Performance Evaluation of Short Time Dead Reckoning for Navigation of an Autonomous Vehicle
Performance Evaluation of Short Time Dead Reckoning for Navigation of an Autonomous Vehicle

Examensarbete utfört i Reglerteknik vid Tekniska högskolan vid Linköpings universitet av

David Enberg

LiTH-ISY-EX--15/4826--SE

Supervisor: Michael Roth
ISY, Linköpings universitet
Peter Reigo
Flowscape AB

Examiner: Daniel Axehill
ISY, Linköpings universitet

Linköping, 18 mars 2015
Prestandautvärdering av Dödräkning för Navigering av Förarlöst Fordon
Performance Evaluation of Short Time Dead Reckoning for Navigation of an Autonomous Vehicle

David Enberg

Utilizing a Global Navigation Satellite System (GNSS) together with an Inertial Navigation System (INS) is today a common integration method to obtain a positioning solution for autonomous systems. Both GNSS and INS have benefits and weaknesses where the best parts from both systems can be combined with a Kalman filter. Because of this complementary nature, it is of interest to look at the robustness of the positioning solution when the Global Navigation Satellite System is temporarily not available.

The aim of the thesis has been to investigate different vehicle models and to evaluate their short-time performance using a Dead Reckoning approach. The goal has been to develop a system for a Heavy Duty Vehicle (HDV) and to find out for which time interval a specific model can stay within a certain range when the GNSS is lost. A GNSS outage could for example happen when driving on a highway and passing signs, bridges and especially when driving inside tunnels. Also, for a solution to become commercially interesting, it must be cheap. Therefore, it is common to use so called Micro-Electro-Mechanical-Systems (MEMS) sensors which are of low-cost but suffer from biases, scale factors and temperature dependencies which must be compensated for.

The results from the tests show that some models are able to estimate the position with good precision during short time GNSS outages whereas other models do not deliver the required accuracy.

The main conclusion is that care should be taken when choosing the vehicle model so that it fits its usage area and the complexity needed to describe its motion. There are also lots of parameters to look at when investigating the best solution, where modeling of the low-cost sensors is one of them.

Keywords
INS, GNSS, vehicle models, EKF, dead reckoning
Abstract

Utilizing a Global Navigation Satellite System (GNSS) together with an Inertial Navigation System (INS) is today a common integration method to obtain a positioning solution for autonomous systems. Both GNSS and INS have benefits and weaknesses where the best parts from both systems can be combined with a Kalman filter. Because of this complementary nature, it is of interest to look at the robustness of the positioning solution when the Global Navigation Satellite System is temporarily not available.

The aim of the thesis has been to investigate different vehicle models and to evaluate their short-time performance using a Dead Reckoning approach. The goal has been to develop a system for a Heavy Duty Vehicle (HDV) and to find out for which time interval a specific model can stay within a certain range when the GNSS is lost. A GNSS outage could for example happen when driving on a highway and passing signs, bridges and especially when driving inside tunnels. Also, for a solution to become commercially interesting, it must be cheap. Therefore, it is common to use so called Micro-Electro-Mechanical-Systems (MEMS) sensors which are of low-cost but suffer from biases, scale factors and temperature dependencies which must be compensated for.

The results from the tests show that some models are able to estimate the position with good precision during short time GNSS outages whereas other models do not deliver the required accuracy.

The main conclusion is that care should be taken when choosing the vehicle model so that it fits its usage area and the complexity needed to describe its motion. There are also lots of parameters to look at when investigating the best solution, where modeling of the low-cost sensors is one of them.
Acknowledgments

This Master’s Thesis has been conducted at Flowscape AB in Kista, Sweden, in collaboration with Scania CV AB in Södertälje, Sweden.

I would like to express my gratitude to my supervisor Michael Roth for his patience and extensive answers to all my questions as well as detailed feedback and remarks on the report. In addition, I would like to thank my examiner Daniel Axehill for his comments and suggestions throughout the report. I would also like to thank Per Sahlholm and Pär Degerman at Scania for making time in their busy schedule to collect data and their input on the vehicle models. Last, I would like to thank for the all help I have received from Flowscape and especially from Isak Tjernberg, Jakob Almqvist, Tommy Palm and Peter Reigo for their time and support.

Linköping, March 2015
David Enberg
# Contents

## Notation

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ix</td>
<td>ix</td>
</tr>
</tbody>
</table>

## 1 Introduction

<table>
<thead>
<tr>
<th>Subsection</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1 Background</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Approach</td>
<td>2</td>
</tr>
<tr>
<td>1.3 Limitations</td>
<td>3</td>
</tr>
<tr>
<td>1.4 Related Work</td>
<td>3</td>
</tr>
<tr>
<td>1.5 Outline</td>
<td>4</td>
</tr>
</tbody>
</table>

## 2 Theory

<table>
<thead>
<tr>
<th>Subsection</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1 Global Navigation Satellite System</td>
<td>5</td>
</tr>
<tr>
<td>2.2 Inertial Navigation System</td>
<td>6</td>
</tr>
<tr>
<td>2.2.1 Accelerometers</td>
<td>6</td>
</tr>
<tr>
<td>2.2.2 Gyroscopes</td>
<td>7</td>
</tr>
<tr>
<td>2.2.3 Magnetometers</td>
<td>7</td>
</tr>
<tr>
<td>2.2.4 Odometer</td>
<td>7</td>
</tr>
<tr>
<td>2.3 Controller Area Network</td>
<td>8</td>
</tr>
<tr>
<td>2.4 Coordinate Frames</td>
<td>8</td>
</tr>
<tr>
<td>2.4.1 Earth-Centered Inertial</td>
<td>8</td>
</tr>
<tr>
<td>2.4.2 Earth-Centered Earth-Fixed and WGS84</td>
<td>8</td>
</tr>
<tr>
<td>2.4.3 Local Navigation Frame</td>
<td>10</td>
</tr>
<tr>
<td>2.4.4 Body Frame</td>
<td>11</td>
</tr>
<tr>
<td>2.5 Kinematics</td>
<td>12</td>
</tr>
<tr>
<td>2.5.1 Euler Angles</td>
<td>12</td>
</tr>
<tr>
<td>2.6 Filtering</td>
<td>16</td>
</tr>
<tr>
<td>2.6.1 Extended Kalman Filter</td>
<td>16</td>
</tr>
</tbody>
</table>

