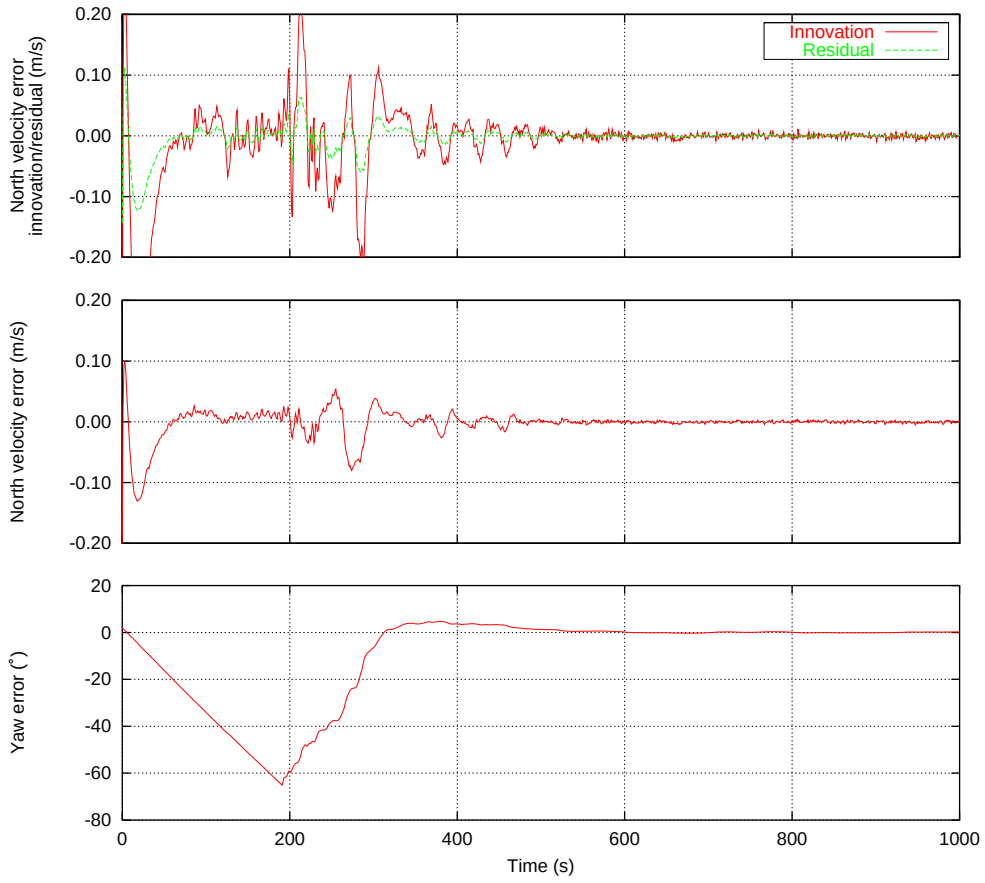


Figure 6.6: Errors during alignment a) Innovation and residual sequence b)North Velocity c) Yaw

estimate provided by the Kalman filter i.e. the state correction sequence in Equation 4.31.

As the inertial errors are compensated, the biases in the innovation/residual sequences reduce and follow an approximately normal distribution. The Lilliefors test is used to test if the sequence follows a normal distribution (Conover, 1999). The Lilliefors test compares the observed cumulative distribution function taken from the data against the theoretical cumulative distribution. The Lilliefors test statistic for the north velocity innovation sequence is computed to be 0.048 between 500 and 600 seconds. This is

less than the the value of 0.088 that is required to reject the null hypothesis that the sequence is Gaussian at the 95% confidence level. So, after 500 seconds, the innovation sequence for the north velocity innovation sequence approximates a Gaussian distribution. Before, 500 seconds, the sequence does not approximate a normal distribution, for example, the Lilliefors test statistic between 400 and 500 seconds is computed to be 0.16.

The figure also shows the INS yaw error. It is observed that the yaw error has a very different characteristic to the velocity estimate due to the yaw error observability. The yaw error causes an error in the north and east velocity measurements only when the vessel undergoes horizontal acceleration through Equation 2.26. Therefore, if the vessel experiences zero horizontal acceleration, the yaw error does not contribute to the error in the innovation/residual sequence, i.e. it is unobservable. This is reflected in the figure. During the first 50 seconds, the filter estimates the states with higher observability. When these are resolved, the magnitude of the innovation/residual sequence is reduced during the period of 50-200 seconds, even though the yaw error is increasing. When the vessel undergoes the alignment manoeuvre at 200s, the error is observable through the horizontal velocity which results in the spikes in the innovation/residual sequence.

As the errors are resolved using the Kalman filter estimates, the magnitude of the innovation/residual sequence reduces and the biases are removed. This shows that the information contained in the innovation/ residual sequence is dependent on the observability of the system errors. This also demonstrates that the innovation and residual sequences contain useful information about the stochastic properties of the sensor errors which can be used for the adaptive estimation algorithms examined in

the following chapter.

Chapter 7

On-line Stochastic Estimation for Low cost INS

7.1 Introduction

This chapter demonstrates the use of the different adaptive Kalman filtering approaches described in Chapter 4 for the alignment and initialisation of an integrated GPS and low cost INS system. The data used is the simulated data for a marine trial used in the previous section. As the INS becomes aligned, the stochastic properties of the inertial system change significantly and the adaptive estimation techniques are employed in order to track these changes.

The adaptive Kalman filtering algorithms in this chapter are compared to the CKF estimate that provided the best navigation performance in the previous chapter. This was achieved by tuning the Kalman filter by performing multiple processing runs and identifying the stochastic matrices that provide the best performance. The optimised initial covariance esti-

mate that was shown to provide the fastest alignment time in Section 6.3 is also used for the CKF in this chapter. In a real system, this would not be possible and the performance of the CKF alignment is likely to be degraded because of this. Therefore, the adaptive Kalman filtering algorithms are tested against the best performance that is expected to be obtained from the CKF using *a priori* stochastic information.

Initial results on which this chapter is based are presented in Hide et al. (2002) and Hide et al. (2003).

7.2 Covariance scaling

The first method considered in this chapter for adapting the stochastic properties of the Kalman filter is to artificially scale the predicted covariance using a scale factor as given in Equation 4.25. A number of techniques can be used in order to estimate the scale factor, S_k. The first method considered to evaluate the effect of covariance scaling is to use an *a priori* scale factor which has been empirically estimated by analysing several processing runs.

7.2.1 *A priori* scale factor method

To examine the effect of scaling the predicted covariance matrix, four scale factors are used at 50 second intervals. The scale factors used are S_k =1.2, 1.15, 1.1 and 1.05, with conventional Kalman filtering resumed after 200 seconds by using the scale factor, $S_k = 1$. The scale factors used reduce after each 50 second interval since it is expected that the inertial errors are reduced due to the dynamic alignment. The scale factors were estimated

using several processing runs to find values that result in a significant improvement in time to alignment for this particular dataset. The filter again uses the process noise matrix from §6.3 with the bias process noise of 1×10^{-4} since this provided the best navigation performance in the previous analysis once the system is aligned.

Figure 7.1 shows the z-axis gyro bias estimate with the covariance scale factors applied. From the examination performed with the conventional Kalman filter in §6.3, it was shown that the z-axis gyro bias is difficult to estimate due to its weak observability. Furthermore, the convergence of other bias estimates requires all states to be successfully resolved, hence the z-axis gyro bias is used as a performance indicator for examining the effect of the adaptive estimation algorithms.

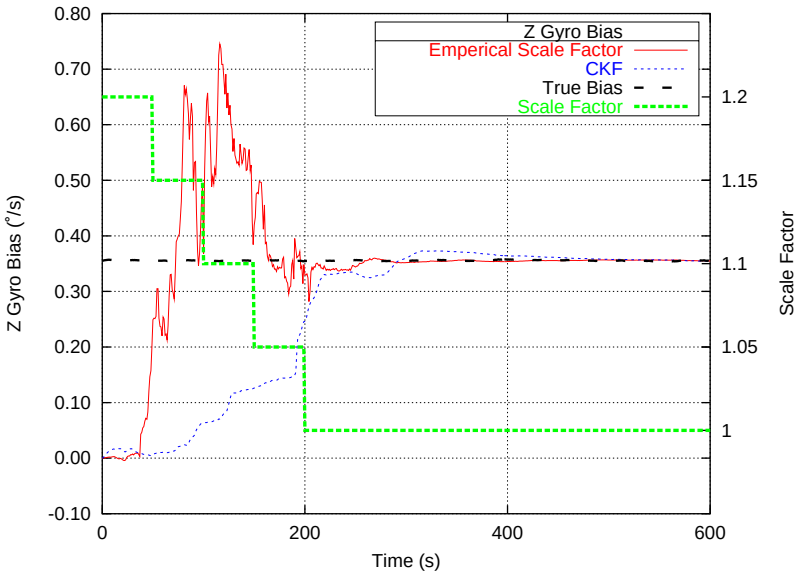


Figure 7.1: Covariance scaling z-axis gyro bias alignment

Figure 7.1 demonstrates that the empirical scale factors result in the z-axis gyro bias being resolved to within 0.01°/s after 252 seconds, compared to the CKF estimate of 385 seconds (which was obtained using empirical

testing). The estimate is improved over the conventional Kalman filter because the initial covariance, P_0, and the *a priori* process noise are underestimated at the beginning of the processing run in the conventional Kalman filter. The covariance matrix is therefore increased at each epoch by scaling the covariance matrix. Since the process noise is valid once the initial large biases have been resolved, the conventional Kalman filter algorithm resumes with $S_k = 1$.

The covariance scaling algorithm results in the attitude errors converging to within 0.5° after 280 seconds which is a significant improvement over the 453 seconds required for the conventional Kalman filter using the same process noise matrix. The amount of additional processing over the conventional Kalman filter algorithm is negligible since S_k is a scalar value. However, using *a priori* estimates for estimating the scale factors is impractical for an adaptive Kalman filter design since the magnitude of the inertial navigation estimates will vary from switch-on to switch-on, and the speed of the Kalman filter alignment also depends on the dynamics of the vehicle in which the system operates. The scale factors for the process noise scaling filter were selected using knowledge of the navigation errors and are therefore unlikely to produce similar results for every other environment in which the inertial system is likely to operate. The results do however demonstrate that using a predicted covariance scaling algorithm can result in a significant improvement in the time to alignment. The following sections describe how the scale factor can be automatically selected using the innovation sequence.

7.2.2 Development of a process noise scaling algorithm

Hu et al. (2001) provide a method for estimating the scale factor by using what is termed a fading memory algorithm. The fading memory algorithm uses a window on the previous measurement innovations. The new innovation is used to form a scale factor using the equation,

$$S_k \geq \frac{v_k^{(-)T} v_k^{(-)}}{\frac{1}{N}\sum_{j=0}^{N-1} v_{k-N+j}^{(-)T} v_{k-N+j}^{(-)}} \tag{7.1}$$

where N is the window length. The denominator in Inequality 7.1 is an estimate of the covariance of the innovation sequence. Therefore the algorithm selects a scale factor using the expected innovation covariance at that epoch. When $S_k \leq 1$, the filter is assumed to be in steady state processing and conventional Kalman filtering is continued. For $S_k > 1$, the sum of the square of the innovations is larger than expected, so the scale factor is used to give more weight to the measurements. Hu et al. (2001) used this algorithm in order to reduce the mis-modelling errors associated with a constant velocity model for pseudorange GPS observations during dynamic manoeuvres. The algorithm relies on the filter having reached steady state processing in order for the covariance approximation to give an accurate estimate of the innovation covariance. For the alignment problem with the GPS/INS integrated system, the algorithm is unsuitable since the Kalman filter has not reached steady state.

Consequently, a modified innovation based scale factor was developed by the author using an estimation window on the innovation sequence of the system when the INS is aligned. Therefore Inequality 7.1 is modified

to give the equation,

$$S_k \geq \frac{v_k^T v_k}{E} \tag{7.2}$$

where E is the expected covariance of the innovations when the system is aligned. An estimate of E can be easily obtained from analysis of the conventional Kalman filter using field trials. A further advantage is obtained with this implementation since the window function is removed therefore removing the lag that occurs when accumulating innovations inside the window function.

Testing this equation with the low cost INS dataset caused problems with filter stability due to the large uncompensated inertial errors. The estimated scale factor was too large which resulted in divergence of the filter. Therefore, Equation 4.25 was modified to apply the scale factor only to the process noise matrix to give,

$$P_k^{(-)} = \Phi P_{k-1}^{(+)} \Phi^T + S_k Q_{k-1} \tag{7.3}$$

The new equation is termed the *process noise scaling* algorithm. The above equation can be formed by following the proof in Hu et al. as follows.

Consider the equation,

$$v_k^{(-)^T} v_k^{(-)} \leq Tr(H_k P_{k-1}^{(-)} H_k^T + R_k) \tag{7.4}$$

where $Tr()$ is the matrix trace function and $H_k P_{k-1}^{(-)} H_k^T + R_k$ is the covariance of the innovation sequence.

Substituting the new process noise scaling equation from Equation 7.3 results in,

$$v_k^{(-)^T} v_k^{(-)} \leq Tr(H_k \Phi P_{k-1}^{(+)} \Phi^T H_k^T + S_k H_k Q_{k-1} H_k^T + R_k) \tag{7.5}$$

and given that $S_k \geq 1$ we have,

$$v_k^{(-)^T} v_k^{(-)} \leq S_k Tr(H_k P_{k-1}^{(-)} H_k^T + R_k) \tag{7.6}$$

which results in the formulation of Inequality 7.1. The only difference in this formulation is that the required scale factor is underestimated compared to the scale factor used for the covariance scaling algorithm.

7.2.3 Process noise scaling

Figure 7.2 shows the z-axis gyro bias estimate obtained using the new process noise scaling algorithm compared to the estimate obtained using the CKF. The CKF estimate is that obtained from §6.3.1 where the initial covariance matrix was defined using multiple processing runs to find an initial covariance that results in a fast time to alignment. The figure shows that the resolution of the initial bias estimate is improved over the CKF estimate. The z-axis gyro bias is obtained to within 0.01°/s within 210 seconds compared to 385 seconds for the CKF. It is shown in the figure that the estimated bias responds more than the CKF estimate to the small change in heading manoeuvres during the period of low dynamics. However, the presence of the larger dynamic manoeuvres beginning at approximately 190 seconds are still required in order to fully resolve the bias state to within an error that allows navigation performance to be maintained.

Figure 7.3 shows the corresponding attitude estimation from the process noise scaling filter. The figure shows that the maximum yaw misalignment is approximately 20°. For the CKF the maximum error was 67° and it was not until the period of higher dynamics that the yaw error was reduced. The presence of the small dynamic manoeuvres is clearly observed

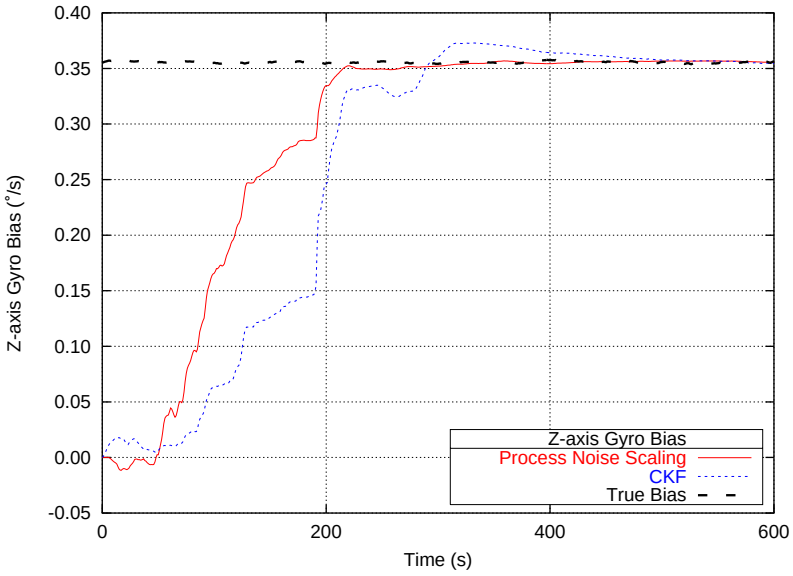


Figure 7.2: Process noise scaling z-axis gyro bias alignment

at 70 and 120 seconds indicating that the filter has adapted to give more weight to the GPS measurements than the CKF. Although the alignment is initially obtained to within 0.5° after 212 seconds, the alignment is not maintained because the z-axis gyro bias is not resolved sufficiently to maintain the heading accuracy, which drifts to a heading error of 0.62°. The yaw alignment is obtained to within 0.5° after 265 seconds. The figure also shows that the pitch and roll estimation is improved over the CKF estimate during the initial part of the trajectory. This indicates that the x and y-axis accelerometer bias states are also better estimated during alignment than the estimates obtained using the CKF.

In order to improve the integrity of the filter, two modifications to the algorithm were made. Firstly, the scale factor is not calculated for the first epoch. This condition was introduced since the initial INS position and velocity estimates may contain large errors due to incorrect initialisation, therefore the magnitude of the initial innovation may result in a poor

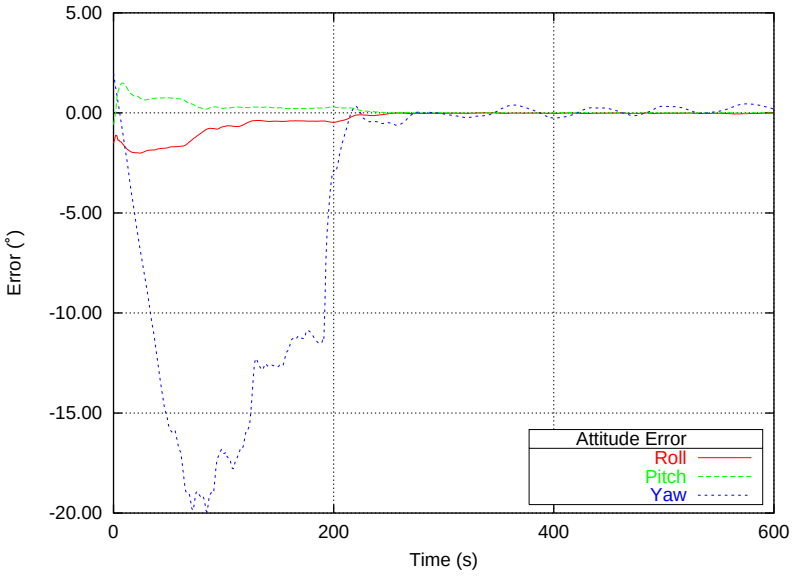
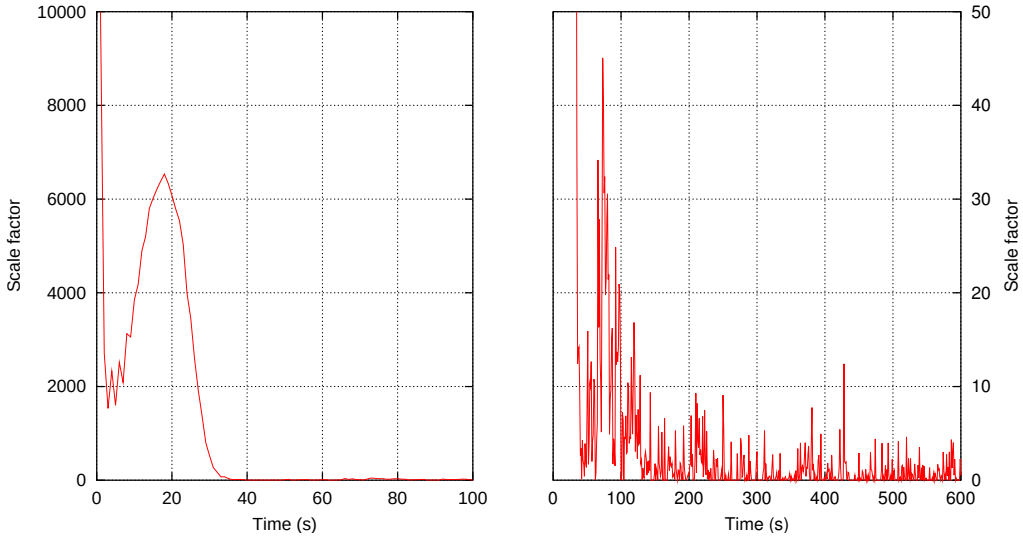


Figure 7.3: Process noise scaling attitude alignment

estimate of the scale factor. The initial position and velocity errors are greatly reduced after the first epoch, therefore the bias in the innovation sequence is more likely to be caused by the incorrect initialisation of the inertial sensor errors and attitude errors. The second condition is that the scale factor must not exceed 10,000. This condition was imposed to help ensure the stability of the filter. This is because previous analysis showed that if the covariance is increased too much, then too much weight can be given to the measurements resulting in very large incorrect bias estimates which can result in divergence of the filter.

The estimated process noise scale factors are shown in Figure 7.4, with the figure being split into two sections for clarity. Figure 7.4(a) shows the first 100 seconds of the scale factor estimates. From the figure it is shown that no scale factor is used for the first epoch, and at the second epoch, the maximum scale factor estimate is used. The application of the scale factor results in the next estimated scale factor to be greatly reduced with the



(a) Scale factor between 1-100 seconds (b) Scale factor between 100-600 seconds

Figure 7.4: Estimated process noise scale factors

scale factor estimated to be 2713. For the first 30 seconds of the dataset the scale factor estimates are all computed to be 1000 or greater. After this initial 30 seconds, the estimated scale factors reduce significantly and Figure 7.4(b) shows that the values are all computed to be less than 50.

The initial period of large scale factors estimates is caused by the large values in the innovation sequence indicating that there is some form of mis-modelling in the Kalman filter. Therefore, the algorithm uses the assumption that any mis-modelling is caused by the incorrect stochastic model for the INS. Therefore, as the process noise is increased, the innovation sequence shows that the system is being modelled better resulting in the scale factor estimates to reduce. After 130 seconds, with the exception of 1 epoch, the scale factors are computed to be less than 10. This indicates that the filter has adapted to allow the innovation sequence to

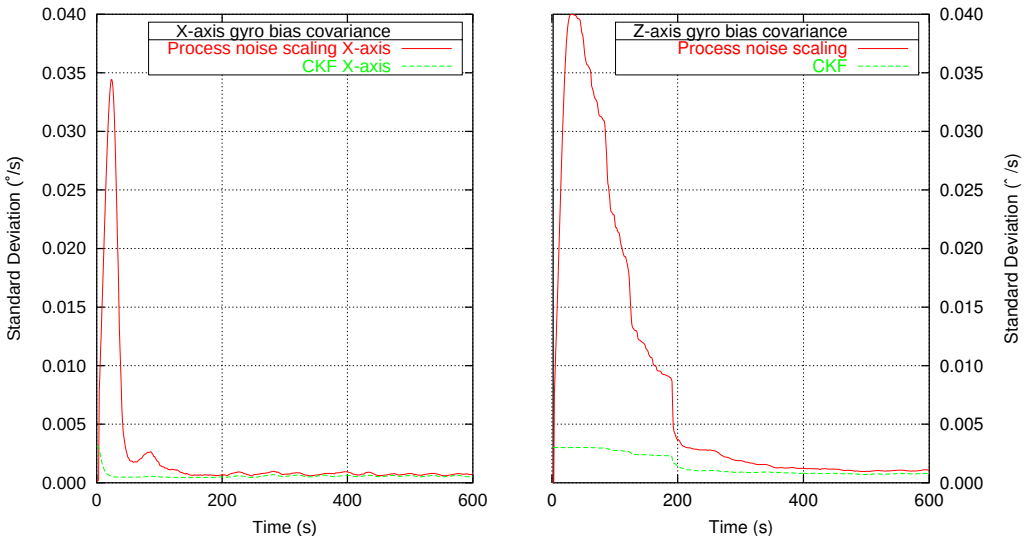
follow an approximately normal distribution of errors, again indicating that the system is properly modelled by the stochastic matrices. The values of between 1 and 10 are consistent with using the estimated innovation covariance approximation.

A potential problem with the process noise scaling filter is stability. The covariance scaling method caused divergence when the scale factor was overestimated. This resulted in the state estimates being too large, particularly for the bias states. The large bias estimates resulted in divergence since the bias estimate resulted in very large navigation errors which resulted in larger state estimates at the next Kalman filter iteration. This was specifically a problem using Inequality 7.2 since the scale factor estimate was very large. The new process noise scaling algorithm results in a much more stable estimate due to not scaling the state covariance at each epoch. Stability is improved over the covariance scaling approach due to the approximation used in moving from Inequality 7.5 to Inequality 7.6 in the derivation of the scale factor. This results in a scale factor that is more likely to be underestimated.

7.2.4 Covariance

Figure 7.5 shows the x and z-axis updated state covariance estimates for the process noise scaling filter, again compared to the CKF estimate using the empirically derived initial covariance matrix. The process noise scaling filter was initialised using a low initial covariance value for the initial matrix. Figure 7.5(a) shows that the standard deviation of the x-axis gyro bias is increased to 0.034°/s after 24 seconds due to the application of the scale factor on the process noise estimates. In other words the large

process noise estimate has increased the uncertainty of the x-axis gyro bias estimate whereas with the CKF, the covariance is decreased. A similar characteristic is shown for the z-axis gyro bias, although the maximum standard deviation is 0.04°/s due to the lower observability of the state.



(a) Bias process noise 0.01 (b) Bias process noise 1×10^{-4}

Figure 7.5: Process noise scaling updated state covariance during attitude alignment

From the figures, it would appear that the algorithm could be replaced by setting the initial covariance to 0.04°/s, however the analysis in §6.3.1 resulted in the optimum initial covariance for the bias states of 0.1°/s. There are two reasons why this is so. Firstly the initial innovation in the Kalman filter is usually mainly caused by the initial position and velocity errors. This results in a large initial innovation which can result in a large bias estimate which may actually increase the initial bias error. Secondly, the initial errors vary from switch-on to switch-on, therefore it is not possible to estimate an initial covariance that will always result in the

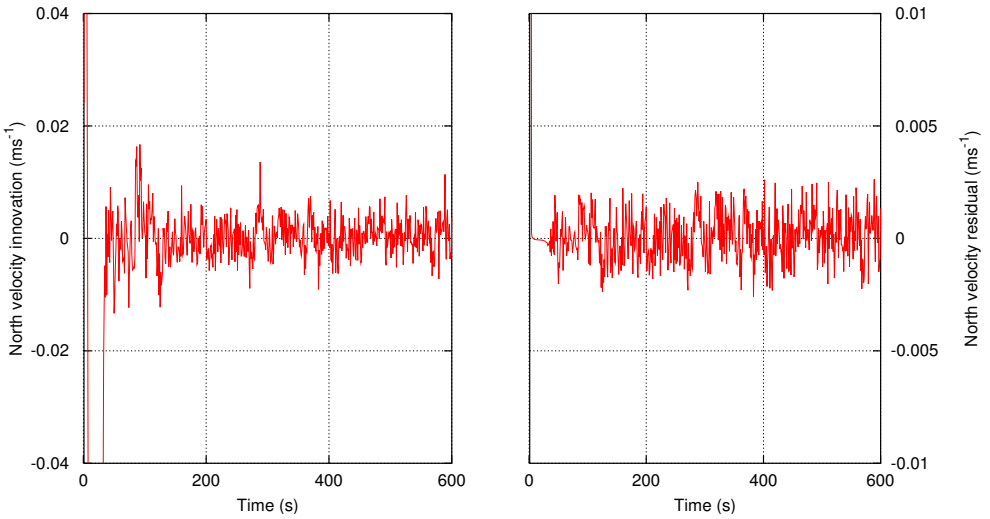
fastest time to alignment.

Another factor to consider with the process noise scaling algorithm is that the algorithm works by multiplying all the elements in the process noise matrix by the scale factor, S_k. When a deficiency in the state estimation is identified through the innovation sequence, the process noise is increased for all states. Consequently, the algorithm is not able to identify an individual state that has been poorly estimated. Intuitively, it is desirable to maintain the correct filter estimates once they have been initialised for the states that are strongly observable. For example the x-axis gyro bias which is shown for the conventional Kalman filter in Figure 6.3. Therefore, it may be possible to improve the time to alignment using an algorithm such as the adaptive Kalman filter examined in §7.3.

7.2.5 Innovation and residual sequence

Figure 7.6 shows the innovation and residual sequences for the north velocity error using the process noise scaling algorithm. The figure shows that the biases that exist under the conventional Kalman filter shown in Figure 6.6 have been removed. In other words, the algorithm has identified that there is a deficiency in the state estimation and has attributed this deficiency to the INS through increasing the process noise. For the initial 35 seconds, there is a significant bias in the north velocity innovation sequence. This is removed through using the scale factor to increase the process noise. As a result, the north velocity residual sequence has very small magnitude at the start of the dataset since the state is heavily weighted by the new measurement. Using the Lilliefors test for normality using 100 samples, the test is passed using data starting at 40 seconds. At the 95% confidence

level, the Lilliefors test statistic is computed to be 0.078 which is less than 0.088 indicating that the sequence approximates a Gaussian distribution.



