GEOMATICS
ENGINEERING

UCGE Reports

Number 20320

Department of Geomatics Engineering

Multiple Inertial Measurement Unit Integration for Pedestrian Navigation

(URL: http://www.geomatics.ucalgary.ca/graduatetheses)

by

Jared B. Bancroft

December 2010

SCHULICH
School of Engineering
UNIVERSITY OF
CALGARY

UNIVERSITY OF CALGARY

Multiple Inertial Measurement Unit Fusion for Pedestrian Navigation

by

Jared B. Bancroft

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE

DEGREE OF DOCTOR OF PHILOSOPHY

DEPARTMENT OF GEOMATICS ENGINEERING

CALGARY, ALBERTA

DECEMBER, 2010

Abstract

The Global Positioning System (GPS) is commonly used for pedestrian navigation. Unfortunately, GPS is often unable to provide the accuracy and availability in environments where pedestrian navigation is commonly required. One low cost inertial measurement unit (IMU) is often used to increase the accuracy and improve the availability of the navigation solution. This research develops several fusion methods of using multiple IMUs to enhance performance. In particular, this research seeks to understand the benefits and detriments of each fusion method.

Three fusion methods are proposed. First, all raw IMU measurements are mapped into a common frame (i.e. a virtual frame) and processed in a typical combined GPS-IMU Kalman filter. Second, a large stacked filter is constructed of several IMUs. This filter construction allows for relative information between the IMUs to be used as updates. Third, a federated filter is used to process each IMU as a local filter. The output of each local filter is shared with a master filter, which in turn, shares information back with the local filters. The construction of each filter is discussed and improvements are made to the virtual IMU (VIMU) architecture, which is the most commonly used architecture in the literature.

Since accuracy and availability are the most important characteristics of a pedestrian navigation system, the analysis of each filter‟s performance focuses on these two parameters. The various approaches are implemented in software, a hardware configuration is designed and actual data is collected in two environments, one where GPS signals are moderately attenuated and another where signals are severely attenuated, to a point where the standalone GPS solution is unusable. Accuracy is shown as a function of architecture and the number of IMUs used.

Results indicate that the stacked filter provides a linear increase in accuracy, while other architectures typically have less improvement with the addition of more than three IMUs. Areas where GPS is sufficient show little improvement with additional IMUs. Only the stacked filter decreases the minimal detectable blunder of GPS observations by a significant amount. Federated filters provide a comparable, but less accurate, solution to that of the stacked filter at a much lower computational cost.

Acknowledgements

I would first like to thank my Supervisor, Professor Gérard Lachapelle. Thank you for your support, advice, example and friendship. My training has advanced rapidly resulting from so many opportunities and projects in the PLAN Group. Your success has become my success and I thank you for it. I would also like to thank my co-supervisor, President Elizabeth Cannon, who never accepted anything less than my best.

I would like to thank Professor Mark Petovello, who was instrumental in helping me understand the numerous facets of navigation. You have always been generous in your time and have always entertained my questions.

Thank you to Tom Williams who was very influential during the theory and software development of the work included herein. Thank you also to Dr. Valérie Renaudin and Sid Kwakkel who were both very helpful in perfecting the text of this thesis.

I would like to thank my Dad, Professor John Bancroft, who inspired my continuation and advanced education. I also must thank my Mom, Fay Bancroft, who always believed in me and taught me to live a balanced life.

Finally, I would like to thank my wife Candace Bancroft. Your support to complete this degree was crucial. Thank you for all that you are and all that you have allowed me to become.

Dedication

I dedicate this thesis to my Dad.

Thank you for inspiring me and teaching me the true meaning of life through the statement below.

"No other success can compensate for failure in the home."

-David McKay

Table of Contents

List of Tables

List of Figures and Illustrations

List of Symbols and Abbreviations

Symbol	**Definition**
$\hat{\bullet}$	Estimated variable
$\tilde{\bullet}$	Observed variable
$\bullet^{a}$	Variable referenced to the „a‟ frame
$\bullet \quad t$	Variable as a function of time
$\bullet^{-}$	Predicted (prior to update)
$\bullet^{+}$	Updated quantity (post update)
$\dot{\bullet}$	Time derivative of variable
S_{L1}	Signal transmitted on L1
A_i, A_q	In-phase and quadrature amplitudes
$P \; Y \quad t$	Encrypted version of the Y code
$D \; t$	Navigation data message
$C \; t$	C/A PRN code
f_{L1}	GPS L1 carrier frequency
ρ	Geometric range
$\delta\rho$	Satellite position error
c	Speed of light
dt	Satellite clock error
dT	Receiver clock error
d_{ion}	Ionospheric error
d_{trop}	Tropospheric error

$\varepsilon_{P,m}$	Code multipath
ε_{P}	Other code errors
λ	Carrier phase wavelength
N	Integer ambiguity
$\varepsilon_{\phi,m}$	Carrier phase multipath
ε_{ϕ}	are the other carrier phase errors
$a_{f_0}, a_{f_1}, a_{f_2}$	Broadcast clock correction coefficients
t	Time
Δt_r	Correction due to relativistic effects
t_{gd}	Group delay
e	Satellite orbit eccentricity
a	Satellite orbit semi-major axis
E_k	Satellite orbit the eccentric anomaly
TEC	Total electron density
R_E	Approximation of the earth‟s radius
ζ	Zenith angle
h_I	Approximate mean height of the ionosphere
OF_{ζ_j}	Obliquity factor
P	Total pressure
T	Temperature
P_s	Signal power
N_o	Noise power with respect to a 1 Hz bandwidth
B	Receiver‟s processing bandwidth
β	Quaternion‟s rotation about the Euler axis
$\mathbf{e}$	Quaternion‟s unit vector of the Euler axis

α	A rotation about the X axis
β	A rotation about the Y axis
χ	A rotation about the Z axis
ω	Angular Velocity
ϕ	Longitude
λ	Latitude
ψ	Yaw
θ	Roll
φ	Pitch
$\mathbf{a}_c$	Coriolis force
$\mathbf{v}$	Velocity
$\mathbf{\Omega}$	Skew symmetric form of angular velocity
$\mathbf{f}$	Specific force vector
$\mathbf{b}$	Bias vector (accelerometer and gyro measurements)
S	Scale factor and cross coupling error matrix (accelerometer and gyro measurements)
$\boldsymbol{\eta}$	Noise
G_g	Specific force dependent bias matrix
σ	Standard deviation
γ_1	Skewness
γ_2	Kurtosis
H_o	Spectral density
$\mathbf{r}$	Position
R_a^b	Rotation matrix from the a frame to the b frame
$\boldsymbol{\gamma}^e$	Normal gravity vector
Q_a^b	Quaternion describing the rotation from the „a‟‟ to „b‟‟frame
$\mathbf{l}$	Observations vector
H	Design matrix

$\mathbf{x}$	State or unknown parameter vector
$\boldsymbol{\varepsilon}$	Measurement noise
J	Cost function
W	Weighting matrix
P	Covariance matrix of the estimated parameters
R	Covariance matrix of the observations
$\mathbf{r}$	Residuals
C_{r_k}	Residuals covariance matrix.
$F\ t$	Coefficient matrix describing the differential equations
$G\ t$	Coefficient matrix shaping the input noise
$w\ t$	White noise vector
Q	Process noise
$G\ \tau$	Shaping matrix
$Q_c\ \ \tau$	Continuous time spectral density
$\hat{C}_{\upsilon_k}$	Innovation based covariance matrix of the innovation sequence
β_i	Information constant shared between the master filter and the ith local filter.
M	Blunder mapping matrix and
∇	Vector of known blunders.
δ_0	Non-centrality parameter.
α	Type I error
β	Type II errors
n	The number of ensembles (e.g. IMUs).
$\dot{\mathbf{L}}$	Relative velocity between the two rigidly mounted points
$\mathbf{L}$	Vector between two points
χ^2	Chi-squared distribution
T_k	Test Statistic

Abbreviation	**Definition**
A/D	Analog Digital
AKF	Adaptive Kalman Filter
AS	Anti Spoofing
BPSK	Binary Phase Shift Keyed
C/N_o	Carrier to Noise Density
CD	Cumulative Distribution
CDMA	Code Division Multiple Access
DoD	Department of Defense
E911	Enhanced 911
ECEF	Earth Centered Earth Fixed
EIRP	Effective Isotropically Radiated Power
EKF	Extended Kalman filter
FDE	Fault Detection and Exclusion
FFR	Federated Fusion Reset
FNR	Federated No Reset
FZR	Federated Zero Reset
GDOP	Geometric Dilution of Precision
GLONASS	Global Navigation Satellite System
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
HDOP	Horizontal Dilution of Precision
HSGPS	High Sensitivity GPS
IB	Innovation Based
IMU	Inertial Measurement Unit
INS	inertial navigation system
IONEX	IONosphere map Exchange
IRNSS	Indian Regional Navigation Satellite System
JPO	Joint Program Office

LL	Local Level
LOS	Line of Sight
LSQ	Least-Squares
MDB	Minimum Detectable Blunder
MEMS	micro electro-mechanical systems
MEO	medium earth orbits
MM	Multiple Mode
NAVSTAR	Navigation System by Timing and Ranging
NLOS	Non-LOS
PDR	Pedestrian Dead Reckoning
PPS	Precise Positioning Service
PPS	Pulse Per Second
PV	Position and Velocity
PVA	position, velocity and attitude
PVT	position, velocity, and time
QZSS	Quasi-Zenith Satellite System
RAUPT	Relative Attitude Update
RF	Radio Frequency
RHCP	right hand circularly polarized
RIMU	Redundant IMU
RPUPT	Relative Position Update
RVUPT	Relative Velocity Update
SA	Selective Availability
SBAS	Satellite Based Augmentation System
SINS	Single Inertial Navigation System
SNR	Signal-to-Noise Ratio
SPS	Standard Positioning Service
SRIMU	Skew Redundant IMU
SVN	Satellite Vehicle Number
TCXO	Temperature Compensated Crystal Oscillator

TEC	Total Electron Content
TOA	Time of Arrival
UTC	Coordinated Universal Time
VIMU	Virtual Inertial Measurement Unit
ZUPT	Zero-Velocity Update

Chapter One: Introduction

The Global Positioning System (GPS) is a freely available satellite constellation that is maintained by the United States Department of Defense (DoD). Over the past few decades a multi-billion dollar industry has grown around the services made available through GPS. Declining hardware costs and increasingly user friendly interfaces continue to propel both commercial growth and the public‟s interest in satellite borne positioning. One sector that has experienced some of the fastest advancement of late has been pedestrian navigation, including cellular phone and first responder navigation. However, due to error sources that cannot be mitigated in real time, the navigation solution can extend beyond acceptable error limits or be completely unavailable. Thus, research to improve the accuracy and availability of a personal navigation system for pedestrians is warranted.

As GPS markets continue to expand and new applications are found every day, any new application must abide by a key requirement; namely, direct line-of-sight between the satellites and the receiver. So stringent is this requirement that the simple occlusion of satellites renders many navigation systems useless or at the least highly degraded. As users travel in urban canyons, parkades, indoors or in high foliage areas, the ability for GPS to provide a navigation solution is compromised. Although High Sensitivity GPS (HSGPS) receivers can track weak signals through fading, this renders them susceptible to high noise and multipath errors (Lachapelle 2007). Thus, researchers are examining other sensors to integrate with GPS.

Inertial measurement units (IMU) are a common complement to GPS, although it is technically more correct to state that GPS augments an inertial navigation system (INS). The advantage being that the GPS and inertial sensors can provide a continuous navigation solution, where GPS alone cannot. As competitive consumer markets drive the price of mobile navigation devices lower, an increasingly common choice for IMUs is micro electro-mechanical systems (MEMS). The size, cost, weight, and low power consumption make these an attractive grade of IMU; however their in-run biases, scale factors and high noise require the integration scheme to mitigate these errors (Titterton & Weston 2004).

While existing INS research has involved one IMU, the purpose of this research is to investigate the use of multiple IMUs in tandem with GPS. In particular, this research will investigate various approaches to integrate multiple IMUs with several filter architectures and constraints that can be used to further improve the accuracy and availability of the navigation solution, with emphasis on pedestrian navigation.

The objectives of this thesis are to:

1. Design, implement and test different techniques to utilize multiple IMUs and GPS observations for pedestrian navigation. Estimation architectures include:
    a. Virtual IMU observation fusion
    b. Centralized filter design
    c. Federated filter design
2. Assess fault detection capability on the IMU and GPS measurements, discussing any limitations.
3. Analyze and compare the performance of the different estimation architectures selected and the number of IMUs used.
4. Analyze the performance of each architecture in residential and indoor conditions.
5. Discuss the advantages and disadvantages of each architecture.

1.1 Pedestrian Navigation

Potential pedestrian navigation users include:

- first responders (e.g. emergency search and rescue)
- cellular phone users (E911 and navigation)
- health and activity monitoring

- recreational users (e.g. hikers, climbers, skiers)
- self guided tourists
- athletes and athletic trainers
- consensual tracking (e.g. elderly, parolees, employees)
- navigation for the visually impaired
- military forces

Each application requires a specific level of navigation accuracy which depends on various factors. Table 1 shows the requirements of some instances of pedestrian navigation and some commercial products that address them (marked *). In proposing a new technology, the proposed system should be equivalent or better in accuracy to those that it seeks to replace, but preferably be less expensive and have longer availability in GPS-compromised environments.

Table 1 – Various Accuracy Requirements and Realizations

Application	**Accuracy Requirement or Realization***
Cellular Phone Users	50 m (1σ) - GPS Enabled (E911, Phase II)
Emergency Land Applications	5-20 m (Hofmann-Wellenhof et al 2003)
Public Transport	20-50 m (Hofmann-Wellenhof et al 2003)
Athletic Training	0.01-50 m
Tourism	50-100 m (CEP) (Prost et al 2008)*
Military Ground Forces	< 10.5 m (2 σ) (Rockwell Collins - DAGR Technical Data Sheet)*
First Responders	< 3 m (ENSCO GEO/NAV Technical Data Sheet)* < 30 m in 30 min GPS outage (personal discussion with D. Taylor of ENSCO, Inc.)*

A key to the success of many INS pedestrian navigation applications is the placement of the IMU on a foot (e.g. Mezentsev (2005)) where the IMU experiences the repetitive and predictable motion of the human gait during walking. This allows for zero velocity updates while the foot in is contact with the ground, which plays a critical role in maintaining the long term accuracy of the system. There are two typical approaches for pedestrian navigation involving IMUs, (i) pedestrian dead reckoning (PDR), and (ii) a strap down INS (Groves et al 2007).

PDR uses the step length of a user and propagates the position forward using a known heading. Sensors used in PDR typically include GPS, IMUs, and magnetometers. This common approach is found in Gebre-Egziabher (2002), Mezentsev (2005), Stirling et al (2005), Beauregard (2007), Tan et al (2008), Zhao et al (2009) and Sun et al (2009). PDR has been successful when subjects walk normally, however it degrades quickly when users jog, sprint, shuffle, crawl (e.g. firefighters crawling inside a building), travel uphill or climb stairs. It is noteworthy that in PDR mode, the position error is typically a function of distance, rather than time (Mezentsev 2005).

Alternatively, an IMU is placed on the foot and treated as an INS. Examples for this method include: Brand & Phillips (2003), Lachapelle et al (2003), Foxlin (2005), Kasameyer et al (2005), Grejner-Brzezinska et al (2006), Mather et al (2006), Beauregard (2007), Groves et al (2007), Bancroft et al (2008) and Godha & Lachapelle (2008). This configuration reduces the necessity for magnetometers, although these can be used to aid with attitude determination as in Groves et al (2007). The INS method also allows for direct analysis of sport and biomedical applications such as gait kinematics and posture analysis (Kwakkel et al 2007, Renaudin et al 2007, Kwakkel 2008, Kwakkel et al 2008). However, a disadvantage to this approach is the time varying lever arm between the GPS antenna and IMU(s). To date, this error has been ignored and thus the magnitude of the lever arm‟s effect has not been quantified. Another limitation to the foot-mounted INS is the degraded accuracy over extended time periods. This result is common to all low cost INS setups and is primarily due to heading errors (Bancroft et al 2008).

Magnetometers are often integrated with the IMU to provide additional heading information. Often, a three axis magnetometer is mounted in conjunction with an IMU. In this manner the pitch and roll of the INS can be used to orient the magnetometer data in the local level frame. From this, the heading can be computed and used to update the system (Goldenberg 2007). The success of multisensor fusion technology has meant that manufacturers of MEMS IMUs are more commonly including magnetometers within the IMU triad (e.g. Analog Devices ADIS16405). Magnetometers however, are susceptible to magnetic interference from hard and soft iron effects. These biases can be self induced (from current inside the data collection apparatus), or externally induced from the materials in the environment. Mitigating such magnetic disturbances is very challenging considering the environments that pedestrian navigation includes (Goldenberg 2007).

Barometers are commonly used to mitigate temporal height variations. This improves the vertical channel accuracy, as long as the bias between the true height and current pressure is resolved. It is common for commercial pedestrian systems to include an IMU, magnetometer and barometer (Grejner-Brzezinska et al 2006, Mather et al 2006, Groves et al 2007).

Ultra wide band measurements in pedestrian navigation are receiving more attention due to their “multipath free” benefits in addition to non-line-of-sight requirements (Chui & O'Keefe 2008, Renaudin et al 2008, Chui & O'Keefe 2009). While this approach requires additional infrastructure, the potential use in first responders and other pedestrian applications is promising.

1.2 Multiple IMU Integration

GPS and IMUs have been successfully integrated since the formal introduction of GPS. More recently, attention has been placed on integration with MEMS IMUs to reduce cost, but still provide robust navigation solutions. A natural progression is to use more IMU sensors, and thus capitalize on the decreasing cost of MEMS sensors, in order to improve overall accuracy. As such, researchers commonly fuse multiple IMU measurements in the raw observation (i.e. specific force and angular velocity) domain, but have not pursued any other fusion methods. Thus, multi-IMU fusion can either occur in two categorical domains: the observation or estimation domain. The following sections explain these concepts in detail.

1.2.1 Raw IMU Observation Fusion

Numerous studies have taken an observation domain approach to redundant IMU (RIMU) integration whereby the observations of several IMUs are fused, generating a single virtual IMU measurement (Sturza 1988, Brown & Sturza 1990, Sukkarieh et al 2000, Allerton & Jia 2002, Colomina et al 2004, Giroux et al 2004, Pittelkau 2005, Osman et al 2006, Pittelkau 2006, Waegli et al 2008). The term virtual IMU (VIMU) will be used herein to describe fusion architectures in the observation domain. RIMU is more commonly used in the literature and can be confused with reduced IMU which has the same acronym.

In the development of VIMU theory, optimizing the configuration of the IMU sensor axes is an important consideration. Pejsa (1974) mathematically determined the optimal configuration for sensor axes; with sensors in a skewed formation rather than an orthogonal one (although the ideal 3 axis sensor is orthogonal). This work named this optimal setup the Skew Redundant IMU (SRIMU). Further work derived the GDOP (Geometric Dilution of Precision) for a multi-sensor cluster to provide theoretical estimations, incorporating correct weighting schemes and providing fault detection through statistical misclosure testing (Sturza 1988, Brown & Sturza 1990).

The prominent method of RIMU fusion fuses raw IMU observations using least squares estimation, mapping each IMU observation to a virtual IMU frame (which requires *a priori* knowledge of the transformation into the virtual fame). The estimation is described in Allerton & Jia (2002), Colomina et al (2004), and Waegli et al (2008). This methodology is fundamentally flawed in that the IMU observations contain un-modeled errors prior to fault testing thus negating fundamental rules of input/output covariance estimation. The work described in one paper (Waegli et al 2008) of the ten referred to in this section (1.2.1) report an actual improvement in performance (approximately 30% to 45%), rather than theoretical derivations of what could be done.

Figure 1 shows the VIMU observation fusion and integration with GPS.

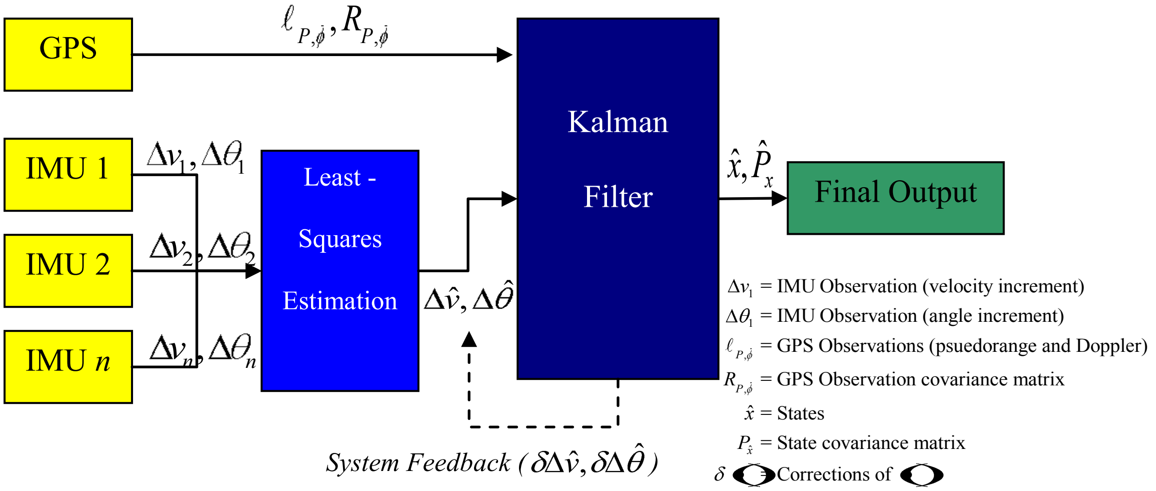

GPS
$\ell_{P,\dot{\phi}}, R_{P,\dot{\phi}}$
IMU 1
$\Delta v_1, \Delta\theta_1$
IMU 2
$\Delta v_2, \Delta\theta_2$
IMU n
$\Delta v_n, \Delta\theta_n$
Least - Squares Estimation
$\Delta\hat{v}, \Delta\hat{\theta}$
Kalman Filter
$\hat{x}, \hat{P}_x$
Final Output
System Feedback ($\delta\Delta\hat{v}, \delta\Delta\hat{\theta}$)
Δv_1 = IMU Observation (velocity increment)
$\Delta\theta_1$ = IMU Observation (angle increment)
$\ell_{P,\dot{\phi}}$ = GPS Observations (psuedorange and Doppler)
$R_{P,\dot{\phi}}$ = GPS Observation covariance matrix
$\hat{x}$ = States
$P_{\hat{x}}$ = State covariance matrix
δ = Corrections of

Figure 1 – IMU Observation Fusion Architecture

Often, the purpose of virtual IMU integration is not to improve the accuracy (although this is a desirable outcome), but to facilitate the detection and exclusion of faulty observations (Sturza 1988, Sukkarieh et al 2000). Of the papers listed in this section (1.2.1), only Waegli et al (2008) provides the actual statistical thresholds for fault detection as well as the quantity of the observations actually rejected. It therefore is prudent that more analysis be performed on the behaviour of the distribution of the raw observations since the true impact of faults in the data remains unexplored. It is unclear whether or not any faults even occur within the IMU data.

1.2.2 Architectural Filter Fusion

The fusion of GPS and a single IMU is typically implemented in a Kalman filter (Groves 2008). Other estimation techniques such as particle and sigma point filters are gaining popularity (e.g. Kubo & Wang (2008), Li et al (2008)), but go beyond the scope of this thesis.

Several architectures have been proposed for the integration of multisensor systems (Allerton & Jia 2005). While the literature is not consistent in terms of nomenclature, Figure 2 represents a selection of different architectures. For simplicity, the architectures discussed within this thesis are either centralized or decentralized, although Mutambara (1998) categorizes architectures into three categories. "Full" decentralization architectures provide random inter-communication between filters and do not provide one "final" solution making this filter unsuitable for navigation purposes and therefore will not be discussed further. Subsets of the decentralized architecture include the federated filter and sensor observation fusion. The bottom three architectures in Figure 2 represent the architectures that will be considered in this thesis; namely the stacked and federated filters as well as the VIMU.

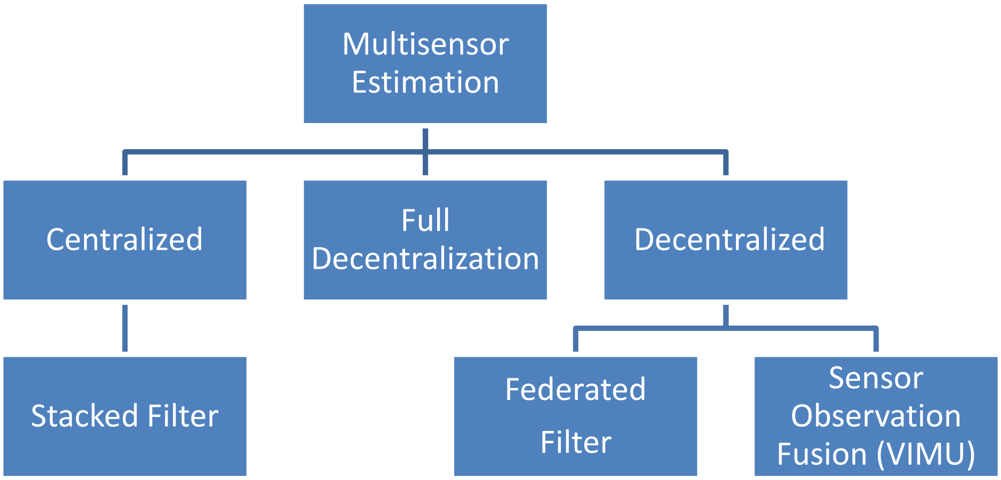


Figure 2 – Categorization of Multisensor Architectures

1.2.2.1 Centralized Filter Fusion

This multi-IMU approach uses a centralized filter that is composed of several individual block filters (e.g. Brand & Phillips (2003), Colomina et al (2004), Bancroft et al (2008), (Bancroft 2009)). The technique allows for the inclusion of relative geometry constraints, such as relative position, velocity and attitude between IMUs. The use of these constraints represent an advantage over the VIMU estimation techniques since VIMU architectures fail to utilize for this useful information.

The process of fault detection has not been investigated thoroughly for the centralized approach. The filter has shown promise with a 58 to 71% accuracy improvement in position, velocity and attitude relative to a truth solution (Bancroft et al 2008, Bancroft 2009). The centralization architecture is depicted in Figure 3.

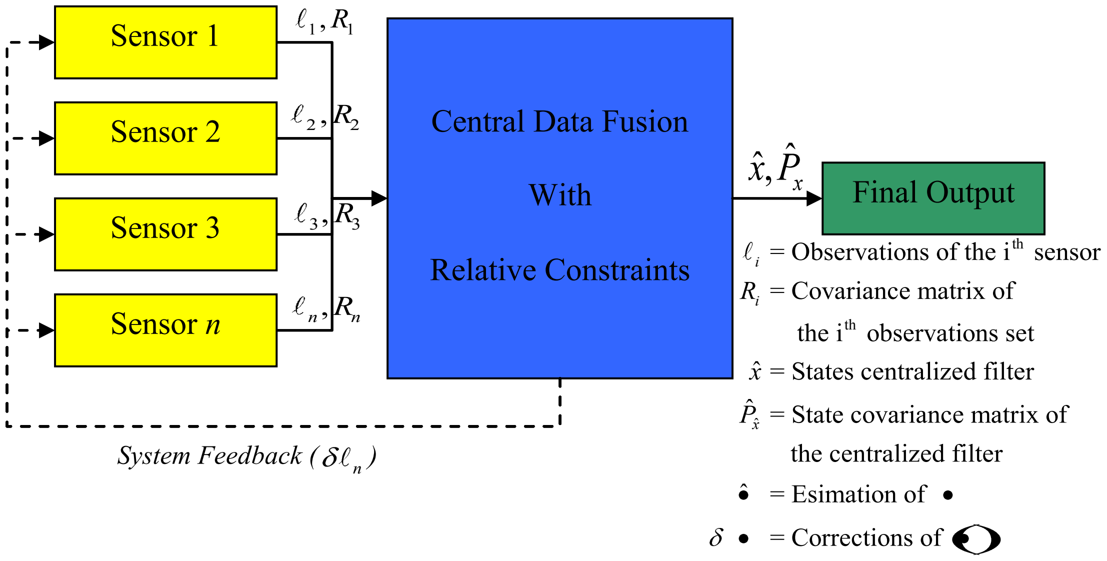

Sensor 1
Sensor 2
Sensor 3
Sensor n
ℓ_1, R_1
ℓ_2, R_2
ℓ_3, R_3
ℓ_n, R_n
Central Data Fusion
With
Relative Constraints
$\hat{x}, \hat{P}_x$
Final Output
ℓ_i = Observations of the i^{th} sensor
R_i = Covariance matrix of the i^{th} observations set
$\hat{x}$ = States centralized filter
$\hat{P}_{\hat{x}}$ = State covariance matrix of the centralized filter
$\hat{\bullet}$ = Esimation of $\bullet$
$\delta \bullet$ = Corrections of
System Feedback ($\delta\ell_n$)

Figure 3 – Centralized Data Fusion Architecture

Also not discussed thoroughly in the current literature is the numerical stability of such a large filter (noting that a five IMU filter can consists of 105 states).

1.2.2.2 Federated Filter Fusion

To the author‟s knowledge, there has been no published work in the domain of decentralized filters incorporating multiple IMUs. Federated filters were introduced in the late 1980‟s and early 90‟s for GPS and INS integration (e.g. Carlson (1990) and Wei & Schwarz (1990)), but have not been extended to the multi-IMU case. Federated filters utilizing several other navigation systems such as radar altimeters, terrain aided navigation systems and synthetic aperture radar have been discussed, but not restricted to IMUs (Carlson 2002, Allerton & Jia 2005).

Figure 4 shows a decentralized federated filter, where *n* individual "local" filters provide the input to a master filter (Mutambara 1998, Allerton & Jia 2005). Federated filters use an information sharing algorithm between the master and local filters (Gao et al 1993). Carlson and Berarducci (1994) discuss variations of the sharing principles, essentially tuning the configuration. The decentralized architecture, and like derivatives, are conducive to a multi-IMU scenario where each IMU represents its own local INS filter. An attractive by-product of the decentralized filter is a decrease in processing time, relative to its centralized counterpart (Gao et al 1993). Processing time is an important consideration in real-time applications where the use of multiple inertial units requires increased matrix computations and lengthy inversions.

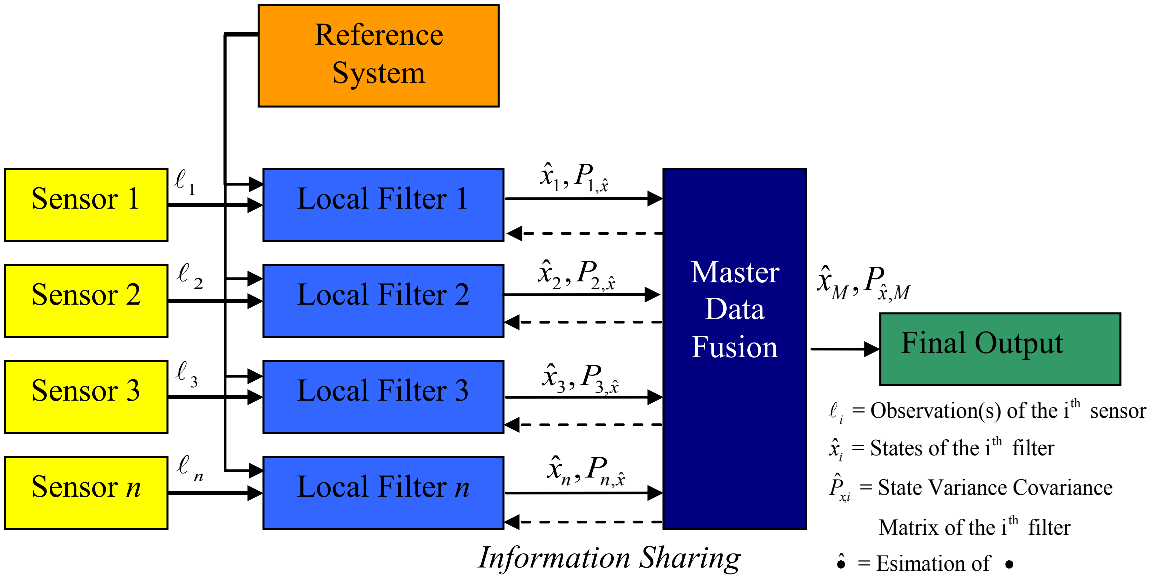

Reference System
Sensor 1
Sensor 2
Sensor 3
Sensor *n*
ℓ_1
ℓ_2
ℓ_3
ℓ_n
Local Filter 1
Local Filter 2
Local Filter 3
Local Filter *n*
$\hat{x}_1, P_{1,\hat{x}}$
$\hat{x}_2, P_{2,\hat{x}}$
$\hat{x}_3, P_{3,\hat{x}}$
$\hat{x}_n, P_{n,\hat{x}}$
Master Data Fusion
$\hat{x}_M, P_{\hat{x},M}$
Final Output
Information Sharing
ℓ_i = Observation(s) of the ith sensor
$\hat{x}_i$ = States of the ith filter
$\hat{P}_{xi}$ = State Variance Covariance Matrix of the ith filter
$\hat{\bullet}$ = Esimation of $\bullet$

Figure 4 – Federated Data Fusion Architecture

1.3 Thesis Overview

GNSS (Global Navigation Satellite System) aided INS is built on three key foundations: GNSS fundamentals, inertial navigation fundamentals, and the fusion of GPS and INS through an estimation process. The following sections provide the context for this thesis and the associated research. The review provided here is not intended to be comprehensive, but rather to provide background knowledge in order to facilitate the understanding of the integration schemes used within this thesis.

Chapter Two of this thesis provides the necessary background knowledge of GPS for pedestrian navigation users. GPS errors are discussed in detail to provide context to the errors as well as their mitigation techniques. Chapter Three provides background knowledge of inertial navigation, inertial measurement devices and errors contained within inertial measurement. Chapter Four provides the theoretical background for the estimation theory behind the filters proposed in Chapter Five. Chapter Four discusses the Kalman filter, its limitations and the input data requirements. Chapter Five discusses specific architectures used to fuse multiple IMUs for pedestrian navigation, namely the

VIMU, stacked filter and the federated filters. The chapter provides the construction and theory behind each proposed filter, in addition to several advantages and disadvantages for each architecture. Chapter Six provides results of two data sets in typical pedestrian environments. Chapter Seven concludes with several recommendations and conclusions.

Chapter Two: Satellite Based Navigation

2.1 GNSS Terminology

Global Navigation Satellite System (GNSS) is a generic term referring to worldwide radio-navigation satellite systems. The United States GPS was the first GNSS, declared operational in April 1995. The term was not used with respect to the U.S. Navy‟s Transit (GPS‟s predecessor), although arguably it was the first true GNSS, having started operating in 1964. Today, many GNSS systems are either operational, under replenishment, or under deployment. The Russian Global Navigation Satellite System (GLONASS) was developed simultaneously with GPS, operating with a full 24 satellite constellation in 1996, although the number of satellites has at times been reduced due to political and financial strife. In 1999, the European Union decided to proceed with its own GNSS, Galileo, although it will be managed by the public and private sectors rather than from within a military jurisdiction. Japan, under government and industry management, has developed Quasi-Zenith Satellite System (QZSS), a localized augmentation to GPS and future GNSSs. China has developed its own GNSS, called BeiDou (or Compass), which anticipates usage of a full 30 satellite constellation and satellite based augmentation operations. The Indian Regional Navigation Satellite System (IRNSS) is also under implementation with anticipation for both SBAS and global capabilities. Thus, it is clear that the term GNSS widely refers to available radionavigation satellite systems extending beyond only GPS.

2.2 The Global Positioning System

The Navigation System by Timing and Ranging (NAVSTAR) GNSS, now referred to simply as GPS, provides Radio Frequency (RF) ranging capabilities enabling military and civilian users to compute precise position, velocity, and time (PVT) through trilateration. The complete system was conceived during the 1960‟s after the success of Transit, and in 1969 the U.S. Office of the Secretary of Defense organized the Defense Navigation

Satellite System to consolidate the development of the system. The Joint Program Office (JPO) was instituted in 1973 to facilitate the construction, ground control management and receiver development for NAVSTAR, and currently still operates in that capacity today (Kaplan & Hegarty 2006).

GPS consists of three major segments: the space, control and user segments. The space segment manages the construction and launching of replacement satellites. The control segment monitors the health of the satellites and arranges satellite navigation data uploads. The user segment includes all military and civilian users and the manufactures that produce GPS receivers.

2.2.1 GPS Space Segment

The GPS satellite constellation is nominally at least 24 satellites set in 6 orbital planes, each orbital plane consisting of four or five satellites. Every orbital plane is earth centered, separated by 60 longitudinal degrees, and elevated 55 degrees from the equatorial plane (Lachapelle 2007). Each satellite orbit is slightly elliptical with eccentricities typically near 0.01, has an 11 hours 58 minutes period (one-half sidereal day) and are medium earth orbits (MEO), approximately 20,200 km from the mean surface of the earth. At the time of writing the GPS constellation consisted of 31 active satellites (United States Naval Observatory 2010). On 24 March 2009, Satellite Vehicle Number (SVN) 49, a GPS IIR-M satellite, was launched, beginning a new era of L5 signal transmission. SVN 49 was retrofitted with an L5 transmitter to fulfill requirements under the International Telecommunication Union Radiocommunication Sector and therefore avoided any foreign claim to the L5 frequency band.

2.2.2 GPS Control Segment

The control segment consists of several satellite tracking stations around the world including monitoring stations and master control stations, used to determine the position, velocity and time (PVT) of each satellite. It is the control segment"s responsibility to precisely determine the PVT of each satellite, upload ephemeris parameters and provide a predicted clock model of the satellite. These predicted ephemeris and time models are broadcast for use by the user segment.

Timing upon each satellite is maintained by an atomic clock. GPS Block I, II, and some IIA satellites used Caesium clocks, but all subsequent satellites contain stable Rubidium clocks.

The U.S. Naval Observatory maintains GPS time to be within 1μs of Coordinated Universal Time (UTC) modulo 1 integer second (IS-GPS-200E 2010), although GPS time is typically within 50 ns (Kaplan & Hegarty 2006). This time dissemination is a product of both the ground control stations, using International Atomic Time Standards, and GPS satellite measurements. GPS time and UTC time were set coincident at 0h January 6, 1980. GPS time is a continuous time scale and is not adjusted for leap seconds, which results in GPS time being biased by integer seconds (hence the modulo 1 integer second adjustment).

2.2.3 GPS User Segment

GPS (and in general GNSS) users use Time of Arrival (TOA) transmission observations, attempting to directly measure the travel time between the user and satellite. In order to measure the travel time, GPS uses three ultra high radio frequencies, namely: L1 (1575.42 MHz), L2 (1227.60 MHz), and L5 (1176.45 MHz). Pseudorandom noise (PRN) code sequences and the navigation message are modulated onto each satellites carrier, thus utilizing Code Division Multiple Access (CDMA) spread spectrum methods. Gold PRN codes were selected to provide minimal self and cross correlation between satellite PRN codes (Gold 1967).

Two positioning services are currently supported by GPS: Standard Positioning Service (SPS) and Precise Positioning Service (PPS). SPS is available to all users while PPS is restricted for the U.S. military and other authorized users. Currently, Anti Spoofing (AS) encrypts the Y code restricting use of the Y code to PPS users. The encrypted Y code is referred to as the P(Y) code and is modulated onto the L1 and L2 frequencies. Selective Availability (SA) was also used to segregate SPS and PPS users prior to 2000. SA degraded the accuracy of the measurements by introducing satellite clock dithering and degraded ephemeris parameters providing false satellite positions. SA reduced the accuracy of single point GPS receivers by up to 100 m. SA was officially turned to zero on May 2, 2000 by U.S. presidential order and the capability for SA on Block III satellites was later removed (GPS World Staff 2007). With SA off, the benefit of the PPS service over the SPS is: (1) access to the codes sequences on L2 providing dual frequency ionosphere corrections, (2) longer P(Y) code lengths and higher chipping rates providing higher range measurement precision and (3) lower multipath effects. With the addition of L5, SPS users will be able to achieve similar precision to those of PPS, although with less ability to mitigate interference and jamming.

2.3 GPS Signal Structure

Each GPS satellite transmits three components: the carrier, the ranging code and the navigation data message. The PRN code is modulated onto the carrier phase and the navigation data message is added via modulo-2, forming the Binary Phase Shift Keyed (BPSK) digital modulation. Mathematically, the transmitted signal on L1 is represented as

$$S_{L1}^{j} = A_i P \; Y^{j} \; t \oplus D \; t \; \cos \; 2 f_{L1} t + A_q C^{j} \; t \oplus D \; t \; \sin \; 2 f_{L1} t \qquad (2.1)$$

where:

S_{L1}^{j} is the signal transmitted on L1 of the jth satellite,

A_i, A_q are the in-phase and quadrature amplitude,

$P \; Y^{j} \; t$ is the encrypted version of the Y code, known as the P(Y) code of the jth satellite,

$D \; t$ is the navigation data message,

$C^j \quad t$ is the C/A PRN code of the jth satellite, and

f_{L1} is the L1 frequency (1575.42 MHz).

The $\oplus$ symbol refers to the modulo-2 addition and is commonly referred to as the XOR operator.

The code modulation yields a spread spectrum proportional to the chipping rate of the code. The main lobe of the spread spectrum for the C/A code is 2 MHz (1.023 MHz chipping rate), and 20 MHz for P(Y) code (10.23 chipping rate).

GPS signals are broadcasted right hand circularly polarized (RHCP). This counters the change in polarization, resultant from Faraday rotation that would otherwise occur on a linearly polarized signal. The RHCP signal can provide improved multipath rejection (i.e. rejecting LHCP signals) at the antenna level for high precision applications.

A satellites navigation data message consists of five 300 bit subframes. Each subframe consists of ten 30 bit words. Because the bit rate of the navigation message is 50 bps, it takes 750 seconds (12.5 min) to completely observe the navigation message, although subframes 1 to 3 repeat the same information. Subframe 1 contains the current GPS week number, satellite accuracy and health, clock correction terms and differential group delay information. Subframe 2 and 3 contain ephemeris parameters. Subframes 4 and 5 contain almanac data, special messages, satellite configuration flags, ionospheric model parameters, and UTC offset data. Subframes 4 and 5 contain 25 pages and are sequentially broadcasted (Lachapelle 2007).

The GPS signal structure is intricate and will not be further reviewed herein. Further information on the structure of the signal can be found in Misra & Enge (2001), Kaplan & Hegarty (2006), and IS-GPS-200E (2010).

2.4 GPS Observables

Three main observations are typically provided by GPS receivers: pseudorange, carrier phase, and Doppler frequency.

2.4.1 Code Pseudorange Observation

The apparent time shift between the receiver‟s replicated code and the code received at the antenna yields a function of the travel time between satellite transmission and antenna recipient (Kaplan & Hegarty 2006). The transmission is referenced to the satellites time and the reception is referenced to the receiver time, thus actually not measuring the time of travel between satellite and antenna, but the apparent time of travel. Multiplying this time difference by the speed of light yields a biased range; hence the range is called a pseudorange. The pseudorange observation equation of the j^th^ satellite is given as

$$\tilde{P}^j = \rho^j + \delta\rho^j + c\ \ dt^j - dT\ \ + d^j_{ion} + d^j_{trop} + \varepsilon^j_{P,m} + \varepsilon^j_P \qquad (2.2)$$

where:

ρ^j is the geometric range between satellite and receiver antenna [m],

$\delta\rho^j$ is the satellite position error (broadcast ephemeris) [m] ,

c is the speed of light [m/s],

dt^j is the satellite clock error with respect to GPS time [s],

dT is the receiver clock error with respect to GPS time [s],

d^j_{ion} is the ionospheric error [m],

d^j_{trop} is the tropospheric error [m],

$\varepsilon^j_{P,m}$ is the code multipath [m], and

ε^j_P are the other code errors (considered to be stochastic) [m].

2.4.2 Carrier Phase Observation

The carrier phase observation is the most precise range measurement a GPS receiver can provide. While the carrier phase observation still contains the same satellite based propagation errors as that of the pseudorange, the carrier phase typically provides much better precision in terms of noise and multipath. The cost of the better precision however comes at the expense of an unknown integer ambiguity term and is to be resolved by the processing software.

The ionospheric delay term for the carrier phase has the same magnitude, but opposite sign to the pseudorange. This effect is a result of carrier phase advancement versus code delay experienced by the signal during propagation through the atmosphere.

The phase observation equation of the jth satellite is given as

$$\tilde{\phi}^j = \rho^j + \delta\rho^j + c\ \ dt^j - dT\ \ - d^j_{ion} + d^j_{trop} + \lambda N^j + \varepsilon^j_{\phi,m} + \varepsilon^j_{\phi} \qquad (2.3)$$

where:

λ is the carrier phase wavelength [m] and

N^j is the integer ambiguity

$\varepsilon^j_{\phi,m}$ is the carrier phase multipath [m], and

ε^j_{ϕ} are the other carrier phase errors (considered to be stochastic) [m].

Due to the difficulty in resolving the ambiguity in single point mode (which most pedestrian navigation applications are) and the inability to maintain carrier phase lock in attenuated environments, the carrier phase observation is not used and therefore will not be further discussed.

2.4.3 Doppler Frequency Observation

The relative motion between the satellite and user antenna results in an apparent frequency shift called the Doppler frequency. The Doppler frequency is an instantaneous measurement made on the carrier phase tracking loop and since it is a time derivative of the phase observation it is impervious to the carrier phase ambiguity. The Doppler can be scaled by the wavelength to provide a relative velocity between the satellite and antenna. Since the satellites velocity can be precisely calculated, multiple Doppler observations can provide the absolute velocity determination of a receiver.

The Doppler observation equation of the jth satellite is given as

$$\tilde{\dot{\phi}}^j = \dot{\rho}^j + \delta\dot{\rho} + c\ d\dot{t}^j - d\dot{T} + \dot{d}^j_{ion} + \dot{d}^j_{trop} + \dot{\varepsilon}^j_{\dot{\phi},m} + \dot{\varepsilon}^j_{\dot{\phi}} \qquad (2.4)$$

where:

$\dot{\rho}^j$ is the geometric range rate between satellite and receiver antenna [m/s],

$\delta\dot{\rho}^j$ is the satellite velocity error (broadcast ephemeris) [m/s] ,

$d\dot{t}^j$ is the satellite clock drift error [s/s],

$d\dot{T}$ is the receiver clock drift error [s/s],

$\dot{d}^j_{ion}$ is the ionospheric error drift [m/s],

$\dot{d}^j_{trop}$ is the tropospheric error drift [m/s],

$\dot{\varepsilon}^j_{\dot{\phi},m}$ is the Doppler multipath rate of change [m/s], and

$\dot{\varepsilon}^j_{\dot{\phi}}$ are the other Doppler errors (considered to be stochastic) [m/s].

2.5 GPS Error Sources

The errors in observation Equations (2.2), (2.3), and (2.4) can be categorized into three sources: satellite based, propagation, and receiver based. Satellite based errors include the satellite clock and ephemeris errors (i.e. satellite position and velocity errors). Propagation errors include the effects resultant from the ionosphere, troposphere, surrounding environment, and interference. Receiver-based errors are a function of the receiver clock, antenna type, noise characteristics and inter-channel biases. All GPS error sources discussed within this thesis are considered either stochastic or systematic. In some cases, systematic errors will be represented as stochastic processes, thus creating the inability to correct, solve or remove the error, but enabling some form of error mitigation.

2.5.1 Satellite Based Errors

Satellite based errors include the broadcast ephemeris errors and the satellite clock error. These errors are ultimately controlled by the GPS control segment. The satellite ephemeris parameters and clock are reverse computed using known control stations around the world. The control segment then predicts ephemeris parameters and clock corrections, uploading them to the satellite approximately every two hours. The longer the delay between updates the more error present in the broadcast parameters. In post mission applications, these parameters can be estimated with very high precision, reducing the final GPS satellite orbital error to 5 cm and satellite clock time to 0.5 ns (i.e. ~15 cm) (Kaplan & Hegarty 2006).

2.5.1.1 Broadcast Ephemeris Errors

GPS satellites serve as control points used in determining a user‟s PVT, and thus any error in the control points are observed in the observations. As part of the satellite navigation message, subframes 2 and 3 contain the necessary parameters to compute the satellite position and velocity. These broadcasted parameters contain a residual error, introducing an error into the PVT estimation of the satellite. A total of 16 ephemeris parameters are sent by the satellite, six of which represent the fundamental Keplarian elements describing an orbit. The remaining parameters provide the perturbations of the Keplarian orbit, which account for the gravitational effects of the earth, moon and sun, solar radiation pressure and varying gravitation effects around the earth (Misra & Enge 2001).

The satellite trajectory can be defined in terms of a space vehicle body frame, consisting of the three orthogonal axes, pointing along track, across track and radially from the user. The magnitude of the ephemeris error is dominated by the along and across track errors of the satellite (resulting from weak geometry in estimating the PVT of a satellite from multiple stations on earth). It is common for satellites to have a 1 to 6 m residual position error (3D); however the effective range error is only typically 0.8 m (1σ)(Kaplan & Hegarty 2006). The broadcast satellite velocity error is around 0.6 mm/s (1σ) (Olynik 2002).