## 3 Vehicle Models

<table>
<thead>
<tr>
<th>Subsection</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1 Vehicle Coordinates</td>
<td>19</td>
</tr>
<tr>
<td>3.2 Motion Models</td>
<td>21</td>
</tr>
<tr>
<td>3.2.1 Dead Reckoning Vehicle Model</td>
<td>21</td>
</tr>
<tr>
<td>3.2.2 Longitudinal Model</td>
<td>22</td>
</tr>
<tr>
<td>3.2.3 Bicycle Model</td>
<td>24</td>
</tr>
</tbody>
</table>
3.2.4 Four-Wheel Vehicle Model ........................................ 26

4 Sensor Models .................................................. 29
4.1 Accelerometer Model ........................................... 29
4.2 Gyroscope Model ............................................... 30
4.3 Odometer Model ............................................... 30
4.4 Sensor Characteristics ......................................... 32
  4.4.1 Sensor Bias ............................................... 32

5 Vehicle State Estimation ......................................... 41
5.1 Error-state Kalman Filter ...................................... 42
5.2 Model-based Kalman Filter .................................... 46
5.3 Data Generation ............................................... 47
5.4 Filter Implementation ......................................... 48

6 Results .......................................................... 51
6.1 Implementation Comparison .................................. 51
6.2 Performance Evaluation ....................................... 52

7 Conclusions and Future Work ................................. 65
7.1 Conclusions .................................................... 65
7.2 Future Work ................................................... 66

A Appendix ......................................................... 71
  A.1 Vehicle Models .............................................. 71
  A.2 Covariance Matrices ........................................ 75
  A.3 Vehicle Parameters ......................................... 77

Bibliography ....................................................... 79
### Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>EKF</td>
<td>Extended Kalman Filter</td>
</tr>
<tr>
<td>GNSS</td>
<td>Global Navigation Satellite System</td>
</tr>
<tr>
<td>GPS</td>
<td>Global Positioning System</td>
</tr>
<tr>
<td>DGPS</td>
<td>Differential GPS</td>
</tr>
<tr>
<td>RTK</td>
<td>Real Time Kinematic</td>
</tr>
<tr>
<td>GLONASS</td>
<td>GLObalnaya NAVigatsionnaya Sputnikovaya Sistema</td>
</tr>
<tr>
<td>INS</td>
<td>Inertial Navigation System</td>
</tr>
<tr>
<td>DR</td>
<td>Dead Reckoning</td>
</tr>
<tr>
<td>IMU</td>
<td>Inertial Measurement Unit</td>
</tr>
<tr>
<td>CAN</td>
<td>Controller Area Network</td>
</tr>
<tr>
<td>HDV</td>
<td>Heavy Duty Vehicle</td>
</tr>
<tr>
<td>MEMS</td>
<td>Micro-Electro-Mechanical-Systems</td>
</tr>
<tr>
<td>ECI</td>
<td>Earth Centered Inertial</td>
</tr>
<tr>
<td>ECEF</td>
<td>Earth-Centered Earth-Fixed</td>
</tr>
<tr>
<td>WGS</td>
<td>World Geodetic System</td>
</tr>
<tr>
<td>ENU</td>
<td>East-North-Up</td>
</tr>
<tr>
<td>PSD</td>
<td>Power Spectral Density</td>
</tr>
<tr>
<td>ACS</td>
<td>Auto Correlation Sequence</td>
</tr>
<tr>
<td>OXTS</td>
<td>Oxford Technical Solutions</td>
</tr>
</tbody>
</table>
## Symbols

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p$</td>
<td>Position states</td>
</tr>
<tr>
<td>$v$</td>
<td>Velocity states</td>
</tr>
<tr>
<td>$a$</td>
<td>Acceleration states</td>
</tr>
<tr>
<td>$\phi$</td>
<td>Roll angle</td>
</tr>
<tr>
<td>$\theta$</td>
<td>Pitch angle</td>
</tr>
<tr>
<td>$\psi$</td>
<td>Yaw angle</td>
</tr>
<tr>
<td>$\omega$</td>
<td>Angular velocity states</td>
</tr>
<tr>
<td>$\beta$</td>
<td>Sideslip state</td>
</tr>
<tr>
<td>$s$</td>
<td>Longitudinal slip state</td>
</tr>
<tr>
<td>$\delta_{rw}$</td>
<td>Right wheel radii state</td>
</tr>
<tr>
<td>$\delta_{lw}$</td>
<td>Left wheel radii state</td>
</tr>
<tr>
<td>$b$</td>
<td>Bias states</td>
</tr>
<tr>
<td>$f()$</td>
<td>Motion model</td>
</tr>
<tr>
<td>$h()$</td>
<td>Measurement model</td>
</tr>
<tr>
<td>$P$</td>
<td>Covariance of states</td>
</tr>
<tr>
<td>$Q$</td>
<td>Covariance of model</td>
</tr>
<tr>
<td>$R$</td>
<td>Covariance of measurements</td>
</tr>
<tr>
<td>$x$</td>
<td>State vector</td>
</tr>
<tr>
<td>$z$</td>
<td>Measurement vector</td>
</tr>
</tbody>
</table>
Introduction

The autonomous vehicle has in recent years taken a big leap towards becoming a part of the modern society. Many car manufacturers today are testing new functionality to increase the safety for the people inside and in the vehicle’s environment. One important aspect of the autonomous vehicle is the positioning solution, which also is the objective in this thesis. A Global Navigation Satellite System (GNSS) and a Dead Reckoning (DR) system is utilized and implemented together for the usage in a Heavy Duty Vehicle (HDV). The focus lies on developing a Dead Reckoning system which is able to accurately estimate the vehicle position during GNSS position outages. The performance between different models and solutions is investigated.

1.1 Background

Autonomous systems are developed to prevent and substantially reduce the accidents in traffic. Vehicle control systems already existing are for example the Anti-lock Braking System and the Electronic Stability Control which have made a great improvement of the vehicle safety [Doumiati et al., 2009]. Autonomous systems could be in the form of a supporting system, where the vehicle can steer autonomously only in situations when the driver becomes sick or falls asleep, or a completely autonomous system where a vehicle can transport goods between destinations. This, however, puts great demands on the solution and the developed algorithms.