(a) Innovation sequence (b) Residual Sequence

Figure 7.6: Process noise scaling residual and innovation sequences

Another factor that must be addressed in using the innovation sequence to scale the process noise is that it is important that blunders due to the measurements have been removed from the innovation sequence. In the decentralised filter used in this section, this means ensuring that there are no blunders in the GPS position and velocity estimates. Furthermore, the correct measurement noise for the GPS measurements is required in order to calculate the covariance of the innovation sequence. For the results obtained in this section, only carrier phase measurements are used with the ambiguities fixed. If an unresolved cycle slip were to cause the ambiguities to be reset, the GPS solution is obtained from the pseudorange measurements. It is important that the measurement model is adjusted in order to accommodate for the increased measurement noise, otherwise the

error is attributed to the INS by increasing the process noise scale factor.

7.2.6 Dependence on initial covariance

A significant advantage of using the process noise scaling algorithm is that it reduces the dependence on the initial covariance as considered for the conventional Kalman filter in §6.3.1. Table 7.1 shows the alignment times obtained using process noise scaling when using different initial covariance estimates. The table demonstrates that for the CKF, the time to alignment is highly dependent on the initial covariance of the bias states. The optimum value was a standard deviation of 0.01, with values larger than this resulting in a rapid increase in the time to alignment due to the large bias estimates. The lower covariance estimates all result in similar performance with a prolonged period of high dynamic manoeuvres required to resolve the biases.

The covariance scaling algorithm greatly reduces the dependence on the initial covariance. The initial standard deviation of 0.1 to 1×10^{-5} only results in a 1 second difference in the time to alignment. Surprisingly, the initial standard deviation of 0.1 did not result in large bias estimates as the CKF experienced. This is thought to be due to the scale factor increasing the process noise of all the states, resulting in comparatively less weight given to just the bias states. However, the alignment time was greatly increased for the initial estimate of 1.0. These results demonstrate that the new algorithm provides a robust method for improving the time to alignment, and reduces the dependence of the initial covariance estimate.

	Initial standard deviation (°/s, ms^{-2})					
	1.0	0.1	0.01	1×10^{-3}	1×10^{-4}	1×10^{-5}
CKF (s)	>1000	>1000	453	551	556	556
Process noise scaling (s)	730	258	265	266	266	266

Table 7.1: Processing noise scaling alignment times for different initial bias state standard deviations

7.2.7 Centralised filtering

The process noise scaling algorithm is more complex to implement for the centralised filtering structure than for the decentralised structure. This is because the number of observations in the centralised filter (i.e. the double differenced satellite range observations) varies depending on the number of satellites available. Therefore, the expected sum of the square of the innovations in Inequality 7.2 will depend on the number of observations used in the filter.

The centralised filter was adapted in KinPos^i to include the covariance scaling algorithm. To accommodate the different number of measurements and measurement types, the approximation for the aligned innovation covariance was estimated for each observation for each type of measurement. The estimate of the innovation variance was then formed from the sum of the variances for each observation. A more thorough statistical model could be developed using the models described in §3.6.2 to allow for the variation in measurement noise on each of the measurements.

The process noise scaling algorithm using the centralised filtering structure results in an alignment time of 263 seconds. This is 5 seconds slower

than the decentralised filter using the same initial covariance matrix. The small difference in the time to alignment is considered to be negligible.

Another factor to consider is that it was mentioned in §6.2 that the decentralised filter is used in this section because the large initial INS errors can actually degrade ambiguity resolution. For the simulated data, the ambiguities were resolved after 1 epoch, however the initial position was specified to within approximately 10 centimetres. For real data, if too much weight is initially given to the INS, the estimated position will drift with the INS. The adaptive algorithms should reduce this problem because, in the case of large initial errors, the INS process noise is increased so that the position is obtained primarily from the GPS measurements. This is considered further in Chapter 8.

7.2.8 Navigation performance

So far, it has been demonstrated that the process noise scaling algorithm provides a significant performance improvement in the time it takes to initialise the bias estimates and align the INS attitude. This section examines the effect that the scaling algorithms have on the navigation errors once the INS is aligned.

Figure 7.7 shows the navigation errors for the position, velocity and attitude using the conventional Kalman filter and the process noise scaling algorithm. The navigation performance of the fading memory covariance scaling algorithm from Hu et al. (2001) given by Inequality 7.1 is also shown. The fading memory algorithm is examined in this section as the algorithms are being tested when the system is aligned and the Kalman gain has reached steady state.

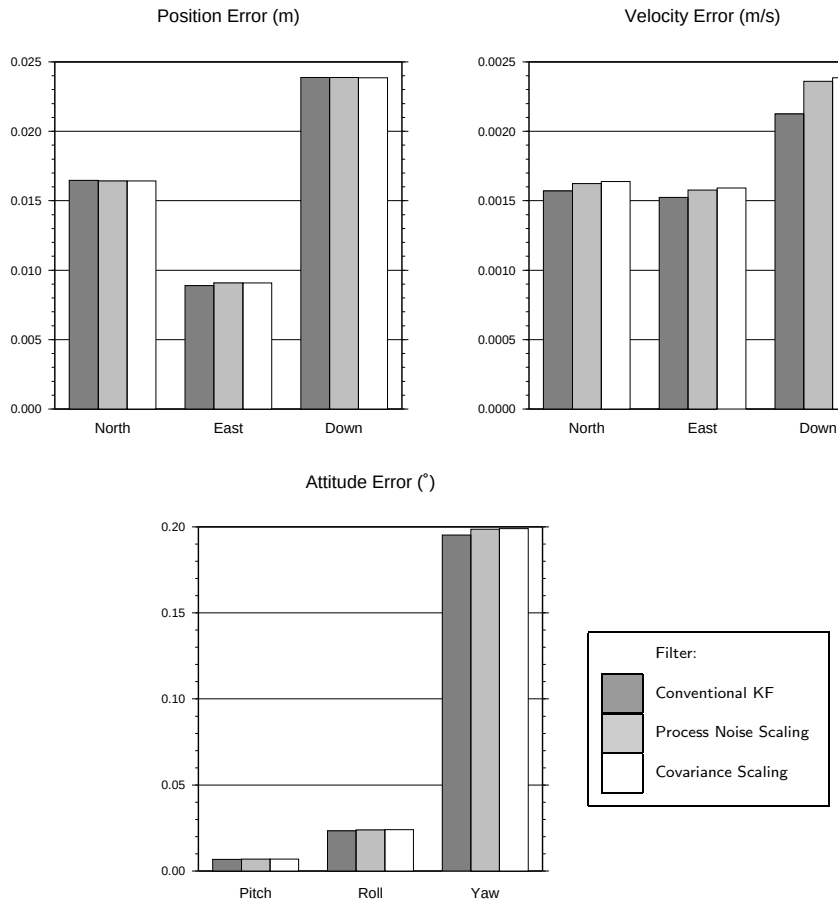


Figure 7.7: Covariance and Process Noise scaling RMS errors

The figure shows that the RMS position errors are at the centimetre level for the CKF. The mean position errors are at the level of a few millimetres for the trial. This is because the atmospheric errors cancel over the short baseline. The position error is thought to be larger in the North direction due to the latitude at which the data was simulated. This is a result of the current orbit patterns and satellite geometry of 55° inclination (Meng et al., 2003). The Doppler errors are at the millimetre level which is thought to be of higher accuracy than expected for real data (see §8.2.5). The roll and pitch errors are less than 0.03° as they are continually observable, but the yaw error is significantly larger at 0.20° for the CKF due to the lower observability.

The figure shows that the adaptive algorithms both result in a small increase in all the navigation errors over the CKF except for a small decrease in the north axis position error of less than 1mm. The largest percentage increase in the navigation errors is experienced by the velocity error which increases by 12.3% in the down axis for the covariance scaling filter. This is however, still an increase of less than 1mm/s.

It was anticipated that the process noise scaling filter could result in a slight deterioration of navigation accuracy because of the approximation used to form the innovation covariance. The approximation is formed from a sample of data taken from the innovation sequence when the system is aligned. The covariance approximation is therefore only valid for the sample of data that it was obtained from. The reduction in navigation performance of the process noise scaling filter is however, only at the millimetre level for position and velocity and less than 1.5 arc minutes in attitude.

The covariance scaling filter was anticipated to improve the navigation performance by providing improved modelling of the inertial errors. However, the algorithm results in a small deterioration in navigation accuracy, which is at the millimetre level for position and velocity and less than 2 arc minutes in attitude.

7.3 Adaptive Kalman filtering

The adaptive Kalman filter is described in §4.3.3. This section describes the use of the innovation and residual based adaptive process noise algorithms for the alignment and initialisation of the INS errors. Therefore, the

adaptive algorithm is used to identify the process noise matrix as opposed to identifying the measurement noise matrix. The adaptive Kalman filter estimate of the measurement noise is not considered in this chapter because the large low cost inertial navigation sensor errors are the dominant error source at the initialisation stage. Use of the measurement noise adaptive estimate would result in the inertial navigation sensor errors being incorrectly attributed to the GPS measurements. The adaptive Kalman filter algorithm for process noise estimation is termed *adaptive-Q* filtering for the rest of this thesis.

7.3.1 Innovation based adaptive-Q

The innovation based adaptive-Q estimate is formed using the state covariance approximation in Equation 4.33 substituted into Equation 4.30. The process noise estimate is computed after the Kalman filter update and it is used to form the covariance prediction at the following epoch. The adaptive estimate requires an initial window length of observations to be accumulated using conventional Kalman filtering to form the approximation for the state covariance. The window length used in this section is 20 epochs. Initially in this section, only the diagonal elements of the estimated process noise matrix are used.

Figure 7.8 shows the innovation based adaptive-Q estimate using two different initial process noise matrices. The first initial process noise matrix considered is the matrix used in §6.3.1 which was identified as the estimate that resulted in the best navigation performance using the CKF. This is labelled as the normal process noise in the figure. The figure shows that the z-axis gyro bias estimate formed using this process noise matrix is slow

to converge to the correct estimate. The bias estimate only converges to within 0.01°/s after 473 seconds, which is slower to converge to the correct value than the CKF.

In order for the bias estimate to converge at all using the normal navigation process noise, the adaptive-Q filter was initialised using the empirically estimated initial covariance matrix that was identified in §6.3.1. When the filter is initialised with a smaller initial covariance for the bias states, the process noise of the bias states converges to a small value which results in the bias states not being updated by the new observations.

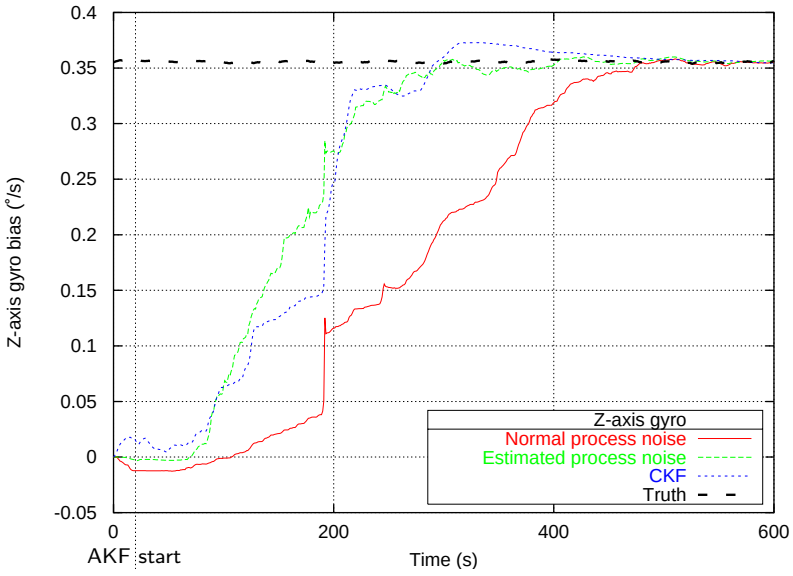


Figure 7.8: Innovation based adaptive-Q filter z-axis gyro bias alignment

The adaptive-Q estimate shown in Figure 7.8 is slow to converge to the correct bias estimate due to the recursive nature of the adaptive-Q algorithm. The adaptive-Q algorithm uses the state covariance approximation given by Equation 4.33 which uses the Kalman gain to convert the innovation covariance approximation to the state covariance approximation. Therefore, the algorithm uses the information obtained from

the first 20 epochs using the CKF algorithm to form the process noise estimate. Because the initial process noise elements correspond to those expected when the INS is aligned, the resultant process noise estimate is based on this information, hence the process noise is underestimated. As more information is obtained from the innovation sequence, the process noise estimates increase. For example, in the figure at approximately 190 seconds at the beginning of the alignment manoeuvre, there is a significant increase in the z-axis gyro bias due to the large innovation. However, the bias estimation is still slower to align than the CKF because the initial covariance and process noise estimates do not accurately model the initial stochastic properties of the INS errors. Therefore the adaptive-Q algorithm is sensitive to the initial covariance and the initial process noise values used. The sensitivity to these values is investigated further §7.3.4.

As a result of the poor bias estimates obtained using the normal process noise matrix for the INS during navigation, the algorithm was changed so that the initial process noise matrix is formed using estimates of what the initial process noise elements are likely to be. The adaptive-Q algorithm is then used to adapt the process noise to reduce the process noise as the states become aligned. This is different to the process noise scaling algorithm where the algorithm was used to automatically increase the process noise.

Figure 7.8 shows the adaptive-Q estimate obtained using the estimated initial process noise matrix. This figure shows that the z-axis gyro bias is resolved to within $0.01^\circ/s$ at 340 seconds which is 45 seconds faster than the optimised CKF. The yaw alignment converges to within 0.5° after only 264 seconds which is 189 seconds faster than the CKF and 1 second faster than the process noise scaling filter. It is significant that the yaw alignment

is achieved to within the 0.5° threshold after 264 seconds even though the z-axis gyro bias is not obtained to within 0.01°/s until 340 seconds. In the process noise scaling filter, the z-axis gyro bias was obtained to within 0.01°/s after only 210 seconds. This indicates that the adaptive-Q algorithm has adapted the process noise elements individually, unlike the process noise scaling algorithm where all the elements are increased by the same amount. In the adaptive-Q filter, the yaw state process noise is estimated so that the yaw state responds to the high dynamic manoeuvre during alignment. The z-axis gyro bias process noise is comparatively underestimated which results in a longer alignment time.

One of the potential advantages of using the adaptive-Q process noise estimate is that it provides a fully populated variance-covariance matrix. However, use of the full variance-covariance matrix resulted in the filter becoming divergent after approximately 200 seconds. This is because the number of unknowns that the adaptive-Q filter is attempting to estimate is increased significantly when including the off-diagonal elements. The full variance-covariance matrix results in a more stable performance when the INS errors are resolved, i.e. when there is less variation in the estimated process noise. Analysis in Mohamed (1999) identified that the correlations in the estimated process noise reduce when a longer estimation window on the innovation sequence is used, and that the navigation performance is similar to that of just using the diagonal elements. Therefore, although it is theoretically appealing to use the full process noise matrix, only the diagonal elements of the process noise are considered for the adaptive-Q filters in this thesis due to the more stable performance.

Another important result of using the adaptive-Q formulation is the

effect on the innovation sequence. The adaptive-Q filter described results in the innovation sequence sharing similar properties with the innovation sequence from the process noise scaling algorithm in §7.2.5 in that the biases are removed from the sequence. For the first 20 seconds during conventional Kalman filtering, there is a bias in the innovation sequence that increases to 27cm/s for the north velocity. When the adaptive filter begins, the bias reduces to 6cm/s after 40 seconds and after 45 seconds there is no distinguishable bias in the innovations. The Lilliefors test statistic computed for the innovation sequence between 45 and 145 is 0.044 which indicates that the sequence during this time period approximates a normal distribution. Therefore, the adaptive-Q algorithm has adapted to make the innovation sequence consistent with the estimated covariance which is the basis for the formulation of the adaptive-Q algorithm. Therefore, in a similar way to the process noise scaling filter, it is important that blunders are removed from the GPS measurements to avoid the errors caused by the GPS measurements resulting in the process noise for the INS being increased incorrectly.

7.3.2 Residual based adaptive-Q

Figure 7.9 shows the residual based adaptive-Q estimate of the z-axis gyro bias. The approximation to the state correction covariance for the residual based adaptive-Q filter is given by Equation 4.32. The figure shows the z-axis gyro bias estimate using the same initial covariance and the same two initial process noise matrices that were defined in the previous section. The bias estimate in the figure formed using the normal navigation process noise shows that the residual based adaptive-Q bias estimate takes longer

to converge than the equivalent innovation based estimate. The filter takes 793 seconds for the z-axis gyro bias to converge to within 0.01°/s. Again this shows that the initial covariance and process noise that are used to initialise the filter are important in the convergence of both the innovation and residual based adaptive-Q algorithms.

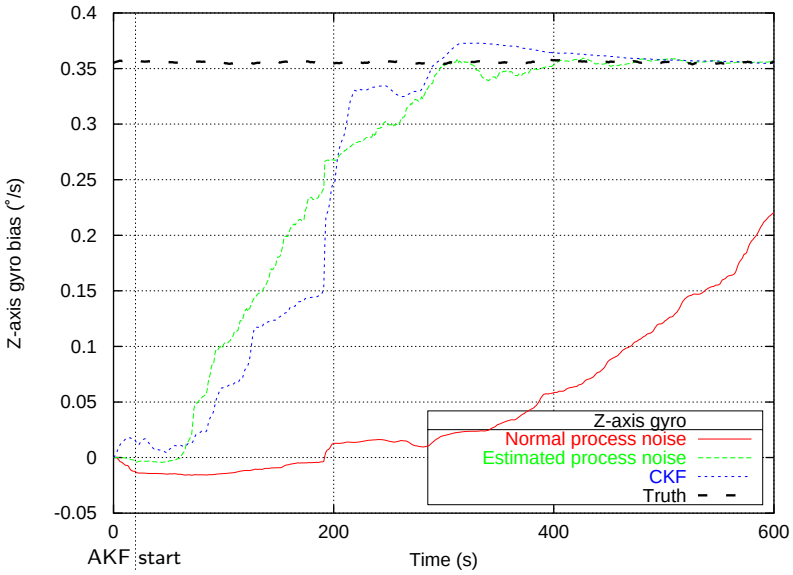


Figure 7.9: Residual based adaptive-Q filter z-axis gyro bias estimate

Figure 7.9 also shows that the estimated initial process noise matrix results in a very similar performance to the innovation based estimate. In fact, the yaw estimate is obtained to within 0.5° after 283 seconds, which is only 19 seconds slower to align than the innovation based filter. The z-axis gyro bias is obtained to within 0.01°/s after 370 seconds which is 30 seconds slower than the innovation based adaptive-Q filter, but still 15 seconds quicker than the CKF. Therefore both the innovation and residual based algorithms show the potential to provide a more rapid initialisation and alignment than the CKF algorithm.

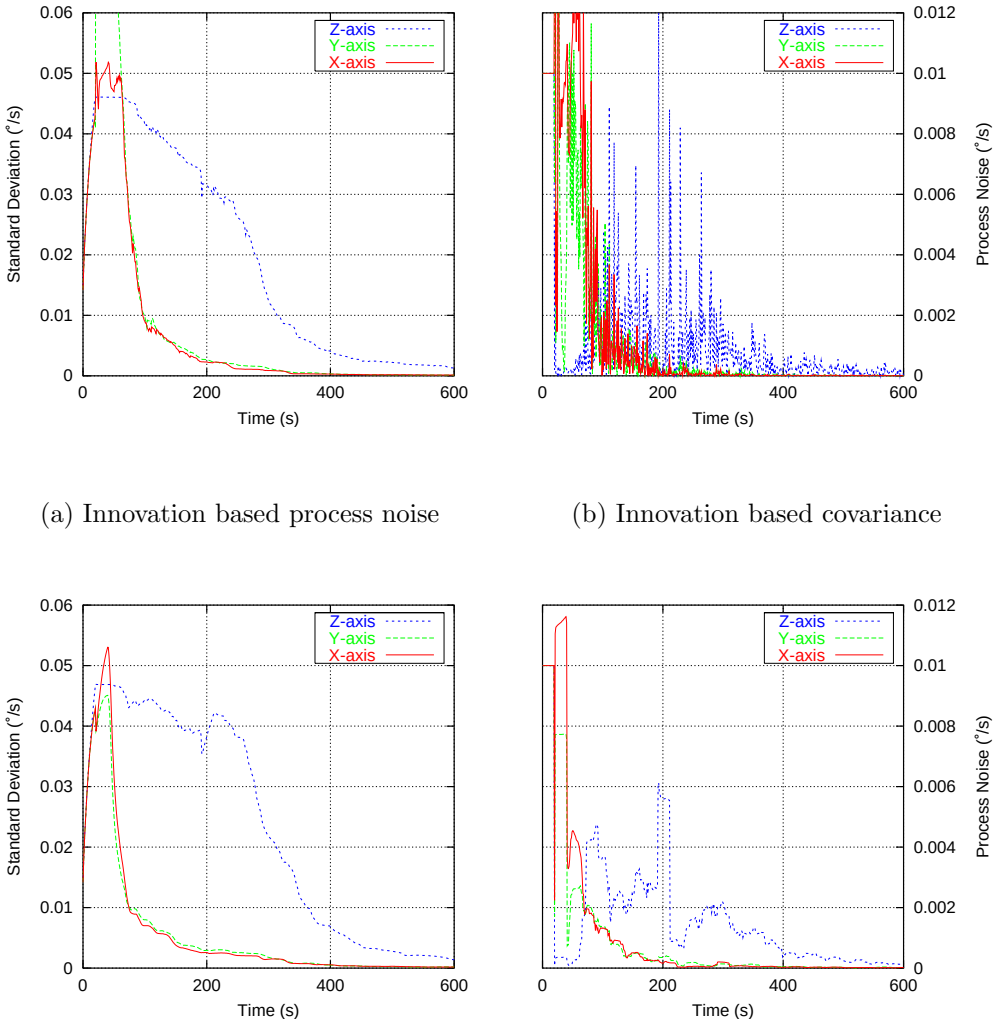
The residual based adaptive-Q algorithm was also tested using the full

variance-covariance process noise matrix. However, this again resulted in filter divergence after approximately 200 seconds which is the same as the innovation based estimate. Therefore only the diagonal elements of the estimated process noise are used in the residual based adaptive-Q filter. From the results obtained from the residual and innovation based filters it is clear that both filters need to be initialised with suitable covariance and process noise matrices.

7.3.3 Covariance

Figure 7.10 shows the covariance and estimated process noise elements for the gyro bias states using the innovation and residual based adaptive-Q algorithms. From the figures it is clear that the innovation and residual based estimates show different characteristics for the estimated process noise.

The residual based filter estimates the x-axis bias process noise to be approximately 0.015°/s at the first adaptive epoch (i.e. after the initial 20 epochs where the CKF is used) with the y-axis bias process noise estimated to be approximately 0.008°/s. The residual based estimate follows what the bias estimates are expected to be since initially the x and y-axis bias estimates are -0.28°/s and -0.18°/s respectively. Both states are observable at the start of the dataset with the state correction expected to be larger for the x-axis bias since the initial error is larger. As the state correction sequence is used to form the process noise matrix, the initial process noise for the x-axis is larger than for the y-axis, hence the estimates shown. However, at the start of the dataset, the z-axis process noise estimate is the smallest even though the initial bias error is the largest at 0.35°/s. This



(a) Innovation based process noise

(b) Innovation based covariance

(c) Residual based process noise

(d) Residual based covariance

Figure 7.10: Adaptive-Q filter gyro bias process noise and covariance

is considered later in this section. The figure shows that the process noise is individually estimated for each state, unlike the process noise scaling filter where all the process noise elements are increased by the same amount.

For the innovation based filter, the estimates vary at a much higher frequency than the residual based filter. Initially at the first adaptive epoch, the y-axis gyro bias is estimated to be 0.15°/s with the subsequent 20 epochs predominantly estimated to be less than 0.01°/s. The initial

estimate of 0.15°/s increases the covariance significantly to 0.154°/s. This actually deteriorates the y-axis bias estimate because too much weight is given to this state. From the innovation covariance for the x and y gyro bias states is is shown that the covariance reduces slower than for the residual based estimate. This is because the x and y-axis bias states are not as accurately obtained as the residual based estimates. Similarly the z-axis bias is quicker to reduce than the residual based estimate because the z-axis bias estimate is better. Therefore the filter is adapting depending on the actual errors. After approximately 400 seconds when both the residual and innovation estimates are obtained to approximately the same accuracy, the covariance estimates converge.

The main characteristic that is identified in Figure 7.10 is that the values for the x and y-axis estimates reduce as more data is made available to the filter. Conversely, the z-axis estimate increases after approximately 70 seconds. This is caused by the observability of the states that are being estimated. For the z-axis gyro, during the period of low dynamics, the state correction sequence is of small magnitude due to the low observability. Therefore, the estimated process noise is small for this state. However, when the small dynamic manoeuvre occurs at approximately 70 seconds, the state correction sequence increases and hence the process noise increases. Similarly, as the correct bias estimates are obtained for the x and y-axes, the state correction sequence reduces and therefore the process noise reduces. This shows that the estimation is recursive which identifies that for the alignment problem, the initial values for the covariance and process noise matrices are critical for the performance of the filter.

The covariance estimates for the gyro bias states in Figure 7.10 both

show that during the initial 20 epochs using the CKF, the covariance increases from the initial standard deviation of 0.1°/s to approximately 0.45°/s for all states (the actual values vary slightly due to the observability of the states). The initial increase should be the same as initialising the bias state covariance with a standard deviation of 0.45°/s but this is not the case. This is because the initial few innovations in the innovation sequence are dominated by the initial position and velocity errors. If a large initial covariance for the bias states is used, the initial error is incorrectly absorbed by the bias states, potentially resulting in large bias errors. By increasing the initial covariance after the first few epochs using the process noise, the filter provides a much better estimate of the biases. This is considered further in the following section.

7.3.4 Dependence on initial covariance and initial process noise

One of the characteristics required for the alignment filter is to reduce the requirement for accurate *a priori* estimates for the elements in the process noise matrix. So far, the analysis has indicated that the initial process noise and covariance information used in the filter is potentially more critical than that used for the CKF. Table 7.2 gives the alignment times for the adaptive-Q formulation given different initial covariance estimates for the gyro and accelerometer bias states. The initial process noise matrix used for the CKF initialisation is the process noise that was identified for normal navigation performance. In other words, Table 7.2 shows whether the standard navigation process noise matrix can be used to initialise the

filter by changing the initial covariance. The dependence of the initial bias states is investigated here due to their weak observability.

	Initial covariance ($^\circ$/s, ms^{-2})		
	0.1	0.01	0.001
Innovation based (s)	271	413	>1000
Residual based (s)	378	700	>1000

Table 7.2: Adaptive-Q alignment times for different initial bias state covariance

The table shows that the adaptive-Q filters are highly sensitive to the initial covariance used. For the initial covariance of the bias states, the estimate of 0.001 resulted in the filter not being aligned even after 1000 epochs. In fact, the filter had only estimated the z-axis bias to be 0.03°/s after 1000 seconds, whereas the actual bias is approximately 0.35°/s. This indicates that the filter will take far longer than 1000 seconds to become aligned. The reason this occurs is that initial bias state corrections using the CKF were very small. Therefore, the process noise estimates are small and the filter continues to underestimate the state correction.

The larger initial covariance estimate of 0.01 for the bias states resulted in the z-axis bias shown in the previous sections when using the navigation process noise. The initial estimate of 0.1 provided the fastest alignment time for the adaptive filters. An initial estimate of 1.0 resulted in the filter becoming divergent as too much weight was given to the bias states which resulted in very large bias estimates which quickly make the filter unstable. Therefore, unlike the process noise scaling filter, the adaptive-Q algorithms

are more sensitive to the initial covariance matrices used than the CKF. Furthermore, from the results obtained, it is clear that the algorithm is slow to increase the process noise but is capable of reducing the estimate relatively rapidly.

Table 7.3 shows the alignment times for the adaptive-Q filters when using different initial process noise elements for the bias states. A scale factor was multiplied to the standard navigation process noise matrix as it is assumed that all states will have a larger process noise at the beginning of the dataset. The table shows that in addition to the sensitivity of the initial covariance, the filter is also sensitive to the initial values used for the process noise. For the innovation based filter, the most rapid time to alignment is 264 seconds and for the residual based filter the fastest time to alignment is 279 seconds using a process noise matrix of 100 and 1000 times larger respectively than the standard navigation process noise matrix.

