GENERIC MULTISENSOR INTEGRATION STRATEGY AND INNOVATIVE ERROR ANALYSIS FOR INTEGRATED NAVIGATION

KUN QIAN

A DISSERTATION SUBMITTED TO THE
FACULTY OF GRADUATE STUDIES IN
PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

GRADUATE PROGRAM IN EARTH AND SPACE SCIENCE
YORK UNIVERSITY
TORONTO, ONTARIO

APRIL 2017

Abstract

A modern multisensor integrated navigation system applied in most of civilian applications typically consists of GNSS (Global Navigation Satellite System) receivers, IMUs (Inertial Measurement Unit), and/or other sensors, e.g., odometers and cameras. With the increasing availabilities of low-cost sensors, more research and development activities aim to build a cost-effective system without sacrificing navigational performance. Three principal contributions of this dissertation are as follows:

i) A multisensor kinematic positioning and navigation system built on Linux Operating System (OS) with Real Time Application Interface (RTAI), York University Multisensor Integrated System (YUMIS), was designed and realized to integrate GNSS receivers, IMUs, and cameras. YUMIS sets a good example of a low-cost yet high-performance multisensor inertial navigation system and lays the ground work in a practical and economic way for the personnel training in following academic researches.

ii) A generic multisensor integration strategy (GMIS) was proposed, which features a) the core system model is developed upon the kinematics of a rigid body; b) all sensor measurements are taken as raw measurement in Kalman filter without differentiation. The essential competitive advantages of GMIS over the conventional error-state based strategies are: 1) the influences of the IMU measurement noises on the final navigation solutions are effectively mitigated because of the increased measurement redundancy upon the angular rate and acceleration of a rigid body; 2) The state and measurement vectors in the estimator with GMIS can be easily expanded to fuse multiple inertial sensors and all other types of measurements, e.g., delta positions; 3) one can directly perform error analysis upon both raw

sensor data (measurement noise analysis) and virtual zero-mean process noise measurements (process noise analysis) through the corresponding measurement residuals of the individual measurements and the process noise measurements.

iii) The *a posteriori* variance component estimation (VCE) was innovatively accomplished as an advanced analytical tool in the extended Kalman Filter employed by the GMIS, which makes possible the error analysis of the raw IMU measurements for the very first time, together with the individual independent components in the process noise vector.

Acknowledgments

I would like to thank everyone who has encouraged and supported me in this meaningful but long journey.

Among them, I would firstly like to thank my supervisor, Dr. Ing Jianguo Wang, for his great guidance and tremendous supports in both academic and financial perspectives. His expertise and enthusiasm upon the research have been motivating me throughout my whole Ph.D. study period. Also, I would like to thank my committee members Dr. Baoxin Hu and Dr. ZhengHong Zhu for their invaluable advices and supports. Special thanks go to the EOL laboratory at York University directed by Drs. Baoxin Hu and Jianguo Wang, and also Mr. William Chan at GeoPixel Air. Not the last, I would send my special acknowledgement to NSERC for partially supporting my research through Dr. Jianguo Wang's NSERC Discovery Grant Individual.

To my wife, thank you for your standing-by me all these years. Your love and support invigorates me every day. I owe you a big hug and a birthday gift.

To my beloved daughter, Ashely, wish you more happiness.

To my mother, "You made me who I am today, and I will always remember you in life's passings for no one could touch my life as you have." (Laura M. Phipps-Kelley, "Remembrance" 14 -16).

Kun Qian

Toronto, February 2017

Table of contents

List of Tables

List of Figures

List of Acronyms

C/A	Coarse Acquisition
CRLB	Cramer-Rao Lower Bound
DCM	Direction cosine matrix
DD	Double differenced
DIAPM	Dipartimento di Ingegneria Aerospaziale - Politecnico di Milano
DMI	Distance measurement instrument
DR	Dead reckoning
ECEF	Earth centered Earth fixed
EKF	Extended Kalman Filter
ENU frame	East, North, Up Cartesian frame
ErrINSEKF	Error-state based inertial navigation system using EKF
FOG	Fiber optic gyroscope
GLONASS	Global Navigation Satellite System
GMIKF	Generic Multisensory integration Kalman Filter
GMIS	Generic multi-sensor integration strategy
GNSS/INS	GNSS aided INS
GPS	Global Positioning System
GNSS	Global Navigation Satellite System
IMU	Inertial Measurement Unit
INS	Inertial Navigation System
IPC	Inter-process communication
LC	Loosely-Coupled

LKF	Linearized Kalman filter
LS	Least squares
LXRT	Linux Real time
MEMS	Micro-electro-mechanical systems
MMS	Mobile Mapping System
MTBF	Mean time between failure
NED frame	North, East, Down Cartesian frame
NLLS	Non-linear least square
OS	Operating system
PFIX	Position fix
PPS	Pulse per second
RTAI	Real Time Application Interface
RW	Random walk
RWWN	Random walk white noise
SINS	Strapdown Inertial Navigation System
TC	Tightly-Coupled
UAV	Unmanned aerial vehicles
VCE	Variance component estimation
WGS	World Geodetic System
YUMIS	York University Multisensor Integrated System
ZUPT	Zero velocity update

List of Symbols

$\mathbf{v}^a$	a vector in ***a*** frame
C_a^b	Transformation matrix between from ***a*** frame to ***b*** frame
ψ	Yaw
ϕ	Roll
p	Pitch
$\partial\mathbf{r}/\partial\mathbf{v}$	Partial derivative of vector **r** with respect to vector **v**
$\boldsymbol{\omega}$	Angular rate/velocity vector
$\times$	Cross product operator
$\mathbf{R}_{a(k)}^{a(k+1)}$	Transformation matrix which transform the frame ***a*** at epoch *k* to the frame ***a*** at epoch *k*+1
$[\mathbf{a}\times]$	3 x 3 Skew matrix of a vector **a**
ϕ	Carrier phase measurement
$\dot{\phi}$	Doppler measurement
ρ	Pseudo-range measurement
$\dot{\mathbf{a}}$	The first order derivative of a vector **a** with respect to time *t*
$\ddot{\mathbf{a}}$	The second order derivative of a vector **a** with respect to time *t*
σ	Standard deviation of a random variable
$\int x\,ds$	Integral of a variable *x*
$\lvert\mathbf{a}\rvert$	Norm of a vector **a**
a	Acceleration of a particle
b	Sensor body frame

d**x**/dt	Derivative of x vector with respect to the time *t*
e	Earth centered Earth fixed (ECEF) frame
F	Total force vector
g	Local geodetic frame
H	Measurement Jacobian matrix/the design matrix of the measurement model
i	Inertial reference frame
m	Mass of a particle
n	Navigation frame
N	Ambiguity integer
p	Probability of a random variable
P	Covariance matrix
Θ	quaternion
Q	covariance matrix of the process noise vector
R	DCM matrix
r	Position vector
u	Rotation vector
v	Vehicle body frame
v	Velocity vector
x	vector (bold)

1. Introduction

1.1 Background and motivation

"Navigation is the science of getting ships, aircraft, or spacecraft from place to place; especially: the method of determining position, course, and distance traveled." [Merriam-Webster, *online*]. The modern navigation starts from the system integrating various electronic sensing devices [Ismaeel, 2003] to determine position, course, and distance traveled. Among all navigation systems, the inertial navigation system (INS) is the most prevalent modern navigation system because of its accuracy, long mean time between failure (MTBF), and self-reliance [Bekir, 2007].

A modern aided inertial integrated navigation system is nothing but a digital computer attached with an inertial measurement unit (IMU) and all other aiding sensors. The central computer is responsible for computing the "continuous" navigation solution including position, velocity and attitude of the vehicle based on the outputs from the inertial sensor while the aiding sensors provide information to estimate and/or suppress the errors in navigation solution. Because of four decades of continuous efforts in the semiconductor and computer industry, the device miniaturization and the ever-growing computation power have revolutionarily transformed the size and the weight of an IMU unit. Accordingly, the high-speed yet affordable microprocessors make it possible the old-fashion bulky gimbaled INS system to be replaced by the lightweight strapdown INS system (SINS) [Lawrence, 1998]. Secondly, the advances in sensor technology, e.g., fiber optic gyroscope (FOG) and micro-electro-mechanical systems (MEMS) technologies, have further brought out the opportunities for the low-cost SINS in many small-sized applications

such as personal navigation, car-navigation, unmanned aerial vehicles (UAVs) and so on [Hasan, 2009].

Although the high-end INS system could maintain stable and satisfactory performance within a rather long period, the remarkable expenses prohibited its application on those small-size civilian systems [Gao, 2007]. Therefore, in the early development stages of inertial navigation technology, the rapidly growing errors in a low-cost navigation system cannot be well controlled merely using the time-intermittent zero velocity updates (ZUPTs) and/or the static position fixes (PFIXs). It is after the full deployment of the Global Navigation Satellite System (GNSS) that the low-cost INS system becomes thriving because that the all-weather 24/7 GNSS system is able to provide the reliable position, velocity and the heading solution with a reasonable degree of accuracy [Bekir, 2007] so that the quickly growing errors in an INS system is effectively suppressed. In other words, the fusion of the complementary GNSS and inertial navigation technologies enables a continuous, high-bandwidth, complete navigation solution with long- and short-term high accuracy [Groves, 2008; Antonio, 2010].

Nowadays, as the major beneficiary of the GNSS/INS technology, the modern kinematic Mobile Mapping System (MMS) is able to work in a fast, cost-effective and accurate manner to geo-reference 3D geospatial objects, such as natural landmarks, roads, targets, images or point clouds even without the use of Ground Control Points (GCPs) which usually consume great amount of time and efforts. Consequently, MMS for the applications in remote sensing/photogrammetry, gravimetry, laser scanning with airborne, land and marine vehicles have become an emerging trend in mapping applications [El-Sheimy, 2007]. In MMS, the positioning and navigation performance relies on the core navigation and positioning engine [Skaloud, 2002; Lopez, 2003; Yudan Yi, 2007], which fuses all sensor data in order to achieve the optimal

estimates of positions, velocities and attitudes for a rigid body of interest. Therefore, the researches about multisensor (e.g., GNSS receivers, IMUs, etc.) integrated navigation receive lots of interests in mobile mapping community [Bossler et al., 1991; Schwarz, et al, 1993; Grejner-Brzezinska, 1997; Tao, 2000; Yudan Yi, 2007; Jaakkola, A., 2010; Wu, D. J. et al., 2013; Karasaka, L., 2013; Munguía, R. et al, 2016; Nilsson, et al, 2016 and etc.]. The research activities in the integrated navigation mainly focus on:

- Improvements of *a priori* stochastic error models of inertial sensors,
- Improvements of INS system initialization and alignment algorithms,
- Improvements of INS mechanization algorithms,
- Applications of the advanced non-linear filtering technique,
- Improvements of GNSS-aided inertial navigation integration strategies.

From author's point of view, the GNSS and inertial navigation integration strategy receives the least interest among all above listed topics, as Farrell [1995] pointed out that the most probable reason for this fact is that the conventional method is widely accepted and hence may inhibit further progress when working with low performance MEMS inertial sensors. Originated from the optimal error control and estimation theory, the conventional integration strategy is focused on the calibration/suppress the errors of the primary INS incurred from the initial misalignment, position errors, as well as systematic and random errors of the primary inertial sensor. First, in the conventional implementation, our *a priori* knowledge about the rigid body's kinematics is often overlooked or not properly utilized. Second, the support for multiple IMU units in the traditional integration strategy becomes a complicate issue because that the error regulation aided INS requires one instance of the algorithm for each IMU. Therefore, the author is motivated to make a significant changeover in the multisensor integration strategy because one may not be

able to reach an innovative and cost-effective breakthrough if one keeps staying in the traditional rut [Wagner and Wieneke, 2003]. Besides the innovation of the multisensor integration strategy, the further accuracy improvements of the navigation solution give rise to the subsequent objective in this dissertation: tuning the stochastic models in the estimation filter for inertial sensors and other participating sensors.

In addition to the exploration of those theoretical algorithms, the author also needs to address the challenge of developing a high-performance yet economically feasible (low-cost) multisensor positioning and navigation system, which leads to the first objective introduced in Section 1.2.

1.2 Objectives of the dissertation

With the conventional integration mechanism [Jekeli, 2001; Rogers, 2003; Titterton, 2004; Bekir, 2007], the performance of an INS system subject to the quality of the IMU sensor rapidly deteriorates when the system lacks of sufficient and accurate aiding data, e.g., in GNSS degraded or denied environments. This dissertation primarily aims at innovating a generic multisensor integration strategy (GMIS) with the full usage of *a priori* knowledge of a rigid body's kinematics for a moving vehicle (or plateform) so that the rapidly deteriorated navigation performance can significantly be improved in comparison with the conventional integration strategies. The objectives of this dissertation are to:

1) Design and develop a hard real-time multisensor integrated system (YUMIS) which can collect and time-tag data from GNSS receivers, IMU unit, and 1394 [1394 Trade Association, 2010] cameras. YUMIS sets a good example of a low-cost yet high-performance multisensor inertial navigation system for the purpose of the follow-up

research upon navigation algorithms and lays the ground work in a practical and economic way for the personnel training in following academic researches

2) Derive and develop a GMIS-based Kalman filter to estimate navigational parameters with the full usage of *a priori* knowledge of a rigid body's kinematics.
3) Tune and evaluate the stochastic models of random errors for all participating sensors within the estimation filter by means of the variance component estimation (VCE) technique in Kalman filter.

1.3 Structure of the dissertation

This dissertation is structured as follows.

Following the introduction in this chapter, Chapter 2 briefs the preliminary knowledge of multisensor integrated inertial navigation and further provides the literature review.

Chapter 3 describes the design and development of an in-house low-cost multisensor system, York University Multisensor Integrated System (YUMIS), based on Linux with Real Time Application Interface (Linux/RTAI).

Chapter 4 presents the generic multisensor integration strategy (GMIS), the proposed extended Kalman filter (GMIKF) and the post-processed navigation solutions that demonstrate the success of the novel integration strategy. Besides, the comparisons of the filter structure and the performance boundaries are also given to show the advantages of GMIS.

Chapter 5 details the *a posteriori* variance component estimation algorithm as an advanced analytical tool for the purpose of the random error analysis. The stochastic models of random errors attached with inertial sensors and process noises in GMIKF are shown to be successfully re-established through the real datasets.

Chapter 6 presents the experimental results of a real dataset associated with specific discussions and analysis.

Chapter 7 concludes the entire work of this dissertation. Recommendations for future work and certain remarks are also summarized at the end.

2. Preliminaries and related literature

In this chapter, the preliminaries and the principles of strapdown inertial navigation are provided as the knowledge base of the follow-up chapters in this dissertation. Moreover, the literatures review is conducted to present the theoretical advances and the state-of-art in the modern strapdown inertial navigation. In addition, the historical works related to the problems tackled in this dissertation (integration strategy and random error analysis) are also summarized. Without further notice, the inertial navigation in this dissertation will be always referred to the strapdown inertial navigation in consideration of the scope of this dissertation.

2.1 Coordinate frame

2.1.1 Cartesian frame

The most common three-dimension Cartesian coordinate frame in navigation is a reference frame consisting of three mutually orthogonal base vectors (axes). The 3D-Cartesian coordinate frame can be defined as a right-handed system or a left-handed system. The Figure 2.1 depicts the differences between two frames. Because the right-handed system is dominant in modern navigation, all coordinate systems introduced in this dissertation are right-handed without further notice. In consideration of the nature of space recognized by human being, the position and velocity of a rigid body are usually represented by a 3×1 column vector.

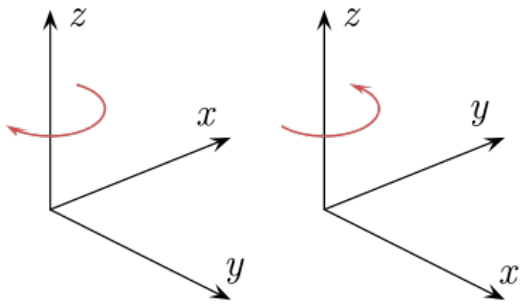


Figure 2-1 The left-handed and right-handed Cartesian coordinate systems

2.1.2 Coordinate frames

The most widely used coordinate frames in modern inertial navigation near the surface of the Earth are:

- Inertial reference frame (***i***)

  Inertial frame is an eternal stationary Cartesian coordinate frame which is either stationary in space or moving at constant velocity (i.e. no acceleration). The frame used in inertial navigation is the Earth-centered inertial (ECI) frame [Grewal et al. 2007; Noureldin et al, 2013]. In inertial navigation, measurements from inertial sensors are physical quantities with respect to inertial frame.

- Earth Center Earth Fixed (ECEF) frame (***e***)

  Earth fixed reference frame has fixed orientation with respect to the rotating Earth. For example, most of the paper map with the legend of North and East directions for personal navigation are actually Earth fixed reference frames.

  ECEF (***e***) is an Earth fixed reference frame with its origin being the center of the Earth. The X-axis is located in the equatorial plane and points toward the mean Greenwich Meridian. The Z-axis is pointing towards the north (along the spin axis of the Earth) being

perpendicular to the X-Y plane. The Y-axis is 90 degrees east of the mean Greenwich Meridian to complete the 3D Cartesian system [Wang, 2003; Gao 2007].

- Local geodetic frame (or local geographic frame) (***g***)

  Local geodetic frame is a topocentric frame with three axes pointing to the local east, north and zenith. Since the height is preferred to be positive when the point is above the Earth surface, East-North-Up (ENU) Cartesian system is more adopted than North-East-Down (NED) system in Geodesy. On the other hand, NED frame is more preferred in navigation because the vehicle's heading azimuth is the same as the yaw angle around DOWN axis when NED frame is used. The local geodetic frame is also known as the local level frame or local navigation frame [Noureldin et al, 2013].

- Navigation frame (***n***)

  Inertial navigation mechanization [Schmidt et al, 1973; Heller, 1975] is conducted in navigation frame. In practical implementation, the ***n*** frame can be realized as a variant of the local ENU frame. For example, the ***n*** frame can deviate from the local ENU frame by certain time-variant amount of rotation around its Z axis. Three popular navigation frames in navigation calculation are due North, free azimuth and wander angle and they are distinguished from each other through the amount of the rotation angle.

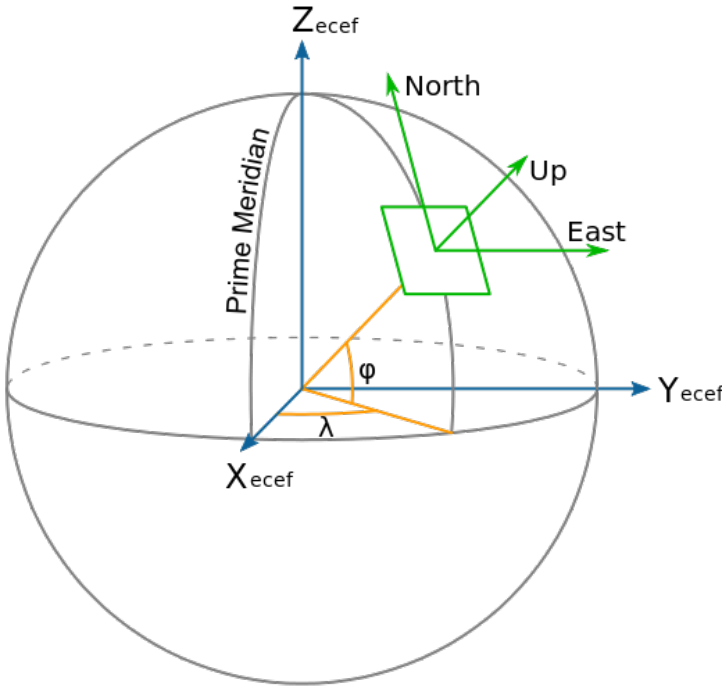


Figure 2-2 ECEF frame and Local Geodetic ENU frame

- Vehicle's body frame (***v***)

  The vehicle's body frame is the coordinate frame rigidly fixed to the vehicle's body. Similar to ENU and NED local geodetic frames, two popular frames are Forward-Right-Down and Right-Forward-Up. Accordingly, if a leveled vehicle is moving toward due north, the Right-Forward-Up frame will coincide with ENU local geodetic frame.

- Sensor's body frame (***b***)

  The sensor's body frame is the 3D Cartesian coordinate frame referred by the sensor measurements. The misalignment angles between frame ***b*** and frame ***v*** are usually named as the sensor mounting (boresight) angles. In this thesis, frame ***b*** coincides with frame ***v*** for the purpose of simplicity.

2.2 Rotation matrix

2.2.1 Overview

The coordinates of a vector of interest are resulted from the projection of the vector onto the base axes of the coordinate system. A single physical vector could take different coordinates in different coordinate systems and the transformation matrix is the tool used to transform the coordinates of a vector from one frame to another [Rogers, 2003]. Let $\mathbf{v}^a$ be a vector in frame ***a*** and $\mathbf{v}^b$ be its counterpart in frame ***b***, the transformation from $\mathbf{v}^a$ to $\mathbf{v}^b$ is achieved through the transformation matrix $\mathbf{C}_a^b$ as follows:

$$\mathbf{v}^b = \mathbf{C}_a^b \mathbf{v}^a \tag{2.2.1}$$

In a three dimensional Cartesian coordinate system, the transformation matrix $\mathbf{C}_a^b$ (from ***a*** to ***b***) would be always of the size of 3×3. Therefore, the transformation matrix $\mathbf{C}_a^b$ in this dissertation shall always be a nonsingular square matrix of the size of 3×3 without further notice. The transformation matrix incurred from the frame rotation becomes a rotation matrix which must be an orthonormal matrix.

Assume that $\mathbf{C}_a^b$ (stands for $\mathbf{C}_a^b(t)$) be the rotation matrix from frame ***a*** to frame ***b*** at time *t*, the time derivative of $\mathbf{C}_a^b$ implicates the dynamics of frame ***b*** with respect to frame ***a***. Among the various mathematical decompositions of $\mathbf{C}_a^b(t)$, the most popular four approaches are: DCM, Euler angles, Rotation vector and Quaternion [Shuster, 1993; Bekir, 2007].

2.2.2 Direction cosine matrix

A direction cosine matrix (DCM) is one particular interpretation of a rotation matrix. Any rotation matrix can be casted as a DCM by interpreting the matrix columns as inner products and hence direction cosines between the unit orthogonal axes of the two coordinate frames. Namely, a DCM consists of nine elements resulted from the dot product of the unit base vectors in two

coordinate systems. It is proved to be a transformation matrix [Bekir, 2007] and has its differential equation as follows [Titterton, 2004]:

$$\dot{\mathbf{C}}_{a}^{b} = \mathbf{C}_{a}^{b}\tilde{\boldsymbol{\omega}}_{ba}^{a} \tag{2.2.2}$$

wherein the vector $\boldsymbol{\omega}_{ba}^{a}$ denotes the instantaneous angular velocity of frame ***a*** with respect to frame ***b*** projected on frame ***a*** and the tilde notation denotes skew symmetric matrix of the vector.

2.2.3 Euler angles

According to Euler's theory, three Euler-angles can also describe the orientation of a rigid body in three dimensional space [Euler, 1776], using which the transformation matrix between the vehicle's body frame and the local geodetic frame can be achieved through the product of three sequential simple rotations each of which is around one instantaneous principle axis [Slabaugh, 1999]. Correspondingly, three associated scalar angles are called Euler angles, which could be defined in different ways. One of the most widely used sequential three rotations from frame ***n*** **(ENU)** to frame ***b***, called Tait-Bryan or Cardan angles [Rogers, 2003; Ardakani & Bridges, 2010; etc.] is defined as the yaw angle around z (Up) axis (ψ), the pitch angle around x (East) axis (p) and the roll angle around y (North) axis (ϕ). Accordingly, the DCM matrix resulted from above three rotations is as follows:

$$\begin{aligned}\mathbf{C}_{n}^{b} &= \begin{bmatrix} \mathrm{c}\gamma & 0 & -\mathrm{s}\gamma \\ 0 & 1 & 0 \\ \mathrm{s}\gamma & 0 & \mathrm{c}\gamma \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 \\ 0 & \mathrm{cp} & \mathrm{sp} \\ 0 & -\mathrm{sp} & \mathrm{cp} \end{bmatrix}\begin{bmatrix} \mathrm{c}\psi & -\mathrm{s}\psi & 0 \\ \mathrm{s}\psi & \mathrm{c}\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} \mathrm{c}\gamma\,\mathrm{c}\psi + \mathrm{s}\gamma\,\mathrm{s}\psi\,\mathrm{sp} & -\mathrm{c}\gamma\,\mathrm{s}\psi + \mathrm{s}\gamma\,\mathrm{c}\psi\,\mathrm{sp} & -\mathrm{s}\gamma\,\mathrm{cp} \\ \mathrm{s}\psi\,\mathrm{cp} & \mathrm{c}\psi\,\mathrm{cp} & \mathrm{sp} \\ \mathrm{s}\gamma\,\mathrm{c}\psi - \mathrm{c}\gamma\,\mathrm{s}\psi\,\mathrm{sp} & -\mathrm{s}\gamma\,\mathrm{s}\psi - \mathrm{c}\gamma\,\mathrm{c}\psi\,\mathrm{sp} & \mathrm{c}\gamma\,\mathrm{cp} \end{bmatrix}\end{aligned} \tag{2.2.3}$$

wherein are

p, γ, ψ three Euler angles (pitch, roll, heading),

cx, sx the cosine and sine functions of a scalar variable x,

Then,

$$\mathbf{C}_{\mathrm{b}}^{\mathrm{n}} = \mathbf{C}_{\mathrm{n}}^{\mathrm{b}^{\mathrm{T}}} = \begin{bmatrix} \mathrm{c}\gamma\,\mathrm{c}\psi + \mathrm{s}\gamma\,\mathrm{s}\psi\,\mathrm{sp} & \mathrm{s}\psi\,\mathrm{cp} & \mathrm{s}\gamma\,\mathrm{c}\psi - \mathrm{c}\gamma\,\mathrm{s}\psi\,\mathrm{sp} \\ -\mathrm{c}\gamma\,\mathrm{s}\psi + \mathrm{s}\gamma\,\mathrm{c}\psi\,\mathrm{sp} & \mathrm{c}\psi\,\mathrm{cp} & -\mathrm{s}\gamma\,\mathrm{s}\psi - \mathrm{c}\gamma\,\mathrm{c}\psi\,\mathrm{sp} \\ -\mathrm{s}\gamma\,\mathrm{cp} & \mathrm{sp} & \mathrm{c}\gamma\,\mathrm{cp} \end{bmatrix} = \begin{bmatrix} \mathrm{C}_{11} & \mathrm{C}_{12} & \mathrm{C}_{13} \\ \mathrm{C}_{21} & \mathrm{C}_{22} & \mathrm{C}_{23} \\ \mathrm{C}_{31} & \mathrm{C}_{32} & \mathrm{C}_{33} \end{bmatrix} \tag{2.2.4}$$

The calculation of the attitude vector $\boldsymbol{\theta}$ can be realized from the DCM matrix $\mathrm{C}_{\mathrm{b}}^{\mathrm{n}}$ as follows:

$$\boldsymbol{\theta} = \begin{bmatrix} p \\ \gamma \\ \psi \end{bmatrix} = \begin{bmatrix} \arcsin(\mathrm{C}_{32}) \\ \arctan(-\mathrm{C}_{31}, \mathrm{C}_{33}) \\ \arctan(\mathrm{C}_{12}, \mathrm{C}_{22}) \end{bmatrix} \tag{2.2.5}$$

The differential equation of Euler angle with respect to the angular velocity components in **ENU** frame is given as [Magnus, 1971; Bekir, 2007]:

$$\begin{bmatrix} \dot{p} \\ \dot{\gamma} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} c\gamma & 0 & s\gamma \\ s\gamma\,tgp & 1 & -c\gamma\,tgp \\ s\gamma\sec p & 0 & -c\gamma\sec p \end{bmatrix} \begin{bmatrix} \omega_{nbx}^{b} \\ \omega_{nby}^{b} \\ \omega_{nbz}^{b} \end{bmatrix} = C_{3\times3}\boldsymbol{\omega}_{nb}^{b} \tag{2.2.6}$$

wherein are

tgx the tangent function of a scalar variable x,

$\sec x$ the secant function of a scalar variable x.

2.2.4 Rotation vector

A rotation vector (3×1) mathematically defines a rotation through a unit vector representing rotation axis and a scalar representing the angular rotation about this axis [Shuster, 1993]. Given $\mathbf{u}$ is a unit vector and ϕ is the scalar, the rotation vector is defined by $\mathbf{\Phi} = \phi \mathbf{u}$. The corresponding differential equation for rotation vector is given by [Bortz, 1971; Shuster, 1993; Bekir, 2007]:

$$\dot{\bar{\varphi}} = \boldsymbol{\omega} + \frac{1}{2}\bar{\varphi} \times \boldsymbol{\omega} + \frac{1}{\phi^2}\left(1 - \frac{\phi}{2}\cot\left(\frac{\phi}{2}\right)\right)\bar{\varphi} \times (\bar{\varphi} \times \boldsymbol{\omega}) \tag{2.2.7}$$

2.2.5 Quaternion

The four-element quaternion is an alternative to the three-element rotation vector [Kuipers, 1999; Kong, 2000]. It is defined by one scalar component q_0 and one vector $\mathbf{q}$ of 3 components. The quaternion Θ can be equivalently presented by scalar angle ϕ about the unit vector $\mathbf{u}$ as follows:

$$\Theta = \{q_0, \quad \mathbf{q}\} = \left\{c\left(\frac{\phi}{2}\right), \quad s\left(\frac{\phi}{2}\right)\mathbf{u}\right\} \tag{2.2.8}$$

In a modern digital computer with certain round-off errors, the attitude update through the quaternion technique is numerically more accurate because it performs the computation using half magnitude of the angular rate in comparison with the rotation vector method or DCM matrix. However, this dissertation still uses Euler angle approach because of its instinctiveness. The differential equation of the quaternion is [Zhe, 1985; Vathsal, 1991; Kong, 2000; Bekir, 2007]:

$$\frac{d}{dt}\begin{bmatrix} q_0 \\ \mathbf{q} \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 0 & -\boldsymbol{\omega}' \\ \boldsymbol{\omega} & -\tilde{\boldsymbol{\omega}} \end{bmatrix}\begin{bmatrix} q_0 \\ \mathbf{q} \end{bmatrix} \tag{2.2.9}$$

2.3 Particle kinematics

2.3.1 Law of motion for a particle

Newton's three laws of motion stated [Isaac Newton, 1687]

i. Every body continues in its state of rest, or of uniform motion in a straight line, unless compelled to change that state by forces acting upon it.

ii. The time rate of change of linear momentum of a body is proportional to the force acting upon it and occurs in the direction in which the force acts.

iii. To every action there is an equal and opposite reaction; that is, the mutual forces of two bodies acting upon each other are equal in magnitude and opposite in direction.

The first two of Newton's laws, as applied to a particle, can be summarized by the law of motion:

$$\boldsymbol{F} = m\mathbf{a} \tag{2.3.1}$$

wherein $\boldsymbol{F}$ is the total force applied to the particle of mass inclusive of both direct contact and field forces such as gravity or electromagnetic forces, m is the mass of the particle, and $\mathbf{a}$ is the acceleration of the particle that must be measured relative to an inertial or Newtonian frame of reference.

Newton's law of motion for a particle reveals the relationship between the external force and the acceleration of the particle. Therefore, it is Newton's law of motion makes possible navigation through measures of specific force.

2.3.2 Kinematics

2.3.2.1 Position, velocity and acceleration

Mathematically, the location of the point *P* at time *t* in Figure 2-3 is described by vector $\vec{P}(t)$ which depicts the time function of the position vector between a point *P* and the origin *O*. In general, one has

$$\mathbf{r} = \vec{P}(t) = [x(t), y(t), z(t)]^T = x(t)\vec{\mathbf{i}} + y(t)\vec{\mathbf{j}} + z(t)\vec{\mathbf{k}} \tag{2.3.2}$$

wherein

$x(t)$ is the projection of position vector $\vec{P}(t)$ on axis x

$y(t)$ is the projection of position vector $\vec{P}(t)$ on axis y

$z(t)$ is the projection of position vector $\vec{P}(t)$ on axis z

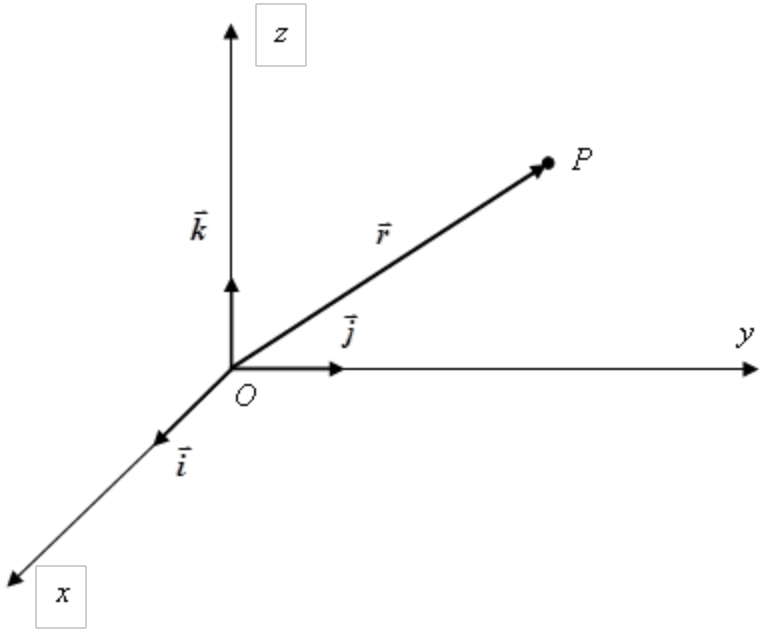


Figure 2-3 Position vector in a 3D Cartesian coordinate system

The velocity vector of point *P* projected into the same static coordinate frame *Oxyz* is defined as the derivative of the point function with respect to time, is given as

$$\mathbf{v} = \vec{V}(t) = \frac{d\vec{P}(t)}{dt} = [x'(t), y'(t), z'(t)]^T \tag{2.3.3}$$

wherein

$x'(t)$ is the first order derivative of $x(t)$ w.r.t. time t

$y'(t)$ is the first order derivative of $y(t)$ w.r.t. time t

$z'(t)$ is the first order derivative of $z(t)$ w.r.t. time t

Similarly, the acceleration vector is defined as the first order derivative of velocity vector in the same static coordinate frame *Oxyz* with respect to time:

$$\mathbf{a} = \vec{A}(t) = \frac{d\vec{V}(t)}{dt} \tag{2.3.4}$$

2.3.2.2 Velocity in a moving frame

In Equation (2.3.3), three base axes of the reference frame *Oxyz* are assumed to be static. However, when these three axes ($\vec{i}$, $\vec{j}$, $\vec{k}$) are also rotating with respect to an inertial reference frame, the velocity vector relative to the inertial reference frame becomes:

$$\mathbf{v} = \frac{\mathrm{d}\mathbf{r}}{\mathrm{dt}} = x'(t)\vec{\mathbf{i}} + y'(t)\vec{\mathbf{j}} + z'(t)\vec{\mathbf{k}} + x(t)\vec{\mathbf{i}}' + y(t)\vec{\mathbf{j}}' + z(t)\vec{\mathbf{k}}' \tag{2.3.5}$$

Alternatively, the right hand side of Equation (2.3.5) can be broken down into two parts: the velocity vector $\mathbf{v}_s = \mathrm{x}'(\mathrm{t})\vec{\mathbf{i}} + \mathrm{y}'(\mathrm{t})\vec{\mathbf{j}} + \mathrm{z}'(\mathrm{t})\vec{\mathbf{k}}$ as if the coordinate frame *Oxyz* is stationary and $\mathbf{v}_r = \mathrm{x}(\mathrm{t})\vec{\mathbf{i}}' + \mathrm{y}(\mathrm{t})\vec{\mathbf{j}}' + \mathrm{z}(\mathrm{t})\vec{\mathbf{k}}' = \boldsymbol{\omega} \times \mathbf{r}^{\mathbf{i}}$ representing the velocity of the coordinate frame *Oxyz* relative to an inertial reference frame [Bekir, 2007]. Thus, Equation (2.3.5) can be rewritten into

$$\mathbf{v} = \frac{\mathrm{d}\mathbf{r}}{\mathrm{dt}} = \mathbf{v}_s + \boldsymbol{\omega} \times \mathbf{r}^{\mathbf{i}} \tag{2.3.6}$$

2.3.2.3 Fundamental navigation equations

Based on (2.3.6), the velocity of a particle with respect to the rotating Earth expressed in the inertial frame (***i***) is

$$\mathbf{v}_{\mathbf{ev}}^{\mathbf{i}} = \left.\frac{\mathrm{d}\mathbf{r}^{\mathbf{i}}}{\mathrm{dt}}\right|_{e} = \left.\frac{\mathrm{d}\mathbf{r}^{\mathbf{i}}}{\mathrm{dt}}\right|_{i} - \boldsymbol{\omega}_{\mathbf{ie}}^{\mathbf{i}} \times \mathbf{r}^{\mathbf{i}} = \mathbf{v}_{\mathbf{iv}}^{\mathbf{i}} - \boldsymbol{\omega}_{\mathbf{ie}}^{\mathbf{i}} \times \mathbf{r}^{\mathbf{i}} \tag{2.3.7}$$

where $\boldsymbol{\omega}_{\mathbf{ie}}^{\mathbf{i}}$ is the constant angular velocity of the rotating Earth in frame ***i*** , $\mathbf{r}^{\mathbf{i}}$ is the position vector and $\mathbf{v}_{\mathbf{iv}}^{\mathbf{i}}$ is the velocity vector of the particle with respect to an inertial frame ***i***. Accordingly, the first order time derivative of $\mathbf{v}_{\mathbf{ev}}^{\mathbf{i}}$ in frame ***i*** is

$$\left.\frac{\mathrm{d}\mathbf{v}_{\mathbf{ev}}^{\mathbf{i}}}{\mathrm{dt}}\right|_{\mathrm{i}} = \left.\frac{\mathrm{d}\mathbf{v}_{\mathbf{iv}}^{\mathbf{i}}}{\mathrm{dt}}\right|_{\mathrm{i}} - \boldsymbol{\omega}_{\mathbf{ie}}^{\mathbf{i}} \times \left.\frac{\mathrm{d}\mathbf{r}^{\mathbf{i}}}{\mathrm{dt}}\right|_{\mathrm{i}} \tag{2.3.8}$$

The further expansion of the last term on the right hand side of (2.3.8) yields

$$\left.\frac{\mathrm{d}\mathbf{v}_{\mathbf{ev}}^{\mathbf{i}}}{\mathrm{dt}}\right|_{\mathrm{i}} = \left.\frac{\mathrm{d}\mathbf{v}_{\mathbf{iv}}^{\mathbf{i}}}{\mathrm{dt}}\right|_{\mathrm{i}} - \boldsymbol{\omega}_{\mathbf{ie}}^{\mathbf{i}} \times \left.\frac{\mathrm{d}\mathbf{r}^{\mathbf{i}}}{\mathrm{dt}}\right|_{\mathrm{i}} = \left.\frac{\mathrm{d}^{2}\mathbf{r}^{\mathbf{i}}}{\mathrm{dt}^{2}}\right|_{\mathrm{i}} - \boldsymbol{\omega}_{\mathbf{ie}}^{\mathbf{i}} \times (\mathbf{v}_{\mathbf{ev}}^{\mathbf{i}} + \boldsymbol{\omega}_{\mathbf{ie}}^{\mathbf{i}} \times \mathbf{r}^{\mathbf{i}}) \tag{2.3.9}$$

On the other hand, the term on the left hand side of (2.3.8) is expanded as follow:

$$\left.\frac{\mathrm{d}\mathbf{v}_{\mathbf{ev}}^{\mathbf{i}}}{\mathrm{dt}}\right|_{\mathrm{i}} = \left.\frac{\mathrm{d}(\mathbf{C}_{\mathbf{n}}^{\mathbf{i}}\mathbf{v}_{\mathbf{ev}}^{\mathbf{n}})}{\mathrm{dt}}\right|_{\mathrm{i}} = \mathbf{C}_{\mathbf{n}}^{\mathbf{i}} \left.\frac{\mathrm{d}\mathbf{v}_{\mathbf{ev}}^{\mathbf{n}}}{\mathrm{dt}}\right|_{\mathrm{n}} + \boldsymbol{\omega}_{\mathbf{in}}^{\mathbf{i}} \times \mathbf{v}_{\mathbf{ev}}^{\mathbf{i}} = \left.\frac{\mathrm{d}\mathbf{v}_{\mathbf{ev}}^{\mathbf{n}}}{\mathrm{dt}}\right|_{\mathrm{i}} + \boldsymbol{\omega}_{\mathbf{in}}^{\mathbf{i}} \times \mathbf{v}_{\mathbf{ev}}^{\mathbf{i}} \tag{2.3.10}$$

Then, by combining (2.3.9) and (2.3.10), one obtains

$$\left.\frac{\mathrm{d}\mathbf{v}_{\mathbf{ev}}^{\mathbf{n}}}{\mathrm{dt}}\right|_{\mathrm{i}} + \boldsymbol{\omega}_{\mathbf{in}}^{\mathbf{i}} \times \mathbf{v}_{\mathbf{ev}}^{\mathbf{i}} + \boldsymbol{\omega}_{\mathbf{ie}}^{\mathbf{i}} \times (\mathbf{v}_{\mathbf{ev}}^{\mathbf{i}} + \boldsymbol{\omega}_{\mathbf{ie}}^{\mathbf{i}} \times \mathbf{r}^{\mathbf{i}}) = \left.\frac{\mathrm{d}^{2}\mathbf{r}^{\mathbf{i}}}{\mathrm{dt}^{2}}\right|_{\mathrm{i}} \tag{2.3.11}$$

And furthermore,

$$\left.\frac{d\mathbf{v}_{ev}^{n}}{dt}\right|_{i} + (\boldsymbol{\omega}_{in}^{i} + \boldsymbol{\omega}_{ie}^{i}) \times \mathbf{v}_{ev}^{i} + \boldsymbol{\omega}_{ie}^{i} \times \boldsymbol{\omega}_{ie}^{i} \times \mathbf{r}^{i} = \left.\frac{d^{2}\mathbf{r}^{i}}{dt^{2}}\right|_{i} = \mathbf{a}_{iv}^{i} \tag{2.3.12}$$

Finally, after all vectors in (2.3.12) are transformed into the navigation frame (***n***), the velocity $\mathbf{v}_{ev}^{n}$ is related to the IMU measurements $\mathbf{a}_{iv}^{b}$ in body frame as follows:

$$\frac{d\mathbf{v}_{ev}^{n}}{dt} + (\boldsymbol{\omega}_{in}^{n} + \boldsymbol{\omega}_{ie}^{n}) \times \mathbf{v}_{ev}^{n} + \mathbf{g}^{n} = \left.\frac{d^{2}\mathbf{r}^{i}}{dt^{2}}\right|_{n} = \mathbf{C}_{b}^{n}\mathbf{a}_{iv}^{b} \tag{2.3.13}$$

2.4 Principle of strapdown inertial navigation

This section briefs the realization of the traditional strapdown inertial navigation algorithms (inertial navigation mechanization) for free inertial navigation calculation. For the sake of simplicity, the sensor body frame (***b***) is assumed to be aligned with the vehicle frame (***v***) in the following discussion.

The inertial navigation mechanization is the mechanism that is made of a set of equations to propagate the navigation parameters in a specific coordinate frame through the high rate IMU outputs: specific forces and angular rates. It can be executed at the same or slower rates than the rate of the original IMU outputs. The navigation parameters of interest are normally position, velocity and attitude of a moving vehicle. Three choices of the navigation coordinate frames with which the INS mechanization proceeds are: the Earth Centered inertial Frame (ECI or ***i*** frame), the Earth centered Earth Fixed frame (ECEF or ***e*** frame) or the local level navigation frame (***n***) during the navigation mission. Despite of their physical and computational differences, the net results are the same for the given set of frames and the navigation equations [Jekeli, 2001]. Here,

only the INS mechanization in the local level navigation frame is overviewed because the navigation parameters defined in the local level navigation frame are more intuitive and preferred for the navigation mission near the Earth. Therefore, the INS mechanizations in ECI (***i***) frame and ECEF (***e***) frame are omitted here. Refer to [Jekeli, 2001; Bekir, 2007; Grove, 2013] for corresponding details.

Three variants of inertial navigation mechanizations in the local level frame are: the north seeking (slaved) mechanization, the free azimuth mechanization and the wander azimuth mechanization. The main difference among them is how the z axis of the local level navigation frame changes along the changing position and time during the mission. Specifically, the north seeking (slaved) mechanization always tracks the local geodetic (geographic) frame, and the free azimuth mechanization maintains the z axis of the navigation frame fixed relative to the inertial space, while the wander azimuth mechanization torques the navigation frame around the z axis to follow the Earth's rotation rate. For the ease of the understanding and the simplicity of the algorithm development, the north seeking INS mechanization is briefed below and the overview of the free azimuth and wander azimuth mechanization can be found in [Kelly, 1994; Maybeck, 1973; Grove, 2013; etc.].

Based on Section 2.2 and 2.3, the continuous strapdown mechanization equations using the north seeking method are given as [Titterton, 2004]:

$$\begin{aligned} \dot{\mathbf{C}}_{\mathbf{b}}^{\mathbf{n}} &= \mathbf{C}_{\mathbf{b}}^{\mathbf{n}}[\boldsymbol{\omega}_{\mathbf{nb}}^{\mathbf{b}}\times] \\ \dot{\mathbf{v}}^{\mathbf{n}} &= \mathbf{C}_{\mathbf{b}}^{\mathbf{n}}\mathbf{f}^{\mathbf{b}} - (\boldsymbol{\omega}_{\mathbf{en}}^{\mathbf{n}} + 2\boldsymbol{\omega}_{\mathbf{ie}}^{\mathbf{n}})\times\mathbf{v}^{\mathbf{n}} - \mathbf{g}^{\mathbf{n}} \\ \dot{\mathbf{r}}^{\mathbf{n}} &= \mathbf{v}^{\mathbf{n}} \end{aligned} \tag{2.4.1}$$

where are:

$\mathbf{r}^{\mathbf{n}}$ the position vector in the local navigation (ENU) frame,

$\mathbf{v}^{\mathbf{n}}$ the velocity vector in the local navigation frame,

$\mathbf{f}^{\mathbf{b}}$ the specific force with respect to the vehicle body frame,

$\mathbf{g}^{\mathbf{n}}$ the gravity vector in the local navigation frame,

$\boldsymbol{\omega}_{\mathbf{en}}^{\mathbf{n}}$ the crate rate vector in the local navigation frame,

$\boldsymbol{\omega}_{\mathbf{ie}}^{\mathbf{n}}$ the Earth rotation rate in the local navigation frame,

$\boldsymbol{\omega}_{\mathbf{nb}}^{\mathbf{b}}$ the turn rate of body frame with respect to the local navigation frame,

$\mathbf{C}_{\mathbf{b}}^{\mathbf{n}}$ transformation matrix from body frame to navigation frame, and

$\times$ the cross product operator.

In the strapdown inertial navigation, the above strapdown mechanization is executed in the discrete form through: attitude update, velocity update, and position update.

2.4.1 Attitude computation

Mathematically, the analytical solution of the instantaneous vehicle attitude should be achieved through solving the attitude differential equations given the initial orientation of the vehicle and the assumption of "continuous" error free angular rates of the vehicle based on gyroscopes' outputs. However, in reality, attitude update has to be conducted in the form of difference equation because the real sensor outputs are discrete in time and the navigation algorithm is executed in a digital computer.

Taking the example of DCM method, the determination of the vehicle attitude is to seek the solution of the transformation matrix $\mathbf{C}_{\mathrm{b}}^{\mathrm{n}}(\mathrm{k})$ at epoch *k* which implies the orientation of the rigid

page body. Under the assumption that an error-free gyroscope is employed, the DCM matrix at epoch *k*+1 can be obtained as follows [Savage, 1998; Rogers, 2003; Titterton, 2004; Bekir, 2007]:

$$\mathbf{C}_{b}^{n}(k+1) = e^{\int_{t_{k+1}}^{t_k} \boldsymbol{\Omega}_{mi}^{n} dt} \mathbf{C}_{b}^{n}(k) e^{\int_{t_k}^{t_{k+1}} \boldsymbol{\Omega}_{ib}^{b} dt} = \mathbf{R}_{n(k)}^{n(k+1)} \mathbf{C}_{b}^{n}(k) \mathbf{R}_{b(k+1)}^{b(k)} \tag{2.4.2}$$

wherein $\mathbf{R}_{a(k)}^{a(k+1)}$ is the intermediate transformation matrix of frame ***a*** from epoch *k* to epoch *k*+1.

In summary, discrete attitude update can be realized through two steps: a) intermediate transformation matrices update on $\mathbf{R}_{n(k-1)}^{n(k)}$ and $\mathbf{R}_{b(k-1)}^{b(k)}$; b) final DCM matrix update on $\mathbf{C}_{b}^{n}(k)$ using (2.4.2).

2.4.2 Velocity computation

The velocity update is to compute the instantaneous velocity at epoch *k* +1 and starts from the integration of the acceleration vector. In the first place, the integrated delta velocity $\boldsymbol{\upsilon}^{n}$ (incremental velocity from epoch *k* to epoch *k*+1) is defined as:

$$\boldsymbol{\upsilon}^{n} = \int_{t_k}^{t_{k+1}} \boldsymbol{f}^{n} dt \tag{2.4.3}$$

where $\boldsymbol{f}^{n}$ is the instantaneous specific force vector in navigation frame ($\boldsymbol{f}^{n} = \mathbf{C}_{b}^{n} \boldsymbol{f}^{b}$). After the omission of the higher order terms, the solution of the incremental velocity $\boldsymbol{\upsilon}^{n}$ from epoch *k* to epoch *k*+1 consists of three terms resulted from: i) the term resulted from direct accumulation of specific force vector in body frame, ii) the rotation correction due to the changing body frame, and iii) the dynamic correction (e.g., sculling motion) [Titterton, 2004].

$$\boldsymbol{\upsilon}^{n} = \mathbf{C}_{b(k)}^{n(k)} \left(\mathbf{v}_{k+1} + \frac{1}{2} \boldsymbol{\alpha}_{k+1} \times \mathbf{v}_{k+1} + \frac{1}{2} \int_{t_k}^{t_{k+1}} (\boldsymbol{\alpha} \times \boldsymbol{f}^{b} - \boldsymbol{\omega}^{b} \times \mathbf{v})) dt \right) \tag{2.4.4}$$

Chawherein $\boldsymbol{f}^{b}$ is the specific force vector in body frame, $\boldsymbol{\omega}^{b}$ is the angular rate vector in body frame, the delta velocity $\mathbf{v}_{k+1} = \int_{t_k}^{t_{k+1}} \boldsymbol{f}^{b} dt$, and the delta angle $\boldsymbol{\alpha}_{k+1} = \int_{t_k}^{t_{k+1}} \boldsymbol{\omega}^{b} dt$.

At last, the instantaneous velocity $\mathbf{v}_{k+1}^{n}$ at epoch *k*+1 can be simply updated using the incremental velocity $\boldsymbol{\upsilon}^{n}$ as follows [Savage, 1998; Rogers, 2003; Titterton, 2004; Grove, 2008]:

$$\mathbf{v}_{k+1}^{n} = \mathbf{v}_{k}^{n} + \boldsymbol{\upsilon}^{n} \tag{2.4.5}$$

2.4.3 Position computation

There are various realizations of the position computation depending on the system performance requirement as well as the availability of the aiding position sensor [Savage, 1998; Jekeli, 2001; Titterton; 2004; Bekir, 2007]. In a low-cost land vehicle navigation system, using the NED local navigation frame defined at the point vector $\mathbf{r}_{k-1}$ at epoch *k*-1, the position vector $\mathbf{r}_{k} = [r_N, r_E, r_D]^T$ at epoch *k* can simply be propagated using trapezoid integration method [Titterton, 2004]:

$$\mathbf{r}_{k} = \mathbf{r}_{k-1} + \delta \mathbf{r} = \mathbf{r}_{k-1} + (\mathbf{v}_{k}^{n} + \mathbf{v}_{k-1}^{n})\frac{\Delta t}{2} \tag{2.4.6}$$

In consideration of the computation of the transport rate $\boldsymbol{\omega}_{en}^{n} = \left[\dot{\lambda}\cos\phi \quad -\dot{\phi} \quad -\dot{\lambda}\sin\phi\right]^T$ of the local navigation frame (***n***) about the Earth frame (***e***), an easier method for position computation is to use the polar coordinates (latitude λ, longitude ϕ, altitude *h*) with respect to the Earth frame through integrating the corresponding rate vector as follows:

$$\dot{\phi} = v_n / (R_m + h)$$

$$\dot{\lambda} = v_e / (R_p + h) / \cos\phi$$

$$\dot{h} = -v_d \tag{2.4.7}$$

wherein R_m is the meridian radius of curvature, R_p is the prime radius of curvature in the plane normal to the meridian [Bekir, 2007] and v_n, v_e, v_d are velocity scalars in three axes of the NED local navigation frame.

2.4.4 IMU measurement models

Without taking into account the installation errors of an inertial sensor, the single component of the rate-output IMU measurement vector for either angular rates or specific forces can be modelled as a function of the true signal (angular rate or specific force) in the body frame [G. Egziabher, 2004] as follow:

$$s_m = (1 + s_f)\, s_t + b_0 + b_1(t) + b_\omega(t) \tag{2.4.8}$$

In (2.4.8), s_m is the sensor measurement and s_t is the truth of s_m. In addition, the measurement is also corrupted by: a random constant b_0, a time varying bias drift $b_1(t)$ and the white noise $b_\omega(t)$. For most of the inertial sensors, $b_1(t)$ is usually modeled as a first order zero-mean Gauss-Markov process [Nassar 2003] whose mathematical form is usually granted after [Gelb, 1974; G. Egziabher, 2004; El-Diasty et al, 2008; Petkov, 2010]:

$$\dot{b}_1(t) = -\frac{1}{\tau} b_1(t) + \omega_{b_1} \tag{2.4.9}$$

where the variable τ is the time constant (correlation time) and ω_{b_1} is the white noise.

2.4.4.1 Accelerometer measurement model

The measurement from an accelerometer in a rate-output IMU is called the specific force. Based on (2.3.12) and (2.4.8), the measurement model for the specific force vector $\boldsymbol{f}_{\text{ib-imu}}^{\text{b}}$ can be constructed as:

$$\boldsymbol{f}_{\text{ib-imu}}^{\text{b}} = (\mathbf{I} + \mathbf{S}_{\text{a}})\left(\mathbf{a}_{\text{nb}}^{\text{b}} + \mathbf{C}_{\text{n}}^{\text{b}}\left((2\boldsymbol{\omega}_{\text{ie}}^{\text{n}} + \boldsymbol{\omega}_{\text{en}}^{\text{n}}) \times \mathbf{v}_{\text{nb}}^{\text{n}}\right) - \mathbf{C}_{\text{n}}^{\text{b}}\mathbf{g}^{\text{n}}\right) + \mathbf{b}_{\text{a}} + \nabla_{\text{a}} \tag{2.4.10}$$

wherein is

$\boldsymbol{\omega}_{\text{ie}}^{\text{n}}$ the Earth's rotation rate vector in the local navigation frame,

$\boldsymbol{\omega}_{\text{en}}^{\text{n}}$ the craft rate vector in the local navigation frame,

$\mathbf{a}_{\text{nb}}^{\text{n}}$ the acceleration vector of vehicle with respect to the local navigation frame,

$\mathbf{v}_{\text{nb}}^{\text{n}}$ the velocity vector of vehicle with respect to the local navigation frame

$\mathbf{b}_{\text{a}}$ the bias vector including start off biases and bias residuals for accelerometers,

$\mathbf{S}_{\text{a}}$ the 3×3 scalar and misalignment error matrix for accelerometers,

$\mathbf{C}_{\text{n}}^{\text{b}}$ the transformation matrix from navigation frame to body frame, and

∇_{a} the white noise vector for accelerometer measurement vector.

2.4.4.2 Gyroscope measurement model

The measurement of a gyro in a rate-output IMU is called the angular rate. Similarly, the measurement model for the angular rate vector $\boldsymbol{\omega}_{\mathbf{ib}}^{\mathbf{b}}$ can be constructed as:

$$\boldsymbol{\omega}_{\text{ib-imu}}^{\text{b}} = (\mathbf{I} + \mathbf{S}_{\boldsymbol{\omega}})\left(\boldsymbol{\omega}_{\text{nb}}^{\text{b}} + \mathbf{C}_{\text{n}}^{\text{b}}(\boldsymbol{\omega}_{\text{ie}}^{\text{n}} + \boldsymbol{\omega}_{\text{en}}^{\text{n}})\right) + \mathbf{b}_{\boldsymbol{\omega}} + \nabla_{\boldsymbol{\omega}} \tag{2.4.11}$$

wherein is

$\boldsymbol{\omega}_{ie}^{n}$ the Earth's rotation rate vector in the local navigation frame,

$\boldsymbol{\omega}_{en}^{n}$ the craft rate vector in the local navigation frame,

$\boldsymbol{\omega}_{nb}^{n}$ the angular rate vector of vehicle with respect to the local navigation frame,

$\mathbf{b}_{\omega}$ the bias vector including start off biases and bias residuals for gyroscopes,

$\mathbf{S}_{\omega}$ the 3×3 scalar and misalignment error matrix for gyroscopes ,

$\mathbf{C}_{n}^{b}$ the transformation matrix from navigation frame to body frame, and

∇_{ω} the Gaussian white noise vector for gyroscope measurement vector.

2.5 Principle of GNSS positioning

Due to the sensor errors, initial misaligned attitude and position errors, the inertial navigation solution computed in Section 2.4 suffers from rapidly growing systematic errors. In the early development stages of inertial navigation technology, most of the inertial navigation systems merely have available the time-intermittent zero velocity updates (ZUPTs) and the static position fixes (PFIXs) for the calibration of the rapidly growing errors in navigation solution. It is after the full deployment of the Global Navigation Satellite System (GNSS) that the low-cost inertial navigation system becomes thriving because the all-weather GNSS system is able to provide the reliable position, velocity and the heading solution with a reasonable degree of accuracy [Bekir, 2007] in a constant rate so that these quickly growing errors can be effectively suppressed down to centimeter level.

This section briefs the principle of absolute positioning using GNSS technology. GNSS is the collective term for those navigation systems that provide the user with a three-dimensional positioning solution by passive ranging using radio signals transmitted by orbiting satellites [Groves, 2008]. There are four popular systems available: the Global Positioning System (GPS)

by United State, Global Navigation Satellite System (GLONASS) by Russia, Galileo by European Union, and Beidou by China. Because the fundamental principle of GPS system applies to all other GNSS, the positioning techniques based on GPS system are overviewed in this section.

2.5.1 Trilateration

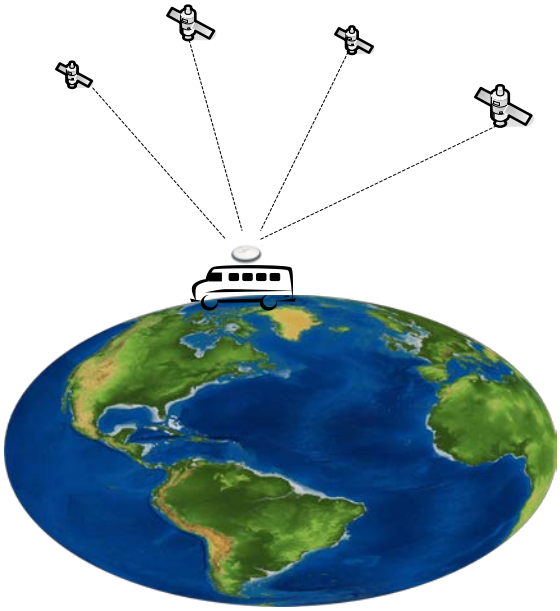


Figure 2-4 Trilateration in GNSS

Trilateration uses distance measurements from at least three fixed points to determine the geospatial coordinates of an unknown position [Murphy et al, 1995; Navidi et al, 1999]. On the ground of the idea of trilateration principle, GNSS technology makes possible the automated real-time positioning through acquiring and tracking radio signals from multiple satellites with their known orbits (Figure 2-4) to determinate the user position using a GNSS receiver.

Assume that satellite ***i*** has its known coordinates (X_i, Y_i, Z_i) and a GNSS receiver generates a range (distance) measurement r_i to it, the unknown position (x, y, z) of the receiver relates to the measurement and satellite's coordinates as follows

$$r_i = \sqrt{(x - X_i)^2 + (y - Y_i)^2 + (z - Z_i)^2} \qquad (i = 1...n) \tag{2.5.1}$$

Provided that more than four range measurements are acquired at the same time, a Least-Squares solution of position (x, y, z) can be reached together with the receiver clock error.

2.5.2 GNSS measurements

In this dissertation, the author describes and uses only the GPS measurements from the GNSS receiver because that: 1) Most of the author's available receivers track only GPS satellites; 2) GLONASS observables processing has its unique issues such as ephemeris decoding, satellite position calculation we didn't get involved; 3) the LAMBDA method fixing the integer ambiguities for the purpose of the cm-level positioning accuracy works only with GPS carrier phases.

There are normally three types of the GPS measurements available from a GPS receiver: L1 C/A pseudo-ranges, L1 carrier phases and L2 carrier phases. In addition, Doppler (range rate) measurements can also be made available. Once the GPS modernization project is completed, one can expect more measurements such as L2 C/A pseudo-range, L5 C/A pseudo-ranges and L5 carrier phases. In a GPS receiver, a carrier phase measurement in either of L1, L2 and L5 band carries the range information as well as an integer number (ambiguity) in cycles of the associated wave length. On the other hand, the Doppler observables measure the rates of instantaneous carrier phase measurements [Kaplan et al, 2006]. The three measurement equations corresponding to a GPS satellite are [Han, 1999; Gao, 2007]

$$\rho = \|r\| + d_o + c(dt - dT) + d_{ion} + d_{trop} + \varepsilon_M + \varepsilon_r \tag{2.5.2}$$

$$\phi = \frac{1}{\lambda}\left[\|r\| + d_o + c(dt - dT) - d_{ion} + d_{trop} + \varepsilon_M\right] + N + \varepsilon_\phi \tag{2.5.3}$$

$$\dot{\phi} = \frac{1}{\lambda}\left[\dot{\|r\|} + \dot{d}_o + c(\dot{dt} - \dot{dT}) - \dot{d}_{ion} + \dot{d}_{trop} + \dot{\varepsilon}_M\right] + \varepsilon_{\dot{\phi}} \tag{2.5.4}$$

wherein ρ is the pseudo-range measurement in meter, ϕ is the carrier phase measurement in cycles, $\dot{\phi}$ is Doppler measurement (range rate), $\|r\|$ is the geometrical distance between a satellite and a receiver, λ is the wave length associated with the carrier phases, d_o is the effect of the ephemeris errors(orbit error), *dt* and *dT* are the satellite and receiver clock errors, d_{ion} is the ionospheric delay error, d_{trop} is the tropospheric delay error, ε_M is the multipath error, ε_r, ε_ϕ and $\varepsilon_{\dot{\phi}}$ are the respective measurement noises, and *N* is the ambiguity parameter.

2.5.3 GNSS positioning

With only pseudo-range measurements, a single receiver can accomplish *absolute positioning* at the accuracy of meters. On the other hand, *relative positioning* can provide a baseline solution between two receivers (stations) at the much higher accuracy because of significant reduction of systematic errors, especially the ionospheric and tropospheric delays, in the differenced observables between receivers (stations). Specifically, with the double differenced (DD) carrier phase measurements for a short baseline (< 10km), the relative positioning can reach an accuracy of a few centimeters if the integer ambiguities are fixed correctly. Accordingly, three DD measurement equations can be derived base on (2.5.2), (2.5.3) and (2.5.4) as follows:

$$\nabla\Delta\rho_{\mathbf{AB}}^{\mathbf{jk}} = \nabla\Delta\|r\|_{\mathbf{AB}}^{\mathbf{jk}} + \varepsilon_{\nabla\Delta r}^{\mathbf{jk}} \tag{2.5.5}$$

$$\nabla\Delta\phi_{\mathbf{AB}}^{\mathbf{jk}} = \nabla\Delta\|r\|_{\mathbf{AB}}^{\mathbf{jk}} + \lambda\nabla\Delta N_{\mathbf{AB}}^{\mathbf{jk}} + \varepsilon_{\nabla\Delta\phi}^{\mathbf{jk}} \tag{2.5.6}$$

$$\nabla\Delta\dot{\phi}_{\mathbf{AB}}^{\mathbf{jk}} = \nabla\Delta\|\dot{r}\|_{\mathbf{AB}}^{\mathbf{jk}} + \varepsilon_{\nabla\Delta\dot{\phi}}^{\mathbf{jk}} \tag{2.5.7}$$

where $\nabla\Delta$ is the double differencing operator between the receivers **A** and **B** with respect to the satellites *j* and *k*.

2.5.4 GNSS attitude determination

Given a baseline between two GNSS antennas, the azimuth of the baseline can be derived and the accuracy of which is closely related to the length and the accuracy of the baseline. Consequently, in most of the land-based low-cost inertial navigation systems, dual GNSS antennas with a fixed short baseline tied on a moving vehicle become a standard configuration to provide the aiding heading measurement. Occasionally, the heading (azimuth) of a moving vehicle is solved through velocity heading vector.

2.6 Inertial navigation

As mentioned in Section 1.1, most of the research activities related to inertial navigation technology can be classified into following sub-areas:

- Improvements of stochastic error models for inertial sensors
- Improvements of inertial navigation system initializations and alignments
- Improvements of inertial navigation mechanization algorithms
- Application of the advanced filtering techniques
- Improvements of multisensory-aided integration strategy

This section reviews above topics except the INS mechanization covered in Section 2.4. The major emphasis of this review focuses on the multisensor integration strategy in an inertial navigation system because it leads to the achieved research contribution presented in Chapter 4.

2.6.1 Error modeling of inertial sensors

In general, IMU outputs are contaminated by the deterministic error (***D***) and the stochastic error (***S***). The deterministic error (***D***) can be caused by scale factor error and misalignment error [Woodman, 2007], non-linearity error involving scale factor [Flenniken, 2005; Amitava, 2008], spinning-mass [Groves, 2008], and other environmental sensitivity effects, for instance, G-sensitivity effects in gyro [Weinberg, 2011], temperature related effects [Mohammed El-Diasty, 2009] and so on. Normally, these deterministic errors are compensated with the *a priori* coefficients determined in laboratory environment by certain regression techniques [Skog, 2006; Mohammed El-Diasty, 2008; Hayal, 2010; Unsal, et al, 2012].

On the other hand, the stochastic error (***S***) associated with an inertial sensor is generally modelled as the sum of the turn-on biases, the stationary time correlated drifts, and the random errors. Firstly, the turn-on biases are commonly modelled as random constants [Shin, 2001; Titterton, 2004; Syed, 2007; Artese, 2008; Fong, et al, 2008; Bancroft, 2010]. The corresponding estimation process is also termed as the calibration. Secondly, the time correlated random drifts are usually modelled on the basis of the repeated laboratory tests and then estimated through a filter [Allan, 1966; Nassar, 2004; Yudan Yi, 2007]. Thirdly, the random errors are normally characterized as the normal distributed noises incurred from sampling or quantizing a continuous signal with a finite word length conversion [Yudan Yi, 2007]. Depending on its hardware design and material selection, the measurements of an inertial sensor could demonstrate various combinations of errors. In order to ensure the success of a navigation mission, the first and paramount task is always the IMU error modeling process which includes: error identification and error modeling.

