DESIGN AND ANALYSIS OF TRANSFER ALIGNMENT ALGORITHMS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

YİĞİTER YÜKSEL

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
ELECTRICAL AND ELECTRONICS ENGINEERING

FEBRUARY 2005

Approval of the Graduate School of Natural and Applied Sciences

Prof. Dr. Canan Özgen
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.

Prof. Dr. İsmet Erkmen
Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

Prof. Dr. Mübeccel Demirekler
Supervisor

Examining Committee Members

Prof. Dr. Kemal Leblebicioğlu (Chairman), (METU, EE)

Prof. Dr. Mübeccel Demirekler, (METU, EE)

Prof. Dr. Aydan Erkmen, (METU, EE)

Assist. Prof. Dr. Yakup Özkazanç, (Hacettepe Unv., EE)

Dr. Ayşe Pınar Koyaz, (TÜBİTAK – SAGE)

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Yiğiter YÜKSEL

ABSTRACT

DESIGN AND ANALYSIS OF TRANSFER ALIGNMENT ALGORITHMS

YÜKSEL, Yiğiter
M.S., Department of Electrical and Electronics Engineering
Supervisor : Prof. Dr. Mübeccel Demirekler

February 2005, 169 Pages

Transfer Alignment is the process of simultaneously initializing and calibrating a weapon inertial navigation system (INS) using data from host aircraft's navigation system. In general, this process is accomplished by calculating the difference of navigation solutions between aircraft and weapon INSs to form observations which are then used in a Kalman filter to generate desired estimates. Numerous techniques about the problem of transfer alignment exist in the literature. In this thesis, those techniques that can be applied in the presence of elastic motion of aircraft wing were analyzed. Several transfer alignment algorithms each of which process different measurement types were designed and implemented. In order to evaluate the performance of implemented algorithms under realistic conditions, a transfer alignment simulation environment was developed. Using this simulation environment, the advantages and disadvantages of each algorithm were analyzed and the dependence of transfer alignment performance on Kalman filter system model, aircraft maneuvers and alignment duration were investigated.

Keywords: Transfer Alignment, In-motion Alignment, Inertial Navigation

ÖZ

YÖNELİM AKTARIMI ALGORİTMALARININ TASARIM VE ANALİZİ

YÜKSEL, Yiğiter
Yüksek Lisans., Elektrik ve Elektronik Mühendisliği Bölümü
Tez Yöneticisi : Prof. Dr. Mübeccel Demirekler

Şubat 2005, 169 Sayfa

Yönelim aktarımı, uçağın seyrüsefer sistemi çıktılarını kullanarak mühimmat üzerindeki ataletsel seyrüsefer sistemine ilk değer atama ve sistemin kalibrasyonunun eş zamanlı yapılması işlemidir. Genel olarak bu işlem, uçağın ve mühimmatın seyrüsefer sonuçları arasındaki farktan bir ölçüm oluşturulması ve bu ölçümün Kalman süzgecinde işlenmesiyle gerçekleştirilir. Konuyla ilgili kaynaklarda, aktarım yönlendirme ile ilgili birçok teknik ortaya konmuştur. Bu tezde, uçak kanadının elastiki hareketinin mevcut olduğu durumlarda uygulanabilecek aktarım yönlendirme teknikleri incelenmiştir. Farklı ölçüm çeşitlerini işleyebilen birçok Kalman filtresi tasarlanmıştır. Tasarlanan Kalman filtrelerinin performanslarını gerçekçi bir ortamda sınayabilmek için, bir yönelim aktarımı benzetim ortamı geliştirilmiştir. Geliştirilen benzetim ortamı kullanılarak, her bir yönelim aktarımı algoritmasının başarımı analiz edilmiş ve Kalman filtresi sistem ve ölçüm modeli, uçak manevrası ve algoritma uygulama süresinin yönelim aktarımı algoritmaları performansı üzerindeki etkileri incelenmiştir.

Anahtar Kelimeler: Yönelim Aktarımı, Ataletsel Seyrüsefer, Hareket Halinde İlklendirme

To my mother and father who still love me despite of my endless peevishness.

ACKNOWLEGMENTS

I am greatly thankful to my supervisor Prof. Dr. Mübeccel Demirekler for her guidance throughout the preparation of this thesis. Thanks to her advices and appreciated criticisms, I had the chance of improving my studies.

I am also grateful to Uğur Topay, Burak Kaygısız, Yüksel Subaşı and Ayhan Erdim for their help and friendship during this period.

I would like to express my sincere thanks to my family for their support and understanding.

Finally, I would like to thank my brother. Even his sole existence makes me feel more confident.

TÜBİTAK-SAGE who supported this work is greatly acknowledged.

TABLE OF CONTENTS

LIST OF SYMBOLS

L Latitude

ℓ Longtitude

h Height

V_e^n Velocity of the system with respect to Earth defined in NED frame

V_N North component of V_e^n

V_E East component of V_e^n

V_D Down component of V_e^n

v_x^y Velocity of the system with respect x defined in y. x, y can be one of the i,b,e and n where “n” refers to NED frame, “e” refers to ECEF frame, “i” refers to inertial frame and “b” refers to body frame of reference. If x is not explicitly indicated, then velocity is defined with respect to earth frame. If neither of x nor y written explicitly then, v represents “V_e^n”.

C_x^y Direction cosine matrix that transforms x to y where x and y can be equal to one of the n,e,i,b. “n” refers to NED frame, “e” refers to ECEF frame, “i” refers to inertial frame and “b” refers to body frame of reference. Instead of “b”, “M” or “S” can be used which corresponds to body frame defined for Master or Slave system respectively.

$S(X)$ A vector to matrix operator that transforms vector $X = [x_1, x_2, x_3]^T$ into a skew symmetric matrix

$$s(X) = \begin{vmatrix} 0 & -x_3 & x_2 \\ x_3 & 0 & -x_1 \\ -x_2 & x_1 & 0 \end{vmatrix}$$

ω_{xy}^z Rotation rate of y with respect to x defined in z. x,y,z can be any one of the i, b, e, n where “i” represents inertial frame, “b” represents body frame, “e” represents ECEF frame and “n” represents NED frame. Instead of b, “M” or “S” can be used to refer body frame of Master and Slave systems

Ω Rotation rate of Earth (15°/hour).

g Local gravity vector defined in WGS 84

a_X^Y Acceleration of the system X with respect to inertial frame of reference defined in Y frame of reference. X can be M or S or null, where M refers Master and S refers Slave systems. Y can be “i” or null, where “i” refers to inertial frame of reference. If neither “x” nor “y” is written explicitly, then a represents acceleration of the system with respect to inertial frame of reference defined in body frame of reference.

R_N Meridian radius of curvature defined in WGS84

R_E Transverse radius of curvature defined in WGS84

R Length of semi major axis of Earth

$\tilde{X}$ Actual / Observed value of X.

δX Instantaneous error on X which is equal to “ $\tilde{X} - X$ ”

δa_{BR} Accelerometer bias repeatability error

δa_{SFR} Accelerometer scale factor repeatability error

δa_{BS} Accelerometer bias instability error

δa_{SFS} Accelerometer scale factor instability error

δa_{RW} Accelerometer random walk error

$\delta f^{\,b}$, δf Total error on accelerometer outputs

δg_{DR} Gyroscope drift repeatability error

δg_{SFR} Gyroscope scale factor repeatability error

δg_{RW} Gyroscope (angular) random walk error.

$\delta\omega_{ib}^{b}$, $\delta\omega$ Total error on gyroscope outputs

φ Euler angles which define the orientation of erroneous navigation Frame with respect to true navigation frame defined on erroneous frame.

$\delta\omega_{in}^{n}$ The sum of earth rotation rate and transport rate errors calculated in NED frame which is equal to "$\delta\omega_{en}^{n} + \delta\omega_{ie}^{n}$"

ϕ, θ, ψ Euler angles which defines the rotation from body to NED frame. Rotation order is z, y and x.

r^{x} Lever arm vector defined in x coordinate frame. If x is not written explicitly, then r is defined in body frame.

r_{Stat} Static component of lever arm. ($\dot{r}_{Stat} = 0$)

r_{Dyn} Dynamic component of lever arm. $\dot{r}_{Dyn} = vib_{vel}$

δr_{X} Total error in "X" component of lever arm vector. If no x is specified, then it represents total error on lever arm vector

R_{X}^{Y} Position vector of the system X, defined in Y frame of coordinate. "x" can be M or S where "M" refers master and "S" refers slave

V_{X}^{Y} Velocity of system "x" with respect to Earth defined in "y" frame of reference. "x" can be "M" or "S". If no "y" is defined than y=x.

ω_{n} Natural frequency of the isolator

ξ Damping coefficient of the isolator

$\mathrm{E} = [\alpha, \beta, \gamma]^{T}$ Euler angles which defines master to slave transformation

e , e_{Tot} Small Euler angles which define the transformation from erroneous slave coordinate frame to true frame defined on erroneous frame. (Error on "E")

e_{Stat} Static component of e

e_{Flex} The component of e which arise from flexure of wing structure

e_{Vib} The component of e which arise from vibration of wing structure

$C(X)$ Direction Cosine Matrix computed using Euler angles X

υ Kalman Filter measurement noise

z_{Obs} Observation used in Kalman filter

z_{Meas} Measurement equation of observations defined in terms of states used in Kalman filter

$\mathrm{E}[xx^{T}]$ Correlation of 'x'

α_{comp} Compensated angle increment in 1/100 sec.

$\alpha_{n}(t_{n}, t)$ Uncompensated angle increment between time t_n and t

$\alpha_n(m)$ Uncompensated angle increment in 1/600 sec. (mth raw gyroscope output in nth minor interval)

$\delta\alpha(t_n, t)$ Coning compensation between time t$_n$ and t

$\delta\alpha_n$ Coning compensation in 1/200 seconds computed in nth interval

$\phi(t_n, t)$ Rotation vector which defines the orientation of a frame at time “t” with respect to time “t$_n$”

LIST OF ABBREVATIONS

PVA	Position, Velocity and Attitude
IMU	Inertial Measurement Unit
INS	(Strapdown) Inertial Navigation System
GPS	Global Positioning System
TA	Transfer Alignment (Transfer of Alignment)
NED Frame	North, East, Down frame of reference
ECEF	Earth Centered Earth Fixed frame of reference
DCM	Direction Cosine Matrix
WGS 84	1984 Word Geodetic System
MEMS	Micro Electro-Mechanical Systems
LA	Lever Arm
MC	Monte Carlo
PSD	Power Spectral Density
SD	Standard Deviation
JDAM	Joint Direct Attack Munitions
RHS	Right Hand Side
PSD	Power Spectral Density
VM	Velocity Match Method
AVM	Attitude and Velocity Match Method
AM	Attitude Match Method
IVM	Integrated Velocity Match Method
SF	Scale Factor
CG	Center of Gravity

CHAPTER I

INTRODUCTION

Inertial navigation is the process of calculating position, velocity and attitude (PVA) of the system by continuously integrating system acceleration and rotation rate measured by inertial measurement unit (IMU) (Titterton (1997), Savage (2000)).

Inertial navigation systems (INS) can be classified into two groups according to their IMU placements. In strapdown inertial navigation systems, IMU is directly mounted on the vehicle rather than on a gimbaled platform which is used to isolate IMU from vehicle rotations. Currently, strapdown systems are used in almost every kind of navigation process (Savage (2000)). Therefore, in the scope of this study, only strapdown inertial navigation systems are considered.

For an inertial navigation system to start operation, an initial PVA solution has to be supplied to the system. Any error in the initial PVA solution causes the navigation system to produce erroneous outputs as soon as system starts operation. Furthermore, due to the structure of strapdown algorithms, these initialization errors cause the navigation errors to grow so fast that, in a very short period of time, the navigator outputs become totally useless.

Therefore, in the past, the initialization problem of inertial navigators was studied extensively. As initial attitude errors cause the navigation errors to increase much rapidly than initial velocity and position errors, studies was especially focused on determining correct initial attitude of the system.

In the literature, solution methods to initialization problems are classified in to two groups according to the system's being in motion or stationary during initialization process (Savage (2000), Titterton (1997)). These groups are called as;

i. Ground Alignment

ii. In Motion Alignment / Transfer Alignment

Ground alignment is the initialization of the system during which the system is stationary. In this method, initial velocity is taken to be zero without considering the environmental disturbances. Initial position is provided by some external aiding such as GPS or manually by means of a known position. To initialize attitude, several different techniques can be implemented, among which gyrocompass is the most preferred method. In the gyrocompass technique, horizontal attitude is determined by comparing local gravity vector with accelerometer outputs, and azimuth angle is calculated by comparing the Earth's rotation rate with the gyroscope outputs. On the other hand, as the accuracy of initial attitude obtained by this technique is insufficient due to the IMU errors, the initial attitude solution should be revised by employing additional estimation techniques such as zero velocity or known position updates (Savage (2000)).

In-motion alignment is the common name of initialization methods which are employed during the motion of the system. In this case, as the system undergoes an arbitrary movement during initialization, it is not possible to assign initial values without an aid of some external reference source. Therefore, the essential part of in-motion alignment is the process of transfer of navigation information (especially attitude information) from a reference source to the navigation system. Hence, these techniques are also called as "Transfer Alignment" (TA) in order to emphasize the basic property of transfer of alignment information.

There are lots of situations where the need for transfer alignment occurs. Especially, when a small system with an inertial navigator is to be carried and deployed by an aircraft, the initialization of small system's navigator is accomplished by utilizing transfer alignment algorithms. A typical example of this is the JDAM (Joint Direct Attack Munitions) type guided munitions (Klotz (1998)). Prior to the launch of the ammunition, a transfer alignment procedure is utilized in order to initialize the navigation system and calibrate the inertial measurement unit of the ammunition's navigator. At the end of this transfer alignment process, the ammunition's inertial navigator become ready to operate for sufficiently long time without a need for any external aiding.

From the papers written in the mid-60s, it can be inferred that, initial studies on transfer alignment problem were started during the NASA's space mission programs. In order to initialize service module's inertial navigator system, main INS outputs (the INS that is used throughout boost powered flight) were need to be processed by some form of transfer alignment algorithms (Baziw

(1972)). Later efforts on this subject were mainly focused on utilization of transfer alignment algorithms on aircraft's avionics systems. To transfer information between different sensors that are located apart from each other, the relative orientation of each sensor cluster must be determined which is also accomplished by implementation of transfer alignment procedures (Harris (1977)).

With the reduction of cost of good quality IMUs after 80s, almost every kind of guided weapon was utilized with some form of inertial navigator. At first, inertial navigators were implemented to meet the requirements of long range mid course guidance phase such as in Penguin missiles (Hallingstad (1989)). However, with the increase of low cost IMU performance, it became possible to use inertial navigators as the main source of guidance information including terminal guidance phase. JDAM is one of the typical examples, which guarantees success of its mission with its inertial navigator outputs only.

Today, not having an electronic counter measure, inertial navigators became an indispensable part of modern guided munitions. Hence, it is increasingly necessary to be able to design low cost INSs which operate longer durations and more accurately without any aid from external sources. To achieve these aims within the allowed cost limits, it is essential to estimate and correct (calibrate) any error that effects the INS performance before the system starts to operate. This essential task is also accomplished by transfer alignment algorithms. Therefore, although the initial motivation in the development of transfer alignment algorithms is to just initialize inertial navigator systems, today, a successful algorithm must be able to accomplish calibration of IMU during the motion of the system as well.

In the literature, several methods have been proposed for the solution of transfer alignment problem. The diversity of the solutions stems from the fact that, different arguments are considered as the primary point of interest in the design of algorithms. Some solutions focus on decreasing the total duration required to complete transfer alignment procedures, whereas others may focus on increasing the final estimation accuracy. On the other hand, regardless of whatever the main objective is, all algorithms depend on the method of comparing two or more common vectors obtained from master and slave navigation systems (there are also some primitive methods which depend on mechanical structures such as optical bore sight etc, however in the scope of this study these methods are completely ignored). In master/slave configurations, master refers to the system which provides reference navigation values such as velocity, attitude, acceleration and slave refers to the system to be aligned and calibrated. The reference values obtained from master system are usually assumed to be almost errorless. Hence any difference between master and slave system navigation outputs is a measurement of error that exists in the slave system. By processing these measurements, transfer alignment algorithms can generate estimates about navigation errors and IMU calibration parameters of slave navigator.

Theoretically all kind of navigation outputs such as velocity, attitude position, acceleration, and rotation rate can be used for comparison in transfer alignment algorithms as long as those values can be obtained both from master and slave systems. Therefore, transfer alignment algorithms are classified and named according to the type of vectors compared to form measurements. For instance, “velocity matching technique” denotes the class of transfer alignment algorithms which process measurements formed using velocity differences and “attitude matching” refers to algorithms which use attitude differences, etc. In addition to these methods, it is also possible to calculate measurements by comparing more than one vector at the same time such as in “velocity and attitude matching” method (Spalding, 1992), (Shortelle 1995)) or by using preprocessed form of basic vector types like in “integrated velocity matching” method (Stovall (1996), (Ross, 1994), (Kain, 1989)).

1.1 THESIS OBJECTIVES

One of the main objectives of this study is to provide both qualitative and quantitative comparison between different transfer alignment algorithms. As explained above, several transfer alignment algorithms can be developed by using different measurement structures. Analyzing the performance of these structures under different conditions, it was aimed to explore the advantages and disadvantages of each method with respect to each other.

In this study, the transfer alignment algorithms were designed and evaluated by considering air to ground guided munitions applications. Therefore, it was assumed that aircraft’s main navigator serves as master navigation system and ammunition’s inertial navigation system is the slave system to be calibrated and aligned to the aircraft master navigator. The environmental conditions that both master and slave systems are exposed to during the transfer alignment procedure and location of slave with respect to the master system were also adjusted according to a fighter aircraft configuration. Furthermore, the transfer alignment trajectory was specified by using maneuver definitions that only an aircraft can perform.

In this thesis, transfer alignment methods were evaluated by primarily considering the resultant error estimation performance that can be achieved. Therefore, the total duration required to complete the transfer alignment procedure was regarded only as a secondary factor during the design of algorithms.

1.2 ORGANIZATION OF THE THESIS

The performance of transfer alignment algorithms is highly dependent to inertial systems qualifications and environmental conditions including flight trajectory followed during transfer alignment procedure. Therefore, in order to make a comparison between different methods, it is necessary to define a complete simulation environment that is used to assess algorithm performances. Hence, this thesis is divided into two major parts. The first part, which consists of Chapter 2 to 3, was devoted to developing a computer simulation environment.

In Chapter 2, the physical structure and specifications of inertial navigation systems that were used in this study was described. Also, the mathematical models of error sources that effect the operation of these inertial navigation systems were derived.

In Chapter 3, trajectory generation program used in the simulation environment was explained. Also, the vibration models which were used throughout in this study were presented in this chapter.

The second part of this thesis was devoted to developing, analyzing and comparing different transfer alignment algorithms. This part consists of Chapter 4 and 5.

In Chapter 4, five different transfer alignment algorithms were developed and the performances of these algorithms under different conditions were compared. Also, the structure of complete transfer alignment simulation environment which is used to assess the algorithm performances is presented in this chapter.

In chapter 5, the transfer alignment algorithms that are based on velocity matching methods were analyzed in detail. Furthermore, the effects of the trajectory that the system follows during transfer alignment algorithms were discussed in the same chapter.

Chapter 6 concludes the thesis.

CHAPTER II

INERTIAL NAVIGATION SYSTEM

Inertial navigation is the process of calculating position, velocity and attitude (PVA) of the system with respect to a known reference frame by using inertially measurable units such as acceleration and rotation rate. This process is accomplished by utilizing Newton's law of classical mechanics. In simple words, given the initial PVA, by continuously integrating the acceleration and rotation rate of the system, instantaneous PVA is calculated. Therefore, an inertial navigation system (INS) can be considered as being composed of two basic parts;

i. An inertial measurement unit (IMU) to sense linear and rotational motion (acceleration / rotation rates)

ii. A processor to properly integrate IMU outputs

The biggest problem of this approach is the complexity of designing an inertial measurement unit which is capable of sensing total acceleration and rotation rate acting on the system with a sufficient accuracy. Basically, IMU is the main part of an INS in determining the overall system performance and price. The very high cost of good quality IMUs forces system designers to develop an inertial navigation system by using an IMU which are just enough (if not at all) to provide sufficient accuracy. Therefore, navigation system designer first has to decide which kind of IMU to use, and then optimize the rest of design process by considering that specific IMU.

In this chapter, the inertial navigation system that was used as slave navigator throughout the study will be introduced. The errors on selected slave IMU types and their effects on slave INS outputs will be described. The general slave INS structure and its frame of reference mechanizations will be outlined.

It is not the aim of this chapter to derive and/or explain all the theory related with inertial navigation systems. Rather, this chapter focuses on introducing the system structure and corresponding navigation equations used throughout the study.

2.1 SLAVE NAVIGATION SYSTEM STRUCTURE

In this study, it is assumed that slave system has an inertial navigator with the structure presented in Figure 2-1.

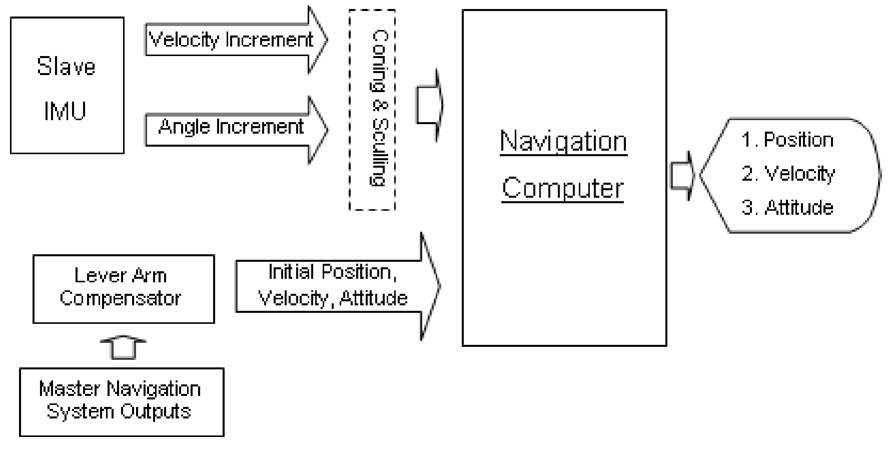


Figure 2-1 : Structure of Slave Inertial Navigation System

As seen from the above figure, initial navigation solution is provided to the slave system using master navigation system output. This kind of initialization is called as “one shot transfer alignment”. For a typical configuration, as the two systems are located apart from each other, output of master system can not be transferred to the slave directly. The effect of separation should be corrected before the transfer. This compensation process is called as lever arm compensation. It is apparent that, accuracy of lever arm compensation process puts a lower limit to the accuracy of initialization. Although, with this technique, velocity and position of slave system can be initialized with negligible errors, resultant attitude accuracy can not be sufficient for almost any mission. Therefore, some additional techniques are needed to be incorporated in order to increase initial attitude accuracy. In fact, this is one of the main reasons why transfer alignment algorithms are very curicial for this kind of systems.

In the above system, which is used as slave INS throughout this study, the navigation computer is mechanized in local level navigation frame of reference with the x, y, z coordinate axes pointing north, east, and down directions respectively (NED frame). For this system, the following navigation equations are used (Titterton (1997)):

$$\dot{C}_b^n = C_b^n S\left(\omega_{ib}^b\right) - S\left(\omega_{in}^n\right)C_b^n \tag{2.1}$$

$$\dot{V}_e^n = C_b^n f - (2\omega_{ie}^n + \omega_{en}^n)\times V_e^n + g \tag{2.2}$$

$$\dot{L} = \frac{V_N}{R_N + h} \qquad \dot{\ell} = \frac{V_E \sec L}{R_E + h} \qquad \dot{h} = -V_D \tag{2.3}$$

These equations have the following properties:

i. System position is defined as Latitude (L), Longitude (l) and Height (h) and updated using the same variables.

ii. System velocity is defined with respect to earth and is defined in NED frame.

iii. The attitude information is updated in Body to NED Frame Direction Cosine Matrix (DCM), and presented to the user with the associated Euler Angles.

The Inertial Measurement Unit (IMU) provides velocity and angle increments at 600Hz. Therefore, navigation computer can execute equations (2.1) to (2.3) at 600Hz. But, considering the conning environment generated by aircraft structure and environmental conditions, 6 sample coning and sculling algorithms were also implemented by reducing the computation frequency of navigation computer to 100Hz. The implemented conning algorithm is taken from Ignagni (1996) (Algorithm 6). According to this algorithm, total compensated angle increment between T and T+0.01 seconds (which is called major interval) is computed using the following equation:

$$\alpha_{comp} = \sum_{n=1}^{2}\sum_{m=1}^{3}\alpha_n(m) + \frac{1}{2}\left[\alpha_1 \times \alpha_2\right] + \sum_{n=1}^{2}\delta\alpha_n \tag{2.4}$$

where $\delta\alpha_n$ correspond to coning compensation term among nth minor interval (0.005 second duration in a major interval) which is calculated as:

$$\delta\alpha_n = \left[\frac{-1}{420}\alpha_{n-1}(2) + \frac{1}{40}\alpha_{n-1}(3) + \frac{157}{420}\alpha_n(1) + \frac{1207}{840}\alpha_n(2)\right]\times\alpha_n(3) \tag{2.5}$$

and each “α” corresponds to an uncompensated angle increment (gyroscope raw angle increment output) during a specific time increment represented in the following figure:

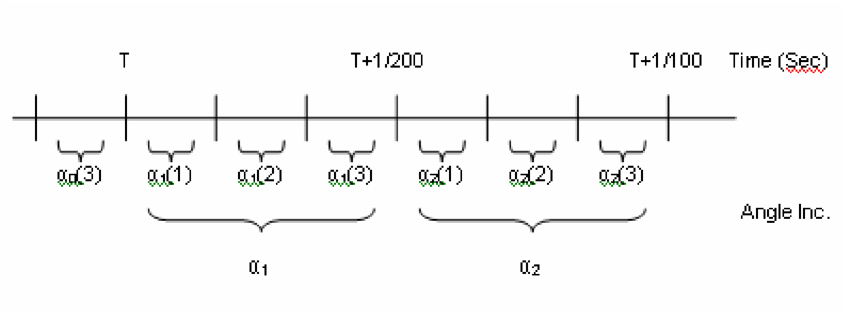


Figure 2-2 : Angle increments used in conning algorithm

Sculling algorithm was also derived based on above conning algorithm using the equivalency rule shown in Roscoe (2001). A brief analysis and derivation of coning and sculling algorithms was also presented in Appendix A.

During the transfer alignment procedure, position of slave INS can be easily updated using master INS position solution. So, by periodically correcting slave’s position solution using master’s position, instability in vertical channel can be avoided. Therefore, for position calculations, instead of using special integration rules like “scrolling algorithm” (Savege (2000)) or “trapezoid rule”, just simple Euler integration was implemented during transfer alignment procedure.

2.2 NAVIGATION SYSTEM ERRORS

Navigation computer of an INS is essentially a differential equation solver. The corresponding equations, which are given from (2.1) to (2.3), represent a nonlinear, time varying system. As shown in Koyaz (2003), this system is unstable in the sense of Liapunov. Therefore, every disturbance that affects the system causes the output errors to grow unbounded. The rate at which errors grow determined by the source of error and the trajectory that system follows.

For the INS systems, major error sources can be classified into 3 groups:

i. IMU Errors (Input Errors)

ii. Initialization Errors (Initial state Errors)

iii. Computation Errors

In the subsequent subsections, each of the above errors is explained regarding to the slave INS system that is under consideration.

2.2.1 Slave IMU Errors

In the literature more than 20 different types errors are defined for IMU outputs. However, for the system point of view, most of these errors are out of concern. This is because, during the field use of an IMU, combined effect of most of the errors can not be separated by just observing the raw IMU outputs. To localize each error sources, some specialized test methodologies (like Allen variance tests) should be incorporated and obviously this is not possible during the active operation.

Therefore, in this study, actual IMU errors are grouped according to their effects on raw IMU outputs. Errors which represent similar output characteristics are modeled using just a single model based on dominant error source belonging to that group. For instance, quantization error of sensors was ignored and their effects on sensor outputs were represented by adjusting random walk variance in constructing models. This is because, it is impossible to distinguish these two errors by using sensor outputs recorded at a constant rate.

The list of IMU errors and their mathematical models with related parameter values that are used in this study are represented in Table 2-1. The error parameters were adjusted in such a way that, the modeled IMU represents a generic commercial grade IMU which consists of MEMS accelerometers and fiber optic gyroscopes. In Table 2-1, repeatability errors represent so called day to day random bias components. It is assumed that, the value of this bias component does not change during the active operation. On the other hand, instability errors represent the varying nature of error components. The "n" in the model equations denotes disturbances in the form of white noise.

Using the error definitions given in Table 2-1, the relation between true and actual IMU outputs can be represented as follows:

$$\begin{vmatrix} \tilde{a}^{x} \\ \tilde{a}^{y} \\ \tilde{a}^{z} \end{vmatrix} = \begin{vmatrix} a^{x} \\ a^{y} \\ a^{z} \end{vmatrix} + \begin{vmatrix} \delta a_{BR}^{x} \\ \delta a_{BR}^{y} \\ \delta a_{BR}^{z} \end{vmatrix} + \begin{vmatrix} \delta a_{BS}^{x} \\ \delta a_{BS}^{y} \\ \delta a_{BS}^{z} \end{vmatrix} + \begin{vmatrix} \delta a_{RW}^{x} \\ \delta a_{RW}^{y} \\ \delta a_{RW}^{z} \end{vmatrix} + \begin{vmatrix} \delta a_{SFR}^{x} & 0 & 0 \\ 0 & \delta a_{SFR}^{y} & 0 \\ 0 & 0 & \delta a_{SFR}^{z} \end{vmatrix} \begin{vmatrix} a^{x} \\ a^{y} \\ a^{z} \end{vmatrix} \quad (2.6)$$

$$\begin{vmatrix}\tilde{\omega}^x\\ \tilde{\omega}^y\\ \tilde{\omega}^z\end{vmatrix} = \begin{vmatrix}\omega^x\\ \omega^y\\ \omega^z\end{vmatrix} + \begin{vmatrix}\delta g_{DR}^x\\ \delta g_{DR}^y\\ \delta g_{DR}^z\end{vmatrix} + \begin{vmatrix}\delta g_{DS}^x\\ \delta g_{DS}^y\\ \delta g_{DS}^z\end{vmatrix} + \begin{vmatrix}\delta g_{RW}^x\\ \delta g_{RW}^y\\ \delta g_{RW}^z\end{vmatrix} + \begin{vmatrix}\delta g_{SFR}^x & 0 & 0\\ 0 & \delta g_{SFR}^y & 0\\ 0 & 0 & \delta g_{SFR}^z\end{vmatrix}\begin{vmatrix}\omega^x\\ \omega^y\\ \omega^z\end{vmatrix} \qquad (2.7)$$

In the above equations, $\tilde{a}$ and $\tilde{\omega}$ represent the actual IMU outputs whereas . a and ω denotes true values.

Table 2-1 : List of IMU errors that are used in this study

Name	Mathematical Model	Parameter Value
Acc. Bias Repeatability	$\delta\dot{a}_{BR} = 0$	Bias Repeatability : 1.5mg (1σ)
Acc. Bias Instability	$\delta\dot{a}_{BS} = -\frac{1}{\tau}\delta a_{BS} + n_{BS}$	Bias Instability at steady state : 0.2mg (1σ) Time Constant : 60sec.
Acc. Scale Factor Rep.	$\delta\dot{a}_{SFR} = 0$	Scale Factor Repeatability : 500 ppm (1σ)
Acc. Random Walk	$\delta a_{RW} = n_{ARW}$ (White Noise)	Standard Deviation of WN : 250µg/√s
Gyro. Drift Rep.	$\delta\dot{g}_{DR} = 0$	Rate Drift Repeatability : 10deg/h (1σ)
Gyro. Drift Instability	$\delta\dot{g}_{DS} = -\frac{1}{\tau}\delta g_{dS} + n_{DS}$	Drift Instability at steady state : 0.35deg/h (1σ) Time Constant : 100sec.
Gyro. SF. Rep.	$\delta\dot{g}_{SFR} = 0$	Scale Factor Repeatability : 500ppm (1σ)
Angle Random Walk	$\delta g_{RW} = n_{GRW}$ (White Noise)	Standard Dev. of WN : 0.1°/hr/√Hz

Although it was assumed that fiberoptic gyroscopes were used in the IMU assembly, the gyroscope scale factor nonlinearities were not considered in this study. Furthermore, despite of the fact that temperature, misalignment and "g / g^2" dependent errors are extremely important, they were also ignored because of the following 2 reasons:

i. It was assumed that IMU compensation algorithms reduce the effect of these errors to an acceptable level.

ii. The inclusion of the empirical models which were derived for these errors makes overall system analysis unnecessarily complex.

As a matter of fact, deriving an equivalent simplified IMU error model form a huge IMU error list is also another challenging problem which is closely related with reduction of order problem. Usually, in practice, the errors that cannot be estimated by the estimation algorithms are eliminated and their effects on IMU outputs are handled by adjusting other dominant IMU error model parameters. On the other hand, in the scope of this thesis, no special study was conducted to derive a simplified error model, and the errors given in Table 2-1 are assumed to be the exact error model of IMU.

2.2.2 Initialization Errors

As mentioned before, initial PVA values for slave navigation computer is calculated by compensating the lever arm effect on the solution provided by master navigator. In order to initialize slave navigation system without error, however, following information must also be supplied to lever-arm compensator:

i. Instantaneous lever arm vector.

ii. Instantaneous orientation of slave with respect to master.

iii. Instantaneous velocity of slave with respect to master.

Due to the non-rigid structure of the aircraft, it is impossible to determine the instantaneous values of the above information accurately. Therefore, calculated initial values for slave system contain errors.

The exact reason for these initialization errors and corresponding mathematical models will be described and derived at Chapter 3 and Chapter 4. However, for the error analysis purposes presented at Chapter 2.3.2, it is assumed that initial PVA error of slave is the sum of error on master navigator outputs and error induced during lever arm compensation process. During straight flight of the aircraft, error induced by the lever arm compensation process is equal to the error on nominal lever arm information which has the following 1σ uncertainty values including flexure effect of aircraft wing:

Table 2-2 : Errors on nominal lever arm values provided by mission computer

Uncertainties in the position of slave with respect to master	X axes	15 cm (1σ)
	Y axes	15 cm (1σ)
	Z axes	30 cm (1σ)
Uncertainties in the orientation of slave with respect to master	Roll	20 mrad (1σ)
	Pitch	20 mrad (1σ)
	Yaw	10 mrad (1σ)

It is assumed that master navigator outputs contain jitter type errors. Therefore, during initialization, these errors are also directly transferred to the slave INS. In this study, it is assumed that jitter errors of master navigator have the standard deviation values presented in Table 2-3.

Table 2-3 : Standard deviation of master navigation system errors

Master Velocity Error Std. Dev.	0.05 m/s (1σ) (for each axes)
Master Rotation Rate Std. Dev.	1.7453e-004 rad/sec (1σ)
Master Attitude Error Std. Dev	3.4907e-004 rad (1σ)

2.2.3 Computational Errors

The discrete and quantized nature of navigation processors tends to produce computational errors on navigation solution. This situation arises especially in the high vibratory environment. The importance of this kind of error depends on the fact that this error can neither be estimated nor compensated. Therefore this error puts a lower limit in the accuracy of inertial navigation system.

For the real implementation (when real IMU increments are used), with the use of appropriate conning and sculling algorithms and sufficient processing frequency, computational errors can be reduced to very low level. However, one should be very careful when designing a simulation environment in computer. In a computer simulator implementation, calculating simulated velocity and angle increments instead of acceleration and rotation rate can be very difficult under vibration environment. Usually, this difficulty is overcome by simply taking Euler integration of calculated acceleration and rotation rate to obtain associated increments. However, such an operation causes computation errors to grow significantly. Therefore, when developing a simulation environment in computer, this point should always be considered, and necessary precautions should be taken to reduce the effect of computational errors during simulations. In Chapter 3 and 4, the measures that are used to reduce the computational errors in the simulations performed in this study are described.

2.3 NAVIGATION SYSTEM ERROR ANALYSIS

In this section, effects of above errors on the navigation system outputs are described. To achieve this, first linear error propagation models for the navigation system is introduced. Then, the effectiveness of these error propagation equations is verified by “Monte Carlo” Analysis (MC Analysis). At the end, the sensitivity of error propagation models to the perturbation on nominal trajectory is presented using simulation results.

2.3.1 Linear Error Propagation Models

Linear error propagation models are derived basically by linearizing the navigation equations (Equation 2.1 – 2.3) around a nominal trajectory. In this study, errors are defined in such a way that resultant error propagation models turns out to be what is known as “Φ - Formulation” in the literature (Titterton, (1997)).

2.3.1.1 Attitude Error Propagation

Let φ = [$\delta\alpha$ $\delta\beta$ $\delta\gamma$] represents the Euler angles which define the orientation of erroneous navigation Frame with respect to true navigation frame. In this case, for small angles, the relation between true and erroneous transformation matrices turns out to be as follows:

$$\tilde{C}_b^n = \left[\mathrm{I} - \Phi\right] C_b^n \tag{2.8}$$

where

$$\Phi = S(\varphi) = \begin{bmatrix} 0 & -\delta\gamma & \delta\beta \\ \delta\gamma & 0 & -\delta\alpha \\ -\delta\beta & \delta\alpha & 0 \end{bmatrix} \tag{2.9}$$

For the above error relation, error propagation equation can be found as (Titterton (1997)):

$$\dot{\varphi} = -\omega_{in}^{n} \times \varphi + \delta\omega_{in}^{n} - C_{b}^{n}\delta\omega_{ib}^{b} \tag{2.10}$$

where "$\delta\omega_{ib}^{b}$" represents the total gyro output error which is equal to "$\tilde{\omega} - \omega$" given in Equation (2.7) and "$\delta\omega_{in}^{n}$" represents the total error in earth and transport rate calculations. Ignoring position errors, $\delta\omega_{in}^{n}$ can be expressed as follows:

$$\delta\omega_{in}^{n} = \delta\omega_{en}^{n} = \begin{bmatrix} \frac{1}{R}\delta V_{e} \\ -\frac{1}{R}\delta V_{n} \\ -\frac{\tan(L)}{R}\delta V_{e} \end{bmatrix} \tag{2.11}$$

"ω_{in}^{n}" can be calculated as follows:

$$\omega_{in}^{n} = \omega_{ie}^{n} + \omega_{en}^{n} \tag{2.12}$$

$$\omega_{ie}^{n} = \begin{vmatrix} \Omega\cos L \\ 0 \\ -\Omega\sin L \end{vmatrix} \tag{2.13}$$

$$\omega_{in}^{n} = \begin{bmatrix} \frac{V_{E}}{R_{E}+h} & -\frac{V_{n}}{R_{N}+h} & -\frac{V_{E}\tan L}{R_{E}+h} \end{bmatrix}^{T} \tag{2.14}$$

2.3.1.2 Velocity Error Propagation

Velocity error is defined as:

$$\delta v = \tilde{v} - v \tag{2.15}$$

For this definition error propagation equation becomes (Titterton (1997)):

$$\dot{\delta v} = C_{b}^{n}\delta a^{b} - \Phi C_{b}^{n} a^{b} - (2\omega_{ie}^{n} + \omega_{en}^{n}) \times \delta v - (2\delta\omega_{ie}^{n} + \delta\omega_{en}^{n}) \times v \tag{2.16}$$

In the above equation, "δa^{b}" represents total accelerometer which is equal to "$\tilde{a} - a$" given in Equation (2.6).

2.3.1.3 Position Error Propagation

Position errors are defined as follows:

$$\tilde{L} = L + \delta L \qquad \tilde{\ell} = \ell + \delta\ell \qquad \tilde{h} = h + \delta h \tag{2.17}$$

For these definitions, error propagation equations for position became (Titterton (1997)):

$$\begin{bmatrix} \delta\dot{L} \\ \delta\dot{\ell} \\ \delta\dot{h} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & \frac{1}{R} & 0 & 0 & 0 & 0 & \frac{-V_N}{R^2} \\ 0 & 0 & 0 & 0 & \frac{1}{R\cos L} & 0 & \frac{V_E \tan L}{R\cos L} & 0 & \frac{-V_E}{R^2\cos L} \\ 0 & 0 & 0 & 0 & & -1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \delta\alpha \\ \delta\beta \\ \delta\gamma \\ \delta V_N \\ \delta V_E \\ \delta V_D \\ \delta L \\ \delta\ell \\ \delta h \end{bmatrix} \tag{2.18}$$

2.3.2 Verification of Linear Error Propagation Models

The above error propagation models are derived depending on the assumption that errors are small (product of two error terms are negligible). For a typical inertial navigation system, however, errors build up in time and evidently ruin the above assumption. Therefore, in order to verify that above propagation models are valid during transfer alignment procedure, MC analyses are performed to compare the true and predicted error variances.

MC analysis is performed on a typical pre-launch path of an aircraft. The trajectory consists of three coordinated turns with a bank angle of approximately 30 degrees each followed a by level and straight flight segment. Total duration of flight is taken to be 120 seconds. The generation of this trajectory is described in Chapter 3.

At each trial in Monte Carlo analysis, erroneous IMU outputs are processed by the navigation system whose initial conditions also have errors. The errors are generated using the models presented above. After 5000 trials, variance of the difference between the navigation system output and true trajectory is calculated and the result is compared with the variance computed by linear error propagation models.

The following table represents the maximum percent difference between true error variances and variance calculated by linear error propagation models when the initial attitude uncertainty is 20mrad in horizontal and 10mrad in vertical.

Table 2-4 : Maximum percent difference between true and estimated error variances when initial attitude uncertainty is 20mrad in horizontal and 10mrad in vertical

	Max % Difference
Height Error Variance	13.95576
North Velocity Error Variance	1.434847
East Velocity Error Variance	0.732525
Down Velocity Error Variance	16.64879
Roll Error Variance	0.689604
Pitch Error Variance	0.468076
Yaw Error Variance	0.174904

As seen in the above table, except vertical channel, linear error propagation models are very successful in calculating the true error variances. In the vertical channel, however, linear models become insufficient due to the relatively big initial horizontal attitude uncertainties which violate small error assumption. On the other hand, even with the simplest estimation algorithm, it is possible to reduce the initial horizontal attitude uncertainties below 2mrad within in very small period of time. Therefore, in practice, it will be enough for the linear error propagation models to track actual error variance under the initial attitude error uncertainty of 2mrad in horizontal and 10mrad in vertical. Under this new condition when the above MC analysis was performed again, it was seen that maximum difference between calculated and true error variances of down velocity and height errors reduces to 1.403% and 1.214% respectively.

Above result implies that, just reducing the initial horizontal attitude uncertainty below 2mrad level, it is assured that linear error propagation models can track true error variances very successfully. Therefore, even if an open loop Kalman filter structure is implemented, at the beginning of the transfer alignment procedure, attitude uncertainties have to be reduced by some feedback mechanism.

Another point that needs to be verified is the sensitivity of linear error propagation models to the nominal trajectory values. In the calculation of above results, true trajectory values are used as nominal values for the propagation models. However, in a typical Kalman filter application for inertial navigation systems, nominal trajectory is taken to be the outputs of the inertial navigation system. This means that, some erroneous navigation values are used as nominal trajectory. In order to prove that this action does not have any effect of the filter performance, one has to show that linear error propagation models are insensitive to the variations in nominal trajectory.

In order to show this, error variance is recalculated using error models which are linearized among a trajectory whose position and velocity values are perturbated by 10 percent and horizontal and vertical attitude values are perturbated with 2mrad and 10mrad (attitude differences are kept in this level in order to satisfy small angle assumption). The maximum percentage difference between computed error variances calculated using true and erroneous trajectories is presented in the following table.