2.5.1.2 Satellite Clock Errors

Aboard all GPS satellites are atomic clocks which maintain the onboard timing operations, signal generation and broadcasting. While the clocks are extremely stable, absolute timing of the satellite clock drifts with respect to GPS time. The control segment approximates the clock errors using a second order polynomial fit and uploads the coefficients of the polynomial to the satellite to include in the broadcasted navigation message. The satellite clock correction is represented as

$$dt^j = a_{f_0} + a_{f_1} \quad t - t_{oc} \quad + a_{f_2} \quad t - t_{oc} \quad ^2 + \Delta t_r + t_{gd} \tag{2.5}$$

where:

$a_{f_0}, a_{f_1}, a_{f_2}$ are the broadcast clock correction coefficients [s, s/s, s/s^2],

t is current epoch time [s],

t_{oc} is the referenced epoch [s],

Δt_r is the correction due to relativistic effects [s], and

t_{gd} is the group delay [s].

The clock correction parameters are predicted by the control segment and therefore still contain residual errors. The magnitudes of the residual errors are typically 0.3 to 4 m, and vary as a function of satellite clock type and time since the last control segment upload (Kaplan & Hegarty 2006). The magnitude of the residual clock drift typically resides around 0.084 mm/s (1σ) for Block II, 0.070 mm/s (1σ) for Block IIA, and 0.052 mm/s (1σ) for Block II R (Olynik 2002).

Since the satellite experiences high velocities and travels through varying gravitational potentials the clock corrections also include a relativistic correction. When the satellites are at perigee (i.e. closest to earth) the velocity reaches a maximum and the gravitational potential reaches a minimum (and vice versa). This change in speed and gravitational potential causes the satellite clock to run more slowly and when the satellite is at apogee the clock will run more quickly relative to earth. This effect, shown mathematically below, can have a maximum magnitude of 21 m (70 ns) and is computed as (IS-GPS-200E 2010)

$$\Delta t_r = Fe\sqrt{a}\sin E_k \tag{2.6}$$

where:

F is the a predetermined constant, -4.442807633 e-10 [s/m$^{1/2}$],

e is the satellite orbit eccentricity,

a is the satellite orbit semi-major axis [m], and

E_k is the satellite orbit the eccentric anomaly.

The average time it takes the signal to travel from the satellite to earth can reach 75 ms. During this time, the earth rotates resulting in a relativistic error, resulting in a small discrepancy between the broadcast and received time. The error is commonly referred to as the Sagnac effect and the correction is simply referred to as the earth rotation correction.

Prior to launch, the satellite clock frequency is adjusted from 10.23 MHz to 10.22999999543 MHz so that GPS users do not have to correct for this portion of relativity theory.

2.5.2 Propagation Based Errors

Electro-magnetic waves experience effects resulting from the medium they travel in, thus GPS signals experience errors resulting from the atmosphere and surrounding environment. The medium and local environments are categorized as propagation-based errors.

2.5.2.1 The Ionosphere and Ionospheric Errors

The ionosphere is a portion of the atmosphere extending 50 to 1,000 km above the surface of the earth. For GNSS L band EM waves, this portion of the atmosphere is dispersive, meaning that different frequencies propagate differently. The dispersive nature of the ionosphere is a result of the free electrons present in it. Free electrons are created by the ionization of gas molecules by ultraviolet cosmic rays. The ionospheric effect on group velocity and phase velocity are equal, but of opposite sign (see Equations (2.2) and (2.3)). The magnitude of the delay is proportional to the electron density and inversely proportional to the square of the frequency. Total Electron Content (TEC) is a value used to represent the density and refers to the number of free electrons in a 1 m^2 cylinder cross section extending from satellite to antenna. Mathematically, the ionospheric error is represented as

$$d_{ion} = \frac{-40.3 \cdot TEC}{f^2} \tag{2.7}$$

where

TEC is the total electron density [electrons/m^2] and

f is the carrier wave frequency (e.g. L1) [Hz]

The TEC is diurnal, but is also a function of antenna location, season, satellite elevation angle, ionizing flux, magnetic activity, sun spot cycle and scintillation (Kaplan & Hegarty 2006).

The signal path of each satellite-to-user observation will have a different ionospheric delays and therefore has to be corrected. A common approach to determine this error is to define the user‟s zenith ionospheric delay (d_{ion}^{z}) and multiply it by an obliquity factor. Figure 5 shows a cross sectional diagram of the situation.

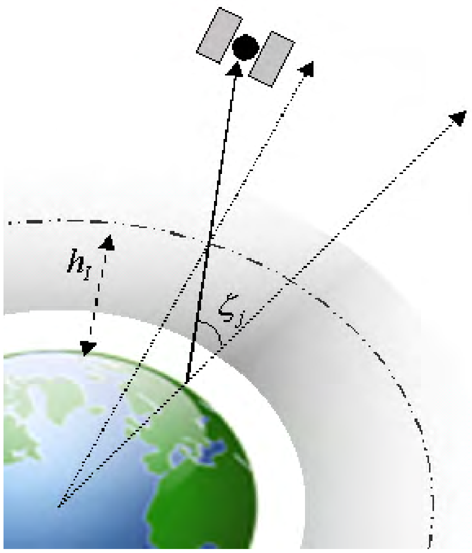


Figure 5 – Obliquity Factor of the zenith Angle

Mathematically, the process can be modeled as

$$d_{ion}^{j} = d_{ion}^{z} \underbrace{\left[1-\left(\frac{R_E \sin \ \zeta_j}{R_E + h_I}\right)^{2}\right]^{-\frac{1}{2}}}_{OF_{\zeta_j}} \tag{2.8}$$

where:

R_E is an approximation of the earth‟s radius [km],

ζ_j is the zenith angle of the jth satellite [°],

h_I is the approximate mean height of the ionosphere (nominally 350 km) [km], and

OF_{ζ_j} is the obliquity factor of the jth satellite.

Mitigating the ionospheric error is done in a variety of ways. Satellite navigation users capable of making multiple frequency observations can remove the first order effect of the ionospheric error. The first order effects account for 99% of the ionosphere error magnitude (Lachapelle 2007). Users capable of differential GPS can determine and mitigate the ionosphere error, although the accuracy is a function of the distance between receivers. Unfortunately, the use of multiple receivers and multiple frequencies adds cost to hardware and is thus typically too expensive for pedestrian navigation applications.

The Klobuchar model (also known as the broadcast model) approximates the zenith ionospheric error with a constant value during the night and a half cosine function during the day. The model is a function of user time as follows

$$d_{ion}^{z} = \begin{cases} cA_1, \text{ if } |t - A_3| > \frac{A_4}{4}, \text{otherwise} \\ c\left(A_1 + A_2 \cos\left(\frac{2\pi(t - A_3)}{A_4}\right)\right) \end{cases} \tag{2.9}$$

where:

A_1 is the constant bias (5e-9 s) [s]

A_2 is the amplitude of the cosine function [s]

A_3 is the corresponding peak of the cosine function (50,400 s or 1400 h local time) [s], and

A_4 is the period of the cosine function [s].

Parameters A_2 and A_4 are determined by the control segment and broadcast in the fourth subframe of the navigation message. It has been estimated that the model only accounts for 50 % of the true ionospheric error (Misra & Enge 2001).

Other methods of mitigating the ionosphere error include using 2D and 3D wide area models, grid and spherical harmonic global ionospheric models, such as IONosphere map

Exchange (IONEX) and Bernese ION format, Voxel (box) approach and radio occultation from low earth orbiting satellites.

The zenith ionosphere error typically ranges between 3 and 45 m (Kaplan & Hegarty 2006), although in times of high ionospheric activity can be much greater. The rate of change in ionospheric error is on the order of 1.4 mm/s (1σ) (Olynik 2002), but is a function of the level of ionospheric activity.

2.5.2.2 Tropospheric Errors

The troposphere extends from the surface of the earth to approximately 50 km and consists of mostly nitrogen, oxygen gas and water vapour. About 75 % of all the dry gas atmosphere of the earth is contained within the troposphere. The majority of water vapour is within 4 km of mean sea level and all of the water vapour is within 12 km. The troposphere is non dispersive and its refractive index is typically 0.03 % of unity. The troposphere error consists of two categorized sources: the dry and wet components. Since the dry and wet components behave very differently they are modeled differently, but ultimately the sum of the dry and wet troposphere errors equals the total tropospheric error.

Errors of the dry and wet components are computed by integrating the refractivity over the entire path length. Thus, determining refractivity throughout the troposphere can provide estimates of the troposphere error. Refractivity is a function of temperature, pressure and water vapour. Mathematically the dry component refractivity can be computed as

$$N_{dry} = a_1 \frac{P}{T} \tag{2.10}$$

where:

a_1 is an empirically determined constant (~77.624 K/mbar)

P is the total pressure [mbar], and

T is the temperature [K].

Moreover, refractivity for the wet component can be computed as

$$N_{wet} = a_2 \frac{e}{T} + a_3 \frac{e}{T^2} \tag{2.11}$$

where

a_2 and a_3 are empirically determined constants (approximately -12.92 K/mbar and $3.719 \cdot 10^5$ K^2/mbar, respectively), and

e is the partial pressure of water vapour [mbar].

The troposphere error is then represented as

$$d_{trop} = 10^{-6} \int N_{dry} dl + 10^{-6} \int N_{wet} dl \, . \tag{2.12}$$

Unfortunately, determining temperature, pressure and humidity along the entire travel path is unrealistic in most geomatics applications, including pedestrian navigation. Thus, several models to approximate the troposphere error have been developed. Most, at present, provide the zenith troposphere delay as a function of temperature, pressure, humidity and latitude, and require an additional obliquity factor to map the error to the satellite zenith. The obliquity factor of the ionosphere and troposphere errors are typically not the same.

Numerous troposphere models have been developed including the Saastamoinen, Hopfield, modified Hopfield and Black and Eisner, which assume different atmospheric characteristics and models. The magnitude of the zenith troposphere error is approximately 2.4 m, and typically the accuracy of these models is about 80 to 90 %, leaving residuals on the order of 25 to 50 cm (Misra & Enge 2001). The rate of change in the residual error is approximately 1 mm/s at the zenith and 2 mm/s at the horizon (Olynik 2002).

2.5.2.3 Multipath Errors

Multipath errors, or simply multipath, are created when antennas receive non line of sight signals which interfere with the transmitted line of sight signal. In some cases, such as indoors and in urban canyons, there may not be line of sight signals received by the antenna.

Multipath is categorized into two categories: specular and diffuse (Lachapelle 2007). Specular multipath is created when the transmitted signal is reflected in a particular direction and follows Snell"s law. Diffuse multipath is created when the signal is reflected in multiple ways. Materials that are smooth (e.g. glass buildings and cars) typically produce specular multipath and normally contain equivalent signal strengths, while rough materials (e.g. trees or ocean surfaces) produce diffuse multipath with reduced power levels.

Multipath is a systematic error that is very difficult to quantify in real time and thus difficult to correct. However, there are a few characteristics which are important to consider (Kuusniemi 2005):

1. Multipath decorrelates both temporally and spatially very rapidly.
2. Line of Sight (LOS) signals are always RHCP and all LHCP are multipath signals.
3. Multipath is observable and repeatable and is a function of the satellite and user position and surrounding environment.
4. Multipath signals can have seemingly normal power levels, but generally are lower in power.
5. Multipath is unbounded for Non-LOS (NLOS) signals.

There is very little that can be done to mitigate multipath at the pseudorange level. Most mitigation is performed internally in the receiver and antenna. HSGPS receivers are particular susceptible to multipath, but have the advantage of providing users with more observation availability.

Multipath signals always arrive after the LOS signal; however the change in the phase of the reflected signal can result in estimated ranges that are less than the true distance (ignoring other ranging errors). Multipath on the pseudorange is limited to one half chip length (~150 m on L1 C/A code), provided LOS signals are present. However, in some environments such as indoors, only NLOS signals are available and therefore multipath can theoretically be unbounded.

Multipath is mitigated by appropriate site selection, receiver design, antenna selection and use of an antenna ground place (choke ring). Unfortunately, none of these attributes are conducive to pedestrian navigation, thus making multipath one of the largest error sources in pedestrian navigation applications.

2.5.3 Receiver Based Range Errors

2.5.3.1 Receiver Clock Errors

Inherent to all GNSS receivers is a time bias between GPS time and the receiver‟s internal time. Since the receiver estimates the range based on the transmit time (referenced to GPS time) and the received time (referenced to the receiver‟s time), all measured ranges will contain a bias known as the receiver clock error. The error is estimated within the estimation process as a fourth parameter (in addition to three position parameters). This pattern also follows with the receiver‟s clock drift, which is also estimated as part of the navigation solution if Doppler observations are used.

Typically a GNSS receiver will incorporate a relatively low cost quartz clock, although some high end geodetic receivers will use a temperature compensated crystal oscillator (TCXO). These high-end clocks are typically more expensive, require more power and are physically larger thus limiting their applications.

The receiver clock error is a systematic quantity. The clock error can also be predicted from apriori information contained in the clock drift, thus in a filter, the receiver clock drift is typically estimated as a random walk or Gauss-Markov process.

2.5.3.2 Noise

Random noise is inherent in all GNSS observations, albeit it may have different magnitudes. Noise defined herein is a stochastic error. Noise results from amplifiers, antenna, cables, inter-channel biases, thermal noise jitter (receiver clock), dynamic stress on the oscillator and simple resolution of the observed code. The L1 C/A code has a noise standard deviation of about 0.8 m (1σ), although this value varies between receiver types. A 6 cm/s (1σ) noise standard deviation on the Doppler observation is commonly used. Noise error magnitude is also fundamentally based on the strength of the received signal power.

Noise can be determined through a zero baseline test as discussed in Lachapelle (2007). To provide context, a 10 hour zero baseline test was conducted in open skies and in a residential house. In the latter case the signal attenuation reached 17 dB, but was commonly about 10 dB. Since zero baselines remove all systematic errors, the residual error is stochastic. It is commonly assumed that these errors are normally distributed, however as shown in Figure 6 and Figure 7 the errors are not completely Gaussian. For the purposes of this thesis, the pseudorange and Doppler observations are however assumed to be Gaussian.

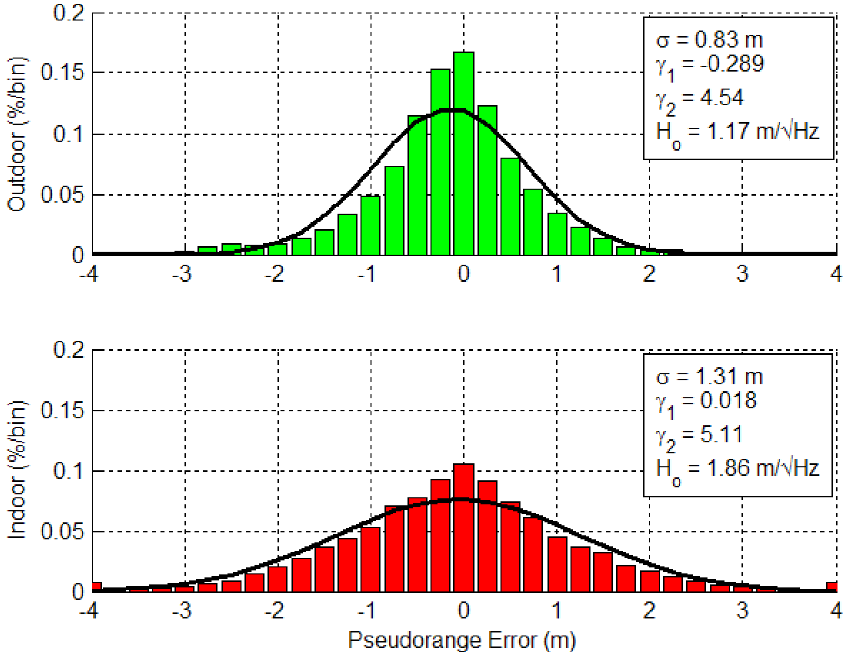


Figure 6 – Zero Baseline Pseudorange Errors (u-blox 10 Hours @ 1 Hz)[1]

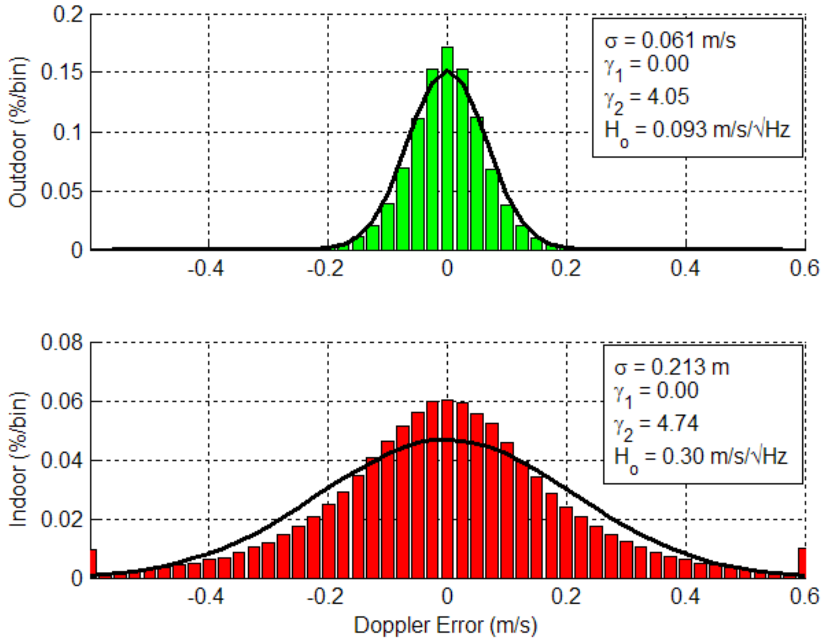


Figure 7 – Zero Baseline Doppler Errors (u-blox 10 Hours @ 1 Hz)[1]

[1] Note - σ is the standard deviation and shown in the figure with a black line, γ_1 is the skewness of the data, γ_2 is the Kurtosis of the data and H_o is the spectral density of the data.

2.5.4 Received Signal Power

Carrier to Noise density (C/N_o) is a measure of the relative signal strength compared to the surrounding noise environment. This metric provides a concise measurement of the integrity of the signal received at the antenna. C/N_o (in units of db-Hz) is computed by

$$C/N_o = \left[\frac{P_s}{N_o} \right]_{dB} \tag{2.13}$$

where

P_s is the signal power [W] and

N_o is the noise power with respect to a 1 Hz bandwidth [W/Hz].

Typical C/N_o values for direct LOS signals are greater than 35 dB-Hz. Signals between 28 and 35 dB-Hz are marginal and any signal less than 28 dB-Hz is poor. Signals less than 28 dB-Hz are likely reflected signals or signals with severe attenuation and usually occur indoors.

The Signal-to-Noise Ratio (SNR) alternatively provides an indication on the receiver performance given the observed signal. This metric accounts for the receivers processing bandwidth, nominally 2 MHz, and is computed by

$$SNR = \left[\frac{P_s}{BN_o} \right]_{dB} \tag{2.14}$$

where

B is the receiver"s processing bandwidth [Hz].

The signal power is weakened by several factors including free space loss, ionospheric and tropospheric loss, depolarization loss, and antenna location. Power gains include the SV Effective Isotropically Radiated Power (EIRP). These power levels typically sum to a nominal value of -160 dB and may vary depending on latitude, satellite elevation angle, atmospheric conditions and local environment. Figure 8 graphically shows the magnitude of each power loss and gain, free space loss accounting for 97 % of power loss (Lachapelle 2007).

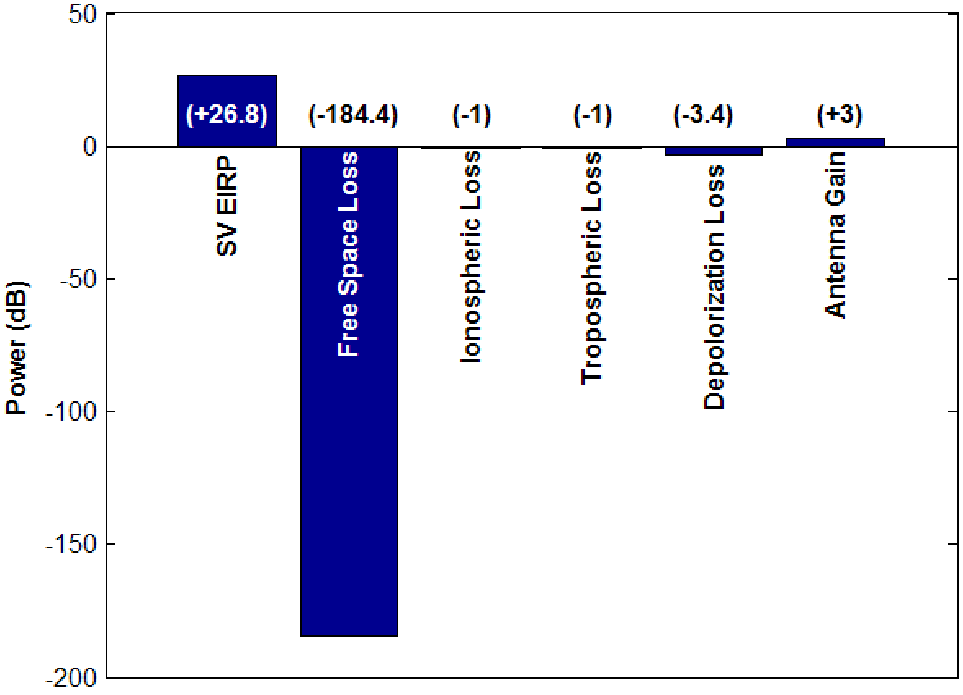


Figure 8 – Power Losses and Gains

Goldhirsh & Vogel (1998) indicate that tree canopies can reduce power levels by 11 dB with approximately 5 dB RMS variation at 1.6 GHz (L1). The study also indicates that indoors the L1 signal power can be reduced by up to 25 dB and that the human body shielding can reduce power levels between 6 to 10 dB.

Figure 9 shows the cumulative probability distribution of the 10 hour data set presented in Section 2.5.4. This distribution shows that indoors, the availability of untainted signals is reduced dramatically. This particular data set shows that 41 % of the C/N$_o$ are lower than what would be observed in open sky conditions.

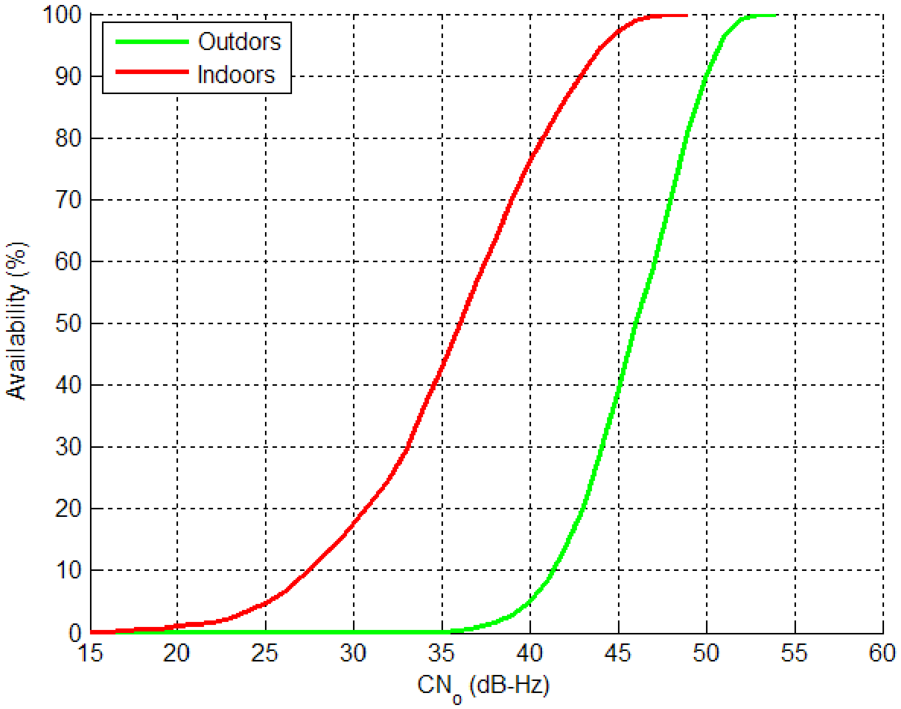


Figure 9 – Cumulative Distribution of C/N$_o$ (u-blox 10 Hours @ 1 Hz)

Chapter Three: Inertial Navigation

Inertial navigation derives relative navigation parameters from a self contained autonomous sensor system (i.e. an IMU). An IMU contains an accelerometer triad, that measures the specific force exerted on the IMU, as well as a gyroscope triad, that measures the angular velocity of the IMU. Once earth rotation and gravity effects are removed from the IMU observations, the integration of the angular velocity over a finite period yields the rotation over the interval and the double integration of the specific force yields the change in position (e.g. Farrell & Barth (1998)).

The measurement errors in IMUs are a chief concern, since they accumulate with successive integrations and vary with time, making them hard to quantify over a single interval. In order to estimate the IMU errors, measurements are checked against other navigation information. The fusion of GPS and IMU measurements is attractive since satellite measurements do not accumulate like those of inertial units. GPS therefore provides absolute positions (and velocity) to the system, in addition to providing essential information to estimate IMU errors. Other updates to the INS can provide enhanced IMU error estimation including zero velocity updates (Godha & Lachapelle 2008), speed updates (i.e. observations from wheel speed sensors) (Gao 2007), coordinate updates and non-holonomic constraints (Godha 2006).

3.1 Coordinate Frames

Four major coordinate frames are often used in inertial navigation. Since each plays a critical role in the INS, a brief description of these frames is necessary. The four coordinate frames discussed are: the inertial frame (*i* frame), Earth Centered Earth Fixed (ECEF), local level (LL) frame and the body frame (*b* frame). The inertial frame, which is a non-rotating, non-accelerating frame, is fixed with respect to celestial objects (e.g.

stars). Table 2 shows the definition (and convention) of the frames used within this thesis.

Table 2 – Coordinate Frame Convention

Frame	Origin	X Axis	Y Axis	Z Axis
Inertial (*i*)	Earth‟s Centre of Mass	Toward Vernal Equinox	Orthogonal to X and Z Forming a Right Handed Frame	Earth Rotation Axis (mean)
Earth Centered Earth Fixed (ECEF) (*e*)	Earth‟s Center of Mass	Mean Meridian of Greenwich	Orthogonal to X and Z axis Forming a Right Handed Frame	Earth Rotation Axis (mean)
Local Level (LL) Frame (*l*)	IMU Vertex	Geodetic East	Geodetic North	Orthogonal to the Reference Ellipsoid, Upward
Body Frame (*b*)	IMU Vertex	Across Track of IMU	Along Track of IMU	Orthogonal to X and Y Axis Forming a Tight Handed System

3.2 Attitude Representation

Transformations between coordinate frames are very common in an INS. IMU observations are made in the body frame, the user typically desires output navigation information in the ECEF or LL frame and the earth rotation vector is easiest to determine in the inertial frame.

This work will make use of three types of attitude representations: direction cosine matrix (or rotation matrix), quaternion, and Euler angles. Each representation has its own advantages and disadvantages, and typically a combination of all three is used in software. Schleppe (1996) provides an excellent review of the various attitude representations and their respective conversions.

3.2.1 Direction Cosine Matrix

The direction cosine matrix represents a rotation between two arbitrary frames (e.g. rotation between frames *a* and *b*). An element within the direction cosine matrix (represented by row r and column c) contains the cosine of the angle between the *c* axis of the *a* frame and the *r* axis of the *b* frame (Savage 2007). Direction cosine matrices are unambiguous and contain no singularities. The direction cosine matrix however does have an inherent weakness because each element within the matrix does require the use of trigonometric processing, a large processing burden.

3.2.2 Quaternion

Quaternions are based on Euler‛s theorem stating that between two coordinate systems one single invariant axis exists with one rotation about that axis. The four parameter quaternion is defined by a scalar rotation angle and a unit vector, which is commonly called the Euler axis. Mathematically, the quaternion can be expressed as

$$\mathbf{q} = \begin{bmatrix} \cos\frac{\beta}{2} \\ \mathbf{e}\sin\frac{\beta}{2} \end{bmatrix} \tag{3.1}$$

where

β is the rotation about the Euler axis and

$\mathbf{e}$ is the unit vector of the Euler axis

When two coordinate systems are equivalent (e.g. parallel), the Euler axis vector is zero and the scalar quantity is 1. Opposite rotations are created by multiplying the unit vector by -1. This moves the Euler axis to the opposite quadrant, thereby allowing for inverted rotations. Quaternion parameters are dependent upon each other and satisfy the constraint that their sum of squares is unity (Schleppe 1996). This latter characteristic is why many applications favour the use of quaternions over direction cosine matrices.

3.2.3 Euler Angles

Euler angles are three sequential rotations that represent the difference in orientation between two coordinate frames. Euler angles are ambiguous and contain singularities, but are commonly preferred because they can provide a conceptually simple understanding of the attitude in 3D Euclidean space. The Euler angle ambiguity arises because no restriction is placed on the order of the sequential rotations, thus there are several potential rotations sequences that could be performed. Since Euler angles provide no ability to directly rotate a vector, they are commonly converted to a rotation matrix. Each Euler angles is used to compute a rotation matrix however the sequential order must be known. The three Euler angles (*α, β, χ*) are represented as their equivalent direction cosine matrices as follows:

$$\mathrm{R}_1 \ \alpha = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\ \alpha & -\sin\ \alpha \\ 0 & \sin\ \alpha & \cos\ \alpha \end{bmatrix} \tag{3.2}$$

$$\mathrm{R}_2 \ \beta = \begin{bmatrix} \cos\ \beta & 0 & \sin\ \beta \\ 0 & 1 & 0 \\ -\sin\ \beta & 0 & \cos\ \beta \end{bmatrix} \tag{3.3}$$

$$\mathrm{R}_3 \ \chi = \begin{bmatrix} \cos\ \chi & -\sin\ \chi & 0 \\ \sin\ \chi & \cos\ \chi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.4}$$

where:

α is the rotation about the X axis,

β is the rotation about the Y axis, and

χ is the rotation about the Z axis.

Euler angles therefore are not commonly used within software to store the rotation between frames, but are used for simple user input and output. The three parameters are also easily interpolated which makes them ideal candidates for Kalman filter states, whereas the other rotation representations are not as conformal.

3.2.4 Rotation between Frames

For the general purpose of this review, rotations will be shown using the direction cosine (rotation) matrix. Rotation from the inertial frame to the ECEF frame is a function of the earth‟s rotation rate and the interval over which to integrate the rotation. The rotation is given as

$$R_i^e = R_3 \ \omega_{ie} \bullet \Delta t \tag{3.5}$$

where:

ω_{ie} is earth‟s rotation rate vector [rad/s],

Δt is the interval of the integration time [s], and

R_j is the rotation matrix about the j^th axis

The ECEF to LL rotation matrix is given as

$$R_e^l = R_1 \ 90^\circ - \phi \ R_3 \ 90^\circ + \lambda \tag{3.6}$$

where

ϕ is the longitude [deg] and

λ is the latitude [deg].

The rotation from the body frame to the local level frame is (convention specific)

$$R_b^l = R_3 \ \psi \, R_2 \ \theta \ R_1 \ \varphi \tag{3.7}$$

where:

ψ is the yaw (rad),

θ is the roll (rad), and

φ is the pitch (rad).

The rotation from the body frame to the ECEF frame is commonly computed in the following manner

$$R_b^e = R_l^e \bullet R_b^l \tag{3.8}$$

Using a, b and c as arbitrary frames, there are a few properties of rotation matrices that are noteworthy, explicitly:

$$R_b^a = \ R_a^{b\ \ T} \tag{3.9}$$

$$R_b^a = R_c^a R_b^c \tag{3.10}$$

$$I_{3x3} = R_b^a R_a^b \tag{3.11}$$

Equation (3.9) shows that a reverse rotation can be performed by transposing the matrix. Equation (3.10) shows that subsequent rotations can be combined into one rotation matrix and that rotation matrices themselves can be rotated. Equation (3.11) shows that a rotation matrix to one frame and back to the original frame is in fact no rotation.

3.3 MEMS IMU Sensor Design

The past few decades has seen increased attention on MEMS IMU sensor design resulting from the following (Titterton & Weston 2004):

- a broad market
- low cost
- small size and weight
- extensive longevity
- no maintenance
- low power consumption
- low part number
- mass production capabilities
- “simple” construction procedures relative to the higher end IMUs
- manufactures are not required to calibrate each unit

However, the benefits come at the cost of obtaining high quality (and integrity) measurements. While this is apparent in today‟s market, Titterton & Weston (2004) claim that MEMS IMU quality may rival the current tactical grade IMU and eventually the navigation grade sensors.

3.3.1 MEMS Gyroscope Technology

Vibratory gyros are the most common type of MEMS gyro technology. MEMS vibratory gyros are based on the Coriolis force resulting from vibrating proof masses experiencing a velocity. The Coriolis force is computed as

$$\mathbf{a_c} = 2\mathbf{v} \times \mathbf{\Omega} \tag{3.12}$$

where:

$\mathbf{a_c}$ is the Coriolis force,

$\mathbf{v}$ is the velocity of the object, and

$\mathbf{\Omega}$ is the angular velocity.

The premise for a vibratory MEMS gyro is to provide a proof mass with a predetermined velocity (from vibrations), observe the Coriolis force via a change in electrical capacitance and mathematically compute the angular velocity of the proof mass. There are several components of a MEMS gyro including proof mass, motor, sensing electrodes and A/D conversion. Each component is discussed next.

3.3.1.1 Proof Masses

Proof masses are generally divided into three categories based on their design: simple beam, balanced beam and cylindrical shell oscillators. Simple beams are susceptible to external vibrations yielding incorrect observations, thus the more effective balanced beams are typically used and are commonly referred to as tuning fork gyros. Manufacturers also use a cylinder shell based proof mass, which yields even better performance parameters than the balanced beam. Proof masses are made of silicon or quartz and are typically on the order of 20 to 100 μm thick. The thickness of the mass typically translates into the stability of the gyro.

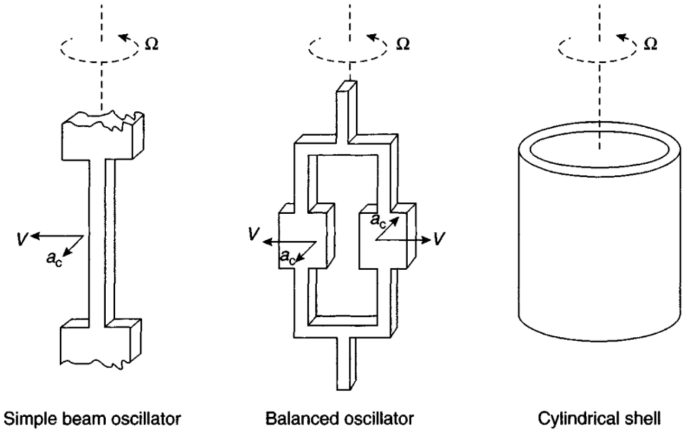


Figure 10 – MEMS GYRO Test Masses (Titterton & Weston 2004)

3.3.1.2 Motor

The proof mass requires velocity in order for a perpendicular Coriolis reaction to be measured. The motor design, shown in Figure 11, is an electrostatic comb drive which vibrates when an alternating current is applied. The proof mass commonly vibrates consistently at about 12 kHz with a consistent amplitude of about 10 μm (Titterton & Weston 2004).

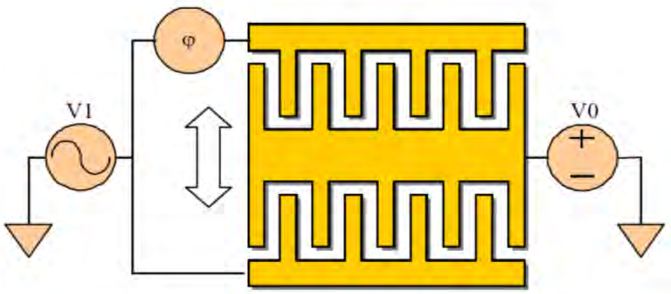


Figure 11 – Comb Drive Motor Schematics (Apostolyuk 2006)

3.3.1.3 Sensing Electrodes

As rotation (from user motion) is experienced by the vibrating mass, the Coriolis force pushes the mass out of the vibrating plane. The magnitude of this movement is measured by the change in capacitance. The capacitance variation is extremely sensitive, measuring changes near 1 aF (10^{-18} Farads). Because gyro capacitors must be so sensitive robust manufacturing techniques are used to enhance their ability. The capacitance sensitivity typically translates linearly into the gyros resolution and noise parameters (Titterton & Weston 2004).

3.3.2 MEMS Accelerometers Technology

Two general categories exist for MEMS accelerometers: pendulous and vibrating beam.

3.3.2.1 Pendulous Accelerometers

A pendulous accelerometer uses a similar proof mass as the gyros proof mass described in Section 3.3.1.1. The objective of the accelerometer is to measure the mass‟s deflection based on a change in capacitance and convert the magnitude of the deflection into a specific force. Just as in the MEMS gyro case, this capacitance must be extremely sensitive to changes on the order of 12 fF (10^{-15} F). The capacitor‟s sensitivity similarly dictates the resolution and noise parameters of the accelerometer.

3.3.2.2 Vibrating Beam Accelerometers

The vibrating beam accelerometer uses an electrostatic comb drive as described in Section 3.3.1.2. The comb drive is set to vibrate at a consistent frequency (approximately 20 kHz) and when a force is applied, a change in resonant frequency is measured (rather than an actual displacement via capacitance change).

3.4 IMU Observation Equations

The observations that an IMU provides are contaminated with both stochastic and systematic errors. The accuracy of any INS is directly related to the mitigation of these errors and therefore identification of the errors in an observation equation is critical.

The accelerometer observation equation, after calibration, is given as

$$\tilde{\mathbf{f}}_{ib}^{b} = \mathbf{b}_a + S_a \mathbf{f}_{ib}^{b} + \boldsymbol{\eta}_a \qquad (3.13)$$

where:

$\mathbf{f}_{ib}^{b}$ is the true specific force vector,

$\mathbf{b}_a$ is the bias vector of the accelerometers,

S_a is the scale factor and cross coupling error matrix, and

$\boldsymbol{\eta}_a$ is noise (assumed to be white Gaussian).

The observation is in vector form, each axis containing a specific force measurement.

Similarly, the gyro observation equation after calibration is given as

$$\tilde{\boldsymbol{\omega}}_{ib}^{b} = \mathbf{b}_g + S_g \boldsymbol{\omega}_{ib}^{b} + G_g \mathbf{f}_{ib}^{b} + \boldsymbol{\eta}_g \qquad (3.14)$$

where:

$\boldsymbol{\omega}_{ib}^{b}$ is the true angular velocity,

$\mathbf{b}_g$ is the bias vector of the gyros,

G_g specific force dependent bias matrix,

S_g is the scale factor and cross coupling error matrix, and

$\boldsymbol{\eta}_g$ is white Gaussian noise.

3.5 IMU Error Sources

Accelerometer and gyro (gyroscope) errors typically include biases, scale factors, triad non-orthogonalities and noise. The magnitude of the noise and the stability of biases are the two common attributes that usually determine the quality of the sensor. The quality of an IMU is typically reflected in its cost. Considering that this research is for pedestrian navigation, system cost becomes a major factor in the development of a marketable product. MEMS provide the IMU of choice for pedestrian applications.

3.5.1 Biases

Biases are errors independent and uncorrelated of the specific force and angular velocity experienced by the unit (IEEE 2001). The bias of a MEMS grade IMU consists of two parts, the turn-on bias and the time variant bias. The biases are typically estimated in a filter, in addition to the position, velocity, attitude and other sensor errors. Godha (2006), for example, models both the turn on bias and the time variant bias in a Kalman filter. The turn on bias commonly contains 90 % of the bias (Groves 2008), however MEMS IMU‟s are more prone to temperature based biases which can often have similar magnitudes as the turn on bias. Table 3 shows the magnitude of the biases for several grades of IMUs.

Table 3 – Typical IMU Grade Bias Specifications (Titterton & Weston 2004, Groves 2008)

IMU Grade	Gyro Bias (°/hr)	Accelerometer Bias (mg) (1 mg ~ 0.009807m/s^2)
Strategic	0.0001	0.01
Navigation	0.01	0.1
Tactical	0.1-10	1-10
Automotive/Pedestrian	>10	>10

3.5.2 Scale Factor Errors

A scale factor error is a ratio of change in the output of the sensor with respect to true intended measurement (IEEE 2001). Scale factors describe the first order trend of the bias. The scale factor can be estimated in the filter, alongside the biases. Table 4 provide typical scale factor specifications for varying grades of IMUs.

Table 4 –Typical IMU Grade Scale Factor Specifications

IMU Grade	Gyro Scale Factor (ppm)	Accelerometer Scale Factor (ppm)
Strategic	Nil	Nil
Navigation	5	100

Tactical	150	300
Automotive/Pedestrian	10,000	10,000

3.5.3 Cross Coupling Errors

Cross coupling or misalignment errors result from the non-orthogonality of the sensor triad making the observing axis sensitive to input from the adjoining normal axes (Groves 2008). The errors can be quantified by coefficients, which occupy the off-diagonal terms of the $S_{a,g}$ matrix and as such can also be expressed in PPM. These errors are most commonly estimated through calibration. The measurements are then corrected prior to being used in the navigation filter or estimation process.

3.5.4 Gyro Specific Force Errors

Gyros that use a spinning or vibratory mass incur additional errors because of imbalances in the proof mass. These errors are a function of the amount of specific force applied to the gyro triad and commonly have magnitudes between 1 and 100 °/hr/g. Thus, an inertial unit located on a foot, which can experience 4 g of acceleration during heel strike, could produce a 400 °/hr error.

3.5.5 Random Noise

Electrical limitations and mechanical instabilities generate noise in all IMU observations. MEMS IMUs suffer from significant noise because their signals are extremely weak and the ability to measure minute fluctuations is challenging. Figure 12 and Figure 13 show the histogram of 15 hours of static data of a MEMS grade IMU (Cloudcap‟s Crista IMU at 100 Hz). A visual inspection shows that the errors follow a Gaussian distribution. This test required that the mean be removed to account for levelling errors and systematic IMU errors. The mean was estimated by using a 100 epoch mean filter.

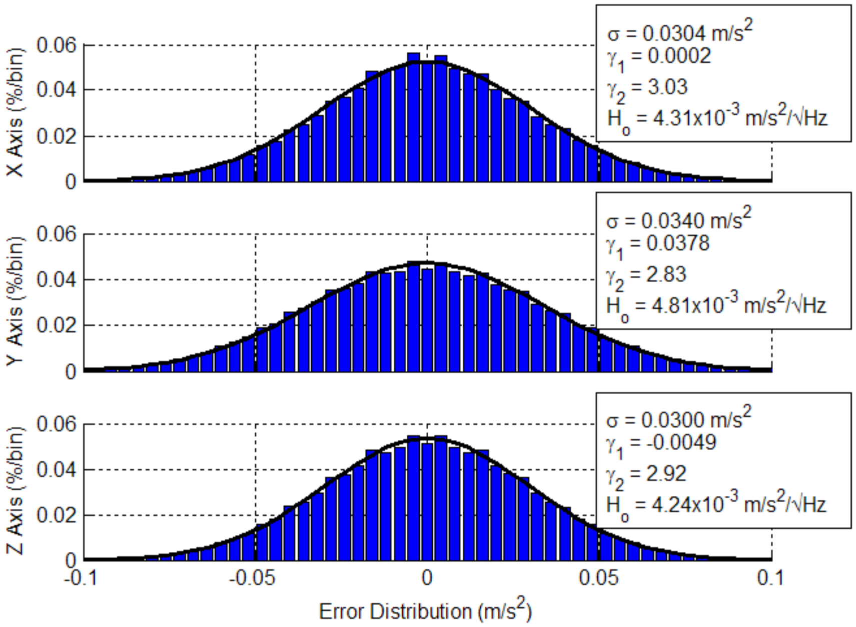


Figure 12 – Accelerometer Noise Characteristics of a Static MEMS Grade IMU (15 hours @ 100 Hz)[1]

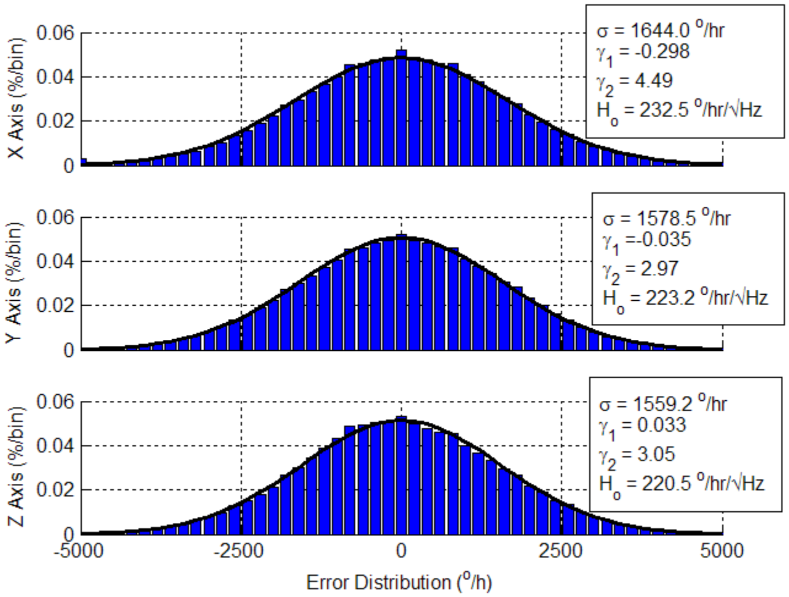


Figure 13 – Gyroscope Noise Characteristics of a Static MEMS Grade IMU (15 hours @ 100 Hz)

[1] σ is the standard deviation and shown in the figure with a black line, γ_1 is the skewness of the data, γ_2 is the Kurtosis of the data and H_o is the spectral density of the data.

3.6 Mechanization Equations

The mechanization of the raw IMU measurements converts the specific force ($\mathbf{f}_{ib}^{b}$) and angular velocity ($\boldsymbol{\omega}_{ib}^{b}$) to an incremental change in position, velocity and attitude over a time interval. Mechanization only provides relative information and consequently can only provide the accumulated change in position. Therefore it is ideal to combine these measurements with an absolute positioning system such as GPS. The relative navigation information of mechanization can be determined in several frames, such as a local level frame, the ECEF frame, wander frame or even the inertial frame. Mechanization frame implementations each contain their own advantages and disadvantages. Ultimately the frame used is selected for the application, computational burden and developer preference. This thesis will use the ECEF mechanization. Mathematically, the ECEF mechanization equations are represented as

$$\begin{bmatrix} \dot{\mathbf{r}}^{e} \\ \dot{\mathbf{v}}^{e} \\ \dot{R}_{b}^{e} \end{bmatrix} = \begin{bmatrix} \mathbf{v}^{e} \\ R_{b}^{e}\mathbf{f}_{ib}^{b} - 2\Omega_{ie}^{e}\mathbf{v}^{e} + \boldsymbol{\gamma}^{e} \\ R_{b}^{e} \;\; \Omega_{ei}^{b} + \Omega_{ib}^{b} \end{bmatrix} \tag{3.15}$$

where:

$\mathbf{r}^{e}$ is the ECEF position vector,

$\mathbf{v}^{e}$ is the ECEF velocity vector,

R_{b}^{e} is the rotation matrix from the body frame to the ECEF frame,

Ω_{ie}^{e} is the skew symmetric form of the angular velocity between the inertial and ECEF frame, as viewed in the ECEF frame,

$\boldsymbol{\gamma}^{e}$ is the normal gravity vector (i.e. an estimation of local gravity provided by a mathematical model),

Ω_{ei}^{b} is the skew symmetric form of the angular velocity between the ECEF and inertial frame, as viewed in the body frame, and

Ω_{ib}^{b} is the skew symmetric form of the angular velocity between the inertial and body frame, as viewed in the body frame (the Gyro measurements)

In practice, the mechanization of inertial measurements into position, velocity and attitude information is not implemented as a set of differential equations as above but rather as a series of computational steps. These steps are discussed here.

The first step is to correct the incoming IMU data with the best estimate of their respective errors as described in Section 3.5. Estimates of the errors are derived either from a filter (i.e. Kalman filter), an *a priori* calibration process, or both. Some errors such as the sensor axis non-orthogonality are also performed in a pre-filtering step, sometimes within the IMU processor. In the case of the gyro, the specific force dependent bias errors are also corrected in a pre-filtering step. The correction of the raw IMU measurements is performed mathematically as:

$$\hat{\omega}_{ib}^{b} = \frac{\tilde{\omega}_{ib}^{b} - b_g}{S_g} \tag{3.16}$$

$$\hat{f}_{ib}^{b} = \frac{\tilde{f}_{ib}^{b} - b_a}{S_a} \tag{3.17}$$

.

The noise of each sensor cannot be removed at this stage and therefore the residuals are assumed to be stochastic. In cases where residual errors remain, this assumption may not reflect the truth.