	Process noise scale factor					
	10000	1000	100	10	1	0.1
Innovation based (s)	347	291	264	472	413	345
Residual based (s)	341	279	283	686	700	702

Table 7.3: Adaptive-Q alignment times for different initial process noise

The results obtained from using different initial process noise matrices are consistent with the results obtained in Busse et al. (2002). Busse et al. used adaptive Kalman filtering for formation flying missions in low Earth orbits. Busse et al. observed that adaptive routines were always able to identify the process noise if the initial estimate was larger than the actual

value, but the adaptive routines were not always able to converge to the correct value if the initial guess was underestimated. The analysis in this section has shown that the algorithm does not increase the process noise rapidly enough to be useful for the alignment problem. The performance of the alignment algorithm is reduced when the initial covariance and process noise is too high or low. Therefore although the algorithm is potentially able to improve alignment over the CKF, its use is restricted because of this problem.

7.3.5 Optimising window length

A further consideration for the adaptive-Q algorithms is the effect of the window length used to form the approximation to the covariance of the state correction sequence. Mohamed (1999) indicated that the number of epochs of observations used to form the approximation should be at least the number of states that are to be estimated. In this instance, the number of states is 15 so as an approximate guide, the window length should be larger than 15. Table 7.4 shows the effect that the window length has on the alignment time for the innovation and residual based adaptive-Q filters using the optimised initial covariance and process noise from §7.3.1. From the table, the fastest alignment times are obtained using a window length of 30 or 40 observations using the innovation based algorithm.

One of the factors for the INS alignment is that using a longer window length increases the initial period for which the filter uses the CKF when accumulating the observations. For example, the filters using a window of 60 epochs will require an initial 60 epochs of processing using the CKF at the start of the dataset. However, the approximation to the innovation or

	Initial covariance ($^\circ$/s, ms^{-2})					
	10	20	30	40	50	60
Innovation based (s)	349	264	247	245	346	342
Residual based (s)	342	284	282	282	279	272

Table 7.4: Adaptive-Q alignment time for different window lengths

state correction covariance should be improved with more data contained inside the estimation window. Another consideration is that the approximation will not reflect a change in the stochastic properties of the filter as rapidly when using a longer estimation window. Therefore the window needs to provide a balance between these factors. From the table it is shown that the window lengths of 30 and 40 show a good balance between these factors for the innovation based adaptive-Q filter, with the alignment time being increased for shorter or longer estimation windows.

For the residual based filter, the alignment times are improved when using the longer estimation windows. This is expected to be because the CKF is used for a longer time which increases the state covariance. The residual based filter tends to underestimate the process noise during alignment which results in the covariance being reduced too quickly. Therefore, the longer estimation windows compensate for the smaller estimated process noise.

7.3.6 Centralised filtering

The adaptive-Q algorithms can also be implemented using the centralised filtering structure. The residual based algorithm is simpler to implement

in the centralised filter than the innovation based algorithm because the number of observations is always constant. For the innovation based approximation, the number of observations varies due to the different number of satellite observations that are available at each epoch. There are two situations that need to be considered for the innovation based filter when satellites come into view. The first is that the satellite is a new observation. In this instance the satellite has not been recently viewed and therefore only *a priori* stochastic information is available for the observation. In this instance, a standard approximation to the innovation covariance for the satellite is assumed. The second situation when a satellite comes into view is when it has been momentarily blocked by an obstruction. In this instance, the previously accumulated residual information is used to form the innovation covariance approximation as it is assumed that the approximation is still valid.

The adaptive-Q algorithms were implemented in KinPos^i using the method described. For the innovation based adaptive-Q filter, the alignment time was 266 seconds which is 2 seconds slower than the decentralised innovation based filter. This is considered to be a negligible difference. The z-axis gyro bias estimate was obtained to within 0.01°/s in 264 seconds compared to 340 seconds for the decentralised filter. This large difference was obtained because for the decentralised filter, the estimate only just failed the 0.01°/s criterion. Similarly the residual based centralised adaptive-Q filter resulted in an alignment time of 289 seconds which is 6 seconds slower than the decentralised filter.

The results obtained using the centralised filtering structure are particularly important for the adaptive-Q algorithm as the centralised and decen-

tralised filters contain different measurement models. One of the factors that needs to be considered with the adaptive-Q algorithm is that in the derivation of the algorithm, it is assumed that the the measurement noise matrix is known. If there is a deficiency in the measurement noise model, it will be incorrectly absorbed by the process noise estimate. The measurement noise for the centralised filter is formed using *a priori* estimates of the noise of the double differenced observations. For the measurement simulation, this model does not vary significantly for different elevation angles which may be expected in a real world environment. Therefore the assumption that each observation is of equal weight for the simulation trials is valid. This should be considered when using real GPS observations. Possible weighting schemes for the GPS measurements were described in §3.6.

For the decentralised filter, the measurement noise matrix is formed using the position and velocity variances from the GPS filter covariance matrix. The GPS filter covariance elements are used because the GPS position and velocity cannot be assumed to be constant due to factors such as changes in the geometrical strength of the satellite constellation, and most importantly the types of GPS observation that were available to form the position estimate. Because the results obtained for the centralised and decentralised filters are very similar, it is assumed that the measurement model for the decentralised filter is a good approximation of the true measurement noise. This is examined in more detail for real data in §8 where the change in position and velocity accuracy is more significant.

7.3.7 Navigation performance

So far, the adaptive-Q algorithm has only been assessed for its suitability for the alignment of a low cost INS. To date, the adaptive-Q algorithm has been examined for GPS and INS in Mohamed (1999) with the adaptive filter resulting in improved orientation accuracy for a tactical grade INS. Figure 7.11 shows the navigation performance obtained from the innovation and residual based adaptive-Q filters compared to the CKF performance.

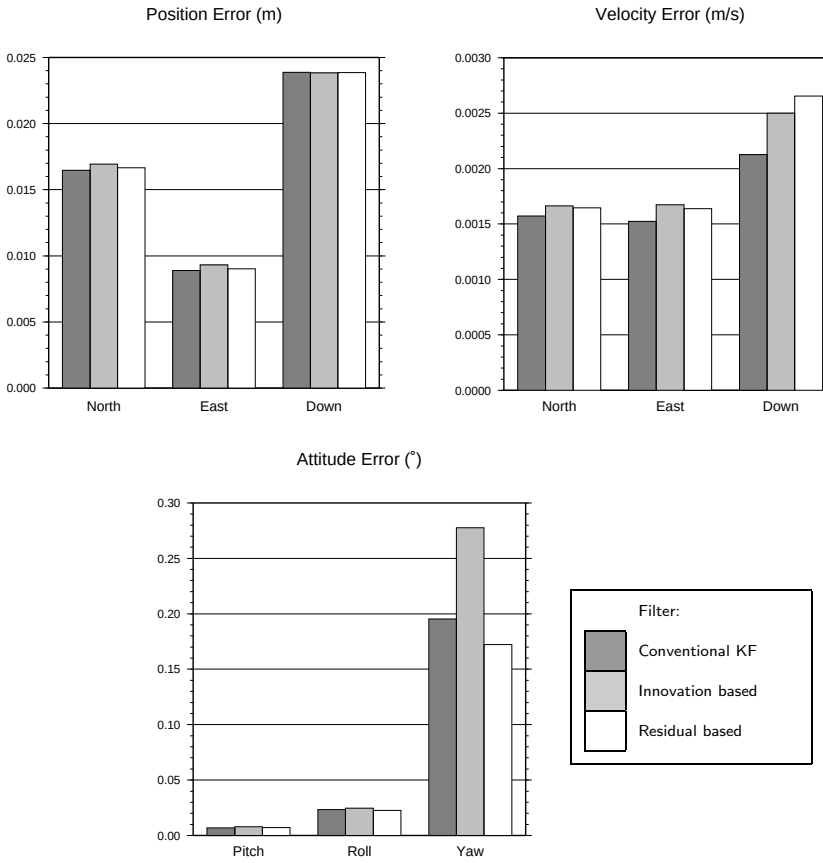


Figure 7.11: Adaptive-Q Kalman filter RMS errors

From the graphs it is clear that there is a small performance decrease in most of the navigation parameters for both the innovation and residual based filters. However, it is shown that for position, the differences are at the millimetre level and for velocity, the differences are at the sub-

millimetre per seconds level. The most significant difference is for the yaw error where the innovation based estimate is 0.08° (5 arc min) worse than the CKF and the residual estimate is an improvement of 0.02° (1.5 arc min). This is thought to be because the residual based filter tends to estimate the process noise to be smaller than the innovation based filter. This resulted in a slower alignment and initialisation but results in improved navigation performance.

Another consideration of the adaptive-Q filters is that they potentially provide a self-tuning algorithm for GPS and INS integration. Therefore the process noise estimate that the adaptive filter converges to may be useful for tuning the CKF algorithm. This was attempted using the average errors obtained from the end of the processing run when the filter is aligned. This resulted in the process noise estimates being much too small for use as *a priori* estimates in the CKF. States such as the yaw error drifted from the true value because the process noise was set too small. This is thought to be because the adaptive algorithm increases and decreases the process noise estimates depending on the new information provided by the estimated state correction covariance. Therefore, for example, during turning manoeuvres, the process noise estimates are increased. Taking an average of the process noise does not provide a good estimate of the process noise. The use of adaptive-Q filtering for identifying the *a priori* process noise for the CKF therefore requires further investigation.

7.4 Multiple Model Adaptive Estimation

The final adaptive Kalman filtering technique for alignment examined here is the multiple model approach as described in Section 4.3.4. The multiple model filter is generally not considered for navigation applications due to the large computational burden imposed from running the multiple filters. However, continued improvements in computer processor technology now allow multiple models to be genuinely considered as a potential method for an adaptive Kalman filtering algorithm.

7.4.1 A six model adaptive filter

The first trial considered here uses six Kalman filters all operating using different process noise matrices. Six models were used since it provides the filter with a broad range of potential filters to cover the change in stochastic properties of the INS during alignment. The process noise matrices are formed using the process noise matrix from §6.3 with the bias process noise of 1×10^{-4}. Again, this process noise matrix is used since it provided the best navigation performance in §6.3. Each filter is formed using a different scale factor, S_n, for the n^{th} Kalman filter to scale the process noise matrix. Therefore this implementation of the multiple model filter is similar to the process noise scaling filter in §7.2.2, but uses a different method to identify the scale factor. This method also differs in that the state estimate from each filter can be used to form the weighted combination of the state at each epoch.

The scale factors for the six individual Kalman filters are defined as S_n = 10000, 1000, 100, 10, 1 and 0.1. The scale factors are predominantly

larger than the standard Kalman filter of $S_n = 1$ since the inertial errors are expected to be larger during alignment. Since the multiple model filter is required to remain adaptive during the processing run, a minimum probability condition is imposed on the probability $p_n(k)$ for each model. If the computed probability is below this threshold, the probability is replaced with the threshold value and the probabilities for the other models are normalised in order to ensure that the sum of the probabilities is equal to unity. The minimum probability threshold used is $1{\times}10^{-3}$. Figure 7.12 shows the estimate of the z-axis gyro bias using the multiple model filter.

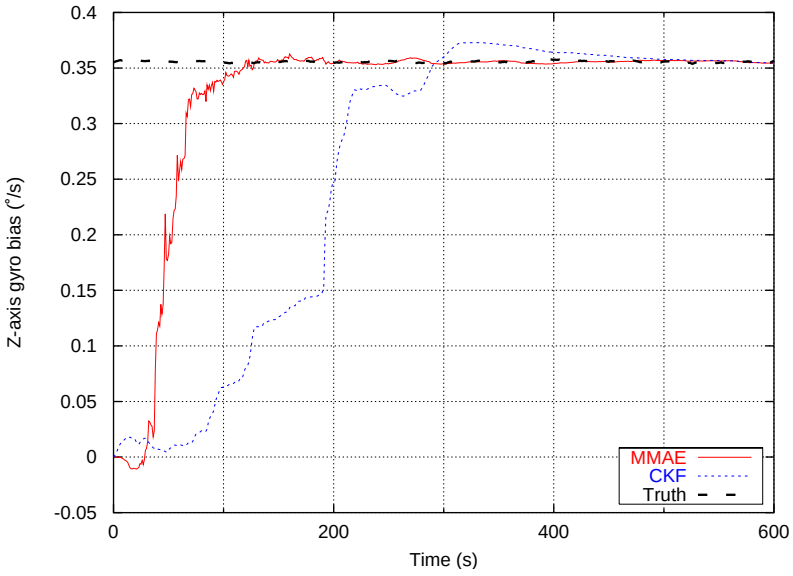


Figure 7.12: 6-model MMAE filter z-axis gyro bias alignment

The MMAE estimate shown in Figure 7.12 provides the fastest alignment achieved by any of the algorithms considered in this chapter. The figure demonstrates that the MMAE algorithm is able to adapt to use different process noise information during initialisation and alignment. From the figure it is shown that during the initial period of low dynamics, the z-axis gyro bias estimate is continually being updated which indicates that

the filters with the larger process noise are being used. After only 114 seconds, the filter estimates the bias to within 0.01° which is still during the period of low dynamics. The yaw alignment is achieved to within 0.5° after 127 seconds. This indicates that the initialisation and alignment is achieved while the vessel is still in the period of low dynamics. Therefore the algorithm removes the requirement for the vessel to undergo the period of high horizontal acceleration manoeuvres in order to align the INS.

From Figure 7.12 it is observed that the filter is constantly adapting to the new information provided by the innovation sequence. At the beginning of the processing run, the filter with the largest process noise scale factor is identified as the filter with the largest probability. In fact, the estimated conditional probability for this filter is 1 to 10 decimal places. This is why it is necessary for the threshold probability condition to be imposed to ensure that this does not result in this being the only filter used for the entire processing run. The filter continually adapts the process noise so that there is a smooth transition to the correct bias estimate. The figure shows that at 114 seconds when the bias estimate is obtained to within 0.01°/s, the bias estimate follows the true bias estimate indicating that the process noise is reduced.

To show how the MMAE estimate was formed, Figure 7.13 shows the conditional probabilities that form the weights for each individual Kalman filter. For example, at 300 seconds, the Kalman filters formed using the process noise scale factors of 0.1 and 1 both have a probability of approximately 0.18. The filter using the process noise scale factor of 10 has a probability of 0.64. The conditional probabilities for the remaining models are calculated to be less than 2×10^{-5} and consequently are not visible on

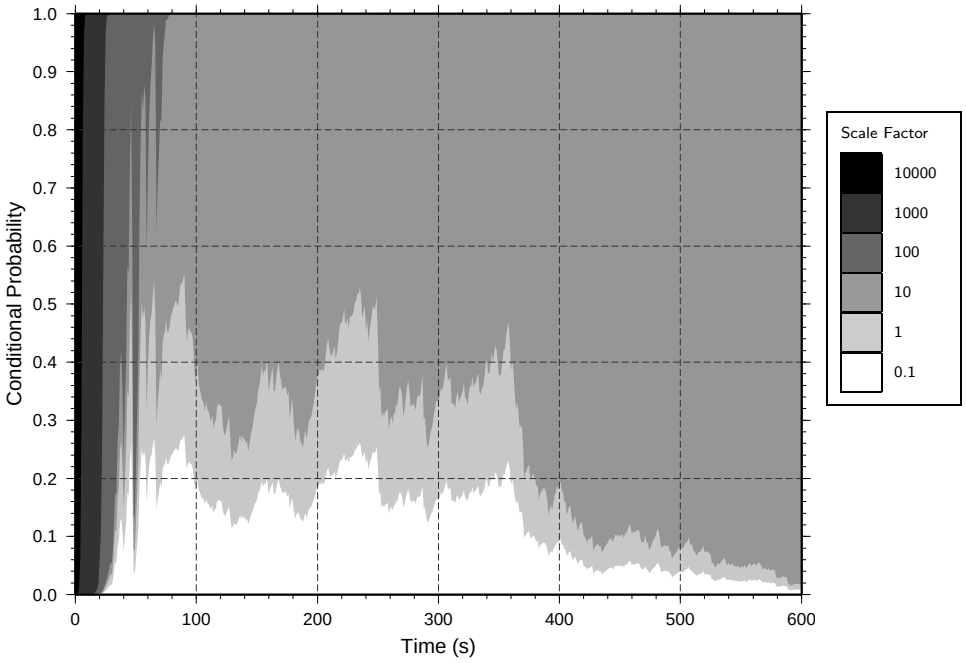


Figure 7.13: 6-model MMAE filter conditional probabilities

the graph.

The filter with the largest process noise shown in Figure 7.13 was identified as the correct filter at the first epoch since the innovation is large compared to the predicted innovation covariance. After 10 seconds the MMAE filter adapts to weight the second Kalman filter with a conditional probability of 0.999 to 3 decimal places and at 26 seconds the filter adapts to weight the third Kalman filter with a probability of 0.966 to 3 decimal places. The figure shows that using the threshold probability results in the filter continually adapting the process noise. During the vessel manoeuvres, the inertial errors reduce and the innovation sequence reduces in magnitude resulting in the MMAE algorithm using the Kalman filters with the smaller process noise. The filter continually adapts during the data set which allows the filter to adapt to the correct bias estimate for the z-axis gyro bias.

After approximately 80 seconds, the MMAE algorithm predominantly

uses the filters with the scale factors of 0.1, 1 and 10, with the scale factor of 10 providing the dominant filter after 600 seconds. This is because these filters result in similar covariance matrices, and hence similar PDF estimates. As more observations become available the MMAE algorithm gradually converges to a single filter even though the minimum threshold is applied. The MMAE converges to the scale factor of 10, not 1 which was expected to give the best navigation performance as identified in §6.3. In fact, as shown in §7.4.7 later on in this chapter, the scale factor does result in a small improvement in some of the navigation states, although there is also a small deterioration in other navigation states. The navigation performance is considered further in §7.4.7.

Figure 7.14 shows the PDF plotted for a range of innovations and the innovation covariances. For the GPS/INS MMAE integration filter, the innovation sequence is the same for each model. However, each filter uses a different process noise matrix which results in a different innovation covariance. In the figure, it is shown that given a small innovation, a filter with a small covariance will most likely be the correct filter. Similarly, for larger innovations, a filter with a larger covariance is more likely to be the correct filter. In other words, MMAE chooses the Kalman filter with an innovation covariance consistent with the the new innovation. This results in an innovation sequence with similar properties to those shown for the process noise and adaptive-Q filters.

7.4.2 Covariance

Figure 7.15 shows the the covariance of the gyro bias states using the MMAE algorithm. The covariance of the multiple model filter is given

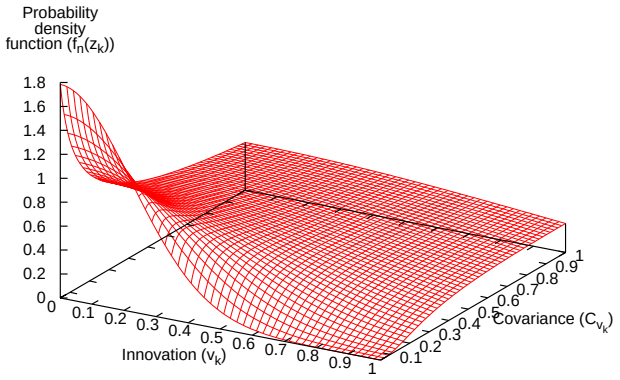


Figure 7.14: Probability density function

by Equation 4.37 and is formed using all of the multiple model estimates weighted by the conditional probabilities. The covariance of the bias states shows similar properties to the covariance obtained using the other adaptive algorithms in that the covariance is increased rapidly after the first epoch. This appears to reduce the initial bias errors that are caused at the first few epochs when a large initial covariance is used. Large initial bias errors are caused when using a large initial covariance because the some of the initial position and velocity errors are absorbed incorrectly by the bias estimates. Because the bias estimates have a relatively low observability, the errors can take a long time to reduce. After the first few epochs, the position and velocity errors in the innovation sequence are primarily caused by the INS attitude misalignment and sensor errors.

The MMAE algorithm increases the covariance to the largest value of all the adaptive filters by increasing the process noise which results in the fastest alignment performance of any of the filters considered. It is also shown that the initial covariance reduces rapidly and by 315 seconds, the covariance estimate for the z-axis gyro bias state is within 0.001°/s of the CKF covariance.

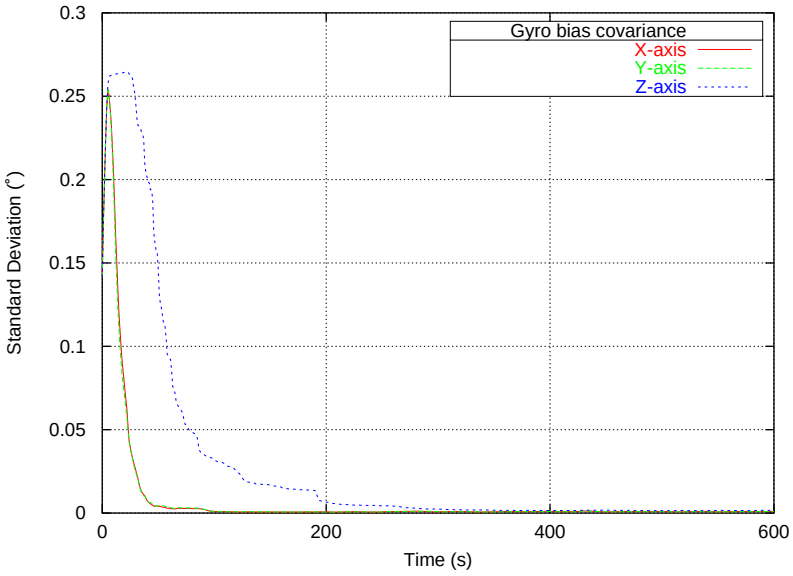


Figure 7.15: 6-model MMAE filter gyro bias covariance

7.4.3 Dependence on initial covariance

The alignment times for different initial bias state covariance values are shown in Table 7.5. The initial covariance estimates of less than or equal to 0.01 all result in the same time to alignment. The initial covariance of 0.1 resulted in the fastest time to alignment, with the yaw estimate resolved to within 0.5° after only 92 seconds. The larger initial covariance estimate of 1 increased the time to alignment by causing too much weight to be given to the bias states resulting in larger, erroneous bias estimates that take longer to converge. The table shows that the filter significantly reduces the dependence on the initial covariance which is a key characteristic that is required for an alignment filter.

	Initial covariance ($^\circ$/s, ms^{-2})					
	1.0	0.1	0.01	1×10^{-3}	1×10^{-4}	1×10^{-5}
MMAE (s)	191	92	127	127	127	127

Table 7.5: MMAE alignment times for different initial process noise

7.4.4 Processing time

The principle drawback of the multiple model filter is the significant increase in processing time required for computing the multiple model estimates. The previous trial used a decentralised bank of six, 15-state INS Kalman filters. Table 7.6 shows the average processing time required per 1 second epoch using KinPos^i on a 1.4GHz PC using the Linux operating system. The results were obtained by averaging the results from 10 processing runs. The table shows that for the decentralised filter, the time required for processing the INS filter is less than 10% of the total GPS and INS processing time per 1 second epoch. The most significant processing time is required by the mechanisation of the inertial measurements. In a real-time system, the inertial mechanisation takes place as soon as the measurement is available. Therefore, at each GPS update, the integrated navigation solution using the decentralised CKF structure requires approximately 6ms of processing time including the processing of the GPS measurements.

Table 7.6 shows that each additional Kalman filter using the MMAE algorithm requires an additional 1ms of extra processing time (the extra processing time required for computing the PDF for each filter and calculating the weighted combination of states and covariance at each epoch is considered negligible). Consequently for the six model Kalman filter

	Time (ms)
GPS	5
INS mechanisation (100Hz)	7
INS Kalman filter	1
Total	13

Table 7.6: KinPos^i CKF processing time per 1 second epoch

described, total processing time per 1 second epoch is 18ms, with the integrated solution available in a real time system with a latency of approximately 11ms. Therefore, modern processing technology allows MMAE to meet the latency requirements for a number of real time applications. It should also be noted that KinPos^i is still in its development stage. This means that there are many potential efficiency improvements that can be made to the software to improve the speed of processing.

Two further considerations are that, firstly, the MMAE algorithm uses parallel Kalman filters. This provides the potential for parallel processors to compute the solution for each filter concurrently, hence removing any extra processing time. Secondly, it is also possible when processing GPS and INS measurements for the Kalman filter to run behind the main mechanisation loop. This means that the correction from the Kalman filter can be applied when the solution becomes available, see for example in Watts and Neads (2003). This removes any latency other than that occurring from the INS mechanisation. Further modifications that can be made to the actual MMAE algorithm to reduce the processing time are considered in Section 7.4.6.

7.4.5 Dynamic MMAE

For the previous six model MMAE filter, when the minimum threshold condition is not used, the filter uses the largest filter for 40s before adapting to the second largest filter. The second largest filter is then used for the entire processing run. This is unsuitable for the GPS/INS filter, especially during alignment, since it ignores new measurements and converges to an incorrect model.

The previous six-model filter used a minimum threshold on the conditional probabilities in order to keep the filter adaptive. This is termed *dynamic* MMAE. This section examines the effect of varying the magnitude of the minimum threshold on the way the filter responds to new measurements. This section also examines using the weighted combination of states formed using only the PDF at each epoch as suggested in Welch (1999) as a method of keeping the MMAE filter adaptive.

7.4.5.1 Conditional Probability Threshold

Using a threshold value for the conditional probability threshold is one method for forcing the MMAE filter to remain adaptive. As identified in the previous analysis, the selection of the value for the threshold is important for a number of reasons. Firstly, the threshold level needs to be large enough to allow the filter to react rapidly to changes in the innovation sequence. For example, for the six-model filter, initially the filter with the largest process noise is selected by the algorithm as the best model. As a result, the conditional probabilities for the other filters tend to zero. The recursive algorithm for computing the conditional probability using the PDF means

that a filter that has a very small computed probability will take a long time to increase again if the filter becomes valid. If the conditional probability equals zero due to the finite precision of the computer word length, the filter will never be included.

It is also expected to be important that the threshold value is not too large so that filters that are not a good model of the system are not incorporated in the weighted solution. The use of the threshold value means that filters that may be poor a model of the system which should be removed from the solution will always be used to form the weighted estimate to some level. Therefore the threshold value needs to be selected to be small enough to allow some filters to be effectively removed from the solution by allowing the conditional probability to be small enough. In the implementation of the MMAE algorithm in KinPos^i, the threshold probability is applied to the estimated probabilities *after* the weighted combination of state estimates has been formed. This is necessary to avoid increasing the weight given to filters that have been identified by the algorithm as containing incorrect stochastic information.

In order to assess the impact that the selection of the threshold value has on the time to alignment and the navigation performance, the filter was run using different threshold values. The results of which are shown in Table 7.7. The table shows that the fastest alignment time was obtained using the minimum threshold of 1×10^{-2}. However, this estimate resulted in the navigation errors being increased since the threshold does not allow the models with the larger process noise estimates to be removed sufficiently from the weighted combination of states. Therefore the resultant navigation errors were noisy. Similarly, the minimum threshold of 1×10^{-1} was also

tested but this resulted in the yaw alignment not converging to within 0.5° at all because the models with large process noise estimates were still included in the estimation. In fact, the filter resulted in similar performance to the weighted PDF filter considered in the following section.

	Threshold						
	10^{-2}	10^{-3}	10^{-4}	10^{-5}	10^{-10}	10^{-20}	10^{-30}
Alignment Time (s)	107	127	284	286	286	368	502

Table 7.7: MMAE alignment times for different initial process noise

As expected, the smaller threshold values result in a longer time to alignment because the MMAE algorithm takes longer to adapt from one filter to another. For example using the minimum threshold of 1×10^{-20}, the filter using the largest scale factor is predominantly used for 25 seconds, followed by the second filter for 20 seconds and then the third filter for 300 seconds. After 600 seconds the MMAE algorithm fully converges to the filter using the scale factor of 10. This is unsuitable for the alignment algorithm as transition between filters is too slow resulting in noisy filter estimates at the beginning of the dataset and a slower alignment time. It is interesting to note, however, that the alignment time for this filter is 368 seconds which, despite the sub-optimal performance, is faster to align than the CKF.

All of the MMAE filters using a minimum threshold of 1×10^{-3} or less resulted in the algorithm converging to the filter using the process noise scale factor of 10. For all of these minimum threshold filters, after 1000 seconds, the conditional probability for the filter using the process noise

scale factor of 10 was 0.99 or larger. This is due to the point identified in §7.4.1 where the filter is not able to adapt quickly between filters that have a similar covariance. It requires many epochs in order for the conditional probabilities to change when the computed PDF is similar for each filter. As a result, the transition between using different Kalman filters is slower which may result in reduced performance if the filter does not react quickly to a change in the stochastic properties.

7.4.5.2 Weighted combination of PDF at each epoch

Another method to keep the MMAE filter adaptive suggested by Welch (1999) is to use the weighted combination of PDF at each epoch as in Equation 4.38. Figure 7.16 shows the z-axis gyro bias estimate obtained by directly using the PDF estimate to compute the filter weight factors. The figure shows that the z-axis gyro bias error is resolved slower than when using the conditional probabilities to weight the filters. Furthermore the filter does not estimate the gyro bias as accurately as the conditional probability weighted filter once the filter has converged to the correct value. In other words the filter does not appear to adapt the process noise once the bias state has been resolved.

