



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA STROJNÍHO INŽENÝRSTVÍ
LETECKÝ ÚSTAV**

FACULTY OF MECHANICAL ENGINEERING
INSTITUTE OF AEROSPACE ENGINEERING

NAVIGATION ALGORITHM FOR INS/GPS DATA FUSION

NÁVRH ALGORITMU PRO FÚZI DAT NAVIGAČNÍCH SYSTÉMŮ GPS A INS

DIPLOMOVÁ PRÁCE

MASTER'S THESIS

AUTOR PRÁCE

AUTHOR

Bc. MARKÉTA PÁLENSKÁ

VEDOUCÍ PRÁCE

SUPERVISOR

doc. Ing. JAN ČIŽMÁR, CSc.

BRNO 2013

Vysoké učení technické v Brně, Fakulta strojního inženýrství

Letecký ústav

Akademický rok: 2012/13

ZADÁNÍ DIPLOMOVÉ PRÁCE

student(ka): Bc. Markéta Pálenská

který/která studuje v **magisterském studijním programu**

obor: **Letecký provoz (3708T011)**

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma diplomové práce:

Návrh algoritmu pro fúzi dat navigačních systémů GPS a INS

v anglickém jazyce:

Navigation algorithm for INS/GPS Data Fusion

Stručná charakteristika problematiky úkolu:

Bezkartanové inerciální navigační systémy (BINS) pro svoji autonomnost představují strategicky významný navigační prostředek, na jehož dalším vývoji se neustále pracuje. Jeho důležitou součástí je i software, zajišťující vlastní základní funkčnost BINS (mechanizace) a software, který zajišťuje optimální fúzi dat získaných ze senzorů BINS a GPS.

Kvalita návrhu algoritmu BINS se přímo promítá do přesnosti a stability výpočtu navigačních a referenčních parametrů.

Cíle diplomové práce:

- 1) Návrh a realizace vhodného algoritmu mechanizace inerciálního navigačního systému
- 2) Algoritmus pro fúzi dat se systémem GPS prostřednictvím metody rozšířeného Kalmanova filtru. Algoritmus navrhnout v prostředí MATLAB - Simulink.
- 3) Experimentální ověření a hodnocení přesnosti a stability vytvořeného algoritmu.

Seznam odborné literatury:

- [1] Rogers, R., M.: Applied Mathematics in Integrated Navigation Systems, American Institute of Aeronautics and Astronautics, 2003, Reston, Virginia, ISBN 1-56347-656-8
- [2] Grewal, M., S. and Andrews, A.P.: Kalman Filtering Theory and Practice Using MATLAB, John Wiley & Sons, 2001, New York, ISBN 0-471-39254-5
- [3] Farrell, J., Barth, M.: The Global Positioning System & Inertial Navigation, McGraw-Hill Companies, Inc., New York, 1999, ISBN 0-07-022045-X

Vedoucí diplomové práce: doc. Ing. Jan Čížmár, CSc.

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2012/13.

V Brně, dne 20.11.2012



Ja Juračka

doc. Ing. Jaroslav Juračka, Ph.D.
Ředitel ústavu

M. Doupovec

prof. RNDr. Miroslav Doupovec, CSc., dr. h. c.
Děkan

Abstrakt

Diplomová práce se zabývá návrhem algoritmu rozšířeného Kalmanova filtru, který integruje data z inerciálního navigačního systému (INS) a globálního polohovacího systému (GPS). Součástí algoritmu je i samotná mechanizace INS, určující na základě dat z akcelerometrů a gyroskopů údaje o rychlosti, zeměpisné pozici a polohových úhlech letadla. Vzhledem k rychlému nárůstu chybovosti INS je výstup korigován hodnotami rychlosti a pozice získané z GPS. Výsledný algoritmus je implementován v prostředí Simulink. Součástí práce je odvození jednotlivých stavových matic rozšířeného Kalmanova filtru.

Summary

This diploma thesis deals with Extended Kalman Filter algorithm fusing data from inertial navigation system (INS) and Global Positioning System (GPS). The part of the developed algorithm is a mechanization of INS which processes data from accelerometers and gyroscopes to provide velocity, position and attitude angles. Due to rapid increase of INS output errors, the EKF is used to correct INS outputs by velocity and position from GPS. The final algorithm is developed in Simulink environment. This thesis includes derivation of EKF state matrices.

Klíčová slova

Rozšířený Kalmanův filtr, inerciální navigační systém, integrace INS/GPS, algoritmus pro fúzi dat

Key words

Extended Kalman Filter, inertial navigation system, INS/GPS integration algorithm, sensor data fusion

Bibliographic citation

PÁLENSKÁ, M. Navigation algorithm for INS/GPS data fusion. Brno: Vysoké učení technické v Brně, Fakulta strojního inženýrství, 2013. 80 s. Vedoucí diplomové práce doc. Ing. Jan Čížmár, CSc.

Declaration

I declare that I have written the master's thesis *Navigation algorithm for INS/GPS data fusion* on my own according to the instructions of my master's thesis supervisor doc. Ing. Jan Čižmár, CSc., and using the sources listed in the references.

May 20, 2013

Markéta Pálenská

I would like to thank my supervisor doc. Ing. Jan Čižmár, CSc. for presiding over my master's thesis. Many thanks belong to my consultants from Honeywell, Ing. Marek Fojtách and Ing. Jan Tomáš, Ph.D. for their support and provided knowledge.

Contents

1	Introduction	1
1.1	Outline of thesis	2
1.2	Current status of solved problematics	3
2	Inertial Navigation Systems	4
2.1	Coordinate frames	5
2.1.1	Earth-Centered Inertial Frame	5
2.1.2	Earth-Centered Earth-Fixed Frame	6
2.1.3	Local Navigation Frame	7
2.1.4	Wander Azimuth Frame	7
2.1.5	Body Frame	7
2.2	Transformation Equations	8
2.2.1	Quaternion	8
2.2.2	Direction Cosine Matrix	9
2.2.3	Euler Angles	10
2.2.4	Rotation Vector	11
2.3	INS Mechanization	11
2.3.1	Velocity Update	12
2.3.2	Position Update	13
2.3.3	Attitude Update	13
2.3.4	Gravity Vector Update	14
2.4	Inertial Sensors Error Model	15
2.4.1	Biases	15
2.4.2	Scale Factor and Cross-Coupling Errors	15
2.4.3	Random noise	16
2.4.4	Alignment of INS	16
3	INS/GPS Data Fusion	17
3.1	Global Positioning System	18
3.1.1	GPS signal and its processing	19

3.2	Integration Schemas	20
3.2.1	Loosely Coupled Integration	20
3.2.2	Tightly Coupled Integration	21
3.2.3	Deep integration	22
3.3	Kalman Filtering	22
3.4	Extended Kalman Filter	26
4	Development of integration algorithm	28
4.1	Attitude Error Analysis	30
4.2	Velocity Error Analysis	32
4.3	Position Error Analysis	35
4.4	Applying corrections	36
5	Achieved Results	38
5.1	Simulink model description	38
5.2	Results	40
6	Conclusion	60
6.1	Summary	60
6.2	Future work recommendations	61
6.3	Conclusion	61

List of Figures

2.1	Attitude angles [2]	4
2.2	ECI frame [5]	6
2.3	ECEF frame [5]	6
2.4	N-frame [5]	7
2.5	B-frame [3]	8
2.6	Schematic of an inertial navigation processor [1]	12
2.7	The scale factor error [2]	16
3.1	Open and closed loop of INS/GPS data fusion [1]	18
3.2	Loosely coupled integration [1]	21
3.3	Tightly coupled integration [1]	22
3.4	Deep integration [1]	23
3.5	Kalman filter algorithm steps [1]	24
3.6	Operation of EKF [15]	26
5.1	Schematic of my INS solution	39
5.2	Schematic of my EKF architecture	40
5.3	True values of position	42
5.4	True values of velocity	43
5.5	True values of attitude	44
5.6	Position errors from INS with ideal sensors	45
5.7	Velocity errors from INS with ideal sensors	46
5.8	Attitude errors from INS with ideal sensors	47
5.9	Position errors from INS with noisy accelerometers	48
5.10	Velocity errors from INS with noisy accelerometers	49
5.11	Position errors from INS with noisy sensors	50
5.12	Velocity errors from INS with noisy sensors	51
5.13	Attitude errors from noisy sensors	52
5.14	GPS Position errors	53
5.15	GPS Velocity errors	54
5.16	Simple EKF Position errors (noise only in accelerometers)	55

5.17	Simple EKF Velocity errors (noise only in accelerometers)	56
5.18	Full EKF Position errors	57
5.19	Full EKF Velocity errors	58
5.20	Full EKF Attitude errors	59
6.1	The Allan Variance on iFOG-IMU-1-A for accelerometer and gyro- scope in x-axis	69

List of Symbols

\cdot	Time derivative
δ	Error of
-1	Inverse matrix
T	Transposed matrix
\times	Cross product
(\times)	Skew symmetric form of a vector
$E(\cdot)$	Expectation of
$\Sigma(\cdot)$	Summation
A	Coriolis force
b	Bias vector
C_a^b	Direction cosine matrix
$C(t)$	C/A code
$D(t)$	GPS Navigation message
ε	Attitude error vector
e	First eccentricity of ellipsoid
f	Flattening of the ellipsoid
f	Specific force vector
F	State transition matrix
θ	Pitch angle
φ	Geodetic Latitude

Φ	Transition matrix
g	Gravity vector
H	Measurement matrix
h	Geodetic altitude
I	Identity matrix
K	Kalman gain
λ	Geodetic longitude
ρ	Rotation vector
ϕ	Roll angle
ω	Angular rate vector
Ω	Skew-symmetric matrix of angular rate vector
P	Covariance matrix of state vector
q	Quaternion
Q	Covariance matrix of system noise vector
r	Position vector
R_0	Equatorial Earth radius
R_N	Prime vertical radius of curvature
R_M	Meridian radius of curvature
R	Covariance matrix of measurement vector
T	Sample time
u	Output vector of IMU
x	State vector
y	Measurement vector

Chapter 1

Introduction

There are two possible approaches to navigation. The first is position fixing method based on measuring the ranges and bearings to known objects [1]. The Global Positioning System is an example of position fixing method. The second way called dead reckoning method which is based on measuring the changes in navigation quantities. These changes are integrated and added to previous values to get the actual value. These two approaches exhibit complementary positives and negatives. Thus, integrated navigation solution provides benefits of both methods.

Inertial navigation systems are among navigation systems based on dead reckoning. The INS has an impressively simple physical background. Fundamental idea comes from Newton's second motion of law and basic kinematics. Newton's second motion of law says that sum of forces acting the body is directly proportional to the mass and acceleration of the body. When acceleration of the body is measured with knowledge of initial conditions there can be simply found immediate velocity and position by integrating this with respect to time.

This implies the important advantage of these systems which is autonomy - inertial navigation systems do not need any external equipment except sensors (accelerometers and gyroscopes sensing the accelerations and angular rates) and navigation processor providing navigation solution. Thus, they are independent of external electromagnetic signals. Next advantages of INS are high short-term accuracy and short period output rate [1]. The continuous operation provides not just velocity and position, but also attitude¹, angular rates and accelerations. Disadvantages include rapid increase of error with time due to integration in the calculation. Outputs from accelerometers and gyroscopes are corrupted by noises and biases and without corrections result in unbounded errors. This is more noticeable in case of low-cost sensors as sensors based on MEMS technology. High performance sensors used e.g. in military and spacecraft application provide more precision but costs are about hundred thousands dollars [1].

¹When magnetometers are not part of sensors, the precision in heading is not sufficient

Global Positioning System as a system using position fixing method offers complementary positives and negatives. Low cost of user equipment and high long-term accuracy belong among the advantages of GPS. On the contrary this system is characterized by long period of output rate and possible unavailability because GPS is vulnerable to interference. Also high bandwidth noise is characteristic for GPS. Next potential drawback is a high short-term noise.

Complete navigation system fusing both of them results in high performance and robustness due to complementary attributes of INS and GPS. A typical integration architecture means that measurements from GPS used by an estimation algorithm to apply corrections to the navigation solution of INS.

Estimation theory provides a powerful tool to estimate states of interest in aided navigation system. This tool is Kalman filter and it is a key to get an optimal solution from both measurements. Although the name, Kalman filter is an estimation algorithm, rather than a filter [1]. It was invented in 1960s by a Hungarian mathematician R.E.Kalman [8]. This tool applies to wide range of disciplines but has irreplaceable function in control systems, avionics and space applications.

The first aim of the thesis is to create mechanization of inertial navigation system. Inputs to the system are data from three accelerometers and three gyroscopes and outputs are immediate values of attitude (roll, pitch and yaw angle), velocity and position. Model of INS is created in Simulink environment and sources of data are supplied by company Honeywell International s.r.o. The second step is to implement an Extended Kalman Filter to apply corrections based on GPS measurements to outputs of INS. Thus, the final algorithm should be INS aided by GPS. The assigned task is to create an error state model of EKF which estimates values of corrections to attitude, velocity and position and further noise parameters of sensors to correct sensor outputs directly. The aim of the work is a system corrected as at the input (removing computed biases in every step) and as at the output (corrections to computed navigation quantities in every step).

1.1 Outline of thesis

This diploma thesis contains six chapters. Chapter 1 focuses on the basic summary of the thesis with emphasis on the current status of the INS/GPS fusion common approach. Chapter 2 describes the overview of INS, focused on the mathematical equations describing the implemented INS mechanization. The chosen approach to the creation of functional model of INS is presented and the necessary theory concerning the reference frames and transformation of the quantities between them. The term of INS mechanization is understood as a set of equations giving a solution in the form of position, velocity and attitude from inputs to the system (measured accelerations and angular rates).

At the beginning of Chapter 3, different integration schemas of INS and GPS are studied with their pros and cons. Thus, the selected algorithm of data fusion - Extended Kalman filter (EKF) is presented. In Chapter 4 single matrices necessary to create an EKF algorithm are derived and some implementation issues are discussed. The Chapter 5 presents achieved results and describes developed Simulink models. The Chapter 6 summarizes the thesis and provides suggestions for next work. Appendixes to thesis present discretization of INS velocity equation and practical use of Allan Variance (method suitable for analyzing inertial sensors).

1.2 Current status of solved problematics

Kalman filter as an extremely powerful tool for analyzing estimation problems was developed in 1960s. The next step was the development of the MEMS technology in 1970s and their progress in the next decades enabling the quartz and silicon sensors to be mass produced at low cost using etching techniques with several sensors on a single silicon wafer [1]. The MEMS sensors are small and light, but currently their performance is relatively low. These lower grade inertial sensors are not suitable for inertial navigation, but their performance can be rapidly increased by integrating with a different navigation system (often GPS). Then even using these sensors in aerospace industry (rather in AHRS² than in INS) is possible. The connection of one of the great breakthroughs in estimation theory, Kalman filter, with MEMS technology significantly expands possibilities in use of inertial sensors.

My work is focused on Extended Kalman Filter algorithm to fuse data from INS and GPS. Nowadays some advanced INS/GPS integrations schemes exist. For example differential GPS improves position accuracy by calibrating out much of the temporally and spatially correlated biases in the pseudo-range values due to ephemeris prediction errors and residual satellite clock errors, ionospheric refraction or even tropospheric refraction [1]. The next possibility to increase performance of the MEMS sensors is the use of the adaptive Kalman filter to vary the assumed system noise. This approach results in speeding up the time of convergence of the state estimates with their true counterparts [1].

While Kalman filter is an ideal solution for real-time application, Kalman smoother is the way in such a case when measurements are needed after as well as before the time of interest. Kalman smoothing is realized by two main methods - the forward-backward method and Rauch, Tung and Striebel method. More detailed information and next extensions to Kalman Filter can be found e.g. in [1].

Recommended and actual sources to get detailed information about the topic discussed in this thesis are mainly [1], [2] and [5].

²Attitude and Heading Reference System

Chapter 2

Inertial Navigation Systems

Inertial Navigation System contains Inertial Measurement Unit, which is a set of three mutually orthogonal accelerometers and three gyroscopes measuring angular rates. Outputs from accelerometers are processed to get position and velocity, angular rates are processed to get attitude. Thus, outputs from INS are following:

$$vel = [v_N \ v_E \ v_D]^T$$

$$pos = [\varphi \ \lambda \ h]^T$$

$$att = [\phi \ \theta \ \psi]^T \quad (2.1)$$

where attitude angles (roll, pitch and yaw angle) describing rotation about axes of aircraft. The aeronautical convention defining these parameters as shown at Figure 2.1.

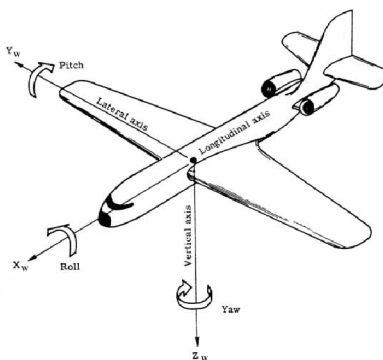


Figure 2.1: Attitude angles [2]

Values measured by accelerometers are called specific forces. Specific force is the acceleration due to all forces except gravity [1]. An important conclusion is that the acceleration does not equal to specific force except situation with zero tilt angle. This is one reason to know attitude to compensate it.