The second step is to use the integrated gyro measurement to provide a change in rotation experienced during the interval of mechanization. Since earth rotation is observed within the gyro observation it first must be removed. This ensures that only the specific rotation for the IMU is computed and does not include ambient rotations. The gyro outputs the observations within the body frame and therefore earth rotation must be determined in the body frame. The earth rotation rectification is given as

$$\begin{aligned}\boldsymbol{\omega}_{eb}^{b} &= \hat{\boldsymbol{\omega}}_{ib}^{b} - \boldsymbol{\omega}_{ie}^{b} \\ &= \hat{\boldsymbol{\omega}}_{ib}^{b} - R_{e}^{b}\boldsymbol{\omega}_{ie}^{e} \\ &= \hat{\boldsymbol{\omega}}_{ib}^{b} - R_{e}^{b}\begin{bmatrix}0\\0\\\omega_{ie}\end{bmatrix}\end{aligned} \tag{3.18}$$

where

ω_{ie} is the earth"s rotation rate (~ 15.041 °/hr).

The angular velocity expressed in Equation (3.18) is then integrated over the epoch to provide a small rotation vector between the k^{th} and $k^{th}+1$ epoch. The rotation vector is then converted to a quaternion as detailed in Savage (2007). Then through quaternion multiplication the quaternion representing the rotation between the body and ECEF frame is updated. The attitude update step is shown as

$$Q_{eb}^{e}\,{}_{k+1} = Q_{eb}^{e}\,{}_{k} * \Delta Q_{eb}^{e} \tag{3.19}$$

where

Q_{ba}^{c} is the quaternion describing the rotation from the „a" to „b" frame, as seen in the c frame.

Step 3 is to provide incremental velocities in the ECEF frame. The specific force measurements are integrated over the interval to yield incremental velocities. As a result the rotation into the ECEF frame is applied to the velocity increments. However, because the rotation matrix from the body to the ECEF frame is only available at the discrete ends of the integration period, the average of the rotation matrices and the slight incremental rotation during the interval must be accounted for. If the interval occurs over k and k+1 epochs, it is mathematically shown as

$$\begin{aligned}\Delta\mathbf{v}_{k}^{e} &= R_{b}^{e}\,{}_{k}\left(I_{3x3} + \frac{1}{2}S_{b}\right)\Delta\mathbf{v}^{b} \\ \Delta\mathbf{v}_{k+1}^{e} &= R_{b}^{e}\,{}_{k+1}\left(I_{3x3} - \frac{1}{2}S_{b}\right)\Delta\mathbf{v}^{b}\end{aligned} \tag{3.20}$$

where

S_b is the skew symmetric form of the rotation vector.

Thus, the average of the computed Δv_{ib}^{e} is used to finally provide the updated velocity, explicitly as

$$\mathbf{\Delta v}_{k,k+1}^{e} = \frac{\mathbf{\Delta v}_{k}^{e} + \mathbf{\Delta v}_{k+1}^{e}}{2} \quad (3.21)$$

At this point the Coriolis and the gravity effects must be compensated for. Gravity is approximately estimated using a global model. These models typically account for gravitation and centripetal acceleration, but do not provide the resolution required for estimating local gravitation anomalies. Regardless, the normal gravity vector nominally contain residual gravitation errors on the order of 10^{-8} m/s^2, resulting in minuscule errors for pedestrian navigation applications (Schwarz & Wei 1990). Coriolis acceleration is computed using the average velocity over the interval. The adjustment is given as

$$\mathbf{\Delta v}_{k,k+1}^{e} = \mathbf{\Delta v}_{k,k+1}^{e} - 2\mathbf{\Omega}_{ie}^{e}\mathbf{v}_{k,k+1}^{e} - \boldsymbol{\gamma}^{e}\ \mathbf{\Delta t} \quad (3.22)$$

The incremental position is then determined by integrating the velocity and added to the previous epoch"s position as

$$\mathbf{r}_{k+1}^{e} = \mathbf{r}_{k}^{e} + \mathbf{v}_{k,k+1}^{e}\mathbf{\Delta t} \quad (3.23)$$

3.7 Initial Alignment

The attitude of an IMU with respect to the mechanization frame is required prior to updating the position, velocity and attitude (PVA) as seen in Equation (3.18), thus requiring an initial attitude. The process of determining the heading from gyro observations is called gyro-compassing, however due to the large biases and high noise of MEMS gyros, gyro-compassing is not possible and therefore not discussed here. Other possible initial heading alignment procedures are performed through magnetic compasses or having the user input their heading manually.

Assuming the IMU is stationary the pitch and roll of the IMU can be determined through the accelerometers as

$$\sin \varphi = \frac{\mathrm{f}^{\mathrm{b}}_{\mathrm{ib}\ \ \mathrm{x}}}{\|\gamma\|} \qquad (3.24)$$

$$\sin \theta = \frac{\mathrm{f}^{\mathrm{b}}_{\mathrm{ib}\ \ \mathrm{y}}}{\|\gamma\|} \; . \qquad (3.25)$$

Based on the gravity vector and the observed magnitude of the IMU horizontal axes, the pitch and roll can be determined which is shown in Equations (3.24) and (3.25). It is noted that since the observed accelerometer values are uncorrected (i.e. the filter has not yet started to estimate biases) the initial attitude still contain errors. Given a maximum bias of 0.3 m/s^2 (as per the specifications sheet), the error in the pitch and roll would be approximately 1.8°.

Chapter Four: Estimation Theory

When trying to estimate a parameter such as position or velocity from pseudoranges or Doppler measurements, there can be three separate scenarios: (1) the number of independent observations is less than the number of desired parameters; (2) the number of observations is equal to the number of parameters; or (3) the number of independent observations is greater than the number of parameters. The first case is said to be under-determined. In such a scenario, the parameters may not be fully resolved since the observation space is not sufficient to translate the entire parameter space. The latter two scenarios are called fully-determined and over-determined respectively, since the observation space is completely defined and adequate for translation into the desired parameters.

The method of estimating parameters from redundant observations and determining the estimated covariance of the estimated parameters is known as estimation theory. Measurement systems convert redundant measurements into one set of parameters, most commonly through a least squares approach. If the system contains dynamics the dynamics characteristics can be fused with the redundant measurements in an optimal form. Kalman filtering is a classical estimation algorithm utilizing both measurements and dynamics.

4.1 Measurement Systems

The observation equation relates the parameters to be estimated and the observations. Equations (2.2) and (3.13) are examples of observation equations.

Consider a parametric system of the kth epoch that is described as follows

$$\mathbf{l}_k = H_k \mathbf{x}_k + \boldsymbol{\varepsilon}_k \tag{4.1}$$

where:

$\mathbf{l}_k$ is the vector of observations,

H_k is the design matrix,

$\mathbf{x}_k$ is the parameter vector, and

$\boldsymbol{\varepsilon}_k$ is the measurement noise.

The design matrix is a Jacobian matrix of the observation equations, effectively a linear conversion from the parameter to the observation domain.

In systems that are over-determined, the goal is to minimize the magnitude of the residual error. This magnitude is described as the cost since it is an undesirable outcome of observation errors. The cost function of least squares (i.e. the error to be minimized) is mathematically shown as

$$J = \mathbf{l}_k - \mathbf{H}_k x_k \;{}^{T} \mathbf{W}_k \; \mathbf{l}_k - \mathbf{H}_k x_k \qquad (4.2)$$

where

J is the cost function to be minimized and

$\mathbf{W}_k$ is a weighting matrix.

Least squares equations are derived by differentiating Equation (4.2) with respect to the parameters (x_k), setting the differentiated function to zero and solving for the parameters (Gao 2008). The covariance of the parameters is determined by propagating the covariance of the observations as given by

$$P = \left(\frac{\partial f}{\partial \ell}\right) R \left(\frac{\partial f}{\partial \ell}\right)^{T} \qquad (4.3)$$

where:

P is covariance matrix of the estimated parameters,

R is the covariance matrix of the observations,

ℓ is the observation, and

f is the parametric mathematical model.

In least squares it is common to let the weighting matrix equal to the inverse of the covariance matrix of the observations (R^{-1}). With this approach the estimated parameters and the corresponding covariance matrix can be computed as

$$\hat{\mathbf{x}}_k = \; H_k^T R_k^{-1} H_k \; ^{-1} H_k^T R_k^{-1} \mathbf{l}_k \tag{4.4}$$

$$P_k = \; H_k^T R_k^{-1} H_k \; ^{-1} \tag{4.5}$$

After the adjustment, the residual error vector and its covariance matrix are computed as

$$\mathbf{r}_k = \mathbf{l}_k - H_k \hat{\mathbf{x}}_k \tag{4.6}$$

where

$\mathbf{r}_k$ are the residuals, and

$$C_{r_k} = R_k - H_k P_k H_k^T \tag{4.7}$$

where

C_{r_k} is residuals covariance matrix.

4.2 Dynamic Systems

If a differential relationship exists between states[2], the dynamic system can be incorporated into the estimation process thereby increasing the quality of the estimation. Dynamic systems are described as

$$\dot{x} \; t \; = F \; t \; x \; t \; + G \; t \; w \; t \tag{4.8}$$

where:

$F \; t$ is a coefficient matrix describing the differential equations,

$G \; t$ is the coefficient matrix shaping the input noise,

$w \; t$ is the white noise vector, and

[2] The nomenclature for dynamic systems typically uses states rather than parameters, although they are effectively the same thing.

t is time.

Note that the system dynamics do not contain observations. This is because dynamic systems only relate the parameters within the system. The combination of system dynamics and measurements is usually fused through a Kalman filter.

4.3 Kalman Filtering

The Kalman filter is a recursive algorithm forming a minimum variance estimation of system dynamics and measurements. The filter operates in two steps: a prediction that utilizes system dynamics to predict the next state, and an update which relates the measurements to the parameter space and combines them with the prediction to give the final estimates. The combination of prediction and measurements requires a concept called the Kalman gain. Effectively, the Kalman gain is a method of optimizing the weight of incoming measurements with respect to the prediction. In this manner the filter bases the solution on a fused prediction and update. To distinguish between a prediction and an update stage, the superscript "-" and "+" are used, respectively.

There are three fundamental assumptions in a Kalman filter (Gao 2008). These assumptions are that (1) there is no time correlation of the process noise (w), (2) there is no time correlation of the measurement errors (ε), and (3) there is no correlation between the process noise and measurement noise. These are expressed as:

$$\mathrm{E}\left[\mathbf{w}_k\mathbf{w}_i^T\right]=\begin{Bmatrix}Q_k, i=k\\ 0, i\neq k\end{Bmatrix} \tag{4.9}$$

$$\mathrm{E}\left[\boldsymbol{\varepsilon}_k\boldsymbol{\varepsilon}_i^T\right]=\begin{Bmatrix}R_k, i=k\\ 0, i\neq k\end{Bmatrix} \tag{4.10}$$

$$\mathrm{E}\left[\mathbf{w}_k\boldsymbol{\varepsilon}_i^T\right]=\begin{Bmatrix}0, i=k\\ 0, i\neq k\end{Bmatrix} \tag{4.11}$$

where

Q_k is the process noise.

These assumptions are critical to the propagation of the state covariance matrix, which is used within the filter to determine the weight of incoming observations. Thus, failure in these three assumptions typically results in overly optimistic state variances, poor performance and reduced fault detection and exclusion capabilities. Unfortunately, in navigation applications these assumptions are not always true. GPS multipath errors for example cause biased time correlations (Lachapelle 2007). For the purpose of this research, increasing the measurement variance provides sufficient results. There are other approaches to deal with the violation of the three assumptions, but extend beyond the focus of this thesis.

The Kalman filter operates in discrete mode for navigation applications. Discrete mode refers to the fact that observations occur at discrete times rather than through a continuous series. Thus, the system dynamic model must first be converted to discrete time in order to predict the states. The state vector ($\mathbf{x}$) is predicted by

$$\hat{\mathbf{x}}_{k+1}^{-} = \Phi_{k,k+1}\hat{\mathbf{x}}_{k} \tag{4.12}$$

where

$\Phi_{k,k+1}$ is the transition matrix from the k to k+1 epochs.

The transition matrix is a time transformation matrix, effectively translating a state vector in time. It is computed directly from the dynamics matrix. The transition matrix (Φ) is theoretically defined as

$$\Phi_{k,k+1} = e^{F\ t_{k+1}-t_k} = e^{F\Delta t}, \tag{4.13}$$

but practically computed through the following Taylor series expansion

$$\Phi_{k,k+1} \approx I + F\Delta t + \frac{F\Delta t^{\ 2}}{2!} + \frac{F\Delta t^{\ 3}}{3!} + \ldots. \tag{4.14}$$

The covariance of the states in the prediction is computed as

$$P^{-}_{k+1} = \Phi_{k,k+1} P_k \Phi^{T}_{k,k+1} + Q_k . \tag{4.15}$$

This equation conceptually propagates the current covariance in time and adds a user defined amount of uncertainty to the prediction. This compensates for any incorrect assumptions about the dynamic process, such as un-modelled movements or higher order effects.

The process noise is typically expressed in the continuous time domain as

$$Q_k = \int_{t_k}^{t_{k+1}} \Phi_{\tau,k+1} G \ \tau \ Q_c \ \tau \ G^{T} \ \tau \ \Phi^{T}_{\tau,k+1} d\tau \tag{4.16}$$

where

$G \ \tau$ is the shaping matrix and

$Q_c \ \tau$ is the continuous time spectral density of the white noise vector.

However, it must be computed in discrete time. Equation (4.16) shows the computation of process noise through trapezoidal integration as

$$Q_k = \Phi_{k,k+1} G_k Q_c G_k{}^{T} \Phi^{T}_{k,k+1} + G_k Q_c G_k{}^{T} \ \frac{t_{k+1} - t_k}{2} . \tag{4.17}$$

The Kalman filter can operate in prediction mode as long as necessary (i.e. in the absence of measurements) and at any frequency. This is an attractive feature in that a solution and its propagated variance-covariance matrix can be provided to the user at any time even when measurements are not available.

As measurements become available the filter enters the update stage, where the measurements are fused with the prediction(s). The Kalman gain matrix is a weighting matrix used to determine the amount of new information added to the system. In the event that measurements are overly noisy, the Kalman gain decreases their effect on the final solution. The Kalman gain equation is given as

$$K_k = P_k^- H_k^T \;\; H_k P_k^- H_k^T + R_k \;\; ^{-1} \,. \tag{4.18}$$

The innovation sequence is given as

$$\boldsymbol{\upsilon}_k = \mathbf{l}_k - H_k \hat{\mathbf{x}}_k^-, \tag{4.19}$$

where

$\boldsymbol{\upsilon}_k$ is the innovation sequence.

and represents the difference between the observations and the current states mapped back into the observation domain. The innovation sequence covariance matrix is computed as

$$C_{\upsilon_k} = H_k P_x H_k^T + R_k \,. \tag{4.20}$$

Any erroneous observations are detected by exceeding a threshold (e.g. 3σ). This fact forms the basis for fault detection and observation exclusion.

After the computation of the Kalman gain, the states can then be updated as

$$\hat{\mathbf{x}}_k^+ = \hat{\mathbf{x}}_k^- + \mathbf{K}_k \boldsymbol{\upsilon}_k \,. \tag{4.21}$$

The covariance of the states is computed during the update stage and is given as

$$P_k^+ = \;\; I - K_k H_k \;\; P_x^- \,. \tag{4.22}$$

A more common covariance update, which provides better symmetry and positive definiteness (at the cost of computational burden), known as the Joseph form is given as (Grewal & Andrews 2001)

$$P_k^+ = \;\; I - K_k H_k \;\; P_k^- \;\; I - K_k H_k \;\; ^T + K_k R_k K_k^T \,. \tag{4.23}$$

4.4 Extended Kalman Filter

Often the measurement and dynamic systems are *not* linear functions, as the case of GPS and IMU mechanization. Thus, a linearization about the last filter estimate is applied to the standard form of the Kalman filter. The derived filter is typically referred to as the Extended Kalman filter (EKF).

4.4.1 Non Linear Dynamic Model

To perform a linearization on the dynamic model, first consider rewriting the form of Equation (4.8) to incorporate a Taylor series expansion, such as

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}, t) + \mathbf{G}(t)\mathbf{w}(t) . \tag{4.24}$$

The expansion is formed as

$$\dot{x}(t) = f(x_0, t) + f'(x_0, t)(x - x_0) + ... + G(t)w(t) \tag{4.25}$$

and after neglecting the higher order terms of the Taylor series expansion, Equation (4.25) then simplifies to

$$\begin{aligned} \dot{x}(t) &\approx f(x_0, t) + f'(x_0, t)(x - x_0) + G(t)w(t) \\ \dot{x}(t) &= \dot{x}_0(t) + F\delta x(t) + G(t)w(t) \\ \delta\dot{x}(t) &= F\delta x(t) + G(t)w(t) , \end{aligned} \tag{4.26}$$

which has the same general form as Equation (4.8). The EKF then estimates the perturbation of the expansion point (more correctly, the expansion vector) and therefore the perturbation must be added to the expansion point to provide the absolute values of the state vector. The state vector after each update is reset to zero. This restricts the state vector to be zero mean. With this restriction Equation (4.21) now has the form

$$\delta\hat{\mathbf{x}}_k^+ = \mathbf{K}_k \boldsymbol{\upsilon}_k , \tag{4.27}$$

and all remaining covariance equations remain the same.

4.4.2 Non Linear Measurement Model

The measurement model linearization is performed in a similar fashion to that of the dynamic model. Equation (4.1) is rewritten as

$$\mathbf{l}_k = h(\mathbf{x}_k) + \boldsymbol{\varepsilon}_k . \tag{4.28}$$

The linearization is performed in the same manner as Equations (4.24) to (4.26) and the resulting equation becomes

$$\delta\mathbf{l}_k = H\delta\mathbf{x}_k + \boldsymbol{\varepsilon}_k . \qquad (4.29)$$

When an EKF is used with non-linear measurements, Equation (4.21) is adapted in the following way

$$\begin{aligned}
\hat{\mathbf{x}}_k^+ &= \hat{\mathbf{x}}_k^- + \underbrace{\delta\mathbf{x}_k^- + K_k \quad \delta\mathbf{l}_k - H\delta\hat{\mathbf{x}}_k^-}_{\delta\hat{x}_k^+} \\
\hat{\mathbf{x}}_k^+ &= \hat{\mathbf{x}}_k^- \cancel{+\delta\hat{\mathbf{x}}_k^-} + K_k \left(\delta\mathbf{l}_k - \cancel{H\delta\hat{\mathbf{x}}_k^-}\right) \\
\hat{\mathbf{x}}_k^+ &= \hat{\mathbf{x}}_k^- + K_k\delta\mathbf{l}_k .
\end{aligned} \qquad (4.30)$$

4.5 Adaptive Kalman Filter

In some filtering applications, determining the correct amount of process noise to be added can be extremely difficult. Filter tuning, whereby the actual values are tuned to the particular data set can improve the overall performance. However, it is not always realistic to tune each data set or is always possible to derive tuning parameters that will work in all situations.

There are two common approaches to adaptive processing: Multiple Model (MM) and Innovation Based (IB). Multiple Model adaptive processing uses a bank of Kalman filters to derive numerous solutions and then weights each solution based on hypothesis testing of the innovation sequence relative to each filter in the bank. This brute force approach is computationally expensive and excessive. Innovation based adaptive filtering has therefore gained more attention and is the method chosen for this thesis.

In IB adaptive filtering, the Q_k and/or R_k are adapted over time based on the whiteness of the innovation sequence. IB adaptive filtering has shown improvements in some situations, but can be extremely volatile if any of the three Kalman filter assumptions fail (Mohamed & Schwarz 1999).

To estimate Q_k and R_k, the innovation sequence covariance matrix is determined from the innovation sequence over a set period of time (or epochs). Thus, the estimated innovation covariance matrix is computed as (Mohamed & Schwarz 1999)

$$\hat{C}_{\upsilon_k} = \frac{1}{N}\sum_{i=i_0}^{k} \boldsymbol{\upsilon}_i \boldsymbol{\upsilon}_i^T \tag{4.31}$$

where

$\hat{C}_{\boldsymbol{\upsilon}_k}$ is the innovation based covariance matrix of the innovation sequence,

i_0 is the index at the beginning of the period,

k is the index at the end of the period, and

N is the number epochs within the period.

Using Equation (4.31), Q_k and R_k are computed as

$$\hat{Q}_k = K_k \hat{C}_{\upsilon_k} K_k^T \tag{4.32}$$

$$\hat{R}_k = \hat{C}_{\upsilon_k} - HP_k^- H^T . \tag{4.33}$$

4.6 Decentralized Filtering

Decentralized filtering is a two-stage data processing technique. Decentralized filters involve a series of filters where the outputs are input into a final fusion technique that ultimately provides a final solution. Decentralized filters are composed of two types of filters, namely local and master. The local filters process their own data in parallel with the other local filters to provide the best possible local estimate. The master filter then uses the output of the local filters as input to provide an optimal global estimate. Figure 14 shows a decentralized architecture. Loosely coupled GPS and IMU integration is a prime example of a decentralized filter where the IMU mechanization occurs within the master filter and GPS is processed in a local filter to provide position and velocity information. There is no restriction on the estimation type, for example a local filter could be a least squares estimator (in which the local filter should be referred to as a local estimator). However, the local filters are usually Kalman filters and therefore present some issues that are discussed below.

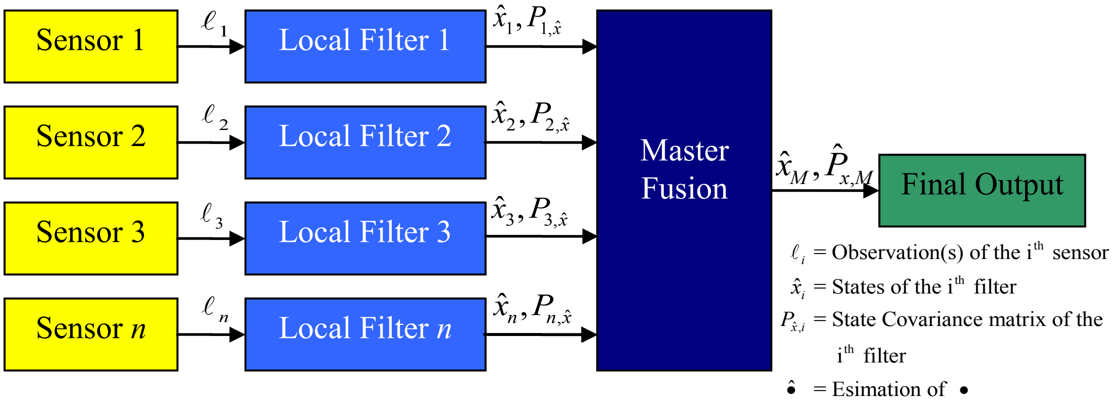

Sensor 1
Sensor 2
Sensor 3
Sensor *n*
ℓ_1
ℓ_2
ℓ_3
ℓ_n
Local Filter 1
Local Filter 2
Local Filter 3
Local Filter *n*
$\hat{x}_1, P_{1,\hat{x}}$
$\hat{x}_2, P_{2,\hat{x}}$
$\hat{x}_3, P_{3,\hat{x}}$
$\hat{x}_n, P_{n,\hat{x}}$
Master Fusion
$\hat{x}_M, \hat{P}_{x,M}$
Final Output
ℓ_i = Observation(s) of the ith sensor
$\hat{x}_i$ = States of the ith filter
$P_{\hat{x},i}$ = State Covariance matrix of the ith filter
$\hat{\bullet}$ = Esimation of $\bullet$

Figure 14 – Decentralized Filtering (No Information Sharing)

There are two fundamental weaknesses within decentralized filter architectures, namely:

1. If the master filter is a Kalman filter, it requires that all input must be stochastic with no time correlation. Typical of any local filtering, time correlation results from the dynamics inherent to the system. No time correlation within observations is a fundamental characteristic and maintains the optimal solution of the master filter. If not respected, the results can be biased and provide overly optimistic covariance estimates. Brown & Hwang (1997) state that using the output of the local filters at the approximate time correlation period can yield satisfactory results and bypass this effect. However, this requires previous knowledge of the time correlation characteristics, something that is not always available.
2. The local filters may not necessarily make use of all the information available. This in turn reduces the quality of the local filter, when in fact other pertinent information is available. The master filter may not be able to recover all the information from the data provided by the local filters; to which point the master filter is then considered sub-optimal and consequently the information passed back to the local filters is also sub-optimal (Brown & Hwang 1997). For

example, consider a black box altimeter, which only outputs an estimated elevation, used in conjunction with a local GPS elevation filter. The black boxed altimeter could provide a more accurate elevation output if an approximate elevation was fed into the altimeter. In this manner the altimeter could be calibrated, rather than operate in an autonomous mode. In this configuration the altimeter"s filter (or whatever estimation technique employed within) is sub-optimal when considering the elevation from the GPS filter could be used.

Often the results of the master filter can be fed back to the local filters. This can help assist the local filter performance thereby improving the master filter performance. Moreover, this approach optimizes the local filters performance (since local filters only use a portion of the observed data). This is the basis for a federated filter.

4.7 Federated Filtering

Federated filtering is defined within this thesis as a decentralized filter that incorporates information sharing between local and master filters. Federated filtering is a two stage process, segmenting information processing between local and master filters. The method of sharing information varies but there are typically four genres of sharing information: no reset, fusion reset, zero reset and cascaded. Before each method is outlined, it is important to first understand how to conserve information between any given number of local filters and the master filter.

4.7.1 Information Conservation Principle

The basic principle of information sharing follows three steps (Carlson & Berarducci 1994):

1. Divide the total information among several local filters.
2. Perform local filter propagation and measurement update within each local filter.
3. Recombine the information from the local filters within an optimal master fusion algorithm.

In order to show the mathematics behind the information conservation principle, alternative Kalman filter equations are used (Brown & Hwang 1997). The derivation will follow from Brown & Hwang (1997) and Carlson (2002) and shows that the information conservation principle within the federated filter is optimally equal to the centralized version, although practically this may not always be the case. A rigorous derivation is available in Carlson (1990). He shows that when all inputs to local filters are independent, the federated filter is equivalent to the centralized Kalman filter. In the context of describing the information principle the subscript "C" will represent a centralized filter, "M" will represent a master filter within the federated filter and "L" will represent a local filter within the federated filter.

The state information matrix (the inverse of the state covariance matrix), is used in the update and is given as

$$P_C^{+\ -1} = P_C^{-\ -1} + H^T R^{-1} H . \tag{4.34}$$

Conceptually, Equation (4.34) shows that the final information of the centralized update is the sum of the information provided in the prediction and the information provided in the observations. It can be shown in terms of the information matrix that

$$P_C^{+\ -1} = P_C^{-\ -1} + P_k^{\ -1} . \tag{4.35}$$

If the observation covariance matrix is assumed to be block diagonal, Equation (4.34) can be rewritten as

$$\begin{aligned} P_C^{+\ -1} &= P_C^{-\ -1} + H_1^T R_1^{-1} H_1 + H_2^T R_2^{-1} H_2 + \ldots + H_n^T R_n^{-1} H_n \\ &= P_C^{-\ -1} + \sum_{i=1}^{n} H_i^T R_i^{-1} H_i \\ &= P_C^{-\ -1} + \sum_{i=1}^{n} P_i^{\ -1} . \end{aligned} \tag{4.36}$$

To complete the alternate version of the Kalman filter equations, the Kalman gain is computed using the state covariance as

$$\mathrm{K} = \mathrm{P}^{+}\mathrm{H}^{\mathrm{T}}\mathrm{R}^{-1}. \tag{4.37}$$

The update of the state vector can also be rewritten, using Equations (4.22) and (4.37)

$$\begin{aligned}
\hat{\mathbf{x}}^{+} &= \hat{\mathbf{x}}^{-} + \mathrm{K}\boldsymbol{\upsilon} \\
&= \hat{\mathbf{x}}^{-} + \mathrm{K} \;\; \mathbf{l} - \mathrm{H}\hat{\mathbf{x}}^{-} \\
&= \hat{\mathbf{x}}^{-} + \mathrm{K}\mathbf{l} - \mathrm{K}\mathrm{H}\hat{\mathbf{x}}^{-} \\
&= \;\; \mathrm{I} - \mathrm{K}\mathrm{H} \;\; \hat{\mathbf{x}}^{-} + \mathrm{K}\mathbf{l} \\
&= \mathrm{P}^{+} \;\; \mathrm{P}^{-} \;\; {}^{-1}\hat{\mathbf{x}}^{-} + \;\; \mathrm{P}^{+}\mathrm{H}^{\mathrm{T}}\mathrm{R}^{-1} \;\; \mathbf{l} \\
\hat{\mathbf{x}}^{+} &= \mathrm{P}^{+} \;\; \mathrm{P}^{-} \;\; {}^{-1}\hat{\mathbf{x}}^{-} + \mathrm{H}^{\mathrm{T}}\mathrm{R}^{-1}\mathbf{l}
\end{aligned} \tag{4.38}$$

which shows that the updated states are a blend of the old state information and the new measurements.

It is now possible to form the master filter version of Equation (4.36) of the sum of information from the master and local filters, (again assuming no correlation between the local filters) as

$$\begin{aligned}
\mathrm{P}_{\mathrm{M}}^{+} \;\; {}^{-1} &= \;\; \mathrm{P}_{\mathrm{M}}^{-} \;\; {}^{-1} + \sum_{i=1}^{n} \;\; \mathrm{P}_{\mathrm{L}_{i}} \;\; {}^{-1} \\
\mathrm{P}_{\mathrm{M}}^{+} \;\; {}^{-1} &= \;\; \mathrm{P}_{\mathrm{M}}^{-} \;\; {}^{-1} + \sum_{i=1}^{n} \Big[\;\; \mathrm{P}_{\mathrm{L}_{i}}^{-} \;\; {}^{-1} + \mathrm{H}\mathrm{R}_{i}^{-1}\mathrm{H}^{\mathrm{T}} \Big] \\
\underbrace{\;\; \mathrm{P}_{\mathrm{M}}^{+} \;\; {}^{-1}}_{\mathrm{P}^{+} \;\; {}^{-1}} &= \underbrace{\;\; \mathrm{P}_{\mathrm{M}}^{-} \;\; {}^{-1} + \sum_{i=1}^{n} \;\; \mathrm{P}_{\mathrm{L}_{i}}^{-} \;\; {}^{-1}}_{\mathrm{P}^{-} \;\; {}^{-1}} + \underbrace{\sum_{i=1}^{n} \mathrm{H}\mathrm{R}_{i}^{-1}\mathrm{H}^{\mathrm{T}}}_{\mathrm{H}\mathrm{R}^{-1}\mathrm{H}^{\mathrm{T}}}
\end{aligned} \tag{4.39}$$

.

Equation (4.39) shows that the total information contained in the master filter is the sum of the current master filter information, the sum of all the local filters information and the information contained in the observations of each local filter. This shows the principle of information conservation, as Equation (4.39) must conform to the standard form set in (4.34), otherwise the master filter will be sub-optimal. Thus, a federated filter incorporating the information sharing principle must therefore satisfy the equation

$$ {P_C^-}^{-1} = {P_M^-}^{-1} + \sum_{i=1}^{n} {P_{L_i}^-}^{-1} \tag{4.40} $$

and ensure that the information contained in the master filter is not duplicated in the local filters.

Similarly the state vector version in Equation (4.38) can be rewritten as

$$ \begin{aligned} \mathbf{x}_M^+ &= P_M\left({P_M^-}^{-1} \mathbf{x}_M^- + \sum_{i=1}^{n} P_i^{-1}\mathbf{x}_i \right) \\ \mathbf{x}_M^+ &= P_M\left({P_M^-}^{-1} \mathbf{x}_M^- + \sum_{i=1}^{n}\left[{P_{L_i}^-}^{-1} \mathbf{x}_i + HR_i^{-1}\mathbf{l}_i \right]\right) \\ \mathbf{x}_M^+ &= P_M\left({P_M^-}^{-1} \mathbf{x}_M^- + \sum_{i=1}^{n} {P_{L_i}^-}^{-1} \mathbf{x}_i + \sum_{i=1}^{n} HR_i^{-1}\mathbf{l}_i \right) \end{aligned} \tag{4.41} $$

provided that the master filter complies with the following condition

$$ {P_C^-}^{-1} \mathbf{x}_C^- = {P_M^-}^{-1} \mathbf{x}_M^- + \sum_{i=1}^{n} {P_{L_i}^-}^{-1} \mathbf{x}_i \, . \tag{4.42} $$

The process noise within the federated filter must also be shared. Using a similar process as above the process noise condition can be reached as follows (Carlson 2002)

$$ {Q_C}^{-1} = {Q_M}^{-1} + \sum_{i=1}^{n} {Q_{L_i}^-}^{-1} \, . \tag{4.43} $$

Thus, the conservation of information includes the state information and the process noise information.

At this point it is necessary to clarify that only states common between the master and local filter require the conditions in Equations (4.40), (4.42), (4.43). Any states that are particular to the local filters and are not passed or shared between filters can operate in a normal fashion and should not be modified. This ensures that the best possible filtering can occur when sharing information between filters.

4.7.2 No Reset Federated Filter

Figure 15 represents the federated no reset (FNR) filter architecture. This architecture only sends information from the reference system to the local filters. The reference system is the source of observations that are provided to the local filters. The reference system, for example, could be the observations of GPS receiver, the filtered navigation solution of a GPS receiver, or an INS solution. More on the selection of the reference system is discussed in Section 5.4.

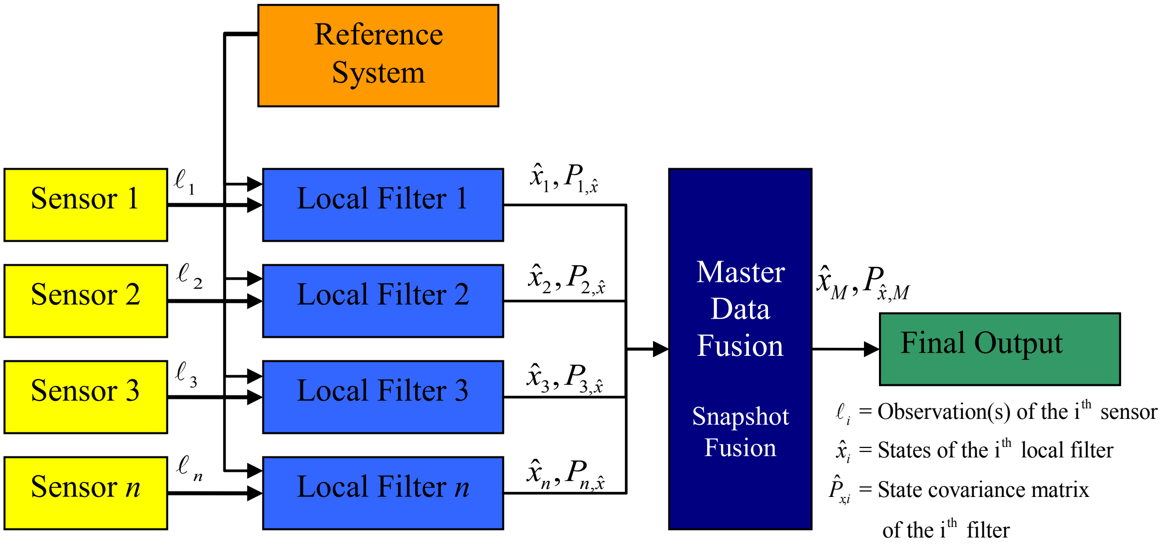

Reference System
Sensor 1
Sensor 2
Sensor 3
Sensor n
ℓ_1
ℓ_2
ℓ_3
ℓ_n
Local Filter 1
Local Filter 2
Local Filter 3
Local Filter n
$\hat{x}_1, P_{1,\hat{x}}$
$\hat{x}_2, P_{2,\hat{x}}$
$\hat{x}_3, P_{3,\hat{x}}$
$\hat{x}_n, P_{n,\hat{x}}$
Master Data Fusion
Snapshot Fusion
$\hat{x}_M, P_{\hat{x},M}$
Final Output
ℓ_i = Observation(s) of the ith sensor
$\hat{x}_i$ = States of the ith local filter
$\hat{P}_{xi}$ = State covariance matrix of the ith filter

Figure 15 – Federated No Reset Filter Architecture

The master filter must be a snapshot fusing algorithm (e.g. least-squares), whereby the input is fused in a single epoch basis (and therefore independent of time). In this manner, the information sharing principle is maintained because no *a priori* information is contained within the master filter and all of the information is stored within the local filters (Carlson 2002). Mathematically this is represented as:

FNR:

$$P_C^{-1} = \sum_{i=1}^{n} P_{L_i}^{-1} \tag{4.44}$$

$$\mathrm{P}_{\mathrm{C}}{}^{-1}\mathbf{x}_{\mathrm{C}} = \sum_{i=1}^{n} \mathrm{P}_{\mathrm{L}_i}{}^{-1}\mathbf{x}_{i} \tag{4.45}$$

$$\mathrm{Q}_{\mathrm{C}}{}^{-1} = \frac{1}{n}\sum_{i=1}^{n} \mathrm{Q}_{\mathrm{L}_i}{}^{-1} \tag{4.46}$$

The FNR architecture is useful in "black box" type navigation systems where access to the local filters is not possible. However, because input into the local filters is the output from the reference system, the output of the local filters is correlated, thus violating the condition of Equation (4.10). Therefore, the weighting of the local filters is skewed and the final master fusion can produce overly optimistic covariance matrices (Groves 2008). It is conceivable that the correlation could be approximated and incorporated into the fusion estimator.

4.7.3 Fusion Reset Federated Filter

The federated fusion reset (FFR) architecture is represented in Figure 16.

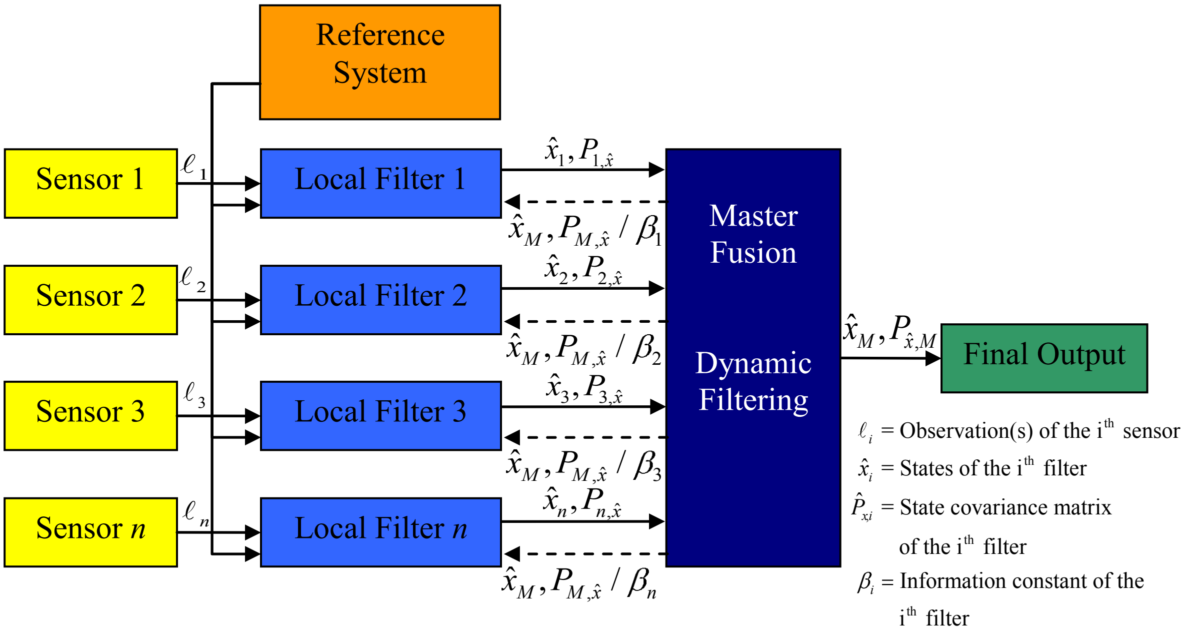

Reference System
Sensor 1
Sensor 2
Sensor 3
Sensor n
ℓ_1
ℓ_2
ℓ_3
ℓ_n
Local Filter 1
Local Filter 2
Local Filter 3
Local Filter n
$\hat{x}_1, P_{1,\hat{x}}$
$\hat{x}_M, P_{M,\hat{x}} / \beta_1$
$\hat{x}_2, P_{2,\hat{x}}$
$\hat{x}_M, P_{M,\hat{x}} / \beta_2$
$\hat{x}_3, P_{3,\hat{x}}$
$\hat{x}_M, P_{M,\hat{x}} / \beta_3$
$\hat{x}_n, P_{n,\hat{x}}$
$\hat{x}_M, P_{M,\hat{x}} / \beta_n$
Master Fusion
Dynamic Filtering
$\hat{x}_M, P_{\hat{x},M}$
Final Output
ℓ_i = Observation(s) of the ith sensor
$\hat{x}_i$ = States of the ith filter
$\hat{P}_{xi}$ = State covariance matrix of the ith filter
β_i = Information constant of the ith filter

Figure 16 – Fusion Reset Federated Filter Architecture

The FFR extends the FNR architecture to include feedback from the master fusion. The states and the covariance from the master fusion replace the matching states and covariance within the local filter. The covariance from the master fusion is scaled by the information factor (i.e. its inverse). In this manner the information is shared from the master fusion to the local filters conserving the information conservation principle. The filter sharing equations can be written as follows:

FFR:

$$\mathrm{P_C}^{-1} = \sum_{\mathrm{i=1}}^{\mathrm{n}} \beta_{\mathrm{i}}\ \mathrm{P_{L_i}}^{-1} \tag{4.47}$$

$$\mathrm{P_C}^{-1}\mathbf{x}_{\mathrm{C}}^{-} = \sum_{\mathrm{i=1}}^{\mathrm{n}} \beta_{\mathrm{i}}\ \mathrm{P_{L_i}}^{-1}\mathbf{x}_{\mathrm{i}} \tag{4.48}$$

$$\mathrm{Q_C}^{-1} = \sum_{\mathrm{i=1}}^{\mathrm{n}} \beta_{\mathrm{i}}\ \mathrm{Q_{L_i}}^{-1} \tag{4.49}$$

The information factor must adhere to the following rule

$$\beta_1 + \beta_2 + \ldots + \beta_{\mathrm{n}} = 1 \tag{4.50}$$

where

β_{i} is the information constant shared between the master filter and the ith local filter.

4.7.4 Zero Reset Federated Filter

Figure 17 represents a Federated Zero Reset (FZR) architecture which periodically resets the covariance of the local filters.

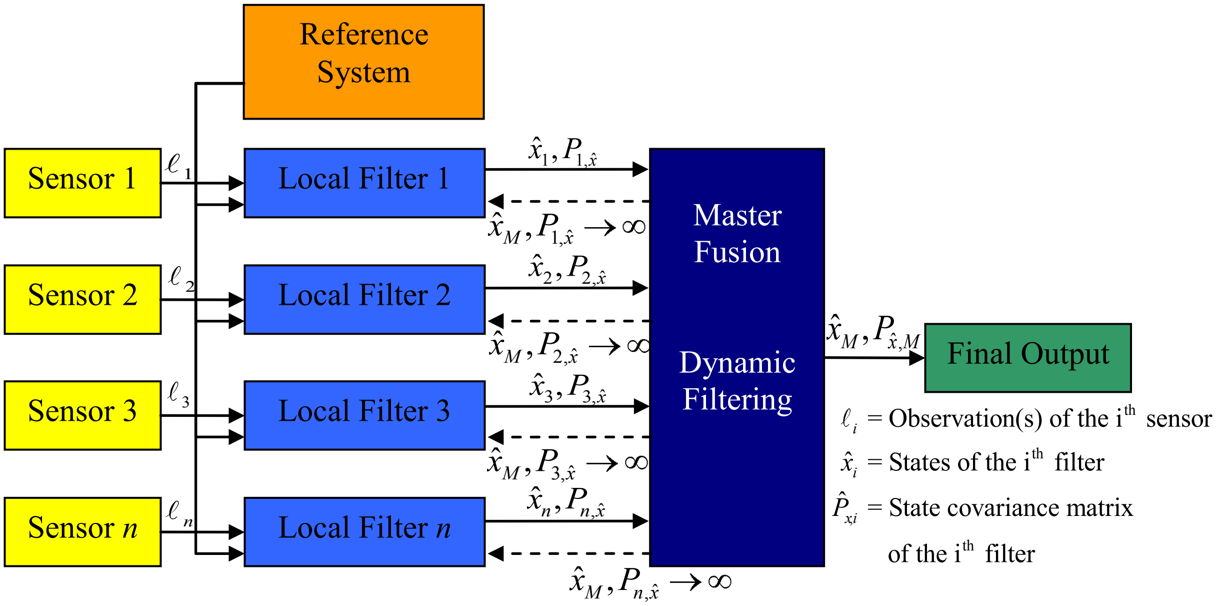

Reference System
Sensor 1
Sensor 2
Sensor 3
Sensor n
ℓ_1
ℓ_2
ℓ_3
ℓ_n
Local Filter 1
Local Filter 2
Local Filter 3
Local Filter n
$\hat{x}_1, P_{1,\hat{x}}$
$\hat{x}_M, P_{1,\hat{x}} \to \infty$
$\hat{x}_2, P_{2,\hat{x}}$
$\hat{x}_M, P_{2,\hat{x}} \to \infty$
$\hat{x}_3, P_{3,\hat{x}}$
$\hat{x}_M, P_{3,\hat{x}} \to \infty$
$\hat{x}_n, P_{n,\hat{x}}$
$\hat{x}_M, P_{n,\hat{x}} \to \infty$
Master Fusion
Dynamic Filtering
$\hat{x}_M, P_{\hat{x},M}$
Final Output
ℓ_i = Observation(s) of the ith sensor
$\hat{x}_i$ = States of the ith filter
$\hat{P}_{xi}$ = State covariance matrix of the ith filter

Figure 17 – Zero Reset Federated Filter Architecture

Zero resets set the information matrices within the local filter to zero, which then eliminates all information in the local filters being repeatedly passed into the master filter. The zero reset is performed at the same interval that information is passed to the master filter. In this manner the local filters perform sub-optimally because the ability to converge is restricted by the master fusion update rate. However, the master filter performs optimally (Carlson & Berarducci 1994). All information is then stored within the master filter. States that are not passed into the master filter are not required to be reset. The sharing information equations are as follows:

FZR:
$$P_C^{-1} = P_M^{-1} \tag{4.51}$$

$$P_C^{-1}\mathbf{x}_C = P_M^{-1}\mathbf{x}_M \tag{4.52}$$

$$Q_C^{-1} = \sum_{i=1}^{n} \beta_i \, Q_{L_i}^{-1} \tag{4.53}$$

4.7.5 Cascaded Federated Filter

The federated cascaded (FC) filter, shown in Figure 18, shows a filter with no resets or information shared and a master filtering algorithm which blindly processes the output. In this architecture, errors from the reference are propagated into each local filter thus correlating the input into the master Kalman filter. This filter architecture must be used with caution because any incorrect tuning within the reference system will provide poor results in all local filters, resulting in overall performance degradation (Brown & Hwang 1997). This architecture is only useful in black-box local filters where limited access is available and filtering at the master fusion level is possible. It is under this consideration that this filter is not analyzed within this research. The sharing information equations can be written as follows:

FC:

$$\mathrm{P_C}^{-1} = \mathrm{P_M}^{-1} + \sum_{i=1}^{n} \mathrm{P_{L_i}}^{-1} \tag{4.54}$$

$$\mathrm{P_C}^{-1}\mathbf{x}_\mathrm{C} = \mathrm{P_M}^{-1}\mathbf{x}_\mathrm{M} + \sum_{i=1}^{n} \mathrm{P_{L_i}}^{-1}\mathbf{x}_\mathrm{i} \tag{4.55}$$

$$\mathrm{Q_C}^{-1} = \mathrm{Q_M}^{-1} + \sum_{i=1}^{n} \mathrm{Q_{L_i}}^{-1} \tag{4.56}$$

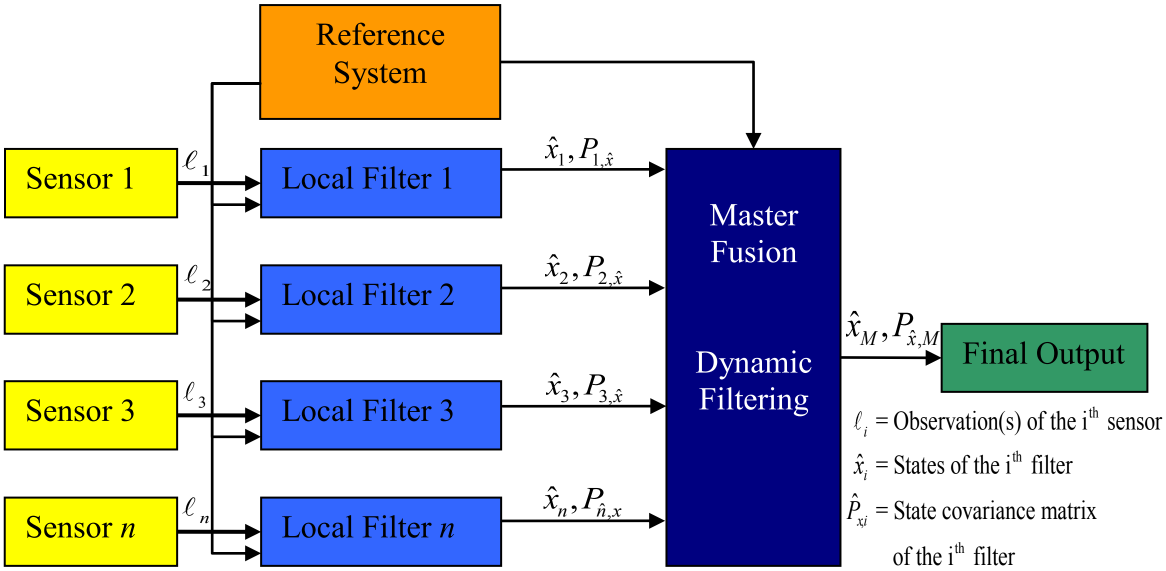

Reference System
Sensor 1
Sensor 2
Sensor 3
Sensor *n*
ℓ_1
ℓ_2
ℓ_3
ℓ_n
Local Filter 1
Local Filter 2
Local Filter 3
Local Filter *n*
$\hat{x}_1, P_{1,\hat{x}}$
$\hat{x}_2, P_{2,\hat{x}}$
$\hat{x}_3, P_{3,\hat{x}}$
$\hat{x}_n, P_{\hat{n},x}$
Master Fusion
Dynamic Filtering
$\hat{x}_M, P_{\hat{x},M}$
Final Output
ℓ_i = Observation(s) of the ith sensor
$\hat{x}_i$ = States of the ith filter
$\hat{P}_{xi}$ = State covariance matrix of the ith filter

Figure 18 – Cascaded Federated Filter Architecture

4.8 Reliability of Observations

Reliability is a term generically used to describe the practice of detecting and rejecting gross errors (also known as faults or blunders) in observations. The process by which this is accomplished is referred to as Fault Detection and Exclusion (FDE). FDE is based on *a priori* knowledge of the statistical behaviour of the observations, whereby any deviation from the observations behaviour can be detected and removed to keep the remaining observations within the assumed statistical model and blunder free. In terms of a filter, removing faults not only improves the reliability, but improves the fusion between the dynamics and observations because the covariance matrix of the observations is more accurately represented. Filtering theory states that if the input noise is zero mean and Gaussian, the innovation sequence will also be zero mean and Gaussian (Gao 2008). This premise forms the basis for hypothesis testing of the innovation sequence. This section will first describe the scenario where any number of faults are possible, and then proceed to a recursive single fault detection algorithm that is able to identify all possible faults in an epoch.