Table 2-5 : Maximum percent difference between computed error variances calculated using true and erroneous trajectories

	Max % Difference
Height Error Variance	0.206334
North Velocity Error Variance	1.253082
East Velocity Error Variance	0.37991
Down Velocity Error Variance	0.257211
Roll Error Variance	0.074917
Pitch Error Variance	0.074015
Yaw Error Variance	0.002848

Above results show that error variance values are highly insensitive to the arbitrary change in nominal position and velocity values. This is also true for attitude values as long as change in attitude values is limited by small angle assumption.

On the other hand, the same situation does not exist for the instantaneous acceleration and rotation rates. As seen in equation (2.12) and (2.18) these values are required in the calculation of error variances. However, when the above analysis is preformed by just increasing these nominal values %1, it is observed that as much as 10% error is occurred on calculated variances. So, linear error propagation models are highly sensitive to the changes in instantaneous acceleration and rotation rates.

Therefore, in the discretization of propagation models, maximum frequency content of acceleration and rotation rates should be used as the basis for the determination of step size. Also, in the calculation of nominal trajectory values, instead of raw IMU outputs, the IMU outputs which are corrected (calibrated) by the estimation algorithms should be used.

CHAPTER III

KINEMATICS OF MASTER AND SLAVE SYSTEMS AND CHARACTERIZATION OF VIBRATION ENVIRONMENT

Kinematics is a branch of mechanics that concern with the geometry of motion without reference to masses or forces acting on the system. It is a collection of vector/matrix methods to describe positions, velocities, and accelerations of particles and rigid bodies as viewed from various reference frames without regard to forces causing the motion (Junkins (2003)).

The base of kinematics depends on the knowledge of position and orientation of the system. The rest of the kinematic variables such as velocity, angular rate, acceleration etc can be derived by taking time derivative of position and orientation of the system with respect to some fixed or moving frame.

These properties of kinematics make it an indispensable tool in the design and analysis of navigation systems. In fact, the navigation equations presented in the Chapter 2 are just a simple application of kinematics to calculate position and orientation of the system with respect to earth by using the knowledge of acceleration and rotation rate of the system with respect to inertial space.

Apart from this fact, kinematics is also important from the point of view of transfer alignment for the following two reasons:

First of all, transfer alignment is a maneuver dependent process. Therefore, in order to verify the algorithms, it is essential to be able to calculate total acceleration and rotation rate of the system which follows a predetermined path. These kinematics calculations are called as trajectory generation.

Secondly, during the transfer alignment procedure, the outputs of master INS have to be transformed to slave frame of reference before being used in estimation algorithms. This process is called lever arm compensation. In order to perform this operation, the kinematics relations between master and slave systems should be derived.

In Chapter 3.1 and 3.2, the kinematics equations used for the above tasks are derived. In the first section the method used in generation of the trajectory that master INS follows is explained. After that, kinematic relations between master and slave systems' PVA solutions and IMU outputs are presented.

In the rest of this chapter, the vibration environment that a slave INS is exposed to during transfer alignment process is presented. As indicated above, the lever arm compensation is an essential part of transfer alignment algorithms. However, in order to perform an accurate lever arm compensation it is required to characterize the relative motion of slave with respect to master navigation system. Due to the non rigid structure of the aircraft, the slave inertial navigation system continuously vibrates during the captive flight. Therefore, in order to perform realistic simulations it is necessary to utilize some vibration models that generate the relative motion of slave with respect to master navigator. In Chapter 3.3, the vibration models used in this study are presented. Furthermore, in Chapter 3.4, the effect of vibration environment on the theoretical inertial navigation performance of slave system is briefly introduced.

3.1 TRAJECTORY GENERATION

One of the essential properties of transfer alignment algorithms designed for the guided munitions is the special maneuver of the aircraft defined for that algorithm. During the transfer alignment, by forcing the aircraft to make some special maneuvers, the observability of the navigation system errors are enhanced (Rehee (2004)). Therefore, every transfer alignment algorithm should be assessed under some maneuver conditions. In the computer environment, simulating a maneuver consists of simultaneous calculation of the PVA of the system and the associated accelerations and rotation rates occurred on the system. This procedure is called as Trajectory Generation (TG).

In general, trajectory generators consist of 2 basic parts called as shaping and regeneration (Savage (2000)). Trajectory shaping lets user define a trajectory that the system follows, whereas regeneration function calculates PVA solutions and accelerations - rotation rates that an IMU connected to the system measures.

As a matter of fact, developing a trajectory generator that reflects the real word environment in a realistic manner can be very complicated. Such a generator designed for an aircraft should include effects of environmental conditions, maneuver induced vibrations, loading conditions (which also vary due to fuel consumption, weapon release etc.), and involves implementation of aircraft control mechanism.

On the other hand, as shown in previous section, the error propagation models are highly insensitive to the PVA solutions. Therefore, for the transfer alignment point of view, developing a simple generator that generates a nominal trajectory suffices. After that solution is obtained, in order to represent the effect of vibration induced motion, a vibration model outputs are added to that solution.

Moreover, despite of the fact that a fighter aircraft is capable of performing a variety of different maneuver, in the design of TA maneuver, the designer has to confine himself only a very small subset of possible maneuvers because of general operational concept of aircrafts. In general, only the selection of bank angle and heading change parameters are left to the designers (these parameters have also some strict upper limits). Therefore, it will be enough for a trajectory generator developed for TA procedure to just simulate level flight and coordinated turn commands.

Because of the above reasons, instead of a highly complicated one, development of a simple and flexible trajectory generator was preferred. In this study, the generator described in Musick (1976) was used as a base. The same methodology given in Musick (1976) was used in deriving the PVA solution, however a completely different trajectory shaping function was implemented.

In the next section, kinematic equations used in trajectory generator are described. In the derivation of equations the following assumptions were made:

i. IMU is at the center of gravity (CG) of aircraft and aligned with the body axes of aircraft.

ii. The maneuvers are performed with zero angle of attack

iii. During the horizontal turns, no altitude change occurs.

iv. Total acceleration acting on the system at y-z plane is always lies along “z” axis. (Coordinated turn requirement)

v. Velocity of the system lies only along x axes of the aircraft body frame and system can accelerate / decelerate only on this axes.

The trajectory is generated for the master INS system. The transformation of these results to the slave system and inclusion of vibration environment will be described at Chapter 4.

In the following discussions, trajectory generation task is divided into two major parts:

i. Trajectory Regeneration

ii. Trajectory Shaping

Trajectory shaping function is responsible for generating first and second time derivatives of Euler angles for the system performing a used defined motion. By using that information, trajectory regeneration part calculates the rest of the navigation parameters.

3.1.1 Trajectory Regeneration Function

Most of the equations derived in this section are taken from Musick (1976). Therefore for more general treatment on the issue, it is suggested to resort that study.

Assume that initial PVA (entered by the user) and time derivative of Euler angles which defines the rotation from body to NED frame (supplied by the trajectory shaping function) are known. In this case, rotation rate of master IMU with respect to NED frame (defined also in NED frame) can be found as follows:

$$\begin{vmatrix} \omega_{nb}^{n}(x) \\ \omega_{nb}^{n}(y) \\ \omega_{nb}^{n}(z) \end{vmatrix} = \begin{vmatrix} 0 \\ 0 \\ \dot{\psi} \end{vmatrix} + \begin{vmatrix} c\psi & -s\psi & 0 \\ s\psi & c\psi & 0 \\ 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} 0 \\ \dot{\theta} \\ 0 \end{vmatrix} + \begin{vmatrix} c\psi & -s\psi & 0 \\ s\psi & c\psi & 0 \\ 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} c\theta & 0 & s\theta \\ 0 & 1 & 0 \\ -s\theta & 0 & c\theta \end{vmatrix} \begin{vmatrix} \dot{\phi} \\ 0 \\ 0 \end{vmatrix} \quad (3.1)$$

where parenthesis represents the element of vector "ω_{nb}^{n}"

By using "ω_{nb}^{n}", time derivative of direction cosine matrix which transform body frame to NED frame can be calculated as follows:

$$\dot{C}_{b}^{n} = S\left(\omega_{nb}^{n}\right)C_{b}^{n} \quad (3.2)$$

Using transport theorem (Junkins (2003)), time derivative of velocity with respect to ECEF frame defined in NED frame can be written as:

$$\dot{v}_{e}^{n} = C_{b}^{n}\dot{v}_{e}^{b} + S\left(\omega_{nb}^{n}\right)v_{e}^{n} \quad (3.3)$$

In the above equation "$\dot{v}_{e}^{b}$" represent the derivative of velocity of the system with respect to ECEF frame defined in body frame. If it is assumed that velocity lies along only x-axes and system can accelerate only on this axis, then:

$$\dot{v}_e^b = \begin{vmatrix} \dot{V}_{Total} \\ 0 \\ 0 \end{vmatrix} = \begin{vmatrix} A \\ 0 \\ 0 \end{vmatrix} \quad (3.4)$$

"A" denotes the acceleration of the system along its path which should also be supplied by the user. "$\dot{V}_{Total}$" represent the total velocity of the system with respect to ECEF frame and equals to:

$$V_{Total} = \sqrt{V_N^2 + V_E^2 + V_D^2} \quad (3.5)$$

Time derivative of positions can be calculated as follows:

$$\dot{L} = \frac{V_N}{R_N + h} \qquad \dot{\ell} = \frac{V_E \sec L}{R_E + h} \qquad \dot{h} = -V_D \quad (3.6)$$

After specifying initial PVA point, time derivative of Euler angles and path acceleration, above set of equations can be used to calculate the PVA solution of the system along its trajectory. To do so, first Equation (3.1) is used to find rotation rates in NED frame, after than Equation (3.2) is used to update DCM. Finally, by using Equation (3.3) and (3.6) velocity and position variables are updated (see Figure 3-1).

On the other hand, the trajectory generator also has to compute the IMU outputs of the system which follows the above calculated trajectory. This process can be accomplished as follows:

Rotation rate of master IMU with respect to inertial frame defined in body frame can be written as:

$$\omega_{ib}^b = C_n^b \omega_{ib}^n = C_n^b \left(\omega_{in}^n + \omega_{nb}^n \right) \quad (3.7)$$

"ω_{nb}^n" is calculated using Equation. (3.1). Also "ω_{in}^n" can be calculated as follows:

$$\omega_{in}^n = \begin{bmatrix} \frac{V_E}{R_E + h} + \Omega \cos L & -\frac{V_n}{R_N + h} & -\frac{V_E \tan L}{R_E + h} - \Omega \sin L \end{bmatrix} \quad (3.8)$$

In order to calculate slave IMU outputs using master IMU outputs, the time derivative of "ω_{ib}^b" will also be required in Chapter 4. This parameter can be calculated as follows:

$$\dot{\omega}_{ib}^b = C_n^b \left[S\left(\omega_{nb}^b \right) \omega_{ib}^n + \dot{\omega}_{ib}^n \right] \quad (3.9)$$

$$\dot{\omega}_{ib}^n = \dot{\omega}_{nb}^b + \dot{\omega}_{en}^n + \dot{\omega}_{ie}^n \quad (3.10)$$

$$\dot{\omega}_{ie}^{n} = \begin{vmatrix} -\Omega\sin(L)\dot{L} \\ 0 \\ -\Omega\cos(L)\dot{L} \end{vmatrix} \tag{3.11}$$

$$\dot{\omega}_{en}^{n} = \begin{vmatrix} -\frac{v_e}{(R_e+h)^2}\dot{h} + \frac{\dot{v}_e}{R_e+h} \\ \frac{v_e}{(R_n+h)^2}\dot{h} - \frac{\dot{v}_n}{R_n+h} \\ -\frac{v_e \sec L}{R_e+h}\dot{L} + \frac{v_e\tan(L)}{(R_e+h)^2}\dot{h} - \frac{\tan(L)}{R_e+h}\dot{v}_e \end{vmatrix} \tag{3.12}$$

$$\dot{\omega}_{ib}^{n} = \begin{vmatrix} -c\psi\dot{\psi}\dot{\theta} - s\psi\ddot{\theta} - s\psi\dot{\psi}c\theta\dot{\phi} - c\psi s\theta\dot{\theta}\dot{\phi} + c\psi c\theta\ddot{\phi} \\ -s\psi\dot{\psi}\dot{\theta} + c\psi\ddot{\theta} + c\psi\dot{\psi}c\theta\dot{\phi} - s\psi c\theta\dot{\theta}\dot{\phi} + s\psi c\theta\ddot{\phi} \\ \ddot{\psi} - c\theta\dot{\theta}\dot{\phi} - s\theta\ddot{\phi} \end{vmatrix} \tag{3.13}$$

As seen from Equation. (3.13), if the second time derivative of Euler angles is also supplied with the inputs, it is possible to calculate continuously differentiable rotation rate which is very advantageous in terms of reducing computational errors arise from the discrete nature of computer programs.

Accelerometers sense the sum of change in velocity with respect to inertial frame and the gravitational force. Velocity with respect to inertial frame can be calculated in NED frame using transport theorem as follows:

$$v_i^b = C_e^b\left(v_e^e + S\left(\omega_{ie}^e\right)r^e\right) \tag{3.14}$$

Differentiate above equation with respect to time again:

$$\dot{v}_i^b = C_e^b\left[S\left(\omega_{ie}^e\right)v_e^e + S\left(\omega_{ie}^e\right)S\left(\omega_{ie}^e\right)r^e + \dot{v}_e^e + S\left(\omega_{ie}^e\right)v_e^e\right] \tag{3.15}$$

Using transport theorem $\dot{v}_e^e$ can be written in terms of $\dot{v}_e^n$ as follows:

$$\dot{v}_e^e = C_n^e S\left(\omega_{en}^n\right)v_e^n + C_n^e\dot{v}_e^n \tag{3.16}$$

Substituting Equation (3.16) in (3.15) and rearranging terms, following equation is obtained:

$$\dot{v}_i^b = C_n^b\left[\dot{v}_e^n + \left(S\left(\omega_{en}^n\right) + 2S\left(\omega_{ie}^n\right)\right)v_e^n\right] + C_n^b\left(S\left(\omega_{ie}^e\right)S\left(\omega_{ie}^e\right)r^e\right) \tag{3.17}$$

where ω_{ie}^n is the rotation rate of Earth defined in NED frame. Therefore it is equal to "$C_e^n\begin{bmatrix}\Omega & 0 & 0\end{bmatrix}^T$" (see Equation (2.13)).

Accelerometer output is equal to the sum of above equation and (minus) gravity. Therefore:

$$a^b = C_n^b\left[\dot{v}_e^n + \left(S\left(\omega_{en}^n\right) + 2S\left(\omega_{ie}^n\right)\right)v_e^n\right] + C_n^b\left(S\left(\omega_{ie}^e\right)S\left(\omega_{ie}^e\right)r^e - g^n\right) \qquad (3.18)$$

The term in the last parenthesis is called as blump-bob gravity and can be calculated using WGS-84 gravity model. By simply denoting it as " g ", master IMU accelerometer output can be calculated as follows:

$$a^b = C_n^b\left[\dot{v}_e^n + \left(S\left(\omega_{en}^n\right) + 2S\left(\omega_{ie}^n\right)\right)v_e^n + g\right] \qquad (3.19)$$

Using Equation (3.7) and (3.19), gyroscope and accelerometer outputs of master IMU can be calculated.

Therefore, continuously solving Equation (3.2), (3.3), (3.6), (3.7) and (3.19) sequentially in a loop, the PVA solution and respective IMU outputs of a system which follows a predetermined path characterized by time derivative of Euler angles and path acceleration can be calculated. A simplified flow diagram of Trajectory Regeneration function is represented in the following figure.

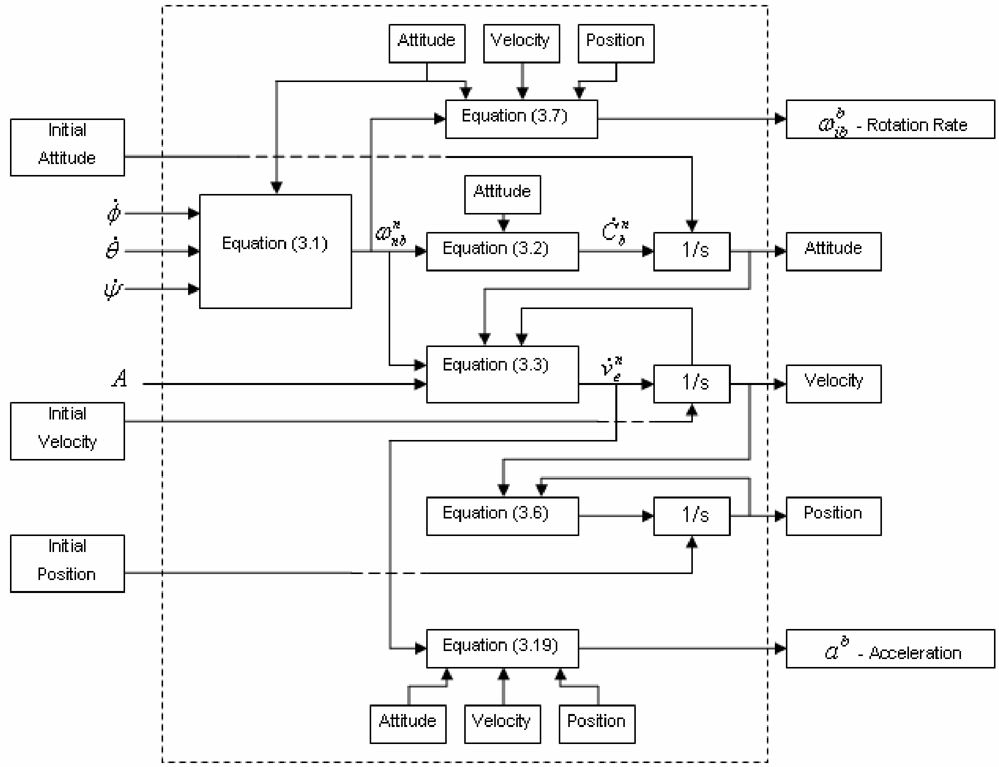


Figure 3-1 : Simplified flow diagram of Trajectory Regeneration function

In the above figure, the calculation of intermediate variables such as “ω_{ie}^{n}” or “g”, which are just a function of instantaneous position, velocity and attitude, are not shown explicitly.

The only assumption in trajectory regeneration equations is that, total velocity of the system lies along its “x” axes of body frame, and this constraint is satisfied by the equations implicitly. Therefore, whatever input is used, the program generates a consistent set of outputs regardless of whether this kind of motion can be realizable or not. That realizable motion definition should be provided by the trajectory shaping function which is described in the next section.

3.1.2 Trajectory Shaping Function

As shown above, for the trajectory regenerator to function properly, following inputs should be supplied to it:

i. Time derivative of roll rate ($\dot{\phi}$)

ii. Time derivative of pitch rate ($\dot{\theta}$)

iii. Time derivative of yaw rate ($\dot{\psi}$)

iv. Acceleration along its “x” axes (A)

For the overall trajectory simulator to simulate the flight of an aircraft, these inputs should be adjusted in such a way that they represent an actual maneuver of an aircraft. On the other hand, for the transfer alignment point of view, it is not required to simulate all kind of maneuvers that an aircraft can perform. This is because, as the transfer alignment maneuvers have to be completed in hostile territory, only very simple maneuvers such that level flight or coordinated turn can be realized during transfer alignment procedures. Therefore, it will be enough for the trajectory simulator to generate following motions:

i. Level Flight

ii. Vertical Turn

iii. Coordinated Turn

In the following sections, calculation of associated time derivatives of Euler angles occurred on an aircraft that performs the above maneuvers is described. As the axial acceleration (A) does not affect the type of maneuver, it can be adjusted to any value as long as it does not exceed the limit of the aircraft.

3.1.2.1 Level Flight

During the level flight, aircraft does not rotate with respect to NED frame. Therefore:

$$\begin{vmatrix}\dot{\phi}\\ \dot{\theta}\\ \dot{\psi}\end{vmatrix} = \begin{vmatrix}0\\ 0\\ 0\end{vmatrix} \tag{3.19}$$

3.1.2.2 Vertical Turn

During the vertical turn, only the derivative of pitch angle can take some nonzero value. A positive value corresponds to nose up, and so makes the aircraft climb, whereas negative values create a dive.

During the vertical turn, roll angle and derivative of azimuth angle should be strictly zero. Otherwise, the change in pitch angle corresponds to an unrealizable motion for an aircraft. It should also be noted that for 90° pitch angle, infinite number of Euler angle sequences yields same DCM matrix. Therefore, 90° pitch angle causes a divide by zero error which should also be avoided when specifying inputs.

3.1.2.3 Coordinated Turn

A coordinated turn means that, aircraft first has to bank in order to change heading. As it is assumed that aircraft does not change altitude, during the heading change the sum of centrifugal acceleration and gravity should lie along the z axis of aircraft.

To satisfy above constraint, total centrifugal acceleration acting on the system should be as follows:

$$a_n = g\cos(\theta)\tan(\phi) \tag{3.20}$$

In order to achieve above centrifugal acceleration, the rate of heading change during the turn should be equal to (Musick (1976)):

$$\dot{\psi} = \frac{g\tan(\phi)}{V_{Total}} \tag{3.21}$$

As seen in Equation (3.21), for the coordinated turn without altitude loss, the rate of heading change is a function of only roll angle and total velocity. On the other hand, change in velocity for an aircraft is not preferred. Therefore, the most important parameter in determining the heading change rate is the total bank angle during the turn.

Furthermore, in order to generate slave INS trajectory (which is described in Chapter 4), the second time derivative of Euler angles is also required. As the $\dot{\psi}$ is completely determined by roll angle during coordinated turn, $\ddot{\psi}$ should also be determined by ϕ and $\dot{\phi}$. By taking time derivative of Equation (3.21), $\ddot{\psi}$ can be calculated as follows:

$$\ddot{\psi} = \frac{g}{\cos^2 \phi V_{Total}^2}\left[\dot{\phi} V_{Total} - \frac{\sin 2\phi A}{2}\right] \tag{3.22}$$

In the application of coordinated turn maneuver, three consecutive phase of heading change occurs for a single turn. For instance, suppose that, it is desired to perform a heading change with a 30° bank angle. In the first phase, the aircraft starts to bank until the bank angle reaches to 30 degrees. In this period, the rate of heading change increases as the bank angle increase. In the second phase, aircraft changes its heading with a constant rate. At the last stage, aircraft returns to level flight and during which the rate of heading change decreases as the bank angle decrease and finally reaches to 0 bank angle.

Hence, in order to perform a specified heading change exactly, the change in all of three stages should be calculated separately. In Musick (1976), this problem is solved by calculating the results analytically. In order to find an analytical solution, it was assumed that, roll rate is constant during the maneuver. However this solution has two drawbacks:

i. There is no such constant roll rate for an aircraft

ii. Constant roll rate implies a jump from 0 to that specified level, which causes a significant artificial computational error in the discrete time calculations.

In order to avoid above problems, instead of dealing with analytical solutions, the following structure is implemented to generate roll rate and time derivative of roll rate:

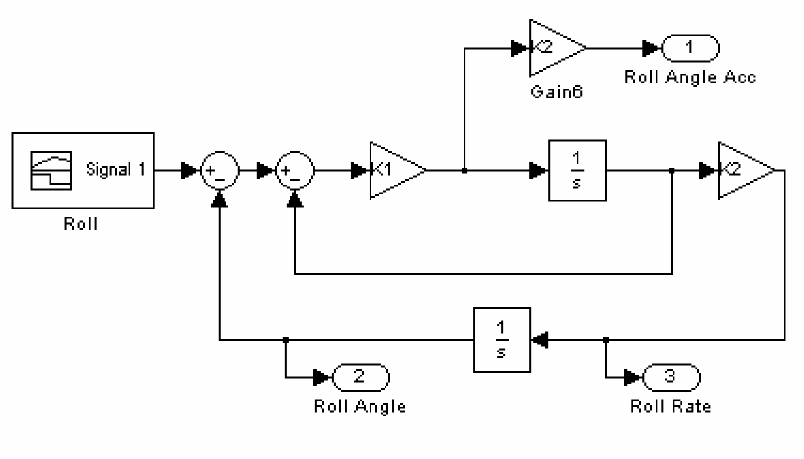


Figure 3-2 : Diagram of roll rate generator

As seen from the above structure, roll rate is generated with a feedback structure. The input to the structure is the roll angle of the system in the form of step and ramp functions. The differentiability of the outputs provides minimizing computational errors in the trajectory generator.

The feedback gains are adjusted so that system represents the real response of an aircraft. In order to determine these coefficients, a recorded roll angle history of a real test flight is used. To simulate this flight test segment, input data shown in Figure 3-3 is constructed.

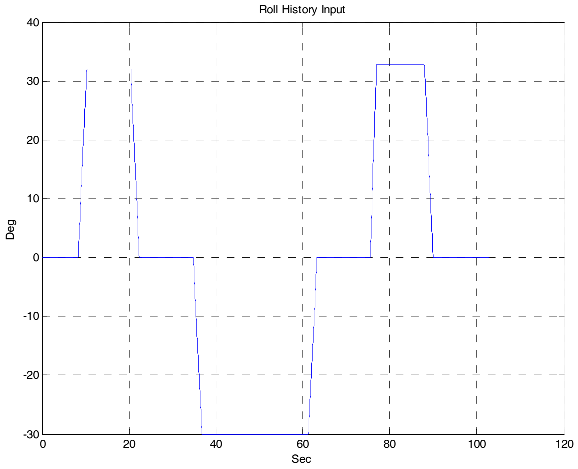


Figure 3-3 : Roll Angle Input to the system

As seen from the above figure, in the input the change in roll angles is represented as a ramp function with a slope of 15°/sec. This is because of two reasons. First of all, in the designed structure there is a direct coupling between input and "Roll Angle Acc". Therefore, any step in the input, causes the roll angle acceleration output not to be continuous which leads to computational errors in generating trajectory. Secondly, during the real flight, the roll rate is determined by the pilots, and the recorded data shows that it is generally preferred to roll with a rate of approximately 15°/sec.

The feedback gains are adjusted with trials depending on experience. Best results are obtained with K1=1.7 and K2=1.3, which corresponds to unity steady state gain. Comparison of simulated roll angle with these coefficients and real roll angle history recorded during flight test is represented in Figure 3-4.

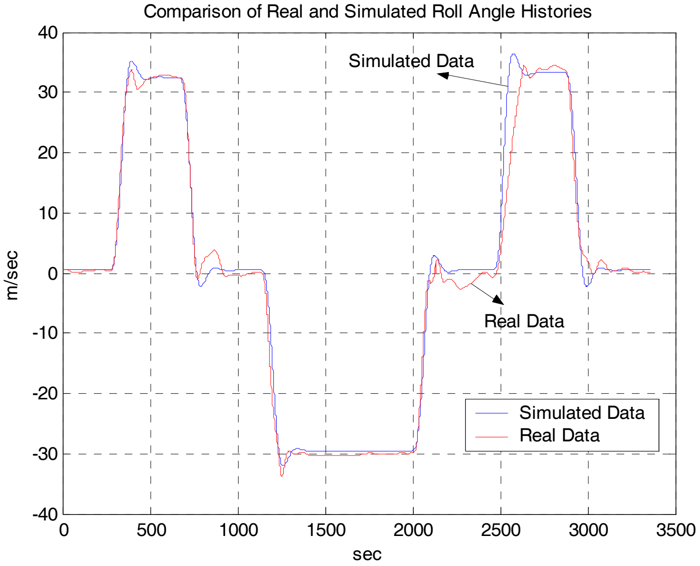


Figure 3-4 : Comparison of simulated and real roll angle histories

As shown in the Figure 3-4, the designed roll rate generator structure is successful enough to follow real data. The difference in the last roll is due to the fact that, in order to make total net heading change 0°, a slower bank-to-turn was performed at the last segment.

The comparisons of recorded and simulated IMU outputs are presented at Figure 3-5 and Figure 3-6.

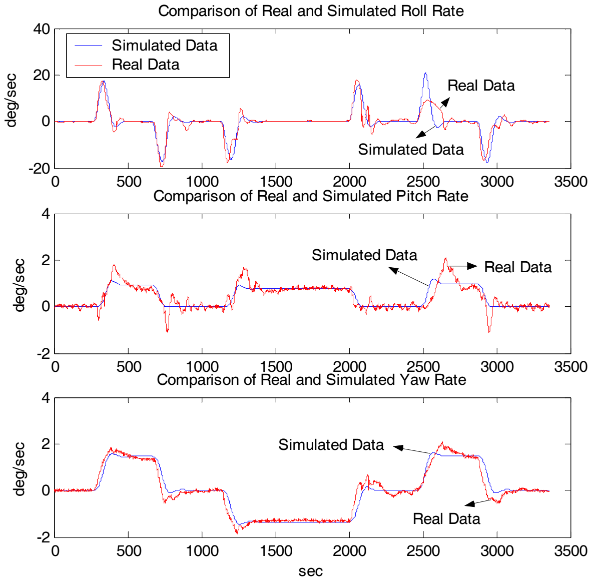


Figure 3-5 : Comparison of real and simulated gyroscope outputs

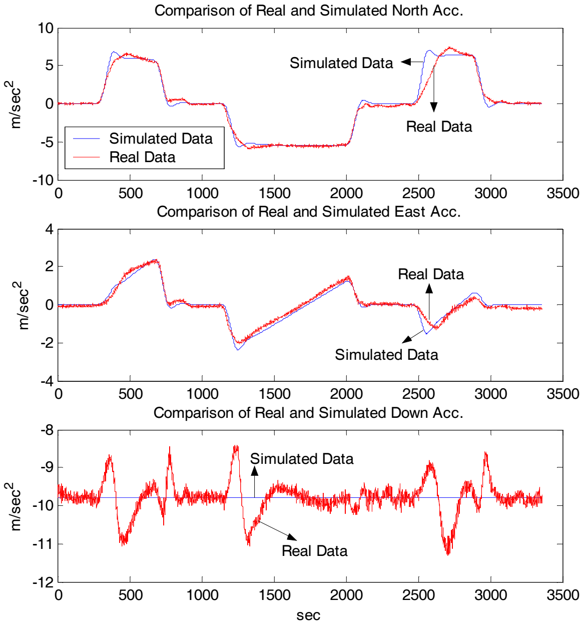


Figure 3-6 : Comparison of real and simulated North-East-Down Accelerations

As seen from these figures, trajectory generator is fairly successful in generating IMU outputs for an aircraft which performs coordinated turn maneuver. On the other hand, there is an apparent difference in the vertical acceleration. (Note that acceleration outputs are compared in NED frame instead of body frame). This is because, in generating trajectory, it was assumed that weight and lift vectors are always equal so that aircraft can complete its maneuvers without any altitude change. However, recorded flight data showed that, this is generally not true for an aircraft. Any roll change causes the aircraft to lose or gain altitude which cause some additional acceleration to appear in vertical channel. Moreover, in the trajectory generator, it was assumed that the angle between “x” axes of aircraft body and total velocity vector is zero. But recorded data shows that, during maneuvers this angle can take values between 0 to 7 degrees which also contributes the difference between simulated and real data.

3.2 KINEMATIC RELATIONS BETWEEN MASTER AND SLAVE

In a typical transfer alignment application, the reference system (master) and the system to be aligned (slave) are located apart from each other on the aircraft. Hence, during the flight, two navigation systems undergo different motion and their outputs are different from each other. For the master navigation system to be used as a reference for slave system, the outputs of master should be compensated for the difference arise from physical separation. This process is known as lever arm compensation.

In this section, kinematic relations between master and slave navigation systems that are located apart from each other are derived.

3.2.1 Velocity Relation

Let “r” represents the relative position of slave with respect to master. Then, relative position of master and slave systems can be represented in inertial frame of coordinates as follows:

$$R_M^i = R_S^i + r^i \tag{3.23}$$

The navigation systems that are under consideration define velocity with respect to earth. Therefore, above relation should be handled in Earth Centered Earth Fixed Frame (ECEF).

$$R_S^e = R_M^e + r^e \tag{3.24}$$

Take the time derivative of both sides:

$$V_S^e = V_M^e + \frac{dr^e}{dt} \tag{3.25}$$

For a fighter aircraft the maximum separation between master and slave systems is 15 meters. For this distance, the navigation frame of reference (NED Frame) defined for the 2 systems can be considered to be same. Therefore, above relation can be written in navigation frame as follows:

$$V_S^N = V_M^N + C_E^N \frac{dr^E}{dt} \tag{3.26}$$

The lever arm vector “r” can be defined in master’s body frame of reference. Hence;

$$V_S^n = V_M^n + C_e^n \frac{dC_M^e r^M}{dt} \tag{3.27}$$

Take the derivative explicitly:

$$V_S^n = V_M^n + C_e^n \left[\frac{dC_M^e}{dt} r^M + C_M^e \frac{dr^M}{dt} \right] \tag{3.28}$$

$$V_S^n = V_M^n + S\left(\omega_{eM}^n\right) r^M + C_M^n \frac{dr^M}{dt} \tag{3.29}$$

Rearranging the above equation, the relation for velocities is found to be as follows:

$$V_S^n = V_M^n + C_M^n \left[(\omega_{iM}^M - \omega_{ie}^M) \times r^M \right] + C_M^n \frac{dr^M}{dt} \tag{3.30}$$

In the above derivations all parameters related with lever arm vector are defined in master’s body frame of reference. In some circumstances, it is not possible to get these values from master navigation system. For such a situation, Equation (3.30) should be revised by defining “r” in slave’s body frame of reference. In this case, Equation (3.30) takes the following form:

$$V_S^n = V_M^n + C_S^n \left[(\omega_{iS}^S - \omega_{ie}^S) \times r^S \right] + C_S^n \frac{dr^S}{dt} \tag{3.31}$$

Due to the much better navigation accuracy of master system and considering the time origin consistency of parameters used in equations, Equation (3.30) is always preferred to the Equation. (3.31), whenever possible.

In the above equations, "dr/dt" term represents the relative velocity of slave with respect to master due to non-rigid structure of the wing excited by vibration and flexural motion. In order to prevent this term directly appear on the velocity relation, the following velocity formulation can also be used instead on equation (3.30).

In equation (3.27) define "r^e" in navigation frame of reference. Hence;

$$V_S^n = V_M^n + C_e^N \frac{dC_n^e r^n}{dt} \tag{3.32}$$

Perform the differentiation and rearrange the terms;

$$V_S^n = V_M^n + S\left(\omega_{en}^n\right) r^n + \dot{r}^n \tag{3.33}$$

The usefulness of above equation comes from the fact that, it directly defines the integral of velocity differences in terms of readily calculated quantities. This point will further be investigated in Chapter 5.

3.2.2 Attitude Relation

As indicated in the above section, navigation frame of reference for the two locations can be considered to be same. Therefore, the relation between body to NED Frame direction cosine matrices defined for the master and slave system can be represented as follows:

$$C_S^n = C_M^n C_S^M \tag{3.34}$$

It should be noted that, "C_S^M" direction cosine matrix represents the relative orientation of slave with respect to master system. Because of structural flexure and vibration of wing, however, the value of this matrix changes dynamically which is governed by the following differential equation:

$$\dot{C}_S^M = S\left(\omega_{MS}^M\right) C_S^M \tag{3.35}$$

3.2.3 Acceleration Relation

In order to find the acceleration relations defined in inertial reference of frame, inertial position vectors should be differentiated twice. This process can be accomplished as follows:

The position of slave can be defined as follows

$$R_S^i = R_M^i + C_M^i r^M \tag{3.36}$$

Differentiate the both sides of above equation;

$$V_S^i = V_M^i + C_M^i S\left(\omega_{iM}^M\right) r^M + C_M^i \frac{dr^M}{dt} \tag{3.37}$$

To obtain the acceleration, take the derivative of above equation again;

$$a_S^i = a_M^i + \frac{dC_M^i S\left(\omega_{iM}^M\right) r^M}{dt} + \frac{dC_M^i \dot{r}^M}{dt} \tag{3.38}$$

After performing indicated differentiations and rearranging the resultant term, the following equation is obtained:

$$a_S^i = a_M^i + C_M^i \left\{ \omega_{iM}^M \times \omega_{iM}^M \times r^M + \dot{\omega}_{iM}^M \times r^M + 2\left(\omega_{iM}^M \times \dot{r}^M\right) + \ddot{r}^M \right\} \tag{3.39}$$

An accelerometer defines the accelerations in its body frame of reference. Hence, to obtain the relation between accelerometer outputs located at master and slave positions, above equation should be multiplied with "C_i^S" DCM:

$$a_S^S = C_M^S \left[a_M^M + \omega_{iM}^M \times \omega_{iM}^M \times r^M + \dot{\omega}_{iM}^M \times r^M + 2\left(\omega_{iM}^M \times \dot{r}^M\right) + \ddot{r}^M \right] \tag{3.40}$$

As seen from the above equation, the relation between 2 accelerometers depends on "$\dot{r}$" and "$\ddot{r}$", both of which is a function of vibratory motion of the wing. Therefore, the most effected term from vibration turns out to be acceleration of the slave system.

3.2.4 Rotation Rate Relation

Rotation rates are vectorial quantities. Therefore, rotation rate relation of master and slave systems can be found by using simple vector addition as follows:

Let "ω_{MS}^M" represent the rotation rate of slave with respect to master defined in master's body frame of reference, then

$$\omega_{iS}^S = C_M^S \left(\omega_{iM}^M + \omega_{MS}^M\right) \tag{3.41}$$

3.2.5 Position Relation

The relation between latitude, longitude and height of the master and slave systems can be calculated with a lever arm defined in master frame of reference as follows:

$$\begin{bmatrix} L \\ lon \\ h \end{bmatrix}^S = \begin{bmatrix} L \\ lon \\ h \end{bmatrix}^M + \begin{bmatrix} 1/(R_N + h) & 0 & 0 \\ 0 & 1/[\cos L(R_E + h)] & 0 \\ 0 & 0 & -1 \end{bmatrix} C_M^n r^M \qquad (3.42)$$

3.3 CHARACTERIZATION OF VIBRATION ENVIRONMENT

As seen from the previous section, in order to perform accurate lever arm compensation it is necessary to know the relative motion of slave with respect to master navigation system. However, in practice, it is not possible to calculate this relative motion during the captive flight. Hence, the effect of relative motion appears as a noise in the lever arm compensation process which limits the effectiveness of any kind of transfer alignment algorithm. As a matter of fact, transfer alignment is the process of transferring navigation values obtained form master navigator to the slave navigator in the presence of unknown relative motion. In an aircraft configuration, that relative motion mostly induced by the non rigid structure of the aircraft wing and is one of the most critical factors that determine the overall estimation performance of any transfer alignment algorithm. Therefore, every transfer alignment algorithm should be assessed under a realistic vibration environment.

On the other hand, the realistic characterization of the vibration environment is a very challenging problem. Usually it is not possible to derive mathematical models which can reflect the vibration environment of an aircraft for real flight conditions. From the related literature, it can be inferred that several studies were conducted on this issue. However, except the general guidelines about their methodology, none of the results were published open so far. The two most explicit works in the literature are Kain (1989) and Spalding (1992). However, they are also lack of completeness. In the related papers, generally, the effect vibration and flexure on acceleration and rotation rates are represented using third order Markov models with different system parameters which are claimed to be determined using some sort of test data.

In this theses study, the vibration models presented in Appendix B are utilized. These models represent the effect of vibration on acceleration and rotation rate values as stochastic system outputs. In this approach, the very low frequency structural changes due to the change in loading is named as flexure, and characterized by deterministic models. Considering the short operation duration, the errors on these models are taken to be in the form of random constants (or very nearly random constant) (refer to Chapter 4.4). On the other hand, the high frequency vibration effects are characterized by using stochastic models which represent the vibration induced acceleration and rotation rates. Therefore, in order to specify the effect of vibration on slave acceleration and rotation rate, slave IMU outputs are calculated in two separate steps. In the first step, theoretical slave IMU outputs which are connected to the master navigator via a rigid structure are calculated. By summing the outputs of vibration models with these rigidly connected IMU outputs, theoretical acceleration and rotation rates sensed by a slave inertial measurement unit under vibration environment is obtained. By processing those IMU outputs with navigation algorithms, theoretical PVA solution of a slave navigator can also be calculated under vibration environment.

As shown in Figure 3-7, the vibration induced acceleration and rotation rates are modeled as a sum of appropriate number of Markov processes. In an INS structure, accelerometer outputs are integrated twice whereas gyroscope outputs are integrated only once. Therefore, in order to have stable integrals, vibration on acceleration and rotation rate is modeled by using third and second order Markov processes respectively.

In Figure 3-7, the model used to generate vibration induced acceleration on “x” axes of slave body frame of reference is shown.

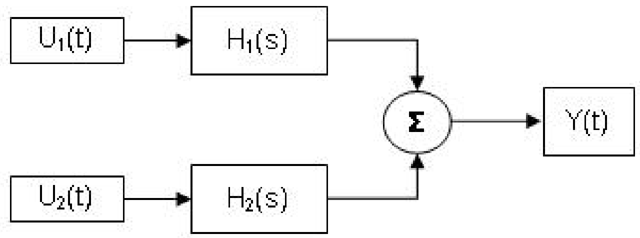


Figure 3-7 : Vibration model for X axis acceleration

In the above figure, $U_1(t)$ and $U_2(t)$ represent independent zero mean unity variance white Gaussian noises, whereas $H_1(s)$ and $H_2(s)$ represent shaping filters with the following form:

$$\mathrm{H}(s) = \frac{\sqrt{N} s^2}{s^3 + As^2 + Bs + C} \tag{3.43}$$

Y(t) denotes the output of the system which is assumed to have the same power spectral density characteristics as the vibration induced acceleration on the “x” axes. Therefore in the simulation environment Y(t) is used as the vibration induced acceleration on “x” axes.

For each axes, similar structures as shown in Figure 3-7, is used to generate vibration induced acceleration and rotation rates. However, for each axes, the shaping filters have different characteristics. This is because, at each axes the spectra of vibration induced motion appears to group around different major (center) frequencies. Therefore for each axes different filter coefficients and different number of filters are used to generate vibration induced motion.

The structure and filter parameters of vibration induced acceleration and rotation rate models for all three axes used in this study are presented in Appendix B. It should be noted that, the models presented in Appendix B also contains the effect of a theoretical mechanical isolator.

On the other hand, the most important deficiency of the models utilized in this study is their inability to model low frequency components accurately. The models presented in Appendix B were derived by using a real test data based on some acceleration and rotation rate sensors outputs. Therefore the models cannot represent low frequency vibration effects as good as they represent high frequency effects.

This can easily be seen from the fact that it is possible to construct 2 different models with almost same PSD shape but with different second integral statistics. This can be shown by changing the smallest pole of derived models. For example the poles of $H_1(s)$ defined for “x” axes vibration induced acceleration are as follows (see Appendix B):

p_1=-51.7943 +46.8018i

p_2=-51.7943 -46.8018i

p_3=-3.4014

Let (p_3) is replaced by “-0.0.05”. If the numerator is kept constant, then $H_1(s)$ becomes:

$$\mathrm{H}_1^{'}(s)=\frac{6.1424\,s^2}{s^3+103.59\mathrm{s}^2+4873.6s+24.365} \tag{3.44}$$

In this case the maximum difference between the squared magnitudes of transfer functions reduces to below 10 percent after 3Hz and below 1 percent after 7Hz. On the other hand, when excited by unity variance white noise, steady state standard deviations of second integrals of model outputs turn out to be as follows:

σ = 0.00047377 (For the original $H_1(s)$)

σ’= 0.012604 (For the modified $H_1(s)$)

As seen from the above results, by adjusting smallest pole of transfer function, it is possible to adjust steady state variance of second integrals of model outputs without disturbing original model PSD. Therefore, it is possible to generate a variety of different vibration induced acceleration models which has same PSD characteristics but have different first and second integral characteristics. This shows that, the models utilized in this study are not able to model vibration induced displacement and velocity effects very accurately.

On the other hand, it should be noted that the above procedure can be used to incorporate the knowledge of maximum position deflection information to the obtained acceleration models. However, in the scope of this study, it is preferred to handle the effect of vibration induced displacement as a part of displacement induced by flexure effects and therefore modeled as random constants (refer to Chapter 4.4).