There are two commonly used mechanizations:

- Strapdown or gimbaless INS means that sensors are strapped into vehicle and are aligned with the navigation body. Therefore, the sensors rotate with vehicle. The alignment is done in navigation processor analytically. Strapdown INS has much simple mechanical construction but exhibits higher error rates due to sensor movement in all directions and following gravity influencing of all sensors.
- Stabilized mechanization means that the instrument platform is isolated not to rotate with vehicle. The angle to the gravity vector is constant. Transformation of the measured values from body to navigation frame is not needed. Due to smaller dynamic range, the sensing provides higher accuracy [5]. Gimbals arrangement can be very complicated and with high maintenance costs. When need of change some sensor occurs, complicated system must be dismantled and then completed in surgically clean environment, not to mention long calibration procedures [6].

Further strapdown INS will be considered.

2.1 Coordinate frames

Transformation of measured and computed quantities between various reference frames is needed. User wants to get his position in geographic coordinates. Sensor outputs are relative to inertial frame of reference, but the rotating Earth does not satisfy conditions of inertial frame. GPS also determines the position of receiver with a respect to satellites. Reference frames and transformation procedure are described in this chapter.

2.1.1 Earth-Centered Inertial Frame

Newton's laws are valid only in inertial frame of reference. Inertial frame of reference means non-rotating and non-accelerating frame. Earth-Centered Inertial Frame has a center at the Earth's center of gravity. Axes are non-rotating and directed to the distant stars. Z-axis is parallel to the spin axis of the Earth, x-axis points to the vernal equinox point and the y-axis is the axis completing right-handed orthogonal frame. The vernal equinox is the point of intersection of ascending node of ecliptic and the celestial equator. ¹ECI frame is marked by a notation i .

¹This reference frame also doesn't satisfy conditions for inertial frame due to non-constant velocity of Earth's motion around the Sun and due to Galaxy's rotation. This approximation is sufficient for navigation purposes.

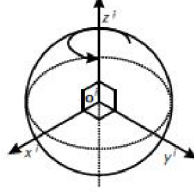


Figure 2.2: ECI frame [5]

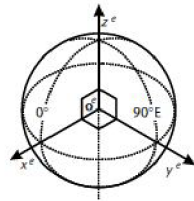


Figure 2.3: ECEF frame [5]

2.1.2 Earth-Centered Earth-Fixed Frame

The origin of the ECEF frame is at the Earth's center of gravity. Axes are fixed with respect to the Earth. Z-axis is the same as at the i-frame, x-axis points to mean meridian of Greenwich and y-axis completes right-handed orthogonal frame. The rotation rate vector of the e-frame with respect to the i-frame projected in e-frame is defined as [3]:

$$\omega_{ie}^e = [0 \ 0 \ \omega_{ie}]^T \quad (2.2)$$

The angular velocity of e-frame relative to i-frame is $\omega_{ie} = 7,292115 \times 10^{-5} \text{ rad s}^{-1}$ according to WGS 84. The constant rotation rate is assumed for navigation purposes although sidereal day² has not constant length. Variations are caused by wind, ice forming and melting and their size do not exceed several milliseconds [5]. ECEF frame is marked by a notation e.

Position vector in the e-frame is computed according to the equation 2.3 [3]:

$$r^e = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} (R_N + h) \cos \varphi \cos \lambda \\ (R_N + h) \cos \varphi \sin \lambda \\ (R_N(1 - e^2) + h) \sin \varphi \end{bmatrix} \quad (2.3)$$

where φ means geodetic latitude, λ denotes geodetic longitude, h altitude, R_N is the radius of curvature in the prime vertical and e is the first eccentricity of the reference

²Sidereal day is the period of rotation of the Earth with respect to the distant stars, solar day is defined with respect to the Sun

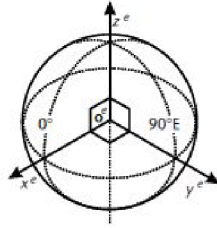


Figure 2.4: N-frame [5]

ellipsoid.

2.1.3 Local Navigation Frame

The origin of local navigation frame is at the point of the initial location of sensors. This frame is crucial for user because there is a need to know a position relative to north, east and down directions. Thus, this frame is also called as a NED coordinate system. X-axis points to the true north, z-axis is orthogonal to the surface of the reference ellipsoid and y-axis points to the east and completes the right-handed orthogonal frame. The notation of the local frame is n-frame.

The use of n-frame causes problems around the Earth's poles. To maintain orientation of x-axis to the north, the rotation about z-axis occurs. When strapdown system is used, rotation is performed just analytically, but problem appears when the stabilised platform is used. The solution is the use of w-frame.

2.1.4 Wander Azimuth Frame

The distinction between n-frame and w-frame is in the shift of x and y-axes. The angle of displacement is called a wander angle. The wander angle is equal to the meridian convergence from the point of alignment [7]. Using w-frame instead of n-frame avoids problems with singularity around Earth's poles.

2.1.5 Body Frame

The body frame or also vehicle frame is tightly bounded to the platform where sensors are placed. Its origin is identical with the origin of the local navigation frame, but the axes are aligned with the movement of vehicle. Outputs from accelerometers and gyroscopes are measured in the body frame and then are transformed. The x-axis of the body frame corresponds to roll axis, y-axis corresponds to pitch axis and z-axis corresponds to yaw axis.

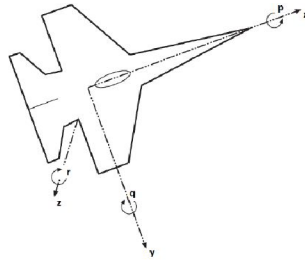


Figure 2.5: B-frame [3]

2.2 Transformation Equations

Approaches to transform quantities from one coordinate frame to another are described in this section. Commonly used techniques are quaternions, Euler angles, direction cosine matrixes and rotation vectors. Detailed description can be found e.g. in [1]. Imaginary frames 'a' and 'b' are used to describe each method of transformation.

2.2.1 Quaternion

Quaternion is described as a non-commutative extension of complex numbers with the scalar part and three dimensional vector part. The scalar part $s = q_0$ represents the magnitude of rotation and the vector part $v = [q_1 \ q_2 \ q_3]$ represents the axis about which that rotation takes place [1]. The main advantage is that quaternions do not suffer from gimbal lock. The next advantage is an amount and linearity of equations, on the other hand this approach is not too intuitive. The quaternion attitude definition:

$$q = \begin{bmatrix} s \\ v \end{bmatrix} = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} \cos(\frac{\mu}{2}) \\ u_1 \sin(\frac{\mu}{2}) \\ u_2 \sin(\frac{\mu}{2}) \\ u_3 \sin(\frac{\mu}{2}) \end{bmatrix} \quad (2.4)$$

where μ represents the rotation angle and $u = [u_1 \ u_2 \ u_3]$ is the unit vector describing rotation axis.

Definition of product of two quaternions [3]:

$$q_b^a = q_c^a * q_b^c = \begin{bmatrix} s_1 s_2 - v_1^T v_2 \\ s_1 v_2 + s_2 v_1 + v_1 \times v_2 \end{bmatrix} \quad (2.5)$$

Quaternion obtained from DCM [1]:

$$q_0 = \frac{1}{2} \sqrt{1 + c_{11} + c_{22} + c_{33}}$$

$$\begin{aligned}
q_1 &= \frac{c_{32} - c_{23}}{4q_0} \\
q_2 &= \frac{c_{13} - c_{31}}{4q_0} \\
q_3 &= \frac{c_{21} - c_{12}}{4q_0}
\end{aligned} \tag{2.6}$$

Quaternion obtained from Euler angles [3]:

$$q = \begin{bmatrix} \cos \frac{\phi}{2} \cos \frac{\psi}{2} \cos \frac{\theta}{2} + \sin \frac{\phi}{2} \sin \frac{\psi}{2} \cos \frac{\theta}{2} \\ \sin \frac{\phi}{2} \cos \frac{\psi}{2} \cos \frac{\theta}{2} - \cos \frac{\phi}{2} \sin \frac{\psi}{2} \sin \frac{\theta}{2} \\ \cos \frac{\phi}{2} \sin \frac{\psi}{2} \cos \frac{\theta}{2} + \sin \frac{\phi}{2} \cos \frac{\psi}{2} \cos \frac{\theta}{2} \\ \cos \frac{\phi}{2} \cos \frac{\psi}{2} \sin \frac{\theta}{2} - \sin \frac{\phi}{2} \sin \frac{\psi}{2} \cos \frac{\theta}{2} \end{bmatrix} \tag{2.7}$$

Quaternion obtained from rotation vector [3]:

$$q = \begin{bmatrix} \cos \|0.5\rho\| \\ \frac{\sin \|0.5\rho\|}{\|0.5\rho\|} 0.5\rho \end{bmatrix} \tag{2.8}$$

where

$$\cos \|0.5\rho\| = 1 - \frac{\|0.5\rho\|^2}{2!} + \frac{\|0.5\rho\|^4}{4!} - \dots$$

$$\sin \|0.5\rho\| = 1 - \frac{\|0.5\rho\|^2}{3!} + \frac{\|0.5\rho\|^4}{5!} - \dots$$

Symbol $\|\cdot\|$ means Euclidean form.

2.2.2 Direction Cosine Matrix

Direction Cosine Matrix (DCM) is the matrix of size 3×3 . C_a^b means DCM transforming vector from coordinate frame a to coordinate frame b . When using DCM, differential equations are linear and no singularity can occur, but the number of equations which needs to be considered is higher than other approaches.

DCM expressed using rotation vector [3]:

$$C_a^b = I + \frac{\sin \|\rho\|}{\|\rho\|} (\rho \times) + \frac{1 - \cos \|\rho\|}{\|\rho\|^2} (\rho \times)(\rho \times), \tag{2.9}$$

where symbol $\rho \times$ denotes skew-symmetric matrix³ of vector $\rho = [\rho_x \rho_y \rho_z]$:

³Skew-symmetric matrices are used to avoid vector multiplication

$$(\rho \times) = \begin{bmatrix} 0 & -\rho_z & \rho_y \\ \rho_z & 0 & -\rho_x \\ -\rho_y & \rho_x & 0 \end{bmatrix} \quad (2.10)$$

DCM obtained from quaternions [1]:

$$C_a^b = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 - q_3q_0) & 2(q_1q_3 + q_2q_0) \\ 2(q_1q_2 + q_3q_0) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 - q_1q_0) \\ 2(q_1q_3 - q_2q_0) & 2(q_2q_3 + q_1q_0) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \quad (2.11)$$

DCM obtained from Euler angles [1]:

$$C_a^b = \begin{bmatrix} \cos \theta \cos \psi & (-\cos \phi \sin \psi + \sin \phi \sin \theta \cos \psi) & (\sin \phi \sin \psi + \cos \phi \sin \theta \cos \psi) \\ \cos \theta \sin \psi & (\cos \phi \cos \psi + \sin \phi \sin \theta \sin \psi) & (-\sin \phi \cos \psi + \cos \phi \sin \theta \sin \psi) \\ -\sin \theta & \sin \phi \cos \theta & \cos \phi \cos \theta \end{bmatrix} \quad (2.12)$$

2.2.3 Euler Angles

Euler angles are a very intuitive kind of representation attitude. This trinity of angles $[\theta \ \phi \ \psi]$ describes rotation around three axes x , y and z . In such a case when the attitude of the body frame with respect to the local navigation frame is described, the roll rotation ϕ represents bank, the pitch rotation θ is elevation and ψ is known as heading. Disadvantage of this method is non-linearity of equations and singularity at ± 90 degrees.

Euler Angles obtained from DCM [3]:

$$\theta = \tan^{-1} \frac{-c_{31}}{\sqrt{c_{32}^2 + c_{33}^2}} \quad (2.13)$$

$$\phi = \tan^{-1} \frac{c_{32}}{c_{33}} \quad (2.14)$$

$$\psi = \tan^{-1} \frac{c_{21}}{c_{11}} \quad (2.15)$$

Euler Angles computed from quaternion [1]:

$$\theta = \tan^{-1} \frac{2(q_0q_1 + q_2q_3)}{1 - 2(q_1^2 - q_2^2)} \quad (2.16)$$

$$\phi = \sin^{-1}(2(q_2q_0 - q_1q_3)) \quad (2.17)$$

$$\psi = \tan^{-1} \frac{2(q_0 q_3 + q_1 q_2)}{1 - 2(q_2^2 - q_3^2)} \quad (2.18)$$

2.2.4 Rotation Vector

Describing rotation by rotation vector is a method coming from Euler's and Chasles's theorems. Information about vector which rotation occurs around and rate of rotation is needed for describing an attitude relative to local navigation frame [8]. The positive features of this method include the fact that differential equations explicitly account for non-commutativity effects. Among the disadvantages are the non-linearity of equations and singularity at $2n\pi$ rad. When small angle approximation is used, the rotation vector coincides with Euler angles.

Rotation vector obtained from quaternion [3]:

$$\rho = \frac{1}{k} [q_1 \ q_2 \ q_3]^T \quad (2.19)$$

$$k = \frac{1}{2} \left(1 - \frac{\|0,5\rho\|^2}{3!} + \frac{\|0,5\rho\|^4}{5!} - \frac{\|0,5\rho\|^6}{7!} + \dots \right) \quad (2.20)$$

$$\|0,5\rho\| = \frac{\sqrt{q_1^2 + q_2^2 + q_3^2}}{q_0} \quad (2.21)$$

Rotation vector obtained from DCM [3]:

$$\rho = \frac{\chi}{2 \sin \chi} \begin{pmatrix} c_{23} - c_{32} \\ c_{31} - c_{13} \\ c_{12} - c_{21} \end{pmatrix} \quad (2.22)$$

where

$$\chi = \arccos \left[\frac{\text{tr}(C_a^b) - 1}{2} \right]$$

where $\text{tr}(C_a^b)$ denotes trace of matrix.

2.3 INS Mechanization

Mechanization of INS is a process where outputs from accelerometers and gyroscopes are used to obtain navigation solution - values of attitude, velocity and position in n-frame. There is a consensus at continuous time equations as a result of more than twenty years of strapdown INS development [3]:

$$\dot{v}^n = C_b^n f^b + g^n - (2\omega_{ie}^n + \omega_{en}^n) \times v^n \quad (2.23)$$

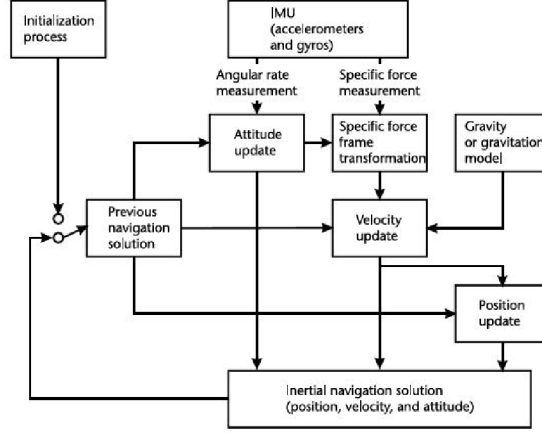


Figure 2.6: Schematic of an inertial navigation processor [1]

$$\dot{C}_n^e = C_n^e (\omega_{en}^n \times) \quad (2.24)$$

$$\dot{h} = -v_D \quad (2.25)$$

$$\dot{C}_b^n = C_b^n (\omega_{ib}^b \times) - (\omega_{in}^n \times) C_b^n \quad (2.26)$$

where C_b^n means direction cosine matrix from b-frame to n-frame and g^n is a vector of gravity. The angular rates have following meaning: ω_{ie}^n is Earth's rotation rate with respect to i-frame and ω_{en}^n means turn rate of the n-frame with respect to the e-frame. C_n^e is a Direction Cosine Matrix from n-frame to e-frame and v_D is a downward velocity. Angular rate ω_{ib}^b is measured by IMU and ω_{in}^n is computed as a sum of ω_{ie}^n and ω_{en}^n . Implemented equations are given by discretizing these equations.

Common process of computing navigation quantities is described at Figure 2.6.

The model is realized in the Simulink environment and works on frequency 100 Hz. Subsystem Trajectory Generator was supplied by Honeywell International company. It contains following data: IMU sensor outputs and true values about position, velocity and attitude used for verification of the correct function.

2.3.1 Velocity Update

Velocity in n-frame is calculated using equation 2.23. This equation is in continuous form. Laplace transform and Z-transform was applied to get this equation in discrete form. The process of derivation is provided in Appendix A. After discretization, the implemented equation is as follows:

$$v_k^n = (I + A \frac{T}{2})^{-1} \left[\left(I - A \frac{T}{2} \right) v_{k-1}^n + \frac{T}{2} (C_{b(k)}^n F_k + C_{b(k-1)}^n F_{k-1} + G_k + G_{k-1}) \right] \quad (2.27)$$

Symbols k and $k - 1$ mean current and previous state. T is a sample time and I is an unit matrix of size 3×3 . Matrix A is given by the equation 2.28 and represents Coriolis force correction. The actual value of Coriolis force correction is a sum of doubled Earth rate and transport rate. This sum is transformed from vector to skew-symmetric matrix.

$$A = (2\omega_{ie}^n + \omega_{en}^n) \times \quad (2.28)$$

2.3.2 Position Update

Equations in continuous form [8]:

$$\dot{\varphi} = \frac{v_n}{R_M + h} \quad (2.29)$$

$$\dot{\lambda} = \frac{v_E}{(R_N + h) \cos \varphi} \quad (2.30)$$

Derivation of altitude h is computed according to equation 2.25.