The following equation shows a measurement model with known faults present (∇_k) where faults are biases appearing in the observation vector

$$\mathbf{l}_k = H_k \mathbf{x}_k + M_k \nabla_k + \boldsymbol{\varepsilon}_k \tag{4.57}$$

where

M_k is the blunder mapping matrix and

∇_k vector of known blunders.

The null hypothesis (H_0) assumes the innovation sequence is zero mean and white Gaussian noise, with no faults present. The alternative hypothesis (H_a) is that innovation sequence contains faults. The distributions for the hypothesis testing are therefore

$$\boldsymbol{\upsilon}_k\big|_{H_0} \sim N\ 0, C_{\boldsymbol{\upsilon}_k} \tag{4.58}$$

(4.59)

$$\boldsymbol{\upsilon}_k\big|_{H_a} \sim N\left(M_k \nabla_k, C_{\boldsymbol{\upsilon}_k}\right),$$

where $N(x, y)$ is a Gaussian distribution with x mean and y variance.

The test statistic, as given in Teunissen & Salzmann (1989), is

$$T_k = \boldsymbol{\upsilon}_k^T C_{\boldsymbol{\upsilon}_k}^{-1} M_k \left(M_k^T C_{\boldsymbol{\upsilon}_k}^{-1} M_k\right)^{-1} M_k^T C_{\boldsymbol{\upsilon}_k}^{-1} \boldsymbol{\upsilon}_k \tag{4.60}$$

Assuming any combination of faults implicitly requires that the mapping blunder matrix, M, needs to be constructed for each possible combination of faults. However, the number of fault combinations that can be formed in the M matrix is extensive. If faults occur in any combination of 10 observations, this can produce 1023 potential M matrices ($\sum_{i=1}^{10} {}_{10}C_i$). Thus, it is more practical to recursively assume that one fault occurs in the current set of measurements. By setting the M matrix to contain only a single one (and assuming no correlation within the innovation covariance), the test statistic can be reduced to

$$t_{k_i} = \frac{\left(\boldsymbol{\upsilon}_k\right)_i}{\sqrt{\left(C_{\boldsymbol{\upsilon}_k}^{-1}\right)_{ii}}} \tag{4.61}$$

where

i is the ith index of the innovation sequence.

In this manner the innovation sequence element is normalized by its estimated variance, effectively indicating the probability of this innovation occurring within the Gaussian distribution (commonly referred to as a *z* score). As such, the null and alternative hypothesis simplify to

$$t_{k_i}\big|_{H_0} \sim N(0, 1) \tag{4.62}$$

$$t_{k_i}\big|_{H_a} \sim N(\delta_0, 1) \tag{4.63}$$

where

δ_0 is the non-centrality parameter.

The null hypothesis is accepted if $\left|t_{k_i}\right| < n_{1-\alpha/2}$ and rejected if $\left|t_{k_i}\right| > n_{1-\alpha/2}$. The testing is performed at both ends of a normal distribution (two tailed test), thus the threshold testing value typically associated with a percentage (α).

As mentioned earlier, single fault detection can be applied recursively to detect multiple faults within an epoch. Unfortunately, the pure mathematics of this modification is not entirely correct and there are several assumptions that are associated with this approach (Petovello 2003). For example, when multiple faults exceed the threshold, the largest t_k is rejected and the sequence is estimated again. This appears to be correct, but it is conceivable for one fault to negatively affect another element in the innovation sequence, and falsely identify a fault. This leads to the theory behind statistical reliability.

4.9 Statistical Reliability

The previous section outlined the method of detecting outliers within the innovation vector. Extending the theory to predict the largest fault detectable and its impact on the states is known as statistical reliability. Statistical reliability aims to quantify the probability of false alarm and misdetection. These two probabilities are fundamental in hypothesis testing.

False alarms are Type I errors in hypothesis testing. This percentage indicates how often the system will detect a fault when, in fact, no fault is present. In statistical terms, it the probability of rejecting H_0 when in fact H_0 is true. Misdetections are Type II errors where the system fails to detect a fault when, in fact, a fault occurred. In statistical terms, it is the probability of accepting H_a when H_0 is true. The probability of a Type I error is α and the probability of Type II errors is β. Figure 19 graphically shows the relationship between the two types of errors.

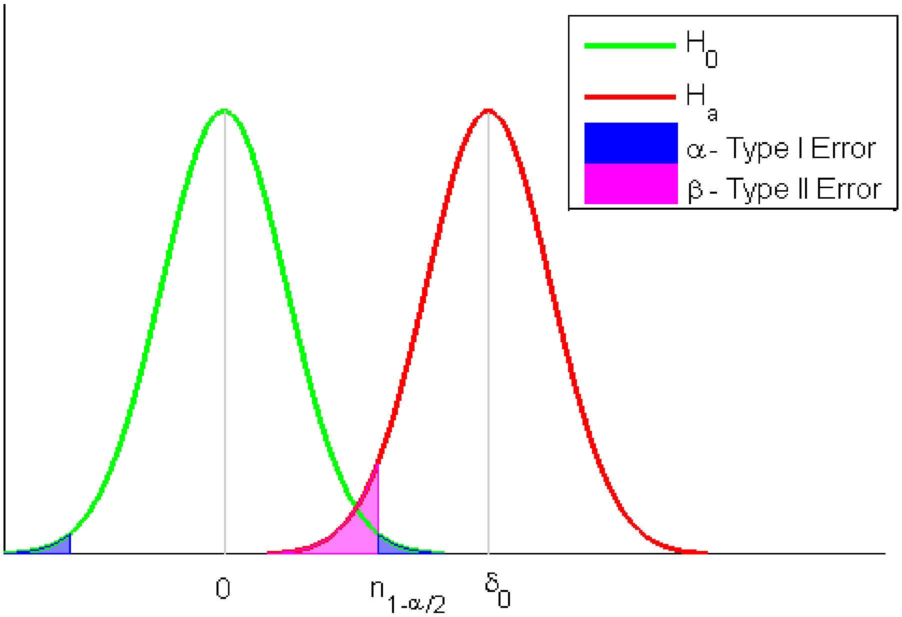


Figure 19 – Probability of Type I and Type II Errors

The non-centrality parameter can be determined from the possibility of accepting type I and type II errors and is given as

$$\delta_0 = \eta_{1-\alpha/2} + \eta_{1-\beta} . \tag{4.64}$$

The non-centrality parameter can be mapped into the measurement domain, if α and β are first set, to provide the smallest possible fault detected, or commonly known as the Minimum Detectable Blunder (MDB) (Petovello 2003)

$$\nabla_i^{MDB} = \frac{\delta_0}{\sqrt{C_{\mathbf{v}_k}^{-1}{}_{ii}}} . \tag{4.65}$$

Once the MDB has been computed, the effect it can have on the estimated states is computed as

$$\delta \mathbf{x}_k = K_k M \nabla_i^{MDB} . \tag{4.66}$$

Chapter Five: Multiple IMU Estimation Architectures

Prior to describing the multi-IMU architectures designed and systematically tested herein, a short introduction of the Single Inertial Navigation System (SINS) architecture is given. Specifics of pedestrian navigation applications where a single IMU located on the foot are first discussed below in order to outline the important factors in processing inertial data.

The sub-sections of the chapter then present the novel theoretical research developed for this thesis. The estimation architectures described in Section 5.2 are more thoroughly developed than previously published, essentially correctly identifying an effective virtual IMU fusion. The validity discussion of FDE testing for VIMUs is also an important contribution. Sections 5.3 and 5.4 are entirely novel to the realm of multi-IMU architectures, specifically the design, implementation and comparative results of each filter have not previously been studied in the literature, to the knowledge of the author. The architectures are presented in this chapter will be field tested in Chapter 6.

5.1 Single Inertial Navigation with IMU on the Foot

The SINS Kalman filter estimates three dimensional accelerometer and gyro biases (6 states), scale factor errors (6 states) and corrections to the expansion point of the position (3 states), velocity (3 states), and Euler angle attitude (3 states). This combination is commonly referred to as the 21 state filter. GPS observations used in this filter (and all multi-IMU filters) are only double differenced pseudoranges. Doppler observations are unusable since the GPS antenna is mounted to the backpack and the IMU is on the foot, which results in a time varying lever arm. However, pseudorange observations have a noise level larger than the time variant portion of the lever arm and are therefore still usable. As this study utilizes double differenced GPS observations, no additional states are required to estimate the GPS receiver clock errors.

The filter operates in a tight integration mode, allowing the system to be updated with less than four satellites, a key performance consideration when poor GPS observability is expected (Knight 1999). Inertial measurements are corrected by their estimated error states, mechanized into the ECEF navigation frame and subsequently used to predict the filter forward in time. The system is updated as GPS observations become available.

Zero-Velocity Updates (ZUPTs) are applied while the foot is at rest during the stance phase. The detection method checks the magnitude of the acceleration and the three sample variances of the accelerometer signal. Several authors have discussed zero velocity detection in detail including Mezentsev (2005), Godha (2006), and Kwakkel (2008). Readers are referred to these publications for more information on the detection process.

5.2 Virtual IMU Architectures

In many cases, such as aviation multi-IMU navigation systems, the purpose of adding additional IMUs to a navigation system is to facilitate IMU fault detection rather than improving accuracy. For pedestrian navigation applications, the opposite is true. Improving accuracy and availability are more important than high levels of reliability, although the latter can also becomes important as soon as accuracy and availability requirements are met. This is most often the case because most pedestrian applications are not required to meet strict safety-of-life standards. Therefore, it will be shown herein that accuracy is improved through the use of a virtual IMU architecture. However, the validity and practicality of FDE may not be acceptable for low cost IMUs and their applications.

As introduced in Section 1.2.1, VIMU is the fusion of raw IMU observations which are rigidly mounted on a body. When IMU observations are simply averaged together (assuming they are already in the same frame), the relative noise reduction of the VIMU data relative to a SINS is readily observed. The standard deviation of the mean is given as

$$\sigma_{\bar{x}} = \sqrt{\sum_{i=1}^{n} \omega_i^2 \sigma_i^2} \tag{5.1}$$

where

ω is the weight associated with each observation set and

n is the number of ensembles (e.g. IMUs).

The equation is further simplified assuming that the ensembles are homogeneous (an acceptable assumption provided all IMUs are of the same make and model) in which case

$$\sigma_{\bar{x}} = \frac{\sigma_x}{\sqrt{n}} \tag{5.2}$$

where

σ_x is the standard deviation of each ensemble.

Thus, as n IMUs are added to a virtual IMU navigation system the noise reduction is reduced by a factor of $n^{-1/2}$. This assumes that a single axis is mapped into a single VIMU axis (and consequently a single IMU is mapped into the VIMU frame). Figure 20 shows the theoretical percentage improvement as a function of the number of IMUs used.

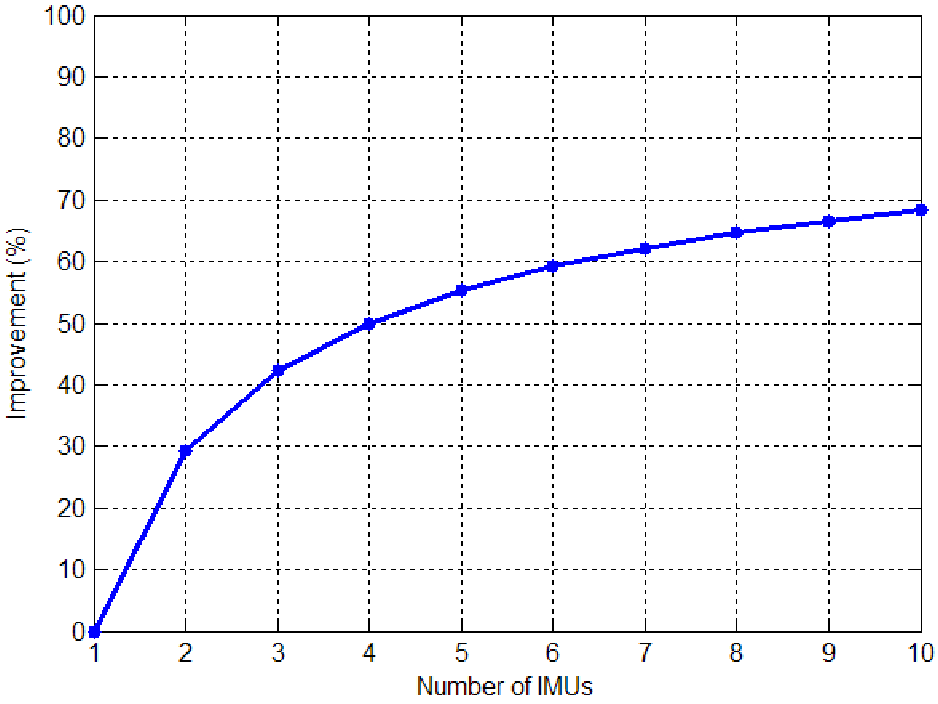


Figure 20 – Noise Reduction of Multiple IMUs (VIMU)

Another benefit of the virtual IMU scenario is a direct real time estimate of the VIMU process noise, as derived from each IMU (Guerrier 2008). This is beneficial when the IMUs have time variant process noise characteristics or filter tuning is not possible for each application or data set.

Averaging of IMUs‟ observations is simple and the least computationally burdensome method of forming a VIMU, however because each IMU is located at a different point on the body, the IMUs measure different specific forces based on the relative location to the VIMU origin. Consequently, the fusion must be performed in the same reference frame and the transformation of each gyro and accelerometer observation set into this frame must be preformed. The transformation is assumed to be known *a priori* from pre-surveyed parameters, namely the vector between the IMUs and VIMU origin and the rotation from one IMU‟s frame to the VIMU‟s frame. From Kane & Levinson (2005), rigid body equations of the angular velocity from a VIMU as follows

$$\boldsymbol{\omega}_{ib_n}^{b_n} = R_v^n \boldsymbol{\omega}_{ib_v}^{b_v} \qquad (5.3)$$

where:

$\boldsymbol{\omega}_{ib_n}^{b_n}$ is the angular velocity of the nth IMU in its body frame,

R_v^n is the rotation matrix from the VIMU body frame to the body frame of the nth IMU (known *a priori*) and

$\boldsymbol{\omega}_{ib_v}^{b_v}$ is the angular velocity of the VIMU in the VIMU body frame.

For the purposes of this thesis, all rotation matrices were determined independently, prior to processing the VIMU data. Further research could determine the IMU and VIMU rotation matrix automatically, without the need for a pre-surveyed calibration, or in situ, where the IMUs are not rigidly mounted.

The specific force, as derived from a VIMU relative to a rigidly attached body, is given in the equation below (Kane & Levinson 2005)

$$\mathbf{f}_{ib_n}^{b_n} = R_v^n \mathbf{f}_{ib}^{b_v} + R_v^n \; \boldsymbol{\alpha}_{ib}^{b_v} \times \mathbf{r}_{nv}^{b_v} \; + R_v^n \; \boldsymbol{\omega}_{ib_v}^{b_v} \times \; \boldsymbol{\omega}_{ib_v}^{b_v} \times \mathbf{r}_{nv}^{b_v} \qquad (5.4)$$

where:

$\mathbf{f}_{ib_n}^{b_n}$ is the specific force vector of the nth IMU,

$\mathbf{f}_{ib}^{b_v}$ is the specific force vector of the virtual IMU,

$\boldsymbol{\alpha}_{ib}^{b_v}$ is the angular acceleration of the VIMU, and

$\mathbf{r}_{nv}^{b_v}$ is the lever arm vector between the nth IMU and VIMU origins within the VIMU body frame.

To the author‟s knowledge, the second and third term on the right hand side of Equation (5.4) have been neglected in previous VIMU systems proposed in the literature. This adjustment to the mapping equation presents an important improvement in accuracy. As will be shown later in this thesis, these terms can be significant.

Equation (5.4) uses the angular acceleration of the virtual frame, which may or may not be output by an IMU. In the event that the angular acceleration is not output by the IMU (as is the case in this thesis), the angular acceleration must be estimated as an additional component of the VIMU fusion procedure in order to correctly determine the specific force. Neglecting this term results in a specific force mapping error related to the lever arm between an IMU and the VIMU origin and the angular acceleration experienced.

When a VIMU fusion is formed, the lever arms and angular accelerations may be assumed to be so small that the correction is negligible. However, when angular accelerations are large, as the case when IMUs are located on a foot, the second and third terms of Equation (5.4) cannot be ignored. Angular accelerations can reach maximum of 900 rad/s^2 in the X axes, and 100-200 rad/s^2 in the Y and Z axes while on the forefoot during regular walking gaits (Kwakkel 2008). Assuming a 7 cm lever arm between the VIMU and the IMU, the corresponding acceleration correction is written as

$$\Delta\mathbf{f}_{\mathrm{ib_n}}^{\mathrm{b_n}} = \boldsymbol{\alpha}_{\mathrm{ib}}^{\mathrm{b_v}} \times \mathbf{r}_{\mathrm{nv}}^{\mathrm{b_v}}$$

$$\begin{bmatrix} -13.1 \\ 31.5 \\ 31.5 \end{bmatrix} \mathrm{m/s^2} = \begin{bmatrix} 800 \\ 200 \\ 100 \end{bmatrix} \mathrm{rad/s^2} \times \begin{bmatrix} -0.035 \\ 0.035 \\ 0.05 \end{bmatrix} \mathrm{m} \, .$$

Figure 21 shows the maximum effect that the angular acceleration can have on the specific force. The maximum error occurs when the lever arm and angular acceleration unit vectors are 90 degrees to each other (i.e. $\boldsymbol{\alpha}_{\mathrm{ib}}^{\mathrm{b_v}} \times \mathbf{r}_{\mathrm{nv}}^{\mathrm{b_v}} = \left\|\boldsymbol{\alpha}_{\mathrm{ib}}^{\mathrm{b_v}}\right\| \left\|\mathbf{r}_{\mathrm{nv}}^{\mathrm{b_v}}\right\| \sin\theta$). Conversely the minimum value (i.e. zero effect) occurs when the two unit vectors are parallel.

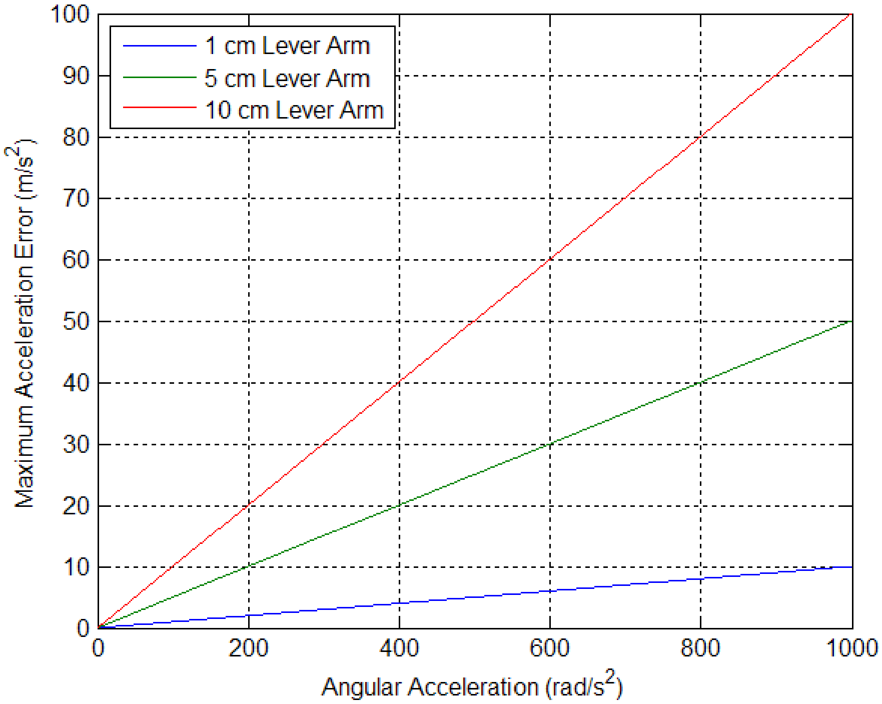


Figure 21 – Maximum Acceleration Errors Due to Angular Acceleration

Figure 22 shows five specific force measurements (rigidly mounted) during the heel lift of the gait cycle starting at time 588.0 s. As the heel lifts the forefoot experiences a high angular acceleration, causing large variations in the magnitude of the specific force. Additionally, the third term of Equation (5.4) will also contribute to the residual error. If the angular acceleration or the third term of Equation (5.4) were neglected, the fusion of the IMU measurements would be compromised.

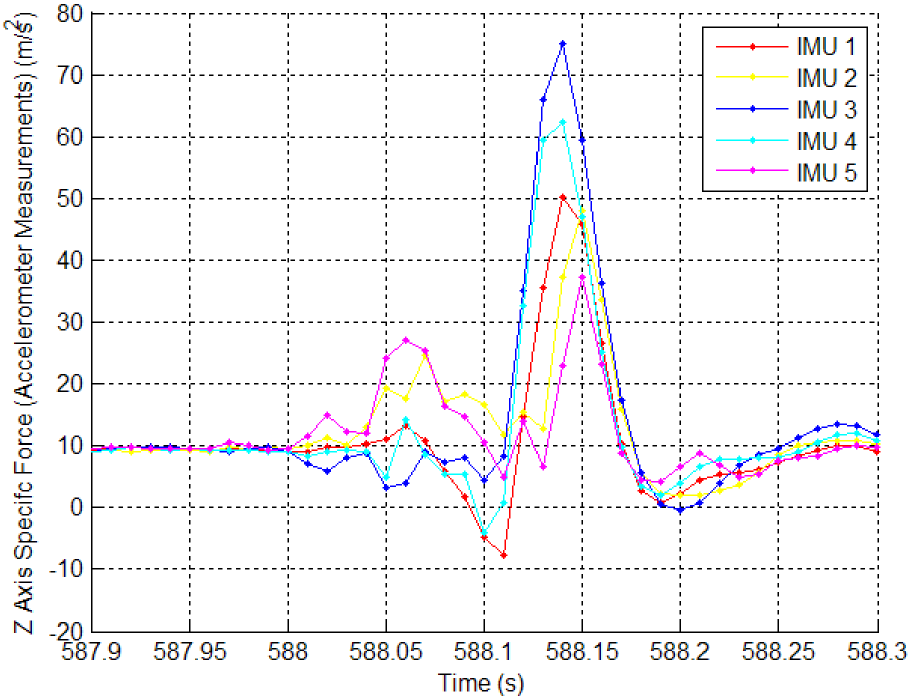


Figure 22 – Specific Force Observations of Five IMUs (IMUs Rigidly Mounted on Foot) during Gait Cycle.

Therefore the nine state estimation model is now described for estimation of angular acceleration, in addition to the angular velocity and specific force.

5.2.1 VIMU – Nine Parameter Least-Squares Estimator

In the VIMU least-squares model, the unknown parameters are the angular velocity, angular acceleration and specific force vectors of the VIMU. As a result of the cross products within Equation (5.4), the 9 state model is non-linear and therefore the system must be linearized. Appendix A provides the derivation of the linearization process. The linearized observation equation, in a similar format to Equation (4.1) is given as

$$\begin{bmatrix} \partial \boldsymbol{\omega}_n \\ \partial \mathbf{f}_n \end{bmatrix} = \begin{bmatrix} R_v^n & 0 & 0 \\ -R_v^n A & R_v^n & -R_v^n\ \mathbf{r}\times \end{bmatrix} \begin{bmatrix} \partial \boldsymbol{\omega}_v \\ \partial \mathbf{f}_v \\ \partial \boldsymbol{\alpha}_v \end{bmatrix}, \ \forall n \in \quad 1,\ldots,N \qquad (5.5)$$

where $A = \boldsymbol{\omega}_v \times \ \mathbf{r}\times + \left[\ \boldsymbol{\omega}_v \times \mathbf{r} \ \times \right]$ and N is the number of IMUs.

The nine parameter least-squares estimation operates in a standard fashion. It uses all gyro and accelerometer measurements as observations and provides an estimation of the virtual IMU accelerometer and gyro measurements. If five IMUs are used, the system has 30 observations and operates at the same frequency as the incoming observations. For this thesis, measurements were weighted equally because the IMUs are all the same make and model.

5.2.1.1 Two IMU Least-Squares Rank Deficiency

This nine-parameter least-squares model shown in Section 5.2.1 has a unique circumstance when two IMUs are used. When using two IMUs, the design matrix will only ever have a maximum rank of 8, indicating that only 8 of the 9 parameters are actually solvable. Further, because the design matrix will only ever have a rank of 8, the rank of the normal equation will only be 8 and therefore the normal equation will not be invertible. It should be noted that the rank of the design matrix for three IMUs is nine and no issues arise estimating all nine parameters, provided no linear relationships exist between IMUs.

Conceptually, the linear dependency arises due to the fact that any angular acceleration about the vector between the two IMUs (i.e. the angular velocity vector and the vector between the IMUs is parallel) will result in zero acceleration. Therefore, all three axis components of the angular acceleration cannot be estimated. As additional IMUs are added the angular acceleration between the two IMUs is observable from other non parallel angular velocities.

To provide a better understanding of the problem the design matrix using two IMUs can be written as

$$H = \begin{bmatrix} R_v^1 & 0 & 0 \\ -R_v^1 \;\; \boldsymbol{\omega}_v \times \;\; \mathbf{r}_1 \times + \left[\; \boldsymbol{\omega}_v \times \mathbf{r}_1 \;\; \times \right] & R_v^1 & -R_v^1 \; \mathbf{r}_1 \times \\ R_v^2 & 0 & 0 \\ -R_v^2 \;\; \boldsymbol{\omega}_v \times \left[\mathbf{r}_2 \times \right] + \left[\; \boldsymbol{\omega}_v \times \mathbf{r}_2 \;\; \times \right] & R_v^2 & -R_v^2 \; \mathbf{r}_2 \times \end{bmatrix} \qquad (5.6)$$

A series of Matlab scripts were written to use symbolic math to compute the rank of the above matrix. This method completely eliminates any singularities arising from linear dependencies resulting from similar rotations to the VIMU frame (R_v^n) and the IMU to VIMU vector ($\mathbf{r}_n$) (which could be generated in columns 4 to 9 of the design matrix). It was determined that the design matrix always had a maximum rank of 8, which proves that the system could not estimate all nine parameters, no matter the values of R_v^n and $\mathbf{r}_n$. It is conceivable that the rank could be lower, again if some linear combination between the columns exist.

To further show which parameter is unsolvable, the design matrix from a data collection described in Chapter 6 was used. These design matrices are time variant and therefore this analysis is only provided to clarify the problem. The design matrix was modified to eliminate one column (i.e. one parameter) at a time, thereby determining which parameter (or parameters) was unsolvable. Table 5 shows results of the test including the number of parameters in the modified least-squares estimation, the condition number of H, the rank of H and whether the normal matrix of H is invertible.

Table 5 – Two IMU Least-Squares Rank Deficiency

	All 9 Parameters Estimated	Angular Velocity (X, Y or Z Axis Removed)	Specific Force (X, Y or Z Axis Removed)	Angular Acceleration (X Axis Removed)	Angular Acceleration (Y Axis Removed)	Angular Acceleration (Z Axis Removed)
Estimated Parameters	9	8	8	8	8	8

Condition of H	1.0e17	2.3e16	2.2e16	58	58	41
Rank	8	7	7	8	8	8
Normal Matrix Invertible	No	No	No	Yes	Yes	Yes

The analysis shows that the system must remove one component of the angular acceleration vector in order to create a solvable system. The specific X, Y or Z component to be removed is not restricted to any specific axis. Using a twin IMU configuration, as per the data collected in Chapter 6, the best results would occur by removing the Z axis of the angular acceleration. Because the modified least-squares version only allows two angular acceleration axes to be estimated, the second term of Equation (5.4) is incomplete and the configuration is unfit for comparison to the other VIMU architectures since it operates in a reduced parametric fashion. The two VIMU is therefore not analyzed within Chapter 6.

5.2.2 VIMU – Nine State Adaptive Kalman Filter

The angular acceleration is the time derivative of the angular velocity and therefore a differential equation exists that relates these states. This forms the basis of a VIMU Kalman filter. A VIMU Kalman filter further reduces noise and can enhance navigation performance. The differential equations of the nine states are as follows:

$$\dot{\boldsymbol{\omega}}_{\mathrm{v}}{}_{ib}^{b} = \boldsymbol{\alpha}_{\mathrm{v}}{}_{ib}^{b} \tag{5.7}$$

$$\dot{\mathbf{f}}_{\mathrm{v}} = \boldsymbol{\eta}_{\mathrm{f}} \tag{5.8}$$

$$\dot{\boldsymbol{\alpha}}_{\mathrm{v}} = \boldsymbol{\eta}_{\alpha} \tag{5.9}$$

where

$\boldsymbol{\eta}_{\mathbf{f}}$ is the process noise of the uncertainty in the time derivative of the specific force vector and

$\boldsymbol{\eta}_{\boldsymbol{\alpha}}$ is the process noise of the uncertainty in the time derivative of the angular acceleration.

The matrix form of the filter, similar to Equation (4.8), is

$$\begin{bmatrix}\dot{\boldsymbol{\omega}}_{\mathrm{v}}\\ \dot{\mathbf{f}}_{\mathrm{v}}\\ \dot{\boldsymbol{\alpha}}_{\mathrm{v}}\end{bmatrix}=\begin{bmatrix}0 & 0 & \mathrm{I}\\ 0 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix}\boldsymbol{\omega}_{\mathrm{v}}\\ \mathbf{f}_{\mathrm{v}}\\ \boldsymbol{\alpha}_{\mathrm{v}}\end{bmatrix}+\begin{bmatrix}0 & 0\\ \mathrm{I} & 0\\ 0 & \mathrm{I}\end{bmatrix}\begin{bmatrix}\boldsymbol{\eta}_{\mathrm{f}}\\ \boldsymbol{\eta}_{\alpha}\end{bmatrix}\tag{5.10}$$

Determining the optimal values for η_{f} and η_{α} is challenging, given the dynamics of the foot throughout the gait cycle. To resolve this issue, an adaptive Kalman filter is used to determine the process noise in real time. A 0.5 s (50 epoch) window is used to determine the process noise and is computed using Equations (4.31) and (4.32). The observation variance, as computed in Equation (4.33), is not used as the input variance, but is held constant to a pre-determined value.

The filter predicts and updates at the same frequency as the incoming measurements (i.e. 100 Hz) which makes this version of the VIMU fusion the most computationally expensive. Updates are performed in an "epoch" mode (all measurements at a given epoch), although it is conceivable to process them sequentially for optimal processing speed.

The VIMU filter must operate with IMUs which are time synchronized. The adaptive Kalman filter could still function if the IMUs are synchronized but have output observations at different data rates or if the observations had different time stamps. The required time synchronization is related to the angular dynamics, specifically the angular acceleration, and will incorrectly determine the specific force at the VIMU location. Given the angular acceleration can reach a magnitude of 40,000 $°/s^2$ for an IMU located on the foot (see Section 6.8.2) and the noise of the gyro observation is 0.7 °/s (as per specification sheet of the IMUs used within this research), a time synchronization between IMU observations of at least 0.02 ms level is required.

5.2.3 Validity of FDE for MEMS Grade VIMU Fusion

This section will demonstrate that FDE is not always a viable option for MEMS IMUs with large biases and scale factors, in particular when IMUs experience large accelerations and angular velocities. Fault detection works on the premise that the misclosure or innovation sequence has zero mean (see Equations (4.62) and (4.63)). As the biases and scale factors of each IMU have not been estimated, and therefore not removed from the observations, the observation model is not zero mean and therefore FDE effectiveness is compromised.

Residuals computed from a 9 state least-squares estimation of each sensor axis are shown below in Figure 23 and Figure 24 (and Appendix B). The period shows a complete gait cycle where all the IMUs were rigidly mounted on the foot. The full details of this data collection are provided in Chapter Six. The residuals are shown with the raw IMU measurements of each sensor in the VIMU frame. The residuals for the accelerometer have a peak magnitude of about 4 m/s^2, which corresponds to the highest acceleration within the gait cycle. Large gyro residuals of nearly 20 °/s are also observed and also correspond to high dynamics. During the stance phase of the gait, the residuals are much smaller, often in the range of the biases. Therefore, the magnitude of the residuals is clearly correlated to high dynamics.

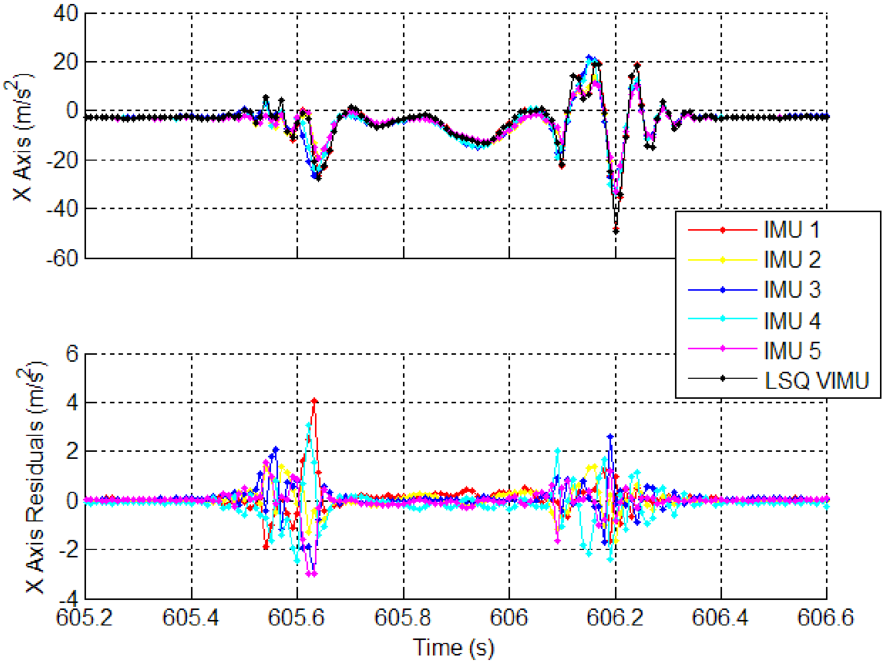


Figure 23 – Specific Force Residuals from a Virtual IMU Computed from Least-Squares (Y and Z Axis are shown in Appendix B)

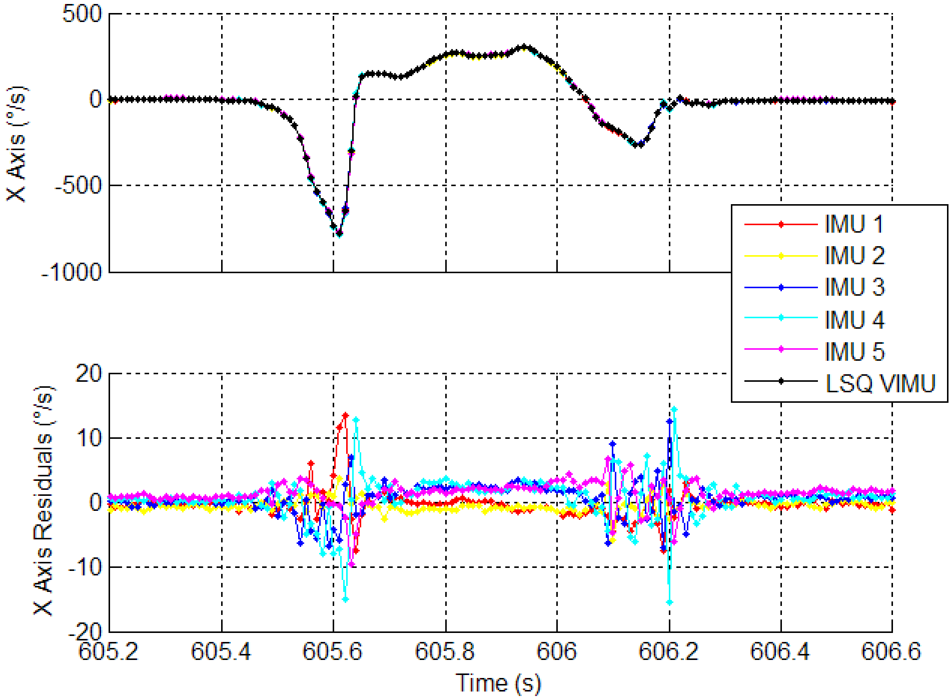


Figure 24 – Angular Velocity Residuals from a Virtual IMU Computed from Least-Squares (Y and Z Axis are shown in Appendix B)

Figure 25 shows the residuals of the Y axis accelerometer as a function of the specific force experienced in the Y axis from a dataset described in Chapter Six. A trend line was fitted to the data, which is shown in red. The general trend of the data indicates that the specific force is linearly related to the residuals. This result shows the impact of the scale factor on the VIMU estimation process. The slope of the red line (0.0339) is the approximate scale factor of the Y axis accelerometer during the test (although it does fluctuate by a few percent). It is also interesting to note that the bias (0.105 m/s^2) is also similar to the bias estimated in the SINS filter (but fluctuates in similar fashion to the scale factor). The two parameters of the trend line serve as confirmation that the estimation is working correctly and that the remaining systematic errors have been identified.

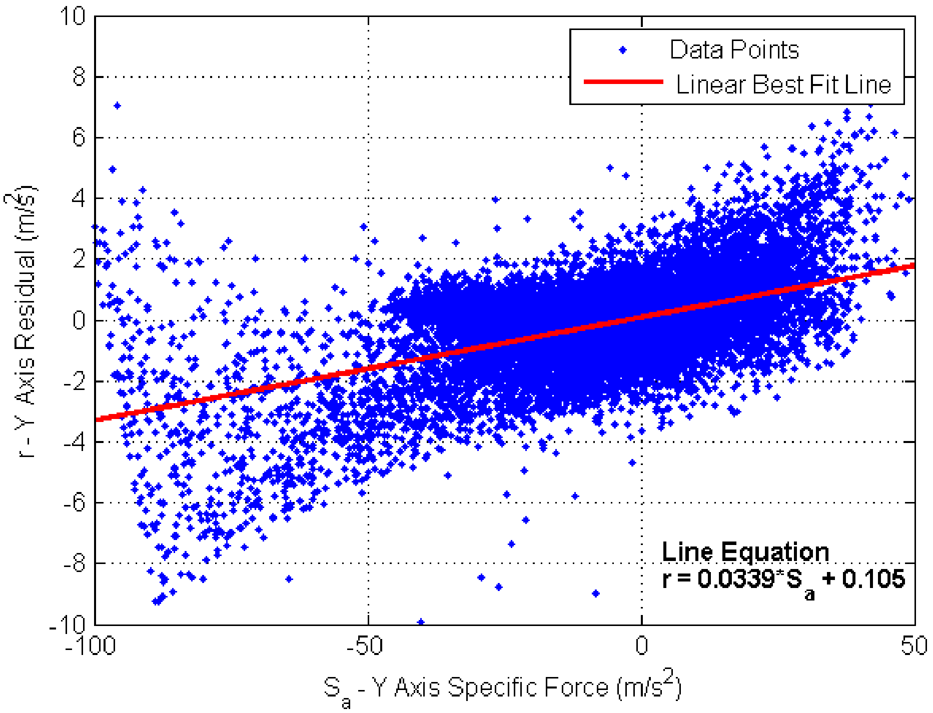


Figure 25 – Y Axis Residuals Vs Y Axis Acceleration (VIMU Frame)

Because the magnitude of the residuals is a function of dynamics rather than sensor error, the input covariance matrix must accommodate these large variations, otherwise faults will be detected during every gait cycle (or whenever the IMU experiences high dynamics). With a VIMU architecture, each IMUs sensor errors cannot be modeled individually. Thus, if FDE was to be performed, the input covariance matrix could not be a function of sensor noise, but rather must contain an increased amount of error to account for biases and the scale factor errors. This increase in error would also be a function of dynamics since the scale factor would create a large bias in the results. Therefore it is a recommendation of this thesis that FDE *not* be performed on MEMS-based VIMU fusion.

5.3 Centralized IMU Estimation Architectures

The centralized filter proposed in this thesis is referred to as a stacked filter, consisting of several individual INS filters. In this manner several "block" filters (i.e. SINS filters) are contained within one centralized filter, ultimately operating as one. This filter architecture has been previously described in Bancroft (2009) and Bancroft et al (2008) and is elaborated upon herein. Specifics are also given concerning a more robust GPS observation FDE algorithm.

The stacked filter contains parameters for position, velocity, attitude, accelerometer and gyro biases and accelerometer and gyro scale factors for each IMU. If five IMUs are used then there are five 21 states filters contained within one centralized 105 state filter. Each block filter can be updated at the same time or individually, but the entire filter prediction cycle must be synchronized (to avoid different block times, within the stacked filter). An attractive characteristic of the stacked filter (and federated filters) is that each block filter could contain additional or different IMU error states, thus facilitating varying types and qualities of IMUs and error state models, which the VIMU architecture does not. Since the IMUs are all the same make and model, the block filters are identical with slightly varied input process noise parameters for each IMU.

The block form of Equation (4.12) and the block form of the stacked filter, similar to Equation (4.1) are as follows

$$\begin{bmatrix} \partial\mathbf{x}_{k+1}^{1} \\ \partial\mathbf{x}_{k+1}^{2} \\ \vdots \\ \partial\mathbf{x}_{k+1}^{n} \end{bmatrix} = \begin{bmatrix} \Phi_{k,k+1}^{1} & 0 & 0 & 0 \\ 0 & \Phi_{k,k+1}^{2} & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & \Phi_{k,k+1}^{n} \end{bmatrix} \begin{bmatrix} \partial\mathbf{x}_{k}^{1} \\ \partial\mathbf{x}_{k}^{2} \\ \vdots \\ \partial\mathbf{x}_{k}^{n} \end{bmatrix} + \begin{bmatrix} \mathbf{w}_{k}^{1} \\ \mathbf{w}_{k}^{2} \\ \vdots \\ \mathbf{w}_{k}^{n} \end{bmatrix} \tag{5.11}$$

$$\begin{bmatrix} \partial\mathbf{z}_{k+1}^{1} \\ \partial\mathbf{z}_{k+1}^{2} \\ \vdots \\ \partial\mathbf{z}_{k+1}^{n} \end{bmatrix} = \begin{bmatrix} H_{k+1}^{1} & 0 & 0 & 0 \\ 0 & H_{k+1}^{2} & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & H_{k+1}^{n} \end{bmatrix} \begin{bmatrix} \partial\mathbf{x}_{k+1}^{1} \\ \partial\mathbf{x}_{k+1}^{2} \\ \vdots \\ \partial\mathbf{x}_{k+1}^{n} \end{bmatrix} + \begin{bmatrix} \boldsymbol{\eta}_{k+1}^{1} \\ \boldsymbol{\eta}_{k+1}^{2} \\ \vdots \\ \boldsymbol{\eta}_{k+1}^{n} \end{bmatrix} \tag{5.12}$$

where:

$\Phi_{k,k+1}^{n}$ is the n^{th} block filter transition matrix,

$\partial \mathbf{x}_{k+1}^{n}$ is the n^{th} block filter states (21 state model),

$\partial \mathbf{z}_{k+1}^{n}$ is the misclosure vector from the n^{th} block filter of the observations,

$\mathbf{w}_{k}^{n}$ is the process driving noise of the n^{th} block filter, and

$\boldsymbol{\eta}_{k+1}^{n}$ is the measurement noise of the n^{th} block filter.

The stacked transition matrix of (5.11) and the design matrix of (5.12) are block diagonal. This important characteristic makes the block filters operate independently, unless additional updates are applied. Thus, if the stacked filter operated without additional updates, the block results would theoretically be identical to the independent IMU filters. In practice however, round off errors and minute computational correlation between block filters result in small differences (i.e. the position varies a few centimetres).

During a GPS update, each block filter requires its own misclosure vector, derived from the GPS observations. However, if each block requires its own misclosure vector, the GPS observations must be repeatedly used for each IMU, thereby directly violating Equation (4.10). The stacked filter innovation vector would have the following form

$$\begin{bmatrix} \boldsymbol{\upsilon}_k^1 \\ \boldsymbol{\upsilon}_k^2 \\ \vdots \\ \boldsymbol{\upsilon}_k^n \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{P}}_k \\ \tilde{\mathbf{P}}_k \\ \vdots \\ \tilde{\mathbf{P}}_k \end{bmatrix} - \begin{bmatrix} h \;\; \mathbf{x}^{(-)1}_{\;\;k} \\ h \;\; \mathbf{x}^{(-)2}_{\;\;k} \\ \vdots \\ h \;\; \mathbf{x}^{(-)n}_{\;\;k} \end{bmatrix} \tag{5.13}$$

where

$h \;\; \mathbf{x}^{(-)n}_{\;\;k}$ is the predicted observation derived from the observation equation using the nth block state vector of the kth epoch.