With the technology available today a common procedure is the usage of a Global Navigation Satellite System together with an Inertial Navigation System. This is usually a good approach since a Global Navigation Satellite System can give position measurements with a fairly high accuracy, but with drawbacks such as slow update rate and external disturbance sensitivity. An Inertial Navigation
System, which generally contains an accelerometer and a gyroscope, outputs acceleration data and angular velocity data respectively and can give data at a high rate. However, the accuracy of the output data will degrade with time, because small errors will accumulate when integrating the measurements. Therefore, fusion of the Global Navigation Satellite System and Inertial Navigation System output data can create a more reliable solution, a system which handles GNSS position outages as well as errors present in the sensors.

Dead Reckoning means to determine your current position from previous measurements or estimates of your position and without input from a system measuring your absolute position. One major issue when relying on a Dead Reckoning solution is the sensor bias drift and a Dead Reckoning system should only be used uncorrected for short intervals. Information from an Inertial Navigation System is always available and can provide measurements with a high frequency. Inertial sensors are however affected by disturbances, which can be classified as a combination of a slowly changing biases and uncorrelated white noise. When measuring angular velocities and linear accelerations, where integration of these measurements provides the attitude, velocities and positions, small errors will accumulate over time. Thus, for a more reliable positioning solution, GNSS data are utilized to initialize the system as well as give corrections of the absolute position.

For control purposes, it is important that an autonomous vehicle has knowledge of its own position and should therefore handle situations when satellite data is not available. A situation with satellite outages could for example be when a vehicle is driving under a bridge, traffic signs or in urban areas where the satellite signal could be corrupted due to reflections from buildings and other objects. If the position on the road is known with good precision, the system should be able to position itself even during satellite outages of moderate duration. Hence, the development of a reliable positioning system, which will work even during absence of satellite data, is the objective.

1.2 Approach

This thesis focuses on developing a Dead Reckoning (DR) solution based on a vehicle model for a Heavy Duty Vehicle (HDV). The vehicle model uses measurement data from the vehicle’s Controller Area Network, CAN-bus, as well as sensor data from an Inertial Navigation System (INS). The Dead Reckoning system is implemented with a Kalman Filter to estimate and localize the vehicle. Two different implementation methods are investigated where the first one is called an error-state implementation and utilizes the position difference between the INS and GNSS to estimate an error which is used to correct the output from the INS. The other method is called a model-based implementation and combines the INS and GNSS together with a vehicle model to estimate the position of the vehicle. The hardware used in this thesis is based on low-cost Micro-Electro-Mechanical-Systems (MEMS). These components are very cheap which often affects the quality of the sensor data output. Errors occur due to temperature sensitivity, mis-
alignment of the sensor axes, quantization and more. Hence, the output data from the sensors must be calibrated in order to work as intended.

1.3 Limitations

Some assumptions had to be made due to limited time and resources.

All of the parameters used in the vehicle models are assumed to be known, where a few have been guessed to a plausible value. Especially the vehicle parameters used in a model called bicycle model have been hard to estimate because they depend on vehicle dynamics which requires a more thorough investigation.

Due to the usage of a single channel low-cost GPS there is no possibility to decouple the road bank angle from the vehicle’s own roll motion, which can be measured with a gyroscope. Therefore, the road is assumed to have minimum sideways slope.

Also, the GPS receiver and the INS are not mounted in the same location and thus producing an error called lever-arm effect, which however can be accounted for but has not been done within this thesis. The same problem occurs when the true trajectory was measured, with a high precision GPS receiver. Since the high precision GPS receiver and the low-cost GPS receiver also are not mounted in the same location, the trajectories from both systems will not be aligned. However, the receivers are placed on a common longitudinal line which makes it possible to detect lateral deviations.

1.4 Related Work

Combining GNSS and INS have been covered in many papers and books during the last decades and are almost ubiquitous today, but with no easy solution yet. Shin [2001] looks on a new calibration technique for low-cost MEMS sensors. Tham et al. [1998] addresses the sensor fusion problem for the navigation of a four-wheel steerable autonomous vehicle used for material handling in an outdoor environment. A localisation system based on the detection of landmarks laid along the lane is used to provide the absolute position corrections. Salmon and Bevly [2014] discusses the benefits of using vehicle sensors and an accurate vehicle model to replace/assist a low-cost Inertial Measurement Unit for ground vehicle navigation. Tightly-coupled Extended Kalman Filter algorithms are utilized combining multiple sensor sets. Bevly and Cobb [2010] has studied the applications of GNSS and written extensively about integration with ground vehicles and control algorithms. Groves [2008] describes the principles of integrating GNSS, inertial and multisensor navigation systems and provides an introduction to the investigated navigation systems. Leung et al. [2011] proposes an Integrated Kalman Filter design that uses low-cost GPS, INS, Wheel Speed Sensors (WSS) and Vehicle Models (VM). A single antenna GPS is combined with existing in-car sensors and WSS. The estimated states are then used in a two degree of freedom VM to estimate the steering bias and the sideslip angles. The proposed Integrated Kalman Filter is able to predict the biases in the steering wheel and
the tyre radius, hence giving an accurate estimation of the vehicle longitudinal velocity. Doumiati et al. [2009] develops an estimation method which uses a vehicle model together with real measurements in order to get real-time estimates of the lateral force at each tire-road contact point as well as the sideslip angle. The observer is based on the Unscented Kalman Filter.

1.5 Outline

This thesis is organized into 7 chapters:

- Chapter 2 describes the underlying theory and concepts considered within this thesis.
- Chapter 3 presents the investigated vehicle models.
- Chapter 4 describes the sensor models and sensor characteristics.
- Chapter 5 describes the integration of the GNSS/INS system.
- Chapter 6 presents the result. The performance of the different models is compared.
- Chapter 7 discusses conclusions based on the results and suggests areas for future work.
This chapter intends to give the reader an introduction to the theory used and will be the backbone for the following chapters. The chapter begins with information about the Global Navigation Satellite System and the Inertial Navigation System. A description of different coordinate frames is given and the transformations between the coordinate frames are presented. The chapter ends with a brief explanation of the filtering technique that are used.