Multiple techniques have been developed to determinate the stochastic model and the associated coefficient parameters for IMU drifting errors, for example, autocorrelation method [Brown and Hwang, 1992], autoregressive (AR) process [Nassar, 2004], power spectral density (PSD) [Yudan Yi, 2007], Allan variance [Allan, 1966], wavelet de-noising [Kang, 2010]. The model parameters by autocorrelation sequence method are rarely accurate due to the limited length of the experimental data [Nassar, 2004]. The AR method can accurately model the medley of all time-correlated errors if the short-term white noise is removed, i.e., wavelet de-noising method. However, based on AR's results, the independent stochastic errors cannot be distinguished from each other. Allan variance and PSD are the two most commonly used approaches to distinguish and model the independent stationary error sources. In frequency domain, they are theoretically related to each other [Stein, 1985; Zhang et al, 2008; Lansdorp, et al, 2012] and their results match well [Yudan Yi, 2007; Quinchia, et al, 2013]. Practically, Allan variance method is more preferred than PSD due to its simplicity and efficiency [Hou, 2005; Flenniken, 2005; Aggarwal, 2008; Claudia C, 2008; De Agostino, 2009].

2.6.2 System initialization and alignment in inertial navigation

The initial position, velocity and orientation must be provided to an inertial navigation system in order to carry out the navigation mission. The corresponding process is traditionally referred as the initialization and alignment [Yudan Yi, 2007]. As mention in Section 2.6.1, the IMU errors to be estimated shall be also initialized during the initialization period, which is known as calibration [Bar-Itzhack, 1988]. Therefore, the goal of the initialization, alignment and calibration is nothing but determining the initial values of the relevant states in the employed filter before an inertial navigation system is ready to proceed with the positioning and navigation

mission. The calibration process can be accomplished either separately or in parallel with the alignment after the initialization of position and velocity. The popular external aiding data for INS initialization could come from GNSS positioning results (the absolute position, velocity and heading), or other pre-surveyed stationary points or attitude.

A stationary inertial navigation system can align itself by exploiting the direction of gravity and the direction of the Earth's spin axis, referred to as self-alignment [Yudan Yi, 2007]. The traditional method of the self-alignment is completed in two phases: analytic coarse (leveling) alignment [Kenneth, 1971; Schimelevich, 1996; Bekir, 2007; Silson, 2011, etc.] and gyro-compassing fine alignment [Cannon Jr, 1961; Kouba, 1962; Kong, 2000; Jekeli, 2001; Bekir, 2007; Sitaraa, 2012; Ma, et al, 2013; Sun, et al, 2013; etc.]. However, it is noteworthy that the low cost INS with the lower resolution of angular velocity than the Earth's rotation rate cannot perform self-alignment unless the extra data are provided.

Both the output feedback control method in classical error theory and the state feedback control method in modern control theory can accomplish the alignment task for an inertial navigation system [Wan, 1998]. In modern control theory, the state components are estimated using a state-based filter which can be categorized into state observer for a deterministic system and Kalman filter for a stochastic system. By iteratively applying the pre-established gain constants upon the system innovation in a deterministic system, the attitude estimates will gradually converge to a stable solution with the bounded errors. The convergence is guaranteed through analysis of system response in frequency domain. On the other hand, Kalman Filter gives an iterative linear minimum variance estimates of the states of interest if the system is treated as a stochastic system in which significant random errors and bias drifting in the measurements are

considered. For more discussion about feedback control method applied in INS alignment, refer to [Cannon Jr, 1961; Kouba, 1962; Wan, 1998; Kong, 1999; Jekeli, 2001; Bekir, 2003].

2.6.3 Filtering technique

2.6.3.1 Filter model

Filtering is a process of estimating the value or probability density distribution of an unknown or a group of unknowns in a stochastic system given a series of noisy data. The generic model for a stochastic system is described by a system model

$$\dot{\mathbf{x}}_n = \mathrm{f}_n(\mathbf{x}_n, \mathbf{w}_n) \tag{2.6.1}$$

and a measurement model

$$\mathbf{z}_n = \mathrm{h}_n(\mathbf{x}_n, \mathbf{v}_n) \tag{2.6.2}$$

where

- $\mathbf{x}_n$ is the state vector at epoch n,
- $\mathbf{w}_n$ is the dynamic (process) noise vector,
- $\mathbf{z}_n$ is the measurement vector at epoch n
- $\mathbf{v}_n$ is the measurement noise vector
- f_n is the system model,
- h_n is the measurement model.

2.6.3.2 Extended Kalman filter

This dissertation applies the extended Kalman filter (EKF) to estimate the state vector $\mathbf{x}_n$. The main features of an EKF are:

i) Predicting the state vector $\hat{\mathbf{x}}_{\mathbf{k+1}}$ at epoch *k*+1 using the original non-linear system model with the latest estimated state vector $\hat{\mathbf{x}}_{\mathrm{k}}$ at epoch *k*;

ii) Predicting the measurement vector needed in system innovation using the original non-linear measurement model with the latest predicted state vector $\hat{\mathbf{x}}_{\mathbf{k+1/k}}$ in (i).

The variance and covariance propagation is based on the total differential equations after the linearization about a latest best approximation of the true states. The proper model for process noises should take into account those ignored higher order terms in system model. Or else, the estimated states and their associated covariance matrix may diverge. Given a non linear system described in (2.6.1) and (2.6.2), the estimation of the states in the discrete form of the extended Kalman filter can be summarized as follows [Simon, 2006]:

a) State and covariance prediction (the time update)

$$\hat{\mathbf{x}}_{\mathbf{k|k-1}} = \hat{\mathbf{x}}_{\mathbf{k}}^{-} = \mathrm{f}_{\mathbf{k}}(\hat{\mathbf{x}}_{\mathbf{k-1|k-1}}) \tag{2.6.4}$$

$$\mathbf{P}_{\mathbf{k|k-1}} = \mathbf{F}_{\mathbf{k}}\mathbf{P}_{\mathbf{k-1|k-1}}\mathbf{F}_{\mathbf{k}}^{\mathrm{T}} + \mathbf{Q}_{\mathbf{k}} \tag{2.6.5}$$

b) State and covariance update (the measurement update)

$$\tilde{\mathbf{y}}_{\mathrm{k}} = \mathbf{z}_{\mathrm{k}} - \mathrm{h}_{\mathrm{k}}(\mathbf{x}_{\mathrm{k}}^{-}) \tag{2.6.6}$$

$$\mathbf{S}_{\mathrm{k}} = \mathbf{H}_{\mathrm{k}}\mathbf{P}_{\mathrm{k|k-1}}\mathbf{H}_{\mathrm{k}}^{\mathrm{T}} + \mathbf{R}_{\mathrm{k}} \tag{2.6.7}$$

$$\mathbf{K}_{\mathrm{k}} = \mathbf{P}_{\mathrm{k|k-1}}\,\mathbf{H}_{\mathrm{k}}^{\mathrm{T}}\,\mathbf{S}_{\mathrm{k}}^{-1} \tag{2.6.8}$$

$$\hat{\mathbf{x}}_{\mathrm{k|k}} = \hat{\mathbf{x}}_{\mathrm{k|k-1}} - \mathbf{K}_{\mathrm{k}}\tilde{\mathbf{y}}_{\mathrm{k}} \tag{2.6.9}$$

$$\mathbf{P}_{\mathrm{k|k}} = (\mathbf{I} - \mathbf{K}_{\mathrm{k}}\mathbf{H}_{\mathrm{k}})\mathbf{P}_{\mathrm{k|k-1}} \tag{2.6.10}$$

with $\mathbf{F}_k = \left.\frac{\partial f_k}{\partial \mathbf{x}}\right|_{\hat{x}_k^-}$ and $\mathbf{H}_k = \left.\frac{\partial h_k}{\partial \mathbf{x}}\right|_{\hat{x}_k^-}$ are the respective Jacobian matrices for system model and measurement model.

2.6.4 Sensor integration strategy

This section reviews the sensor integration strategies. To certain extent, the development of the integration strategy is interchangeable with the determination of the variables to be solved (state selection) and corresponding modeling work (the construction of the system and measurement equations). Based on author's literature survey, two approaches to differentiate the integration strategies in inertial navigation are to distinguish the selection of the states and the mechanism to couple the aiding non-inertial sensors.

2.6.4.1 Categorization by state selection

Because the choice of the states (variables of interest) leads to the system model as well as the mechanism performing error control in an inertial navigation system, many reference papers define the sensor integration strategy according to the selection of the states. Because Kalman filter is the standard filtering method in most of the researches and industrial products, the sensor integration strategies can also be named after the types of state vector in Kalman filter, for example, indirect (error-state-based) Kalman filter and direct (total-state-based) Kalman filter [Maybeck, 1979; Wendel, 2001; Giroux, 2005]. As the matter of the fact, given one chosen integration strategy, various nonlinear filtering candidates (e.g., EKF, UKF, Particle filter, etc.) are available. Nevertheless, this dissertation chooses to use the EKF instead of other nonlinear

filtering algorithms because the focus of the author's research is not laid on the filtering techniques.

Indirect Kalman Filter

The indirect Kalman filter approach is to estimate the error state vector $\Delta\mathbf{x}$ instead of the whole state vector $\mathbf{x_0}$. Given the original system model $f(.)$ in (2.6.1), the system differential equations of the error state vector $\Delta\mathbf{x}$ can be obtained through the perturbation analysis of $f(.)$ after ignoring the higher order term (H.O.T.). For example, assume the error state vector $\Delta\mathbf{x}$ applies to the whole state vector $\mathbf{x_0}$ and the disturbance vector $\Delta\mathbf{w}$ applies to the process noise vector $\mathbf{w_0}$, the differential equations of the estimated state vector $\hat{\mathbf{x}}$ ($\hat{\mathbf{x}} = \mathbf{x_0} + \Delta\mathbf{x}$) can be approximated using the first order of Taylor expansion:

$$\dot{\mathbf{x}}_0 + \Delta\dot{\mathbf{x}} = f(\mathbf{x}_0 + \Delta\mathbf{x}, \mathbf{w}_0 + \Delta\mathbf{w}, \mathrm{t}) = f(\mathbf{x}_0, \mathbf{w}_0, \mathrm{t}) + \mathbf{F}\Delta\mathbf{x} + \mathbf{L}\Delta\mathbf{w} + \delta\mathbf{w} \tag{2.6.11}$$

where: $\mathbf{x}_0$ is the whole state vector; $\mathbf{w}_0$ is the process noise vector; *t* is the time; $\mathbf{F}$ is the Jacobian matrix with respect to $\Delta\mathbf{x}$; $\mathbf{L}$ is the Jacobian matrix with respect to $\Delta\mathbf{w}$; $\delta\mathrm{w}$ is the ignored higher order terms.

Because $\dot{\mathbf{x}}_0 = \mathrm{f}(\mathbf{x}_0, \mathbf{w}_0, \mathrm{t})$, then

$$\Delta\dot{\mathbf{x}} = \mathbf{F}\Delta\mathbf{x} + \mathbf{L}\Delta\mathbf{w} + \delta\mathbf{w} \tag{2.6.12}$$

Restructure $\mathbf{L}\Delta\mathbf{w} + \delta\mathbf{w}$ to one vector $\tilde{\mathrm{w}}$, the dynamics equation for error state vector $\Delta\mathbf{x}$ at epoch *k*+1 can be discretized as follows:

$$\Delta\mathbf{x}_{\mathrm{k+1}} = \mathbf{F}_{\mathrm{k}}\Delta\mathbf{x}_{\mathrm{k}} + \mathbf{L}_{\mathrm{k}}\Delta\mathbf{w}_{\mathrm{k}} + \delta\mathbf{w}_{\mathrm{k}} = \mathbf{F}_{\mathrm{k}}\Delta\mathbf{x}_{\mathrm{k}} + \tilde{\mathbf{w}}_{\mathrm{k}} \tag{2.6.13}$$

Similarly, the measurement vector $\mathbf{z}$ as the function of the whole state vector $\mathbf{x}_0$ is approximated around its nominal value $h(\hat{\mathbf{x}}, t)$:

$$\mathbf{z} = h(\hat{\mathbf{x}} - \Delta\mathbf{x}) + \mathbf{v} = h(\hat{\mathbf{x}}, t) - \mathbf{H}\Delta\mathbf{x} + \delta\mathbf{v} + \mathbf{v} \tag{2.6.14}$$

where $\mathbf{v}$ is the measurement noise vector; $\mathbf{H}$ is the Jacobian matrix with respect to $\Delta\mathbf{x}$ and $\delta\mathbf{v}$ accounts for the ignored higher order terms (H.O.T.). The re-arrangement of (2.6.14) at epoch *k* gives

$$\Delta\mathbf{z}_k = \mathbf{H}_k\Delta\mathbf{x}_k - \mathbf{v}_k - \delta\mathbf{v}_k = \mathbf{H}_k\Delta\mathbf{x}_k + \tilde{\mathbf{v}}_k \tag{2.6.15}$$

where the error measurement vector $\Delta\mathbf{z} = h(\hat{\mathbf{x}}, t) - \mathbf{z}$ and $\tilde{\mathbf{v}}$ is the measurement noise vector.

In summary, the linearized system and measurement equations for error state vector $\Delta\mathbf{x}$ at epoch *k* are given as:

$$\Delta\mathbf{x}_{k+1} = \mathbf{F}_k\Delta\mathbf{x}_k + \mathbf{L}_k\Delta\mathbf{w}_k + \delta\mathbf{w}_k = \mathbf{F}_k\Delta\mathbf{x}_k + \tilde{\mathbf{w}}_k \tag{2.6.16}$$

$$\Delta\mathbf{z}_k = \mathbf{H}_k\Delta\mathbf{x}_k - \mathbf{v}_k + \delta\mathbf{v}_k = \mathbf{H}_k\Delta\mathbf{x}_k + \tilde{\mathbf{v}}_k \tag{2.6.17}$$

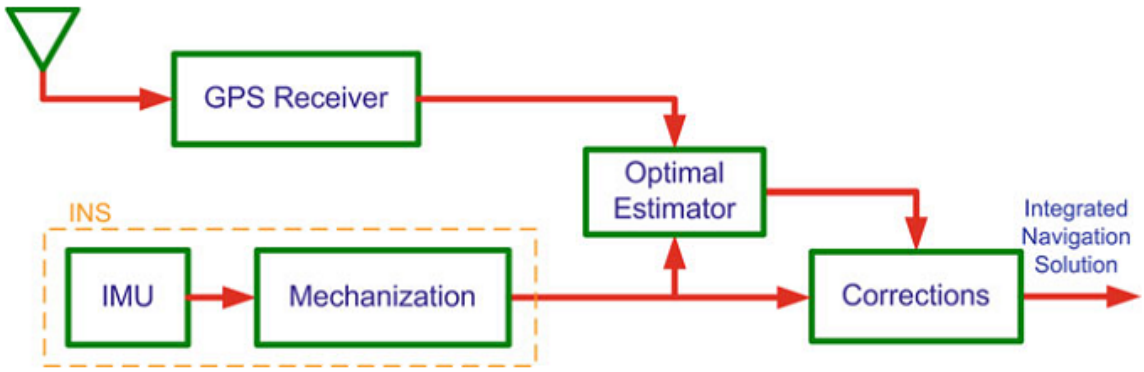
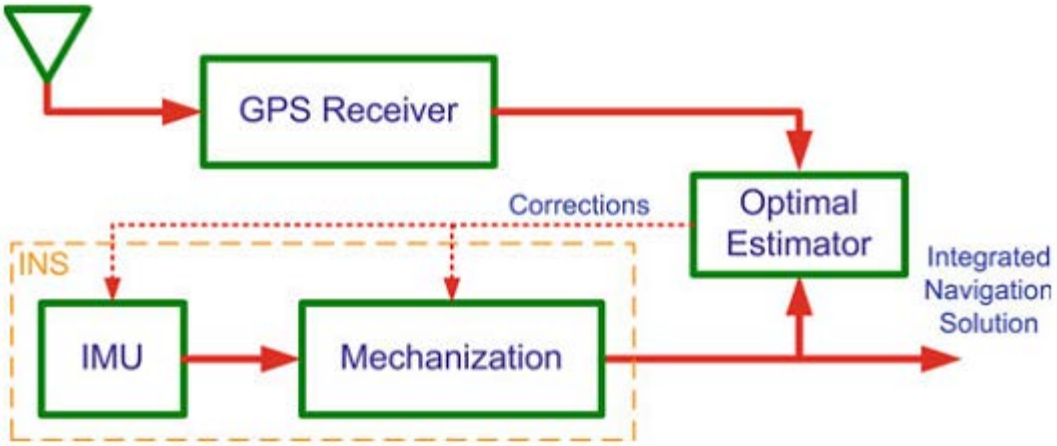


Figure 2-5 Open-loop error feedward in INS [Noureldin, et al, 2013]



Add

Figure 2-6 Close-loop error feedback in INS [Noureldin, et al, 2013]

At epoch *k*, the error state vector $\Delta\mathbf{x}_k$ resulted from the indirect Kalman filter can be fed forward/back to the inertial navigation system in either type of two non-mutually exclusive schemes: open-loop and closed-loop [Noureldin, et al, 2013]. In the open-loop scheme (Figure 2-5), the inertial navigation solution supports integrity monitoring and continuing service in the event of a problem with the Kalman filter [Groves, 2008]. While in the close-loop scheme (Figure 2-6), the error states are fed back (to calibrate the erroneous IMU measurements and internal variables in mechanization) to inertial navigation component on every iteration of the mechanization so that the errors with the inertial navigation will not grow without bound.

Direct Kalman Filter

The main difference between the indirect Kalman filter and the direct Kalman filter in multisensor integrated inertial navigation are the choice of the system model and the associated measurement model. More specifically, the aiding sensors in indirect Kalman filter aims to calibrate the error states in the primary system (e.g., navigation solution errors resulted from INS

mechanization). On the other hand, all sensors equally participate in direct Kalman filter to estimate the navigation solution states.

The generic system and measurement models in the form of the direct Kalman filter can be given as follows:

$$\mathbf{x}_n = \mathrm{f}_n(\mathbf{x}_{n-1}) + \mathbf{w}_n \tag{2.6.18}$$

$$\mathbf{z}_n = \mathrm{h}_n(\mathbf{x}_n) + \mathbf{v}_n \tag{2.6.19}$$

where

$\mathbf{x}_n$ = the state vector at epoch n,

$\mathbf{w}_n$ = the dynamic (process) noise vector,

$\mathbf{z}_n$ = the measurement vector at epoch n

$\mathbf{v}_n$ = the measurement noise vector

f_n = the nonlinear system model,

h_n = the nonlinear measurement model.

As its name implies, the direct Kalman filter takes the variables of interest as the states in the filter. In opposite to the indirect Kalman filter formulation, all sensor data including IMU outputs may be processed as the raw measurements in the direct Kalman filter [Maybeck, 1979]. This configuration is proposed for the low-cost sensors with large inertial sensor errors so that the INS errors remain small and the linearity assumption is upheld [Noureldin, et al, 2013]. The inherent disadvantage of this filter algorithm is the increased computational cost from the more frequent and time consuming Kalman gain calculation (involving matrix inversion) [Maybeck, 1979; Wendel, 2001]. Obviously, this conclusion is somehow outdated in consideration of the much more computation power in a modern CPU than those in early days.

More characteristics of the direct Kalman filter are revealed in chapter 4 since the new integration strategy is implemented by direct Kalman filter. A summary of the indirect EKF and the direct EKF is listed in Table 2-1 [Maybeck, 1979].

Table 2-1 Comparison between indirect and direct Kalman filter

	Advantages	Disadvantages
Indirect EKF	- Less computation with lower measurement update rate - Well behaved linear dynamics equation for the error state given the accurate INS mechanization solution	- Complex filter structure - Not easy to detect measurement blunders
Direct EKF	- Better performance during GNSS gap - Easier for blunder detection	- Heavy computation load - Less system robustness and reliability

2.6.4.2 Categorization by aiding data

In consideration of the growing systematic errors in an inertial navigation system, one or more independent measurements derived from external sources are necessary to calibrate the inertial navigation engine so as to yield a system with the greater precision than either of the components operating in isolation [Titterton, 2004]. Typical aiding data are the measurements from radar, GNSS satellite system, laser ranging sensor, speed sensor (e.g., distance measurement indicator (DMI)), altimeters and etc. Depending on the mechanism to integrate all participant sensors, the integration architecture may be broken down into five classes: snapshot fusion (least square) integration, cascade integration, centralized integration, federal integration, and hybrid integration [Grove, 2008]. Conventionally, in a navigation system involving both IMU and GNSS

receivers, the corresponding integration architectures are also jargoned as uncoupled, loosely coupled, tightly coupled and deeply coupled [Titterton, 2004].

Accordingly, in GNSS aided INS (GNSS/INS), the uncoupled system matches the snapshot integration architecture, the loosely coupled system corresponds to the cascade architecture and the tightly coupled system corresponds to the centralized architecture. Different from the other three strategies, the deeply coupled approach is usually implemented in an IMU-augmented GNSS receiver to improve GNSS signal track capability, especially in GNSS hostile environment (e.g., during short GNSS data gaps).

Snapshot integration

Snapshot fusion algorithm (Figure 2-7) delivers the final navigation solution $\hat{\mathbf{x}}_\mathbf{f}$ using the navigation solutions from *m* independent navigation processors ($\hat{\mathbf{x}}_\mathbf{1}, \hat{\mathbf{x}}_\mathbf{2} \ldots \hat{\mathbf{x}}_\mathbf{m}$). The typical fusion algorithm is an optimal estimator e.g., weighted least square, and an extreme case of which will be a simple switcher choosing one of candidate solutions as the final solution. Conventionally, the GNSS/INS system using snapshot strategy is also named the uncoupled system due to the independence among individual navigation processors. Due to the scope of this dissertation, no more discussion about the snapshot integration will be proceeded.

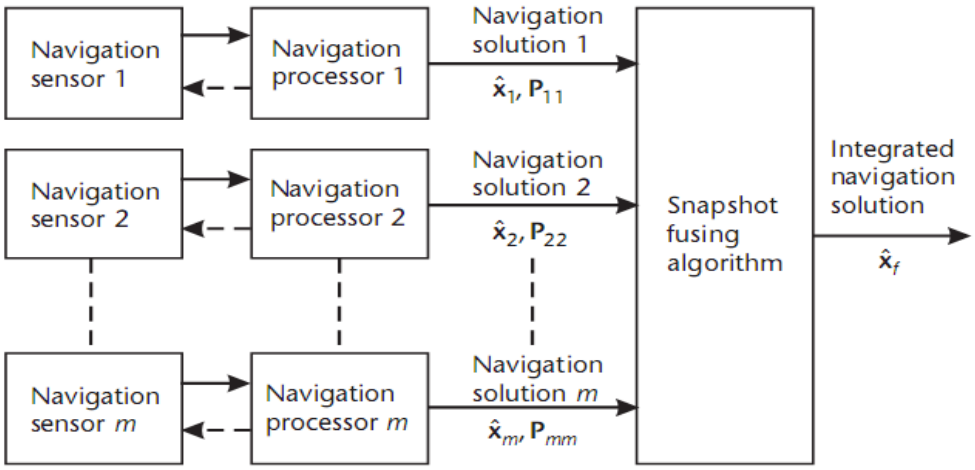


Figure 2-7 Snapshot fusion integration [Grove, 2008]

Cascade integration

If the snapshot fusion algorithm is replaced by an iterative time-domain estimator (e.g., Kalman filter), the system is termed as cascade system. Like in snapshot system, the aiding data is in the form of the navigation solution (e.g., position, velocity, and etc.) resulted from the independent navigation processors. Depending on the form of states, the cascade integrated architecture can be used in direct (total state) filter or indirect (error state) filter. The error measurement (difference between aiding navigation solution and reference navigation solution) in indirect Kalman filter shall be derived according to the 2.6.15. In direct Kalman filter, the navigation solutions from individual navigation processors will equally take part in the filter as stochastically independent measurements.

Centralized integration

If the aiding data (navigation solutions) in a cascade system is replaced by the raw sensor data, e.g. the GNSS pseudoranges and range rates, the system is termed as a centralized system because the whole system is built upon a central estimator by combining all sorts of raw sensor measurements. Due to the disappearance of the independent sub-navigators and the closer relationship between sensors and system, this integration architecture is conventionally named the tightly-coupled system, especially in the case of the GNSS/INS system. Similar to the cascade system, the centralized integrated architecture can be also used in direct (total state) filter or indirect (error state) filter.

Federal and hybrid integration

Unlike the cascade and centralized system with only one core Kalman filter, the federated filter provides a more robust and reliable structure integrating a few navigation sub-systems each of which is attached with a local (Kalman) filter. A master filter is responsible for the fusion of the results from local filters. The implementation of each local Kalman filters may be cascaded or centralized. Obviously, the failure of one local Kalman filter shall not collapse the whole federal system because each local Kalman filter is isolated from the other systems. In other words, the enhanced robustness of the federated system benefits from the separated processes of the individual subsystems. It is conceivable that the advantages of the federal integration architecture are at the cost of physical size and computation power [Carlson, N. A., 1990].

Compared to the federated system dealing with sub-navigation systems, a hybrid system defines a more flexible and compact architecture integrating subsystems and sensors. Figure 2-8 depicts a typical hybrid INS system [Grove, 2008] with centralized GNSS, cascaded Loran (long range navigation), and federated-cascaded TRN (Terrain-referenced navigation). Because the

federal and hybrid integration is beyond the scope of this dissertation, this overview will not be further extended.

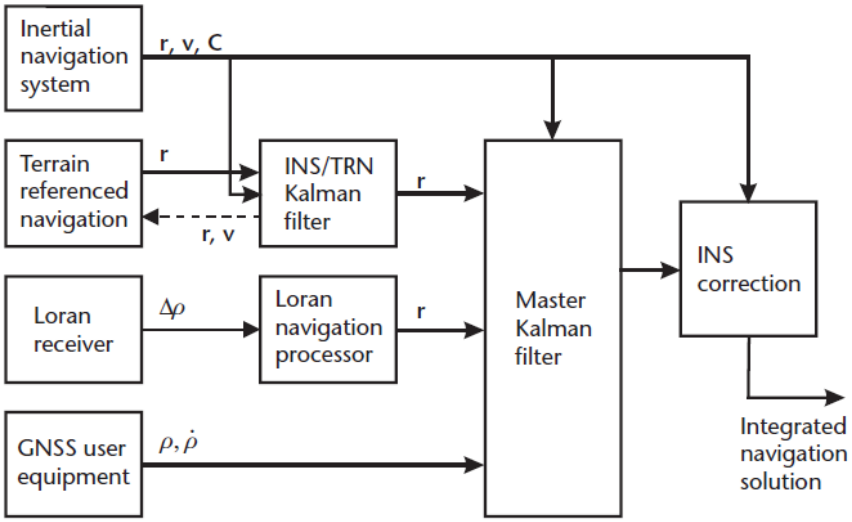


Figure 2-8 Hybrid navigation system [Grove, 2008]

2.6.4.3 Error-state based inertial navigation

Originated from the error control theory, the indirect Kalman filter is the most popular sensor integration strategy for an inertial navigation system, which aims to estimate the error states using the aiding information (e.g., positions and/or velocities from GNSS receivers, fixed positions, zero velocities, velocities from Doppler radar devices, etc.) so as to assure that the observable error states are well-controlled not to grow infinitively.

In principle, the error states must consist of: i) the navigational error states (position, velocity, and attitude) and ii) sensors' systematic error states. Various forms of system model [BAR-Itzhack, 1988; Pham, 1992; Scherzinger, 1994; Dmitriyev, et al, 1997; Kong, 1999] were developed to describe the transition of navigational error states in the time domain. The choice of the system model should rest with applications and depend on the adaptive habituation by users.

Although different by the definitions and the associated differential equations, the *psi*-angle based INS error model and the *phi*-angle based model are proved to be identical [Bar-Itzhack, 1981]. Without loss of the generality, the more intuitive *phi*-angle inertial navigation error model and its associated measurement models for GNSS measurements are here summarized in order to overview the application of the indirect Kalman filter in the inertial navigation.

INS error model

Take as example the reduced 15-state INS indirect Kalman filter, 9 error states (3 position components, 3 velocity components and 3 attitude components) and 6 inertial sensor error states (errors in a 3-axis gyroscope and a 3-axis accelerometer) make up an error state vector $\Delta\mathbf{x}$ ($\Delta\mathbf{x} = \left[\delta\mathbf{r}^{\mathbf{n}} \quad \delta\mathbf{v}^{\mathbf{n}} \quad \phi \quad \delta\boldsymbol{\omega}_{\mathbf{ib}}^{\mathbf{b}} \quad \nabla_{\mathbf{ib}}^{\mathbf{b}}\right]^{\mathbf{T}}$). In the inertial navigation calculation, the IMU outputs, therefore, are simply modelled as the sum of the true signals and the error signals as follows:

$$\boldsymbol{f}_{\mathrm{ib-imu}}^{\mathrm{b}} = \boldsymbol{f}_{\mathrm{ib}}^{\mathrm{b}} + \nabla_{\mathbf{ib}}^{\mathbf{b}} \text{ , and}$$

$$\boldsymbol{\omega}_{\mathrm{ib-imu}}^{\mathrm{b}} = \boldsymbol{\omega}_{\mathrm{ib}}^{\mathrm{b}} + \delta\,\boldsymbol{\omega}_{\mathrm{ib}}^{\mathrm{b}}$$

Under the assumption of the limited magnitudes of all error states, the differential equations of the error states built upon the INS mechanization equations are given as follows [Titterton, 2004; Bekir, 2007]:

$$\delta\dot{\mathbf{r}}^{\mathbf{n}} = \delta\mathbf{v}^{\mathbf{n}} \tag{2.6.14}$$

$$\begin{aligned}\delta\dot{\mathbf{v}}^{\mathbf{n}} = {} & \mathbf{C}_{\mathbf{b}}^{\mathbf{n}}\nabla_{\mathbf{ib}}^{\mathbf{b}} - \phi\times(\mathbf{C}_{\mathbf{b}}^{\mathbf{n}}\,\mathbf{f}^{\mathbf{b}}) \\ & -(\delta\,\boldsymbol{\omega}_{\mathbf{en}}^{\mathbf{n}} + 2\delta\,\boldsymbol{\omega}_{\mathbf{ie}}^{\mathbf{n}})\times\mathbf{v}^{\mathbf{n}} - (\boldsymbol{\omega}_{\mathbf{en}}^{\mathbf{n}} + 2\boldsymbol{\omega}_{\mathbf{ie}}^{\mathbf{n}})\times\delta\,\mathbf{v}^{\mathbf{n}} - \delta\,\mathbf{g}^{\mathbf{n}}\end{aligned} \tag{2.6.15}$$

$$\dot{\phi} \quad = -\boldsymbol{\omega}_{\mathbf{in}}^{\mathbf{n}}\times\phi + \delta\,\boldsymbol{\omega}_{\mathbf{in}}^{\mathbf{n}} - \mathbf{C}_{\mathbf{b}}^{\mathbf{n}}\delta\,\boldsymbol{\omega}_{\mathbf{ib}}^{\mathbf{b}} \tag{2.6.16}$$

$$\delta\dot{\boldsymbol{\omega}}_{\mathbf{ib}}^{\mathbf{b}} = \mathbf{0} \tag{2.6.17}$$

$$\dot{\nabla}_{ib}^{b} = \mathbf{0} \tag{2.6.18}$$

where is:

$\delta\mathbf{r}^{n}$ the 3×1 position error vector in the local navigation (ENU) frame,

$\delta\mathbf{v}^{n}$ the 3×1 velocity error vector in the local navigation frame,

$\delta\mathbf{g}^{n}$ the 3×1 gravity error vector in the local navigation frame,

$\delta\,\boldsymbol{\omega}_{en}^{n}$ the 3×1 crate rate error vector in the local navigation frame,

$\delta\,\boldsymbol{\omega}_{ie}^{n}$ the 3×1 Earth rotation rate error vector in the local navigation frame,

$\delta\,\boldsymbol{\omega}_{ib}^{b}$ the 3×1 gyroscope drift error vector,

$\mathbf{C}_{b}^{n}$ the 3×3 true transformation matrix from body frame to navigation frame,

ϕ the 3×1 attitude error vector so that $\hat{\mathbf{C}}_{b}^{n} = [\mathbf{I} - (\phi\times)]\,\mathbf{C}_{b}^{n}$ and

$\delta\mathbf{C}_{b}^{n} = [-(\phi\times)]\,\mathbf{C}_{b}^{n}$

$\mathbf{f}^{b}$ the 3×1 true specific force in body frame, and

∇_{ib}^{b} the 3×1accelerometer drift error vector.

Aiding GNSS error measurements

Two measurement coupling approaches in GNSS-aided inertial navigation are termed the loosely and tightly coupled architecture. Their major distinction is the aiding measurements. The GNSS navigation solutions are used in the loosely-coupled integration while the raw GNSS measurements are used in the tightly coupled integration.

Loosely coupled integration

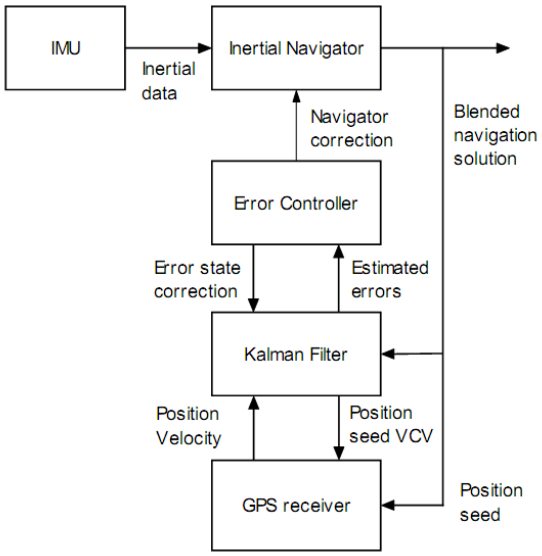


Figure 2-9 Loosely-coupled GNSS/INS integration [Scherzinger, 2000]

In a loosely-coupled GNSS/INS integrated system (Figure 2-9), an independent GNSS navigation processor is parallelly executed for the GNSS measurements to feed the position and velocity solution to the error-states based EKF filter in which the error measurements are constructed as follows:

$$\mathbf{r}_{\mathbf{INS}}^{\mathbf{n}} - \mathbf{r}_{\mathbf{GPS}}^{\mathbf{n}} = \mathbf{H}_{Position}\,\Delta\mathbf{x} + \varepsilon_{\mathbf{r}}^{\mathbf{j}} \tag{2.6.19}$$

$$\mathbf{v}_{\mathbf{INS}}^{\mathbf{n}} - \mathbf{v}_{\mathbf{GPS}}^{\mathbf{n}} = \mathbf{H}_{Velocity}\,\Delta\mathbf{x} + \varepsilon_{\mathbf{v}}^{\mathbf{j}} \tag{2.6.20}$$

where

$$\mathbf{H}_{Position} = \begin{bmatrix}\mathbf{I}_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3}\end{bmatrix}, \text{ and}$$

$$\mathbf{H}_{Velocity} = \begin{bmatrix}0_{3\times3} & \mathbf{I}_{3\times3} & 0_{3\times3} & 0_{3\times3} & 0_{3\times3}\end{bmatrix}.$$

Tightly coupled integration

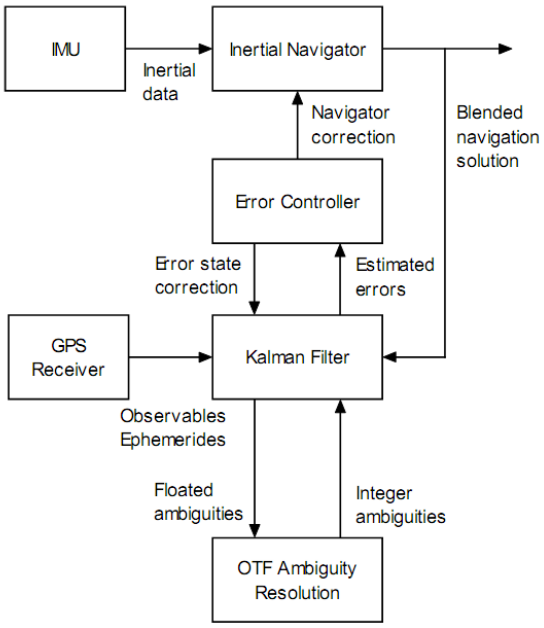


Figure 2-10 Tightly-coupled GNSS/INS integration [Scherzinger, 2000]

The tightly-coupled integration (Figure 2-10) is a natural extension of the loosely-coupled integration to overcome the main drawback of the loosely-coupled system: the GNSS navigation processor cannot provide position and velocity solutions when the number of the GNSS satellites in view is less than four. By directly coupling the raw GNSS observables, the error measurements are still available even with only one GNSS satellite in view.

2.6.4.4 Total-state based inertial navigation

Depending on the state selection, two kinds of total-state direct Kalman filter are full-state filter and compact-state filter. In the full-state direct Kalman filter, the state vector includes position, velocity and attitude and all other states of interest while the measurements are IMU outputs and other external source signals [Maybeck, 1979]. Thus, the INS mechanization will be directly used as the non-linear system model for the core navigation states in the full-state direct Kalman filter.

However, the total-state direct Kalman filter was hardly discussed in literatures until the compact total-state direct Kalman filter was proposed by Wagner et al [1997] because that the computer in the early days cannot afford to the huge computation cost spent on the processing of the inertial sensor data in the measurement updates of the Kalman filter. As a computation-efficient alternative, the compact total-state direct Kalman filter takes fewer variables as states while treating the inertial sensor measurements as known input vector **u** [Wendel, 2001].

Nowadays, the great advances in computation power and the trends of the usage of low-cost inertial sensors prompt researchers to rethink or even overturn the stereotypes about the application of the direct Kalman filter in the low-cost INS system. Most importantly, the nonlinear system model in the total-state Kalman filter provides the flexibility to refine the system model [Phuong, 2009; Edwan, 2012] and even modify the navigation mechanization equations so as to achieve the statistically superior navigation solution. Inspired by this thought, this dissertation proposes a novel generic integration strategy (Chapter 4) applied to a multisensor integrated kinematic positioning and navigation system, in which the rigorous 3D kinematic trajectory model is deployed as the core of the system model.

Full total-state inertial navigation

Similar to the simplified 15-state INS indirect Kalman filter, the state vector in a typical 21-state direct inertial navigation Kalman filter can be defined as follow [Wendel, 2001]:

$$\mathbf{x_f} = \begin{bmatrix} \mathbf{r^n} & \mathbf{v^n} & \boldsymbol{\theta} & \boldsymbol{\omega}_{\mathbf{ib}}^{\mathbf{b}} & f_{\mathbf{ib}}^{\mathbf{b}} & \delta\boldsymbol{\omega}_{\mathbf{ib}}^{\mathbf{b}} & \nabla_{\mathbf{ib}}^{\mathbf{b}} \end{bmatrix}^{\mathrm{T}}$$

wherein is

$\mathbf{r^n}$ the 3×1 position vector in the local navigation frame,

$\mathbf{v^n}$ the 3×1 velocity vector in the local navigation frame,

$\boldsymbol{\theta}$ the 3×1 attitude vector (e.g., Euler angle),

$\boldsymbol{\omega}_{\mathbf{ib}}^{\mathbf{b}}$ the 3×1 true angular rate of body frame with respect to the inertial frame,

$f_{\mathbf{ib}}^{\mathbf{b}}$ the 3×1 true specific force with respect to the vehicle body frame,

$\delta\boldsymbol{\omega}_{\mathbf{ib}}^{\mathbf{b}}$ the 3×1 systematic error in turn angular rate of body frame, and

$\nabla_{\mathbf{ib}}^{\mathbf{b}}$ the 3×1 systematic error in true specific force,

For a full total-state-based INS, the simplest choice of the system model is the continuous strapdown mechanization driven by the IMU outputs as given in (2.4.1). Namely, the differential equations for position, velocity and attitude can directly follow the inertial navigation mechanization equations. In addition, the error vectors $\delta\boldsymbol{\omega}_{\mathbf{ib}}^{\mathbf{b}}$ and $\nabla_{\mathbf{ib}}^{\mathbf{b}}$ as well as the angular rate vector $\boldsymbol{\omega}_{\mathbf{ib}}^{\mathbf{b}}$ and the specific force vector $f_{\mathbf{ib}}^{\mathbf{b}}$ are all modeled as random constants. As a result, by taking into account the process noise vector $\mathbf{w}_{\mathbf{f}}$ (modeling errors for $\boldsymbol{\omega}_{\mathbf{ib}}^{\mathbf{b}}$ and $f_{\mathbf{ib}}^{\mathbf{b}}$), given the continouse state model being $\dot{\mathbf{x}}_{\mathbf{f}} = \boldsymbol{f}(\mathbf{x}_{\mathbf{f}}, \mathbf{w}_{\mathbf{f}})$, the linearized model through Taylor expansion used for covariance propagation in the full total-state-based EKF can be formulated as:

$$\mathbf{x}_{(\mathbf{k})} = \mathbf{F}_{\mathbf{f}}\ \mathbf{x}_{(\mathbf{k-1})} + \mathbf{G}_{\mathbf{f}}\ \mathbf{w}_{\mathbf{f}} \tag{2.6.21}$$

where

$$\mathbf{x}_{(\mathbf{k})} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \mathbf{x}_3 \end{bmatrix}_{(\mathbf{k})}, \quad \mathbf{w}_{\mathbf{f}} = \begin{bmatrix} \mathbf{w}_{\boldsymbol{\omega}} \\ \mathbf{w}_{\mathbf{a}} \end{bmatrix}$$

$$\mathbf{x}_1 = \begin{bmatrix} \mathbf{r}^{\mathbf{n}} & \mathbf{v}^{\mathbf{n}} & \boldsymbol{\theta} \end{bmatrix}^{\mathbf{T}}, \ \mathbf{x}_2 = \begin{bmatrix} \boldsymbol{\omega}_{\mathbf{ib}}^{\mathbf{b}} & f_{\mathbf{ib}}^{\mathbf{b}} \end{bmatrix}^{\mathbf{T}}, \mathbf{x}_3 = \begin{bmatrix} \delta\boldsymbol{\omega}_{\mathbf{ib}}^{\mathbf{b}} & \nabla_{\mathbf{ib}}^{\mathbf{b}} \end{bmatrix}^{\mathbf{T}}$$

$\mathbf{F_f} = \begin{bmatrix} \mathbf{F_{11}} & \mathbf{F_{12}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}$, $\mathbf{G_f} = \begin{bmatrix} \mathbf{0} \\ \mathbf{I_{2x2}} \\ \mathbf{0} \end{bmatrix}$, and $\mathbf{F_{11}}$ and $\mathbf{F_{12}}$ are corresponding Jacobian matrices for the nonlinear model $\boldsymbol{f}(\mathbf{x_f}, \mathbf{w_f})$.

Let the IMU's angular rate and specific force vectors be $\boldsymbol{\omega}_{ib-imu}^{b}$ and $\boldsymbol{f}_{ib-imu}^{b}$, their corresponding measurement equations are:

$$\boldsymbol{f}_{ib-imu}^{b} = \boldsymbol{f}_{ib}^{b} + \nabla_{\mathbf{ib}}^{\mathbf{b}} + \boldsymbol{\varepsilon}_{f} \tag{2.6.22}$$

$$\boldsymbol{\omega}_{ib-imu}^{b} = \boldsymbol{\omega}_{ib}^{b} + \delta\boldsymbol{\omega}_{ib}^{b} + \boldsymbol{\varepsilon}_{\boldsymbol{\omega}} \tag{2.6.23}$$

where $\boldsymbol{\varepsilon}_{f}$ and $\boldsymbol{\varepsilon}_{\omega}$ are the random noises in gyro and accelerometer triads, respectively. Furthermore, taking as example a cascaded (loosely coupled) GNSS/INS system using full total-state filter, the measurement equations for GNSS position and velocity are

$$\mathbf{r}_{\mathbf{GNSS}}^{\mathbf{n}} = \mathbf{H}_{\mathbf{GNSS-POS}}\,\mathbf{x} + \boldsymbol{\varepsilon}_{\mathbf{r}} \tag{2.6.24}$$

$$\mathbf{v}_{\mathbf{GNSS}}^{\mathbf{n}} = \mathbf{H}_{\mathbf{GNSS-VEL}}\,\mathbf{x} + \boldsymbol{\varepsilon}_{\mathbf{v}} \tag{2.6.25}$$

where $\boldsymbol{\varepsilon}_{\mathbf{r}}$ and $\boldsymbol{\varepsilon}_{\mathbf{v}}$ are the random noises in position and velocity, and the coefficient matrices are $\mathbf{H}_{\mathbf{GNSS-POS}} = [\mathbf{I}_{3\times3} \quad \mathbf{0}_{3\times3} \quad \mathbf{0}_{3\times3} \quad \mathbf{0}_{3\times3} \quad \mathbf{0}_{3\times3} \quad \mathbf{0}_{3\times3} \quad \mathbf{0}_{3\times3}]$ and $\mathbf{H}_{\mathbf{GNSS-VEL}} = [\mathbf{0}_{3\times3} \quad \mathbf{I}_{3\times3} \quad \mathbf{0}_{3\times3} \quad \mathbf{0}_{3\times3} \quad \mathbf{0}_{3\times3} \quad \mathbf{0}_{3\times3} \quad \mathbf{0}_{3\times3}]$.

Compact total-state inertial navigation

A pioneering compact integration mechanism was proposed and has shown pleasing solution quality by directly estimating the optimal navigation parameters instead of the error states [Wagner et al, 1997; Wagner& Wieneke 2003], in which the inertial navigation mechanization was assimilated into the equivalent system equations in the extended Kalman filter. Following

Wagner's work, Edwan [2012] also developed a new loosely coupled DCM based GPS-aided inertial integration using the compact total-states, in which the state vector was composed of nine DCM elements, the gyroscope triad bias vector, the accelerometer triad bias vector, and the position and velocity vector expressed in the local navigation frame. Benefiting from the constant input vector **u**, the DCM states are linearly propagated along with the covariance and the filter produces satisfied performance in the low-cost inertial navigation system.

By taking as an example a 15-state compact total-state INS system, the system model of the individual states in the compact total-state INS takes exactly the same form as their counterparts in the full total-state INS except that: 1) the angular rate vector $\boldsymbol{\omega}_{ib}^{b}$ and the specific force vector $\boldsymbol{f}_{ib}^{b}$ are removed from the state vector; 2) the inertial sensor measurements for the angular rate vector $\boldsymbol{\omega}_{ib-imu}^{b}$ and the specific force vector $\boldsymbol{f}_{ib-imu}^{b}$ joined the system model as the input vector **u**. Similar to (2.6.21), the generic model for covariance propagation in the compact total state EKF was achieved through transforming the term $\mathbf{F_f}\ \mathbf{x_f}$ in the full total-state INS into $\mathbf{F_c}\ \mathbf{x_c} + \mathbf{G_c u}$, which was defined as follows:

$$\mathbf{x_{(k)}} = \mathbf{F_c}\ \mathbf{x_{(k-1)}} + \mathbf{G_c}\ \mathbf{u} + \mathbf{G_c w_c} \tag{2.6.28}$$

where:

$$\mathbf{x_{(k)}} = \begin{bmatrix} \mathbf{x_1} \\ \mathbf{x_3} \end{bmatrix}_{\mathbf{(k)}},\quad \mathbf{w_c} = \begin{bmatrix} \mathbf{w_{\omega\text{-imu}}} \\ \mathbf{w_{a\text{-imu}}} \end{bmatrix},\quad \mathbf{u} = \begin{bmatrix} \boldsymbol{\omega}_{\mathbf{ib\text{-imu}}}^{\mathbf{b}} & \mathbf{f}_{\mathbf{ib\text{-imu}}}^{\mathbf{b}} \end{bmatrix}^{\mathbf{T}},$$

$$\mathbf{x_1} = \begin{bmatrix} \mathbf{r^n} & \mathbf{v^n} & \boldsymbol{\theta} \end{bmatrix}^{\mathbf{T}},\ \mathbf{x_3} = \begin{bmatrix} \delta\boldsymbol{\omega}_{\mathbf{ib}}^{\mathbf{b}} & \nabla_{\mathbf{ib}}^{\mathbf{b}} \end{bmatrix}^{\mathbf{T}},$$

$$\mathbf{F_c} = \begin{bmatrix} \mathbf{F_{11}} & \mathbf{-F_{12}} \\ 0 & 0 \end{bmatrix},\ \mathbf{G_c} = \begin{bmatrix} \mathbf{F_{12}} \\ 0 \end{bmatrix}, \text{ and}$$

$\mathbf{F}_{11}$, $\mathbf{F}_{12}$ are the same matrices as the ones in full total-state INS, process noise vector $\mathbf{w}_{\omega\text{-imu}}$ and $\mathbf{w}_{\text{a-imu}}$ are the random errors in inertial sensor.

Regarding to the aiding position and velocity, their measurement equations are exactly the same as those in the full total-state INS because the states in both systems are the navigation parameters.

2.7 Random error analysis in KF

In this section, the random error analysis in Kalman Filter is detailed to inspire the work in Chapter 5. The solution optimality of a Kalman filter relies on the appropriate stochastic models which are numerically represented by two variance-covariance (VC) matrices **Q** associated with the process noise and **R** associated with the measurement vector. Estimation (tuning) of these two matrices has been actively pursued by plenty of researchers since the advent of Kalman filter. Despite of the variations in numerous application-specific algorithms, the tuning of the Kalman filter can be simply classified into two categories: VC matrix estimation (VCME) and variance component estimation (VCE) based on the *a priori* matrices. Mehra [1970, 1972] published his pioneering work about the direct estimation of matrices **Q** and **R** using the system innovations in steady-state KF. Four relevant estimation methods summarized in his work are: Bayesian, maximum likelihood, correlation and covariance matching. Mehra's work is still of directive significance to many recent researches [Dunik, et al, 2008; Bavdekar, et al, 2011; Bulut, 2011; Matisko, et al, 2013]. However, his method targeting on the linear control system can not be directly applied to a non-linear time-variant system. Similar to the Mehra's work, an Auto-covariance Least Square (ALS) was developed [Odelson, 2006] as a one-step estimation technique to directly solve for the unknown elements in **Q** and **R**. Nevertheless, the strict

assumptions on stability and observability of the system model make it not applicable to the inertial navigation system.

Alternatively, variance-factor based estimation strategy is more attractive because of its computation effectiveness and reliable accuracy when the skeleton (*a priori*) matrices for **Q** and **R** are known. This type of the VCE methods applied in Kalman filter (VCE-KF) was originated from the variance and covariance estimation in Least Squares after Helmert [1907]. Along with the continuous theoretical developments in [Forstner, 1979; Grafarend, et al, 1980; Koch, 1986; Ou, 1989; Xu, et al, 2006, 2007; Amiri-Simkooei, 2007; Teunissen, et al, 2008], VCE in Least Squares has been extensively used in many researches and engineering applications [Wang, 1997; Sieg and Hirsch, 2000; Wang, 2000; Wang and Rizos, 2002; Tiberius, 2003; Rietdorf, 2004; Tesmer, 2004; Zhou, et al, 2006; Hermann Bahr, et al, 2007; Gopaul, et al, 2010; Xiao, et al, 2014].

Compared to the rigorous variance and covariance estimators in Least Squares after Helmert's method, the iterative variance component estimator is more practical because it is computation-efficient and the stochastic independence among the measurements from different sensors holds true in most of the applications including the multisensor integrated navigation system.

2.7.1 Variance component estimation in Least Squares after Helmert

This section overviews the VCE method after Helmert in Least Squares. Let the measurement equation system be represented by

$$\mathbf{l} + \Delta = \mathbf{B}\,\delta\,\hat{\mathbf{x}} + \mathbf{F}(\mathbf{x}^{(0)}) \tag{2.7.1}$$

where are

$\mathbf{l}$ the measurement vector,

Δ the measurement noise vector,

$\mathbf{B}$ the design matrix,

$\mathbf{F}$ the nonlinear observation equations,

$\mathbf{x}^{(0)}$ the approximate of the parameter vector $\mathbf{x}$,

$\delta\hat{\mathbf{x}}$ the correction vector for $\mathbf{x}^{(0)}$.

Assume that the measurement vector $\mathbf{l}$ consists of m statistically independent measurement types (or measurements), (2.7.1) can be partitioned into

$$\begin{bmatrix} \mathbf{l}_1 \\ \vdots \\ \mathbf{l}_i \\ \vdots \\ \mathbf{l}_m \end{bmatrix} + \begin{bmatrix} \Delta_1 \\ \vdots \\ \Delta_i \\ \vdots \\ \Delta_m \end{bmatrix} = \begin{bmatrix} \mathbf{B}_1 \\ \vdots \\ \mathbf{B}_i \\ \vdots \\ \mathbf{B}_m \end{bmatrix} \delta\hat{\mathbf{x}} - \begin{bmatrix} \mathrm{f}_1(\mathrm{x}^{(0)}) \\ \vdots \\ \mathrm{f}_i(\mathrm{x}^{(0)}) \\ \vdots \\ \mathrm{f}_m(\mathrm{x}^{(0)}) \end{bmatrix} \tag{2.7.2}$$

where are

$\mathbf{l}_i$ $n_i \times 1$ sub-vector of the i-th type of the measurements

Δ_i the noise vector of $\mathbf{l}_i$

$\mathbf{B}_i$ the design matrix associated with $\mathbf{l}_i$

The weight matrix $\mathbf{P}$ of measurement vector $\mathbf{l}$ can be divided into m blocks of diagonal sub-matrices $\mathbf{P}_i$

$$\mathbf{P} = diag(\mathbf{P}_1 \quad \cdots \quad \mathbf{P}_i \quad \cdots \quad \mathbf{P}_m) \tag{2.7.3}$$

with its corresponding covariance matrix

$$\begin{aligned} \mathbf{D} &= diag(\mathbf{D}_1 \quad \cdots \quad \mathbf{D}_i \quad \cdots \quad \mathbf{D}_m) \\ &= diag(\sigma_{01}^2\mathbf{P}_1^{-1} \quad \cdots \quad \sigma_{0i}^2\mathbf{P}_i^{-1} \quad \cdots \quad \sigma_{0m}^2\mathbf{P}_m^{-1}) \end{aligned} \tag{2.7.4}$$

where $\sigma_{0i}^2\ (1, ..., m)$ is the *i*-th variance component (variance factor) of the unit weight to be estimated for the *i*-th group of the measurements ($\mathbf{l}_i$). The Least-Squares solution for the unknowns ($\hat{\mathbf{x}}$) and its covariance matrix $\mathbf{D}(\hat{\mathbf{x}})$ is

$$\hat{\mathbf{x}} = \mathbf{x}^{(0)} + \delta\ \hat{\mathbf{x}} = \mathbf{x}^{(0)} + \left(\mathbf{B}^{\mathrm{T}}\mathbf{P}\mathbf{B}\right)^{-1}\mathbf{B}^{\mathrm{T}}\mathbf{P}\mathbf{l} \tag{2.7.5}$$

$$\mathbf{D}(\hat{\mathbf{x}}) = \hat{\sigma}_0^2\left(\mathbf{B}^{\mathrm{T}}\mathbf{P}\mathbf{B}\right)^{-1} \tag{2.7.6}$$

Furthermore, the corrected measurement vector is $\hat{\mathbf{l}} = \mathbf{B}\,\delta\,\hat{\mathbf{x}} + \mathbf{F}(\mathbf{x}^{(0)})$ and the measurement residual vector $\mathbf{v} = \hat{\mathbf{l}} - \mathbf{l} = \left[\mathbf{B}\left(\mathbf{B}^{\mathrm{T}}\mathbf{P}\mathbf{B}\right)^{-1}\mathbf{B}^{\mathrm{T}}\mathbf{P} - \mathbf{I}\right]\mathbf{l}$, and each independent component of which follows the normal distribution $\mathbf{v}_{\mathrm{i}} \sim \mathrm{N}(\mathbf{0}, \mathbf{D}_{\mathbf{v}_{\mathrm{i}}})$ under the assumption of $\Delta_i \sim N(0, \mathbf{D}_{\Delta_i})$. Accordingly, the expectation of the sum of the weighted squared residuals for the *i*-th group observable ($\mathbf{L}_i$) is [Cui, et al, 2001]:

$$\begin{aligned}\mathrm{E}(\mathbf{v}_{\mathrm{i}}^{\mathrm{T}}\mathbf{P}_{\mathrm{i}}\mathbf{v}_{\mathrm{i}}) = &\left[\mathrm{n}_{\mathrm{i}} - 2\mathrm{tr}(\mathbf{N}^{-1}\mathbf{N}_{\mathrm{i}}) + \mathrm{tr}(\mathbf{N}^{-1}\mathbf{N}_{\mathrm{i}})^2\right]\ \sigma_{0\mathrm{i}}^2\ \ + \\ &\sum_{\mathrm{j}=1,\mathrm{j}\neq\mathrm{i}}^{\mathrm{m}}\{\mathrm{tr}(\mathbf{N}^{-1}\mathbf{N}_{\mathrm{i}}\mathbf{N}^{-1}\mathbf{N}_{\mathrm{j}})\sigma_{0\mathrm{j}}^2\}\end{aligned} \tag{2.7.5}$$

where n_i is the number of the measurements in the *i*-th group, *m* is the number of the unknowns, $\mathbf{N} = \mathbf{B}^{\mathrm{T}}\mathbf{P}\mathbf{B}$, and $\mathbf{N}_i = \mathbf{B}_i^T\mathbf{P}_i\mathbf{B}_i$ (*i* = 1, 2, …, m).

By reformulating 2.7.5, the unknown variance factor vector is related to the coefficient matrix **S** and the observable vector **W** as follows:

$$\underset{\mathrm{m}\times\mathrm{m}}{\mathbf{S}}\quad\underset{\mathrm{m}\times 1}{\hat{\sigma}^2} = \underset{\mathrm{m}\times 1}{\mathbf{W}} \tag{2.7.6}$$

where

$$\underset{m\times m}{\mathbf{S}} = \begin{bmatrix} n_1 - 2tr(\mathbf{N^{-1}N_1}) + tr(\mathbf{N^{-1}N_1})^2 & tr(\mathbf{N^{-1}N_1})tr(\mathbf{N^{-1}N_2}) & \cdots & tr(\mathbf{N^{-1}N_1})tr(\mathbf{N^{-1}N_m}) \\ & n_2 - 2tr(\mathbf{N^{-1}N_2}) + tr(\mathbf{N^{-1}N_2})^2 & & tr(\mathbf{N^{-1}N_2})tr(\mathbf{N^{-1}N_m}) \\ & & \ddots & \vdots \\ & & & n_m - 2tr(\mathbf{N^{-1}N_m}) + tr(\mathbf{N^{-1}N_m})^2 \end{bmatrix},$$

with $\underset{m\times 1}{\hat{\boldsymbol{\sigma}}^2} = \begin{bmatrix} \hat{\sigma}_{01}^2 & \hat{\sigma}_{02}^2 & \cdots & \hat{\sigma}_{0m}^2 \end{bmatrix}^{\mathbf{T}}$ and $\underset{m\times 1}{\mathbf{W}} = \begin{bmatrix} \mathbf{v_1^T P_1 v_1} & \mathbf{v_2^T P_2 v_2} & \cdots & \mathbf{v_k^T P_k v_k} \end{bmatrix}^{\mathbf{T}}$.

If the matrix **S** in 2.7.6 is invertible, the solution for $\sigma_{0i}^2(1,...,m)$ can be achieved as follows

$$\underset{m\times 1}{\hat{\boldsymbol{\sigma}}^2} = \underset{m\times m}{\mathbf{S^{-1}}}\ \underset{m\times 1}{\mathbf{W}} \tag{2.7.7}$$

In consideration of the computation cost due to the large scale of the matrix **S** in general, the solution in (2.7.7) is often simplified. Among various candidate algorithms, one iterative solution of $\sigma_{0i}^2(1,...,m)$ can be solved as

$$E(\mathbf{v}_i^T\mathbf{P}_i\mathbf{v}_i) = \sigma_{0i}^2(n_i - tr(\mathbf{N^{-1}N}_i)) \tag{2.7.8}$$

by assuming that

$$\sigma_{01}^2 = \sigma_{02}^2 = \cdots = \sigma_{0m}^2 = \sigma_{0i}^2 \tag{2.7.9}$$

Theoretically, above solution of $\sigma_{0i}^2(1,...,m)$ shall gradually converge to the true value after a few iterations, i.e., the solved $\sigma_{0i}^2(1,...,m)$ tends to satisfy (2.7.9). With the definition of redundancy contribution term $r_i = n_i - tr(\mathbf{N^{-1}N}_i)$, the practical estimation of $\sigma_{0i}^2(1,...,m)$ becomes [Förstner, 1979]

$$\hat{\sigma}_{0i}^2 = \mathbf{v}_i^T\mathbf{P}_i\mathbf{v}_i / r_i \tag{2.7.10}$$

where the r_i represents the total redundancy contribution of the measurement vector $\mathbf{l}_i$, which reflects the extent of the influence of raw observable $\mathbf{l}_i$ on the parameter estimates. The bigger the

r_i is, the less $\mathbf{l}_i$ affects the parameter estimation. With a group of independent measurements, the redundant index of each measurement always satisfies $0< r_i <1$. When $r_i = 1$, the measurement $\mathbf{l}_i$ is completely redundant. In other words, the measurement $\mathbf{l}_i$ becomes a high leverage measurement when the associated scalar r_i tends to zero.

2.7.2 Variance component estiamtion in Kalman Filter after Helmert

A practical VCE algorithm for Kalman filter was proposed based on following two discoveries [Wang, 1997]:

- Under the assumption that all noises are normal distributed, Kalman filter is constructed epoch wise by applying the least squares principle, which utilizes all of the random information as three groups of statistically independent measurements: the predicted state vector as a group of pseudo-measurements, the zero mean process noise vector also as a group of pseudo-measurements, and the raw measurement vector, whose residual or correction vectors can directly be calculated as the projection of the system innovation vector.
- The redundancy distribution associated with the above mentioned three groups of measurements, and the redundant index for each of the individual independent measurements can be calculated epoch wise after the reliability theory transplanted from Least Squares into Kalman filter.

Given the redundancy contribution indexes and residuals for the individual variance components, the simplified VCE algorithm has accordingly been developed [Förstner, 1979; Wang, 1997; Wang, et al, 2009; etc.]. Let the linear or linearized system described by KF at time t_k be

$$\mathbf{x}_k = \mathbf{\Phi}_k \mathbf{x}_{k-1} + \mathbf{\Lambda}_k \mathbf{w}_k \quad \text{(System model)} \tag{2.7.11}$$

$$\mathbf{z}_k = \mathbf{H}_k \mathbf{x}_k + \boldsymbol{\varepsilon}_k \quad \text{(Measurement model)} \tag{2.7.12}$$

where is

$\mathbf{x}_k$ the $n_x \times 1$ state vector,

$\mathbf{w}_k$ the $n_w \times 1$ process noise vector following $\mathbf{w}_k \sim \mathrm{N}(\mathbf{0}, \mathbf{Q}_w)$

$\mathbf{\Lambda}_k$ the coefficient matrix of w_k,

$\mathbf{z}_k$ the $n_z \times 1$ measurement vector,

$\boldsymbol{\varepsilon}_k$ the measurement noise vector following $\boldsymbol{\varepsilon}_k \sim \mathrm{N}(\mathbf{0}, \mathbf{R}_k)$

$\mathbf{\Phi}_k$ the state transition matrix,

$\mathbf{H}_k$ the design matrix, and

$\mathbf{w}_k$ and $\boldsymbol{\varepsilon}_k$ are assumed uncorrelated with each other.

By considering three independent groups of the measurements and pseudo-measurements at an arbitrary epoch *k* [Wang, 1997]:

(1) the raw measurement vector $\mathbf{l}_z = \mathbf{z}_k$ with its variance $\mathbf{D}_{\mathbf{l}_z} = \mathbf{R}_k$,

(2) the pseudo-measurement vector $\mathbf{l}_w = \mathbf{w}_k$ with its variance $\mathbf{D}_{\mathbf{l}_w} = \mathbf{Q}_k$,

(3) the third pseudo-measurement vector $\mathbf{l}_x = \mathbf{x}_{\mathbf{k/k-1}} = \mathbf{\Phi}_\mathbf{k} \mathbf{x}_{\mathbf{k-1}}$ with its variance $\mathbf{D}_{\mathbf{l}_x} = \mathbf{\Phi}_\mathbf{k} \mathbf{D}_{\mathbf{x}_{k-1}} \mathbf{\Phi}_\mathbf{k}^\mathrm{T}$, where $\mathbf{D}_{\mathbf{x}_{k-1}}$ is the variance matrix of $\mathbf{x}_{k-1}$,

one alternative measurement equation system can be constructed for Kalman filter at epoch *k* as follows:

$$\begin{bmatrix} \mathbf{v}_z \\ \mathbf{v}_w \\ \mathbf{v}_x \end{bmatrix} = \begin{bmatrix} \mathbf{H}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \\ \mathbf{I} & -\mathbf{\Lambda}_k \end{bmatrix} \begin{bmatrix} \hat{\mathbf{x}} \\ \hat{\mathbf{w}} \end{bmatrix} - \begin{bmatrix} \mathbf{l}_z \\ \mathbf{l}_w \\ \mathbf{l}_x \end{bmatrix} = \begin{bmatrix} \mathbf{H}_k & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \\ \mathbf{I} & -\mathbf{\Lambda}_k \end{bmatrix} \begin{bmatrix} \hat{\mathbf{x}} \\ \hat{\mathbf{w}} \end{bmatrix} - \begin{bmatrix} \mathbf{z}_k \\ \mathbf{w}_k \\ \mathbf{\Phi}_k \hat{\mathbf{x}}_{k/k-1} \end{bmatrix} \tag{2.7.13}$$

to which the Least Squares Principle can be applied to derive the identical solution for Kalman filter [Wang, 1997; Caspary and Wang, 1998]. One of the significant contributions made by this alternate derivation of KF was about to handle the process noise vector separately, which has made possible the simultaneous estimation of the variance components associated with the process noise vector $\mathbf{w}_k$ and the measurement noise vector $\boldsymbol{\varepsilon}_k$.