The transformation from $\tilde{\mathbf{P}}_{\mathbf{k}}$ to $\begin{bmatrix} \tilde{\mathbf{P}}_k & \tilde{\mathbf{P}}_k & \cdots & \tilde{\mathbf{P}}_k \end{bmatrix}^T$ can be performed through a transformation matrix B, $I \quad I \quad \cdots \quad I^{\;T}$, which effectively repeats the original observation vector. Therefore, the covariance transformation of the R matrix is BRB^T, resulting in a new covariance matrix for the repeated observation vector. However, the use of this covariance matrix results in a divergent filter, yielding spurious and incorrect results. A closer analysis of the covariance matrix shows a 100% correlation between observations. Take for example a three observation covariance matrix, repeated two times (three observations feeding two identical states). The observations vector is transformed to

$$\begin{bmatrix} I_{3x3} \\ I_{3x3} \end{bmatrix}_{6x3} \begin{bmatrix} a \\ b \\ c \end{bmatrix}_{3x1} = \begin{bmatrix} a \\ b \\ c \\ a \\ b \\ c \end{bmatrix}_{6x1} \text{, where } a, b, c \text{ are the observations.}$$

The covariance matrix is transformed to

$$\begin{bmatrix} I_{3x3} \\ I_{3x3} \end{bmatrix}_{6x3} \begin{bmatrix} \sigma_a^2 & \sigma_{ab} & \sigma_{ac} \\ \sigma_{ba} & \sigma_b^2 & \sigma_{bc} \\ \sigma_{ca} & \sigma_{cb} & \sigma_c^2 \end{bmatrix}_{3x3} \quad I_{3x3} \quad I_{3x3} \;{}_{3x6} =$$

$$\begin{bmatrix} \begin{bmatrix} \sigma_a^2 & \sigma_{ab} & \sigma_{ac} \\ \sigma_{ba} & \sigma_b^2 & \sigma_{bc} \\ \sigma_{ca} & \sigma_{cb} & \sigma_c^2 \end{bmatrix}_{3x3} & \begin{bmatrix} \sigma_a^2 & \sigma_{ab} & \sigma_{ac} \\ \sigma_{ba} & \sigma_b^2 & \sigma_{bc} \\ \sigma_{ca} & \sigma_{cb} & \sigma_c^2 \end{bmatrix}_{3x3} \\ \begin{bmatrix} \sigma_a^2 & \sigma_{ab} & \sigma_{ac} \\ \sigma_{ba} & \sigma_b^2 & \sigma_{bc} \\ \sigma_{ca} & \sigma_{cb} & \sigma_c^2 \end{bmatrix}_{3x3} & \begin{bmatrix} \sigma_a^2 & \sigma_{ab} & \sigma_{ac} \\ \sigma_{ba} & \sigma_b^2 & \sigma_{bc} \\ \sigma_{ca} & \sigma_{cb} & \sigma_c^2 \end{bmatrix}_{3x3} \end{bmatrix}_{6x6} .$$

The transformed covariance matrix will then have the following correlation coefficient matrix

$$\begin{bmatrix} \begin{bmatrix} 1 & \frac{\sigma_{ab}}{\sigma_a\sigma_b} & \frac{\sigma_{ac}}{\sigma_a\sigma_c} \\ \frac{\sigma_{ba}}{\sigma_b\sigma_a} & 1 & \frac{\sigma_{bc}}{\sigma_b\sigma_c} \\ \frac{\sigma_{ca}}{\sigma_c\sigma_a} & \frac{\sigma_{cb}}{\sigma_c\sigma_b} & 1 \end{bmatrix}_{3x3} & \begin{bmatrix} 1 & \frac{\sigma_{ab}}{\sigma_a\sigma_b} & \frac{\sigma_{ac}}{\sigma_a\sigma_c} \\ \frac{\sigma_{ba}}{\sigma_b\sigma_a} & 1 & \frac{\sigma_{bc}}{\sigma_b\sigma_c} \\ \frac{\sigma_{ca}}{\sigma_c\sigma_a} & \frac{\sigma_{cb}}{\sigma_c\sigma_b} & 1 \end{bmatrix}_{3x3} \\ \begin{bmatrix} 1 & \frac{\sigma_{ab}}{\sigma_a\sigma_b} & \frac{\sigma_{ac}}{\sigma_a\sigma_c} \\ \frac{\sigma_{ba}}{\sigma_b\sigma_a} & 1 & \frac{\sigma_{bc}}{\sigma_b\sigma_c} \\ \frac{\sigma_{ca}}{\sigma_c\sigma_a} & \frac{\sigma_{cb}}{\sigma_c\sigma_b} & 1 \end{bmatrix}_{3x3} & \begin{bmatrix} 1 & \frac{\sigma_{ab}}{\sigma_a\sigma_b} & \frac{\sigma_{ac}}{\sigma_a\sigma_c} \\ \frac{\sigma_{ba}}{\sigma_b\sigma_a} & 1 & \frac{\sigma_{bc}}{\sigma_b\sigma_c} \\ \frac{\sigma_{ca}}{\sigma_c\sigma_a} & \frac{\sigma_{cb}}{\sigma_c\sigma_b} & 1 \end{bmatrix}_{3x3} \end{bmatrix}_{6x6} .$$

This matrix has 100% correlated observations, therefore making the observation set unusable. When data is processed with this covariance matrix, the filter diverges rapidly.

This problem, however, is easily solved by removing the inter block correlation within the covariance matrix, thus assuming that all "repeated" observations are independent (aside from any serial correlation that is introduced from observation differencing). This makes the stacked filter observation covariance matrix block diagonal. In this manner the block filters can act similarly to local filters within a federated filter, which also use the observations repeatedly, albeit in their own INS filters. Information conservation principles apply when the final solution is combined, noting that the outputs of each block filter are highly correlated. The result of the stacked filter is five navigation solutions that are combined via least squares to give the final navigation solution. It is therefore important to observe the correlation between the navigation solutions of each block filter

5.3.1 Stacked Filter Relative Updates

Because the stacked filter contains multiple position, velocity and attitude states, one for each IMU, the filter can be updated with relative PVA information that is known *a priori*.

This section describes the relative updates that can be used as constraints to improve the performance of the filter.

5.3.1.1 Relative Position Update (RPUPT)

If the relative position vector between the IMUs is known, it can be used to update the filter. This information does not update the absolute position of the block filters, but constrains the rate of divergence between the IMUs. It also aids in the estimation of the bias and scale factors of the IMUs. If the IMUs are rigidly mounted with respect to each other, than the vector between them remains constant. If the IMUs are not rigidly held together the vector must be determined from other means in order to utilize the update. Bancroft et al (2008), for example, uses the step length to perform a RPUPT on two IMUs located on each forefoot while Brand & Phillips (2003) use an additional RF ranging technique to perform a RPUPT.

The inter-IMU vector is measured in one of the IMU"s body frame and is computed by differencing the lever arms (i.e. the vector from the GPS antenna to the IMU in the body frame). The relative position update, in the form of Equation (4.1), is given by

$$\begin{bmatrix} \hat{r}_x^1 - \hat{r}_x^2 & -L_x^{2,1} \\ \hat{r}_y^1 - \hat{r}_y^2 & -L_y^{2,1} \\ \hat{r}_z^1 - \hat{r}_z^2 & -L_z^{2,1} \end{bmatrix} = \begin{bmatrix} 1 & & & 0_{1x18} & -1 & & & \cdots_{1x18} \\ & 1 & & 0_{1x18} & & -1 & & \cdots_{1x18} \\ & & 1 & 0_{1x18} & & & -1 & \cdots_{1x18} \end{bmatrix} \delta\mathbf{x} + \boldsymbol{\eta} \tag{5.14}$$

where

$\hat{r}_x^1$ is the estimated X coordinate of the 1st block filter and

$L_x^{2,1}$ is the *a priori* known X component of the vector between the IMUs.

The misclosure is computed by differencing the positions in each block filter and the known vector between the IMUs. It is important to note that by differencing the lever arms to generate the inter-IMU vector, the lever arms must be in the same frame and not the independent body frames. Since the navigation frame for this research is the ECEF frame, the inter-IMU vector must be rotated into that frame. Consequently, there is an inherent relationship between the efficacy of the RPUPT and the error in the orientation of the body frame relative to the ECEF frame.

The update is applied periodically to facilitate a convergence of the block INS filter, reduce numerical computations and limit the inter-block correlation accumulation. Using experimental filter tuning, a periodicity of 6 s and a standard deviation of 1 cm (a diagonal matrix) provided the best performance.

5.3.1.2 Relative Velocity Update (RVUPT)

The relative velocity of a point on a moving rigid body is given by Marion & Thornton (1995). In the context of two rigidly mounted IMUs the relative velocity is expressed as

$$\dot{\mathbf{L}}^{2,1} = \boldsymbol{\omega}_1 \times \mathbf{L}^{2,1} \qquad (5.15)$$

where:

$\dot{\mathbf{L}}^{2,1}$ is the relative velocity between the IMUs 2 and 1,

$\boldsymbol{\omega}_1$ is the angular velocity vector measured by IMU 1, and

$\mathbf{L}^{2,1}$ is the vector between IMUs 1 and 2.

The vector between the IMUs is assumed to be known *a priori* and the angular velocity vector is observed by the inertial unit. This update therefore derives its input from the observation of the IMU. The accuracy is a function of the noise characteristics of the IMU and the filter‟s ability to correctly estimate the systematic IMU errors. The relative velocity update, in the form of Equation (4.1), is given by

$$\begin{bmatrix} \hat{v}_x^1 - \hat{v}_x^2 & -\dot{L}_x^{2,1} \\ \hat{v}_y^1 - \hat{v}_y^2 & -\dot{L}_y^{2,1} \\ \hat{v}_z^1 - \hat{v}_z^2 & -\dot{L}_z^{2,1} \end{bmatrix} = \begin{bmatrix} 0_{1x3} & 1 & & & 0_{1x18} & -1 & & & 0_{1x15} \\ 0_{1x3} & & 1 & & 0_{1x18} & & -1 & & 0_{1x15} \\ 0_{1x3} & & & 1 & 0_{1x18} & & & -1 & 0_{1x15} \end{bmatrix} \delta\mathbf{x} + \boldsymbol{\eta} \qquad (5.16)$$

where

$\hat{v}_x^1$ is the *X* velocity of the 1st block filter and

$\hat{v}_x^2$ is the *X* velocity of the 2nd block filter.

As with the relative position update, the relative velocity observation is derived in the body frame and must be rotated into the navigation frame, thus creating a similar relationship between the error of the rotation and the RVUPT. The standard deviation used for RVUPTs was 2 cm/s and was derived using propagation of variances of Equation (5.15), assuming nominal values of the IMUs noise characteristics and the accuracy of the known lever arm, as discussed in Section 5.3.1.1.

5.3.1.3 Relative Attitude Update (RAUPT)

The relative attitude update follows a similar procedure to the relative position update. The misclosure vector is formed using the difference in estimated Euler angles of each IMU and the pre-surveyed Euler angles describing the rotation between them. In this research the IMUs are fixed on the same platform and mounted on adjacent faces thereby allowing simple Euler angle identification. The relative attitude update, in the form of Equation (4.1), is given as

$$\begin{bmatrix} \theta_{B1}^{B2} \\ \varphi_{B1}^{B2} \\ \psi_{B1}^{B2} \end{bmatrix} = \begin{bmatrix} 0_{1x6} & 1 & & & 0_{1x18} & -1 & & & 0_{1x12} \\ 0_{1x6} & & 1 & & 0_{1x18} & & -1 & & 0_{1x12} \\ 0_{1x6} & & & 1 & 0_{1x18} & & & -1 & 0_{1x12} \end{bmatrix} \delta\mathbf{x} + \boldsymbol{\eta} \tag{5.17}$$

where:

θ_{B1}^{B2} is the misclosure of the roll between the first and second IMU body frames,

φ_{B1}^{B2} is the misclosure of the pitch between the first and second IMU body frames, and

ψ_{B1}^{B2} is the misclosure of the yaw between the first and second IMU body frames.

The standard deviation of this observation is 0.1 rad (i.e. 5.7°).

5.3.2 Stacked Filter Fault Detection and Exclusion of GPS Measurements

Since GPS observations are repeated within the stacked filter, the FDE process is slightly modified for GPS observations. The modification eliminates the possibility that GPS observations may be rejected for one block filter and accepted for another, while at the same time improving the reliability of the fault detection scheme.

Equation (4.57) describes the effect of the blunder vector and its mapping matrix on the observation vector. It is in this equation that the FDE algorithm will be modified to test a series of observations (corresponding to a single GPS measurement) rather than elements of the innovation sequence. The M matrix is generated based on the GPS observations and number of IMUs used. For example, the M matrix with three pseudoranges, repeated for two IMUs, with a single fault in the first element will be $M = 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0^{T}$. The test statistic is then computed from Equation (4.60) with direct reference to the GPS observations.

The test statistic, now different than the single element FDE algorithm presented in Section (4.8), is a chi-squared distribution. The null and alternate hypotheses are

$$\mathrm{T_k}\big|_{\mathrm{H_0}} \sim \chi^2 \ \mathrm{d},0 \tag{5.18}$$

$$\mathrm{T_k}\big|_{\mathrm{H_a}} \sim \chi^2 \ \mathrm{d},\delta_0^2 \ , \tag{5.19}$$

where d is the degrees of freedom (the number of times an observation is used) and δ_0 is the non-centrality parameter. With these hypotheses, the test is conducted by rejecting the null hypothesis if $\mathrm{T_k} \geq \chi_{\alpha}^2 \ \mathrm{d},0$.

The computation of the MDB for the stacked filter is also modified. From Petovello (2003) the non-centrality parameter can be computed as $\delta_0^2 = \nabla_k^T C_{\nabla_k}^{-1} \nabla_k$. Reforming this equation to solve for ∇_k is the premise for determining the MDB provided the non-centrality parameter has been set. Using M, the fault mapping matrix, the non-centrality parameter can be simplified in the following manner, noting that in this particular case, ∇_k, is a scalar

$$\delta_0^2 = \nabla_k^T C_{\nabla_k}^{-1} \nabla_k$$

$$\delta_0^2 = \nabla_k^T \quad M^T C_{\mathbf{v}}^{-1} M \quad \nabla_k$$

$$\delta_0 = \nabla_k \quad M^T C_{\mathbf{v}}^{-1} M \quad ^{1/2} .$$

The MDB of the stacked filter can then be determined as

$$MDB_{\text{Stacked Filter}} = \frac{\delta_0}{M^T C_{\mathbf{v}}^{-1} M \quad ^{1/2}} . \tag{5.20}$$

Assuming that the innovation covariance matrix is equivalent between block filters, the improvement in the MDB versus a SINS MDB is given as follows

$$\begin{aligned} \frac{MDB_{\text{Stacked Filter}}}{MDB_{1\text{ IMU}}} &= \frac{\dfrac{\delta_0}{\sqrt{M^T C_{\mathbf{v}_k}^{-1} M}}}{\dfrac{\delta_0}{\sqrt{C_{\mathbf{v}_k}^{-1} \quad _{ii}}}} \\ &= \frac{\delta_0}{\sqrt{M^T C_{\mathbf{v}_k}^{-1} M}} \frac{\sqrt{C_{\mathbf{v}_k}^{-1} \quad _{ii}}}{\delta_0} \\ &= \frac{\sqrt{C_{\mathbf{v}_k}^{-1} \quad _{ii}}}{\sqrt{M^T C_{\mathbf{v}_k}^{-1} M}} \\ &= \sqrt{C_{\mathbf{v}_k}^{-1} \quad _{ii} \quad M^T C_{\mathbf{v}_k}^{-1} M \quad ^{-1}} \\ &= \sqrt{C_{\mathbf{v}_k}^{-1} \quad _{ii} \quad n C_{\mathbf{v}_k}^{-1} \quad _{ii} \quad ^{-1}} \\ &= \sqrt{\frac{1}{n}}. \end{aligned} \tag{5.21}$$

The performance however will not be exactly as derived above, as a result of inter-block correlation present in the covariance matrix introduced by the relative updates.

5.4 Decentralized IMU Estimation Architectures

Decentralized filtering was discussed in Section 4.6. This section will apply the filter architecture to the case of multiple IMUs and provide the implementation details.

The decentralized filter separates the processing of each IMU into its own INS filter. In the context of this thesis, a decentralized filter that shares the GPS observations will be referred to as a federated filter, as this is more consistent with the nomenclature in the literature.

The federated filters discussed within this chapter contain common states. Specifically, the shared states are position ($\mathbf{r}$), velocity ($\mathbf{v}$) and the Euler angles representing the rotation from the body frame to the ECEF frame ($\boldsymbol{\alpha}$). The local filters estimate these parameters as part of their 21 state filters. The master fusion filter (or least squares estimator as in this case) also contain the same shared states ($\mathbf{r}$, $\mathbf{v}$ and $\boldsymbol{\alpha}$). In this manner, only these states are shared, all biases and scale factors within the local filters remain unmodified.

The reference data of the federated filter can be formed by one of two methods. The first method is to use GPS observations, whereby each local filter operates in a tightly coupled manner (i.e. GPS observations are used in each of the local filters). The second method is to use one of the IMUs to form an INS aided by the GPS observations, the output thereof providing updates to the local filters. In this manner, the federated filter operates in a loosely coupled architecture. If the INS provides the reference to the local filter it provides a time correlated input into the observations of the local filters. This time correlation violates the rules of observation input into a filter and therefore would generate an overly optimistic variance of the states.

It is conceivable that a standalone GPS navigation solution could be used as the reference system, however it has been previously shown that a loosely coupled INS produces suboptimal FDE and overall performance (Godha 2006).

5.4.1 Federated No Reset Filter

The federated no reset (FNR) filter architecture for multiple IMUs is shown in Figure 26. The filter is fundamentally equivalent to running each IMU through an INS filter and combining the final results of each solution via least squares. However, in the software developed for this thesis, each local filter represents one INS and is processed simultaneously. The master fusion is performed via least squares with each local filter‟s PVA providing the observations.

The observation vector for the master fusion is

$$\mathbf{l}_M = \mathbf{r}_1 \quad \mathbf{v}_1 \quad \boldsymbol{\alpha}_1 \quad \cdots \quad \mathbf{r}_n \quad \mathbf{v}_n \quad \boldsymbol{\alpha}_n{}^T \tag{5.22}$$

and the corresponding R_M matrix is given as

$$R_M = \begin{bmatrix} P_{1\ 9x9} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & P_{n\ 9x9} \end{bmatrix}. \tag{5.23}$$

Thus, if there are five IMUs, the master estimator contains 45 observations and correspondingly, a 45x45 observation covariance matrix. The master‟s R_M input observation covariance matrix is block diagonal, however the internal PVA correlation remains within the off diagonal elements (i.e. $P_{n\ 9x9}$ is not diagonal). Because R_M is block diagonal, the least-squares algorithm processes the multiple PVA as independent observations which result in overly optimistic variances for the final states. The PVA of the local filters is in reality correlated as a result of using the same GPS observations and moreover by potentially similar dynamics if the IMUs are rigidly mounted together. Figure 26 shows the flow of information for the multi-IMU FNR filter.

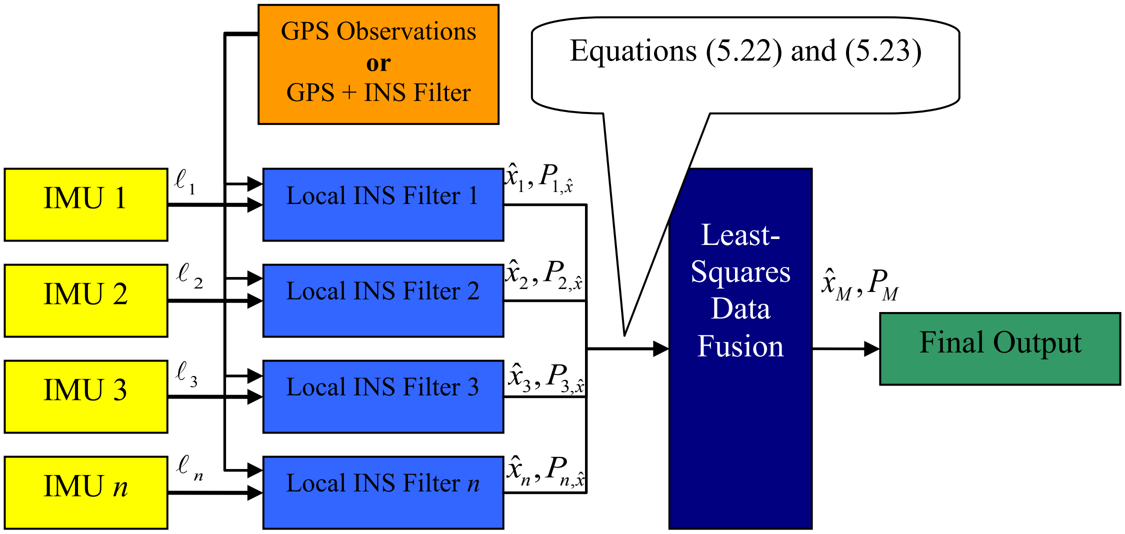


Figure 26 – FNR Multiple IMU Filter

5.4.2 Federated Fusion Reset Filter

The FFR filter has a similar structure to the FNR filter, but the master fusion parameters (and its corresponding covariance matrix) are shared with the local filters. The information factor for each local INS filter (see Section 4.7.3) is n^{-1} because the IMUs are all the same make and model. The input to the master fusion is the same as the FNR filter. Furthermore, since the states of the INS extended Kalman filter are zero, the PVA of the master fusion replaces the PVA used to provide the expansion point, rather than the actual values in the state vector. The covariance information, however, replaces the actual values with the local filters. Additionally, because correlation is developed within the local filter between the PVA and other filter states, these values must be set to zero, otherwise the filter will diverge. Further the covariance replacement of the ith local filter with the master state covariance matrix is as follows, the first nine states representing the PVA having been replaced.

$$P_i = \begin{bmatrix} \frac{1}{\beta_i} P_{m_{9x9}} & 0 \\ 0 & P_{i_{12x12}} \end{bmatrix}. \qquad (5.24)$$

$P_{i_{12x12}}$ remains unmodified during the covariance replacement because it contains the bias and scale factors of the ith IMU which are not shared. Figure 27 shows the data flow of information in the multi-IMU FFR filter.

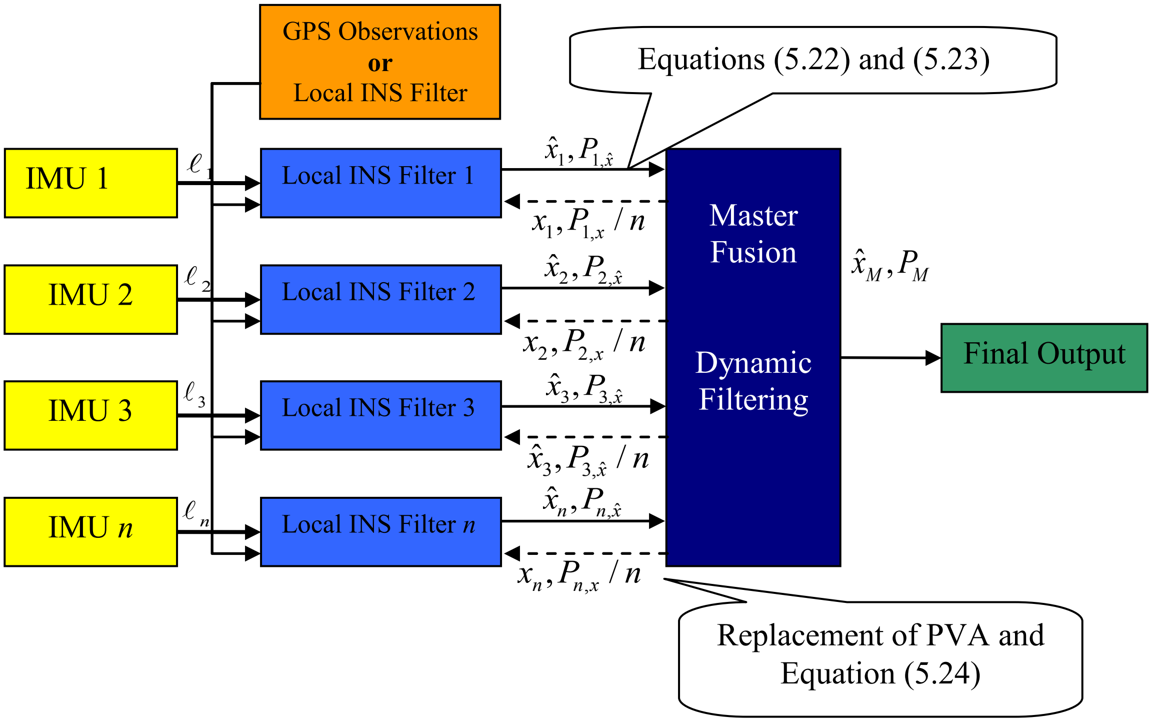


Figure 27 – FFR Multiple IMU Filter

5.4.3 Federated Zero Reset Filter

The FZR filter contains a Kalman filter for the master position and velocity information and a least squares estimator for the attitude information. This varies slightly from the form given in Section 4.7.4, although it still conforms to the federated filtering information sharing principles, essentially combining the general form of FZR and the FFR into one federated filter. The attitude is shared among master and local filters, but since it does not contain a differential relationship within the master filter (as position and velocity do), there is no benefit in including attitude as a parameter in the master Kalman filter. Additionally, since the attitude dynamics are extremely high, the filter would require a high amount of process noise to accommodate the dynamics. Future research could address estimating the attitude derivative within the master filter.

The attitude sharing is performed on a FFR basis, with the position and velocity on a FZR basis. The inputs of the master Kalman filter are the local filter‟s position and velocity and the input into the least squares estimator is the attitude. The output of each estimator is then shared back into the local filters. The position and velocity (PV) covariance is set to a diagonal matrix, i.e. 100^2 m^2 for position and 10^2 m^2/s^2 for velocity; this effectively resets the PV portion of the filter. The attitude variance is shared using the information constant discussed in Section 5.4.2. Figure 28 shows the data flow of information in the multi-IMU FZR filter.

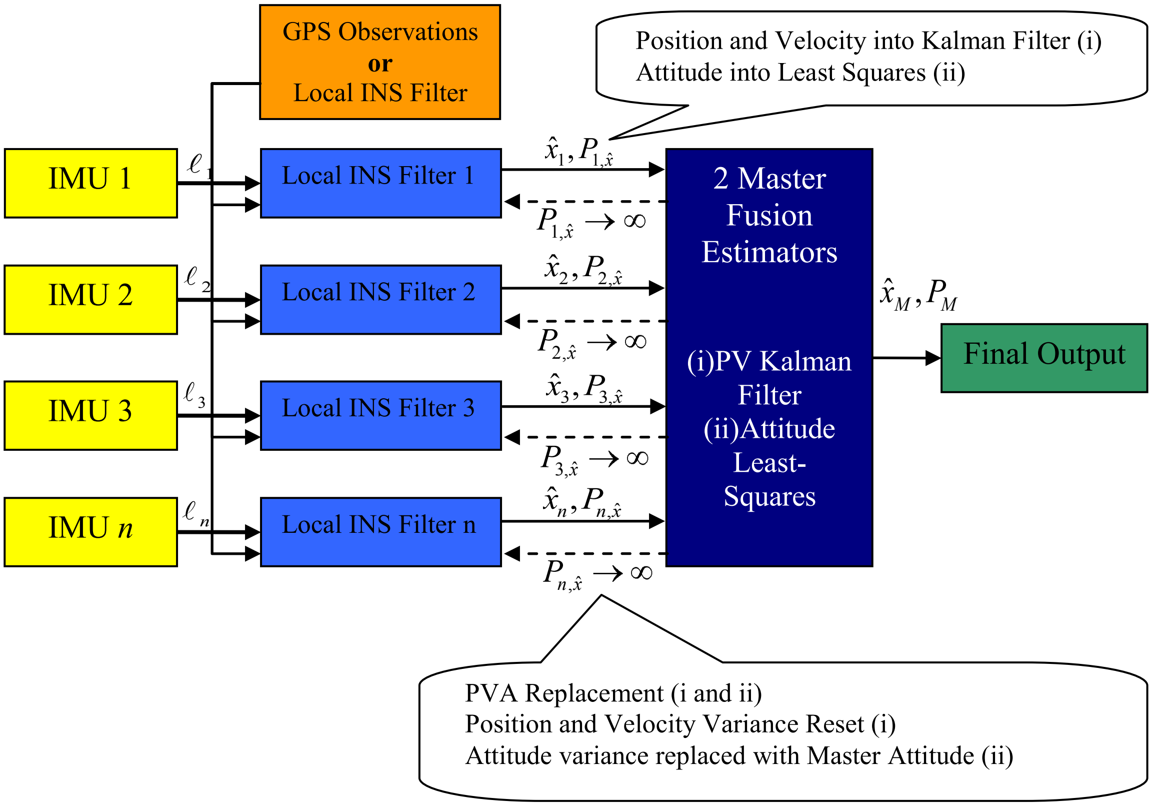

GPS Observations
or
Local INS Filter
Position and Velocity into Kalman Filter (i)
Attitude into Least Squares (ii)
IMU 1
IMU 2
IMU 3
IMU n
ℓ_1
ℓ_2
ℓ_3
ℓ_n
Local INS Filter 1
Local INS Filter 2
Local INS Filter 3
Local INS Filter n
$\hat{x}_1, P_{1,\hat{x}}$
$P_{1,\hat{x}} \to \infty$
$\hat{x}_2, P_{2,\hat{x}}$
$P_{2,\hat{x}} \to \infty$
$\hat{x}_3, P_{3,\hat{x}}$
$P_{3,\hat{x}} \to \infty$
$\hat{x}_n, P_{n,\hat{x}}$
$P_{n,\hat{x}} \to \infty$
2 Master Fusion Estimators
(i)PV Kalman Filter
(ii)Attitude Least-Squares
$\hat{x}_M, P_M$
Final Output
PVA Replacement (i and ii)
Position and Velocity Variance Reset (i)
Attitude variance replaced with Master Attitude (ii)

Figure 28 – FZR Multiple IMU Filter

5.5 Comparison of Architectures

Table 6 shows a comparison of the different architectures described in the chapter and each architecture‟s strengths and weaknesses.

Table 6 – Comparison of the Various Architectures

Filter/Estimation Characteristic	**VIMU**	**Centralized**	**Federated Filter**
Enhanced GPS Observation FDE	No	Yes	No
IMU Observation FDE	Not Recommended	No	No
Reduced Noise at Mechanization Input	Yes	No	No
Constrains Estimator using Relative PVA	No	Yes	No
Estimates Each IMUs Bias and Scale Factor	No	Yes	Yes
IMU Time Synchronization *Not* Required	No	Yes	Yes

In the following chapter these models will be implemented and compared with a focused discussion on the advantages and disadvantages of each. Particular attention will be paid to the navigational accuracy of each model since it has important implications for everyday use.

5.6 Software Implementation

The architectures discussed within this research were implemented in a C++ class based program. The 34 class development was strictly modular, allowing for the repeated use of classes, in particular the IMU fusion.

The user can select any of the architectures from an option file, modify the various parameters each architecture requires, modify the IMU noise characteristics/models and process the data. The program operates in post-mission, but all the algorithms used are capable of operating in real-time.

Several GPS and estimation classes were adapted from the previously written source code within the PLAN Group. These classes were adapted to accommodate where the GPS data could be used, its format and the numerous estimation classes that were required to use the data.

A *SingleIMU* class was created to contain all the information regarding one IMU. Protected member variables included the position, velocity, quaternion (body to ECEF rotation), covariance matrices of the PVA, type of alignment performed, spectral densities of the sensors noise, bias and scale factor models, initial values of errors and their variances, lever arm to the GPS antenna, and numerous file streams for I/O. This class worked in conjunction with the IMUs option file class, where the user could modify any of the initial parameters listed above.

A virtual template class was created to process any type of IMU architecture. Virtual functions were created to force future class implementations to process data in a similar manner within the main function. For example, such functions included loading the next IMU epoch, performing the mechanization of the IMU data, predicting the filter forward, applying a GPS update and detecting/applying ZUPTs. This base class function was called *IMUProcessor*.

Using the *IMUProcessor* template class, the *SingleIMUProcessor, VirtualIMUProcessor, StackedIMUProcessor, and FederatedIMUProcessor* were written and inherited all the public functions of the *IMUProcessor* base class (which only contained public functions). With this organization, the number of private variables within each implementation of the *IMUProcessor* class was minimal. For example, the *SingleIMUProcessor* had one instance of one *SingleIMU* class, an *Estimator* class, a pointer to the *GPSProcessor* class, which was defined in the main function. The *VirtualIMUProcessor* contained an array of *SingleMU* classes and one *SingelIMUProcessor* classes to process the VIMU data. The *FederatedIMUProcessor* had an array of *SingleIMUProcessors* for each local filter, and then had its own master filter variables contained within the protected member variables.

In this manner, new architectures could easily be implemented, rather than modifying the existing code several times for each architecture.

The software developed was not optimized to run in an embedded system, but developed as a tool for evaluating the proposed architectures. The software was continually tested against previously developed single IMU integration packages to ensure that the results were correct.

5.7 Filter Tuning

Tuning the filters developed within this thesis presented a significant (and time consuming) problem. There are five tunable parameters for each sensor (i.e. axis) within the IMU. With a five IMU configuration there are potentially over 150 potential parameters to tune, aside from parameters custom to each architecture (e.g. federated filter sharing information rate). It should be noted that in the VIMU case only one IMU (i.e. the VIMU) requires tuning. For the stacked and federated filters, achieving a high level of tuning for each parameter is simply unrealistic given the quantity. It is conceded that there could be better results with more advanced filter tuning for each architecture. However, it presents similar results that would be seen in industry where each sensor could not be specifically tuned due to the effort required.

Therefore, a generic set of tuning parameters was used during each data set for all IMUs. Only minor modifications to the spectral densities were allowed to accommodate each sensor noise range. Consequently, the same parameters used in the single IMU solution were used in every other multi-IMU solution. Although the solutions may be somewhat sub-optimal, the methodology facilitates better filter performance comparisons, rather than tuning performance comparisons.

Chapter Six: Data Collection and Analysis

Estimation architectures discussed in Chapter 5 were developed in C++ software. Data was collected in typical pedestrian navigation environments (e.g. urban neighbourhoods with a mixture of open and occluded sky) and processed using the software. The results of the different processing architectures are compared within this chapter, accuracy being the primary interest and improvement as a function of IMUs used.

6.1 Data Collection Environments

Data was collected in two environments: a typical North American residential home and inside the Olympic Oval at the University of Calgary. The residential home, as shown in Figure 29, provided an excellent example of an area where GPS was attenuated by 4 to 18 dB and provided reasonable standalone GPS accuracies of a few metres. Although GPS can typically provide reasonable accuracy in such an environment, the benefit of an integrated system to reject multipath is valuable and the ability to position an individual within a room can be of great value to first responders.

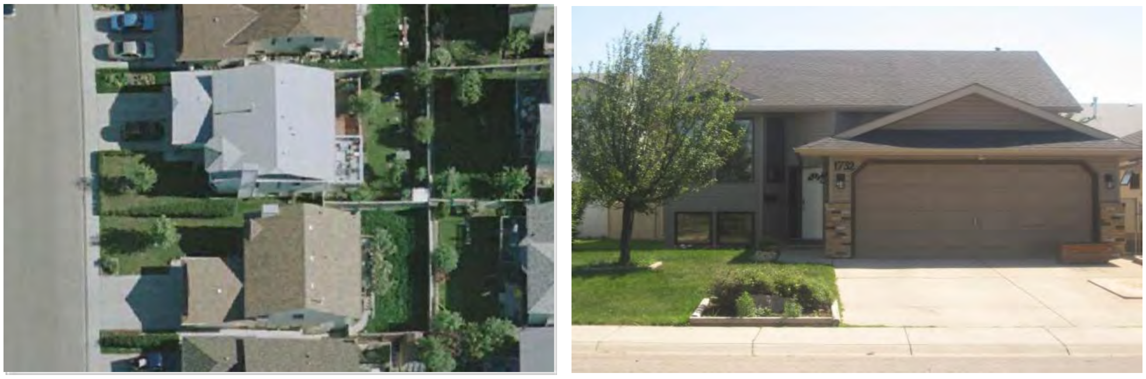


Figure 29 – Residential House used for Data Collection

The Olympic Oval, shown in Figure 30, is an ideal location for indoor testing as GPS signals are attenuated by 25 to 35 dB and are yet observable with high sensitivity receivers. Because of the severe signal attenuation, the effects of multipath and noise are large, often to a point where the GPS solution is completely unreliable and unusable. In this environment there must be an integrated system to provide useful navigation information.

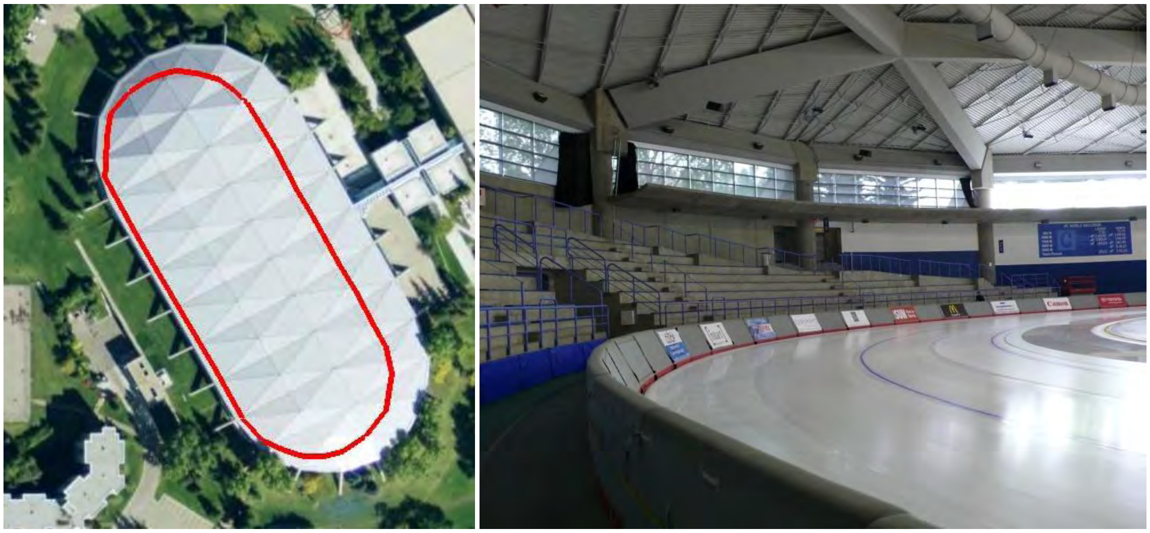


Figure 30 – Olympic Oval (Left: roof top with trajectory in red, Right: inside showing track and ice level)

6.2 Data Collection Set Up

To collect the data, the test subject carried a rigid aluminum backpack to house a reference INS, two laptops to collect the GPS and IMU data, and batteries to power all the equipment.

A NovAtel SPAN system was used to provide the reference solution. It consists of a Honeywell HG1700 AG58 IMU and a NovAtel OEM4 GPS receiver. The receiver and IMU operate in an ultra-tight mode in real time and the GPS and IMU data is logged onboard the OEM4 receiver. The data in this case was differentially post-processed with a nearby (< 1 km) reference station to provide a reference trajectory. The data was processed in NovAtel‟s Inertial Explorer in forward and reverse directions, smoothed using RTS smoothing (Gelb 1974) and then combined for the final reference solution.

The reference solution was accurate to within a few metres in the Oval, and better than 0.5 m while in the residential house.

The GPS receiver used was a u-blox Antaris 4 Precision Timing AEK-4T evaluation kit with firmware 5.0. The antenna was a u-blox ANN-MS, which was designed and manufactured by Allis Communications Co Ltd as antenna M827B (M827B Data Sheet 2006). The antenna was attached to the top of the backpack, rather than the head, to avoid the effects of antenna detuning (Bancroft et al 2010). All GPS data was differentially processed to eliminate the satellite position and clock errors and reduce the effect of the ionosphere and troposphere errors. Differential processing was used to enable a clear analysis of the multi IMU method rather than errors derived from single point (GPS) positioning.

The IMUs used within this research was Cloudcap Technology‟s Crista IMU. The error characteristics of the Crista IMU and the HG1700 AG11-58 tactical are shown in Table 7. The single retail unit cost associated with a tactical grade IMU is typically around $50,000 and for a MEMS grade IMU is $1,500. Figure 31 shows a picture of the IMUs rigidly mounted on a platform attached to the author‟s foot.

Table 7 – Reference and MEMS Grade IMU Maximum Errors

		HG1700 AG11-58 Tactical Grade IMU	Cloudcap Crista MEMS Grade IMU
Accelerometer	In Run Bias (mg)	1	51
	Turn on Bias (mg)	-	30
	Scale Factor (PPM)	300	10,000
	Random Walk (g/√Hz)	2.16×10^{-6}	370×10^{-6}
Gyro	In Run Bias (°/h)	1	2,160
	Turn on Bias (°/h)	-	5,400
	Scale Factor (PPM)	150	10,000
	Random Walk (°/h/√Hz)	7.5	226.8

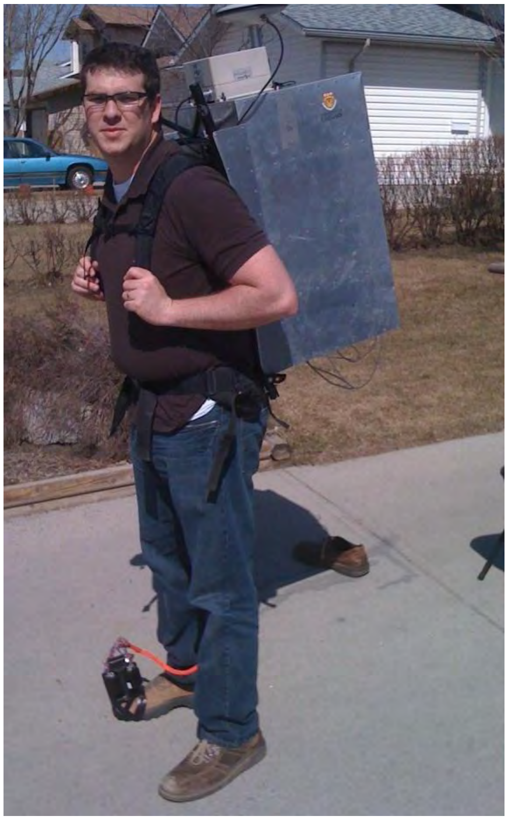


Figure 31 – Rigidly Mounted IMUs on the Foot

6.3 Disjunction Error

Unfortunately, in view of the size and weight of the reference IMU, it had to be housed on top of the backpack. This provides a reference solution for the backpack‟s location and not for the IMU location(s). This makes an exact comparison of the multi-IMU PVA impossible, since the reference solution does not represent the position estimated by the multi-IMU algorithm. This error, referred to as the disjunction error, can however be approximated and is less than the users step length and is negligible compared to the error of the algorithms tested within this thesis.

The IMU was located on the forefoot and the antenna was located on the backpack. The lever arm is the vector from the forefoot to the antenna on the back pack while standing.

As the user walked, the lever arm was subject to a periodic change due to the mechanics of the gait cycle. The most error was accumulated in the longitudinal direction, which varies with half the step length (i.e. the distance from the standing position to the heel strike); a minimal latitudinal error is induced if the user has any abduction/adduction (hip rotation) or varus/valgus (knee rotation); and a minimal vertical effect based on the step height is also induced. Step lengths typically range from small steps of a few centimetres to large steps over a metre (Kwakkel 2008) with step heights typically less than 20 cm. The latitudinal motion was typically less than 10 cm, but varied when the user changed direction or stepped side to side. Each person"s gait cycle exhibits different lever arm errors and contains a variety of aberrations due to the variability in walking mechanics.

Although the lever arm is time variant, the variation is symmetric about the fixed lever arm. It is under this assumption (i.e. the disjunction errors are symmetric about a predetermined lever arm) that solutions can be compared to within a decimetre error envelope.

6.4 Residential Data

During the residential data collection, the subject walked along the street, between the houses, then walked down stairs to the basement, proceeded up the stairs to the main level and outside again. The walk through mimicked a first responder"s walk through of the house. The trajectory is shown in Figure 32. To account for inaccuracies in Google Earth"s geo-referencing, the trajectory was shifted to correctly align with the house.

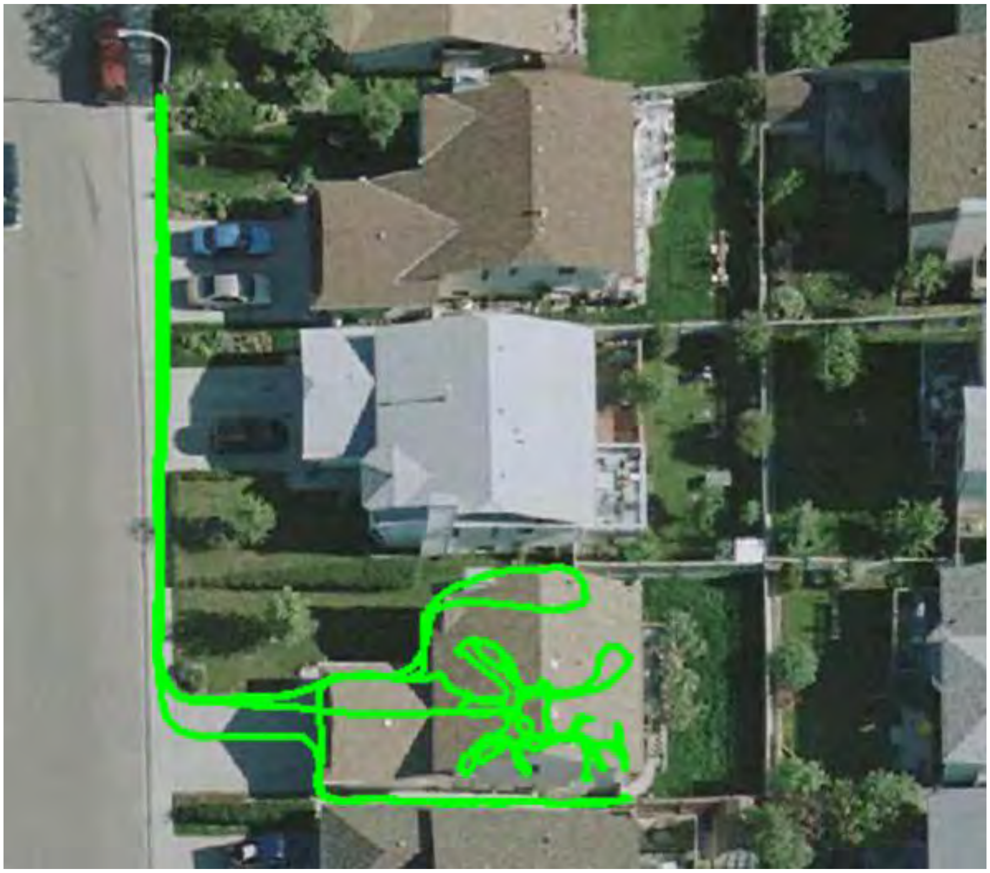


Figure 32 – Truth Trajectory (Residential Data Set)

A 10° satellite elevation mask was used, which generated a Horizontal Dilution of Precision (HDOP) profile shown in Figure 33. This figure also shows the average power of all satellites tracked. While in the basement the average C/N$_o$ dropped below 30 dB-Hz and on the main floor the C/N$_o$ was approximately 37 dB-Hz. Between the houses the C/N$_o$ decreased to 37 dB-Hz. On the sidewalk the C/N$_o$ averaged 43 dB-Hz. The total walkthrough took 7.5 minutes.

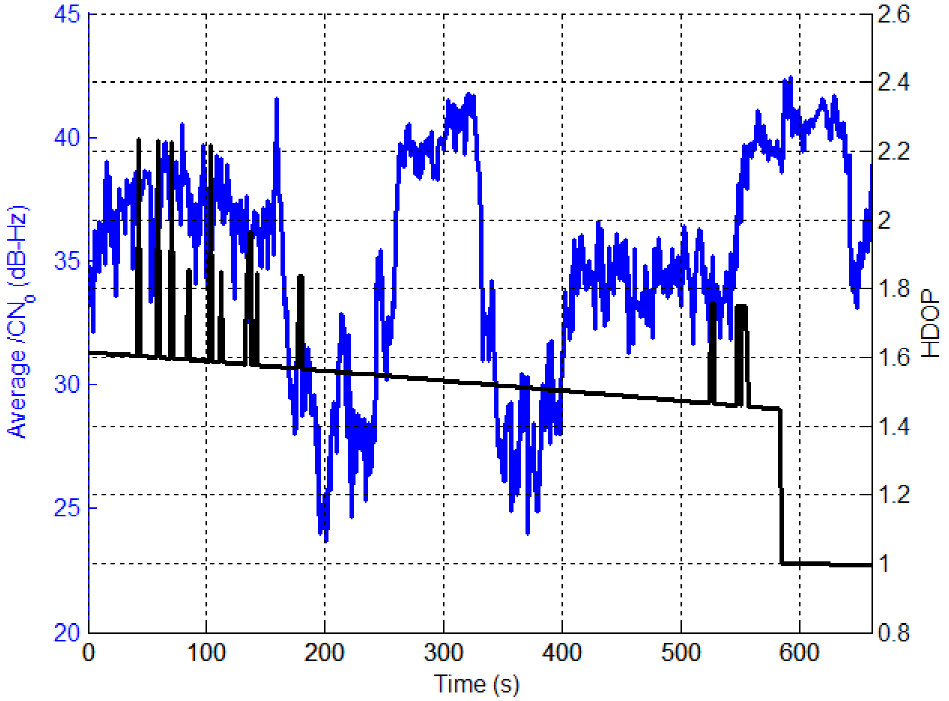


Figure 33 – Average C/N$_0$ and HDOP (Residential Data Set)

6.4.1 Position Accuracy

This section analyzes the accuracy of the architectures discussed in this experiment and Table 8 provides a summary of the statistical values of each. This table allows for the comparison of each estimation technique used. This table is discussed throughout Section 6.4.1.