Thus, implemented equations are following:

$$\varphi = \int \frac{v_n}{R_M + h} dv_N \quad (2.31)$$

$$\lambda = \int \frac{v_E}{(R_N + h) \cos \varphi} dv_E \quad (2.32)$$

$$h = \int -v_D dv_D \quad (2.33)$$

2.3.3 Attitude Update

There is a need to say that results in yaw angle exhibit more inaccuracy than in the roll and pitch angle. The justification is that for complex AHRS solution magnetometres are next necessary sensors to precise heading computation. Thus, greater inaccuracy in the yaw angle is not caused by an error in algorithm.

Attitude is computed using equation 2.24. Implementation was obtained from [5]:

$$q_k = \left[\cos \frac{\|v\|}{2} I + \frac{2}{\|v\|} \sin \frac{\|v\|}{2} \Omega \right] q_{k-1} \quad (2.34)$$

q is a symbol of quaternion, v and Ω are quantities computed according to [5]:

Constant name	Value
Earth's gravitational constant	$\mu = 3.986004418 \cdot 10^{14} \text{ m}^3 \text{ s}^{-2}$
Earth's angular rate	$\omega_{ie} = 7.2921158 \cdot 10^{-5} \text{ rads}^{-1}$
Flattening of the ellipsoid	$f = \frac{1}{298.257223563}$
Eccentricity of the ellipsoid	$e=0.0818191908426$
Equatorial radius	$R_0 = 6378137\text{m}$
Polar radius	$R_p = 6356752.3142\text{m}$

Table 2.1: WGS 84 values used for the computing of the gravity vector

$$\mathbf{v} = \dot{\boldsymbol{\omega}}_{bn}^n = \left\{ \left[\boldsymbol{\omega}_{ib(k-1)}^b - C_{n(k-1)}^b \boldsymbol{\omega}_{in}^n \right] + \left[\boldsymbol{\omega}_{ib(k)}^b - C_{n(k)}^b \boldsymbol{\omega}_{in}^n \right] \right\} \frac{T}{2} \quad (2.35)$$

$$\boldsymbol{\Omega} = \begin{bmatrix} 0 & -v_x & -v_y & -v_z \\ v_x & 0 & v_z & -v_y \\ v_y & -v_z & 0 & v_x \\ v_z & v_y & -v_x & 0 \end{bmatrix} \quad (2.36)$$

Thus DCM is obtained from quaternion using 2.11 and Euler angles are obtained from DCM using equations 2.13 to 2.15⁴. Matrix $\boldsymbol{\Omega}$ is called as a skew-symmetric matrix of vector \mathbf{v} .

2.3.4 Gravity Vector Update

Gravity vector is computed from actual position according to the following equations [1]:

$$\mathbf{g} = g(\varphi) \left[1 - \frac{2}{R_0} \left(1 + f + \frac{\omega_{ie}^2 R_0^2 R_p}{\mu} \right) h + \frac{3}{R_0^2} h^2 \right] \quad (2.37)$$

$$g(\varphi) = 9.7803253359 \frac{1 + 0.01931853 \sin^2 \varphi}{\sqrt{1 - e^2 \sin^2 \varphi}} \quad (2.38)$$

The equation 2.38 describes a simple WGS-84 model of the acceleration due to gravity at the ellipsoid as a function of latitude. This is a gravity field model including outward centrifugal acceleration due to Earth's rotation (virtual force arising from the use of rotating resolving axes). To get rid of this acceleration and obtain just gravitational acceleration there is a need to subtract this. Thus, gravitational acceleration varies with height and needs to be scaled. This resulted in equation 2.37 [1].

Used values of Earth's parameters are taken from WGS 84 and are summarized in Table 2.1.

⁴The direct conversion from quaternion to Euler angles is not used, because direction cosine matrix is used for the computing of velocity

2.4 Inertial Sensors Error Model

Inertial sensors are accelerometers and gyroscopes measuring specific forces and angular rates without external device. There are various types of inertial sensors. Accelerometers can operate on the piezoelectric or capacitive principle to convert mechanical motion into an electrical signal. Technology of gyroscope can be fiber-optical, ring laser or based on spinning mass [1]. Used technology differs in cost, size, performance, possible use and other factors. The current status of the development of low-cost sensors is marked by using MEMS technology.

MEMS (Micro-Electrical-Mechanical Systems) technology uses sophisticated placement of electrical and micro-mechanical elements to silicon wafer via etching technology. The advantages of MEMS sensor are small size and weight, low cost and power consumption and relatively great shock tolerance. Of course, their performance is poor in comparison to very accurate laser-ring sensors.

There is no perfect sensor without any error. Every sensor output is corrupted and there is a need to analyze errors and remove the maximum possible amount of them. The errors of IMU sensors can be divided into several categories. They include biases, scale factor, cross-coupling errors and random noise.

2.4.1 Biases

The bias is a constant error exhibited by all accelerometers and gyros. The bias is often one of major contributors to overall error. Bias is the composition of static and dynamic part. The static component of bias (also known as bias offset) is constant through IMU operation, but it differs run-to-run [1]. Bias offset can be calibrated by measurement without any IMU input. The dynamic part is called bias instability or in-run bias variation and varies over period of order a minute. Typically it represents about 10 % size of the static bias [1]. Accelerometer bias is denoted as $b_a = (b_{a,x}, b_{a,y}, b_{a,z})$ and respectively gyro bias is denoted as $b_g = (b_{g,x}, b_{g,y}, b_{g,z})$.

2.4.2 Scale Factor and Cross-Coupling Errors

The scale factor error represents error in the mapping of the input-output relationship. The scale factor error of sensor output is proportional to the true value. The denoting of accelerometer scale factor errors is $s_a = (s_{a,x}, s_{a,y}, s_{a,z})$ and the denoting of gyroscope scale factor error is $s_g = (s_{g,x}, s_{g,y}, s_{g,z})$. The scale factor is illustrated at Figure 2.7.

Cross-coupling error (or misalignment error) comes from misalignment of the sensitive axes. It is caused by production constraints. Generally, misalignment errors do not significantly contribute to overall error, even though some low-cost IMU sensors can be exception.

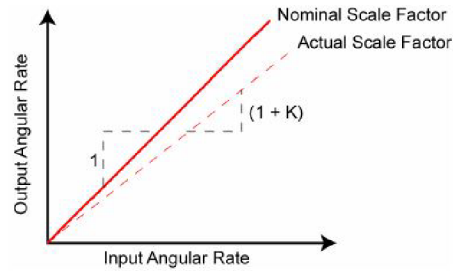


Figure 2.7: The scale factor error [2]

2.4.3 Random noise

Random noise occurs in all IMU sensors and it is caused by various reasons. Mechanical instabilities are manifested at spinning-mass gyros, high-frequency resonances cause noise in vibratory gyros etc. MEMS sensors are loaded with a lot of random noise. The denotation is $w_a = (w_{a,x}, w_{a,y}, w_{a,z})$ at accelerometres and $w_g = (w_{g,x}, w_{g,y}, w_{g,z})$ at gyroscopes.

2.4.4 Alignment of INS

IMU sensor outputs are integrated to get the navigational solution. This implies the need of the initial estimation of values. The estimation includes initialization, calibration and alignment. The initial phase of ground alignment is the levelling which provides the estimate of initial attitude. The alignment is defined as determining the relative orientation of the INS platform and the reference navigation frame axes. The initialization means initial estimate of velocity and position and calibration is a process of determining of various factors to calibrate inertial instruments [2].

The calibration of IMU sensors can be off-line in laboratory or on-line using state-augmentation techniques. The initial estimate of velocity, position and attitude with their error is needed at least at the start of operating. When aiding INS is considered, estimates are further needed. Correct initialization is important not just to get integrating constants. For example, the incorrect estimate of position affects gravity vector and Coriolis force compensation. Also wrong estimate of initial attitude results in incorrect velocity computation etc. Thus, alignment, initialization and calibration are critical to system performance [2]. The problem of initialization is often solved via estimation theory. Therefore, the initialization is performed using the same tools as normal operation (when aided INS is used).

Chapter 3

INS/GPS Data Fusion

Brief reasons to integrate INS and GPS were presented in Chapter 1. Complementary disadvantages of these systems exhort to use both of them together. The dominant errors of INS are low frequency drifts, on the contrary the main GPS error consists in high bandwidth noise. But there are other possible navigation systems which can be used to aid INS. In the area of personal navigation, odometres or some other low-cost dead-reckoning systems are used [1]. In aviation, the fusion of INS and GPS is an ideal solution to get continuous high bandwidth information. Due to the development of MEMS technology, the price of inertial sensors has been significantly decreased.

The weaknesses of GPS include the vulnerability to outages, low data rate and no attitude information¹. On the other hand, the GPS errors are bounded and the GPS equipment is cheap. INS is an autonomous system with high data rate, but is susceptible to error drifting. Any inaccuracy in sensor informations due to integration in INS mechanization leads to errors which grow with time. Prices of IMUs which provide sufficient-quality solutions for periods longer than the order of minutes are very high. So, integration of GPS and INS allows to create a reliable navigational system even with inexpensive inertial sensors.

There exists several integration approaches. They differ in a way how corrections gained from integration algorithm are applied back to the INS, what types of GPS information are used (if just position and velocity or directly pseudorange and carrier phase) and if GPS equipment uses outputs from integration algorithm or not. Integration algorithm can work as the total state or as the error state - in the first case the algorithm computes true values of navigational quantities, in the case of error state approach the algorithm computes corrections (distinctions from true values are intended). As the assignment of my thesis is the error state approach, only this method will be discussed further.

There is no precise definition of particular integration approaches, so commonly

¹If carrier phases with multiple antennas are not used [8].

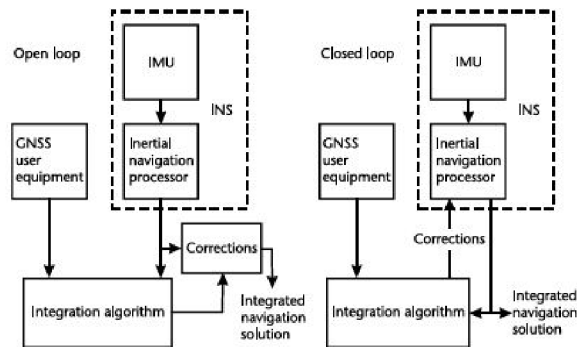


Figure 3.1: Open and closed loop of INS/GPS data fusion [1]

used terms according to Groves are presented. Basic types are loose, tight and deep integration differing in the degree of integration. The greater degree of integration brings better performance and higher reliability, but on the other hand the independence of both systems is lost. My thesis focuses on the loose integration, thus this type is described more detailed.

The integration schemas can also be divided into open or closed solutions. In other words, if computed corrections of sensors are fed back to the INS (closed loop) or if just corrections to INS outputs are processed (open loop). Decision about suitable approach is a question of quality of inertial sensors and integration algorithm. Higher quality sensors with not so powerful algorithm justifies use of open loop. When ratio of quality is opposite, the use of closed loop makes more sense. The issue is if raw data from sensors are meaningful to maintain and use the open-loop configuration as integrity monitoring. Based on quality of analyzed and simulated sensors, the closed loop solution was chosen.

3.1 Global Positioning System

GPS is the global navigation satellite system operated by U.S. Department of Defense. The GPS user equipment measures time delays and decodes messages from visible satellites to determine position and accurate time anywhere on Earth. Limited accuracy services are freely available to civil users. This section describes the basic principle of operating and typical error sources.

GPS consists of three segments - space, control and user. The space segment contains from 24 to 32 satellites. Satellites move in six circular orbits at altitude of 20,200 km. These satellites broadcast signals to control segment and to users. The inclination angle relative to the equator is 55 degrees, the spacing of orbits is 60 degrees. The constellation is designed with at least four satellites visible anytime [9].

The control segment represents the network of monitor stations which provides correction for each satellite. The next part of control segment are control stations which calculate the needed correction to the satellite motion² and the navigation data message to be send back to the space segment. The validity of each navigation message is several hours and contains data about ephemeris (precise satellite orbit), ionospheric refraction model etc. The user segment consists of passive users.

Global Positioning Service provides two levels of accuracy - Precise Positioning Service (PPS) and Standard Precisioning Service (SPS). PPS is available just for authorized users (U.S. Army and selected allies armies). The approach to PPS is allowed by the decoding key. The PPS enables to achieve accuracy in tens of centimetres. SPS is a free service accesible to unlimited number of users with accuracy about 100 metres³.

The principle of GPS is based on time-of-arrival ranging [5]. The GPS receiver calculates the required time for transmission from the satellite to the receiver. This time interval is converted to distance by the multiplication by speed of light. This distance is called pseudorange due to errors in the satellite's and the receiver's clock. At least four satellites are needed for succesful positioning. One known range gives information that the searched position is on the sphere. Three ranges lead to the intersection of three spheres which gives two points. One of them can be excluded because approximate position is known. Thus, the fourth measure is needed for the elimination of the receiver clock's bias.

Receiver clock's bias is time varying error that affects all simultaneous range measurements in the same manner [5]. This error can be estimated when four satellites are visible. The next type of error is a satellite clock's bias. This bias is estimated by monitor stations and then sent to the user segment to correct range. The atmospheric delay is an error caused by conditions influencing speed of light as temperature, pressure and humidity changes (tropospheric atmospheric delay) and level of air ionization (ionospheric atmospheric delay).

Selective Availability errors are purposely added to decrease accuracy for nonauthorized users. The SA error can be achieved in two ways - by corrupting the broadcasting of ephemeris data or by dithering satellite oscillator frequency [5].

3.1.1 GPS signal and its processing

This subsection is based on [9]. Every GPS satellite broadcasts on two frequencies:

$$L_1 = 1575,42 \text{ MHz}$$

²Maneuvers are known as station keeping used to keep correct satellite track.

³Using differential GPS in the surroundings of control stations significantly improves accuracy.

$$L_2 = 1227,6 \text{ MHz}$$

The GPS signal is described by a following equation [9]:

$$s(t) = A_c C(t) D(t) \sin(2\pi L_1 t) + A_{p1} P(t) D(t) \cos(2\pi L_1 t) + A_{p2} P(t) D(t) \cos(2\pi L_2 t) \quad (3.1)$$

Carrier waves are modulated by $C(t)$ and $D(t)$ codes. Codes and data have values $+1$ and -1 . Thus, the modulation type is Binary-Phase Shift Keying (BPSK). The meaning of codes $C(t)$ and $D(t)$ is enabling of distance measuring and separation of individual satellite's signals. Data $D(t)$ are used for ephemeris data transmission. Satellite positions are computed in user's receiver from ephemeris data.

Codes are pseudorandom sequences used to increase resistance to interference (due to spread spectrum transmission). The next reason to use codes is for separation of satellites (code multiplex). Code $C(t)$ is called Gold code. Gold codes have bounded small cross-correlations within a set, which is used for sharing frequency by multiple transmitters. The bit rate is 1,023 MB/s. The designation of this code is C/A, Coarse Acquisition.

The code $P(t)$ is so-called Precision code. It is a pseudorandom sequence with period of an approx. 266 days (but only seven days long part is used). The bit rate is 10,230 MB/s. This code enables greater precision due to longer and faster code which increases frequency spreading. Also user can measure at both frequencies when using $P(t)$ code. Using of both frequencies helps to deal with ionospheric refraction.

GPS errors is possible to divide into three groups. The first of them are errors formed in the space segment. These errors come mainly from stability of satellite's frequency issues and prediction of satellite's perturbation. Greater error obtained from the control segment comes from ephemeris prediction. At the side of user, the main part of error is ionospheric refraction. Further, the tropospheric refraction, receiver's noise or multipath propagation of signal contribute to overall error.

3.2 Integration Schemas

3.2.1 Loosely Coupled Integration

The loosely coupled integration of INS and GPS is characterized by the use of GPS outputs (position and velocity) to compare with INS output. Differences between INS and GPS solution are utilized as measurement. System model of estimation algorithm is based on INS error dynamics equations. The output from estimation algorithm, of-

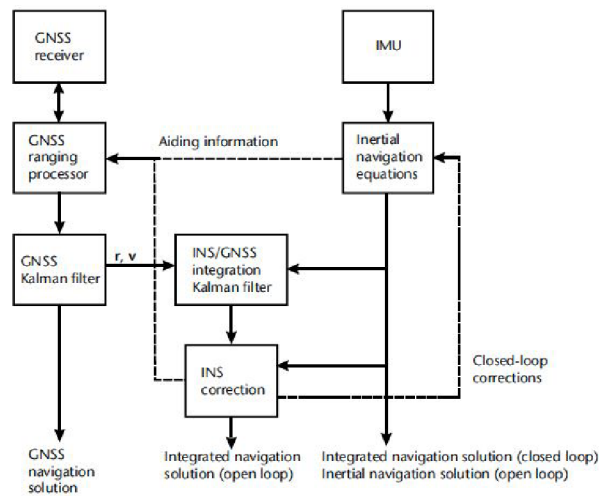


Figure 3.2: Loosely coupled integration [1]

ten the Extended Kalman Filter ⁴, is used as correction and applied back to the INS. The integrated navigation solution is then the corrected INS navigation solution. INS and GPS provide the independent solution and the estimation algorithm forms the third solution. The advantage is a simplicity, redundancy and existence of the independent GPS solution in case of the open-loop solution. Another advantage is the faster processing time (compared to the tightly coupled integration) due to smaller dimensions in state vectors [13]. When the loosely coupled integration scheme is selected, the important question is the choice of measurement iteration rate. Too rapid rate leads to instability, on the contrary too slow rate can cause less observability [1]. This method needs four visible satellites which limits the use for terrestrial applications. Also the situation with frequent outages of GPS requires more accurate sensors (to maintain acceptable precision during outages).