3.4 ANALYSIS OF VIBRATION DEPENDENT ERRORS

Under vibration environment, some additional error effects appear in inertial navigation systems. These errors can be grouped as follows:

i. Instrument Errors

ii. Algorithmic Errors

Instrument errors arise due to the fact that, inertial sensors' dynamic responses can change under vibration environment. In addition to this, sensor errors which are a function of product of two inertial measurements (such as anisoelastic bias error for gyroscopes and vibro-pendulous error for accelerometers) cause to an additional bias/drift error under oscillatory motion.

Therefore, during the captive flight, error characteristic of slave IMU system will be different from the free flight during which system exposes to lower vibration than captive flight. Any estimation algorithm that is used to estimate net bias error during captive flight will converge to a different result with respect to free flight conditions. This phenomenon especially occurs for accelerometers and called as vibration rectification. Therefore, after the transition from captive to free flight, the quality of estimation algorithm results become worse than what is calculated.

On the other hand, the complete treatment of this issue is out of scope of this thesis. In this study (where necessary), this point is simply accounted by artificially increasing error variances of estimated values after the transition to free flight.

Algorithmic errors mainly consist of errors occur during strapdown calculations due to the imperfect discretization of computational elements. On the other hand, the inclusion of coning and sculling algorithms greatly reduces the effect of these errors. An extremely detailed analysis of vibration environments on the algorithmic computations is presented in Savage (2000). Therefore, in this study, only the performance of selected coning and sculling algorithms (shown in Chapter 2) under the vibration environment derived in previous section is presented using the results given in Savage (2000).

In Appendix A, it is shown that, the total and algorithmic coning and sculling errors can be calculated using power spectral densities of rotation rates and accelerations acting on the system. When Equation (A-34) and (A-35) was evaluated using PSD of x and z axis gyroscope output models presented in Appendix B with “1/600” second increment period and “1/200” computational interval, the total coning error during “1/100” second was found to be 0.0452°/hour, whereas total algorithm error was found to be 1.1243e-006°/hour. Although, the total coning error is found to be less than expected, the ratio between true and algorithmic error indicates that, the implemented coning compensation algorithm are very successful to compensate net gyro drift occur due to the coning environment.

Similar calculations are also carried to find numerical values for sculling errors under the environment characterized with the vibration models presented in Appendix B. To this extent, x axes gyroscope output and z axes accelerometer outputs were used in the calculations with “1/600” sec increment period and “1/200” computational interval. Using Equation (A-36) and (A-37) total sculling error and algorithm error during “1/100” second was found to be 9.4003e-005m/s^2 and 3.7529e-009m/s^2 respectively. Again, these results verify that, net bias effect appear due to the sculling motion can be greatly reduced by implementing sculling compensation algorithms.

Comparing the effect of coning and sculling errors, it can be seen that, under vibration environment conning errors are more dominant than sculling errors, and the effect of these errors can be greatly reduced by the addition of conning/sculling compensation algorithms. However, it should be noted that, instrument errors for accelerometers which are excited under vibration can be much higher than sculling errors. Therefore, in the design of estimation algorithm, the effect of vibration on instrument errors should always be considered.

CHAPTER IV

DESIGN OF TRANSFER ALIGNMENT ALGORITHMS

4.1 INTRODUCTION

As explained in Chapter 1, the main function of transfer alignment algorithms is to estimate both slave IMU calibration parameters and slave INS PVA solution errors during the captive flight of the slave. This is achieved by comparing similar vectors obtained from master and slave navigation systems outputs. As it is assumed that master navigator is almost errorless, any difference between master and slave systems outputs is an indication of slave system errors.

On the other hand, not all errors of slave system are directly coupled with navigation outputs. For instance, it is not possible to calculate gyroscope drift errors by just using the velocity differences between the two systems. This is because, drift errors generates attitude errors and this attitude error generates a velocity error. In order to estimate those indirect errors, an estimation algorithm which can process the observed differences for sufficiently long time is necessary.

The most significant factor that affects the design of such an estimation algorithm is the selection of vector types that are compared. The choice of vector type specifies the overall structure of any estimation algorithm. Therefore, transfer alignment algorithms are classified according to the type of vectors compared to generate an estimate. According to this criterion, transfer alignment algorithms can be classified as follows:

i. Acceleration / Rotation Rate Matching
i. Velocity Matching
ii. Integrated Velocity Matching
iii. Attitude Matching
iv. Velocity and Attitude Matching
v. Position Matching

Each of the above groups is named according to the type of the observation vectors used in the estimation algorithms. In the following sections, each of the above methods is described separately except position matching method. This is because, in real systems, due to the GPS aided and barometric damped nature of position calculations, there are big variations between samples of position outputs of master navigator. Therefore, it is not possible to use position vectors directly as observation in estimation algorithms. Furthermore, both the integrated velocity matching and position matching methods are conceptually very similar methods. So the analysis of integrated velocity matching method also provides sufficient information about effectiveness of position matching method. Thus, in this study, position matching method is not considered.

In the literature, there are also some less utilized algorithms such as doubly integrated velocity matching (Stovall (1996)) or velocity and rotation rate matching methods (Rogers (1991)). However, because of the reasons stated in the following sections, it is not possible to implement these techniques for real systems. Therefore, in this study they are not considered alone and their effectiveness are analyzed in the scope of other methods.

In almost all kind of transfer alignment procedures, Kalman filters are used as the primary method of estimation. This is because, Kalman Filters are optimal filters and they constitute a general framework for all kind of recursive filters including deterministic recursive least square filters (Haykin (2002)). Therefore, in the following discussions, for each kind of method the Kalman Filter structures are presented first and the performance of each method is compared by using Kalman filter estimates which utilize corresponding matching method.

The application of a generic Kalman filter in a typical transfer alignment problem is represented in the following figure.

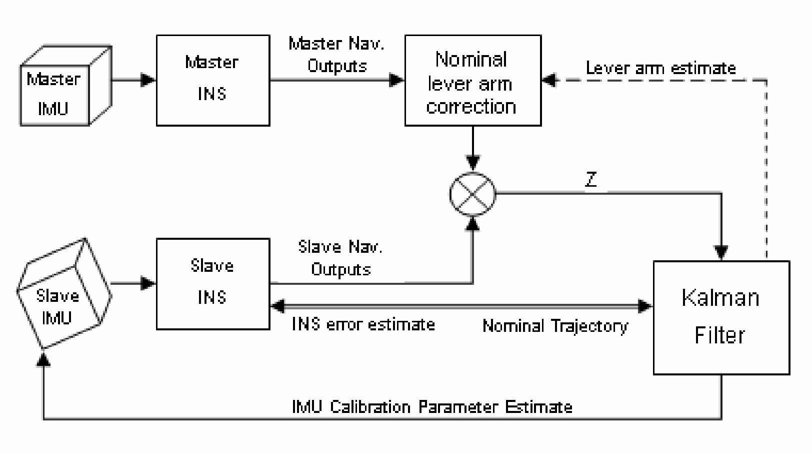


Figure 4-1 : Application of a Kalman filter in transfer alignment problem

As indicated in Chapter 1, in this study, the transfer alignment algorithms are designed by considering air to ground guided munitions applications. Therefore, it is assumed that aircraft's main navigator serves as master navigation system, whereas ammunition's inertial navigator serves as the slave system. As these two systems are located apart from each other on the aircraft, a lever arm correction should be made on the master navigator output before a comparison is made. As shown in Figure 4-1, the difference between the slave and corrected master navigator outputs are processed by the Kalman filter to generate estimates about the error on slave PVA solution, slave IMU calibration parameters and the lever arm vector between master and slave systems. In some applications, lever arm vector can be known accurately prior to start of transfer alignment procedures, and therefore estimation of lever arm vector is not necessary for every application.

Kalman filters require a mathematical propagation model for the states they try to estimate. On the other hand, as the navigation equations are nonlinear, they cannot be used as propagation models. Therefore, Kalman filter system model is obtained by linearizing the navigation equations around a nominal trajectory. This form of Kalman filter is called as extended Kalman filter. These linearized equations model the propagation of errors about the given nominal trajectory. Hence extended Kalman filters are used to estimate the amount of errors on states instead of the states itself (Brown, (1997)). The structure and computational steps of this type of Kalman filters are summarized in Appendix C.

As show in Figure 4-1, the nominal trajectory used by the Kalman filter is supplied by the slave INS. Therefore, Kalman filter is used to estimate errors of slave INS. Furthermore, together with the navigation errors, as it is also required to estimate IMU calibration parameters and lever arm errors, the propagation models for these variables should be augmented to the Kalman filter system model too. In Chapter 2, it was explained how to linearize navigation equations and augment IMU models to these linearized equations. In the following table a summary of the states and the corresponding propagation models used in a typical transfer alignment Kalman filter are presented.

Table 4-1 : Summary of system states used in Kalman filter

Name of State	Error Propagation Model
North, East, Down Vel. Err.	$\delta\dot{V} = C_b^n \delta a^b - S(\varphi) C_b^n a^b - (2\omega_{ie}^n + \omega_{en}^n) \times \delta V - (\delta\omega_{en}^n) \times V$
Roll, pitch yaw err.	$\dot{\varphi} = -\omega_{in}^n \times \varphi + \delta\omega_{in}^n - C_b^n \delta\omega_{ib}^b$
Acc bias rep. Err.	$\delta\dot{a}_{BR} = 0$
Acc. Scale Factor Rep. Err	$\delta\dot{a}_{SFR} = 0$
Acc. Bias Stability Err.	$\delta\dot{a}_{BS} = -\frac{1}{\tau_{DS}} \delta a_{BS} + n_{BS}$
Gyro. Drift Rep.	$\delta\dot{g}_{DR} = 0$
Gyro Scale Factor Err.	$\delta\dot{g}_{SFR} = 0$
Gyro. Drift Stability Err.	$\delta\dot{g}_{DS} = -\frac{1}{\tau_{DS}} \delta g_{DS} + n_{DS}$
Static lever arm err	$\delta\dot{r} = 0$

The states shown in Table 4-1 constitutes the common states used in all transfer alignment methods listed above. Together with these states, some additional states are also augmented to the above system model depending on the matching method used in transfer alignment procedure.

In order to asses the performance of each implemented algorithm under different error and environmental characteristics a transfer alignment simulation environment is developed. The simulation environment is capable of generating both theoretically true and erroneous navigation values for master and slave navigation systems of an aircraft which follows a predetermined path. By processing the erroneous slave navigation values with the transfer alignment algorithms and by comparing the algorithm results with the theoretically true navigation values the performance of each algorithm is evaluated. Furthermore, by performing Monte Carlo analysis using the simulation environment, the sensitivity of each algorithm to the several unmodelled environmental factors is also characterized.

In Chapter 4.1, the structure of developed simulation environment is presented. In the rest of this chapter, each transfer alignment method listed above is derived and explained one by one.

4.2 TRANSFER ALIGNMENT SIMULATION ENVIRONMENT

As shown in Chapter 2, navigation equations have non – linear and time varying nature which makes the error equations obtained from them highly complicated to derive analytical relations. Moreover, there are lots of environmental error sources that cannot be modeled as simple mathematical models. Therefore, usually it is not possible to make a theoretically true covariance analysis.

Hence, Monte Carlo analysis turns out to be an important tool to assess the real performance of developed transfer alignment algorithms. Using Monte Carlo analysis, not only the performance of the filters is evaluated but also, the sensitivity of the filter to unmodeled/unexpected error sources can also be discovered. Therefore, Monte Carlo analyses are indispensable tools in the development of transfer alignment algorithms.

On the other hand, in order to perform a Monte Carlo analysis, a simulation environment in which all kind of error sources are accurately modeled and included in the simulation is required. Furthermore, this simulation environment must be able to generate true navigation values as well. Therefore, developing a simulation environment can be even much harder than developing the transfer alignment algorithm itself.

In this section, the main components of the implemented simulation environment used in this study are explained. In the following figure, the general structure of simulation environment is presented.

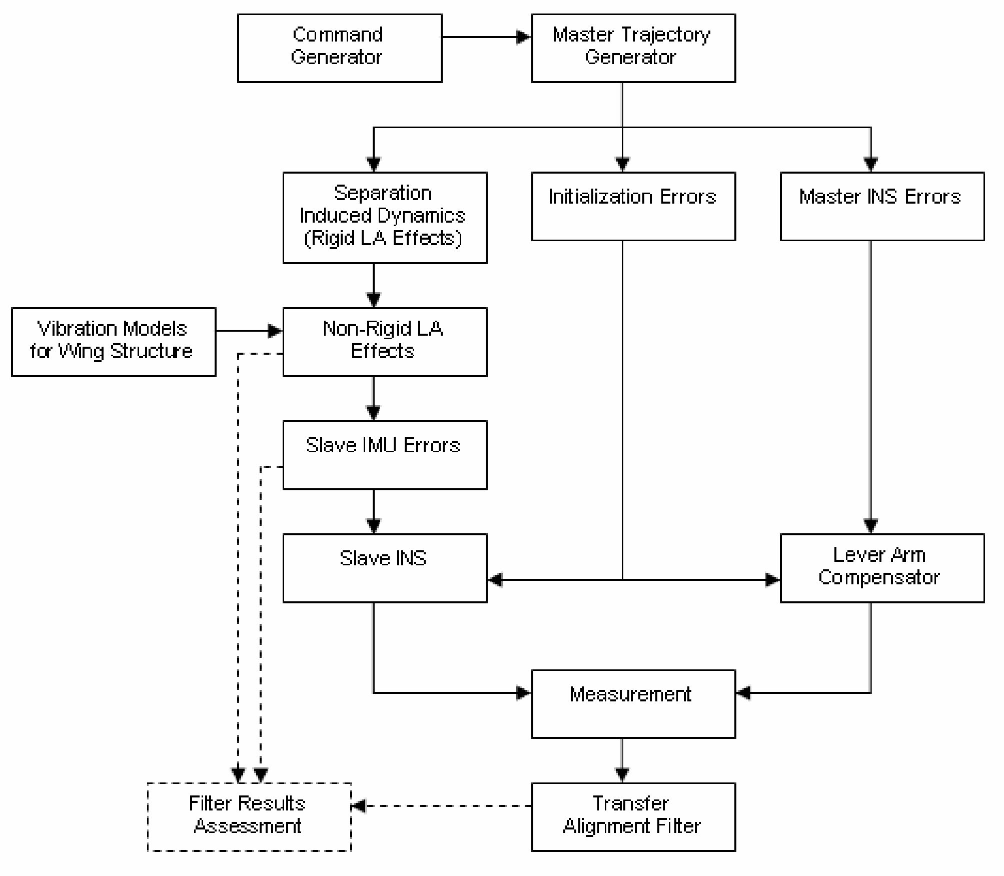

Command Generator
Master Trajectory Generator
Separation Induced Dynamics (Rigid LA Effects)
Initialization Errors
Master INS Errors
Vibration Models for Wing Structure
Non-Rigid LA Effects
Slave IMU Errors
Slave INS
Lever Arm Compensator
Measurement
Filter Results Assessment
Transfer Alignment Filter

Figure 4-2 : Block Diagram of Transfer Alignment Simulation Environment

The function of each block can be summarized as follows:

i. Command Generator Module

This module is the trajectory shaping part of trajectory generator. Using the system shown in Figure 3-2 and Equation (3.21), it generates required Euler angle derivatives that comply with the desired trajectory shape supplied by the user.

ii. Master Trajectory Generator Module

Using Equation (3.2), (3.3), (3,6), (3.7) and (3,19) and Euler angles derivatives provided by Command Generator module (refer to Figure 3-1), it generates master system's true position, velocity and attitude solution as well as corresponding IMU outputs for master IMU.

iii. Separation Induced Dynamics Module

Using Equation (3.30), (3.34), (3.40), (3.41) and (3.42), it transforms the navigation solutions and IMU outputs defined for master system to the slave system. During these transformations, this block assumes that master and slave systems are connected via a rigid structure (omits the effect of "$\dot{r}, \ddot{r}, \omega_{MS}$").

iv. Vibration Models Module

This block simulates the relative motion of slave with respect to master (vibration of slave). In order to perform this operation, it uses stochastic models presented in Appendix B. As explained before, these models represent the effect of vibration on acceleration and rotation rates. Thus, transfer functions defined for the vibration models were converted to the state space representation in controllable form so that, each state in state space form corresponds to vibration induced acceleration, velocity and displacement for third order models and rotation rate and angle increment for second order models. Therefore, this module is capable of generating the instantaneous values of "r_{Dyn}", "$\dot{r}$", "$\ddot{r}$", "ω_{ms}^{m}". "r_{Dyn}" is the integral of "$\dot{r}$" which represents the dynamical changing component of lever arm vector (for more information refer to Chapter 4.4), whereas "$\ddot{r}$" denotes the vibration induced acceleration. "ω_{ms}^{m}" correspond to vibration induced rotation rate of slave with respect to master.

v. Non Rigid Lever Arm Module

Using the random variable values provided by Vibration Models module, it adds the effect of vibration on the slave system's navigation outputs by inserting the values of "r_{Dyn}", "$\dot{r}$", "$\ddot{r}$", "ω_{ms}^{m}" in Equation (3.30), (3.35), (3.40) and (3.41). It should be noted that the lever arm vector "r" appear in these equation can be calculated using Equation (4.25) (refer to Chapter 4.4).

As explained in Chapter 2, implemented navigation system uses velocity and angle increments to calculate instantaneous PVA solutions. To calculate the velocity increment (integral of raw accelerometer output) in "1/600" second duration, Equation (3.40) should be integrated as follows:

$$u = \int_{T}^{T+1/600} a_S^S(t)\,dt = u_1 + u_2 \tag{4.1}$$

$$u_1 \cong C_M^S \left[a_M^M + \omega_{iM}^M \times \omega_{iM}^M \times r^M + \dot{\omega}_{iM}^M \times r^M + 2\left(\omega_{iM}^M \times \dot{r}^M\right)\right]\frac{1}{600} \tag{4.2}$$

$$u_2 = C_M^S \dot{r} \tag{4.3}$$

Otherwise, if "u" is calculated using just multiplying Equation (3.40) with "1/600" seconds, the computational error induced by numerically integrating acceleration values can change error characteristics of generated IMU outputs so much that simulation environment produces unreliable results.

vi. Slave IMU Error Module

This module adds error to the generated true slave IMU outputs according to the Equation (2.6) and (2.7). The error definitions are given in Table 2-1.

vii. Slave INS Module

This module calculates the position, velocity and attitude of slave system by processing erroneous slave IMU outputs supplied by slave INS module and erroneous initial values supplied by initialization module.

viii. Initialization Errors Module

This module calculates initial position, velocity and attitude of slave system using erroneous lever arm and relative orientation values. This initialization can be represented as follows:

$$\tilde{V}_S^n = \tilde{V}_M^n + \tilde{C}_M^n \left[\tilde{\omega}_{eM}^M \times \tilde{r}^M\right] \tag{4.4}$$

$$\tilde{C}_S^n = \tilde{C}_M^n \tilde{C}_S^M \tag{4.5}$$

$$\begin{bmatrix} L \\ lon \\ h \end{bmatrix}^S = \begin{bmatrix} L \\ lon \\ h \end{bmatrix}^M + \begin{bmatrix} 1/(R_N + h) & 0 & 0 \\ 0 & 1/[\cos L(R_E + h)] & 0 \\ 0 & 0 & -1 \end{bmatrix} \tilde{C}_M^n \tilde{r}^M \tag{4.6}$$

where $\tilde{r}^M$ and $\tilde{C}_S^M$ represents the erroneous lever arm and relative orientation values. It is assumed that, mission computer of aircraft can supply these values with an accuracy specified by the standard deviation values shown in Table 2-2. Therefore this module adds error on correct lever arm and relative orientation values which comply with these specifications and supply this information to Lever-arm Compensator Module.

"$\tilde{V}_M^n$" and "$\tilde{C}_M^n$" denotes the instantaneous velocity and attitude outputs of master navigator. It was assumed that these instantaneous values have jitter type errors with standard deviation values presented in Table 2-3 (these errors are generated in Master INS Error Module).

It should be noted that, Equation (4.4), (4.5) and (4.6) constitutes the sufficient equations required to complete one-shot transfer alignment procedure described in Chapter 2.1. As seen from these equations, even if a perfect master navigator is used, due to the errors on lever arm and relative orientation values, initial values of slave navigator become erroneous.

ix. Master Navigator Errors Module

This module adds EGI errors on generated master solution and IMU outputs. The errors added to the master outputs are selected to be white Gausian noise with variance values given in Table 2-3.

In this simulation structure, the effect of environmental conditions on master's solutions was totally ignored and their effects were considered as a part of master INS errors. This is because, modeling such effects can be extremely complex, and therefore is assumed to be out of scope of this thesis.

x. Lever Arm Compensator

Using erroneous lever arm vector and relative orientation information supplied by the initialization module, it calculates the lever arm compensation on master navigation outputs. The compensated master navigator outputs are then used as reference values for Transfer Alignment Filter.

xi. Measurement Module

This module calculates the observations such as velocity or attitude differences which are then used in transfer alignment filter. The calculations of observations for each transfer alignment method are described in following sections.

xii. Transfer Alignment Filter

The estimation process of Slave INS and IMU errors is accomplished in this module. Although not represented in the figure, the estimated error values are used to correct related navigation states in a close loop structure.

xiii. Filter Results Assessment

By comparing true and estimated error values, this module evaluates the performance of implemented transfer alignment filter structure.

In the following sections, the performance of each transfer alignment algorithms is evaluated using the above simulation environment. The nominal maneuver for each simulation is taken to be a C-shaped maneuver which last for 100sec and consist of consecutive straight flight and horizontal coordinated turn maneuvers. The aircraft velocity during these maneuvers is adjusted to be approximately 210m/s. The roll angle profile of the maneuver is presented in the following figure.

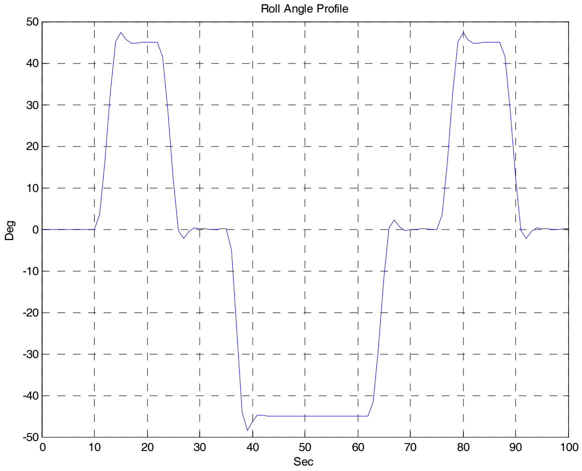


Figure 4-3 : Roll angle profile used in algorithm performance assessment simulations.

4.3 ACCELERATION / ROTATION RATE MATCHING METHOD

As the name implies, in this method, master and slave systems' IMU outputs are compared to form an observation. The comparison results are used as measurement in a Kalman filter structure to find misalignment between the two systems. The misalignment angles between two systems are dynamically changing due to the random vibration and flexure. Therefore, first, it is necessary to derive differential equation that governs the change in the misalignment angle.

Let the master to slave transformation be represented by the following Euler angles which is defined in slave body frame of reference with the rotation order of z,y,x:

$$\mathrm{E} = \begin{vmatrix} \alpha \\ \beta \\ \gamma \end{vmatrix} \tag{4.7}$$

The relation between the derivative of these Euler angles and rotation rate of master with respect to slave can be found as follows (Titterton, (1997)):

$$\begin{aligned} \dot{\alpha} &= \left(\cos\alpha\omega_x + \sin\gamma\omega_y\right)\sec\beta \\ \dot{\beta} &= \omega_y \cos\alpha - \sin\gamma\omega_x \\ \dot{\gamma} &= \omega_z + \tan\beta\left(\cos\gamma\omega_x + \sin\gamma\omega_y\right) \end{aligned} \tag{4.8}$$

where

$$\omega_{SM}^{S} = C_M^S(\mathrm{E})\omega_{IM}^{M} - \omega_{IS}^{S} = \left[\omega_x \ \omega_y \ \omega_z\right]^T \tag{4.9}$$

If it is assumed that Euler angles are small such that "$\sin\mathrm{E} \approx \mathrm{E}$" and "$\cos\mathrm{E} \approx 1$", then by neglecting higher order terms, Equation (4.8) can be written as:

$$\begin{aligned} \dot{\alpha} &= \omega_x - \gamma\omega_y \\ \dot{\beta} &= \omega_y - \gamma\omega_x \\ \dot{\gamma} &= \omega_z - \beta\omega_x \end{aligned} \tag{4.10}$$

Using small angle assumption, master to slave transformation matrix can be written as follows:

$$C_M^S(\mathrm{E}) = \mathrm{I} + S(\mathrm{E}) = \mathrm{I} + \begin{vmatrix} 0 & -\gamma & \beta \\ \gamma & 0 & -\alpha \\ -\beta & \alpha & 0 \end{vmatrix} \tag{4.11}$$

Inserting Equation (4.11) into (4.9) and combining (4.9) with (4.10) and after rearranging the terms following equation can be obtained:

$$\dot{\mathrm{E}} = \begin{vmatrix} 0 & \omega_{iM}^{M}(z) & -\omega_{iS}^{S}(y) \\ \omega_{iM}^{M}(z) & 0 & \omega_{iS}^{S}(x) \\ \omega_{iS}^{S}(y) & -\omega_{iS}^{S}(x) & 0 \end{vmatrix} \mathrm{E} + \left(\omega_{iM}^{M} - \omega_{iS}^{S}\right) \tag{4.12}$$

where $\omega(\bullet)$ represents the corresponding vector element.

Equation (4.12) defines the differential equation that governs change in the Euler angles between master and slave systems. The last term in the right hand side of equation represents the deterministic input function. In case of Euler angles between two systems being not small, the above equation can also be used by first transferring master rotation rate to a nominal frame of reference.

In a Kalman filter structure, Equation (4.12) should be continuously solved by using the observed master and slave rotation rate. Therefore Equation (4.12) should be written as:

$$\dot{\mathrm{E}} = M\left(\omega_{iM}^{M}, \tilde{\omega}_{iS}^{S}\right)\mathrm{E} + \left(\omega_{iM}^{M} - \tilde{\omega}_{iS}^{S}\right) + \delta\omega_{iS}^{S} \tag{4.13}$$

where $\tilde{\omega}_{iS}^{S} = \omega_{iS}^{S} + \delta\omega_{iS}^{S}$ represents observed slave rotation rate, $\delta\omega_{iS}^{S}$ represents slave gyroscope's errors (it is assumed that master IMU is errorless) and M represent an operator which convert its arguments to the matrix shown in Equation (4.12).

Equation (4.13) defines the propagation model for the misalignment angle “E”. The aim is to implement a Kalman filter to estimate “E”. The measurement for such a filter can be calculated using acceleration and/or rotation rate outputs of master and slave IMUs. By employing small angle assumption in Equation (3.40), the difference between master and slave systems can be represented as follows:

$$\tilde{a}_{S}^{S} - a_{M_{Comp}}^{M} = -S\left(a_{M_{Comp}}^{M}\right)\mathrm{E} + \delta a_{S} + \delta a_{Comp} \tag{4.14}$$

where “δa_{S}” and “$\tilde{a}_{S}^{S} = a_{S}^{S} + \delta a_{S}^{S}$” denotes slave accelerometer error and observed slave acceleration. “$a_{M_{Comp}}^{M}$“ corresponds to lever arm compensated master acceleration output which is equal to (See Equation (3.40)):

$$a_{M_{Comp}}^{M} = a_{M}^{M} + \omega_{iM}^{M} \times \omega_{iM}^{M} \times r^{M} + \dot{\omega}_{iM}^{M} \times r^{M} \tag{4.15}$$

“δa_{Comp}” represents lever arm compensation error which is equal to

$$\delta a_{Comp} = C_{M}^{S}\left[\ddot{r} + 2\left(\omega_{IM}^{M} \times \dot{r}^{M}\right)\right] \tag{4.16}$$

Equation (4.14) represents a measurement equation that defines the relationship between E and acceleration differences between master and slave systems.

On the other hand, measurement for Kalman filter based on rotation rate differences can be formed using Equation (3.41) and (4.11) as follows:

$$\omega_{iS}^{S} - \omega_{iM}^{M} = -S\left(\omega_{iM}^{M}\right)\mathrm{E} + C_{M}^{S}\omega_{MS}^{M} \tag{4.17}$$

Using the observed variables, Equation (4.17) can be written as:

$$\tilde{\omega}_{iS}^{S} - \omega_{iM}^{M} = -S\left(\omega_{iM}^{M}\right)\mathrm{E} + \delta\omega_{iS}^{S} + \omega_{MS}^{S} \tag{4.18}$$

where “$\tilde{\omega}_{iS}^{S}, \delta\omega_{iS}^{S}$” represents observed slave rotation rate and corresponding gyroscope error, and “ω_{MS}^{S}” represents rotation rate of slave with respect to master. Equation (4.18) defines the relationship between master and slave rotation rate difference and misalignment angle.

In order to asses performance of acceleration/rotation rate matching method, a Kalman filter with the system equation given in Equation (4.13) and measurement equations shown in (4.14) and (4.18) is formed. Although the vibration related terms "ω_{MS}^{S}" and "$a_{M_{Comp}}^{M}$" in Equations (4.14) and (4.18) represents correlated noises they are not augmented to the Kalman filter system model and considered as a part of measurement noise. Furthermore, none of the gyroscope and accelerometer errors other than random walk (refer to Table 2-1) is excited in simulation environment. Therefore δa_S and $\delta\omega_{iS}^{S}$ consist of only white noises and considered as a part of measurement noise also. Finally, it is assumed that known lever arm vector "r" is almost errorless. Both update and propagation routines of the Kalman filter is run at 100Hz (for the definition of update and propagation routines refer to Appendix C).

Three different measurement structures are assessed with the designed Kalman filter. In the first structures, only accelerometer outputs are used as measurement (Equation (4.14)). In the second case, only rotation rates are used (Equation (4.18)) and in the third case both accelerations and rotation rates are used at the same time (In a sequential Kalman filter structure). The covariance estimates of the Kalman filter for each case are represented in Figure 4-4, Figure 4-5 and Figure 4-6. The average of standard deviations obtained in the last 10 seconds of each structure is summarized in Table 4-2.

Table 4-2 : Misalignment error standard deviation estimates comparison for accelerometer/rotation rate matching method

	Roll Error SD (mrad)	Pitch Error SD (mrad)	Yaw Error SD (mrad)
Acceleration	10	0.9	0.784
Rotation Rate	0.774	0.481	0.484
Acc. & Rot. Rate	0.62604	0.42644	0.42110

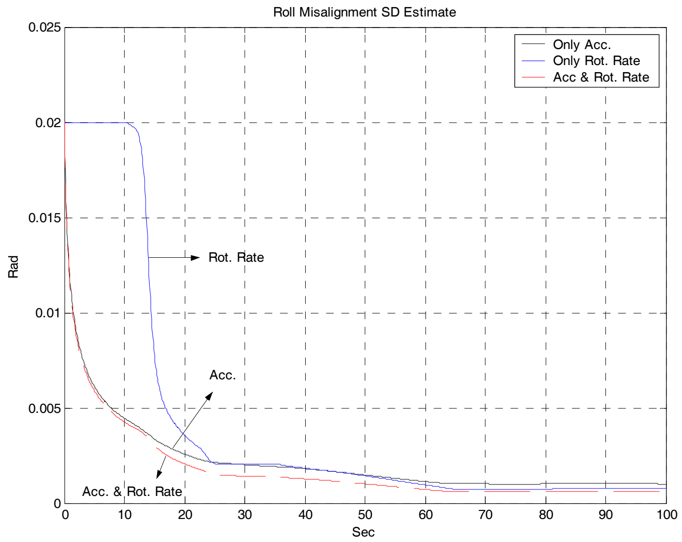


Figure 4-4 : Roll misalignment angle standard deviation estimate

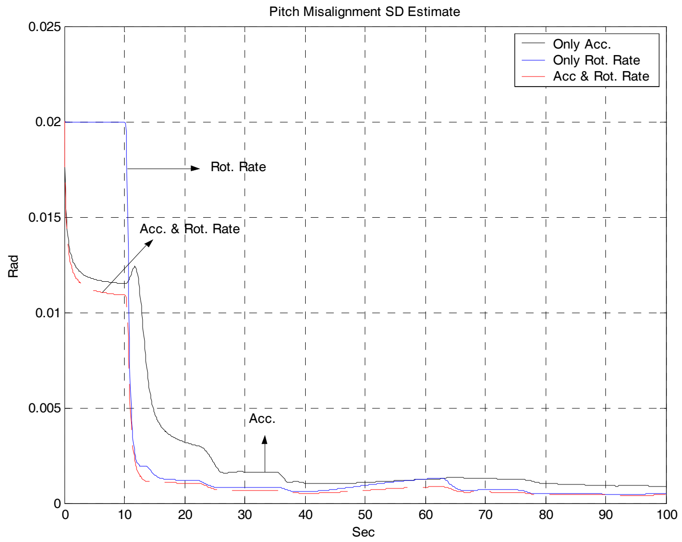


Figure 4-5 : Pitch misalignment angle standard deviation estimate

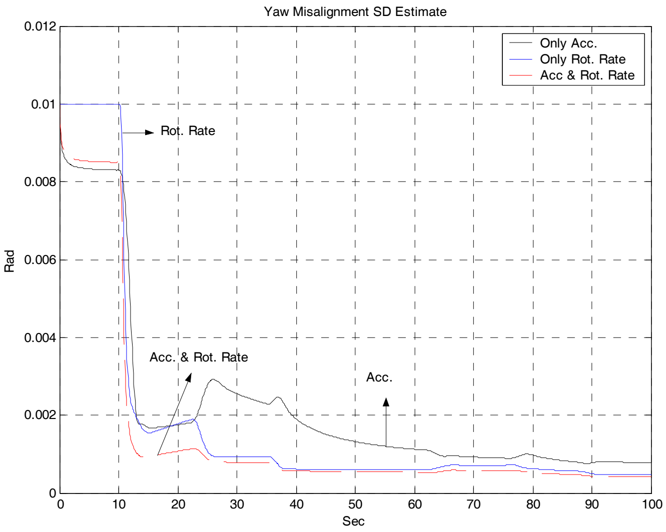


Figure 4-6 : Yaw misalignment angle standard deviation estimate

As seen from above figures, even in the optimal situation the accelerometer/rotation rate matching technique is not sufficient to estimate misalignment angles with a desired accuracy. Best results are obtained with the utilization of acceleration and rotation rates at the same time; however, its roll error estimation performance is also fall behind requirements which is estimated to be less than 0.4mrad.

In addition to the insufficient estimation performance, the most important drawback of this technique is the requirement for very high speed propagation of state equations. For the Kalman filter to work properly, Equation (4.13) has to be solved accurately in discrete time. In order to discretize Equation (4.13) with sufficient accuracy, master rotation rate (which is used as input in the equation) must be obtained at least 100Hz. However, for a standard mux-bus structure of most aircraft, it is not possible to acquire data from aircraft avionics at that frequency. Therefore, the applicability of this method is only limited to some special data bus setups.

On the other hand, the need to solve Equation (4.13) comes from the requirement of estimating dynamical change in misalignment angle. If it is assumed that the amount of dynamical change can be neglected, then the estimation problem can be reduced to estimating a constant parameter. In this case, the problem can be handled by just solving the measurement equations given in Equation (4.14) and (4.18) in a least square sense to obtain constant misalignment angle “E”. In the past, a number of study was conducted to solve Equation (4.14) and (4.18) by using deterministic least square algorithms (Boch (1989), Setterlend 1972)). However, the problem in this approach is that, until the first maneuver of the aircraft, observation equations constitute an underdetermined system which increases the complexity of recursive least square algorithms. Furthermore, if only acceleration measurements are used, regardless of whether aircraft performs a coordinated turn or not, observation equation for acceleration measurement stays always underdetermined which prevents the estimation of azimuth misalignment error. This situation can be shown as follows:

Suppose that a misalignment estimate is formed by using just 2 different observations. One of the observations is obtained at initial time (t_0), and the other estimate is obtained in a later time. In this case, a simple estimation can be formed by using pseudoinverse as follows :

$$\mathrm{E} = \left(\mathrm{Y}^T \mathrm{Y}\right)^{-1} \mathrm{Y}^T z \tag{4.19}$$

where for acceleration match Y and z correspond to

$$\begin{aligned} \mathrm{Y} &= \left[S\left(a_{M_{Comp}}^{M}(t_o)\right) : S\left(a_{M_{Comp}}^{M}(t_1)\right)\right]^T \\ z &= \left[\tilde{a}_S^S(t_0) - a_{M_{Comp}}^{M}(t_0) : \tilde{a}_S^S(t_1) - a_{M_{Comp}}^{M}(t_1)\right]^T \end{aligned} \tag{4.20}$$

and for both acceleration and rotation rate match Y and z correspond to

$$\begin{aligned} \mathrm{Y} &= \left[S\left(a_{M_{Comp}}^{M}(t_o)\right) : S\left(a_{M_{Comp}}^{M}(t_1)\right) : S\left(\omega_{IM}^{M}(t_o)\right) : S\left(\omega_{IM}^{M}(t_1)\right)\right]^T \\ z &= \left[\tilde{a}_S^S(t_0) - a_{M_{Comp}}^{M}(t_0) : \tilde{a}_S^S(t_1) - a_{M_{Comp}}^{M}(t_1) : \omega_{IS}^{S}(t_0) - \omega_{IM}^{M}(t_0) : \omega_{IS}^{S}(t_1) - \omega_{IM}^{M}(t_1)\right]^T \end{aligned} \tag{4.21}$$

In the above equations “t_0” corresponds to initial time. In the following figure the change in smallest eigenvalue of “$\left(\mathrm{Y}^T \mathrm{Y}\right)$” with respect to “$t_1$” is presented for only acceleration and for both rotation rate and acceleration measurements.

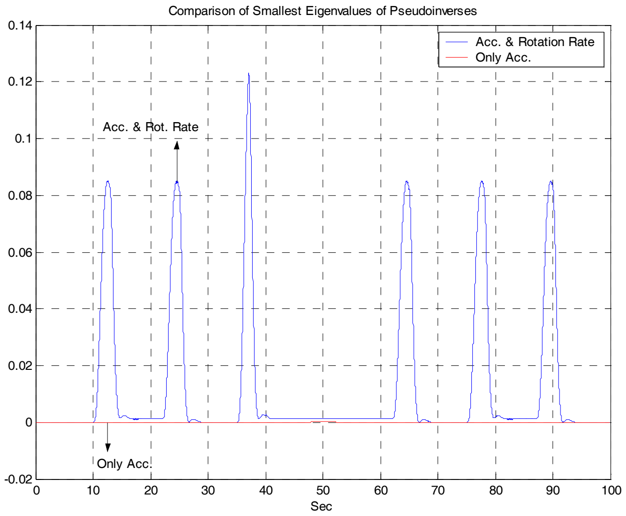


Figure 4-7 : Comparison of smallest eigenvalues of only acceleration measurement and both acceleration and rotation rate measurements.

As seen from the above figure, in acceleration and rotation rate matching technique, the occurrence of roll rate is very crucial in azimuth misalignment estimate. This is verified by the fact that, each local maximum (eigenvalue increase) in the Figure 4-7 corresponds to the roll rate occurrence during the maneuver of the aircraft (refer to Figure 4-3 for maneuver definition). In the only acceleration measurement case the maximum value of minimum eigenvalue turns out to be in the order of 3e-5, which in effect represents a singular matrix.

The reason for the insufficiency of acceleration measurements comes from the fact that, during the coordinated turn, the net acceleration occurring on the y axes of the master's body frame of reference is zero. Therefore, the maneuvers can not produce an acceleration both in x and y axes, which prevents the estimation of azimuth misalignment. On the other hand, as shown in Chapter 3, even during the coordinated turn maneuver in real time, aircraft undergoes a vertical motion, which creates "y" axes acceleration. Therefore, it is expected that performance of this technique can be much better in real environment.

One of the biggest advantages of this type of filter is its robustness to almost any kind of modeling error. 3 more states representing gyroscope drift repeatability errors were augmented to the simplest filter structure presented above, and that 6 state Kalman filter was run under the simulation environment where all error sources were excited (all kind of IMU errors presented in Table 2-1 and lever arm errors are applied). The only modification made on the filter structure was to multiply measurement covariance matrix of Kalman filter with 4 to compensate the suboptimal effects in filter structure. The results of this highly suboptimal filter structure reveal that, Kalman filter was still able to continue to estimate misalignment errors without diverging. Monte Carlo analysis showed that the total degradation in estimation of misalignment angle for the suboptimal filter was in the order of 0.4mrad. In the following figure, the real error variance of misalignment angles calculated by Monte Carlo simulations based on suboptimal Kalman filter structure which uses both acceleration and rotation rate as measurement is presented.

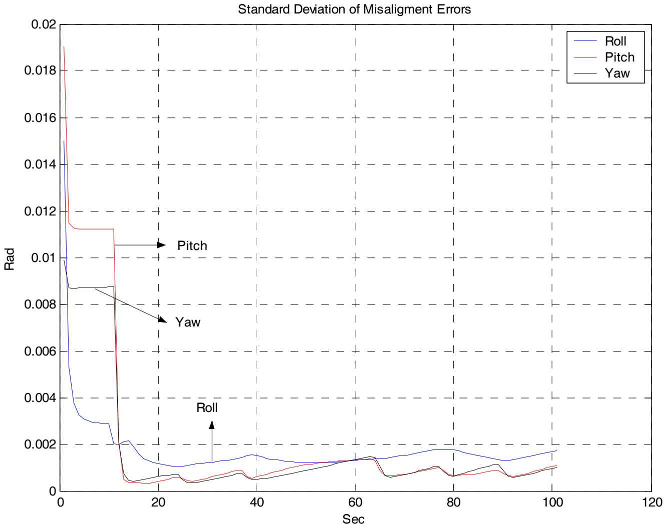


Figure 4-8 : Standard deviations estimates of misalignment errors obtained by Monte Carlo analysis based on suboptimal Kalman filter structure

In the above filter configuration, together with misalignment states gyroscope drift errors were also augmented to the Kalman Filter system model. This is because, Monte Carlo analysis showed that if the drift errors are not augmented to the system model, then the suboptimal Kalman filter tends to diverge for long operation durations. Therefore, although the inclusion of drift models does not increase overall estimation performance, they should be used together with misalignment states. On the other hand, it should be noted that, for tactical grade IMUs, which has approximately 1°/sec drift error, such inclusion of drift errors is not necessary as that amount of drift error does not corrupt the measurements as much as in the case of gyroscopes with 10°/sec drift error.

In order to see that, whether this technique is capable of estimating any IMU calibration parameter, all IMU and lever arm models presented in Table 4-1 were augmented to the Kalman filter system model. When the Kalman filter was run under these conditions, it was observed that this transfer alignment method is not capable of producing acceptable IMU calibration parameter estimation performance (the estimation performance was twice worse than the estimation performance of velocity matching method). In Boch (1989), it is proposed to slow down filter update rate in order to support estimation of calibration parameters. However, even this technique did not contribute to the estimation performance of Kalman filter at all. In Harris (1977), it is suggested to augment system models with vibration related terms such as "ω_{MS}^{S}" in Equation (4.18). However, as also explained by Harris (1977), there is no guarantee that more complex filters depending on less accurate models such as vibration effects can increase filter performance. Even if such a performance increase is obtained with these techniques, it is possible that Kalman filter can diverge in real time operation due to the vibration modeling errors.

Another disadvantage of this method is that, when all IMU models are utilized in Kalman filter system model, the Kalman filter become highly sensitive to any error in misalignment model presented in Equation (4.13). As explained above, in order to track dynamical misalignment changes during flight, very high frequency Kalman filter update rate should be utilized. With such high frequency update rate, any erroneous misalignment model quickly disturbs IMU calibration parameter estimates. On the other hand, when update frequency is reduced, filter cannot estimate misalignment changes and therefore cannot estimate IMU calibration parameters either. Therefore, in order to estimate any IMU calibration parameter with this method, a very accurate misalignment model is required.