2.1 Global Navigation Satellite System

A Global Navigation Satellite System (GNSS) provides information from satellites orbiting Earth. There are several different satellite systems where the Global Positioning System (GPS) developed by USA, is the most known one. Another existing satellite system is, for example, GLONASS which is developed and maintained by Russia. All these satellite systems consists of 20 to 30 satellites forming a constellation. Each satellite constellation orbits Earth at a distance of 25000 to 30000 km above ground to make it possible to receive signals from four satellites at any location on Earth [Groves, 2008]. By using a GNSS receiver the user can pick up the signals transmitted along a line of sight from the satellites. The position of the receiver is determined through a technique called triangulation where the signals from each satellite are calculated into a range measurement. The satellite signal consists of a carrier wave at a frequency near 1.6 gigahertz and as the receiver interprets the signals, it subtracts the time from its own internal clock, when a signal is received, with the time transmitted from the satellite clocks, when the signal was sent [Bevly and Cobb, 2010]. The difference is the time it took for the signal to reach the receiver and by multiplying with the speed of light the pseudorange between the satellite and the receiver has been determined. A pseudorange corresponds to a range measurement with some un-
corrected errors, mainly due to an offset in the receiver’s internal clock, which results in an incorrect range measurement. When range measurements from at least three satellites are available, a plane can be set up from these three satellites which will define two possible receiver locations, one on the surface of Earth and one far out in space [Bevly and Cobb, 2010]. A range measurement from a fourth satellite is used to solve for the internal clock bias in the receiver.

With a basic receiver a position accuracy of one to four meters in the horizontal plane and one to six meters in the vertical plane can be expected [Groves, 2008]. If supported by the GNSS receiver, the position estimate can be improved by also using information from stationary reference GNSS receivers called base stations. These are located at fixed positions on the surface of Earth and are used to correct the range errors in the measurements. This technique is called Differential GPS. Base stations near a mobile GNSS receiver will be exposed to approximately the same atmospheric conditions [Groves, 2008]. Since a base station has a known position, a comparison can be made of that position with the position of the mobile GNSS receiver, if also utilizing the same satellites. The errors can then be accounted for which improves the position accuracy to approximately one meter [Groves, 2008]. Finally, by also calculating the carrier-phase, i.e. by determining the number of phases in the signal sent between a satellite and the receiver, the position can be given with centimeter accuracy. In short, this is done by computing the distance between two cycles peaks corresponding to one wave-length of the satellite signal and this technique is called Real Time Kinematics navigation. A more detailed explanation of GNSS is given by Bevly and Cobb [2010], Groves [2008] and Subirana et al. [2013].

### 2.2 Inertial Navigation System

An Inertial Navigation System (INS) is a Dead Reckoning navigation system consisting of different sensors such as accelerometers, gyroscopes, magnetometers, odometers and barometers. Inertial sensors measure changes in the environment with respect to some reference frame. As previously stated, the advantage of using inertial sensors is their ability to be independent from external systems and thus unaffected by disturbances or blocking of signals compared to signals received from a GNSS [Bevly and Cobb, 2010].

#### 2.2.1 Accelerometers

Accelerometers translate sensed forces along a particular sensor axis into a signal level. The sensed forces can occur due to interference with Earth’s gravitation field but specifically from a change in motion. The latter is useful because given some initial position, the future positions can be determine by integrating the accelerations sensed by the accelerometer. Most of the accelerometers used in the vehicle field today are of a type called Micro-Electro-Mechanical-Systems (MEMS) [Bevly and Cobb, 2010]. From Newton’s second law, $F = ma$, the accelerometer is modeled as a dynamic system with a proof mass connected to a
2.2 Inertial Navigation System

damper and a spring. The displacement created by motion along a specific axis are measured and related to a force. Since the sensor is measuring the force applied to the proof mass, placing the sensor in Earth's gravitation field will give a reading of approximately $9.82 \text{ m/s}^2$ in the Up direction. This is due to a counter-acting force that keeps the mass in place. If the accelerometer is mounted with its sensitive axis in the direction of travel, a positive acceleration will give a measurement in the direction of travel because the proof mass will be forced into motion by the sensor body. Thus, an accelerometer is not measuring the direct forces but can sense them from the reaction forces applied to the proof mass. As the gravity component is always present, it must be taken into consideration when calculating the true acceleration of a vehicle.

2.2.2 Gyrosopes

Gyrosopes measure the rotation rate about a sensitive axis and are used for attitude calculations [Bevly and Cobb, 2010]. There are many types at different price ranges. MEMS gyroscopes are based on the principle of measuring vibrations. They generate harmonic motion at a known frequency which is translated into a specific rotation rate for that frequency. Earth's rotation will also be measured by a gyroscope and must be subtracted to receive the true rotation rate of the measured object. But unlike the gravity impact on accelerometers, this effect can usually be negligible in MEMS gyroscopes since it is not distinguishable from the gyroscopes’ bias and noise [Bevly and Cobb, 2010].

2.2.3 Magnetometers

Magnetometers measure the total magnetic flux density sensed along the axis of the sensor [Bevly and Cobb, 2010]. Earth's magnetic field points from the magnetic north pole to the magnetic south pole and by using a magnetometer the direction of the magnetic field can be measured at a point in space. For road applications, changes in the magnitude of the magnetometer measurements may be used to estimate the pitch and correct the heading accordingly [Bevly and Cobb, 2010]. However, a major problem is that magnetic fields are produced by other objects such as the vehicle itself and will cause an measurement error. Combining a magnetometer with other sensors measuring heading, such as integrated yaw rate from a gyroscope or from an differential odometer, these magnetic errors can be smoothed out [Bevly and Cobb, 2010].

2.2.4 Odometer

The distance traveled is usually calculated by measuring the rotations of the wheels together with a known wheel radii. These measurements are given by an odometer and can be recalculated into a speed and heading estimation. As the wheels will not spin at the same speed when turning, these measurements can also be used to estimate a virtual heading. This, together with the other inertial sensors can be very useful because of the independency from external influences.
2.3 Controller Area Network

In many vehicles today, there exists a network for communication called Controller Area Network, abbreviated CAN. The data is sent on a bus which makes it possible to send or receive messages from one system to another. In other words, a CAN-bus is a message-based communication protocol between components and systems within the vehicle. In this application, messages from seven sensors are utilized, namely the steering wheel angle, engine torque, wheel speed, current gear, current gear ratio, gear shift and brake.