3.2.2 Tightly Coupled Integration

In case of tightly coupled integration input as measurement to the estimation algorithm are pseudo-range and pseudo-range rates from GPS [1]. The use of just one of them is possible, but in practice the use of both is advantageous due to their complementarity.⁵ Satellite ephemeris data are used to derive pseudo-range from INS. The advantage of this way of integration is a greater resistance against interference and jamming and the fact that the integrated solution is available even in such a case when less than four satellites are visible. The difficulty is that the system designer must have access to

⁴Next possibilities are unscented Kalman filter or particle filters [12]

⁵Pseudo-ranges come from code tracking and pseudo-range rates come from carrier tracking which is more accurate and less robust than code tracking [1].

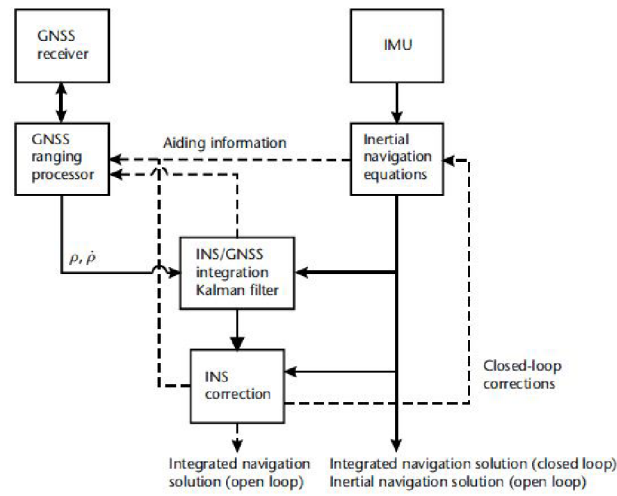


Figure 3.3: Tightly coupled integration [1]

GPS equipment hardware. So this access is not suitable for general using [5]. Another disadvantage is the loss of independent GPS solution and greater computational load. According to Solimeno, the computational load (number of operations) of the tightly coupled system compared to the loosely coupled system increases linearly with the number of visible satellites.

3.2.3 Deep integration

Deep integration approach (also known as ultra-tight) enables GPS outputs to be generated by means of the corrected INS solution, information from navigation data message and GPS error estimates. This architecture improves noise resistance. As well as the previous solution for short time intervals, this kind of system is operational when less than four satellites are available [1].

3.3 Kalman Filtering

The Kalman filter represents one of most widely used method to estimate variables in navigation systems. Its use enables to integrate measurements from various sources into optimal solution which provides better performance than either of them alone. This chapter presents basic principles of this algorithm.

The Kalman filter is a linear statistical estimation algorithm invented by R. E. Kalman. The assumption is that the observed system is driven by noise which is characterized by stochastic quantities and that sensors are disrupted by the same type of noise. The Kalman filter is an optimal filter minimizing variance for known stochastic

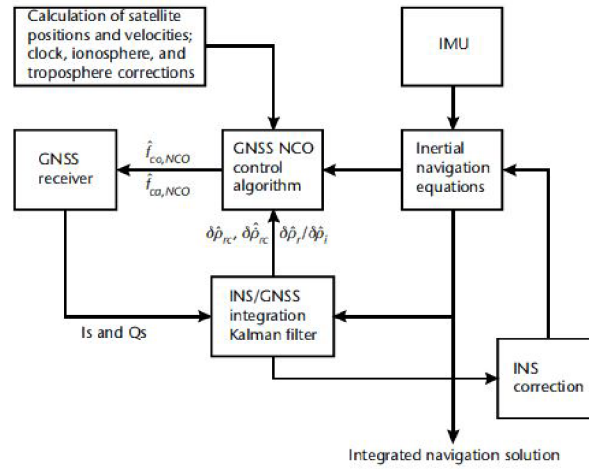


Figure 3.4: Deep integration [1]

linear model with gaussian zero-mean noise and known covariance [6]. The estimator is driven by same inputs as the observed system and the estimate is based on comparing both inputs (computed and measured).

The algorithm estimates system parameters in real time based on measuring. Measurements are function of estimated parameters [1]. This results in rapid decrease of noise and other errors inseparably associated with each type of measurement. The algorithm contains two phases - prediction and then update of estimate based on measurement.

Basic elements of Kalman filter are following - the state vector and covariance matrix, the process model, the measurement vector and covariance matrix, model of measurement and, finally, the algorithm. The state vector is understood as system parameters which are needed to be estimated. In this case estimated variables are attitude errors, velocity errors, position errors and biases of accelerometers and gyroscopes. This results in fifteen components of state vector. The error covariance matrix describes correlation between estimated errors. The process model (or system model) describes the dynamics of the system - the variance of states with time. The measurement vector is a vector of measured quantities. Analogously to error covariance matrix, the measurement noise covariance matrix defines noise characteristics of measurement. Measurement model describes dependence of measurement vector on state vector.

Algorithm itself contains ten steps at one iteration. First four steps belong to system propagation phase, next six steps to measurement-updated phase. First phase predicts the state vector estimate and error covariance matrix from the time of validity of the last measurement set to the time of current set of measurements using the known properties of the system. The measurement-update phase occurs after gain of new measurement.

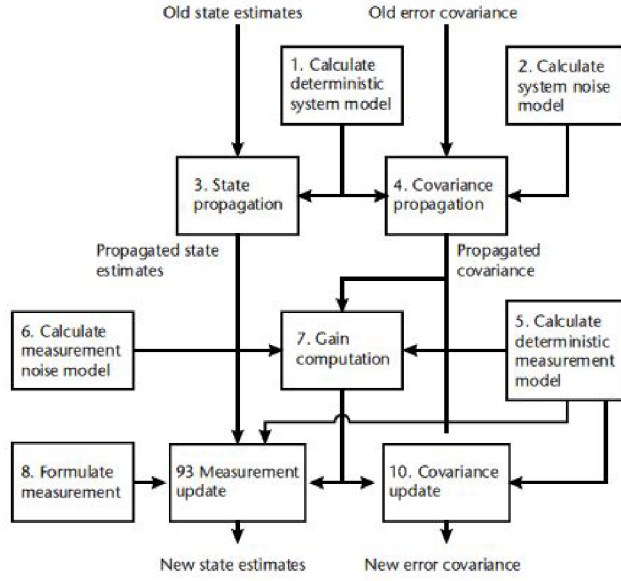


Figure 3.5: Kalman filter algorithm steps [1]

Detailed description of algorithm can be found in [1] or in [14].

Discrete-time system is described by following equations:

$$x_{k+1} = F_k x_k + G_k u_k + w_k \quad (3.2)$$

$$y_{k+1} = H_{k+1} x_{k+1} + v_{k+1} \quad (3.3)$$

where x denotes state vector, F is the state transition matrix, G is the system noise distribution matrix, u is the input vector, w is the process noise, y is the vector of measurement, H is the measurement matrix and v is measurement noise. Noises are considered to be uncorelated, zero-mean and satisfying following conditions [14]:

$$w_k \sim (0, Q_k) \quad (3.4)$$

$$v_k \sim (0, R_k) \quad (3.5)$$

$$E[w_k w_j^T] = Q_k \delta_{k-j} \quad (3.6)$$

$$E[v_k v_j^T] = R_k \delta_{k-j} \quad (3.7)$$

$$E[v_k w_j^T] = 0 \quad (3.8)$$

Symbol δ_{k-j} denotes Kronecker delta function acquiring value 1 in case $k = j$ and value 0 otherwise. Q is process noise covariance matrix, R is measurement noise covariance matrix and $E[x]$ means expected value of parameter x .

System-propagation phase of Kalman filter is described by:

$$x_{k|k-1} = F_{k-1}x_{k-1|k-1} + G_{k-1}u_{k-1|k-1} \quad (3.9)$$

$$P_{k|k-1} = F_{k-1}P_{k-1|k-1}F_{k-1}^T + Q_{k-1} \quad (3.10)$$

Significance of different variables:

- $x_{k-1|k-1}$ an estimate of state vector in time t_{k-1} .
- $x_{k|k-1}$ prediction of state vector in time t_k before the processing of measurement vector y_k
- $P_{k|k-1}$ prediction of covariance matrix in time t_k before the processing of measurement vector y_k

Measurement-update phase of Kalman filter:

$$K_k = P_{k|k-1}H_k^T(H_kP_{k|k-1}H_k^T + R_k)^{-1} \quad (3.11)$$

$$x_{k|k} = x_{k|k-1} + K_k(y_k - H_kx_{k|k-1}) \quad (3.12)$$

$$P_{k|k} = (I - K_kH_k)P_{k|k-1}(I - K_kH_k)^T + K_kR_kK_k^T \quad (3.13)$$

Significance of different variables:

- K_k Kalman gain
- $x_{k|k}$ Estimate of state vector x after the measurement update
- $P_{k|k}$ Estimate of covariance matrix after the measurement update
- I Identity matrix

Discretization of matrix F is usually performed using power-series expansion [1]:

$$\phi_{k-1} = I + F_{k-1}\tau_s + \frac{1}{2}F_{k-1}^2\tau_s^2 + \frac{1}{6}F_{k-1}^3\tau_s^3 + \dots \quad (3.14)$$

Common solution to discretize matrix Q is following [4]:

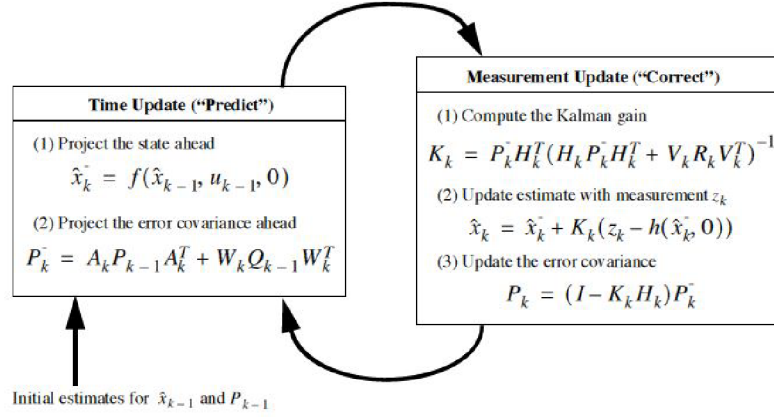


Figure 3.6: Operation of EKF [15]

$$Q_k = \phi_k G Q G^T \phi_k^T \quad (3.15)$$

The tuning of the Kalman filter means selecting optimal values of matrixes Q_k , R_k and initial values of the error covariance matrix P_0 . The computing of a Kalman gain as a ratio of $P_{k|k-1}$ and R_k is an essential feature of the system. The underestimation of this ratio leads to a very slow convergence to correct values. On the contrary too large Kalman gain can cause instability or noising of state estimates due to the measurement noise having too great influence on them [1]. Thus, it can be said that the tuning of Kalman filter is kind of decision between convergence rate and stability.

Thus, the summary is a following: succesfull implementation of Kalman filter means correct setting of matrices F, G, Q, R , and H . The own Kalman Filter algorithm is then realized by five equations 3.9 to 3.13.

3.4 Extended Kalman Filter

In case of nonlinear processes like a navigation system, the extension to the Kalman filter is used. The Extended Kalman filter (EKF) linearizes values about the current mean and covariance [15]. Because EKF is a just local linear approximation, it can easily diverge. The next distinction from Kalman filter is in fact that computing of covariance matrices is not possible to do off-line. The reason is that covariance matrices are function of measurement in EKF [16].

In a standard Kalman filter vector of measurement y is a linear function of the state vector x . In case of EKF, matrices F and H are substituted by nonlinear functions of the state vector $f(x)$ and $h(x)$. Thus, the discrete system is described by following

equations [1]:

$$x_{k+1} = f_k(x_k, u_k, w_k) \quad (3.16)$$

$$y_{k+1} = h_{k+1}(x_{k+1}, w_{k+1}) \quad (3.17)$$

$$w_k \sim (0, Q_k) \quad (3.18)$$

$$v_k \sim (0, R_k) \quad (3.19)$$

where f is a nonlinear function of state vector and h is a measurement vector.

Extended Kalman filter algorithm is the same in equations 3.9 to 3.13 with following Jacobian evaluating in each step:

$$F_{k-1} = \frac{\partial f(x)}{\partial x} \Big|_{x_{k|k}} \quad (3.20)$$

$$H_k = \frac{\partial h(x)}{\partial x} \Big|_{x_{k|k}} \quad (3.21)$$

Linearization of F and H matrices results in fact that error covariance matrix P and Kalman gain K are dependent on the state estimates. This can lead to problems with stability of EKF [1]. To maintain this independency linearized Kalman filter can be used⁶.

⁶In this solution F and H matrixes are linearized about predetermined state vector, which determines use in application where approximated trajectory is known [1]

Chapter 4

Development of integration algorithm

There are two possibilities of designing EKF for INS and GPS data fusion - total state or error model approach. The total state approach means that estimated states are directly navigation outputs as attitude, velocity and position. For the purpose of my thesis, the developing of error model EKF was assigned. The final developed algorithm estimates errors in attitude, velocity and position and accelerometer and gyro biases. Thus, the state vector has a following form:

$$x = \begin{pmatrix} \delta\psi_{ib}^i \\ \delta v_{ib}^i \\ \delta r_{ib}^i \\ b_a \\ b_g \end{pmatrix} \quad (4.1)$$

Perturbation of navigation quantities is given by following equations¹ [4]:

$$\hat{r}_n = r_n + \delta r_n \quad (4.2)$$

$$\hat{v}_n = v_n + \delta v_n \quad (4.3)$$

$$\hat{C}_b^n = (I - E^n) C_b^n \quad (4.4)$$

In equation 4.4 E^n means skew symmetric form of attitude error matrix:

¹There are two possibilities how to correct attitude - to DCM or to quaternion. The quaternion approach is presented in Section 4.4

$$E^n = \begin{pmatrix} 0 & -\varepsilon_D & \varepsilon_E \\ \varepsilon_D & 0 & -\varepsilon_N \\ -\varepsilon_E & \varepsilon_N & 0 \end{pmatrix} \quad (4.5)$$

Symbol $\hat{\cdot}$ denotes quantity perturbed by error which is designated by δ . Next, the symbol for attitude error $\delta \psi_{ib}^i$ is simplified to ε .

The system matrix F has a following form [1]:

$$F = \begin{pmatrix} F_{11} & F_{12} & F_{13} & 0_3 & C_b^n \\ F_{21} & F_{22} & F_{23} & C_b^n & 0_3 \\ 0_3 & F_{32} & F_{33} & 0_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & 0_3 & 0_3 \\ 0_3 & 0_3 & 0_3 & 0_3 & 0_3 \end{pmatrix} \quad (4.6)$$

Then, matrix G has the following form:

$$G = \begin{pmatrix} 0 & -C_b^n \\ C_b^n & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \quad (4.7)$$

Forcing (or input) vector function u is given by IMU outputs:

$$u = \begin{pmatrix} \delta f^b \\ \delta \omega_{ib}^b \end{pmatrix} \quad (4.8)$$

The system noise covariance matrix contains standard deviations of sensors noise²:

$$Q = \begin{pmatrix} \sigma_{ax}^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma_{ay}^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_{az}^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_{\omega_x}^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_{\omega_y}^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma_{\omega_z}^2 \end{pmatrix} \quad (4.9)$$

The transformation of matrixes F and Q to discrete form was described in equation 3.15 and 3.14. The meaning of matrix Q_k is in the level of confidence to the measurements. According to Shin, the norm of Q_k bigger than the real one means the measurements have more confidence³ then INS alone. On the contrary, the norm of Q_k smaller than the real one, EKF has a tendency to diverge. For purpose of my thesis the result is to

²Setting correct values of noise covariance matrix is a question of tuning Kalman filter. Not always precise standard deviations give the best performance

³Measurements are noisy due to measurement noise, thus estimates exhibit the same level of noise

give more confidence to GPS, consequently to measurements. Reason is that low-cost sensors are considered and simulated. The tuning of the EKF gives us the proper values of Q matrix members. My EKF was tuned manually until parameters of accuracy and stability were optimal, but some more sophisticated methods can be used. One of automatic real-time tuning methods is Adaptive Kalman Filter. More information can be found for example in [1].

Measurement model is given by equation 6.1. Vector of measurement is given by difference between INS and GPS values:

$$z = \begin{pmatrix} vel_{INS} - vel_{GPS} \\ pos_{INS} - pos_{GPS} \end{pmatrix} \quad (4.10)$$

Matrix H known as measurement matrix has the following form in my implementation:

$$H = \begin{pmatrix} 0_3 & 0_3 & -I_3 & 0_3 & 0_3 \\ 0_3 & -I_3 & 0_3 & 0_3 & 0_3 \end{pmatrix} \quad (4.11)$$

where 0_3 means zero matrix 3x3 and I_3 represents unit matrix of size 3x3.