As a result, despite the simplicity and robustness of the filter structure, the estimation performance of this filter turns out to be insufficient. Especially, its lack of ability to estimate any calibration error parameter makes the filter very useless for advanced transfer alignment needs.

4.4 VELOCITY MATCHING METHOD

In this transfer alignment technique, master and slave velocities are compared via a Kalman filter structure. However, before the comparison is made, the velocity obtained from aircraft should be compensated for lever arm effect. As the lever arm compensation errors act like IMU errors in the measurement, the correctness of lever arm compensation process is highly crucial in the estimation of calibration parameters (Boch (1989)).

A typical structure of a transfer alignment filter which uses velocity match technique is represented in the following figure.

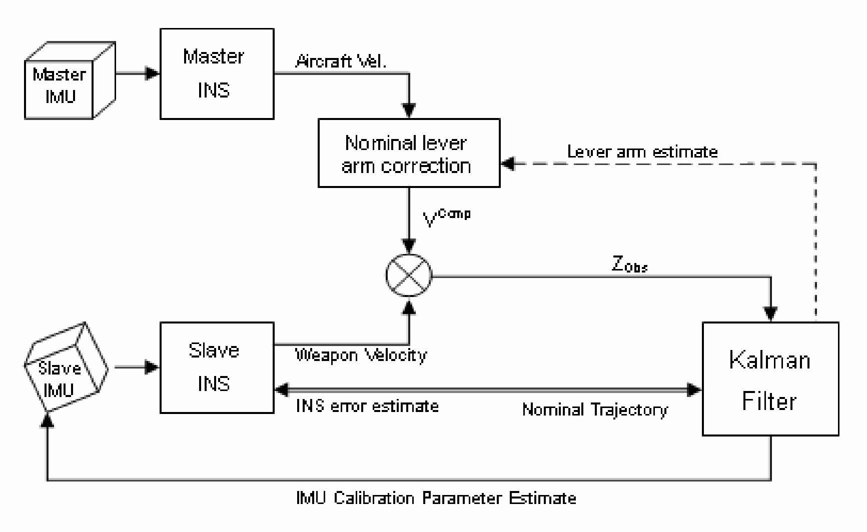


Figure 4-9 : Structure of transfer alignment filter which uses velocity match method

As seen from the above figure, measurements for the Kalman filter are formed using slave and compensated aircraft velocities. The lever arm compensation on master velocity output is calculated using Equation (3.30) as follows:

$$V_M^{Comp} = V_M^n + C_M^n \left[(\omega_{iM}^M - \omega_{ie}^M) \times r^M \right] \tag{4.22}$$

As seen from Equation (4.22), it is assumed that, rotation rate and lever arm vector are also supplied by master INS along with the velocity information. If those variables cannot be obtained from master INS, than lever arm compensation process can be accomplished using variables obtained from slave

INS by utilizing Equation (3.31). However, as indicated in Stovall (1993), use of slave supplied values can decrease the performance of the filter.

By subtracting the slave velocity from compensated master velocity, an observation for the Kalman filter can be calculated as follows:

$$z_{Obs} = \tilde{V}_M + C_M^n \left[\left(\tilde{\omega}_{iM}^M - \omega_{ie}^M\right) \times \tilde{r}^M\right] - \tilde{V}_S \tag{4.23}$$

In Equation (4.23), the variables with "~" on top represent erroneous values (the values that are actually supplied by the master and slave systems). The relation between true and erroneous value can be represented as $\tilde{x} = x + \delta x$ where δx denotes error on the corresponding variable.

In order to process this observation in a Kalman filter, a measurement model that relates the system states with the observation is required. Using Equation (3.30), a measurement equation for the observation given in (4.23) can be written in terms of error values as follows:

$$z_{Meas} = -\delta V_S + C_M^N S\left(\tilde{\omega}_{EM}^M\right)\delta r^M + C_M^n \frac{dr^M}{dt} + \delta V_M^n - C_M^n S\left(\tilde{r}^M\right)\delta\omega_{iM}^M \tag{4.24}$$

In Equation (4.24) δV_S and δr^M represents the error on slave velocity and lever arm vecotrs. If it is assumed that only these two variables are modeled in system model of Kalman filter then the remaining terms constitute the measurement noise. Hence measurement noise can be represented as follows:

$$\upsilon = C_M^n \dot{r}^M + \delta V_M^n - C_M^n S\left(\tilde{r}^M\right)\delta\omega_{iM}^M \tag{4.25}$$

The first term in RHS of Equation (4.25) represents the lever arm compensation error due to the elastic motion of aircraft structure (vibration term). The last two terms in RHS represents the error in master INS outputs. These master INS errors were accepted to be in the form of white noise with standard deviations given in Table 2-3 and errors in master supplied attitude information was ignored.

The estimation of slave navigation errors, slave IMU parameters and lever arm errors was performed using an extended Kalman filter with a closed loop feedback structure (refer to Appendix C). The model given in Table 4-1 was implemented as the Kalman filter system model. Therefore the system model was composed of the following 27 states:

i. 3 states for slave velocity error each represents north, east and down velocity errors. The error propagation models for these states are shown in Equation (2.16).

ii. 3 states for slave attitude error. The error propagation models for these states are shown in Equation (2.10)

iii. 3 states for slave's accelerometer bias repeatability

iv. 3 states for slave's accelerometer scale factor repeatability

v. 3 states for slave's gyroscope drift repeatability

vi. 3 states for slave's gyroscope scale factor repeatability

vii. 3 states for slave's accelerometer bias instability

viii. 3 states for slave's gyroscope drift instability

ix. 3 states for static lever arm error

The propagation models and stochastic properties for IMU error states were summarized in Table 4-1 and Table 2-1. As seen from Table 4-1, lever arm errors were modeled as random constants in the system model. On the other hand, due to the elastic nature of aircraft structure, the lever arm between master and slave system continuously changes. Therefore, it would be more realistic to model the lever arm vector as the sum of two vectors one of which represents the static and the other represents the dynamic component of lever arm. This can be represented as follows:

$$\begin{aligned} r &= r_{Stat} + r_{Dyn} \\ \dot{r}_{Stat} &= 0 \end{aligned} \tag{4.26}$$

However, in the Kalman filter, only the static lever arm component is modeled as random constant and the effect of dynamic lever arm is ignored in the system model. This is because for two reasons. First of all, it is expected that steady state standard deviation of dynamic lever arm component does not exceed 5 cm. Therefore, the effect of dynamical lever arm errors on lever arm compensation process would not be too much. Secondly, as shown in Chapter 3 and Appendix B, to model the dynamical lever arm component, it is required to augment at least 9 states to the Kalman filter system model. However, the accuracy of these models is not sufficient to represent vibration dependent displacement during real flight conditions. Therefore considering the increased computational complexity as well as unreliable modeling problems, it is preferred to consider the effect of dynamical lever arm error as a part of measurement noise.

Hence, replacing Equation (4.26) in Equation (4.24) and (4.25), the following revised measurement equations are obtained:

$$z_{Meas} = -\delta V_S + C_M^N S\left(\tilde{\omega}_{EM}^M\right)\delta r_{Stat}^M + \upsilon \tag{4.26}$$

$$\upsilon = C_M^n\left(\dot{r}_{Dyn}^M - \tilde{\omega}_{eM}^M \times r_{Dyn}^M\right) + \delta V_M^n - C_M^n S\left(\tilde{r}_{Stat}^M\right)\delta\omega_{iM}^M \tag{4.27}$$

As can be seen in Equation (4.27), with the utilization of new measurement equation, Kalman filter estimates only the error on static lever arm component. Furthermore, because of the effects of "$\dot{r}_{Dyn}^M$" and "r_{Dyn}^M", measurement noise contains correlated noise components. Although these errors contradicts with the Kalman filter white measurement noise constraint, because of the reasons stated above, those correlated noise terms was handled as white noise processes in the Kalman filter's measurement model. The variances of these components were taken to be same as the steady state variance of vibration models presented in Appendix B.

As seen from Table 4-1, slave position errors were not modeled in Kalman filter. This is because; lever arm compensation for position can be performed with an accuracy of at worst 30cm. Therefore, instead of adding position states, it was preferred to directly update position states of INS by the position information supplied by master INS so that processing load of filter hardware could be reduced.

In the following figures attitude and bias errors standard deviation estimates of the Kalman filter which uses Velocity match technique with an update rate of 1Hz is presented.

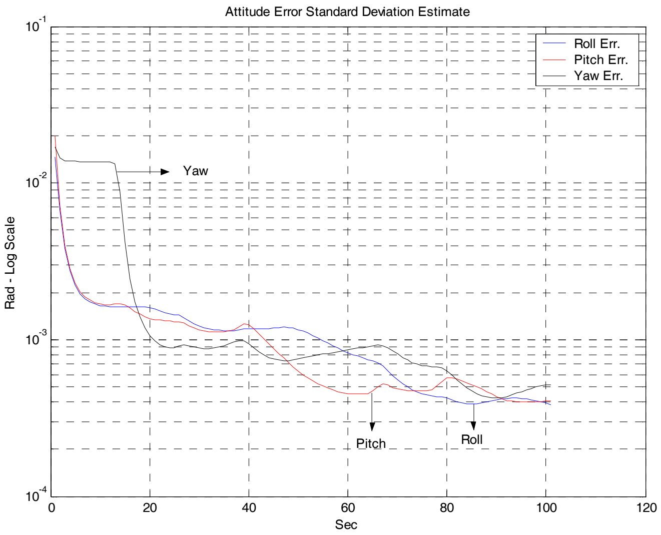


Figure 4-10 : Attitude error standard deviation estimate of Kalman filter

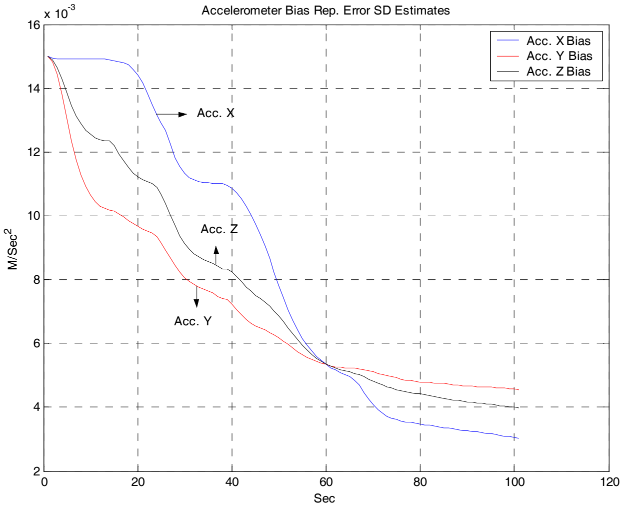


Figure 4-11 : Accelerometer bias repeatability error standard deviation estimates of Kalman filter

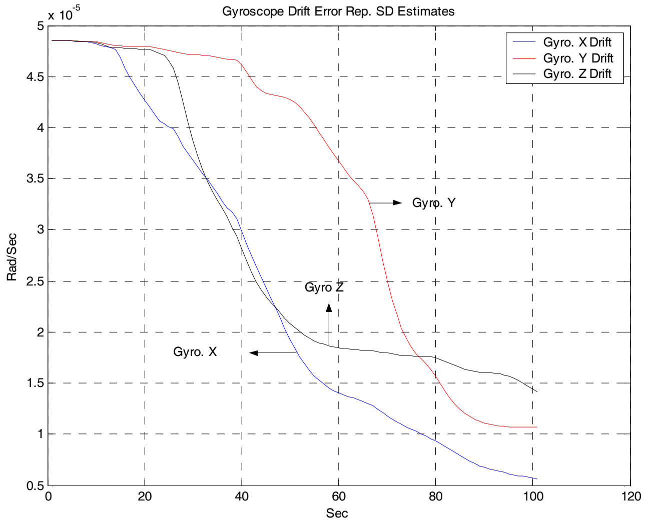


Figure 4-12 : Gyroscope drift repeatability error standard deviation estimates of Kalman filter

As seen from the above figures, velocity match technique is highly successful in estimating both attitude and calibration error parameters. Filter can reduce the attitude error in the order of 0.45 mrad level as well as it reduces the bias error to 0.4 mg level. Also, above figures roughly reveals the relationship between aircraft maneuver and estimation performance of the filters. For instance, as shown in Figure 4-10, error in azimuth angle can not be observed until the aircraft banks to turn.

A complete performance analysis of this filter structure will be presented at the next chapter. Therefore, no more result about this technique is presented in this section.

4.5 INTEGRATED VELOCITY MATHCING METHOD

As shown in Equation (4.27), vibration induced velocity directly appear as measurement noise in velocity match technique. Therefore the level of vibration has a limiting effect on the estimation performance of the filter. Moreover, as the bandwidth of the vibration is much higher than the filter update rate, basically it is not possible to reduce the effect of vibration on filter efficiency by augmenting the vibration models to the Kalman filter’s system model. Hence, if the measurements can be separated from the effect of vibration induced velocity, estimation performance of the Kalman filter will increase.

Integrated velocity match technique is a direct consequence of above reasoning. In this technique, the velocity measurements are integrated in order to reduce the effect of vibration induced velocity, before the measurements are processed by Kalman filter. In this method, integration operation acts as a low pass filter such that it can eliminate the high frequency vibration motion from measurements.

On the other hand, the effectiveness of this method should not be overestimated. As a matter of fact, Kalman filter is an optimal estimator. Therefore, it is not possible to increase the estimation performance by performing a linear operation on measurements. This statement is true only if system and measurement noises satisfy white noise property, however, as explained before, vibration induced motion is a correlated process. In fact, this property of measurement error makes it possible to increase estimation performance by integrating measurements.

To derive an analytical reasoning for above the statements can be extremely tedious. Some derivations depending on continuous time Kalman filter formulation with suboptimal gain may be possible, but analytically solving continuous time Ricatti equation with a suboptimal gain seems not possible. Therefore, it was preferred to present effectiveness of integration concept with the following simple example.

Suppose it is desired to estimate the velocity of a system which has only one degree of freedom by observing some noisy velocity measurement or integral of that measurement. That problem can be represented as follows:

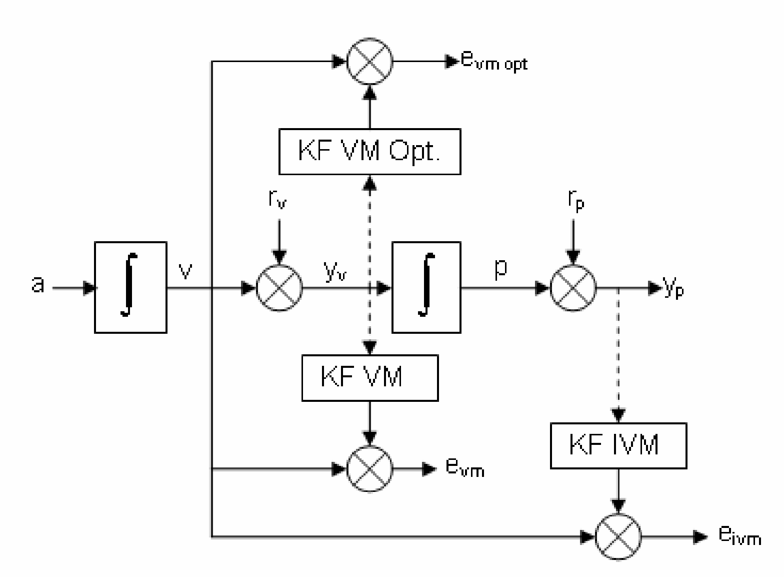


Figure 4-13 : System which is used to verify integration approach

In the above figure, “a” represents the acceleration in the form of white noise process with PSD level of $0.1(m/s^2)^2$, “r_v,y_v” represents velocity measurement noise, and velocity observation. The integral of observation is called as “p”. Note that, “p” represents the erroneous position of the system as well (the position which is calculated using velocity observation). Finally, “r_p“ denotes the error arise from the discrete time integration of velocity observation and “y_p” denotes velocity integral observations.

For the above system 3 different Kalman filter structures are formed. “KF VM” uses velocity observation, whereas, “KF IVM” uses integral of velocity observation to estimate actual velocity of the system. “KF VM Opt” also uses velocity observations, however it always satisfy Kalman filter system and measurement model assumptions (optimal Kalman filter structure for the system). The estimation errors for the Kalman filters are represented as “e_{vm}”, “$e_{vm\ opt}$” and “e_{ivm}”.

First of all, it should be noted that, if measurement noises are white and Kalman filter structures are implemented in continuous time, then the steady state values of velocity estimation error variances will be as follows:

$$\sigma^2_{e_{vm}} = \sqrt{N_a R_v} \tag{4.28}$$

$$\sigma^2_{e_{ivm}} = \sqrt{N_a\left(2\sqrt{N_a R_p} + R_v\right)} \tag{4.29}$$

where N_a represents PSD level of input acceleration, and R_v and R_p denotes PSD level of white measurement errors. As seen from Equation (4.29) if R_p is zero (if no error occurs in the integration of velocity measurements) than the steady state error variances of both Kalman filters become equal as expected.

Above system was run in several different configurations to compare the estimation performance of "KF VM" and "KF IVM". In the first configuration, r_p was set to zero, and a white noise process with a PSD level of $0.01(\text{m/sec}^2)^2$ was applied as r_v. During the discretization of the system, it was assumed that measurement errors were averaged such that:

$$\begin{aligned} R_v^{Disc} &= R_v \Delta T \\ R_p^{Disc} &= R_p \Delta T \end{aligned} \tag{4.30}$$

where ΔT corresponds to observation interval. When both Kalman filters were run at 1Hz update rate, the following estimation error standard deviation estimates were obtained.

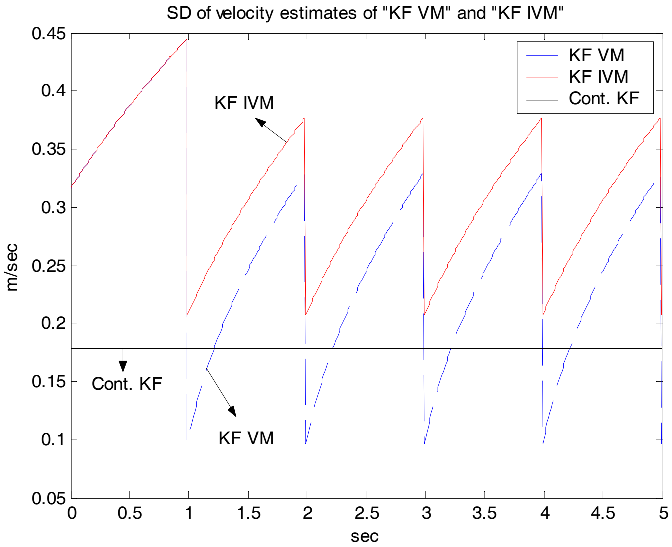


Figure 4-14 : Comparison of SD estimates of Kalman filters which uses velocity and integrated velocity measurements

As seen from the above figure, with this configuration VM filter performs better than IVM filter. However as the update frequency increases, both standard deviation estimates converge to the strait black line which represents the theoretical standard deviation estimate of a continuous time Kalman filter estimate. This example also shows the implicit suboptimality of the continuous time Kalman filters which arise from the fact that white noise processes are not realizable.

In the second configuration, r_p was again set to zero, however, a second order Markov process which has the following PSD was applied as the velocity observation noise:

$$R_v(s) = \frac{sN_v}{s^2 + 2\alpha\beta s + \beta^2} \tag{4.31}$$

Steady state variances of this process can be calculated as follows:

$$\begin{aligned} \mathrm{E}_{SS}\left[r_v r_v^T\right] &= \sigma^2_{v_{vib}} = \frac{N_v}{4\alpha\beta} \\ \mathrm{E}_{SS}\left[\left(\int r_v\right)\left(\int r_v\right)^T\right] &= \sigma^2_{d_{vib}} = \frac{N_v}{4\alpha\beta^3} \end{aligned} \tag{4.32}$$

Parameters of this process are adjusted to be as follows:

$$\begin{aligned} N_v &= 1.3 \\ \alpha &= 0.65 \\ \beta &= 50 \end{aligned} \tag{4.33}$$

which corresponds to following steady state variances:

$$\begin{aligned} \sigma^2_{v_{vib}} &= 0.01(m/s)^2 \\ \sigma^2_{d_{vib}} &= 4.0000\mathrm{e}\text{-}006\,\mathrm{m}^2 \end{aligned}$$

In this configuration, suboptimal Kalman filters were implemented such that they handled correlated measurement noises as if they were white noise sequences with standard deviation equal to $\sigma^2_{v_{vib}}$ for “KF VM” and $\sigma^2_{d_{vib}}$ for “KF IVM”. As a matter of fact, this is the point where the importance of correlatedness of measurement errors appears. As given in Equation (4.33), the steady state variance of integral of correlated noise is $1/\beta^2$ of noise itself. Therefore, in this system, each integrated measurement has 1/2500 times less uncertainty than the measurement itself, which under some circumstances provides an increase in the estimation performance of the filter.

In Figure 4-15, the comparison of standard deviation estimates of Kalman filter and real error standard deviations obtained using Monte Carlo simulations is presented for both “KF VM” and “KF IVM” which were run at 1Hz update rate.

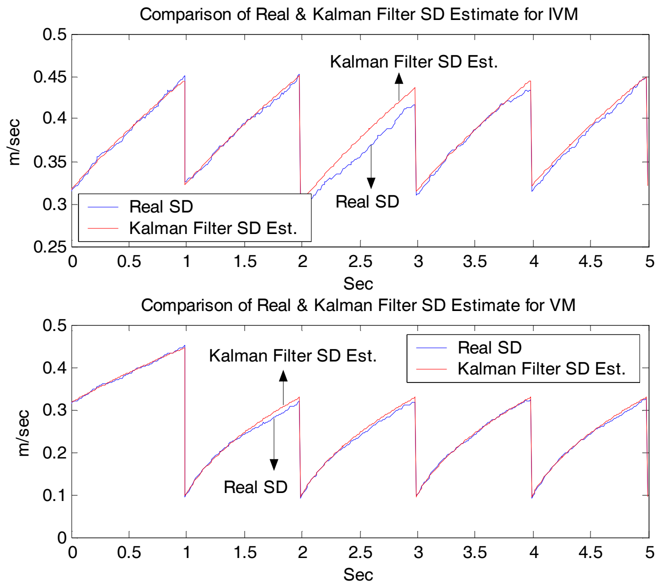


Figure 4-15 : Comparison of real and Kalman filter error estimate standard deviations

As seen from the above figure, although the measurement noises are correlated, Kalman filter still performs as if it were optimal. Also note that, “KF VM” filter still produce better estimates than “KF IVM” filter.

In the third configuration, excitation power of velocity measurement noise (N_v) was increased 600 times, and the rest was kept same as the previous configuration. Standard deviations of estimation errors obtained using this configuration is shown Figure 4-16.

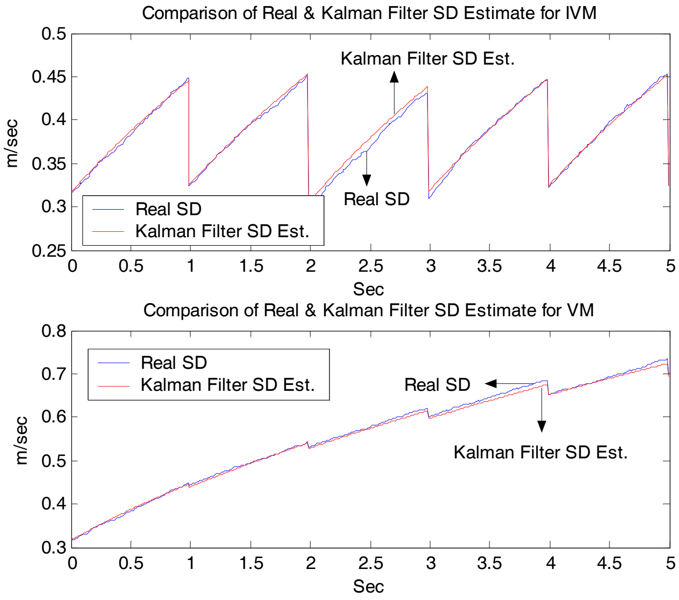


Figure 4-16 : Comparison of real and Kalman filter error estimate standard deviations

In the second subplot in Figure 4-16 (SD estimates for VM), the standard deviation estimate of Kalman filter was obtained using the optimal Kalman Filter structure; however, the real error standard deviation was obtained by Monte Carlo simulations using suboptimal Kalman filter outputs (“e_{vm}”). Therefore, this figure suggests, under the specified conditions, “KF VM” filter can still works as if it were optimal. The reason for this is the selected 1Hz update rate of Kalman filters. The coefficients shown in Equation (4.33) correspond to a bandpass process whose center frequency is around 10Hz. Therefore 1Hz sample of this process becomes relatively uncorrelated and representing this sample in Kalman filter as a white noise sequence turns out to be a very good approximation.

In the first subplot of Figure 4-16 (SD estimates for IVM), the standard deviation estimate of “KF IVM” filter and real standard deviation of error calculated with Monte Carlo simulations using “e_{ivm}” were presented. Comparing this figure with the second one, it can be seen that, “KF IVM” filter performs much better than “KF VM” filter.

At a first glance, the result presented in Figure 4-16 seems to be violating the optimality of Kalman Filter. However, it should be noted that, in order to keep integration error at negligible levels, the integration of measurements was performed at 500Hz and thus R_p is set to zero. On the other hand, both “KF VM” and “KF IVM” were run at 1Hz. Therefore, as a matter of fact, “KF IVM” filter incorporates 500 observations in one update period whereas “KF VM” filter uses just single observation. If the frequency of measurement integration process is decreased, than an additional error component on integrated measurements starts to appear which has to be handled in Kalman filter structure properly. Monte Carlo simulations showed that for this specified system, integration frequency can be reduced to 100Hz safely by just using an artificial measurement error (R_p) in the Kalman filter formulations. However below 100Hz, in order to prevent divergence of estimates, this integration error should also be modeled by augmenting additional states in the Kalman Filter system model.

The power of integrated measurement approach comes from the fact that, even if suboptimal Kalman filter structure is used, it can present near optimal performance. To show this, same system used in above analysis (Figure 4-13) was used again. However this time, optimal Kalman filter (“KF VM opt”) was run at 100 Hz. Also integration of measurements was performed at 100Hz and “KF IVM” was run using these integrated measurements at 20Hz. A small artificial (R_p) was added to KF IVM formulations in order to compensate integration errors. The comparison of standard deviation of errors is represented in the following figure.

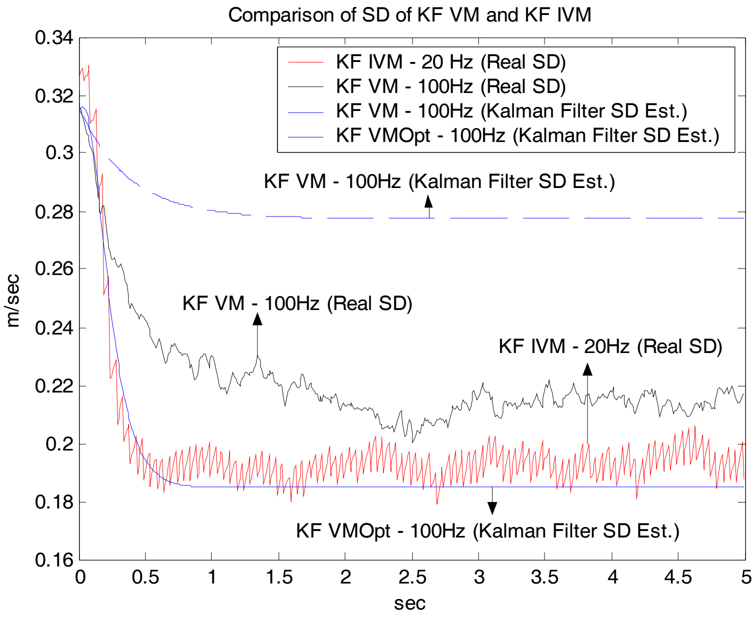


Figure 4-17 : Comparison of SD obtained using "KF VM", "KF IVM" and "KF VM Opt"

As seen from the above figure, suboptimal Kalman filter which process integrated measurements at 20Hz can perform almost same with the optimal Kalman filter which uses velocity observations at 100Hz. In the figure, results of "KF VM" filter are also presented. As can be seen from this figure, at 100Hz, suboptimal filter's standard deviation estimate and real standard deviation obtained using Monte Carlo simulations substantially differed which was due to the unmodelled correlated nature of velocity measurement noise.

On the other hand, it should be noted that, integration errors, which was modeled as white measurement noise, may cause "KF IVM" to diverge for long operation durations. Therefore, for Kalman filters using integrated measurements, integration errors should be handled very carefully.

In a typical transfer alignment algorithm, every velocity measurement is contaminated with some form of correlated noise. As can be seen from Equation (4.27), especially vibration induced velocity and dynamic displacement error add significant amount of correlated error which causes problems for high Kalman Filter update rates. Moreover, as unmoldelled lever arm errors behave like IMU errors, during the maneuver of aircraft it distorts the calibration parameter estimation very much.

In order to avoid divergence problems, one can reduce the filter update rate. However, this causes the estimation performance of filter to decrease. On the other hand, as shown above, measurement integration method can be very effective under correlated measurement noise. Therefore, in the literature, integrated velocity matching method has also been frequently employed in the design of transfer alignment algorithms.

Basically two different integrated velocity matching method can be implemented. In the first method, the observation used in velocity matching technique, which is given in Equation (4.22), is directly integrated. In this approach, three new states representing integrated measurements are augmented to the Kalman filter system model. This can be shown as follows:

Let the original Kalman filter model be as follows:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Hx + v \end{aligned} \tag{4.34}$$

Then in the integrated measurement approach, the system model turns out to be as follows:

$$\begin{aligned} \begin{vmatrix} \dot{x} \\ \dot{\hat{y}} \end{vmatrix} &= \begin{vmatrix} A & 0 \\ H & 0 \end{vmatrix} \begin{vmatrix} x \\ \hat{y} \end{vmatrix} + \begin{vmatrix} B & 0 \\ 0 & \mathrm{I} \end{vmatrix} \begin{vmatrix} u \\ v \end{vmatrix} \\ y &= \begin{vmatrix} 0 & 1 \end{vmatrix} \begin{vmatrix} x \\ \hat{y} \end{vmatrix} + \hat{v} \end{aligned} \tag{4.35}$$

In Equation (4.35), theoretically there should not be any measurement error. However, as described before, to model the effect of integration error, an artificial measurement error is also utilized in the measurement model.

The structure of this kind of transfer alignment filter can be represented as in the following figure.

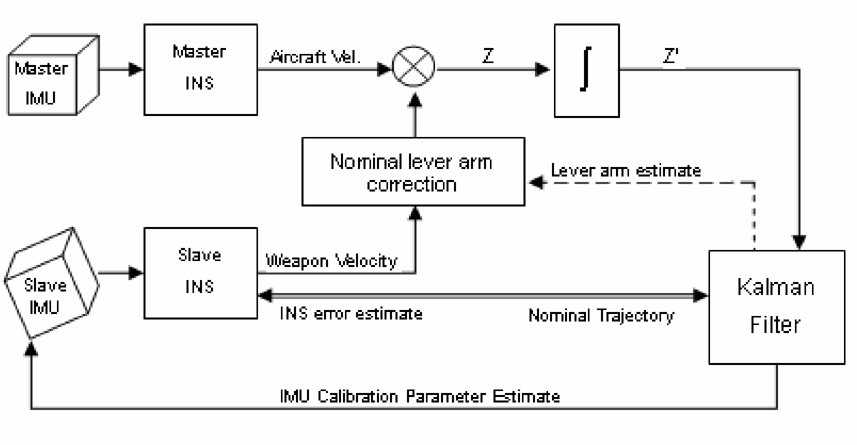


Figure 4-18 : The structure of transfer alignment filter which uses integrated velocity matching technique

As seen from the above figure, the compensated velocity differences are integrated in this method. However, the compensation process involves dynamically changing terms such as rotation rate of master. Therefore, in order to keep integration error at acceptable levels, it is required to perform integration operation very frequently.

In order to reduce the need for frequent integration, the method is modified such that, integral of velocities are computed before lever arm compensation is made (Savage, (2000)). The structure of this modified integrated velocity matching method is shown in Figure 4-19.

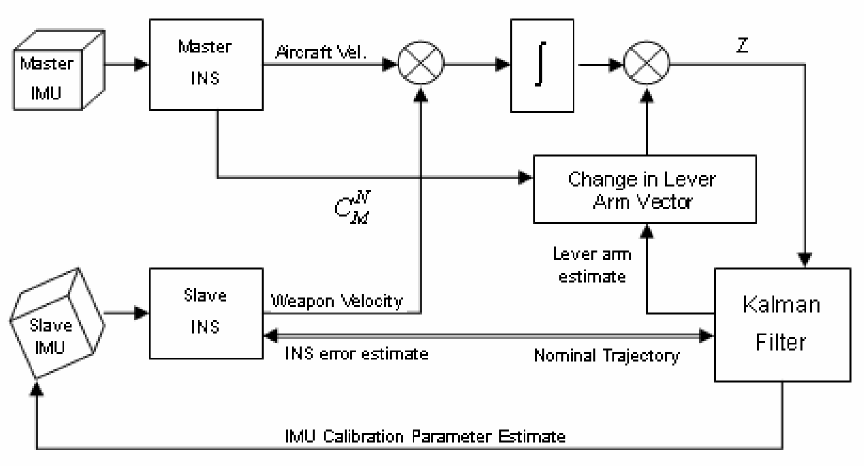


Figure 4-19 : The structure of Kalman filter which uses integrated velocity measurements

In this method, the direct difference between master and slave velocities is integrated (without any lever arm compensation). As shown in Equation (3.33) this difference corresponds to the following:

$$V_S^n - V_M^n = S\left(\omega_{en}^n\right)r^n + \dot{r}^n \tag{4.36}$$

Therefore the integral of velocity difference can be represented as:

$$\int_0^{\Delta T}\left(V_S^n - V_M^n\right)dt = r^n(\Delta T) - r^n(0) + \int_0^{\Delta T} S\left(\omega_{en}^n\right)r^n dt \tag{4.37}$$

Using Equation (4.37) an observation can be computed as follows:

$$z_{Obs}^{Integ} = \int_0^{\Delta T}\left(\tilde{V}_s^n - \tilde{V}_M^n\right)dt - \left[\tilde{C}_M^n(\Delta T) - \tilde{C}_M^n(0)\right]\tilde{r}_{Stat}^M + \int_0^{\Delta T} S\left(\omega_{en}^n\right)\tilde{r}^n dt \tag{4.38}$$

Using the same state structure shown in Table 4-1, and lever arm model given in Equation (4.25), a measurement model for the above observation can be constructed as follows:

$$z_{Meas}^{Integ} = \int_0^{\Delta T}\left(\delta V_s^N - \delta V_M^N\right)dt + \left[\tilde{C}_M^N(\Delta T) - \tilde{C}_M^N(0)\right]\delta r_{Stat}^M + \upsilon + \upsilon_{Integ} \tag{4.39}$$

$$\upsilon = \left[\hat{C}_M^N(\Delta T) - \hat{C}_M^N(0)\right]r_{Dyn}^M - \left[\delta\hat{C}_M^N(\Delta T) - \delta\hat{C}_M^N(0)\right]\hat{r}^M + \int_0^{\Delta T}\Omega_{EN}^N \delta r^N \tag{4.40}$$

Equation (4.39) represents the measurement model and Equation (4.40) represents the measurement error (measurement noise). The last term in

Equation (4.39) (υ_{Integ}) corresponds to fictitious integration error. As shown in Equation (4.38), observations are calculated by integrating velocities. As this integration is performed in discrete time, an unavoidable integration error occurs in real time implementation. In order to represent this error, a fictitious noise component is added to the measurement model. In the following discussions, it is assumed that the amount of this additional error is negligible and thus completely ignored. However, in Chapter 5, the effect of this error is analyzed in detail.

In deriving above measurement models, it is assumed that, the product of two error terms are negligible, and r_{Dyn} does not change much in single observation duration.

As seen from Equation (4.39), the measurement equation in this method contains integral of velocity errors differences (the first term in the right hand side of Equation (4.39)). Therefore, it is required to augment this integral as a new state in Kalman filter's system model. However, it should be noted that, this new state does not represent the integral of velocity error. Instead it represents the error in integral of difference between slave and master system velocities. Thus, denoting this new state as "i", the propagation model of "i" can be represented as follows:

$$\dot{i} = \delta V_s^N - \delta V_M^N \qquad (4.41)$$

As stated before, error on master supplied information is assumed to be in the form of white noise. Therefore, "δV_M^N", which represents the error on master velocity, is handled as a white process noise for the "i" state. As the integration of measurements starts at the same time filter starts to operate, neglecting data synchronization errors, the initial uncertainty of "i" state is taken to be zero (Refer to Savage (2000) for a detailed explanation about the calculation of initial uncertainty of "i" state).

With the augmentation of Equation (4.41) to the states defined in Table 4-1, the number of states modeled in the Kalman filter system model for integrated velocity match technique increases to 30.

The last term in Equation (4.40) represents the cross product of lever arm vector error and the rotation rate of Navigation frame of reference with respect to Earth. As this rotation rate is very small, the effect of that error term can simply be ignored.

Comparing Equation (4.40) and Equation (4.27), it can be seen that, in Equation (4.40) instead of vibration induced velocity term there is only vibration induced displacement term. As a matter of fact, this is the point where benefits of integrated velocity matching technique come from.

In the following figures attitude bias and drift errors standard deviation estimates of the Kalman filter which uses integrated velocity match technique with an update rate of 1Hz is presented. The integral of measurements are calculated at 20Hz.

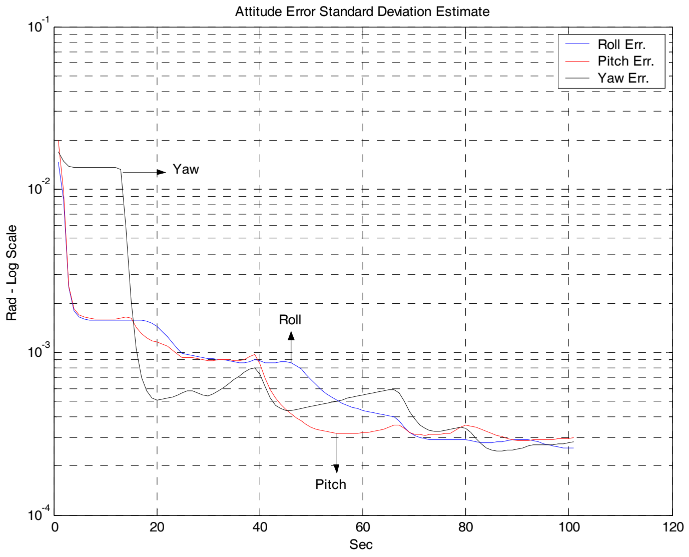


Figure 4-20 : Attitude error SD estimates of the Kalman filter which uses integrated velocity match technique

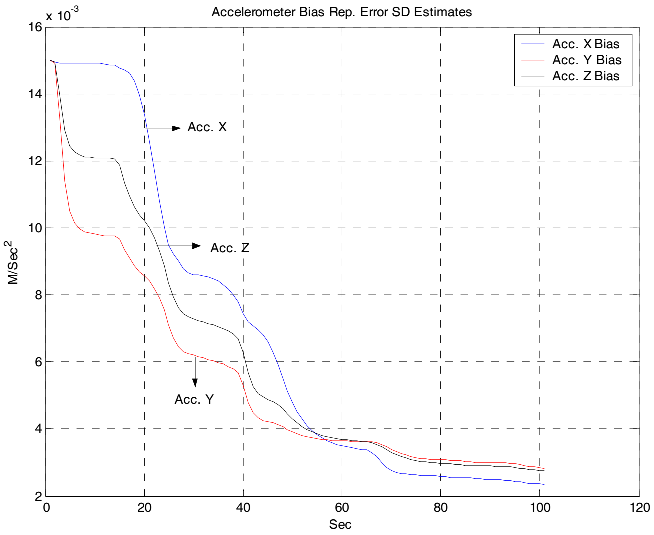


Figure 4-21 : Bias repeatability error SD estimation of Kalman filter which uses integrated velocity measurements.

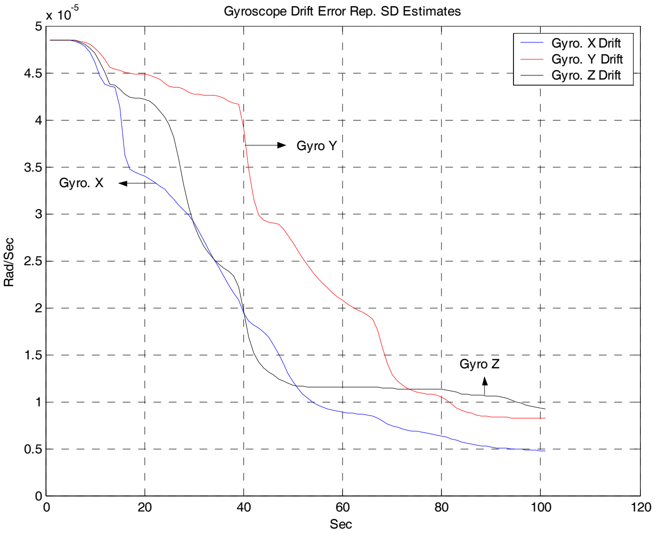


Figure 4-22 : Gyroscope drift error repeatability SD estimations of Kalman filter which uses integrated velocity measurements

Comparing Figure 4-10 and Figure 4-11 with Figure 4-20 and Figure 4-21, it can be seen that estimation performance of integrated velocity match technique is better than velocity match technique under the specified conditions. However, it should be noted that, in the above calculations it was implicitly assumed that 20 Hz integration frequency is sufficient. The more detailed treatment of the results and comparison of this technique with velocity match method under different conditions will be presented at the next chapter.

4.6 ATTITUDE MATCHING METHOD

In a typical transfer alignment application, it is possible to obtain attitude of the master system along with the velocity information. Therefore, it is usually desirable to use attitude information as another source of measurement in Kalman filter structure.

In order to asses the effect of attitude measurements in estimation performance, first, a Kalman filter which uses only attitude information as measurement is constructed. For this filter, observations are obtained using the following equation:

$$z_{obs} = \tilde{C}_S^n \tilde{C}_M^S C_n^M \tag{4.42}$$

Using the state variables used in Kalman filter, a measurement model for the above observation can be constructed as follows:

Combine Equation (2.8) with (4.42)

$$z_{obs} = [\mathrm{I} - \Phi] C_S^n \tilde{C}_M^S C_n^M \tag{4.43}$$

Let the relation between slave and master transformation matrices are as follows:

$$\tilde{C}_S^M = C_S^M [\mathrm{I} + S(e)] \tag{4.44}$$

where “e” represents the error on slave to master transformation matrix. In other words, it represents the small Euler angles which define the transformation from slave erroneous body frame of coordinates to the slave true body frame defined on slave erroneous body frame of coordinates.

Then using Equation (3.34) and (4.44), the following relation is obtained:

$$C_S^n \tilde{C}_M^S C_n^M = \mathrm{I} - C_S^n S(e) \tilde{C}_M^S C_n^M \tag{4.45}$$

Replace (4.45) in (4.43):

$$z_{obs} = [\mathrm{I} - \Phi][\mathrm{I} - C_S^n S(e) \tilde{C}_M^S C_n^M] \tag{4.46}$$

Neglecting the product of error terms, Equation (4.46) can be written as follows:

$$z_{obs} = \mathrm{I} - \left[C_S^n S(e) \tilde{C}_M^S C_n^M + \Phi\right] \tag{4.47}$$

Assume that

$$C_S^n S(e) \tilde{C}_n^S = C_S^n S(e) C_n^S \tag{4.48}$$

Then, using similarity transformation, measurement equation in vector form can be obtained as follows:

$$z_{meas} = \begin{vmatrix} z_{obs}(2,3) \\ z_{obs}(3,1) \\ z_{obs}(1,2) \end{vmatrix} = C_S^N e + \varphi \tag{4.49}$$

where, indices inside the parenthesis represent the corresponding element of “z_{obs}”. As seen from Equation (4.49), in order to implement attitude measurements in Kalman filter, the error in slave to master transformation matrix (‘e’) has to be modeled and added to Kalman filter’s system model.