Wang [1997] proved that the measurement residual vectors for three independent measurement groups at epoch *k* as in 2.7.13 can be computed as the projections of the same innovation vector as follows:

$$\mathbf{v}_z = (\mathbf{I} - \mathbf{H}_k\mathbf{K})(\mathbf{z}_k - \mathbf{H}_k\hat{\mathbf{x}}_{k/k-1}) \tag{2.7.14}$$

$$\mathbf{v}_w = \mathbf{Q}_k\mathbf{\Lambda}_k^T\mathbf{D}_{\hat{\mathbf{x}}_{k/k-1}}^{-1}\mathbf{K}(\mathbf{z}_k - \mathbf{H}_k\hat{\mathrm{x}}_{k/k-1}) \tag{2.7.15}$$

$$\mathbf{v}_x = \mathbf{\Phi}_k\mathbf{D}_{\hat{\mathbf{x}}_{k-1}}\mathbf{\Phi}_k^T\mathbf{D}_{\hat{\mathbf{x}}_{k/k-1}}^{-1}\mathbf{K}(\mathbf{z}_k - \mathbf{H}_k\hat{\mathbf{x}}_{k/k-1}) \tag{2.7.16}$$

where **K** is the Kalman gain matrix at epoch *k*, the covariance matrix of the predicted state $\hat{\mathbf{x}}_{k/k-1}$ is $\mathbf{D}_{\hat{\mathbf{x}}_{k/k-1}} = \mathbf{\Phi}_k\mathbf{D}_{\hat{\mathbf{x}}_{k-1}}\mathbf{\Phi}_k^T + \mathbf{\Lambda}_k\mathbf{Q}_k\mathbf{\Lambda}_k^T$. Consequently, three residual vectors ($\mathbf{v}_z$, $\mathbf{v}_w$ and $\mathbf{v}_x$) for three measurement vectors ($\mathbf{l}_z$, $\mathbf{l}_w$ and $\mathbf{l}_x$) are actually correlated with each other through the same innovation vector ($\mathbf{d} = \mathbf{z}_k - \mathbf{H}_k\hat{\mathbf{x}}_{k/k-1}$). In addition, the corresponding redundancy indices for each measurement group are [Wang 1997]:

$$\mathbf{r}_{\mathbf{l}_z} = \mathrm{tr}(\mathbf{I} - \mathbf{H}_k\mathbf{K}) \tag{2.7.17}$$

$$\mathbf{r}_{\mathbf{l}_w} = \mathrm{tr}(\mathbf{Q}_k \mathbf{\Lambda}_k^T \mathbf{H}_k^T \mathbf{D}_{\mathbf{d}}^{-1} \mathbf{H}_k \mathbf{\Lambda}_k) \tag{2.7.18}$$

$$\mathbf{r}_{\mathbf{l}_x} = \mathrm{tr}(\mathbf{\Phi}_k \mathbf{D}_{\hat{\mathbf{x}}_{k-1}} \mathbf{\Phi}_k^T \mathbf{H}_k^T \mathbf{D}_{\mathbf{d}}^{-1} \mathbf{H}_K) \tag{2.7.19}$$

where the total system redundancy $\mathbf{r} = \mathbf{r}_{\mathbf{lz}} + \mathbf{r}_{\mathbf{lw}} + \mathbf{r}_{\mathbf{lx}}$, and the covariance matrix of the innovation vector is denoted by $\mathbf{D}_d = \mathbf{R}_k + \mathbf{H}_k \mathbf{D}_{\hat{x}_{k/k-1}} \mathbf{H}_k^T$. Accordingly, for an arbitrary epoch *k*, the variance factors of the three measurement groups defined in (2.7.13) can be estimated as follows:

$$\sigma_{0j}^2(k) = \mathbf{v}_j^T \mathbf{D}_{\mathbf{L}_j}^{-1} \mathbf{v}_j / \mathbf{r}_j \quad (j = \mathbf{z}, \mathbf{w}, \mathbf{x}) \tag{2.7.20}$$

As for a global variance component estimate over a specific or the whole time duration, a simple accumulation can obtain a reliable estimate due to the cross-epoch-orthogonal properties of the measurement residuals [Wang, 1997]. For example, the global variance component up to the current epoch *k* can be computed as:

$$\sigma_{0j}^2(k \mid 1...k) = \frac{\sum_{i=1}^{k} \mathbf{v}_{j_i}^T \mathbf{D}_{\mathbf{l}_j}^{-1} \mathbf{v}_{j_i}}{\sum_{i=1}^{k} \mathbf{r}_{j_i}} \quad (j = \mathbf{z}, \mathbf{w}, \mathbf{x}) \tag{2.7.21}$$

Commonly, the components in the process noise vector $\mathbf{w}_k$ and the measurement noise vector $\boldsymbol{\varepsilon}_k$ are modeled as uncorrelated, i.e., $\mathbf{Q}_k$ and $\mathbf{R}_k$ are both diagonal. As a result, the redundant index of each independent component in either $\mathbf{w}_k$ or $\boldsymbol{\varepsilon}_k$ is given by

$$r_{\mathbf{l}_z}^{i_z} = 1.0 - (\mathbf{H}_k \mathbf{K})_{i_z i_z} \qquad (i_z = 1 \ldots n) \tag{2.7.22}$$

$$r_{\mathbf{l}_w}^{j_w} = (\mathbf{Q}_k \mathbf{\Lambda}_k^T \mathbf{H}_k^T \mathbf{D}_{\mathbf{d}}^{-1} \mathbf{H}_k \mathbf{\Lambda}_k)_{j_w j_w} \qquad (j_w = 1 \ldots m) \tag{2.7.23}$$

Accordingly, the individual variance components for $\mathbf{w}_k$ and $\boldsymbol{\varepsilon}_k$ can be estimated in analogy to (2.7.20) and (2.7.21).

Intuitively inspired by VCE techniques in Kalman filter, the author of this dissertation attempted to analyze the random errors of the sensors measurements during a navigation mission. However, in the traditional indirect (error-state based) Kalman filter, above VCE technique is not suitable for the tuning of an individual sensor's stochastic model because the random errors from aiding sensors and inertial sensors are always blended in the error measurements. This issue is naturally avoided by taking advantage of the novel integration strategy [Qian et al, 2013, 2015; Wang et al, 2014, 2015]. Chapter 5 details the application of VCE technique in GMIKF in order to tune stochastic models of the process and measurement noise vectors toward their truth.

3. York University Multisensor Integrated Kinematic Positioning and Navigation System

This chapter is mainly based on the published work:

Kun, Qian; Jianguo Wang; Nilesh Gopaul and Baoxin Hu (2012): ***Low Cost Multisensor Kinematic Positioning and Navigation System with Linux/RTAI,*** Journal of Sensor and Actuator Networks, 2012, 1, 166-182, DOI10.3390/jsan1030166, www.mdpi.com/journal/jsan/.

Facing the challenge of developing a high-performance yet economically feasible (low-cost) multisensor positioning and navigation system, the author designed and implemented York University Multisensor Integrated System (YUMIS) based on Linux operating system with RTAI interface so that the raw sensor data can be collected and modified for the follow-up research upon navigation algorithms.

First, the author modified the interrupt handlers in Linux kernel space for parallel port and RS232 serial port as a RTAI process. Benefitting from RTAI IPC and global RTAI buffer, the data collector modules in the form of LXRT programs are also developed in user space for NovAtel OEM GNSS receivers and IMU receivers. In addition, the Coriander software package (https://damien.douxchamps.net/ieee1394/coriander/) is modified so that the raw image data can be collected from dual 1394 cameras made by PointGrey. In the follow-up chapters, this dissertation only discusses the post-processing scenario instead of the real-time one because the innovative integration strategy and the associated variance component estimation on inertial sensors are our research goals.

3.1 Hardware components

YUMIS consists of a central computer (controller) and multiple sensors, e.g., OEM GNSS receivers, IMU, and cameras. In addition to these devices tied with the moving vehicle, another GNSS receiver is usually setup as the static base station close to the working area in order to achieve centimeter positioning accuracy using the double differencing (DD) GNSS technique through DD carrier phase measurements with fixed integer ambiguities.

Figure 3-1 and Figure 3-2 show the conceptual hardware configuration of YUMIS system during a van test. As the prototype of the YUMIS system, one IMU unit, one primary GNSS receiver, and two cameras are placed on the roof of the vehicle. In order to obtain the aiding azimuth information, a second GNSS receiver is tied (together with the primary GNSS receiver) to a metal bar aligned with the vehicle longitudinal direction. The central computer connects to all sensors through their dedicated cables, e.g., RS232 serial cables, FireWire 1394 cables and Ethernet cables.

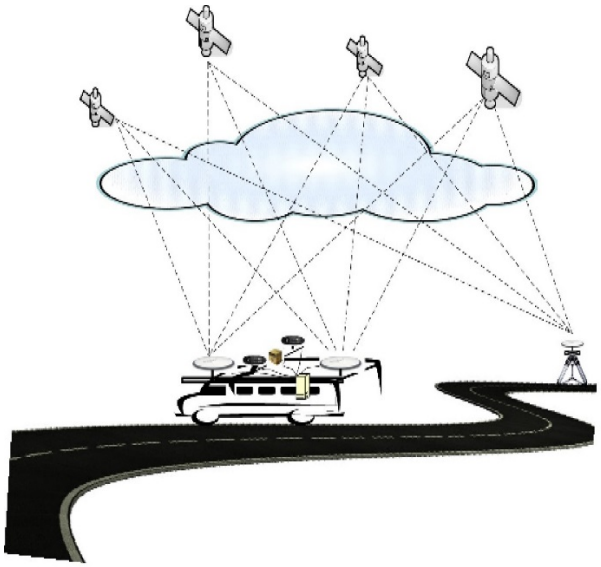


Figure 3-1 YUMIS system in a van test

Figure 3-2 GPS and IMU, and Controller in YUMIS system

3.2 Real time OS kernel

Despite its popularity, the development of a real-time multisensor kinematic positioning and navigation system discourages many researchers and developers due to its complicated hardware environment setup and time consuming device driver development in real-time operating system. These issues can be addressed in a fast and economical manner through using Linux with Real Time Application Interface (Linux/RTAI).

Michael [1997] first designed and developed RTLinux as the hard real-time feature for the Linux operating system in 1996. His work was acquired by Wind River in 2007 and converted into the commercial Wind River Linux. Another main hard real-time Linux variant, RTAI, was independently developed by DIAPM (Dipartimento di Ingegneria Aerospaziale - Politecnico di Milano) in 2000 [Mourot, 2011]. Since then, there have been many successful RTAI applications implemented on the numerical control machines in the manufacturing industry [Shackleford et al, 2001; Morat et al, 2007; Leto et al, 2008; Li et al, 2010]. The application of Linux/RTAI in YUMIS is because it not only demonstrates superior performance but also is well maintained by the RTAI community.

3.2.1 OS Scheduler

In principle, the performance of the scheduler in operation system determines the response time for tasks. Like all other hard real-time operating systems, RTAI scheduler runs with the pre-emptive strategy which is different from the non-pre-emptive task scheduler in a normal Linux system. Figure 3-3 shows the parallel co-existence of a RTAI micro scheduler and a normal Linux scheduler. By modifying the hardware interrupt dispatcher, RTAI scheduler seizes the higher priority than the Linux scheduler does while handling the peripherals' interrupts (hard external interrupt, timer, signal, message, etc.)

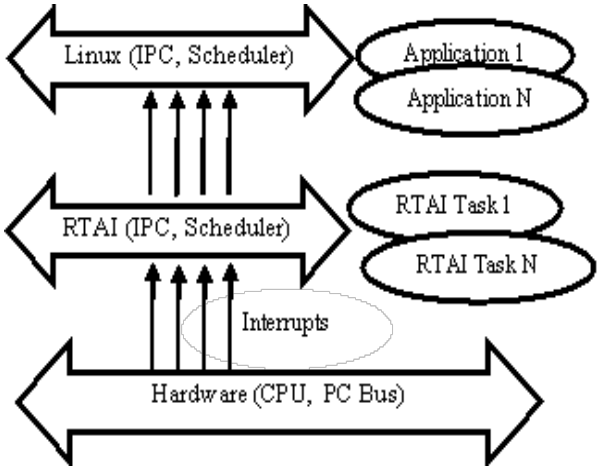


Figure 3-3 RTAI architecture (Mourot, 2011)

Three striking characteristics of RTAI need to be pointed out. First, the RTAI micro kernel guarantees that the higher execution priorities are assigned to the hard real-time RTAI tasks than those to the normal Linux tasks by intercepting and redistributing all interrupts and signals. Thus, all real-time tasks created under RTAI context always have the superior execution priorities when competing with the normal Linux tasks. Second, RTAI avoids uncontrolled hard disk I/O time in the real-time module, which can become disastrous while doing virtual memory swapping

[Michael, 1997; Aeolean Inc., 2002]. Third, RTAI provides the flexible real-time inter-process communication (IPC) methods, e.g., pipes and message boxes, to allow the applications in user-space to communicate with the real-time modules.

In order to verify its hard real-time performance, RTAI comes with the test suites (a set of user space applications) which evaluate several critical real-time kernel indices such as the timer latency, the task switching time, number of overruns and so on. In different hardware platforms, these indices might vary. One sample results of RTAI timer latency test using the Jetway mini control board is shown in Figure 3-4, in which the column "ovl max" represents the worst latency measurement during the test. Usually, a system with excellent timing accuracy has the value of "ovl max" being less than 15-20 μs. More performance comparisons among various real-time operating systems can be referred to [Barbalace, 2008].

```
## RTAI latency calibration tool ##
# period = 100000 (ns)
# avrgtime = 1 (s)
# do not use the FPU
# start the timer
# timer_mode is oneshot

RTAI Testsuite - KERNEL latency (all data in nanoseconds)
RTH|    lat min|    ovl min|    lat avg|    lat max|    ovl max|    overruns
RTD|      -1148|      -1148|       -713|       7395|       7395|          0
RTD|      -1065|      -1148|       -703|       4598|       7395|          0
RTD|       -930|      -1148|       -741|       7350|       7395|          0
RTD|      -1028|      -1148|       -722|      10582|      10582|          0
RTD|      -1043|      -1148|       -702|       8557|      10582|          0
RTD|      -1088|      -1148|       -730|       6795|      10582|          0
RTD|      -1148|      -1148|       -754|       8340|      10582|          0
RTD|      -1178|      -1178|       -755|       6600|      10582|          0
RTD|      -1088|      -1178|       -780|       5003|      10582|          0
```

Figure 3-4 Real-time kernel performance test result

3.2.2 Task management

A task is a process or thread in an operating system (OS). In general, multiple tasks co-exist in a multitasking OS with their own dedicated resources (register, stack, etc.). Due to the limited computing resources, the OS scheduler needs to control (switches) all tasks' working statuses by

allocating or confiscating resources such as CPU, memory, file, etc. An active task can be in three statuses: running, ready, or blocked [Kamal, 2008].

As pointed out in Section 3.2.1, the tasks in a Linux/RTAI system can be divided into RTAI tasks and normal Linux tasks. By default, a RTAI task is registered as a soft real-time task [Michael, 1997; Jones, 2012]. A soft real-time RTAI task is indeed a normal Linux task which is able to communicate with other RTAI hard/soft real-time threads/modules. In order to achieve the superior execution priorities with the quicker responses in a Linux/RTAI system, the hard real-time RTAI tasks need to distinguish themselves from the soft real-time RTAI tasks through the task registration. Because that the RTAI scheduler will not hand over the CPU to the normal Linux's scheduler until all corresponding RTAI tasks finish their work or block themselves, the RTAI tasks' superior priorities are guaranteed. In addition, the transparency of the existence of hard real-time RTAI tasks to all other Linux programs maximizes the system's utilization so that hard real-time tasks can work together with other normal programs in one machine. This dual-OS-scheduler architecture not only allows our system to take the full advantages of all existing Linux features but also ensures the proper balance of the CPU power between the Linux programs and the real-time RTAI tasks.

3.2.3 LXRT service/module

Most of the non-pre-emptive multitask OSs including Linux separate the running program into system (kernel) space module and user space module. Usually, a program starts from being user space task and when necessary, enters into the kernel space by invoking the system API. Then, it returns back to the user space once the system API is returned. It is noteworthy that the module in kernel space can crash the system simply because of one careless hardware I/O

instruction. Therefore, for the sake of the robustness and security of the system, the user space part in Linux is prohibited from directly accessing to the hardware registers so that the system hardware registers are protected from any unauthorized vandalism.

Similarly, the real-time program is separated into two components: the bottom half and the top half. The top half is coded as a real-time kernel module in the kernel space to quickly respond to the incoming data/signals/interrupts. It then pushes the incoming data/signals/interrupts to the bottom half which could be a data processing module in user space running time consuming algorithm. The top half has to acquire and forward the data within a limited amount of time, otherwise the accumulated incoming data can cause delayed outputs or crash the system in the worst case. It is always the developer's responsibility to ensure the real-time task meets the time constraint.

In the case of Linux/RTAI, the most time-constrained task shall be coded as hard real-time RTAI task in order to achieve the quickest response and hence the best performance. It is well-known that debugging a fragile kernel (real-time) module is a painful job because of the frequent system crashes and the anti-intuitive and unfriendly debug tools. As a main advantage of LXRT, a real-time LXRT module can be realized under a user space LXRT environment and also use all the application programming interface (API) available in the user space [Perter, 2004; RTAI API documentation, 2013]. In addition, one can enjoy the luxury of a user space graphic debugger to debug the hard real-time module without worrying about the system crashes. Moreover, it is very convenient to convert a Linux application into a RTAI/LXRT application through only a few lines of the source codes.

In theory, a LXRT hard real-time module consumes a few more microseconds than a traditional hard real-time module does. However, because the influences of the microsecond time

latency on the navigation solution is insignificant for the land vehicle kinematic positioning and navigation system [Ding, 2008], a LXRT module running in user space can be a full substitute for the traditional kernel module in terms of functionality and performance.

3.3 System software architecture

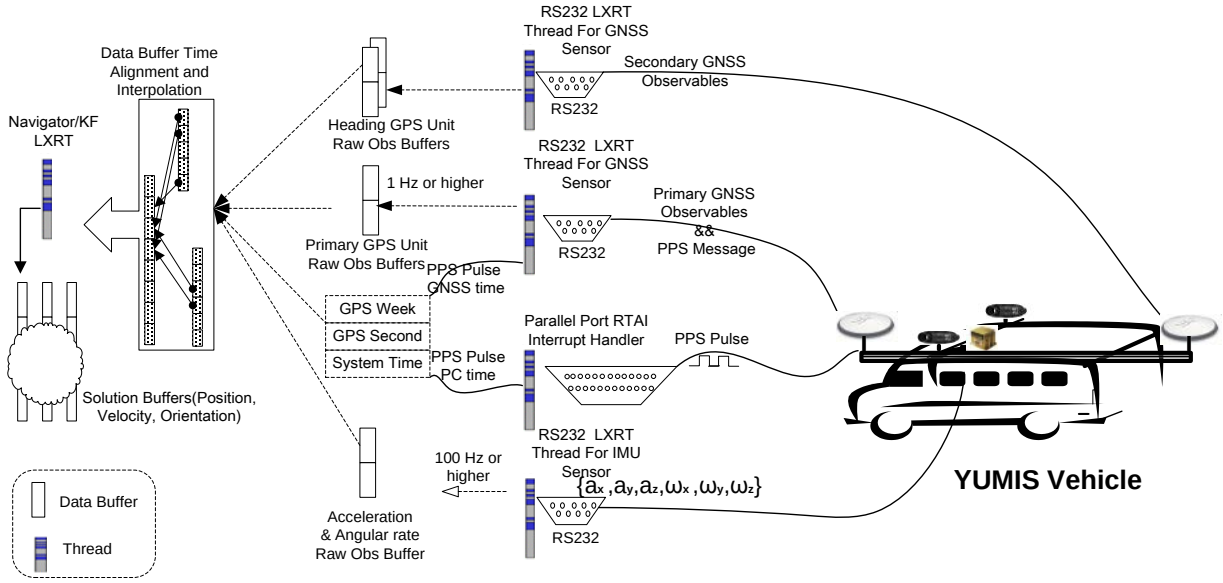


Figure 3-5 YUMIS software architecture

The software system in YUMIS (Figure 3-5) is built upon four major software components: data collectors, time-tagging module using GNSS time, the data buffers, and data processors.

3.3.1 Data collector

The responsibility of a data collector is to grab the raw data from various sensors and transfer them into a data buffer the navigator algorithm can access. As in Section 3.2.3, the data collector in YUMIS is a combination of a device driver (in kernel space) and a LXRT thread (in user space). For example, in the case of the sensor connecting to YUMIS using a RS232 serial ports, the

RS232 device driver in kernel space (the bottom half) first intercepts the sensor data and push them into RTAI global buffers. Then the data collector in another LXRT thread access the raw sensor data through RTAI APIs and relay them into another data buffer visited by the navigator later on (Figure 3-6). In order to capture the data with the minimum time delay, the device driver supporting RS232 serial port communication in YUMIS is developed as a hard real-time RTAI task in kernel space, which interfaces RTAI Global buffers and hardware registers of RS232 serial ports.

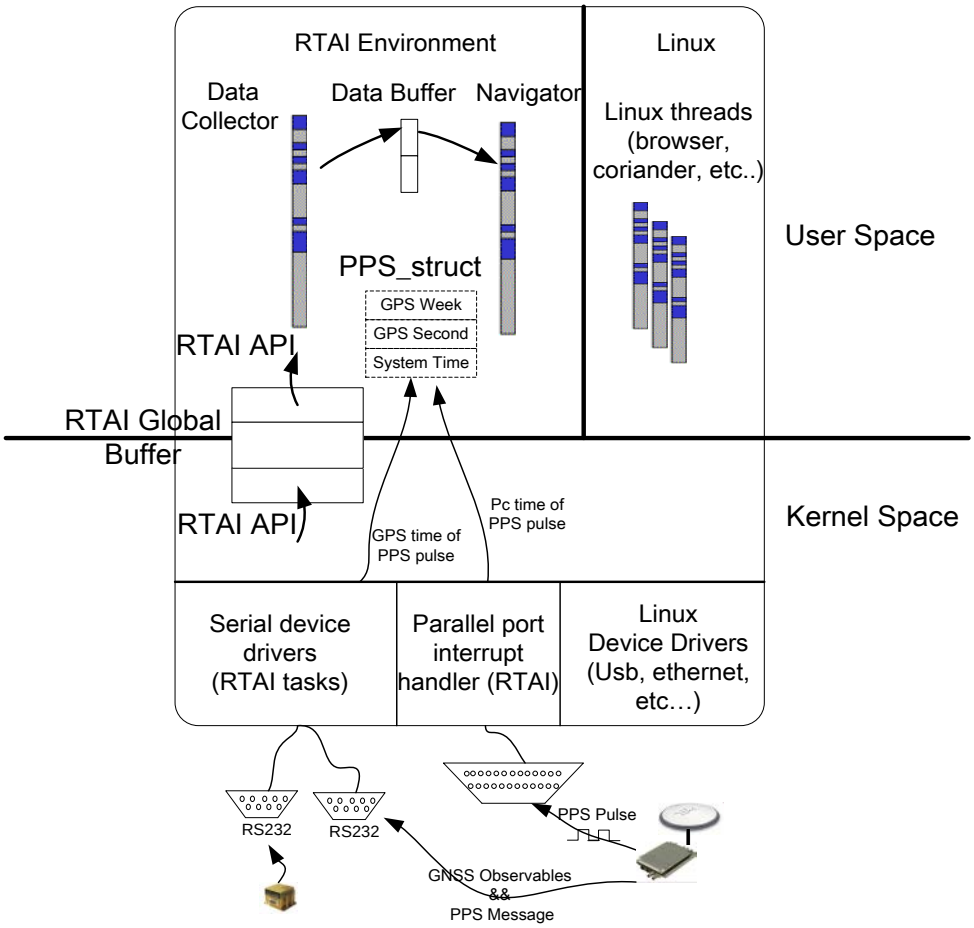


Figure 3-6 Data collector in real time mode

In contrast to the data collector in real time mode, the data collector in the post-processing mode (Figure 3-7) is developed as a file reader which sequentially reads the sensor measurements from the binary data file and stores them into buffers without concerning the timing latency.

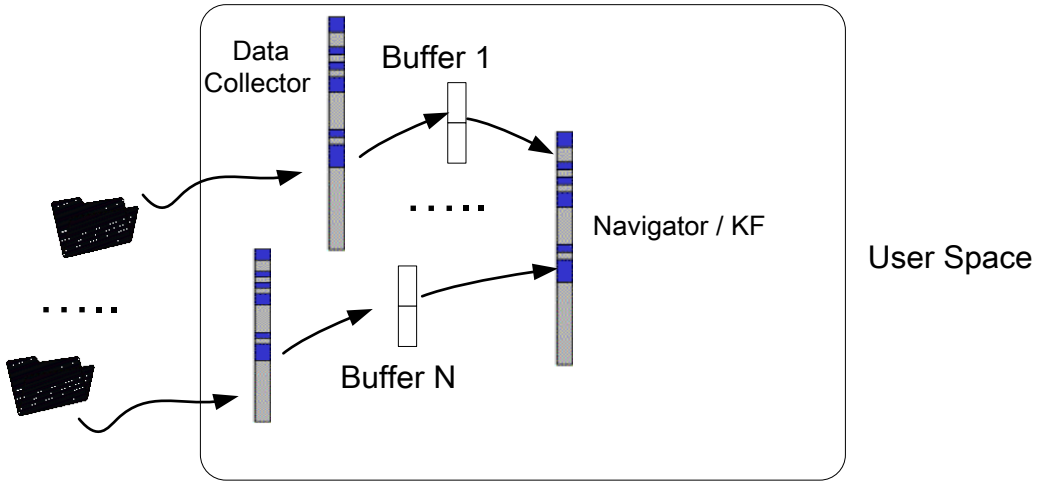


Figure 3-7 Data collector in post-processing mode

3.3.2 Data buffer

Corresponding to each sensor in YUMIS, one dedicated data collecting thread is created and is allocated with a sensor data buffer in use space. The data collector and the navigator shall exchange the sensor data through the data buffers instead of direct thread-to-thread communication. Advantages of this data exchange mechanism are: unlimited reading threads, clear structure, and code efficiency.

In order to avoid any data corruption caused by simultaneous I/O operations on the same data block, each data buffer manages its own mutual exclusive data access (MUTEX) in the multithread environment so that a thread can get the exclusive access permission to read or write the data block (Figure 3-8). It is noted that the configurable size of the buffer (number of blocks) is fixed after it is created so that the time consuming dynamic memory allocation and problematic

memory release operations are prevented. In this mechanism, the system gains the highest data I/O speed without concerning about notorious memory leak.

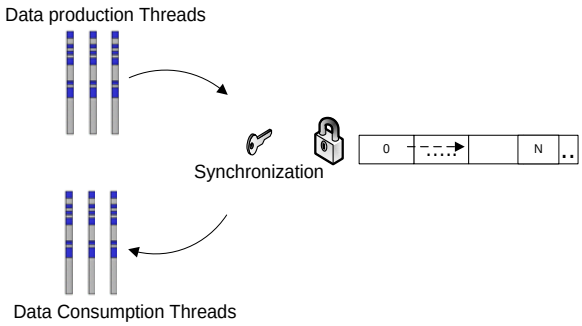


Figure 3-8 Protected data buffer

3.3.3 Time synchronization

3.3.3.1 Standard GNSS time import

The goal of the time synchronization in YUMIS is to tag the acquired sensor data with certain time stamp with respect to the one time reference frame. In nature, we may use the local computer time reference frame as the default frame for all sensors. However, the clock oscillators in most of the desktop computers suffer from drifts so that their long term instabilities cannot satisfy the requirement of high-accuracy timing system (e.g. micro-second level). Moreover, the final navigation solution tagged in the local computer time frame is usually not desirable. Consequently, a stable external clock source with both high resolution and long term stability is crucial to minimize the magnitude of the internal clock's drift and instability in YUMIS.

Nowadays, GPS time has become a standard time frame for a multisensor integrated navigation systems because the PPS (pulse per second) pulse train can be easily retrieved from a GPS OEM unit at the accuracy of a few tens of nanoseconds, for example, 50 ns accuracy from

NovAtel GPS OEM unit. Physically, the 1 Hz timing signal in YUMIS is nothing but a square wave (pulse) with 1ms width (Figure 3-9).

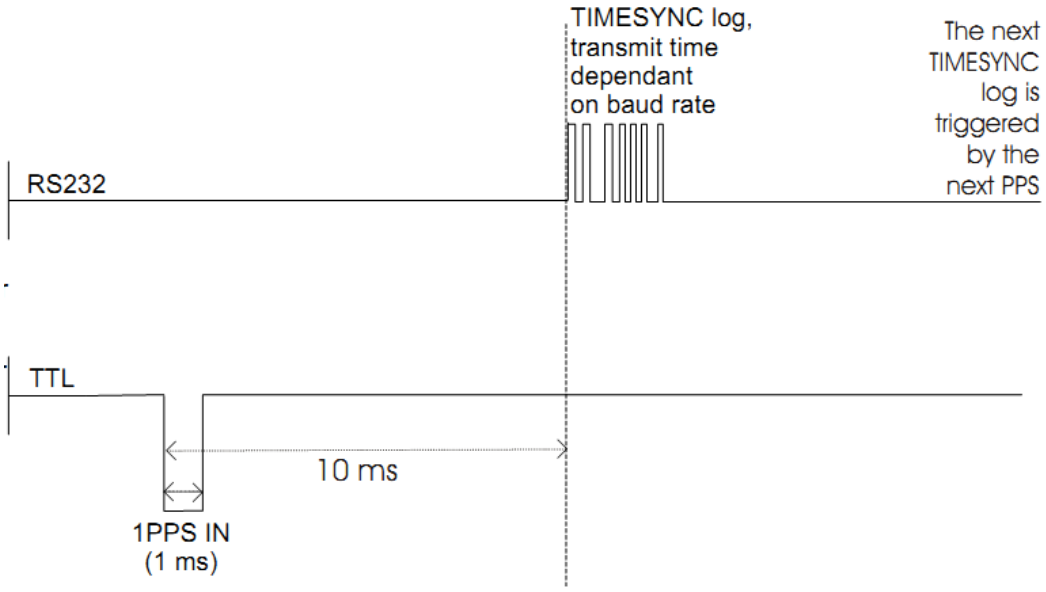


Figure 3-9 PPS pulse in TTL level (NovAtel OEM4 Manual)

The absolute GPS time when the PPS signal voltage drops suddenly (left edge of the PPS wave) is usually dispatched in the form of a special PPS message by the GNSS receiver through a RS232 serial port (Figure 3-10). Once the local computer time corresponding to the PPS signal is known, the GPS time for those data acquired from all other sensors can be computed as follows:

$$T_{sensor} = t_{local} - t_{PPS} + T_{PPS} \tag{3.3.1}$$

where T_{sensor} is the sensor data time stamp in the GPS time frame, t_{local} is the local computer time stamp in the sensor data packet, t_{PPS} is the local computer time stamp for 1 Hz PPS pulse, and T_{PPS} is the time stamp in the GPS time frame corresponding to the 1 Hz PPS.

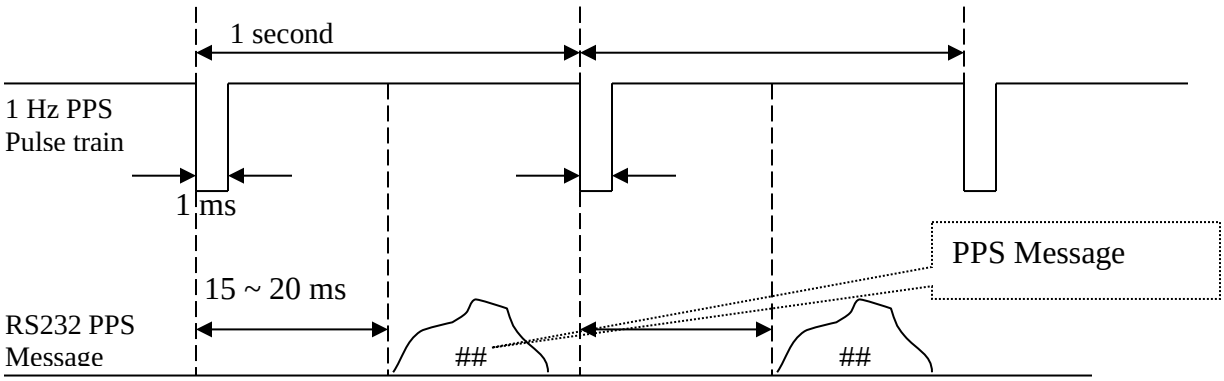


Figure 3-10 PPS signal and message diagram

In YUMIS, the PPS hardware signal from the GNSS receiver is hardwired into pin #10 of the parallel port in order to trigger the interrupt IRQ7, whose processing handler (a RTAI task) intercepts the pulse and retrieves the corresponding local computer time using the Linux system API function “clock_gettime( )” and assigns it into the time synchronization variable “PPS_struct” (Figure 3-5). The GPS time corresponding to the PPS signal is then captured through the RS232 serial port later (e.g., about 20ms with NovAtel OEM GPS receivers). Herein lays the connection between the GPS time and the computer time.

As the most accurate representation of the PPS signal in the local computer time frame, the hardware pulse suffers a lag of a few tens of microseconds due to the interrupt response delay (10 ~ 30 us). However, its influences on the navigation solution are surely ignorable in the YUMIS as a low dynamic system equipped on a land vehicle.

3.3.3.2 Sensor data time-stamp

For each sensor data packet, a time stamp needs to be assigned for the exact time instant when measurements are taken. In YUMIS, because the observables from GNSS receivers are

already time-tagged by the standard GNSS time, only the time synchronization solution for the Crossbow MEMS IMU units is discussed here. All other sensors such as cameras and LiDAR sensors shall apply the same method to tag their data (images or ranges).

Two types of measurements are supported with the Crossbow MEMS IMU sensors: the continuous analog signal and the digital data packet. In both situations, the time delay caused by data acquisition and data transmission has to be taken into account. According to the Crossbow DMU (Dynamic Measurement Unit) manuals [Crossbow 440 Manual, Crossbow DMI user's manual], the Crossbow DMU working in digital data packet mode will experience three stages in term of one data packet (Figure 3-11): sampling (T1), processing (T2) and transferring (T3). On the contrary, the sensor data is presented immediately on the analog output pins after the processing step (T2) is over when a Crossbow DMU is working in the continuous analog signal mode. According to Table 3-1 [Li, 2004], the analog output has a time delay of 2.3 ms with the 400Hz output rate while the scaled digital packet has a time delay of 6.4 ms when the working frequency is 156Hz. In YUMIS, the IMU data are captured through a dedicated RS232 real time serial port module. As a result, the timing delay caused by the software can be assumed an ignorable constant compared to the values in Table 3-1.

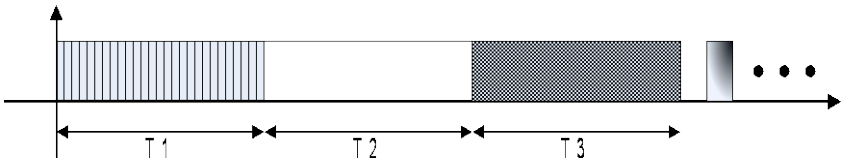


Figure 3-11 Time sequence of the IMU data traffic

Table 3-1 IMU300 Series, IMU400 Series and VG300CB Timing

Operating Mode	T1	T2	T3	Total time	Max Rate
Voltage	0.8 ms	0.5 ms	4.7 ms	6.0 ms	166 Hz
Scaled	0.8 ms	0.9 ms	4.7 ms	6.4 ms	156 Hz
VG(VGX only)	0.8 ms	1.5 ms	5.2 ms	7.5 ms	133 Hz
Analog Output	0.8 ms	1.5 ms	-	2.3 ms	400 Hz

According to the analysis of the timing error tolerance in [Ding, 2008], the manufacturer's calibration results for the time error can be directly applied if the jerk of a moving vehicle is ten times smaller than the absolute magnitude of the system innovation because the estimation error in the final navigation solution caused by the timing error could become ignorable. Thus, the time offset of the IMU data can be assumed to be a virtually constant in YUMIS as the dynamics of the land vehicle is relatively smooth (low dynamic) under normal conditions. That is to say, the known timing offset in Table 3-1 will be directly applied to the time stamps of the raw IMU measurements (assigned when the digital data packets are acquired).

3.3.4 Navigation processor

Essentially, the navigator is nothing but a computer program running a group of numerical algorithms to estimate the navigational parameter of our interests. In YUMIS, if the real-time navigation solution is requested, the navigator shall be implemented as a RTAI thread with high execution priority. In the follow-up chapters, this dissertation only discusses the post-processing scenario instead of the real-time one because the innovative integration strategy and the associated variance component estimation on inertial sensors are our research goals.

3.4 Image sensor integration

Besides GNSS and IMU sensor, two optical image sensors with IEEE firewire 1394 interface are actually integrated in YUMIS system (Figure 3-1) for the future researches work (e.g. visual SLAM or visual odometry). However, because the usage of the image data is beyond the scope of this dissertation, the image sensor data collection is briefed in this section without elaboration.

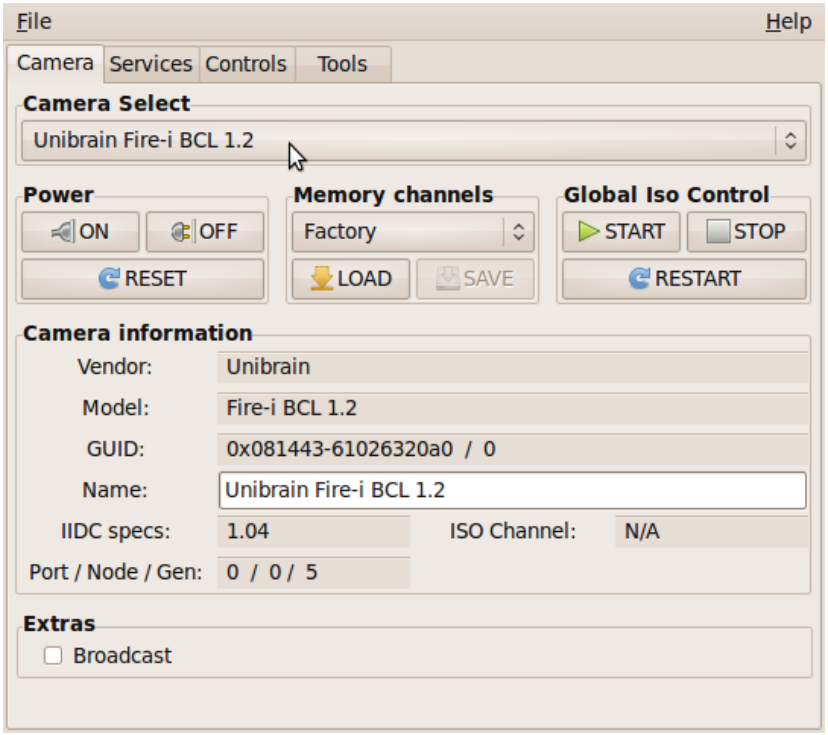


Figure 3-12 Coriander Linux GUI

YUMIS system succeeds in integrating the image sensor through converting the third party open source software – Coriander (http://damien.douxchamps.net/ieee1394/coriander/) into a Linux/RTAI process. Coriander is a full featured Linux GUI program (Figure 3-12) interfacing digital cameras (image sensors). The optical images collected by one image sensor in YUMIS are always sequentially stored in a binary file.

The computer timestamp ($t_{a\text{-}PC}$) for each image is first acquired through certain Linux system APIs. Based on the global variable "PPS_struct" (time synchronization) which is periodically updated by the IRQ7 interrupt handler triggered by GNSS receiver's PPS signal pulse, YUMIS system can achieve the standard GPS time $t_{a\text{-}GPS}$ corresponding to $t_{a\text{-}PC}$. In addition, the time latency ($|t| = t_{t\text{-}GPS} - t_{a\text{-}GPS} = t_{t\text{-}PC} - t_{a\text{-}PC}$) between the moment when the image data is available ($t_{a\text{-}GPS}$) and the exact moment when the image is being taken ($t_{t\text{-}GPS}$) can be obtained by reading the hardware register of the image sensor [Point Grey, 2011]. Finally, the GPS time for the image is computed as:

$$t_{t\text{-}GPS} = t_{a\text{-}GPS} + |t| \tag{3.4.1}$$

3.5 Summary

This chapter introduces the architecture of the YUMIS system from the perspective of both hardware and software. As the pre-requisite component of the multisensory system, real-time kernel (Linux/RTAI) is successfully applied in YUMIS system. Besides, a generic system architecture is also proposed for a multisensory positioning and navigation system allowing for the whole life cycle of the sensor data, e.g., data collection, data processing and time synchronization. YUMIS presents a low-cost alternative to the expensive commercial system, which can acquire real geospatial data from multiple sensors such as the IMU units, the OEM GNSS sensors and cameras. YUMIS demonstrates not only excellent hard real-time performance but also offers the convenience in the real-time software development for multisensor integration. Moreover, YUMIS lays the foundation for the follow-up training of highly qualified Geomatics Engineers at the EOL laboratory at York University.

4. Generic integration strategy for multisensor integrated kinematic positioning and navigation

This chapter describes a generic multisensor integration strategy for multisensor integrated kinematic positioning and navigation using GNSS receivers and IMUs [Qian et al, 2013, 2015; Wang et al, 2014, 2015] and is mainly based on the published work:

Qian, Kun; Wang, Jianguo and Hu, Baoxin (2015): ***Novel integration strategy for GNSS-aided inertial integrated navigation***, *Geomatica*, 2015, 2, Vol. 69, pp. 217-230.

Wang, Jianguo; *Kun Qian* and Baoxin Hu (2015): ***An Unconventional Full Tightly-Coupled Multi-Sensor Integration for Kinematic Positioning and Navigation***, Chapter 65, in J. Sun et al. (eds), China Satellite Navigation Conference (CSNC) 2015 Proceedings: Volume III, Volume 342 of the series Lecture Notes in Electrical Engineering, pp. 753-765, Springer-Verlag Berlin Heidelberg 2015.

The core of a multisensor integrated navigation system is the fusion of the measurement data from the IMU and other heterogeneous aiding sensors such as GNSS receivers etc. In multisensor-aided inertial integrated kinematic positioning and navigation, there are four different integration architectures: uncoupled, loosely-coupled, tightly-coupled and deeply-coupled [Greenspan, 1996; Schmidt, 2010; etc.] as summarized in Section 2.6.4.2. Under each integration architecture, there are so-called direct and indirect methods to realize the estimator, e.g., Kalman filter [Yi and Wang, 1987; Greenspan, 1996; Qi and Moore, 2002; Giroux et al, 2005; Munguía, 2014; etc.]. The former uses the whole-value states (navigation parameters) while the latter estimates the error states.

As a traditional and dominant integration strategy in the engineering practice, the indirect integration mechanism consists of two parallel components [Jekeli, 2001; Rogers, 2003; Titterton, 2004; Bekir, 2007; Grove, 2008]: inertial navigation mechanization and the associated optimal estimation of the error states rooted in the linearized error models for the navigation parameters and the IMU's systematic errors. Because the inertial navigation mechanization between two adjacent aiding measurements is driven by the IMU measurements, the performance of the conventional integration strategy is closely related to the *a priori* error characteristics of the IMU. In the system using a low-cost IMU, the error models around the mechanization solution may suffer from the intolerable drift of IMU systematic errors, especially, during outages of aiding measurements, for instance, GNSS outages in poor GNSS environment. Some researchers have suggested to apply the complementary parametric constraints, e.g., assuming zero velocity along the vertical axis [Shin et al, 2001; Ma et al, 2003; Godha, 2006; G. Dissanayake et al, 2001; Wang and Gao, 2010]. However, these virtual measurements are not rigorously modelled and even cause serious position drifting if these constraints are breached.

What else can one do to suppress the influences of the random errors on the navigation solution, especially, in the case of the system using a low-cost IMU? A number of researches attempted to construct a standalone system model using the knowledge of the kinematics of a rigid body. Wang [1997] employed a 2D kinematic model as the system model for kinematic positioning and integrated navigation in the Kalman filter. Furthermore, Wang and Sternberg [2000] tentatively utilized the 3D kinematic model after the uniform circular motion as the system model in the Kalman filter with the six whole-value states and the nine measurements for the GNSS-aided IMU integrated navigation.

Along the same path, a generic multisensor integration strategy is proposed [Qian et al, 2013; Wang et al, 2014], by which the rigorous trajectories and smooth rotation dynamics can be naturally regulated over the IMU data interval or a specific limited time interval. The adjective "generic" comes from the fact that IMU sensor will not be differentiated from all other sensors and take part in estimation filter as raw measurements. Consequently, the influences of the time-variant errors of an inertial sensor on navigation solution are mitigated or suppressed by the rigorous kinematic trajectory model.

Because it is realized through the extended Kalman filter, the novel integration strategy is referred as Generic Multisensor Integration Kalman Filter (GMIKF) in this dissertation without further notice. In this chapter, the description of the system model of the proposed GMIKF is followed by the introduction of its state vector and measurement model. Then, the characteristics of the GMIKF is discussed along with the analysis of its performance boundary. At the end, a sample dataset is taken to demonstrate the proposed generic multisensor integration strategy.

4.1 Generic multisensor integration Kalman Filter

4.1.1 The system model

4.1.1.1 Trajectory model after kinematics

The mathematic representation of position, velocity and acceleration of a mechanic system involves two coordinate systems which are moved relatively to each other (Figure 4-1). One system is called the space-fixed navigation system $S_n(o\text{-}xyz)$ while the other is referred to as the moving body system $S_b(o_b\text{-}x_by_bz_b)$. In the context of the strapdown inertial navigation, the navigation parameters of interests are the position vector ($\mathbf{r}_{nb}^{n}$) referred to the local navigation

frame ***n*** and three Euler angles (pitch p, roll γ and heading ψ) describing the attitude of the IMU body frame ***b*** with respect to the local navigation frame ***n***.

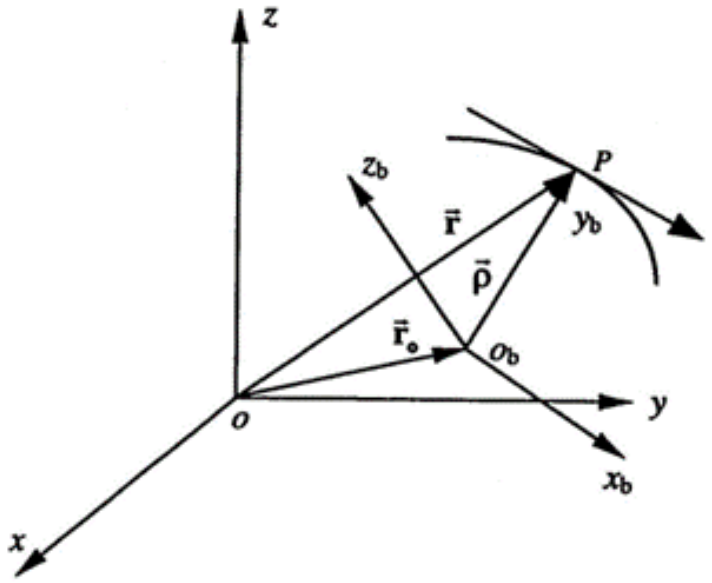


Figure 4-1 Coordinate systems (body frame and navigation frame)

By definition, the first derivative of the position vector $\mathbf{r}_{nb}^{n}$ is the velocity vector $\mathbf{v}_{nb}^{n}$ projected in the navigation frame, which relates to its counterpart velocity vector $\mathbf{v}_{nb}^{b}$ projected in the IMU body frame through a DCM $\mathbf{C}_{b}^{n}$ from ***b*** to ***n***:

$$\dot{\mathbf{r}}_{nb}^{n} = \mathbf{v}_{nb}^{n} = \mathbf{C}_{b}^{n}\mathbf{v}_{nb}^{b} \tag{4.1.1}$$

According to the rule of the vector dynamics [Bekir, 2007], the differential equation of $\mathbf{v}_{nb}^{n}$ is further developed as:

$$\dot{\mathbf{v}}_{nb}^{n} = \dot{\mathbf{C}}_{b}^{n}\mathbf{v}_{nb}^{b} + \mathbf{C}_{b}^{n}\dot{\mathbf{v}}_{nb}^{b} = \mathbf{C}_{b}^{n}\left[\boldsymbol{\omega}_{nb}^{b}\times\right]\mathbf{v}_{nb}^{b} + \mathbf{C}_{b}^{n}\left[\dot{v}_{nbx}^{b} \quad \dot{v}_{nby}^{b} \quad \dot{v}_{nbz}^{b}\right]^{T} \tag{4.1.2}$$

Similarly, the differential equation of the acceleration vector $\mathbf{a}_{nb}^{n}$ is further developed as follow:

$$\dot{\mathbf{a}}_{nb}^{n} = \dot{\mathbf{C}}_{b}^{n}\mathbf{a}_{nb}^{b} + \mathbf{C}_{b}^{n}\dot{\mathbf{a}}_{nb}^{b} = \mathbf{C}_{b}^{n}[\boldsymbol{\omega}_{nb}^{b}\times]\mathbf{v}_{nb}^{b} + \mathbf{C}_{b}^{n}\left[\dot{a}_{nbx}^{b} \quad \dot{a}_{nby}^{b} \quad \dot{a}_{nbz}^{b}\right]^{T} \tag{4.1.3}$$

Additionally, by the definition of the velocity, acceleration and jerk in kinematics, one has

$$\dot{\mathbf{v}}_{nb}^{n} = \mathbf{a}_{nb}^{n} = \mathbf{C}_{b}^{n}\mathbf{a}_{nb}^{b} \tag{4.1.4}$$

$$\dot{\mathbf{a}}_{nb}^{n} = \mathbf{j}_{nb}^{n} = \mathbf{C}_{b}^{n}\mathbf{j}_{nb}^{b} \tag{4.1.5}$$

wherein $\mathbf{v}_{nb}^{b}$, $\mathbf{a}_{nb}^{b}$, $\mathbf{j}_{nb}^{b}$, $\boldsymbol{\omega}_{nb}^{b}$ are the velocity, acceleration, jerk and angular velocity vector in the IMU body frame ***b***, respectively, and the $\mathbf{C}_{b}^{n}$ is the direction cosine matrix (DCM) from the body frame ***b*** to the navigation frame ***n***.

Based on (4.1.2) – (4.1.5), the derivatives of the velocity vector $\mathbf{v}_{nb}^{b}$ and the acceleration vector $\mathbf{a}_{nb}^{b}$ in body frame are given as [Qian et al, 2013]:

$$\dot{\mathbf{v}}_{nb}^{b} = \begin{bmatrix} \dot{v}_{nbx}^{b} \\ \dot{v}_{nby}^{b} \\ \dot{v}_{nbz}^{b} \end{bmatrix} = \begin{bmatrix} a_{nbx}^{b} \\ a_{nby}^{b} \\ a_{nbz}^{b} \end{bmatrix} - \begin{bmatrix} 0 & -\omega_{nbz}^{b} & \omega_{nby}^{b} \\ \omega_{nbz}^{b} & 0 & -\omega_{nbx}^{b} \\ -\omega_{nby}^{b} & \omega_{nbx}^{b} & 0 \end{bmatrix} \begin{bmatrix} v_{nbx}^{b} \\ v_{nby}^{b} \\ v_{nbz}^{b} \end{bmatrix} = \mathbf{a}_{nb}^{b} - \boldsymbol{\omega}_{nb}^{b} \times \mathbf{v}_{nb}^{b} \tag{4.1.6}$$

$$\dot{\mathbf{a}}_{nb}^{b} = \begin{bmatrix} \dot{a}_{nbx}^{b} \\ \dot{a}_{nby}^{b} \\ \dot{a}_{nbz}^{b} \end{bmatrix} = \begin{bmatrix} j_{nbx}^{b} \\ j_{nby}^{b} \\ j_{nbz}^{b} \end{bmatrix} - \begin{bmatrix} 0 & -\omega_{nbz}^{b} & \omega_{nby}^{b} \\ \omega_{nbz}^{b} & 0 & -\omega_{nbx}^{b} \\ -\omega_{nby}^{b} & \omega_{nbx}^{b} & 0 \end{bmatrix} \begin{bmatrix} a_{nbx}^{b} \\ a_{nby}^{b} \\ a_{nbz}^{b} \end{bmatrix} = \mathbf{j}_{nb}^{b} - \boldsymbol{\omega}_{nb}^{b} \times \mathbf{a}_{nb}^{b} \tag{4.1.7}$$

wherein are

$j_{nbx}^{b}, j_{nby}^{b}, j_{nbz}^{b}$	the body jerk scalar components,
$\omega_{nbx}^{b}, \omega_{nby}^{b}, \omega_{nbz}^{b}$	the body rotation scalar components,
$v_{nbx}^{b}, v_{nby}^{b}, v_{nbz}^{b}$	the body velocity scalar components, and
$a_{nbx}^{b}, a_{nby}^{b}, a_{nbz}^{b}$	the body acceleration scalar components.

In summary, three equations (4.1.1), (4.1.6) and (4.1.7) describe the 3D trajectory of a rigid body using the velocity, acceleration, jerk and angular rate vector. Particularly, the acceleration

equation (4.1.7) is missing in the traditional INS mechanization, which is one of the key points of the GMIKF: accuracy improvement through the acceleration prediction based on a rigorous trajectory model.

4.1.1.2 Attitude model

There are four approaches for attitude representation: DCM, Euler angles, rotation vector and quaternion. The vehicle attitude (state components) in this dissertation is realized by three Euler angles in the local ENU (east-north-up) navigation frame as pitch, roll and heading. In strapdown inertial navigation, the differential equation of the three Euler angles is as follows [Magnus, 1971; Bekir, 2007; etc.]:

$$\begin{bmatrix} \dot{p} \\ \dot{\gamma} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} c\gamma & 0 & s\gamma \\ s\gamma\, tgp & 1 & -c\gamma\, tgp \\ s\gamma \sec p & 0 & -c\gamma \sec p \end{bmatrix} \begin{bmatrix} \omega_{nbx}^{b} \\ \omega_{nby}^{b} \\ \omega_{nbz}^{b} \end{bmatrix} = C_{3\times3}\boldsymbol{\omega}_{nb}^{b} \tag{4.1.8}$$

wherein are

p, γ, ψ three Euler angles (pitch, roll, heading),

cx, sx the cosine and sine functions of a scalar variable x,

tgx the tangent function of a scalar variable x,

$\sec x$ the secant function of a scalar variable x, and

$C_{3\times3}$ $= \begin{bmatrix} cr & 0 & sr \\ sr\, tgp & 1 & -cr\, tgp \\ sr \sec p & 0 & -cr \sec p \end{bmatrix}$

4.1.1.3 Angular velocity (rate) model

Similar to the differential equation of the acceleration vector in 4.1.7, a realistic system model for angular rates in the GMIKF is also developed to enhance the system performance. Because YUMIS is equipped with a land vehicle, the land vehicle's dynamics is of our interests. According to the vehicle dynamics, the land vehicle's angular acceleration vector ($\boldsymbol{\alpha}$) is actually related to the torque ($\boldsymbol{\tau}$) and its moment of inertial ($\boldsymbol{I}$) as follows:

$$\boldsymbol{\alpha} = \dot{\boldsymbol{\omega}} = \frac{\boldsymbol{\tau}}{\boldsymbol{I}} \tag{4.1.9}$$

Obviously, the computation of these quantities requires the vehicle's physical properties such as dimension, mass distribution and etc. In the case of the spacecraft with known physical properties (e.g., mass, size, and so on), the angular rate vector can be alternatively modeled via the inertial tensor and angular momentum [Bar-Itzhack, 2004].

On the other hand, in navigation, given unknown physical models of the land vehicle, the angular rate is often modelled as a Gauss-Markov process [Singer, 1970; Brown and Hwang, 1997; Li & Jilkov, 2000]. Furthermore, under the normal driving condition with a smooth steering, the three components in the body-frame angular rate vector $\boldsymbol{\omega}_{nb}^{b} = \begin{bmatrix} \omega_{nbx}^{b} & \omega_{nby}^{b} & \omega_{nbz}^{b} \end{bmatrix}^{T}$ of a land vehicle over a short time interval can be modeled as three independent random processes disturbed by random noises, of which ω_{nbx}^{b} and ω_{nby}^{b} are two scalar zero-mean random processes and ω_{nbz}^{b} is a non-zero mean random process. Usually, the first order Gauss-Markov process is adequate to carve the dynamics of a zero-mean random variable through exploring the correlation between two random variables at two consecutive epochs. For example, the zero-mean Singer

motion model [Singer, 1970] expresses the dynamic variation of two quasi-zero angular rate components ω_{nbx}^{b} and ω_{nby}^{b} of the land vehicle as follows:

$$\dot{\omega}_{nbx}^{b} = -\beta_{x}\omega_{nbx}^{b} + w_{x}(t) \tag{4.1.10}$$

$$\dot{\omega}_{nby}^{b} = -\beta_{y}\omega_{nby}^{b} + w_{y}(t) \tag{4.1.11}$$

Meanwhile, the "current" model as the modified Singer model [Zhou et al, 1984; Li and Jilkov, 2000] is adapted to describe the dynamics of the angular rate ω_{nbz}^{b} around its non-zero mean $\overline{\omega}_{nbz}^{b}$.

$$\dot{\omega}_{nbz}^{b} = -\beta_{z}\omega_{nbz}^{b} + \beta_{z}\overline{\omega}_{nbz}^{b} + w_{z}(t) \tag{4.1.12}$$

Actually, in practice, a constant $\boldsymbol{\omega}_{nb}^{b}(t)$ could also be a reasonable assumption, when: 1) the sensor sampling rate is high enough so that the better approximation (more complicated model) may not yield a significantly better accuracy, 2) the error levels of sensors are of the same order as the skipped higher order terms in the constancy model [Bekir, 2007]. In consideration of its simplicity and feasibility on the low-cost MEMS IMU based land vehicle inertial navigation, $\boldsymbol{\omega}_{nb}^{b}(t)$ in GMIKF is modeled as a constant vector contaminated by certain process noises during a short time period over which the state vector prediction proceeds.

4.1.2 Formulation of the generic multisensor integration Kalman filter

4.1.2.1 The state vector

The state vector of an estimator in navigation usually consists of the core navigation states, inertial sensor states and the supplementary states. For example, in the traditional inertial navigation system, the core navigation states are the navigational errors of the inertial mechanization results (position errors, velocity errors and attitude errors), and the inertial sensor states are the inertial sensor's non-white (systematic) errors, e.g., turn-on biases, bias drifts. While

the inertial sensor states in GMIKF still remain the same, the core navigation states in GMIKF are redefined to be the vehicle's positions, velocities, accelerations, attitudes, and angular rates.

Moreover, in both cases, some supplementary states are needed to complete the measurement equations to model the aiding sensor parameters, e.g., GNSS receiver's clock drift, the ambiguity integer numbers for the GNSS carrier phase observables, lever arm vectors of the sensors, and etc.

In consideration of the integrity of the system model targeting a low-cost inertial navigation system, the 21 state variables (15 core navigation states and 6 inertial sensor states) are selected as follows:

$$\mathbf{X}_{21\times1} = \left[\mathbf{r}_{1\times3} \quad \mathbf{v}^{b}_{nb(1\times3)} \quad \mathbf{a}^{b}_{nb(1\times3)} \quad \boldsymbol{\theta}_{(1\times3)} \quad \boldsymbol{\omega}^{b}_{nb(1\times3)} \quad \mathbf{b}_{\omega(1\times3)} \quad \mathbf{b}_{a(1\times3)}\right]^{T} \tag{4.1.13}$$

herein

$$\mathbf{r}_{1\times3} = \begin{bmatrix} e & n & u \end{bmatrix}, \quad \mathbf{v}^{b}_{nb\,(1\times3)} = \begin{bmatrix} v^{b}_{nbx} & v^{b}_{nby} & v^{b}_{nbz} \end{bmatrix},$$

$$\mathbf{a}^{b}_{nb\,(1\times3)} = \begin{bmatrix} a^{b}_{nbx} & a^{b}_{nby} & a^{b}_{nbz} \end{bmatrix},$$

$$\boldsymbol{\theta}_{(1\times3)} = \begin{bmatrix} p & \gamma & \psi \end{bmatrix}, \quad \boldsymbol{\omega}^{b}_{nb\,(1\times3)} = \begin{bmatrix} \omega^{b}_{nbx} & \omega^{b}_{nby} & \omega^{b}_{nbz} \end{bmatrix},$$

$$\mathbf{b}_{\boldsymbol{\omega}\,(1\times3)} = \begin{bmatrix} b_{\boldsymbol{\omega}_x} & b_{\boldsymbol{\omega}_y} & b_{\boldsymbol{\omega}_z} \end{bmatrix}, \quad \mathbf{b}_{\mathbf{a}\,(1\times3)} = \begin{bmatrix} b_{\mathbf{a}_x} & b_{\mathbf{a}_y} & b_{\mathbf{a}_z} \end{bmatrix}$$

with

e	n	u	the east, north, up position components,
v^{b}_{nbx}	v^{b}_{nby}	v^{b}_{nbz}	three body velocity components,
a^{b}_{nbx}	a^{b}_{nby}	a^{b}_{nbz}	three body acceleration components,
p	γ	ψ	three Euler angels of pitch, roll and heading angles,
ω^{b}_{nbx}	ω^{b}_{nby}	ω^{b}_{nbz}	the body angular rate components,

$\mathbf{b}_{\omega(1\times3)}$ the 1×3 gyroscope bias vector including start off biases and bias residuals, and

$\mathbf{b}_{a(1\times3)}$ the 1×3 accelerometer bias vector including start off biases and bias residuals.

It is worth mentioning that, the choice of the state vector for estimation filter with GMIS integration strategy varies according to the navigation frame, the navigational parameters of interests, and the balance between the accuracy and the algorithm complexity. Although the gimbal lock at 90^{o} pitch might removes the ability to use the roll angle, the Euler angle is still selected in this dissertation to represent the attitude is because of 1) its intuitional simplicity; 2) the fact that our land vehicle using YUMIS used in our study never runs into the situation with pitch close to 90^{o}.

4.1.2.2 Discretization of system model

In order to apply the extended Kalman filter, a system model in discrete time is needed for the purpose of the propagation of the states and associated covariance matrix across two discrete time instances. For narrative convenience, the continuous system models of 21 states defined in (4.1.13) are rephrased as follows:

$$\dot{\mathbf{r}}_{nb}^{n} = \mathbf{C}_{b}^{n}\mathbf{v}_{nb}^{b} \tag{4.1.14}$$

$$\dot{\mathbf{v}}_{nb}^{b} = \mathbf{a}_{nb}^{b} - \boldsymbol{\omega}_{nb}^{b} \times \mathbf{v}_{nb}^{b} \tag{4.1.15}$$

$$\dot{\mathbf{a}}_{nb}^{b} = \mathbf{j}_{nb}^{b} - \boldsymbol{\omega}_{nb}^{b} \times \mathbf{a}_{nb}^{b} \tag{4.1.16}$$

$$\dot{\boldsymbol{\theta}} = \mathbf{C}(\boldsymbol{\theta})_{3\times3}\boldsymbol{\omega}_{nb}^{b} \tag{4.1.17}$$

$$\dot{\boldsymbol{\omega}}_{nb}^{b} = \mathbf{0} \tag{4.1.18}$$

$$\dot{\mathbf{b}}_{\omega} = \mathbf{0} \tag{4.1.19}$$

$$\dot{\mathbf{b}}_{a} = \mathbf{0} \tag{4.1.20}$$

For demonstration purpose, the angular rate vector $\boldsymbol{\omega}_{nb}^{b}$, gyro bias vector $\mathbf{b}_{\omega}$ and accelerometer bias vector $\mathbf{b}_{a}$ are all modeled to be random walks associated with certain process noises in YUMIS implementation. However, their models are subject to the changes of the vehicle dynamics, the IMU sampling rate, the IMU type, and so on.

i. Position equation

The computation of the position vector $\mathbf{r}_{nb}^{n}(k+1)$ at epoch *k* +1 can be realized as follows:

$$\mathbf{r}_{nb}^{n}(k+1)=\mathbf{r}_{nb}^{n}(k)+\delta\mathbf{r}_{nb}^{n}=\mathbf{r}_{nb}^{n}(k)+\mathbf{C}_{b}^{n}(k)\int_{k}^{k+1}\mathbf{R}\mathbf{v}_{nb}^{b}dt \tag{4.1.21}$$

The rotation matrix **R** can be approximated using the equivalent rotation vector $\boldsymbol{\sigma}$ [Titterton, 2004]:

$$\mathbf{R}=\mathbf{I}+\frac{\sin\sigma}{\sigma}[\boldsymbol{\sigma}\times]+\frac{(1-\cos\sigma)}{\sigma^{2}}[\boldsymbol{\sigma}\times]^{2} \tag{4.1.22}$$

where $\boldsymbol{\sigma}=\boldsymbol{\alpha}+\boldsymbol{\varepsilon}=\int_{k}^{k+1}\boldsymbol{\omega}dt+\boldsymbol{\varepsilon}$, in which $\boldsymbol{\alpha}=\int_{k}^{k+1}\boldsymbol{\omega}dt$ and $\dot{\boldsymbol{\varepsilon}}$ is the non-commutativity rate vector due to the non-measurable motion, which (coning error) equals [Bortz, 1971]:

$$\dot{\boldsymbol{\varepsilon}}=\frac{1}{2}\boldsymbol{\sigma}\times\boldsymbol{\omega}+\frac{1}{\sigma^{2}}\left[1-\frac{\sigma\sin\sigma}{2(1-\cos\sigma)}\right]\boldsymbol{\sigma}\times\boldsymbol{\sigma}\times\boldsymbol{\omega} \tag{4.1.23}$$

where the scalar σ is the norm of the rotation vector $\boldsymbol{\sigma}$.

Substituting for **R** using its numerical approximation, the delta position vector $\delta\mathbf{r}_{nb}^{n}$ becomes:

$$\delta\mathbf{r}_{nb}^{n}=\mathbf{C}_{b}^{n}(k)\int_{k}^{k+1}\left[\mathbf{v}_{nb}^{b}+\boldsymbol{\sigma}\times\mathbf{v}_{nb}^{b}+0.5[\boldsymbol{\sigma}\times]^{2}\mathbf{v}_{nb}^{b}-\cdots\right]dt \tag{4.1.24}$$

If the second- and higher-order terms are omitted, it gives:

$$\delta\mathbf{r}_{nb}^{n}=\mathbf{C}_{b}^{n}(k)\left[\int_{k}^{k+1}\mathbf{v}_{nb}^{b}\,dt+\int_{k}^{k+1}\boldsymbol{\sigma}\times\mathbf{v}_{nb}^{b}dt\right] \tag{4.1.25}$$

The attitude drift due to the ignored high order terms in (4.1.22) is termed as the attitude loss of the first order attitude approximation, which can be evaluated as [Titterton, 2004]:

$$\delta_{1st} = \frac{1}{dt}(\sigma\cos\sigma - \sin\sigma)$$

when the rigid body is experiencing a single *x*-axis rotation symbolized by $\boldsymbol{\sigma} = [\sigma \quad 0 \quad 0]^T$ and dt is the update period. For the application of low-cost IMU in land vehicle navigation where the maximum angular rate is ~ 40 °/sec and the IMU update rate is 100 Hz, the influence on the position update caused by δ_{1st} is surely ignorable within 1 second (a typical data rate for the measurements based on the GNSS position and heading solution).