2.4 Coordinate Frames

When navigating over a large area, position coordinates will most likely be specified using different coordinate frames. For instance, the measurements received by an accelerometer could be given in a coordinate frame that might differ from the coordinate frame the user wants to work in. Using the output from a gyroscope a transformation matrix can be formed to transform the accelerations to an appropriate coordinate frame. In the following section the different coordinate frames and their relation are given.

2.4.1 Earth-Centered Inertial

The Earth-Centered Inertial frame (ECI) is a coordinate frame with its origin centered at Earth's center of mass. This frame is a non-rotating and non-accelerating frame with respect to the fixed stars. The z-axis is parallel to Earth's spin axis, the x-axis is pointing towards the vernal equinox and the y-axis completes a right-handed orthogonal frame [Shin, 2001]. An overview of the ECI frame is shown in Figure 2.1 which also includes the Earth-Centered Earth-Fixed coordinate frame described in Section 2.4.2.

2.4.2 Earth-Centered Earth-Fixed and WGS84

An Earth-Centered Earth-Fixed frame (ECEF) is similar to the ECI frame except that all axes remain fixed by allowing the coordinate frame to move with the rotation of Earth. The ECEF coordinate frame enables the user to navigate with respect to the center of Earth. However, for most practical navigation problems, the user wants to know their position relative to Earth's surface. The model of Earth's surface often used in navigation systems is a type of ellipsoid due to the fact that Earth's surface is not completely spherical. The ellipsoid is defined by two radii. The equatorial radius, which is the length of the semi-major axis and the polar radius, which the length of the semi-minor axis. Coordinates given in the ECEF frame can be defined as Cartesian or geodetic coordinates. A Cartesian coordinate $R$ on the ellipsoid surface is defined as $R = [x, y, z]$ and the distance to that point from the center of Earth is computed as $R = |r| = \sqrt{x^2 + y^2 + z^2}$. Before satellites existed, several models were needed to describe the shape of Earth since one model could not fit the surface of Earth everywhere. But with satellites,
Figure 2.1: ECI and ECEF coordinate frame: The relation between the ECI and ECEF coordinate frame.

positions across the whole surface of the world have been measured with respect to a common reference, the satellite constellation, leading to the development of global ellipsoid models [Bevly and Cobb, 2010]. One standard is the World Geodetic System WGS84 which uses geodetic coordinates latitude $\varphi$ and longitude $\lambda$. As well as defining an ECEF coordinate frame and an ellipsoid, WGS84 provides models of Earth’s geoid and gravity field and a set of fundamental constants [Noureldin et al., 2012].

**World Geodetic System**

WGS84 is an Earth model developed by the United States Department of Defence in year 1984, where the surface of Earth has been modeled as an ellipsoid. The surface is located at the mean sea level and this model defines the relationship between geodetic coordinates given by satellites in latitude and longitude, and the ECEF cartesian coordinate system. The conversion from longitude $\lambda$, latitude $\varphi$ and height over sea level $h$ to ECEF is defined as

$$X = (N(\varphi) + h) \cos \varphi \cos \lambda,$$

$$Y = (N(\varphi) + h) \cos \varphi \sin \lambda,$$

$$Z = \left(N(\varphi)(1 - e^2) + h\right) \sin \varphi,$$
where \( N(\varphi) = \frac{a}{\sqrt{1-e^2 \sin^2 \varphi}} \), \( e \) is the eccentricity of the ellipsoid and \( a \) is the distance from the center to turning point of the ellipsoid denoted the semi-major axis.

### 2.4.3 Local Navigation Frame

The local navigation frame is defined as a plane on the ellipsoid Earth surface with the origin in the vehicle's center of mass. In this thesis the \( x \)-axis is pointing towards east and is hence known as the East axis. The \( y \)-axis is pointing in the north direction and is called the North axis, and the \( z \)-axis is denoted Up axis which is orthogonal to the surface of Earth. Together this frame is commonly known as the ENU-frame. The local navigation frame is important in navigation because the user usually wants to know their attitude relative to the east, north and up directions [Noureldin et al., 2012]. Another common way to define the local coordinate frame is the NED-frame which consists of the North axis, the East axis and the Down axis. This coordinate frame is often used in aviation since the axes coincide with the body-fixed orientation of the vehicle and where the \( z \)-axis is pointing towards the center of Earth [Shin, 2001].

![ENU coordinate frame](image_url)

**Figure 2.2:** ENU coordinate frame: A local navigation frame defined as ENU.
2.4.4 Body Frame

The measured output from an accelerometer is given in the body frame of the sensor. If the body frame of the sensor is assumed to be coherent with the vehicle’s body frame, there is a direct connection between the sensor frame and the body frame. The origin in the body frame is often assumed to be in the vehicle’s center of mass and so its coordinate frame will be aligned with the local navigation frame. The axes of the body frame are as this thesis defined as $x$-axis in the forward direction, $y$-axis pointing towards the left and $z$-axis pointing upwards, and thus completing the orthogonal set.

When describing angular motion, a rotating $x$-axis denotes a roll angle, a rotating $y$-axis denotes a pitch angle and a rotating $z$-axis denotes a yaw angle. Hence, the axes of the body frame are sometimes known as roll ($\phi$), pitch ($\theta$) and yaw ($\psi$) as shown in Figure 2.3. The body frame is essential in navigation because it describes the object that is moving. All inertial sensors measure the motion of the body frame with respect to some inertial frame. In most systems, the sensitive axes of the inertial instruments are aligned with the body frame axes, but there will always be some deviation. Prior to computation of the navigation solution, the inertial sensor outputs must be transformed to a common point of reference. This transformation is usually performed within the Inertial Navigation System (INS) [Noureldin et al., 2012].

Figure 2.3: Body frame: The vehicle frame defined as an orthogonal system.
2.5 Kinematics

In navigation, the linear and angular motion of one coordinate frame must be described with respect to another. In fact, the object frame and reference frame must be different, otherwise there will be no motion. There are different ways for attitude representation, for instance by using Euler angles or quaternions. All methods for representing attitude describes the orientation of one coordinate frame with respect to another and when defining the velocity and acceleration, it is important to correctly account for any rotation of the reference frame and the navigation frame. In this thesis, Euler angles are utilized and explained in Section 2.5.1.