The results from Figure 7.16 are explained by looking at the weight factors obtained using the PDF in Figure 7.17. The figure shows that for the first two epochs the Kalman filter uses the process noise scale factor of 10000 which is identified as the single filter to use. After this the Kalman filter using the scale factors of 1000 and 100 result in the largest estimated weight. However, after approximately 10 seconds, the estimated filter weights remain relatively constant with no single filter being identified

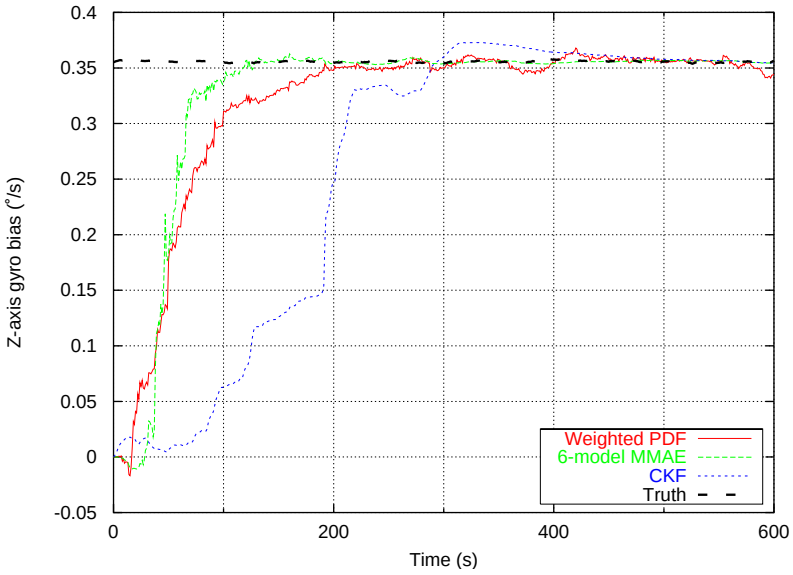


Figure 7.16: Weighted PDF MMAE filter z-axis gyro bias alignment

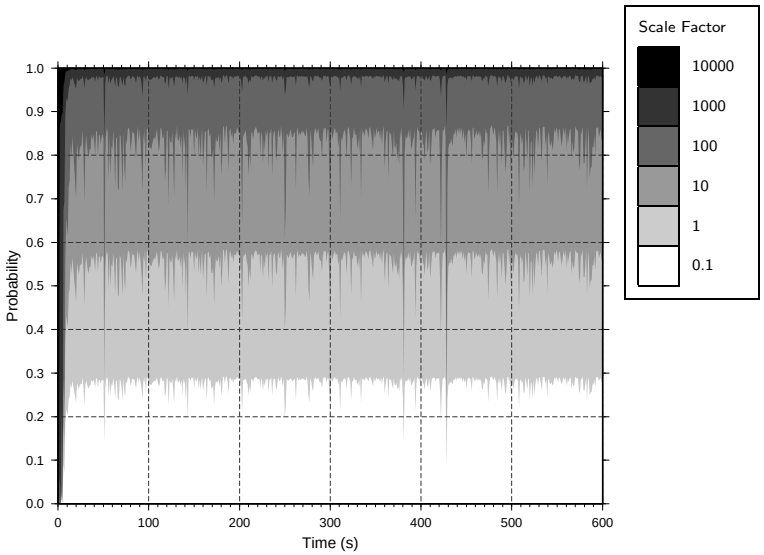


Figure 7.17: Weighted PDF MMAE filter weights

as the correct filter. This is similar to the situation that occurred in the last section where the minimum threshold was set too large. This does not provide the characteristics required for the alignment filter because the filter results in the navigation errors being too large.

The PDF is not suitable for weighting the Kalman filters at each epoch since the difference between the PDF for a given innovation is not significant enough to remove filters that do not model the inertial errors correctly.

7.4.6 Reducing computational burden

This section identifies two simple methods that can be used to help reduce the computational burden of running multiple model Kalman filters. The approaches considered are reducing the number of parallel Kalman filters used, and using the filter with the maximum conditional probability at each epoch.

7.4.6.1 Reducing the number of models

This section examines the effect of reducing the number of models used in the MMAE filter in order to reduce the computational burden of MMAE. Figure 7.18 shows the z-axis gyro bias estimate from the MMAE filter using two, three and four models. The different models were formed in the same way as the six model filter in §7.4.1 using the scale factors 10 to 10000.

Figure 7.18a shows the two model filter estimate. The filters use the scale factors of 10 and 10000. The MMAE filter therefore effectively consists of an alignment filter formed using the large scale factor, and a navigation filter formed using the smaller scale factor. The scale factor of 10 is used for the navigation filter since it is the filter that the MMAE algorithm

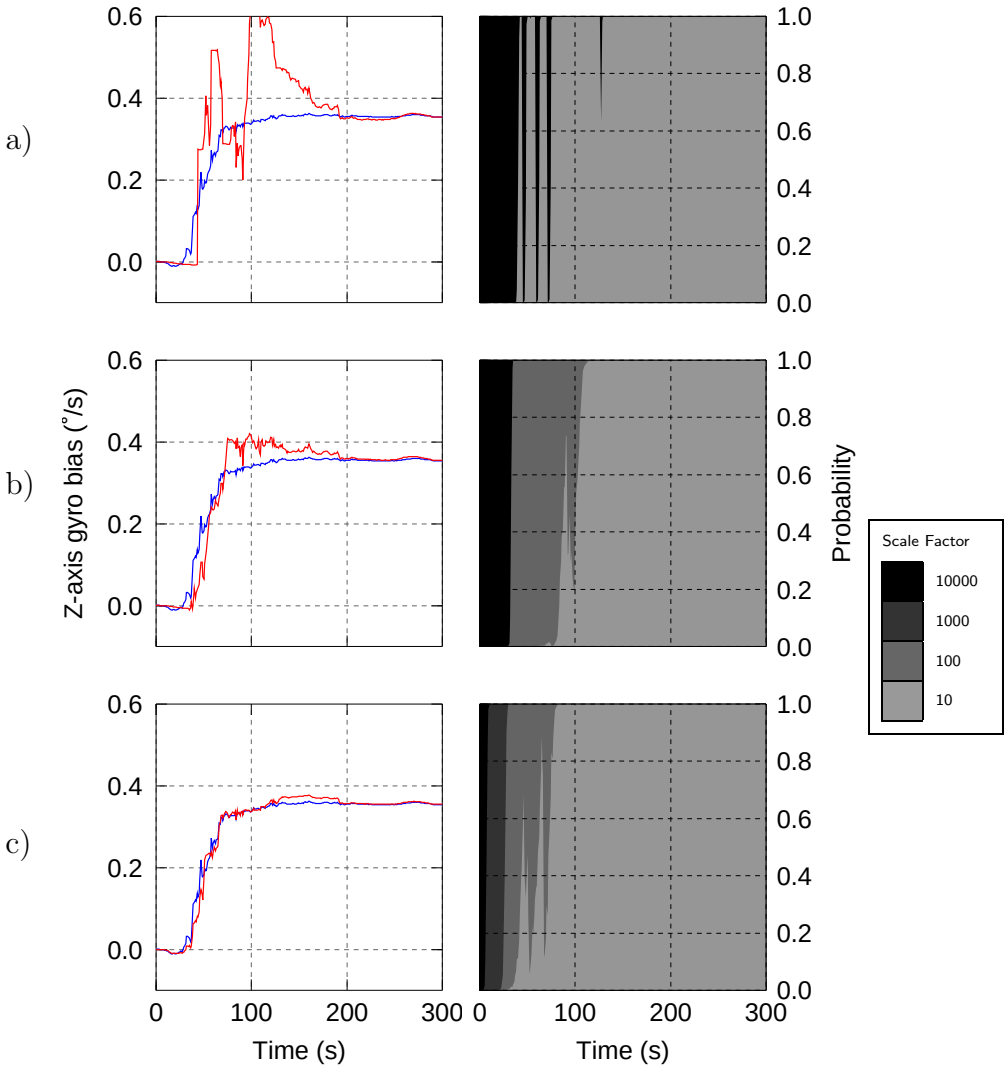


Figure 7.18: Conditional probability and z-axis gyro alignment using a) 2 models, b) 3 models, c) 4 models

converged to in §7.4.1. The MMAE algorithm is used to switch between the two filters depending on the information provided by the innovation sequence.

From Figure 7.18a it is shown that the two model estimate does not provide a smooth transition to the correct bias estimate. The MMAE algorithm identifies the alignment filter at the start of the processing run and transfers to the navigation filter after 40 seconds. The MMAE adapts between the alignment filter and the navigation filter at 45, 60 and 70 seconds. From the bias estimate it is clear that the alignment filter gives too much weight the measurements resulting in the large jumps in the bias estimation. When the algorithm converges to the navigation filter, the bias estimate from the navigation filter is poor and the bias estimate takes over 300 seconds to converge, with the attitude alignment achieved after 370 seconds.

The two model estimate is poor because the MMAE filter is unable to identify a correct model during the transition period between the alignment filter and the navigation filter. Instead, the algorithm selects the nearest model to approximate the process noise which is not a good estimate. The three model filter in Figure 7.18b provides an improved transition to the bias estimate over the two model filter. This is because an intermediate model is used using a scale factor of 100. This provides an improved estimate of the process noise during the transition period from the scale factor of 10000 to the scale factor of 10. Yaw alignment is achieved to within 0.5° for the three model filter in 285 seconds.

The four model filter provides an estimate much closer to that of the 6 model filter. This is because the MMAE algorithm has a filter that

accurately models the process noise information during each phase of the alignment as the bias estimate improves. However, the full attitude alignment is still not obtained until 285 seconds due to the z-axis bias state not being resolved to a high enough accuracy for the yaw alignment to be maintained. This indicates that in the 6 model filter, the period from approximately 100 seconds to 600 seconds is significant where the filter uses the models with the scale factors of 0.1 and 10. From the estimate for the four model filter, it appears that the alignment is maintained by effectively using smaller process noise estimates during this time.

In order to reduce the number of models that are used in the filter, it is important that there still remain enough models to accurately model the process noise during all conditions. In the case of the alignment filter, there needs to be a smooth transition from the larger process noise to the lower process noise. The number of models used in the MMAE filter can be reduced once navigation performance has been achieved so that the MMAE is just used during alignment. This would decrease the processing requirements which will reduce the latency at which the multiple model estimate would be made available. Also, in the case of the filters shown in this chapter, the MMAE algorithm has converged to a single Kalman filter estimate. This filter could be then used as a single filter which reduces to the standard CKF.

7.4.6.2 Maximum conditional probability filter

Another method that can be used to improve the efficiency of the algorithm is to use the filter with the maximum conditional probability as the single 'correct' filter at each epoch. This significantly improves the efficiency

of the algorithm since the multiple models are only required in order to calculate the innovation covariance at each epoch. Therefore, once a single filter has been identified, only one model is used for the Kalman filter update. In addition to examining using the filter with the maximum conditional probability, this section also examines using the filter with the maximum PDF at each epoch. The z-axis gyro bias estimates for the

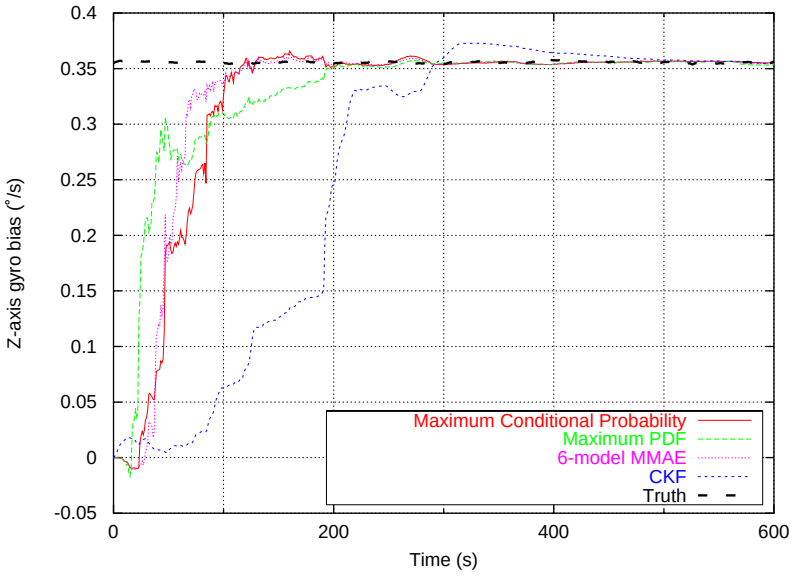


Figure 7.19: MPDF and MCP filter z-axis gyro bias alignment

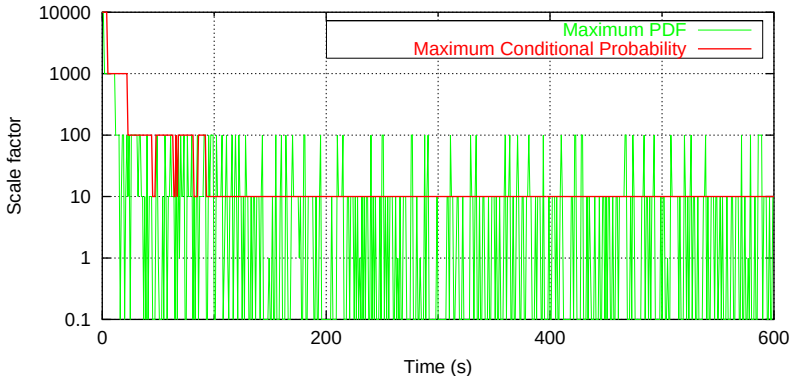


Figure 7.20: MPDF and MCP filter scale factors

maximum conditional probability (MCP) filter and the maximum PDF (MPDF) filter are compared to the conventional MMAE algorithm estimate in Figure 7.19. The filters are formed using the same six models that are used in §7.4.1. From the figure, it is clear that the MCP filter provides similar characteristics to the standard MMAE filter for the z-axis gyro bias estimate. The bias is aligned during the period of low dynamics, and the noise reduces once the correct estimate is obtained. However, the yaw alignment for the MCP filter is not obtained to within 0.5° until 285 seconds. This is because the MCP filter uses only one of the six filters instead of forming the optimal weighted combination. The yaw estimate is however obtained to within 0.8° after only 101 seconds.

Figure 7.20 shows the scale factor estimate from the maximum conditional probability filter. After 100 seconds only the filter using the scale factor of 10 is used. The figure shows that the MCP filter adapts from the largest scale factor filter to the filter using the scale factor of 10 in the first 100 seconds. Unlike the standard MMAE algorithm, the single filter is used after this point. The 6 model filter uses a combination of the filters using the scale factors 0.1, 1 and 10. This appears to be significant from the results shown for the MCP filter in that it makes the navigation estimate less noisy when the INS is aligned.

Figure 7.19 also shows that the MPDF estimate is slightly slower to converge to the correct bias with the large manoeuvre at 190 seconds required for the bias to be fully resolved. The attitude alignment is obtained to within 0.5° after 195 seconds which is faster than the MCP filter but is still slower than the conventional MMAE algorithm. From Figure 7.20 it is shown that after the initial selection of the filters with the large scale

factors at the start of the processing run, the maximum PDF fluctuates between the scale factors of 0.1 to 100 very rapidly. It is interesting to note, however, that the maximum PDF is very rarely identified as the scale factor of 1. During the first 1000 epochs of processing, for the scale factors 100 to 0.1, the number of epochs for which the filter is identified as the maximum PDF are 128, 338, 22 and 500 respectively. It is not known why the scale factor of 1 should only be used for such a comparatively small number of epochs. Because there is a much faster transition between the filters, the MPDF filter potentially provides an algorithm that can change rapidly which may be useful for during normal navigation. The navigation performance for the filters is examined in §7.4.7.

From the analysis in this section it is clear that both of the algorithms result in a faster time to alignment than the CKF. However, both the filters are slower to align than the standard MMAE algorithm.

7.4.7 Navigation performance

It has been shown in the previous sections that the MMAE filtering approach can greatly improve the time for aligning a low cost INS over the CKF. The other potential benefit of MMAE is that the multiple models can identify a 'correct' filter from a bank of Kalman filters and can also adapt the stochastic properties during navigation. The identification of a single correct filter can be useful in the design stage of a Kalman filter in that the algorithm essentially provides a method for self-tuning. Also, with the identification of the dynamic MMAE algorithms, it is also anticipated that the algorithms will be able to provide improved navigation performance over the CKF. Figure 7.21 shows the navigation performance

of the standard MMAE filter and also the MPDF and MCP filters. The reduced model filters from §7.4.6.1 are not included as they all converged to a single filter which results in the same performance as the MCP filter.

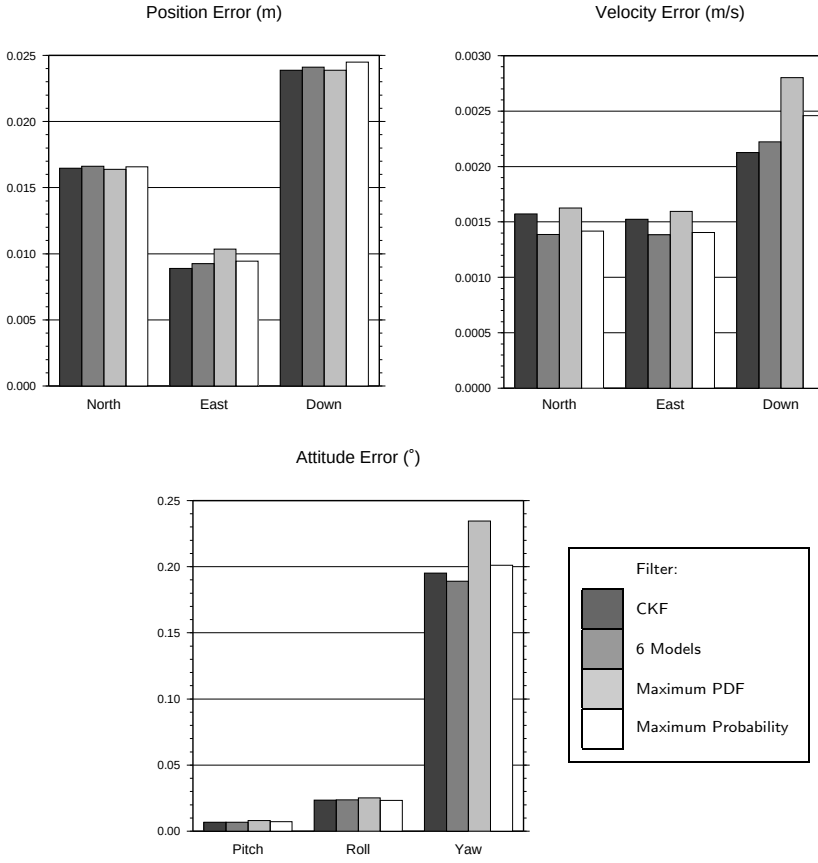


Figure 7.21: MMAE filter navigation errors

Firstly, the standard 6 model estimate gives a small increase in the position and vertical velocity errors, and a small decrease in the horizontal velocity errors. The differences are at the sub centimetre level for position and sub millimetre per second level for velocity and can therefore be considered negligible. There is also a small improvement in the yaw attitude error of 0.006° (23 arcseconds). It was mentioned in §6.3 that the base process noise filter used to form the MMAE process noise matrices resulted in the best navigation performance for the CKF, so it was expected that the

MMAE algorithm would converge to the filter using a scale factor of 1. In fact, it is shown in Figure 7.21 that the filter using a scale factor of 10 results in very similar navigation performance which gives a small improvement in some states and a small deterioration in others. The analysis in §6.3 considered only changing the process noise for the bias states and not the process noise for the navigation states.

The navigation errors for the MPDF filter are the same navigation errors that result from using the reduced order filters as during the period of data used for analysing the navigation performance, only the process noise using the scale factor of 10 is used. The figure shows that there is a small increase in all of the navigation errors, although the increase can be considered negligible.

Also shown in Figure 7.21 is the navigation performance of the MCP filter. From the figure it is clear that there is a deterioration in the navigation errors for all of the states. The most significant of which is the yaw error which is increased by 0.039° (2.5 arc min). In §7.4.6.2 it was mentioned that the rapid change in the estimation of the process noise matrix that resulted when using the MCP filter may potentially provide an MMAE filter that is able to adapt quickly to new measurements. In fact, the filter results in a more noisy estimate since the filter with a scale factor of 100 is used relatively often (in the initial analysis it was used for 128 epochs in the first 1000 epochs). Unfortunately, none of the variations of the MMAE filters appear to lend themselves to adapting the process noise during normal navigation. However, the MMAE algorithm does operate well when the stochastic properties of the filter change significantly over time.

7.5 Summary

This chapter has investigated the use of three different adaptive Kalman filtering algorithms for the initialisation and alignment of a low cost INS using carrier phase GPS updates. Each of the filters identified have different characteristics that provide many different advantages and disadvantages for the alignment problem. These are summarised in Table 7.8. Figure 7.22 also shows the yaw alignment obtained using the adaptive algorithms compared to the CKF estimate. It should also be noted that the CKF estimate is the estimate obtained from §6.3 where the filter is initialised using an empirically derived initial covariance matrix that provided a fast alignment.

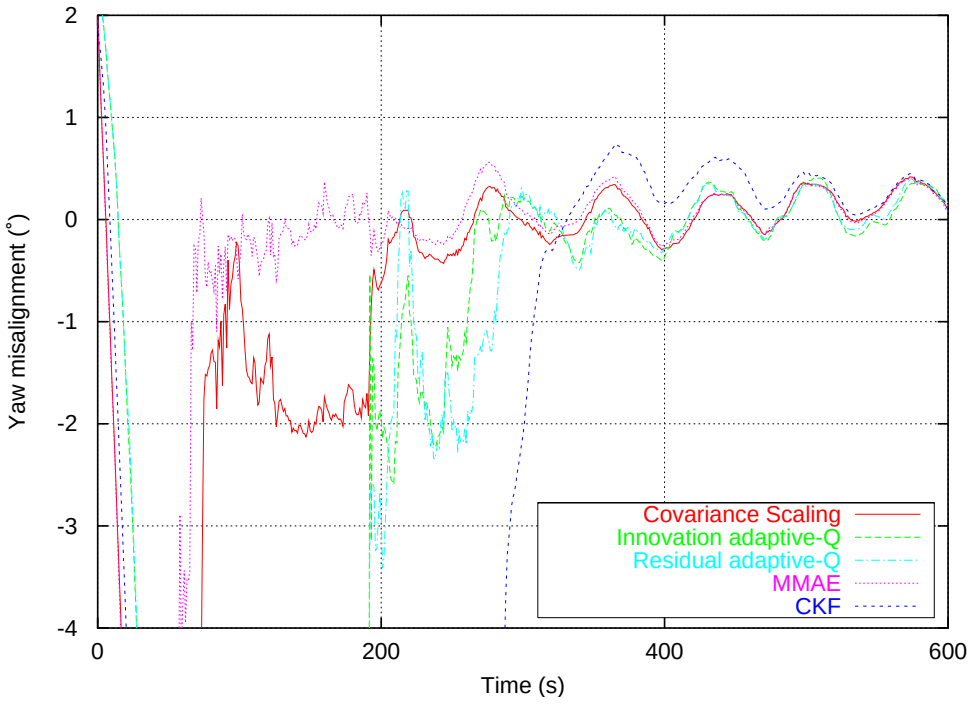


Figure 7.22: Attitude alignment

From Figure 7.22 and Table 7.8, it is clear that all of the adaptive

algorithms investigated result in an improvement in the time to alignment over the CKF. For the process noise scaling filter and adaptive-Q filter, the alignment is achieved in just over half the time required for the CKF. The actual improvement is slightly better than this because the adaptive algorithms only require the vessel to undergo a short period of high horizontal acceleration alignment manoeuvres. The alignment manoeuvre begins at approximately 190 seconds so that the adaptive algorithms require only 75 seconds of the alignment manoeuvre compared to the CKF that requires 260 seconds. Of course the MMAE estimate is resolved in 127 seconds which completely removes the requirement for the dynamic manoeuvre to take place and just uses the small horizontal accelerations that occur in the normal trajectory to align the filter. The MMAE estimate requires less than one third of the time that the CKF requires to obtain the yaw alignment to within 0.5°. The improvement in the time to alignment for other datasets will depend on factors such as the initial attitude misalignment, the initial sensor errors and the vessel or vehicle trajectory.

Of all the adaptive algorithms, the adaptive-Q filters are the least suited to the alignment problem. The high dependence on the initial covariance and process noise matrix means that the adaptive-Q filters do not always result in an improvement over the CKF. In fact, the filter may not align the INS at all if the initial stochastic information is not correct. However, the adaptive-Q algorithm has shown some promise, however, for the integration of low cost INS and GPS during normal navigation due to its ability to adapt the individual states separately and potentially model any correlations that may exist. The adaptive-Q algorithm has also been used successfully with a tactical grade IMU in Mohamed (1999).

The process noise scaling algorithm is the simplest to implement of all the filters and it provided a significant improvement in the time to alignment. The MMAE filter provided the best alignment performance but the improvement is at the cost of a significant increase in the amount of computer processing required. None of the simple efficiency improvements were able to match the performance of the 6-model filter.

The most important consideration that applies to all of these filters is that they all require the definition of the initial process noise matrices (or matrices for the MMAE algorithm). None of the filters are able to remove the requirement for *a priori* stochastic information although the process noise scaling filter and the MMAE filter both reduce the requirement for the definition of the correct initial covariance. The following chapter investigates the use of the adaptive algorithms with real data.

Method	Alignment time	Advantages	Limitations
Conventional	453s	+Stability	-Slow convergence given poor *a priori* information -Inability to adapt stochastic information
Process Noise Scaling	265s	+Simple to implement +Negligible computational increase +Good performance +Reduces dependence on initial covariance +No initialisation lag	-Still relies on definition of *a priori* process noise -Cannot individually estimate process noise for each state -Cannot estimate correlations -Cannot reduce process noise
Adaptive-Q (Innovation based) (Residual based)	264s 283s	+Minimal computational increase +Process noise for each state is individually estimated +Good performance if correct initial conditions identified +Self tuning for navigation	-High dependence on initial *a priori* process noise matrix -High dependence on initial *a priori* covariance matrix -Convergence not always guaranteed -Poor performance when estimating correlations -Requires correct definition of measurement noise -Slow initialisation
Multiple Models	127s	+Robust +No initialisation lag +Best performance +Alignment obtained during period of low dynamics +Reduces dependence on initial covariance +Self tuning for navigation	-Computationally intensive -Still requires definition of *a priori* process noise matrix -Cannot individually estimate each state -Cannot estimate correlations -Slow to adapt between filters

Table 7.8: Summary of adaptive Kalman filters for INS alignment

Chapter 8

Test results

8.1 Introduction

This chapter describes the results of two kinematic field trials using a low cost INS with carrier phase GPS. The field trials are used to assess the performance of a low cost MEMS INS integrated with GPS. The trials are also used to investigate the adaptive Kalman filtering algorithms examined in Chapter 7.

This chapter first describes the results obtained from a marine trial conducted in February 2002 in Plymouth, UK, from which the trajectory in §6.2 was derived. The integration of GPS with low cost inertial sensors is an innovative area of research as current GPS/INS hydrographic surveying systems mostly use medium cost tactical grade INS.

The second part of this chapter describes the results obtained from a vehicle trial conducted in July 2003 in Nottingham, UK, using the same low cost IMU as used in the marine trial.

8.2 Marine Trial

This section describes the kinematic GPS and low cost INS trial conducted at the Royal Naval Hydrographic School in Plymouth. Some of the issues related to collecting GPS and INS data, including sensor synchronisation and calculation of the lever-arm separation of the IMU and the GPS antenna, are also presented.

The marine trial is investigated as it is a potential application area in which low cost INS based systems can be used to replace higher cost systems. The attitude accuracy requirement for a modern multibeam sonar with a swath width of 150° is 0.05° in the roll axis, with a similar requirement for pitch and yaw Trethewey et al. (1999). This results in a depth error of 0.5% at the largest scan angle (ibid.). This research investigates whether this accuracy is attainable from a GPS and low cost INS system.

8.2.1 Test description

The trial was conducted in February 2002 on the River Tamar at Plymouth, UK on a small hydrographic survey vessel owned by the Royal Naval Hydrographic School. The survey vessel was used since it has a TSS Position and Orientation System for Marine Vessels (POS/MV 320) installed which is used for geo-referencing a multibeam sonar mounted on the underside of the vessel.

8.2.1.1 POS/MV

The POS/MV system was used to provide an attitude reference for the trial. The POS/MV utilises two dual frequency Novatel GPS receivers with a tactical grade Litton LN200 inertial navigation unit (see §2.2.2 for the IMU sensor specification). The specification of the integrated POS/MV solution is given in Table 8.1.