If the nominal master to slave transformation matrix “$\tilde{C}_M^S$” is assumed to be constant, then, in a deterministic sense, the propagation model of ‘e’ can be derived by utilizing Equation (4.8), where in this case “$\left[\omega_x\ \omega_y\ \omega_z\right]^T$” in Equation (4.8) corresponds to the following:

$$\left[\omega_x\ \omega_y\ \omega_z\right]^T = \omega_{S\tilde{S}}^S = \left[\mathrm{I} + S(e)\right]\tilde{C}_M^S \omega_{iM}^M - \omega_{iS}^S \tag{4.50}$$

On the other hand, as stated in the Chapter 4.3, due to the insufficient transmission rate of aircraft avionics, usually it is not possible to solve Equation (4.8) with a sufficient rate. Furthermore, the rotation rate supplied by the slave will be erroneous which also affects the solution. Hence, it is not possible to solve Equation (4.8) with sufficient accuracy. Therefore, in Kalman filter implementations, it is preferred to use pure stochastic processes to model the error in slave to master transformation matrices.

To evaluate the performance of Kalman filter which uses only attitude measurements, first of all, the filter was assessed under the rigid aircraft body assumption. According to this assumption, there is no relative motion between master and slave, and therefore the orientation of slave with respect to master stays constant throughout the flight and can be modeled as random constant in Kalman filter system model. Thus, the following model should be augmented to the system model given in Table 4-1 in order to process attitude measurements in Kalman filter.

$$\dot{e} = 0 \qquad (4.51)$$

In the above model, 'e' represents the error on master to slave transformation matrix which is defined in Equation (4.44). The initial uncertainty of 'e' was taken to be the same as initial relative orientation error given in Table 2-2.

The error standard deviation estimates of this filter structure (which has 30 states including 'e') are presented in the following figures.

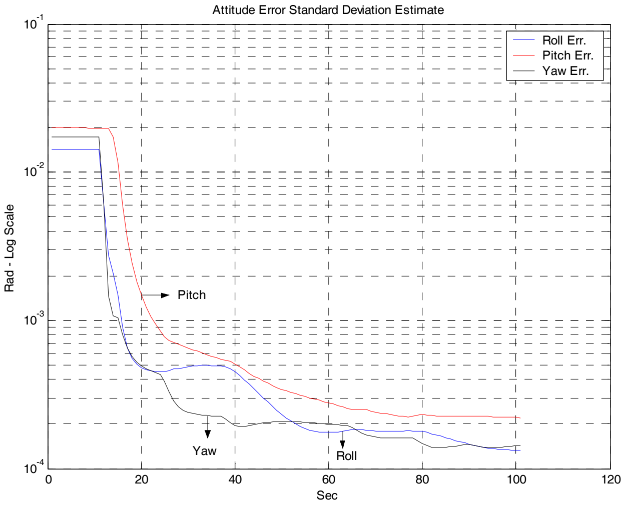


Figure 4-23 : Kalman filter's standard deviation estimate of attitude error states

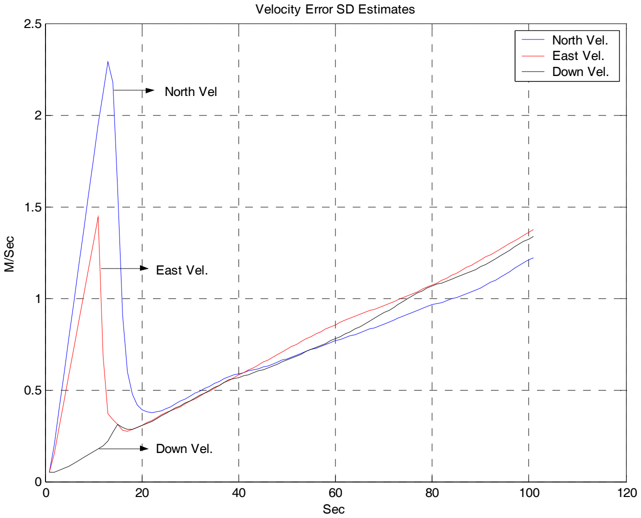


Figure 4-24 : Kalman filter's standard deviation estimate of velocity error states

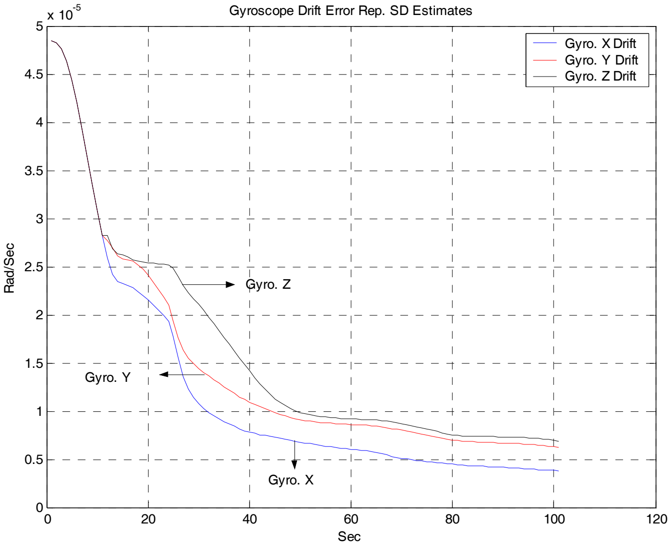


Figure 4-25 : Kalman filter's standard deviation estimate of gyroscope drift errors

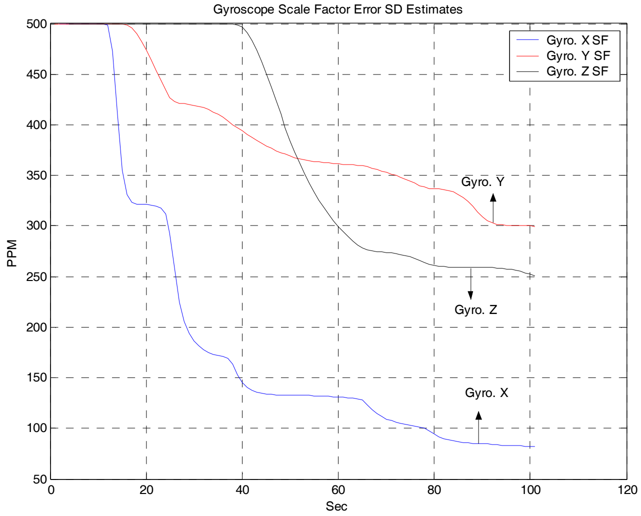


Figure 4-26 : Kalman filter’s standard deviation estimate of gyroscope scale factor errors.

As seen from Figure 4-24, by using only attitude measurements, it is not possible to estimate velocity errors. This is because, the effect of velocity errors on attitude errors are too weak. As shown in Equation (2.10), velocity error excites attitude error via only “ $\delta\omega_{in}^{n}$ ” term, which is calculated by dividing the velocity with the radius of Earth. Therefore, in this method, the effect of velocity errors on attitude errors can not be separated from other major source of attitude errors.

Furthermore, because of this insufficiency in estimating velocity errors, the accelerometer calibration parameters can not estimated either. On the other hand, as can be seen from Figure 4-25 and Figure 4-26, using attitude measurements gyroscope calibration parameters can be estimated quite successfully.

As shown in Figure 4-23, until the first maneuver of aircraft, attitude errors cannot be estimated. This is because; attitude observation method is the same as the initialization method of attitude. This can be shown as follows:

Initial attitude of slave is calculated as follows:

$$\tilde{C}_S^n(0) = C_M^n(0)\tilde{C}_S^M \tag{4.52}$$

Replace Equation (4.44) in (4.52);

$$\tilde{C}_S^n(0) = C_M^n(0)C_S^M\left[\mathrm{I} + S(e(0))\right] = C_S^n(0)\left[\mathrm{I} + S(e(0))\right] \tag{4.53}$$

Replace Equation (2.8) in (4.53):

$$\left[\mathrm{I} - S(\varphi(0))\right]C_S^n(0) = C_S^n(0)\left[\mathrm{I} + S(e(0))\right] \tag{4.54}$$

Rearrange (4.54);

$$\varphi(0) = -C_S^n(0)e(0) \tag{4.55}$$

where "φ" represents the initial attitude error and "e" represents the initial misalignment error. Initial covariance matrix of Kalman filter's attitude and misalignment states are adjusted using Equation (4.55). Therefore, this shows that, during the initialization of state covariance matrix, attitude and misalignment states are not initialized separately but rather, "$\varphi(0) + C_S^N(0)e(0)$" is initialized to be as zero. Thus, unless C_S^N changes (which requires the aircraft to perform a maneuver) attitude error and misalignment states cannot be estimated separately.

Another disadvantage of only attitude matching method is the highly nonlinear structure of observation equation (Equation (4.42)). To obtain a measurement relation from the given observation, small angle assumptions are utilized several times, which in some instances may not be true. For instance, assume that initial misalignment error is [20 25 10]mrad. Regarding the initial misalignment variances presented in Table 2-2, the likelihood of encountering these values is considerably high. Assuming the initial attitude of slave system is [0 0 0.5236] rad, then the error in linearizing the observation equation can be approximated as follows:

$$err_{Meas} = dcm2euler(C_S^n C(e) C_n^S) - C_S^n e \tag{4.56}$$

where "e" is initial misalignment error and C(e) is real direction cosine matrix calculated using "e". Using the above angle values, measurement error can be found as $\mathrm{err_{Meas}}$ = [2.8e-6, 0.8e-6, 0.1738e-3]. As seen from this result, measurement error can be too high to simply ignore. Especially, as this error has a bias like characteristic, this amount of error can easily destroy the gyroscope calibration parameter estimates of Kalman filter.

The effect of this measurement error on filter performance can easily be observed by comparing the Kalman filter standard deviation estimate with that of real standard deviation of z gyroscope drift estimation error which is obtained by Monte Carlo analysis. In Figure 4-27 this comparison is presented. As seen from this figure, after the aircraft starts its maneuver, the difference between estimated and real standard deviations grows. During the first strait flight segment, no difference is observed. This is because, due to the initialization and measurement calculation methods, no measurement error occurs until the aircraft performs a maneuver.

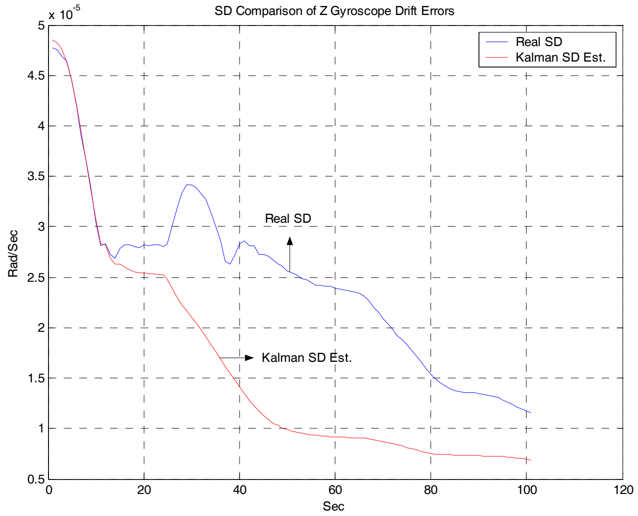


Figure 4-27 : Comparison of standard deviation estimate of Kalman Filter with that of real standard deviation of z gyroscope drift estimation error

In order to avoid this, for only attitude match method, initial misalignment error standard deviations are reduced to [5 5 10] mrad.

To see the effect of wing vibration on attitude match method, vibration models were activated again in the simulation environment. In this case, due to the wing flexure and vibration effects, misalignment errors can no longer be represented as only random constants. Along with random constant, additional states should be augmented to Kalman filter system model in order to represent the dynamically varying component of misalignment angles.

To represent dynamically changing components, following model was used for misalignment angle:

$$e_{Tot} = e_{Stat} + e_{Flex} + e_{vib} \tag{4.57}$$

where “e_{Tot}” represents total misalignment error. “e_{Stat}” denotes the total error arise from the mechanical mismatch such as mounting misalignments or so. Therefore it can be modeled as random constant.

"e_{Flex}" represents the low frequency large amplitude misalignment due to flexure of wing and "e_{Vib}" represents high frequency small amplitude misalignment arise from vibration of wing. In order to model these components usually Markov processes which are at least second order are utilized in Kalman filter system model. However, such an approach requires at least 12 states to be augmented to Kalman filter, which increase the computational complexity very much. Therefore, in this study a simpler approach was implemented. In this approach, "e_{Flex}" component was modeled as a random walk process. Process noise of random walk component was calculated by performing Monte Carlo analysis. This procedure can be summarized as follows; in the simulation environment, by performing Monte Carlo trials, the variance of change in Euler angles (which defines master to slave orientation) due to the non rigid structure was calculated with respect to time, and this variance versus time curve was fitted to a straight line. The slope of this line was used as process noise for random walk component. For the vibration models shown in Appendix B, the variance of process noise was found to be approximately "14e-10" (rad/sec)^2 for 1 second samples.

On the other hand "e_{Vib}" is modeled as white noise. This is because, for 1 second Kalman filter update time, observations for vibration induced motion become essentially uncorrelated, and hence cannot be estimated. Therefore, the effect of "e_{Vib}" was modeled as a part of measurement noise in Kalman filter. The variance of this component is assumed to be "8.7e-009" (rad/sec)^2 which is close to the steady state variance of rotational vibration model shown in Appendix B.

Regarding to the above modifications, the following six states are augmented to the Kalman filter system model given in Table 4-1 (note that Equation (4.58) and (4.59) represents vector variables):

$$\dot{e}_{Stat} = 0 \tag{4.58}$$

$$\dot{e}_{Flex} = n_{Flex} \tag{4.59}$$

where n_{Flex} represents system disturbance in the form of white noise with a power spectral level of "14e-10" (rad/sec)^2/Hz. Also to consider the new states, the measurement model given in Equation (4.49) is modified as follows:

$$z_{meas} = C_S^N\left(e_{Stat} + e_{Flex}\right) + \varphi + \upsilon \tag{4.60}$$

$$\upsilon = e_{Vib} \tag{4.61}$$

Since “e_{Vib}” was modeled as white noise, Equation (4.61) represents the measurement noise with a variance “8.7e-009” (rad/sec)^2.

The estimation performance of this Kalman filter structure was evaluated by performing Monte Carlo analysis. In the following figures, the comparison of Kalman filter standard deviation estimates and real error standard deviations calculated using Monte Carlo trials are presented.

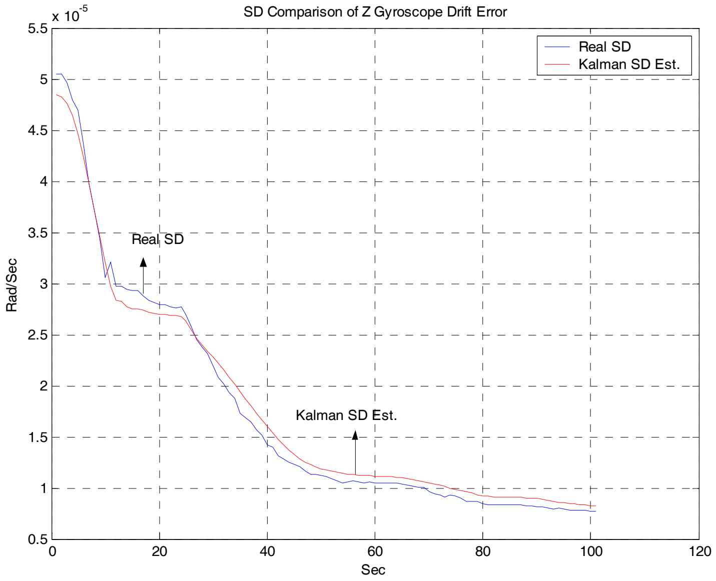


Figure 4-28 : Comparison of real standard deviation and Kalman filter standard deviation estimate for Z gyroscope drift error

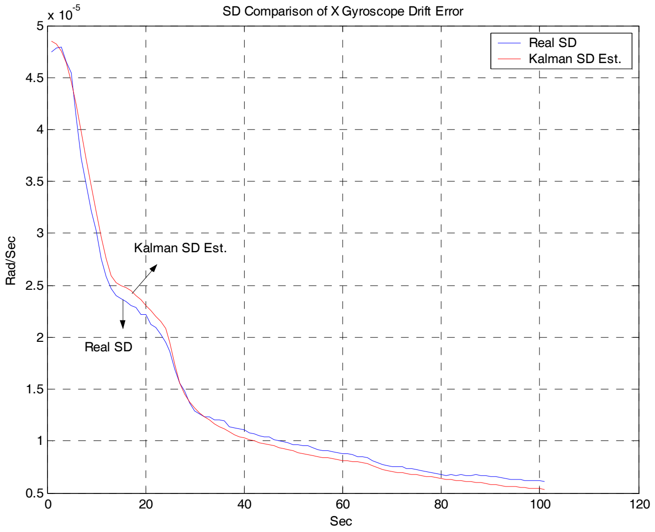


Figure 4-29 : Comparison of real standard deviation and Kalman filter standard deviation estimate for X gyroscope drift error.

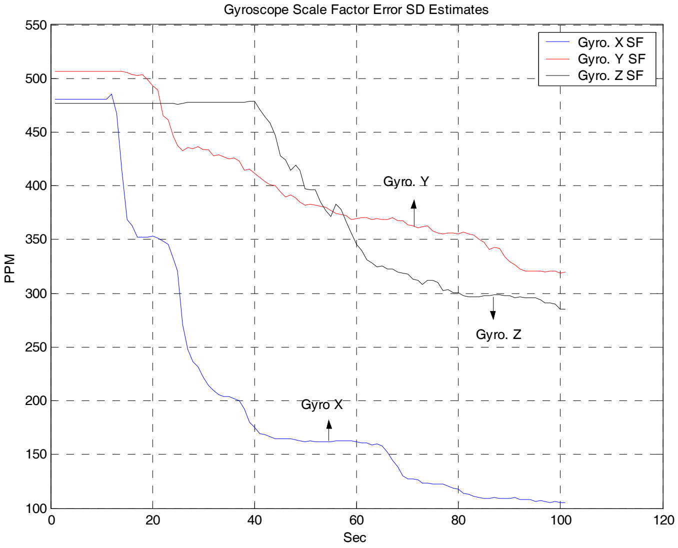


Figure 4-30 : Real standard deviations of gyroscope scale factor errors obtained using Monte Carlo trials.

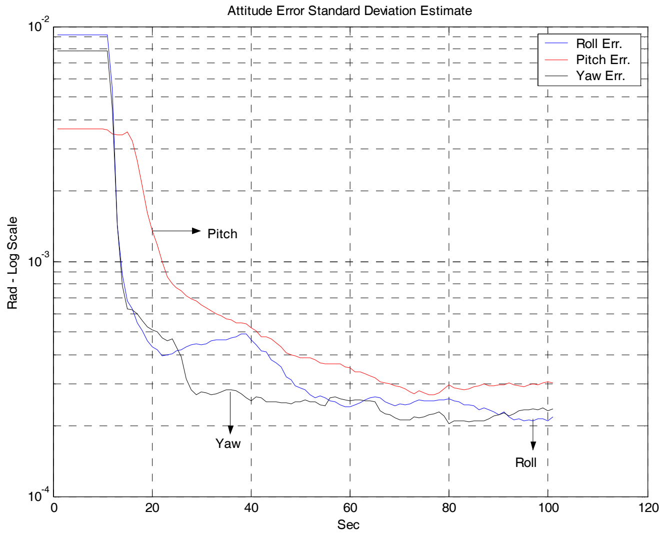


Figure 4-31 : Real standard deviations of attitude error estimates obtained using Monte Carlo trials

As seen from Figure 4-28, by reducing initial uncertainty of misalignment angles, the error occur on gyroscope calibration parameters estimates during aircraft maneuvers can be prevented.

Also, by comparing Figure 4-29, Figure 4-30, Figure 4-31 with Figure 4-25, Figure 4-26 and Figure 4-23, it can be seen that the flexing nature of misalignment angles causes the attitude, gyroscope drift and scale factor error estimates to degrade. Therefore, true modeling of misalignment angles is highly crucial in the implementation of this method. Any error on the model of “e_{Flex}” can either degrade the estimation performance (for models with high uncertainty) or in long term cause the filter to diverge due the wrong standard deviation estimate (for models with low uncertainty). This situation is depicted in Figure 4-32 and Figure 4-33. In these figures, the real standard deviations obtained by Monte Carlo trials and standard deviation estimates of Kalman filter for azimuth error and z gyroscope scale factor errors are compared when process noise for “e_{Flex}” state is reduced to “14e-12” in the Kalman filter system model.

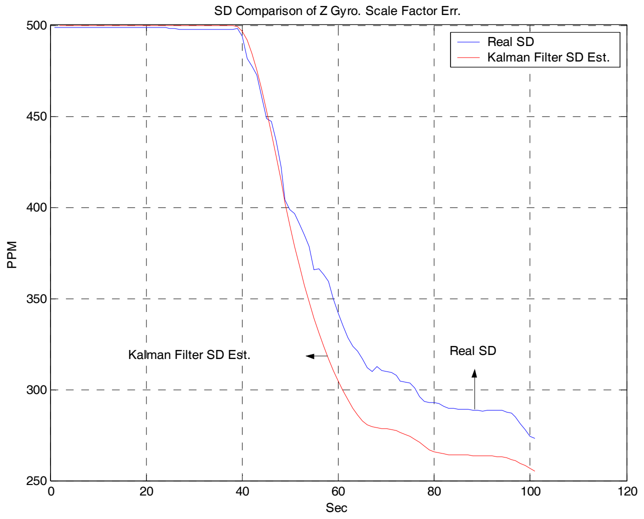


Figure 4-32 : Comparison of real standard deviation obtained by Monte Carlo trials and Kalman filter standard deviation estimate for Z gyroscope scale factor error

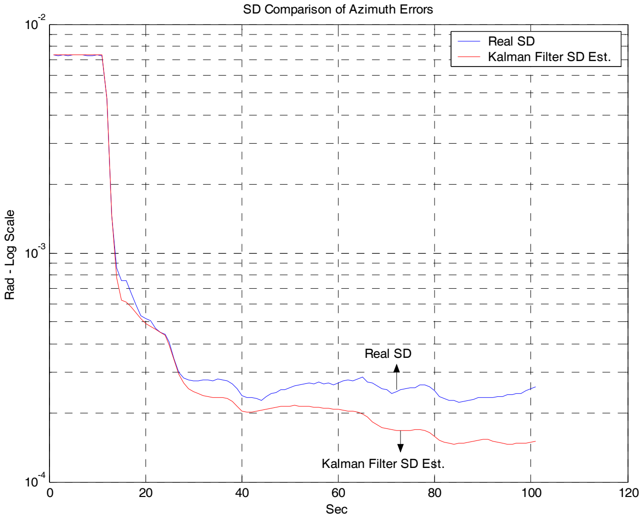


Figure 4-33 : Comparison of real standard deviation obtained by Monte Carlo trials and Kalman filter standard deviation estimate for azimuth error

As shown in the above figures, both the gyroscope calibration parameters and attitude error estimates can easily be disrupted by errors in “e_{Flex}” models.

Therefore, in real time applications, special attention has to be given to accurate modeling of misalignment changes due to the flexure effect.

4.7 VELOCITY AND ATTITUDE MATCHING METHOD

As shown in previous section, by using only attitude measurements it is not possible to estimate accelerometer calibration parameters. Furthermore, attitude matching method is not guaranteed to work for high initial misalignment errors. On the other hand, by comparing Figure 4-31 with Figure 4-10, it can be seen that in the attitude match method, attitude errors can be estimated faster and better than velocity match method. Also, attitude measurements enhance the estimation of gyroscope calibration parameter estimates. Thus, in order to get the advantage of both types of methods in a single filter structure, both velocity and attitude observations should be processed in a Kalman filter at the same time. This technique is called attitude and velocity matching method.

The implementation of this method is as almost same as the previous methods. As the attitude and velocity observation noises are almost uncorrelated, these two observations can be processed in Kalman filter in a sequential manner in order to prevent grow of code mass in the real time implementation.

For the Kalman filter structure using attitude and velocity match methods, Equation (4.22) and (4.49) constitutes the observation equations. Measurement equation for velocity observations is also same as Equation (4.26). However, a slight modification is made for the attitude measurement and related Kalman filter states. As shown in Equation (4.57), total misalignment between master and slave systems can be represented as the sum of static, flexure and vibration components. On the other hand, as stated before, static component is modeled as a random constant whereas flexure component modeled as random walk. Therefore, it is possible to combine these two components and model them as a single state in Kalman filter system model. This new state is also modeled as random walk, but with an initial variance equal to the sum of dynamic and flexure components initial variances. This new misalignment state can be represented as follows:

$$\dot{e} = n_{Flex} \tag{4.62}$$

$$\mathrm{E}\left[ee^{T}\right] = \mathrm{E}\left[e_{Stat}e_{Stat}{}^{T}\right] + \mathrm{E}\left[e_{Flex}e_{Flex}{}^{T}\right] \tag{4.63}$$

Hence, together with the states presented in Table 4-1, total number of states for attitude and velocity match method can be reduced to 30. In this case, the measurement equation for the attitude observations can be represented as follows:

$$z_{meas} = C_S^N(e) + \varphi + \upsilon \tag{4.64}$$

$$\upsilon = e_{Vib} \tag{4.65}$$

In the following figures, the standard deviation estimates of Kalman filter which process velocity and attitude information in a sequential manner are presented. It should be noted that, although not shown below, those results were also verified with Monte Carlo simulations.

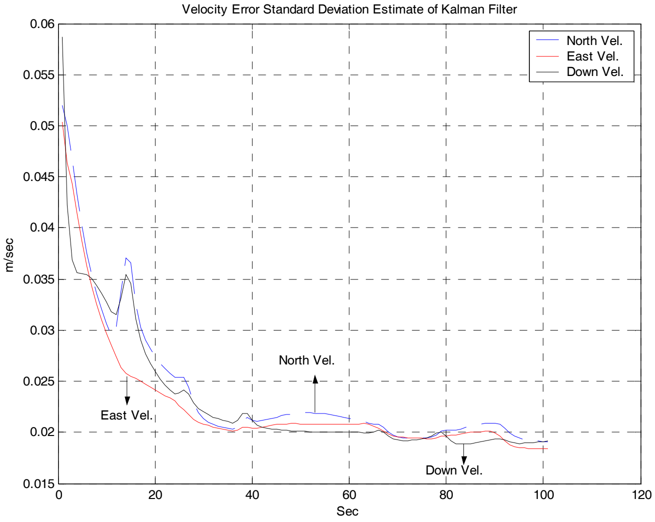


Figure 4-34 : Kalman Filter SD estimate of velocity error for attitude and velocity match method

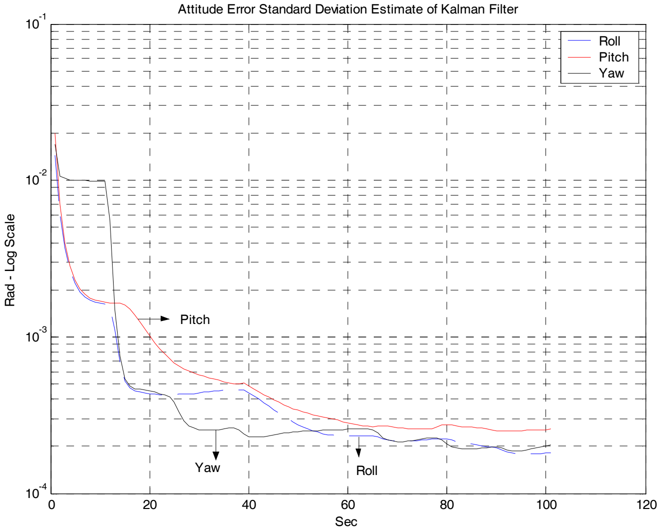


Figure 4-35 : Kalman Filter SD estimate of attitude error for attitude and velocity match method.

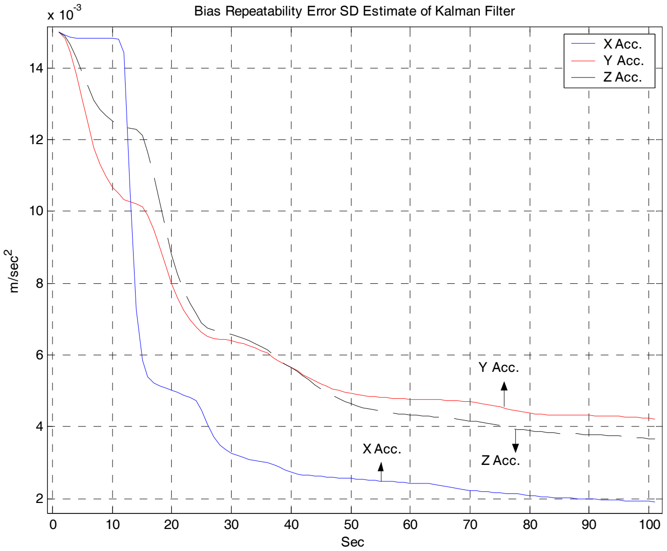


Figure 4-36 : Kalman Filter SD estimate of accelerometer bias error for attitude and velocity match method.

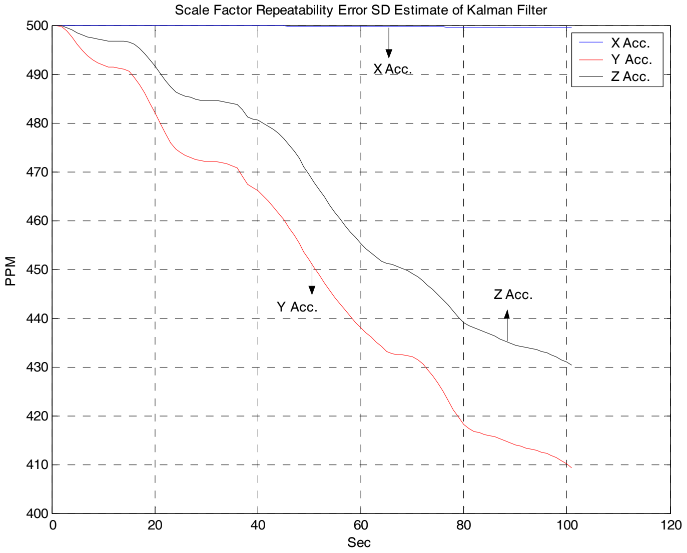


Figure 4-37 : Kalman Filter SD estimate of accelerometer scale factor error for attitude and velocity match method.

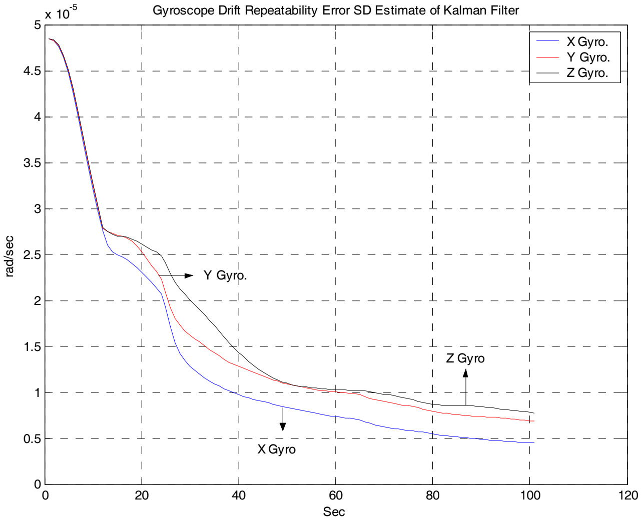


Figure 4-38 : Kalman Filter SD estimate of gyroscope drift error for attitude and velocity match method

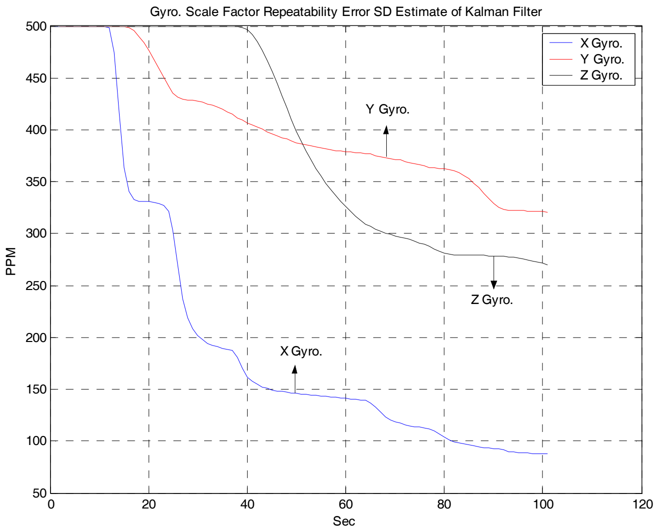


Figure 4-39 : Kalman Filter SD estimate of gyroscope scale factor error for attitude and velocity match method

As shown in Figure 4-35, although attitude observations were utilized, initial misalignment variances were adjusted to be [20 20 10]mrad. This is because, with the utilization of velocity together with attitude observations, horizontal tilt errors can be reduced to under 5mrad within a very short period of time, even if aircraft does not perform a maneuver. Therefore, errors due the nonlinearity of observation do not affect the system performance as in the only attitude match method.

The superiority of attitude and velocity match over only velocity match is represented in the following figures. In these figures, standard deviation estimates of two Kalman filters, one of which uses only velocity observation and the other uses both velocity and attitude observations are compared.

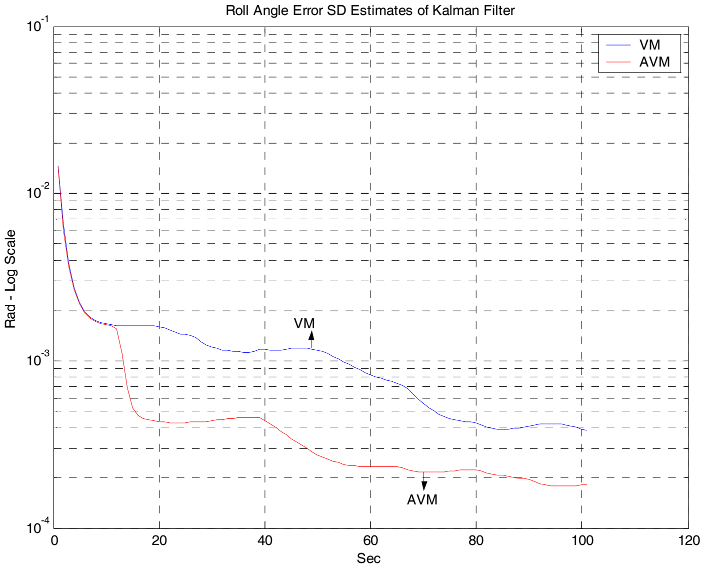


Figure 4-40 : Kalman filter standard deviation estimates of roll angle error for “velocity match” and “velocity and attitude match” methods

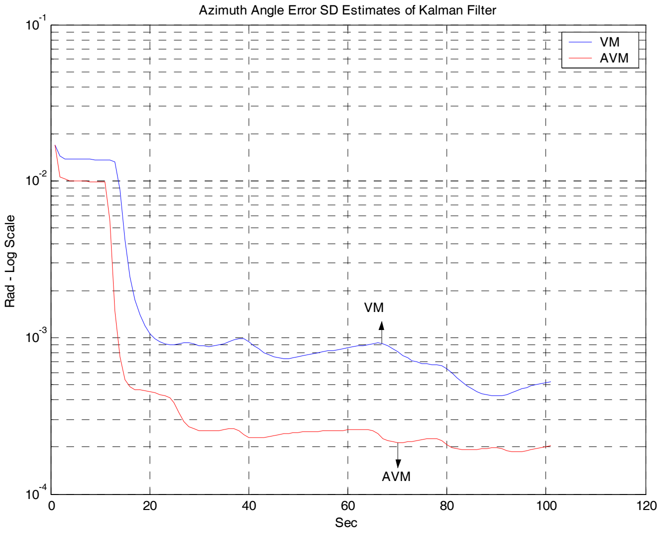


Figure 4-41 : Kalman filter standard deviation estimates for azimuth error for “velocity match” and “velocity and attitude match” methods

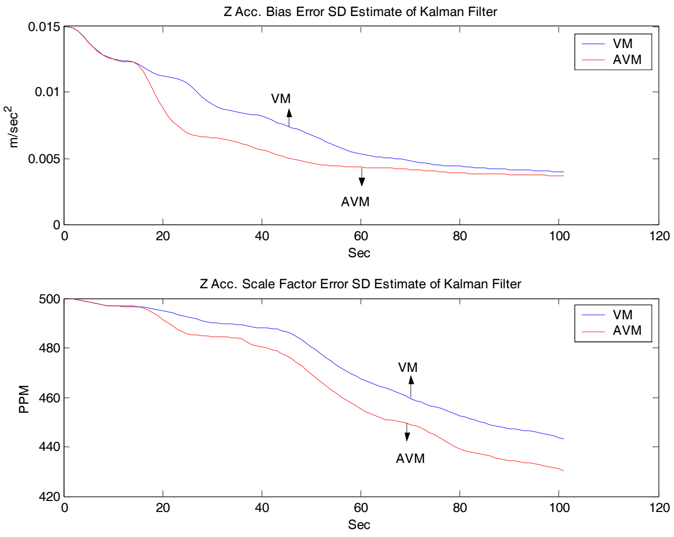


Figure 4-42 : Kalman filter standard deviation estimates for Z accelerometer bias and scale factor errors for "velocity match" and "velocity and attitude match" methods

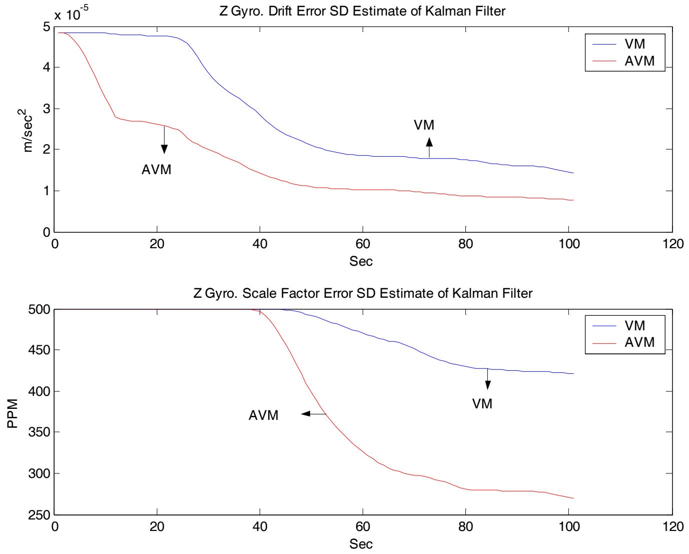


Figure 4-43 : Kalman filter standard deviation estimates for Z gyroscope drift and scale factor errors for "velocity match" and "velocity and attitude match" methods

As seen from the above results, processing attitude and velocity observations at the same time enhance estimation performance of the Kalman filter both in terms of duration and accuracy. As the time required for the Kalman filter to reach a steady state (especially for the attitude error states) are extremely important for most weapon systems (especially for the air to air munitions), this method is the most preferred technique utilized in applications requiring rapid transfer alignment.

On the other hand, the most important drawback of this technique is its sensitivity to modeling errors of misalignment states. As stated above, any error in misalignment models causes the Kalman filter error variance estimates to be substantially erroneous (See Figure 4-32 and Figure 4-33). For long operation durations, such a situation may cause the filter to diverge. However, modeling the flexure effect correctly can be extremely difficult. As shown in Roger (1996), for an F16, the flexure on the aircraft wing can reach to 3mrad under the maneuver with 30°-40° bank angle and return to 0° when aircraft completes its maneuver. This proves the fact that; along with many other parameters, flexural misalignment is a function of aircraft bank angle. Therefore fitting a generic stochastic model for misalignment without considering the maneuver of aircraft can lead to very erroneous results. In the past some studies about relating the aircraft maneuver with the stochastic models which represent flexural misalignment was published. For instance in Spalding (1992), flexure is modeled with third order Markov processes and the effective correlation time of the process is adjusted using the aircraft rotation rate. Even if such approaches are sufficient to consider the effect of aircraft maneuver on misalignment, it is not possible to model all the effects leading to spontaneous alignment changes such as weapon release, whether condition change etc. using stochastic models.

Therefore, for attitude observations, there is always a risk of occurring modeling errors which degrade the filter performance too much. On the other hand, the azimuth difference between master and slave systems is not affected much by the wing flexure effect as the roll and pitch differences do. Therefore, as suggested in Reiner (1996), instead of using all attitude information, by using just attitude difference in “yaw” axes, a measurement which is less sensitive to flexure can be formed. This method can be implemented as follows:

In Equation (4.42) the observation is formed in such a way that, a linear combination of misalignment angles appears in measurement model (see Equation (4.49)). However, in this method, the aim is to use just the misalignment in "yaw" axes. Therefore, instead of Equation (4.42), the following observation is used:

$$z_{obs} = \tilde{C}_n^S C_M^n \tilde{C}_S^M \tag{4.66}$$

In this case, a measurement model for only "yaw" misalignment can be constructed as follows:

$$z_{meas} = z_{obs}(1,2) = e_z + \left(C_N^S \varphi\right)(3,1) \tag{4.67}$$

where e_z represents the total misalignment in yaw axes.

As only "yaw" misalignment appears in measurement model, in the Kalman filter system model only this variable is modeled as a state with the following propagation model (see Equation (4.62)):

$$\dot{e}_z = n_{Flex\,z} \tag{4.68}$$

where $n_{Flex\,z}$ represents the "z" component of n_{Flex}.

Therefore, together with the states shown in Table 4-1, total state number of Kalman filter turns out to be 28 (note that in attitude and velocity match method total state number was 30).

In the following figures, the standard deviation estimates of 3 Kalman filters are compared. In the first structure, a Kalman filter which process velocity and "yaw" misalignment angle measurements was implemented using the method described above. The other two structures corresponded to Kalman filters which were in the form of "velocity match" and "velocity and attitude match" structures explained in previous sections.

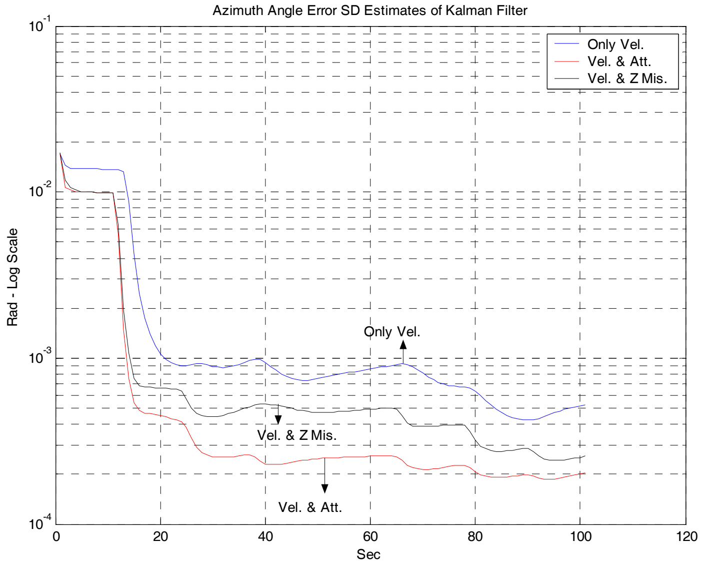


Figure 4-44 : Kalman filter azimuth error standard deviation estimate comparisons for “velocity match” (VM) method, “attitude and velocity match” (AVM) method and “velocity and yaw misalign match” method.