In addition, if the travelled distance along the trajectory is denoted by $\Delta\mathbf{r} = \int_k^{k+1} \mathbf{v}_{nb}^b \, dt$, after integrated the cross product term in 4.1.25, one can obtain that

$$\begin{aligned} \delta\mathbf{r}_{nb}^n &= \mathbf{C}_b^n(k)\left[\Delta\mathbf{r} + \boldsymbol{\sigma}\times\Delta\mathbf{r} - \int_k^{k+1}[\dot{\boldsymbol{\sigma}}\times\Delta\mathbf{r}]dt\right] \\ &= \mathbf{C}_b^n(k)\left[\Delta\mathbf{r} + (\boldsymbol{\alpha}+\boldsymbol{\varepsilon})\times\Delta\mathbf{r} - \int_k^{k+1}[(\boldsymbol{\omega}+\dot{\boldsymbol{\varepsilon}})\times\Delta\mathbf{r}]dt\right] \end{aligned} \tag{4.1.26}$$

In inertial navigation, the attitude drift $\delta\boldsymbol{\alpha}$ due to $\boldsymbol{\varepsilon}$ in (4.1.22) is usually evaluated through a typical coning motion when the body is oscillating at frequency *f* about the *x* and *y* axes with the angular amplitudes being θ_x and θ_y. For such a coning motion, the attitude drift $\delta\boldsymbol{\alpha}$ in z-component is $\pi f \theta_x \theta_y \delta t \sin\phi\left[1 - \frac{\sin 2\pi f \delta t}{2\pi f \delta t}\right]$ [Titterton, 2004], in which ϕ is the phase difference and δt is the time, e.g. 0.01 second when the position update frequency is 100 Hz. After drop $\boldsymbol{\varepsilon}$ and $\dot{\boldsymbol{\varepsilon}}$ in equation (4.1.26) due to their ignorable influences on the position, the delta position vector $\delta\mathbf{r}_{nb}^n$ turns out to be

$$\delta\mathbf{r}_{nb}^{n} = \mathbf{C}_{b}^{n}(k)\left[\Delta\mathbf{r} + \boldsymbol{\alpha}\times\Delta\mathbf{r} - \int_{k}^{k+1}\left[\boldsymbol{\omega}\times\Delta\mathbf{r}\right]dt\right] \tag{4.1.27}$$

Now, after ignoring higher order terms, $\mathbf{r}_{nb}^{n}(k+1)$ is approximated by:

$$\begin{aligned}\mathbf{r}_{nb}^{n}(k+1) &\approx \mathbf{r}_{nb}^{n}(k) + \mathbf{C}_{b}^{n}(k)\left[\left(\mathbf{I} + [\boldsymbol{\omega}_{\mathbf{k}}\times]\Delta t + [\dot{\boldsymbol{\omega}}_{\mathbf{k}}\times]\frac{\Delta t^{2}}{2}\right)\Delta\mathbf{r} - \frac{(\boldsymbol{\omega}_{\mathbf{k}}\times\mathbf{v}_{\mathbf{k}})}{2}\Delta t^{2} - \frac{(\boldsymbol{\omega}_{\mathbf{k}}\times\mathbf{a}_{\mathbf{k}})}{6}\Delta t^{3}\right] \\ &\approx \mathbf{r}_{nb}^{n}(k) + \mathbf{C}_{b}^{n}(k)\left[\Delta\mathbf{r} + \frac{(\boldsymbol{\omega}_{\mathbf{k}}\times\mathbf{v}_{\mathbf{k}})}{2}\Delta t^{2} + \frac{(\boldsymbol{\omega}_{\mathbf{k}}\times\mathbf{a}_{\mathbf{k}})}{3}\Delta t^{3}\right] \\ &\quad + \mathbf{C}_{b}^{n}(k)\left[(\mathbf{I} + [\boldsymbol{\omega}_{\mathbf{k}}\times]\Delta t)\right]\frac{\mathbf{j}_{nb}^{b}}{6}\Delta t^{3} - \mathbf{C}_{b}^{n}(k)[\Delta\mathbf{r}\times]\frac{\dot{\boldsymbol{\omega}}_{\mathbf{k}}}{2}\Delta t^{2}\end{aligned} \tag{4.1.28}$$

wherein are

$\boldsymbol{\omega}_{k+1} = \boldsymbol{\omega}_{\mathbf{k}} + \dot{\boldsymbol{\omega}}_{\mathbf{k}}\Delta t$,

$\boldsymbol{\alpha} = \int_{k}^{k+1}\boldsymbol{\omega}dt \approx \boldsymbol{\omega}_{\mathbf{k}}\Delta t + \frac{\dot{\boldsymbol{\omega}}_{\mathbf{k}}}{2}\Delta t^{2}$, and

$\Delta\mathbf{r} = \mathbf{v}_{k}\Delta t + \frac{\mathbf{a}_{k}}{2}\Delta t^{2} + \frac{\mathbf{j}_{nb}^{b}}{6}\Delta t^{3}$ where $\mathbf{j}_{nb}^{b}$ being the jerk vector in the body frame.

If GMIS is implemented through an EKF, the propagation of the covariance matrix of the position vector $\mathbf{r}_{nb}^{n}(k+1)$, as another necessary computation, relies on the linearization of (4.1.28). When computing the covariance matrix of $\mathbf{r}_{nb}^{n}(k+1)$, an extra intermediate matrix is needed since the attitude is currently modelled by the Euler angle vector $\boldsymbol{\theta} = [p \quad \gamma \quad \psi]^{T}$ instead of the attitude error $\boldsymbol{\varphi} = \left[\phi_{nbx}^{b} \quad \phi_{nby}^{b} \quad \phi_{nbz}^{b}\right]^{T}$ (in the **ENU** frame). Theoretically, the Euler angle vector $\boldsymbol{\theta}$ is related to the attitude error $\boldsymbol{\varphi}$ in the **ENU** frame as follows:

$$\boldsymbol{\delta\theta} = \begin{bmatrix} \delta p \\ \delta\gamma \\ \delta\psi \end{bmatrix} \approx \begin{bmatrix} \mathrm{c}\hat{\gamma} & 0 & \mathrm{s}\hat{\gamma} \\ \mathrm{s}\hat{\gamma}\,\mathrm{tg}\hat{p} & 1 & -\mathrm{c}\hat{\gamma}\,\mathrm{tg}\hat{p} \\ \mathrm{s}\hat{\gamma}\sec\hat{p} & 0 & -\mathrm{c}\hat{\gamma}\sec\hat{p} \end{bmatrix} \begin{bmatrix} \omega_{\mathrm{nbx}}^{\mathrm{b}}\Delta\mathrm{t} \\ \omega_{\mathrm{nby}}^{\mathrm{b}}\Delta\mathrm{t} \\ \omega_{\mathrm{nbz}}^{\mathrm{b}}\Delta\mathrm{t} \end{bmatrix}$$
$$= \begin{bmatrix} \mathrm{c}\hat{\gamma} & 0 & \mathrm{s}\hat{\gamma} \\ \mathrm{s}\hat{\gamma}\,\mathrm{tg}\hat{p} & 1 & -\mathrm{c}\gamma\,\mathrm{tg}\hat{p} \\ \mathrm{s}\hat{\gamma}\sec\hat{p} & 0 & -\mathrm{c}\hat{\gamma}\sec\hat{p} \end{bmatrix} \begin{bmatrix} \phi_{\mathrm{nbx}}^{\mathrm{b}} \\ \phi_{\mathrm{nby}}^{\mathrm{b}} \\ \phi_{\mathrm{nbz}}^{\mathrm{b}} \end{bmatrix}$$

Thus,

$$\boldsymbol{\varphi} = \begin{bmatrix} \phi_{\mathrm{nbx}}^{\mathrm{b}} \\ \phi_{\mathrm{nby}}^{\mathrm{b}} \\ \phi_{\mathrm{nbz}}^{\mathrm{b}} \end{bmatrix} = \begin{bmatrix} \mathrm{c}\hat{\gamma} & 0 & \mathrm{c}\hat{p}\,\mathrm{s}\hat{\gamma} \\ 0 & 1 & -\mathrm{s}\hat{p} \\ \mathrm{s}\hat{\gamma} & 0 & -\mathrm{c}\hat{p}\,\mathrm{c}\hat{\gamma} \end{bmatrix} \boldsymbol{\delta\theta} = \mathbf{T}\,\boldsymbol{\delta\theta} \tag{4.1.29}$$

For instance, given $\hat{\mathbf{v}}^{\mathrm{n}} = \mathbf{C}_{\mathrm{b}}^{\mathrm{n}}\hat{\mathbf{v}}^{\mathrm{b}}$, the first order derivative of $\hat{\mathbf{v}}^{\mathrm{n}}$ with respect to time t is then computed as follows:

$$\begin{aligned} \frac{\mathrm{d}\hat{\mathbf{v}}^{\mathrm{n}}}{\mathrm{dt}} &= \hat{\mathbf{C}}_{\mathrm{b}}^{\mathrm{n}}\frac{\mathrm{d}\mathbf{v}^{\mathrm{b}}}{\mathrm{dt}} - [\boldsymbol{\varphi}\times]\hat{\mathbf{C}}_{\mathrm{b}}^{\mathrm{n}}\hat{\mathbf{v}}^{\mathrm{b}} = \hat{\mathbf{C}}_{\mathrm{b}}^{\mathrm{n}}\frac{\mathrm{d}\mathbf{v}^{\mathrm{b}}}{\mathrm{dt}} + \hat{\mathbf{v}}^{\mathrm{n}}\times\boldsymbol{\varphi} \\ &= \hat{\mathbf{C}}_{\mathrm{b}}^{\mathrm{n}}\frac{\mathrm{d}\mathbf{v}^{\mathrm{b}}}{\mathrm{dt}} + [\hat{\mathbf{v}}^{\mathrm{n}}\times]\mathbf{T}\frac{\mathrm{d}\boldsymbol{\theta}}{\mathrm{dt}} \end{aligned} \tag{4.1.30}$$

Consequently, the covariance matrix of $\hat{\mathbf{v}}^{\mathrm{n}}$ is then expanded as follows:

$$\mathrm{cov}(\hat{\mathbf{v}}^{\mathrm{n}}) = \hat{\mathbf{C}}_{\mathrm{b}}^{\mathrm{n}}\,\mathrm{cov}(\hat{\mathbf{v}}^{\mathrm{b}})(\hat{\mathbf{C}}_{\mathrm{b}}^{\mathrm{n}})^{\mathrm{T}} + [\hat{\mathbf{v}}^{\mathrm{n}}\times]\mathbf{T}\,\mathrm{cov}(\boldsymbol{\theta})\mathbf{T}^{\mathrm{T}}[\hat{\mathbf{v}}^{\mathrm{n}}\times]^{\mathrm{T}} \tag{4.1.31}$$

ii. Velocity equation

The approximation for the velocity vector in body frame at epoch *k*+1 is simply performed using the Taylor expansion as follows:

$$\mathbf{v}_{\mathrm{nb(k+1)}}^{\mathrm{b}} = \mathbf{v}_{\mathrm{nb(k)}}^{\mathrm{b}} + \dot{\mathbf{v}}_{\mathrm{nb(k)}}^{\mathrm{b}}\Delta\mathrm{t} + \frac{\ddot{\mathbf{v}}_{\mathrm{nb(k)}}^{\mathrm{b}}}{2}\Delta\mathrm{t}^{2} \tag{4.1.32}$$

Given that $\dot{\mathbf{v}}_{\mathrm{nb}}^{\mathrm{b}} = \mathbf{a}_{\mathrm{nb}}^{\mathrm{b}} - \boldsymbol{\omega}_{\mathrm{nb}}^{\mathrm{b}}\times\mathbf{v}_{\mathrm{nb}}^{\mathrm{b}}$, the second order derivative of the body frame velocity vector is achieved by

$$\begin{aligned}\ddot{\mathbf{v}}^{b}_{nb} &= \dot{\mathbf{a}}^{b}_{nb} - \dot{\boldsymbol{\omega}}^{b}_{nb} \times \mathbf{v}^{b}_{nb} - \boldsymbol{\omega}^{b}_{nb} \times \dot{\mathbf{v}}^{b}_{nb} \\ &= \mathbf{j}^{b}_{nb} - \boldsymbol{\omega}^{b}_{nb} \times \mathbf{a}^{b}_{nb} - \dot{\boldsymbol{\omega}}^{b}_{nb} \times \mathbf{v}^{b}_{nb} - \boldsymbol{\omega}^{b}_{nb} \times \left(\mathbf{a}^{b}_{nb} - \boldsymbol{\omega}^{b}_{nb} \times \mathbf{v}^{b}_{nb}\right) \\ &= -2\left(\boldsymbol{\omega}^{b}_{nb} \times \mathbf{a}^{b}_{nb}\right) + \left(\boldsymbol{\omega}^{b}_{nb} \times \left(\boldsymbol{\omega}^{b}_{nb} \times \mathbf{v}^{b}_{nb}\right)\right) + \mathbf{v}^{b}_{nb} \times \dot{\boldsymbol{\omega}}^{b}_{nb} + \mathbf{j}^{b}_{nb}\end{aligned} \tag{4.1.33}$$

Then, the approximation of the velocity vector at epoch k+1 around the components of the state vector at epoch k is given by:

$$\begin{aligned}\mathbf{v}^{b}_{nb(k+1)} = &\left(\mathbf{I}_{3x3} - \left[\boldsymbol{\omega}^{b}_{nb(k)} \times\right]\Delta t + \frac{\left[\boldsymbol{\omega}^{b}_{nb(k)} \times\right]^2}{2}\Delta t^2\right)\mathbf{v}^{b}_{nb(k)} + \left(\mathbf{I}_{3x3}\Delta t - \left[\boldsymbol{\omega}^{b}_{nb(k)} \times\right]\Delta t^2\right)\mathbf{a}^{b}_{nb(k)} \\ &+ \left[\mathbf{v}^{b}_{nb(k)} \times\right]\frac{\dot{\boldsymbol{\omega}}^{b}_{nb}}{2}\Delta t^2 + \frac{\mathbf{j}^{b}_{nb}}{2}\Delta t^2\end{aligned} \tag{4.1.34}$$

iii. Acceleration equation

Similarly, the acceleration vector is approximated by

$$\mathbf{a}^{b}_{nb(k+1)} = \mathbf{a}^{b}_{nb(k)} + \dot{\mathbf{a}}^{b}_{nb(k)}\Delta t + \frac{\ddot{\mathbf{a}}^{b}_{nb(k)}}{2}\Delta t^2 \tag{4.1.35}$$

wherein

$$\dot{\mathbf{a}}^{b}_{nb} = \mathbf{j}^{b}_{nb} - \boldsymbol{\omega}^{b}_{nb} \times \mathbf{a}^{b}_{nb} \quad ,$$

$$\ddot{\mathbf{a}}^{b}_{nb} \approx \boldsymbol{\omega}^{b}_{nb} \times \boldsymbol{\omega}^{b}_{nb} \times \mathbf{a}^{b}_{nb} + \mathbf{a}^{b}_{nb} \times \dot{\boldsymbol{\omega}}^{b}_{nb} - 2\boldsymbol{\omega}^{b}_{nb} \times \mathbf{j}^{b}_{nb}$$

Then, the acceleration vector in the discrete form at epoch k+1 is

$$\begin{aligned}\mathbf{a}^{b}_{nb(k+1)} = &\left(\mathbf{I}_{3\times3} - [\boldsymbol{\omega}^{b}_{nb(k)}\times]\Delta t + \frac{[\boldsymbol{\omega}^{b}_{nb(k)}\times]^2}{2}\Delta t^2\right)\mathbf{a}^{b}_{nb(k)} \\ &+ \left[\mathbf{a}^{b}_{nb(k)} \times\right]\frac{\dot{\boldsymbol{\omega}}^{b}_{nb}}{2}\Delta t^2 + \left(\mathbf{I}_{3x3}\Delta t - \left[\boldsymbol{\omega}^{b}_{nb(k)} \times\right]\Delta t^2\right)\mathbf{j}^{b}_{nb}\end{aligned} \tag{4.1.36}$$

iv. Attitude equation

According to (2.2.6), the approximation of the Euler angles can be achieved through the first order Taylor series expansion as follows:

$$\begin{bmatrix} p \\ \gamma \\ \psi \end{bmatrix}_{k+1} \approx \begin{bmatrix} p_{(k)} \\ \gamma_{(k)} \\ \psi_{(k)} \end{bmatrix} + \begin{bmatrix} c\gamma_{(k)} & 0 & s\gamma_{(k)} \\ s\gamma_{(k)}\, tg p_{(k)} & 1 & -c\gamma_{(k)}\, tg p_{(k)} \\ s\gamma_{(k)} \sec p_{(k)} & 0 & -c\gamma_{(k)} \sec p_{(k)} \end{bmatrix} \begin{bmatrix} \omega^{b}_{nbx(k)} \\ \omega^{b}_{nby(k)} \\ \omega^{b}_{nbz(k)} \end{bmatrix} \Delta t \qquad (4.1.37)$$

v. Angular rate equation

According to the discussion in Section 4.1.1, under the assumption of the constant angular rate, the angular rate vector at epoch k+1 is:

$$\begin{bmatrix} \omega^{b}_{nbx} \\ \omega^{b}_{nby} \\ \omega^{b}_{nbz} \end{bmatrix}_{\mathbf{k+1}} = \begin{bmatrix} \omega^{b}_{nbx} \\ \omega^{b}_{nby} \\ \omega^{b}_{nbz} \end{bmatrix}_{\mathbf{k}} \qquad (4.1.38)$$

vi. IMU error equation

In GMIKF, the error vectors for gyroscope ($\mathbf{b}_{\omega}$) and accelerometer ($\mathbf{b}_{a}$) are also taken as random constants under the assumption of the high measurement noise level in a low cost inertial sensor [Kubrak, 2007; Edwan, 2012]. Nevertheless, the random constant model for inertial sensor's systematic errors can be extended to a first order zero-mean Gauss-Markov process presented in (2.4.9) if necessary.

4.1.3 The measurement model

4.1.3.1 IMU measurement model

Without loss of the generality, three orthogonal gyroscopes in an IMU measure the angular rate vector $\boldsymbol{\omega}^{i}_{ib}$ while three accelerometers measure the specific force vector $\boldsymbol{f}^{i}_{ib}$. An important assumption about inertial sensors is that their measurements are mainly contaminated by the following error sources: constant start-off biases, drifting bias residuals, linear scale factors, misalignments of sensor axes and Gaussian white noises. Targeting on the low-cost MEMS

Crossbow 440CA IMU unit used in YUMIS, the associated measurement models for three gyroscopes and three accelerometers are rephrased as follows based on (2.4.11) and (2.4.12):

$$\boldsymbol{\omega}_{\text{ib-imu}}^{\text{b}} = (\mathbf{I} + \mathbf{S}_{\omega})\left(\boldsymbol{\omega}_{\text{nb}}^{\text{b}} + \text{C}_{\text{n}}^{\text{b}}(\boldsymbol{\omega}_{\text{ie}}^{\text{n}} + \boldsymbol{\omega}_{\text{en}}^{\text{n}})\right) + \mathbf{b}_{\omega} + \nabla_{\omega} \tag{4.1.39}$$

$$\boldsymbol{f}_{\text{ib-imu}}^{\text{b}} = (\mathbf{I} + \mathbf{S}_{\text{a}})\left(\mathbf{a}_{\text{nb}}^{\text{b}} + \text{C}_{\text{n}}^{\text{b}}\left((2\boldsymbol{\omega}_{\text{ie}}^{\text{n}} + \boldsymbol{\omega}_{\text{en}}^{\text{n}}) \times \mathbf{v}_{\text{nb}}^{\text{n}}\right) - \text{C}_{\text{n}}^{\text{b}}\mathbf{g}^{\text{n}}\right) + \mathbf{b}_{\text{a}} + \nabla_{\text{a}} \tag{4.1.40}$$

wherein I is the identity matrix.

4.1.3.2 GNSS measurement model

i. Loosely coupled (cascade) integration

In a loosely-coupled (cascade) integration system, the aiding measurements are usually the direct outputs from other standalone navigation systems, e.g., the position, velocity, the GNSS heading and etc. The corresponding GNSS measurement equations are:

$$\mathbf{r}_{\mathbf{k}}^{\mathbf{GNSS}} = \mathbf{r}_{\mathbf{k}} + \mathbf{C}_{\mathbf{b(k)}}^{\mathbf{n}}\mathbf{L}^{\mathbf{b}} + \mathbf{w}_{\mathbf{P\text{-}GNSS}} \tag{4.1.41}$$

$$\mathbf{v}_{\mathbf{k}}^{\mathbf{GNSS}} = \mathbf{C}_{\mathbf{b(k)}}^{\mathbf{n}}\mathbf{v}_{\mathbf{nb(k)}}^{\mathbf{b}} + \mathbf{C}_{\mathbf{b(k)}}^{\mathbf{n}}[\boldsymbol{\omega}_{\mathbf{nb(k)}}^{\mathbf{b}}\times]\mathbf{L}^{\mathbf{b}} + \mathbf{w}_{\mathbf{v\text{-}GNSS}} \tag{4.1.42}$$

$$\boldsymbol{\psi}_{\mathbf{k}}^{\mathbf{GNSS}} = \boldsymbol{\psi}_{\mathbf{k}} + \mathbf{w}_{\mathbf{H\text{-}GNSS}} \tag{4.1.43}$$

wherein are

$\mathbf{L}^{\mathbf{b}}$	the lever arm vector in body frame,
$\mathbf{C}_{\mathbf{b}}^{\mathbf{n}}$, $\boldsymbol{\omega}_{\mathbf{nb}}^{\mathbf{n}}$	the DCM matrix and angular rate states,
$\mathbf{r}$, $\mathbf{v}_{\mathbf{nb}}^{\mathbf{b}}$, $\boldsymbol{\psi}$	the position, velocity and heading states,
$\mathbf{r}^{\mathbf{GNSS}}$, $\mathbf{v}^{\mathbf{GNSS}}$, $\boldsymbol{\psi}^{\mathbf{GNSS}}$	the GNSS position, GNSS velocity and heading measurements, and
$\mathbf{w}_{\mathbf{x\text{-}GNSS}}$	the associated noise vectors of the GNSS measurements (x=p, v, h).

ii. Tightly coupled (centralized) integration

Because the states in GMIKF are the navigation parameters, the measurement equations in tightly-coupled system take the same format as in the GNSS standalone navigation. For example, with the GPS system, the generic observation equations for the pseudorange ρ_i^j and the carrier phases φ_i^j (only L1 phases is considered for demonstration purpose) from receiver *i* to satellite *j* are:

$$\rho_i^j = \left\|\mathbf{r}_i^j\right\| + c(\delta t_i - \delta t^j) + d_{i-trop}^j + d_{i-ion}^j + \varepsilon_{\rho_i^j} \tag{4.1.44}$$

$$\phi_i^j = \left\|\mathbf{r}_i^j\right\| + c(\delta t_i - \delta t^j) + d_{i-trop}^j - d_{i-ion}^j + \lambda_1 N_i^j + \varepsilon_{\varphi_i^j} \tag{4.1.45}$$

wherein *i*=*A*, *B* (one as a base station and the other one as a rover) , *j*=1, 2, …, n, $\left\|\mathbf{r}_i^j\right\|$ is the geometric range from receiver *i* to satellite *j*, *c* is the speed of light, $\delta t_i, \delta t^j$ are the receiver and satellite clock errors, respectively, $d_{i-trop}^j, d_{i-ion}^j$ are the tropospheric and ionospheric delays, λ_1, N_i^j are the L1 band's wave length and the ambiguity parameter, and $\varepsilon_{\rho_i^j}$, $\varepsilon_{\varphi_i^j}$ are the random noises with zero means. As usual, the single differencing performs between two GNSS receivers with respect to the same satellite and further the double differencing is introduced to two single differenced measurements between two satellites. Specifically, with the double-differenced pseudo-ranges at L1 band, the generic observation equation becomes:

$$\nabla\Delta\rho_{AB}^{jk}(t) = \left\|\nabla\Delta\mathbf{r}_{AB}^{jk}(t)\right\| + \nabla\Delta d_{AB-Iono}^{jk}(t) + \nabla\Delta d_{AB-Trop}^{jk}(t) + \varepsilon_{\nabla\Delta\rho_{AB}}^{jk}(t) \tag{4.1.46}$$

where $\nabla\Delta$ is the double differencing operator between stations ***A*** and ***B*** with respect to satellites *j* and *k*, $\left\|\nabla\Delta\mathbf{r}_{AB}^{jk}(t)\right\|$ is the double differenced range, $\nabla\Delta d_{AB-Iono}^{jk}(t)$ is the double differenced

ionospheric error, $\nabla\Delta d^{jk}_{AB-Trop}(t)$ is the double differenced tropospheric error and $\varepsilon^{jk}_{\nabla\Delta\rho_{AB}}(t)$ is the random error.

With the short baseline (between rover and the GPS base station), e.g., < 10km, the double differenced tropospheric and ionospheric delays become small enough to be ignored so that (4.1.46) can be simplified to

$$\nabla\Delta\rho^{jk}_{AB}(t) = \left\|\nabla\Delta\mathbf{r}^{jk}_{AB}(t)\right\| + \varepsilon^{jk}_{\nabla\Delta\rho_{AB}}(t) \tag{4.1.47}$$

with $\nabla\Delta\rho^{jk}_{AB} = [\rho^{k}_{B}(t) - \rho^{k}_{A}(t)] - [\rho^{j}_{B}(t) - \rho^{j}_{A}(t)]$. Correspondingly, with the carrier phase observables at L1 band, the double differenced observation equation is modelled to be

$$\nabla\Delta\varphi^{jk}_{AB}(t) = \left\|\nabla\Delta\mathbf{r}^{jk}_{AB}(t)\right\| + \lambda_1\nabla\Delta N^{jk}_{AB} + \varepsilon^{jk}_{\nabla\Delta\varphi_{AB}}(t) \tag{4.1.48}$$

Furthermore, the measurement equation for the double-differenced Doppler observable shall be

$$\nabla\Delta\dot{\varphi}^{jk}_{AB}(t) = \left\|\nabla\Delta\dot{\mathbf{r}}^{jk}_{AB}(t)\right\| + \varepsilon^{jk}_{\nabla\Delta\dot{\varphi}_{AB}}(t) \tag{4.1.49}$$

In particular, the state vector in GMIKF must be extended to include the float ambiguity parameters ($\nabla\Delta N^{jk}_{AB}$) before they are fixed as constant integer numbers.

4.2 Analysis and characteristics of GMIKF

4.2.1 Characteristics and advantages

The Figure 4-2 shows three paramount distinctions in GMIKF (right side) in comparison with the conventional counterpart (left side):

a) The embedment of the prediction of the kinematic states, i.e., position, the velocity, acceleration, attitude and angular rate vector in the navigation Kalman filter as the system model. Thus, the core of the system model after kinematics can effectively restrain the

potential drifts due to the inertial sensor systematic errors, especially with the low-cost IMUs.

b) The direct participation of the IMU measurements in Kalman filter's measurement updates. Consequently, the superior performance (solution accuracy) arises from the improvement of the overall system measurement redundancy upon the angular rates and accelerations. In addition, it opens up the possibility to realize the variance component estimation (VCE) for the individual inertial sensors.

c) No need to distinguish between the core sensor and aiding sensors so that all the individual measurements can be processed directly through their measurement updates in Kalman filter. As a result, this structure makes possible of the simultaneous estimation of variance components in the process noise vectors and measurement noise vectors for all sensors participating in GMIKF because the random errors in IMU measurements are statistically separated from the other error sources.

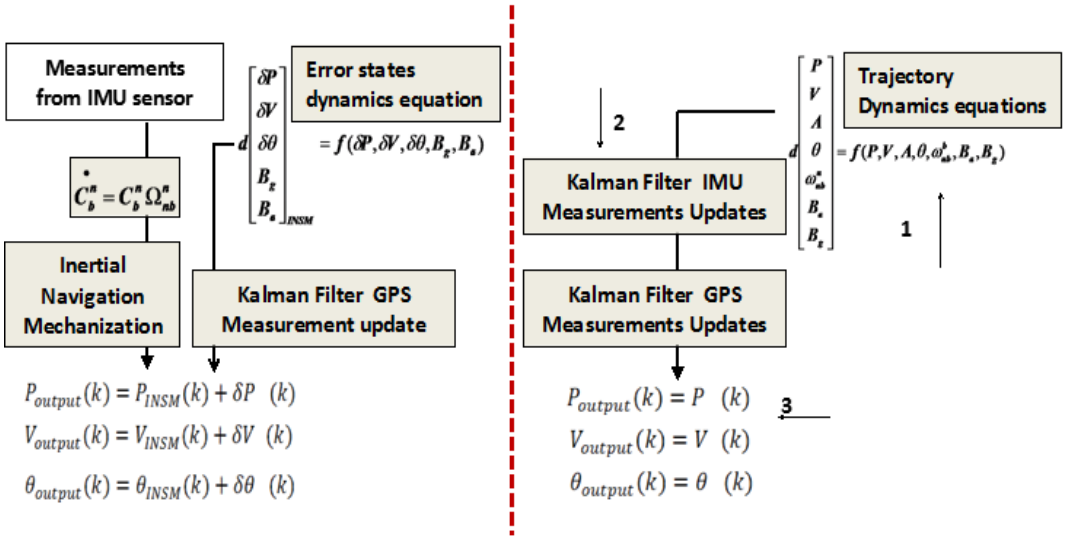


Figure 4-2 Comparison between the traditional GNSS-aided IMU integration mechanism and the proposed novel GMIKF [Qian et al, 2013]

In principle, under the assumption that all noises are normal distributed, Kalman filter is equivalent to a sequential least square with a time-variant state vector and the process noises [Anderson et al, 1979; Dan Simon, 2006]. In other words, the system model is actually a group of virtual measurements for the state vector which reveals the connections between the state vectors from time to time. It is because of these extra virtual measurements that the measurement redundancies in the proposed GMIKF for $\boldsymbol{\omega}_{\mathrm{nb}}^{\mathrm{b}}$ and $\mathbf{a}_{\mathrm{nb}}^{\mathrm{b}}$ are obviously improved than that in the traditional error state-based Kalman filter, in which the acceleration vectors, angular rate vectors and the associated IMU systematic errors between two adjacent aiding epochs are only treated as the input (driving force) of the mechanization. As a result, the accuracy of the angular rate vector $\boldsymbol{\omega}_{\mathrm{nb}}^{\mathrm{b}}$ and the acceleration vector $\mathbf{a}_{\mathrm{nb}}^{\mathrm{b}}$ in GMIKF will be surely improved for the low accuracy IMUs. It can also be seen that the dynamics constraints introduced by Dissanayake [2001] is equivalent to (4.1.6) with the zero noise vector. Furthermore, by assuming that $v_{nbz}^{b} = 0$ and/or $v_{nbx}^{b} = 0$, the predicted velocity components shall be reduced into the same quasi-zero virtual measurements as in [Shin and El-Sheimy, 2001; Ma et al, 2003; Godha, 2006].

As mentioned before, the traditional error-state based algorithm propagates the position solution $\mathbf{r}_{\mathrm{nb}}^{\mathrm{n}}$ based on the acceleration vector $\mathbf{a}_{\mathrm{ib}}^{\mathrm{b}}$ and the angular rate vector $\boldsymbol{\omega}_{\mathrm{ib}}^{\mathrm{b}}$ resulted from removing biases from the raw IMU measurements. Therefore, the performance of the navigation mechanization (free inertial solution) in the traditional algorithm is directly driven by the accuracy of the IMU outputs. Accordingly, the larger the random errors in an inertial sensor are, the larger the errors in the free inertial navigation solution are. In an extreme case, the significantly large random errors in low-cost IMU measurements could even substantially influence the nominal trajectory model, and lead to the distortion of the assumed INS error model which can be greatly mitigated in the GMIKF.

Furthermore, the *a priori* error model defined for a static low-cost MEMS inertial sensor needs to be checked and/or modified in the dynamic working environment because the vibration on a low cost MEMS IMU might cause significant changes of its noise level compared to those in the static case [De Pasquale, 2010]. In the GMIKF, the system model (prediction) for $\boldsymbol{\omega}_{nb}^{b}$ and $\mathbf{a}_{nb}^{b}$ does provide a rigorous reference for checking on the IMU performance without complicating the filtering structure.

In summary, the significant advantages of the generic multisensor integration strategy lie in 1) the performance improvement due to the rigorous trajectory model of a rigid body; 2) the improvement of the overall system redundancy upon the angular rate and acceleration of a rigid body; 3) the influence of the IMU measurement noises on the final navigation solutions is effectively mitigated as the result of the participation of the direct IMU measurement update in Kalman Filter; and 4) The potential introduction of the error analysis of individual sensors (Chapter 5) due to the structure of the novel GMIKF.

4.2.2 Performance boundary analysis

It is well known that the assessment of the performance of the discrete-time nonlinear filter is difficult due to the lack of optimal estimator and the true states. In practice, the alternative suboptimal filters (e.g., EKF, particle filter) are developed to deliver the sub-optimal state estimates which approximate the true states. Theoretical performance evaluation of suboptimal filters is usually accomplished through deriving a lower bound on the error covariance as an indicator of the achievable performance. A commonly used lower bound on the error covariance of a suboptimal filter is the Cramer-Rao lower bound (CRLB) established for the error

performance evaluation of suboptimal nonlinear estimators and the assessment of nonlinear approximation [Lei et al, 2011].

In this section, two CRLBs for both the traditional error-state based INS EKF (ErrINSEKF) and GMIKF upon the same simulated deterministic trajectory are developed. Subsequently, the superior theoretical performance of GMIKF to the ErrINSEKF are revealed through the comparison of their nominal CRLB bounds. The CRLBs for various types (grades) of IMUs are also calculated in order to show the gradually diminished performance discrepancy between the GMIKF and the ErrINSEKF when the random errors in IMU measurements become smaller.

4.2.2.1 Cramer Rao bound for EKF

Let $\hat{\mathbf{x}}$ be an unbiased estimate of the unknown vector $\mathbf{x}$ based on the measurement vector $\mathbf{z}$, and let $\mathbf{p_{x,z}(x,z)}$ be the joining probability density of the pair $\mathbf{(x,z)}$, the Cramer Rao lower bound (CRLB) for the covariance of $\hat{\mathbf{x}}$ is expressed by $\mathbf{P}^*$, therefore,

$$\mathbf{cov}(\hat{\mathbf{x}}) = \mathrm{E}\left[(\mathbf{x}-\hat{\mathbf{x}})((\mathbf{x}-\hat{\mathbf{x}})^{\mathrm{T}})\right] \geq \mathbf{P}^* = \mathbf{J}^{-1} \tag{4.2.1}$$

where the element of the Fisher matrix ($\mathbf{J}$) is

$$[\mathbf{J}]_{\mathbf{ij}} = -\mathrm{E}\left\{\frac{\partial \ln \mathrm{p}_{\mathbf{x,z}}(\mathbf{x,z})}{\partial \mathrm{x}_{\mathrm{i}}}\frac{\partial \ln \mathrm{p}_{\mathbf{x,z}}(\mathbf{x,z})}{\partial \mathrm{x}_{\mathrm{j}}}\right\} = -\mathrm{E}\left\{\frac{\partial^2 \ln \mathrm{p}_{\mathbf{x,z}}(\mathbf{x,z})}{\partial \mathrm{x}_{\mathrm{i}}\partial \mathrm{x}_{\mathrm{j}}}\right\} \tag{4.2.2}$$

The CRLB establishes the lower limits on how much “information” a set of measurements carries about an unknown parameter vector [Derpanis, 2006]. In many instances, obtaining $\mathbf{P}^*$ for a nonlinear filter is difficult [Taylor, 1978]. However, in the case of the extended Kalman filter, Taylor [1978] pointed out that its covariance propagation equations linearized about the true unknown trajectory provide the Cramer-Rao lower bound to the estimation error covariance matrix in EKF. For the multisensor integrated inertial navigation system, the approximated CRLB

of the estimated non-linear states using EKF is exactly analogous to the EKF covariance matrix when one replaces the state estimates with its real values [Benzerrouk, 2012]. In other words, the performance of navigation and positioning algorithm can be evaluated using a deterministic (simulated) trajectory [Bessell, 2003] upon which the CRLB bound $\left(\mathbf{P}_{\mathbf{k}}^{*}\right)^{-1}$ for the state vector $\hat{\mathbf{x}}_{\mathbf{k}}$ at epoch k can be written in a recursive form as follows:

$$\left(\mathbf{P}_{\mathbf{k}}^{*}\right)^{-1} = \left(\mathbf{F}_{\mathbf{k-1}}\mathbf{P}_{\mathbf{k-1}}^{*}\mathbf{F}_{\mathbf{k-1}}^{\mathbf{T}} + \mathbf{Q}_{\mathbf{k}}\right)^{-1} + \mathbf{H}_{\mathbf{k}}^{\mathbf{T}}\mathbf{R}_{\mathbf{k}}^{-1}\mathbf{H}_{\mathbf{k}} \tag{4.2.3}$$

wherein $\mathbf{P}_{\mathbf{k}}$, $\mathbf{F}_{\mathbf{k}}$, $\mathbf{Q}_{\mathbf{k}}$, $\mathbf{H}_{\mathbf{k}}$ and $\mathbf{R}_{\mathbf{k}}$ can be referred to the matrices defined in (2.6.4)- (2.6.10) in Section 2.6.3. As a result, CRLBs analysis in the case of EKF is the same as the covariance analysis predicated on the stochastic models being correct.

4.2.2.2 A simulated example

Figure 4-3 shows a simulated trajectory used in CRLB computation. It consists of several segments with various types of maneuvering motions: constant velocity, constant acceleration, circular turn and non-zero roll constant velocity. Because the land vehicle system is of our most research interest, the 3D trajectory with the constant height is used in the following CRLB comparison.

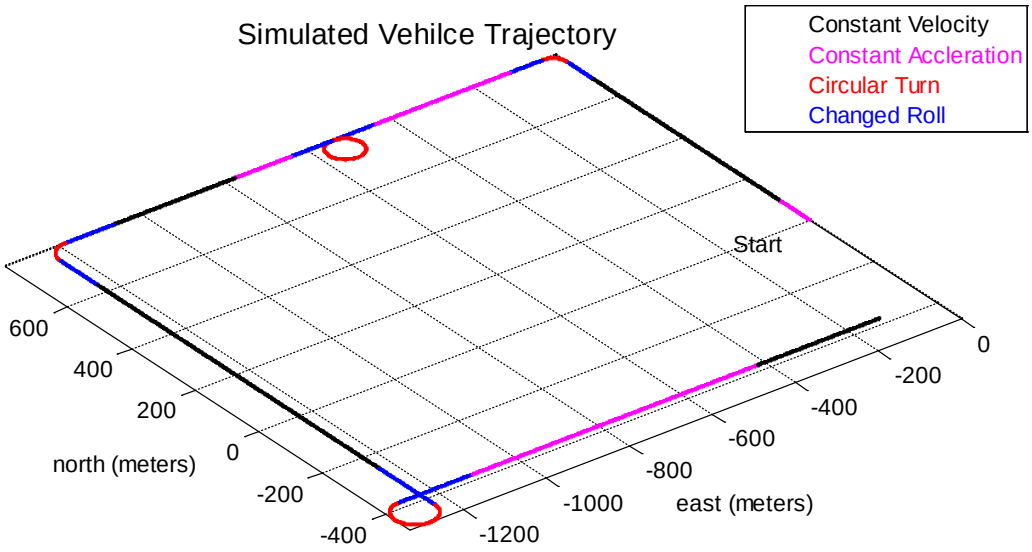


Figure 4-3 The simulated trajectory for GMIKF's CRLB evaluation

It is assumed that the due north oriented stationary vehicle is perfect levelled before it is accelerated at the start point $[0, 0, 0]^T$ at the time 0 (sec). Throughout the entire trajectory, four constantly-accelerated motions, four uniform linear motions with the constant-velocity and four uniform circular motions are simulated. The corresponding velocity and acceleration profiles in the local navigation ENU frame, and the angular rate profile in the body frame are given in Figure 4-4.

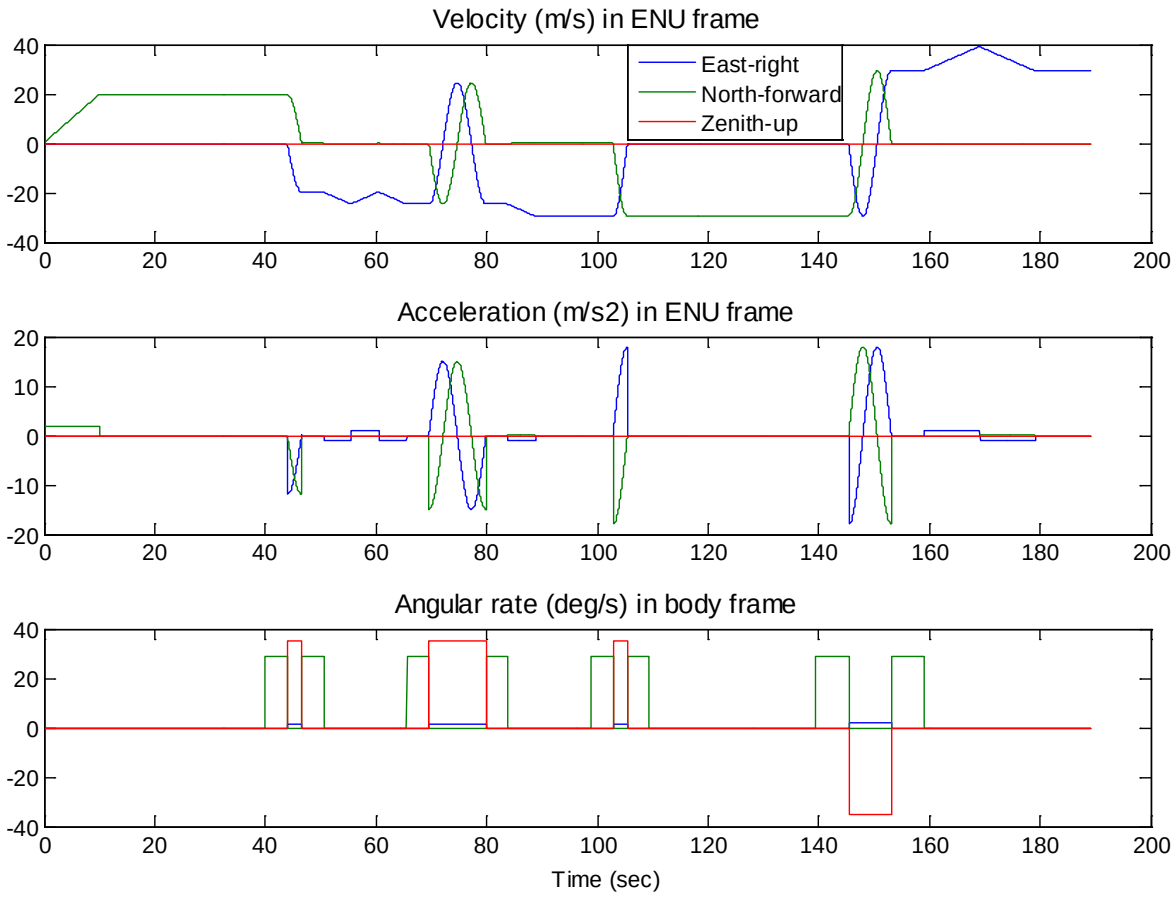


Figure 4-4 The velocity, acceleration and angular rate profiles

In this example, the comparison of the CRLB bounds between the 15-state ErrINSEKF, derived from (2.6.11) – (2.6.17) and the 21-state GMIKF, derived from (4.1.14)- (4.1.20) are proceeded upon the static motion and the accelerated motion, both of which span 180 seconds in time.

In CRLB computation, $\mathbf{y_k}$ in (4.2.5) represents the error vector in the instantaneous raw inertial observables (the angular rates or the specific forces) modeled as the sum of a random error vector $\mathbf{x}_\mathbf{k}^{\mathrm{wn}}$ and a random constant bias vector $\mathbf{x}_\mathbf{k}^{\mathrm{b}}$, while all other errors (e.g. installation misalignment, temperature and etc.) are ignored.

In addition, the error vector $\mathbf{x}_k^{wn}$ is taken as the process noise vector in the traditional ErrINSEKF while it is treated as measurement noise vector in the GMIKF. $\mathbf{x}_\mathbf{k}^{wn}$ is further modeled as the linear combination of two independent white noise vectors $\mathbf{x}_\mathbf{k}^{RW_{wn}}$ and $\mathbf{x}_\mathbf{k}^{BS_{wn}}$ which are translated from random walk noise and bias stability, respectively.

As a result, the inertial sensor error vector $\mathbf{y}_\mathbf{k}$ is mathematically modeled as follows:

$$\mathbf{x}_\mathbf{k}^\mathbf{b} = \mathbf{x}_\mathbf{k-1}^\mathbf{b} \tag{4.2.4}$$

$$\begin{aligned} \mathbf{y}_\mathbf{k} &= \mathbf{x}_k^b + \mathbf{x}_k^{wn} \\ &= \mathbf{x}_k^b + \mathbf{x}_k^{RW_{wn}} + \mathbf{x}_k^{BS_{wn}} \end{aligned} \tag{4.2.5}$$

with $\mathbf{x}_k^{RW_{wn}} \sim N(0, \boldsymbol{\sigma}_{RW_{wn}}^2)$, $\mathbf{x}_k^{BS_{wn}} \sim N(0, \boldsymbol{\sigma}_{BS_{wn}}^2)$, and $\mathbf{x}_k^{wn} \sim N(0, \boldsymbol{\sigma}_{RW_{wn}}^2 + \boldsymbol{\sigma}_{BS_{wn}}^2)$.

In order to conduct the comprehensive comparison of the CRLB bounds between ErrINSEKF and GMIKF, several IMU units from the full range of the IMU grade in the civil applications: navigation, tactic, industrial and consumer are considered. Table 4-1 lists the specifications of used IMUs in the CRLB computation.

Table 4-1 Specifications of the used IMUs

Grade	Name	Random Walk (RW)	Random Walk White noise (RWN)	Bias (BS)	Non-Linearity	Resolution
Navigation	HG9900	0.002 $^0/\sqrt{h}$	-	0.003 ^{0}/h	-	-
		-	0.005 mg/$\sqrt{\text{HZ}}$	0.025 mg	-	-
Tactical	HG1700	0.088 $^0/\sqrt{h}$	-	1.0 ^{0}/h	-	-
		0.2 m/s/$\sqrt{h}$	-	1.0 mg	-	-
Tactical	VG800	0.1 $^0/\sqrt{h}$	-	3.0 ^{0}/h	0.1% FS	< 0.025 °/s
		0.5 m/s/$\sqrt{h}$	-	1.0 mg	0.1% FS	< 0.5 mg
Industrial	SMDIMU	2.0 $^0/\sqrt{h}$	-	20 ^{0}/h	0.1% FS	
		0.08 m/s/$\sqrt{h}$	-	0.075 mg	0.4% FS	
Industrial	VN-100	-	0.035 ^{0}/s$\sqrt{\text{HZ}}$(256 Hz)	10 ^{0}/h	0.1% FS	< 0.02 °/s
		-	0.14 mg/$\sqrt{\text{HZ}}$(260 Hz)	0.04 mg	0.5% FS	< 0.5 mg
Industrial	IMU440	4.5 $^0/\sqrt{h}$	-	10 ^{0}/h	0.5% FS	< 0.02 °/s
		1.0 m/s/$\sqrt{h}$	-	1.0 mg	1% FS	< 0.5 mg
Consumer	BP3010	12.02 $^0/\sqrt{h}$	-	96.9 ^{0}/h	-	-
		0.47 m/s/$\sqrt{h}$	-	0.450 mg	-	-

Consumer	iPhone 4	144 $^0/\sqrt{h}$	-	75.0 ^{0}/s	-	-
		8.0 m/s/$\sqrt{h}$	-	20.0 mg	-	-

In addition, three necessary matrices for Kalman filter's CRLB computation: the covariance matrix **Q** of the process noise vector, the covariance matrix **R** of the measurement noise vector and the covariance matrix $\mathbf{P_0}$ of initial state vector are initialized as the diagonal matrices with their standard deviations (1σ) shown in Table 4-2, 4-3 and 4-4.

Table 4-2 Standard deviations of the initial states for ErrINSEKF and GMIEKF

ErrINSEKF		1 σ	GMIKF		1 σ
Position (m)	$\delta\mathbf{r}^{\mathbf{n}}_{\mathrm{nb}}$	10, 10, 10	Position (m)	$\mathbf{r}^{\mathbf{n}}_{\mathrm{nb}}$	10, 10, 10
Velocity (m/s)	$\delta\mathbf{v}^{\mathbf{n}}_{\mathrm{nb}}$	0.001, 0.001, 0.001	Velocity (m/s)	$\mathbf{v}^{\mathbf{b}}_{\mathrm{nb}}$	0.001, 0.001, 0.001
-	-	-	Acceleration (m/s2)	$\mathbf{a}^{\mathbf{b}}_{\mathrm{nb}}$	10e-7, 10e-7, 10e-7
Misalignment (°)	ϕ	0.1, 0.1, 0.1	Euler angle (°)	$\boldsymbol{\theta}$	0.1, 0.1, 0.1
-	-	-	Angular rate (°/s)	$\boldsymbol{\omega}^{\mathbf{b}}_{\mathrm{nb}}$	10e-7, 10e-7, 10e-7
Gyroscope bias	$\delta\boldsymbol{\omega}^{\mathbf{b}}_{\mathbf{ib}}$	$\mathrm{BS}_{\omega}\sqrt{\frac{\delta t}{T}}$	Gyroscope bias	$\mathbf{b}_{\boldsymbol{\omega}}$	$\mathrm{BS}_{\omega}\sqrt{\frac{\delta t}{T}}$
Accelerometer bias	$\nabla^{\mathbf{b}}_{\mathbf{ib}}$	$\mathrm{BS}_{a}\sqrt{\frac{\delta t}{T}}$	Accelerometer bias	$\mathbf{b}_{\mathbf{a}}$	$\mathrm{BS}_{a}\sqrt{\frac{\delta t}{T}}$

Note: BS_{ω} and BS_{a} are the respective 1σ bias stabilities (Allan Variance) for gyroscopes and accelerometers listed in the "Bias" column in Table 4-1, δt is the IMU's data rate (e.g. 0.01 s for 100 Hz), and T is the correlation time when bias stability computed.

Table 4-3 Standard deviations of process noises used in ErrINSEKF and GMIEKF

ErrINSEKF	1 σ	GMIKF	1 σ
Gyroscope noise $\mathbf{x}_{\omega}^{\text{wn}}$ (°/s)	$\sqrt{\boldsymbol{\sigma}_{\text{RW}_{\omega}}^2 + \boldsymbol{\sigma}_{\text{BS}_{\omega}}^2}$	Jerk $\mathbf{j}_{\text{nb}}^{\mathbf{b}}$ (m/s^3)	72, 72, 72 (moving)
			0, 0, 0（Static）
Accelerometer noise $\mathbf{x}_{\text{a}}^{\text{wn}}$ (m/s^2)	$\sqrt{\boldsymbol{\sigma}_{\text{RW}_{\text{a}}}^2 + \boldsymbol{\sigma}_{\text{BS}_{\text{a}}}^2}$	Angular acceleration $\dot{\boldsymbol{\omega}}_{\text{nb}}^{\mathbf{b}}$ (°/s^2)	50, 50, 50 (moving)
			0, 0, 0（Static）

Note: $\boldsymbol{\sigma}_{\text{RW}_{\omega}} = \text{RW}_{\omega} / \sqrt{\delta\, t}$ and $\boldsymbol{\sigma}_{\text{RW}_{\text{a}}} = \text{RW}_{\text{a}} / \sqrt{\delta\, t}$ where $\delta\, t$ is IMU data rate (e.g. 0.01 s for 100 Hz data); RW_{ω} and RW_{a} are the standard deviations of the white noise listed in the "Random Walk" column in Table 4-1, by which the random walk effects for gyroscopes and accelerometers are triggered. In addition, $\boldsymbol{\sigma}_{\text{BS}_{\omega}} = \text{BS}_{\omega}\sqrt{\frac{\delta\, t}{\text{T}}}$ and $\boldsymbol{\sigma}_{\text{BS}_{\text{a}}} = \text{BS}_{\text{a}}\sqrt{\frac{\delta\, t}{\text{T}}}$ where BS_{ω} and BS_{a} are the same as in Table 4-2.

Table 4-4 Measurement standard deviations in loosely-coupled ErrINSEKF and GMIEKF

ErrINSEKF	1 σ	GMIKF	1 σ
-	-	Gyroscope noise $\mathbf{x}_{\omega}^{\text{wn}}$ (°/s)	$\sqrt{\boldsymbol{\sigma}_{\text{RW}_{\omega}}^2 + \boldsymbol{\sigma}_{\text{BS}_{\omega}}^2}$
-	-	Accelerometer noise $\mathbf{x}_{\text{a}}^{\text{wn}}$ (m/s^2)	$\sqrt{\boldsymbol{\sigma}_{\text{RW}_{\text{a}}}^2 + \boldsymbol{\sigma}_{\text{BS}_{\text{a}}}^2}$
External Position (m)	1.0, 1.0, 1.0	External Position (m)	1.0, 1.0, 1.0
External Heading (°)	0.1	External Heading (°)	0.1

Note: The definition for the four IMU measurement noises $\boldsymbol{\sigma}_{\text{RW}_{\omega}}$, $\boldsymbol{\sigma}_{\text{RW}_{\text{a}}}$, $\boldsymbol{\sigma}_{\text{BS}_{\omega}}$ and $\boldsymbol{\sigma}_{\text{BS}_{\text{a}}}$ are the same as in Table 4-3.

CRLBs in free inertial mode

In the conventional error-state-based inertial integrated navigation (ErrINS), the inertial only mode is conventionally termed as either the free inertial navigation calculation, or dead-reckoning. On the other hand, the free inertial mode of GMIEKF means that only IMU measurements are applied through measurement updates in the navigation Kalman filter. Figure 4-5 and Figure 4-6 show the time series of the square-rooted CRLB of the 3D position and attitudes with HG9900 (high end) and BP3010 (low cost) where ErrINS refers to the ErrINSEKF in free inertial mode and GMIINS refers to the GMIEKF in free inertial mode.

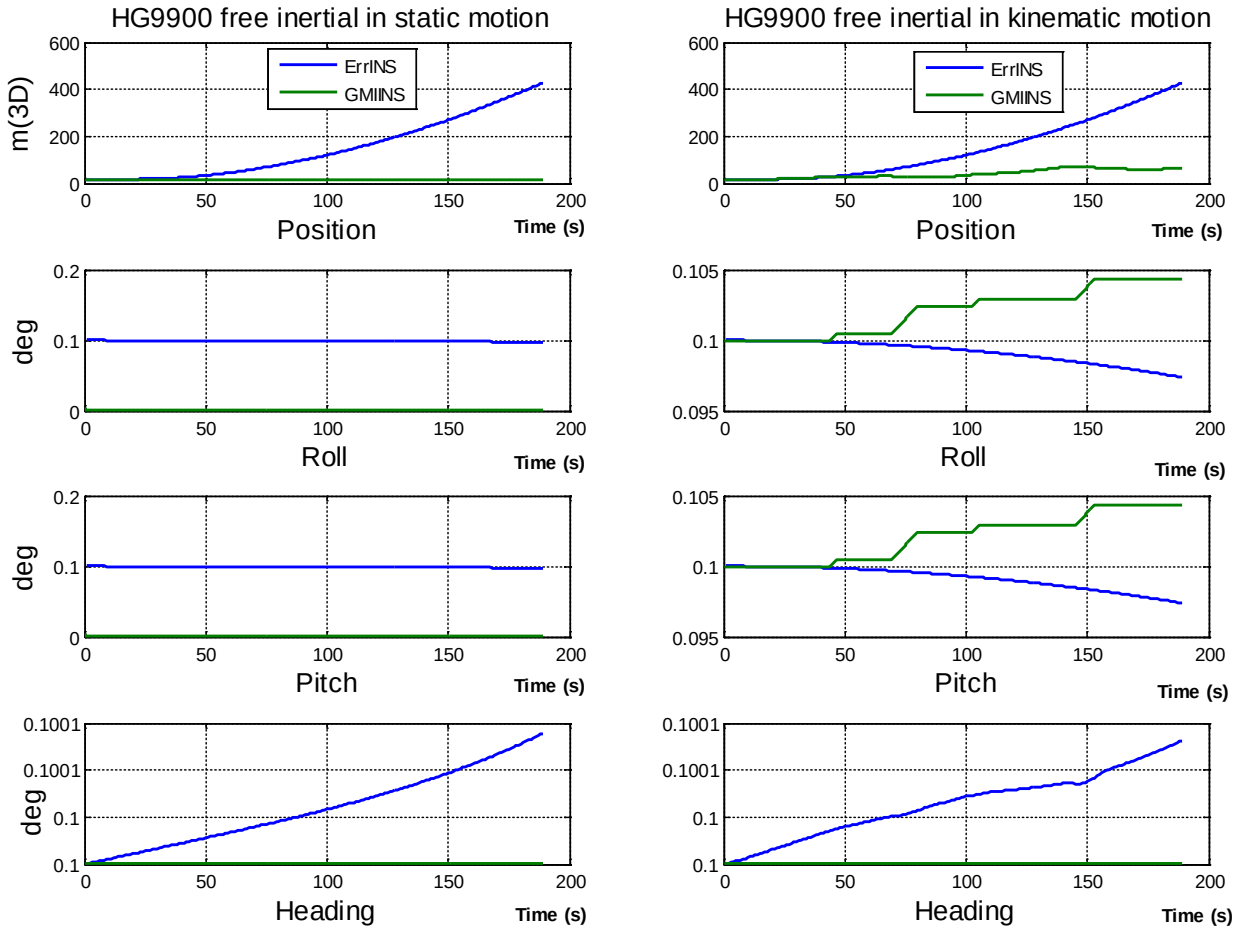


Figure 4-5 Sqrt(CRLB) of position and attitudes using HG9900

During the 180 second period of stationary motion (left side in both figures), the position performance (CRLB) of GMIEKF is well constrained through the prediction of the trajectory. Meanwhile, the position CRLBs of the ErrINSEKF show slight growth which clearly follows the IMU's specification data. In kinematic situation, the performance of the GMIEKF algorithm is also superior to the traditional Error state based algorithm for both IMUs.

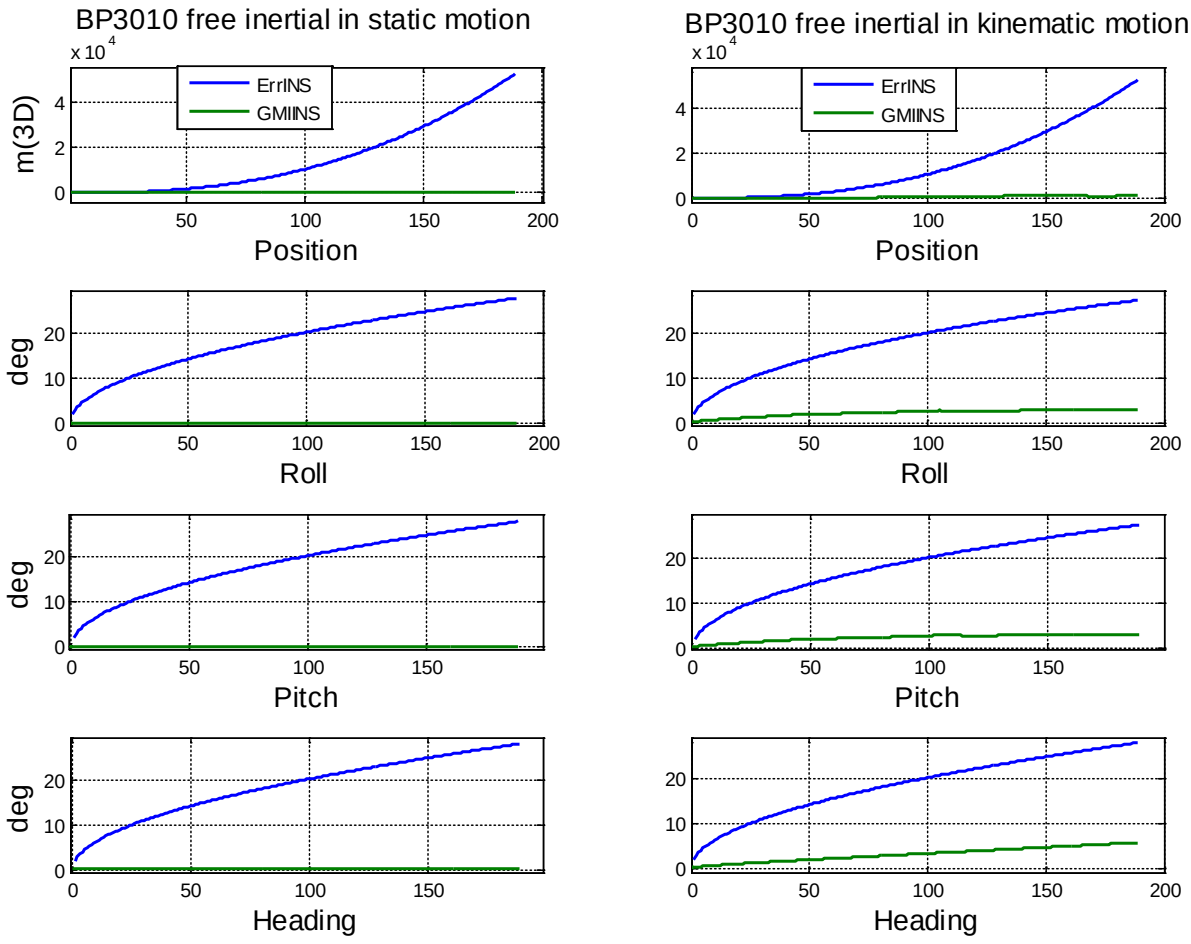


Figure 4-6 Sqrt(CRLB) of position and attitudes using BP3010

In order to compare position solution performance between ErrINSEKF and GMIEKF, Sqrt(CRLB) for all IMUs in Table 4-1 are computed at the end of kinematic trajectory shown in

Figure 4-3 and depicted in Figure 4-7. Unsurprisingly, the solution performance of GMIEKF for all IMUs are unanimously superior to the solution performance of ErrINSEKF.

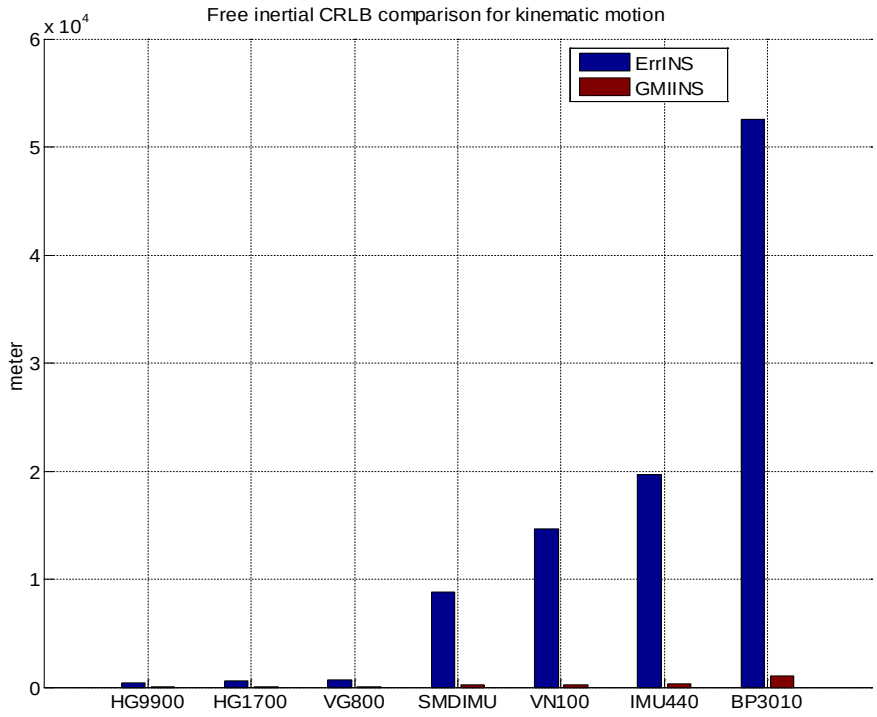


Figure 4-7 $\sqrt{\mathrm{CRLB}}$ of the position states at the end of trajectory spanning 180 s in time

CRLBs in loosely-coupled integration mode

Similarly, the CRLBs from ErrINSEKF and GMIEKF in loosely-coupled architecture are further compared with the integration of GPS measurements. Table 4-4 shows the accuracies of the external GNSS measurements in its third and fourth rows. It can be seen that the position and heading performance from ErrINSEKF and GMIEKF are getting closer together incurred from the GPS position and heading measurements. However, the CRLBs from GMIEKF are still better than those from ErrINSEKF because of the trajectory prediction in GMIEKF discussed in Section 4.2.1.

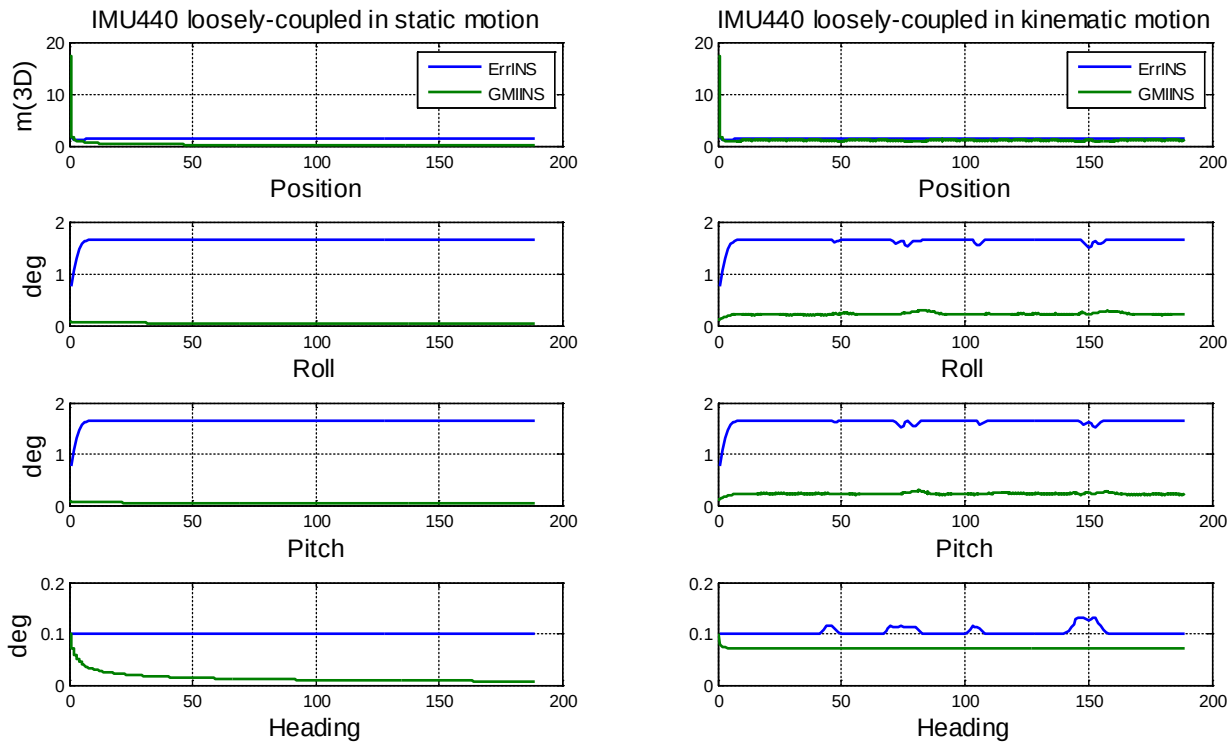


Figure 4-8 CRLBs of position and attitudes in loosely-coupled system using IMU440

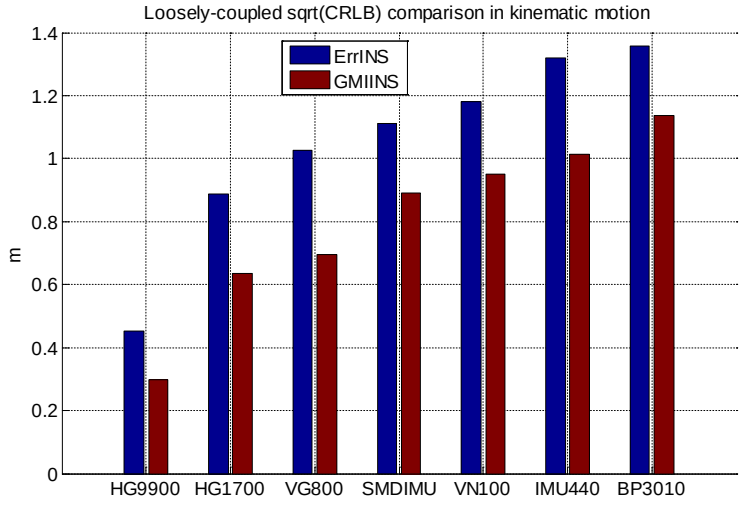


Figure 4-9 CRLBs of the position states in loosely-coupled integration architecture using different IMUs

4.3 Road test dataset

4.3.1 Overview

In order to demonstrate the success of the application of GMIKF upon the multisensor inertial navigation, a short land-based road test was conducted on April 1, 2012 using our in-house YUMIS system, which consisted of one Crossbow IMU440CA (MEMS) IMU and two NovAtel OEM4 GPS receivers (rovers). The lever arm parameters of two GPS receivers were measured beforehand at the accuracy of 0.5cm. The raw IMU data were collected at 100 Hz and the measurements (C/A code, carrier phase, Doppler) from two GPS receivers were acquired at 4Hz. Besides, one GPS base station near the start point was set up at the 1Hz data rate throughout the whole road test for the sake of relative positioning solution. Given that the number of GPS satellites in view was equal to 6 or greater, the high accuracy (~ 2cm) relative baseline solutions between the primary GPS receiver and the base receiver were achieved, and so were the baseline solutions for the vehicle's heading between two GPS rovers mounted on the vehicle's roof. Provided that the fixed baseline between two rovers is about 2 meters long, the overall 1σ accuracy of the derived DGPS heading can reach about $\pm 0.5^{o}$. The IMU specification and GPS hardware configuration are summarized in Table 4-5 and Table 4-6, respectively. The top view (2D) of the land vehicle's trajectory and the associated velocity profile are given in Figure 4-10. Figure 4-11 and Figure 4-12 show the GPS satellite availability during the road test.