	C/A	DGPS	RTK
Horizontal Position (m) CEP	15-40	0.5-4	0.04-0.1
Velocity (ms^{-1})	0.5	0.03	<0.03
Roll and pitch (°) RMS	0.02	0.02	0.02
Yaw with GAMS (°) RMS	0.02-0.1	0.02-0.1	0.02-0.1
Yaw without GAMS (°) RMS	0.02-2	0.02-2	0.02-2

Table 8.1: POS/MV 320 specification (TSS (UK) Ltd, 1999)

The POS/MV system uses a GPS Azimuth Measurement Subsystem (GAMS) in order to maintain the yaw accuracy during periods of low horizontal acceleration. The GAMS system consists of two choke ring antennas mounted on the roof of the survey vessel over a baseline of approximately 1.7m. The system utilises GPS carrier phase measurements in order to determine a GPS-only yaw estimate to approximately 0.1-0.5° depending on baseline separation and the level of multipath (TSS (UK) Ltd, 1999). The yaw estimate from the GPS phase measurements is noisy, yet drift free which makes the measurement ideal for use in the POS/MV Kalman filter.

The POS/MV system was used in real time to obtain position, velocity and attitude. The system used DGPS corrections from local beacons and

the carrier phase measurements were only used for the GAMS solution.

8.2.1.2 Crossbow AHRS-DMU-HDX

The Crossbow AHRS-DMU-HDX is an Attitude and Heading Reference System (AHRS) intended for applications such as UAVs, avionics and platform stabilisation (Crossbow Technology Inc., 2000). The Crossbow AHRS-DMU-HDX is herein referred to as the AHRS sensor. The IESSG's AHRS sensor was used in the marine trial as a low-cost MEMS IMU.

The sensor contains three orthogonally mounted gyros, accelerometers and magnetometers to provide a three dimensional standalone attitude solution. The AHRS sensor uses the magnetometers to obtain magnetic heading and the accelerometers to provide roll and pitch. A Kalman filter is used with the gyros to stabilise the attitude solutions from the magnetometers and accelerometers in the presence of magnetic disturbances and vehicle acceleration. The Kalman filter is also used to estimate the sensor errors.

The AHRS therefore contains orthogonally mounted gyro and accelerometer triads which allows the AHRS to be used as a conventional IMU. The specification of the inertial sensors is the same as that used for simulation, the values for which are given in Table 6.1. The sensor specification is at the very low end of sensors used as an INS, and the INS alone cannot be used for navigation without GPS updates due to the large sensor drifts. The AHRS has been modified through several revisions since the IESSG obtained the sensor, and it can now been considered as providing similar sensor specification to current sensors that are less than £2,000.

The AHRS is capable of operating in three different modes termed

voltage, scaled and *angle* and provides both serial and analogue outputs. In voltage mode, the AHRS sensor uses the internal A-D converter to sample the accelerometer and gyro sensors, and outputs the raw voltages over the serial interface at approximately 163Hz. In scaled sensor mode the AHRS attempts to compensate the raw inertial measurements for bias and scale factor errors using the internal Kalman filter and original calibration information. The scaled measurements are available over the serial interface at approximately 100Hz. Angle mode also compensates and scales the raw inertial measurements, and in addition provides the roll, pitch and yaw estimates over the serial interface at approximately 70Hz.

The raw analogue output of the accelerometers can be sampled by an external A-D converter using any of the operational modes of the AHRS. However, the raw analogue output from the gyros is not available. The gyro output is only available via a D-A converter which converts the compensated gyro output in scaled or angle mode to an analogue signal. When the sensor is operated in voltage mode, the gyro measurements are not available using the analogue output. The raw analogue output of the gyros is, however, available using the serial output. As a result, the AHRS sensor was operated in voltage mode and the serial data was used for data processing.

8.2.1.3 Sensor configuration

The sensor configuration for the marine trial is depicted in Figure 8.1. The POS/MV system is permanently installed in the survey vessel with the two GPS antennas for the POS/MV rigidly mounted to the roof of the vessel. The POS/MV processor is rack mounted in the main cabin with a

notebook computer used to control and log the data from the POS/MV. The IMU for the POS/MV is located on the floor of the front cabin. The AHRS sensor was attached directly to the housing of the POS/MV IMU as shown in the figure.

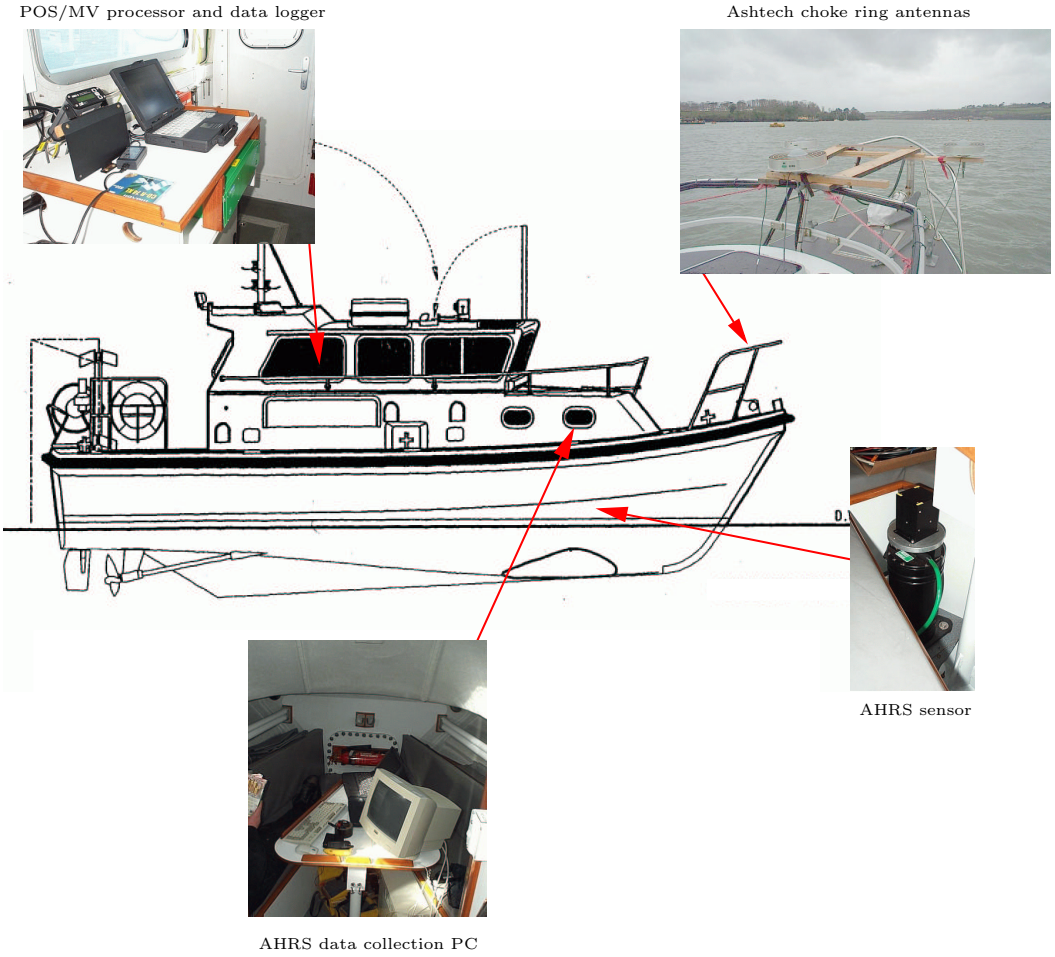


Figure 8.1: Marine trial sensor configuration

Three Ashtech receivers were connected to three Ashtech choke ring antennas which were rigidly attached to a wooden frame at the front of the vessel. The antennas were located at the front of the vessel in order to reduce the obstruction caused by the masts on top of the vessel. Three GPS receivers were used in order to provide the potential for the derivation of GPS attitude measurements.

Two computers were used to log the data from the AHRS sensor located in the front cabin of the vessel. A notebook PC with an ADAC PCM-5516-16 A-D converter card was used to log and time-tag the analogue data using the time tagging method described in §5.4.1. Unfortunately, due to a firmware upgrade in the Ashtech Z-XII receiver, the time tagging method for the analogue data collection failed. A modified method for data collection was therefore used which is described in Section 8.2.2. An additional desktop PC was used in order to record the serial data output from the AHRS sensor using the Crossbow GyroView software.

8.2.1.4 Trajectory description

The vessel trajectory for the marine trial is shown in Figure 8.2. The initial alignment part of the trajectory is the same as is used for the simulation analysis in Chapter 7. After this initial alignment section of the trajectory, the vessel then performs a series of survey lines up and down the course of the River Tamar with the survey lines increasing in distance. The survey lines last from 2 minutes to 8 minutes. This is used to assess the performance of the sensor during periods of low horizontal acceleration.

8.2.1.5 Lever-arm separation

The lever-arm separation between the AHRS, GPS antennas and POS/MV systems was measured using a total station. It was not possible to measure the lever-arm offsets in a dry dock environment, therefore the measurements were taken with the vessel resting on the water where the boat was subject to some movement. Two sets of measurements to the AHRS, the

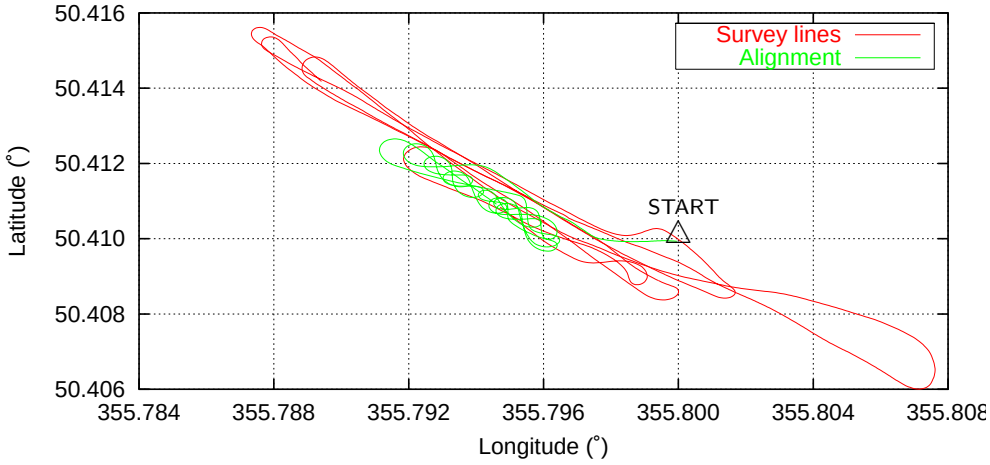


Figure 8.2: Marine trial trajectory

POS/MV IMU, the three Ashtech antennas and the two GAMS antennas were recorded. The total station was mounted on a tripod on a fixed point on the boat. The measurements could only be taken using the total station when the boat was approximately level due to the total station detecting that it was not level.

Since the points that the total station was measuring to were also fixed to the boat, it was thought that the movement of the boat on the water would not greatly affect the ability to measure the lever-arm offsets. The prism for the total station was located above each of the GPS antennas and on two points on the centre line of the boat. The prism was attached to a survey pole to measure the reference point for the IMU which was mounted beneath the hatch over the front cabin.

The points along the centre line of the survey vessel were used to rotate the measurements from the total station to the vehicle/ vessel frame. The reference point of the AHRS sensor was then used as the origin of the coordinate system to form the IMU frame using the assumption that the

IMU was aligned to the vessel frame. Table 8.2 shows the differences between the Ashtech GPS antenna baselines measured using the total station and a steel tape measure. The table shows that the measurements compare to within 3mm for all of the baselines.

Antenna	Total Station	Tape measure	Difference
Left to right	1.718	1.715	0.003
Centre to right	1.511	1.508	0.003
Centre to left	1.495	1.497	0.002

Table 8.2: Comparison of total station and tape-measured antenna separation

8.2.1.6 Doppler generation

Due to the firmware update in the Ashtech receivers, raw Doppler observations were not recorded by the GPS receiver. Consequently, Doppler data was derived from the raw carrier observations using a first order Taylor approximation. The Doppler data was derived using the equation,

$$\dot{\Phi}(k) = \frac{\Phi(k+dt) - \Phi(k-dt)}{2dt} \tag{8.1}$$

where k is the current epoch and dt is the epoch separation. A separate pre-processing program was developed to derive the Doppler observations before processing in KinPos^i.

As a consequence of using this formula to derive the Doppler observations, the first epoch of Doppler data is not available at satellite acquisition. This loss of data is not considered a problem since there was good satellite

coverage throughout the dataset. The minimum number of GPS observations available for any epoch was 6, with 10 satellites available for much of the dataset.

The first order Taylor approximation was used due to results presented in Bruton (2000). Bruton assessed the performance of higher order Taylor approximations and other differentiating filters for deriving high accuracy Doppler data in static and kinematic environments. The first order Taylor approximation produced the results that most closely matched the raw Doppler observations using Ashtech Z-XII receivers in a dynamic environment. This approximation was selected since the algorithms examined are designed for use in real time systems.

8.2.2 Sensor synchronisation

The time tagging of the Crossbow IMU measurements was planned to be achieved using the method described in §5.4.1. However, as the Ashtech Z-XII event marker input failed to work, a cross-correlation technique was implemented to match the output of the Crossbow IMU measurements with the GPS time-stamped IMU measurements from the POS/MV system.

The cross correlation equation used to time-tag the AHRS data is given as,

$$r(d) = \frac{\sum \left[(x(i) - \bar{x})(y(i-d) - \bar{y})\right]}{\sqrt{\sum (x(i) - \bar{x})^2}\sqrt{\sum (y(i-d) - \bar{y})^2}} \tag{8.2}$$

where $r(d)$ is the cross correlation function between the data set x and y for the time offset d. The cross correlation is calculated for different time offsets and the maximum value of the cross correlation corresponds to the closest match between the measurements. From analysis of Equation 8.2

it can be demonstrated that the cross correlation coefficient is insensitive to constant sensor bias and scale factor errors.

The timing for the serial data from the AHRS was obtained from the unit's internal free running counter which is taken at the time the A/D channels are sampled (see Crossbow Technology Inc. (1999)). The counter value is contained within the serial data collected from the Crossbow IMU and a single tick of the counter corresponds to approximately 0.79μs (ibid.). This scale factor was used with an approximate start time to approximately time-tag the serial data.

A C program was developed to compute the cross correlation between the POS/MV IMU and Crossbow measurements at a number of points throughout the dataset for different time offsets. The results are shown in Figure 8.3 where 200 points were tested with a time offset resolution of 1 millisecond. Each test sample contained 40,000 points which corresponds to 200 seconds of POS/MV IMU data. This sample length was chosen since it provides a high number of samples over which the sensor errors can be considered constant. Since the data from the two IMU's was not recorded at exactly the same epoch, the POS/MV IMU data was linearly interpolated to the AHRS time. Furthermore, several gaps occurred in the collection of the POS/MV data recorded using the Ethernet link. The interpolated POS/MV IMU data was only used if the difference between the POS/MV IMU record and the Crossbow record was less than half the POS/MV IMU epoch separation.

Figure 8.3 shows that there is a drift of the Crossbow time in relation to GPS time. Linear regression was used on the gyro records to remove this drift and also the constant time offset. The cross-correlation was

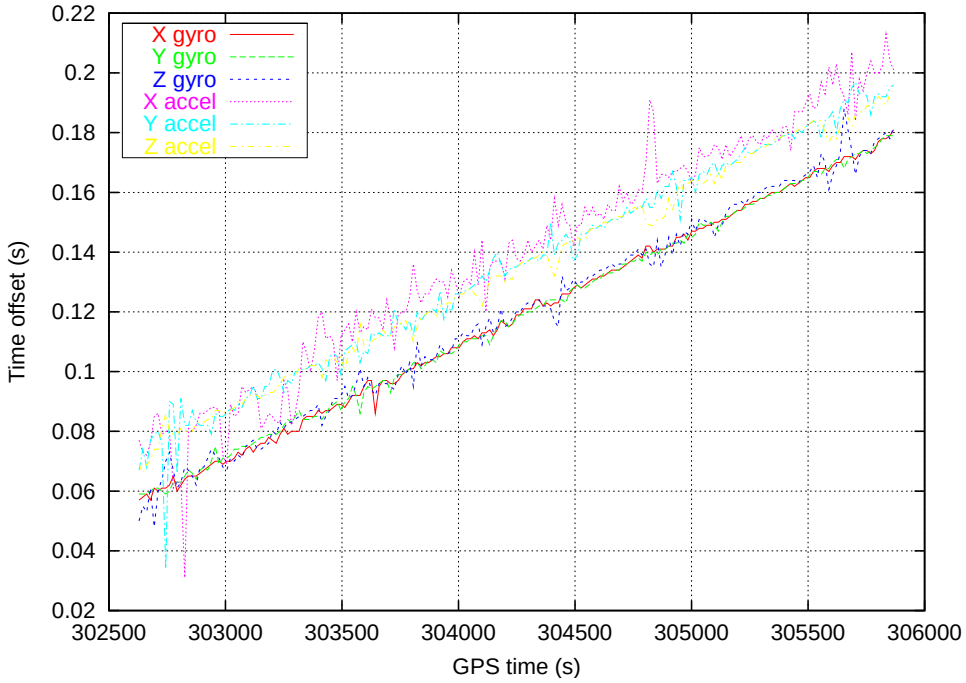


Figure 8.3: Coarse cross correlation

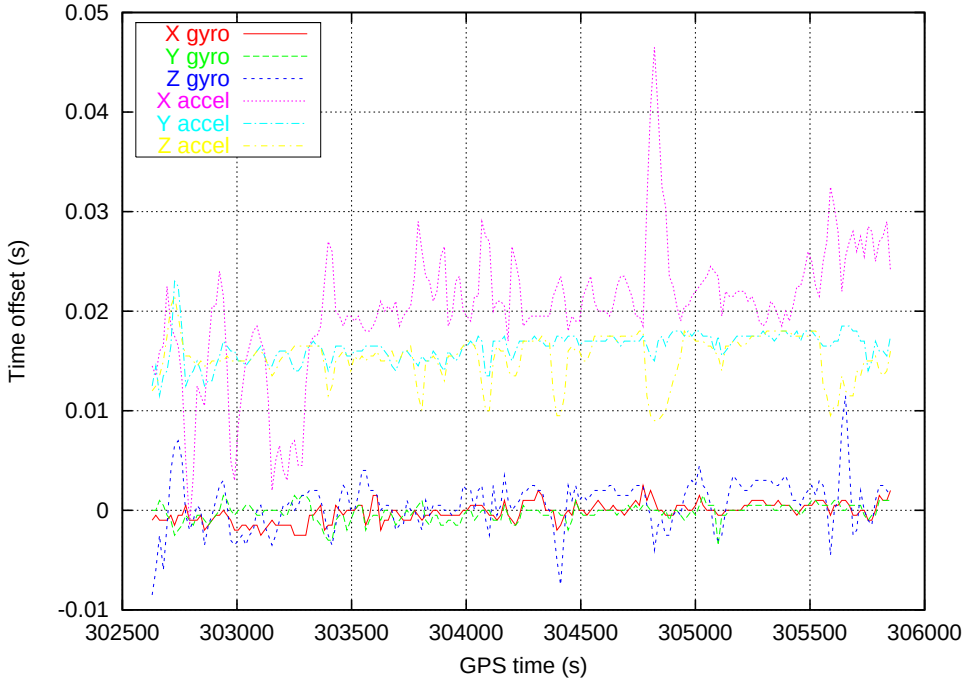


Figure 8.4: Adjusted cross correlation

recomputed and is shown in Figure 8.4. From this figure it is clear that there is a constant offset of approximately 17ms in the time at which the gyro measurements are sampled and the time at which the accelerometer measurements are sampled. This is thought to be due to the sampling electronics in the AHRS sensor. The accelerometer measurements were therefore shifted in the AHRS IMU file by three IMU epochs which corresponds to an offset of 18ms. This is not shown in the figure to preserve clarity.

The cross correlation method is suitable to use with the marine data set because of the high frequency movement of the vessel on the water. The movement of the vessel on the surface of the water ensures that there is always some low frequency motion in the output of the inertial sensors. This technique is likely to be less effective for trajectories with less movement such as those obtained from land vehicles. Table 8.3 shows the standard deviation of the final cross correlation of the IMU records. The largest standard deviation of 6.49ms for the x-axis accelerometer record can be attributed to the dynamics of the vessel. Given that the axes of the IMU were approximately aligned with the vehicle frame of the survey vessel, there is less variation in acceleration in the x-axis accelerometer than in the y and z axes. Similarly, for the z-axis gyro, there is less variation in turn rate in this axis due to the trajectory.

8.2.3 Conventional Kalman filter performance

Figure 8.5 shows the roll, pitch, yaw and vessel speed for the marine trial. The attitude of the vessel shown in the figure is the attitude obtained using the POS/MV, and the speed shown is obtained from the KinPos GPS

	Gyro			Accelerometer		
	x	y	z	x	y	z
Standard deviation (ms)	0.96	0.85	2.50	6.49	1.49	2.26

Table 8.3: Standard deviation of final cross correlation

processing using the derived Doppler observations. The change in pitch at approximately 303600 seconds is caused by the increase in the speed of the vessel which causes the front of the vessel to pitch up. The speed of the vessel mainly varies between 4 and 6m/s (6m/s is approximately 13 miles per hour). The speed of the vessel depends on the direction of travel because the vessel is either heading up or down stream, with the upstream direction being slower because of the current of the river.

The attitude errors shown in the figure are obtained by processing the AHRS IMU and carrier phase GPS measurements using KinPos^i with the decentralised Kalman filter. The GPS ambiguities are resolved in 5 epochs using the GPS only filter and the GPS ambiguities remain fixed for all the satellites throughout the dataset. The inertial measurements are integrated with the GPS measurements only when the GPS carrier phase ambiguities have been resolved. The number of satellites in view is 10 at the beginning of the dataset which reduces to 7 by the end of the dataset. The GPS antenna used for updating the INS is the antenna mounted on the port side of the vessel. This antenna was used because the masts on the vessel obstructed the satellite view less than the antenna that was mounted on the centre of the vessel. The AHRS sensor was initialised at the start of the dataset using bias estimates that were obtained from several post

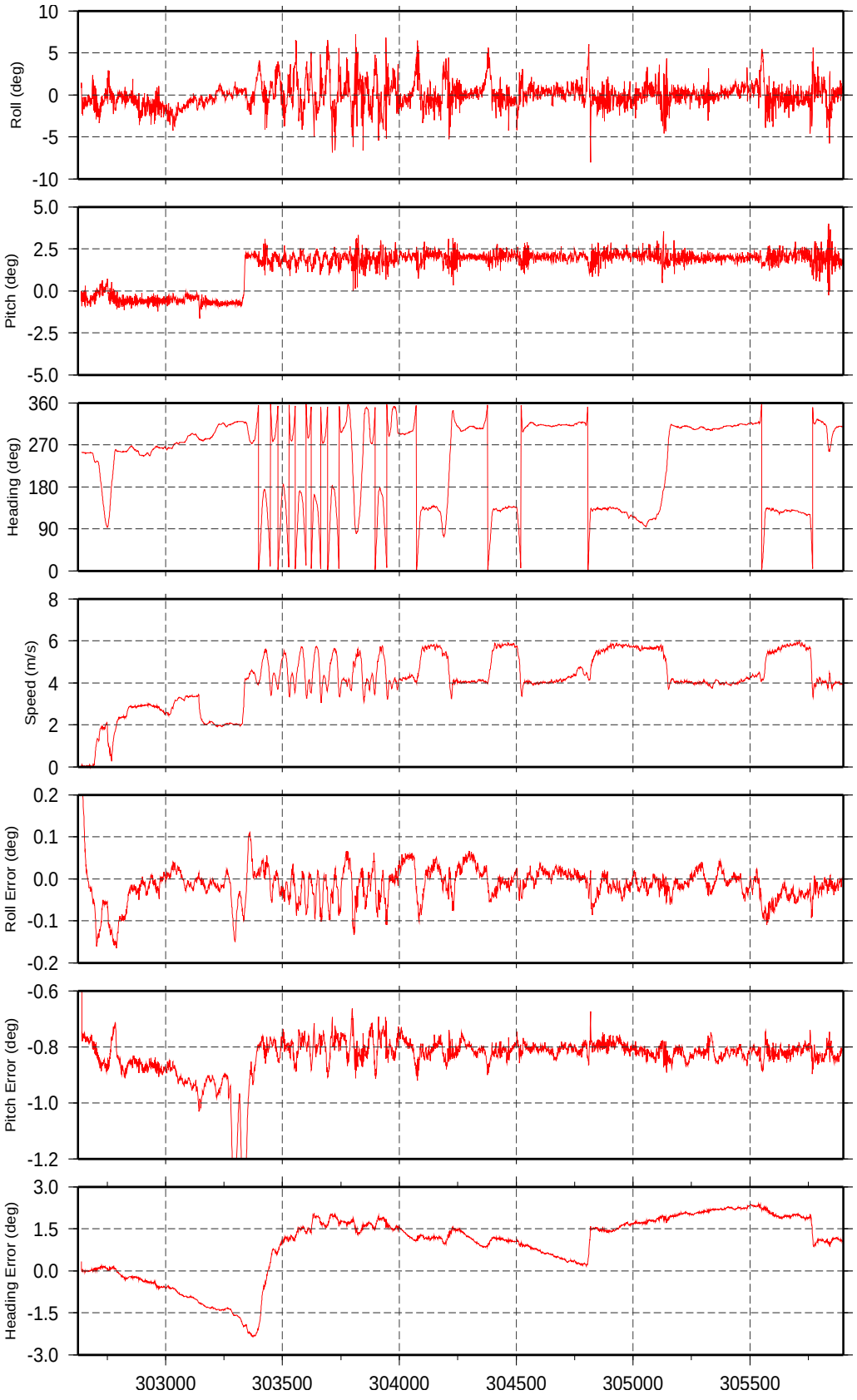


Figure 8.5: CKF attitude error

processing runs. The attitude was initialised using the estimated attitude from the POS/MV system. It can be seen at the beginning of the dataset before the alignment manoeuvre that the attitude drifts which indicates that the INS has still not been fully initialised with the correct estimates.

From the roll and pitch errors in the figure, it is shown that there is a constant attitude bias throughout the dataset. This is assumed to be due to the misalignment of the AHRS sensor with the POS/MV IMU. The AHRS sensor was mounted directly on top of the POS/MV IMU which is not a completely level surface. Therefore, it was expected that there would be a small misalignment between the x and y-axes of the two sensors. The z-axis misalignment was anticipated to be potentially a larger error because the sensor could only be aligned 'by eye'. The POS/MV IMU is a cylindrical shape (as shown in Figure 8.1), therefore there were no parallel edges that could be physically aligned. As a result of the unknown misalignment between the two sensors, it is assumed for the trial that the mean attitude error is caused only by the misalignment between the two sensors. In fact, error sources such as unresolved scale factor and sensor axis misalignments can potentially result in a constant attitude error. Therefore, this assumption should be considered for all the results obtained in this section.

The attitude misalignments for the AHRS sensor are summarised in Table 8.4. The statistics are obtained for two sections of the vessel trajectory. The first statistics are taken during the alignment manoeuvre. This is for two reasons. Firstly, all the states in the Kalman filter are observable during this period, therefore, there should not be any drift in the data. The second reason is that other error sources such as the scale factor errors

should result in a random error due to the motion of the boat. Therefore, the mean errors during this section are less likely to be contaminated by the unresolved inertial sensor errors. The table also shows the errors for the entire dataset starting at the beginning of the alignment manoeuvre. These errors reflect the navigation errors that can be expected using the low cost Crossbow INS.

		Roll	Pitch	Yaw
During alignment	Standard deviation (°)	0.043	0.051	0.161
	Mean (°)	-0.033	-0.785	1.719
Whole dataset	Standard deviation (°)	0.034	0.041	0.522
	Mean (°)	-0.034	-0.774	1.667

Table 8.4: Marine trial attitude errors using CKF

Table 8.4 shows that the roll and pitch errors for the whole processing run are 0.034° and 0.041° respectively. The RMS error specification for the POS/MV system is 0.02° which means that the roll and pitch errors for the low cost IMU are approaching the same accuracy as the higher specification system. Without using a higher grade system still, it is difficult to assess the roll and pitch errors to a more precise level because of the similarity in the accuracies of the two systems. It should be noted that the above AHRS sensor performance is the standard deviation whereas the POS/MV error is an RMS. The table shows that the mean roll and pitch errors for the entire dataset are different to the mean offset during alignment. For the roll axis there is only a 0.001° difference, but for the pitch axis the difference is 0.011° which indicates that the RMS error for the AHRS will

be larger.