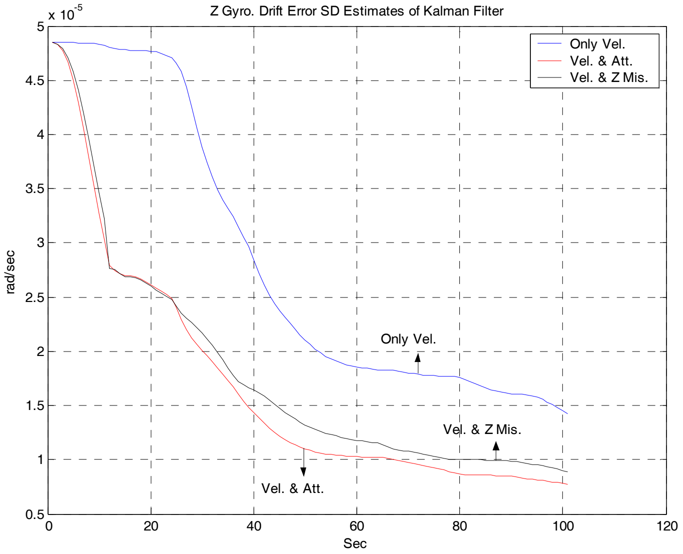


Figure 4-45 : Kalman filter Z gyroscope drift error standard deviation estimate comparisons for “velocity match” (VM) method, “attitude and velocity match” (AVM) method and “velocity and yaw misalign match” method.

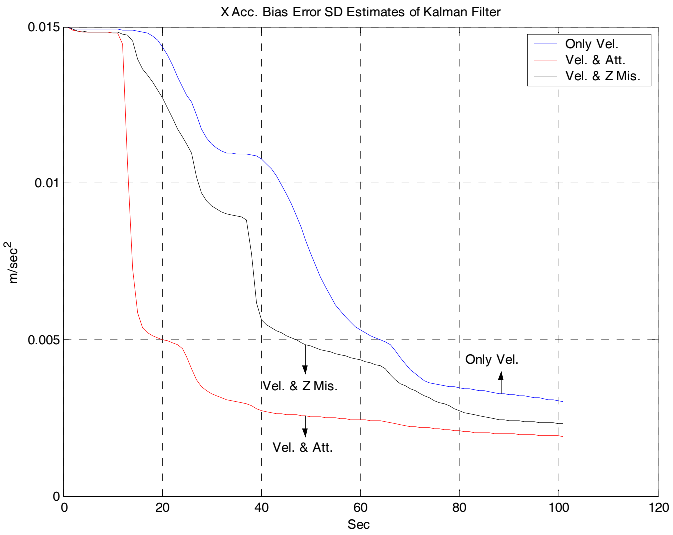


Figure 4-46 : Kalman filter X accelerometer bias error standard deviation estimate comparisons for "velocity match" (VM) method, "attitude and velocity match" (AVM) method and "velocity and yaw misalign match" method.

As seen from the above figures, by using only yaw misalignment observations together with velocity observations, filter performance can be enhanced with respect to only velocity match method in terms of both accuracy and duration.

The most important advantage of utilizing attitude observations (both for three and single axis measurements) in Kalman filter is their ability to limit the azimuth error increase during the level and straight flight of the aircraft. As seen from Figure 4-44 after the aircraft completes its maneuvers and starts straight flight segment (which corresponds to approximately after 90th seconds), the azimuth error begins to increase in Kalman filters which use only velocity observations. This is because, during straight flight, azimuth error and velocity observations become almost uncoupled, and hence no azimuth error estimates can be calculated. However, if the misalignment in "yaw" axes can be estimated before last straight flight segment, then the azimuth error grow will also be bounded by the grow of "yaw" axis misalignment error. This situation is represented in Figure 4-47. In this figure, azimuth error standard deviation estimates of the previous three Kalman structures are presented for a 25 minutes trajectory. In this trajectory, after 90th seconds, the aircraft totally undergoes a level and straight flight.

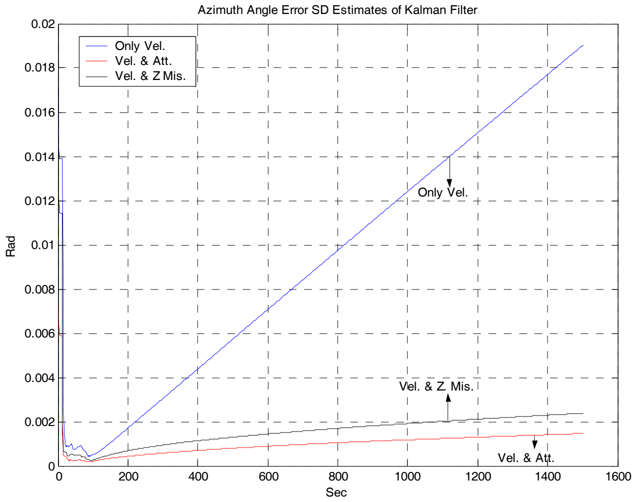


Figure 4-47 : Kalman filter azimuth error standard deviation estimate comparisons for "velocity match" (VM) method, "attitude and velocity match" (AVM) method and "velocity and yaw misalign match" method.

As seen from the above figure, the utilization of attitude information in Kalman filter measurements greatly reduces the azimuth error increase during the level and straight flight segment. For velocity match method, the increase in azimuth error mainly characterized by the gyroscope residual drift and gyroscope random walk error after the transfer alignment maneuvers are completed. However, with the utilization of attitude information, that increase can be limited by the model used for misalignment states. In the Kalman filter structures used in above simulations, misalignment states were modeled as random walk components (see Equations (4.62) and (4.68)). Therefore, in filter structures where attitude observations are used, the azimuth error increase was mainly characterized by the process noise variance used in misalignment models. It should be noted that, if a stable model were used for misalignment states, then the azimuth error increase could also be bounded above for Kalman filters which utilize attitude observations.

As seen from above explanations, the accurate modeling of misalignment states is extremely important in Kalman filter structures which use attitude observations. However, in order to derive an accurate model for the effect of flexure and vibration on the misalignment states, lots of flight tests have to be conducted to determine aircraft structural properties. As the developed vibration models were not able to model those effects accurately, in this study techniques which use attitude observations are not considered any more.

On the other hand, it should be noted that, misalignment modeling error basically distorts calibration parameter estimates of Kalman filter. Therefore; if the main objective of the designed transfer alignment algorithm is to estimate attitude errors in a shortest time without considering calibration parameter estimates, then velocity and attitude matching method would still be the optimal solution for these kinds of problems. However, in the scope of this study, the robustness of transfer alignment algorithms in estimating calibration parameters are evaluated as more important than reducing the total transfer alignment duration. Hence, in Chapter 5, only velocity and integrated velocity techniques are studied further without considering methods which also utilize attitude observations.

CHAPTER V

ANALYSIS OF TRANSFER ALIGNMENT ALGORITHMS BASED ON VELOCITY OBSERVATIONS

As shown in Chapter 4, Kalman filters which process velocity based observations turn out to be most appropriate transfer alignment structures in terms of satisfying following objectives:

i. Algorithms are not affected much from flexure and vibration environment.

ii. Algorithms can estimate the IMU calibration parameters and attitude errors fairly well.

In this chapter, the effectiveness and performance of transfer alignment algorithms based on velocity and integrated velocity matching methods under several conditions will be assessed by simulations. The change in estimation performance under different Kalman filter configurations will also be evaluated.

In the following analysis, it was assumed that, maximum rate of data supplied by master navigation system is 20Hz. Therefore, both maximum update rate for Kalman filter (refer to Appendix C for the definition of update rate) and integration frequency for the integrated velocity match technique was limited to 20Hz.

In the following explanations, the Kalman filter performances are evaluated using their standard deviation estimates even for configurations which are not optimal. It should be noted that, even if not shown explicitly, those estimates were also verified by Monte Carlo simulations. In the simulations, the maneuver defined in Figure 4-3 was used as the nominal trajectory unless otherwise is stated.

5.1 COMPARISION OF VELOCITY AND INTEGRATED VELOCITY MATCHING METHODS

As shown in Chapter 4, in velocity matching method, compensated velocity differences between master and slave systems are used as observation for Kalman filter, whereas in integrated velocity matching method the integral of direct difference of velocities are used. In this section, the estimation performances of Kalman filters which use these methods are compared.

In the following figures, the standard deviation estimates of 3 different Kalman filters are presented. The filter configurations are as follows:

i. Configuration I (VM – 1Hz): In this configuration a Kalman filter which process velocity observations at 1Hz using velocity match technique was implemented.

ii. Configuration II (VM – 10Hz): In this configuration a Kalman filter which process velocity observations at 10Hz using velocity match technique was implemented.

iii. Configuration III (IVM – 1Hz): In this configuration a Kalman filter which process integrated velocity observations at 1Hz using integrated velocity match technique was implemented. The integration of velocity differences was performed at 20Hz. Integration error was assumed to be in the form of white noise with variance equals to 4e-5 (m^2) ("υ_{Integ}" term in Equation (4.39))

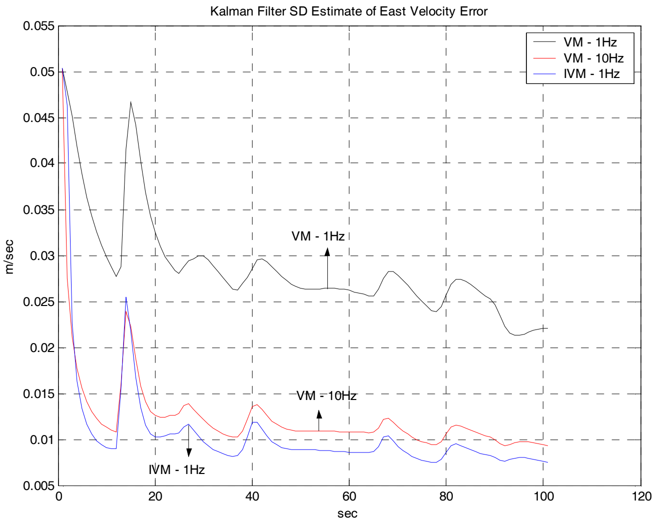


Figure 5-1 : Kalman filter standard deviation estimates for east velocity error

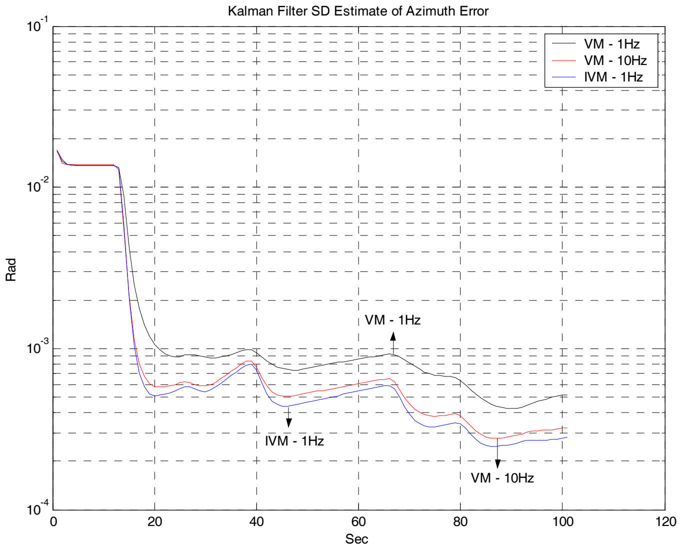


Figure 5-2 : Kalman filter standard deviation estimates for azimuth error

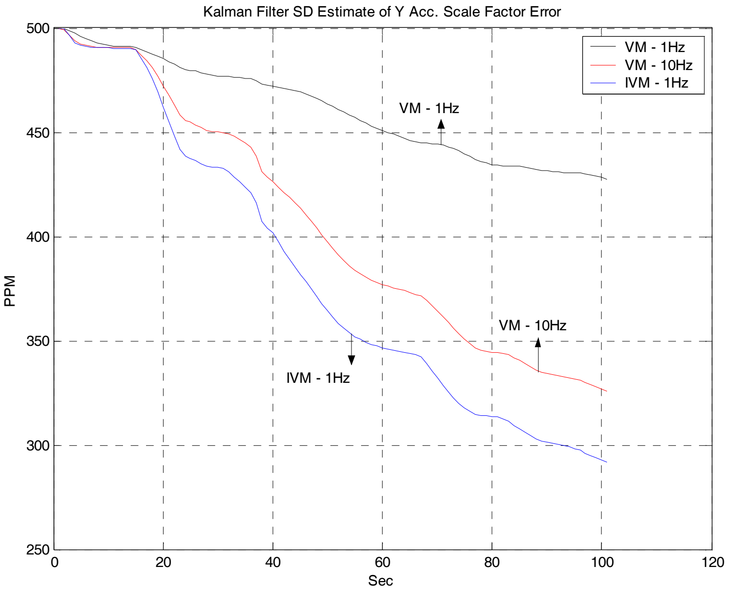


Figure 5-3 : Kalman filter standard deviation estimates for Y accelerometer scale factor error

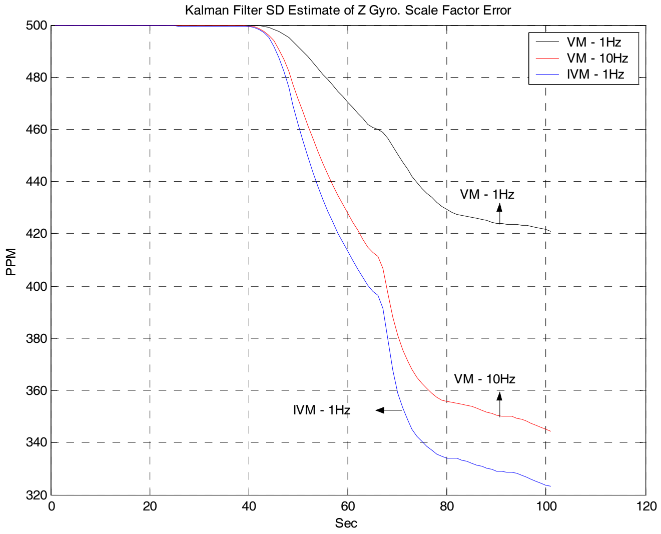


Figure 5-4 : Kalman filter standard deviation estimates for Z gyroscope scale factor error.

As shown in above figures, integrated velocity match technique calculates better estimates for all states. This is expected, because, as shown in Chapter 4, under unmodeled correlated measurement noise, Kalman filter with low update rate can generate better estimates with integrated measurements. Moreover, it should be noted that the Kalman filter which uses velocity match technique at 10Hz is able to show similar performance with integrated velocity match technique at 1Hz.

On the other hand, unfortunately, 20Hz is a critical lower limit for the integration of measurements. As shown in Chapter 4, measurement integration errors are represented as white noise in Kalman filter measurement equation (see Equation (4.39)). However, at 20Hz, significant amount of integration error occurs at the integral of measurements which can not be modeled by simple stochastic models. This is shown at Figure 5-5. In this figure, the difference between integral of North velocity of master system calculated at 20Hz and 600Hz with Euler integration method is presented.

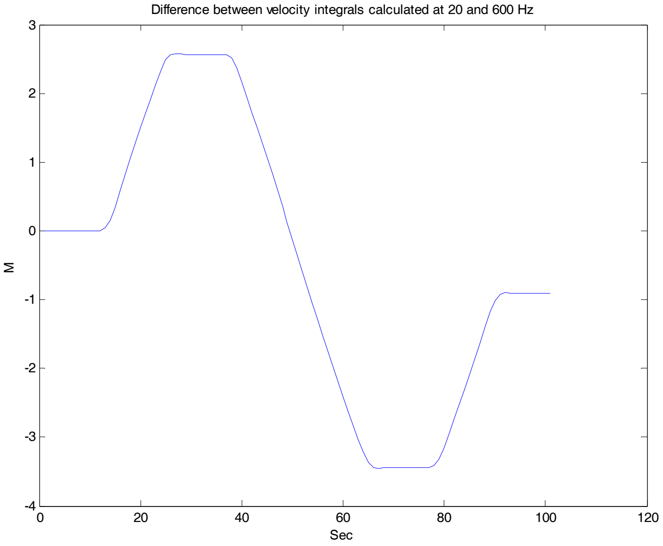


Figure 5-5 : The difference between integral of North velocity calculated at 20Hz and 600Hz with Euler method.

As shown in the above figure, 20Hz is insufficient to take integral of velocities. On the other hand, as shown in Equation (4.38), observations are calculated by taking the integral of velocity differences between master and slave system. Therefore, it is not needed to take integrals of master and slave velocities separately at different rates. If the integral of both velocities are calculated at the same rate, then approximately same amount of error occurs on both integrals and by taking the difference between integrals, integrations errors can be eliminated. To show this, the mean and variance of following error component is calculated by using Monte Carlo simulations:

$$err = \int (V_S - V_M) dt \Big|^{20Hz} - \int (V_S - V_M) dt \Big|^{600Hz} \qquad (6.1)$$

where, superscripts over bars represents discrete time integration frequency (Integral operators are implemented with Euler method in discrete time). In the Monte Carlo simulations, real slave velocities under vibration environment are used. In the following figure, the change of mean and standard deviation of North component of above error term (Equation (6.1)) with respect to time is presented.

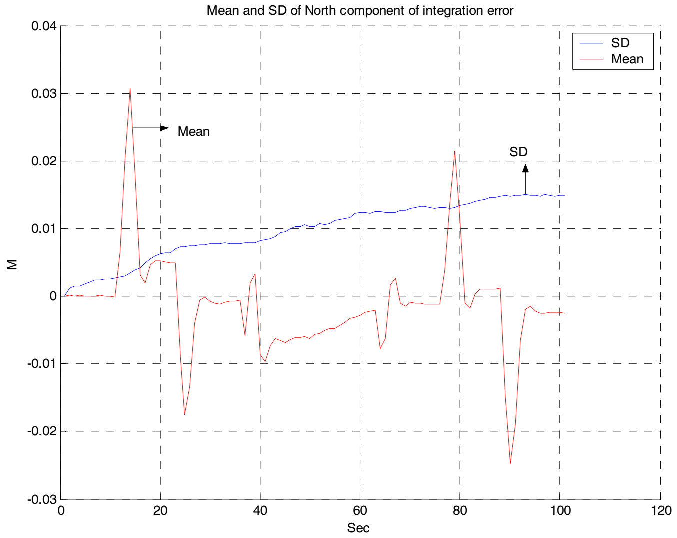


Figure 5-6 : The mean and standard deviation of North Component of integration error

By comparing Figure 5-5 and Figure 5-6, it can be seen that, by taking the integral of velocities at the same rate, we can reduce the integration error. However, still an error with a non-zero mean occurs on measurement, which cannot be represented by stochastic models. Thus, this error with non-zero mean directly contradicts with the Kalman filter's white measurement noise assumption, and therefore can distort the error estimation in an unpredicted way. To show how this kind of error effects the estimation, the Monte Carlo simulation results of the same Kalman filter defined as configuration III is represented in the following figures. In the figures, 'SD of error' and 'Mean of error' denotes the real mean and variance of error calculated using Monte Carlo trials, whereas 'Kalman filter SD est.' denotes the Kalman filter standard deviation estimate of corresponding error state.

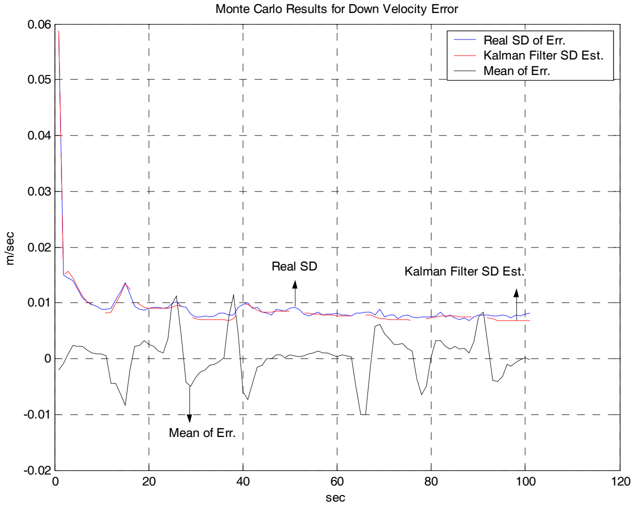


Figure 5-7 : Monte Carlo results for down velocity error estimate of the Kalman filter which uses IVM method.

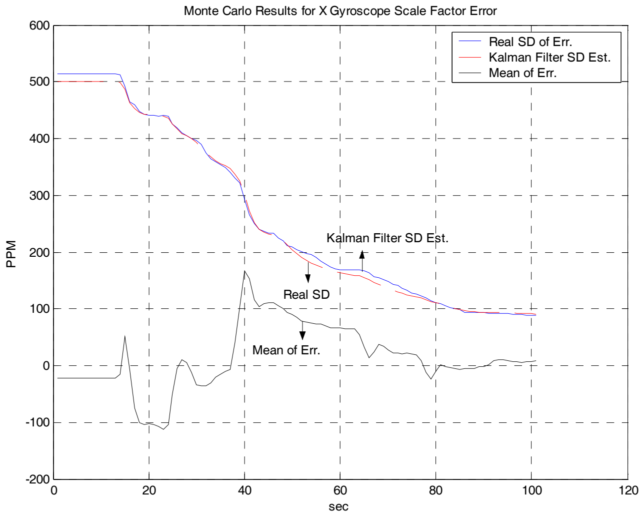


Figure 5-8 : Monte Carlo results for X gyroscope scale factor error estimate of the Kalman filter which uses IVM method.

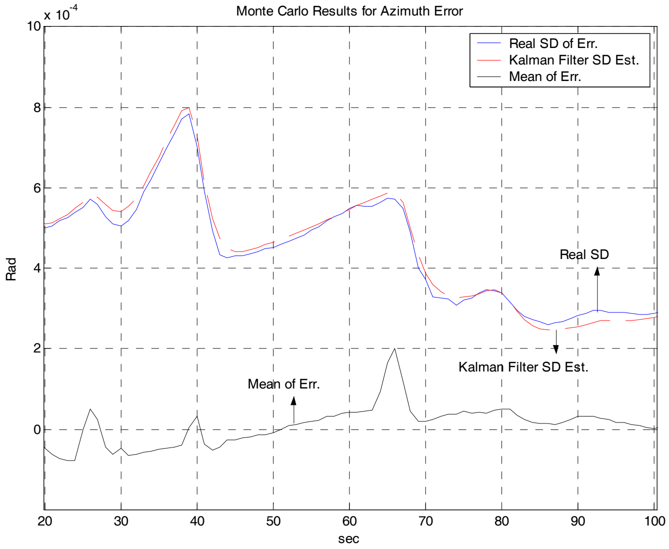


Figure 5-9 : Monte Carlo results for azimuth error estimate of Kalman filter which uses IVM method

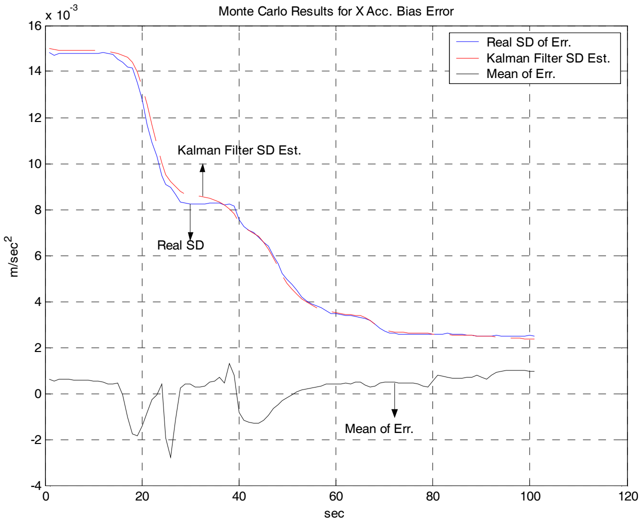


Figure 5-10 : Monte Carlo results for X accelerometer bias error estimate of Kalman filter which uses IVM method.

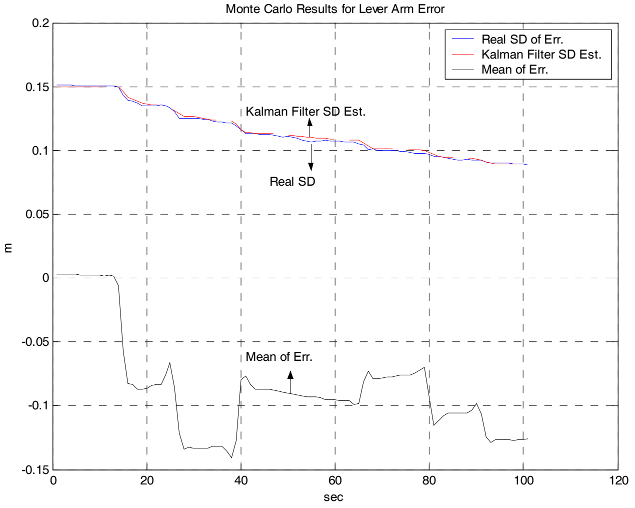


Figure 5-11 : Monte Carlo results for X axes lever arm static error estimate of Kalman filter which uses IVM method

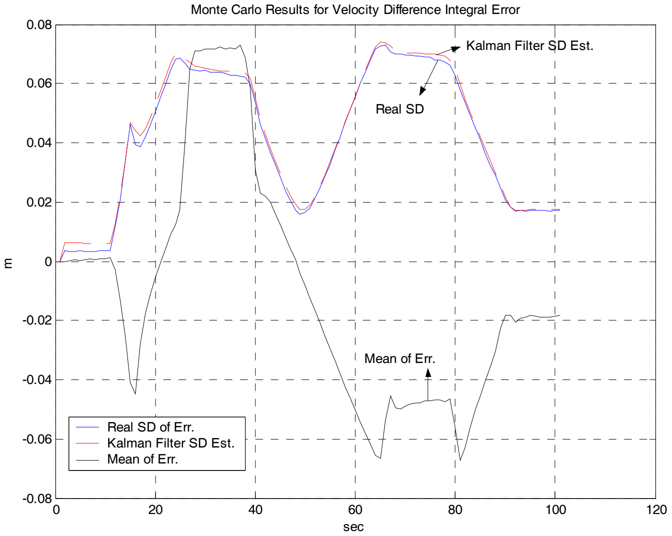


Figure 5-12 : Monte Carlo results for integral of velocity difference estimate of Kalman filter which uses IVM method

As shown in above figures, the measurement integration errors at 20Hz cause the Kalman filter estimates to have errors which are not zero mean. Especially, integral of velocity difference state and static lever arm estimates turns out to be highly erroneous (see Figure 5-11 and Figure 5-12). Furthermore, as shown in Figure 5-10 and Figure 5-8 some of the IMU calibration parameter estimates are also disturbed. However, as presented in Figure 5-9, attitude estimates are not distorted much.

On the other hand, it should be noted that, real error variances calculated by Monte Carlo simulations and Kalman filter’s standard deviation estimates were perfectly matched to each other. This shows that, the errors on estimates are truly result of measurement noise with non-zero mean. Therefore, if somehow integral of measurements can be computed correctly, than, despite of the suboptimalities, Kalman filter structure using integrated velocity match method will work very well. This is presented in Figure 5-13. In this figure same Kalman filter structure (IVM – 1Hz) described above was also used, however, this time integral of velocity differences was calculated at 150Hz. In the following figure, Monte Carlo simulation result for x gyroscope scale factor error estimate is presented.

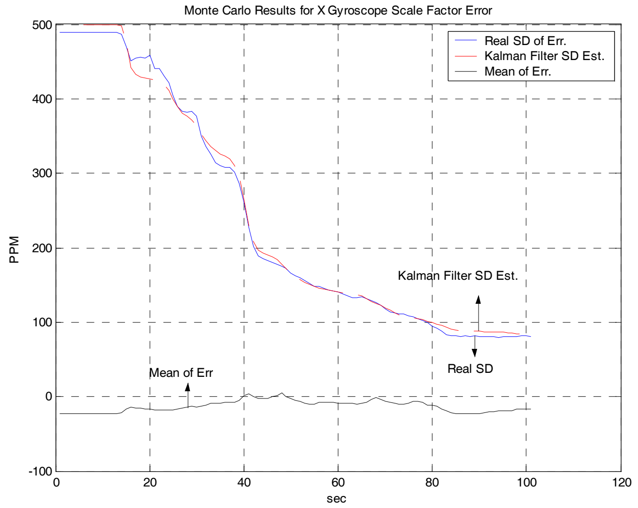


Figure 5-13 : Monte Carlo results for X Gyroscope scale factor error estimate of Kalman filter which uses IVM method (integrals are calculated at 150 Hz).

Comparing Figure 5-13 with Figure 5-8, it can be seen that, by calculating integral of velocity difference with a sufficient rate, the mean error on estimates can be corrected. Also note that, with the faster integration frequency, better estimates can be obtained. (Compare the final error standard deviation estimates obtained at Figure 5-8 and Figure 5-13 which are 91 and 83 PPM respectively). This is because; it was assumed that, the noise at master navigation system velocity output (δV_M^N) is always a white sequence with a variance of 0.05m/sec independent of sampling rate. Therefore, with more frequent measurement integration the effect of this master velocity noise on the integral of velocity difference state (refer to Equation (4.41)) can also be reduced and hence, better estimates can be calculated.

In summary, above results show that, although integrated velocity match technique is generally better than velocity match technique, 20Hz integration frequency is insufficient. Furthermore, considering the transmission errors and data latency/synchronization errors on mux-bus structure, it is expected that during real time implementation integration errors will be much higher than shown in Figure 5-6. Also this integration error will increase as the aircraft performs more dynamic maneuver. Therefore, it is concluded that integrated velocity match technique is not applicable with master navigators which have 20Hz output rate.

Hence, in the rest of this study, main analyses are focused on just velocity matching method.

5.2 EFFECT OF KALMAN FILTER UPDATE RATE

As stated before, it is assumed that master navigation system can provide its output with a rate up to 20Hz. Therefore, theoretically it is possible to process all these information as observations in a Kalman filter which has an update rate of 20Hz. In this section, the effect of update rate on estimation performance of transfer alignment Kalman filter is presented.

To evaluate how update rate effects Kalman filter estimates, a Kalman filter which uses velocity match method was run with 1Hz, 2Hz, 5Hz, 10Hz and 20Hz update rates (the structure of the Kalman filter was the same as described in Chapter 4.4). In the following figures, the standard deviation estimates of the Kalman filters are presented.

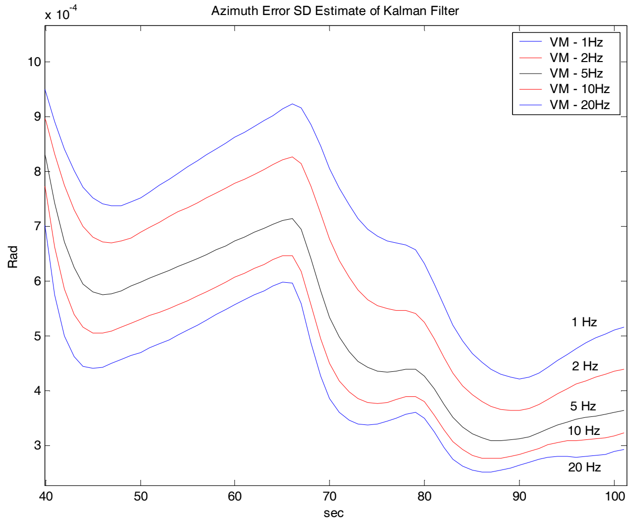


Figure 5-14 : Effect of update rate on Kalman filter azimuth error standard deviation estimate

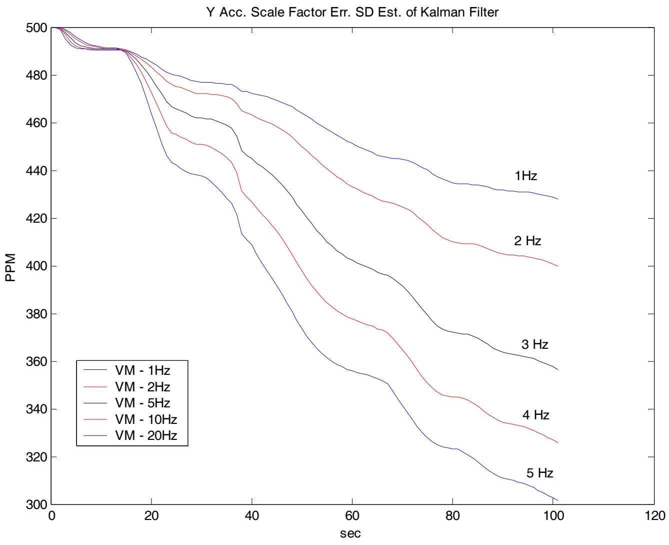


Figure 5-15 : Effect of update rate on Kalman filter Y accelerometer scale factor error standard deviation estimate

As shown in Figure 5-14 and Figure 5-15, as the filter update rate increases, Kalman filter can produce better estimates. This is also true for all other filter states which are not shown here. However, it should be noted that, while performing above simulations, it is assumed that, measurement noise on master supplied information has always a constant variance which does not change with sampling rate (for instance δV_M^n is assumed to have same stochastic properties for all transmission frequency).

On the other hand, as shown in Chapter 4, as the filter update rate increases, Kalman filter estimates become more vulnerable to unmodeled correlated measurement noises. Considering the Kalman filter structure based on velocity matching method described in Chapter 4, there are 3 major source of correlated measurement noise. These are:

i. Vibration induced velocity difference between master and slave (the effect of "$\dot{r}$").

ii. Lever arm correction error due to simplified lever arm models (the effect of "δr").

iii. Correlated noise on master supplied information (the effect of correlated "δV_M^n").

In the implemented Kalman filter structures, the effect of vibration induced velocity is considered as purely white noise and handled as a part of measurement noise although it is not white. However, considering the power spectral properties of vibration induced acceleration models shown in Appendix B, it can be seen that, this approximation does not affect the estimation performance much.

In contrary to this, it is expected that the models which are used to represent lever arm error can be erroneous. As shown in Chapter 3.3 the vibration models are not sufficient to accurately characterize vibration induced displacement effects. Therefore, as shown in Chapter 4, lever arm errors were simply modeled as random constants without prior sufficient information. However, it is expected that, during the maneuvers of aircraft, the wing structure may flex and a random change in lever arm vector may appear. Such an unmodelled change in lever arm causes correlated noises to appear on measurements which can disturb the Kalman filter estimations very much.

To show how lever arm modeling errors can affect estimation performance, error estimations of two Kalman filter structures both of which uses velocity match technique were compared. In the first Kalman filter, the static lever arm errors were again modeled as random constants. Since, static lever arm errors were also generated as random constants in the simulation environment, this filter represented the optimal case. In the second Kalman filter structure, the states representing the static lever arm errors were deleted, and their effect was represented as white measurement noise with a variance proportional to the roll rate of master system in the measurement model of the Kalman filter. In the simulation environment, the standard deviations shown in Table 2-2 were used to generate static lever arm errors. Both Kalman filter structures were run at 1Hz and 10Hz respectively. In the following figures, the Monte Carlo simulation results and Kalman filter standard deviation estimates are presented. In the figures, ‘VM’ represents the Kalman filter in which static lever arm errors are modeled as random constants, whereas ‘VMSub’ represents the Kalman filter in which static lever arm errors were not included as states in the filter’s system model. Real standard deviations were obtained using Monte Carlo simulations based on ‘VMSub’ filter structures.

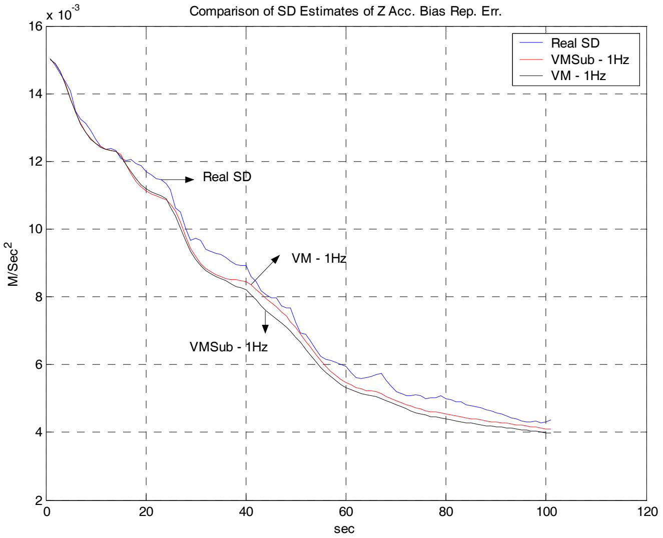


Figure 5-16 : Comparison of standard deviation estimates for Z accelerometer bias error obtained using Kalman filters with 1Hz update rate.

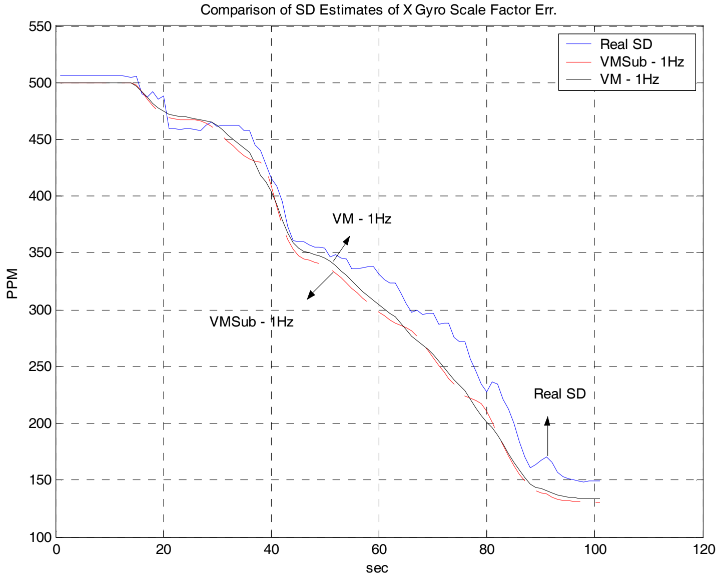


Figure 5-17 : Comparison of standard deviation estimates for X gyroscope scale factor error obtained using Kalman filters with 1Hz update rate.

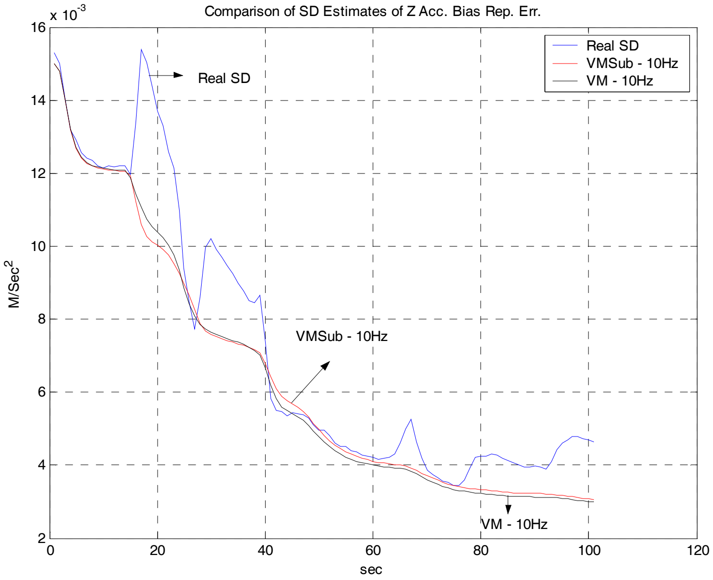


Figure 5-18 : Comparison of standard deviation estimates for Z accelerometer bias error obtained using Kalman filters with 10Hz update rate.

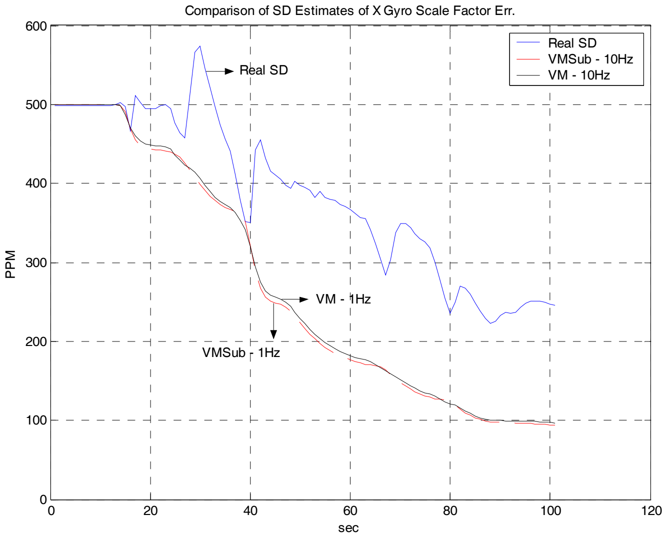


Figure 5-19 : Comparison of standard deviation estimates for X gyroscope scale factor error obtained using Kalman filters with 10Hz update rate

Comparing Figure 5-16 and Figure 5-17 with Figure 5-18 and Figure 5-19, it can be seen that, at 1Hz the unmodeled static lever arm errors does not degrade the filter performance much. This is also verified with Monte Carlo simulations. Although there is a small difference between Kalman filter estimated error standard deviations and real error standard deviations obtained with Monte Carlo simulations, these differences are not too big to cause divergence.

On the other hand, when the filter update rate is increased to 10Hz, the unmodeled static lever arm errors become very significant and totally disturb the Kalman filter IMU calibration parameter estimates. Furthermore, as shown in Figure 5-18 and Figure 5-19, at 10Hz update rate, Kalman filter cannot estimate the real error standard deviations which in long term probably cause divergence in filter structure.

As stated before, due to the maneuver of the aircraft, flexure effect causes changes in lever arm vector which cannot be easily represented by stochastic models. If, for some reason, it is required to run Kalman filter with high update rate, the error caused by mismodeled lever arm state, will degrade the filter performance and probably will cause the filter to diverge.

A simple way of reducing the effect of lever arm modeling errors on estimation performance of Kalman filters which uses velocity match technique is the implementation of Schmidt-Kalman filter. In the Schmidt – Kalman filters, the states that are not of primary interest are considered but not actually implemented in the filter (Schmidt (1966), Brown (1997)). The lever arm errors are estimated not because they are required for slave navigation system, but because they are coupled with navigation states in measurement equations. As they cannot be modeled accurately, the Kalman filter estimates about them are not reliable. Therefore, Schmidt – Kalman filter is an appropriate solution for handling lever arm errors.

In the implementation of Schmidt Kalman filter, the only modification is made to the Kalman gain. Static lever arm errors are also modeled as random constants (no reduction is made to the number of states used). However, after an optimal Kalman gain is computed, gain that corresponds to the lever arm errors are zeroed and this new gain is used to calculate feedback and new covariance matrix value. As a suboptimal gain is utilized, in the calculation of covariance of updated estimates, Joseph Form (Brown (19997) of covariance update is used.

In the following figures standard deviation of errors calculated by a Schmidt Kalman filter which uses velocity match technique at 10Hz is presented.