Measurement noise covariance matrix R is given by standard deviations of measurement sensors:

$$R = \begin{pmatrix} \sigma_\phi^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sigma_\lambda^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sigma_h^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sigma_{v_N}^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sigma_{v_E}^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sigma_{v_D}^2 \end{pmatrix} \quad (4.12)$$

In case of simulated GPS (true values with added noise) the parameters of R matrix are directly known. In case of real measurement, these parameters are identified i.e. by laboratory testing.

4.1 Attitude Error Analysis

Attitude dynamics is given by following equation [3]:

$$\dot{C}_b^n = C_b^n (\omega_{nb}^b \times) = C_b^n (\omega_{ib}^b \times) - C_b^n (C \omega_{in}^b \times) \quad (4.13)$$

The application of the derivative of equation 4.4 to equal the equation 4.13 gives [3]:

$$(I - E^n) \dot{C}_b^n - \dot{E}^n C_b^n = (I - E^n) C_b^n \left[(\omega_{ib}^b \times) - (\omega_{in}^b \times) + \delta(\omega_{ib}^b \times) - \delta(\omega_{in}^b \times) \right] \quad (4.14)$$

$$\begin{aligned}
&= (I - E^n)C_b^n \left[(\omega_{ib}^b \times) - (\omega_{in}^b \times) \right] + (I - E^n)C_b^n \left[\delta(\omega_{ib}^b \times) - \delta(\omega_{in}^b \times) \right] \\
&= (I - E^n)C_b^n (\omega_{nb}^b \times) + (I - E^n)C_b^n \left[\delta(\omega_{ib}^b \times) - \delta(\omega_{in}^b \times) \right]
\end{aligned}$$

Thus, result after reduction is:

$$-E^n C_b^n = (I - E^n)C_b^n \left[\delta(\omega_{ib}^b \times) - \delta(\omega_{in}^b \times) \right] \quad (4.15)$$

$$E^n = -C_b^n \left[\delta(\omega_{ib}^b \times) - \delta(\omega_{in}^b \times) \right] \quad (4.16)$$

Further, error equation for $\delta(\omega_{in}^b \times)$ is derived [3]:

$$\omega_{in}^b + \delta \omega_{in}^b = C_b^n (I + E^n) (\omega_{in}^n + \delta \omega_{in}^n) \quad (4.17)$$

This can be rewritten as:

$$\delta \omega_{in}^b = C_b^n (\delta \omega_{in}^n + E^n \omega_{in}^n) \quad (4.18)$$

Thus, time derivation of attitude error can be expressed as [3]:

$$\dot{\varepsilon}^n = \delta \omega_{in}^n - (\omega_{in}^n \times) \varepsilon^n - C_b^n \delta \omega_{ib}^b \quad (4.19)$$

Attitude error dynamics is given by the following equation [3]:

$$\dot{\varepsilon}^n = F_{12} \delta v^n + F_{13} \delta v^n - (\omega_{in}^n \times) \varepsilon^n - C_b^n \delta \omega_{ib}^b \quad (4.20)$$

Biases are assumed to be constant:

$$\dot{b}_a = 0 \quad (4.21)$$

$$\dot{b}_g = 0 \quad (4.22)$$

Submatrix F_{12} of system matrix is the following 3×3 matrix:

$$F_{12} = \begin{pmatrix} 0 & F_{1212} & 0 \\ F_{1221} & 0 & 0 \\ 0 & F_{1232} & 0 \end{pmatrix} \quad (4.23)$$

Where single submatrices are:

$$F_{1212} = \frac{-1}{R_E + h} \quad (4.24)$$

$$F_{1221} = \frac{1}{R_N + h} \quad (4.25)$$

$$F_{1232} = \frac{\tan \varphi}{R_E + h} \quad (4.26)$$

Submatrix F_{13} of system matrix is the following 3×3 matrix:

$$F_{13} = \begin{pmatrix} F_{1311} & 0 & F_{1313} \\ 0 & 0 & F_{1323} \\ F_{1331} & 0 & F_{1333} \end{pmatrix} \quad (4.27)$$

Where single submatrices are:

$$F_{1311} = -\omega_{ie} \sin \varphi \quad (4.28)$$

$$F_{1313} = \frac{v_E}{(R_E + h)^2} \quad (4.29)$$

$$F_{1323} = \frac{-v_N}{(R_N + h)^2} \quad (4.30)$$

$$F_{1331} = \omega_{ie} \cos \varphi + \frac{v_E}{(R_E + h) \cos^2 \varphi} \quad (4.31)$$

$$F_{1333} = \frac{-v_E \tan \varphi}{(R_E + h)^2} \quad (4.32)$$

4.2 Velocity Error Analysis

This derivation is based on [3] and [4]. Velocity dynamics equation is given by the following expression:

$$\dot{v}^n = C_b^n f^b - (2\omega_{ie}^n + \omega_{en}^n) \times v^n + g^n \quad (4.33)$$

The application of perturbation of 4.33 gives:

$$\delta \dot{v}^n + v^n = (I - E^n) C_b^n (f^b + \delta f^b) - \quad (4.34)$$

$$-(2\omega_{ie}^n + \omega_{en}^n + 2\delta \omega_{ie}^n + \delta \omega_{en}^n) \times (\delta v^n + v^n) + g^n + \delta g^n$$

Therefore, the reduction yields:

$$\begin{aligned}
\delta v^n &= -(2\delta\omega_{ie}^n + \delta\omega_{en}^n) \times v^n + \delta g^n - \\
&-(2\omega_{ie}^n + \omega_{en}^n) \times v^n - \varepsilon^n \times f^n + C_b^n \delta f^b \\
&= v^n \times (2\delta\omega_{ie}^n + \delta\omega_{en}^n) + \delta g^n - \\
&-(2\delta\omega_{ie}^n + \delta\omega_{en}^n) \times \delta v^n + f^n \times \varepsilon^n + C_b^n \delta f^b
\end{aligned} \tag{4.35}$$

With respect to the following expressions

$$\omega_{ie}^n = \begin{pmatrix} \omega_e \cos \varphi \\ 0 \\ -\omega_e \sin \varphi \end{pmatrix} \tag{4.36}$$

$$\omega_{en}^n = \begin{pmatrix} v_E/R_E + h \\ -v_N/R_N + h \\ \frac{-v_E \tan \varphi}{R_E + h} \end{pmatrix} \tag{4.37}$$

The application of perturbation to sum $2\omega_{ie}^n + \omega_{en}^n$ gives:

$$2\delta\omega_{ie}^n + \delta\omega_{en}^n = \delta\Omega_v \delta v^n + \delta\Omega_r \delta r^n \tag{4.38}$$

where

$$\delta\Omega_v = \begin{pmatrix} 0 & 1/(R_E + h) & 0 \\ -1/(R_N + h) & 0 & 0 \\ 0 & -\tan \varphi / (R_E + h) & 0 \end{pmatrix} \tag{4.39}$$

$$\delta\Omega_r = \begin{pmatrix} -2\omega_e \sin \varphi & 0 & -v_E / (R_E + h)^2 \\ -2\omega_e \cos \varphi - [v_E / ((R_E + h) \cos^2 \varphi)] & 0 & v_N / (R_N + h)^2 \\ 0 & v_E \tan \varphi / (R_E + h)^2 & 0 \end{pmatrix} \tag{4.40}$$

From 4.35 and 4.38 the velocity error function is derived:

$$\begin{aligned}
v^n \times (2\delta\omega_{ie}^n + \delta\omega_{en}^n) &= (v^n \times) (\delta\Omega_r \delta r^n + \delta\Omega_v \delta v^n) \\
&= (v^n \times) \delta\Omega_r \delta r^n + (v^n \times) \delta\Omega_v \delta v^n
\end{aligned} \tag{4.41}$$

The next step is the developed multiplication:

$$({}^n v \times) \delta \Omega_r = \begin{pmatrix} -2v_E \omega_e \cos \varphi - \frac{v_E^2}{(R_E+h) \cos^2 \varphi} & 0 & \frac{-v_N v_D}{(R_N+h)^2} + \frac{v_E^2 \tan \varphi}{(R_M+h)^2} \\ 2\omega_e (v_N \cos \varphi - v_D \sin \varphi) + \frac{v_E v_N}{(R_E+h) \cos^2 \varphi} & 0 & \frac{-v_E v_D}{(R_N+h)^2} - \frac{v_N v_E \tan \varphi}{(R_N+h)^2} \\ 2v_E \omega_e \sin \varphi & 0 & \frac{v_E^2}{(R_E+h)^2} + \frac{v_N^2}{(R_N+h)^2} \end{pmatrix} \quad (4.42)$$

$$({}^n v \times) \delta \Omega_v = \begin{pmatrix} \frac{v_D}{R_N+h} & \frac{-v_E}{R_E+h} & 0 \\ 0 & \frac{v_D}{R_E+h} + \frac{v_N \tan \varphi}{R_E+h} & 0 \\ \frac{-v_N}{R_N+h} & \frac{-v_E}{R_E+h} & 0 \end{pmatrix} \quad (4.43)$$

Using previous equations the velocity error dynamics equation 4.33 can be reformulated:

$$\delta v^n = F_{21} \varepsilon^n + F_{22} \delta v^n + F_{23} \delta r^n + C_b^n \delta f^b \quad (4.44)$$

where single submatrices of matrix F are following:

$$F_{21} = C_b^n f^b \times = f^n \times \quad (4.45)$$

$$F_{22} = \begin{pmatrix} F_{2211} & F_{2212} & F_{2213} \\ F_{2221} & F_{2222} & F_{2223} \\ F_{2231} & F_{2232} & 0 \end{pmatrix} \quad (4.46)$$

$$F_{2211} = \frac{v_D}{R_N+h} \quad (4.47)$$

$$F_{2212} = -2\omega_{ie} \sin \varphi - 2 \frac{v_E \tan \varphi}{R_E+h} \quad (4.48)$$

$$F_{2213} = \frac{v_N}{R_N+h} \quad (4.49)$$

$$F_{2221} = 2\omega_{ie} \sin \varphi + \frac{v_E \tan \varphi}{R_E+h} \quad (4.50)$$

$$F_{2222} = \frac{v_N \tan \varphi + v_D}{R_E+h} \quad (4.51)$$

$$F_{2223} = 2\omega_{ie} \cos \varphi + \frac{v_E}{R_E+h} \quad (4.52)$$

$$F_{2231} = -2 \frac{v_N}{R_N+h} \quad (4.53)$$

$$F_{2232} = -2\omega_{ie} \cos \varphi - 2\frac{v_E}{R_E + h} \quad (4.54)$$

Submatrix F_{23} has the following form:

$$F_{23} = \begin{pmatrix} F_{2311} & 0 & F_{2313} \\ F_{2321} & 0 & F_{2323} \\ F_{2331} & 0 & F_{2333} \end{pmatrix} \quad (4.55)$$

$$F_{2311} = -2v_E \omega_{ie} \cos \varphi - \frac{v_E^2}{(R_E + h) \cos^2 \varphi} \quad (4.56)$$

$$F_{2313} = \frac{v_E^2 \tan \varphi}{(R_E + h)^2} - \frac{v_N v_D}{(R_N + h)^2} \quad (4.57)$$

$$F_{2321} = \frac{v_N v_E}{(R_E + h) \cos^2 \varphi} + 2v_N \omega_{ie} \cos \varphi - 2v_D \omega_{ie} \sin \varphi \quad (4.58)$$

$$F_{2323} = -\frac{v_N v_E \tan \varphi - v_E v_D}{(R_E + h)^2} \quad (4.59)$$

$$F_{2331} = 2v_E \omega_{ie} \sin \varphi \quad (4.60)$$

$$F_{2333} = \frac{v_E^2}{(R_E + h)^2} + \frac{v_N^2}{(R_N + h)^2} - 2\frac{g_0}{r_g} \quad (4.61)$$

where g_0 is given by equation 2.38 and geocentric radius at the surface r_g is given by:

$$r_g = R_E \sqrt{\cos^2 \varphi + (1 - e^2)^2 \sin^2 \varphi} \quad (4.62)$$

4.3 Position Error Analysis

This derivation is based on [3] and [4]. The development of position error dynamics is based on time derivative of the position vector:

$$\dot{r}^n = \begin{pmatrix} \dot{\varphi} \\ \dot{\lambda} \\ \dot{h} \end{pmatrix} = \begin{pmatrix} \frac{1}{R_N + h} & 0 & 0 \\ 0 & \frac{1}{(R_E + h) \cos \varphi} & 0 \\ 0 & 0 & -1 \end{pmatrix} \quad (4.63)$$

Position error dynamics is given by:

$$\delta \dot{r}^n = F_{32} \delta v^n + F_{33} \delta r^n \quad (4.64)$$

where

$$\begin{aligned}
F_{32} &= \begin{pmatrix} \frac{\partial \phi}{\partial v_N} & \frac{\partial \phi}{\partial v_E} & \frac{\partial \phi}{\partial v_D} \\ \frac{\partial \lambda}{\partial v_N} & \frac{\partial \lambda}{\partial v_E} & \frac{\partial \lambda}{\partial v_D} \\ \frac{\partial h}{\partial v_N} & \frac{\partial h}{\partial v_E} & \frac{\partial h}{\partial v_D} \end{pmatrix} = \\
&= \begin{pmatrix} F_{3211} & 0 & 0 \\ 0 & F_{3222} & 0 \\ 0 & 0 & -1 \end{pmatrix} \tag{4.65}
\end{aligned}$$

where elements of submatrix F_{32} has the following form:

$$F_{3211} = \frac{1}{R_N + h} \tag{4.66}$$

$$F_{3222} = \frac{1}{(R_E + h) \cos \varphi} \tag{4.67}$$

Matrix F_{33} is given by:

$$\begin{aligned}
F_{33} &= \begin{pmatrix} \frac{\partial \phi}{\partial \varphi} & \frac{\partial \phi}{\partial \lambda} & \frac{\partial \phi}{\partial h} \\ \frac{\partial \lambda}{\partial \varphi} & \frac{\partial \lambda}{\partial \lambda} & \frac{\partial \lambda}{\partial h} \\ \frac{\partial h}{\partial \varphi} & \frac{\partial h}{\partial \lambda} & \frac{\partial h}{\partial h} \end{pmatrix} = \\
&= \begin{pmatrix} 0 & 0 & F_{3313} \\ F_{3321} & 0 & F_{2223} \\ 0 & 0 & 0 \end{pmatrix} \tag{4.68}
\end{aligned}$$

where elements of submatrix F_{33} has the following form:

$$F_{3313} = -\frac{v_N}{(R_N + h)^2} \tag{4.69}$$

$$F_{3321} = \frac{v_E \sin \varphi}{(R_E + h) \cos^2 \varphi} \tag{4.70}$$

$$F_{3323} = \frac{v_E}{(R_E + h)^2 \cos \varphi} \tag{4.71}$$

4.4 Applying corrections

The important finding about corrections is that corrections have to be applied inside the loop. This means that subtracting computed error have to be processed before delaying the value and sending it to next computations. Corrections to velocity, position and sensor outputs are simply subtracted from actual values. The more complicated is

situation in attitude. Correction to DCM can be processed as shown in equation 4.4. The second possibility implemented in my solution is applying correction to quaternion [2]:

$$\begin{bmatrix} \delta q_0 \\ \delta q_1 \\ \delta q_2 \\ \delta q_3 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} q_1 & q_2 & q_3 \\ -q_0 & -q_3 & q_2 \\ q_3 & -q_0 & -q_1 \\ -q_2 & q_1 & -q_0 \end{bmatrix} \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \end{bmatrix} \quad (4.72)$$

The tuning of EKF includes setting of initial values covariance matrix P (overall 15 values in order corresponding to state vector) and elements of matrix Q (6 values corresponding to standard deviations of IMU sensors noise) and analogously, elements of matrix R (6 values corresponding to standard deviations of the GPS noise). It was observed that increasing values of matrix Q leads to expecting higher noise in sensors and thus in constantly higher error. On the other hand underestimating of these values causes worse estimates in single steps, although average error is not higher. Overestimating of elements of matrix R did not cause measurable effects. Decreasing the values leads to significantly higher average error. The course of errors was steady.

The major influence has setting of last six elements of matrix P . These quantities are responsible for initial estimates of biases. Higher than optimal values causes over-estimated bias. Lower values decrease impact of corrections and cause that some noise is not eliminated.

Chapter 5

Achieved Results

The description of Simulink models and achieved performance in different configurations is discussed in this chapter. The subsystem Trajectory Generator was supplied by the company Honeywell International, s.r.o. It contains the following data: the sensor measurements (true values without noise) and true values of output quantities for simulation time more than 30,000 steps (corresponding to approx. 300 seconds when frequency 100 Hz for INS is used). Also necessary initial conditions were supplied.

5.1 Simulink model description

This section describes implemented Simulink models which are attached to this thesis. Two models are attached - the first is called SimpleEKF.mdl, the second model is FullEKF.mdl. The distinction is that the simple model estimates just corrections to velocity and position and the accelerometer biases. So, in this case gyros are not noisy and then there is no need to correct computed attitude angles. This simple EKF has nine states.

The fully implemented model computes also corrections to attitude and gyros biases. The reason to do development in these two steps is that the simple model is significantly easier to implement and tune. As can be seen at the results, the simplified model computes very well and has no problems with stability. The full model is much more complex and has slight problems with stability. Moreover the tuning of this model to get the maximum performance was more demanding.