Table 4-5 IMU 440CA specifications

Angular Rate (100Hz)	Bias Stability [º/hour]	<10.0
	Angle Random Walk [º/sqrt(hour)]	< 4.5
Acceleration (100Hz)	Bias Stability[mg]	<1.0
	Velocity Random Walk [m/s/sqrt(hour)]	<1.0

Table 4-6 GPS hardware configuration

No.	Receiver Type	Data Rate	Data Type	

1	NovAtel OEM4	4 Hz	C/A, L1, L2, D1	Rover
2	NovAtel OEM4	4 Hz	C/A, L1	Heading
3	NovAtel OEM4	1 Hz	C/A, L1, L2, D1	Base

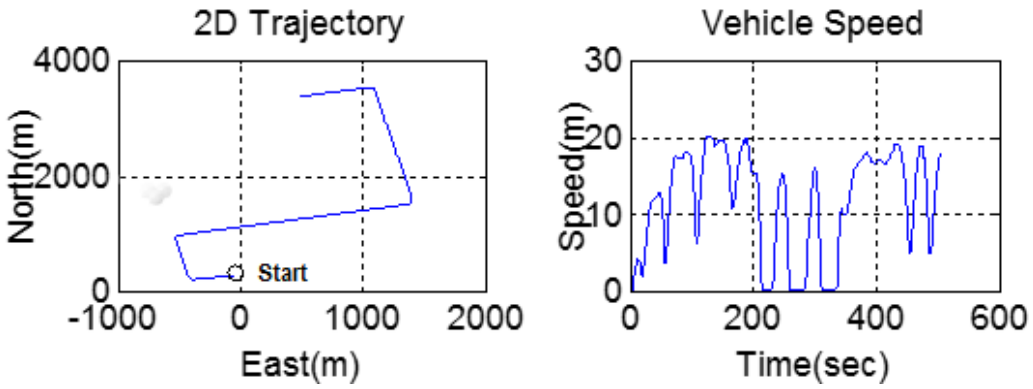


Figure 4-10 The top view of the trajectory and the speed profile

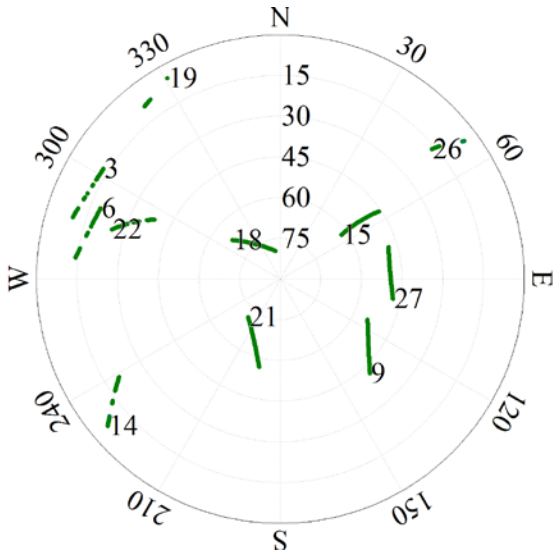


Figure 4-11 The sky plot of the available GPS satellites during the road test

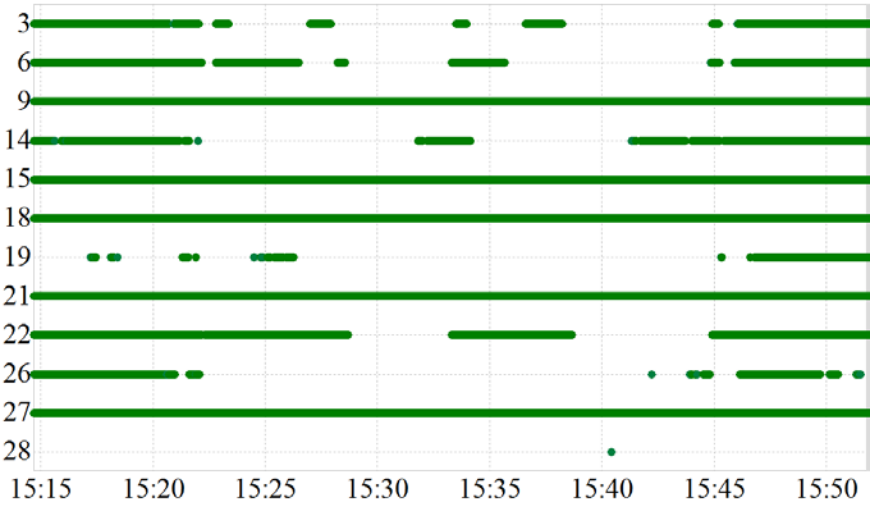


Figure 4-12 The GPS satellite availability during road test in time

4.3.2 The static alignment and initialization

In the beginning of the GMIEKF, the pitch and roll angles are initialized through static alignment for more than 5 minutes, during which the velocity, acceleration and angular rate vectors were all true zeroes. For each pair of IMU measurements (angular rates and specific forces), the pitch and roll angles were solved by using equation (4.3.1) [Bekir, 2007; Jekeli 2001]. The mean values of the pitch and roll angles from the static alignment were used as their initial values. Besides, the position and heading states in GMIEKF were initialized via the GPS position and GPS heading solution.

$$\hat{\mathbf{C}}_{\mathrm{b}}^{\mathrm{n}} = \left[\mathbf{g}^{\mathbf{n}} \quad \mathbf{g}^{\mathbf{n}} \times \boldsymbol{\Omega}^{\mathbf{n}} \quad \mathbf{g}^{\mathbf{n}} \times \boldsymbol{\Omega}^{\mathbf{n}} \times \mathbf{g}^{\mathbf{n}}\right]\left[\mathbf{a} \quad \mathbf{a} \times \boldsymbol{\omega} \quad \mathbf{a} \times \boldsymbol{\omega} \times \mathbf{a}\right]^{-1} \tag{4.3.1}$$

wherein

$\hat{\mathbf{C}}_{\mathrm{b}}^{\mathrm{n}}$ is the initial DCM matrix

$\mathbf{g}^{\mathbf{n}}$ is the local gravity vector,

$\boldsymbol{\Omega}^{\mathbf{n}}$ is the Earth rotation angular rate vector,

a is the accelerometer output, and

ω is the gyroscope output.

4.3.3 The loosely-coupled integration results

This sub-section presents the integrated navigation solution from the proposed GMIKF using loosely-coupled integration strategy. At the very beginning of the data processing, the static alignment module using (4.3.1) was executed until the GPS position and the DGPS heading solution became ready for the initialization of the complete state vector and its covariance matrix. The GMIKF was then carried on with the update rate of 100 Hz IMU measurements and 4 Hz GPS position, velocity measurements in addition to the GPS heading measurements. There are two GPS RTK solution (cm-level) measurement outages around 100 sec and 480 sec, in which the GPS position accuracy was degraded to meter-level.

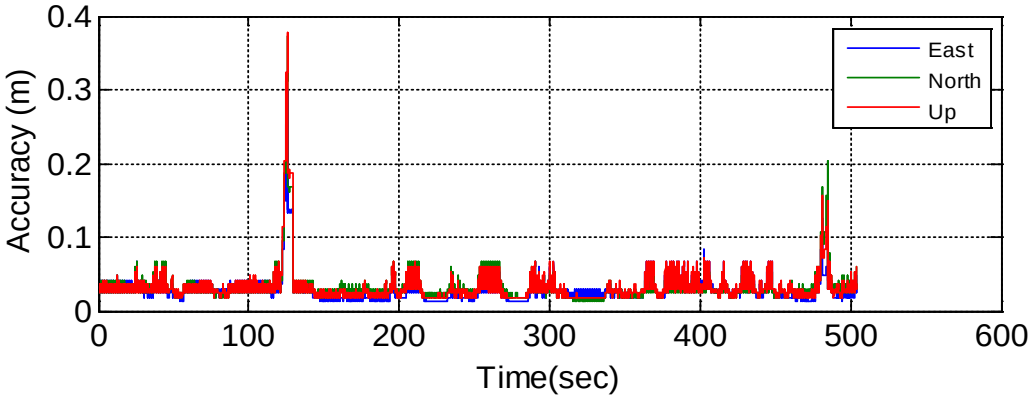


Figure 4-13 1σ (standard deviation) position accuracy in loosely-coupled GMIEKF

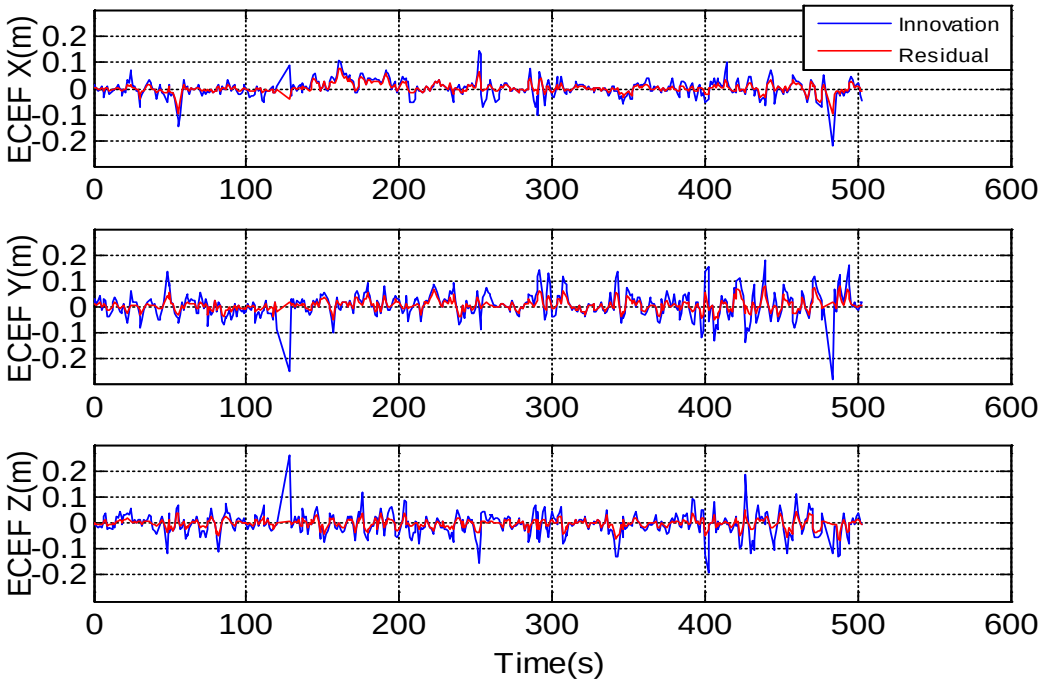


Figure 4-14 System innovations and residuals of GPS position measurements in loosely-coupled GMIEKF

As shown in Figure 4-13, the overall 1σ positional accuracies in east, north and up directions were about ±0.04m, ±0.04m and ±0.05m, correspondingly. Two big spikes were due to the low quality of the GPS position measurements (single point positioning solutions for those specific time periods). The cm-level system innovations and measurement residuals presented in Figure 4-14 for the GPS position measurements verify the correctness of the computed position accuracy.

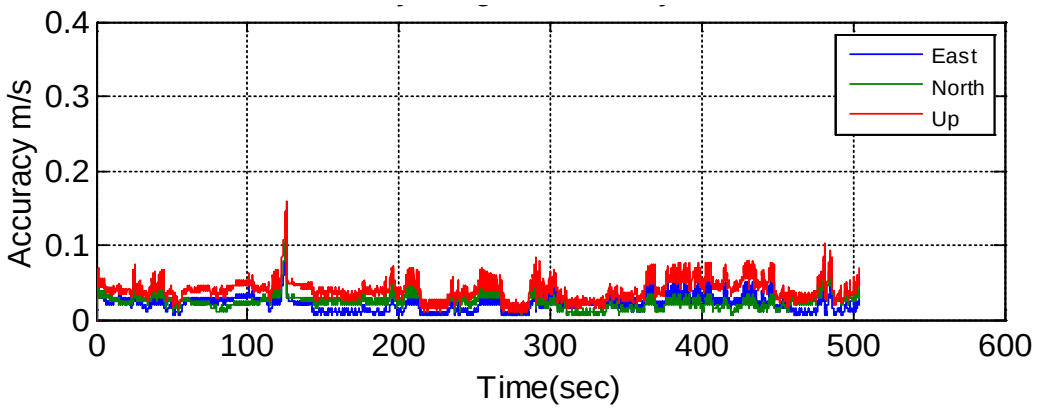


Figure 4-15 1σ (standard deviation) velocity accuracy in loosely-coupled GMIEKF

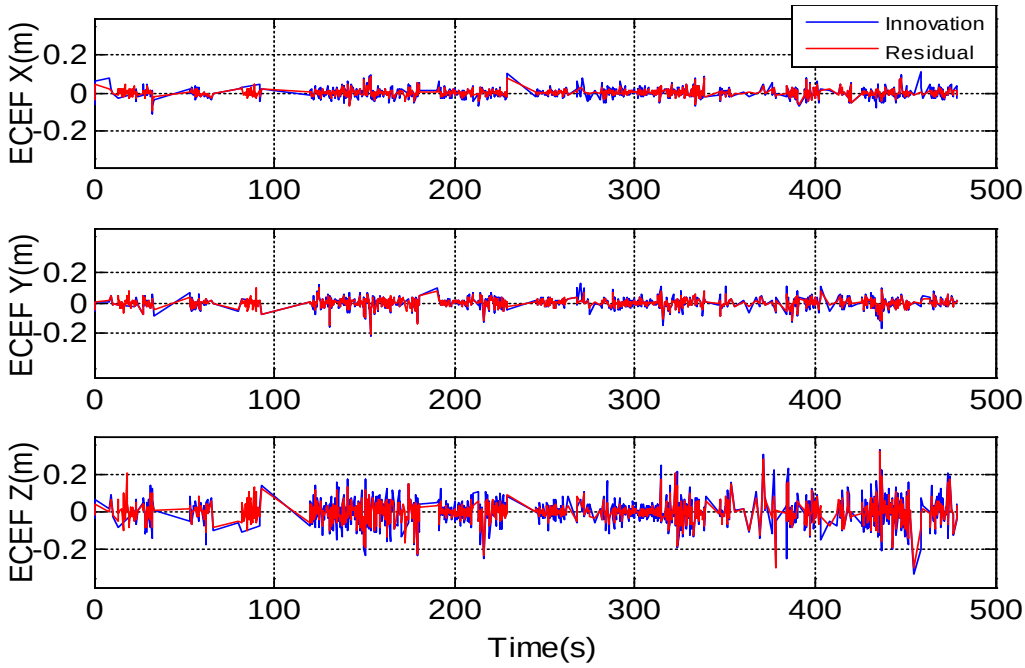


Figure 4-16 System innovations and residuals of GPS velocity measurements in loosely-coupled GMIEKF

Similarly, the accuracy of the velocity solution and the associated GPS velocity innovations and residuals are presented in Figure 4-15 and Figure 4-16, respectively. The overall 1σ accuracies for the velocity states in east, north and up directions were about ±0.04m, ±0.05m and ±0.07m, respectively.

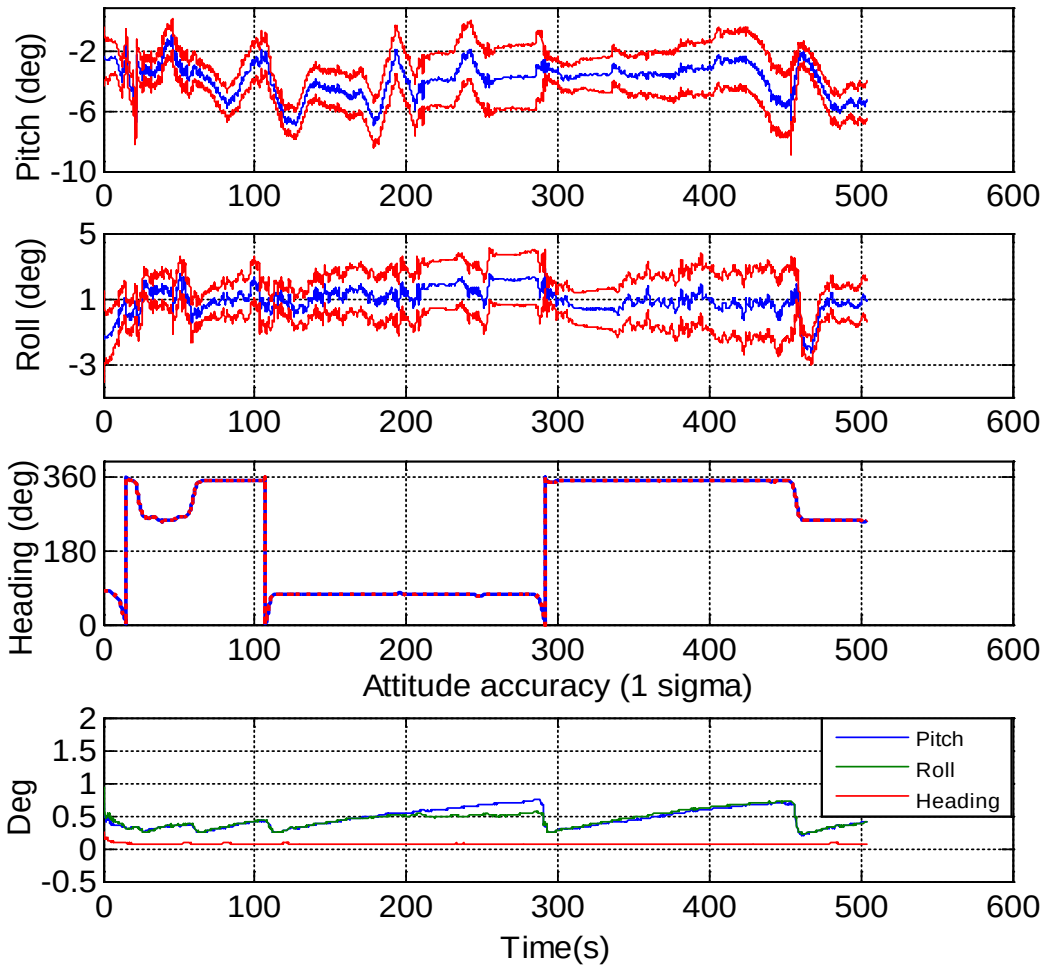


Figure 4-17 Attitude Solution with its 3σ Envelops (the first three plots) and attitude accuracy (the bottom plot) in loosely-coupled GMIEKF

Figure 4-17 shows the attitude (roll, pitch and heading) solution with the estimate in blue and its $\pm 3\sigma$ bounds in red. For the better visual effect, the 1σ attitude accuracies were also plotted. Figure 4-18 shows the time series of the system innovations and measurement residuals of the DGPS heading measurements. Once more, the innovations and residuals verify the computed standard deviation for pitch, roll and heading to be around $\pm 0.5^{o}$, $\pm 0.5^{o}$, and $\pm 0.1^{o}$ respectively.

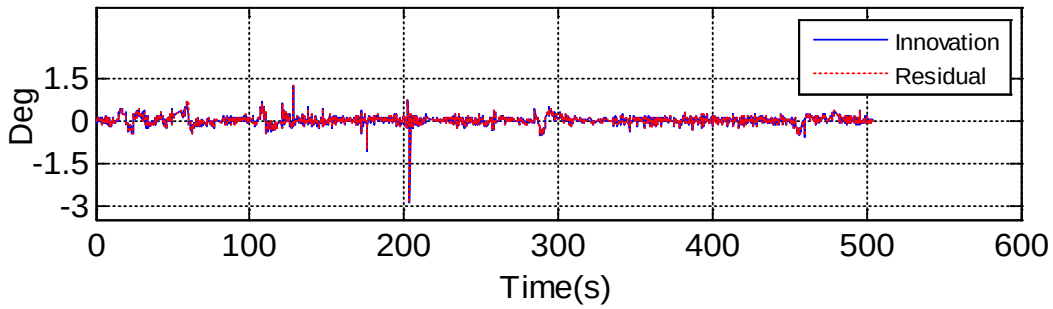


Figure 4-18 System innovations and residuals of DGPS heading measurements in loosely-coupled GMIEKF

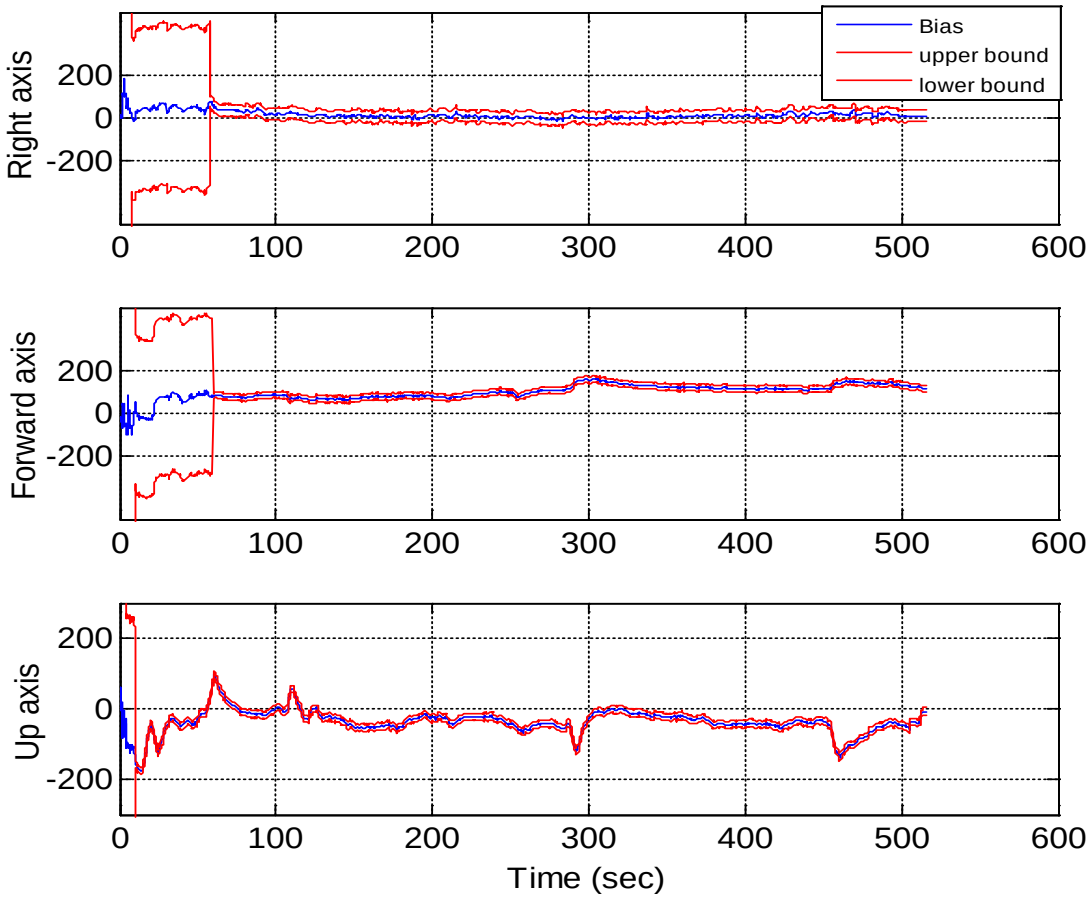


Figure 4-19 Gyroscope biases with their 3σ Envelops in loosely-coupled GMIEKF

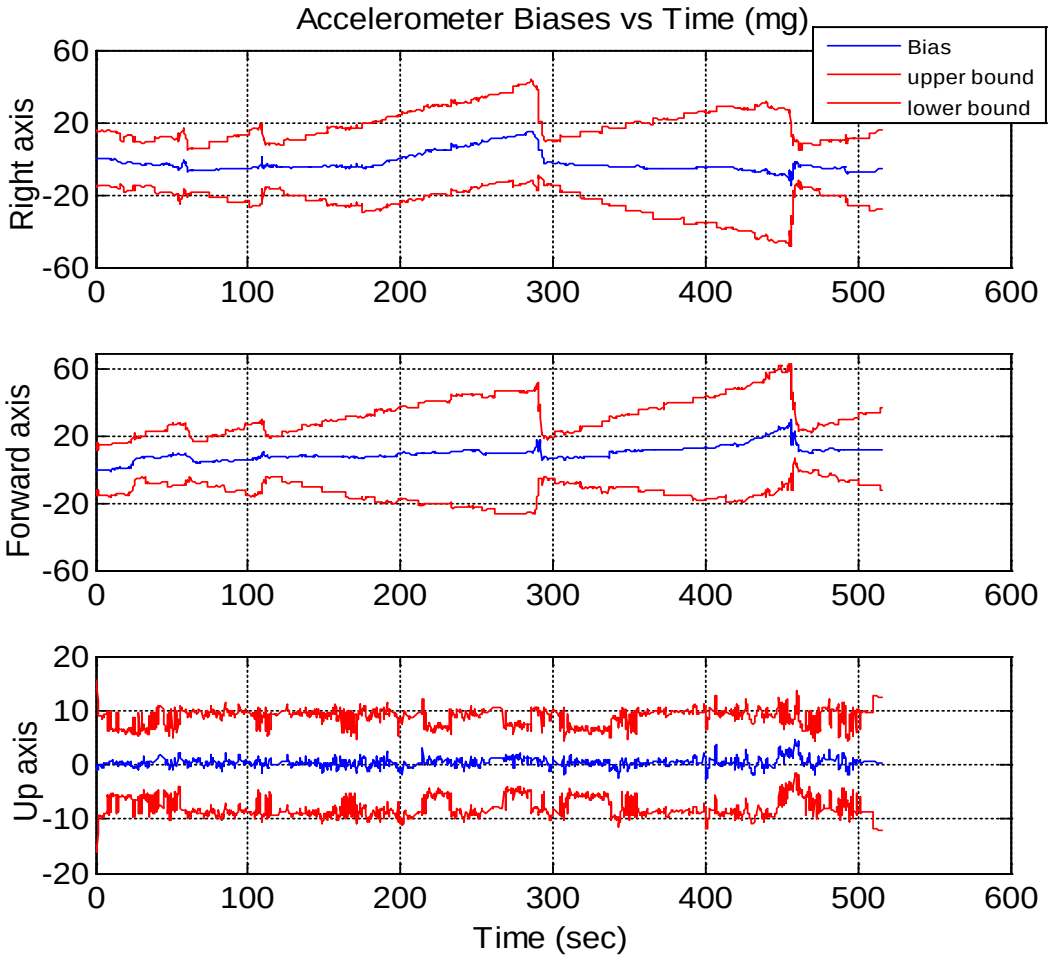


Figure 4-20 Accelerometer biases with their 3σ Envelops in loosely-coupled GMIEKF

Furthermore, the estimated bias state vector and its $\pm 3\sigma$ bounds in GMIEKF are shown in Figure 4-19 and Figure 4-20. The accuracies of the accelerometer bias drifts in right and forward directions are noticeably changed while the vehicle is turning because the observabilities of bias drifts during the turning period are enhanced.

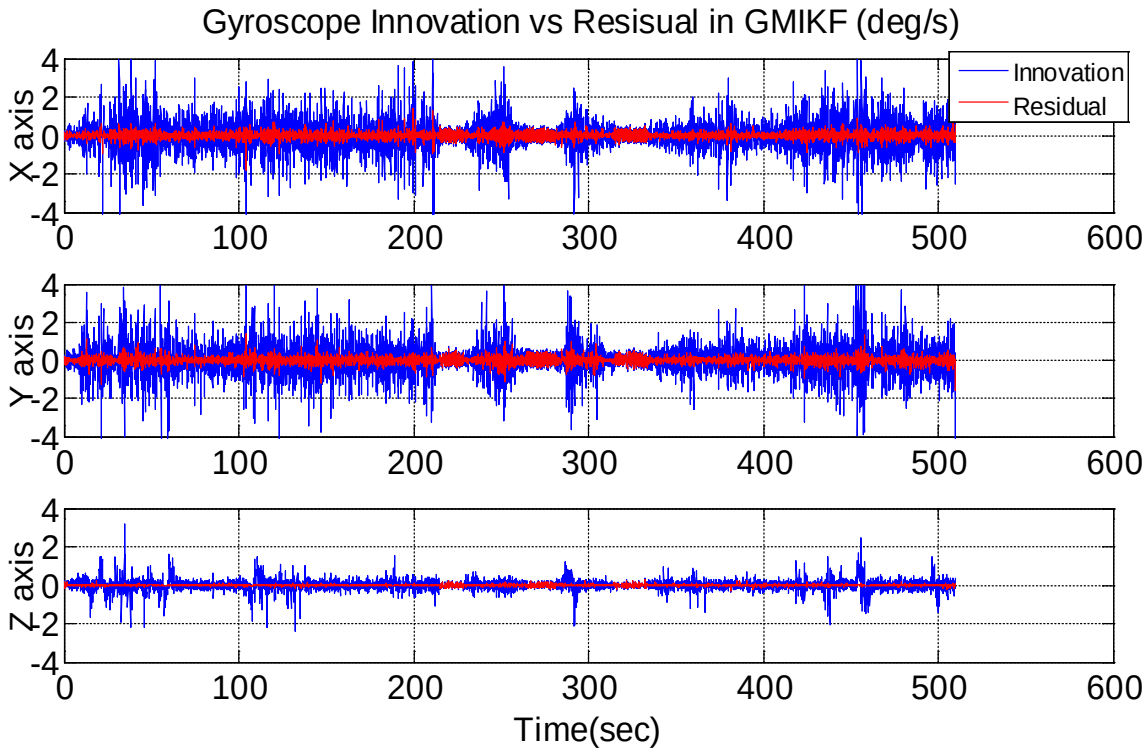


Figure 4-21 Measurement innovations and residuals for three gyroscopes in loosely-coupled GMIEKF

As one essential innovative feature of the GMIEKF, the IMU measurements directly participate in the Kalman filter through the measurement updates where the measurement accuracies are ± 0.5 m/s^2 (1σ standard deviation) for three accelerometers and $\pm 0.5^o$/s for three gyroscopes. In Figure 4-21, the small measurement residuals imply that three angular rate states well follow the raw gyroscopic measurements. In Figure 4-22, the amplitudes of the measurement residuals and the innovations for the specific forces sometimes differ from each other significantly, which indicates that the system model of GMIEKF exerts substantial influences on the estimation of the acceleration states.

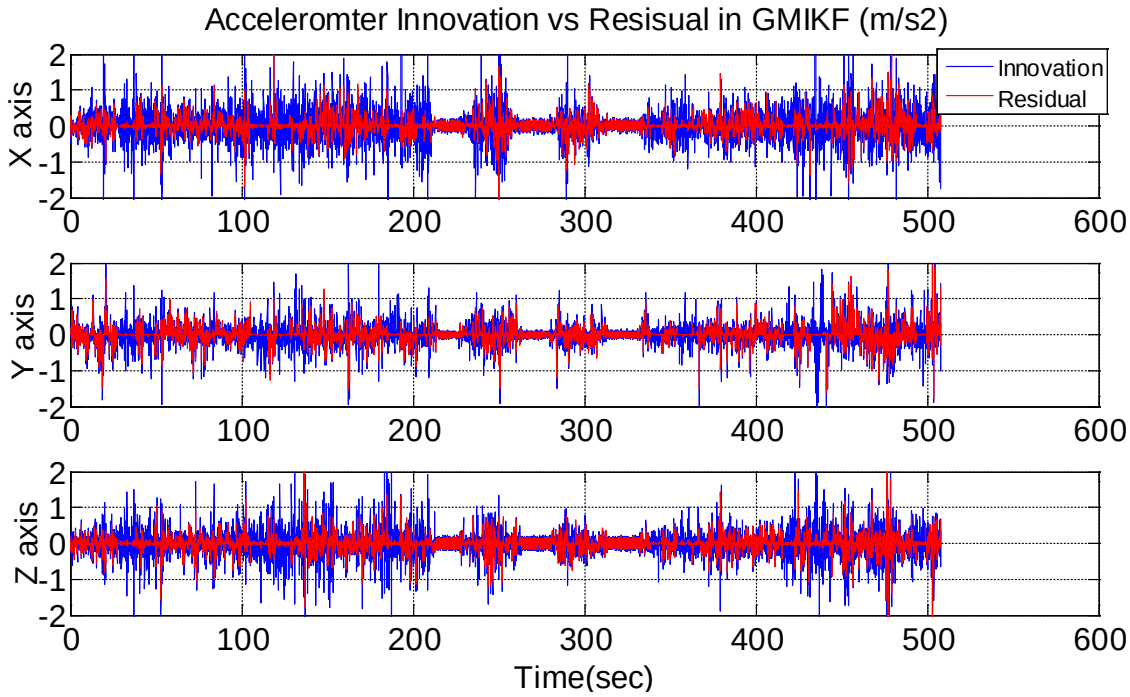


Figure 4-22 Accelerometer innovations and residuals in loosely-coupled GMIEKF

4.3.4 The tightly-coupled integration results

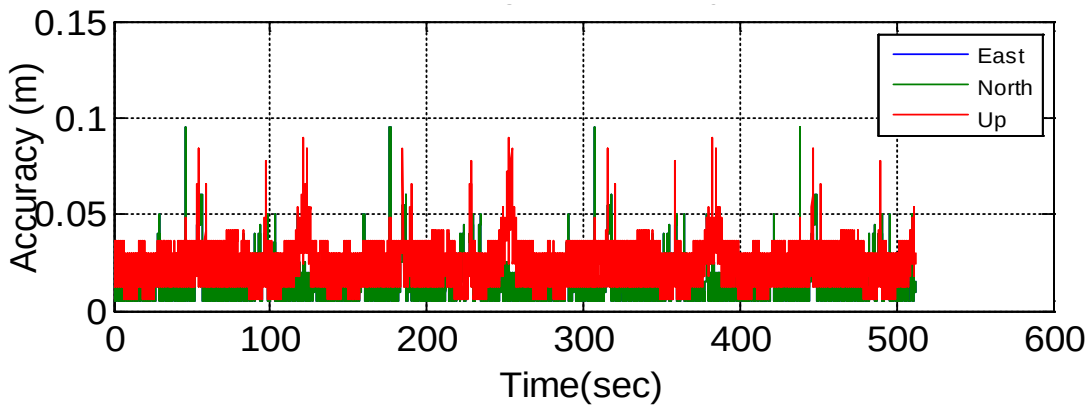


Figure 4-23 1σ (standard deviation) position accuracy in tightly-coupled GMIEKF

This sub-section shows the results from the tightly-coupled GMIEKF with the same dataset where the LAMBDA [Teunissen, 1995] method is used to fix the integer ambiguities for the

L1/L2 GPS carrier phase measurements. More details about ambiguity fix are explained in Section 6.1. Most of the plots were omitted except the 1σ accuracy for the position and velocity solutions because of the similarities between the results from the loosely-coupled GMIEKF and the ones from the tightly-coupled GMIEKF.

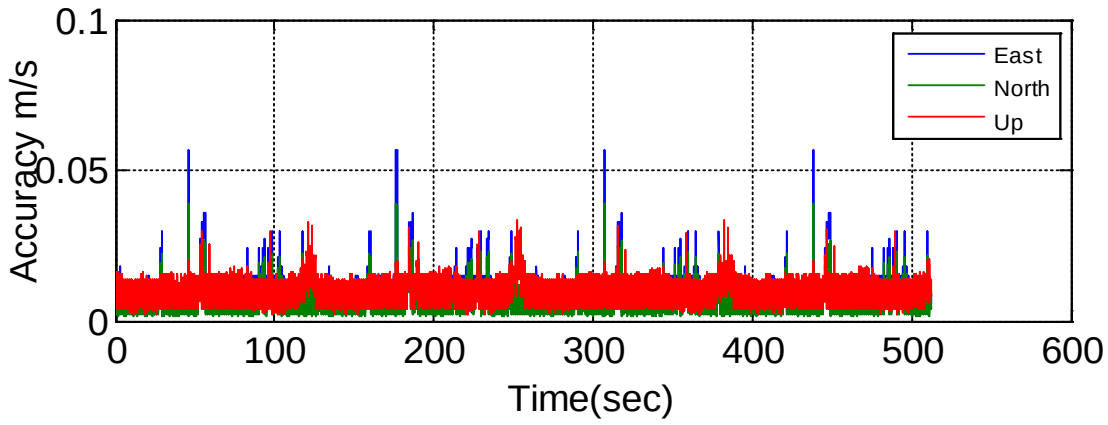


Figure 4-24 1σ (standard deviation) velocity accuracy in tightly-coupled GMIEKF

The advantages of the tightly-coupled GMIEKF were noticed at the two moments when the cm-level GPS RTK solutions could not be reached in the loosely-coupled GMIEKF. As can be seen in Figure 4-23 and Figure 4-24, the accuracy deteriorations of both position and velocity solutions in Figure 4-13 and 4-15 around the seconds 120 and 480 do not occur due to the measurement updates of the limited available carrier phase measurements in tightly-coupled GMIEKF.

In order to verify the success of the fusion of those L1 carrier phase measurements with fixed ambiguity integers in tightly-coupled GMIKF, Figure 4-25 shows the system innovation and residual series of the double differenced L1 carrier phases measurements for satellites SVPRN15, SVPRN18 and SVPRN22 with respect to the reference satellite SVPRN6.

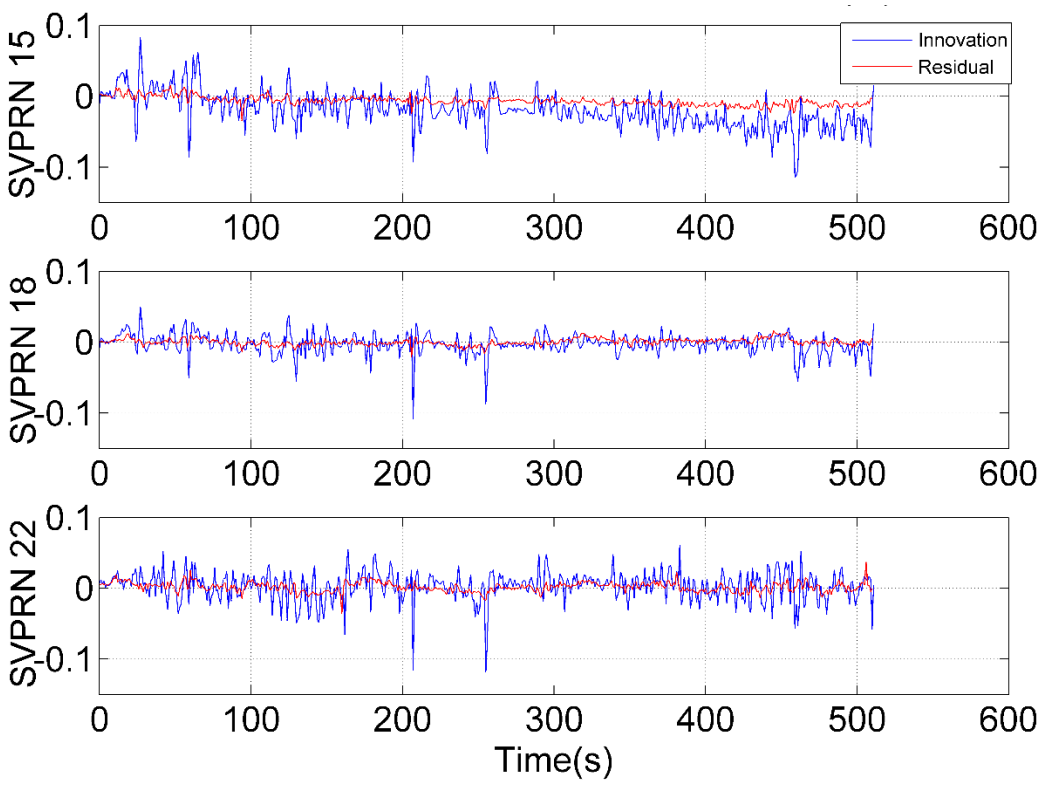


Figure 4-25 Innovations and residuals of DD L1 carrier phase measurements in tightly-coupled GMIEKF

4.3.5 Performance during GPS outages

In order to assess the performance of the free-inertial positioning and navigation solutions after the proposed GMIKF, eight 20-second long GPS outages were simulated based on segments of the reference dataset shown above. During the simulated outage periods, only the IMU raw outputs were used as the measurements. In the beginning of the simulated GPS outages, the navigation parameters for both of the free-inertial solutions were initialized to the same reference inertial integrated navigation solutions.

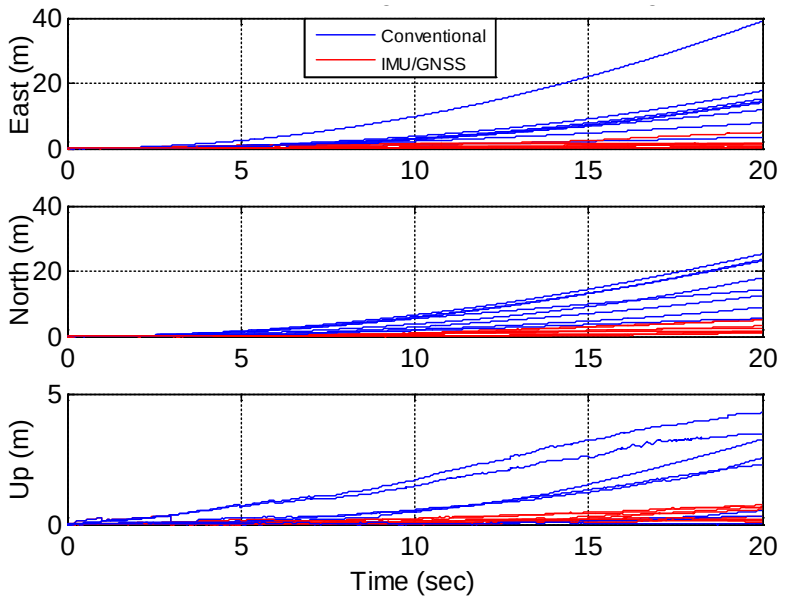


Figure 4-26 Position drifts during eight GNSS 20 seconds outages

For comparison purpose, the free inertial position solutions from the novel GMIEKF and the conventional inertial mechanization (or dead reckoning) are presented together in Figure 4-26. In addition, the largest position drifts for both methods during all eight outages in east, north and up directions are listed in Table 4-7 , in which the bottom row shows that the root mean square (RMS) of the eight (8) position drifts from the conventional dead reckoning (DR) and GMIKF are [±18.32m ±17.76m ±2.61m] and [±2.15m ±3.12m ±0.52m], respectively. As in Figure 4-26, the positions in the GMIEKF were undoubtedly drifted much slower than the ones in the free inertial navigation solution based on the conventional mechanization.

Table 4-7 Statistics with eight simulated GPS outages

	Conventional DR(m)			GMIKF (m)			Remark
	East	North	Up	East	North	Up	Motion
1	3.44	12.52	0.20	0.99	5.33	0.52	Straight line
2	15.27	25.25	0.33	1.60	1.09	0.29	Static
3	17.64	23.56	0.74	1.46	1.76	0.56	Turning

4	14.49	23.18	2.54	2.54	1.19	0.15	Turning
5	38.96	17.98	3.25	0.49	5.04	0.67	Straight line
6	11.87	5.47	4.26	0.12	3.22	0.73	Straight line
7	7.86	14.26	2.29	0.11	1.54	0.65	Static
8	14.05	8.96	3.67	4.95	2.41	0.35	Turning
RMS	18.32	17.76	2.61	2.15	3.12	0.52	Turning

4.3.6 Remarks

The computation cost for the GMIEKF is noticeably higher than the conventional integration strategy due to the high rate IMU measurement updates. Firstly, the prolonged computation time can be greatly shortened by the high performance matrix libraries making the best use of the computation power of CPUs or GPUs, e.g., Eigen, Intel® Math Kernel Library (MKL) and etc. In addition, the reasonable data compression algorithms, e.g., the pre-filter developed by Joglekar, et al [1975], can be employed to effectively reduce the computation cost without significant loss of the performance. Due to the scope of this dissertation, how to improve the computation speed without sacrificing solution accuracy will not be further discussed.

4.4 Summary

This chapter proposed a novel generic multisensor integration strategy for the integrated inertial kinematic positioning and navigation system in which the 3D kinematic trajectory model has been developed as the core of the system model in the proposed GMIEKF. The GMIEKF enables equal utilization of measurements from all sensors, inclusive of the IMU measurement data, through measurement update. The solution performance of GMIEKF has been analyzed through the comparison against the traditional algorithm (Error-state-based) upon both the simulated and real data. Unanimous superior performance is concluded in both situations

(simulation and real-time). The results showed that the proposed integration strategy can improve the accuracy and the robustness of the navigation solutions, especially with the low-cost IMU in poor GNSS environment and/or GNSS-denied environment.

It is worthwhile to mention the following facts:

1) The generic multisensory integration strategy for kinematic positioning and navigation does not distinguish the core sensor, usually an IMU, and the aiding sensors so that any positioning and/or orientation sensor can be directly integrated;
2) The introduction of the core of the system model on the ground of kinematics does significantly enhance the solution performance, especially during the GNSS outages, or in the case of a low-cost IMU;
3) The GMIEKF brings the opportunities for the direct error analysis upon the raw sensor (e.g., IMU and GNSS receiver) measurements in a multisensor system, for example, studying of the system redundancy distribution at the sensor level after [Wang, 1997], measurement outlier detection, the *a posteriori* variance component estimation of the measurement noises and the process noises. Specifically, as part of my PhD research, one achieved advancement is the work in Chapter 5: the *a posteriori* variance component estimation of the individual independent measurements and the individual independent components in the process noise vector using their residuals.

5. Variance component estimation upon generic multisensor integration Kalman filter

This chapter is based on the published work:

Qian, Kun; Jianguo Wang and Baoxin Hu (2016): ***A posteriori estimation of stochastic model for multi-sensor integrated inertial kinematic positioning and navigation on basis of variance component estimation***, Journal of Global Positioning Systems, 2016, vol. 14, 12 pages, SpringerOpen, DOI 10.1186/s41445-016-0005-5, ISSN: 1446-3164.

In this chapter, a generic method of tuning the stochastic models of the random errors in sensor measurements and the process noises in inertial navigation is proposed, especially to estimate the *posteriori* variance components for the IMU measurements directly in the extended Kalman filter for the first time.

Tuning the *a priori* stochastic models of the process and measurement noise vectors in Kalman Filer (KF) has always been a challenge. Especially in the multisensor integrated kinematic positioning and navigation demanding a cm-level accuracy, it becomes paramount because the solution accuracy quantified by the variance and covariance matrix of the state vector is expected to be statistically correct. As a preferable technique to address this challenge, the variance component estimation (VCE) applied to the Kalman Filter's process and measurement noise covariance matrix (**Q** and **R**) has been proved in various applications and in different ways. However, with the conventional error states-based GNSS aided inertial navigation (GNSS/INS) Kalman filter, it is difficult to tune the stochastic model for the measurements at sensor level due to the amalgamation of the inertial sensor measurements with the aiding sensors during the

construction of the error measurements. In other words, tuning the stochastic model of measurements is premised on the statistical independence of random errors among participant sensors.

However, the issue of the statistical dependences between the aiding sensor and inertial sensor measurements is cleared out by taking the innovative multisensor integration strategy described in Chapter 4, which deploys the vehicle's generic kinematic model and directly utilizes the raw outputs of all sensors (e.g., IMU, GPS receivers) through measurement updates in Kalman filter. As a result, the statistical independence of random errors among individual sensors naturally encourages the adoption of the VCE technique to estimate the variance components associated with the process and measurement noise vectors based on their residuals in GMIKF.

In this chapter, the variance component estimation targeting on GMIKF is detailed. Section 5.1 first re-formulates the GMIKF in order to prepare for the development of the VCE method. Then, the VCE process is described in Section 5.2. At last, Section 5.3 presents the numerical results together with the relevant performance analysis regarding one dataset.

5.1 Reformulation of generic multisensor integration Kalman filter

This section details the discretized system model and measurement model of a 27 state GMIKF in tightly-coupled integration architecture in order to facilitate the narratives in Sections 5.2 and 5.3.

5.1.1 System model

According to the continuous differential equations in Section 4.1.1, the discretized system model for 27 states in a tightly-coupled GMIKF is restated as follows:

$$\begin{bmatrix} \mathbf{r}^{n}_{nb(k+1)} \\ \mathbf{v}^{b}_{nb(k+1)} \\ \mathbf{a}^{b}_{nb(k+1)} \\ \boldsymbol{\theta}_{(k+1)} \\ \boldsymbol{\omega}^{b}_{nb(k+1)} \\ \mathbf{b}_{\omega(k+1)} \\ \mathbf{b}_{a(k+1)} \\ \mathbf{S}_{\omega(k+1)} \\ \mathbf{S}_{a(k+1)} \end{bmatrix} = f\left(\mathbf{r}^{n}_{nb(k)},\ \mathbf{v}^{b}_{nb(k)},\ \mathbf{a}^{b}_{nb(k)},\ \boldsymbol{\theta}_{(k)},\ \boldsymbol{\omega}^{b}_{nb(k)}\ \mathbf{b}_{\omega(k)}\ \mathbf{b}_{a(k)}\ \mathbf{S}_{\omega(k)}\ \mathbf{S}_{a(k)}\right) + \Lambda\left(\mathbf{j}^{n}_{nb},\ \dot{\boldsymbol{\omega}}^{b}_{nb}\ \mathbf{w}_{b_\omega}\ \mathbf{w}_{b_a}\ \mathbf{w}_{s_\omega}\ \mathbf{w}_{s_a}\right) \quad (5.1.1)$$

where

$$f\left(\mathbf{r}^{n}_{nb(k)},\ \mathbf{v}^{b}_{nb(k)},\ \mathbf{a}^{b}_{nb(k)},\ \boldsymbol{\theta}_{(k)},\ \boldsymbol{\omega}^{b}_{nb(k)}\ \mathbf{b}_{\omega(k)}\ \mathbf{b}_{a(k)}\ \mathbf{S}_{\omega(k)}\ \mathbf{S}_{a(k)}\right) =$$

$$\begin{bmatrix} \mathbf{r}^{n}_{nb}(k) + \mathbf{C}^{n}_{b}(k)\left[\mathbf{v}_k\Delta t + \frac{\mathbf{a}_k}{2}\Delta t^2 + \frac{(\boldsymbol{\omega}_k \times \mathbf{v}_k)}{2}\Delta t^2 + \frac{(\boldsymbol{\omega}_k \times \mathbf{a}_k)}{3}\Delta t^3\right] \\ \left(\mathbf{I}_{3x3} - \left[\boldsymbol{\omega}^{b}_{nb(k)}\times\right]\Delta t + \frac{\left[\boldsymbol{\omega}^{b}_{nb(k)}\times\right]^2}{2}\Delta t^2\right)\mathbf{v}^{b}_{nb(k)} + \left(\mathbf{I}_{3x3}\Delta t - \left[\boldsymbol{\omega}^{b}_{nb(k)}\times\right]\Delta t^2\right)\mathbf{a}^{b}_{nb(k)} \\ \left(\mathbf{I}_{3x3} - [\boldsymbol{\omega}^{b}_{nb(k)}\times]\Delta t + \frac{[\boldsymbol{\omega}^{b}_{nb(k)}\times]^2}{2}\Delta t^2\right)\mathbf{a}^{b}_{nb(k)} \\ \boldsymbol{\theta}_{(k)} + \mathrm{C}_{3\times3}\boldsymbol{\omega}^{b}_{nb(k)}\Delta t \\ \boldsymbol{\omega}^{b}_{nb(k)} \\ \mathbf{b}_{\omega(k)} \\ \mathbf{b}_{a(k)} \\ \mathbf{S}_{\omega(k)} \\ \mathbf{S}_{a(k)} \end{bmatrix},$$

and $\Lambda\left(\mathbf{j}^{n}_{nb},\ \dot{\boldsymbol{\omega}}^{b}_{nb}\ \mathbf{w}_{b_\omega}\ \mathbf{w}_{b_a}\ \mathbf{w}_{s_\omega}\ \mathbf{w}_{s_a}\right) =$

$$\begin{bmatrix} \mathbf{C}_b^n(k)\left[\left(\mathbf{I}+\left[\boldsymbol{\omega}_\mathbf{k}\times\right]\Delta t\right)\right]\frac{\mathbf{j}_{nb}^b}{6}\Delta t - \mathbf{C}_b^n(k)\left[\Delta\mathbf{r}\times\right]\frac{\dot{\boldsymbol{\omega}}_\mathbf{k}}{2}\Delta t^2 \\ \frac{\mathbf{j}_{nb}^b}{2}\Delta t^2 + \left[\mathbf{v}_{nb(k)}^b\times\right]\frac{\dot{\boldsymbol{\omega}}_{nb}^b}{2}\Delta t^2 \\ \left(\mathbf{I}_{3x3}\Delta t - \left[\boldsymbol{\omega}_{nb(k)}^b\times\right]\Delta t^2\right)\mathbf{j}_{nb}^b + \left[\mathbf{a}_{nb(k)}^b\times\right]\frac{\dot{\boldsymbol{\omega}}_{nb}^b}{2}\Delta t^2 \\ \mathbf{C}_{3\times3}\frac{\dot{\boldsymbol{\omega}}_{nb}^b}{2}\Delta t^2 \\ \dot{\boldsymbol{\omega}}_{nb}^b \\ \mathbf{w}_{\mathbf{b}_\omega} \\ \mathbf{w}_{\mathbf{b}_a} \\ \mathbf{w}_{\mathbf{s}_\omega} \\ \mathbf{w}_{\mathbf{s}_a} \end{bmatrix}.$$

The definitions of the variables in (5.1.1) can be referred to Section 4.1.2. Since the system model is a non-linear function of the state vector $\mathbf{x}_\mathbf{k}$, (5.1.1) is linearized for the purpose of the error (or variance and covariance) propagation as follows:

$$\mathbf{dX_{k+1}} = \begin{bmatrix} d\mathbf{r}_{nb(k+1)}^n \\ d\mathbf{v}_{\mathbf{nb}(k+1)}^\mathbf{b} \\ d\mathbf{a}_{\mathbf{nb}(k+1)}^\mathbf{b} \\ d\boldsymbol{\theta}_{(k+1)} \\ d\boldsymbol{\omega}_{\mathbf{nb}(k+1)}^\mathbf{b} \\ d\mathbf{b}_{\boldsymbol{\omega}(k+1)} \\ d\mathbf{b}_{\mathbf{a}(k+1)} \\ d\mathbf{S}_{\boldsymbol{\omega}(k+1)} \\ d\mathbf{S}_{\mathbf{a}(k+1)} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \frac{\partial\mathbf{r}_{(k+1)}}{\partial\mathbf{v}} & \frac{\partial\mathbf{r}_{(k+1)}}{\partial\mathbf{a}} & \frac{\partial\mathbf{r}_{(k+1)}}{\partial\boldsymbol{\theta}} & \frac{\partial\mathbf{r}_{(k+1)}}{\partial\boldsymbol{\omega}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \frac{\partial\mathbf{v}_{(k+1)}}{\partial\mathbf{v}} & \frac{\partial\mathbf{v}_{(k+1)}}{\partial\mathbf{a}} & \mathbf{0} & \frac{\partial\mathbf{v}_{(k+1)}}{\partial\boldsymbol{\omega}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \frac{\partial\mathbf{a}_{(k+1)}}{\partial\mathbf{a}} & \mathbf{0} & \frac{\partial\mathbf{a}_{(k+1)}}{\partial\boldsymbol{\omega}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \frac{\partial\boldsymbol{\theta}_{(k+1)}}{\partial\boldsymbol{\theta}} & \frac{\partial\boldsymbol{\theta}_{(k+1)}}{\partial\boldsymbol{\omega}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} d\mathbf{r}_{nb(k)}^n \\ d\mathbf{v}_{\mathbf{nb}(k)}^\mathbf{b} \\ d\mathbf{a}_{\mathbf{nb}(k)}^\mathbf{b} \\ d\boldsymbol{\theta}_{(k)} \\ d\boldsymbol{\omega}_{\mathbf{nb}(k)}^\mathbf{b} \\ d\mathbf{b}_{\boldsymbol{\omega}(k)} \\ d\mathbf{b}_{\mathbf{a}(k)} \\ d\mathbf{S}_{\boldsymbol{\omega}(k)} \\ d\mathbf{S}_{\mathbf{a}(k)} \end{bmatrix}$$

$$+\begin{bmatrix} \frac{\partial \mathbf{r}_{(k+1)}}{\partial \mathbf{j}} & \frac{\partial \mathbf{r}_{(k+1)}}{\partial \dot{\boldsymbol{\omega}}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \frac{\partial \mathbf{v}_{(k+1)}}{\partial \mathbf{j}} & \frac{\partial \mathbf{v}_{(k+1)}}{\partial \dot{\boldsymbol{\omega}}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \frac{\partial \mathbf{a}_{(k+1)}}{\partial \mathbf{j}} & \frac{\partial \mathbf{a}_{(k+1)}}{\partial \dot{\boldsymbol{\omega}}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \frac{\partial \boldsymbol{\theta}_{(k+1)}}{\partial \dot{\boldsymbol{\omega}}} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} d\mathbf{j}_{nb}^{n} \\ d\dot{\boldsymbol{\omega}}_{\mathbf{nb}}^{\mathbf{b}} \\ d\mathbf{w}_{\mathbf{b}_\omega} \\ d\mathbf{w}_{\mathbf{b}_a} \\ d\mathbf{w}_{\mathbf{s}_\omega} \\ d\mathbf{w}_{\mathbf{s}_a} \end{bmatrix} = \boldsymbol{\Phi}_k \mathbf{dX}_k + \boldsymbol{\Lambda}_k \mathbf{dw}_k \qquad (5.1.2)$$

where

$$\frac{\partial \mathbf{r}_{(k+1)}}{\partial \mathbf{v}} = \mathbf{C}_b^n(k)\left(\mathbf{I}\Delta t + \frac{[\boldsymbol{\omega}_\mathbf{k} \times]}{2}\Delta t^2\right), \qquad \frac{\partial \mathbf{r}_{(k+1)}}{\partial \mathbf{a}} = \mathrm{C}_b^n(k)\left(\mathbf{I}\frac{\Delta t^2}{\mathbf{2}} + \frac{[\boldsymbol{\omega}_\mathbf{k} \times]}{3}\Delta t^3\right),$$

$$\frac{\partial \mathbf{r}_{(k+1)}}{\partial \boldsymbol{\theta}} = \left[\left(\mathbf{C}_b^n(k)\left[\Delta \mathbf{r} + \frac{(\boldsymbol{\omega}_\mathbf{k} \times \mathbf{v}_\mathbf{k})}{2}\Delta t^2 + \frac{(\boldsymbol{\omega}_\mathbf{k} \times \mathbf{a}_\mathbf{k})}{3}\Delta t^3\right]\right) \times\right]\mathbf{T}, \text{ with } \mathbf{T} = \begin{bmatrix} c\hat{\gamma} & 0 & c\hat{p}\, s\hat{\gamma} \\ 0 & 1 & -s\hat{p} \\ s\hat{\gamma} & 0 & -c\hat{p}\, c\hat{\gamma} \end{bmatrix},$$

$$\frac{\partial \mathbf{r}_{(k+1)}}{\partial \boldsymbol{\omega}} = \mathbf{C}_b^n(k)\left[\frac{[\mathbf{v}_\mathbf{k} \times]}{-2}\Delta t^2 + \frac{[\mathbf{a}_\mathbf{k} \times]}{-3}\Delta t^3\right], \qquad \frac{\partial \mathbf{v}_{(k+1)}}{\partial \mathbf{v}} = \left(\mathbf{I}_{3x3} - [\boldsymbol{\omega}_{nb(k)}^b \times]\Delta t + \frac{[\boldsymbol{\omega}_{nb(k)}^b \times]^2}{2}\Delta t^2\right),$$

$$\frac{\partial \mathbf{v}_{(k+1)}}{\partial \mathbf{a}} = \left(\mathbf{I}_{3x3}\Delta t - [\boldsymbol{\omega}_{nb(k)}^b \times]\Delta t^2\right), \qquad \frac{\partial \mathbf{v}_{(k+1)}}{\partial \boldsymbol{\omega}} = \left([\mathbf{v}_{nb(k)}^b \times]\Delta t + [\mathbf{a}_{nb(k)}^b \times]\Delta t^2\right),$$

$$\frac{\partial \mathbf{a}_{(k+1)}}{\partial \mathbf{a}} = \left(\mathbf{I}_{3x3} - [\boldsymbol{\omega}_{nb(k)}^b \times]\Delta t + \frac{[\boldsymbol{\omega}_{nb(k)}^b \times]^2}{2}\Delta t^2\right), \quad \frac{\partial \mathbf{a}_{(k+1)}}{\partial \boldsymbol{\omega}} = [\mathbf{a}_{nb(k)}^b \times]\Delta t,$$

$$\frac{\partial \boldsymbol{\theta}_{(k+1)}}{\partial \boldsymbol{\theta}} = \begin{bmatrix} 1 & (-s\gamma\, \omega_{nbx}^b + c\gamma\, \omega_{nbz}^b)\Delta t & 0 \\ \frac{(s\gamma\, \omega_{nbx}^b - c\gamma\, \omega_{nbz}^b)\Delta t}{cp^2} & 1 + (c\gamma\, \omega_{nbx}^b + s\gamma\, \omega_{nbz}^b)\mathrm{tg}p\,\Delta t & 0 \\ \frac{\mathrm{tg}p(s\gamma\, \omega_{nbx}^b - c\gamma\, \omega_{nbz}^b)\Delta t}{cp} & \frac{(c\gamma\, \omega_{nbx}^b + s\gamma\, \omega_{nbz}^b)\,\Delta t}{cp} & 1 \end{bmatrix},$$

$$\frac{\partial \boldsymbol{\theta}_{(k+1)}}{\partial \boldsymbol{\omega}} = \begin{bmatrix} c\gamma_{(k)} & 0 & s\gamma_{(k)} \\ s\gamma_{(k)}\, tg p_{(k)} & 1 & -c\gamma_{(k)}\, tg p_{(k)} \\ s\gamma_{(k)} \sec p_{(k)} & 0 & -c\gamma_{(k)} \sec p_{(k)} \end{bmatrix} \Delta t ,$$

$$\frac{\partial \mathbf{r}_{(k+1)}}{\partial \mathbf{j}} = C_b^n(k)\left[\left(\mathbf{I} + [\boldsymbol{\omega}_{\mathbf{k}} \times]\Delta t\right)\right]\frac{\Delta t^3}{6}, \qquad \frac{\partial \mathbf{r}_{(k+1)}}{\partial \dot{\boldsymbol{\omega}}} = C_b^n(k)[\Delta \mathbf{r} \times]\frac{\Delta t^2}{(-2)},$$

$$\frac{\partial \mathbf{v}_{(k+1)}}{\partial \mathbf{j}} = \mathbf{I}\frac{\Delta t^2}{2}, \qquad \frac{\partial \mathbf{v}_{(k+1)}}{\partial \dot{\boldsymbol{\omega}}} = [\mathbf{v}_{nb(k)}^b \times]\frac{\Delta t^2}{2},$$

$$\frac{\partial \mathbf{a}_{(k+1)}}{\partial \mathbf{j}} = \left(\mathbf{I}_{3x3}\Delta t - [\boldsymbol{\omega}_{nb(k)}^b \times]\Delta t^2\right), \qquad \frac{\partial \mathbf{a}_{(k+1)}}{\partial \dot{\boldsymbol{\omega}}} = [\mathbf{a}_{nb(k)}^b \times]\frac{\Delta t^2}{2},$$

and $$\frac{\partial \boldsymbol{\theta}_{(k+1)}}{\partial \dot{\boldsymbol{\omega}}} = \begin{bmatrix} c\gamma_{(k)} & 0 & s\gamma_{(k)} \\ s\gamma_{(k)}\, tg p_{(k)} & 1 & -c\gamma_{(k)}\, tg p_{(k)} \\ s\gamma_{(k)} \sec p_{(k)} & 0 & -c\gamma_{(k)} \sec p_{(k)} \end{bmatrix} \frac{\Delta t^2}{2}.$$

The explanation of the symbols in (5.1.2) can also be referred to Section 4.1.2 and the counterparts in (5.1.1).