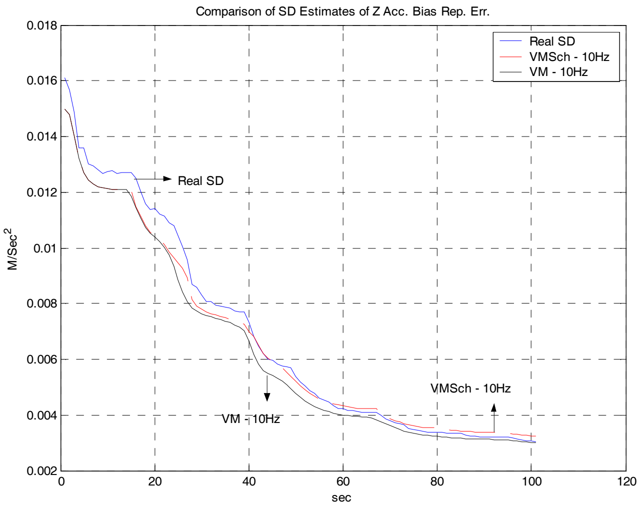


Figure 5-20 : Comparison of standard deviation estimates for Z accelerometer bias error obtained using Schmidt Kalman filter which uses velocity match method with 10Hz update rate

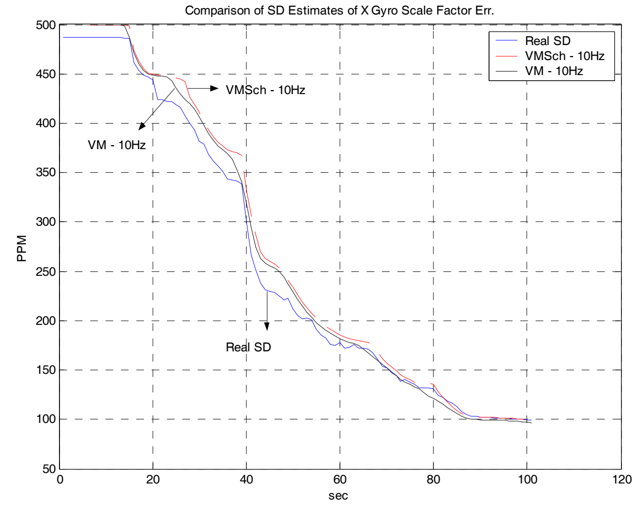


Figure 5-21 : Comparison of standard deviation estimates for X gyroscope scale factor error obtained using Schmidt Kalman filter which uses velocity match method with 10Hz update rate

As shown in above figures, Schmidt Kalman filters work perfectly. Although the filter structure is totally suboptimal, the filter can function as if it were optimal. It should be noted that, in this structure lever arm errors are not estimated at all, however their effects are handled so cleverly that they cannot degrade the filter estimation performance. This shows that, even if the lever arm changes during the maneuvers, that change did not effect the filter operation. Therefore, filter sensitivity to lever arm errors for high update rates can be reduced.

5.3 EFFECT OF AIRCRAFT MANEUVERS ON KALMAN FILTER ESTIMATES

As can be seen from error propagation models presented in Chapter 2, propagation of the errors is affected by the motion that the system follows. This dependence of error propagation to the system's motion can be revealed by examining the existence of navigation parameters (position, velocity, attitude, acceleration, rotation rate) in the propagation models given in Equation (2.10) and (2.16). Although all of the navigation parameters exist as a coefficient in the error models, the models are not sensitive to the variations of these parameters except the instantaneous acceleration of the system which has a great impact on the way the error builds up in time (see Chapter 2). Therefore, by forcing the system to be exposed some specific accelerations, it is possible to change the error propagation characteristic in such a way that errors become observable in a shorter period of time.

The above phenomenon is also utilized during transfer alignment procedures. By forcing the system to perform some predetermined maneuvers, it is possible to modify the structure of observable space defined by the system model of the Kalman filter so that Kalman filter estimation performance can be enhanced (Rehee (2004)).

In the past, several studies about the relation between the aircraft maneuvers and the observability of the system models were published. Especially, in Itzhack (1980), Porat (1981), Itzhack (1988) and Meskin (1991), profound results were derived. On the other hand, most of the studies that published so far on this issue have the following deficiencies:

i. In these studies, it was aimed to figure out the analytical relations between aircraft maneuvers and system observability. However, in order to derive analytical solutions, a number of assumptions were made which limits the usefulness of the solutions for real systems.

ii. The real time application constraints (such as the aircraft maneuver ability) were not considered in these studies.

Therefore, in this study, it was preferred to follow a heuristic method. On the other hand, in order to reduce total number of simulations, results published in previous studies were also utilized very frequently.

In a theoretical sense, it is possible to define infinitely many kind of maneuver for the aircraft. However, for the real time operation, the following constraints have to be considered in defining the maneuver for long duration transfer alignment procedures:

i. A controlled axial acceleration is not realizable.

ii. Maneuvers requiring altitude change is not practical.

iii. Heading changes are performed with bank-to-turn maneuvers. In order to prevent the maneuver to impose additional load to the pilots, the maximum bank angle should be not exceed 60°.

iv. Total duration of maneuver is the most important factor that determines the applicability of the maneuver during real operation conditions.

Considering the above constraints, in this study only the maneuvers that consist of just coordinated turns were investigated. Also, the maximum duration of transfer alignment procedure was limited to 150sec. On the other hand, this limit does not represent the maximum duration that the transfer alignment algorithms should run. Rather, it represents the total duration for the specific maneuver. After this predetermined maneuver is completed, the algorithms may continue to run until the separation of the slave from the main system. The time between the end of the maneuvers and release of the system should be determined by considering the accuracy of the IMU error models used in Kalman filter. For less accurate IMU models, this duration have to be limited in order to prevent any kind of divergence during operation.

As a first step in determining the structure of maneuver, the acceleration and rotation rate profiles occur on slave IMU system during coordinated turns are investigated. In the following figure, the acceleration outputs of a slave IMU that is nominally aligned with aircraft body axes is presented under the coordinated turn maneuver characterized by the roll angle profile shown in Figure 5-22.

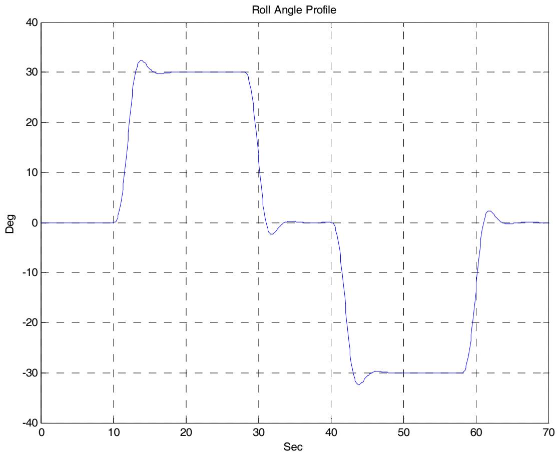


Figure 5-22 : Roll angle profile

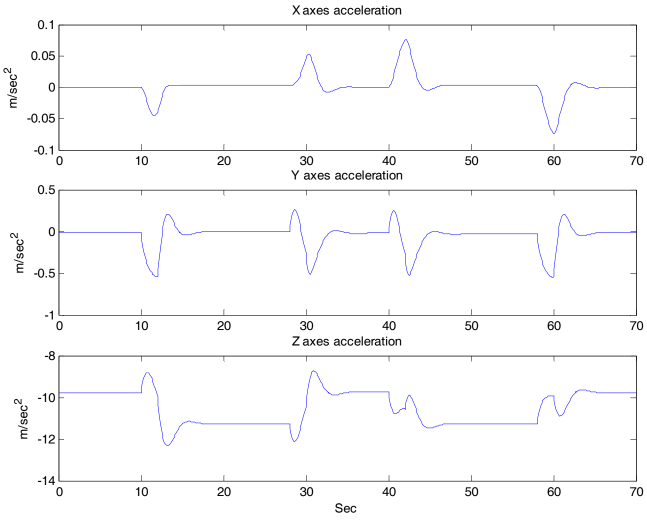


Figure 5-23 : Acceleration profile during coordinated turn

As seen from Figure 5-23, the sum of centrifugal acceleration and gravity appears at only “z” axes during coordinated turn. Only a small amount of additional acceleration appears on “x” and “y” axes which occur due to the roll angle change. However this amount of acceleration is not sufficient to estimate accelerometer scale factor errors. Therefore, in order to be able to estimate these errors, it is necessary to locate slave IMU in such a way that its body axes should not coincide with aircraft body axes. A good choice for slave orientation with respect to aircraft body axes would be [45 45 0]°. However, in this study, considering other implementation issues, the relative orientation was taken to be [52 3 0]°.

In order to figure out the relations between aircraft maneuvers and error estimates, lots of covariance analyses were conducted with different maneuver definitions and filter system models. As the system model is highly complicated and most of the errors are coupled with each other, it was not possible to derive exact relations. However, with these simulations it was possible to gain valuable insights about the error estimates and motion relations which are presented in the following paragraphs.

In Bar-Itzhack (1980 - a) and Porat (1981), it was claimed that under certain conditions, just a single bank-to-turn maneuver is better than any other maneuvers which consist of several bank-to-turns in the same duration. However the covariance analysis shows that, this is not true for the system used in this study. In Figure 5-24, the azimuth error standard deviation estimates of Kalman filter which uses velocity match method along two different maneuvers are presented. In the first maneuver, the aircraft performs just a single 30 degrees bank-to-turn for 70 seconds whereas in the second maneuver 4 consecutive bank-to-turns are performed with 30 or -30 degrees bank angles successively without any intermediate level flight segment.

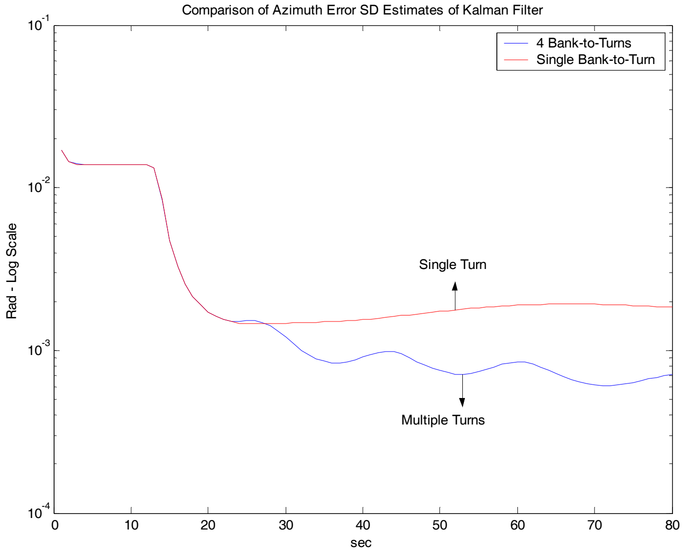


Figure 5-24 : Azimuth error standard deviation estimates of Kalman filter

As seen from Figure 5-24, more than one bank-to-turn maneuver produces better results. It should be noted that, this result does not contradict with the results presented in Porat (1981). This is because, in that study, it was claimed that, their results hold only if the gyroscope drift errors are small. However, the gyroscope drift error standard deviation used in this study is 400 times bigger than the value used in the corresponding paper (See Table 2-1).

Covariance analysis showed that, generally, drift and bias errors are estimated better when there is no horizontal component of acceleration. However, without any horizontal acceleration it is not possible to estimate azimuth and scale factor errors as well as lever arm errors. Therefore both level flight and bank-to-turn maneuvers are highly crucial during transfer alignment procedure. In fact this is one of the reasons why multiple bank-to-turns turn out to better than single bank-to-turn maneuver. This is because, between different bank angles (in the above case between 30 and -30 degrees) the roll angle reduces to zero momentarily, and during that period of time bias and drift estimates can be enhanced which increase the overall estimation performance during heading change portion of the maneuver.

As explained in Chapter 3, a heading change with coordinated turn maneuver consists of 2 different stages. In the first stage, the roll angle is increased from zero to the desired bank angle. This stage can also be seen in Figure 5-22. Between tenth and fourteenth seconds, bank angle gradually increases from 0 to 30 degrees. During this stage, the rate of heading change also increases as the roll angle increases. Also, additional centrifugal acceleration appears on slave system due to the relative motion of slave with respect to master system. In the second stage, heading change is performed under constant bank angle. In this stage the only centrifugal acceleration appears due to the turn of the aircraft as a rigid body. The simulation results show that, the first stage of coordinated turn does not affect the overall estimation performance of Kalman filter. Several simulations are performed with different roll rates, however no significant improvement can be obtained.

On the other hand, it is found that bank angle is highly crucial in defining the maneuver. Any increase in the bank angle also increases the estimation performance of the filter. In the following figures, “x” and “z” axes accelerometer bias estimates of Kalman filter which uses velocity match technique are presented for different maneuvers. Each of the maneuvers has the same roll angle profile presented in Figure 5-22 but with different bank angles.

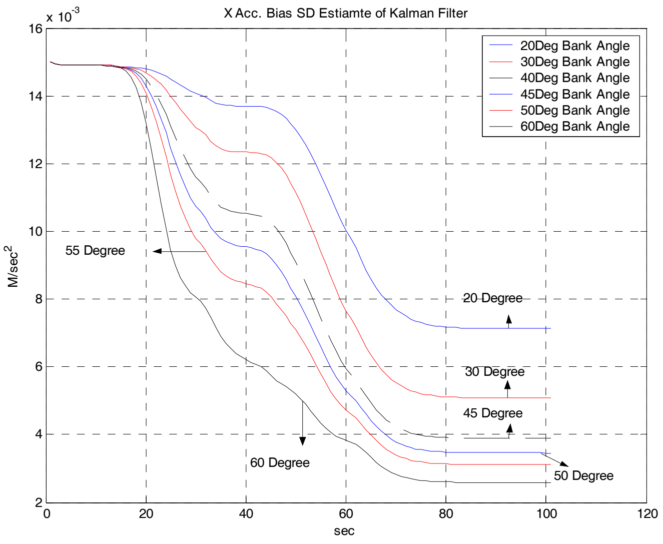


Figure 5-25 : X axes accelerometer bias standard deviation estimates of Kalman filter under maneuvers with different bank angles

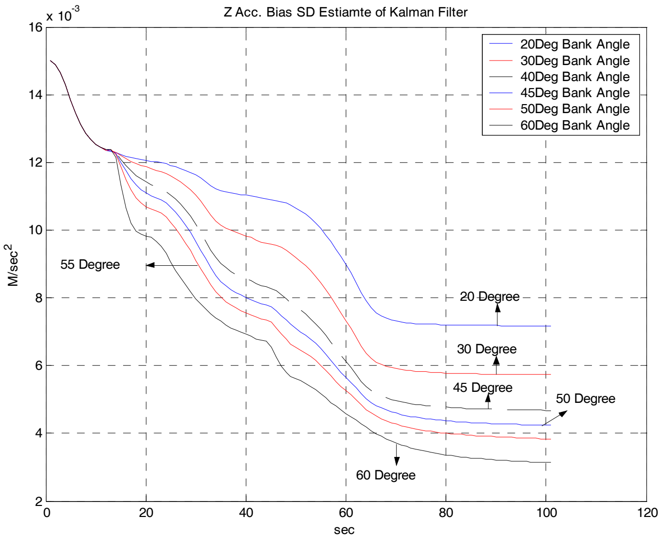


Figure 5-26 : Z axes accelerometer bias standard deviation estimates of Kalman filter under maneuvers with different bank angles

As seen from the above figures, under higher bank angles, Kalman filter can calculate better estimates. However as indicated before, maximum bank angle is usually limited by the aircraft and real time operational constraints. Furthermore, the above results were obtained with the assumption that, the vibration induced on the aircraft does not change with the change in bank angle. However, during real time operation it is expected that any increase in the bank angle also cause the vibration level to increase and this vibration will certainly decrease the overall estimation performance. Therefore, in the remaining part of this study, 45 degree was used as the maximum bank angle during coordinated turns.

Considering the above results, it is inferred that a suitable maneuver should be composed of 2 basic motions. The first motion is level and straight flight and the second one is bank-to-turn maneuver with 45 degree bank angle. Therefore, constructing an optimal maneuver definition problem reduces to determining:

i. How many bank-to-turn and level flight segment should be used

ii. In what order they should be used

iii. What the duration of each segment should be.

In order to find a satisfactory answer to above questions, the estimation performance of Kalman Filter which uses velocity match technique with 4 different maneuvers were compared. The roll angle profiles of each of the maneuvers are presented in the following figures.

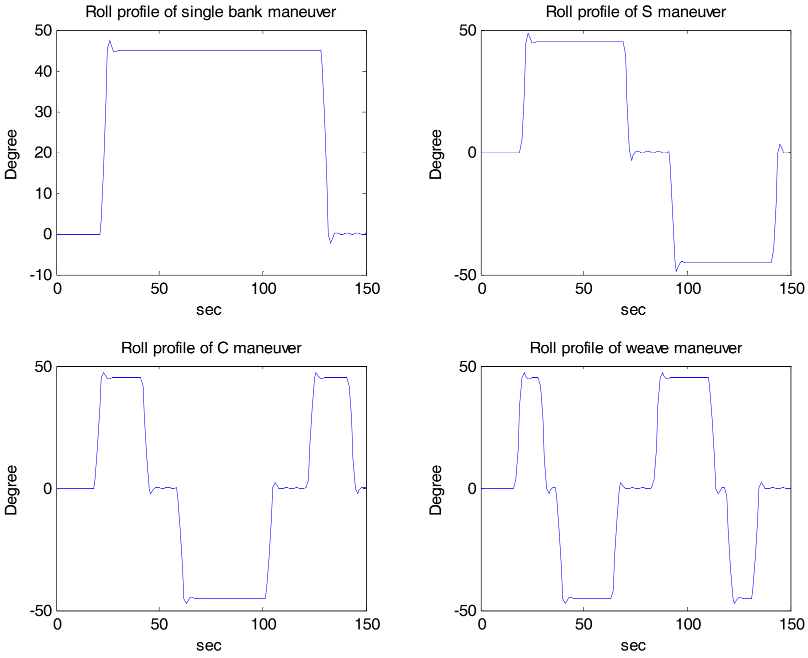


Figure 5-27 : Roll angle profiles for different maneuvers used to assess Kalman filter performance

As seen from the above figures, the successive bank angles were adjusted such that each has an opposite sign. This is because for two reasons. First of all, covariance analysis showed that instead of repeating same bank angle, changing the signs improves the estimation performance. Secondly, alternating bank angle sign decrease the total heading change which is preferable for the real time operations.

The names of the maneuvers come from the similarity of latitude – longitude profiles of the aircraft to the corresponding letter. This situation is represented in the following figure. In this figure, the latitude – longitude profiles of the aircraft which performs the associated maneuver are shown.

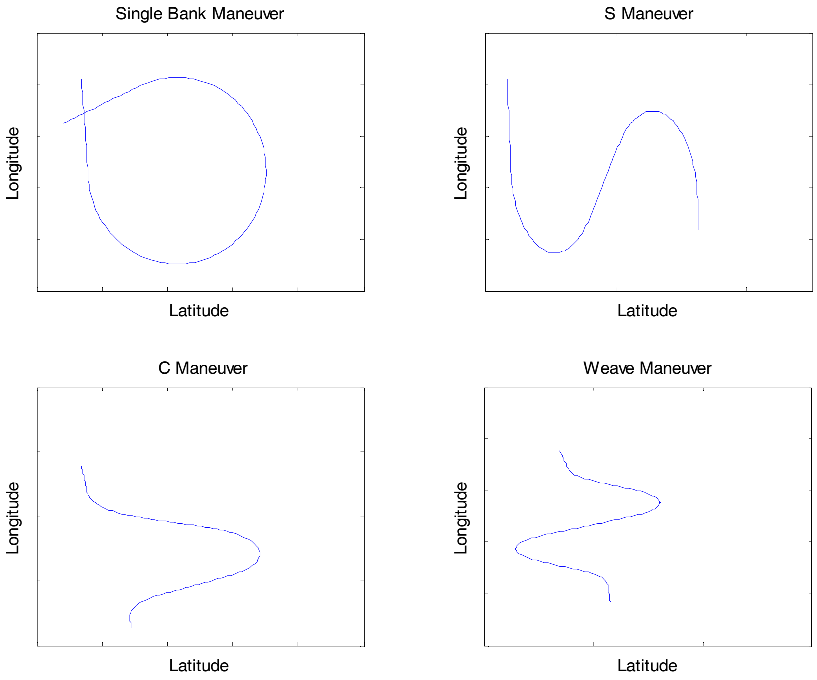


Figure 5-28 : Latitude – Longitude profile of the aircraft which performs the corresponding maneuver

In the following figures, standard deviation estimates of a Kalman filter which uses velocity match technique under the above maneuvers are presented. In each figure, a reference value for corresponding states is also shown. That reference value was obtained by running transfer alignment algorithm for 2500 seconds under a maneuver which consists of successive 200 seconds bank-to-turn maneuvers and 200 seconds level flight started with 200 seconds level flight segment (in order to be able to show the reference values in the same figures, the time axes of reference value is scaled by 10). The reference represents a practical limit where the transfer alignment algorithms should converge in a limited 150 second interval. Therefore, in a given period of time, if any state approaches that limit, then it is inferred that corresponding maneuver is sufficient for estimating the error states and maneuver definitions are revised by just considering the error states that cannot approach the limit.

Although maneuvers shown in Figure 5-27 are completed in 150 seconds, simulations are performed for 250 second duration. In the last 100 seconds, aircraft also continues to perform level and straight flight. This is because, it is expected that, after the transfer alignment maneuvers completed, the aircraft continues to perform mostly level flight until it arrives the release point. Therefore, together with the transfer alignment duration, it would be more realistic to consider the effect of post alignment duration also in the design of maneuvers.

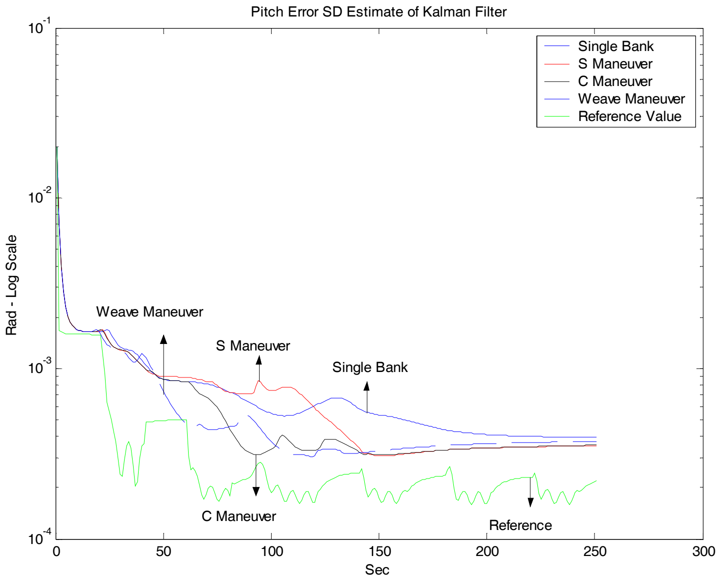


Figure 5-29 : Pitch error standard deviation estimate

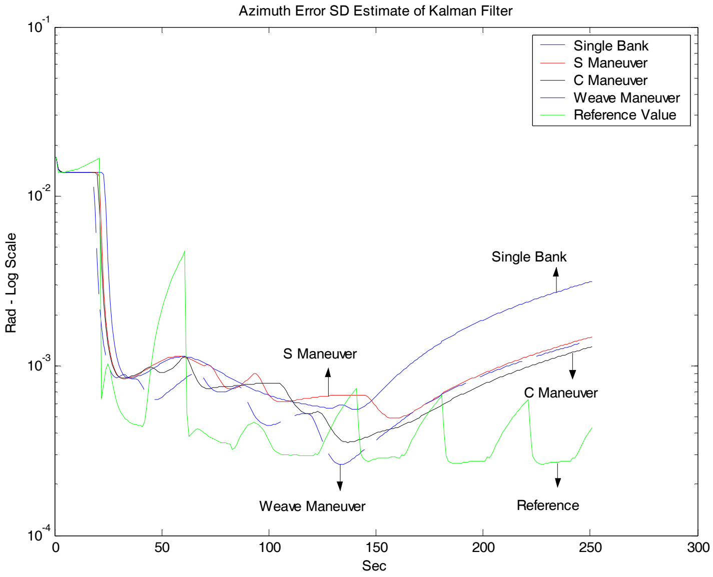


Figure 5-30 : Azimuth error standard deviation estimates

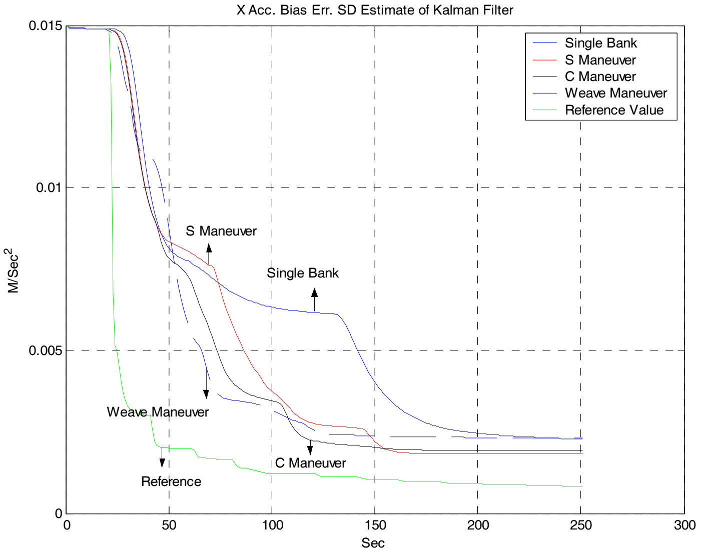


Figure 5-31 : X accelerometer bias repeatability error standard deviation estimate

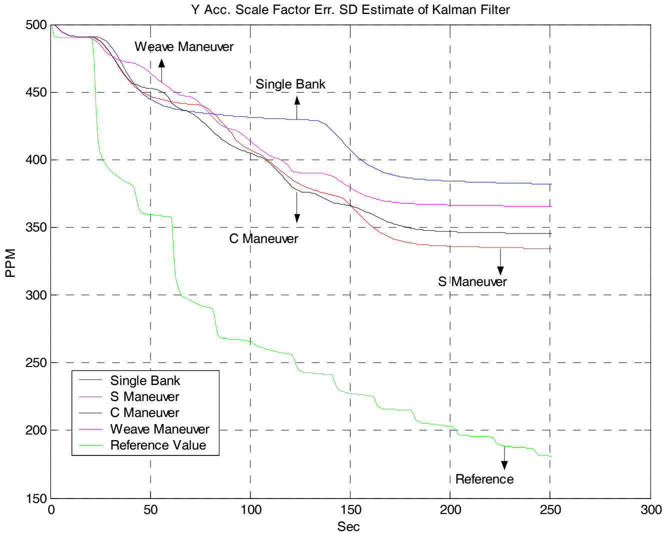


Figure 5-32 : Y accelerometer scale factor repeatability error standard deviation estimate

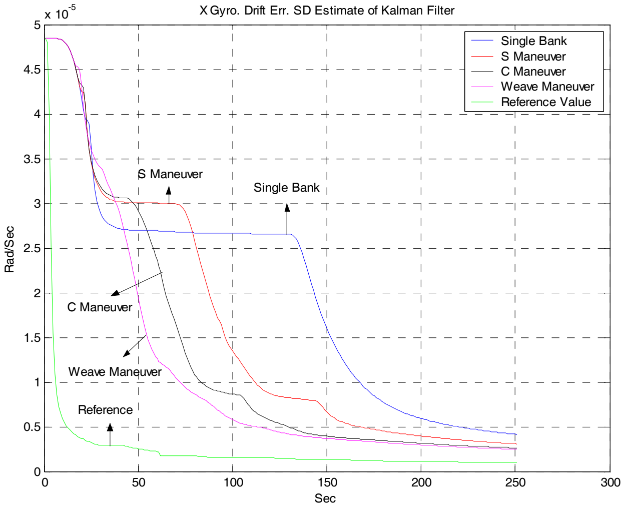


Figure 5-33 : X gyroscope drift repeatability error standard deviation estimate

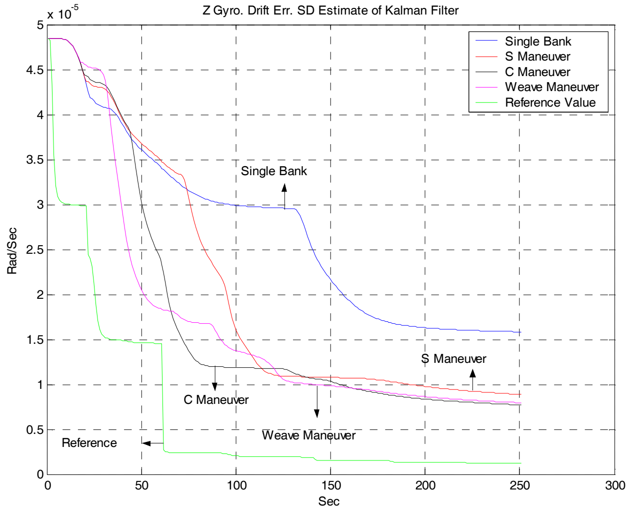


Figure 5-34 : Z axes gyroscope drift repeatability error standard deviation estimate

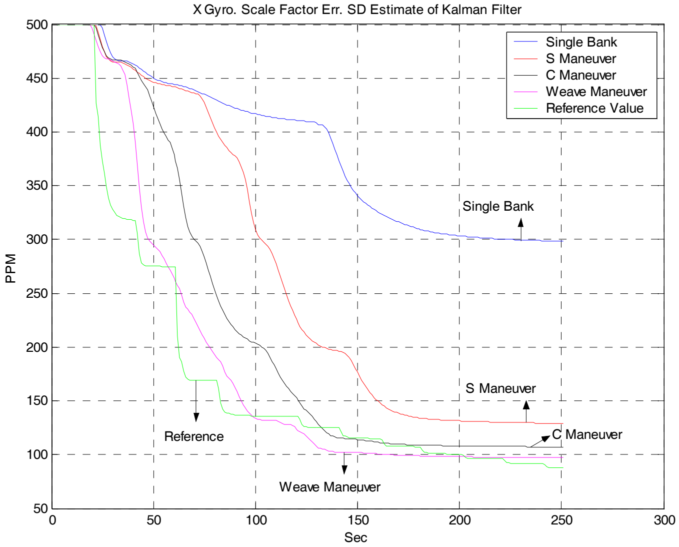


Figure 5-35 : X axes gyroscope scale factor error standard deviation estimate

As seen from the above figures “C” and “S” maneuvers turn out to be the best maneuver in terms of improving Kalman filter estimates. Among them “C” maneuver seems to be better than “S” especially in estimating gyroscope calibration parameter estimates. However, in terms of accelerometer calibration parameter estimates, they show similar performances. Furthermore, both “S” and “C” type maneuvers enhance the accelerometer calibration parameters estimates better than “Weave” type maneuvers which consist of more bank-to-turn motions. This reveals the fact that, performing unnecessary amount of bank-to-turn maneuvers successively can even reduce the total estimation performance of a Kalman filter.

In Chapter 5, it is shown that during the level flight segment (after the transfer alignment maneuvers), the azimuth error increases rapidly due to the residual gyroscope errors on “z” axes (refer to Figure 4-47). However, it should be noted that, this azimuth error can easily be estimated with any kind of bank-to-turn maneuvers. This situation can also be seen in Figure 5-30. At that figure, with the start of first bank-to-turn maneuver the azimuth angle is rapidly estimated. So, even if azimuth error gets bigger when the aircraft approaches to the release point, that error can be quickly estimated with a simple and rapid bank-to-turn maneuver prior to launch.

Therefore, it is concluded that, the optimum maneuver for transfer alignment procedure is composed of an “S” or “C” type maneuver which is performed to estimate IMU calibration parameters in friendly territory and a rapid heading change maneuver with no constraints to reestimate azimuth alignment error just before the launch of the slave system.

On the other hand, it should be noted that, for the transfer alignment point of view, the number of bank-to-turn and level flight segments during the maneuver is more important than the duration of each segment. Therefore, the total duration of transfer alignment maneuvers can be reduced with a little degradation in estimation performance. This is shown in the following figures. In these figures, the Kalman filter standard deviation estimates are presented for “C” maneuver which last for both 100 and 150 seconds. Simulations were again run for 250 seconds, in order to see the effect of post transfer alignment period on estimation performance.

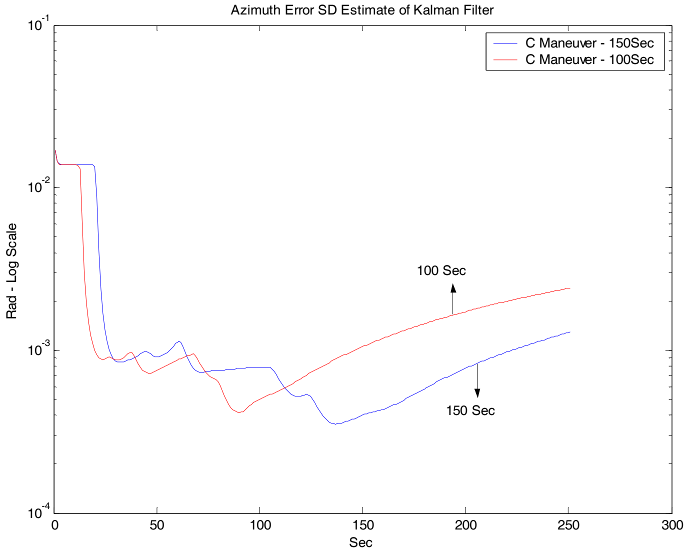


Figure 5-36 : Azimuth error standard deviation estimates

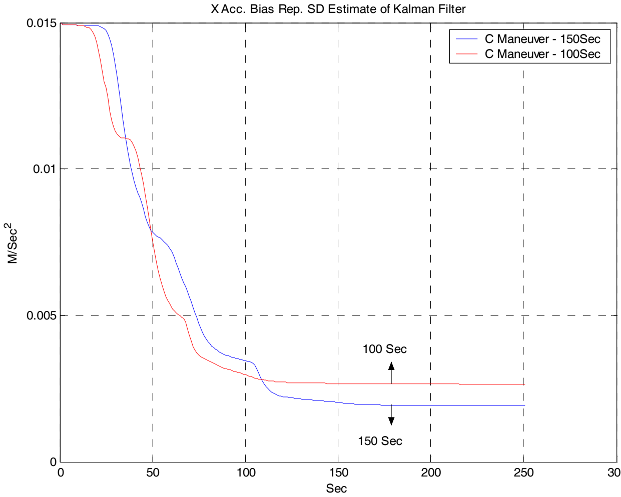


Figure 5-37 : X axes accelerometer bias error standard deviation estimate

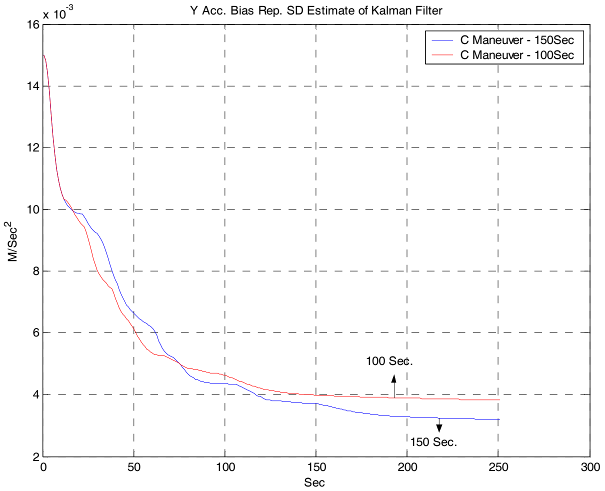


Figure 5-38 : Y axes accelerometer bias error standard deviation estimate

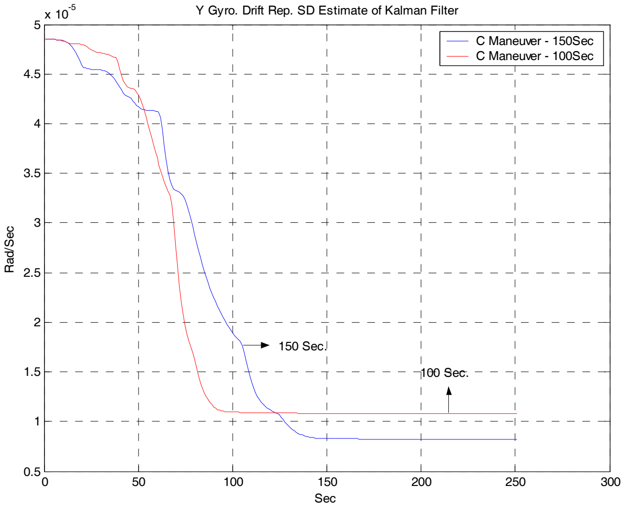


Figure 5-39 : Y Axes gyroscope drift error standard deviation estimate

As seen from the above figures, the total duration of transfer alignment maneuvers can be reduced without sacrificing Kalman filter estimation performance much. It should also be noted that, maximum bank angle is a more effective parameter than the total duration of maneuver. Therefore, it is also possible to reduce total alignment time without reducing estimation performance by increasing the total bank angle.

In the related literature, wing rock maneuver is also suggested as another maneuver that can be applied during transfer alignment procedures. However, since this maneuver is mainly used for rapid transfer alignment procedures where attitude measurement is also included in observation model, in this study it was not considered.

CHAPTER VII

CONCLUSIONS & FUTURE WORKS

The main objective of this study was to form a general framework in the design and analysis of transfer alignment algorithms. Although, many forms of transfer alignment procedures are extensively used especially in military aircrafts, a complete treatment of this problem has not been published openly before. In the literature, there are a number of papers that deal with the several aspects of this problem. However, none of the studies provides quantitative results that can be used as a basis for comparison with other methods. Moreover, most of the studies are far behind being complete and usually the information given in the published text is not enough to repeat the procedure to assess the performance of suggested algorithms.

Therefore, in this study, first of all, a simulation environment which can be used to assess estimation performance of different algorithms was developed. In order to implement a realistic simulation environment, the vibration models presented in Appendix B were utilized in the simulations. On the other hand, as the vibration models are not very accurate, these models are not augmented to the Kalman filter system model for any of the transfer alignment methods implemented in this study.

In this study five different transfer alignment algorithms were developed separately. By comparing estimation performances of these different algorithms, the advantages and disadvantages of each method were analyzed. Together with performance evaluations, the ease of each algorithm from the point of view of real time implementation was also presented. As far as the writer knows, this is the first study that such a comprehensive comparison is made.

For algorithms that use velocity based observations, a detailed analysis was also presented. The effect of update intervals and lever arm errors on algorithm performances was evaluated by simulations. The implementation of Schmidt-Kalman filters to reduce the effect of erroneous lever arm errors was also described.

The effect of maneuver on the performance of transfer alignment procedure was also investigated. Depending on the implementation constraints and previously published results, several candidate transfer alignment maneuvers were formed and their performances were evaluated using covariance simulations. The simulations verified that, an optimum maneuver for transfer alignment procedure should consist of both bank-to-turn and level flight segments. Furthermore, the number of bank-to-turns should not exceed 3 for 150 seconds transfer alignment duration.

The comparison between different transfer alignment methods showed that, it is not possible to achieve a single method which performs well in every condition. Every method implemented in this study has some unique properties which make it superior than others under some conditions. Therefore, before designing a transfer alignment algorithm, it is necessary to define the constraints and environmental conditions first, and then develop an algorithm that performs best at these predetermined conditions.

6.1 RECOMMENDED FUTURE WORKS

In the scope of this study, real time implementation problems which require implementation specific solutions were not analyzed. One of the most important of such problems is the data synchronization with master and slave systems. In digital systems, synchronization of two systems with separate clocks is always a problem. It is very likely that, the data send by the master navigator, reaches to the slave system with some latency. Furthermore, there is always an uncertainty about the amount of latency that these reference values have. Therefore, depending on the application, it may be necessary to model and include these data (measurement) latency effects in the Kalman filter structures (Itzhack (1985), Oshman, (2001)).

Another real time implementation problem is the error in the master navigator outputs. Usually, during the transfer alignment procedure, the master navigator solutions which are blended with aiding mechanism such as GPS or altimeter are used as reference values. However, it is very hard to model errors on such kind of blended outputs. Furthermore, due to the blending mechanism, it is also possible that some jumps at the master navigator output values occur from sample to sample which can totally destroy the error estimates of slave in an unpredicted way (Groves (2003), Grooves (2002)). Therefore, some special precautions have to be taken in order to prevent the use of such kind of erroneous reference values in Kalman filters.

Another important concept that is not considered in this study is the analysis of the effect of error models on transfer alignment performance. In this study, some error models were put forward first and all analysis were conducted using these models. However it is apparent that, the estimation performance of transfer alignment Kalman filter depends on the stochastic properties of these models. This is especially true for stability errors of IMU. Since these types of errors have flat spectral properties, estimation of these errors is harder than IMU repeatability errors, and usually puts a lower limit in the estimation of IMU calibration parameters. Therefore, it is crucial to analyze the dependence of transfer alignment Kalman filter estimation performance to the change of error characteristics.

In addition to this, it is also necessary to analyze the sensitivity of the Kalman filter to the inaccurate system models. As stated in previous chapters, the IMU errors and aircraft structure parameter characteristics are subject to change during real time operation. For instance, IMU bias errors can change depending on the total vibration that the system exposes to (refer to Chapter 3). With constant error models, it is not possible to handle such change in error characteristics. Therefore, if Kalman filter is sensitive to these kinds of modeling errors, it is possible that the filter can diverge during real time operation. In this study, only some simple Monte Carlo analysis were performed to observe the sensitivity of Kalman filter to the aircraft structure models such as lever arm and relative orientation states (refer to Chapter 4 & 5). However, in order to avoid any divergence problem, it is necessary to conduct more systematic sensitivity analysis on all error models. Furthermore, as an alternative to the method presented in this thesis, it is also suggested implementing and evaluating the effectiveness of Kalman filters based on multiple models. It is expected that, multiple model Kalman filter will be less sensitive to inaccurate models.

Throughout this study, comments regarding to the system operation are basically formed by observations. The relationships between reason and results are inferred by performing several simulations and observing the relationship between the results. Therefore, most of the conclusions presented in this study are lack of solid theoretical and analytical reasoning. For instance, the superiority of integrated velocity match technique over velocity match technique is verified by using a highly simplified system definition. However, such an approach is incapable of explaining under what conditions integration of measurements are better than using raw measurements. In order to specify such conditions, it is necessary to analytically analyze the effect of measurement integration on sub-optimal Kalman filter structures.

Another point that needs a theoretical analysis is the dependence of Kalman filter estimation performance on the aircraft maneuvers. In this study, this effect was examined by performing some covariance analysis. However, theoretical reasoning can provide more insight about the aircraft maneuver and state estimation relation.

In the related literature, aircraft maneuver and state estimation relation is usually described by using observability properties of the system model (Itzhack (1980-a), Porat (1981), Itzhack (1988), Meskin (1991), (Rehee (2004)). However, error model of a navigation system is a time varying system. Therefore, unless the observability is defined over some specified duration, the results of such an observability analysis become totally useless due to the time varying nature of the system and random system disturbances. In other words, it is not useful to estimate some initial states which are not correlated with the current state values. Hence, in order to define a suitable observability analysis period, some further studies are required.

REFERENCES

Bar-Itzhack I. Y., Porat B., Azimuth Observability Enhancement during Inertial Navigation System In-Flight Alignment, Journal of Guidance and Control, Vol. 3, No. 4, 1980

Bar-Itzhack I. Y., Vitek Y., The Enigma of False Bias Detection in a Strapdown System During Transfer Alignment, Journal of Guidance and Control, Vol.8, No. 2, 1985

Bar-Itzhack, Berman N., Control Theoretic Approach to Inertial Navigation Systems, Journal of Guidance and Control, Vol.11, No. 3, 1988

Baziw J., Leondes C. T., In Flight Alignment and Calibration of Inertial Measurement Units – Part I General Formulations, IEEE Transactions on Aerospace and Electronic Systems, Vol AES-8, No:4, 1972

Baziw J., Leondes C. T., In Flight Alignment and Calibration of Inertial Measurement Units – Part II Experimental Results, IEEE Transactions on Aerospace and Electronic Systems, Vol AES-8, No:4, 1972

Boch W., Krogmann U., Fast Transfer Alignment for Air-Launched Missile INS, CP 411

Bortz J. E.,, A New Mathematical Formulation For Strapdown Inertial Navigation, IEEE Transactions on Aerospace and Electronic Systems, Vol:AES-7, No:1, 1977

Brown G. B., Hwang P. Y. C., Introduction to Random Signals and Applied Kalman Filtering, John Willey & Sons, 1997

Goshen-Meskin D., Bar-Itzhack I. Y., Observability Analysis of Piecewise Constant Systems – Part I : Theory, IEEE Transactions on Aerospace and Electronic Systems, Vol 28, No:4, 1992

Goshen-Meskin D., Bar-Itzhack I. Y., Observability Analysis of Piecewise Constant Systems – Part II : Application to Inertial Navigation In-Flight Alignment, IEEE Transactions on Aerospace and Electronic Systems, Vol 28, No:4, 1992

Groves P. D., Wilson G. G., Mather C. J., Robust Rapid Transfer Alignment with an INS/GPS Reference, Institute of Navigation National Technical Meeting, 2002

Groves P. D., Optimising The Transfer Alignment of Weapon INS, The journal of Navigation, 2003

Hallingstad O., Design of a Kalman Filter For Transfer Alignment, AGARD Lecture Series No. 166 - Kalman Filter Integration of Modern Guidance and Navigation Systems, 1989

Harris R. A., Wakefield C. D., Coordinate Alignment for Elastic Bodies, NEACON 77 Record, pp665-670

Haykin S., Adaptive Signal Processing, Prentice Hall, 2002

Ignagni M. B., On the Orientation of Vector Differential Equation in Strapdown Inertial Systems, IEEE Transaction on Aerospace and Electronic Systems, Vol:30, No:4, 1994

Ignagni M. B., Optimal Strapdown Attitude Integration Algorithms, Journal of Guidance Control and Dynamics, Vol:13, No:2, 1990

Ignagni M. B., Efficient Class of Optimized Coning Compensation Algorithms, Journal of Guidance Control and Dynamics, Vol:19, No:2, 1996

Junkins J. L., Schaub H., Analytical Mechanics of Space Systems, AIAA Education Series, 2003.