The most significant error identified in Figure 8.5 and Table 8.4 is the yaw error. The yaw error is significantly larger than the roll and pitch errors because of the lower observability of the error as experienced using the simulated data. Figure 8.5 clearly identifies that the yaw error drifts between the vessel turns. Therefore the error increases as the time between the turning manoeuvres increases. For example, the most significant drift in the yaw error occurs between 304800 and 305550 seconds. There is a slower turn during this period but that doesn't result in a significant improvement in the yaw error estimation. By the end of the straight trajectory at 305550 seconds, the yaw error has increased to maximum of 1.13° from the mean. At 305550 seconds, the yaw error is improved once again. Section 8.2.6 examines using GPS attitude observations from a dual antenna GPS system in order to reduce the yaw drift.

8.2.4 CKF innovation sequence and GPS gap bridging

As identified in Chapter 6, the residual and innovation sequences contain significant information about the performance of the sensors in the integrated system. Figure 8.6 shows the innovation sequence for the marine trial for the horizontal position and velocity. The innovation sequence is important as it shows the prediction accuracy of the AHRS sensor. The GPS data for the marine trial was collected at 1Hz. For position obtained using carrier phase GPS measurements with the integer ambiguities fixed, the GPS positioning accuracy should be at the centimetre level. Figure 8.6

shows the position innovation sequence to be at the sub-decimetre level, with standard deviations for the north and east axes of 0.031m and 0.032m respectively. The mean errors are -0.006 for both sets of data, and the initial 300 seconds were not included to obtain these statistics since the filter has not fully resolved the attitude and sensor errors. The innovation sequence for the position in the down axis is slightly larger at 0.048m.

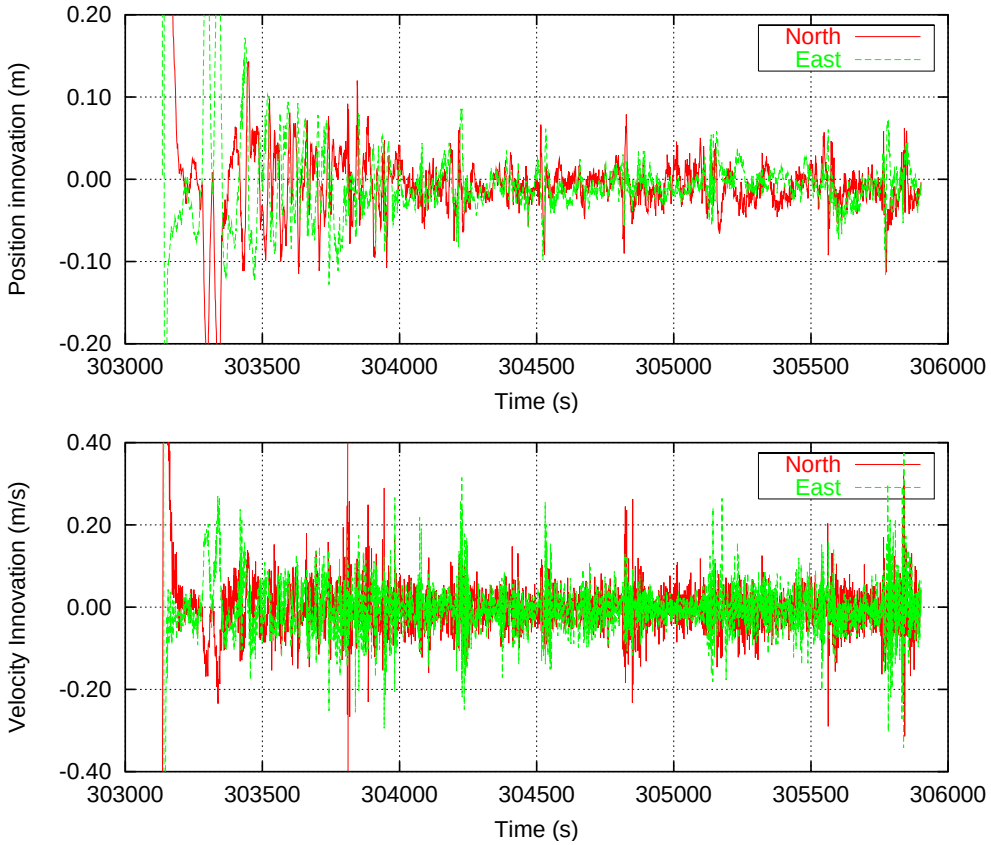


Figure 8.6: Position and velocity innovation sequences

The innovation sequence is of larger magnitude than expected. The innovation sequence represents the combined error of the INS and GPS systems. It is assumed that the positioning accuracy using the GPS measurements is relatively consistent due to the good satellite observability throughout the dataset. Choke ring antennas were used to shield the antenna from

signals arriving beneath the antenna, i.e. to reduce the multipath caused by the GPS signals reflecting from the surface of the water. Therefore the larger magnitude of the innovation sequence is thought to be caused by the AHRS sensor errors. The innovation sequence shows spikes at, for instance, 304220 and 304520 seconds which coincide with the vessel turning. This indicates that INS errors such as the yaw error have become observable through the turn, resulting in a position error.

The velocity innovations are larger than for the position estimation with standard deviation in the north and east axes of 0.055m/s and 0.057m/s and mean of -0.004m/s for both axes. This is thought to be due to using the first order Taylor approximation to derive the Doppler data. In the results obtained for the derived data in Bruton (2000), the velocity errors in a dynamic environment were estimated to be 0.24m/s, and during static conditions the velocity errors were 0.007m/s. Therefore, it is thought that the standard deviation of approximately 0.06m/s is consistent with these results given the moderate dynamics of the vessel.

8.2.4.1 GPS gap bridging

Another test for the prediction accuracy of the INS is to simulate GPS outages, and investigate the drift rate of the standalone INS over time. This gives an indication of the performance that can be expected if a true GPS outage were to occur. This also validates that the INS is working as expected. Table 8.5 shows the maximum position errors obtained during simulated GPS outages of 15, 30 and 60 seconds. Two dynamic conditions are highlighted in the table to identify if there is a difference in performance when the vessel follows an approximately straight trajectory, and when the

vessel undergoes a turning manoeuvre.

		Maximum position error (m)		
Dynamics	Outage start time	15s	30s	60s
Turn	304200	1.53	4.87	18.58
	304350	0.33	1.07	2.94
	304500	0.47	2.91	13.26
	304800	0.58	2.93	11.05
	305150	1.49	5.11	24.12
Straight	304600	0.80	4.03	21.74
	304900	1.09	4.19	21.45
	305000	0.82	3.20	14.18
	305200	0.30	0.76	7.78
	305400	0.69	3.24	17.36

Table 8.5: Standalone INS performance during GPS outages

The table shows that the maximum position errors obtained during 15, 30 and 60 second gaps are 1.53m, 5.11m and 24.12m. The average errors for the 10 outages are 0.81m, 4.18m and 15.24m respectively. These results are better than were expected. Results from, for example Salychev et al. (2000) using a low cost INS sensor in a vehicle trial showed mean position drifts of 3m after 10 seconds, and 10m after 20 seconds. These values are RMS values for the whole outage whereas the values in Table 8.5 are maximum values taken from the end of the outage. In addition, the results in Salychev et al. (2000) increased to 10m and 23m after 10 and 20 seconds respectively when the vehicle accelerated during the outage. These results are significantly larger than the results obtained for the marine trial,

although it is noted that the in the vehicle trial, larger velocities were experienced.

An example of the drift during a GPS outage is shown in Figure 8.7 for the gap starting at 304800. The coordinates are given in metres in the north and east directions to give a clearer indication of the magnitude of the errors. The section of the vessel trajectory shown starts at coordinate (0,0). The green circles indicate the GPS updates that are available and the red crosses indicate the carrier phase position (truth) during the GPS outage. The figure shows that the vessel travels approximately 300m during the 60 second outage. The position error after 60 seconds is 11.05m which is approximately 4% of the total distance travelled. The figure shows how the INS is able to navigate over short distances with the error increasing over time.

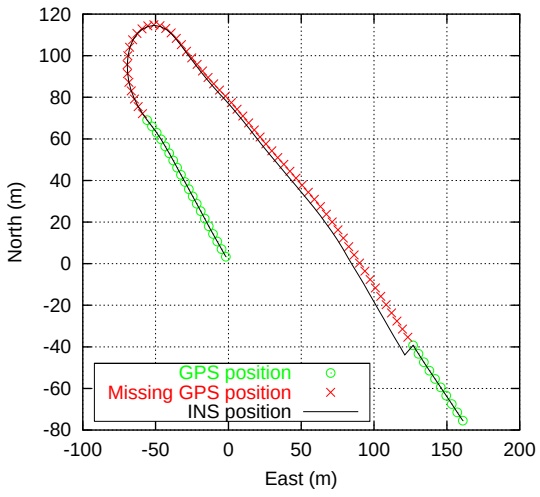


Figure 8.7: Standalone INS performance during 60 second GPS outage

The average maximum position error after 60 seconds for the turn and straight trajectories were 13.98m and 16.5m respectively. The error during turning manoeuvres was expected to be larger due to the increased output

of the sensors during this time. However this was not shown in the results. This is thought to be because the constant movement of the boat on the water results in the output of the sensors being similar during turns or straight trajectories. The results obtained during GPS gap bridging also showed that there was no significant difference in the attitude accuracy after the 60 second outage, and the average maximum velocity error after 60 seconds was 0.6m/s.

8.2.5 Marine trial error sources

The navigation accuracy obtained from the practical trials has been shown to be of lower accuracy than the results obtained when using simulation. The attitude accuracy obtained from the simulated data using the CKF is 0.01°, 0.02° and 0.20° for roll, pitch and yaw respectively, compared to 0.03°, 0.04° and 0.52° for the practical trial. This is thought to be due to a combination of the GPS and inertial sensor errors being larger than the manufacturer's specification, and the increase in the error of the GPS and INS system error sources. This also means that some of the error models that were used for data simulation need to be changed to reflect the errors obtained from the physical system.

For the GPS measurements, the Doppler measurements are significantly larger than those obtained using simulation. The Doppler measurements were consistent with the results obtained in Bruton (2000) indicating that the simulated Doppler error sources need to be modified to be consistent with the results obtained from real data. The first order Taylor approximation is considered to provide similar velocity accuracy to the receiver Doppler data following the results provided in Bruton.

The GPS and INS system errors are lever-arm error, synchronisation error and vibration. The total station measurements in §8.2.1.5 for the GPS separation were consistent with the tape-measured distances. However, the AHRS sensor was located in the front cabin which required the use of a survey pole and further tape measured differences. Therefore, the lever-arm error was expected to be larger for the GPS to IMU measurement.

The synchronisation error was analysed in §8.2.2 which resulted in the standard deviation of the cross correlation at approximately the millisecond level. It was also assumed that there was no synchronisation error in the POS/MV measurements. This is not the case as the POS/MV system will contain some timing errors, the magnitude and characteristics of which are not known. The cross correlation is insensitive to scale factor and bias errors, but the misalignment between the axes of the two sensors may introduce a timing error. Therefore these factors will contribute to the synchronisation error for the marine trial being larger than the error used during simulation. In the future it is recommended that manufacturers of low cost IMUs incorporate some form electronics to allow easier integration of the measurements with GPS.

The vibration of the sensor was not introduced in the simulations in Chapter 6 and Chapter 7. For the marine trial, the vibration of the sensor introduces a significant increase in the output of the sensors. Figure 5.3 shows the frequency response of the x-axis accelerometer with the engines on and the engines off, which shows a significant increase in the sensor noise across the frequency range, and in particular at the frequency of 35Hz. The standard deviation of, for example, the z-axis accelerometer increases from approximately 0.015m/s to 0.98m/s between the engines

on and the engines off. These estimates are only very coarse estimates of the noise in the output because the boat is not completely static, but this gives an indication that vibration affects the raw inertial sensor outputs significantly.

The inertial sensor errors of bias, scale factor and axis misalignment are also potentially larger in the marine trial than simulation. It is thought that the most significant error source is the bias variation in the sensor. The sensor measurements were corrected for temperature dependent variation using previously collected data. However, the repeatability of the temperature dependent bias may not be consistent. The bias estimates obtained still showed some temporal variation for both the gyros and accelerometers.

All of these error sources are thought to be responsible for the increased errors that are obtained using the real data. The most significant error that results from the marine trial is the yaw error. One possible method to reduce the yaw error is considered in the following section.

8.2.6 GPS attitude estimation

This section examines the use of GPS attitude measurements in the integration Kalman filter to improve the yaw estimation. GPS derived attitude measurements from multiple roving antennas can be used to reduce the drift that occurs in the yaw axis between the vessel undergoing horizontal acceleration manoeuvres. Three Ashtech receivers were mounted to the marine vessel as shown in Figure 8.1 to allow the capability for full GPS attitude estimation. No specific GPS processing software was available for processing the GPS attitude measurements, therefore the attitude measurements were obtained directly from the GPS positions. The yaw estimate is

obtained using the equation,

$$\psi = atan2\left(\frac{\delta E}{\delta N}\right) \tag{8.3}$$

where δN and δE is the difference between the positions of the GPS antennas in the north and east axes respectively. Roll and pitch measurements were also derived using the difference in height between the antennas.

In order to use the attitude estimates in the integration Kalman filter, the misalignment between the GPS and inertial sensor body frames has to be estimated. The misalignment between the two body frames was estimated by computing the direction cosine matrices for the GPS and INS systems. The misalignments were computed using POS/MV derived attitudes for the IMU direction cosine matrix because the drift is lower than for the AHRS sensor. The direction cosine matrix of misalignments is formed using the equation,

$$C_{b_G}^{b_I} = C_N^{b_I} C_{b_G}^N \tag{8.4}$$

where the subscripts G and I represent the GPS and inertial sensors respectively. The average of the misalignments estimated using Equation 8.4 is then applied to the GPS measurements so that the GPS measurements can be used as an attitude update for the integration filter.

The GPS attitude errors are given in Table 8.6 and the GPS yaw error is shown in Figure 8.8. The table shows that the roll and pitch errors are of lower accuracy than the yaw. This is because the height estimation from the GPS measurements is of lower accuracy than the horizontal position estimation. The attitude estimation is also affected by the antenna separation with the accuracy being improved over longer baselines. The results shown are consistent with attitude accuracy shown in, for example Zhang

(1995) and Lu (1995), using an Ashtech 3DF system and a similar baseline separation.

	Roll	Pitch	Yaw
RMS error (°)	0.173	0.301	0.155
Integrated system error (°)	0.035	0.042	0.091

Table 8.6: GPS attitude estimation

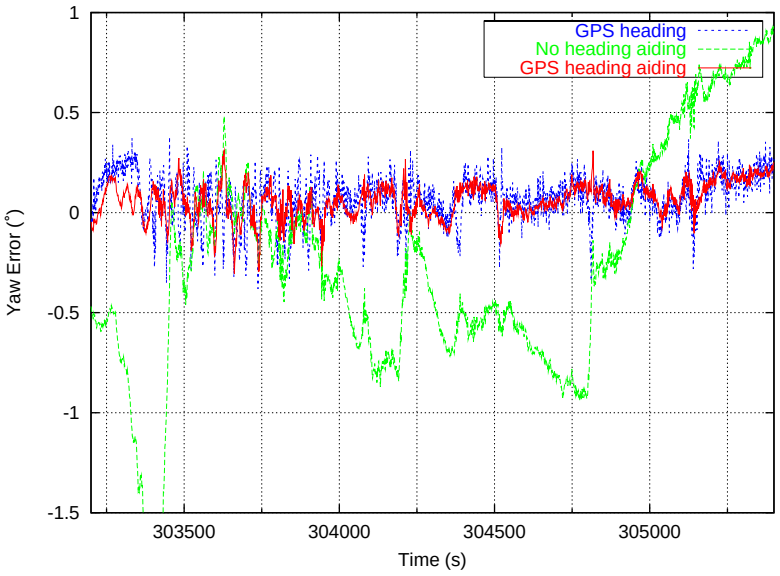


Figure 8.8: Integrated system yaw error using GPS yaw aiding

Because of the poor yaw estimation from the low cost INS integrated with a single GPS antenna, the GPS yaw estimate obtained from two of the GPS antennas was added as an estimate to the integration Kalman filter. The roll and pitch measurements were not used due to their poor accuracy and to reduce the overall potential system cost. Figure 8.8 shows the yaw error using GPS yaw aiding compared to the INS yaw error obtained in §8.2.3. The figure shows that the INS drift has significantly been reduced in the yaw estimate. The figure also shows that the INS is essentially

used to smooth the GPS yaw estimate. Unfortunately, the GPS heading error varies over time which effects the accuracy of the integrated solution. It is thought that the variation in the GPS heading accuracy is caused by multipath as other GPS error sources should effect the GPS range measurements equally.

Table 8.6 shows the attitude error from the integrated INS and dual antenna GPS system compared to the three antenna GPS system. The table shows that the INS system provides significantly improved attitude accuracy over the three antenna GPS attitude system in all axes. The roll and pitch errors both match the requirement for the multibeam sonar attitude accuracy specified in §8.2. The yaw error of 0.091° is an improvement over the GPS-only derived attitude and a significant improvement over the single GPS antenna system. However, the accuracy falls just outside of the 0.05° multibeam requirement. The GPS yaw accuracy can potentially be improved using a longer baseline, new GPS receivers with better multipath mitigation and proper estimation of the attitude parameters using least squares or a Kalman filter (Lu, 1995).

The low cost IMU and dual antenna GPS system provides a significant improvement in attitude accuracy over a multi-antenna GPS system. The cost of the IMU based system is potentially lower since the IMU costs less than a geodetic grade GPS receiver. The cost of the dual antenna system can also be reduced by using a single frequency receiver for the second antenna. The attitude reference from the IMU coupled with the ambiguities from the dual frequency receiver can be used to initialise the ambiguities for the single frequency receiver. The estimated cost of the hardware for such a system using commercial off the shelf components is

estimated to be less than £20,000. This is a significant reduction in the cost for a system such as the POS/MV which is based around a tactical grade IMU.

8.2.7 Adaptive Kalman filtering

This section examines the use of the adaptive Kalman filtering algorithms for the marine trial, that were used with simulated data in Chapters 6 and 7. Figures 8.9 and 8.10 show the alignment of z-axis gyro bias and yaw error respectively using the CKF, process noise scaling and MMAE filters. The adaptive-Q filters are considered separately in §8.2.7.4. The trajectory used is the same as that used in the simulation analysis which allows comparison of the results obtained in this section with those obtained in Chapters 6 and 7. The time axis used in the figures in this section is therefore changed to start at zero to correspond with the simulation analysis. Also, it is highlighted that the sensor bias error in Figure 8.9 is different to that used for simulation because the simulated bias values are formed using the 1-sigma error from the IMU specification. Therefore each processing run results in a random bias.

For the real data, truth is not available for the bias estimates. Instead, the estimated truth is formed using the CKF estimate from §8.2.6 which uses the GPS yaw aiding in the filter and initial bias estimates from multiple processing runs. The following sections investigate the performance during alignment for each of the filters.

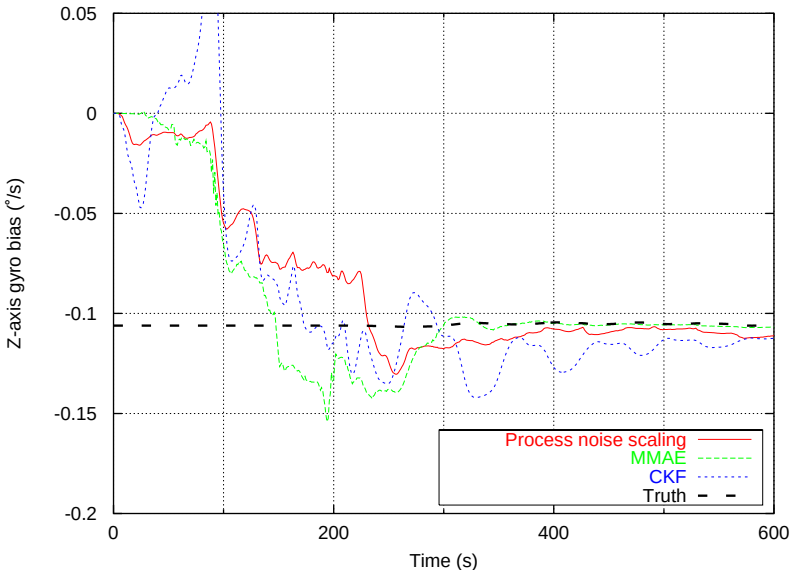


Figure 8.9: Comparison of z-axis gyro bias alignment

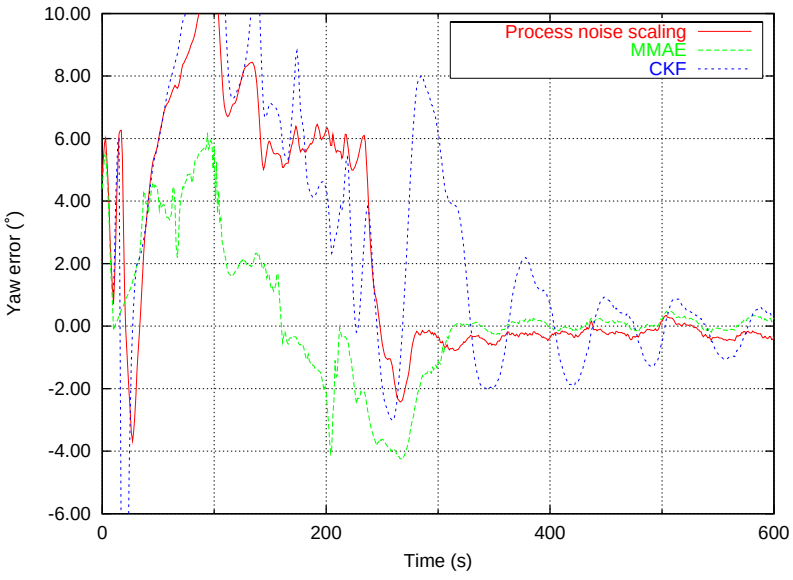


Figure 8.10: Comparison of yaw axis alignment

8.2.7.1 CKF

The CKF filter was initialised using the initial covariance matrix from §6.3.1 where the initial gyro and accelerometer bias variances are 0.01°/s and 0.01ms^{-2}. This initial covariance matrix was used because it is the best estimate of the initial errors that is available based on the specification of the inertial sensor. The *a priori* process noise matrix is taken from §8.2.3, since this matrix provides the best navigation performance once the INS is aligned.

Figures 8.9 and 8.10 show that the CKF estimate is slow to converge to the correct bias and yaw errors. This is because the process noise matrix used is the *a priori* matrix that is expected to give the smallest navigation errors when the system is operating during normal navigation. After 570 seconds the z-axis gyro bias is resolved to within 0.01°/s. In view of the larger yaw errors obtained using real data, alignment is considered to be achieved when the yaw error converges to within 1° of the POS/MV reference. The CKF estimate does not converge to within this value during the alignment manoeuvre (which ends at approximately 800 seconds). An alignment time is therefore not possible to obtain for the CKF estimate because after the alignment manoeuvre, the yaw error drifts between vessel turns. The failure to align the yaw error in this time is thought to be because the x and y-axis accelerometer biases are not fully resolved. After 800 seconds, they are only estimated to within 0.013ms^{-2} and 0.014ms^{-2} respectively. The accelerometer bias error is therefore absorbed by other states in the Kalman filter.

These results indicate that the CKF does not resolve the sensor biases

and align the INS because of using the *a priori* process noise estimate which is defined for normal navigation performance. This reflects the results that were obtained using the simulation analysis and highlights the requirement for the stochastic information to be adapted on-line.

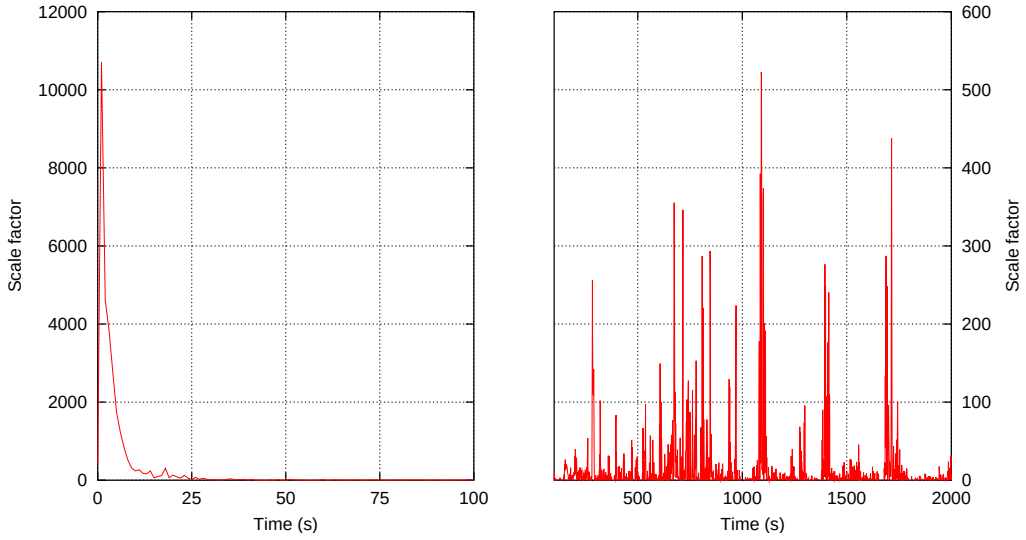
8.2.7.2 Process noise scaling

The z-axis gyro bias for the process noise scaling filter in Figure 8.9 shows that the bias is resolved to within 0.01°/s after 344 seconds. The yaw estimate is resolved to within 1° after 275 seconds. This is a significant improvement over the CKF estimate which indicates that the process noise noise scaling filter can be used to provide a rapid alignment using a simple modification to the CKF.

In order to verify the process noise scaling filter alignment time, the filter was also tested using different start times. Ten different start times were used, with each run starting 30 seconds later so that the final processing run began at 300 seconds. The average alignment time for the yaw error was 267 seconds which indicates that the algorithm works independently of dynamics.

The process noise scale factors for the first processing run are shown in Figure 8.11. Figure 8.11(a) shows that the initial scale factors are estimated to be large, with the magnitude quickly reducing after approximately the first 10 seconds. These large initial scale factors result in the covariance being increased which gives more weight to the measurements. Between 25 and 100 seconds, the scale factors are not larger than 50 which indicates that the covariance has been sufficiently increased by the process noise.

Figure 8.11(b) shows that the scale factors after the initial 100 epochs do



(a) Scale factor between 1-100 seconds

(b) Scale factor between 100-2000 seconds

Figure 8.11: Estimated process noise scale factors

not remain small like they did for the simulated data. Spikes occur in the scale factors which coincide predominantly with the turning manoeuvres in the vessel's trajectory. This is due to some of the INS errors becoming observable through the position error at these points. This is more significant in the real data because the INS drift is higher between manoeuvres. The process noise scaling filter results in more weight being given to the observations during these turns, therefore the filter continually responds to new measurements during navigation. The effect of this property on the attitude errors is considered in §8.2.7.5.

8.2.7.3 MMAE

The MMAE filter used in this section is formed using six models with the scale factors 0.1 to 10000. This is the same as the filter used in §7.4.1. The

six model filter is used because it provided the best performance during data simulation. The MMAE z-axis gyro estimate converges to within $0.01°/s$ after 280 seconds which is 64 seconds faster than the process noise scaling filter. However, the attitude alignment is not obtained to within 1° until 303 seconds which is 28 seconds slower. The average alignment time was 290 using the 10 processing runs as considered for the process noise scaling filter.

For the simulated data, the MMAE algorithm was able to align the INS during the period of low dynamics, however for the real data, both of the process noise scaling and MMAE filters still require a period of higher dynamics to resolve all of the errors. This is again expected to be because of the larger GPS, inertial sensor and system errors experienced by the real system.

Figure 8.12 shows the conditional probability estimate for the MMAE filter during the initial 600 seconds. The figure shows a similar characteristic to the MMAE filter using simulated data. The algorithm transitions from using the Kalman filters with the larger process noise scale factors to the filters using the smaller scale factors. After 200 seconds, the filter using the scale factor of 10 is identified as the dominant filter.

The figure shows that the MMAE filter uses the Kalman filters with the largest scale factors for a shorter time than during simulation. This is thought to be because the initial bias errors are slightly smaller for the real data. This shows that the adaptive filter adapts differently depending on the initial errors to be resolved. This is an essential quality of the adaptive algorithm as the initial bias errors are random and vary each time the INS is switched on.