The models contain five main subsystems. The first of them is called Trajectory Generator and among others contains data from sensors and true values of output quantities. The next subsystem, Correction and initial conditions, consists of computed state vector transferred to correction vector. The transfer is done by Accumulator functions. These functions remove the constant part of the error. Further, initial conditions for velocity, position and attitude are established in this part of the model. So, the state

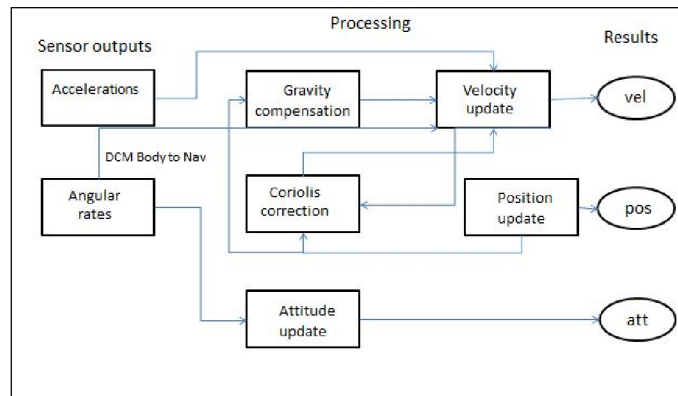


Figure 5.1: Schematic of my INS solution

vector is the only input to this subsystem, and the correction vector and vector of initial conditions are outputs.

The third big subsystem is INS - mechanization of inertial navigation system. Inputs are correction vector, vector of initial conditions, reset signal and sensor outputs. The INS subsystem is divided into next five subsystems. The Attitude update computes attitude angles and DCM matrix from body to navigation frame. The implemented equations are described in section 2.3.3 of this thesis. The Gravity subsystem provides actual value of gravity vector computed from position value in the last step. The calculation procedure is done by section 2.3.4. The Coriolis correction subsystem is responsible for computing Coriolis force correction to use in velocity update. Next outputs are actual values of Earth's radii at given position and angular rate ω_{in}^n . These quantities are needed further in EKF algorithm. The Velocity update subsystem computes velocity and accelerometer output in navigation frame (also needed further). The equations are described in section 2.3.1. The Position update subsystem gives a position output. My developed model of INS is briefly outlined at Figure 5.1.

The key part of this model is EKF subsystem where corrections to INS are computed. GPS outputs are created in the subsystem Measure vector by adding noise to true values of velocity and position. Vector of measurement is then obtained by subtracting simulated GPS outputs from INS values. The Matlab m-function EKF is algorithm where single matrices are computed and then own EKF algorithm is processed. Outputs from EKF algorithm are correction vector and covariance matrix. Values of correction vector are zeroed at every step. The setting of initial values of covariance matrix P is done in the attached m-file. The m-file also reads trajectory data from directory. The brief schematic of my EKF operation is presented at Figure 5.2.

The fifth subsystem Display Results compares computed data to true values of navigation quantities and displays them in graphs.

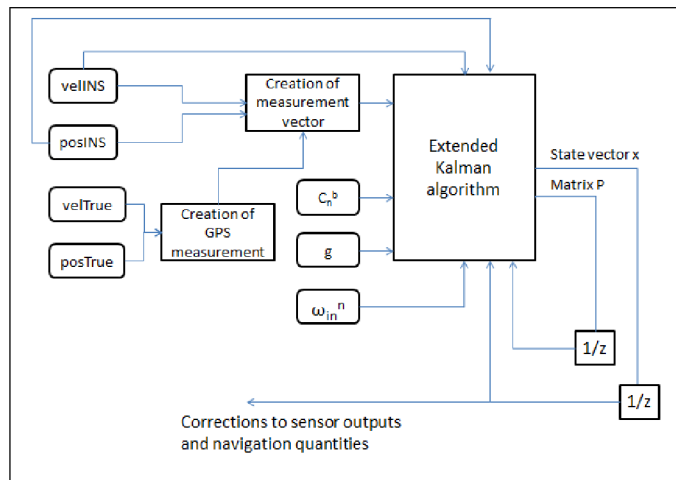


Figure 5.2: Schematic of my EKF architecture

5.2 Results

True values of position, velocity and attitude are described at Figure 5.1 to 5.3. The first step was the creation of inertial navigation system with use of ideal sensors. The performance of the system was observed by comparing to the true values. The outputs from the comparison between developed INS with simulated ideal sensors and true values are described at the Figure 5.4 (position), Figure 5.5 (velocity) and Figure 5.6 (attitude). These errors are given just by internal proceeding of INS. The next step was adding noises to the accelerometers and gyroscopes. The values of added noise was chosen to match common low-cost MEMS sensors. Specifically, parametres of added noises are defined in Table 5.1. Results from INS with noisy accelerometers are presented at the Figure 5.7 (position) and 5.8 (velocity). As can be seen, adding noises to the sensors results in a very rapid increase in error. After 300 seconds, the position error is more than 3,000 metres in altitude. Results from INS with noisy accelerometers and gyroscopes are displayed at the Figure 5.9 to 5.11.

After the succesful development of INS mechanization, GPS measurements were created. To simulate GPS, noises were added to the true values of position and velocity. Values of added noise were chosen to match the GPS accuracy approximately. The added noise has parametres as specified in Table 5.2 and the set GPS precision can be seen at Figure 5.12 for position and Figure 5.13 for velocity. Seeds were set to be not correlated to approximate simulated measurements to real GPS. The next step was the development of simplified EKF estimating only errors in velocity, position and accelerometer bias (gyros were operating without added noise in that case). This simplified EKF state vector has nine elements (correction to attitude and gyros missing). The most demanding task was the tuning of EKF to provide best results. This tuning

Quantity	Noise Type	Mean	Variance	Seed
GPS Latitude	random number	0	1e-12	55
GPS Longitude	random number	0	1e-12	45
GPS Altitude	random number	0	2,5	60
GPS Velocity North	random number	0	0,001	23
GPS Velocity East	random number	0	0,001	15
GPS Velocity Down	random number	0	0,001	30

Table 5.1: GPS noise parameters

Sensor	Type	Mean	Size	Seed
accX	random	0	0.001	55
accY	random	0	0.001	45
accZ	random	0	0.001	60
gyrX	random	0	4.3e-5	55
gyrY	random	0	4.3e-5	45
gyrZ	random	0	4.3e-5	60

Table 5.2: Sensor noise parameters

is done mainly by the setting of covariance matrix P initial values. The covariance matrix is a matrix of size 9×9 where the matrix element on the diagonal corresponds to the order of state vector elements. During the tuning of EKF it was observed that the values corresponding to sensor biases influence the performance of the system most significantly. Results from this reduced EKF model are displayed at Figure 5.14 for position and at Figure 5.15 for velocity. Attitude errors are not displayed because noise was not added to gyroscope in this model.

When this simplified EKF provided satisfactory results, the system was extended to estimate also corrections to attitude angles and gyroscopes outputs. This step implied the enlargement of state vector, F , G and H matrices and the re-tuning of the entire system. The change of covariance matrix P initial values proved to be insufficient, so the tuning of the system has required also to move Q and R matrix elements. Although a lot of time was spent on EKF tuning, some small issues with stability (mainly in velocity as can be seen in graphs) remained. Nevertheless, the achieved performance of the system is in line with expectations as can be seen at Figure 5.16 (position), 5.17 (velocity) and 5.18 (attitude).