5.1.2 Measurement model

5.1.2.1 IMU measurement model

The general IMU measurement equations participating the measurement update in GMIKF are restated as follows:

$$\mathbf{Z}_{\mathbf{IMU}} = \begin{bmatrix} \boldsymbol{\omega}_{\mathbf{ib\text{-}imu}}^{\mathbf{b}} \\ \boldsymbol{f}_{\mathbf{ib\text{-}imu}}^{\mathbf{b}} \end{bmatrix} = \begin{bmatrix} (\mathbf{I} + \mathbf{S}_\omega)\left(\boldsymbol{\omega}_{nb}^b + \mathbf{C}_n^b(\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n)\right) + \mathbf{b}_\omega \\ (\mathbf{I} + \mathbf{S}_a)\left(\mathbf{a}_{nb}^b + \mathbf{C}_n^b\left((2\boldsymbol{\omega}_{ie}^n + \boldsymbol{\omega}_{en}^n) \times \mathbf{v}_{nb}^n\right) - \mathbf{C}_n^b\mathbf{g}^n\right) + \mathbf{b}_a \end{bmatrix} + \begin{bmatrix} \nabla_\omega \\ \nabla_a \end{bmatrix} \qquad (5.1.3)$$

The definitions of variables in 5.1.3 can be referred to those in Section 2.4.4. In addition, the Jacobians matrix **H** for the measurement vector $\mathbf{z}_{\mathbf{IMU}}$ in (5.1.3) is given by:

$$\mathbf{H}_{\mathbf{imu}} = \left.\frac{\partial \mathbf{z}_{\mathbf{IMU}}}{\partial \mathbf{x}}\right|_{\hat{\mathbf{x}}_{\mathbf{k|k-1}}} =$$

$$\begin{bmatrix} \frac{\partial \boldsymbol{\omega}^{\mathbf{b}}_{\mathbf{ib\text{-}imu}}}{\partial \mathbf{r}^{\mathbf{n}}_{\mathbf{nb}}} & \frac{\partial \boldsymbol{\omega}^{\mathbf{b}}_{\mathbf{ib\text{-}imu}}}{\partial \mathbf{v}^{\mathbf{b}}_{\mathbf{nb}}} & \mathbf{0} & \frac{\partial \boldsymbol{\omega}^{\mathbf{b}}_{\mathbf{ib\text{-}imu}}}{\partial \boldsymbol{\theta}} & \mathbf{I}+\mathbf{S}_{\boldsymbol{\omega}} & \mathbf{I} & \mathbf{0} & \frac{\partial \boldsymbol{\omega}^{\mathbf{b}}_{\mathbf{ib\text{-}imu}}}{\partial \mathbf{S}_{\boldsymbol{\omega}}} & \mathbf{0} \\ \frac{\partial \boldsymbol{f}^{\mathbf{b}}_{\mathbf{ib\text{-}imu}}}{\partial \mathbf{r}^{\mathbf{n}}_{\mathbf{nb}}} & \frac{\partial \boldsymbol{f}^{\mathbf{b}}_{\mathbf{ib\text{-}imu}}}{\partial \mathbf{v}^{\mathbf{b}}_{\mathbf{nb}}} & \mathbf{I}+\mathbf{S}_{\mathbf{a}} & \frac{\partial \boldsymbol{f}^{\mathbf{b}}_{\mathbf{ib\text{-}imu}}}{\partial \boldsymbol{\theta}} & \mathbf{0} & \mathbf{0} & \mathbf{I} & \mathbf{0} & \frac{\partial \boldsymbol{\omega}^{\mathbf{b}}_{\mathbf{ib\text{-}imu}}}{\partial \mathbf{S}_{\mathbf{a}}} \end{bmatrix} \tag{5.1.4}$$

with $\frac{\partial \boldsymbol{f}^{\mathbf{b}}_{\mathbf{ib\text{-}imu}}}{\partial \boldsymbol{\theta}} \approx \frac{\partial \left(\mathbf{C}^{\mathrm{b}}_{\mathrm{n}}\right)\mathbf{a}^{\mathrm{n}}_{\mathrm{hazard}}}{\partial \boldsymbol{\theta}}$, and $\mathbf{a}^{\mathrm{n}}_{\mathrm{hazard}} = (2\boldsymbol{\omega}^{\mathrm{n}}_{\mathrm{ie}} + \boldsymbol{\omega}^{\mathrm{n}}_{\mathrm{en}}) \times \mathbf{v}^{\mathrm{n}}_{\mathrm{nb}} - \mathbf{g}^{\mathrm{n}}$ which is the hazard acceleration vector. In (5.1.4), because $\mathbf{C}^{\mathrm{b}}_{\mathrm{n}}\mathbf{C}^{\mathrm{n}}_{\mathrm{b}} = \mathbf{I}^{\mathrm{b}}_{\mathrm{n}}$, the term $(\delta\mathbf{C}^{\mathrm{b}}_{\mathrm{n}})\mathbf{a}^{\mathrm{n}}_{\mathrm{hazard}}$ is further approximated as follows:

$$\begin{aligned} (\delta\mathbf{C}^{\mathrm{b}}_{\mathrm{n}})\mathbf{a}^{\mathrm{n}}_{\mathrm{hazard}} &\approx -\hat{\mathbf{C}}^{\mathrm{b}}_{\mathrm{n}}\delta\hat{\mathbf{C}}^{\mathrm{n}}_{\mathrm{b}}\hat{\mathbf{C}}^{\mathrm{b}}_{\mathrm{n}}\hat{\mathbf{a}}^{\mathrm{n}}_{\mathrm{hazard}} = \hat{\mathbf{C}}^{\mathrm{b}}_{\mathrm{n}}[\boldsymbol{\varphi}\times]\hat{\mathbf{C}}^{\mathrm{n}}_{\mathrm{b}}\hat{\mathbf{C}}^{\mathrm{b}}_{\mathrm{n}}\hat{\mathbf{a}}^{\mathrm{n}}_{\mathrm{hazard}} \\ &= \hat{\mathbf{C}}^{\mathrm{b}}_{\mathrm{n}}[\boldsymbol{\varphi}\times]\hat{\mathbf{a}}^{\mathrm{n}}_{\mathrm{hazard}} = -\hat{\mathbf{C}}^{\mathrm{b}}_{\mathrm{n}}[\hat{\mathbf{a}}^{\mathrm{n}}_{\mathrm{hazard}}\times]\,\boldsymbol{\varphi} \\ &= -\hat{\mathbf{C}}^{\mathrm{b}}_{\mathrm{n}}[\hat{\mathbf{a}}^{\mathrm{n}}_{\mathrm{hazard}}\times]\mathbf{T}(\hat{\boldsymbol{\theta}})\delta\boldsymbol{\theta} \end{aligned} \tag{5.1.5}$$

where $\boldsymbol{\varphi} = \begin{bmatrix}\phi^{\mathrm{b}}_{\mathrm{nbx}} & \phi^{\mathrm{b}}_{\mathrm{nby}} & \phi^{\mathrm{b}}_{\mathrm{nbz}}\end{bmatrix}^{\mathrm{T}}$ is the instantaneous attitude error vector, the operator $\wedge$ represents the nominal value of the unknown variables, and **T** as defined in (5.1.2) is the transformation matrix between the vector $\boldsymbol{\varphi}$ and the Euler angle error vector $\delta\boldsymbol{\theta}$. As a result, the first order partial derivative of $\boldsymbol{f}^{\mathbf{b}}_{\mathbf{ib\text{-}imu}}$ with respect to Euler angle vector $\boldsymbol{\theta}$ is given as follows:

$$\frac{\partial \boldsymbol{f}^{\mathbf{b}}_{\mathbf{ib\text{-}imu}}}{\partial \boldsymbol{\theta}} = \frac{\partial \left(\mathbf{C}^{\mathrm{b}}_{\mathrm{n}}\right)\mathbf{a}^{\mathrm{n}}_{\mathrm{hazard}}}{\partial \boldsymbol{\theta}} = -\hat{\mathbf{C}}^{\mathrm{b}}_{\mathrm{n}}\left[\hat{\mathbf{a}}^{\mathrm{n}}_{\mathrm{hazard}}\times\right]\begin{bmatrix} \mathrm{c}\hat{\gamma} & 0 & \mathrm{c}\hat{p}\,\mathrm{s}\hat{\gamma} \\ 0 & 1 & -\mathrm{s}\hat{p} \\ \mathrm{s}\hat{\gamma} & 0 & -\mathrm{c}\hat{p}\,\mathrm{c}\hat{\gamma} \end{bmatrix} \tag{5.1.6}$$

In practice, the random errors in the measurements from three gyros and three accelerometers can be considered statistically independent after ignoring the small timing error and the tiny installation error compared to the other error sources. Correspondingly, the covariance matrix for these six independent measurements will be a 6×6 diagonal matrix. Taking into account the variance components, the covariance matrix for IMU sensor (3 gyroscopes and 3 accelerometers) can be modeled as follows:

$$\mathbf{R_{IMU}} = \begin{bmatrix} \sigma^2_{\text{gyro-x}} R^2_{\omega_x} & 0 & 0 & 0 & 0 & 0 \\ & \sigma^2_{\text{gyro-y}} R^2_{\omega_y} & 0 & 0 & 0 & 0 \\ & & \sigma^2_{\text{gyro-z}} R^2_{\omega_z} & 0 & 0 & 0 \\ & & & \sigma^2_{\text{acc-x}} R^2_{a_x} & 0 & 0 \\ & \textit{Symmetrical} & & & \sigma^2_{\text{acc-y}} R^2_{a_y} & 0 \\ & & & & & \sigma^2_{\text{acc-z}} R^2_{a_z} \end{bmatrix}$$

where $R^2_{\omega_x}$, $R^2_{\omega_y}$, $R^2_{\omega_z}$, $R^2_{a_x}$, $R^2_{a_y}$ and $R^2_{a_z}$ are six *a priori* variance elements for six sensors in one IMU and $\mathbf{s_{IMU}} = \left[\sigma^2_{\text{gyro-x}} \quad \sigma^2_{\text{gyro-z}} \quad \sigma^2_{\text{gyro-z}} \quad \sigma^2_{\text{acc-x}} \quad \sigma^2_{\text{acc-y}} \quad \sigma^2_{\text{acc-z}}\right]^{\text{T}}$ is the IMU's variance component vector of the unit weight.

5.1.2.2 GPS measurement model

By taking the tightly-coupled GMIKF, three types of GPS measurements, the double-differenced pseudo-ranges and carrier phases (L1 and L2), and range rates (Doppler velocity) are used:

$$\mathbf{Z_{GPS}} = \begin{bmatrix} \nabla\Delta\rho^{\text{jk}}_{\text{AB}}(\text{t}) \\ \nabla\Delta\varphi^{\text{jk}}_{\text{AB}}(\text{t}) \\ \nabla\Delta\dot{\rho}^{\text{jk}}_{\text{AB}}(\text{t}) \end{bmatrix} = \begin{bmatrix} \left\|\nabla\Delta\mathbf{r}(\text{t})^{\text{jk}}_{\text{AB}}\right\| \\ \left\|\nabla\Delta\mathbf{r}(\text{t})^{\text{jk}}_{\text{AB}}\right\| + \lambda_1 \nabla\Delta\text{N}^{\text{jk}}_{\text{AB}} \\ \left\|\nabla\Delta\dot{\mathbf{r}}(\text{t})^{\text{jk}}_{\text{AB}}\right\| \end{bmatrix} + \begin{bmatrix} \varepsilon^{\text{jk}}_{\nabla\Delta\rho_{\text{AB}}}(\text{t}) \\ \varepsilon^{\text{jk}}_{\nabla\Delta\varphi_{\text{AB}}}(\text{t}) \\ \varepsilon^{\text{jk}}_{\nabla\Delta\dot{\rho}_{\text{AB}}}(\text{t}) \end{bmatrix} \tag{5.1.7}$$

where $\nabla\Delta L^{\text{jk}}_{\text{AB}} = [L^{\text{k}}_{\text{B}}(\text{t}) - L^{\text{k}}_{\text{A}}(\text{t})] - [L^{\text{j}}_{\text{B}}(\text{t}) - L^{\text{j}}_{\text{A}}(\text{t})]$ is the double-differenced (DD) operation on the observables L between two stations A and B with respect to satellite j and k. The measurement noises of the DD pseudo-ranges, carrier phases and range rates are denoted by $\varepsilon^{\text{jk}}_{\nabla\Delta\rho_{\text{AB}}}(\text{t})$, $\varepsilon^{\text{jk}}_{\nabla\Delta\varphi_{\text{AB}}}(\text{t})$ and $\varepsilon^{\text{jk}}_{\nabla\Delta\dot{\rho}_{\text{AB}}}(\text{t})$.

Measurement noise model for the GPS measurements has been studied for years [Collins et al, 1999; Wieser et al, 2005; Luo et al, 2009; Gopaul et al, 2010; Wang et al, 2009, 2010; Takasu, 2013; Tay et al, 2014; etc.]. The common assumption about the measurement noises is that their

measurement variances change along with the satellite elevation angles. In this dissertation, the variance of a single GPS observable is modelled as the function of the elevation angle $\beta_{\mathrm{A}}^{\mathrm{j}}$ of the line of sight from station A to satellite j and the *a priori* receiver noise $\sigma_{90^\circ}^2$ at the local zenith [Takasu, 2013]:

$$\sigma_{\mathrm{A,j}}^2 = \sigma_{90^\circ}^2 \left[1.0 + \frac{1.0}{\sin(\beta_{\mathrm{A}}^{\mathrm{j}})^2} \right] = \sigma_{90^\circ}^2 m_{\mathrm{A,j}} \tag{5.1.8}$$

Accordingly, the covariance matrix of $\boldsymbol{n}$ double-differenced GPS observables of each type in the case of short baseline (< 10 km) shall be [Gopaul et al, 2010]:

$$\mathbf{R}_{\Delta\nabla} = \begin{bmatrix} \sigma_{\mathrm{A,j}}^2 + \sigma_{\mathrm{B,j}}^2 + \sigma_{\mathrm{A,1}}^2 + \sigma_{\mathrm{B,1}}^2 & \sigma_{\mathrm{A,j}}^2 + \sigma_{\mathrm{B,j}}^2 & \cdots & \sigma_{\mathrm{A,j}}^2 + \sigma_{\mathrm{B,j}}^2 \\ \sigma_{\mathrm{A,j}}^2 + \sigma_{\mathrm{B,j}}^2 & \sigma_{\mathrm{A,j}}^2 + \sigma_{\mathrm{B,j}}^2 + \sigma_{\mathrm{A,2}}^2 + \sigma_{\mathrm{B,2}}^2 & \cdots & \sigma_{\mathrm{A,j}}^2 + \sigma_{\mathrm{B,j}}^2 \\ \vdots & \vdots & \vdots & \vdots \\ \sigma_{\mathrm{A,j}}^2 + \sigma_{\mathrm{B,j}}^2 & \sigma_{\mathrm{A,j}}^2 + \sigma_{\mathrm{B,j}}^2 & \cdots & \sigma_{\mathrm{A,j}}^2 + \sigma_{\mathrm{B,j}}^2 + \sigma_{\mathrm{A,n}}^2 + \sigma_{\mathrm{B,n}}^2 \end{bmatrix} \tag{5.1.9}$$

where A and B are the reference station and the rover station, respectively, the satellite j is taken as the reference satellite, $\sigma_{\mathrm{A,j}}^2$ and $\sigma_{\mathrm{B,j}}^2$ are the measurement variances corresponding to the reference satellite j while $\sigma_{\mathrm{A,k}}^2$ and $\sigma_{\mathrm{B,k}}^2$ (k = 1, …, n; k ≠ j) are the measurement variances corresponding to the individually locked satellites.

After factoring out the scalar $\sigma_{90^\circ}^2$ at the local zenith, one can reformulate the matrix $\mathbf{R}_{\Delta\nabla}$ in (5.1.9):

$$\mathbf{R}_{\Delta\nabla} = \mathbf{M}\,\sigma_{90^\circ}^2 = \begin{bmatrix} m_{\mathrm{A,j}} + m_{\mathrm{B,j}} + m_{\mathrm{A,1}} + m_{\mathrm{B,1}} & m_{\mathrm{A,j}} + m_{\mathrm{B,j}} & \cdots & m_{\mathrm{A,j}} + m_{\mathrm{B,j}} \\ m_{\mathrm{A,j}} + m_{\mathrm{B,j}} & m_{\mathrm{A,j}} + m_{\mathrm{B,j}} + m_{\mathrm{A,2}} + m_{\mathrm{B,2}} & \cdots & m_{\mathrm{A,j}} + m_{\mathrm{B,j}} \\ \vdots & \vdots & \vdots & \vdots \\ m_{\mathrm{A,j}} + m_{\mathrm{B,j}} & m_{\mathrm{A,j}} + m_{\mathrm{B,j}} & \cdots & m_{\mathrm{A,j}} + m_{\mathrm{B,j}} + m_{\mathrm{A,n}} + m_{\mathrm{B,n}} \end{bmatrix} \sigma_{90^\circ}^2 \tag{5.1.10}$$

Accordingly, the covariance matrix of the observation vector $\mathbf{Z}_{\mathbf{GPS}}$ in (5.1.7) can be given as

$$\mathbf{R}_{\Delta\nabla\mathbf{GPS}} = \begin{bmatrix} \sigma_{\mathrm{c/a}}^2 \mathbf{M}_{\Delta\nabla\rho} & & \\ & \sigma_{\phi}^2 \mathbf{M}_{\Delta\nabla\phi} & \\ & & \sigma_{\dot{\rho}}^2 \mathbf{M}_{\Delta\nabla\dot{\rho}} \end{bmatrix} \tag{5.1.11}$$

where $\mathbf{M}_{\Delta\nabla\rho}$, $\mathbf{M}_{\Delta\nabla\phi}$, and $\mathbf{M}_{\Delta\nabla\dot{\rho}}$ are three matrices resulted from (5.1.10) for each type of GPS observables, and $\mathbf{s}_{\mathbf{GPS}} = \begin{bmatrix} \sigma_{\mathrm{c/a}}^2 & \sigma_{\phi}^2 & \sigma_{\dot{\rho}}^2 \end{bmatrix}^{\mathrm{T}}$ is variance components in vector form corresponding to the scalar receiver noises at the local zenith for each type of GPS observations as in (5.1.8).

5.1.2.3 Heading measurement model

In a multisensor integrated kinematic positioning and navigation system using low-cost IMUs, the heading measurement is necessary in order to suppress the fast heading drift. In the GMIKF implemented in this research, the vehicle's heading measurement is made available through the short baseline solution between two GPS rovers mounted on the vehicle's roof. The cm-level accuracy relative baseline solution is achieved through N (N >= 5) DD L1 phase measurements with their fixed integer ambiguities

$$\mathbf{Z}_{\mathrm{h}} = \psi + \varepsilon_{\mathrm{h}} \tag{5.1.12}$$

where $\mathbf{Z}_{\mathrm{h}}$ is the DGPS heading measurement, ψ is the heading state, and $\varepsilon_{\mathrm{h}} \sim \mathrm{N}\left(0,\ \sigma_{\psi}^2 \mathrm{R}_{\psi}^0\right)$ with R_{ψ}^0 being the *a priori* variance and σ_{ψ}^2 being the variance component of the heading measurements.

5.1.3 Summary

Given the general GMIKF equations at epoch *k*:

$$\mathbf{x}_k = f(\mathbf{x}_{k-1}) + \mathbf{\Lambda}(\mathbf{w}_k) \quad \text{(System model)} \tag{5.1.13}$$

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \boldsymbol{\varepsilon}_k \qquad \text{(Measurement model)} \tag{5.1.14}$$

the linearized (extended) GMIKF is derived:

$$\mathbf{x}_{\mathrm{k}} = \boldsymbol{f}(\hat{\mathbf{x}}_{\mathrm{k-1}}) + \boldsymbol{\Phi}_{\mathrm{k}}\delta\mathbf{x}_{\mathrm{k-1}} + \boldsymbol{\Lambda}_{\mathrm{k}}\mathbf{w}_{\mathrm{k}} \tag{5.1.15}$$

$$\mathbf{z}_{\mathrm{k}} = \mathbf{h}(\hat{\mathbf{x}}_{\mathrm{k/k-1}}) + \mathbf{H}_{\mathrm{k}}\delta\mathbf{x}_{\mathrm{k}} + \boldsymbol{\varepsilon}_{\mathrm{k}} \tag{5.1.16}$$

where

$$\mathbf{x}_{\mathrm{k}(27\times1)} = \left[\mathbf{r}^{\mathrm{n}}_{\mathrm{nb(k)}},\quad \mathbf{v}^{\mathbf{b}}_{\mathbf{nb}(\mathrm{k})},\quad \mathbf{a}^{\mathbf{b}}_{\mathbf{nb}(\mathrm{k})},\quad \boldsymbol{\theta}_{(\mathrm{k})},\quad \boldsymbol{\omega}^{\mathbf{b}}_{\mathbf{nb}(\mathrm{k})}\quad \mathbf{b}_{\boldsymbol{\omega}(\mathrm{k})}\quad \mathbf{b}_{\mathbf{a}(\mathrm{k})}\quad \mathbf{S}_{\boldsymbol{\omega}(\mathrm{k})}\quad \mathbf{S}_{\mathbf{a}(\mathrm{k})}\right]^{\mathrm{T}},$$

$$\mathbf{w}_{\mathrm{k}(6\times1)} = \left[\mathbf{j}^{\mathrm{n}}_{\mathrm{nb}},\quad \dot{\boldsymbol{\omega}}^{\mathbf{b}}_{\mathbf{nb}}\quad \mathbf{w}_{\mathbf{b}_{\boldsymbol{\omega}}}\quad \mathbf{w}_{\mathbf{b}_{\mathbf{a}}}\quad \mathbf{w}_{\mathbf{s}_{\boldsymbol{\omega}}}\quad \mathbf{w}_{\mathbf{s}_{\mathbf{a}}}\right]^{\mathrm{T}},\quad \mathbf{w}_{\mathrm{k}} \sim \mathrm{N}(\mathbf{0},\mathbf{Q}_{\mathrm{k}}),$$

$$\mathbf{z}_{\mathrm{k}} = \left[\mathbf{Z}_{\mathbf{IMU}}\quad \mathbf{Z}_{\mathbf{GPS}}\quad \mathbf{Z}_{\mathbf{h}}\right]^{\mathrm{T}},\ \boldsymbol{\varepsilon}_{\mathrm{k}} \sim \mathrm{N}(\mathbf{0},\mathbf{R}_{\mathrm{k}}),\quad \mathbf{R}_{\mathrm{k}} = \begin{bmatrix} \mathbf{R}_{\mathbf{IMU}} & & \\ & \mathbf{R}_{\mathbf{GPS}} & \\ & & \sigma^2_{\varepsilon_{\mathrm{h}}} \end{bmatrix},$$

$$\mathbf{H}_{\mathrm{k}} = \begin{bmatrix} \mathbf{H}_{\mathbf{IMU}} & & \\ & \mathbf{H}_{\mathbf{GPS}} & \\ & & 1.0 \end{bmatrix},\ \text{and the Jacobian matrices } \boldsymbol{\Lambda}_{\mathbf{k}} \text{ and } \boldsymbol{\Phi}_{\mathbf{k}} \text{ can be referred to 5.1.2.}$$

In summary, the GMIKF at an arbitrary epoch *k* can be translated into a form of Lease Squares estimator as follows:

$$\begin{bmatrix} \mathbf{v}_{\mathbf{z(n\times1)}} \\ \mathbf{v}_{\mathbf{w(18\times1)}} \\ \mathbf{v}_{\mathbf{x(27\times1)}} \end{bmatrix} = \begin{bmatrix} \mathbf{H}_{\mathrm{k(n\times27)}} & \mathbf{0}_{\mathbf{n\times6}} \\ \mathbf{0}_{\mathbf{6\times27}} & \mathbf{I}_{\mathbf{6\times6}} \\ \mathbf{I}_{\mathbf{27\times27}} & -\boldsymbol{\Lambda}_{\mathrm{k(27\times6)}} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{x}}_{\mathrm{k}(27\times1)} \\ \hat{\mathbf{w}}_{\mathbf{18\times1}} \end{bmatrix} - \begin{bmatrix} \mathbf{z}_{\mathrm{k}(27\times1)} \\ \mathbf{l}_{\mathbf{w(18\times1)}} \\ \boldsymbol{\Phi}_{\mathrm{k}(27\times27)}\hat{\mathbf{x}}_{\mathrm{k/k-1}(27\times1)} \end{bmatrix} \tag{5.1.17}$$

where the unknown parameter vector is $\left[\hat{\mathbf{x}}_{\mathrm{k}} \quad \hat{\mathbf{w}}\right]^{\mathrm{T}}$, $\mathbf{l}_{\mathbf{w}} = \mathbf{0}_{\mathbf{18\times1}} = [\mathbf{l}_{\mathbf{j}^{\mathrm{n}}_{\mathrm{nb}}} \quad \mathbf{l}_{\dot{\boldsymbol{\omega}}^{\mathbf{b}}_{\mathbf{nb}}} \quad \mathbf{l}_{\mathbf{b}_{\boldsymbol{\omega}}} \quad \mathbf{l}_{\mathbf{b}_{\mathbf{a}}} \quad \mathbf{l}_{\mathbf{s}_{\boldsymbol{\omega}}} \quad \mathbf{l}_{\mathbf{s}_{\mathbf{a}}}]^{\mathrm{T}}$, and the respective stochastic models of $\mathbf{v}_{\mathbf{w}}$, $\mathbf{v}_{\mathbf{z}}$, and $\mathbf{v}_{\mathbf{x}}$ are

$$\mathbf{v}_{\mathbf{w}(6\times1)} \sim N(\mathbf{0},\mathbf{Q}_{6\times6}),\ \mathbf{v}_{\mathbf{z}} \sim N(\mathbf{0},\mathbf{R}),\ \text{and}\ \mathbf{v}_{\mathbf{x}} \sim N(\mathbf{0},\boldsymbol{\Phi}_{\mathbf{k}}\mathbf{P}_{k/k-1}\boldsymbol{\Phi}^{\mathbf{T}}_{\mathbf{k}}).$$

5.2 Variance component estimation for GMIKF

5.2.1 Variance compoment vector

According to the formulation of the system model in Section 5.1, the initial values of the variance components of the process and measurement noises in GMIKF would be

$$\boldsymbol{\theta}_{\mathrm{w}} = \left[\boldsymbol{\sigma}^2_{\mathbf{j}^{\mathrm{n}}_{\mathrm{nb}}(1\times3)} \quad \boldsymbol{\sigma}^2_{\dot{\boldsymbol{\omega}}^{\mathrm{n}}_{\mathrm{nb}}(1\times3)} \quad \boldsymbol{\sigma}^2_{\mathbf{w}_{\mathbf{b}_\omega}(1\times3)} \quad \boldsymbol{\sigma}^2_{\mathbf{w}_{\mathbf{b}_\mathrm{a}}(1\times3)} \quad \boldsymbol{\sigma}^2_{\mathrm{w}_{\mathrm{s}_\omega}(1\times3)} \quad \boldsymbol{\sigma}^2_{\mathrm{w}_{\mathrm{s}_\mathrm{a}}(1\times3)}\right]^{\mathbf{T}} \tag{5.2.1}$$

$$\boldsymbol{\theta}_{\mathrm{z}} = \left[\sigma^2_{\mathrm{c/a}(1\times1)} \quad \sigma^2_{\phi_{\mathrm{L1}}(1\times1)} \quad \sigma^2_{\phi_{\mathrm{L2}}(1\times1)} \quad \sigma^2_{\dot{\rho}_{\mathrm{L1}}(1\times1)} \quad \sigma^2_{\psi(1\times1)} \quad \boldsymbol{\sigma}^2_{\mathrm{gyro}(1\times3)} \quad \boldsymbol{\sigma}^2_{\mathrm{acc}(1\times3)}\right]^{\mathbf{T}} \tag{5.2.2}$$

As an attractive alternative to the noisier raw Doppler measurements in GPS navigation, the carrier-phase-derived Doppler measurements with more stable quality are often adopted for the velocity estimation [Serrano et al, 2004]. Accordingly, if the DD Doppler measurement $\nabla\Delta\dot{\rho}_{\mathrm{AB}}$ is replaced by the time differenced L1 carrier phase from a rover GPS receiver, the estimation of the variance component corresponding to $\sigma^2_{\dot{\rho}}$ in (5.1.1) should be omitted so that the variance component vector $\boldsymbol{\theta}_{\mathbf{z}}$ is reduced to:

$$\boldsymbol{\theta}'_{\mathbf{z}} = \left[\sigma^2_{\mathrm{c/a}(1\times1)} \quad \sigma^2_{\phi_{\mathrm{L1}}(1\times1)} \quad \sigma^2_{\phi_{\mathrm{L2}}(1\times1)} \quad \sigma^2_{\psi(1\times1)} \quad \boldsymbol{\sigma}^2_{\mathrm{gyro}(1\times3)} \quad \boldsymbol{\sigma}^2_{\mathrm{acc}(1\times3)}\right]^{\mathbf{T}} \tag{5.2.3}$$

In addition, the selection of the variance components is also involved in the redundancy contribution of the measurements and the noise components in the process noise vector. In Least Squares, each redundant index r_i as in (2.7.20) reflects the degree of the influence of a raw observable on the unknown parameters. The bigger r_i is, the less a measurement affects the parameter estimation. With a group of independent measurements, the redundant index of each measurement always satisfies $0< r_i <1$. The measurement is said to be completely redundant (no influence on parameter estimation) if $r_i = 1$. On the contrary, it becomes a high leverage measurement when the associated scalar r_i tends to zero, which may be due to: a) very high accuracy of the measurement, or b) the strong geometrical or physical relation between the

measurement and to-be-estimated parameters, i.e., there exists no any other redundant measurements.

It has been found that there may exist small redundant indices that may result in the divergence of the variance component estimates [Wang et al, 2009, 2010; Gopaul et al, 2010]. For example, given the variance of unit weight σ_0^2 which satisfies $\sigma_0^2 = \sigma_{01}^2 = \sigma_{02}^2 = \cdots = \sigma_{0\mathrm{m}}^2 = \sigma_{0\mathrm{i}}^2$ in (2.7.9), the accuracy of the estimated variance components $\underset{\mathrm{m}\times 1}{\hat{\boldsymbol{\sigma}}^2}$ (as a vector) in (2.7.7) is $2\sigma_0^4\mathbf{S}^{-1}$ [Wang, 2007]. After zeroing the non-diagnal elements in matrix **S**, the accuracy of the *i*-th variance component in (2.7.8) can be approximated by $2\sigma_0^4 \mathrm{r}_\mathrm{i}^{-1}$. That is, if the redundant index of a measurement (should not be expected) or any component in the process noise vector is relatively too small, one may exclude it from the VCE process by fixing it to a reasonable value or taking the previously estimated variance as its variance in order to avoid any potential divergences [Wang et al, 2009].

According to the LS formulation of the GMIKF in (5.1.17), the measurement vector $\mathbf{l}_\mathrm{w}$ for the process noises actually consists of four independent zero vectors ($\mathbf{l}_{\mathbf{b}_\omega}, \mathbf{l}_{\mathbf{b}_\mathbf{a}}, \mathbf{l}_{\mathbf{s}_\omega}$ and $\mathbf{l}_{\mathbf{s}_\mathbf{a}}$) taking into account the IMU bias drifts and the scale factor drifts. Their redundancy contribution indices are closely related to the kinematics, the system configuration and the sensor quality and usually relatively small (< 0.001). Table 5-1 lists a set of typical values of redundancy indices of bias drifts and the scale factor drifts corresponding to the data set used in Chapter 4, in which the Crossbow 440CA IMU was used.

Table 5-1 Typical redundancy contribution indices for high leverage process noise components

Bias drift	$\mathbf{l}_{\mathbf{b}_\omega}^{\mathbf{x}}$	$\mathbf{l}_{\mathbf{b}_\omega}^{\mathbf{y}}$	$\mathbf{l}_{\mathbf{b}_\omega}^{\mathbf{z}}$	$\mathbf{l}_{\mathbf{b}_\mathbf{a}}^{\mathbf{x}}$	$\mathbf{l}_{\mathbf{b}_\mathbf{a}}^{\mathbf{y}}$	$\mathbf{l}_{\mathbf{b}_\mathbf{a}}^{\mathbf{z}}$
Ri	2.9×10^{-9}	2.9×10^{-9}	2.9×10^{-9}	1.4×10^{-6}	1.4×10^{-6}	1.4×10^{-6}

Scale factor	$\mathbf{l}_{s_\omega}^{x}$	$\mathbf{l}_{s_\omega}^{y}$	$\mathbf{l}_{s_\omega}^{z}$	$\mathbf{l}_{s_a}^{x}$	$\mathbf{l}_{s_a}^{y}$	$\mathbf{l}_{s_a}^{z}$
Ri	9.4×10^{-9}	6.1×10^{-9}	5.7×10^{-7}	1.2×10^{-7}	1.9×10^{-7}	3.0×10^{-4}

It is found out that these four types of process noise components consistently show small redundancy indices in despite of the changed kinematics in various datasets. Consequently, they are omitted in the following VCE process and the vector $\boldsymbol{\theta}'_{\mathbf{w}}$ to be estimated in the tightly-coupled GMIKF is reduced into be a 6×1 vector as follows:

$$\boldsymbol{\theta}'_{\mathbf{w}} = \left[\boldsymbol{\sigma}^2_{\mathbf{j}^{n}_{nb}(1\times3)} \quad \boldsymbol{\sigma}^2_{\boldsymbol{\omega}^{n}_{nb}(1\times3)}\right]^{\mathbf{T}} \tag{5.2.4}$$

5.2.2 Iterative VCE process

Because the simplified Helmert method in Least Squares runs under an assumption that all being-estimated variance components of unit weight are assumed to be equal as in Eq. (2.7.9), the variance component estimates will always be biased unless that assumption holds true. Here arises the necessity of an iterative estimation process. In pratice, the estimated variance components shall converge to the unbiased values after a number of iterations. Likewise, the VCE for GMIKF is also performed in an iterative manner. It is also worth mentioning that the rough initial variance values for the VCE process should always be avoided in consideration of the estimation divergence.

The proposed iterative VCE process (Figure 5-1) starts from two initial diagonal variance matrices $\mathbf{Q}_{(0)}$ and $\mathbf{R}_{(0)}$, which are respectively associated with the process and measurement noise vectors. However, these two matrices ($\mathbf{Q}_{(i)}$ and $\mathbf{R}_{(i)}$) in the beginning of the other subsequent i-th iterations shall be adjusted according to the results from previous iteration.

As the second step, given the measurement residual vectors $\mathbf{v}_{j_i}$ for j-th group of measurements at epoch *i* computed by (2.7.14)- (2.7.16), the epoch-wise results of $\mathbf{r}_{j_i}$ (redundant contribution indices) and $\mathbf{v}_{j_i}^{T}\mathbf{D}_{\mathbf{L}_j}^{-1}\mathbf{v}_{j_i}$ (the weighted sum of measurement and process noise residuals squared for each component) are obtained and recorded throughout the whole iteration process. Then for each epoch, the accumulated (global) variance component estimates of each independent component in $\mathbf{v}_{\mathbf{z}}$ and $\mathbf{v}_{\mathbf{w}}$ up to epoch *k* can be computed as:

$$\sigma_{0j}^{2}(k\,|\,1...k)=\frac{\sum_{i=1}^{k}\mathbf{v}_{j_i}^{T}\mathbf{D}_{\mathbf{L}_j}^{-1}\mathbf{v}_{j_i}}{\sum_{i=1}^{k}\mathbf{r}_{j_i}} \tag{5.2.3}$$

$$(j=j_{nbx}^{n},j_{nby}^{n},j_{nbz}^{n},\omega_{nbx}^{n},\omega_{nby}^{n},\omega_{nbz}^{n},s_{c/a},s_{\phi_{L1}},s_{\phi_{L2}},s_{\psi},s_{gyrox},s_{gyroy},s_{gyroz},s_{accx},s_{accy},s_{accz})$$

where $\mathbf{D}_{L_j}$ is the covariance matrix of the j-th group of measurements.

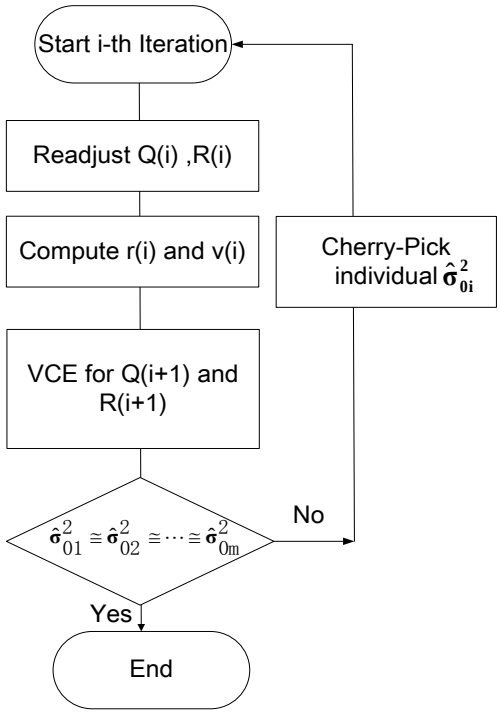


Figure 5-1 Flow chart of VCE in GMIKF [Qian et al, 2016]

5.2.3 Statistic tests and error distribution evaluation

Statistic test against any outlier using system innovations or measurement residuals is a prerequisite for an appropriate VCE process in Kalman Filter because the potential undetected outliers might distort the distribution of these statistics and mislead the estimates of the variance components. By conducting the statistic test in Kalman filter, one can identify the possible outliers under the assumption that the system and measurement models are correct [Wang, 2008]. Stöhr [1986] studied the test statistics in Kalman filter through system innovations after the normal distribution and χ^2- distribution. In addition, the test statistics after t- distribution and *F*-distribution based on system innovations can be found in [Salzmann, 1993]. Wang [1997, 2008] further extended the test statistics to the measurement residual vector and the process noise residual vector, and systematically constructed the global, regional and local test statistics in Kalman filter.

In order to assure the statistical correctness of state solution and variance component estimates, multiple statistics tests are performed for both GMIKF and subsequent VCE work shown in Figure 5-1. The statistics test process may consists of two parts: the outlier test using the system innovations in GMIKF and the distribution test using the measurement residuals in VCE estimation.

5.2.3.1 Outlier detection

According to Wang [1997, 2008], the outlier test in GMIKF at epoch *k* can be organized as follows:

Test 1: The local χ^2-test for each type of the measurements using the system innovation vector with the GNSS observables in the *i*-th group which share the

same stochastic characteristics (e.g., C/A code, carrier phase, Doppler velocity) at epoch *k*. With $\mathbf{d}(\mathrm{k})_{\mathbf{i}}$ being the system innovation vector of the *i*-th group GPS observables. The Null hypothesis is $\mathrm{H}_{\mathbf{0}}\mathbf{:}\ \mathbf{d}(\mathrm{k})_{\mathbf{i}} = \mathbf{0}$ and the alternative is $\mathrm{H}_{\mathbf{A}}\mathbf{:}\ \mathbf{d}(\mathrm{k})_{\mathbf{i}} \neq \mathbf{0}$. The test statistics runs

$$\chi^2_{\mathbf{d}(\mathrm{k})_{\mathrm{i}}} = \mathbf{d}(\mathrm{k})_{\mathrm{i}}^{\mathrm{T}}\mathbf{D}_{\mathrm{d}}^{-1}\mathbf{d}(\mathrm{k})_{\mathrm{i}} \sim \chi^2_{\alpha}(\mathrm{r}(\mathrm{k})_{\mathrm{i}})$$

where α is the specified significant level for Type I Errors, r is the degrees of freedom, and $\mathbf{D}_{\mathbf{d}}$ is the variance matrix of $\mathbf{d}(\mathrm{k})_{\mathrm{i}}$.

Test 2: Each GPS observable in the *i*-th group can also be tested after the normal distribution for potential outlier. The test statistics for normal test runs

$$\mathrm{z}_{\mathrm{ij}} = \mathrm{d}(\mathrm{k})_{\mathrm{ij}} / \sigma_{\mathrm{d}(\mathrm{k})_{\mathrm{ij}}} \sim \mathrm{N}_{\alpha}(0,1)$$

where the significant level is α, and $\mathrm{d}(\mathrm{k})_{\mathrm{ij}}$ is the *j*-th system innovation in *i*-th group.

Test 3: Similarly, the local z-tests can also be performed for each single observable of the non-GNSS measurements, e.g., IMU observables, heading observables.

5.2.3.2 Residual stationary test

The goal of the distribution test against the measurement residual vector is to check if the residuals are a wide-sense stationary random process because variance homogeneity is the premise of the variance component estimation. According to Wang [1997, 2008], a regional *F*-test to examine the variance homogeneity between two groups of epochs, e.g., epochs ($1 \cdots k$-s) and epochs (k-s+$1 \cdots k$) is given by

$$F_{\mathrm{g}_1/\mathrm{g}_2} = \hat{\sigma}^2_{\mathrm{g}_1} / \hat{\sigma}^2_{\mathrm{g}_2} \sim F_{\alpha}\left(\sum \mathrm{r}\big|_{\mathrm{g}_1}, \sum \mathrm{r}\big|_{\mathrm{g}_2}\right)$$

where $\hat{\sigma}_{g_i}^2 = \left(\sum \mathrm{v}^{\mathrm{T}}\mathrm{Pv}\right)\Big|_{g_i} / \left(\sum \mathrm{r}\right)\Big|_{g_i}$ (i=1, 2), $F_\alpha(f_1, f_2)$ is the critical value of the Fisher distribution at the specified significant level α with the 1st degree of freedom f_1 being the numerator and the 2nd degree of freedom f_2 being the denominator under the assumption of the estimated value of $F_{g1/g2} \geq 1$. Otherwise, one can simply swap the numerator and the denominator.

5.3 Numerical Experiment

To prove the success of the proposed VCE algorithm in GMIKF, the results of the variance component estimation targeting on the tightly-coupled GMIKF are presented in this section with the same dataset as in Section 4.3, where the reduced variance component vectors to be estimated are $\boldsymbol{\theta}'_{\mathrm{w}}$ and $\boldsymbol{\theta}'_{\mathrm{z}}$ as discussed in Section 5.2.1.

$$\boldsymbol{\theta}'_{\mathrm{w}} = \left[\boldsymbol{\sigma}^2_{\mathbf{j}^{\mathrm{n}}_{\mathrm{nb}}(1\times3)} \quad \boldsymbol{\sigma}^2_{\dot{\boldsymbol{\omega}}^{\mathrm{n}}_{\mathrm{nb}}(1\times3)}\right]^{\mathbf{T}}$$

$$\boldsymbol{\theta}'_{\mathrm{z}} = \left[\sigma^2_{\mathrm{c/a}(1\times1)} \quad \sigma^2_{\phi_{\mathrm{L1}}(1\times1)} \quad \sigma^2_{\phi_{\mathrm{L2}}(1\times1)} \quad \sigma^2_{\psi(1\times1)} \quad \boldsymbol{\sigma}^2_{\mathrm{gyro}(1\times3)} \quad \boldsymbol{\sigma}^2_{\mathrm{acc}(1\times3)}\right]^{\mathrm{T}}$$

In the beginning of the variance tuning process in Kalman filter, the initial values of the being estimated variance components are usually set with a group of conservative values to guarantee that the Kalman filter can accommodate all sorts of measurements. The initial standard deviations corresponding to the noise factors in the process noise vector in GMIKF are assigned as the maximum jerks [Punzo, 2011] and angular accelerations of a land vehicle driven on the flat road with asphalt pavement (Table 5-2), while the measurement noises of the IMU and GPS receiver are empirical values derived from our experiments and observations (Table 5-3).

Table 5-2 Initial standard deviations of process noises in tightly-coupled GMIKF

Jerk (m/s^3)	Angular acceleration $°/s^2$

$\boldsymbol{\sigma}^{x}_{\dot{\mathbf{j}}^{n}_{nb}}$	$\boldsymbol{\sigma}^{y}_{\dot{\mathbf{j}}^{n}_{nb}}$	$\boldsymbol{\sigma}^{z}_{\dot{\mathbf{j}}^{n}_{nb}}$	$\boldsymbol{\sigma}^{x}_{\dot{\boldsymbol{\omega}}^{n}_{nb}}$	$\boldsymbol{\sigma}^{y}_{\dot{\boldsymbol{\omega}}^{n}_{nb}}$	$\boldsymbol{\sigma}^{z}_{\dot{\boldsymbol{\omega}}^{n}_{nb}}$
72	72	72	300	300	300

Table 5-3 Initial standard deviations of measurement noises in tightly-coupled GMIKF

$\boldsymbol{\sigma}^{x}_{gyro}$	$\boldsymbol{\sigma}^{y}_{gyro}$	$\boldsymbol{\sigma}^{z}_{gyro}$	$\boldsymbol{\sigma}^{x}_{acc}$	$\boldsymbol{\sigma}^{y}_{acc}$	$\boldsymbol{\sigma}^{z}_{acc}$	$\sigma_{c/a}$	$\sigma_{\phi_{L1}}$	$\sigma_{\phi_{L2}}$	σ_{ψ}
0.5°/s	0.5°/s	0.5°/s	0.5m/s^2	0.5m/s^2	0.5m/s^2	0.5m	0.02m	0.02m	0.25°

During the iterative VCE process, in consideration of those high leverage measurements due to either the high accuracy or the strong geometry effect, the variance component estimates after the *i*-th round iteration will be cherry-picked as the initial values for the (*i*+1)-th iteration, where the sifting rule is simply to disregard those variance estimates with small redundancy contribution indices (***r***). That is to say, given a small ***r*** (< 0.1) for the *j*-th group of measurements, the corresponding variance component estimate σ_j^2 in the previous *i*-th round will not be used in the next (*i*+1)-th iteration. For example, if the redundancy contribution index of the angular rate measurements from the X axial gyroscope is less than 0.1 in the 1st iteration, the standard deviation will remain to be 0.5°/s in the next 2nd iteration.

5.3.1 Process noise variance components

Figure 5-2 and Figure 5-3 show the iterative estimated variances corresponding to the process noise vectors: the jerk vector ($\dot{\mathbf{j}}^{b}_{nb}$) and the angular acceleration vector ($\dot{\boldsymbol{\omega}}^{b}_{nb}$) in the body frame. As can be seen, their estimates converge at the 4th iteration.

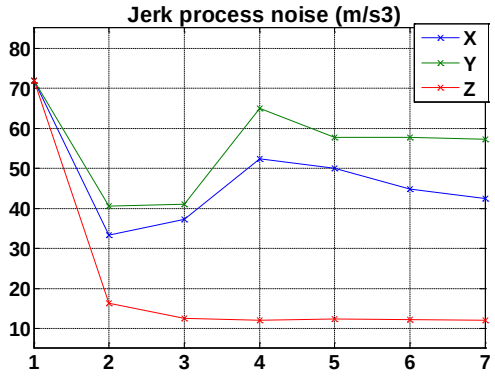


Figure 5-2 Iterative VCE results for the three jerk components

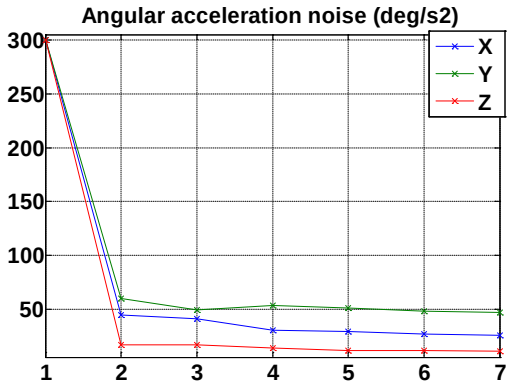


Figure 5-3 Iterative VCE results for the angular accelerations

Particularly, it should be noted that the variance estimates for the jerk vector ($\mathbf{j}_{nb}^{b}$) and the angular acceleration vector ($\dot{\boldsymbol{\omega}}_{nb}^{b}$) achieved from this specific dataset may not be universal to represent the generic land vehicle's dynamics because the vehicle was driven on a fairly flat terrain along the course with straight lines and three sharp turns.

5.3.2 Variance components of IMU measurements

This section presents the estimated variance components for the IMU measurements. In addition, the numerical comparison with the manufacturer specification and laboratory results from Allan variance method is also shown in the end.

5.3.2.1 The estimated variance components

Figure 5-4 and Figure 5-5 show the estimated standard deviations (1σ) of IMU measurement noises. The final estimated standard deviations (1σ) are: 0.18 ^{0}/s, 0.25^0/s, and 0.06^0/s for three gyroscopes, and 0.15 m/s^3, 0.1 m/s^3 and 0.09 m/s^3 for accelerometers as in Table 5-5 and Table 5-6.

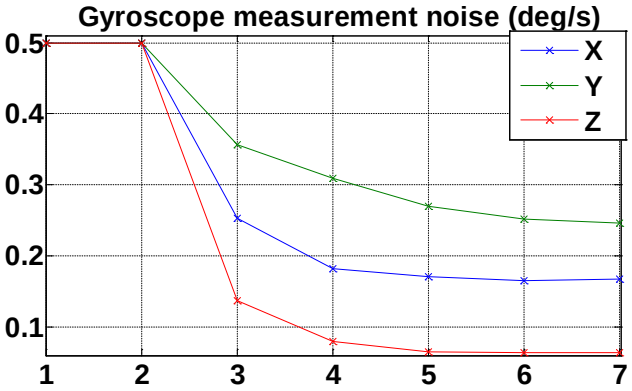


Figure 5-4 Iterative VCE results for gyro measurements

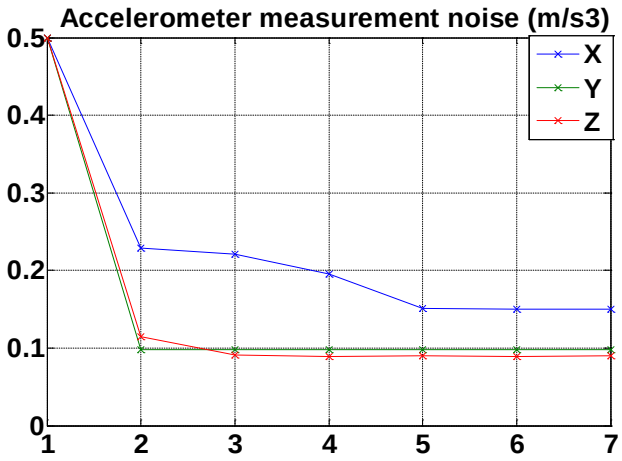


Figure 5-5 Iterative VCE results for the accelerometer measurements

In Figure 5-4, it can be seen that the estimation of the variance components in the 1st iteration are skipped for three high leverage gyroscope measurements incurred from the relative large

initial angular acceleration process noise (300 $^0/s^2$). Similarly, the variance component for the Y axis accelerometer is fixed after the 2nd iteration as well (Figure 5-5).

5.3.2.2 Manufacturer Specification

Table 5-4 IMU440CA technical specification from the manufacturer

	Random Walk (σ_{rw})	Bias Stability (σ_{BS})	Quantization Noise Std. (σ_q)	Equivalent Noise Std.
Gyroscope	4.5 ($^0/\sqrt{hour}$)	10(^{0}/hour)	0.02(^{0}/ sec)	0.7503 (^{0}/s)
Accelerometer	1.0 (m/s/$\sqrt{hour}$)	1.0 (mg)	0.5 (mg)	0.16 (m/s^2)

The technical specification for IMU440CA from the manufacturer is collected as shown in Table 5-4 which identified the standard deviation (1σ) of three major zero-mean error sources in IMU measurements: random walk, bias instability and quantization noises. Because GMIKF consolidates above three independent noises as one white noise component, the standard deviation (1σ) of the equivalent white noise is computed as:

$$\sigma = \sqrt{\sigma_{rw}^2 + \sigma_{bs}^2 + \sigma_q^2}$$

where $\sigma_{rw} = \sigma_{allan-rw} / \sqrt{\delta t}$ is modeled as a white noise, $\sigma_{bs} = \sigma_{allan-bs} \sqrt{\frac{\delta t}{T}}$ is modeled as a random walk sequence, δt = 0.01 (s) is the sampling period of the IMU device, and T is the correlation time at which the bias stability measurement was made (~100 seconds for IMU440CA).

5.3.2.3 Laboratory Allan variance results

The Allan variance (AVAR) proposed by D.W.Allan [1966] is originally for studying the frequency stability of precision oscillators. AVAR is a method of describing the root mean square

(RMS) of a random process as a function of average time. Because of the close similarities, AVAR has been adapted to random-drift characterization of inertial sensors.

This section presents the Allan variance results for the noises in IMU measurements based on a laboratory static dataset lasting 18 hours collected in Dec, 2012 for the same IMU used in the road test.

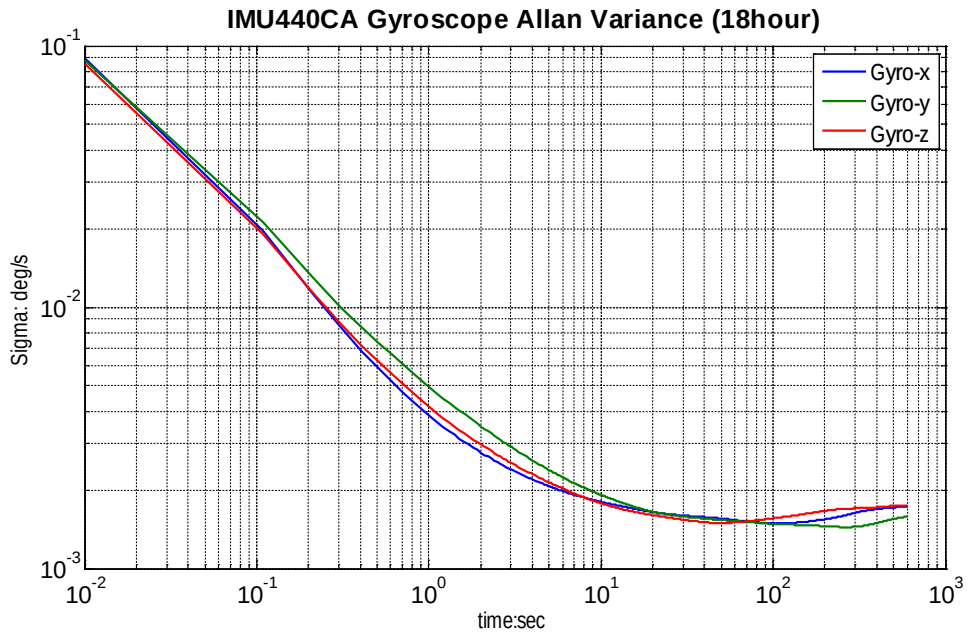


Figure 5-6 IMU440CA Allance Variance results (Gyros)

Table 5-5 IMU440CA Gyroscope measurement noise from Allan Variance

	X-right	Y-forward	Z-up
Random Walk ($^0/\sqrt{\text{hour}}$)	0.196	0.263	0.227
Bias Stability (^{0}/hour)	2.88	2.21	1.14
Quantization Noise Std.(^{0}/ sec)	0.0003	0.00025	0.00027
Equivalent White Noise Std. (^{0}/ sec)	0.0435	0.0438	0.0378

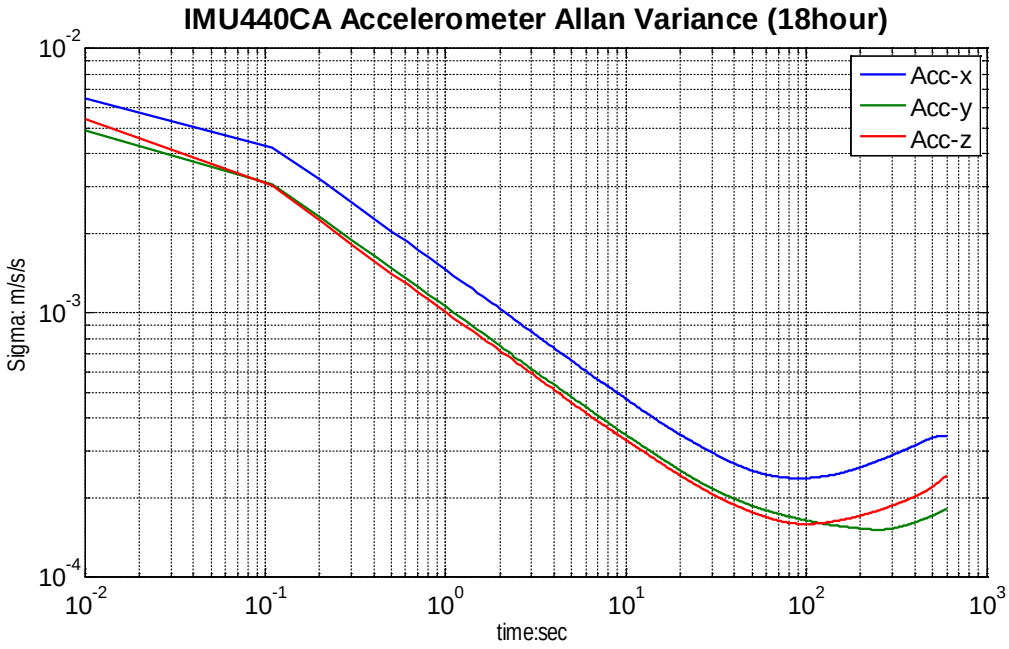


Figure 5-7 IMU440CA Allance Variance results (Acceleromters)

Table 5-6 IMU440CA Accelerometer measurement noise from Allan Variance

	X-right	Y-forward	Z-up
Velocity Random Walk ($m/s/\sqrt{hour}$)	0.1025	0.0660	0.0600
Bias Stability (m/s^2)	0.0036	0.0023	0.0020
Equivalent White Noise Std.(m/s^2)	0.017	0.011	0.010

5.3.2.4 Comparison

Table 5-7 and 5-8 compare the standard deviations ($1\ \sigma$) of the IMU measurements resulted from three independent methods. In consideration of the extremely violent dynamics IMU might experience, the specifications from manufacturer are understandably more conservative (larger) than those from laboratory Allan variance tests which are often too optimistic for the real kinematic applications because an Allan variance is assembled at a single temperature whereas the manufacturer's specification applies to an operational temperature range.

In practice, vehicle's vibration is known to produce unwanted output for MEMS devices [Yoon, 2009], and therefore, is detrimental to the accuracy of the MEMS IMU's measurements and INS system [Wendel, 2001; De Pasquale, 2010; Vasispacher et al, 2015]. In both tables, the distinctions between the results from the Allan variance method and the VCE for kinematic dataset show the negative impacts of vibration on the performance of the IMU's measurements.

In table 5-7, Z axial gyroscope is much more resistant against the vibration than X and Y axial gyroscopes because the noises modelled by VCE algorithm actually contain both the vibration and the measurement noises and the vehicle's vibration in vertical direction is known to be more severe than those in longitudinal and lateral directions [Stein, 2011; Chonhenchob, 2012].

Table 5-7 Evaluation of IMU Gyroscope measurement noises

Gyroscope 1 σ (^{0}/ sec)	X-right	Y-forward	Z-up
Manufacturer Specification=$\sqrt{(\sigma^2_{rw}+\sigma^2_{BS})}$	0.7503	0.7503	0.7503
Allan Variance =$\sqrt{(\sigma^2_{rw}+\sigma^2_{BS}+\sigma^2_{q})}$	0.0435	0.0438	0.0378
VCE results	0.18	0.25	0.06

Table 5-8 Evaluation of IMU Acclerometer measurement noises

Acclerometer 1 σ (m/s^2)	X-right	Y-forward	Z-up
Manufacturer Specification=$\sqrt{(\sigma^2_{rw}+\sigma^2_{BS})}$	0.16	0.16	0.16
Allan Variance=$\sqrt{(\sigma^2_{rw}+\sigma^2_{BS}+\sigma^2_{q})}$	0.017	0.011	0.010
VCE results	0.15	0.1	0.09

In Table 5-8, the noise components resulted from Laboratory Allan variance method is superior to the VCE results upon the kinematic dataset because the vibration is the target input that the accelerometers are designed to measure [Yoon, 2009].

In a Least Squares system described in Section 2.7.1, the residual vector $\mathbf{v}$ shall statistically follow the normal distribution $\mathbf{v}_i \sim N(\mathbf{0}, \mathbf{D}_{\mathbf{v}_i})$. Hence, the histogram of the standardized residuals shall be able to verify the stochastic model of the observables. Similarly, given a pair of true **Q** and **R**, the histogram of the standardized measurement residuals in Kalman filter should also agree with a standardized normal distributed curve. Conversely, the best choice among a number of pairs of **Q** and **R** can be reasonably affirmed by the best graphic agreement between the histogram of the standardized measurement residuals and the standardized normal distribution curve.

Section 5.3.5 carries out the histogram comparison for three pairs of covariance matrices (**Q** and **R**) named as "before VCE", "Allan" and "after VCE", which correspond to three rows in Table 5-7 and Table 5-8. Nevertheless, the $\mathbf{R}_{\mathbf{IMU}}$ in the group "before VCE" is actually assigned with the $\mathbf{R}_{\mathbf{IMU}}$ used at 1st iteration of the VCE process instead of the values from Manufacturer's specification. The best pair of **Q** and **R** is then concluded through the visual comparison of the match between histogram and the standardized normal distribution curve.

Yet again, as mentioned in Section 5.3.1, the results from this dataset definitely do not possess universal meaning to describe the performance changes of the MEMS IMU caused by the variation of the vehicle's vibration which might become our future research topic.

5.3.3 Variance components of GPS measurements

In order to avoid the unstable quality of the raw Doppler velocity measurements, the time differenced L1 carrier phases were substituted for the double differenced Doppler measurements in GMIKF. Accordingly, the variance of the Doppler velocity measurements in GMIKF is directly related to L1 carrier phase measurement's variance. As a result, the variance component estimation of the Doppler velocity measurements $\sigma^2_{\dot{\rho}(1\times1)}$ is skipped for this specific dataset.

Figure 5-8 and Figure 5-9 give the iterative variance estimates for the double differenced GPS measurement including carrier phase and C/A code measurement in L1 band.

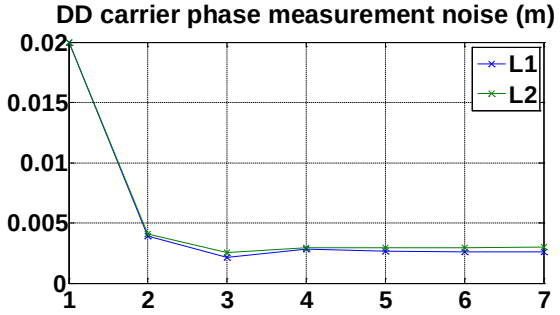


Figure 5-8 Iterative VCE results for the DD GPS L1 carrier phase measurements

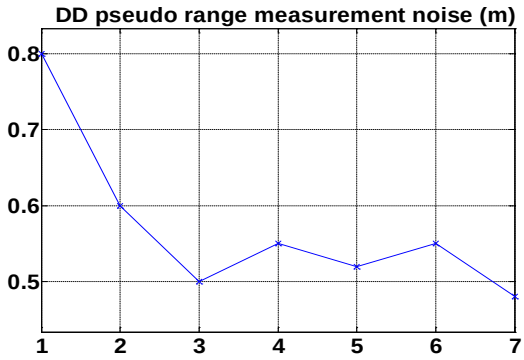


Figure 5-9 Iterative VCE results for the DD GPS C/A code measurements

5.3.4 Variance component of DGPS heading measurements

Because the DGPS heading measurements were derived from the relative baseline solutions using double differenced L1 carrier phase observables between two GPS rovers rigidly tied on the vehicle roof, the variance of the DGPS heading measurements ($\mathbf{Z}_h$ in 5.1.12) should have been directly derived from the stochastic model of those raw L1 carrier phase observables. However, in the GMIKF, the variance of the DGPS heading is quantified by a constant numerical value because of its stable accuracy assured by the fact that the DGPS heading measurement is only made available to GMIKF when at least 5 DD L1 carrier phase measurements reached the

fixed ambiguities. Correspondingly, a single scalar factor is appropriate to represent the variance component of the DGPS heading measurement in GMIKF.

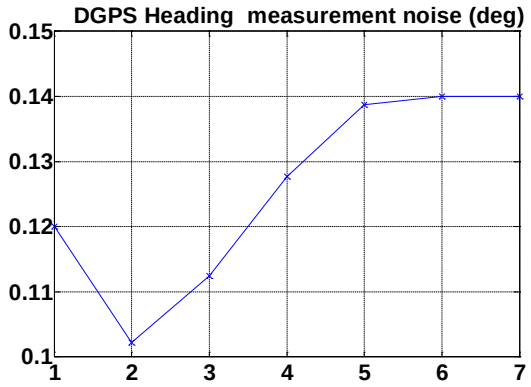


Figure 5-10 Iterative VCE results for heading measurement noise

5.3.5 Histograms of measurement residuals

Statistically, the residual vector $\mathbf{v}$ shall follow the normal distribution $\mathbf{v}_i \sim N(\mathbf{0}, \mathbf{D}_{\mathbf{v}_i})$. In other words, the histogram of the standardized residuals can reveal the correctness of the measurement model and the *a priori* stochastic model of the measurements if the number of the residual samples is large enough. Therefore, in this section, the goal to identify the best pair of covariance matrices (**Q** and **R**) among three candidate pairs named as "before VCE", "Allan" and "after VCE" is fulfilled through comparison of the residual histograms.

However, because the over-optimistic $\mathbf{R}_{IMU}$ of a MEMS IMU derived from the Allan variance method leads to a diverge solution due to many normal IMU measurements being rejected in GMIKF, the histograms of the standardized measurement residuals resulted from two groups ("Before VCE" and "After VCE") are only compared in the following plots.

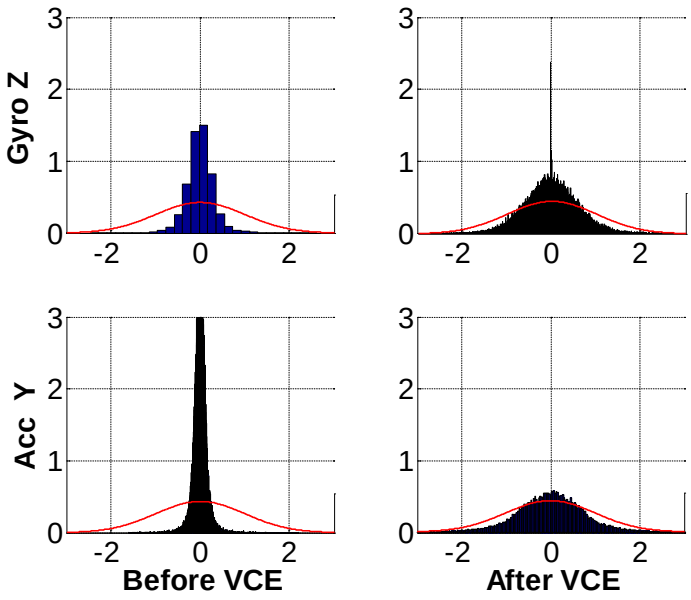


Figure 5-11 Histograms of standardized residuals for Z gyro and Y accelerometer

The positive changes on standardized measurement residuals brought by VCE work can be noticed in Figure 5-11: the histograms from the group "After VCE" conform to the normal distribution much better than those from the group "Before VCE". Essentially, the abnormal residual histogram for group "Before VCE" stems from the unrealistically large values of $\mathbf{R}_{IMU}$ compared to the amplitudes of measurement residuals.

In Figure 5-12, regarding four DD GPS carrier phase measurements (L1 band), the standardized measurement residual histograms from group "After VCE" also fit the standardized normal distribution curves much better than that from group "Before VCE", which tells that the estimated GPS carrier phase measurement variances are more accurate (realistic) than the *a priori* (Before VCE) counterparts.