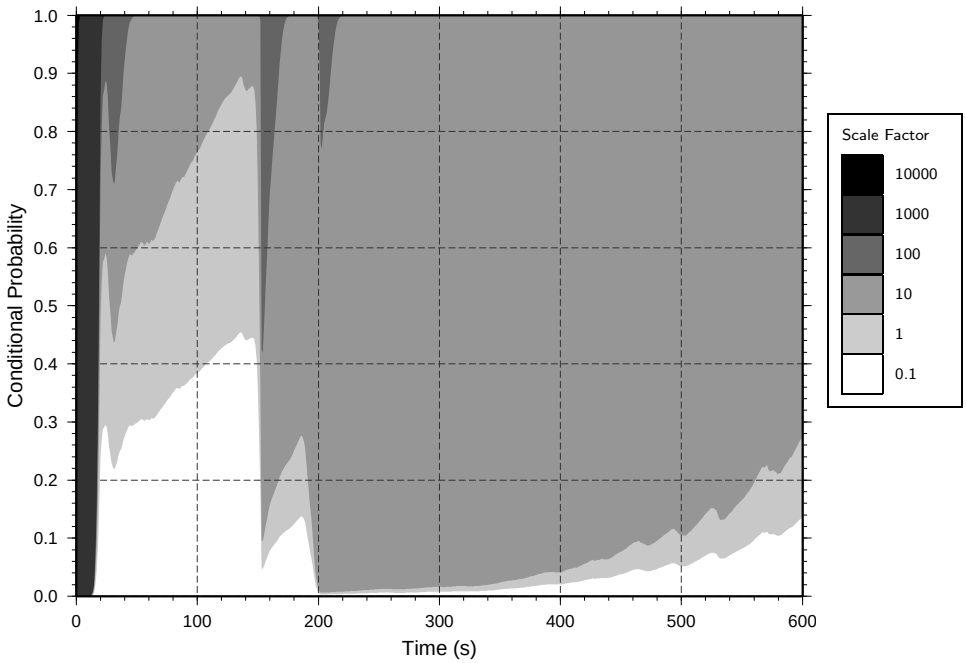


Figure 8.12: MMAE conditional probabilities during alignment

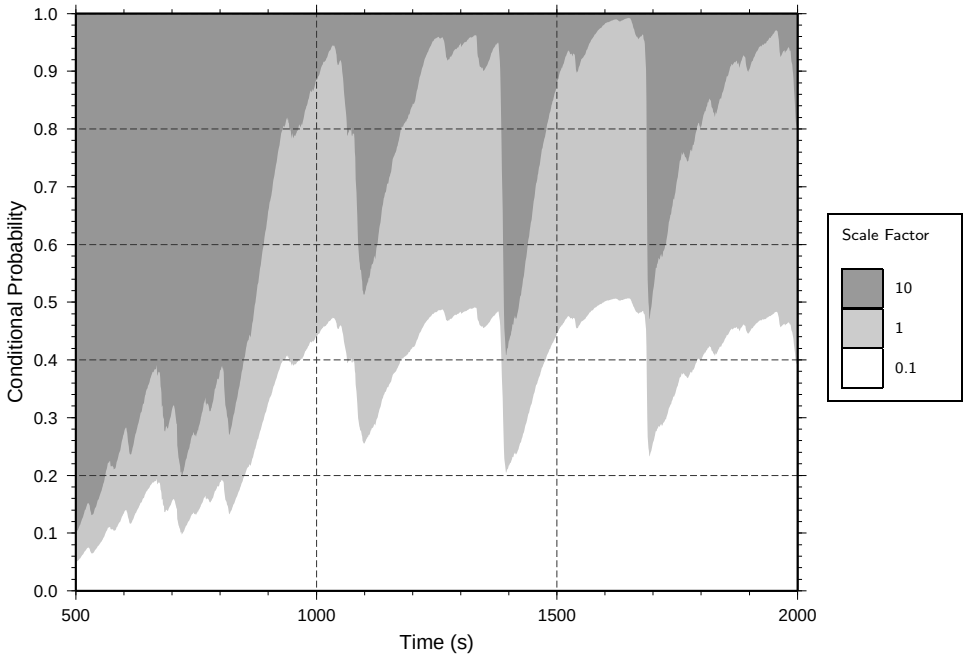


Figure 8.13: MMAE conditional probabilities during navigation

Figure 8.13 shows the conditional probabilities for the MMAE filter during normal navigation. Here it is shown that after the initial alignment period, the MMAE algorithm uses only the Kalman filters using the scale factors of 0.1, 1 and 10. This potentially means that some of the Kalman filters can be removed from the MMAE algorithm once alignment has been achieved. The figure also shows that the filter adapts to use filter with the larger process noise scale factor during the turning manoeuvres. This is similar to the process noise scaling algorithm, except the transition is smoother. The result of this characteristic on the attitude estimation is considered in §8.2.7.5.

8.2.7.4 Adaptive-Q

The innovation and residual based adaptive-Q filters both result in a slow time to alignment and large navigation errors for the real data. The adaptive Kalman filter yaw errors are shown in Figure 8.14 for the entire processing run. The filters were initialised using the process noise and initial covariance matrices from §7.3.1. The figure shows that neither of the filters are able to resolve the yaw error to within 1° during the alignment manoeuvre. The figure does show that after approximately 500 seconds, the yaw error converges, but the estimate is noisy. After the alignment section of the trajectory is completed, the yaw error suffers from a larger drift than the CKF indicating that the bias estimates are not resolved as accurately. The attitude errors for the adaptive-Q filter are considered in the following section.

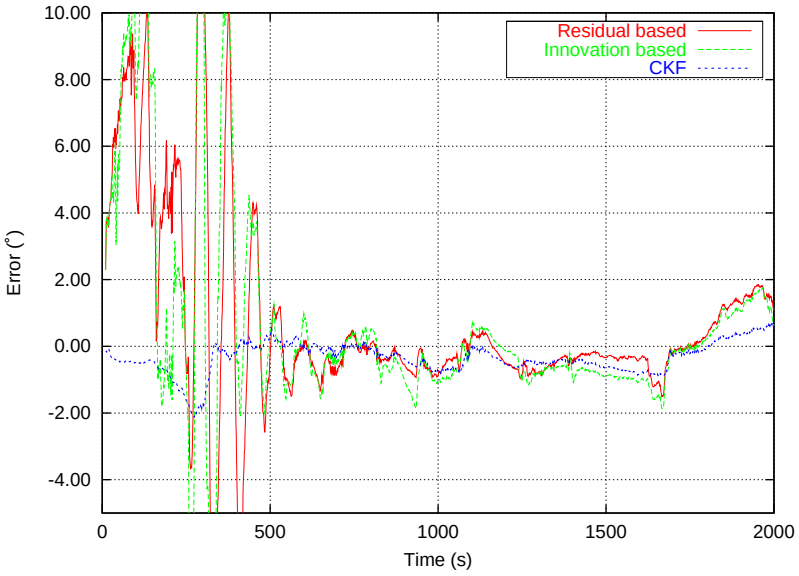


Figure 8.14: Adaptive Kalman filter attitude error

8.2.7.5 Attitude errors

Table 8.7 shows the attitude accuracies obtained using the adaptive Kalman filters compared to the CKF. The table shows that the attitude accuracy in all axes of the adaptive-Q filters is significantly worse than the performance of the CKF. However, both the process noise scaling and MMAE filters result in an improvement in the yaw estimation. The MMAE filter also resulted in a small improvement in the roll and pitch estimation although this is not thought to be significant. The improvement in the yaw estimation is thought to be due to the filters continually adapting to the new information from the innovation sequence as described in the previous sections. This predominantly occurs during vessel turns when some of the errors become observable. The adaptive filters identify that there is a deficiency in the stochastic modelling and attribute this to the INS.

	Roll	Pitch	Yaw
CKF	0.034	0.041	0.522
Process noise scaling	0.040	0.035	0.357
MMAE	0.032	0.034	0.448
Innovation based Adaptive-Q	0.180	0.118	0.832
Residual based Adaptive-Q	0.123	0.100	0.651

Table 8.7: Attitude errors for adaptive Kalman filters

8.2.7.6 Summary

This section has showed that the process noise and MMAE filters result in a faster time to alignment, and improved yaw estimation over the CKF. The CKF algorithm resulted in the CKF not managing to resolve all of the inertial sensor errors even after a prolonged period of high dynamic manoeuvres. The process noise scaling and MMAE filters were able to successfully adapt the stochastic information to provide average alignment times of 267 and 290 seconds respectively. The yaw estimation was also improved over the CKF using these adaptive filters. The adaptive-Q algorithms were unable to successfully adapt the process noise matrix which resulted in a longer alignment time and poor navigation performance.

8.3 Van Trial

This section investigates using adaptive Kalman filtering for integrating GPS and INS for vehicle navigation. Vehicle navigation is a potential area for the use of low cost INS as the INS can be used to bridge periods

when the GPS satellite signals are obstructed. For the vehicle trial, truth information is not available due to not having access to a higher grade system for comparison. Consequently, the performance of the algorithms is examined by investigating the drift of the sensor during GPS outages. Improved compensation of the inertial sensors and alignment of the INS should result in improved performance during gap bridging.

8.3.1 Sensor configuration

Figure 8.15 describes the method used to collect and time-tag the IMU data for the van trial. The figure shows that the serial data stream from the AHRS output was collected using a dedicated notebook PC. Data collection software written by the author was used to collect the serial data. The data collection software was developed due to the poor reliability of the Crossbow GyroView software. The GyroView software frequently missed several epochs of data on previous data collection trials. The AHRS sensor was configured to sample data in voltage mode which is the same mode as used for the marine trial.

A second notebook PC was used to collect the analogue data from the AHRS sensor. Since the AHRS was configured to collect data in voltage mode, only the accelerometer measurements were available using the analogue outputs. The analogue output from the AHRS was sampled at 400Hz to ensure a high resolution for time tagging the IMU measurements. The A-D card used was a Measurement Computing DAS16/16, which is the latest version of the ADAC PCM-5516-16 A-D card that was used in the marine trial. The data was collected using analogue data collection and timing software developed by the author. The data collection software is a

Figure 8.15: Van trial equipment configuration

Figure 8.16: Crossbow AHRS-DMU-HDX and Leica choke ring GPS antenna mounted on roof of van

modified version of the data collection software developed for Section 8.2. The modified software allows the data to be sampled and accurately time tagged with GPS time in real-time. The software also provides a visual interface to configure and check the validity of the raw measurements.

The time tagging of the inertial measurements is achieved using the same principle as described in Section 8.2.2. However, in this trial the Leica System 530 GPS receiver was used which, in addition to logging the received event times, outputs the precise time-tag of the event over the serial output in the form of a PLEI string (for more information see Leica Geosystems (1999)). This data collection principle is shown in Figure 8.15. Here it is shown that the analogue output from the AHRS is sampled using the analogue data collection notebook PC containing the DAS16/16 A-D card. The A-D card sends a pulse to the event input of the GPS receiver and the GPS receiver returns the precise time-tag over the serial output which is received by the data collection software.

As a result of using the modified data collection software, in future trials, the software can be used to provide integration of the GPS and INS measurements in real-time. This capability was not used in this trial due to the requirement for line of sight between the base and roving receiver for differential corrections, and also due to the hardware restrictions of sampling the AHRS raw analogue output described in Section 8.2.1.2.

Figure 8.16 shows the choke ring antenna and AHRS attached rigidly to the roof rack of the van used for the field trial. Due to the small separation between the antenna and the AHRS, the lever-arm separation was measured using a steel ruler. The AHRS was not aligned to the vehicle frame due to the restrictions of the AHRS thread mount.

In addition to the roving GPS antenna and IMU, a static reference GPS receiver was set up on the roof of the IESSG. The baseline separation between the reference and rover is less than 2km for the entire dataset.

8.3.2 Sensor synchronisation

Analysis of the data collected using the A-D output from the AHRS sensor demonstrated that the A-D data was noisier than expected. This was thought to be due to the cable used to connect the AHRS sensor to the A-D card being of poor quality and subject to signal interference. In addition, the raw gyro voltage measurement is not available using the analogue output from the AHRS. Therefore, the serial data from the AHRS sensor had to be used, and cross correlation was used to match the serial data to the time stamped analogue data. This is similar to the data collection method in Section 8.2.2, but should be improved because both data collection systems are sampling the same sensor.

Figure 8.17 shows the cross correlation between the serial and analogue accelerometer output. The serial measurements were sampled at the maximum output of 166Hz and the analogue measurements were sampled at 400Hz.

The figure shows firstly that each accelerometer measurement is offset by a constant time difference of approximately 0.4ms. This is caused by the A-D sampling method of the DAS16/16 A-D card as described in Section 5.4.1. The time difference corresponds to the channel skew which is obtained from the A-D sampling frequency of 400Hz divided by the 6 sampled channels. This results in an inter-channel offset of 0.417ms. Consequently, it can be assumed that the AHRS sensor does not suffer

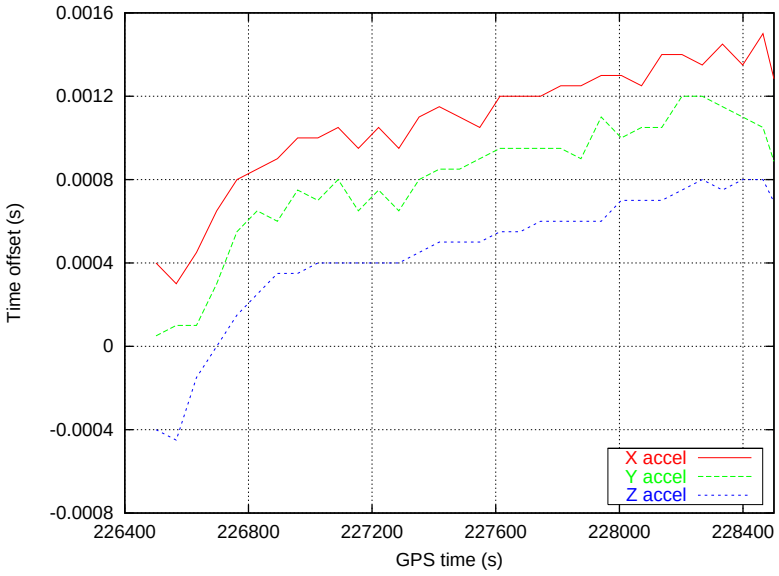


Figure 8.17: Cross correlation

from the inter-channel time skew that occurs using the DAS16/16.

The figure demonstrates that the cross-correlation method allows the time-tag to be attached to the serial data with a high degree of accuracy. The time-tags attached to the serial data were corrected for the drift that was estimated by fitting a polynomial to the data. This resulted in the standard deviation of the cross correlation of less than 1×10^{-5}s for each channel. Using the analysis from this section and §5.4.1, the time tagging accuracy for the dataset is estimated to be at the millisecond level.

8.3.3 Trajectory description

The trajectory for the van trial is shown in Figure 8.18. The trial was performed in an industrial estate located approximately 1.5km south of the IESSG building in Nottingham. The location was selected as there are only a few obstructions that were likely to obstruct the GPS signals. Typical speeds that were reached during the trial were 20-30 miles per

hour. Figure 8.18 shows that the trajectory involved a high number of turns, particularly where the van was driven around a large car park. The turns are expected to ensure that the INS alignment accuracy is maintained throughout the trial.

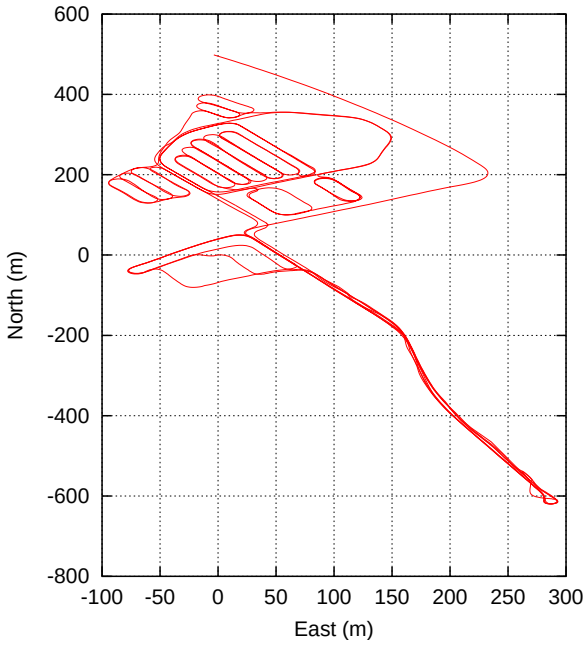


Figure 8.18: Van trial trajectory

8.3.4 CKF innovation sequence

The innovation sequence for north and east positions using the CKF is shown in Figure 8.19. The epoch separation used is 1 second, therefore the figure shows the prediction accuracy of the INS over this time. The figure shows that for the initial two minutes, the innovation sequence is small. In fact the standard deviation is 9mm/s and 6mm/s for the north and east axes respectively. The innovation sequence is small because the van is static with the engines off, the GPS integer ambiguities are fixed, and the initial errors are estimated using a coarse alignment.

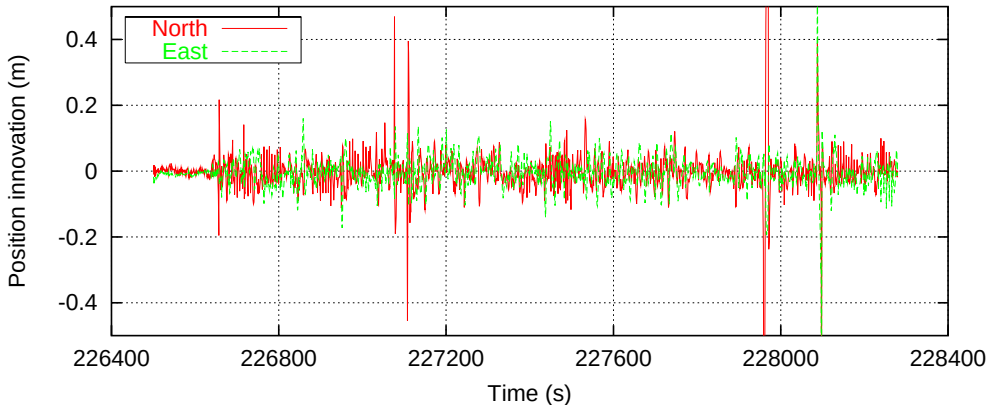


Figure 8.19: Van trial innovation sequence

Because the vehicle is stationary, the initial gyro biases can be estimated by averaging their outputs. A coarse roll and pitch alignment is achieved using Equation 2.15 with the outputs of the accelerometers. The initial accelerometer biases are estimated to be zero. Additionally, the initial yaw estimation is achieved using the change in position of the van before it came to a standstill. Because of this initialisation, the INS does not result in a large initial position error as experienced in the marine trial.

When the van accelerates from being stationary, the residual sequence increases. From the figure, is is also shown that the innovation sequence contains several spikes, for example at 227100 and 227950. These spikes occur where the GPS integer ambiguities are lost due to signal obstructions. At these points, the GPS position is provided by the pseudorange observations.

8.3.5 Adaptive estimation

The analysis of the residual sequence in the previous section identified that the stochastic properties of the INS are different to those experienced in

the marine trial. This is because static initialisation can be used to reduce the initial alignment and sensor errors. The analysis also showed that the GPS position accuracy changes due to obstructions such as buildings and trees blocking the GPS signals. This section looks at the characteristics of the adaptive filters for the vehicle trial.

8.3.5.1 Initialisation and alignment

The adaptive Kalman filters for the vehicle trial were configured in the same way as used for the marine trial. The only difference is that the initial covariance for the gyro bias was reduced to $1\times10^{-4\circ}$/s due to its improved estimation using the static initialisation.

The improved estimation of the initial gyro biases for the vehicle trial has a significant impact on the estimation of the stochastic properties of the INS by the adaptive algorithms. At the beginning of the dataset, the estimated scale factors for the process noise scaling filter are all computed to be less than 40. Similarly, for MMAE filter, the filter initially forms a weighted combination of the Kalman filters that use the scale factors of 0.1, 1 and 10. Once the van moves from stationary, the filter converges to predominantly use the filter with the scale factor of 10. For the adaptive-Q filters, the initial covariance for the gyro bias was set to 1×10^{-3} to ensure that the gyro bias is continually estimated during the trial. After the initial 20 seconds of conventional Kalman filtering for the adaptive-Q filters, the estimated process noise is significantly reduced from the initial estimate.

Because there was no attitude reference available for the vehicle trial, the time to resolve the initial attitude is not possible to ascertain. However, the above analysis identifies that the adaptive filters have adapted to model

the different initial stochastic properties that are experienced for the vehicle trial.

8.3.5.2 Gap bridging

Because there is no attitude reference for the trial, the adaptive estimation algorithms are tested by investigating the gap bridging accuracy of the INS. The performance of the INS during a GPS outage depends on the resolution of the inertial sensor errors, and the alignment of the INS. Therefore, if the position accuracy is improved during a GPS outage, it is inferred that the errors have been better estimated. Table 8.8 shows the maximum position errors after 15, 30 and 60 second GPS outages for each of the adaptive Kalman filtering algorithms used in this thesis. The results are obtained by averaging the errors from 10 separate GPS outages. The GPS outages were simulated which allows the carrier phase position at the end of the outage to provide the truth.

	Outage length (s)		
	15	30	60
CKF	2.33	7.36	23.61
Process noise scaling	1.87	6.81	15.23
MMAE	1.82	5.42	14.98
Innovation adaptive-Q	2.52	8.45	33.02
Residual adaptive-Q	2.43	8.42	33.26

Table 8.8: Standalone INS maximum position errors in metres during GPS outages

From the table, it is again confirmed that the process noise scaling and MMAE filtering algorithms result in better estimation of the INS errors. The table shows that after a 60 second GPS outage, the process noise scaling and MMAE filters result in a 36% and 37% improvement in position estimation over the CKF. The average distance travelled during the outage is 540m. Therefore, the average maximum error is approximately 3% of the distance travelled for these filters.

For the adaptive-Q filters, the gap bridging accuracy is deteriorated over the CKF. This indicates that the adaptive-Q filter is not able to estimate the process noise matrix for the INS effectively. However, for the residual based filter, the gap bridging accuracy was significantly improved by the end of the processing run. Using the last five simulated outages, the average maximum position error was computed to be 15.68m, compared to 24.90m for the CKF. This indicates that the residual based adaptive-Q filter has taken longer to align the INS. Once alignment has been achieved, the position accuracy during the outage is improved over the CKF indicating that the stochastic information is being modelled effectively.

The position errors during the outage are a significant improvement over the results presented in Salychev et al. (2000), which are described in §8.2.4.1. This is thought to be partly due to the constant turns in the trajectory which improves the observability of the errors and restricts the error growth. Further analysis of the vehicle trial using different trajectories is necessary to reach full conclusions about the full bridging accuracy of the GPS and low cost system. The main point from the van trial is to highlight the improved error estimation using adaptive Kalman filtering.

Chapter 9

Summary and Conclusions

This chapter gives an overview of the research undertaken for this thesis and gives conclusions about the results that have been obtained. The chapter concludes with recommendations for future research in the area of GPS and low cost INS integration and adaptive Kalman filtering.

9.1 Summary

The objective of the research undertaken in this thesis is to investigate the integration of GPS and low cost INS measurements and examine potential ways in which the standard Kalman filter integration algorithms can be improved. In order to achieve this objective, GPS and inertial measurement data simulation software was used and extended. GPS and INS integration software was also developed which provides the capability of current integration algorithms. The integration software was extended further to allow the investigation of a number of adaptive Kalman filtering algorithms that provide the ability to adapt the *a priori* stochastic information on-line in real time. The tools were then used to examine the characteristics of

the adaptive algorithms for improving the error estimation for a GPS and low cost INS system. In particular, this involved examining the algorithms during the initial alignment phase of the integration process when the INS attitude and inertial sensor errors are resolved. Finally, two practical trials were conducted using GPS and a low cost IMU in a marine and land-based vehicle environment. The performance from such a low cost system was evaluated, and the performance of the new integration algorithms was also tested.

9.2 Conclusions

The following sections summarise the results and conclusions from the research undertaken in this thesis. This section splits the conclusions into two main areas: results obtained using data simulation and results obtained using practical trials.

9.2.1 Simulated data

GPS and INS integration represents one of the most difficult challenges for Kalman filtering as a large number of states are estimated, some of which have low observability. Three adaptive Kalman filtering algorithms were examined in detail using simulated data. The principle findings from each of algorithms are summarised as:

- **Process noise scaling:** A new algorithm was developed that is used to increase the process noise based on the information contained in the innovation sequence. The algorithm requires very little modification to the CKF algorithm, and requires negligible extra processing. The

algorithm was shown to provide a 42% reduction in the time required for INS alignment compared to the CKF.

- **Adaptive-Q filter:** The adaptive-Q filter was shown to provide some of the characteristics that are required for rapid alignment and initialisation. However, it was shown that the quality of the stochastic information used to initialise the filter is essential in the convergence of the algorithm. The algorithm only converged if the initial estimate of the process noise is larger than the true value. The innovation and residual based algorithms resulted in a reduction in the alignment time over the CKF of 42% and 38% respectively.

- **MMAE:** The MMAE algorithm was implemented for the first time for the application of GPS and INS integration. The computational increase that results from using the MMAE algorithm was shown to be significant. With modern processing technology, however, the MMAE algorithm was shown to be able to meet the latency requirements for a real-time navigation system. The algorithm resulted in the most rapid time to alignment of any of the adaptive algorithms with a 72% reduction in the time required for alignment over the CKF. The alignment was also achieved without the vessel undergoing any special alignment manoeuvres.

  The MMAE was shown to converge predominantly to a single filter once alignment had taken place. Different algorithms were tested that force the MMAE filter to remain adaptive. The threshold value for the conditional probabilities was identified as the best method to achieve this. Modifications to the MMAE to reduce the com-

putational burden were examined but none were able to match the performance of the standard MMAE algorithm.

The following general conclusions about the analysis of the adaptive Kalman filters can be made:

- Using data simulation, all of the adaptive Kalman filtering algorithms were shown to result in a reduction in the time required for alignment.
- Adaptive Kalman filtering is more convenient as it reduces the requirement for *a priori* information.
- The CKF alignment time was obtained by empirically estimating the initial covariance to result in the fastest alignment time which would normally not be possible. Apart from the adaptive-Q filters, the dependence on the initial covariance was significantly reduced using adaptive Kalman filtering.
- None of the adaptive algorithms resulted in an overall improvement in navigation performance with the simulated data. This is thought to be because the stochastic properties of the simulated data remain relatively constant throughout the processing run.
- None of the adaptive algorithms provide a complete solution to the adaptation of the stochastic information. For example, the MMAE algorithm resulted in the best performance at the expense of a significant computational increase.

From the above points, a broader conclusion can be made about the use of adaptive Kalman filtering algorithms. It has been shown that there are

potential improvements that can be made to the CKF when the stochastic information in the filter cannot be considered constant, or is not fully known. These findings have repercussions for a broad range of applications that use Kalman filtering. Some of the navigation based applications are considered in the final section for ideas for future work.

9.2.2 Practical trials

Practical trials were undertaken in this thesis for three principle reasons. Firstly, in the case of the marine trial, the attitude accuracy of a GPS and low cost INS system was analysed for the application of geo-referencing a multibeam sonar for hydrographic surveying. Secondly, the trials were used to investigate the performance of the adaptive Kalman filtering algorithms, and finally, they were used to validate the results obtained using data simulation and identify areas in which the simulation algorithms need to be modified.

The conclusions reached using the marine trial with the conventional Kalman filter are summarised as:

- **Single antenna marine trial:** Using measurements from a single GPS antenna, the roll and pitch accuracies achieved were 0.03° and 0.04° respectively. The yaw error was significantly larger at 0.52° due to a combination of the inertial sensor errors and low observability between the vessel performing turning manoeuvres.

- **Dual antenna marine trial:** To reduce the yaw drift between manoeuvres, measurements from two roving GPS receivers installed on the vessel were used in the integration Kalman filter. The dual

antenna GPS and low cost IMU system was demonstrated to provide an attitude accuracy of 0.04°, 0.03° and 0.09° in roll, pitch and yaw respectively. The yaw accuracy falls just outside of the attitude requirement of 0.05° for geo-referencing a multibeam sonar but can potentially be enhanced by improving the GPS attitude estimation (see §9.3). The attitude performance from such a system is an improvement over a multiple antenna GPS attitude system and is lower cost.

The following conclusions are drawn from the analysis of the adaptive Kalman filtering algorithms used with the real data.

- The CKF failed to fully resolve the inertial sensor errors for the marine trial even after a prolonged period of high dynamic manoeuvres. This again highlighted that some form of adaptation of the stochastic properties is required at the beginning of the processing run for alignment to be achieved rapidly.

- The process noise scaling and MMAE filters resolved the sensor errors and aligned the INS attitude within 6 minutes for the marine trial. The yaw alignment accuracy was also improved for these filters, with an improvement over the CKF of 32% and 14% respectively. This shows that the process noise scaling and MMAE filters provide the capability to adapt the stochastic information for the INS to result in reduced time to alignment and improved attitude estimation.

- The innovation and residual based adaptive-Q filters were slow to resolve the initial errors and align the INS in the marine trial. Nav-

igation performance was deteriorated compared to the CKF using these filters.

- For the vehicle trial, the gap bridging accuracy for the process noise and MMAE filters demonstrated an improvement of 36% and 37% over the CKF. This indicates that these filters are able to estimate the stochastic properties of the INS better than the *a priori* information used in the CKF.

- For the vehicle trial, the adaptive-Q filters both resulted in a deterioration of the gap bridging accuracy. However, some promise was shown with the residual based filter as the gap bridging accuracy was improved later on in the dataset.