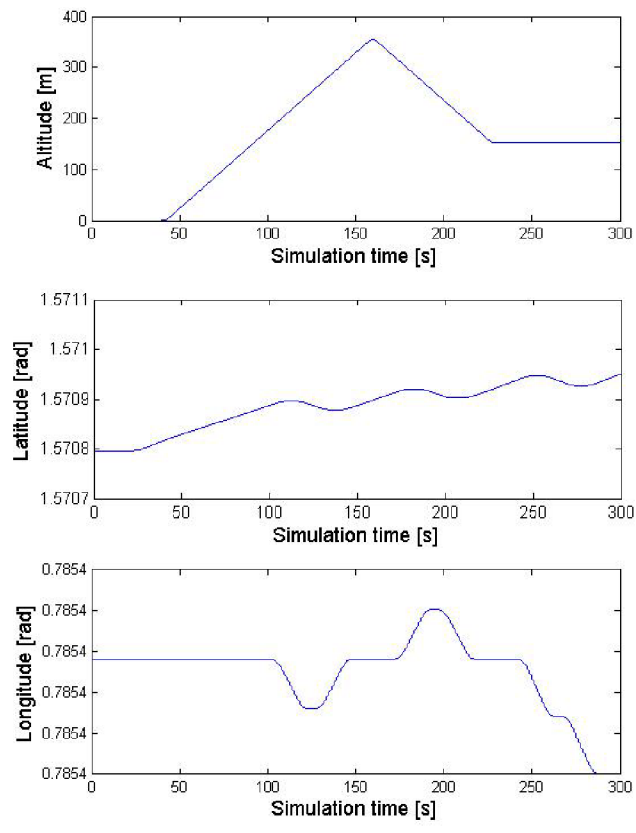


Figure 5.3: True values of position

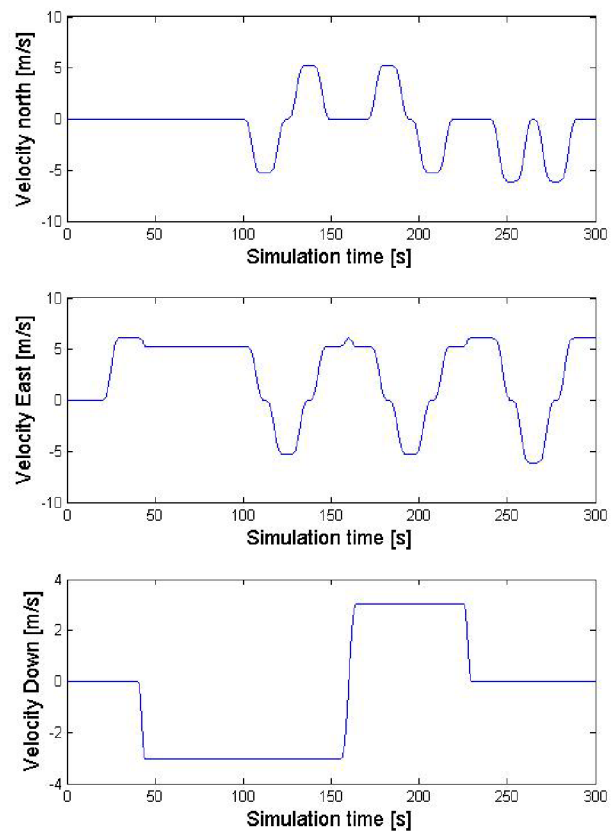


Figure 5.4: True values of velocity

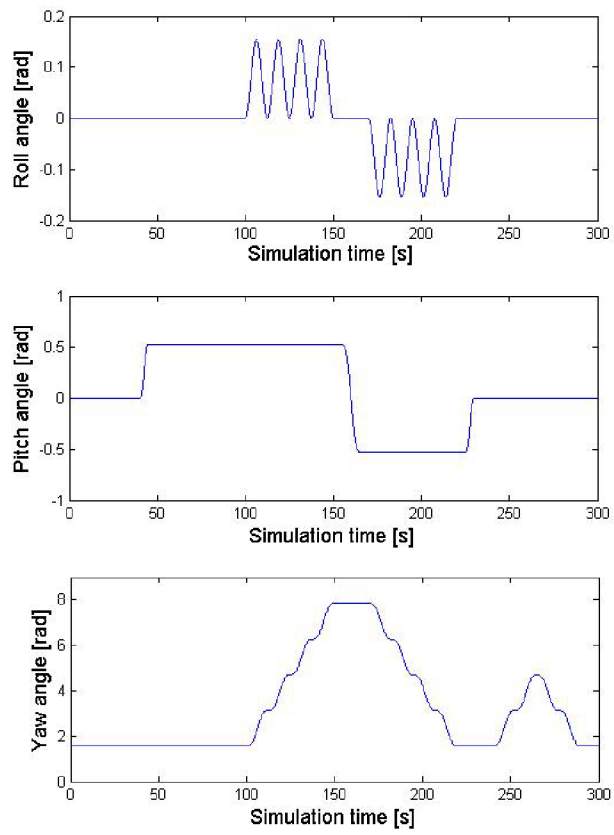


Figure 5.5: True values of attitude

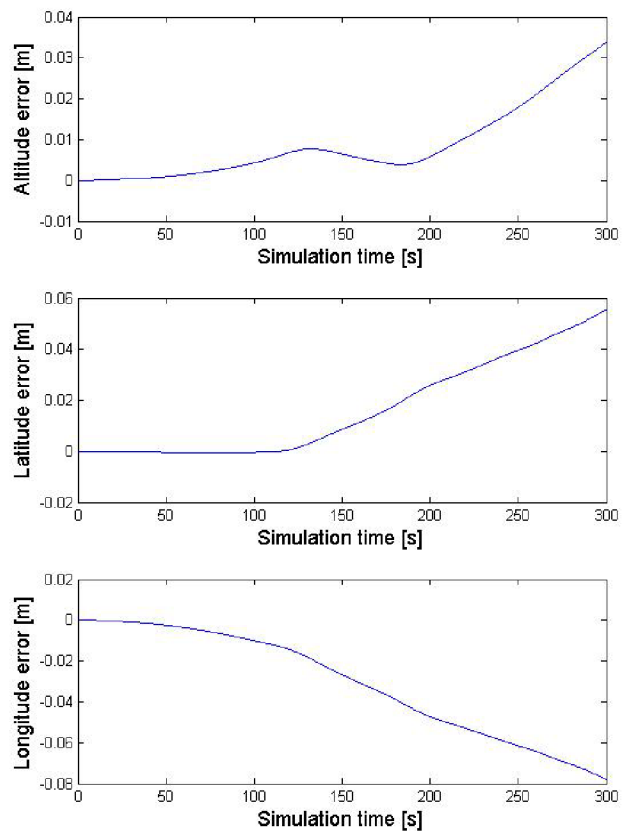


Figure 5.6: Position errors from INS with ideal sensors

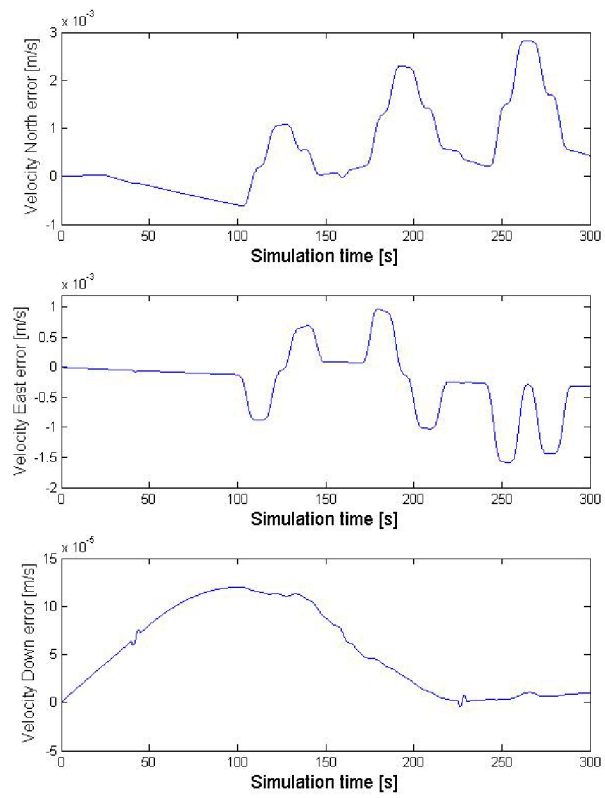


Figure 5.7: Velocity errors from INS with ideal sensors

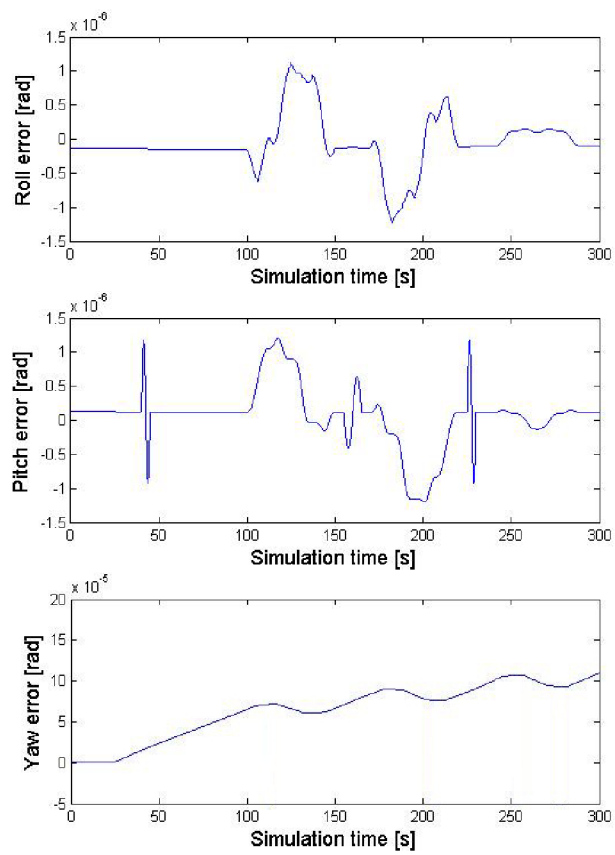


Figure 5.8: Attitude errors from INS with ideal sensors

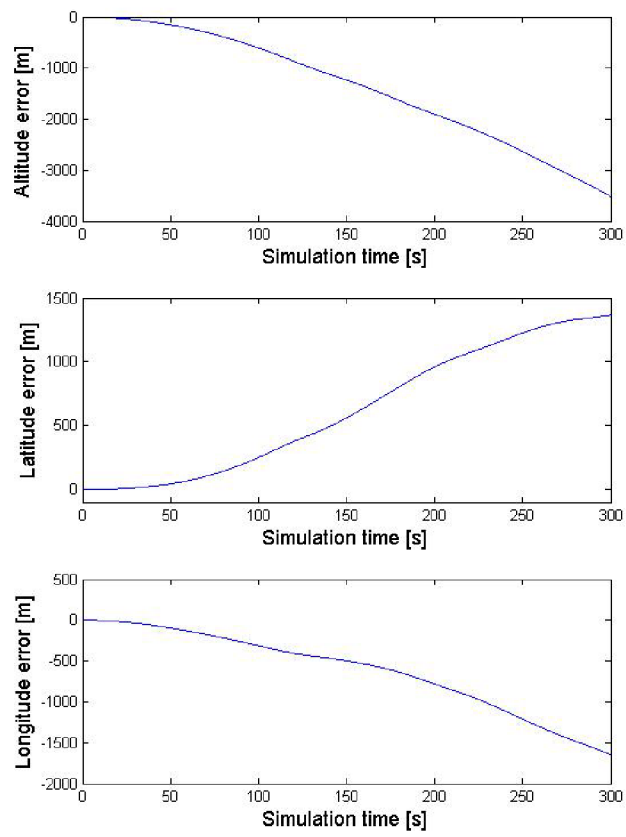


Figure 5.9: Position errors from INS with noisy accelerometers

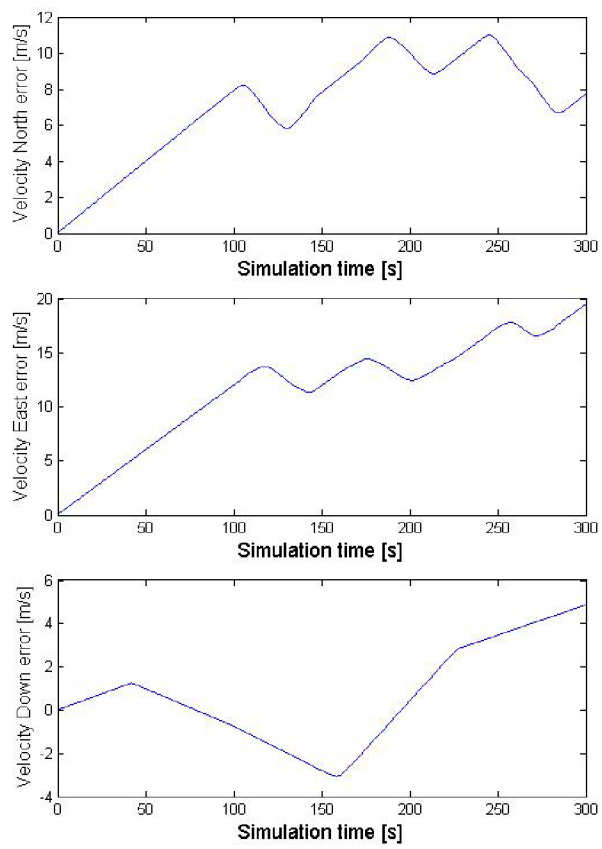


Figure 5.10: Velocity errors from INS with noisy accelerometers

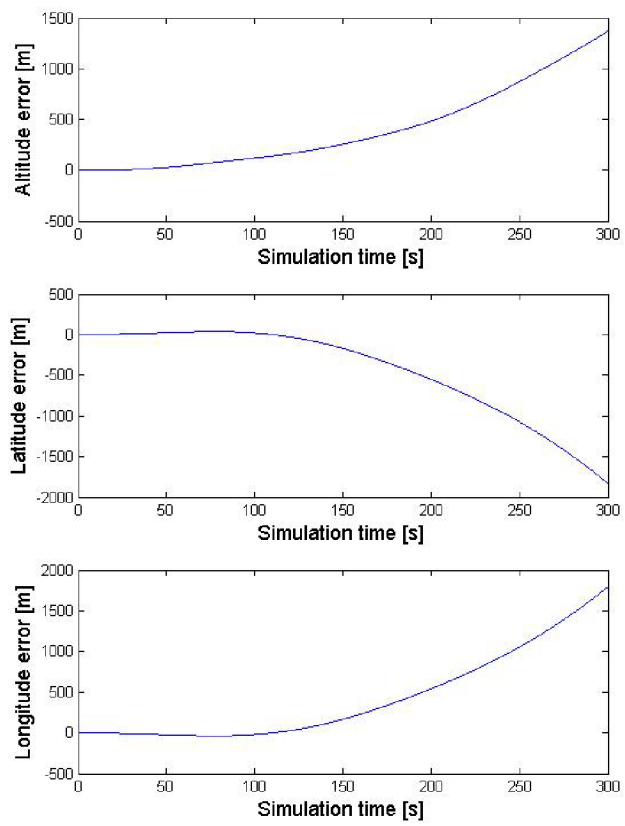


Figure 5.11: Position errors from INS with noisy sensors

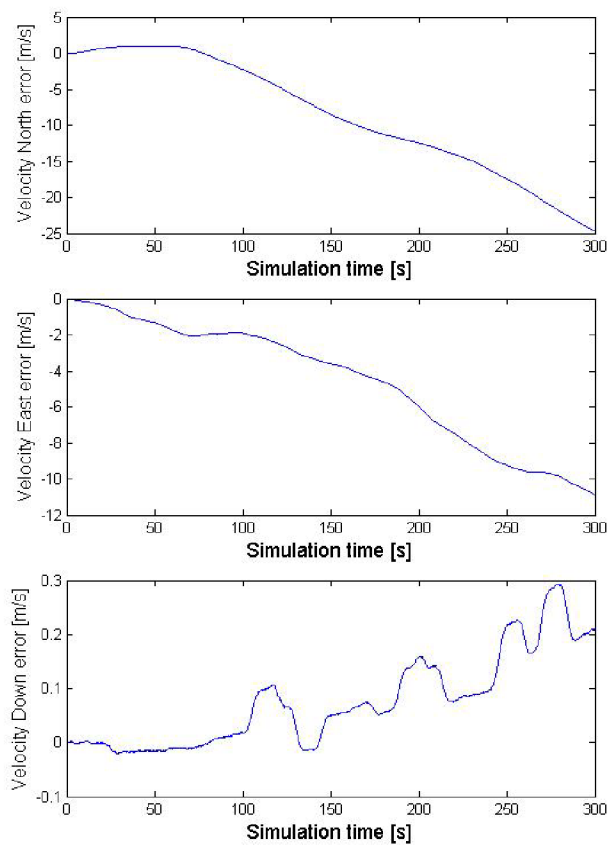


Figure 5.12: Velocity errors from INS with noisy sensors

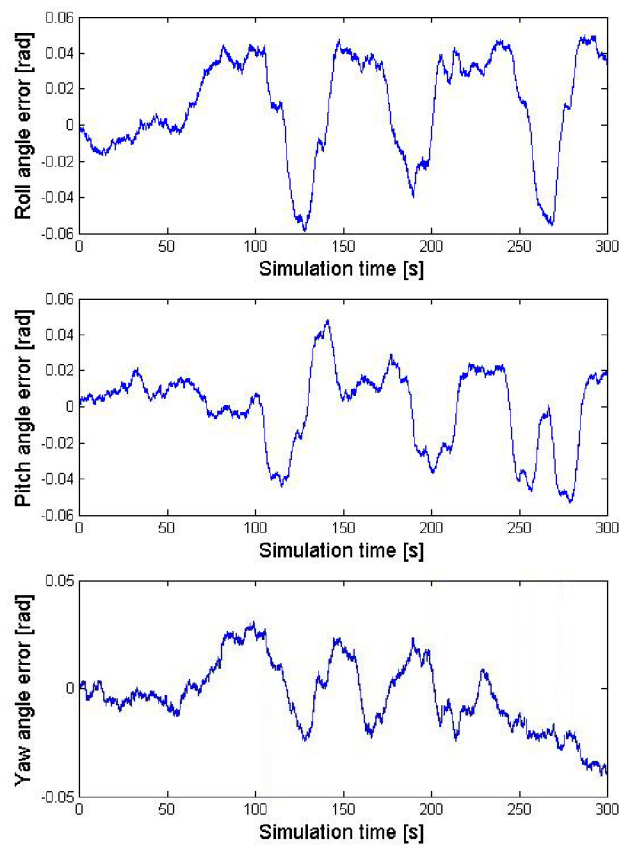


Figure 5.13: Attitude errors from noisy sensors

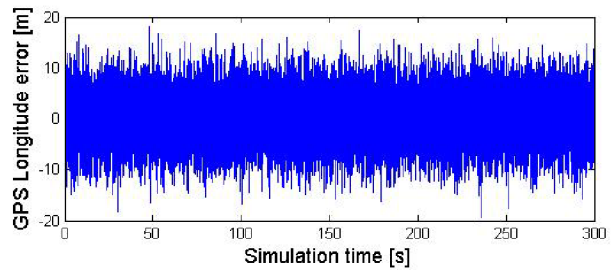
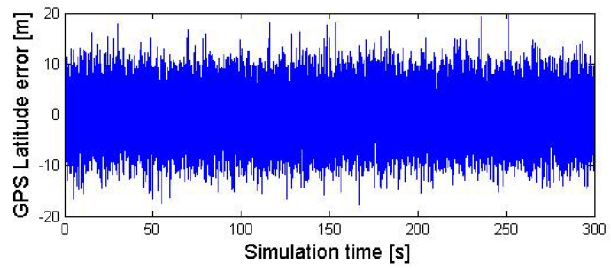
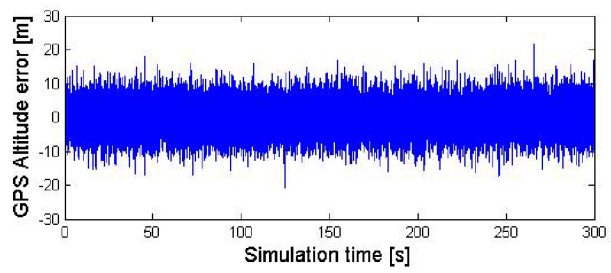


Figure 5.14: GPS Position errors

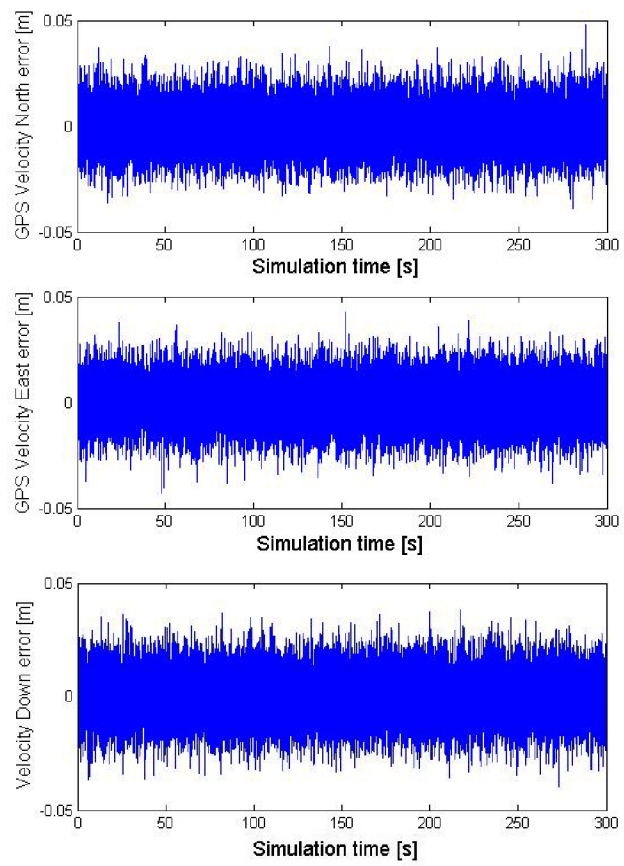


Figure 5.15: GPS Velocity errors

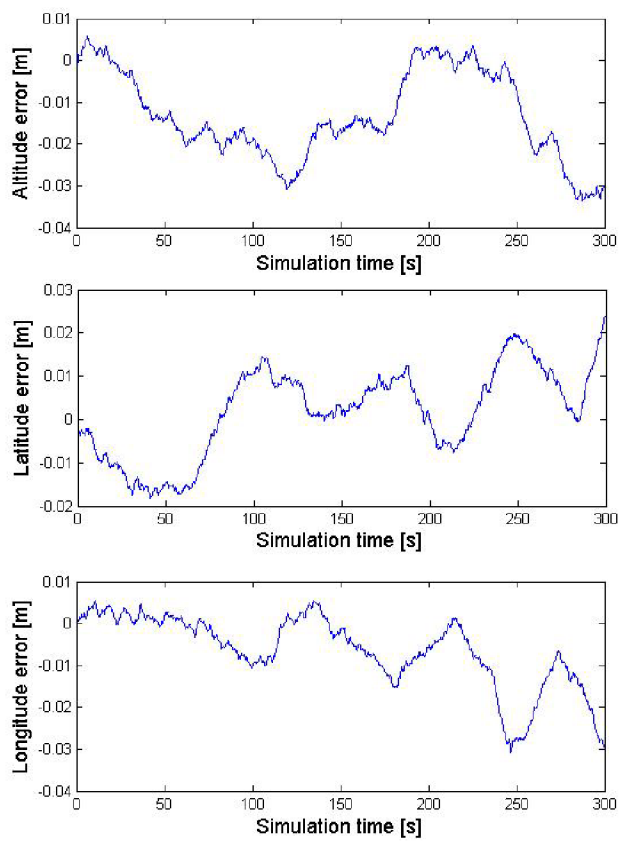


Figure 5.16: Simple EKF Position errors (noise only in accelerometers)

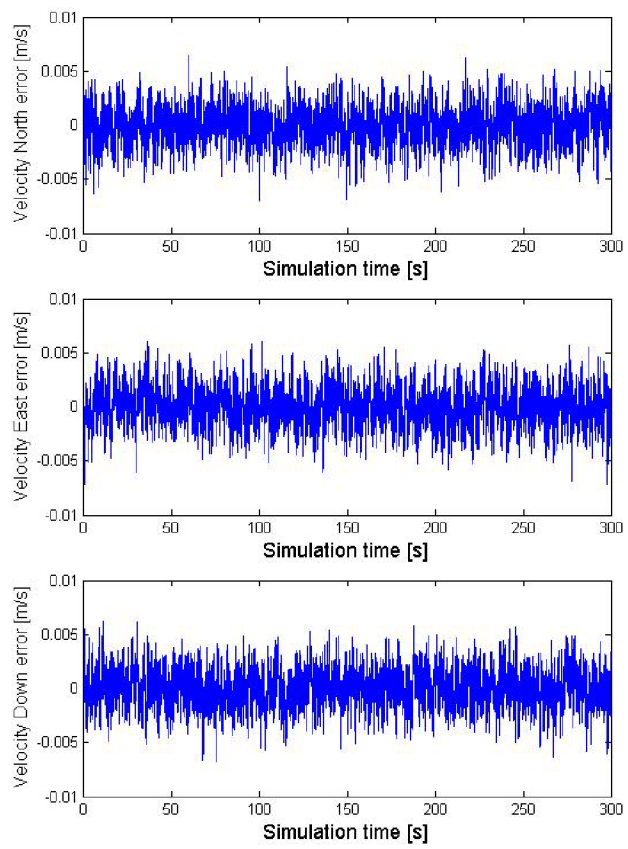


Figure 5.17: Simple EKF Velocity errors (noise only in accelerometers)