2.5.1 Euler Angles

Euler angles provide an intuitive way of describing the attitude between two coordinate frames. The attitude is broken down into three rotations around different axes. The first rotation, $\psi$, is called the yaw rotation. This is performed about the inertial-frame $z$-axis. This rotation produces a new coordinate frame where the $x$- and $y$-axis are rotated by the yaw angle $\psi$ and where the $z$-axis is unchanged, aligned with the inertial frame. The transformation matrix relating yaw rotation to an inertial frame is given by

$$
R_z(\psi) = \begin{bmatrix}
\cos(\psi) & \sin(\psi) & 0 \\
-\sin(\psi) & \cos(\psi) & 0 \\
0 & 0 & 1
\end{bmatrix}.
$$

(2.4)

The pitch rotation, $\theta$, is performed about the vehicle's new $y$-axis from the previous yaw rotation. Here, the $x$ and $z$ components of the vector are transformed and the result is a set of coordinates resolved in the new frame. The transformation matrix relating pitch rotation to an inertial frame is given by

$$
R_y(\theta) = \begin{bmatrix}
\cos(\theta) & 0 & -\sin(\theta) \\
0 & 1 & 0 \\
\sin(\theta) & 0 & \cos(\theta)
\end{bmatrix}.
$$

(2.5)

The roll rotation, $\phi$, is performed about the vehicle's new $x$-axis obtained from the previous pitch rotation. This transforms the $y$ and $z$ components resulting in a vector in the new frame. The transformation matrix relating roll rotation to an inertial frame is given by

$$
R_x(\phi) = \begin{bmatrix}
1 & 0 & 0 \\
0 & \cos(\phi) & \sin(\phi) \\
0 & -\sin(\phi) & \cos(\phi)
\end{bmatrix}.
$$

(2.6)

By multiplying all of these rotations together, a transformation matrix is formed. However, it is important in what order these rotations are performed since a different order will generate a different transformation matrix which describes the orientation of the axes. Also, there is a problem when using Euler angles since a
singularity can occur when the pitch angle equals $\pm \frac{\pi}{2}$. This scenario should however not affect land vehicle navigation because of the unlikely event of a vehicle driving on a 90 degrees-tilted road.

Hence, by using Euler angles the attitude can be stored within a vector containing $[\phi, \theta, \psi]$ which is updated with the transformed measurements received from the gyroscope. The gyroscope measurements must be transformed from the body frame to the local navigation frame. However, the derivatives of the Euler angles are obtained, so the Euler angles in the local navigation frame are then given by an integration.

**Angular Rate Transformation**

The transformation matrix for the angular rates, which is the gyroscope output, can be set up by first moving from the local navigation frame to the body frame. The $x$-axis in the local navigation frame denoted the Euler angle roll axis is fixed and aligned with the gyroscope’s $x$-axis. The $y$-axis which denote the Euler angle pitch rate is rotated to the same plane as the roll rate axis using the transformation matrix describing the relationship between the roll plane and the inertial frame. Last, the Euler yaw rate angle is rotated by the roll- and pitch transformations to be aligned with the gyroscope’s $z$-axis and is expressed as

$$
\begin{pmatrix}
\dot{\phi} \\
0 \\
\end{pmatrix}_{n} + R_{x}(\phi)
\begin{pmatrix}
0 \\
0 \\
\end{pmatrix}_{n} + R_{x}(\phi) \cdot R_{y}(\theta)
\begin{pmatrix}
0 \\
0 \\
\end{pmatrix}_{n} + R_{y}(\theta) \cdot R_{z}(\psi)
\begin{pmatrix}
0 \\
0 \\
\end{pmatrix}_{n}
$$

Writing out the transformation matrix using (2.5)-(2.6), the attitude matrix can be defined as

$$
\begin{pmatrix}
\dot{\phi} \\
\dot{\theta} \\
\dot{\psi} \\
\end{pmatrix}_{n} =
\begin{pmatrix}
1 & 0 & -\sin(\theta) \\
0 & \cos(\phi) & \sin(\phi) \cdot \cos(\theta) \\
0 & -\sin(\phi) & \cos(\phi) \cdot \cos(\theta) \\
\end{pmatrix}
\begin{pmatrix}
\dot{\phi} \\
\dot{\theta} \\
\dot{\psi} \\
\end{pmatrix}_{n}
$$

Multiplying the attitude transformation matrix with its inverse results in the transformation from the body measurements to the local navigation frame, denoted $R_{b}^{n}$, which is written as

$$
\begin{pmatrix}
\dot{\phi} \\
\dot{\theta} \\
\dot{\psi} \\
\end{pmatrix}_{b} =
\begin{pmatrix}
\sin(\phi) \cdot \tan(\theta) & \cos(\phi) \cdot \tan(\phi) \\
\cos(\phi) & -\sin(\phi) \\
\sin(\phi) / \cos(\theta) & \cos(\phi) / \cos(\theta) \\
\end{pmatrix}
\begin{pmatrix}
\dot{\phi} \\
\dot{\theta} \\
\dot{\psi} \\
\end{pmatrix}_{n}
$$

Notice the singularity which can occur in matrix $R_{b}^{n}$, due to the division with $\cos(\theta)$. This is also known as a gimbal lock which means that two axes have been aligned and describe the same plane. For example, when the pitch angle is rotated by $\frac{\pi}{2}$ it will cause the roll and yaw axis to be aligned.