Table 8 – Horizontal Errors of All Architectures for Data Collected in a Residential House

Processing Method	Number of IMUs	Mean	Mean Improvement	RMS	RMS Improvement	Std	Max	Min	Median
		(m)	(%)	(m)	(%)	(m)	(m)	(m)	(m)
Standalone GPS	0	1.5	0.0	1.88	0.0	1.2	6.1	0.0	1.1
SINS	1	1.1	25.8	1.27	32.6	0.6	3.4	0.0	0.9
	1	1.2	18.0	1.36	27.7	0.6	3.0	0.0	1.2
	1	1.3	14.6	1.47	21.7	0.8	4.2	0.0	1.1
	1	1.3	12.9	1.49	20.9	0.8	4.1	0.0	1.1
	1	1.3	13.9	1.41	25.2	0.6	2.9	0.0	1.2
VIMU (6 State LSQ)	2	1.3	13.1	1.46	22.3	0.7	3.8	0.1	1.2
	3	1.2	15.2	1.44	23.1	0.7	3.9	0.0	1.1
	4	1.2	19.7	1.37	27.1	0.7	3.8	0.0	1.0
	5	1.1	22.8	1.31	30.2	0.7	3.5	0.0	1.0
VIMU (9 State LSQ)	3	1.1	26.6	1.27	32.4	0.7	3.3	0.1	0.9
	4	1.1	27.8	1.25	33.5	0.7	3.3	0.0	0.9
	5	1.0	29.3	1.23	34.7	0.7	3.2	0.0	0.8
VIMU (AKF)	2	1.0	28.5	1.24	34.0	0.7	3.4	0.0	0.9
	3	1.0	29.3	1.24	34.0	0.7	3.6	0.0	0.9
	4	1.0	34.7	1.14	39.6	0.6	3.3	0.0	0.8
	5	0.9	35.7	1.10	41.3	0.6	2.7	0.0	0.9
Stacked Filter	2	1.1	23.8	1.28	31.6	0.6	3.0	0.1	1.0
	3	1.1	22.9	1.32	29.9	0.7	3.3	0.1	1.0
	4	1.1	24.0	1.30	30.9	0.7	3.4	0.0	1.0
	5	1.1	25.8	1.26	33.1	0.6	3.1	0.1	0.9
FNR (GPS)	2	1.1	24.7	1.28	32.0	0.6	3.2	0.0	1.0
	3	1.1	23.3	1.31	30.3	0.7	3.5	0.0	1.0
	4	1.1	22.2	1.33	29.3	0.7	3.7	0.1	1.0
	5	1.1	26.9	1.25	33.5	0.6	3.2	0.0	0.9

Table 8 (Cont'd)

Processing Method	Number of IMUs	Mean	Mean Improvement	RMS	RMS Improvement	Std	Max	Min	Median
		(m)	(%)	(m)	(%)	(m)	(m)	(m)	(m)
FNR (INS)	2	1.2	16.5	1.40	25.6	0.7	3.2	0.1	1.1
	3	1.2	16.8	1.39	25.8	0.7	3.2	0.0	1.1
	4	1.2	17.2	1.39	26.2	0.7	3.2	0.0	1.1
	5	1.2	18.4	1.37	27.2	0.7	3.2	0.0	1.1
FFR (INS)	2	1.2	15.0	1.38	26.7	0.6	3.0	0.1	1.2
	3	1.1	25.6	1.27	32.4	0.6	3.4	0.0	0.9
	4	1.1	25.7	1.27	32.5	0.6	3.4	0.0	0.9
	5	1.1	25.8	1.27	32.6	0.6	3.4	0.0	0.9
FZR (INS)	2	4.3	-195.3	4.96	-164.3	2.4	10.7	0.1	4.1
	3	4.5	-208.9	6.07	-223.0	4.0	19.4	0.1	3.2
	4	5.9	-303.7	7.02	-273.7	3.8	16.2	0.3	4.8
	5	4.9	-235.9	6.05	-222.1	3.5	19.7	0.0	3.9

6.4.1.1 SINS Results

In order to depict the accuracy of the proposed filters in this research, Figure 34 shows a time series of the horizontal errors for the five SINS navigation solutions, where multiple IMU fusion is not used. This figure shows that an accuracy of 1 to 4 metres is achievable with GPS alone, but has a maximum error of 6 m which would not, for example, locate a first responder within a specific room of the house. Figure 35 shows the vertical errors for the same SINS solutions. The vertical error of the standalone GPS solution reaches 8 m, while the SINS solution errors are typically within 4 m of the true elevation. It is also important to note that each SINS solution‟s performance is different, often varying by a few metres in both the horizontal plane and vertical axis. The SINS vertical axis errors, however, tend to follow each other more closely, which results from the ZUPTs applied during the stance phase.

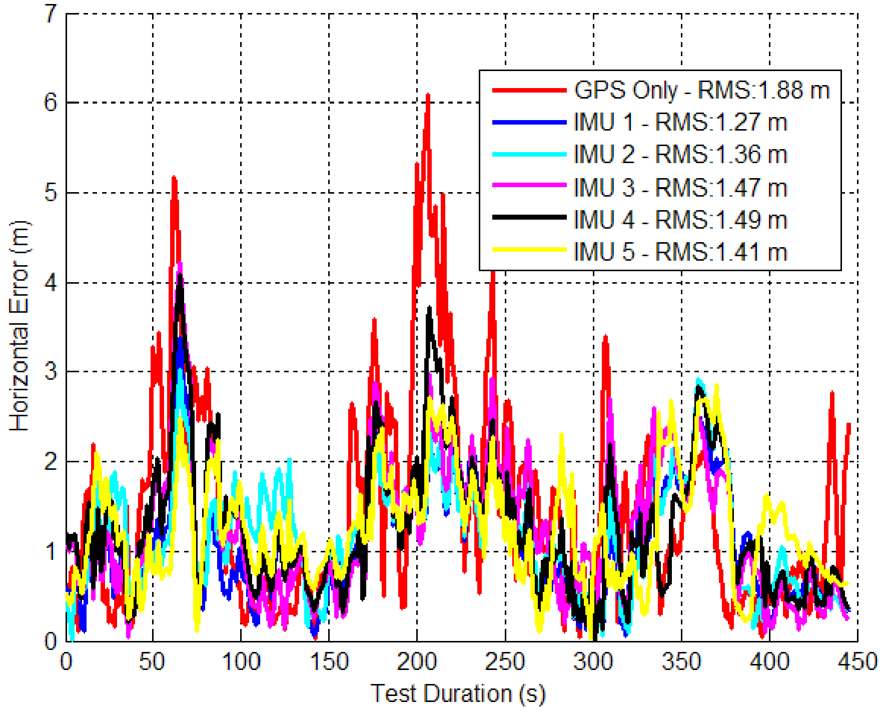


Figure 34 – SINS Horizontal Errors of Five IMUs (Residential Data Set)

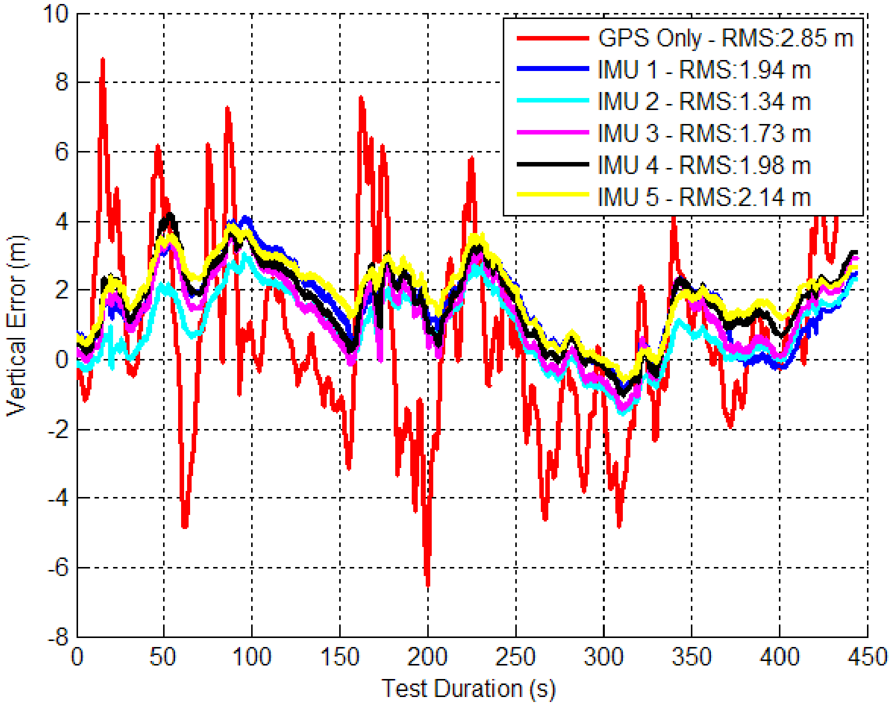


Figure 35 – SINS Vertical Errors of Five IMUs (Residential Data Set)

The Cumulative Distributions (CD) of the SINS horizontal and vertical errors are shown in Figure 36. The CD shows that when a SINS is used there is a significant amount of improvement gained in the vertical channel and a moderate improvement is gained horizontally. The error biases of the SINS solutions" horizontal errors are smaller compared to those of the vertical errors, which shows more deviation resulting from varying levels of performance depending on the common tuning parameters used.

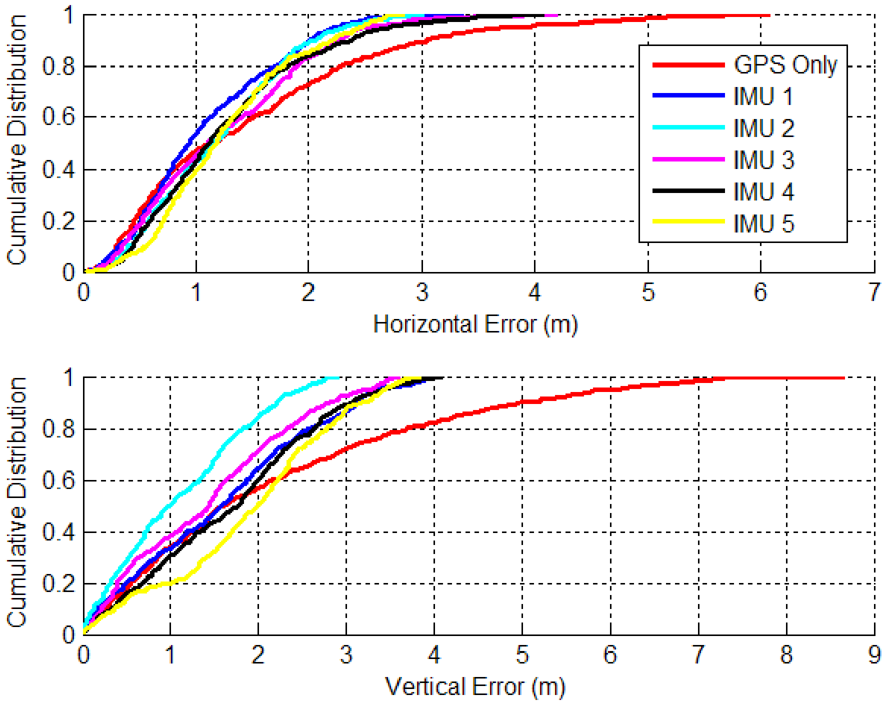


Figure 36 – CDs of SINS Errors of Five IMUs (Residential Data Set)

IMU two was selected as the single IMU solution for comparative results within this section. IMU 2 had an average performing horizontal error amongst all of the IMUs and provided the least amount of vertical RMS error. This IMUs error is shown in the figures within this section as the “Single IMU.”

6.4.1.2 VIMU Results

Figure 37 shows the time series" horizontal error for the three VIMU fusion methods, and the standalone GPS and typical SINS for comparison. The RMS of the horizontal errors is shown in the legend and indicates that moving to the adaptive filter provides a 10.1% and 6.6 % more accurate solution than averaging and the Least-Squares (LSQ) methods, respectively. At time 100 s in Figure 37, the user encounters open sky and the Adaptive Kalman Filter (AKF) quickly accepts the GPS observations, whereas the VIMU and SINS solutions take nearly 35 s longer to converge. When in the basement where standalone GPS has six metre horizontal error, the VIMU filters maintain two metre accuracy whereas the SINS solutions achieve only three to four metre accuracy.

The VIMU solutions contain more noise as a result of the decreased spectral densities used within the filter. This effect was amplified when GPS measurements were stronger (i.e. signal power increased) and the filter weighed the observations more heavily, thus shifting the position. As the filter weighed the GPS measurements less (i.e. when the signal power decreased), the navigation solution displayed a smoother trajectory.

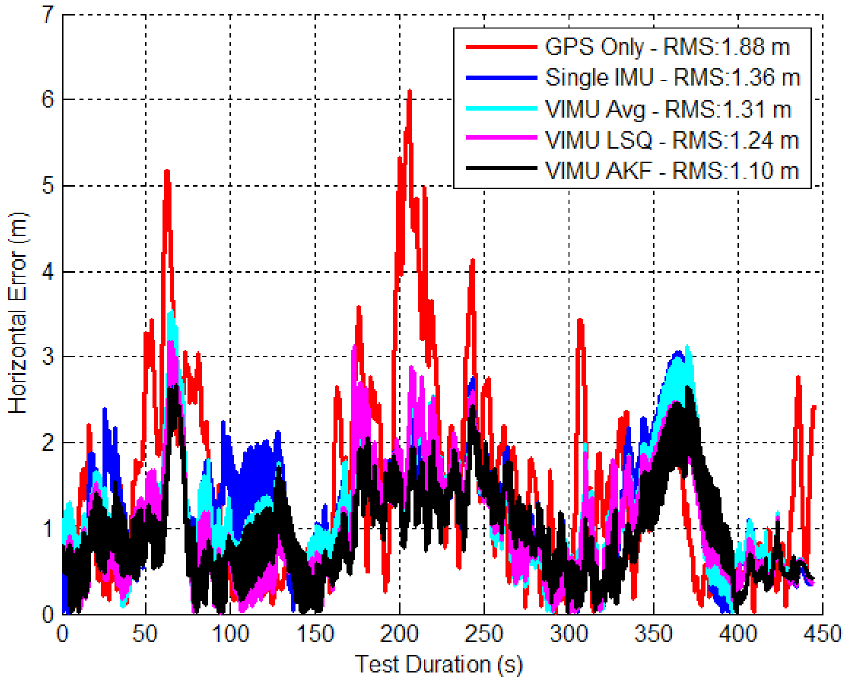


Figure 37 – VIMU Horizontal Errors (5 IMUs Used in Residential Data Set)

The vertical errors alternatively increased with the VIMU architecture as shown in Figure 38. The vertical error of the VIMU AKF architecture was biased by the largest amount, namely a few metres, which is larger than the SINS or the other two VIMU solutions. The VIMU average and VIMU LSQ vertical errors however are only slightly larger than the SINS solution. Again the SINS vertical solution performed exceptionally well. More on this result is discussed in Section 6.8.3.

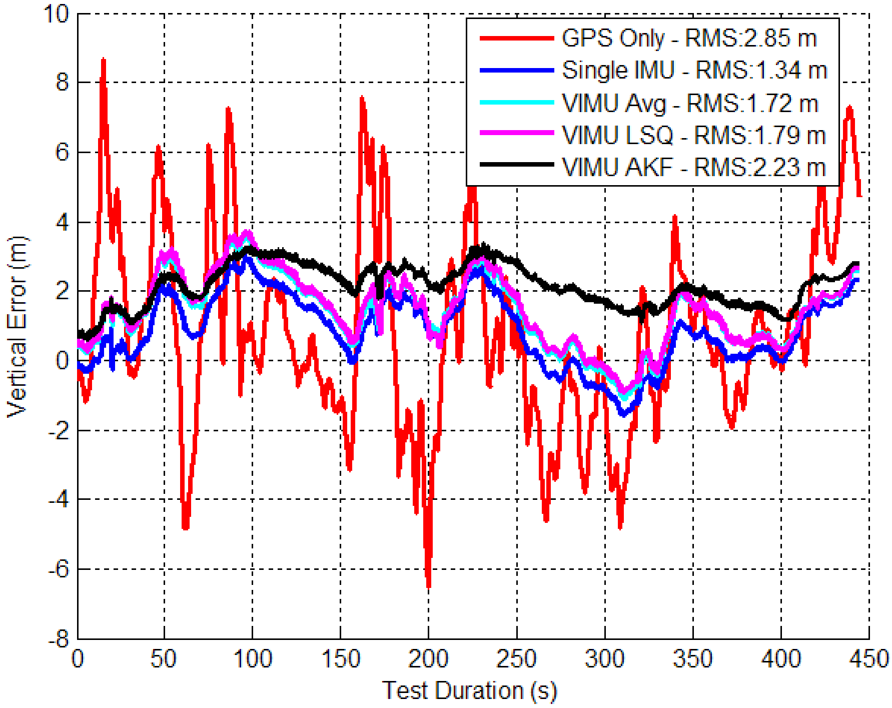


Figure 38 – VIMU Vertical Errors (5 IMUs Used in Residential Data Set)

The CDs of the horizontal and vertical errors are shown in Figure 39. The VIMU AKF performance was best in the horizontal plane and poorest in the vertical axis. In the latter, the VIMUs behaved similarly to the SINS solution, although it was clear that there was no improvement with the VIMU average and VIMU LSQ solutions.

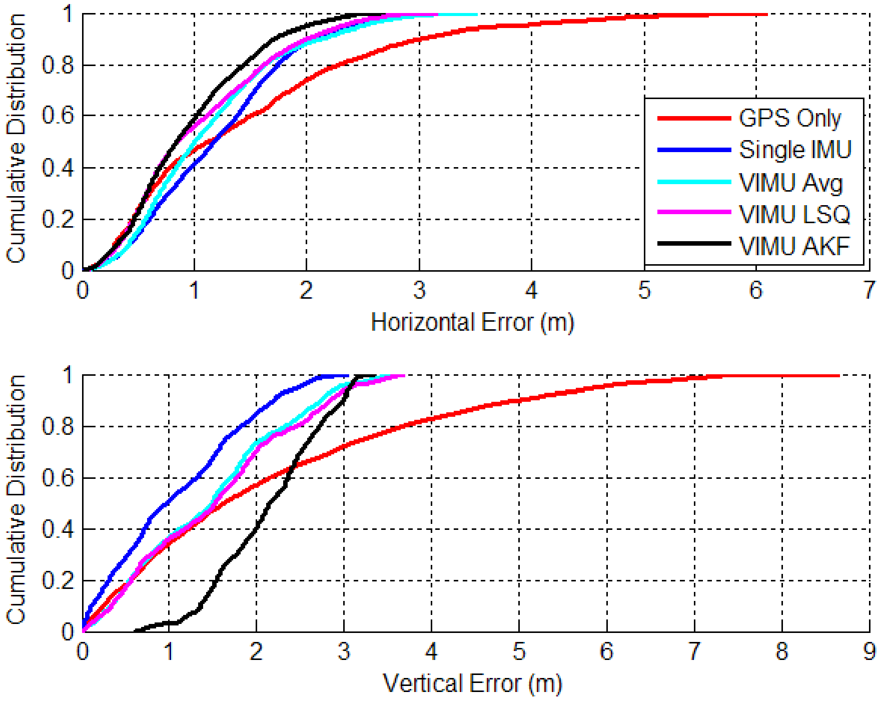


Figure 39 – CD of Horizontal and Vertical Errors (Residential Data Set)

6.4.1.3 Stacked and Federated Filter Accuracy

The stacked filter, FNR and FFR filter‟s horizontal errors are shown in Figure 40. FZR results are discussed in Section 6.7. The FNR (GPS) filter provided the best solution between the stacked and federated filters but only by less than one percent.

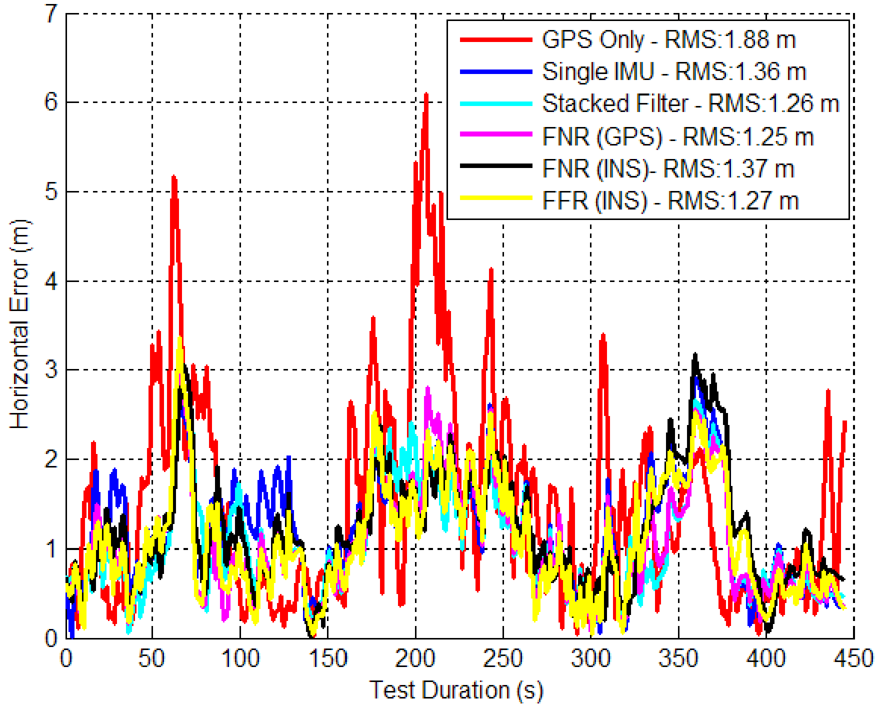


Figure 40 – Stacked and Federated Filter Horizontal Errors (Residential Data Set)

Since GPS signal strength is still reasonable in this environment, the additional information contained within the relative updates did not further improve the accuracy of the final solution. This indicates that the filter‟s biases and scale factors had been resolved and other un-modelled error sources begin to dominate the solution‟s accuracy. The FNR (INS) performed 6.3% worse than the FNR (GPS), which indicates that using the raw ranges of the GPS receiver as input to each local filter is superior.

The vertical errors of the stacked and federated filters are shown in Figure 41. The stacked filter performed the best with a 1.65 m RMS error, but did not outperform the SINS solution. The FNR (GPS) filter provided the second best solution with a 1.86 m RMS error. These errors are consistent with the errors experienced with the SINS cases albeit higher than the specific SINS solution chosen for comparison.

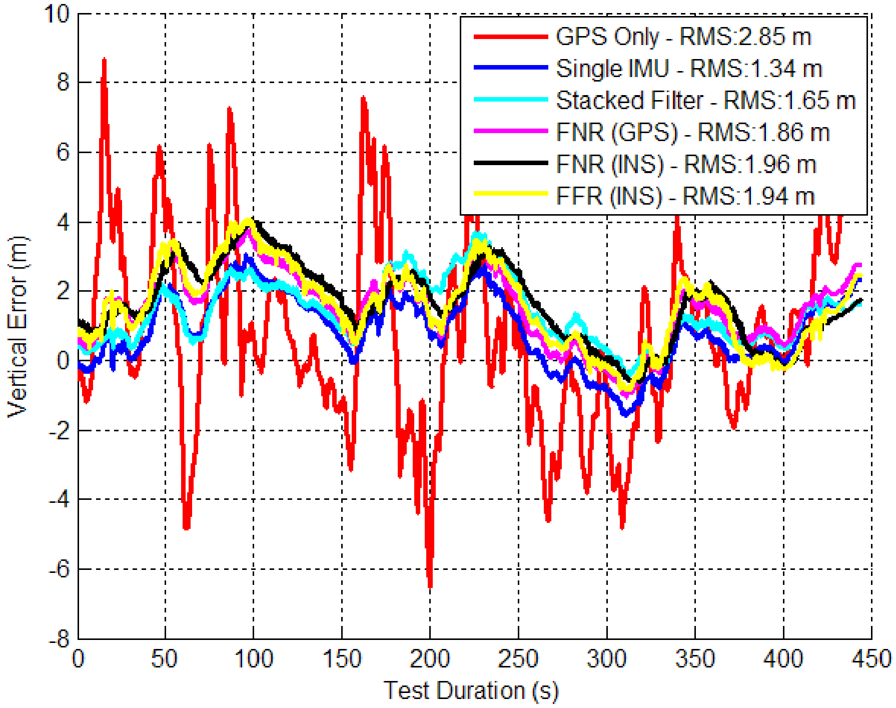


Figure 41 – Stacked and Federated Filter Vertical Errors (Residential Data Set)

Figure 42 shows the CDs of the horizontal and vertical errors. The horizontal distributions have a slightly improved performance with more accurate results below the 1 m level. For example, the SINS filter solution is better than 1 m 38.5% of the time, whereas the stacked filter had 58.3 % and the FNR (GPS) filter had 56.5 %. In the vertical channel the stacked filter had the best CD with 41.5 % less than 1 m error compared to the FNR (GPS) at 33.9 % less.

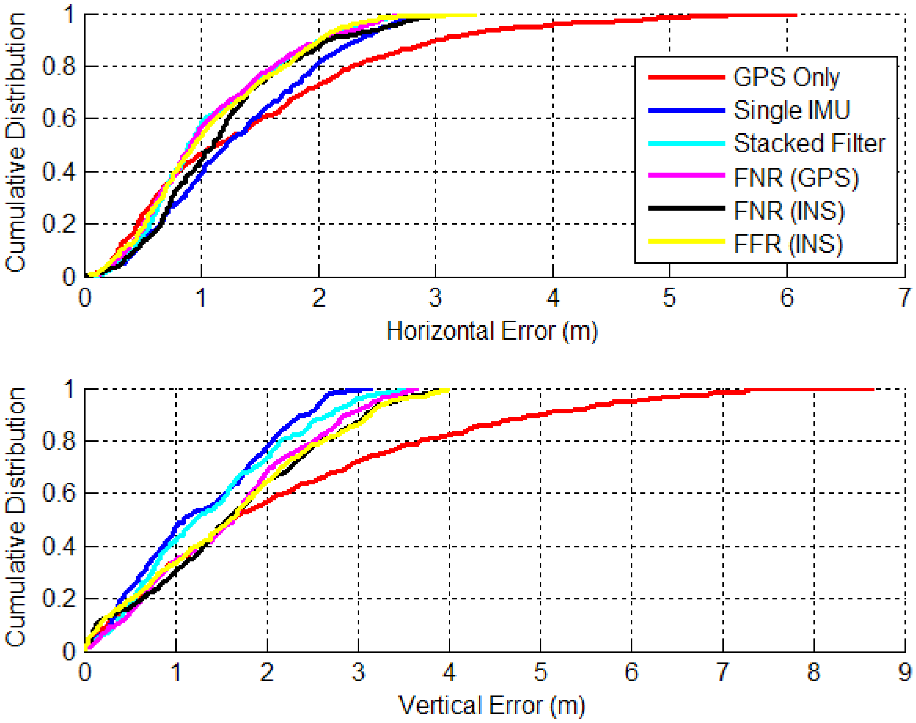


Figure 42 – CD of Horizontal and Vertical Errors for Stacked and Federated (Residential Data Set)

6.4.2 Filters Position Accuracy vs. Number of IMUs

Of particular interest to this research is the cost vs. benefit of adding IMUs. This section addresses this question with respect to the architecture used.

6.4.2.1 VIMU Accuracy vs. Number of IMUs

Figure 43 shows the VIMU architectures RMS percent improvement relative to that of a standalone GPS solution. In all cases, the accuracy improved with each additional IMU. The AKF method had the largest increase when a second IMU was added, although this dramatic increase was not maintained with the addition of the third, fourth and fifth IMU. This is a direct result of estimating the angular acceleration within its filter. Interestingly, applying the averaging technique with five IMUs was less accurate than with two IMUs using the LSQ or AKF method. This confirms that estimating the angular acceleration had a positive impact on the accuracy of the navigation solution, even more so than the number of IMUs used. This was an important practical finding, which makes the use of a dual inertial system considerably more attractive.

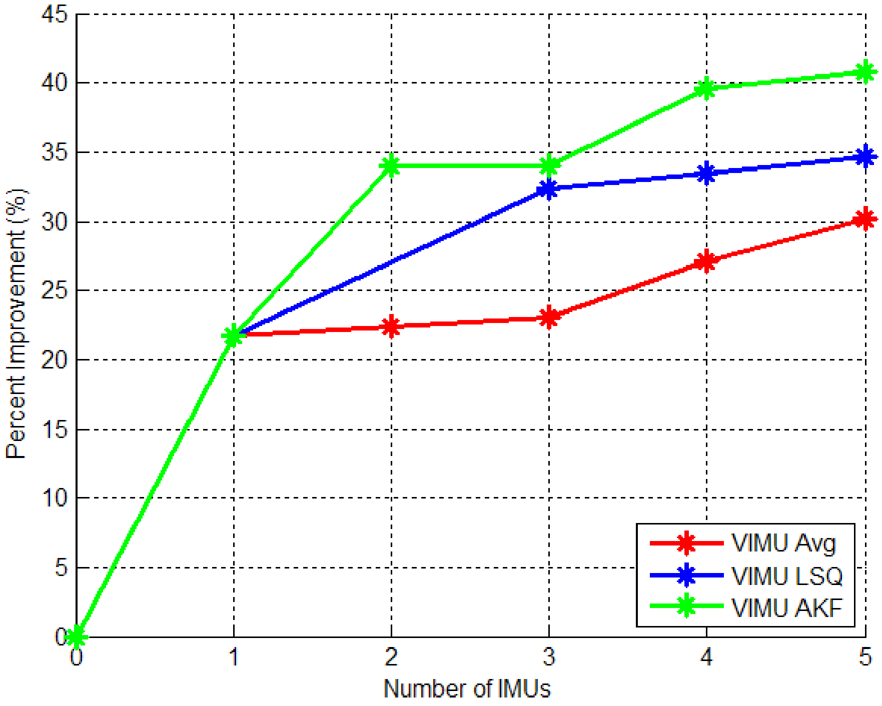


Figure 43 – VIMU Accuracy as a Function of IMUs Used (Residential Data Set)

6.4.2.2 Stacked and Federated Filter Accuracy vs. Number of IMUs

Figure 44 shows the accuracy of the navigation solution as a function of the number of IMUs used for the stacked and federated filters. The stacked filter showed the largest percent increase with two IMUs, but then decreased with the addition of the third and fourth IMU. The third and fourth IMUs were among the least accurate SINS solutions. Thus, when the filter combined the block filter solutions, the final solution was degraded. This contradicts the hypothesis that the relative updates would have provided additional information to improve the accuracy of each block filter. This contradiction is refuted with the data set from the Olympic Oval, which shows that in the absence of reasonable GPS observability, the relative updates significantly improve the navigation solution.

The FNR (GPS) results followed a similar trend to that of the block filter, again suggesting that the relative updates were providing little improvement to navigation solution in this case. The FFR (INS) filter performance plateaued at the third IMU and had similar results with three to five IMUs, only increasing 0.1 % per additional IMU. The FNR (INS) percent improvement was minute with only 0.3, 0.4 and 1.2 % for each additional IMU.

Consistent with the results of the VIMU architecture in Section 6.4.2.1, the addition of the second IMU had the largest percentage increase, even more so than the third, fourth or fifth IMU. This suggests that if two IMUs are used, the stacked, FNR (GPS) or VIMU AKF all show similar performance. However, when using more than two IMUs, the solution accuracy improves at a lower rate.

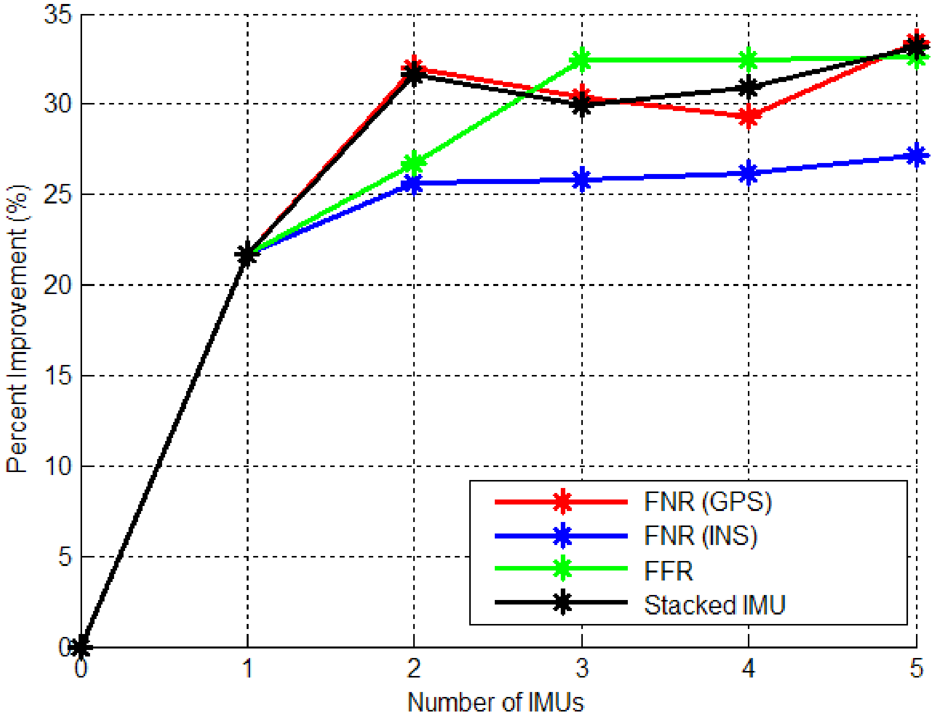


Figure 44 – Stacked and Federated Filter Accuracy as a Function of IMUs Used (Residential Data Set)

6.4.3 Minimum Detectable Blunder (MDB)

The MDB for PRN 22 is shown in Figure 45 when five IMUs were used. PRN 22 had an elevation angle of 22° and average power of 38 dB-Hz. The MDB was 0.5 m lower in the SINS solution when compared to a standalone GPS solution. The VIMU MDB was nearly identical to the SINS, decreasing only by a few centimetres. Thus, the actual improvement of the MDB with the use of the VIMU is negligible, an expected result considering that the improvement of the IMU provided no absolute positioning information. The marginal decrease would also follow suit for the federated filters where GPS observations were only tested with the information within the local filters. In the case where the federated filter used a local filter as the reference (e.g. FNR (INS)), the MDB was identical to the SINS MDB.

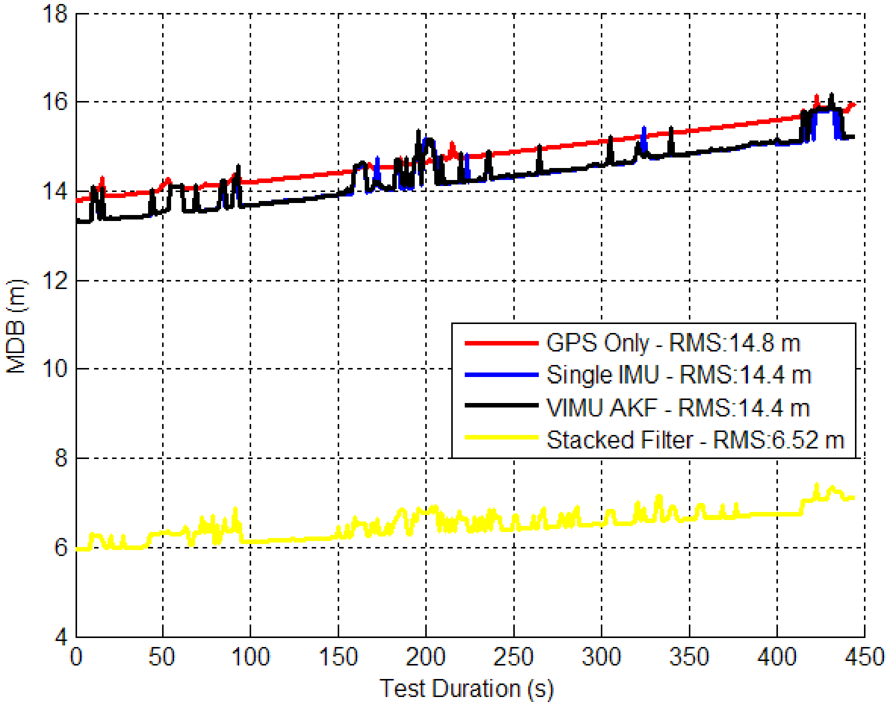


Figure 45 – Comparison of MDBs for PRN 22 for each Filter (Residential Data Set)

Alternatively, the stacked filter decreased the MDB substantially since it incorporated multiple positions into the detection algorithm. The stacked filter MDB RMS decreased by a factor of 2.27 compared to the theoretical value of 2.24 (1/√n) when five IMUs were used. This ratio, as derived in Equation (5.21), is shown in Figure 46 over the duration of the test. The green line shows the theoretical ratio. The discontinuities in the ratio arise when mismatches between the faults detected in one filter are not detected in another. In this case, the discontinuities show that the ratio increases, which indicates that the stacked filter MDB temporality increases as a result of more rejections. Thus, since the stacked filter rejects more observations, the MDB slightly increases with the reduction in geometry and the discontinuities arise. In one case however, the MDB remained constant and the MDB of the single IMU decreased. This would suggest a false detection in the SINS filter where the MDB would increase and cause the ratio to decrease.

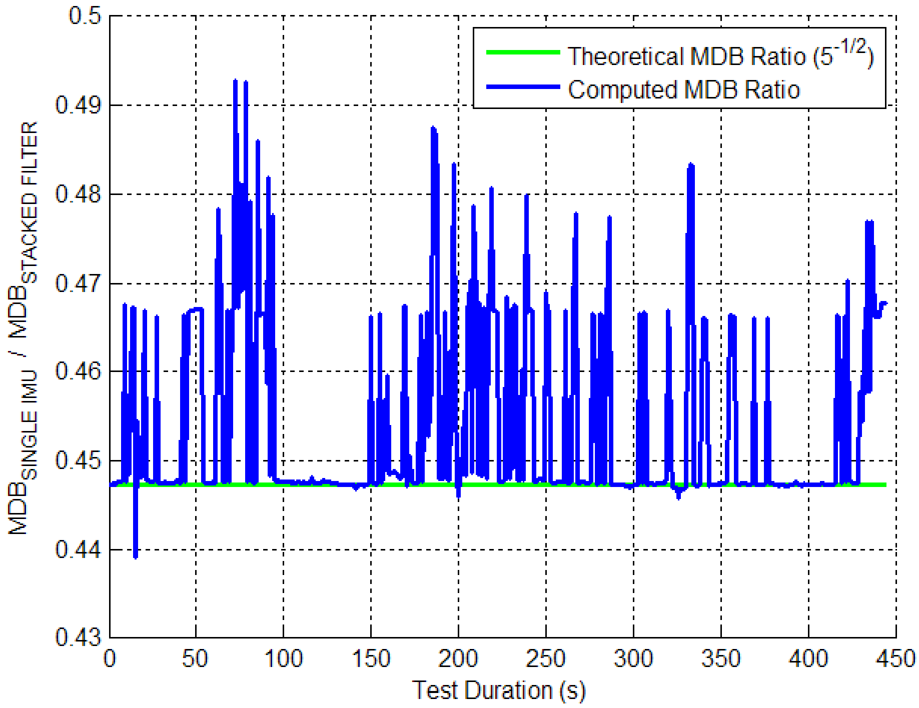


Figure 46 – Ratio of Stacked MDBs and SINS MDBs for PRN 22 (Residential Data Set) (see Equation (5.21))

6.5 Olympic Oval Data Set

The Olympic Oval presents a different approach to that of Section 6.4 where GPS, although moderately attenuated, is still operating within the requirements set out in Table 1. In this environment, GPS will not provide acceptable performance for most applications and an integrated system is needed. Figure 47 shows an average power drop of 24 dB inside the Oval while the HDOP occasionally doubles. This figure also shows the relative power increases when the user is located outside to allow the reference solution to re-estimate the IMU errors (i.e. 500 to 750 s).

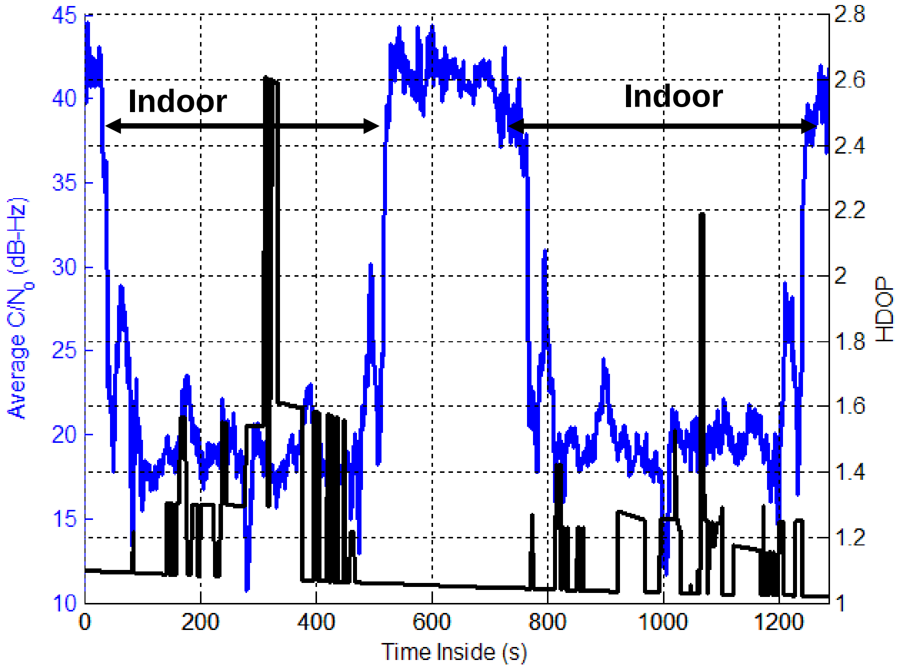


Figure 47 – Average C/N$_0$ and HDOP (Olympic Oval Data Set)

6.5.1 Position Accuracy

Table 9 provides the statistical position errors for the indoor Olympic Oval test. Again this table will not be explicitly discussed, but the information contained within is used throughout Section 6.5.

Table 9 – Horizontal Errors of All Architectures for Olympic Oval with IMUs Rigidly Mounted on the Foot

Processing Method	Number of IMUs	Mean	Mean Improvement	RMS	RMS Improvement	Std	Max	Min	Median
		(m)	(%)	(m)	(%)	(m)	(m)	(m)	(m)
Standalone GPS	0	26.0	0.0	33.6	0.0	21.4	128.2	0.0	25.5
SINS	1	19.9	23.6	24.9	25.9	15.1	52.0	0.3	16.5
	1	16.4	37.0	20.1	40.3	11.6	40.3	1.1	14.1
	1	23.0	11.7	27.5	18.3	15.1	52.0	0.2	27.4
	1	16.5	36.4	22.8	32.4	15.7	51.8	0.1	10.8
	1	19.2	26.3	25.9	23.0	17.4	56.1	0.0	13.6
VIMU (6 State LSQ)	2	16.1	38.1	20.0	40.6	11.8	38.9	0.0	14.6
	3	15.8	39.4	20.4	39.5	12.9	39.5	0.1	9.7
	4	15.5	40.5	19.9	40.8	12.6	38.9	0.2	10.5
	5	16.1	37.9	21.0	37.7	13.4	42.2	0.2	11.4
VIMU (9 State LSQ)	3	16.3	37.3	21.3	36.8	13.7	44.6	0.2	11.0
	4	16.6	36.0	21.5	36.0	15.4	53.6	0.0	18.4
	5	15.5	40.3	20.2	40.1	12.9	45.7	0.0	14.2
VIMU (AKF)	2	16.2	37.7	20.7	38.6	12.8	43.5	0.2	14.3
	3	15.9	38.9	20.8	38.3	13.4	44.3	0.1	11.3
	4	16.5	36.6	21.6	35.9	13.9	48.6	0.1	14.2
	5	15.1	42.1	19.5	42.1	12.3	40.2	0.0	11.9
Stacked Filter	2	17.8	22.2	24.6	27.0	13.3	48.0	0.3	15.3
	3	16.8	21.0	23.2	31.1	12.7	40.2	0.7	12.7
	4	15.8	20.3	20.8	38.3	12.8	41.4	0.4	8.8
	5	16.1	20.8	19.7	41.5	13.1	41.4	0.0	9.9
FNR (GPS)	2	17.4	33.1	21.6	35.8	12.8	45.1	0.1	14.8
	3	17.1	34.1	22.0	34.5	13.9	45.7	0.8	11.0
	4	16.7	35.8	22.0	34.7	14.3	49.1	0.4	9.1
	5	17.1	34.1	22.7	32.6	14.9	50.6	0.1	11.4

Table 9 (Cont'd)

Processing Method	Number of IMUs	Mean	Mean Improvement	RMS	RMS Improvement	Std	Max	Min	Median
		(m)	(%)	(m)	(%)	(m)	(m)	(m)	(m)
FNR (INS)	2	21.5	17.2	27.3	18.9	16.8	60.8	0.8	17.7
	3	21.4	17.5	27.2	19.2	16.7	59.9	0.7	18.4
	4	21.2	18.3	27.0	19.7	16.7	59.9	0.8	17.9
	5	21.3	18.2	27.0	19.6	16.7	59.9	1.0	17.5
FFR (INS)	2	20.5	21.2	25.2	25.1	13.6	52.2	0.3	12.5
	3	20.3	22.0	25.2	25.0	15.0	52.4	0.2	16.7
	4	20.0	23.0	25.1	25.4	15.1	52.0	0.6	16.4
	5	20.0	23.1	25.1	25.3	15.2	52.6	0.3	16.6
FZR (INS)	2	52.0	-100.1	59.3	-76.4	28.6	145.0	1.0	44.4
	3	94.1	-262.0	107.5	-219.5	52.0	217.5	3.2	87.9
	4	89.6	-244.5	101.1	-200.4	46.8	180.9	3.0	77.1
	5	80.6	-209.9	98.7	-193.5	57.1	234.0	10.7	60.1

6.5.1.1 SINS Results

The GPS and SINS horizontal errors for this 21.4 minute test are provided in Figure 48. Once again, the results are presented to show the capabilities of SINS solutions and provide context for adding more IMUs in this environment. Accuracy varied by tens of metres while indoors, but converged when the subject was outside (i.e. in open skies) to within a few metres.

The horizontal RMS errors varied from 20.1 m to 27.6 m, while the maximum error varied between 40.3 and 56.1 metres. GPS alone had a maximum error of 128.2 m and it is obvious that the integrated system was able to mitigate the multipath errors more effectively. Two SINS solutions (IMU 3 and IMU 4) took nearly 60 s to converge to the standalone GPS position when the subject exited the first loop. The other three SINS solutions follow the GPS error trend more consistently when exiting the building. Thus, each solution had varying degrees of convergence times.

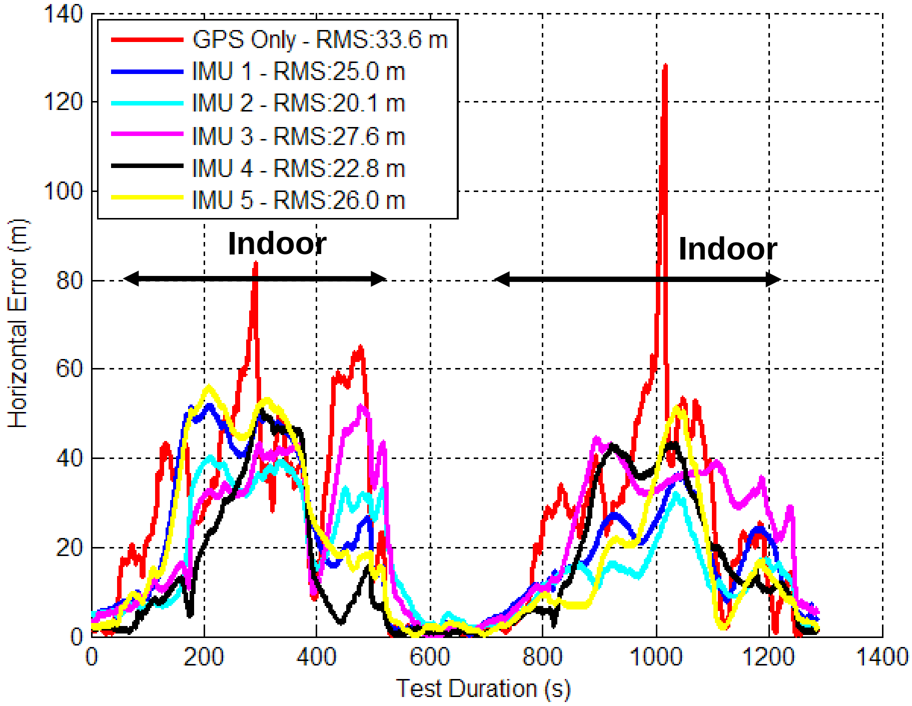


Figure 48 – Horizontal Errors of Five SINS Solutions (Olympic Oval Data Set)

The vertical errors of the solution are shown in Figure 49. Their RMS varies from 4.4 m to 14.4 m, much better than the 47 m error standalone GPS provided. Vertical errors had an extremely low frequency error, which can be attributed to the ZUPTs applied, and provided a slow error accumulation in the vertical axis. A large bias was also observed in some of the IMUs, in particular IMU 2, with a bias of approximately 5 metres. This bias remained stable during the time outside and the GPS aided the solution. Better filter tuning in the vertical axis for this particular IMU could possibly resolve this problem.

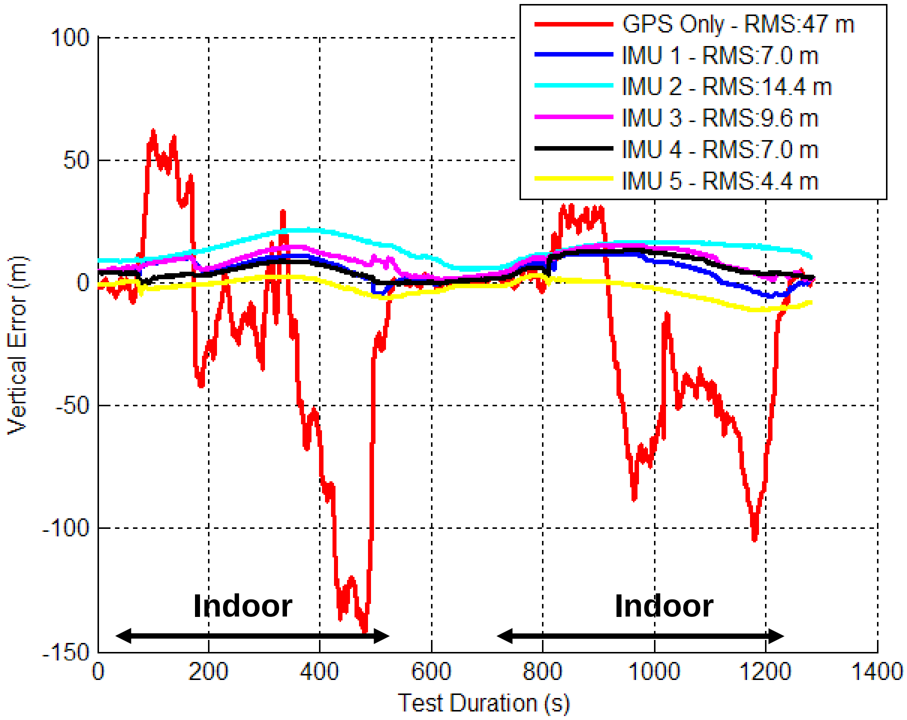


Figure 49 – Vertical Errors of Five SINS Solutions (Olympic Oval Data Set)

Figure 50 shows the CD of the errors in the horizontal plane and vertical axis. The vertical axis errors exhibit a substantial improvement compared to the horizontal plane. This bottom figure shows that 80 % of the elevation errors are less than 5.5, 10.0, 10.0, 13.2, and 16.5 for IMUs 5, 1, 4, 3 and 2 respectively, and that although these show a considerable improvement over the standalone GPS solution, they differ for each IMU. A similar conclusion can also be observed when the horizontal errors at 60 % shows a 14 to 32 m difference.