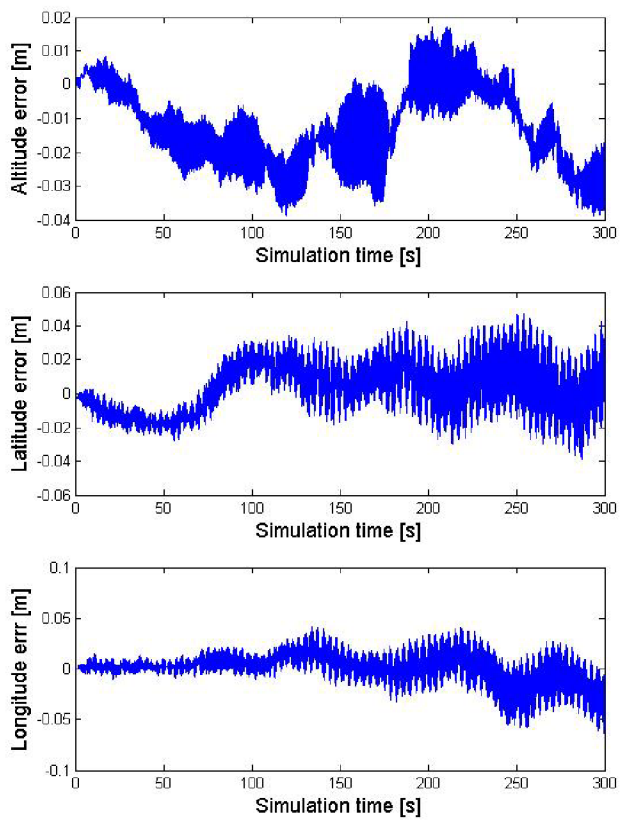


Figure 5.18: Full EKF Position errors

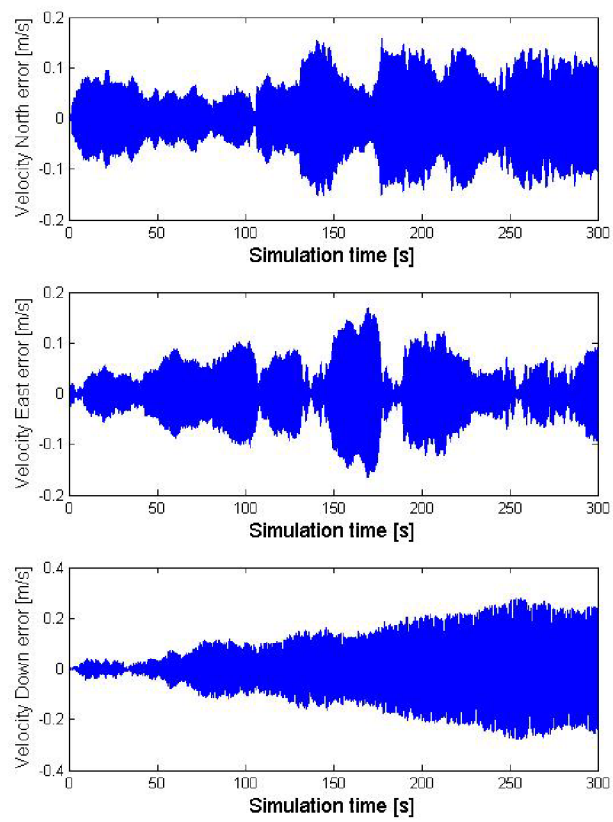


Figure 5.19: Full EKF Velocity errors

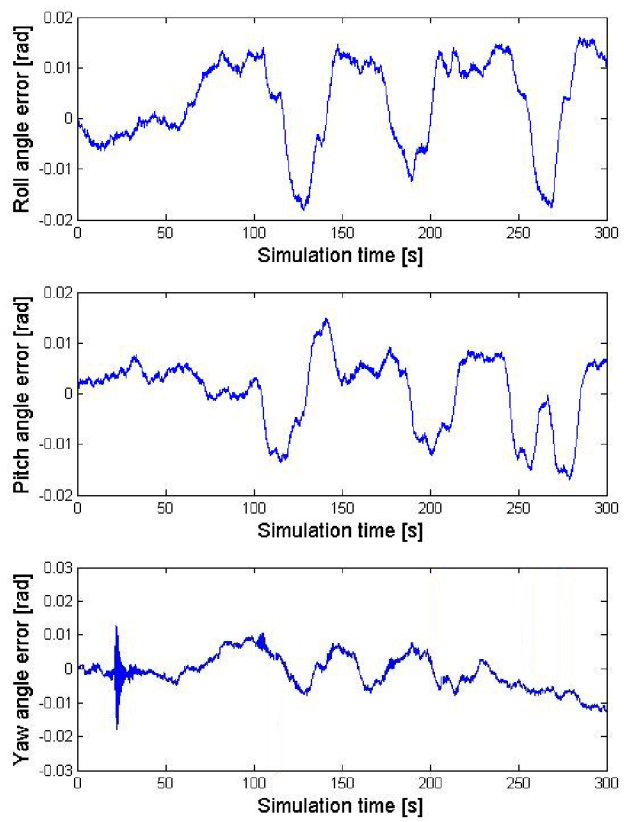


Figure 5.20: Full EKF Attitude errors

Chapter 6

Conclusion

6.1 Summary

The main objective of this thesis is the development of the fully functional algorithm for the inertial navigation system and for the sensor fusion method via Extended Kalman Filter. The parameters of the developed system (choosing INS error model, loosely coupled integration schema etc.) were specified by the assigning company. This thesis presents the necessary knowledge to build the system from INS up to complete solution providing the corrected navigation solution in every step.

Chapter 1 presents the brief current status of problematics with references to actual available sources. INS mechanization equations are presented in Chapter 2, including transformation equations and sensor errors outline. Chapter 3 briefly describes Global Positioning System and presents different ways of integration schemas with emphasis on the Extended Kalman filter. One of the most important knowledge is the content of Chapter 4 where the EKF algorithm equations are developed. Chapter 5 displays the achieved results and the last chapter provides the final summary.

The major objective was to develop a low-cost INS/GPS navigation system. The research led to following contributions:

1. The discrete form of INS mechanization equations was developed using Laplace and bilinear transformation and successfully implemented
2. The error analysis of low-cost MEMS sensors was done via Allan Variance to understand sensor behavior
3. The loosely coupled INS/GPS integration algorithm was developed by the use of fifteen-state Extended Kalman Filter
4. The knowledge about designing and tuning EKF was acquired

6.2 Future work recommendations

The further work based on theme of this thesis can involve:

- dealing with different operational frequencies of fused systems (more information can be found in [17])
- comparing different ways to deal with system non-linearity (e.g. Extended Kalman Filter versus Unscented Kalman Filter)
- research and development in the area of automatic tuning methods (Monte Carlo simulation or similar)
- more detailed model of sensor errors (algorithm considering bias, scale factor errors and nonorthogonalities is presented e.g. in [3])
- more accurate evaluating algorithms (different dynamics of trajectories, simulation of GPS outages, experimental car tests etc.)

6.3 Conclusion

The thesis provides the complete solution to implement Extended Kalman Filter for INS and GPS data fusion. During implementation following conclusions were deduced:

1. To ensure correct function of EKF algorithm, proper way to apply corrections back to INS is crucial. The corrections have to be applied inside the loop to apply the corrections before values are delayed and used for next computations.
2. The tuning of EKF is a question of setting proper initial values of covariance matrices P , Q and R . This includes overall 27 values to be set. The major impact has the setting of initial values of covariance matrix P corresponding to biases of sensors (last six numbers). Higher than optimal values cause overestimated bias. Lower values decrease impact of corrections and cause that some noise is not eliminated.
3. The error state EKF algorithm computes corrections at every step. The important fact is that the state vector has to be zeroed at every step.
4. Corrections applied to sensor outputs (computed biases) have to be accumulated¹ in every step to ensure correct processing.

¹Current and previous values are added

5. Simple EKF model estimating position, velocity and accelerometer errors reduces the following errors achieved after 300 seconds: The error about 3,000 metres in position (altitude) and 20 m/s in velocity (east direction) is reduced to 0.03 metres in position and 0.005 m/s in velocity. Accuracy of GPS which is used as a measurement fed into EKF is about ten metres in position and 0.03 m/s in velocity.
6. The full EKF model estimating moreover attitude and gyro error has the following results after 300 seconds: The position error is reduced from almost 2,000 metres in latitude to 0.05 metres and the velocity error is reduced from about 25 metres per second in north direction to 0.1 m/s. The attitude error values 0.06 radians have been improved to one third of the noncompensated error. The resulting accuracy in attitude is up to 0.02 radians which corresponds to 1.15 degrees. This accuracy in all parameters is sufficient, but the error behavior in time exhibits a mild instability. This is most probably caused by the tuning which needs more sophisticated methods exceeding the format of master's thesis.

References

1. Groves, Paul. D.: Principles of GNSS, Inertial and Multisensor Integrated Navigation Systems. Boston, USA: Artech House, 2008. 518 p. ISBN-13: 978-1-58053-255-6
2. Rogers, Robert M.: Applied Mathematics in Integrated Navigation Systems. Third Edition. Reston, USA: American Institute of Aeronautics and Astronautics, 2007. 408 p. ISBN-13: 978-1-56347-927-4
3. Shin, Eun-Hwan: Estimation Techniques for Low-Cost Inertial Navigation. Calgary, Canada: University of Calgary, Department of Geomatics Engineering. 2005. 206 p. Available from:
<<https://geomatics.ucalgary.ca/research/publications/GradTheses.html>>
4. Shin, Eun-Hwan: Accuracy Improvement of Low-Cost INS/GPS for Land Application. Calgary, Canada: University of Calgary, Department of Geomatics Engineering, 2001. 137 p. Available from:
<<https://geomatics.ucalgary.ca/research/publications/GradTheses.html>>
5. Farrell, Jay. Barth, Matthew: The Global Positioning System and Inertial Navigation. New York, USA: McGraw-Hill Companies, Inc, 1999. 340 p. ISBN 0-07-022045-X
6. Novák, Jakub: Snímání a zpracování údajů lokalizace dopravního prostředku. Brno: VUT v Brně, Fakulta strojního inženýrství, 2006. 65 p.
7. Horemuž, Milan: Integrated Navigation. Stockholm, Royal Institute of Technology, Division of Geodesy, 2006. 107 p. Available from:
<<http://gidec.abe.kth.se/files/2011KTH/Course%20materials/Milan%20Horemuz,%20Integrated%20Navigation.pdf>>
8. Gebre-Egziabher, Demoz: GPS, INS and GPS/INS fusion. Minnesota, USA: University of Minnesota, Department of Aerospace Engineering & Mechanics. 2005. Study material, archiv of autor.

9. Hrdina, Zdeněk. Pánek, Petr: Rádiové určování polohy (Družicový systém GPS). Praha: ČVUT v Praze, Fakulta elektrotechnická. 1995. 267 p. ISBN 80-01-01386-3
10. Stockwell, Walter: Bias Stability Measurement : Allan variance. Available from: <http://bullseye.xbow.com:81/Support/Support_pdf_files/Bias_Stability_Measurement.pdf>
11. Vagner, Martin: MEMS Gyroscope Performance Comparison Using Allan Variance Method. Brno: VUT v Brně, Fakulta elektrotechniky a komunikačních technologií. 2011. 5 p. Available from: <<http://www.feec.vutbr.cz/EEICT/2011/sbornik/03-Doktorske%20projekty/03-Kybernetika%20a%20automatizace/14-xvagne04.pdf>>
12. Yang, Yong: Tightly Coupled MEMS INS/GPS Integration with INS Aided Receiver Tracking Loops. Calgary, Canada: University of Calgary, Department of Geomatics Engineering, 2008. 205 p.
Available from: <<https://geomatics.ucalgary.ca/research/publications/GradTheses.html>>
13. Solimeno, Adriano: Low-cost INS/GPS Data Fusion with Extended Kalman Filter for Airborne Applications. Lisbon: Instituto Superior Tecnico. 2007. 163 p.
Available from: <<https://dspace.ist.utl.pt/bitstream/2295/126262/1/dissertacao.pdf>>
14. Havlena, Vladimír. Štecha, Jan: Moderní teorie řízení. Praha : ČVUT v Praze, Fakulta elektrotechnická. 2000. 297 p. ISBN 8001021653
15. Welch, Greg. Bishop, Gary: An Introduction to the Kalman Filter. Chapel Hill, USA: University of North Carolina, Department of Computer Science. 2006. 16 p. Available from: <http://www.cs.unc.edu/~welch/media/pdf/kalman_intro.pdf>
16. Šimandl, Miroslav: Odhad stavu stochastických systémů. Brno: Centrum pro rozvoj výzkumu pokročilých řídicích a senzorických technologií. 2010. 48 p.
Available from: <http://www.crr.vutbr.cz/system/files/brozura_07_1012.pdf>
17. Farrell, Jay A.: Aided Navigation: GPS with High Rate Sensors. New York, USA: McGraw-Hill Companies, Inc, 2008. 553 p. ISBN 0-07-149329-8

Appendix A - Content of attached CD

Directory Full EKF:

- FulleKF_2007a.mdl
- FulleKF_2010a.mdl
- FulleKF_init.m

Directory Simple EKF:

- SimpleEKF_2007a.mdl
- SimpleEKF_2010a.mdl
- SimpleEKF_init.m

Directory Trajectories

File ReadMe.txt

Appendix B - Discretization of INS Velocity Update

Velocity update equation in the continuous form has the following form:

$$\dot{v} = C_b^n f^b + g^n + (2\omega_{ie}^n + \omega_{en}^n) \times v^n$$

To avoid the vector product, the skew symmetric matrix is used:

$$A = (2\omega_{ie}^n + \omega_{en}^n) \times$$

The Laplace transform is applied:

$$sV(s) = C_b^n f^b(s) + g^n(s) + AV(s)$$

The Z transform is applied:

$$s = \frac{2}{T} \frac{1 - z^{-1}}{1 + z^{-1}}$$

$$\frac{2}{T} \frac{1 - z^{-1}}{1 + z^{-1}} V(z) = C_b^n F^b(z) + G(z) - AV(z)$$

$$\frac{2}{T} \left(V(z) - \frac{V(z)}{2} \right) = C_b^n F^b(z) + C_b^n \frac{F(z)}{2} + G(z) + \frac{G(z)}{2} - AV(z) - A \frac{V(z)}{2}$$

$$\frac{2}{T} (v_k - v_{k-1}) = C_b^n F_k^b + C_b^n F_{k-1}^b + G_k + G_{k-1} - Av_k - Av_{k-1}$$

$$\frac{2}{T} v_k + Av_k = C_b^n F_k^b + C_b^n F_{k-1}^b + G_k + G_{k-1} + \frac{2}{T} v_{k-1} - Av_{k-1}$$

$$\frac{2}{T} v_k \left(I + A \frac{T}{2} \right) = C_b^n F_k^b + C_b^n F_{k-1}^b + G_k + G_{k-1} + \left(I - A \frac{T}{2} \right) v_{k-1} \frac{2}{T}$$

$$v_k = \left(I + A \frac{T}{2} \right)^{-1} \left[\left(I - A \frac{T}{2} \right) v_{k-1} + \frac{T}{2} \left(C_b^n F_k^b + C_b^n F_{k-1}^b + G_k + G_{k-1} \right) \right]$$

Appendix C - Allan Variance

The Allan Variance is a method of measuring frequency stability, mainly of oscillators, clocks or amplifiers. But this method is suitable for simple long term stability analysis at sensors which exhibit non-white noise. The root mean square variation is a sufficient method just for white noise. When this method is applied to non-white noise, the root mean square variation gradually converges to infinity. David W. Allan developed Allan Variance (also known as two-sample variance) to enable convergence in any case.

The Allan Variance basic equation can be seen at 6.1.

$$AVAR^2(\tau) = \frac{1}{2(n(\tau) - 1)} \sum (y(\tau)_{i+1} - y(\tau)_i)^2 \quad (6.1)$$

The basic idea is to divide long sequence of data to bins based on an averaging time. Then the difference in average between successive bins are squared, adding them all up and divided by a rescaling factor . The square root of the result is a quantitative measure of how much the average value changed at that particular values of averaging time [10]. Then the procedure is repeated with increased averaging time τ . The result of the procedure is a graph called sigma-tau. The x-axis displays averaging times and the y-axis displays rate in degrees per time interval, usually at log-log scale. The sigma-tau graph provides characteristics of several basic noise types according to asymptotics properties. The left side of the graph, short averaging times, the Allan Variance is highest due to noise in sensor. The slope of Allan Variance is a characteristic of angle random walk. As the averaging time increases, the Allan Variance decreases up to the minimum point. From this point Allan Variance starts to increase again due to the rate random walk of the sensor [10]. The minimum point on the curve is the best stability which can be reached with fully modeled sensor and active bias estimation. The rate random walk parameters help predict the time scale at which the drift of sensor occurs [11].

The Allan Variance method was applied to IMU sensors iFOG-IMU-1-A of iMAR company. The static tests lasting more than 13 hours were done. The result for accelerometer and gyroscope in x-direction is displayed at Figure 6.1. The bias stability value from Allan Variance gives a better result than the manufacturer indicates.

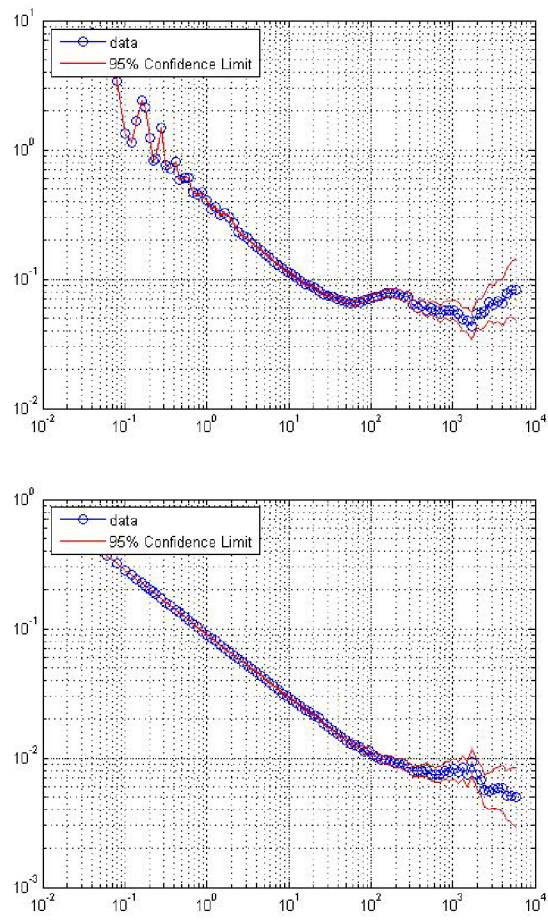


Figure 6.1: The Allan Variance on iFOG-IMU-1-A for accelerometer and gyroscope in x-axis