**Linear Measurement Transformation**

To relate measurements received from the magnetometer or the accelerometer another transformation matrix is needed. This transformation matrix transforms
measurements from the body frame to the local navigation frame and is computed by three consecutive rotations [Britting, 2010] around the body axes using the Euler rotations defined in (2.4)-(2.6) as
\[
C_n^b = (C_n^b)^T = R_z(-\psi)R_y(-\theta)R_x(-\phi)
\]
\[
= \begin{pmatrix}
\cos(\psi) & -\sin(\psi) & 0 \\
\sin(\psi) & \cos(\psi) & 0 \\
0 & 0 & 1
\end{pmatrix}
\begin{pmatrix}
\cos(\theta) & 0 & \sin(\theta) \\
0 & 1 & 0 \\
0 & \sin(\phi) & \cos(\phi)
\end{pmatrix}
\begin{pmatrix}
1 & 0 & 0 \\
0 & \cos(\phi) & -\sin(\phi) \\
0 & \sin(\phi) & \cos(\phi)
\end{pmatrix}
\]
\[
= \begin{pmatrix}
\cos(\theta)\cos(\psi) & -\cos(\phi)\sin(\psi) + \sin(\phi)\sin(\theta)\cos(\psi) & \sin(\phi)\sin(\psi) + \cos(\phi)\sin(\theta)\cos(\psi) \\
\sin(\theta) & \cos(\phi)\cos(\psi) + \sin(\phi)\sin(\theta)\sin(\psi) & -\sin(\phi)\cos(\psi) + \cos(\phi)\sin(\theta)\sin(\psi) \\
-\sin(\phi)\cos(\theta) & \sin(\phi)\cos(\psi) & \cos(\phi)\cos(\theta)
\end{pmatrix}
\]
\[
= \begin{pmatrix}
\cos(\theta)\cos(\psi) & -\sin(\phi)\cos(\psi) + \sin(\phi)\sin(\theta)\cos(\psi) & \sin(\phi)\sin(\psi) + \cos(\phi)\sin(\theta)\cos(\psi) \\
\sin(\theta) & \cos(\phi)\cos(\psi) + \sin(\phi)\sin(\theta)\sin(\psi) & -\sin(\phi)\cos(\psi) + \cos(\phi)\sin(\theta)\sin(\psi) \\
-\sin(\phi)\cos(\theta) & \sin(\phi)\cos(\psi) & \cos(\phi)\cos(\theta)
\end{pmatrix},
\tag{2.10}
\]
where \(\psi\) is the yaw angle, \(\theta\) is the pitch angle and \(\phi\) is the roll angle.

**ECEF to ENU Transformation**

Positions received from a Global Navigation Satellite System (GNSS) must also be transformed into the local navigation frame. The conversation from latitude, longitude and height to the Earth-Centred, Earth-Fixed coordinate frame has already been mentioned in Section 2.4.2 using (2.1). The transformation from the ECEF-frame to the local navigation frame, ENU, can be expressed as
\[
C_n^e = \begin{pmatrix}
-\sin(\lambda) & \cos(\lambda) & 0 \\
-\sin(\phi)\cos(\lambda) & -\sin(\phi)\sin(\lambda) & \cos(\phi) \\
\cos(\phi)\cos(\lambda) & \cos(\phi)\sin(\lambda) & \sin(\phi)
\end{pmatrix},
\tag{2.11}
\]
where \(\phi\) is the latitude and \(\lambda\) is the longitude received from a GPS.

Using a reference station, the conversation from an ECEF position to an ENU position is given by
\[
Pos_{ENU} = C_n^e(\lambda_{\text{ref}}, \phi_{\text{ref}}) \left[ (Pos_{ECEF}(\lambda, \phi) - Pos_{ECEF}(\lambda_{\text{ref}}, \phi_{\text{ref}})) \right].
\tag{2.12}
\]

**Earth Rotation**

The difference between the ECI coordinate frame and the ECEF coordinate frame is Earth’s rotation. The rotation rate vector of the ECEF-frame with respect to the Inertial-frame projected to the ECEF-frame is given by
\[
\omega_{ie}^e = \begin{pmatrix}
0 \\
\omega_c \cos(\phi) \\
\omega_c \sin(\phi)
\end{pmatrix}^T,
\tag{2.13}
\]
where \(\omega_c\) is the magnitude of Earth’s rotation and is equal to \(7.2921158 \times 10^{-5} \frac{\text{rad}}{\text{s}}\). The conversation of Earth’s rotation to the local navigation frame is calculated as in Shin [2001], but with the ENU-frame as navigation frame as
\[
\omega_{ie}^n = C_n^e \omega_{ie}^e = \begin{pmatrix}
0 & \omega_c \cos(\phi) & \omega_c \sin(\phi)
\end{pmatrix}^T.
\tag{2.14}
\]
The turn rate of the local navigation frame with respect to the ECEF-frame is expressed in terms of the rate of change of latitude and longitude [Shin, 2001] as

$$\omega_{en}^n = \begin{pmatrix} -\dot{\phi} & \lambda \cos(\varphi) & \lambda \sin(\varphi) \end{pmatrix}^T. \tag{2.15}$$

Expressing $\dot{\phi} = v_N/(M + h)$ and $\lambda = v_E/(N + h) \cos(\varphi)$ gives

$$\omega_{en}^n = \begin{pmatrix} -\frac{v_N}{(M + h)} & \frac{v_E}{(N + h)} & \frac{v_E \tan(\varphi)}{(N + h)} \end{pmatrix}, \tag{2.16}$$

where $v_N, v_E$ are velocities in the north and east direction, respectively. $h$ is the ellipsoidal height and $M, N$ are radii of curvature in the meridian and prime vertical [Shin, 2001] written as

$$M = \frac{a(1 - e^2)}{(1 - e^2 \sin^2(\varphi))^{3/2}}, \tag{2.17}$$

$$N = \frac{a}{(1 - e^2 \sin^2(\varphi))^{1/2}}, \tag{2.18}$$

where $a$ is the semi-major axis and $e$ is the eccentricity of the reference ellipsoid.

The final impact from Earth’s rotation expressed in the local navigation frame is computed as the sum of (2.14) and (2.16),

$$\omega_{in}^n = \omega_{ie}^n + \omega_{en}^n. \tag{2.19}$$

To get the rotation rate of the local navigation frame with respect to the inertial frame given in the body frame, $\omega_{in}^b$ is used together with the rotation matrix derived in Section 2.5.1.

$$\omega_{in}^b = R_{in}^b \omega_{in}^n. \tag{2.20}$$

These transformations are used in Section 5.1 when expressing the inertial sensor dynamics.