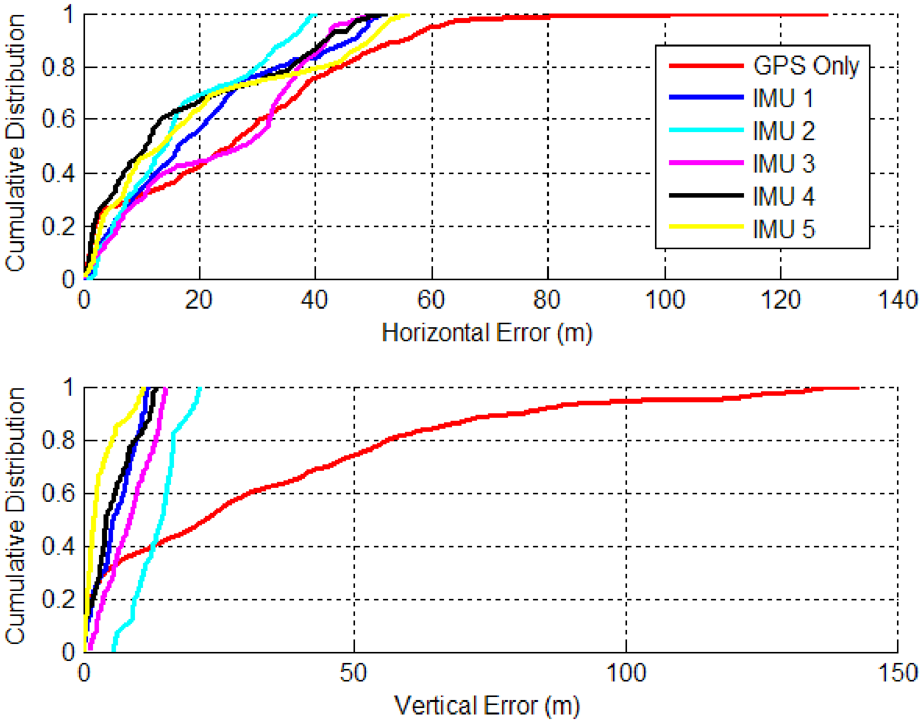


Figure 50 – CD of SINS and Horizontal and Vertical Errors of Five IMUs (Olympic Oval Data Set)

6.5.1.2 VIMU Results

The VIMU horizontal errors are shown in Figure 51. The horizontal error improvement is more significant than that of the residential data set (e.g. Section 6.4.1.2). The VIMU average provided a 37.7 % improvement, and the LSQ and AKF methods were similar with 40.1 % and 42.1 % improvements, respectively. This is further investigated in Section 6.8.1, which shows that the results are also hindered by time tagging issues due to several IMUs using their own independent clocks.

The VIMU tends to diverge much more slowly when entering the indoors and converges much more quickly when exiting, compared to the SINS solution. That said, at time 185 s, the solution very quickly diverged from a 6 m error to nearly a 40 m error. This was a direct result of a strong multipath signal that had a high C/N$_o$. The filter consequently overweighed the GPS measurement and the VIMU filters were unable to reject this information. This effect has been seen in all the filters during this research and presents a problem that could not be solved without manual intervention of the observation covariance matrix. An important observation was that the VIMU filter was able to mitigate the error the longest. This effect can be seen on a map in Figure 57 which occurred in the north east corner of the Oval.

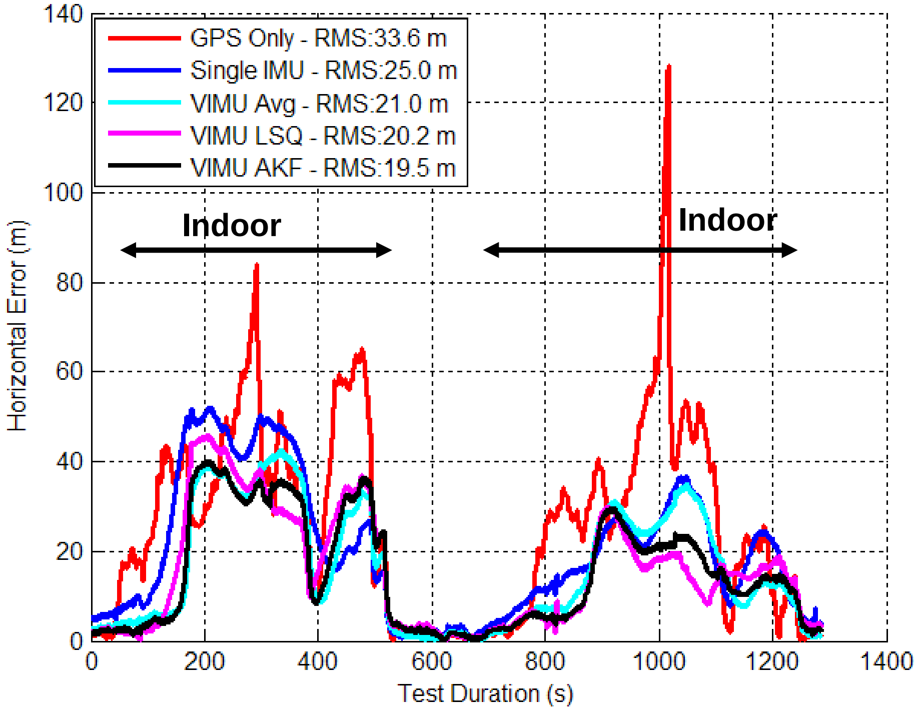


Figure 51 – VIMU Horizontal Errors (5 IMUs Used in Olympic Oval Data Set)

The elevation profile, shown in Figure 52, displays the same elevation divergence as seen in the residential case described in Section 6.4.1.2. This is addressed in Section 6.8.3 and is a function of filter tuning rather than the fusion of raw IMU measurements.

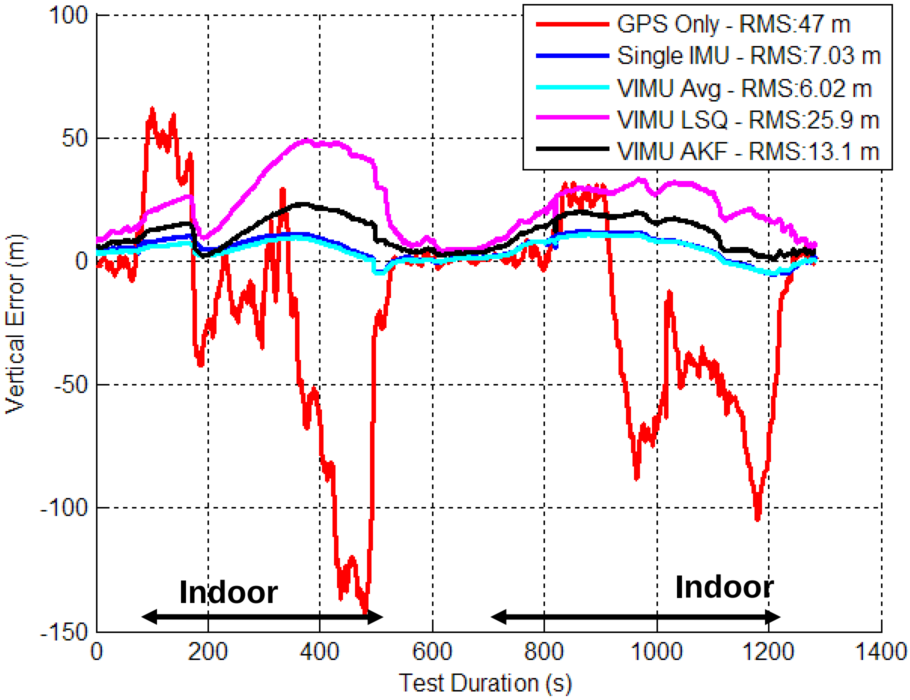


Figure 52 – VIMU Vertical Errors (5 IMUs Used in Olympic Oval Data Set)

Figure 53 shows the CD of the horizontal and vertical errors. The VIMU‟s horizontal errors showed superior performance at 40 % error. This revealed a distinct advantage over the SINS solutions. However, beyond 40 % the advantage was less pronounced and provided only marginal improvement compared to the SINS solution. In the vertical axis the LSQ and the AKF drifted but then slowly converged when GPS was less attenuated. This convergence was much slower than in the SINS and VIMU average solution.

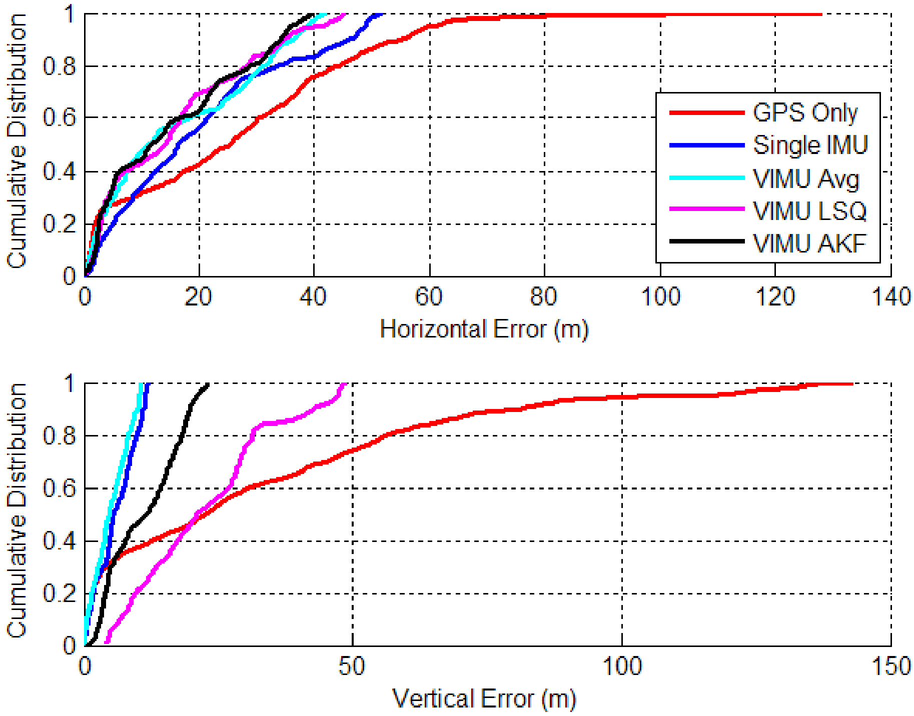


Figure 53 – CD of VIMU Horizontal and Vertical Errors (5 IMUs Used in Olympic Oval Data Set)

6.5.1.3 Stacked and Federated Filter Results

Figure 54 provides the stacked and federated filter horizontal error results. The best solution was the stacked filter which outperformed its FNR (GPS) counterpart by 8.9 %. This is evidence of the effectiveness of the relative updates providing more information to the filter assisting in constraining the divergence of the system when GPS is providing poor observations. The FNR (GPS) filter again provided more accurate results to the FNR (INS) and FFR (INS), which provided similar results as the SINS solutions.

The SINS and the FFR (INS) error profiles in Figure 54 show a similar result. This occurred because the reference INS in the FFR were the same single INS plotted in Figure 54. This introduces a concept where the reference local filter was aiding the other local filters to follow its trajectory because the input “observations” were time correlated.

This, in some cases, is to the detriment of a federated filter using one local filter as its reference solution for other local filters. This result confirmed the theory presented in Section 5.4; the reference system data must yield to assumptions of the Kalman filter shown in Equations (4.9), (4.10), (4.11).

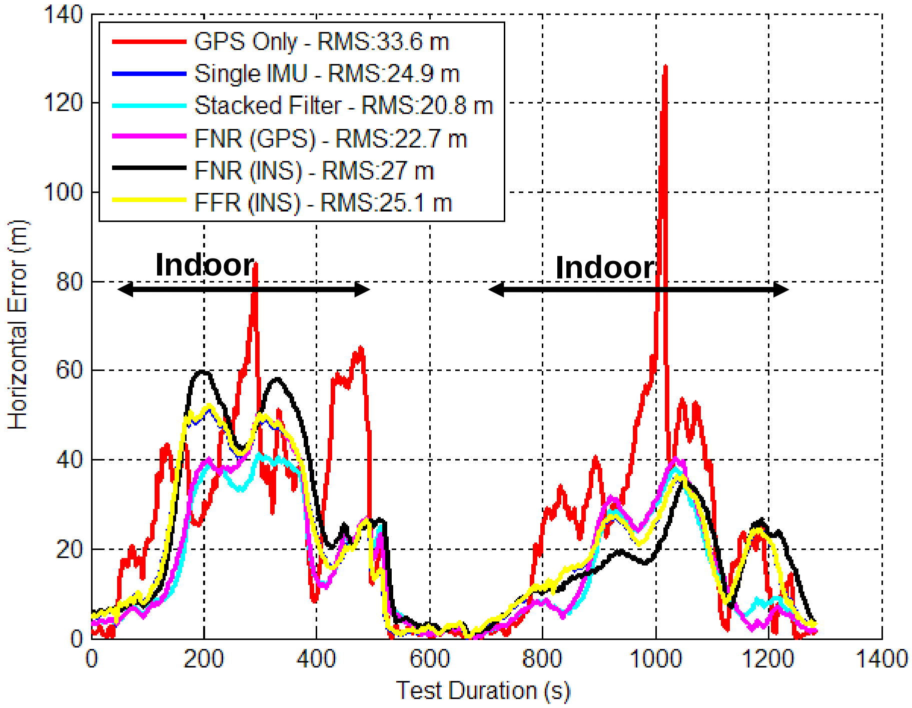


Figure 54 – Horizontal Error of Stacked and Federated Filters (5 IMUs Used in Olympic Oval Data Set)

In the vertical axis, as shown in Figure 55, the filters generally performed similarly to the best SINS solution indicating that additional IMUs and varying architecture do not further improve the elevation accuracy. In both loops, the maximum elevation error was approximately 12 m which was significantly better than the error in the horizontal plane, (i.e. four to six times better). The RMS errors were also consistent between architectures, varying by less than a metre, further confirming that the improvement was a result of the ZUPTs and was not related to the architecture, number of IMUs or relative updates.

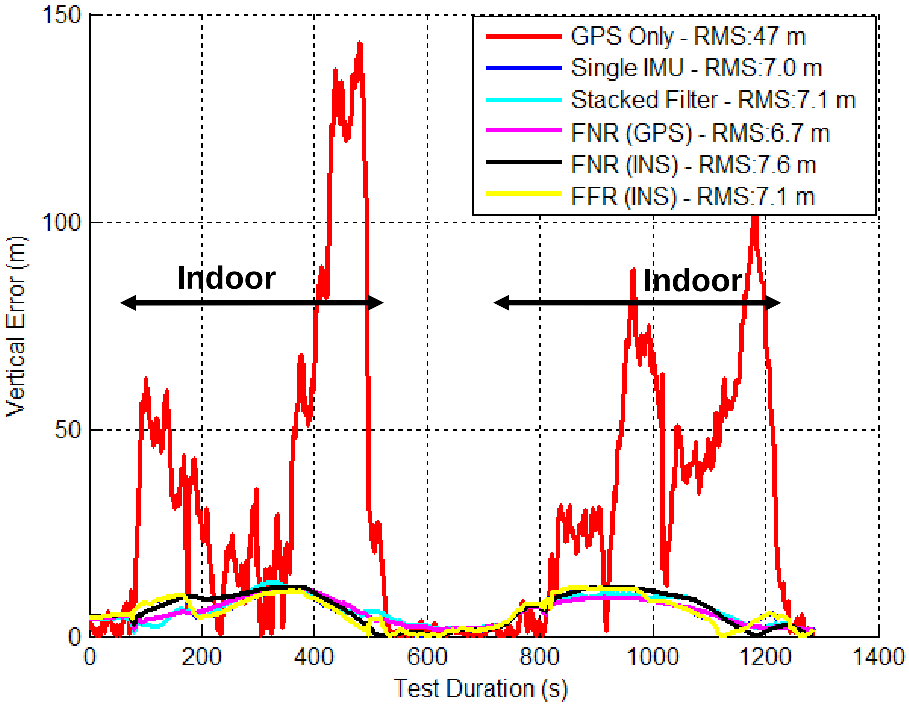


Figure 55 – Vertical Error of Stacked and Federated Filters (5 IMUs Used in Olympic Oval Data Set)

Figure 56 shows the CD of the horizontal and vertical errors. The stacked filter provided a reasonable improvement at 90 % CD where it outperformed the FNR (GPS), but followed a similar trend at lower percentages. Both the FNR (GPS) and stacked filter behaved similarly below 80 %, which showed that, in terms of the distribution, the relative updates were providing improvement at times when the FNR (GPS) did not.

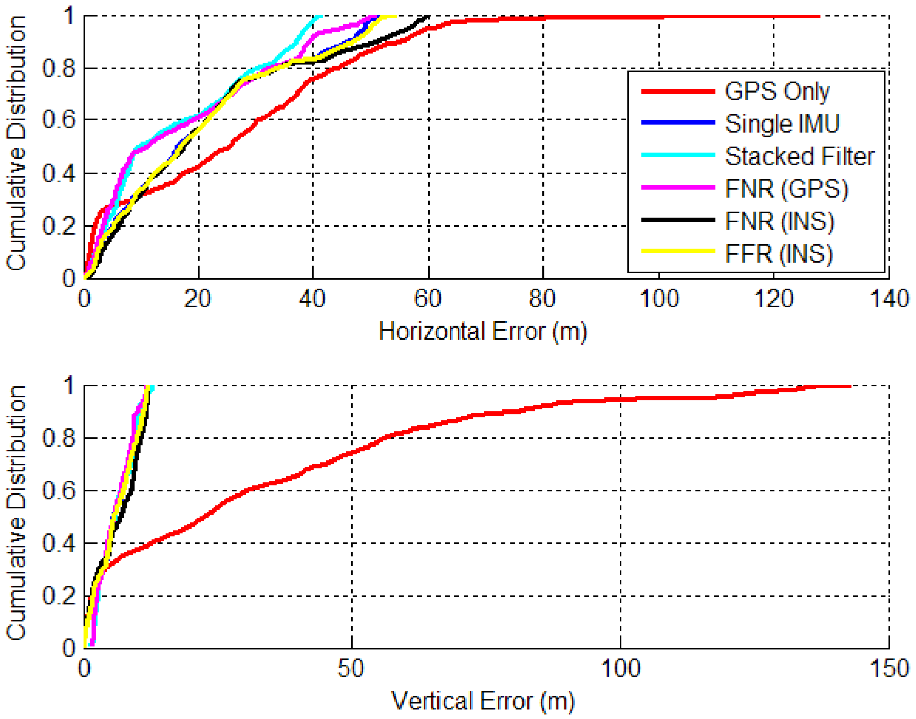


Figure 56 – CD of Horizontal and Vertical Errors for Stacked and Federated Filters (Olympic Oval Data Set)

To compare the results of each filter, Figure 57 and Figure 58 show a map containing the trajectories of each architectures best solution (i.e. least amount of RMS error). A standalone GPS solution and a SINS solution are also provided for context. During the test the subject walked around the Oval in a counter-clockwise direction, then went outside, returned to the Oval and walked around the Oval in a clockwise direction. In the first loop, Figure 57 shows that the VIMU provided an excellent trajectory until the northeast corner where it diverged in the presence of a strong multipath signal (see Section 6.5.1.2). Comparing this to the SINS solution, where the heading immediately diverged after reaching the northward turn at the Oval track, the VIMU heading exhibited an excellent ability to provide a correct heading. The stacked and FNR filters provide a similar trajectory, again the largest error appears to be in the heading which had diverged as the subject exited the SE corner of the track. All solutions at the SE corner have been without good GPS observations for six minutes and were effectively navigating on high multipath signals.

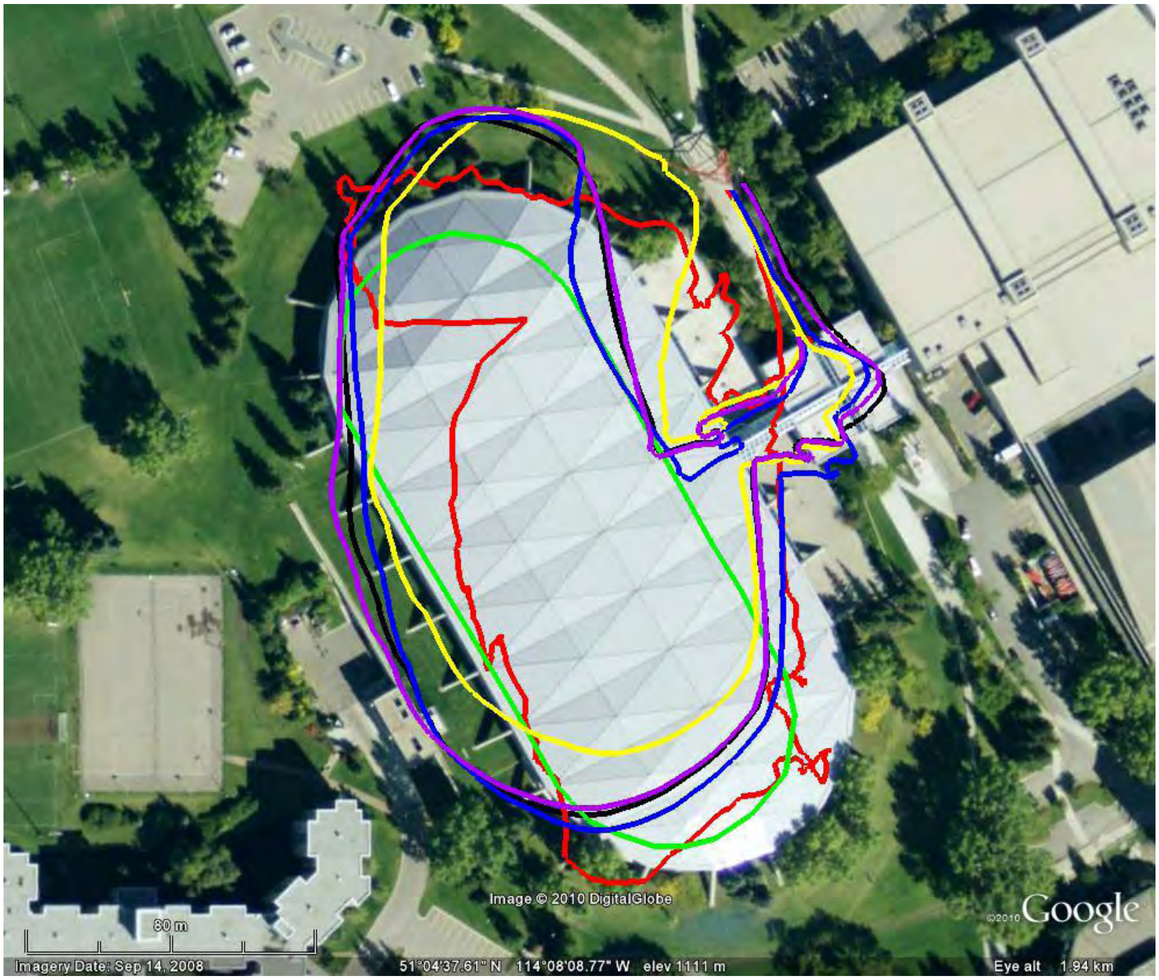


Image © 2010 DigitalGlobe
80 m
Imagery Date: Sep 14, 2008
51°04'37.61" N 114°08'08.77" W elev 1111 m
©2010 Google
Eye alt 1.94 km

Figure 57 – Loop 1 (Counter Clock Wise) Map View of Best Performing Filters - Truth Solution, Standalone GPS Solution, SINS, VIMU (AKF), Stacked Filter, FNR (GPS)

The trajectories of the second loop, shown in Figure 58, appear even better than the first, especially in the north east and south east corners. In this trajectory, the SINS was well aligned, but had acquired an along-track error, that provided the large horizontal error shown in Figure 48. By the time the user exited the track, the SINS solution contained the largest heading error. This was indicative of the heading degrading during the time indoor, which was less prominent in the multi-IMU architectures. With remarkable accuracy, the FNR (GPS) and the stacked filter had aligned themselves with the truth trajectory at the north east corner and appear to have an ideal heading. The subject then made an eastward turn to exit the Oval track where the FNR (GPS) solution provided the best accuracy while exiting the building.

For the Oval data, the user entered and exited the track at the same point and therefore provided an interesting metric to compare the solutions. The FNR (GPS) filter only deviated by 2.5 m, the SINS difference was 13.5 m and the standalone GPS solution had 49.3 m difference. The same check of the reference system yielded a 5.1 m difference.

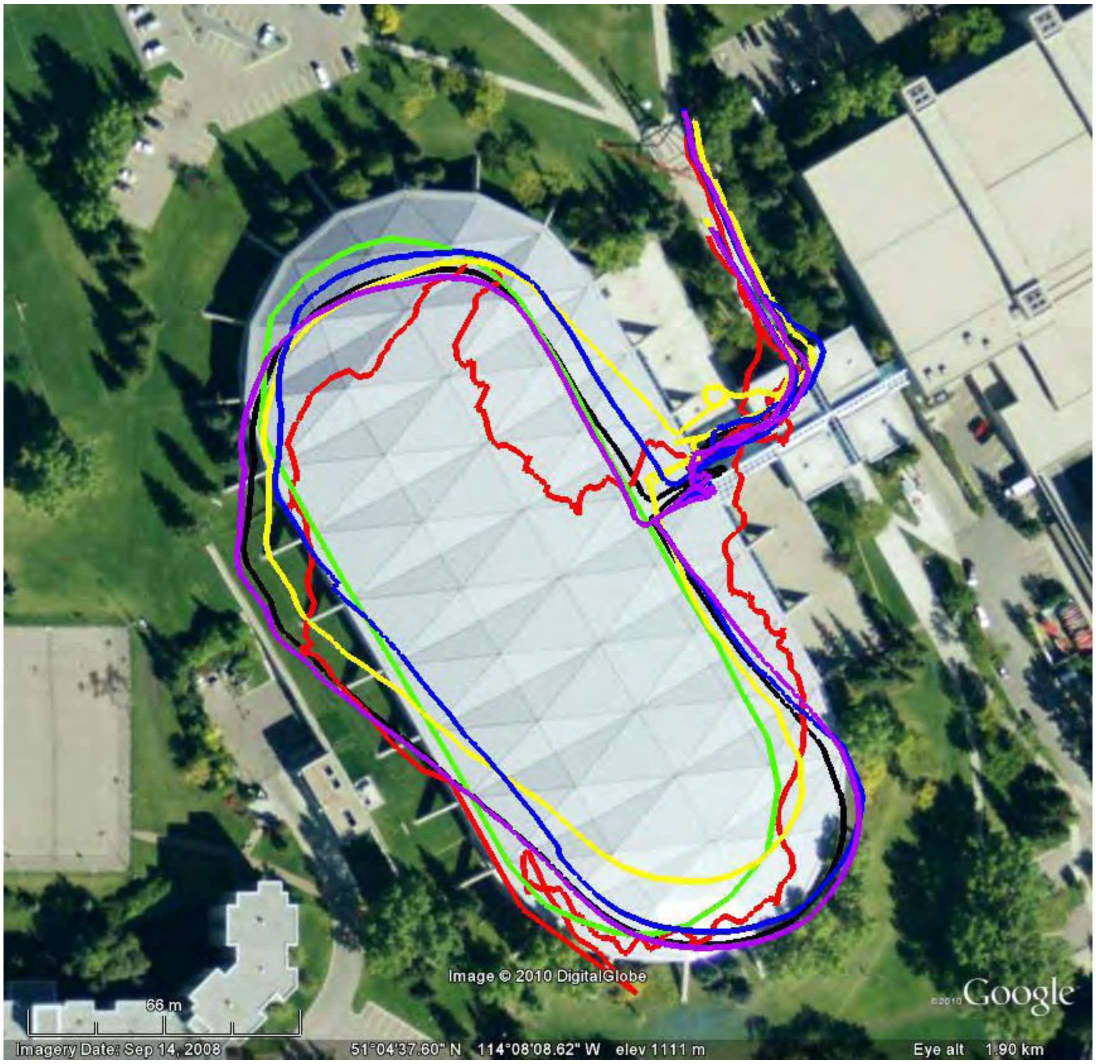


Image © 2010 DigitalGlobe
66 m
Imagery Date: Sep 14, 2008
51°04'37.60" N 114°08'08.62" W elev 1111 m
Eye alt 1.90 km
©2010 Google

Figure 58 – Loop 2 (Clock Wise) Map View of Best Performing Filters - Truth Solution, Standalone GPS Solution, SINS, VIMU (AKF), Stacked Filter, FNR (GPS)

6.5.2 Position Accuracy vs. Number of IMUs

The accuracy of the VIMU as a function of the number of IMUs is shown in Figure 59. This figure provides an indication of the weakness of the VIMU in time tagging. Because of this issue, the improvement was less incremental for all the VIMU fusion methods. In this case the VIMU AKF provided the best solution, albeit with marginal time synchronization.

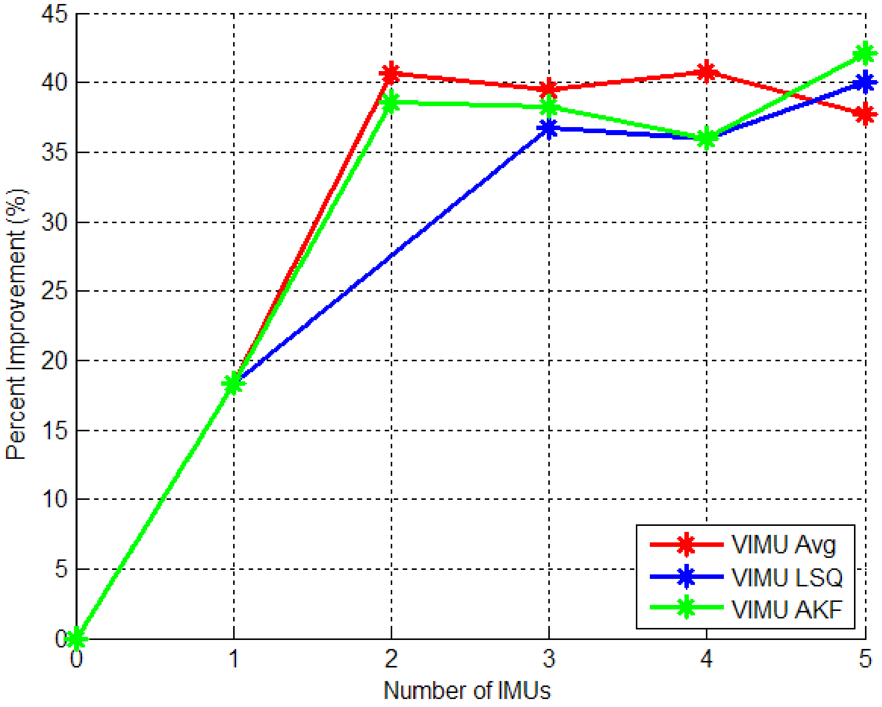


Figure 59 – VIMU Accuracy Improvement as a Function of IMUs Used (Olympic Oval Data Set)

Figure 60 shows the RMS accuracy improvement as a function of the number of IMUs for the stacked and federated filters. The stacked filter had a linear improvement for each additional IMU of about 3 to 7 % per IMU added. This again indicates the value of the relative updates, as each additional IMU provided additional relative information to improve the accuracy of the solution and the error states within the block filters. The FNR (INS) and the FFR (INS) results did not increase linearly, but plateaued similar to the results of two IMUs. The FNR (GPS) slightly decreased with each additional IMU in excess of two.

The FNR (INS) and FFR (INS) results were very similar to the residential data set with very moderate improvements as each IMU was added. The FNR (GPS) also had similar results between data sets with a slight decrease in performance with more IMUs. The two data sets confirm that the federated filter architecture did not increase the accuracy, but merely processed the data in a similar manner to that of the centralized version.

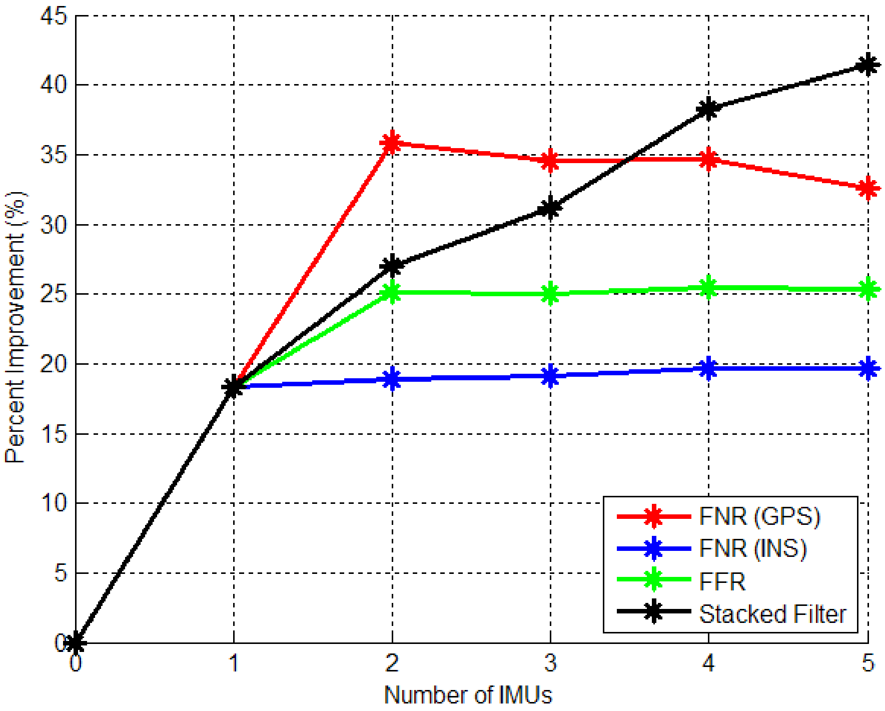


Figure 60 – Stacked and Federated Filter Accuracy Improvement as a Function of IMUs Used (Olympic Oval Data Set)

6.5.3 Minimum Detectable Blunder

The MDBs of the various filters proposed in this thesis are shown for PRN 31 in Figure 61. The MDB of all architectures, with the exception of the stacked filter, were very similar. The MDB decreased for each multi-IMU filter, but by only a few metres. This was consistent with the theory, as the MDB is a function of the absolute information contained in the innovation covariance. Since inertial observations only provide temporal positioning information, there is not a significant increase in the ability to detect faults in GPS measurements. On the other hand, the stacked filter MDB was computed using several absolute positions within its filter, which enhanced its ability to detect faults.

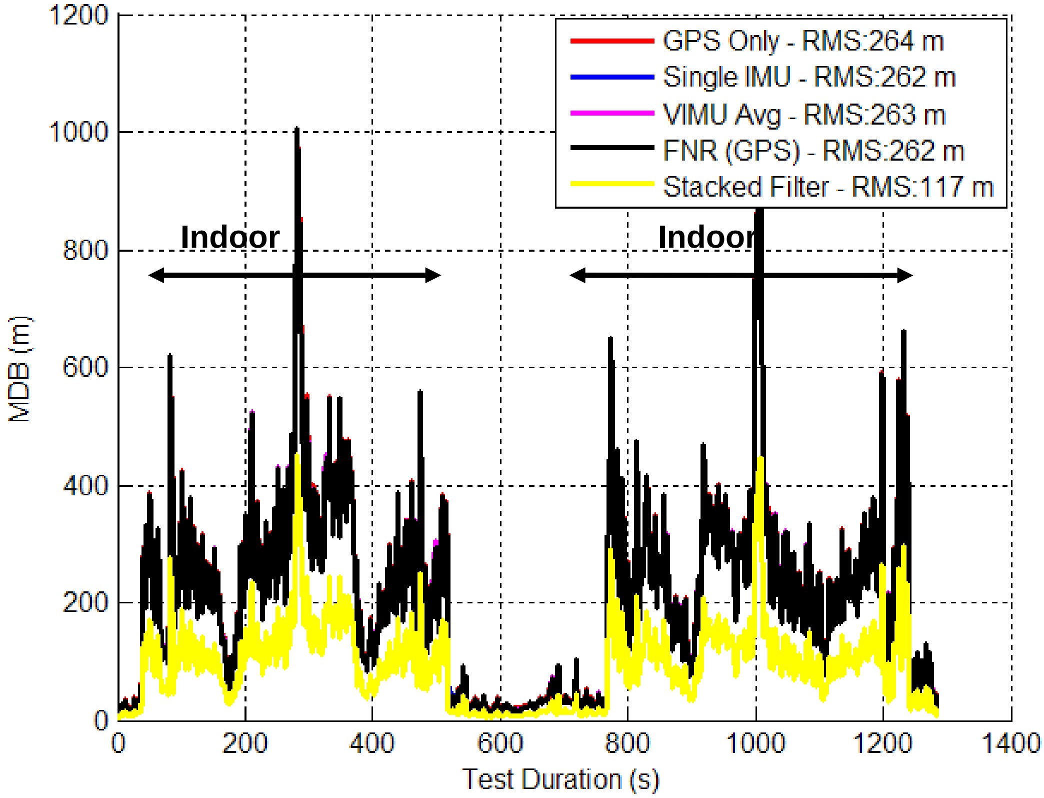


Figure 61 – MDB of PRN 31 for Various Architectures (Olympic Oval Data Set)

6.6 Estimated Position Variances

A major aspect of navigation is the estimated variances of the solution. A system whose estimated covariances are not indicative of the real errors provides an untrustworthy system. Thus validating that the estimated position covariance and the true errors coincide is an important comparison. Figure 62 and Figure 63 show the horizontal position errors and the estimated standard deviations of the position as determined in the software. The figures show the three sigma bounds, and the horizontal standard deviations were determined using the propagation of variances.

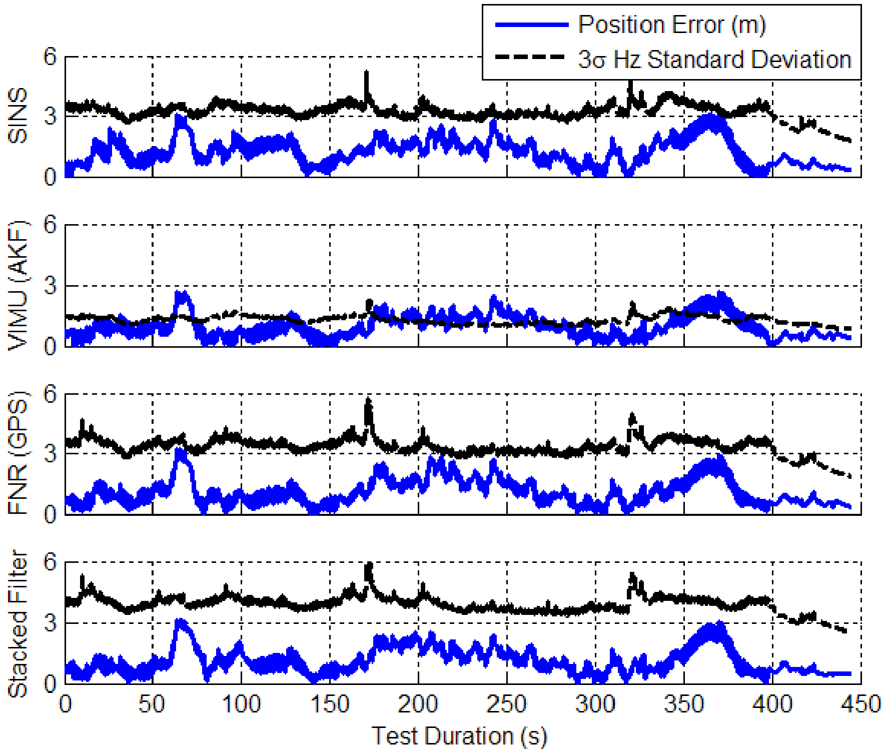


Figure 62 – Comparison of Horizontal Errors and Estimated (3σ) Position Standard Deviations (Residential Data Set)

The analysis shows that the VIMU results are slightly overoptimistic in the residential data set. This is attributed to the fact that the process noise of the IMU has been reduced, but other errors present in the system are still contributing to the navigation solution errors (i.e. GPS multipath).

Both the FNR (GPS) and the stacked filter results show realistic variances. This confirms that the filters are operating at a reasonable level, even more so than would have been anticipated considering the blanket tuning approach used (see Section 5.7).

In the Olympic oval data set, 8.3 % of the SINS horizontal position errors exceeded the 3σ values. Given that multipath is extremely difficult to model and is not well represented in the observation covariance matrix, this percentage is acceptable. The

VIMU data set had 4.0 % of the errors greater than the 3σ estimates, while the FNR (GPS) and stacked filter had 5.2 % and 4.9 %, respectively. These numbers indicate that the position variances output by the filters are indicative of the true errors.

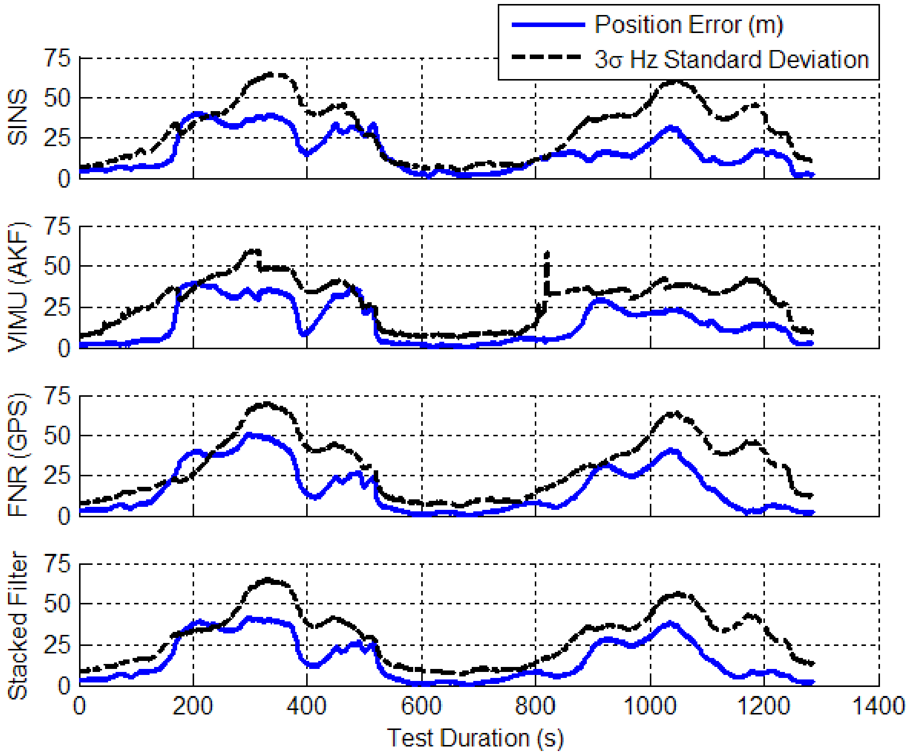


Figure 63 - Comparison of Horizontal Errors and Estimated (3σ) Position Standard Deviations (Olympic Oval Data Set)

6.7 FZR Filter Results

FZR is not suitable for pedestrian navigation applications with multiple IMUs in degraded GPS environments. The FZR filter, for all test cases reported herein, provided unusable results with errors larger than those of the standalone GPS solution. From a practical stand point, the master filter requires navigation solutions every second (or few seconds); the exact output rate could be varied for each application. Therefore, the FZR filter has two approaches to provide the navigation solution at this interval (i.e. 1 Hz). The first approach to update the master filter at 1 Hz (or less) using the local filters and thereby share information at that rate. The second approach is to predict the master filter at 1 Hz and share information at a lower rate, say 5 or 10 s. If the first approach is used and the sharing rate is reduced to 1 Hz, the local filters reset too often to converge back to a reasonable navigation solution. This approach is further compounded by GPS signal attenuation to levels that provide very slow convergence rates (e.g. indoor). The second approach is also unrealistic because of the amount of process noise added in pedestrian navigation filters over 5 or 10 seconds. This effect is even further amplified when the IMU is located on the foot and the prediction of the master filter can occur at any point in the gait cycle where the velocity could either be zero (during the stance phase) or at a maximum (during the swing phase). In either case, the FZR master filter performs poorly at low data rates. Thus, the information stored in the master filter, which is sub optimal, further degrades performance when replaced back into the local filters.

This effect could be theoretically bypassed by changing the information sharing principle to be sequentially processed, where information is shared backward from the master filter to one local filter at a time. This extends beyond the scope of this thesis, but could provide additional federated filter architectures for pedestrian navigation results.

6.8 VIMU Fusion Issues

This section analyzes the capability of the proposed VIMU fusion techniques and provides an explanation of the degraded elevation solution within the VIMU architecture.

6.8.1 VIMU Timing

The Olympic Oval data set saw similar navigation improvement when adding more IMUs when compared to the residential data set, but contained larger residuals within the VIMU fusion after entering the Oval. This result is attributed to the time tagging limitations of the IMU measurements. Each IMU is equipped with an internal clock and is updated with a Pulse Per Second (PPS) signal generated by the reference INS. This serves as the time synchronization between GPS and IMU for integration and was originally considered acceptable for a single IMU and GPS integration. This synchronization had an approximate maximum deviation of 1 ms, which with respect to single IMU integration is acceptable. However, in the case of integration with several IMUs, clock synchronization accuracy for VIMU must be higher.

During the Oval test, the IMUs clocks frequency and consequently the time bias shifted when the subject entered the Oval. It is hypothesized that this was due to a temperature variation (i.e. outside was 28° C vs. inside near the ice rink it was close to 18° C). This change in temperature shifted the observation time measurements only slightly (on the order of 7 ms) and became obvious when the IMU data was compared side by side. Figure 64 shows the difference between the X axis gyro measurements of two IMUs mapped to the same frame. As the user traveled to the Oval track, the differences increased. This effect was clearly seen in the residuals of the VIMU estimation methods.

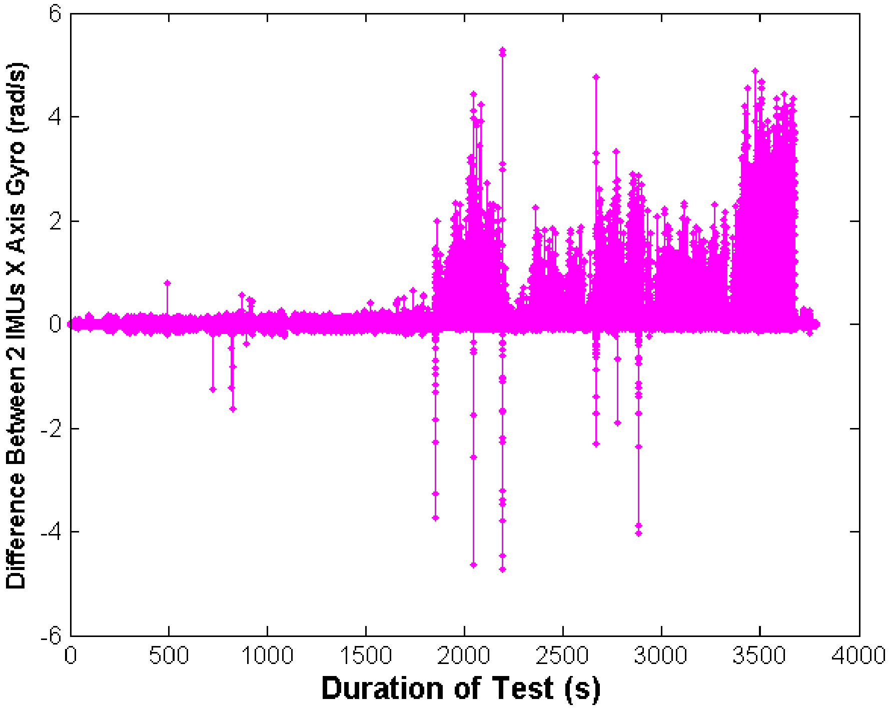


Figure 64 – Differences between 2 IMUs X Axis Gyro Observation when entering the Olympic Oval

There is also another potential source of timing error which stems from the PPS accuracy of the reference INS. Since the PPS was sent to each IMU, exact time synchronization should occur between all IMUs. To confirm this, the clock bias is shown in Figure 65. The error bounds are also shown in the figure. By comparing the time when the residuals increased in Figure 64 and the time when the clock bias increased in Figure 65, it is clear that the increased residuals occur before the large deviation of the clock bias. Further, the clock bias increased when the user entered the building and the residual errors increased when the user reached the lower part of the Oval, where the temperature was coolest.