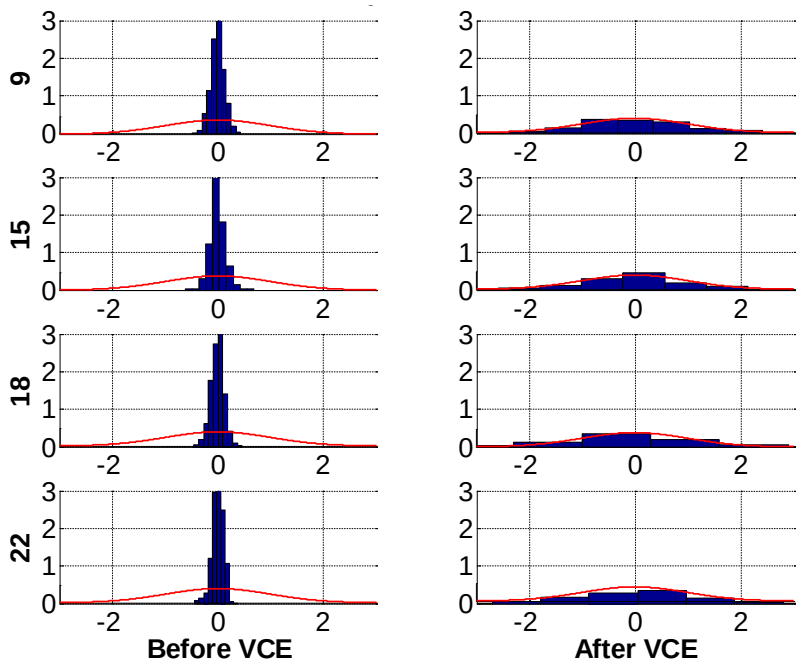


Figure 5-12 Standardized residual histograms for GPS DD L1 phase measurements

In summary, the VCE for GMIKF achieves a better stochastic model for both IMU and GPS carrier phase measurements. With a second dataset, a full set of histogram comparison for all involving measurements including virtual zero process noise measurements is presented in Chapter 6 to show the positive effects of VCE work on tuning of variances of jerk and angular acceleration

5.4 Summary

This section proposed and realized a practical VCE algorithm for multisensor integrated kinematic positioning and navigation. First, the sensor data were processed by using the extended Kalman filter with the GMI strategy (GMIKF) as shown in Chapter 4. Then, the random errors in individual measurements could be statistically separated from the other error sources. This has made possible the simultaneous estimation of the variance components in the process noise vector and measurement noise vector for all sensors participating in GMIEKF. Second, the simplified

VCE method accumulating the redundancy contribution of individual independent measurements or measurement groups was formulated for the individual components in the process noise vector (e.g., the jerks and the angular accelerations) and the measurement vector (e.g., the IMU and the double-differenced GPS measurements). The success of the VCE algorithm is proved through the more normal-distributed histograms of the standardized measurement residuals.

6. Data post processing software, road tests and results

This chapter overviews the post-processing software package developed in C/C++ for the purpose of the post-processing of YUMIS' data including GNSS and IMU data. A complete set of the solution including position, velocity, attitude and other IMU error states resulted from a selected test is presented to show the success of the software package. At last, the results for the iterative VCE tuning of process and measurement noise component in GMIKF are also detailed.

6.1 Overview of data post-processing software

The data post-processing software developed in this research (GMI post processor) is programmed in C/C++ under Microsoft Visual Studio IDE environment. In order to boost computation speed, the executable program is linked to two external high performance matrix libraries: Eigen matrix library and Intel® Math Kernel Library (MKL). Six major modules in the suite are: data importer, GPS only processors, initialization module, static alignment module, GMI Kalman filter and VCE processor.

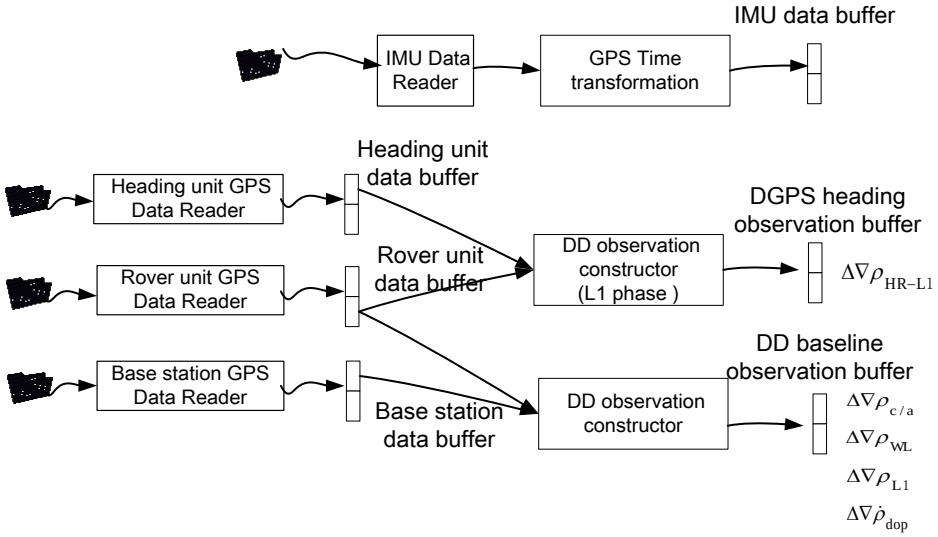


Figure 6-1 Data importer module

In the GMI post processor, the data importer (Figure 6-1) sequentially reads all sensor data from the binary files to the data buffers in memory up to certain processing time (t). Importer's reading pace is controlled by an internal scheduler which advances the processing time (t) when all sensor data up to time (t) have been consumed and processed. In parallel with GMI Kalman filter, GPS only processor module (Figure 6-2) runs five Least Squares based submodules to provide the rover SPP solution ($\mathbf{r}_{c/a}$), the DGPS heading solution (ψ), the DD C/A baseline solution ($\Delta\nabla\mathbf{r}_{c/a}$), the DD Widelane phase (DD-WL) baseline solution ($\Delta\nabla\mathbf{r}_{WL}$), the fixed ambiguities for DD-WL measurements ($\Delta\nabla\mathbf{N}_{WL}$), the DD L1 phase (DD-L1) baseline solution ($\Delta\nabla\mathbf{r}_{L1}$) and the fixed ambiguity for DD-L1 measurements ($\Delta\nabla\mathbf{N}_{L1}$).

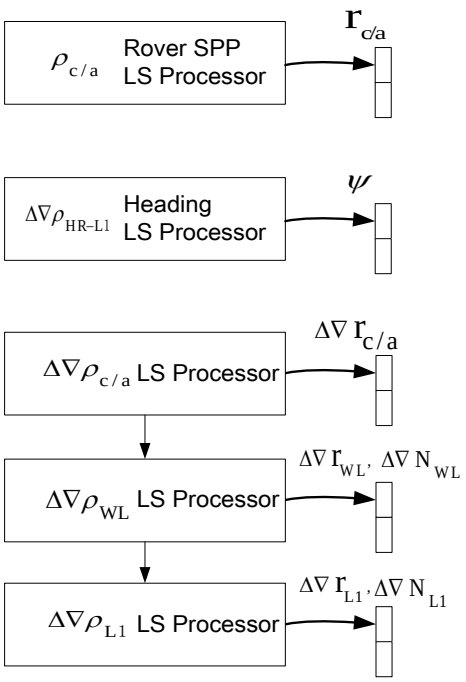


Figure 6-2 GPS only processor module

Among five modules, the WL LS processor uses DD-WL ($\Delta\nabla\rho_{WL}$) measurements and derives the baseline solution $\Delta\nabla\mathbf{r}_{WL}$ and the associated integer ambiguity $\Delta\nabla\mathbf{N}_{WL}$ with the help of the sub-meter accuracy DD C/A baseline solution $\Delta\nabla\mathbf{r}_{c/a}$. Similarly, the L1 LS processor uses

the DD-L1 ($\Delta\nabla\rho_{L1}$) measurements and solves a more accurate baseline solution $\Delta\nabla\mathbf{r}_{L1}$ and the associated integer ambiguity $\Delta\nabla\mathbf{N}_{L1}$.

In the beginning of GMI processor (Figure 6-3), the initialization module and the static alignment module (Section 4.3.2) prepare the initial state vector of GMIKF including position, velocity and attitude. Once the GMIKF state vector is successfully initialized, these two modules are bypassed in the rest of data process.

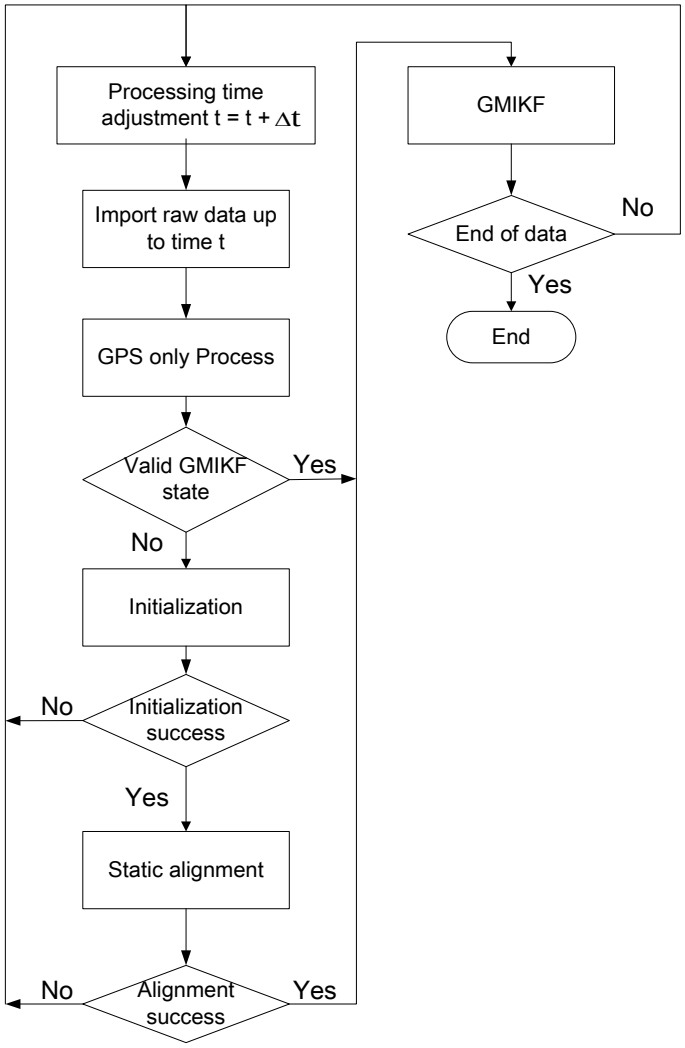


Figure 6-3 Flowchart of GMI processor

Beside the state preidction(1st step) and the measurement update (3rd step), two extra steps in the GMIKF (Figure 6-4) are necessary for the cm-level position accuracy with GPS carrier phase measurements, which are "cycle slip detection and repair" (2nd step) and "integer ambiguity determination" (4th step).

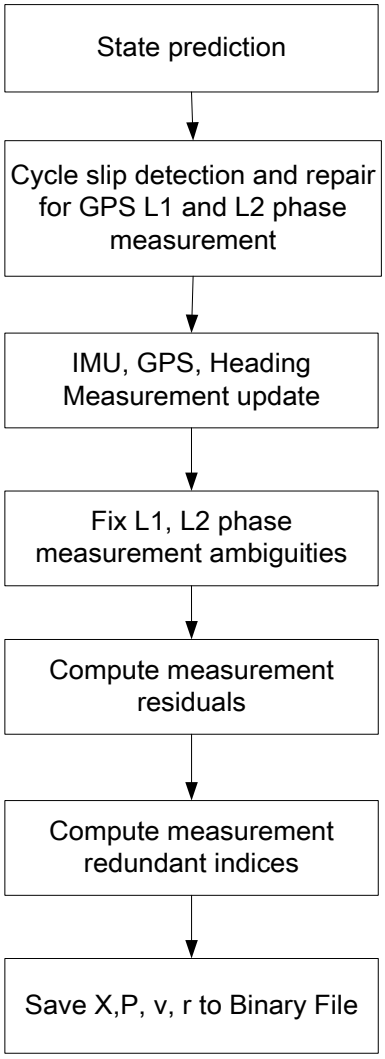


Figure 6-4 Flowchar of GMIKF module

The "cycle slip detection and repair" authenticates the continuity of integer ambiguities through multiple techniques using the raw Doppler measurements [Chang, 2008], the DD geometry free dual fequency observation combination [Bisnath, 2000], the delta position based on the time differenced L1 carrier phase measurement [Liu, 2009], the delta position based on the

predicted position state and the cycle slip status bit output from the GPS OEM receiver [NovAtel OEM4 Manual]. The "integer ambiguity determination" fixes the dual frequency carrier phase ambiguities using LAMBDA method [Teunissen, 1995] with the help of the fixed ambiguities ($\Delta\nabla\mathbf{N}_{WL}$ and $\Delta\nabla\mathbf{N}_{L1}$) from GPS only processors.

6.2 The selected test dataset and its results

Among tens of datasets collected using the YUMIS system introduced in Chapter 3, we take one of them to elaborate the whole process of the GMI post processor. The selected data set (2015-11-22) was collected on Nov 22, 2015, which involved one Crossbow IMU440CA (MEMS) IMU and two GNSS rover receivers mounted on the vehicle roof.

6.2.1 Trajectory and system configuration

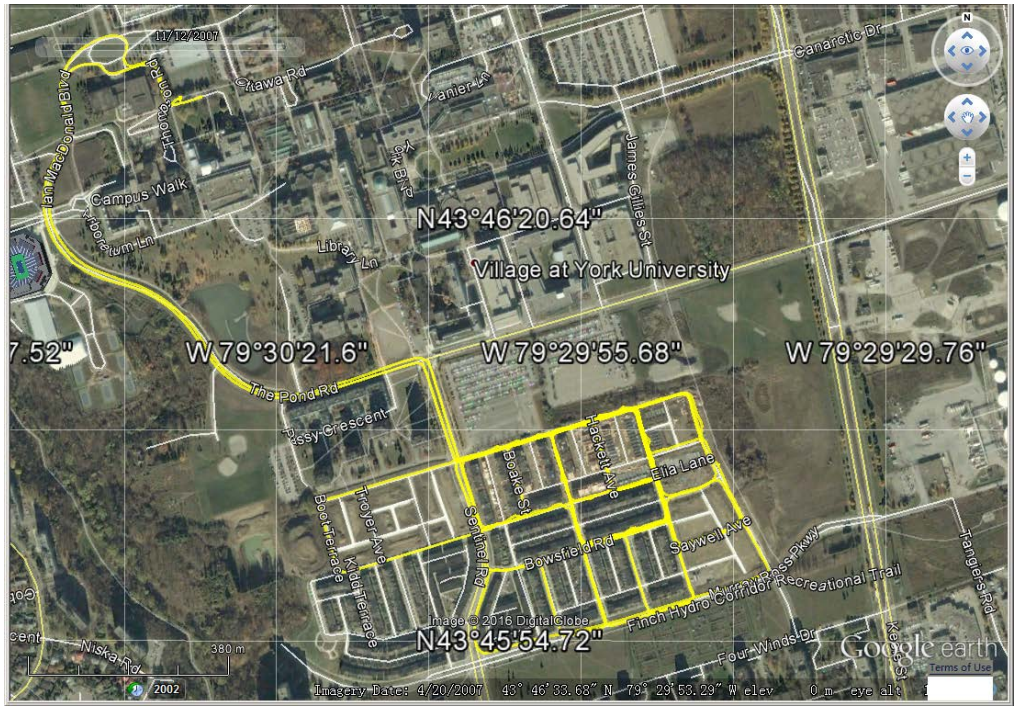


Figure 6-5 Top view of the trajectory

The vehicle's trajectory is around one residential area close to York University's Keele campus, which is highlighted in solid yellow line (Figure 6-5). Because of the speed limitation, the vehicle's max speed is around 10 m/s (30 km/h) as seen in Figure 6-6.

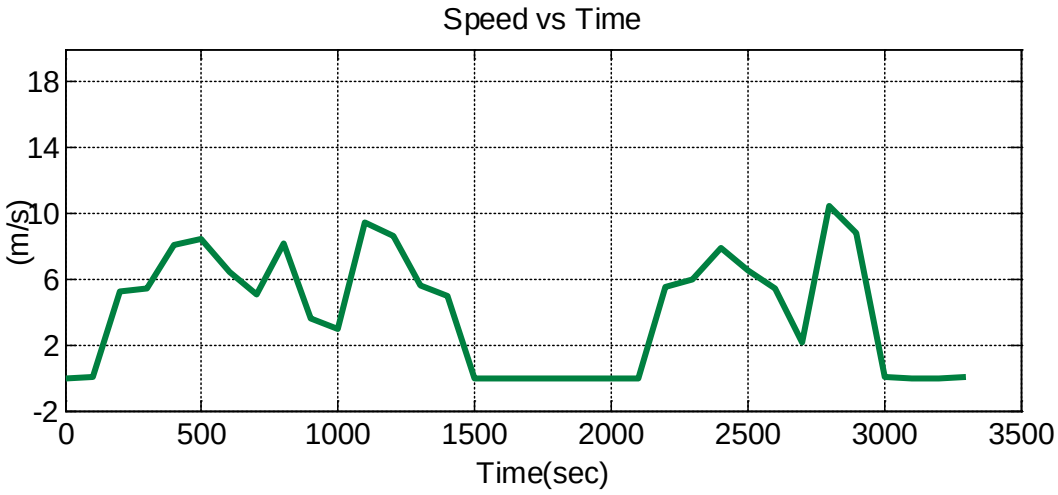


Figure 6-6 Speed profile of the vehicle

In terms of hardware setting, the system configuration with the 2015-11-22 data set was the same as the one presented in Section 4.3 except the GNSS receivers. The raw IMU data were collected at 100 Hz data rate while the primary rove receiver was replaced with the NovATel OEM6 GNSS receiver through the Ethernet port. The lever arm parameters of the GNSS receivers with respect to the IMU unit were manually measured at the accuracy of 0.5cm. Simultaneously, one another GNSS base station near the start point (<10 km) was set up with the 5 Hz data rate throughout the whole road test. Table 6-1 summarizes the GNSS receivers' hardware configuration.

Table 6-1 GNSS receivers' hardware configuration

No.	Receiver Type	Raw Rate	Data Rate in KF	Data Type	
1	NovAtel OEM6	100 Hz	5 Hz	C/A, L1, L2, D1	Rover
2	NovAtel OEM4	5 Hz	5 Hz	C/A, L1	Heading

3	NovAtel OEM4	5 Hz	5 Hz	C/A, L1, L2, D1	Base

6.2.2 GPS Satellite availability

With the third party GNSS post processer tool (RTKLib), the sky plot of all available GPS satellites during the mission is plotted (Figure 6-7). In addition, Figure 6-8 shows the GPS satellite availability with the 5 degree cut-off elevation angle.

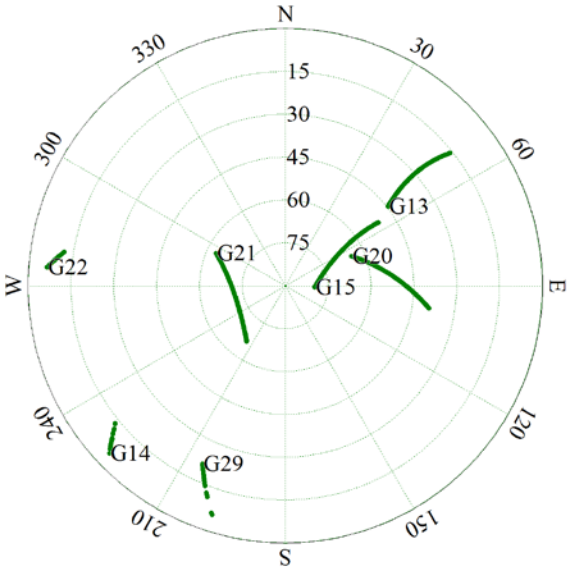


Figure 6-7 GPS Satellite sky plot during mission

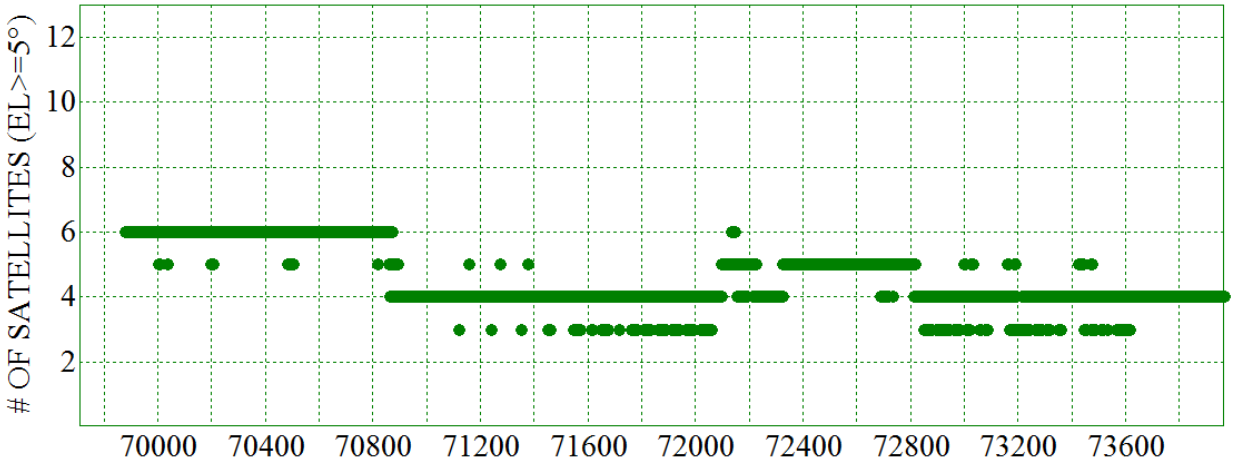


Figure 6-8 Number of GPS Satellites in YUMIS's view

Although maximum seven different GPS satellites were observed, only four of them (SVPRN 13, 15, 20, 21) were persistently available (Figure 6-10) because the half of skyview was frequently blocked by the houses. The incurred sharp change of geometry DOPs is depicted in Figure 6-9.

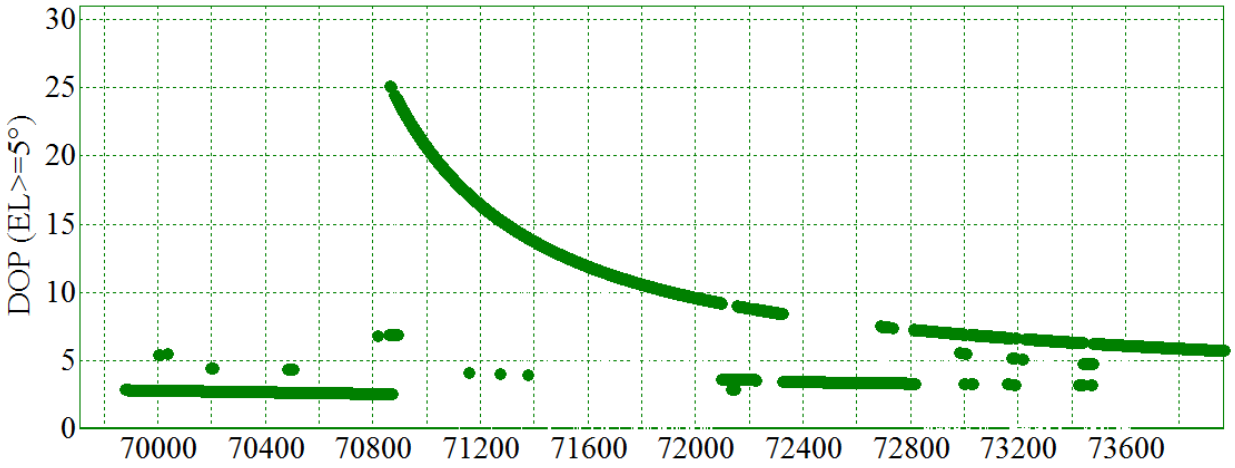


Figure 6-9 GDOP of GPS Satellites in YUMIS's view

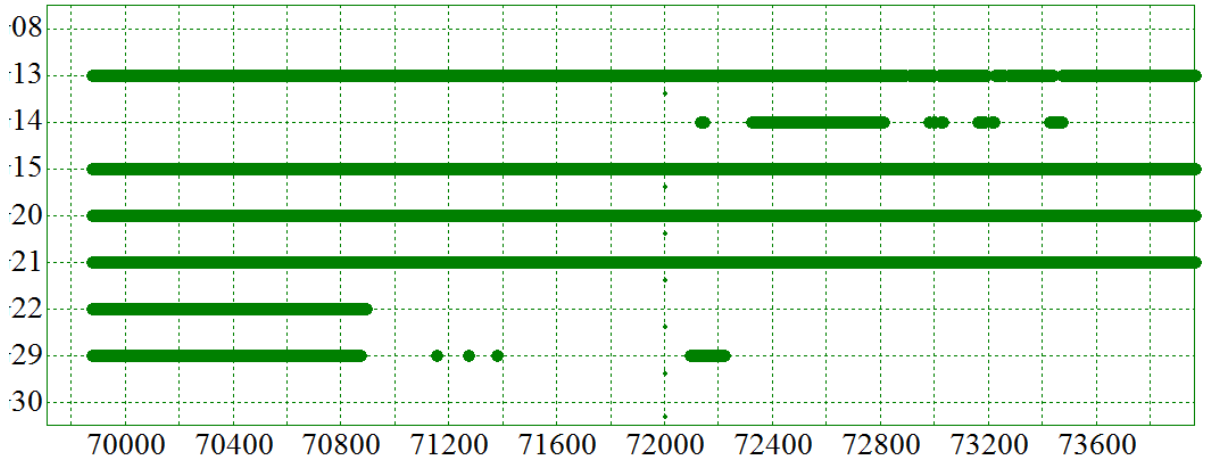


Figure 6-10 GPS Satellites (carrier phase) visibility in YUMIS's view

6.2.3 Solution of 27 state GMIKF

The diagonal covariance matrices of GMI Kalman filter for the process noise vectors and the measurement vector were initially constructed according to the values in Table 6-2.

Table 6-2 Initial standard deviations of process noises and measurement noises

$\boldsymbol{\sigma}_{\dot{\mathbf{j}}_{nb}^{n}}$ -x	72 (m/s3)	$\boldsymbol{\sigma}_{\dot{\boldsymbol{\omega}}_{nb}^{n}}$ -x	300 (°/s2)
$\boldsymbol{\sigma}_{\dot{\mathbf{j}}_{nb}^{n}}$ -y	72(m/s3)	$\boldsymbol{\sigma}_{\dot{\boldsymbol{\omega}}_{nb}^{n}}$ -y	300 (°/s2)
$\boldsymbol{\sigma}_{\dot{\mathbf{j}}_{nb}^{n}}$ -z	72(m/s3)	$\boldsymbol{\sigma}_{\dot{\boldsymbol{\omega}}_{nb}^{n}}$ -z	300 (°/s2)
GNSS C/A	0.5 (m)	GNSS L1 phase	0.02 (m)
GNSS Doppler	0.1 (m/s)	GNSS L2 phase	0.02 (m)
Gyroscope- x	0.75 (°/s)	Accelerometer –x	0.16 (m/s3)
Gyroscope- y	0.75 (°/s)	Accelerometer –y	0.16 (m/s3)
Gyroscope- z	0.75 (°/s)	Accelerometer –z	0.16 (m/s3)
DGPS heading	0.5 (°)		

6.2.3.1 Position, velocity and acceleration

Figure 6-11 and Figure 6-12 present the 1σ accuracy of position, velocity solution. The position accuracy in up component was the worst while the accuracies in east and north components stably maintained at a few cm level during the most of the road test (Figure 6-11). The velocity accuracy in Figure 6-12 shows the similar pattern as in Figure 6-11 where the accuracy in up component is the worst compared to the other two components.

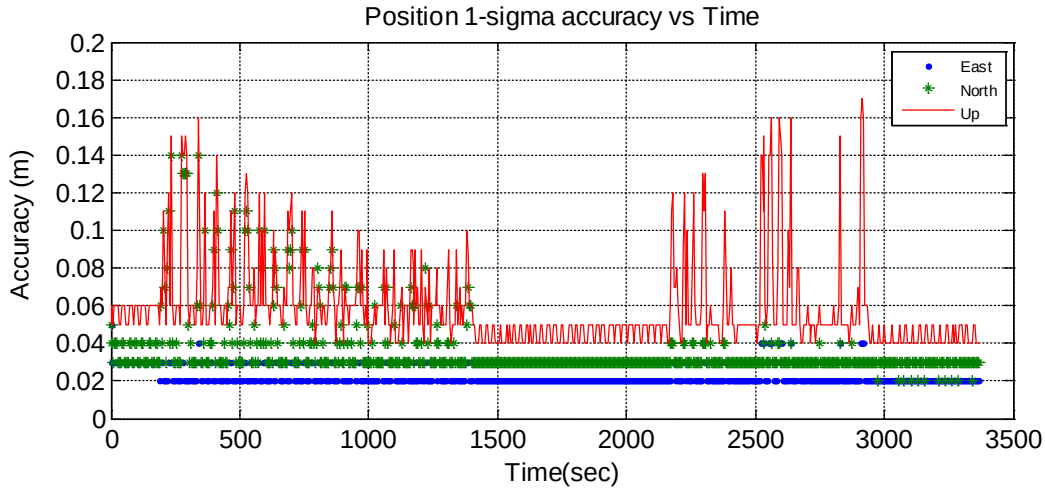


Figure 6-11 Position solution 1σ accuracy

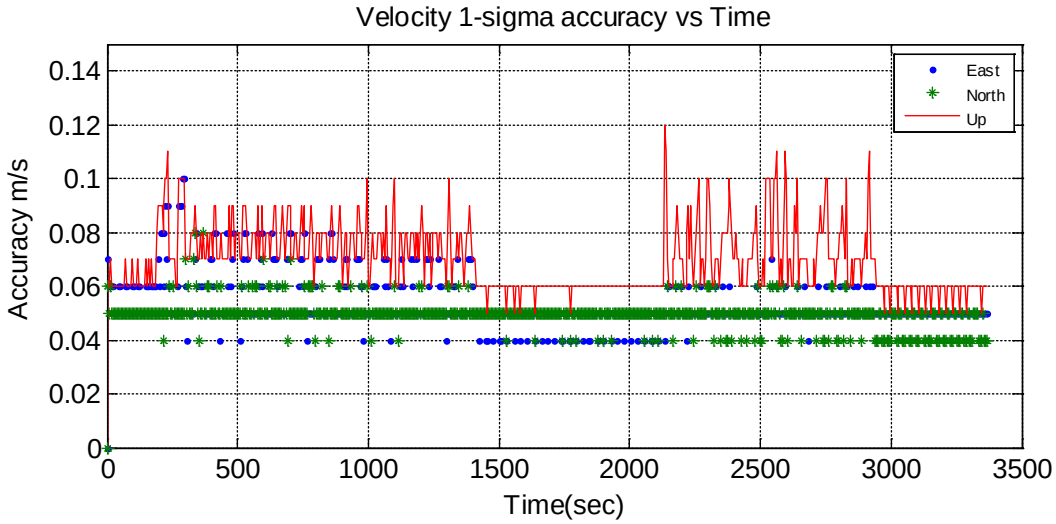


Figure 6-12 Velocity solution 1σ accuracy in ENU frame

Figure 6-13 shows the body acceleration solution with $\pm 3\sigma$ boundaries. The fluctuations of the accelerations in right and forward axes are obviously stronger than the up axis because of the land vehicle's dynamics resides in horizontal direction.

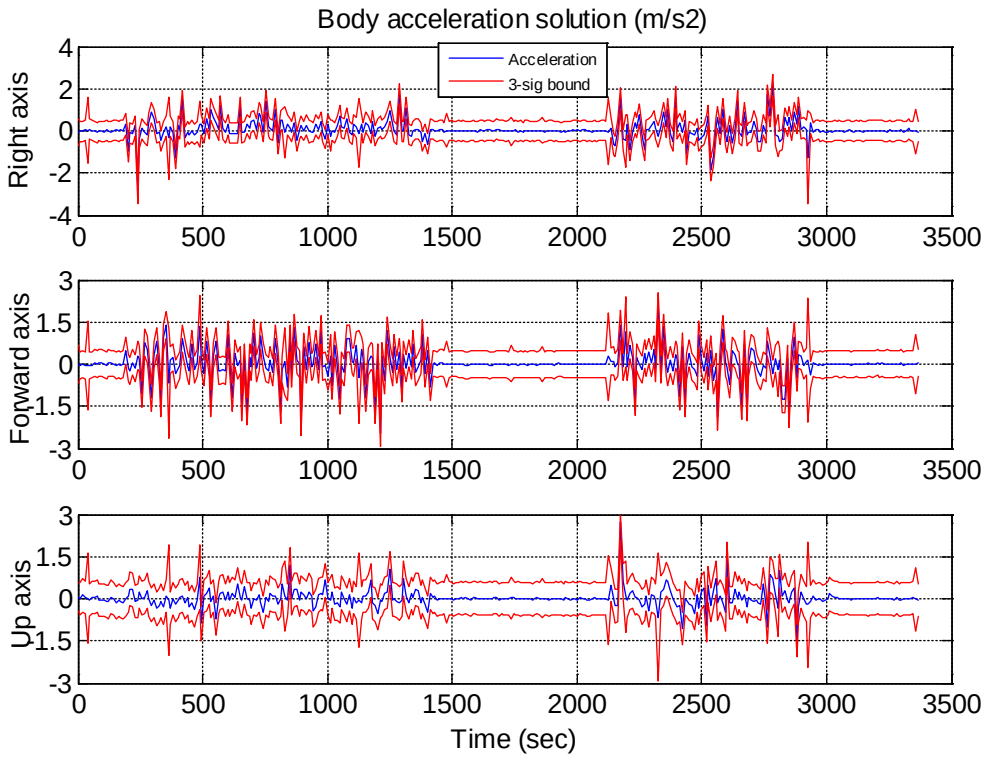


Figure 6-13 Acceleration solution in body frame

In order to verify the solution of GMIKF, one reference positioning solution is computed using the third party GNSS navigation software (RTKLIB) which also reaches the cm level positioning accuracy with the same dataset. Two positioning solutions are compared in ECEF frame and their differences are plotted in Figure 6-14 along with the RTKLIB solution's standard deviation (1σ). Through Figure 6-14, the GMIKF's success is proved since the differences are well within the 1σ boundaries of the RTKLIB solution.

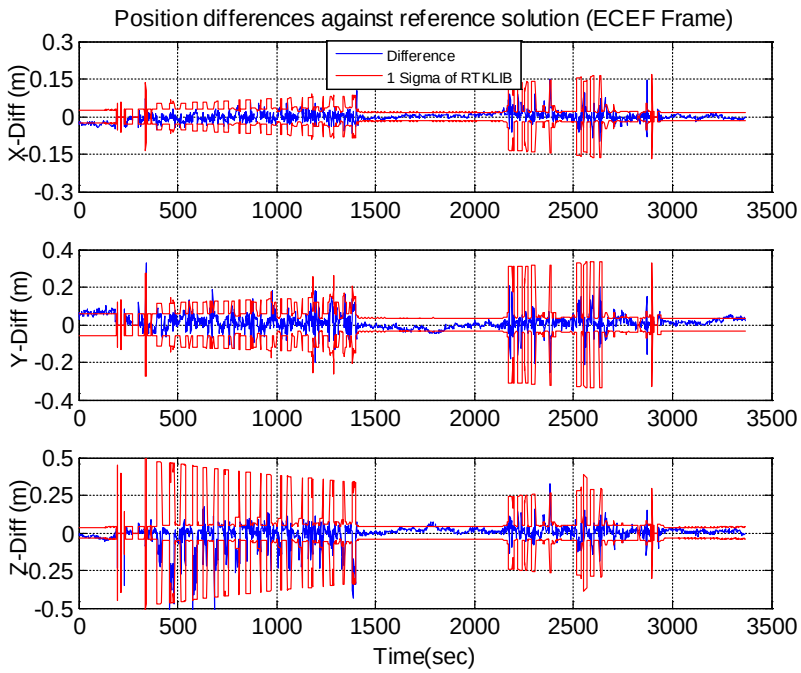


Figure 6-14 Position solution difference against RTKLIB reference solution

6.2.3.2 Attitude and angular rates

Figure 6-15 shows the solutions of the pitch and roll angles with their 3σ accuracy bounds. The accuracies of the pitch and roll angles are well maintained within 0.5 degree except during three static periods, in which the growing accuracy bounds are seen because: a) the weak obervabilities of pitch and roll angles during static motions; and b) no special handling in GMIKF to adapt the stationary status (e.g. apply the smaller variances of process noises, or enforce any zero velocity update etc.).

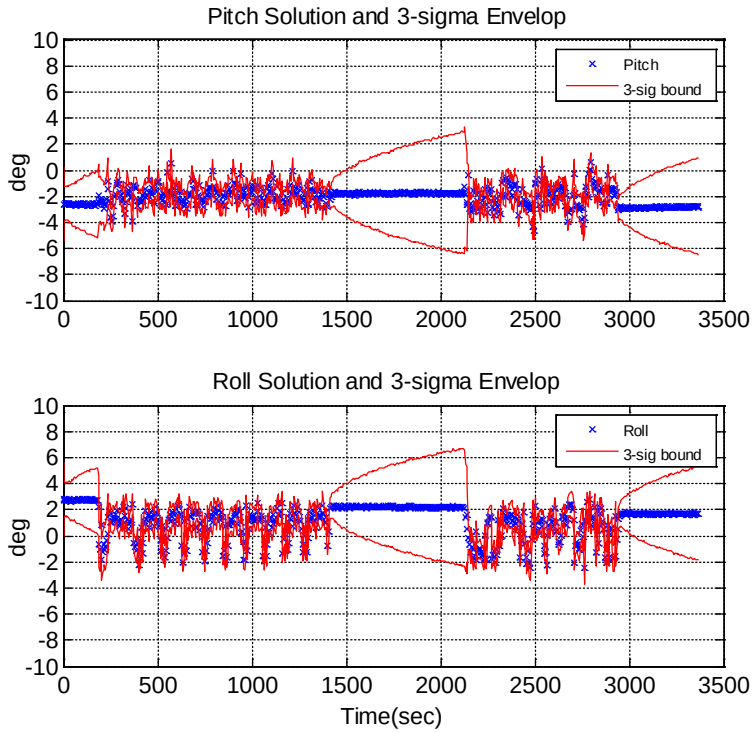


Figure 6-15 Pitch and roll solution and their 3σ accuracy bounds

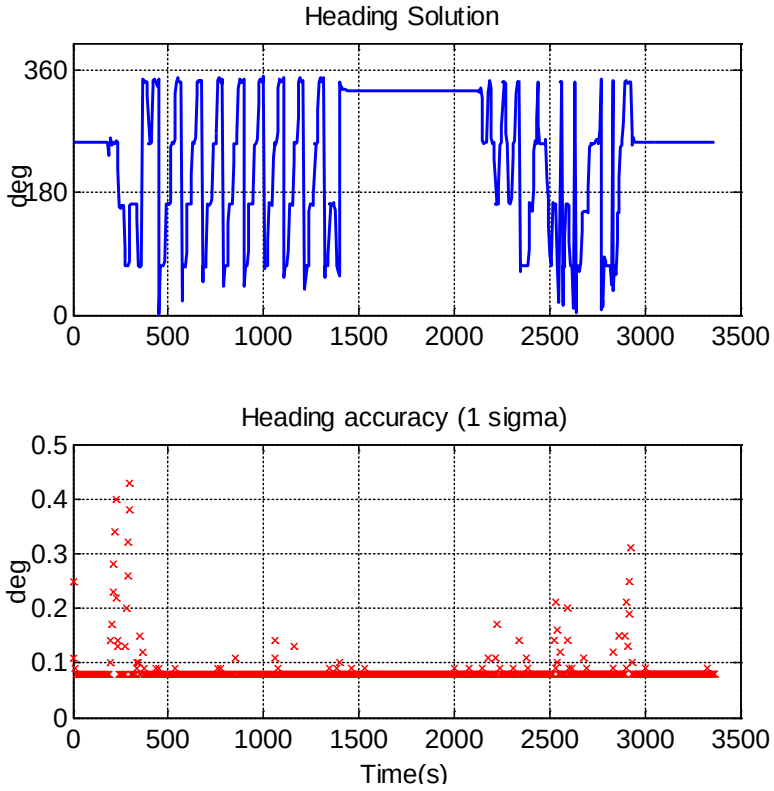


Figure 6-16 Heading solution (top) and its 1σ accuracy (bottom)

Figure 6-16 presents the heading solution (top) along with its 1σ standard deviation (bottom). Due to the contribution of the external DGPS heading measurements, the heading solution maintained throughout the whole road test at good accuracy except at a few epochs when DGPS heading measurements were not available.

Furthermore, the angular rate solutions is shown in Figure 6-17. The angular rates on both the right and the forward axes show the stable estimates close to zero. In contrast, because of the sharp turnings of the land vehicle, the Up axis presented much stronger signals than the other two axes.

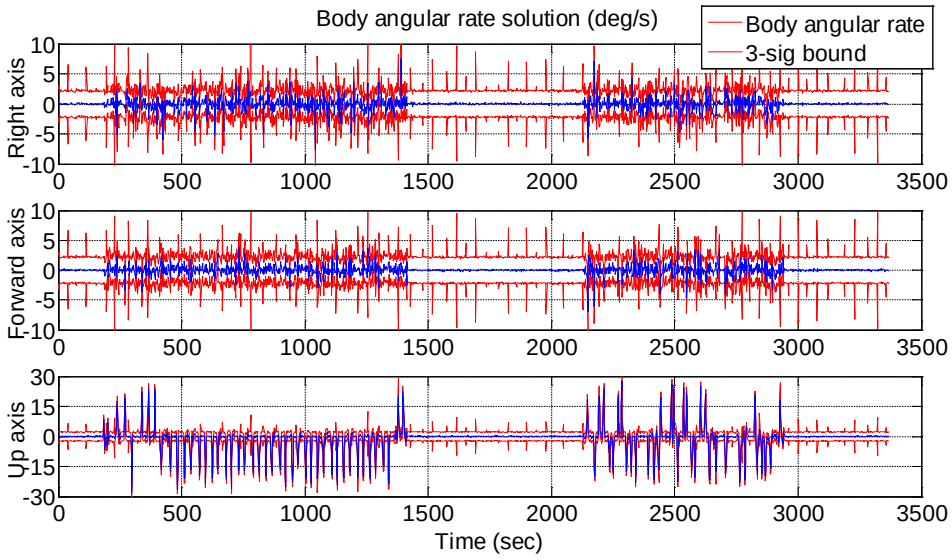


Figure 6-17 Angular rate solution in body frame

6.2.3.3 IMU biases solution

Figure 6-18 and Figure 6-19 present the estimated biases for three gyroscopes and three accelerometers, respectively. In both figures, the estimated biases are stabilized. According to the manufacture's specifications for IMU440CA, the range of gyroscopes' biases shall remain within

± 0.75 °/s while the ones associated with the accelerometers shall be within ± 0.15 mg, which are well satisfied with the bias estimates in GMIKF.

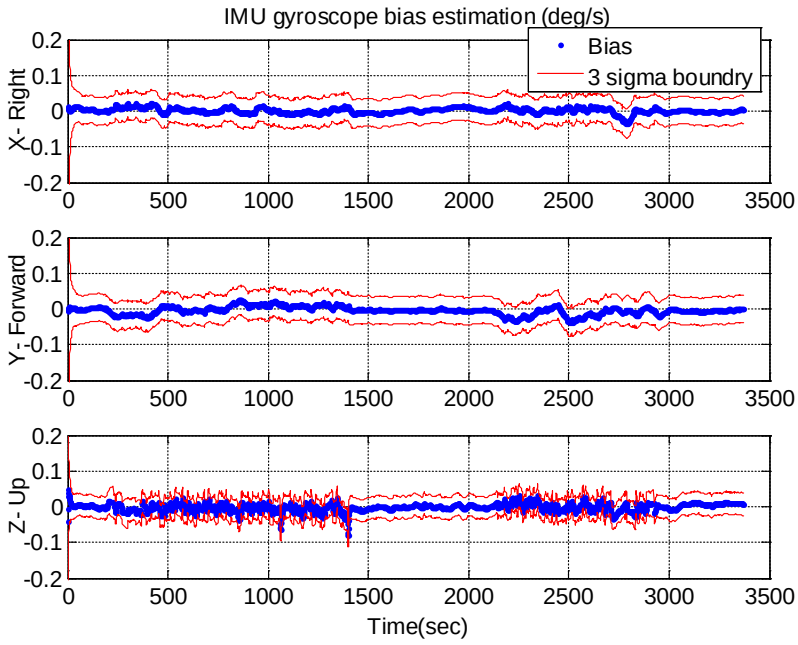


Figure 6-18 Gyroscope biases in three axes with their 3σ bounds

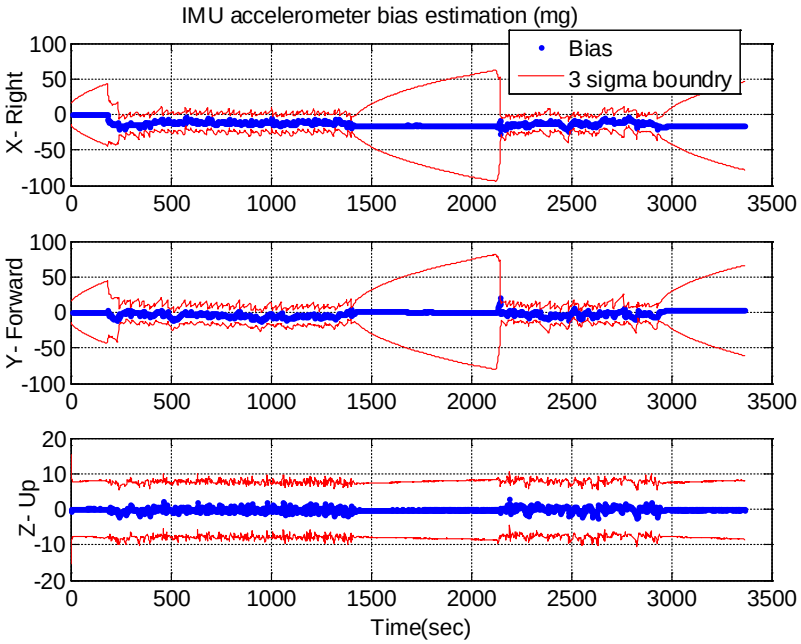


Figure 6-19 Accelerometer biases in three axes with their 3σ bounds

6.2.3.4 IMU scale factors solution

Figure 6-20 and Figure 6-21 present the results of six scale factor drifts regarding three gyroscopes and three accelerometers. Their stable estimates in both figures demonstrates the good quality of the IMU440CA and the success of the GMIKF. When the vehicle is stationary, these drifts nearly stay the same. But their standard deviations are increased because the observabilities of scale factor drifts are closely related to the vehicle's motion.

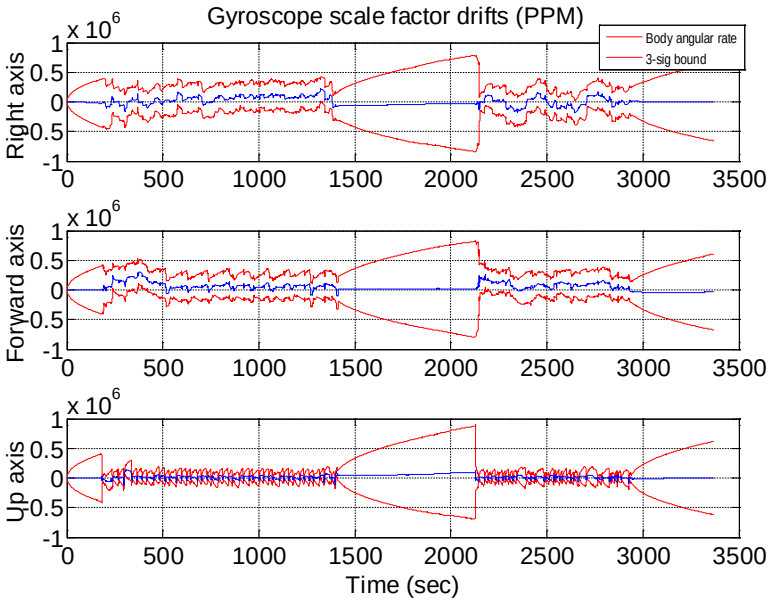


Figure 6-20 Scale factor drifts in three gyroscope axes

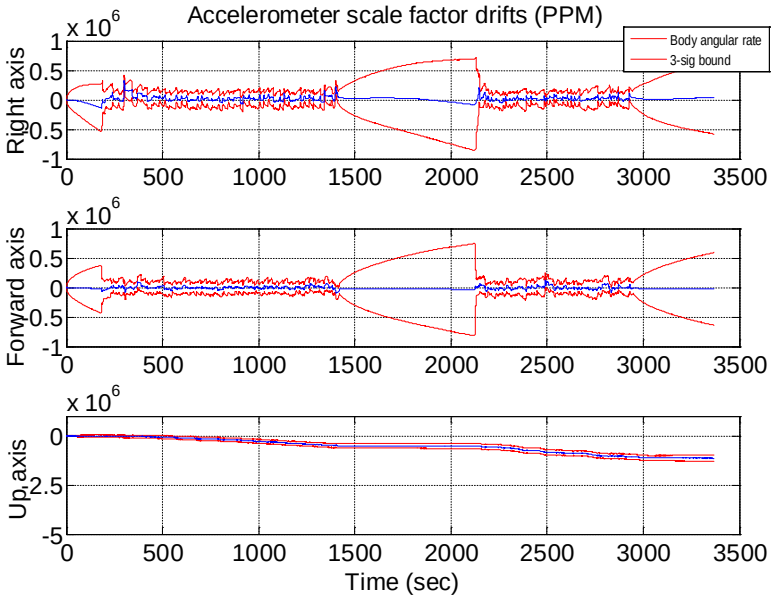


Figure 6-21 Scale factor drifts in three accelerometer axes

6.2.3.5 Measurement residuals in GMIKF

Although the popular analysis of the system innovation series in Kalman filter can validate the success of the Kalman filter, the measurement residuals and the residuals associated with the process noise components were used in this section because the author wants to separately check the correctness of system model, measurement model and stochastic models of the process and measurement noises in Kalman filter since one failure in any of three models will cause misshaped residuals.

In this section, the innovations and residuals corresponding to the virtual zero process noise measurements and the raw measurements from sensor are plotted to verify the success of the GMIKF. The residuals for two types of process noise measurements (jerk and angular accelerations) are plotted in Figure 6-22 and Figure 6-23. As well, the innovation and residuals for raw sensor measurements are shown in Figure 24 – Figure 30 including seven types of

measurements: C/A code, L1 phase, L2 phase, Doppler (L1 band), gyroscope's angular rates, accelerometer's specific force and DPGS heading.

Residuals of process noise components as the zero mean virtual measurements

The formulas to calculate the residuals associated with each of the components in the process noise vector can be found in Section 2.7.2 and Section 5.1.1 after Wang [1997], to which the traditional error analysis in the integrated navigation would not access.

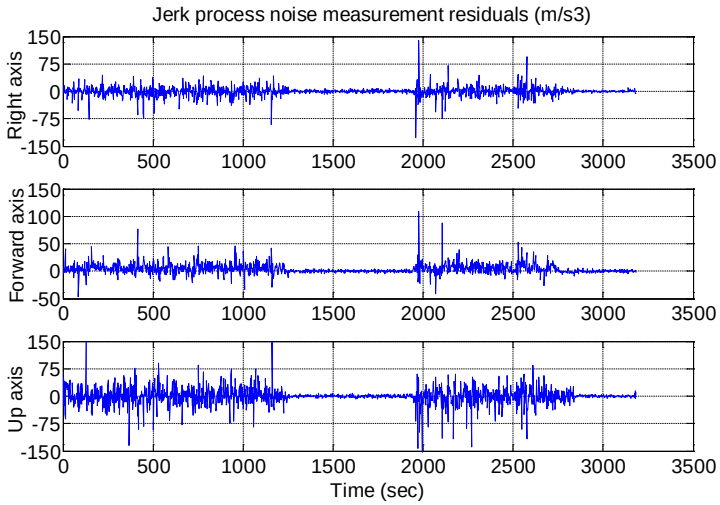


Figure 6-22 Residuals of the virtual zero jerk process noises

Figure 6-22 and Figure 6-33 show the residuals of the virtual zero process noise measurements. Although the actual physical jerks and angular acclerations are not normal distributed random errors, they are treated so in the GMIKF. Hence, a set of numbers is also needed to describe their stochastic properties. In both figures, the initial conservative variances for jerk and angular acceleration process noises succeed in their job because the amplitudes of corresponding residuals are consistently smaller than the initial values, which also hints that there is still room for the further tuning of variance of these noise components.

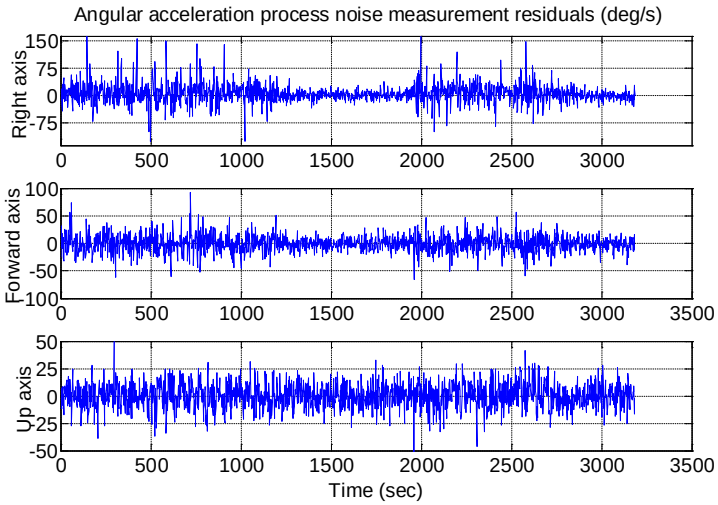


Figure 6-23 Residuals of the virtual zero angular acceleration process noises

Residuals of DD GPS measurements

All raw GPS sensors took part in the GMIKF in the form of the double differenced measurements between the base station and the GPS rover receiver with respect to the reference GPS satellite which is SVPRN 15 in this section. Regarding each type of GPS measurements, the system innovations and residuals of six non-reference GPS satellites (SVPRN 13, 18, 20, 21, 24 and 29) are depicted in Figure 6-24 – Figure 6-27. The close match between innovations and residuals imply the fact that the predicted states well fit the measurement model.

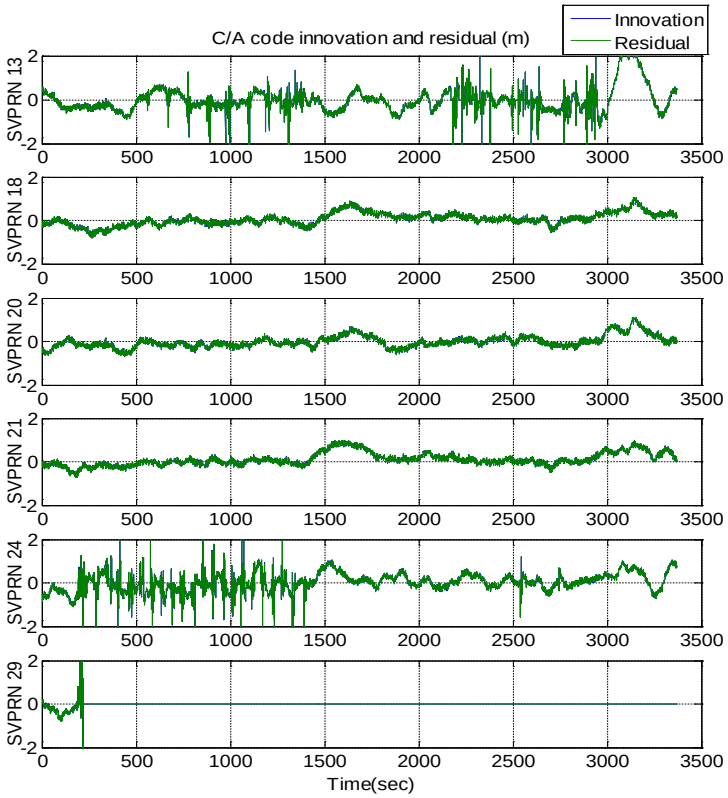


Figure 6-24 System innovations and residuals of GPS satellite C/A code measurements

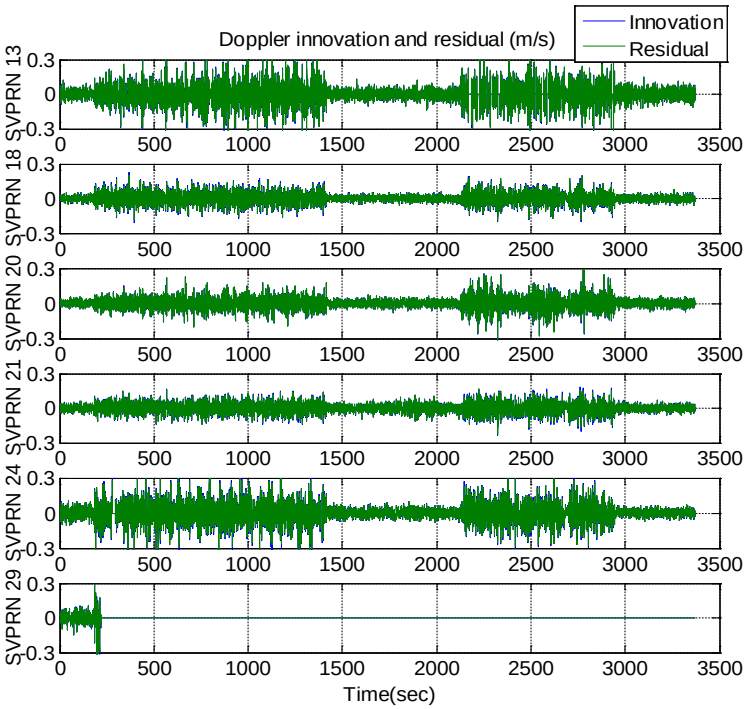


Figure 6-25 System innovations and residuals of GPS satellite Doppler measurements

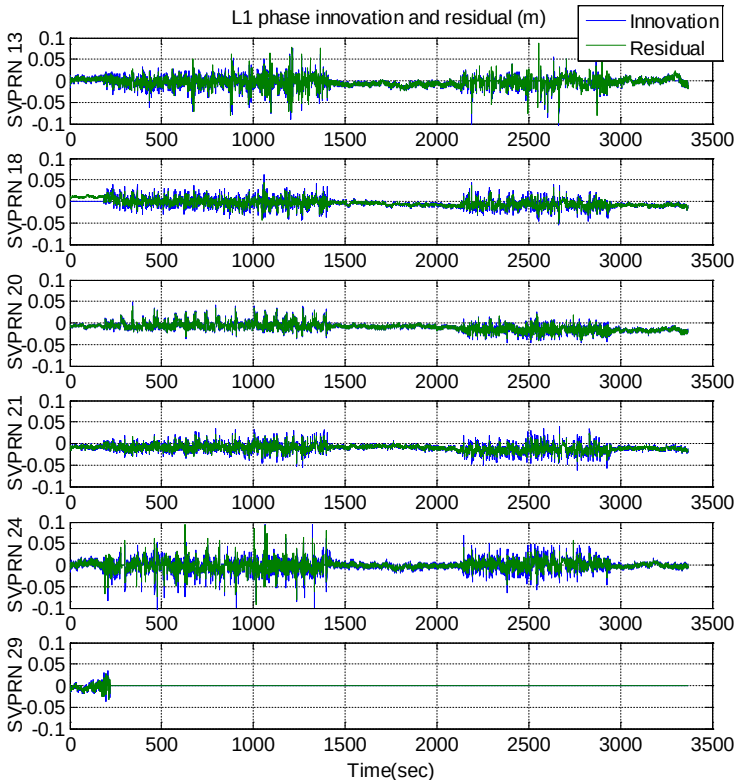


Figure 6-26 System innovations and residuals of GPS L1 band carrier phase measurements

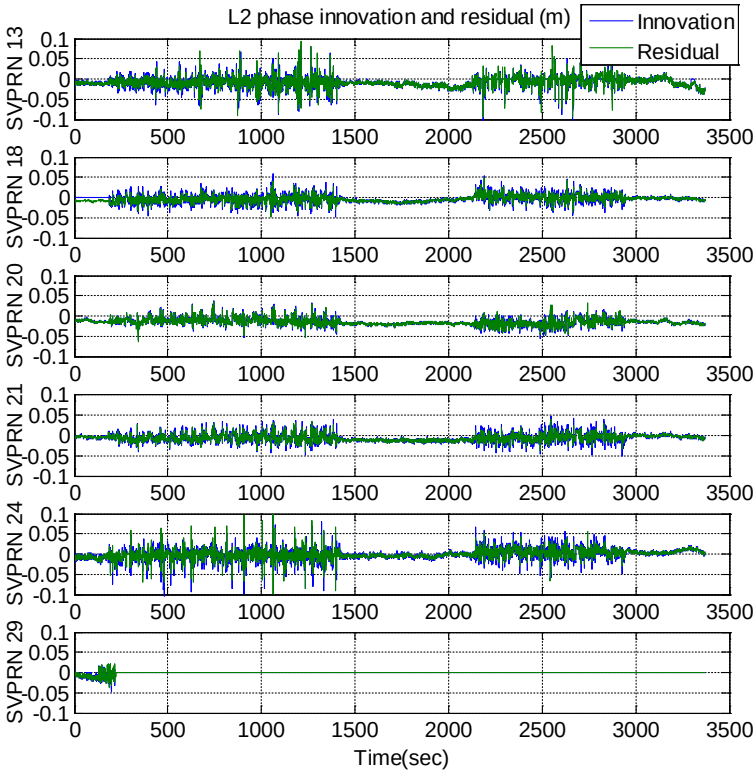


Figure 6-27 System innovations and residuals of GPS L2 band carrier phase measurements

Residuals of IMU measurements

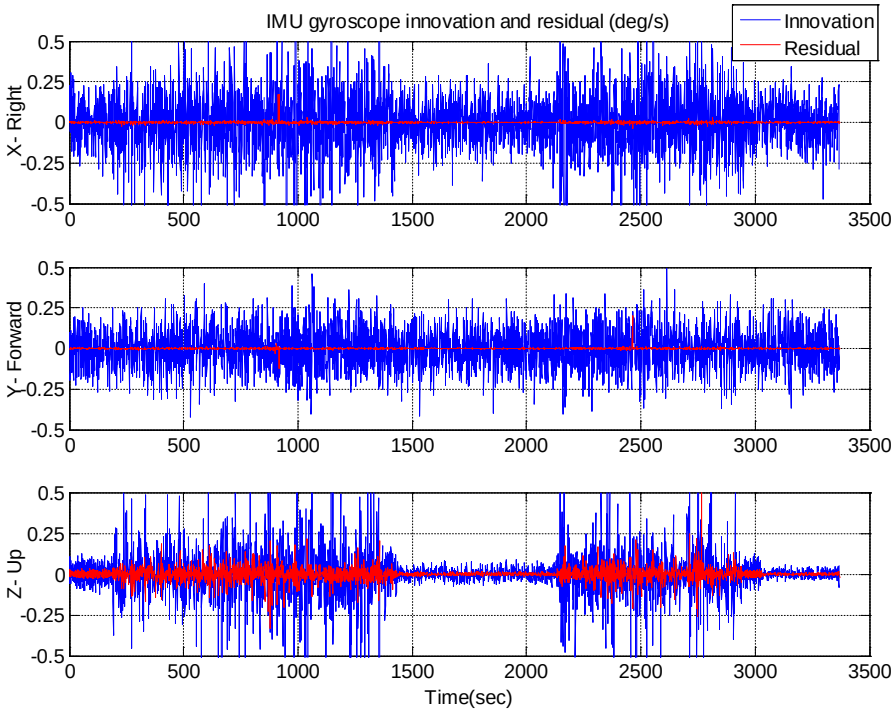


Figure 6-28 System innovations and residuals of gyroscope measurements

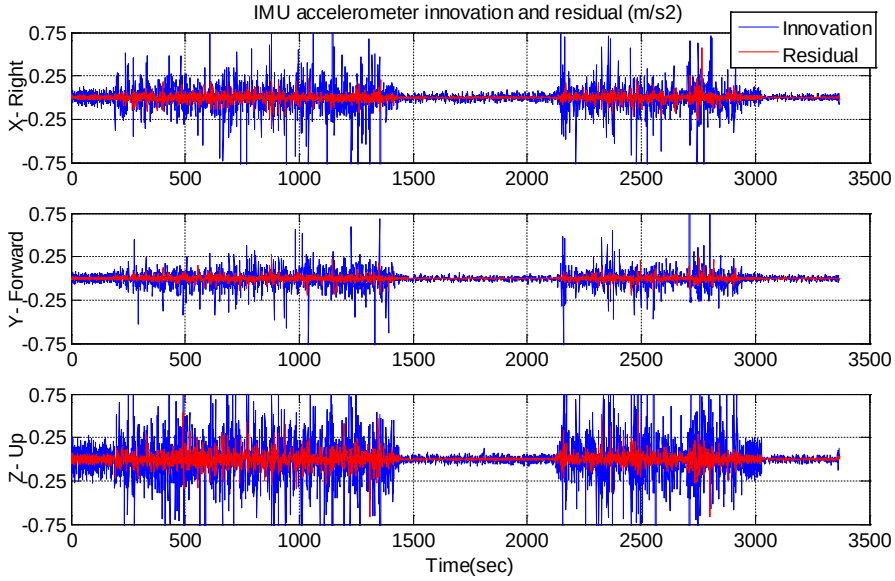


Figure 6-29 System innovations and residuals of accelerometer measurements

The large innovations and small residuals for gyroscope measurements in Figure 6-28 indicate that the estimated angular rate states in GMIKF follow the raw measurements instead of

their prediction, which is directly caused by the high level of process noises and the relatively small measurement noises. On the other hand, the three plots in Figure 6-29 show a much close pair of innovations and residuals compared to the ones in Figure 6-28, which implies that the predicted accelerations nicely fit the measured counterparts in all three axes.

Residuals of DGPS heading measurements

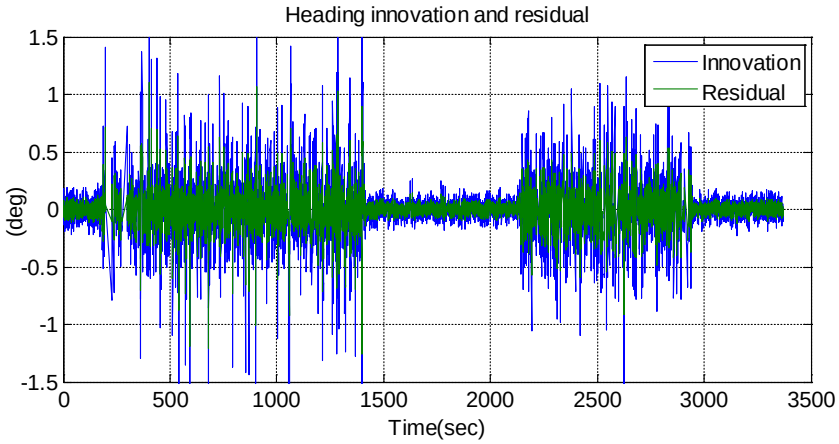


Figure 6-30 DGPS heading measurements innovations and residuals in GMIKF

Similar to the IMU accelerometer measurements, the Figure 6-30 shows that DGPS heading measurements are definitely absorbed into GMIKF for the heading estimation because the amplitudes of the residuals are consistently smaller than that of the innovations.

6.2.4 VCE in GMIKF

In inertial navigation, the tuning of the variance components for a nonlinear Kalman filter experiencing non-consistent dynamics are relatively subjective because no unique set of stochastic numbers can beat all other candidates to achieve unanimous **minimum mean square error (MMSE)** in various environments. In addition, for a non-adaptive Extended Kalman filter, the stochastic model for process noises and measurement noises should be more conservative in

order to handle the intermittent measurement losses and associated rapidly growing errors. Based on the experiences gained from this section, the key aspects of tuning the stochastic model tuning in Kalman filter are summarized as follows:

1) Identify the measurement blunders in Kalman Filter and use them to monitor the fitness of the Kalman filter.
2) Adjust (increase) the variances when the numbers of blunders are significantly increased.
3) Examine the changes of the state solution resulted from the new variance components in order to avoid solution divergence.
4) Avoid over-tuning the components with non-consistent stochastic property, for example, the jerks as the process noise components in determination of the land vehicle's trajectory.
5) Do not tune the noise components with small redundant indices.