Finally, the following conclusions about the use of data simulation for analysing GPS and INS integration can be reached:

- The results from the practical trials using the adaptive Kalman filter reflected the characteristics of the results obtained using the simulation analysis.

- The absolute attitude accuracies and alignment times were different using the simulated data to those obtained using the practical trial. This was thought to be predominantly due to the error values used in the modelling of the simulated data. These values need to be adjusted in future trials to better reflect those obtained from practical trials.

- Data simulation is a valuable tool for investigating the properties of new algorithms. All of the error sources are controllable and

most importantly, truth information is provided for all navigation parameters.

- Data simulation is an iterative process that requires comparison with data obtained from practical trials. The results from practical trials can be used to update the simulated errors.

9.3 Future recommendations

The ideas presented in this thesis can be tested in a broad range of application areas. The research has shown that it is possible to modify the conventional Kalman filtering algorithm to improve the state estimation. The following recommendations are suggested for future research in the area of GPS and low cost INS integration, adaptive Kalman filtering and navigation system simulation:

- **Simulation:** The development of the GPS and inertial measurement simulation software at the IESSG is an ongoing area of research. Recently (summer 2003), the GPS simulation algorithms have been developed for the Future Real-time Location and Navigation (FURLONG) study (Moore et al., 2003). This is a 12 month study into simulating signals for future satellite positioning systems (ibid.). For this study, the simulator has been extended to include measurements from modernised GPS and Galileo satellites. The GPS simulation models have also been modified. In particular, the atmospheric delay models have been modified to provide better modelling of regional variations. See, for example, Moore et al. (2003) and Smith et al. (2003). This demonstrates that navigation system simulation has an

important role in the development and analysis of future processing algorithms and navigation systems.

Data simulation needs to be a recursive process where the results obtained from real data are applied to the simulated data and vice versa. It is recommended that more research is carried out into modifying the error values used in simulation to match those experienced with real data. In addition, the inertial sensor error models can be extended to better characterise individual sensor types. Another extension that has not been considered in this thesis is the change in the sensor characteristics as the sensor warms up after the initial switch-on. This can be incorporated into the existing data simulation algorithms.

Other future recommendations for research using the data simulator include examining the impact of the new satellite observations. By 2008, the European satellite navigation system, Galileo, is expected to be fully operational (Ochieng et al., 2001). Simulation provides an essential part of examining the potential performance from new navigation systems.

- **GPS and low cost INS for marine applications:** The results shown in this thesis for the GPS and low cost INS marine trial are promising. More investigation is required using different qualities of inertial sensor to identify the impact that this has on the attitude accuracy. Also, the effect of different system errors such as lever-arm offset and vibration on the attitude needs to be investigated further. For low dynamic applications such as the marine environment, GPS

attitude measurements will be required to restrict the error growth, predominantly in the yaw axis. The attitude determination from the GPS measurements performed in this thesis used the GPS positions to directly derive the attitude parameters. Algorithms exist, for example in Lu (1995), that can be used to improve the attitude estimation by using an algorithm such as least squares or Kalman filtering. The impact of improved estimation of the GPS attitude on the low cost INS based system is required to determine the full potential accuracy from such a system.

- **Adaptive Kalman filtering:** Further testing of the adaptive algorithms needs to be performed in a variety of environments. For example, flight tests and higher dynamic vehicle trials.

  The adaptive-Q algorithm has been shown to work successfully in Mohamed and Schwarz (1999) for a tactical grade INS. Therefore it is suggested that the algorithms are applied to different grades of inertial sensor, and also different types of GPS receiver. If the estimation problem is not restricted to the real-time case, the performance of the algorithms combined with Kalman filter smoothing can also be investigated.

  The adaptive algorithms can also be applied to a wide range of applications (not necessarily for navigation) where the stochastic properties of the processes being modelled vary over time. One possible application identified in the IESSG is using the algorithms for integrating GPS and accelerometers in bridge deflection monitoring (see Roberts et al. (2002)). The triad of accelerometers used in bridge

deflection monitoring results in the same problems of, for example, initialisation as experienced with this research.

It is hoped that the research presented in this thesis can be successfully applied to other applications in the future.

References

ADAC Corporation, 2002. PCM-5516-16 Manual 1.3.

Agrotis, L., 1984. Determination of satellite orbits and the global positioning system. Ph.D. thesis, University of Nottingham.

Alkan, R., Baykal, O., 2001. Survey boat attitude determination with GPS/IMU systems. Journal of Navigation , 135–144.

Baker, H. C., 1998. GPS water vapour estimation for meteorological applications. Ph.D. thesis, University of Nottingham.

Barbour, N., Madden, P., Socha, M., 1996. A micromechanical gyro package with GPS under development for small pointing satellites. In: The 10th Annual AIAA Utah State University Conference on Small Satellites.

Barnes, J. B., Ackroyd, N., Cross, P. A., July 1998. Stochastic modelling for very high precision real-time kinematic GPS in an engineering environment. In: XXI International Congress of FIG, Brighton.

Bingley, R. M., Roberts, G., May 1998. Global positioning system. Tech. rep., IESSG, University of Nottingham.

Bisnath, S. B., 2000. Efficient, automated cycle-slip correction of dual-frequency kinematic GPS data. In: Proceedings of ION GPS 2000, The 13th Technical Meeting of the Satellite Division of the Institute of Navigation, Salt Lake City, Utah. pp. 145–154.

Bona, P., Tiberius, C., March 2000. An experimental comparison of noise characteristics of seven high-end dual frequency GPS receiver-sets. In: Proceedings of the IEEE Position Location and Navigation Symposium, San Diego, California.

Borre, K., Tiberius, C., September 2000. Time series analysis of GPS observables. Proceedings of ION GPS 2000, The 13th Technical Meeting of the Satellite Division of the Institute of Navigation, Salt Lake City, Utah .

Britting, K., 1971. Inertial Navigation Systems Analysis. John Wiley and Sons Inc.

Brown, R. B., Hwang, P. Y. C., 1997. Introduction to Random Signals and Applied Kalman Filtering, 3rd Edition. John Wiley and Sons Inc.

Bruton, A. M., December 2000. Improving the accuracy and resolution of SINS/DGPS airborne gravimetry. Ph.D. thesis, University of Calgary.

Burden, R., Faires, J., 1993. Numerical Analysis, 5th Edition. Brooks/Cole Publishing Company.

Busse, F., How, J., Simpson, J., 2002. Demonstration of an adaptive extended Kalman filter for low earth orbit formation estimation using CDGPS. In: Proceedings of ION GPS 2002, The 15th Technical Meeting of the Satellite Division of the Institute of Navigation, Portland Oregon.

Cannon, M., Lachapelle, G., Szarmes, M., Hebert, J., Keith, J., Jokerst, S., 1997. DGPS kinematic carrier phase signal simulation analysis for precise aircraft velocity determination. Navigation 44, 231–245.

Caspary, W. F., 1987. Concepts of network and deformation analysis. Monograph, School of Surveying, The University of New South Wales.

Chao, C. H. J., 1996. Improved modelling of high precision wide area differential GPS. Ph.D. thesis, University of Nottingham.

Chatfield, C., 2003. The analysis of time series: an introduction, 6th Edition. Chapman and Hall.

Chen, W., 1992. Integration of GPS and INS for precise surveying applications. Ph.D. thesis, University of Newcastle Upon Tyne.

Chiang, K., Noureldin, A., El-Sheimy, N., 2003. Multisensor integration using neuron computing for land-vehicle navigation. GPS Solutions 6, 209–219.

Collins, J. P., Langley, R. B., March 1999. Possible weighting schemes for GPS carrier phase observations in the presence of multipath. Tech. Rep. DAAH04-96-C-0086 / TCN 98151, U.S. Army Corps of Engineers Topographic Engineering Center.

Comp, C. J., Axelrad, P., January 1998. Adaptive SNR-based carrier phase multipath mitigation technique. IEEE Transactions on Aerospace and Electronic Systems 34 (1), 264–276.

Conover, W., 1999. Practical nonparametric statistics, 3rd Edition. New York.

Cramer, M., Stallmann, D., Haala, N., 2000. On the use of GPS/inertial exterior orientation parameters in airborne photogrammetry. International Archives of Photogrammetry and Remote Sensing XXXIII, 198–205, part B3.

Cross, P. A., 1994. Advanced Least Squares Applied to Position Fixing. University of East London, Department of Land Surveying.

Crossbow Technology Inc., July 1999. DMU-HDX-AHRS User's Manual Version 1.2.

Crossbow Technology Inc., 2000. AHRS300CA product specification. [online]. Available at: <http://www.xbow.com> [Accessed 20 January 2001].

De Jonge, P., Tiberius, C., 1996. The LAMBDA method for integer ambiguity estimation: implementation aspects. Tech. Rep. 12, Delft Geodetic Computing Centre, lGR-Series.

Dodson, A., Moore, T., Pattinson, M., November 2001. Measuring tropospheric delay in a moving receiver. In: Proceedings of ION GPS 2001, The 14th Technical Meeting of the Satellite Division of the Institute of Navigation, Salt Lake City, Utah.

Dorobantu, R., Zebhauser, B., 1999. Field evaluation of a low-cost strapdown IMU by means GPS. Ortung und Navigation 1 , 51– 65.

Evans, A. J., 2001. Long baseline kinematic GPS. Ph.D. thesis, University of Nottingham.

Farrell, J. A., Barth, M., 1999. The Global Positioning System and Inertial Navigation. McGraw-Hill.

Ford, T., Neumann, J., Bobye, M., 2001. OEM4 inertial: An inertial/ GPS navigation system on the OEM4 receiver. In: Proceedings of the International Symposium on Kinematic Systems in Geodesy, Geomatics and Navigation (KIS), Banff, Canada.

Gelb, A. (Ed.), 1974. Applied Optimal Estimation. Analytic Sciences Corporation.

Grewal, M. S., Andrews, A. P., 1993. Kalman filtering: theory and practice. Prentice Hall.

Grewal, M. S., Weill, L. R., Andrews, A. P., 2001. Global Positioning Systems, Inertial Navigation, and Integration. Wiley-Interscience.

Gustafson, D., Dowdle, J., Flueckiger, K., March 2000. A deeply integrated adaptive GPS-based navigator with extended range code tracking. In: Proceedings of the IEEE Position Location and Navigation Symposium, San Diego, California.

Hanlon, P. D., Maybeck, P. S., April 2000. Multiple-model adaptive estimation using a residual correlation Kalman filter bank. IEEE Transactions on Aerospace and Electronic Systems 36.

Hartinger, H., Brunner, F. K., July 1998. Attainable accuracy of GPS measurements in engineering surveying. In: FIG XXI International Congress, Commission 6, Engineering Surveys.

Hide, C., Moore, T., Smith, M., 2002. Adaptive Kalman filtering for low cost GPS/INS. In: Proceedings of ION GPS 2002, The 15th Technical Meeting of the Satellite Division of the Institute of Navigation, Portland Oregon. pp. 1143–1147, (Best presentation award winner in Section C3a: MEMS Inertial Measuring Units).

Hide, C., Moore, T., Smith, M., January 2003. Adaptive Kalman filtering for low cost GPS/INS. Journal of navigation 56 (1), 143–152.

Hofmann-Wellenhof, B., Lichtenegger, H., Collins, J., 1997. GPS Theory and Practice. Springer-Verlag.

Howell, G., Tang, W., 1994. A universal GPS/INS Kalman filter design. In: Proceedings of ION GPS 1994, The 9th Technical Meeting of the Satellite Division of the Institute of Navigation, Salt Lake City, Utah. pp. 443–451.

Hu, C., Chen, Y., Chen, W., 2001. Adaptive Kalman filtering for DGPS positioning. In: Proceedings of the International Symposium on Kinematic Systems in Geodesy, Geomatics and Navigation (KIS), Banff, Canada.

IGS, 2002. International GPS service. [online]. Available at: <`http://igscb.jpl.nasa.gov/`> [Accessed 10 August 2002].

Kalman, R., March 1960. A new approach to linear filtering and prediction problems. Journal of Basic Engineering 82, 35–45.

Kaplan, E. D. (Ed.), 1996. Understanding GPS Principles and Applications. Artech House Publishers.

Kim, D., Langley, R. B., January 2001. Estimation of the stochastic model for long-basline kinematic GPS applications. In: ION National Technical Meeting. Long Beach, CA.

Kreye, C., Eissfellar, E., Winkil, J. Ó., September 2000. Improvements of GNSS receiver performance using deeply coupled INS measurements. In: Proceedings of ION GPS 2000, The 13th Technical Meeting of the Satellite Division of the Institute of Navigation, Salt Lake City, Utah.

Ladetto, Q., 2000. On foot navigation: continuous step calibration using both complementary recursive prediction and adaptive Kalman filtering. In: Proceedings of ION GPS 2000, The 13th Technical Meeting of the Satellite Division of the Institute of Navigation, Salt Lake City, Utah.

Langley, R. B., 1997. GPS receiver system noise. GPS World 8 (6), 40–45.

Lawrence, A., 1999. Modern Inertial Technology, 2nd Edition. Springer.

LCPC, 2003. Fiche signaltique d'un grand quipement. [online]. Available at: <`http://www.lcpc.fr/LCPC/Moyens.equipements/Equipements/Localisation/`> [Accessed 18 March 2003].

Leica Geosystems, 1999. GPS System 500 Technical Reference Manual. Version 2.0.

Leick, A., 1995. GPS Satellite Surveying, 2nd Edition. John Wiley and Sons Inc.

Lithopoulos, E., 1998. The applanix approach to GPS/INS integration. Photogrammetrie Fernerkundung Geoinformation , 363–370.

Lu, G., January 1995. Development of a GPS multi-antenna system for attitude determination. Ph.D. thesis, University of Calgary.

Magill, D., 1965. Optimal adaptive estimation of sampled stochastic processes. IEEE Transactions on automatic control AC-10, 434–439.

Maybeck, P., 1979. Stochastic models, estimation and control. Vol. 1. Academic Press.

Mehra, R., April 1970. On the identification of variances and adaptive Kalman filtering. IEEE Transactions on automatic control AC-15, 175–184.

Mehra, R., October 1972. Approaches to adaptive filtering. IEEE Transactions on automatic control AC-17, 693–698.

Meng, X., Roberts, G. W., Dodson, A. H., Cosser, E., Noakes, C., May 2003. Simulation of the effects of introducing pseudolite data into bridge deflection monitoring data. In: Second Symposium on Geodesy for Geotechnical and Structural Engineering. Berlin, Germany.

Mohamed, A. H., 1999. Optimizing the estimation procedure in INS/GPS integration for kinematic applications. Ph.D. thesis, University of Calgary.

Mohamed, A. H., Schwarz, K. P., 1999. Adaptive Kalman filtering for INS/GPS. Journal of Geodesy 73, 193–203.

Moore, T., Hill, C., Noakes, C., Hide, C., 2003. FURLONG: A study of future real-time location and navigation. In: Proceedings of ION GPS

2003, The 16th Technical Meeting of the Satellite Division of the Institute of Navigation, Portland Oregon.

Northrup Gruman., 2001a. LN100 product specification. [online]. Available at: <http://www.ngnavsys.com/Html/LN-100/> [Accessed 10 December 2001].

Northrup Gruman., 2001b. LN200 product specification. [online]. Available at: <http://www.ngnavsys.com/Html/LN-200/> [Accessed 10 December 2001].

Ochieng, W. Y., Sauer, K., Cross, P., Sheridan, K., Iliffe, J., Lannelongue, S., Ammour, N., Petit, K., 2001. Potential performance levels of a combined Galileo/GPS navigation system. The Journal of Navigation , 185–197.

Pattinson, M., 2002. Tropospheric delay estimation of a moving platform. Ph.D. thesis, University of Nottingham.

Rao, C., 1965. Linear stastical inference and its applications. John Wiley and Sons, Inc.

Rizos, C., 1999. GPS enhancements. Tech. rep., University of New South Wales, [online]. Available at: <http://www.gmat.unsw.edu.au/snap/gps/gps_notes2.pdf> [Accessed 2 April 2003].

Roberts, G., 1997. Real-time on the fly kinematic GPS. Ph.D. thesis, University of Nottingham.

Roberts, G., Meng, X., Dodson, A., April 2002. Using adaptive filtering to detect multipath and cycle slips in GPS/accelerometer bridge deflection

monitoring data. In: Proceedings of the XXII International Congress of the FIG, TS6.2 Engineering Surveys for Construction Works and Structural Engineering II, Washington DC, USA.

Rogers, R. M., 2000. Applied Mathematics in Integrated Navigation Systems. AIAA Education Series.

Salychev, O., 1998. Inertial Systems in Navigation and Geophysics. Bauman MSTU Press, Moscow.

Salychev, O., Voronov, V., Cannon, M., Nayak, R., Lachapelle, G., January 2000. Low-cost INS/GPS integration: Concepts and testing. Proceedings of the ION National Technical Meeting, Anaheim , 98–105.

Satirapod, C., 2002. Improving the accuracy of static GPS positioning with a new stochastic modelling procedure. In: Proceedings of ION GPS 2002, The 15th Technical Meeting of the Satellite Division of the Institute of Navigation, Portland Oregon.

Satirapod, C., Wang, J., 2000. Comparing the quality indicators of GPS carrier phase observations. Geomatics Research Australia 73, 75–92.

Schmidt, G., 1989. Kalman filter integration of modern guidance and navigation systems. AGARD Lecture Series (166).

Schwarz, K., 1999. University of Calgary lecture notes.

Schwarz, K., El-Sheimy, N., April 2000. Kingspad users manual. Tech. Rep. 3, Department of Geomatics Engineering, University of Calgary.

Schwieger, V., May 2001. Time-dependent correlations using the GPS

- a contribution to determine the uncertainty of GPS-measurements. Technical Conference during the FIG Working Week .

Shaw, M., Sandhoo, K., Turner, T., September 2000. Modernization of the global positioning system. GPS World , 36–44.

Shin, E.-H., December 2001. Accuracy improvement of low cost INS/GPS for land applications. Master's thesis, University of Calgary.

Škaloud, J., April 1995. Strapdown INS orientation accuracy with GPS aiding. Master's thesis, University of Calgary.

Škaloud, J., Cramer, M., Schwarz, K., july 1996. Exterior orientation by direct measurement of camera position and attitude. ISPRS Comission III, Working Group II 31.

Smith, M., Moore, T., Hill, C., Noakes, C., Hide, C., 2003. Simulation of GNSS/IMU measurements. In: Proceedings of the International Society for Photogrammetry and Remote Sensing, Theory, Tehcnology and Realities of Inertial/GPS Sensor Orientation, Castelldefels, Spain.

Sukkarieh, S., Nebot, E., Durrant-Whyte, H., June 1999. A high integrity IMU/GPS navigation loop for autonomous land vehicle applications.

Sun, J., May 1998. Development and testing of a real-time DGPS/INS integrated system. Master's thesis, University of Calgary.

Szarmes, M., Ryan, S., Lachapelle, G., Fenton, P., June 1997. DGPS high accuracy aircraft velocity determination using doppler measurements. In: Proceedings of the International Symposium on Kinematic Systems in Geodesy, Geomatics and Navigation (KIS), Banff, Canada. pp. 167–174.

Titterton, D. H., Weston, J. L., 1997. Strapdown Inertial Navigation Technology. Institution of Electrical Engineers.

Trethewey, M., Field, M., Cooper, D., 1999. Making the most of investment in multibeam sonar. International Conference on Shallow Water Survey Technologies, The Defence Science and Technology Organisation, Sydney Australia .

Tsakiri, M., 1995. GPS and DR for land vehicle navigation. Ph.D. thesis, University of Nottingham.

TSS (UK) Ltd, March 1999. POS/MV Model 320 Ver 3, Position and Orientation System for Marine Vessels System Manual.

Wang, J., 1999. Modelling and quality control for precise GPS and Glonass satellite positioning. Ph.D. thesis, School of Spatial Sciences, Curtin University of Technology.

Wang, J., Stewart, M., Tsakiri, M., 1999. Online stochastic modelling for INS/GPS integration. In: Proceedings of ION GPS 1999, The 12th Technical Meeting of the Satellite Division of the Institute of Navigation, Nashville, Tennessee.

Watts, B., Neads, S., July 2003. Robots behind the wheel. GPS World .

Weill, L. R., April 1997. Conquering multipath: the GPS accuracy battle. GPS World , 59–66.

Welch, G., 1999. Multiple models Kalman filter seminar. Tech. Rep. Comp 290-039.

Welch, G., Bishop, G., 2001. An introduction to the Kalman filter. Tech. rep., University of North Carolina at Chapel Hill, [online]. Available at: <`http://info.acm.org/pubs/toc/CRnotice.html`> [Accessed 29 November 2001].

Wieser, A., July 2001. Robust and fuzzy techniques for parameter estimation and quality assessment in GPS. Ph.D. thesis.

Wolf, R., Eissfeller, B., Hein, G. W., 1997. A Kalman filter for the integration of a low cost INS and an attitude GPS. In: Proceedings of the International Symposium on Kinematic Systems in Geodesy, Geomatics and Navigation (KIS), Banff, Canada.

Xin-Xiang, J., November 1996. Theory of carrier adjusted DGPS positioning approach and some experimental results. Ph.D. thesis, Delft University of Technology.

Xu, G., 2003. GPS - Theory, algorithms and applications. Springer Verlag.

Yazdi, N., Farrokh, A., Najafi, K., August 1998. Micromachined inertial sensors. In: Proceedings of the IEEE. Vol. 86. pp. 1640–1659.

Zhang, G., March 1995. A low-cost integrated INS/GPS system. Master's thesis, University of Calgary.

Appendix A

Cubic spline Calculation

Given the cubic equation,

$$y = a_i (x - x_i)^3 + b_i (x - x_i)^2 + c_i (x - x_i) + d_i \tag{A.1}$$

which has derivatives,

$$y' = 3a_i (x - x_i)^2 + 2b_i (x - x_i) + c_i \tag{A.2}$$

$$y'' = 6a_i (x - x_i) + 2b_i \tag{A.3}$$

the second derivatives, S_i, of the cubic equation can be calculated by solving the following system of equations,

$$\begin{pmatrix} & \ddots & & & & \\ h_{i-1} & 2(h_{i-1}+h_i) & h_i & & & \\ & h_i & 2(h_i+h_{i+1}) & h_{i+i} & & \\ & & h_{i+1} & 2(h_{i+1})+h_{i+2} & h_{i+2} & \\ & & & \ddots & & \end{pmatrix} \begin{pmatrix} \vdots \\ S_{i-1} \\ S_i \\ S_{i+1} \\ \vdots \end{pmatrix} \tag{A.4}$$

where $h_i = x_{i+1} - x_i$. The cubic equation in A.1 can then be constructed using the following equations to obtain the coefficients,

$$a_i = \frac{S_{i+1} - S_i}{6h_i} \tag{A.5}$$

$$b_i = \frac{S_i}{2} \tag{A.6}$$

$$c_i = \frac{y_{i+1} - y_i}{h_i} - \frac{2h_i S_i + h_i S_{i+1}}{6} \tag{A.7}$$

$$d_i = y_i \tag{A.8}$$

Appendix B

KinPos^i control file example

```
FILES
     INPUT    (File locations for GPS and IMU files)
          REFREC NOTT2 gerb nss/sunb/gerb118.not XYZ
               3851400.0 -78000.0 5067000.0 0.000
          KINREC NOTT2 movb nss/sunb/movb118.not XYZ
               3851531.9655 -78696.6063 5066401.6861 0.000
          EPH SP3 nss/sp3/igs10674.sp3
          IMU nss/sunb/insfile.dat
          ATTITUDE nss/attitude.out
     END
     OUTPUT
          POS sim.pos
          RES sim.res
          REP sim.rep
     END
END
OPTIONS
     MODE         4 (1:standalone,2:diffpse,3:diffcar,4:diffcar&pse)
     DOPPLER      YES
     SNR          YES    (Use signal to noise ratio GPS measurement model)
     SMOOTHPSE    NO
     FREQUENCY    1
     DPSEUDERR    1.4
     CARRERR      0.01
     DDOPERR      0.005
     INTERVAL     1.0
     START        GPS 1067 355000.0
     STOP         GPS 1067 360000.0
     ELEVMIN      10.0
     POSMODEL     CA    (Constant Acceleration Model)
END
INERTIAL
```

```
     INS          LOOSE    (Centralised or decentralised)
     TYPE         STD    (IMU file type)
     ARM          1.000 1.000 -1.000    (Lever arm separation)
     INITATTI     0.0 0.0 283.0    (Initial INS attitude)
     UPDATE       1    (Kalman filter update rate)
     INTERP       INS    (Interpolate IMU measurements at KF update)
     ATTERR       0.001 0.001    (INS process noise)
     VELERR       0.001 0.0014
     POSERR       0.0 0.0
     GYB          YES 1.0d-4    (Bias process noise)
     ACB          YES 1.0d-4
     GYS          YES 0.0    (Scale factor process noise)
     ACS          NO 0.0
     GYM          NO 0.0    (Misalignment process noise)
     ACM          NO 0.0
     IGYB         0.0 0.0 0.0    (Initial sensor biases)
     IACB         0.0 0.0 0.0
     IGYS         1.0 1.0 1.0
     IACS         1.0 1.0 1.0
     GPSCOV       YES    (Use GPS covariance to form measurement noise)
     GPSATT       0.1 0.1    (GPS attitude measurement noise)
     GPSPVEL      0.2 0.3    (Pseudorange velocity)
     GPSPPOS      0.1 0.15   (Pseudorange position)
     GPSCVEL      0.005 0.008   (Carrier phase velocity)
     GPSCPOS      0.01 0.013   (Carrier phase position)

     ATTITUDE     NO RPH    (Use GPS attitude measurements)
     KALTYPE      ADAPTIVE Q 20 INNOV DIAG    (Adaptive options)
END

ATMOSPERE    (GPS tropospheric delay models)
     BASE
          DRYMD MAGNET MAGNET
          WETMD NONE NEILL
          SOLVE NONE DIRECT
          METDAT STD
          DELAY NO tropdel.ini
     END
     ROVER
          DRYMD MAGNET MAGNET
          WETMD NONE NEILL
          SOLVE NONE DIRECT
          METDAT STD
     END
END
```

Appendix C

NSS control file example

```
FILES
     DATSTYLE sunb/***.01o    (Location and extension for GPS files)
     STATION
          NAME movb118 KINEMATIC    (Kinematic filename)
               ../traj/plymouth.dat    (Location of trajectory file)
          NAME gerb118 STATIC    (Static filename - more than one can be defined)
               4063506.2986 -297194.3999 4890803.0687    (Static coordinates)
     END
     EPH
          SP3 sp3/igs10674.sp3
     END
END
PROCOPT
     MODEL    (Errors to be simulated)
          IONO 1
          TROPO 1
          MULTIPATH 1
          DOPPLER 1
          SPLINE 1    (Use cubic spline interpolation)
     END
     PARAMETERS
          STARTTIME GPS 1067 355000.0
          STOPTIME GPS 1067 358000.0
          INTERVAL 1    (GPS sampling rate)
          SATCLKRANDOMERR 1.0d-12
          RECCLKRANDOMERR 4.0d-07
          SIMSATCLK EPH
          SIMRECCLK YES
          L1PSEUDO 0.75d0
          L2PSEUDO 0.75d0
          L1CARR 0.001d0
          L2CARR 0.001d0
```

```
          ELEVMIN 5.0d0
          SLIP NO
          UNHEALTHLY 1 2 3 4    (Satellites not to use)
     END
END
INERTIAL
     MODEL
          LEVERARM 1.0 1.0 -1.0
          GPSOFFSET 0.0 0.0 0.0    (GPS and INS attitude misalignment)
          NGYRO 3
          NACCEL 3
          INSDAT sunb/insfile.dat    (IMU output filename)
          INTERVAL 0.01
          AERONAV 0
          EARTHROT YES
          GRAVITY 0    (Gravity model)
     END
     GYROS
          AXI1 1.0 0.0 0.0    (Definition of IMU axes)
          AXI2 0.0 1.0 0.0
          AXI3 0.0 0.0 1.0
          BIAS 0.5    (°/s)
          BIASINST 1e-5 100
          NOISE 0.0142
          SF1 0.1    (%)
          SF2 0.0
          TEMP1 0.02    (°/s/ °C)
          TEMP2 0.0
          SATURATION 100
          QUANTI 0.011
          MISALIGMENT 0.0172
     END
     ACCELEROMETERS
          AXI1 1.0 0.0 0.0
          AXI2 0.0 1.0 0.0
          AXI3 0.0 0.0 1.0
          BIAS 0.294    (ms^-2)
          BIASINST 1e-5 60
          NOISE 0.0025
          SF1 0.1    (%)
          SF2 0.0
          TEMP1 0.02    (ms^-2/ °C)
          TEMP2 0.0
          SATURATION 40
          QUANTI 0.003
          MISALIGMENT 0.0172
          CROSS 0.1    (%)
     END
END
```