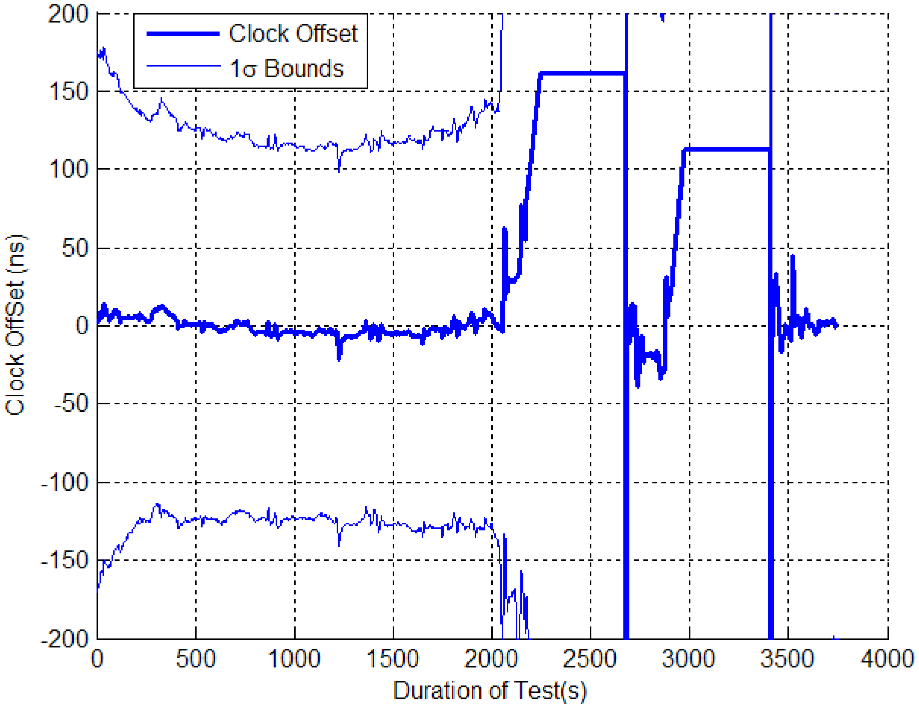


Figure 65 – PPS Timing Accuracy from NovAtel SPAN System during Oval Test

This time tagging issue is very difficult to mitigate and thus makes time synchronization of IMU data sets to within an acceptable interval very difficult and often impossible.

6.8.2 Estimating the Angular Acceleration of the VIMU

Estimating the angular acceleration is a nuisance parameter (i.e. not important for navigation performance) and is only of interest to correctly map the acceleration to the VIMU frame. However, the ability to estimate the angular acceleration is important to consider. Since no truth solution is available to provide the angular acceleration, a time derivative of the angular velocity will suffice. Since the differentiation will remove the scale factor and biases but increase the noise, it still provides a solution to which the estimated parameter can be compared.

The upper graph of Figure 66 shows a short time segment of the VIMU LSQ with the raw differentiated data of one IMU, the raw differentiated data of the VIMU, and the estimated angular acceleration provided by the LSQ VIMU estimator. The full scale of the differences makes it difficult to show the subtleties, so the bottom graph shows a zoomed portion of the top graph. The bottom figure shows that the estimated angular acceleration is very close to the differentiated values and they are within the values of their estimated standard deviations. Although this segment occurs in the stance phase, it is also representative of the results when the foot is in motion.

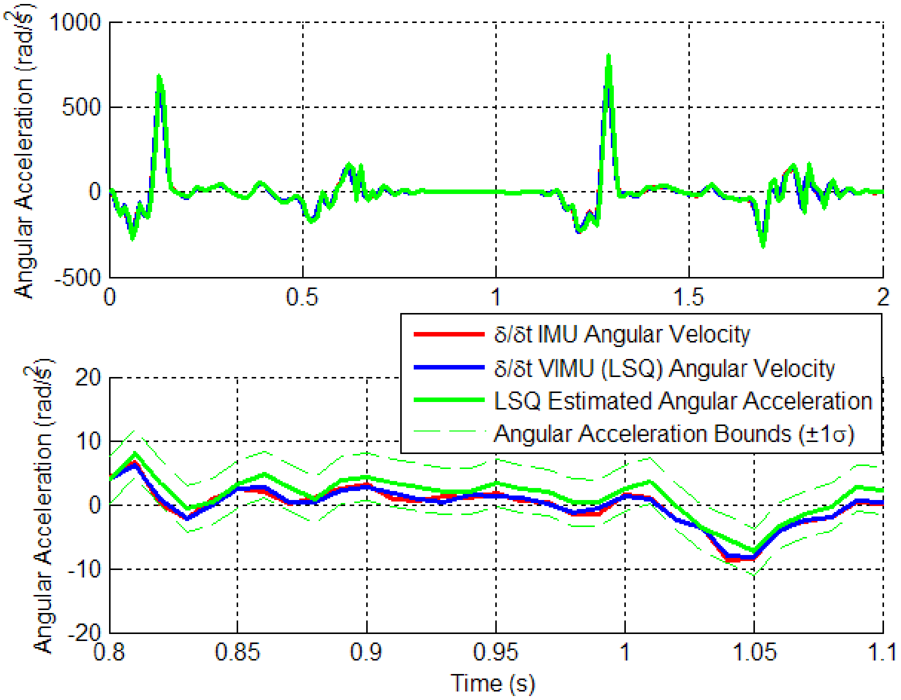


Figure 66 – Estimating the VIMU Angular Acceleration using LSQ

The estimated variances of the angular acceleration (and specific force and angular velocity) are of little importance in the least squares estimator, since all the IMUs are weighted equally. This information, although not used in this research, could be input into the navigation filter as a real time noise characteristic.

The VIMU adaptive Kalman filter input variance of the raw observations are critical for proper fusion. They remain unchanged during the filter operation (noting that only the process noise is adaptive). Since the filter must assume that the input is Gaussian noise, the standard deviation of the input filters was set to 0.03 m/s^2 for the accelerometers (see Figure 12) and 1600 °/h for the gyro (see Figure 13), much below the actual bias and scale factors expected. These values were derived from static data in Section 3.5.5 and were consistent with the spectral densities used within the Kalman filter for the filter‟s process noise.

Since the IMUs were rigidly mounted together, the estimated state variances within the AKF changed very minutely and were effectively constant. However, the variances were extremely optimistic since they did not account for the biases and scale factors and therefore the output of the AKF could only be considered a noise reduction technique, not a true estimate of the value.

6.8.3 Increased VIMU Vertical Error

The VIMU vertical errors shown in Figure 38 and Figure 52 clearly demonstrate the reduced performance of the VIMU vertical estimation. This result was a function of filter tuning. The IMU spectral density was scaled by an equal number for all three axes. However, the actual noise reduction, in the case of the VIMU LSQ, was not equal for all three axes. Figure 67 shows the percent decrease from the LSQ estimator for the accelerometer and gyro measurements. The gyro measurements for each axis all decreased equally per IMU added. The accelerometer measurements noise decreases as a function of the inter-IMU vector and the orientation between each IMU and the VIMU. Therefore the noise reduction for the accelerometers was not simply $n^{-1/2}$, where n represent the number of IMUs.

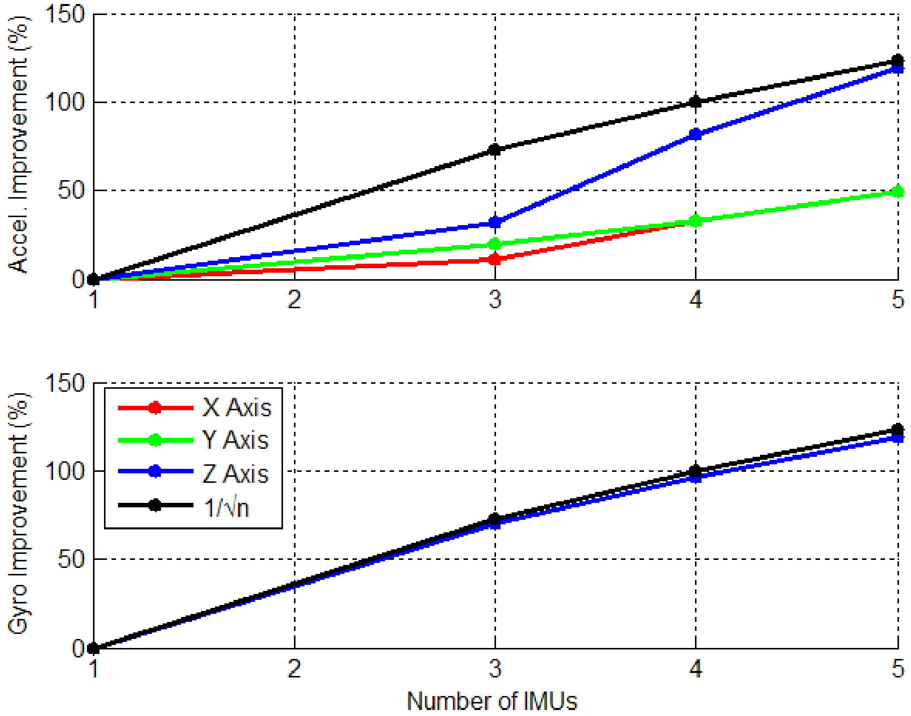


Figure 67 – Estimated Variance Improvement of VIMU LSQ (Theoretical)

6.9 Processing Speed of Architectures and Number of IMUs

There is a large difference in the computer processing speed of each architecture and for the number of IMUs used. An exact comparison of the computational load is beyond the scope of this thesis, but Figure 68 shows the processing rate of each architecture and the number of IMUs added for the software developed by the author. All data was processed on an Intel Core 2 Quad CPU with 3.25 GB of RAM. This analysis is merely intended to be comparative, since there are numerous factors that determine processing speed. The slowest architecture was the stacked filter. This was mostly due to the inversion required for the gain matrix computation, which has n times m rows and columns (n is the number of IMUs and m is the number of GPS observations); propagating the filter forward was also a burden. This was the only filter that was unable to run in real time.

The VIMU AKF was able to process faster than the federated filters, an interesting note considering the VIMU AKF produced solutions at 100 Hz whereas the federated filters operate at 20 Hz.

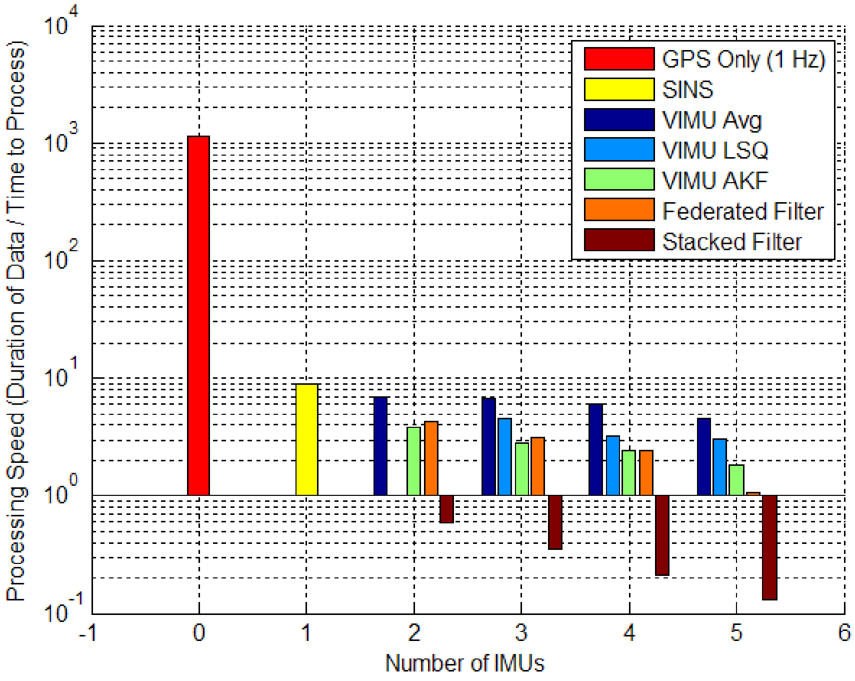


Figure 68 – Processing Speed of Various Architectures

These results are largely influenced by I/O processes such as the input and output of the filters data, which include PVA navigation parameters and estimated variances, biases and scale factors for each IMU with their respective variances, MDB information, satellite number and DOP information. Thus, in the event of a five IMU federated filter, the output was five times greater than that of a SINS filter.

6.10 Recommending an Architecture

The purpose of the thesis was to create, implement and analyze various multi-IMU estimation architectures. This section identifies which architecture might best be suited for specific environments and characteristics that would be expected for a pedestrian navigation system considering a multi-IMU approach, given the results within the chapter. Table 10 provides the recommended architecture as a function of characteristic, rating each architecture from one to three, one being the first choice, three being the last.

Table 10 – Architecture Preference as a Function of Development Characteristics

Development Characteristic	VIMU	Stacked Filter	Federated Filter
High IMU Noise	1	2	3
Low Multipath Environment (e.g. Residential House)	1	3	2
High Multipath Environment (e.g. Deep Indoors, Urban Canyons)	3	1	2
Relative Position and Attitude Known	1	1	3
Low Processing Load	1	3	2
Various Types and Qualities of IMUs Used	3	1	2
More than Two IMUs	2	1	3

The VIMU architecture would be preferred for a system to operate within a residential neighbourhood or areas where GPS in only moderately attenuated. This would allow for good multipath rejection and detection, and bridge any short gaps within the GPS data. In this case, the IMU would operate as a smoother. Alternatively, if the user was going to operate deep indoors, the architecture of choice would be the stacked filter. This method allows for added observations when indoor and can operate without GPS (for a limited time). The stacked filter also has a better FDE rejection method, which would enable the detection of poor GPS observations when a single INS configuration (including a VIMU) could not.

Another aspect that is important to consider in recommending a specific architecture would be whether the relative position and attitude values are known prior to operation. Most design plans would include this information, however this restriction may not always be the case. It is conceivable that a system could add additional IMUs via a module where the user could purchase any number of IMUs and place them on the subject. In this case the relative information would not be known and the benefit of relative updates or the mapping of IMU observations into the VIMU would not be possible.

Several other factors would dictate the architecture including design complexity, number and data rates of IMUs, processing capability and desired battery life. Thus the selection and recommendation of a specific architecture is left to the developer to weigh the benefits and detriments. Finally, IMU technology is currently going through rapid performance enhancements that would possibly affect the above ranking and proposed architectures.

Chapter Seven: Conclusions and Recommendations

This research proposed three architectures for which multi-IMU data can be fused to provide improved navigation performance. The filters proposed specifically assess the integration schemes within the scope of pedestrian navigation. The objective of this thesis was to compare the results of three architectures and provide insight into the advantages and disadvantages of each, providing a better understanding of the accuracy and availability for each filter. This chapter provides conclusions and recommendations that will benefit anyone pursuing multi-IMU fusion.

7.1 Conclusions

1. The stacked filter provided better results compared to its federated no reset counterpart, which showcases the use of the relative updates and a better fault detection algorithm. Although the improvement was minor in the residential data set, the filter was already operating at a high performance level with the use of only moderately attenuated GPS signals. In the Olympic Oval data set, the stacked filter performed 9 % better with five IMUs, than the federated no reset filter.
2. The multi-IMU federated filters accuracy reached a maximum with two IMUs, whereas the stacked filter accuracy linearly increases 3 to 7 % with each additional IMU. This suggests that the relative updates provide a linear relationship with the number of IMUs, at least up to five or so units.
3. When GPS measurements were used as the reference information for the local filters of the federated filter, the performance was 15% better than when a SINS solution was used as the reference for the federated filter. The time correlation of the output of the SINS solution resulted in a dramatic decrease in performance of the local filters, even though the SINS solution was more accurate.

4. The federated zero reset filter provided inadequate navigation solutions in both data sets. The limitation stems from the intermittent dynamics, which require the master filter to be updated and the local filters zero resets to be applied at high data rate (i.e. less than 1 Hz). This results in very short convergence times within the local filters. Increasing the data rate allows the local filters to converge, but results in the master filter continually predicting over extended periods of time and requires an excessive amount of process noise to account for the long prediction time. Further, since the velocity over the prediction interval is constant, it does not accurately account for the velocity during the gait cycle, and resultantly provides a poor prediction of the position.
5. Within the VIMU scope, FDE is not practical unless the systematic errors have been removed prior to testing for faults. Performance within the FDE is severely hindered by the dynamics of the IMU and the magnitude of the scale factors and biases. There is also no evidence within this research to suggest that FDE on IMU measurements would increase navigation accuracy or availability; the primary interests of pedestrian navigation.
6. Estimating the angular acceleration has a positive impact on the accuracy of the VIMU navigation solution, even more-so than the number of IMUs used. This is a very practical finding, which makes the use of a dual inertial system more attractive.
7. There is a linear dependence in the case of a VIMU operating with two IMUs. This eliminates the ability of a dual IMU nine parameter least-squares estimation algorithm. While only two axes of the angular acceleration can be estimated, there is potential for operation using an eight parameter estimation model.
8. Although systematic errors are not removed prior to the VIMU fusion, the least-squares approach and the adaptive filter approach provide an improvement over simple averaging with respect to noise reduction. Accuracy improvements are of the order of 9 to 10% per IMU added up for up to five IMUs added.
9. Time synchronization of the measurements is critical for the VIMU fusion methods, but not necessary for the other multi-IMU architectures. Time

synchronization for VIMU fusion is related to the dynamics of the foot and the noise properties of the IMU. For the test set up used herein, the ideal time synchronization should have been 0.02 ms or better, but was in reality 1 ms. When a large temperature variation was experienced, the clock timing error reached approximately 7 ms, which hindered the VIMU fusion and limited performance.

10. The accelerometer measurements of the VIMU require additional tuning for the Kalman filter. This stems from the fact that the gyro measurement noise decreases at a rate of $n^{-1/2}$, but the accelerometer measurement noise does not decrease at a similar rate. The noise reduction is a function of the lever arm (i.e. geometry between IMUs) and quality of the gyro measurements.
11. The elevation accuracy of the stacked and federated filters is much better than that of the VIMUs, often by several metres. The VIMU results can often have vertical biases resulting from limited filter tuning in the vertical axis.
12. Processing times of the filters differ, but the stacked filter requires the most processing time, followed by the federated filters, VIMU AKF, VIMU LSQ and VIMU average.

7.2 Recommendations

1. The federated zero reset filter was unable to operate at 1 Hz and provided poor overall performance. This could be theoretically improved upon by changing the information sharing principle to sequentially process one local filter at a time. This could provide another federated filter architecture for pedestrian navigation.
2. The estimated variances of the angular acceleration (and specific force and angular velocity) from the VIMU were not used as filter inputs to the VIMU filter. This information could be input into the navigation filter as a real time noise characteristic. More work could be done in comparing error behaviours of the individual IMUs and passing this information into the processing filter.
3. Because the IMUs were mounted on the foot, all filters benefited from the use of zero velocity updates. These updates managed the velocity component of the

filter. If the IMUs were rigidly mounted on the backpack, Doppler measurements could be used. This would result in more GPS dependent velocity solutions, but would be more practical for applications where the IMUs are not located on the foot.

4. A multi-IMU filter estimating one position, velocity and attitude and IMU errors for each IMU should be developed. This filter would essentially be a combined stacked and VIMU filter, where the raw IMU would be fused while the filter would properly estimate the bias and scale factor IMU error sources. This would facilitate IMU observation fault detection.
5. Inertial units that also measure the angular acceleration (in addition to specific force and angular velocity) would be ideally suited for a VIMU filter where the angular acceleration is estimated. This would provide another set of observations and more accurately estimate the angular acceleration. This would also allow for a dual IMU nine state least-squares estimation architecture that could estimate all values of the angular acceleration.
6. The performance of a multi-IMU system using tactical or navigation grade IMUs would provide interesting results that would provide better insight of the architectures performance. Since some error sources of the MEMS IMUs are not estimated or removed in full (i.e. non-orthogonality), these errors sources could induce errors into the solution. Using higher grade inertial units could provide a clearer insight into the estimation of the error states, their convergence rates, FDE performance and overall accuracy.
7. The use of varying types of IMUs together would provide interesting results. It is conceivable that users may have access to several IMU (or INSs) that could be used together to increase navigation accuracy. For example, using two or three tactical grade IMUs as a reference solution, rather than the best performing IMU, thereby increasing availability, accuracy and reliability.
8. The use of multiple GPS receivers is also worth of further investigation for multi-IMU systems. Multiple receiver (and antenna) systems could utilize attitude determination, multipath mitigation and real time receiver noise characteristics.

9. Several of the architectures could be adopted for multiple GPS receiver configurations. An analysis of the fusion results would be interesting, specifically estimating the amount of multipath and noise contained in a signal.

References

Allerton, D.J. and H. Jia (2002) "An Error Compensation Method for Skewed Redundant Inertial Configuration," *ION 58th Annual Meeting/CIGTF 21st Guidance Test Symposium*, Albuquerque, NM, pp. 142-147.

Allerton, D.J. and H. Jia (2005) "A Review of Multisensor Fusion Methodologies for Aircraft Navigation Systems," *The Journal of Navigation of the Royal Institute of Navigation*, 58, pp. 405-417.

Apostolyuk, V. (2006) "*Theory and Design of Micromechanical Vibratory Gyroscopes,"* MEMS/NEMS Handbook, Springer, Vol 1, pp 173-195.

Bancroft, J.B. (2009) "Multiple IMU Integration for Vehicular Navigation " *ION GNSS 2009*, Savannah, GA, pp. 13 pp.

Bancroft, J.B., G. Lachapelle, M.E. Cannon and M.G. Petovello (2008) "Twin IMU-HSGPS Integration for Pedestrian Navigation," *ION GNSS 2008*, Savannah GA, pp. 11.

Bancroft, J.B., G. Lachapelle, T. Williams and J. Garrett (2010) "GPS Observability and Availability for Various Antenna Locations on the Human Body," *ION GNSS 2010*, Portland, OR, pp. 11.

Beauregard, S. (2007) "Omnidirectional Pedestrian Navigation for First Responders," *4th IEEE Workshop on Positioning, Navigation, and Communication 2007 (WPNC'07)*, Hannover, Germany, pp. 33-36.

Brand, T. and T. Phillips (2003) "Foot-to-Foot Range Measurement as an Aid to Personal Navigation," *59th Annual Meeting and CIGTF 22nd Guidance Test Symposium*, Albuquerque, NM, pp. 113-121.

Brown, A. and M. Sturza (1990) "The Effect of Geometry on Integrity Monitoring Performance," *ION GPS 1990*, Atlantic City, NJ, pp. 121-129.

Brown, R.G. and P.Y.C. Hwang (1997) "*Introduction to Random Signals and Applied Kalman Filtering,"* 3rd Edition, John Wiley & Sons.

Carlson, N.A. (1990) "Federated Square Root Filter for Decentralized Parallel Processes," *IEEE Transactions on Aerospace and Electronic Systems*, 26, 3, pp. 517-525.

Carlson, N.A. (2002) "Federated Filter for Distributed Navigation and Tracking Applications," *ION 58th Annual Meeting/CIGTF 21st Guidance Test Symposium*, Albuquerque, NM, pp. 340-353.

Carlson, N.A. and M.P. Berarducci (1994) "Federated Kalman Filter Simulation Results," *NAVIGATION: Journal of The Institute of Navigation*, 41, 3, pp. 297-321.

Chui, D. and K.P. O'Keefe (2008) "Seamless Outdoor-to-Indoor Pedestrian Navigation using GPS and UWB," *ION GNSS 2008*, Savannah, GA, pp. 12.

Chui, D. and K.P. O'Keefe (2009) "*Seamless Outdoor-to-Indoor Positioning - DGPS + UWB,*" GPS World, March, 2009, pp. 32-38.

Colomina, I., M. Giménez, J.J. Rosales, M. Wis, A. Gómez and P. Miguelsanz (2004) "Redundant IMUs for Precise Trajectory Determination," *XXth ISPRS Congress* Istanbul, pp. 7.

Farrell, J.A. and M. Barth (1998) "*The Global Positioning System & Inertial Navigation,"* McGraw-Hill.

Foxlin, E. (2005) "Pedestrian Tracking with Shoe-Mounted Inertial Sensors," *IEEE Computer Graphics and Applications*, 25, 6, pp. 38-46.

Gao, J. (2007) "*Development of a Precise GPS/INS/On-Board Vehicle Sensors Integrated Vehicular Positioning System,"* PhD Thesis, Department of Geomatics Engineering, University of Calgary, Canada.

Gao, Y. (2008) "*Advanced Estimation Methods and Analysis, ENGO 629 Course Notes,*" Department of Geomatics Engineering, University of Calgary, Canada.

Gao, Y., E.J. Krakiwsky, M.A. Abousalem and J.F. McLellan (1993) "Comparison and Analysis of Centralized, Decentralized, and Federated Filters," *NAVIGATION: Journal of The Institute of Navigation*, 40, 8, pp. 69-86.

Gebre-Egziabher, D. (2002) "*Design and Performance Analysis of A Low-Cost Aided Dead Reckoning Navigator,"* PhD Thesis, Department of Aeronautics and Astronautics, Standord University, United States.

Gelb, A. (1974) "*Applied Optimal Estimation,"* The Massachusetts Institute of Technology Press.

Giroux, R., S. Sukkarieh and M. Bryson (2004) "Implementation of A Skew Redundant Low Cost INS in a Fast Prototyping Enviroment," *ION NTM 2004*, San Diego, CA, pp. 954-961.

Godha, S. (2006) "*Performance Evaluation of Low Cost MEMS-Based IMU Integrated with GPS for Land Vehicle Navigation Application,"* MSc Thesis, Department of Geomatics Engineering, The University of Calgary, Canada.

Godha, S. and G. Lachapelle (2008) "Foot mounted inertial system for pedestrian navigation," *Measurement Science and Technology*, 19, pp. 1-9.

Gold, R. (1967) "Optimal Binary Sequences for Spread Spectrum Multiplexing," *IEEE Transactions on Informtion Theory*, 13, 4, pp. 619-621.

Goldenberg, F. (2007) "Magnetic Heading, Achievements and Prospective," *ION NTM 2007*, San Diego CA, pp. 743-755.

Goldhirsh, J. and W.J. Vogel (1998) "*Handbook of Propagation Effects for Vehicular and Personal Mobile Satellite Systems - Overview of Experimental and Modeling Results,*" The Johns Hopkins University, Applied Physics Laboratory and The University of Texas at Austin, Electrical Engineering Research Laboratory, pp. 348.

GPS World Staff (2007) "*GPS Modernization Fits and Starts*," Accessed: Oct, 2010, http://www.gpsworld.com/gnss-system/news/gps-modernization-fits-and-starts-4396.

Grejner-Brzezinska, D.A., C.K. Toth, Y. Jwa and S. Moafipoor (2006) "Seamless and Reliable Personal Navigator," *ION NTM 2006*, Monterey, CA, pp. 597-603.

Grewal, M.S. and A.P. Andrews (2001) "*Kalman Filtering - Theory and Practice Using MATLAB,"* 2nd Edition, John Wiley & Sons.

Groves, P.D. (2008) "*Principles of GNSS, Inertial, and Multisensor Integrated Navigation Systems,"* Artech House.

Groves, P.D., G.W. Pulford, C.A. Littlefield, D.L.J. Nash and C.J. Mather (2007) "Inertial Navigation Versus Pedestrian Dead Reckoning: Optimizing the Integration," *ION GNSS 2007*, Forth Worth, TX, pp. 2043-2055.

Guerrier, S. (2008) "*Integration of Skew-Redundant MEMS-IMU with GPS for Improved Navigation Performance,"* Master Thesis, Geodetic Engineering Laboratory, Ecole Polytechnique Fédérale de Lausanne, Switzerland.

Hofmann-Wellenhof, B., K. Legat and M. Wieser (2003) "*Navigation - Principles of Positioning and Guidance,"* 1st Edition, Springer.

IEEE (2001) "IEEE Standard for Inertial Sensor Terminology (528-2001)."

IS-GPS-200E (2010) "Navstar GPS Spcae Segment/Navigation User Interface."

Kane, T.R. and D.A. Levinson (2005) "*Dynamics - Theory and Applications,"* Cornell University Library.

Kaplan, E.D. and C.J. Hegarty (2006) "*Understanding GPS Principles and Applications,"* 2nd Edition, Artech House.

Kasameyer, P.W., L. Hutchings, M.F. Ellis and R. Gross (2005) "MEMS-based INS Tracking of Personnel in a GPS-denied Environment," *ION GNSS 2005*, Long Beach 2005, pp. 949-955.

Kubo, Y. and J. Wang (2008) "INS/ GPS Integration Using Gaussian Sum Particle Filter," *ION GNSS 2008*, Savannah, GA, pp. 1345-1352.

Kuusniemi, H. (2005) "*User-Level Reliability and Quality Monitoring in Satellite-Based Personal Navigation,"* Doctor of Technology Thesis, Tampere University of Technology, Finland.

Kwakkel, S.P. (2008) "*Human Lower Limb Kinematics Using GPS/INS,"* MSc Thesis, Department of Geomatics Engineering, University of Calgary, Canada.

Kwakkel, S.P., S. Godha and G. Lachapelle (2007) "Foot and Ankle Kinematics During Gait Using Foot Mounted Inertial System," *ION NTM 2007*, San Diego, CA, pp. 1-9.

Kwakkel, S.P., G. Lachapelle and M.E. Cannon (2008) "GNSS Aided *In Situ* Human Lower Limb Kinematics During Running," *ION GNSS 2008*, Savannah, GA, pp. 1-10.

Lachapelle, G. (2007) "*Advanced GNSS Theory and Design, ENGO 625 Course Notes,*" Department of Geomatics Engineering, University of Calgary, Canada.

Lachapelle, G., O.Mezentsev, J. Collin and G. MacGougan (2003) "Pedestrian and Vehicular Navigation under Signal Masking using Integrated HSPG and Self Contained Sensor Technologies," *11th IAIN World Congress*, Berlin, pp. 1-29.

Li, Y., C. Rizos, J. Wang, P. Mumford and W. Ding (2008) "Sigma-Point Kalman Filtering for Tightly Coupoled GPS/INS Integration," *NAVIGATION: Journal of The Institute of Navigation*, 55, 3, pp. 167-177.

M827B Data Sheet (2006) "*M Series Magnetic GPS Antenna Data Sheet,*" Allis Communications Co., Ltd., pp. 11.

Marion, J.B. and S.T. Thornton (1995) "*Classical Dynamics of Paritcles and Systems,"* 4th Edition, Saunders College Publishing.

Mather, C.J., P.D. Groves and M.R. Carter (2006) "A Man Motion Navigation System Using High Sensitivity GPS, MEMS IMU and Auxiliary Sensors " *ION GNSS 2006*, Forth Worth, TX, pp. 2704-2714.

Mezentsev, O. (2005) "*Sensor Aiding of HSGPS Pedestrian Navigation,"* Thesis, Department of Geomatics Engineering, University of Calgary, Canada.

Misra, P. and P. Enge (2001) "*Global Positioning System - Signals, Measurements, and Performance,"* 1st Edition, Ganga-Jamuna Press.

Mohamed, A.H. and K.P. Schwarz (1999) "Adaptive Kalman Filtering for INS/GPS," *Journal of Geodesy*, 73, pp. 193-203.

Mutambara, A.G.O. (1998) "*Decentralized Estimation and Control for Multisensor Systems,"* CRC Press.

Olynik, M.C. (2002) "*Temporal Characteristics Of Gps Error Sources And Their Impact On Relative Positioning,"* MSc Thesis, Department of Geomatics Engineering, University of Calgary, Canada.

Osman, A., B. Wright, S. Nassar, A. Noureldin and N. El-Sheimy (2006) "Multi-Sensor Inertial Navigation Systems Employing Skewed Redundant Inertial Sensors," *ION GNSS 2006*, Fort Worth, TX, pp. 6.

Pejsa, A.J. (1974) "Optimum Skewed Redundant Inertial Navigators," *AIAA Journal*, 12, 7, pp. 899-902.

Petovello, M. (2003) "*Real-time Integration of a Tactical-Grade IMU and GPS for High-Accuracy Positioning and Navigation,"* PhD Thesis, Department of Geomatics Engineering, The University of Calgary, Canada.

Pittelkau, M.E. (2005) "Calibration and Attitude Determination with Redundant Inertial Measurement Units," *Journal of Guidance, Control, and Dynamcis*, 28, 4, pp. 743-752.

Pittelkau, M.E. (2006) "Cascaded and Decoupled RIMU Calobration Filters," *The Journal of the Astronautical Sciences*, 54, 3 & 4, pp. 449-466.

Prost, J.B., B. Godefroy and S. Terrenoir (2008) "*City Walk - Improving GPS Accuracy for Urban Pedestrians,*" GPS World, 19, pp. 32-37.

Renaudin, V., B. Merminod and M. Kasser (2008) "Optimal Data Fusion for Pedestrian Navigation based on UWB and MEMS," *IEEE Position Location and Navigation Symposium 2008*, Monterey, CA, pp. 341-349.

Renaudin, V., O. Yalak and P. Tomé (2007) "*Hybridization of MEMS and Assisted GPS for Pedestrian Navigation,*" InsideGNSS, January/February 2007, pp. 34-42.

Savage, P.G. (2007) "*Strapdown Analytics Part 1,"* 2nd Edition, Strapdown Associates Inc.

Schleppe, J.B. (1996) "*Development of a Real-Time Attitude System Using a Quaternion Parameterization and Non-Dedicated GPS Receivers,"* MSc Thesis, Department of Geomatics Engineering, The University of Calgary, Canada.

Schwarz, K.P. and M. Wei (1990) "Efficient Numerical Formulas for the Computation of Normal Gravity in a Cartesian Frame," *Manuscripta Geodaetica*, 15, 4, pp. 228-234.

Stirling, R., K. Fyfe and G. Lachapelle (2005) "Evaluation of a New Method of Heading Estimation for Pedestrian Dead Reckoning Using Shoe Mounted Sensors," *The Journal of Navigation of the Royal Institute of Navigation*, 58, pp. 31-45.

Sturza, M. (1988) "Navigation System Integrity Monitoring Using Redundant Measurements," *NAVIGATION: Journal of The Institute of Navigation*, 35, 4, pp. 69-87.

Sukkarieh, S., P. Gibbens, B. Brocholsky, K. Willis and H.F. Durrant-Whyte (2000) "A Low-Cost Redundant Inertial Measurement Unit for Unmanned Air Vehicles," *The International Journal of Robotics Research*, 19, pp. 1089-1103.

Sun, Z., X. Mao, W. Tian and X. Zhang (2009) "Activity classification and dead reckoning for pedestrian navigation with wearable sensors," *Measurement Science and Technology*, 20, pp. 1-10.

Tan, H., A.M. Wilson and J. Lowe (2008) "Measurement of stride parameters using a wearable GPS and inertial measurement unit," *Journal of Biomechnanics*, 41, pp. 1398-1406.

Teunissen, P.J.G. and M.A. Salzmann (1989) "A Recursive Slippage Test for Use in State-space Filtering," *Manuscripta Geodaetica*, 14, pp. 383-390.

Titterton, D.H. and J.L. Weston (2004) "*Strapdown Intertial Navigation Technology,"* 2nd Edition, The Institution of Electrical Engineers.

United States Naval Observatory (2010) "*GPS Constellation Status*," Accessed: Aug 2010, ftp://tycho.usno.navy.mil/pub/gps/gpstd.txt.

Waegli, A., S. Guerrier and J. Skaloud (2008) "Redundant MEMS-IMU integrated with GPS for Performance Assessment in Sports," *IEEE Position Location and Navigation Symposium 2008*, pp. 1260-1268.

Wei, M. and K.P. Schwarz (1990) "Testing a Decentralized Filter for GPS/INS Integration," *IEEE Position Location and Navigation Symposium 1990*, Las Vegas, NV, pp. 429-435.

Zhao, X., Z. Syed and N. El-Sheimy (2009) "A Portable Multi-sensor Navigation System for In-vehicle and On-foot Navigations," *ION ITM 2009*, Anaheim, CA, pp. 8.

APPENDIX A: DERIVATION OF VIMU DESIGN MATRIX

This appendix shows the partial derivatives with respect to the unknown parameters in Equation (5.4). For clarity the orientation superscripts and subscripts have been removed. Equation (5.4) is repeated here

$$\mathbf{f}_\mathrm{n} = \mathrm{R}_\mathrm{v}^\mathrm{n}\mathbf{f}_\mathrm{v} + \mathrm{R}_\mathrm{v}^\mathrm{n}\ \boldsymbol{\alpha}_\mathrm{v} \times \mathbf{r}\ + \mathrm{R}_\mathrm{v}^\mathrm{n}\ \boldsymbol{\omega}_\mathrm{v} \times\ \boldsymbol{\omega}_\mathrm{v} \times \mathbf{r}$$

$$\frac{\partial \mathbf{f}_n}{\partial \mathbf{f}_v} = R_v^n$$

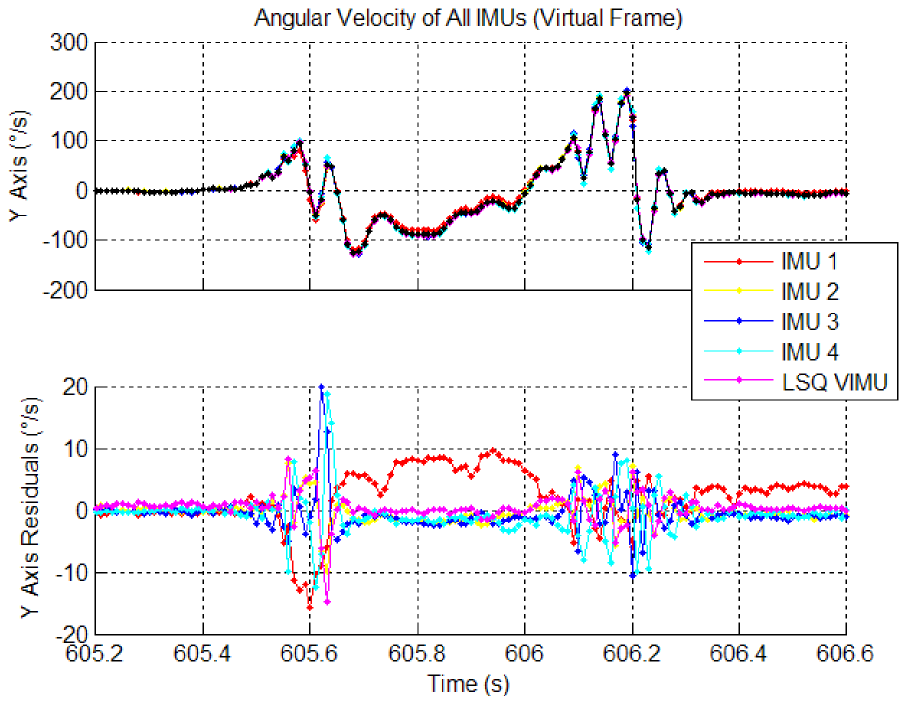
$$\begin{aligned}
\frac{\partial \mathbf{f}_n}{\partial \boldsymbol{\alpha}_v} &= \frac{\partial}{\partial \boldsymbol{\alpha}_v} R_v^n\ \ \boldsymbol{\alpha}_v \times \mathbf{r} \\
&= -R_v^n \frac{\partial}{\partial \boldsymbol{\alpha}_v}\ \ \mathbf{r} \times \boldsymbol{\alpha}_v \\
&= -R_v^n \frac{\partial}{\partial \boldsymbol{\alpha}_v}\ \ \ \mathbf{r} \times\ \boldsymbol{\alpha}_v \\
&= -R_v^n\ \ \mathbf{r} \times
\end{aligned}$$

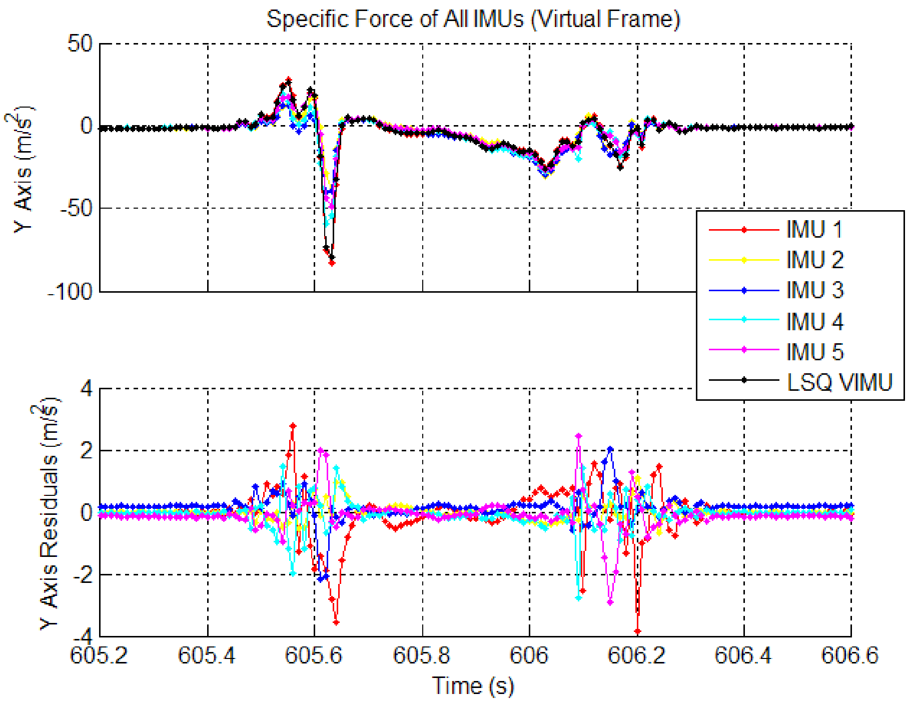
$$\begin{aligned}
\frac{\partial \mathbf{f}_n}{\partial \boldsymbol{\omega}_v} &= \frac{\partial}{\partial \boldsymbol{\omega}_v} R_v^n\ \ \boldsymbol{\omega}_v \times\ \boldsymbol{\omega}_v \times \mathbf{r} \\
&= R_v^n \frac{\partial}{\partial \boldsymbol{\omega}_v}\ \ \ \boldsymbol{\omega}_v \times\ \ \ \ \boldsymbol{\omega}_v \times \mathbf{r} \\
&= R_v^n \left(\left(\frac{\partial}{\partial \boldsymbol{\omega}_v}\ \ \boldsymbol{\omega}_v \times \right)\ \boldsymbol{\omega}_v \times \mathbf{r}\ +\ \boldsymbol{\omega}_v \times \left(\frac{\partial}{\partial \boldsymbol{\omega}_v}\ \ \boldsymbol{\omega}_v \times \mathbf{r} \right) \right) \\
&= -R_v^n\ \left[\ \boldsymbol{\omega}_v \times \mathbf{r}\ \ \times \right] +\ \boldsymbol{\omega}_v \times\ \ \mathbf{r} \times
\end{aligned}$$

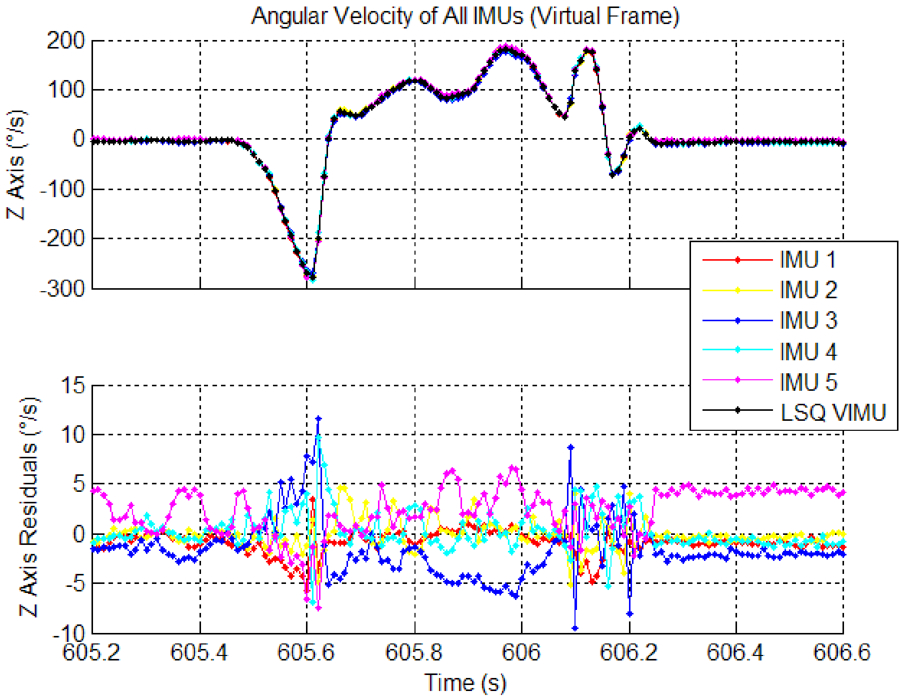
Note that

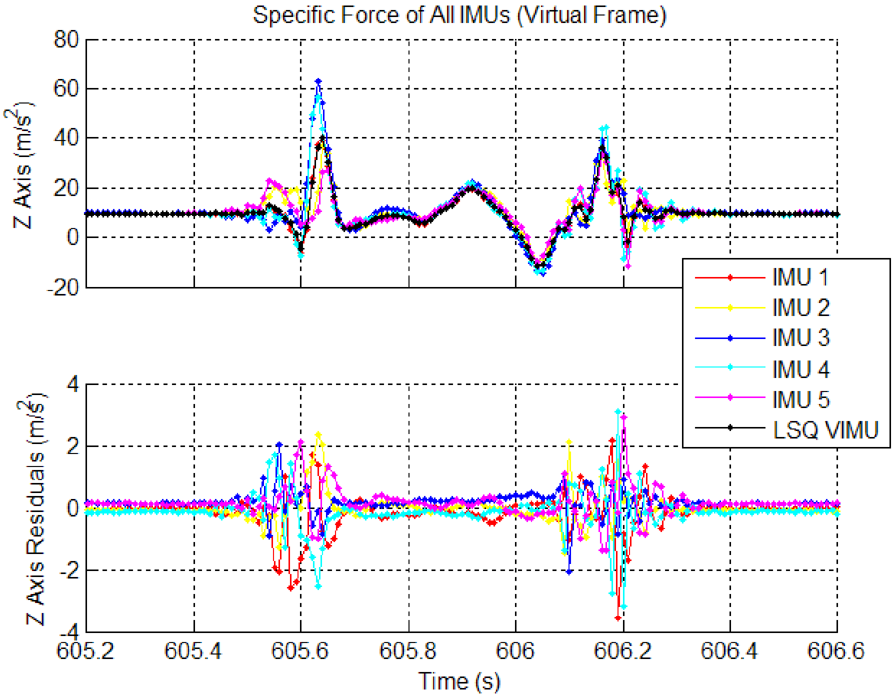
$$\begin{aligned}
\left(\frac{\partial}{\partial \boldsymbol{\omega}_v}\ \boldsymbol{\omega}_v \times \right)\ \boldsymbol{\omega}_v \times \mathbf{r}\ &= \left(\frac{\partial}{\partial \omega_{1v}}\ \boldsymbol{\omega}_v \times \right)\ \boldsymbol{\omega}_v \times \mathbf{r}\ + \left(\frac{\partial}{\partial \omega_{2v}}\ \boldsymbol{\omega}_v \times \right)\ \boldsymbol{\omega}_v \times \mathbf{r}\ + \left(\frac{\partial}{\partial \omega_{3v}}\ \boldsymbol{\omega}_v \times \right)\ \boldsymbol{\omega}_v \times \mathbf{r} \\
&= \left[\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \times \right]\ \boldsymbol{\omega}_v \times \mathbf{r}\ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}^T + \left[\begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \times \right]\ \boldsymbol{\omega}_v \times \mathbf{r}\ \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}^T + \left[\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \times \right]\ \boldsymbol{\omega}_v \times \mathbf{r}\ \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}^T \\
&= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} \boldsymbol{\omega}_v \times \mathbf{r} \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}^T \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix} \boldsymbol{\omega}_v \times \mathbf{r} \begin{bmatrix} \\ 1 \\ 0 \end{bmatrix}^T \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \boldsymbol{\omega}_v \times \mathbf{r} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}^T \\
&= \left[\ \boldsymbol{\omega}_v \times \mathbf{r}\ \times \right]
\end{aligned}$$

APPENDIX B: VIMU RESIDUALS






Angular Velocity of All IMUs (Virtual Frame)
Z Axis (°/s)
200
100
0
-100
-200
-300
IMU 1
IMU 2
IMU 3
IMU 4
IMU 5
LSQ VIMU
Z Axis Residuals (°/s)
15
10
5
0
-5
-10
605.2
605.4
605.6
605.8
606
606.2
606.4
606.6
Time (s)


Specific Force of All IMUs (Virtual Frame)
Z Axis (m/s^2)
80
60
40
20
0
-20
IMU 1
IMU 2
IMU 3
IMU 4
IMU 5
LSQ VIMU
Z Axis Residuals (m/s^2)
4
2
0
-2
-4
605.2
605.4
605.6
605.8
606
606.2
606.4
606.6
Time (s)