Kain J. E., Cloutier J. R., Rapid Transfer Alignment for Tactical Weapon Applications, Proceedings of AIAA Guidance, Navigation and Control Conference, pp1290-1300, 1989

Klotz H. A., Derbak C. B., GPS-Aided Navigation and Unaided Navigation on the Joint Direct Attack Munition, IEEE 1998 Position Location and Navigation Symposium, pp412-419, 1998

Koyaz P., A System Theoretical Approach to Aided Inertial Navigation, METU Thesis, 2003

Musick S. H., PROFGEN : A Computer Program to Generate Flight Profile, AFAL–TR–76–247, Reference System Branch, Reconnaissance and Weapon Delivery Division, 1976

Oshman Y., Rubanenko A., State Estimation Using Measurements with Uncertain Time-Tag, AIAA Guidance Navigation & Control Conference, 2001

Porat B., Bar-Itzhack I. Y., Effect of Acceleration Switching During INS In-Flight Alignment, Journal of Guidance and Control, Vol.4, No. 4, 1981

Reiner J., Method For Airborne Transfer Alignment of an Inertial Measurement Unit, United States Patent, Patent No.:5948045, 1999

Rhee I., Abdel-Hafez M. F., Speyer J. L., Observability of an Integrated GPS/INS During Maneuvers, IEEE Transactions on Aerospace and Electronics Systems, Vol:40, No:2, 2004

Rogers R. M., Velocity-Plus-Rate Matching for Improved Tactical Weapon Rapid Transfer Alignment, Proceedings of AIAA Guidance, Navigation and Control Conference, 1991, pp1580-1588

Rogers R. M., Weapon IMU Transfer Alignment Using Aircraft Position from Actual Flight Tests, IEEE PLANS, pp328-335, 1996

Roscoe, K. M., Equivalency Between Strapdown Inertial Navigation Conning and Sculling Integrals/Algorithms, Journal of Guidance Contral and Dynamics, Vol:24, No:2, 2001

Ross C. C., Elbert T. F., A Transfer Alignment Algorithm Study Based on Actual Flight Test Data From a Tactical Air-to-Ground Weapon Launch, IEEE PLANS, pp431-438, 1994

Savage P. G., Strapdown System Algorithms, Advances in Strapdown Inertial Systems, NATO AGARD Lecture Series, No:133, 1984

Savage, P. G., Strapdown Analytics Part I, Strapdown Associates, 2000

Savage, P. G., Strapdown Analytics Part II, Strapdown Associates, 2000

Schmidt S. F., “Application of State Space Methods to Navigation Problems”, In C. T. Leondes (ed.), Advances in Control Systems, Vol. 3., New York: Academic Press, 1966

Setterlend R. H., A Novel Method for In-Flight Alignment of Master-Slave Inertial Platforms, MIT Theses, 1972

Shortelle K. J., Graham W. R., F-16 Flight Tests of a Rapid Transfer Alignment Procedure, IEEE PLANS 1998, pp379-386, 1998

Spalding K., An Efficient Rapid Transfer Alignment Filter, Proceedings of AIAA Guidance, Navigation and Control Conference, pp1276-1286, 1992

Stovall S. H., Effect of Angular Vibration on Transfer Alignment, Naval Air Warfare Center Weapons Division, China Lake, 1993.

Stovall S. H., Transfer Alignment, Naval Air Warfare Center - Weapons Division, China Lake, 1996.

Titterton D.H. and Weston J.L., “Strapdown Inertial Navigation Technology”, Peter Pergrinus Ltd., London, 1997

APPENDIX A

SUMMARY OF CONING AND SCULLING ALGORITHMS

In this appendix, the basic motivation behind the utilization of coning and sculling algorithms and implementation of these algorithms to strapdown inertial navigators are shown. Furthermore, error calculations of coning and sculling algorithms are also derived at the end of this appendix.

On the other hand, it should be noted that, the results of algorithmic error derivations for coning and sculling algorithms were not tested using a real test setup at all and hence, the assumptions made in the derivations were not verified. Therefore, it is possible that, there may be some difference between real and computed results.

A.1 CONING ALGORITHM

In a conventional strapdown inertial navigation system, the direction cosine matrix is updated using the following method:

$$\dot{C}_X^Y = C_X^Y S\left(\omega_{YX}^X\right) \qquad \text{(A-1)}$$

where C_X^Y represents the transformation matrix from X frame to Y frame and ω_{YX}^X represents the rotation rate of X with respect Y and defined in X. (For notational simplification "ω_{YX}^X" will be denoted simply as "ω" for the rest of this appendix). In this method Equation (A-1) is solved as follows:

$$C_X^Y\left(t\right) = C_X^Y\left(t_n\right) e^{\int_{t_n}^{t} S[\omega(\tau)]d\tau} \qquad \text{(A-2)}$$

By reformulating the exponential term using the Taylor series expansion and skew-symmetric property of integrand, the above equation can be solved In discrete time using the angle increments supplied by gyroscopes as follows (Titterton (1997)):

$$C_X^Y(t)=C_X^Y(t)\left\{\mathrm{I}+\frac{\sin(\sigma_\omega)}{\sigma_\omega}S(\alpha(t_n,t))+\frac{1-\cos(\sigma_\omega)}{\sigma_\omega}S^2(\alpha(t_n,t))\right\} \qquad \text{(A-3)}$$

where;

"$\alpha(t_n,t)=\int_{t_n}^{t}\omega(\tau)d\tau$" is the gyroscope angle increment output between t and t+ΔT and "$\sigma_\omega^2=\left(\alpha_x^2+\alpha_y^2+\alpha_z^2\right)$" the magnitude of total rotation during that period. It should be noted that, the term inside the parenthesis corresponds to a transformation matrix which relates the orientation of x from t to t_n which can be represented as "$C(t_n,t)$". Therefore, given that "$C_X^Y(t_n)$" is given, the calculation of "$C_X^Y(t)$" reduces to the calculation of "$C(t_n,t)$".

On the other hand, in the above approach, the change of direction of the coordinate system where the rotation rates are defined is totally ignored. In Equation (A-2), it is implicitly assumed that the rotation rate "ω" is defined with respect to a constant coordinate frame which is not true as the orientation of X changes during motion. As "$\Delta T=t-t_n$" gets smaller, the change of orientation of X frame become negligible and thus the accuracy of this approach increases. However, computing Equation (A-3) with sufficiently high frequency is usually not possible and because of this the well known phenomenon of non-commutativity of finite rotations occurs (Bortz (1971)).

In order to avoid this effect, the attitude information is updated using the rotation vector concept. "Rotation vector" defines an axis of rotation and a magnitude for that rotation about the rotation vector itself (Savage (2000)). As the rotation vector is the eigenvector of corresponding direction cosine matrix with an eigenvalue of 1, its components on the coordinate system axes are same for both base and transformed coordinate systems. Therefore, if the differential equation governing the change of rotation vector with respect to time can be derived, then that differential equation can be solved without considering the rotation of base coordinate system.

In the literature this concept was first published in Bortz (1971). In the corresponding paper, by using some geometrical reasoning, the differential equation governing the rotation vector was derived and the relation between the direction cosine matrix and rotation vector was calculated. In later studies such as Savage (1984) and Ignagni (1994) these relations were also derived using several other approaches. According to above studies, the differential equation of rotation vector that defines the orientation of X at time “t” with respect to time “t$_n$” can be shown as follows:

$$\dot{\phi}(t_n,t)=\omega(t_n,t)+\delta\dot{\alpha}(t_n,t) \qquad \text{(A-4)}$$

where “$\phi(t_n,t)$” represents the rotation vector which defines the rotation of X at time “t” with respect to time “t$_n$”, “$\omega(t_n,t)$” represents the inertialy measurable rotation rate of X (for instance, gyroscope rotation rate outputs) and “$\delta\dot{\alpha}(t_n,t)$” is called as the non-inertially measurable angular motion which can be calculated as follows (Savage (1984)):

$$\delta\dot{\alpha}(t_n,t)=\frac{1}{2}\alpha(t_n,t)\times\omega(t)+\frac{1}{\sigma_\omega^2(t)}\left(1-\frac{\sigma_\omega(t)\sin(\sigma_\omega(t))}{1-\sin(\sigma_\omega(t))}\right)\alpha(t_n,t)\times(\alpha(t_n,t)\times\omega(t))$$

(A-5)

In practical implementations, the last term in right hand side of Eq (A-5) is usually ignored. Hence, rotation vector “$\phi(t)$” can be calculated as follows:

$$\phi(t_n,t)=\int_{t_n}^{t}\omega(\tau)d\tau+\int_{t_n}^{t}\left(\frac{1}{2}\alpha(t_n,\tau)\times\omega(\tau)\right)d\tau \qquad \text{(A-6)}$$

Once the rotation vector is calculated using (A-6), then the corresponding direction cosine matrix can be calculated as follows (Savage (1984)):

$$C(t_n,t)=\mathrm{I}+\frac{\sin(\sigma_\phi)}{\sigma_\phi}S(\phi(t_n,t))+\frac{1-\cos(\sigma_\phi)}{\sigma_\phi}S^2(\phi(t_n,t)) \qquad \text{(A-7)}$$

where $\sigma_\phi^2=\left(\phi_x^2+\phi_y^2+\phi_z^2\right)$.

As seen from Equation (A-6) and (A-7), if the rotation rate and its integral stays in the same direction (such as in the case of rotation rate along a single axes), then Equation (A.6) becomes direct integral of rotation rate and thus Equation (A-7) becomes equivalent to exponential term in Equation (A-2). Therefore the effect of non-commutative rate vector appears if it least 2 rotation rate which are out of phase is applied on two different axes. Such a motion can be represented as follows:

$$\omega=a\Omega\cos(\Omega t)\vec{i}+b\Omega\sin(\Omega t)\vec{j} \qquad \text{(A-8)}$$

Above motion definition is called as pure conning motion. Therefore the last term in Equation (A-6), which is also shown below, is also called as conning correction.

$$\delta\alpha(t_n, t) = \int_{t_n}^{t} \left(\frac{1}{2} \alpha(t_n, \tau) \times \omega(\tau) \right) d\tau \tag{A-9}$$

The error which occurs due to not accounting this effect is called as conning error.

The importance of this effect comes from the fact that, in vibration environment every axes of inertial measurement unit sense a random rotational vibration. These random vibrations inevitably cause motions such as defined in Equation (A-8) to occur. Therefore, in vibration environment calculating conning effect become highly crucial.

Several different digital algorithms which can be utilized to solve equation (A-9) using gyroscope's angle increment outputs in discrete time were published in the past. Some of them are Ignagni (1990), Ignagni (1996), Savage (1984) Savage (2000). In the literature, these algorithms are called as coning compensation algorithms or simply coning algorithms.

In the corresponding papers, the coning algorithms are derived using a simple optimization procedure. The basic method in deriving a coning algorithm can be summarized as follows:

First of all, under the motion definition given in Equation (A-8), the exact analytical solution of equation (A-9) is solved for "$\Delta T = t - t_n$" period. As shown in Ignagni (1996), this is equal to:

$$\delta\alpha(t_n, t_n + \Delta T) = \frac{ab}{2} \Omega \left(\Delta T - \frac{\sin(\Omega \Delta T)}{\Omega} \right) \vec{k} \tag{A-10}$$

It should be noted that, if any coning compensation term were not included in the computations, then the effect of Equation (A-10) would be totally ignored, and the result would be highly detrimental for large amplitude high frequency oscillations. For instance suppose that at each axes, the amplitude of rotational motion is 1mrad with a frequency of 50Hz. If the attitude is computed at 600Hz, then a net drift of 1.4deg/hour occurs. (In other words, under the rotation rate given in Equation (A-8), the system rotates about z axis but without coning compensation, INS algorithm can not track this rotation accurately).

After calculating the exact value, the aim is to find an approximation to this exact value by using limited number of gyroscope outputs. To do so, it is assumed that a fixed number of (call it as m) gyroscope outputs in the form of angle increment can be obtained within ΔT period. Using these "m" outputs, a general form to calculate coning correction is established with undetermined coefficients. In different studies, different basic forms were utilized. In this study, it is assumed that in a single interval, 3 successive gyroscope outputs can be obtained and hence the following structure which is published in Ignagni (1996) is used:

Lets $\alpha_n(k)$ represents the kth angle increment among 3 outputs in nth interval. Then the form of coning correction term between $[t_n, t_n + \Delta T]$ can be represented as follows:

$$\delta\alpha(t_n, t_n + \Delta T) = [k_1\alpha_{n-1}(2) + k_2\alpha_{n-1}(3) + k_3\alpha_n(1) + k_4\alpha_n(2)] \times \alpha_n(3) \quad \text{(A-11)}$$

Finally, an unconstrained optimization algorithm with respect to coefficients is run to minimize the difference between the exact analytical solution shown in (A-10) and the Equation (A-11).

In ΔT period, the difference between (A-10) and (A-11) can be shown as follows:

$$err_{coning}^{Alg} = ab\begin{Bmatrix} (-k_1)\sin(5\Omega\delta t) - (2k_1 - k_2)\sin(4\Omega\delta t) \\ +(2k_2 - k_1 - k_3 + 0.5)\sin(3\Omega\delta t) \\ -(2k_3 - k_2 - k_4)\sin(2\Omega\delta t) \\ +(2k_4 - k_3)\sin(\Omega\delta t) - 1.5\Omega\delta t \end{Bmatrix} \quad \text{(A-12)}$$

where δt represents the time between 2 successive gyroscope outputs, and Ω represents the coning frequency given in (A-8).

In Ignagni (1996), the coefficients that minimize (A-12) were calculated to be as follows:

$$k_1 = -\frac{1}{420}, \quad k_2 = \frac{1}{40}, \quad k_3 = \frac{157}{420}, \quad k_4 = \frac{1207}{840} \quad \text{(A-13)}$$

In the literature, several other algorithms each of which uses different basic forms and different number of gyroscope outputs were published. Each of the derived algorithms has different error characteristics under coning and benign environments.

The extension of algorithm (A-11) to span more than one interval is also straightforward. Suppose that, within 2ΔT period a total of 6 successive gyroscope outputs can be obtained in the form of angle increments. Then Equation (A-9) can be rewritten as:

$$\delta\alpha(t_n, t_n + 2\Delta T) = \frac{1}{2}\int_{t_n}^{t_n+\Delta T}(\alpha(t_n,\tau)\times\omega(\tau))d\tau + \frac{1}{2}\int_{t_n+\Delta T}^{t_n+2\Delta T}((\alpha(t_n, t_n+\Delta T) + \alpha(t_n+\Delta T,\tau))\times\omega(\tau))d\tau \tag{A-14}$$

By rearranging the above term, the following equation is obtained:

$$\delta\alpha(t_n, t_n + 2\Delta T) = \frac{1}{2}\alpha(t_n, t_n+\Delta T)\times\alpha(t_n+\Delta T, t_n+2\Delta T) + \frac{1}{2}\int_{t_n+\Delta T}^{t_n+2\Delta T}(\alpha(t_n+\Delta T,\tau)\times\omega(\tau))d\tau + \frac{1}{2}\int_{t_n}^{t_n+\Delta T}(\alpha(t_n,\tau)\times\omega(\tau))d\tau \tag{A-15}$$

The last two integrals in the right hand side of Equation (A-13) can be calculated by using the coning compensation algorithms shown in (A-11). As the first term can be calculated exactly, the total error in these calculation is just the twice of the error defined in (A-12). In Chapter 2, this form of coning compensation algorithm was used in attitude update equation.

A.2 SCULLING ALGORITHM

Sculling algorithms are used in velocity update equations in order to reduce the computational error arise from high frequency motion components. As seen from Equation (2.2), the velocity update equations involve the solution of the following integral term:

$$\Delta v(t_n, t) = \int_{t_n}^{t} C_B^N(\tau) f^B(\tau) d\tau = C_B^N(t_n)\int_{t_n}^{t} C(t_n,\tau) f^B(\tau) d\tau \tag{A-16}$$

For small "σ_ϕ" Equation (A-7) can be simplified as follows:

$$C(t_n, t) = \mathrm{I} + S(\phi(t_n, t)) \tag{A-17}$$

Replacing (A-17) in (A-16), following equation is obtained:

$$\Delta v(t_n, t) = C_B^N(t_n)\left\{\int_{t_n}^{t} f^B(\tau)d\tau + \int_{t_\tau}^{t} S(\phi(t_n, t)) f^B(\tau)d\tau\right\} \tag{A-18}$$

By directly applying integration by part rule to the last term in the parenthesis and rearranging the terms, Equation (A-18) can be rewritten as follows (Savage (1984)):

$$\Delta v(t_n, t) = C_B^N(t_n) u(t_n, t) + C_B^N(t_n)\{\Delta v_{Rot}(t_n, t) + \Delta v_{Scull}(t_n, t)\} \tag{A-19}$$

where

$$u(t_n, t) = \int_{t_n}^{t} f^B(\tau) d\tau \tag{A-20}$$

$$\Delta v_{Rot} = \frac{1}{2} \phi(t_n, t) \times u(t_n, t) \tag{A-21}$$

$$\Delta v_{Scull} = \frac{1}{2} \int_{t_n}^{t} [\phi(t_n, \tau) \times f(\tau) + u(t_n, t) \times \omega(\tau)] d\tau \tag{A-22}$$

Equation (A-21) and (A-22) are called as rotation compensation and sculling effects respectively. By using accelerometer and gyroscope increment outputs, Equation (A-20) and (A-21) can be computed easily. However, a digital algorithm which can compute Equation (A-21) in discrete time is required. In the literature, several methods for calculating Equation (A-21) were published (Savage (1984), Savage (2000)). In these methods, again a similar methodology used in calculating coning algorithms is utilized. However, in this case, the error is defined under the following motion definition:

$$\begin{aligned} \omega &= [\Omega a \cos(\Omega t)\ 0\ 0] \\ acc &= [0\ b \sin(\Omega t + \phi)\ 0] \end{aligned} \tag{A-23}$$

This motion definition is called as sculling motion, and therefore the algorithms that are used to compute Equation (A-21) is called as sculling algorithms. As seen from Equation (A-22), the effect of sculling term can be excited most if acceleration and a rotation rate which are out of phase are applied in different axes.

However in this study, the method proposed in Roscoe (2001) is implemented to derive a sculling algorithm. In Roscoe (2001), a general equivalency between coning and sculling algorithms is derived. This equivalency defined in corresponding paper can be summarized as follows:

Let A(x1) represent the following term:

$$A(x1) = \int_{t_n}^{t} \alpha_{x1}(t_n, \tau) \times x1(\tau) d\tau \tag{A-24}$$

where

$$\alpha_{x1}(t_n, t) = \int_{t_n}^{t} x1(\tau) d\tau \tag{A-25}$$

Then,

$$B = \int_{t_n}^{t} \alpha_{x1}(t_n, \tau) \times x2(\tau) + \alpha_{x2}(t_n, \tau) \times x1(\tau) \qquad \text{(A-26)}$$

can be calculated as follows:

$$B = A(x1 + x2) - A(x1) - A(x2) \qquad \text{(A-27)}$$

Therefore if any digital integral algorithm is computed for A(x1), then the same algorithm can also be used to compute B as well.

When the above idea is utilized for the coning and sculling algorithms, sculling algorithm can easily be developed by just using the digital algorithm developed for coning compensation. This can be shown as follows:

Equation (A-11) defines a digital algorithm to compute the (A-9). Let's call this algorithm as "$C(x)$" (Note that, $C(\omega)$ represents the coning compensation term). Then the sculling term which is represented in Eq (A-21) can be calculated as follows:

$$\Delta v_{Scull} = C(\omega + f) - C(\omega) - C(f) \qquad \text{(A-28)}$$

It should be noted that, although in Equation (A-28) the inputs to the algorithms are represented as rotation rate and acceleration, as shown in Equation (A-11), the real inputs are the increment type outputs of gyroscopes and accelerometers.

To find total algorithmic error under pure sculling environment defined in Equation (A-23), similar procedures used in calculating coning algorithm errors can be used. Under the motion defined by Equation (A-23), the exact value of sculling term given in Equation (A-22) can be found as follows:

$$\delta v_{Scull} = \frac{ab}{2}\left[\Delta T - \frac{\sin \Omega \Delta T}{\Omega}\right]\cos(\phi) \qquad \text{(A-29)}$$

Using the same motion definition, the difference between the implemented sculling algorithm shown in Equation (A-28) (which is based on coning algorithm given in Equation (A-11) and the exact solution during ΔT period is found to be as follows:

$$err_{scull}^{\mathrm{Alg}} = \frac{ab}{\Omega}\left\{\begin{array}{l} (-k_1)\sin(5\Omega\delta t) - (2k_1 - k_2)\sin(4\Omega\delta t) \\ +(2k_2 - k_1 - k_3 + 0.5)\sin(3\Omega\delta t) \\ -(2k_3 - k_2 - k_4)\sin(2\Omega\delta t) \\ +(2k_4 - k_3)\sin(\Omega\delta t) - 1.5\Omega\delta t \end{array}\right\} \qquad \text{(A-30)}$$

where δt represent the time between 2 successive gyroscope outputs, and Ω represents the coning frequency given in (A-28). The values of coefficients are the same as the coefficients derived for conning algorithm which are defined at Equation (A-13).

A.3 TOTAL ALGORITHM ERROR UNDER GENERAL MOTION

To find the total effect of complete vibration environment on INS algorithms and to evaluate the effectiveness of the implemented coning and sculling algorithms Equation (A-10), (A-12), (A-29) and (A-30) should be evaluated using the power spectral densities obtained using the flight test data (refer to Chapter 4). As shown in Savage (2000), a net bias effect can be observed only if two frequencies contributing to the environment are same. In other words, no conning/sculling error occurs as a result of rotations/accelerations acting on the system with different frequencies. Therefore, it will be enough to consider each frequency component in power spectral densities separately and then aggregate the total errors.

To evaluate the conning effect on a single axis, power spectral densities of rotation rates acting on different axes can be used. The total power on a single axis can be assumed to be generated from sum of sine functions each has a power equal to the power contained under the region of 1Hz band of corresponding spectra with the same starting frequency of that band.

Thus the rotation rate on any axes can be represented as follows:

$$\begin{aligned}\omega_x &= \ldots a_{n-1}2\pi(n-1)\sin[2\pi(n-1)t] + a_2 2\pi n\sin[2\pi n t] \\ &\quad + a_3 2\pi(n+1)\sin[2\pi(n+1)t]\ldots\end{aligned} \tag{A-31}$$

where each “a” is equal to

$$a_n = \frac{1}{2\pi n}\sqrt{4*\frac{1}{2\pi}\int_{2\pi(n-1)}^{s\pi n} S_{x\_gyro}(\omega)d\omega} \tag{A-32}$$

Using discrete test data (which has a sample frequency of Fs), each “a” in Equation (A-33) can also be computed as follows:

$$a_n = \frac{1}{2\pi n}\sqrt{4*\frac{PSD_{Fs}(n)}{Fs}} \tag{A-33}$$

where PSD_{Fs} represents power spectral density of corresponding rotation rate calculated using Fs point periodogram (same as MATLAB build in function PSD).

If the same process is applied to rotation rate sensed by another axis, and if it is assumed that the phase difference between rotation rates acting on the different axes are always 90°, then total coning error appear on the perpendicular axes (without any coning compensation algorithm) can be found as

$$err_{coning}^{Total} = \sum_n \frac{a_n b_n}{2} (2\pi n)\left(\Delta T - \frac{\sin((2\pi n)\Delta T)}{(2\pi n)} \right) \qquad \text{(A-34)}$$

where, “a” and “b” represents coefficients of sine functions used to represent rotation rates sensed by 2 different axes, and “ΔT” corresponds to compensation period.

The same approach can also be utilized to find total conning algorithm error. In this case, using Equation (A-12) algorithm error can be calculated as follows:

$$err_{conn}^{Total\_Alg} = \sum_n a_n b_n \{(-k_1)\sin(5(2\pi n)\delta t) - \cdots\cdots - 1.5(2\pi n)\delta t\} \qquad \text{(A-35)}$$

Similar calculations can also be utilized to find numerical values for sculling errors under the environment characterized by flight test results. To do so, rotation rate in one axes and acceleration in another axes should be used. Accelerometer output is also represented using Equation (A-31) and sine function coefficients for accelerometer outputs are calculated using the following equation:

$$b_n = 2\sqrt{\frac{1}{2\pi} \int_{2\pi(n-1)}^{s\pi n} S_{x\_acc}(\omega)d\omega}$$

In this case total and algorithmic errors can be calculated as follows:

$$err_{Scull}^{Total} = \sum_n \frac{a_n b_n}{2}\left(\Delta T - \frac{\sin((2\pi n)\Delta T)}{(2\pi n)} \right) \qquad \text{(A-36)}$$

$$err_{scull}^{Total\_Alg} = \sum_n \frac{a_n b_n}{(2\pi n)} \{(-k_1)\sin(5(2\pi n)\delta t) - \cdots\cdots - 1.5(2\pi n)\delta t\} \qquad \text{(A-37)}$$

where a_n and b_n represent sine function coefficients for accelerometer and rotation rates, δt represents IMU output period and ΔT represents the computation interval.

It should be noted that in deriving Equations (A-34), (A-35) and (A-36), (A-37) it is implicitly assumed that ΔT=3 δt. Also, if multiple computational intervals are used such as in Equation (A-15), then the errors are also multiplied with the ratio of total interval to computational interval.

APPENDIX B

VIBRATION MODELS

In this appendix, the vibration models that are presented in Yuksel (2005) are summarized.

In this study, the continuous time models vibration models utilized for each axes is converted to state space representation by using the following controllable canonical form:

Let

$$H(s) = \frac{\sqrt{N}s^2}{s^3 + As^2 + Bs + C} \tag{B.1}$$

and

$$a(t) = h(t) * n(t) \tag{B.2}$$

where n is a zero mean unity variance white Gaussian noise. Assume that, $\dot{v} = a$ and $\dot{d} = v$, then in the state space form, the system defined in Equation (B.2) can be represented as follows:

$$\begin{vmatrix} \dot{a} \\ \dot{v} \\ \dot{d} \end{vmatrix} = \begin{vmatrix} -A & -B & -C \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{vmatrix} \begin{vmatrix} a \\ v \\ d \end{vmatrix} + \begin{vmatrix} \sqrt{N} \\ 0 \\ 0 \end{vmatrix} n \tag{B.3}$$

Note that output is also equal to state "a". Therefore if it is assumed that "a" represents vibration induced acceleration of any system then the other two states directly represent the vibration induced velocity and displacement.

If (B.1) has only negative poles than the system defined in (B.3) reaches to steady state. At the steady state, the covariance matrix of states satisfies the following Lyapunov equation:

$$F\Sigma + \Sigma F^T + Q = 0 \tag{B.4}$$

where "F" represents system matrix of (B.3), Σ refers to covariance matrix of states and "Q" refers to power spectral density of "$\sqrt{N}n$".

All shaping filters used in vibration models have negative poles. Therefore, Equation (B.4) can be used to calculate the steady state variance of vibration induced acceleration velocity and displacement.

B.1 X AXES LINEAR VIBRATION MODEL

Number and form of shaping filter used

2 - $\mathrm{H}(s) = \dfrac{\sqrt{N}s^2}{s^3 + As^2 + Bs + C}$

First model coefficients

$\sqrt{N_1}$ = 6.1424 (m/s^2)

A_1 = 106.99

B_1 = 5225.4

C_1 = 16575

Lyapunov equation solution for first model (state order: acc, vel, disp)

0.18172	3.2765e-018	-3.4776e-005
3.2765e-018	3.4776e-005	-1.7051e-023
-3.4776e-005	-1.7051e-023	2.2446e-007

Second model coefficients

$\sqrt{N_2}$ = 5.5298 (m/s^2)

A_2 = 429.43

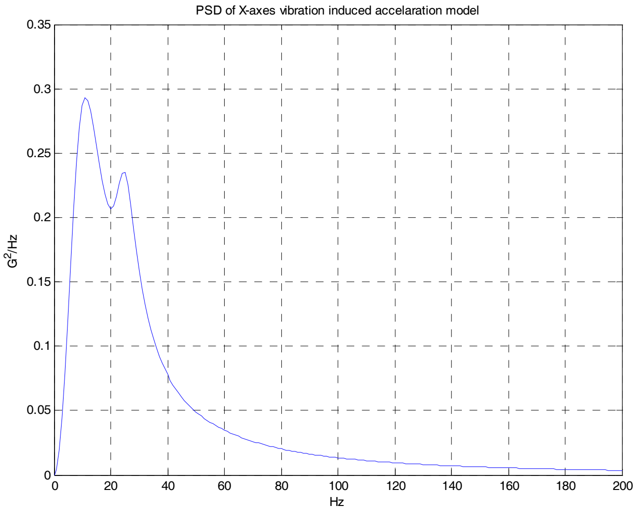
B_2 = 45218

C_2 = 8.8952e+006

Lyapunov equation solution for second model (state order: acc, vel, disp)

0.0657	1.0421e-019	-1.4529e-006
1.0421e-019	1.4529e-006	-3.4328e-023
-1.4529e-006	-3.4328e-023	7.0143e-011

PSD figure of vibration model between 0-200Hz



B.2 Y AXES LINEAR VIBRATION MODEL

Number and form of shaping filter used

$$2 - \mathrm{H}(s) = \frac{\sqrt{N}s^2}{s^3 + As^2 + Bs + C}$$

First model coefficients

$\sqrt{N_1}$ = 13.2483 (m/s^2)

A_1 = 134.59

B_1 = 3966.1

C_1 = 14077

Lyapunov equation solution for first model (state order: acc, vel, disp)

0.66971	-6.0532e-018	-0.00016886
-6.0532e-018	0.00016886	-2.2089e-021
-0.00016886	-2.2089e-021	1.6145e-006

Second model coefficients

$\sqrt{N_2}$ = 4.8894 (m/s^2)

A_2 = 242.31

B_2 = 57740

C_2 = 8.9125e+006

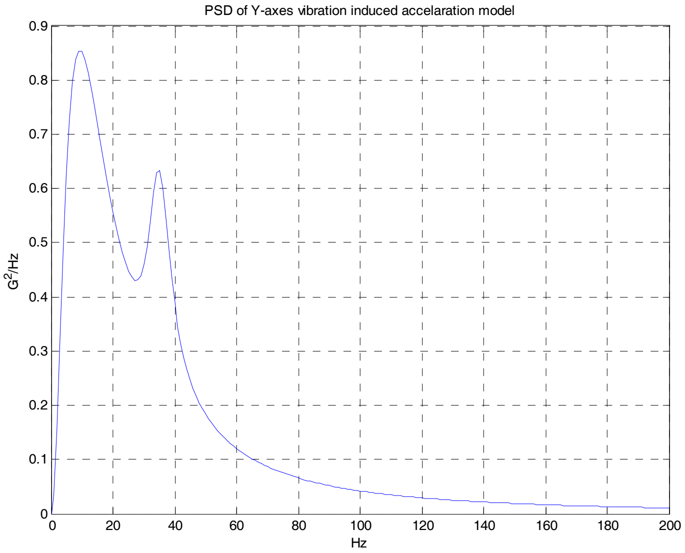
Lyapunov equation

0.13591	-2.7011e-019	-2.3538e-006

solution for second model (state order: acc, vel, disp)

-2.7011e-019	2.3538e-006	-1.0686e-021
-2.3538e-006	-1.0686e-021	6.3993e-011

PSD figure of vibration model between 0-200Hz



B.3 Z AXES LINEAR VIBRATION MODEL

Number and form of shaping filter used

$$2 - \mathrm{H}(s)=\frac{\sqrt{N}s^{2}}{s^{3}+As^{2}+Bs+C}$$

First model coefficients

$\sqrt{N_1} = 24.797$ (m/s^2)

$A_1 = 142.11$

$B_1 = 2255.9$

$C_1 = 9120.1$

Lyapunov equation solution for first model (state order: acc, vel, disp)

2.2268	1.4697e-017	-0.00098708
1.4697e-017	0.00098708	-6.5227e-020
-0.00098708	-6.5227e-020	1.5381e-005

Second model coefficients

$\sqrt{N_2} = 18.514$ (m/s^2)

A_2 = 305.01

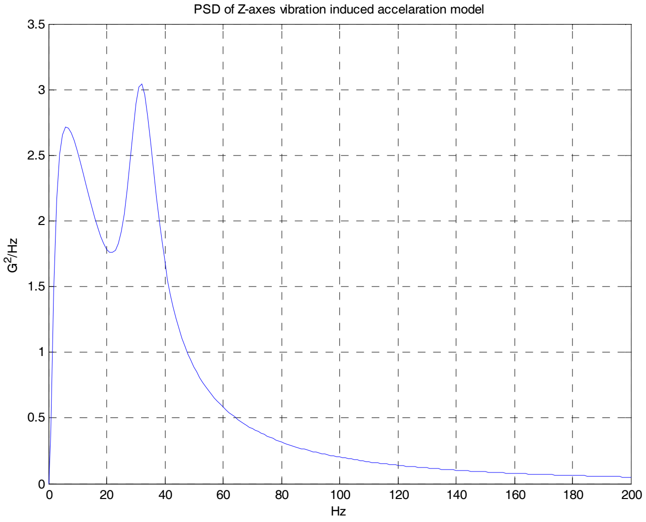
B_2 = 56302

C_2 = 8.9081e+006

Lyapunov equation solution for second model (state order: acc, vel, disp)

1.1675	-3.7892e-018	-2.0735e-005
-3.7892e-018	2.0735e-005	1.3235e-022
-2.0735e-005	1.3235e-022	7.0998e-010

PSD figure of vibration model between 0-200Hz



B.4 ROLL AXES ROTATIONAL VIBRATION MODEL

Number and form of shaping filter used

$$2 \text{ - } \mathrm{H}(s) = \frac{\sqrt{N}s}{s^2 + As + B}$$

First model coefficients

$\sqrt{N_1}$ = 0.05010646 (Rad/s)

A_1 = 38.662981

B_1 = 3818.1879

Lyapunov equation solution for first model (state order: Rot. Rate, Angle.)

3.24685017e-005	5.170525063e-023
5.17052506e-023	8.503641601e-009

Second model coefficients

$\sqrt{N_2}$ = 0.050368 (Rad/s)

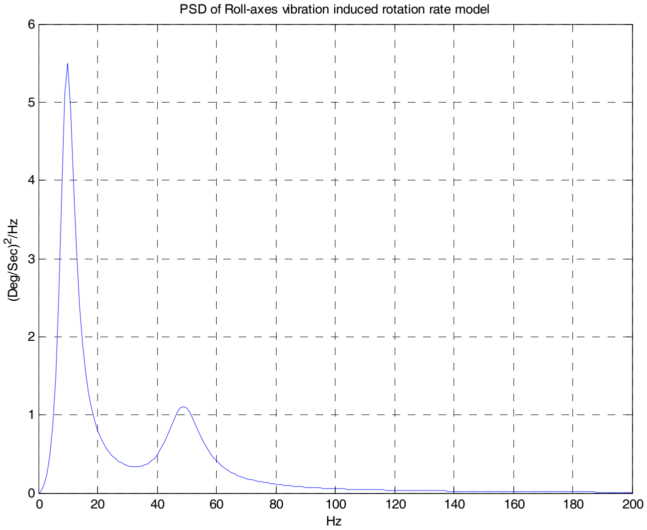
A_2 = 90.467983

B_2 = 95106.782

Lyapunov equation solution for second model (state order: Rot. Rate, Angle Inc.)

1.402157066e-005	-3.90782759e-024
-3.90782759e-024	1.474297655e-010

PSD figure of vibration model between 0-200Hz



B.5 PITCH & YAW AXES ROTATIONAL VIBRATION MODEL

Number and form of shaping filter used

3 - $H(s)=\frac{\sqrt{N}s}{s^2+As+B}$

First model coefficients

$\sqrt{N_1}$ = 0.020301594 (Rad/s)

A_1 = 30.5243766

B_1 = 2049.94463

Lyapunov equation solution for first model (state order: Rot. Rate, Angle

6.751239523e-006	2.0033280460e-025
2.003328046e-025	3.29337651e-009

Inc.)

Second model coefficients

$\sqrt{N_3} = 0.0105172$ (Rad/s)

$A_2 = 10.1528081$

$B_2 = 6816.42742$

Lyapunov equation solution for first model (state order: Rot. Rate, Angle Inc.)

5.447427403e-006	-9.819235677e-025
-9.819235677e-025	7.991616517e-010

Second model coefficients

$\sqrt{N_3} = 0.05214237$ (Rad/s)

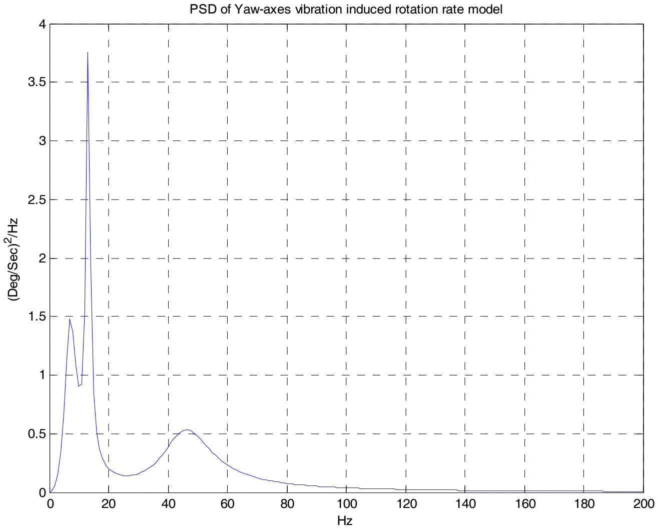
$A_3 = 132.1782843$

$B_3 = 84933.76592$

Lyapunov equation solution for second model (state order: Rot. Rate, Angle Inc.)

1.02846982932e-005	-6.007333565e-024
-6.0073335654e-024	1.2109080742e-010

PSD figure of vibration model between 0-200Hz



APPENDIX C

KALMAN FILTER STRUCTURE USED IN TRANSFER ALIGNMENT ALGORITHMS

Let the system model of the Kalman Filter is represented as follows:

$$\begin{aligned}
&\dot{x} = F(t)x + G(t)\omega(t) \\
&y = H(t)x + v(t) \\
&\mathrm{E}\left[xx^T\right] = P \\
&\mathrm{E}\left[vv^T\right] = R \\
&G(t)\mathrm{E}\left[\omega\omega^T\right]G(t)^T = q
\end{aligned} \tag{C-1}$$

The system given in Equation (C-1) is in continuous time. In order to implement a discrete time Kalman filter, the system model is discretized as follows:

$$\begin{aligned}
&\Delta T = t_{k+1} - t_k \\
&\Phi(t_{k+1}, t_k) \cong e^{F(t_k)(\Delta T)} \cong \mathrm{I} + F\Delta T + \frac{(F\Delta T)^2}{2!} + \frac{(F\Delta T)^3}{3!} \\
&n_k = \int_{t_k}^{t_{k+1}} \Phi(t_{k+1}, \tau)\omega(\tau)d\tau \\
&Q_k = \mathrm{E}\left[n_k n_k^T\right] = \int_{t_k}^{t_{k+1}} \Phi(t_{k+1}, \tau)q(\tau)\Phi(\tau, t_{k+1})d\tau \\
&\qquad \cong \Phi(t_{k+1}, t_k)q\Phi^T(t_{k+1}, t_k)\Delta T \\
&x_{k+1} = \Phi(t_{k+1}, t_k)x_k + n_k \\
&y_k = H(t_k)x_k + v_k \\
&P_{k+1} = \Phi(t_{k+1}, t_k)P_k\Phi^T(t_{k+1}, t_k) + Q_k
\end{aligned} \tag{C-2}$$

In Equation (C-2), ΔT represents the discretization period. It corresponds to maximum duration, in which assumptions utilized in Equation (C-2) hold. Therefore, conversion from continuous to discrete time equations should be performed at least once in ΔT period regardless of Kalman filter update cycle. The computations regarding to this discretization process is called as propagation routine and the maximum duration that this routine is calculated (ΔT) is called as propagation period.

In this study, Kalman Filters are implemented in the following structure:

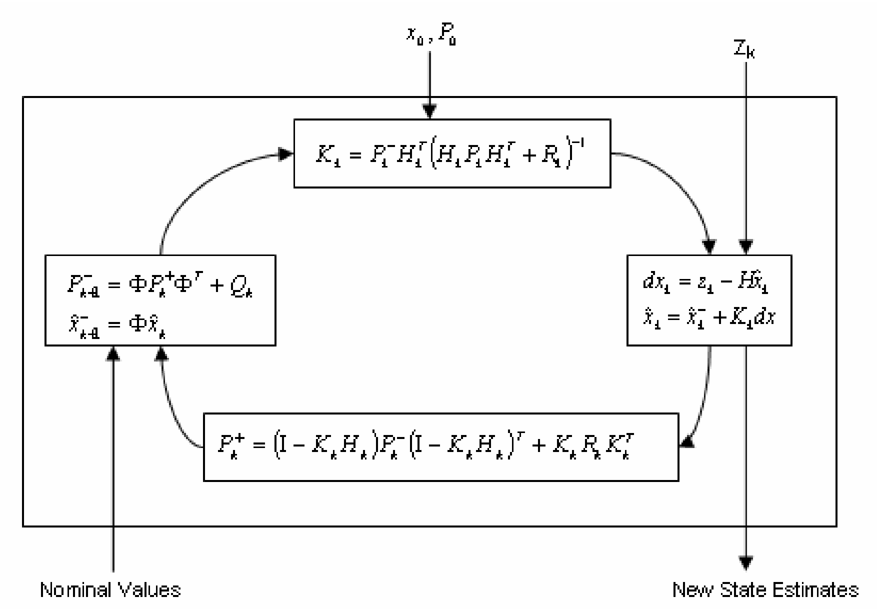

x_0, P_0
Z_k
$K_k = P_k^- H_k^T \left(H_k P_k H_k^T + R_k \right)^{-1}$
$P_{k+1}^- = \Phi P_k^+ \Phi^T + Q_k$
$\hat{x}_{k+1}^- = \Phi \hat{x}_k$
$dx_k = z_k - H\hat{x}_k$
$\hat{x}_k = \hat{x}_k^- + K_k dx$
$P_k^+ = \left(I - K_k H_k \right) P_k^- \left(I - K_k H_k \right)^T + K_k R_k K_k^T$
Nominal Values
New State Estimates

Figure C-1 : General structure of an Extended Kalman Filter

In Figure C-1 “z_k“ represents the new measurement obtained at time “t_k“. Using this measurement Kalman filter calculates an error estimate by using Kalman gain which is computed using states covariance matrix P_k and measurement covariance matrix R_k. The calculation of operations shown in Figure C-1 is called as update routine and the maximum period that this routine is processed is called as update period.