Because the goal of the VCE process is to achieve the best balance between the accuracy and the robustness for GMIKF, the finalized VCE results for all noise components in GMIKF are relatively conservative compared to the ones directly derived from the VCE algorithm outputs.

6.2.4.1 Iterative VCE tuning results

This section first shows the changes of process and measurement noise configurations in GMIKF between the initial settings and the final results (Table 6-3). Then the VCE results of the first and final iteration are tabulated in Table 6-4 - Table 6-7 while the intermediate results are plotted in the figures for the better vision effects.

Table 6-3 Standard deviations of process and measurement noises in GMIKF (1σ)

	First	Final		First	Final
$Jerk_x$	72 (m/s^3)	40 (m/s^3)	$\dot{\omega}_x$ (angular acceleration)	300($°/s^2$)	30 ($°/s^2$)
$Jerk_y$	72(m/s^3)	40 (m/s^3)	$\dot{\omega}_y$ (angular acceleration)	300($°/s^2$)	30 ($°/s^2$)
$Jerk_z$	72(m/s^3)	62 (m/s^3)	$\dot{\omega}_z$ (angular acceleration)	300($°/s^2$)	30 ($°/s^2$)
DD GPS C/A	0.5 (m)	0.15 (m)	DD GPS L1 phase	0.02 (m)	0.003 (m)
DD GPS Doppler	0.1 (m/s)	0.02 (m/s)	DD GPS L2 phase	0.02 (m)	0.003 (m)
Gyroscope- x	0.75 (°/s)	0.15 (°/s)	Accelerometer –x	0.16 (m/s^3)	0.1 (m/s^3)
Gyroscope- y	0.75 (°/s)	0.12 (°/s)	Accelerometer –y	0.16 (m/s^3)	0.1 (m/s^3)
Gyroscope- z	0.75 (°/s)	0.11 (°/s)	Accelerometer –z	0.16 (m/s^3)	0.1 (m/s^3)
DGPS heading	0.25 (°)	0.12 (°)			

Table 6-4 The VCE tuning results for the process noises in GMIKF (first iteration)

	J_x	J_y	J_z	$\dot{\omega}_x$	$\dot{\omega}_y$	$\dot{\omega}_z$
σ_i (initial s.t.d.)	72	72	72	300	300	300
Ri (redundant index)	0.9	0.9	0.9	0.89	0.89	0.9
$\hat{\sigma}_{0i}^2$ (variance components)	0.049	0.034	0.19	0.01	0.004	0.002
$\tilde{\sigma}_i = \sigma_i \times \hat{\sigma}_{0i}$ (next variances)	15.94	13.28	31.28	30	18.97	13.42

Table 6-5 The VCE tuning results for the measurement noises in GMIKF (first iteration)

	σ_ω^x	σ_ω^z	σ_ω^y	σ_a^x	σ_a^y	σ_a^z	σ_{ca}	σ_{L1}	σ_{L2}	σ_{D1}	σ_{hd}
σ_i	0.75	0.75	0.75	0.16	0.16	0.16	0.5	0.02	0.02	0.1	0.5
Ri	0.05	0.05	0.05	0.037	0.04	0.04	0.7	0.8	0.8	0.7	0.9
$\hat{\sigma}_{0i}^2$	-	-	-	-	-	-	0.36	0.028	0.033	0.038	0.12
Blunder number	0	0	0	0	0	0	102	2	2	12	6
$\tilde{\sigma}_i$	0.75	0.75	0.75	0.16	0.16	0.16	0.16	0.003	0.004	0.019	0.17

Table 6-6 The VCE tuning results for the process noises in GMIKF (final iteration)

	J_x	J_y	J_z	$\dot{\omega}_x$	$\dot{\omega}_y$	$\dot{\omega}_z$
σ_i	40	40	62	30	30	30

Ri	0.9	0.9	0.9	0.7	0.8	0.83
$\hat{\sigma}_{0\mathrm{i}}^2$	0.12	0.3	0.25	0.27	0.15	0.06
$\tilde{\sigma}_i$	13.86	21.91	31	15.59	11.62	7.348

Table 6-7 The VCE tuning results for the measurement noises in GMIKF (final iteration)

	$\sigma_{\omega}^{\mathrm{x}}$	$\sigma_{\omega}^{\mathrm{z}}$	$\sigma_{\omega}^{\mathrm{y}}$	$\sigma_{\mathrm{a}}^{\mathrm{x}}$	$\sigma_{\mathrm{a}}^{\mathrm{y}}$	$\sigma_{\mathrm{a}}^{\mathrm{z}}$	σ_{ca}	σ_{L1}	σ_{L2}	σ_{D1}	σ_{hd}
σ_i	0.15	0.12	0.11	0.1	0.1	0.1	0.16	0.003	0.003	0.02	0.12
Ri	0.15	0.15	0.09	0.05	0.05	0.02	0.7	0.8	0.8	0.9	0.934
$\hat{\sigma}_{0\mathrm{i}}^2$	0.39	0.23	-	-	-	-	0.8	0.72	1.12	1.2	0.92
Blunder number	300	118	0	69	38	120	36	42	42	20	16
$\tilde{\sigma}_i$	0.095	0.058	0.11	0.1	0.1	0.1	0.15	0.003	0.003	0.022	0.12

According to Section 5.2, the theoretic optimal unbiased estimates for the variances in a Kalman filter are obtained until all of the estimated variance components of unit weight are close to 1.0. However, in order to tolerate the vast changes of the dynamics, the variance components of the process noise components in GMIKF hardly approach this optimal criterion. In other words, the strict (small) variances of the process noises will harm the filter's fault-tolerant ability. In summary, the equality condition has to be compromised to certain extent so that the algorithm's robustness and performance can be well balanced. Therefore, in Table 6-7, although the variance components of the GPS related measurements are approaching 1.0, the incurred blunders are also greatly increased compared to those in Table 6-5.

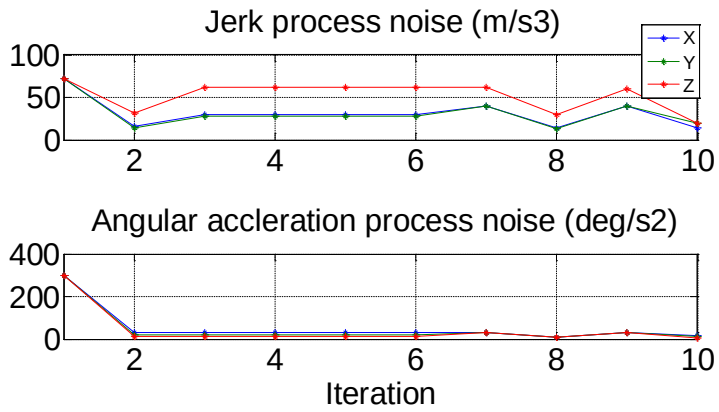


Figure 6-31 Iterative VCE results for jerk and angular acceleration noises

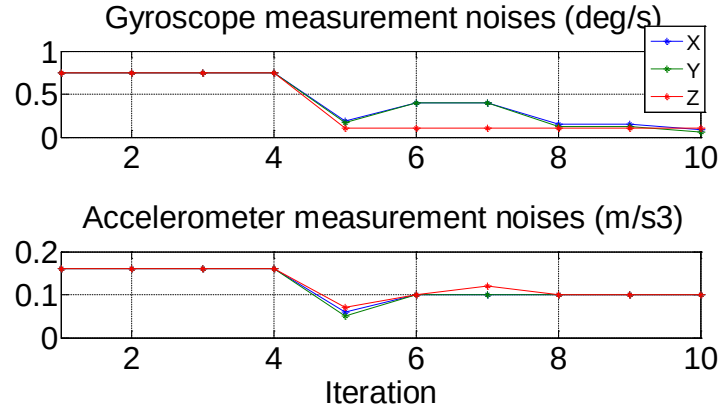


Figure 6-32 Iterative VCE results for gyroscope and accelerometer measurements

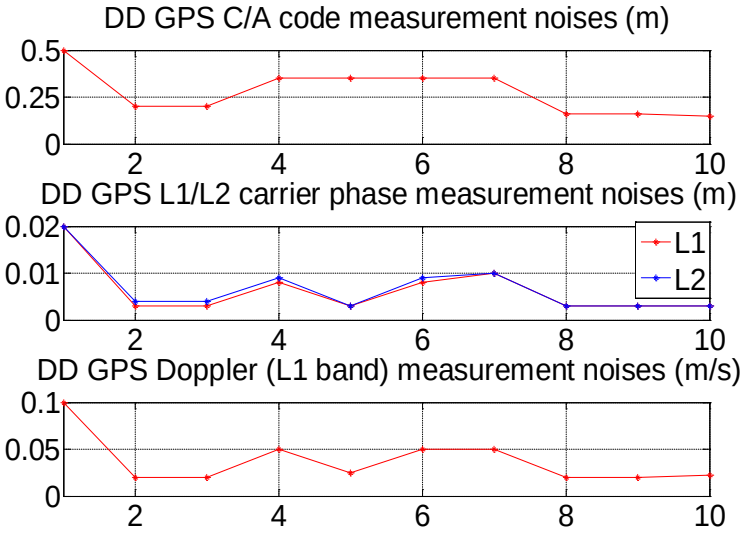


Figure 6-33 Iterative VCE results for DGPS heading measurements

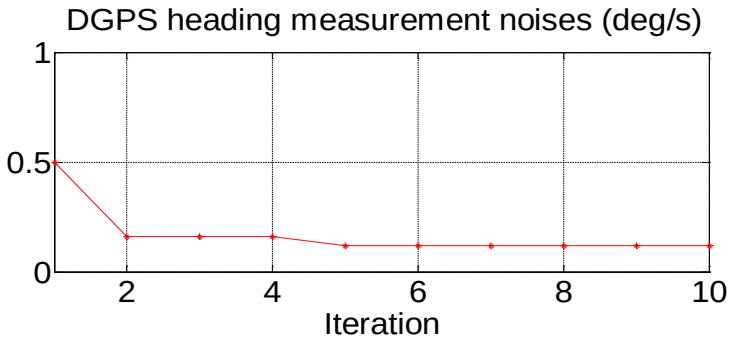


Figure 6-34 Iterative VCE results for DGPS heading measurements

As listed in Table 6-6 and Table 6-7, the iterative VCE process for GMIKF ends with a set of variances balancing the solution performance and the system's robustness. The large variations of the variance estimates in above four figures (Figure 6-31 - Figure 6-34) reflect the conflicts between the enhanced solution standard deviation and the weakened system error tolerance capabilities.

As mentioned in Chapter 5, the final VCE results obtained from one dataset can not represent the universal properties of those noise components. For the future application of the variances of the noise components in Table 6-7 in various datasets, it is recommended that the set of configurations should be moderately applied because the system robustness is of paramount importance.

6.2.4.2 Histograms comparisons

After the corresponding residuals are computed through (2.7.13 – 2.7.16), this section demonstrates the changes of the residual histograms for the virtual process noise measurements (jerk and angular acceleration) and all raw measurements used in GMIKF including DD GPS measurements and DGPS heading measurements.

Histograms of residuals of the process noises

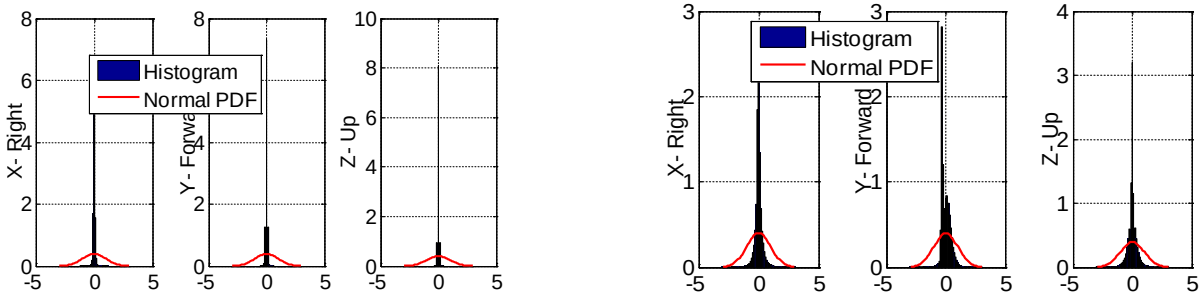


Figure 6-35 Jerk residual histograms (first vs. final iteration)

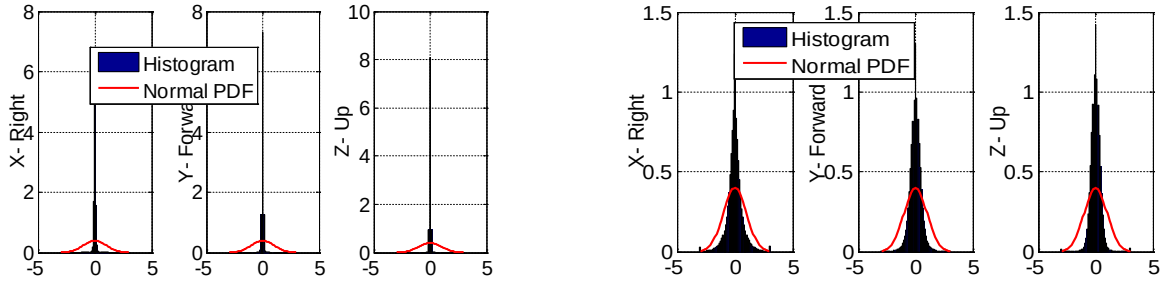


Figure 6-36 Angular acceleration residual histograms (first vs. final iteration)

Figure 6-35 and Figure 6-36 present the histograms of the standardized residuals of six virtual zero process noise measurements based on 336,572 residual samples. According to 2.7.15, the residuals were computed along with other raw measurement residuals. Considering that both jerk and angular acceleration are the properties of the moving vehicle's dynamics, the variance component estimation pauses while the vehicle is stationary. Although the estimated variances of the jerk and the angular accelerations are much moderate ($\hat{\sigma}_{0\mathrm{i}}^2 \neq 1$), the residual histograms still visually match the standardized normal distribution curve much better than their slim column counterparts resulted from the initial variance configurations before the VCE process was introduced.

Histograms of DD GPS measurement residuals

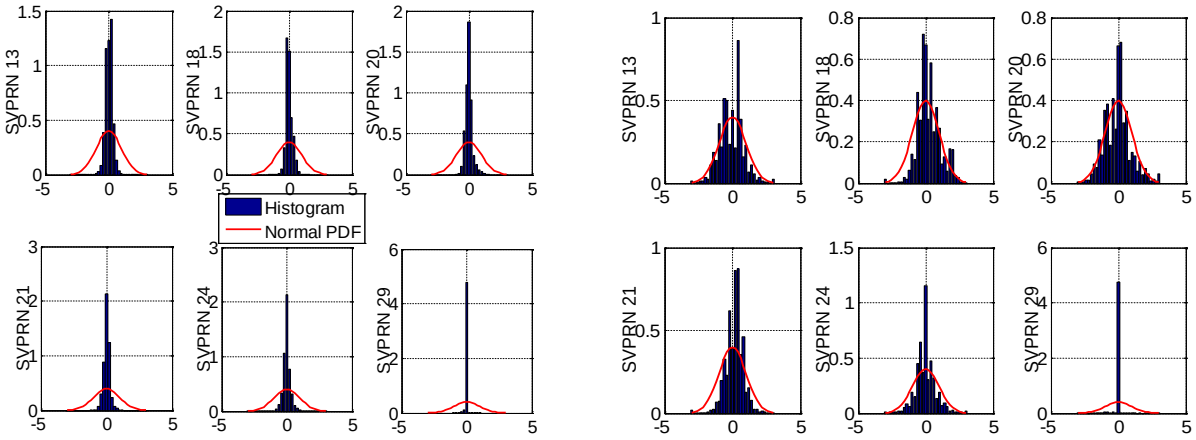


Figure 6-37 GNSS L1 residual histograms (first vs. final iteration)

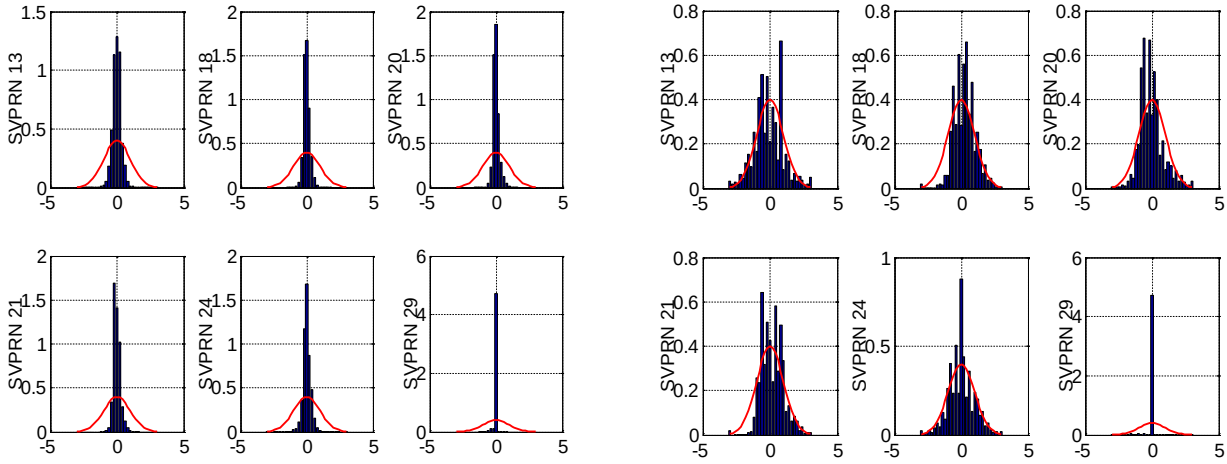


Figure 6-38 GNSS L2 residual histograms (first vs. final iteration)

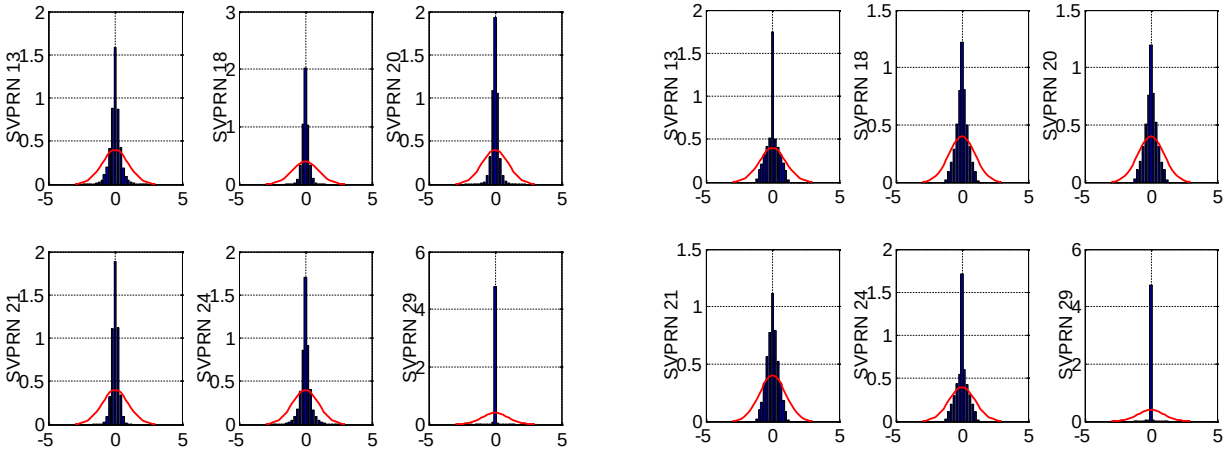


Figure 6-39 GNSS D1 residual histograms (first vs. final iteration)

In the above figures, all histograms (except the GPS measurements for SVPRN 29) resulted from the final iteration unanimously fit the normal distribution PDF curves better than the ones from the initial iteration. The over optimistic measurement residuals for SVPRN 29 are mainly caused by two facts: 1) the number of measurement samples is relatively small due to its short time availability according to Figure 6-10; and 2) almost all measurements with SVPRN 29 were sampled while the vehicle was in stationary. As a result, the optimistic measruement residuals collected in the very beginning of the trajectory distort the residual histograms.

Histograms of IMU measurement residuals

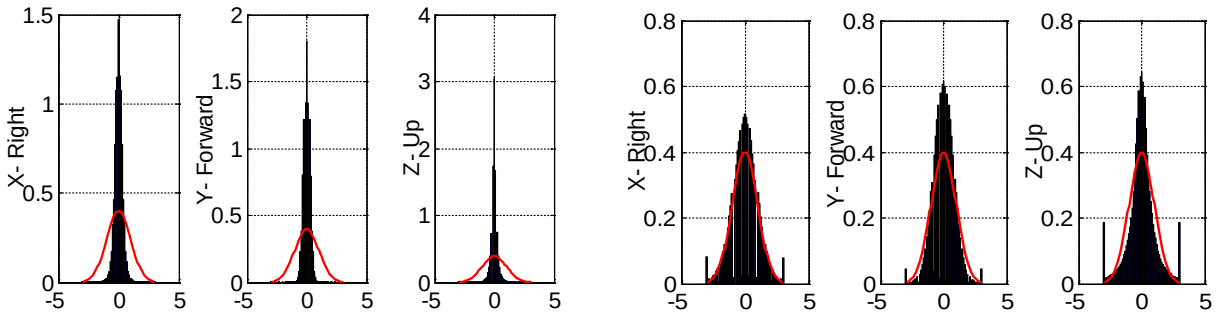


Figure 6-40 IMU gyroscope residual histograms (first vs. final iteration)

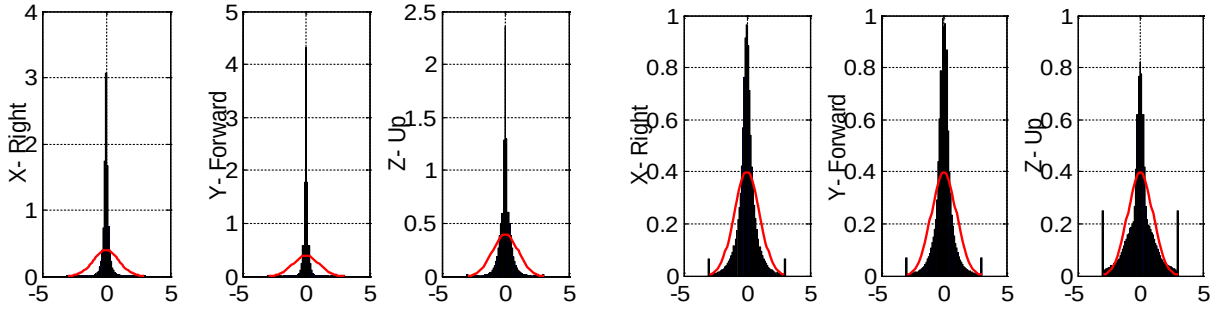


Figure 6-41 IMU accelerometer residual histograms (first vs. final iteration)

According to Table 5-7, the Allan variances for both gyroscopes and accelerometers are significantly superior to the manufacturer specification and the VCE results, which has been verified again by the VCE trials in this section. Although the intermediate VCE outputs can

achieve 0.05 m/s^2 for the accelerometers (Figure 6-32), they are not applicable to GMIKF because the vibrations of the vehicle bring too much noises into the accelerometers. Also, the vibrations might trigger extra noises in IMU440's gyroscopes' measurements because the MEMS-based IMU are sensitive to the vibrations. On the other hand, the VCE results show that the IMU440CA does not exert its maximum power when the conservative specifications from manufacturer are applied.

Histograms of DGPS heading measurement residuals

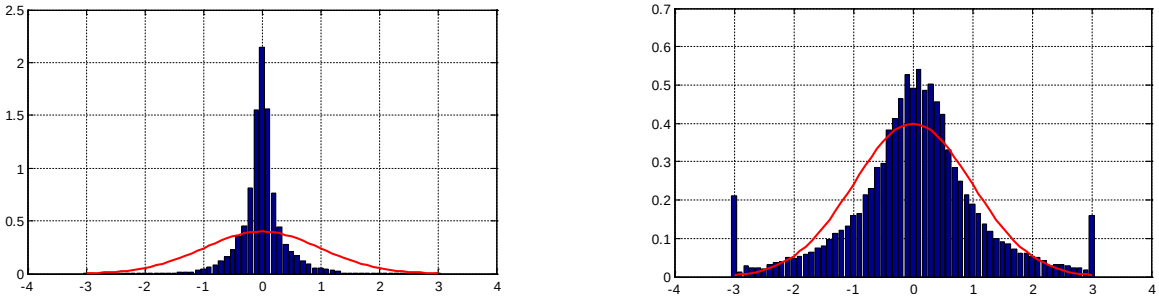


Figure 6-42 DGPS heading residual histograms (first vs. final iteration)

According to Table 6-7, the DGPS heading measurements derived from the 2m long fixed baseline could reach an accuracy of 0.12º (1σ) which is verified through the nice match between the residual histogram and the normal distribution curve in Figure 6-42.

All in all, we confidently conclude that the measurement variances in "final" column of Table 6-3 are the moderately optimal configurations for all noise components for the GMIKF targeting on YUMIS system (land vehicle environment) regarding their stochastical properties of the process and measurement noises.

7. Conclusion and future work

7.1 Conclusions

A breakthrough in multisensor integration for kinematic positioning and navigation has been demanded with the development of the modern sensor and computer technologies. Hence, three principal research objectives were set in Section 1.2, which are 1) Design, develop and validate a hard real-time multisensor system; 2) Derive and develop a GMIS-based Kalman filter to estimate navigational parameters with the full usage of *a priori* knowledge of a rigid body's kinematics; 3) Tune and evaluate the stochastic models of all random error sources by means of the variance component estimation (VCE) technique for all participating sensors, specifically including inertial sensors. Accordingly, the essential outcomes achieved in this research are summarized below.

As the first contribution documented in Chapter 3, York University Multisensor Integrated System (YUMIS), a kinematic positioning and navigation system built upon Linux with Real Time Application Interface (Linux/RTAI), was successfully developed to provide the hardware platform for acquiring real geospatial data from multiple positioning and orientation sensors, such as the low-cost IMU units, OEM GNSS sensors, image sensors and etc. YUMIS exemplifies a low-cost yet high performance hardware platform stereotype for multisensor integration and real-time software development.

1) As the pre-requisite component in YUMIS system, the Linux/RTAI is successfully applied in YUMIS system which meets the stringent requirements upon the high performance real-time multitask management and the flexibilities of system expansion.

2) The generic software architecture proposed in YUMIS provides an easy solution for the multisensor integration, e.g., data collection, data time-tagging, data store and retrieval.
3) YUMIS based on Linux/RTAI offers the convenience of real-time software development for multisensor integrated inertial positioning and navigation by using the LXRT technique which allows the smooth conversion of a normal Linux application to a RTAI/LXRT application at the cost of a few lines of source code.
4) YUMIS lays the foundation for the follow-up systematic training of highly qualified personnel devoted to the data processing in multisensor integration relying on the expensive commercial products, who are usually discouraged by the complicated hardware environment setup and time consuming device driver development.

Secondly, as the most paramount contribution of this research, a novel generic multisensor integration (GMI) strategy for a multisensor integrated kinematic positioning and navigation system is proposed, in which a generic 3D kinematic trajectory model is developed as the system model and the strong dependence on the *a priori* error characteristics of inertial sensors in the conventional integration mechanism is cleared. With the conventional integration strategy, a GNSS (Global Navigation Satellite Systems) aided inertial positioning and navigation system (GNSS/INS) equipped with a low-cost IMU suffers from its low and unstable performance. The competitive advantages with the GMI strategy over the existing implementations are exhibited as follows:

1) GMI strategy maintains the kinematics of a moving platform in the core system model over an appropriate time interval so that the solution drifts due to IMU errors can be efficiently mitigated in the GNSS degraded or denied environments. In addition, the core system model can be helpful to reject the GNSS pseudo-range measurements with large

multipath errors and/or the GNSS carrier phase measurements with wrong fixed ambiguities.

2) GMI strategy directly integrates the individual sensors into the KF without distinguishing the core IMU from other aiding sensors. Hence, the direct IMU measurement updates in KF are made possible, which is essential for the utilization of low-cost IMUs in the high-quality direct-georeferencing technology (DGT). In addition, the superior performance (solution accuracy) arises from the improvement of the overall system measurement redundancy upon the angular rates and accelerations.
3) GMI strategy delivers a compact and efficient system (software) structure directly dealing with the navigational parameters, which streamlines the integration of all kinds of measurements, e.g., delta position, multiple IMUs.
4) With GMI strategy, a thorough error analysis is enabled at the sensor level because no more blended error measurements involving IMU measurements are required, which is paramount in a survey grade DGT system.
5) With GMI strategy, the possibility of the simultaneous estimation of variance components in the process and measurement noise vectors allows the investigation of the multisensor interoperability in terms of the sensor data accuracy.

As the third principal contribution, the improvement (tuning) of the *a priori* stochastic models of the process and measurement noise vectors in Kalman Filer (KF) for a multisensor inertial navigation system as documented in Chapter 5 has been comprehensively studied in this research. It is known that tuning the stochastic model of random measurement errors is premised on the statistical independence of random errors among participant sensors, however, which does

not hold true in the traditional integration mechanism due to the amalgamation of the observables from inertial sensor and other aiding sensors. The highlights of my research on this part are:

1) The issue of the statistical independence among sensors in a multisensor integrated inertial positioning and navigation system is cleared out by the usage of the GMI strategy.
2) Based on the equity theory on Kalman filter and Least Squares [Wang, 1997] and the rigorous VCE in Least Squares after Helmert [1907], for the very first time, the *posteriori* variance component estimation in a multisensor integrated navigation system succeeds in simultaneous estimation of all independent individual raw measurements (e.g., IMU and GNSS measurements) and all independent individual components in the process noise vector (e.g., jerks and angular accelerations in GMIKF).
3) Through the timely accumulation of the epoch wise measurement redundancy indices and measurement residuals, the simplified VCE in Kalman filter after the rigorous Helmert method enables a thorough error analysis at the sensor level under both the real-time and the post-processing environment.

7.2 Future works

In terms of the further researches on the GMI strategy in the multisensor integrated inertial positioning and navigation, the suggestions are:

1) In the current research, only one IMU unit is applied in consideration of the simplicity and focus of the research on core issues in GMIS. However, the GMIS allows for the integration of the multiple IMU units without major changes to the integration strategy. Therefore, the natural expansion of the GMIS is to replace single IMU unit with the multiple units to achieve better balance between performance and cost.
2) Automatically calibrate low-cost IMU unit using the generic multisensor integration strategy in the realistic working environment or on well-designed test sites to discuss the inertial sensor performance changes.
3) Investigate the performance of GMIKF with reduced state components to relieve the heavy computation load, e.g., gyroscope's measurements.
4) Substitute the Euler angle with quaternion technique to achieve better linearity in system model to enhance the EKF's performance.
5) Apply pre-filter technique [Joglekar, et al, 1975] to reduce the computation cost while maintaining the performance.
6) Apply delta position measurements for the low-cost positioning and navigation system, which can be derived from the LiDAR point clouds, the time differenced GNSS carrier phase measurements, and etc.
7) Combine the advanced non-linear filters (e.g. unscented KF) with the GPU computing technique to improve both the solution accuracy and the computation speed.

8. References

Aeolean Inc. (2016): Introduction to Linux for Real-Time Control. Prepared for: Intelligent Systems Division, National Institute of Standards & Technology, U.S.A. http://www.aeolean.com/html/RealTimeLinux/RealTimeLinuxReport-2.0.0.pdf (accessed on Jan 15, 2016).

Aggarwal, P., Syed, Z., Niu, X., & El-Sheimy, N. (2008): A standard testing and calibration procedure for low cost MEMS inertial sensors and units, Journal of navigation, 61(02), 323-336.

Allan, D. W. (1966): Statistics of Atomic Frequency Standards. In: Proc of the IEEE, 54, 221-230.

Amiri-Simkooei, A. (2007): Least-squares variance component estimation: theory and GPS applications (Doctoral dissertation, TU Delft, Delft University of Technology).

Anderson, B. D., & Moore, J. B. (1979): Optimal filtering. Englewood Cliffs, 21, 22-95.

Angrisano, A. (2010): GNSS/INS integration methods, Dottorato di ricerca (PhD) in Scienze Geodetiche e Topografiche Thesis, Universita'degli Studi di Napoli PARTHENOPE, Naple.

Ardakani, H. A., & Bridges, T. J. (2010): Review of the 3-2-1 Euler angles: a yaw-pitch-roll sequence, Department of Mathematics, University of Surrey, Guildford GU2 7XH UK, Tech. Rep.

Artese, G., & Trecroci, A. (2008): Calibration of a low cost MEMS INS sensor for an integrated navigation system. Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci, 877-882.

Bähr, H., Altamimi, Z., & Heck, B. (2007): Variance component estimation for combination of terrestrial reference frames, Universität Karlsruhe Schriftenreihe des Studiengangs Geodäsie und Geoinformatik, 6.

Bancroft, J. B. (2010): Multiple inertial measurement unit integration for pedestrian navigation, Department of Geomatics Engineering, UCGE Reports, (20320), 196.

Bancroft, J. B., & Lachapelle, G. (2011): Data fusion algorithms for multiple inertial measurement units. Sensors, 11(7), 6771-6798.

Bar-Itzhack, I. Y. (2012): Identity between INS position and velocity error models, Journal of Guidance, Control, and Dynamics.

Bar-Itzhack, I. Y., & Berman, N. (1988): Control theoretic approach to inertial navigation systems. Journal of Guidance, Control, and Dynamics, 11(3), 237-245.

Bar-Itzhack, I. Y., & Harman, R. R. (2004): Implicit and explicit spacecraft gyro calibration, In Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit, (Providence, Rhode Island, USA).

Barabanov, M. (1997): A linux-based real-time operating system (Doctoral dissertation, New Mexico Institute of Mining and Technology).

Barbalace, A., Luchetta, A., Manduchi, G., Moro, M., Soppelsa, A., & Taliercio, C. (2008): Performance comparison of VxWorks, Linux, RTAI, and Xenomai in a hard real-time application. IEEE Transactions on Nuclear Science, 55(1), 435-439.

Bavdekar, V. A., Deshpande, A. P., & Patwardhan, S. C. (2011): Identification of process and measurement noise covariance for state and parameter estimation using extended Kalman filter. Journal of Process control, 21(4), 585-601.

Bekir, E. (2007): Introduction to modern navigation systems, World Scientific Publishing Co. Pte. Ltd., 2007.

Bessell, A., Ristic, B., Farina, A., Wang, X., & Arulampalam, M. S. (2003): Error performance bounds for tracking a manoeuvring target, In Proceedings of the 6th International Conference on Information Fusion (pp. 903-910).

Benzerrouk, H. (2012): Gaussian vs. Non-Gaussian noise in inertial/GNSS integration. GNSS Solutions, Inside GNSS Magazine, 32-39.

Bisnath, S. B., & Langley, R. B. (2000): Efficient, automated cycle-slip correction of dual-frequency kinematic GPS data, In proceedings of ION GPS (pp. 145-154).

Bose, A., Puri, S., & Banerjee, P. (2008): Modern inertial sensors and systems. PHI Learning Pvt. Ltd.

Bossler, J., Goad, C., Johnson, P., & Novak, K. (1991): GPS and GIS map the nation's highways. Geo Info Systems, 1(3), 27-37.

Bortz, J. E. (1971): A new mathematical formulation for strapdown inertial navigation, IEEE transactions on aerospace and electronic systems, (1), 61-66.

Britting, K. R. (1971): Inertial navigation systems analysis, Wiley, New York.

Brown, R. G., & Hwang, P. Y. (1997): Introduction to random signals and applied kalman filtering. 1997. NY John Wiley and Sons.

Bulut, Y., Vines-Cavanaugh, D., & Bernal, D. (2011): Process and measurement noise estimation for Kalman filtering. In Structural Dynamics, Volume 3, pp. 375-386, Springer New York.

Carlson, N. A. (1990): Federated square root filter for decentralized parallel processors. *IEEE Transactions on Aerospace Electronic Systems*, *26*, 517-525.

Caspary, Wilhelm and Wang, Jianguo (1998): Redundanz-anteile und Varianzkomponenten im Kalman Filter, Zeitschrift für Vermessungswesen, Vol. 123, No.4, 1998, pp.121-128.

Cannon Jr, R. H. (2012): Alignment of inertial guidance systems by gyrocompassing-linear theory, Journal of the Aerospace Sciences.

CHANG, Z. Q., HAO, J. M., & LI, J. Y. (2008): Cycle Slips and Gross Error Checkout Using Doppler Observations [J], Bulletin of Surveying and Mapping, 3, 008.

Chen, Zhe (1985): Principle of Strapdown Inertial Navigation System, Beijing University of Aeronautics and Astronautics Press, Beijing, China.

Chonhenchob, V., Singh, S. P., Singh, J. J., Stallings, J., & Grewal, G. (2012): Measurement and Analysis of Vehicle Vibration for Delivering Packages in Small - Sized and Me
Trucks and Automobiles. Packaging Technology and Science, 25(1), 31-38.

Claudia, C., & Naranjo, M. (2008): Analysis and Modeling of MEMS Based Inertial Sensors. Signal Processing School of Electrical Engineering. Report XR EE SB.

Coffee, J. R., & Saggio, F. (1989): Strapdown gyro contribution to coning motion errors. In IN: IEEE International Conference on Systems Engineering, Dayton, OH, Aug. 24-26, 1989, Proceedings (A90-52860 24-66). New York, Institute of Electrical and Electronics Engineers, Inc., 1989, p. 55-58.

Collins, J. P. and Langley, R. B. (1999): Possible weighting schemes for GPS carrier phase observations in the presence of multipath, final contract report for the US Army Corps of Engineers Topographic Engineering Center, No. DAAH04-96-C-0086/TCN, 98151.

Crossbow Technology Inc. (2010): 440 Series User's Manual, Document 7430-0131-01, 2010, U.S.A.

Crossbow Technology Inc. (2002): DMU User's Manual. 2002, U.S.A.

Cui, X., Yu, Z., Tao, B., Liu, D., Yu, Z., Sun, H., & Wang, X. (2001): Generalized surveying adjustment. Wuhan Technique University of Surveying and Mapping, Wuhan, 103-104.

De Agostino, M., Manzino, A. M., & Piras, M. (2010): Performances comparison of different MEMS-based IMUs. In Position Location and Navigation Symposium (PLANS), 2010 IEEE/ION, pp. 187-201.

De Pasquale, G., & Somà, A. (2010): Reliability testing procedure for MEMS IMUs applied to vibrating environments. Sensors, 10(1), 456-474.

Derpanis, K. G. "Cramer-Rao Bound" (2006): http://www.cs.yorku.ca/~kosta/CompVis_Notes/cramer-rao.pdf (accessed on Jan, 10, 2016).

Ding, Weidong; et al. (2008): Time synchronization error and calibration in integrated GPS/IMU (MEMS) systems. ETRI Journal, 2008, 2, Volume 30, pp. 59–67.

Dissanayake, G., Sukkarieh, S., Nebot, E., & Durrant-Whyte, H. (2001): The aiding of a low-cost strapdown inertial measurement unit using vehicle model constraints for land vehicle applications, IEEE transactions on robotics and automation, 17(5), 731-747.

Duník, J., & Šimandl, M. (2008): Estimation of state and measurement noise covariance matrices by multi-step prediction, IFAC Proceedings Volumes, 41(2), 3689-3694.

Edwan, E., Zhou, J., Zhang, J., & Loffeld, O. (2012): A new loosely coupled DCM based GPS/INS integration method. Navigation, 59(2), 93-106.

El-Diasty, M., & Pagiatakis, S. (2009): A rigorous temperature-dependent stochastic modelling and testing for MEMS-based inertial sensor errors. Sensors, 9(11), 8473-8489.

El-Diasty, M., & Pagiatakis, S. (2008): Calibration and stochastic modelling of inertial navigation sensor errors. Journal of Global Positioning Systems, 7(2), 170-182.

El-Sheimy, N., & Hassan, T. (2007): Mobile Mapping Systems-The New Trend in Mapping and GIS Applications.

Euler, L. (1776): Novi Commentarii academiae scientiarum: Petropolitanae.

Feng, Y., Quan, P., Yan, L., Yongmei, C., & Wei, L. (2006): Survey of sampling nonlinear filter, In Control Conference, 2006. CCC 2006. Chinese (pp. 535-540). IEEE.

FireWire™ Reference Tutorial - 1394 Trade Association. (2010): http:// http://www.1394ta.org/press/whitepapers/firewire%20reference%20tutorial.pdf (accessed on April, 17, 2017)

Flenniken, W. (2005): Modeling inertial measurement units and analyzing the effect of their errors in navigation applications (Doctoral dissertation).

Fong, W. T., Ong, S. K., & Nee, A. Y. C. (2008): Methods for in-field user calibration of an inertial measurement unit without external equipment, Measurement Science and technology, 19(8), 085202.

Förstner, W. (1979): Ein Verfahren zur Schätzung von Varia und Kovarianzkomponenten, Allgemeine Vermessungs-nachrichten, No. 11-12, 1979, pp. 446-453.

Gao, J. (2007): Development of a precise GPS/INS/on-board vehicle sensors integrated vehicular positioning system, Vol. 68, No. 11.

Gebre-Egziabher, D. (2004): Design and performance analysis of a low-cost aided dead reckoning navigator, Doctoral dissertation, Stanford University).

Gelb, Arthur. (1974): Applied optimal estimation, MIT press.

Giroux, R., Gourdeau, R., & Landry Jr, R. (2005): Inertial navigation system/global positioning system fusion algorithm design in a fast prototyping environment: Towards a real-time implementation. Canadian Aeronautics and Space Journal, 51(3), 133-144.

Giroux, R., Gourdeau, R., & Landry, R. (2005): Extended Kalman filter implementation for lowcost ins/gps integration in a fast prototyping environment, In Symposium on Navigation of the Canadian Navigation Society (pp. 1-11).

Godha, S. (2006): Performance evaluation of low cost MEMS-based IMU integrated with GPS for land vehicle navigation application. Library and Archives Canada= Bibliothèque et Archives Canada.

Grafarend, E., Kleusberg, A., & Schaffrin, B. (1980): An introduction to the variance-covariance component estimation of Helmert type, Zeitschrift für Vermessungswesen, 105(4), 161-180.

Greenspan, R. L. (1996): GPS and inertial integration. Global Positioning System: Theory and applications. 2, 187-220.

Grejner-Brzezinska, D. A. (1997): Airborne Integrated Mapping System- Positioning component. Future of navigation: Facing the challenges, 225-235.

Groves, P. D. (2013): Principles of GNSS, inertial, and multisensor integrated navigation systems, Artech house.

Hasan, A. M., Samsudin, K., Ramli, A. R., Azmir, R. S., & Ismaeel, S. A. (2009): A review of navigation systems (integration and algorithms), Australian journal of basic and applied sciences, 3(2), 943-959.

Hayal, A. G. (2010): Static calibration of the tactical grade inertial measurement units (Doctoral dissertation, The Ohio State University)

Helmert, F. R. (1907): Die Ausgleichungsrechnung nach der Methode der kleinsten Quadrate, Zweite Auflage, Teubner, Lepzig.

Heller, W. G. (1975): Free-Inertial and Damped-Inertial Navigation Mechanization and Error Equations (No. TASC-TR-312-1-1), ANALYTIC SCIENCES CORP READING MA.

Hou, H. (2005): Modeling inertial sensors errors using Allan variance. Library and Archives Canada= Bibliothèque et Archives Canada.

Ignagni, M. B. (1996): Efficient class of optimized coning compensation algorithms. Journal of Guidance, Control, and Dynamics, 19(2), 424-429.

Jaakkola, A., Hyyppä, J., Kukko, A., Yu, X., Kaartinen, H., Lehtomäki, M., & Lin, Y. (2010): A low-cost multi-sensoral mobile mapping system and its feasibility for tree measurements, ISPRS journal of Photogrammetry and Remote Sensing, 65(6), 514-522.

Jekeli, Christopher. (2001): Inertial navigation systems with geodetic applications. Walter de Gruyter.

Joglekar, A. N., and Powell, J. D. (1975): Data Compression in Recursive Estimation with Applications to Navigation Systems. Journal of Aircraft, 12(1), 58-64.

Jones, M. Tim (2008): Anatomy of real-time Linux architectures from soft to hard real-time. IBM developer work, 15.

Julier, S. J., & Uhlmann, J. K. (1997): New extension of the Kalman filter to nonlinear systems, In AeroSense'97 (pp. 182-193), International Society for Optics and Photonics.

Kalman, R. E. (1960): A new approach to linear filtering and prediction problems, Journal of Basic Engineering, 82(1), 35-45.

Kang, C. W., Kang, C. H., & Park, C. G. (2010): Wavelet Denoising Technique for Improvement of the Low Cost MEMS-GPS Integrated System, In International Symposium on GPS/GNSS (p. 2628).

Kaplan, E., & Hegarty, C. (2005): Understanding GPS: principles and applications, Artech House.

Kelly, A. (1994): Modern inertial and satellite navigation systems, No. CMU-RI-TR-94-15, CARNEGIE-MELLON UNIV PITTSBURGH PA ROBOTICS INST.

Koch, K. R. (1986): Maximum likelihood estimate of variance components. Bulletin Gæodésique, 60(4), 329-338.

Kong, X., Nebot, E. M., & Durrant-Whyte, H. (1999): Development of a nonlinear psi-angle model for large misalignment errors and its application in INS alignment and calibration, In Robotics and Automation, 1999, Proceedings of 1999 IEEE International Conference, Vol. 2, pp. 1430-1435.

Kong, X. (2000): Inertial navigation system algorithms for low cost IMU, Department of Mechanical and Mechatronic Engineering, Graduate School of Engineering, University of Sydney.

Kouba, J. T. (1962): Gyrocompass Alignment of an Inertial Platform to Arbitrary Attitudes. ARS Journal, 32(7), 1029-1033.

Kubrak, D. (2007): Hybridisation of a GPS Receiver with Low-Cost Sensors for Personal Positioning in Urban Environment (Doctoral dissertation, Télécom ParisTech).

Kuipers, J. B. (1999): Quaternions and rotation sequences (Vol. 66). Princeton: Princeton university press.

Lansdorp, B. M., & Saleh, O. A. (2012): Power spectrum and Allan variance methods for calibrating single-molecule video-tracking instruments, Review of Scientific Instruments, 83(2), 025115.

Lawrence, A. (2012): Modern inertial technology: navigation, guidance, and control, Springer Science & Business Media.

Lei, Ming, Barend J. van Wyk, and Yong Qi (2011): "Online estimation of the approximate posterior cramer-rao lower bound for discrete-time nonlinear filtering." IEEE Transactions on Aerospace and Electronic Systems 47.1 (2011): 37-57.

Li, Binghao (2004): A cost effective synchronization system for multisensor integration. Proceedings of the 17th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2004), Long Beach, California, U.S.A. 2004, 9, pp. 1627-1635.

Li, X.R. and Jilkov, V.P. (2000): A Survey of Manoeuvring Target Tracking: Dynamic Models, IEEE Transactions on Aerospace and Electronic Systems, Vol. 39, No. 4, October 2003, pp. 1333-1364.

Liu, Zhenkun & Huang, Shunji (2009): GPS dynamic cycle slip detection and correction with baseline constraint, Journal of Systems Engineering and Electronics, 20(1), 60-64.

Luo, X., Mayer, M. and Heck, B. (2009): Improving the stochastic model of GNSS observations by means of SNR-based weighting, In Observing our Changing Earth, pp. 725-734, Springer Berlin Heidelberg

Ma, L. H., Wang, K. L., & Li, H. (2013): Gyrocompass alignment method of sins based on Kalman filtering pretreatment and dynamic gain adjustment on a rocking base, Information Technology Journal, 12(4), 777.

Ma, L., Li, Z., & Wu, T. (2011): A coning compensation algorithm for SINS in high dynamic motion. Journal of Control Engineering and Applied Informatics, 13(3), 32-40.

Ma, Xin, Salah Sukkarieh, and Jong-Hyuk Kim (2003): "Vehicle model aided inertial navigation", Intelligent Transportation Systems, 2003, Proceedings of 2003 IEEE, Vol. 2.

Magnus, Kurt (1971): Kreisel: Theorie und Anwendungen, Springer Berlin, 1971.

Matisko, P., & Havlena, V. (2013): Noise covariance estimation for Kalman filter tuning using Bayesian approach and Monte Carlo. International Journal of Adaptive Control and Signal Processing, 27(11), 957-973.

Maybeck, P. S. (1973): Wander azimuth implementation algorithm for a strapdown inertial system (No. AFFDL-TR-73-80). AIR FORCE FLIGHT DYNAMICS LAB WRIGHT-PATTERSON AFB OH.

Maybeck, P. S. (1994): Stochastic models, estimation, and control, Vol. 1 of Mathematics in Science and Engineering.

Mehra, R. (1970): On the identification of variances and adaptive Kalman filtering. IEEE Transactions on automatic control, 15(2), 175-184.

Mehra, R. (1972): Approaches to adaptive filtering. IEEE Transactions on automatic control, 17(5), 693-698.

Munguía, R. (2014): A GPS-aided inertial navigation system in direct configuration, Journal of Applied Research and Technology, Vol. 12, August 2014, pp. 803-814.

Munguía, R., Nuño, E., Aldana, C. I., & Urzua, S. (2016). A visual-aided inertial navigation and mapping system. International Journal of Advanced Robotic Systems, 13(3), 94.

Murphy, W., & Hereman, W. (1995): Determination of a position in three dimensions using trilateration and approximate distances, Department of Mathematical and Computer Sciences, Colorado School of Mines, Golden, Colorado, MCS-95, 7, 19.

Nassar, S., SCHWARZ, K. P., EL -SHEIMY, N. A. (2005): Modeling inertial sensor errors using autoregressive (AR) models. Navigation, 51(4), 259-268.

Nassar, S. (2005): Improving the inertial navigation system (INS) error model for INS and INS/DGPS applications. National Library of Canada= Bibliothèque nationale du Canada.

Navidi, W., Murphy, W. S., & Hereman, W. (1998): Statistical methods in surveying by trilateration, Computational statistics & data analysis, 27(2), 209-227.

Navigation. In Merriam-Webster's collegiate dictionary. Retrieved from http://www.merriam-webster.com/dictionary/navigation.

Nilesh S. Gopaul, Jianguo Wang and Jiming Guo (2010): On Posteriori Variance and Covariance Components Estimation in GPS Relative Positioning, Proceedings of CPGPS 2010 Navigation and Location Services: Emerging Industry and International Exchanges, pp. 141-148.

Nilsson, J. O., & Skog, I. (2016, May). Inertial sensor arrays—A literature review. In Navigation Conference (ENC), 2016 European (pp. 1-10). IEEE.

Newton, I., Motte, A., & Cajori, F. (1987): Mathematical principles of natural philosophy (pp. xvii-xvii). W. Benton: Encyclopaedia Britannica.

Noureldin, A., Karamat, T. B., & Georgy, J. (2012): Fundamentals of inertial navigation, satellite-based positioning and their integration, Springer Science & Business Media.

Odelson, B. J., Rajamani, M. R., & Rawlings, J. B. (2006): A new autocovariance least-squares method for estimating noise covariances. Automatica, 42(2), 303-308.

Ou, Ziqiang (1989): Estimation of Variance and Covariance Components, Bulletin Géodésique, Vol. 63, Issue 3 (1989), pp. 139-148

Park, M. (2005): Error analysis and stochastic modeling of MEMS-based inertial sensors for land vehicle navigation applications. Library and Archives Canada= Bibliothèque et Archives Canada.

Patrick Mourot (Alcatel, France) (2011): RTAI internals presentation, http://www.aero.polimi.it/~rtai/documentation/articles/patric_mourot_rtai_internal_presentation.html (accessed on Oct, 10, 2011).

Perlmutter, M., & Robin, L. (2012): High-performance, low cost inertial MEMS: A market in motion. In Position Location and Navigation Symposium (PLANS), 2012 IEEE/ION, pp. 225-229.

Point Grey Research Inc. (2011): Flea3 FL3-FW Technical Reference Manual, 2011.

Punzo, V., Borzacchiello, M. T., & Ciuffo, B. (2011): On the assessment of vehicle trajectory data accuracy and application to the Next Generation SIMulation (NGSIM) program data. Transportation Research Part C: Emerging Technologies, 19(6), 1243-1262.

Qi, Honghui and Moore, John B. (2002): Direct Kalman filtering Approach for GPS/INS Integration, IEEE Transactions on Aerospace and Electronic Systems, Vol. 38, No. 2 (2002), pp. 687-693.

Qian, Kun; Wang, Jianguo; Gopaul, Nilesh. and Hu, Baoxin. (2012): Low cost multisensor kinematic positioning and navigation system with Linux/RTAI, Journal of Sensor and Actuator Networks, vol. 1, issue 3, 2012, pp. 166-182.

Qian, Kun; Wang, Jianguo. and Hu, Baoxin (2013): Application of vehicle kinematic model on GPS/ MEMS IMU integration, Joint EOGC 2013 & CIG Annual Conference, June 5-7, 2013, Toronto.

Qian, Kun; Wang, Jianguo. and Hu, Baoxin. (2015): “Novel integration strategy for GNSS-aided inertial integrated navigation”, Geomatica, 2015, 2, Vol. 69, pp. 217-230.

Qian, Kun; Wang, Jianguo and Hu, Baoxin (2016): “A posteriori estimation of stochastic model for multisensor integrated inertial kinematic positioning and navigation on basis of variance component estimation”, Journal of GPS, 2016.

Qing, Yongyuan (2006): Inertial navigation (Chinese). Beijing: Science Press.

Quinchia, A. G., Falco, G., Falletti, E., Dovis, F., & Ferrer, C. (2013): A comparison between different error modeling of MEMS applied to GPS/INS integrated systems, Sensors, 13(8), 9549-9588.

Raj Kamal (2008): Embedded systems – Architecture, Programming and Design, McGraw-Hill.

RealTime Application Interface for Linux (2013): RTAI API documentation https://www.rtai.org/documentation/magma/html/api/ (accessed on Sep, 1, 2015).

Rietdorf, Andreas (2004): Automatisierte Auswertung und Kalibrierung von scanneden Messsystemen mit tachy-metrischen Messprinzip, PhD dissertation, Civil Engineering and Applied Geosciences, Technical University Berlin, 2005.

Rogers, Robert M. (2003): Applied mathematics in integrated navigation systems (Vol. 1). AIAA.

Salzmann, Martin (1993): Least Squares Filtering and Testing for Geodetic Navigation Applications , Number 37, Publication on Geodesy, Netherlands Geodetic Commission, Delft, 1993.

Savage, P. G. (1998): Strapdown inertial navigation integration algorithm design part 2: Velocity and position algorithms. Journal of Guidance, Control, and Dynamics, 21(2), 208-221.

Scherzinger, B. M. (2000): Precise robust positioning with inertial/GPS RTK. In Proceedings of the 13th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GPS) (pp. 115-162).

Schimelevich, L., & Naor, R. (1996): New approach to coarse alignment. In Position Location and Navigation Symposium, 1996., IEEE 1996 (pp. 324-327).

Schmidt, S. F., Bjorkman, W. S., & Conrad, B. (1973): New mechanization equations for aided inertial navigation systems.

Schmidt, G. T. (2010): INS/GPS technology trends, MASSACHUSETTS INST OF TECH LEXINGTON MA.

Schwarz, K. P., Martell, H. E., El-Sheimy, N., Li, R., Chapman, M. A., & Cosandier, D. (1993): VIASAT-A mobile highway survey system of high accuracy. In Vehicle Navigation and Information Systems Conference, 1993., Proceedings of the IEEE-IEE (pp. 476-481).

Shin, E. H., & El-Sheimy, N. (2003): Accuracy improvement of low cost INS/GPS for land applications. National Library of Canada= Bibliothèque nationale du Canada.

Shin, E. H., & El-Sheimy, N. (2003): Backward smoothing for pipeline surveying applications. Proceedings of the 2005 National Technical Meeting of The Institute of Navigation. 2001.

Shuster, M. D. (1993): A survey of attitude representations. Navigation, 8(9), 439-517.

Sieg, Detlef and Hirsch, Milo (2000): Varianzkomponenten-schätzung in ingenieurgeodätischen Netzen, AVN, No. 3, 2000, pp. 82-90.

Silson, P. M. (2011): Coarse alignment of a ship's strapdown inertial attitude reference system using velocity. IEEE Transactions on Instrumentation and Measurement, 60(6), 1930-1941.

Simon, D. (2006): Optimal state estimation: Kalman, H infinity, and nonlinear approaches, John Wiley & Sons.

Singer, R.A. (1970): Estimating Optimal Tracking Filter Performance for Manned Manoeuvring Targets, IEEE Transactions on Aerospace and Electronic Systems, Vol. 6, No. 4, July 1970, pp. 473-483.

Sitaraa, B., Chandrasekhar, R. S., Dadub, V., & Reddya, G. S. (2007): Analytical Gyrocompass Alignment of Strapdown INS on a Stationary Base, Sensors 2007, National Conference for Aerospcae Applications, December 2007

Skaloud, J. (2002): Direct georeferencing in aerial photogrammetric mapping. Photogrammetric Engineering and Remote Sensing, 68(3)

Skog, I., & Händel, P. (2006): Calibration of a MEMS inertial measurement unit, In XVII IMEKO World Congress (pp. 1-6).

Skog, I., Nilsson, J. O., Händel, P., & Nehorai, A. (2016): Inertial Sensor Arrays, Maximum Likelihood, and Cramér–Rao Bound. IEEE Transactions on Signal Processing, 64(16), 4218-4227.

Slabaugh, G. G. (1999): Computing Euler angles from a rotation matrix. Retrieved on August, 6(2000), 39-63.

Slavov, T., & Petkov, P. (2011): Strapdown Inertial System Based on Improved MEMS Error Models. Cybernetics and Information Technologies, 11(4), 3-23.

Soetens, Peter. (2004): How to port your C++ GNU/Linux application to RTAI/LXRT, http://www.isr.uc.pt/~rui/str/rtai_porting.pdf (accessed on Feb, 27, 2012).

Stein, G., Chmúrny, R., & Rosík, V. (2011): Compact vibration measuring system for in-vehicle applications. Measurement Science Review, 11(5), 154-159.

Stein, S. R. (1985): 12 Frequency and Time-Their Measurement and Characterization.

Stöhr, M. (1986), Der Kalman–Filter und seine Fehlerprozeße unter besonderer Berücksichtigung der Auswirkung von Modellfehlern, Forschung-Ausbildung-Weiterbildung Bericht Heft 19, Universität Kaiserlautern, Fachbereich Mathematik, August 1986.

Sun, F., Lan, H., Yu, C., El-Sheimy, N., Zhou, G., Cao, T., & Liu, H. (2013): A robust self-alignment method for ship's strapdown INS under mooring conditions, Sensors, 13(7), 8103-8139.

Sun, H., Fu, J., Yuan, X., & Tang, W. (2008): Analysis of the Kalman Filter with Different INS Error Models for GPS/INS Integration in Aerial Remote Sensing Applications. The International Archives of the Phogrammetry, Remote Sensing and Spatial Information Sciences, 37, 883-890.

Syed, Z. F., Aggarwal, P., Goodall, C., Niu, X., & El-Sheimy, N. (2007): A new multi-position calibration method for MEMS inertial navigation systems, Measurement Science and Technology, 18(7), 1897.

Talaya López, J. (2003): Algorithms and Methods for Robust Geodetic kinematic Positioning.

Takasu, T. (2013): RTKLIB ver. 2.4.2 Manual. http://www.rtklib.com/

Tanizaki, H. (2013): Nonlinear filters: estimation and applications, Springer Science & Business Media.

Tao, C. V. (2000): Mobile mapping technology for road network data acquisition. Journal of Geospatial Engineering, 2(2), 1-14.

Tao, C. V. (2007): Advances in mobile mapping technology: Volume 4 of International Society for Photogrammetry and Remote Sensing book series, Taylor & Francis, ISBN 0-415-42723-1.

Tay, S. and Marais, J. (2013): Weighting models for GPS Pseudorange observations for land transportation in urban canyons, 6th European Workshop on GNSS Signals and Signal Processing (p. 4p).

Taylor, J. H. (1978): The Cramer-Rao estimation error lower bound computation for deterministic nonlinear systems, IEEE Conference on Decision and Control including the 17th Symposium on Adaptive Processes, 1978.

Tesmer, Volker (2004): Das stochastische Modell bei der VLBI-auswertung, PhD dissertation, No. 573, Reihe C, DGK, Munich, 2004.

Teunissen, P. J. (1995). The least-squares ambiguity decorrelation adjustment: a method for fast GPS integer ambiguity estimation, Journal of geodesy, 70(1-2), 65-82.

Teunissen, P. J. G., & Amiri-Simkooei, A. R. (2008): Least-squares variance component estimation. Journal of Geodesy, 82(2), 65-82.

Tiberius, Christian and Kenselaar, Frank (2003): Variance component estimation and precise GPS positioning: case study, Journal of Surveying Engineering, Vol. 129, No. 1 (2003), pp. 11-18.

Titterton, D., & Weston, J. L. (2004): Strapdown inertial navigation technology, 2nd edition, Co-published by the Institution of Engineering and Technology, London, UK and the American Institute of Aeronautics, Reston, Virginia, USA, 2004.

Unsal, D., & Demirbas, K. (2012): Estimation of deterministic and stochastic IMU error parameters. In Position Location and Navigation Symposium (PLANS), 2012 IEEE/ION,pp. 862-868.

Vasispacher, T., Bréda, R., & Madarász, L. (2015): Integration architectures of navigation systems for unmanned vehicles. In Applied Machine Intelligence and Informatics (SAMI), 2015 IEEE 13th International Symposium, pp. 97-101.

Vathsal, S. (1991): Derivation of the relative quaternion differential equation. Journal of Guidance, Control, and Dynamics, 14(5), 1061-1064.

Wan, D., & Fang, J. C. (1998): Initial alignment of inertial navigation. Nanjing: Press of Southeast University.

Wang, C. (2003): Development of a low-cost GPS-based attitude determination system, University of Calgary.

Wang, M., Wu, W., Wang, J., & Pan, X. (2015): High-order attitude compensation in coning and rotation coexisting environment. IEEE Transactions on Aerospace and Electronic Systems, 51(2), 1178-1190.

Wang, Jianguo (1997): Filtermethoden zur fehlertoleranten kinematischen Positionsbestimmung, PhD dissertation, Schriftenreihe Studiengang Vermessungswesen, UniBw München, No. 52, 1997.

Wang, Jianguo (2008): Test Statistics in Kalman Filtering, Vol. 7, No. 1, Journal of GPS, pp. 81-90.

Wang, Jianguo (2009): Reliability Analysis in Kalman Filtering, J. of Global Positioning Systems, Vol. 8, No. 1(2009), pp. 101-111.

Wang, Jianguo and Sternberg, Harald (2000): Model development for kinematic surveying of land vehicle trajectories (in German), Schriftenreihe Studiengang Vermessungswesen UinBw München, No. 60-1, ISSN 0179-1009, Germany, 2000, pp. 317-331.

Wang, Jianguo; Qian, Kun and Hu, Baoxin (2014): A novel and unique IMU/GNSS Kalman filter, Keynote Speaker of Session Invited Presentation (J. Wang), China Satellite Navigation Conference 2014, May 21-23, 2014, Nanjing, China.

Wang, Jianguo; Qian, Kun and Hu, Baoxin (2015): An Unconventional Full Tightly-Coupled Multi-Sensor Integration for Kinematic Positioning and Navigation, Chapter 65, in J. Sun et al. (eds), China Satellite Navigation Conference (CSNC) 2015 Proceedings: Volume III, Lecture Notes in Electrical Engineering 342, Springer-Verlag Berlin Heidelberg 2015.

Wang, Jinling and Rizos, Chris (2002): Stochastic Assessment of GPS Carrier Phase Measurements for Precise Static Relative Positioning, J. of Geodesy, Vol. 76, No. 2.

Wagner, J. F., & Wieneke, T. (2003): Integrating satellite and inertial navigation—conventional and new fusion approaches. Control Engineering Practice, 11(5), 543-550.

Weinberg, H. (2011): Gyro mechanical performance: The most important parameter, Technical Article MS-2158. Analog Devices, Inc, 1-5.

Wendel, J., Schlaile, C., & Trommer, G. F. (2001): Direct Kalman filtering of GPS/INS for aerospace applications, In International Symposium on Kinematic Systems in Geodesy, Geomatics and Navigation (KIS2001).

Wendel, Jan, Trommer, Gert F., "Impact of Mechanical Vibrations on the Performance of Integrated Navigation Systems and On Optimal IMU Specification," Proceedings of the 57th Annual Meeting of The Institute of Navigation (2001), Albuquerque, NM, June 2001, pp. 614-621.

Wieser, A., Gaggl, M. and Hartinger, H. (2005): Improved positioning accuracy with high sensitivity GNSS receivers and SNR aided integrity monitoring of pseudo-range observations, Proc ION GNSS, pp. 13-16.

Wu, D. J., Xiong, W., Zhou, R. X., & He, C. J. (2013): Designing and realization of GNSS continuously operating reference stations system of Hong Kong-Zhuhai-Macao bridge. Science of Surveying and Mapping, 38(2), 62-64.

Woodman, Oliver J. (2007): An introduction to inertial navigation. University of Cambridge, Computer Laboratory, Tech. Rep. UCAMCL-TR-696 14 (2007): 15.

Xiao, G. and Wujiao, D. (2014): The application of robust Helmert varaince component estimationin GPS/BDS integrated navigation (抗差 Helmert 方差分量估计在 GPS/BDS 组合定位中的应用). Geodesy and Geodynamics (大地测量与地球动力学), 34(1), pp. 173-176.

Xu, P., Shen, Y., Fukuda, Y., & Liu, Y. (2006): Variance component estimation in linear inverse ill-posed models. Journal of Geodesy, 80(2), 69-81.

Yi, Yudan (2007): On improving the accuracy and reliability of GPS/INS-based direct sensor georeferencing, Ohio State University, Division of Geodetic Science.

Yoon Sang Won. (2009): Vibration Isolation and Shock Protection for MEMS. ProQuest, 2009

Zhang, S. B., Li, X. C., & Su, Z. (2015): Cone algorithm of spinning vehicles under dynamic coning environment, International Journal of Aerospace Engineering, 2015.

Zhang, X., Li, Y., Mumford, P., & Rizos, C. (2008): Allan variance analysis on error characters of MEMS inertial sensors for an FPGA-based GPS/INS system, in Proceedings of the International Symposium on GPS/GNNS (pp. 127-133).

Zhou, H. and Kumar, K.S.P. (1984): A “current” statistical model and adaptive algorithm for estimating maneuvering targets, Journal of Guidance, Control, and Dynamics, 1984, 7(5), pp. 596-602.

Zhou, X. W., Dai, W. J., Zhu, J. J., Li, Z. W., & Zou, Z. R. (2008): Helmert variance component estimation-based vondrak filter and its application in GPS multipath error mitigation, in VI Hotine-Marussi Symposium on Theoretical and Computational Geodesy (pp. 287-292), Springer Berlin Heidelberg.