**Skew Symmetric Matrix**

A cross product can be computed by utilizing the skew symmetric matrix form according to Britting [2010] as

$$a \times \omega = \begin{pmatrix} a_z \omega_y - a_y \omega_z \\ a_x \omega_z - a_z \omega_x \\ a_y \omega_x - a_x \omega_y \end{pmatrix} = \begin{pmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{pmatrix} a = \Omega a, \tag{2.21}$$

where $a$ is an arbitrary three dimensional vector and $\Omega$ is the skew symmetric form of $\omega$. This alternative way of writing a cross-product are used in Section 5.1.
2.6 Filtering

Fusing the information from a GNSS and INS can be achieved via Kalman filtering, which is an recursive estimation process suitable for real-time implementations [Gustafsson, 2012]. From noisy input data, a Kalman filter optimizes the state estimates such that the estimates are unbiased and the state covariance is minimal. In the 1960s, Rudolf Kalman first developed the filter for linear systems but the filter has been extended to other cases as well. A nonlinear state space model in discrete time is used to describe a certain nonlinear system and can be expressed as

\[
\begin{align*}
x_{k+1} &= f(x_k, u_k) + w_k, \\
z_k &= h(x_k) + \epsilon_k,
\end{align*}
\]  

(2.22)

where \( x_{k+1} \) denotes the state vector created from the previous states \( x_k \), inputs \( u_k \) and the process noise \( w_k \). \( z_k \) is the measurement vector and \( \epsilon_k \) is the measurement noise. \( f() \) and \( h() \) are functions which generate the states and measurements respectively and which are derived from the known properties of the system [Bevly and Cobb, 2010].

In this thesis, the implemented system is nonlinear and an Extended Kalman Filter (EKF) is used for state estimation. The estimated states are set up in the ENU-frame and all the measurements are converted to the ENU-frame.

2.6.1 Extended Kalman Filter

The EKF is used for approximate filtering via nonlinear state space models. The linearization of the nonlinear dynamics is achieved by performing a Taylor expansion around current state estimates. The Jacobian matrices of the motion and measurement equations are calculated with respect to the states and used in the time and measurement updates. However, due to the linearization process, the EKF is not an optimal estimator and tends to underestimate the true covariance matrix which relates to the uncertainties of the measurements and hence, the result could be unsatisfactory. Also, if the error covariance matrix is badly initialized or the process model is incorrect, the filter can quickly diverge [Bevly and Cobb, 2010]. Besides that, the EKF is a popular extension of the Kalman Filter because of its simplicity. The algorithm consists of two parts, the time update and the measurement update which are defined in (2.23) and (2.24).

**Time update:**

\[
\begin{align*}
\hat{x}_{k+1|k} &= f(\hat{x}_{k|k}, u_k), \\
P_{k+1|k} &= f'(\hat{x}_{k|k}, u_k)P_{k|k}(f'(\hat{x}_{k|k}, u_k))^T + G_k Q_k G_k^T.
\end{align*}
\]  

(2.23)

During a time update the states and covariances are predicted, given the previous state estimates and inputs. The prediction is based on the motion model \( f(x, u) \). Since the output is a prediction of how the states are progressing, the uncertainty or error covariance matrix \( P \), is increased.
Measurement update:

\[ S_k = h'(\hat{x}_{k|k-1})P_{k|k-1}(h'(\hat{x}_{k|k-1}))^T + R_k, \]
\[ K_k = P_{k|k-1}(h'(\hat{x}_{k|k-1}))^T S_k^{-1}, \]
\[ \epsilon_k = z_k - h(\hat{x}_{k|k-1}), \]
\[ \hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k \epsilon_k, \]
\[ P_{k|k} = P_{k|k-1} - K_k h'(\hat{x}_{k|k-1})P_{k|k-1}. \]

During the measurement update real measurement data is used to correct the estimated states using the measurement model \( h(x) \). The previous estimated uncertainty \( P_{k|k-1} \) will be used to calculate a weight of how much the measurements can be trusted.

As stated in the beginning of the Section 2.6.1, the nonlinear motion model is linearized around the current state estimate and this is performed by formulating the Jacobians as

\[ f'(\hat{x}_{k|k}, u_k) = \frac{\partial f}{\partial x}|_{\hat{x}_{k|k}, u_k}, \]
\[ h'(\hat{x}_{k|k-1}) = \frac{\partial h}{\partial x}|_{\hat{x}_{k|k-1}}. \]

Filter tuning

The process noise covariance matrix \( Q \) and the measurement noise covariance matrix \( R \) represents the uncertainty of the motion and measurement model respectively, but are often used as a set of tuning parameters. If a diagonal element in the \( Q \)-matrix, corresponding to the variance of a certain state, is increased the state will respond more quickly to rapid changes, thus increasing the responsiveness of the system. In the same manner, if a diagonal element in the \( R \)-matrix, corresponding to the uncertainty of a measurement, is increased, that will affect the state’s sensitivity to noisy measurements. It is usually a good approach to set the state parameters to a reasonable value which should reflect the behavior of the motion and measurement models and then after simulations tune the values to achieve a desired result with a good trade off between responsiveness and noise sensitivity. It is also important to initialize the error state covariance matrix \( P \) with reasonable values from the beginning because if the initial values are too small then the unmodeled errors in the system will be much larger than the uncertainties will reflect. That can cause the Kalman filter to apply wrong weight [Groves, 2008].

One good tuning philosophy according to Groves [2008] is to fix \( P \) and \( Q \) and then vary \( R \) by trial-and-error to find the smallest value that gives stable state estimates. If this does not give satisfactory performance, \( P \) and \( Q \) can also be varied. However, while this approach works for linear systems it might not be the case for a nonlinear system.

As a conclusion, the tuning process of a Kalman filter is basically a trade off between convergence rate and stability and similar to the filter algorithms, an
iterative process. For more information in this matter see Groves [2008], Bevly and Cobb [2010] and Gustafsson [2012].
This chapter derives the equations for the different vehicle models. The chapter begins with a Dead Reckoning vehicle model independent of vehicle parameters, followed by a longitudinal vehicle model which includes inputs from CAN. A bicycle model is then presented which focuses on lateral motion. The last model is based on a four-wheel vehicle model which also includes roll dynamics.

### 3.1 Vehicle Coordinates

First, the relation between a vehicle’s body fixed coordinate system and the ENU-coordinate frame is shown in Figure 3.1 and Figure 3.2. Using velocity and angular measurements received from an INS, the position in the ENU frame is obtained, see Figure 3.1. \( \psi \) is the heading angle and is defined as the angle between a line parallel to the East direction and the longitudinal direction \( v_x \) and the sideslip angle \( \beta \) is defined as the angle between the direction of the velocity \( v \) and the longitudinal direction \( v_x \).

Figure 3.2 shows how the vehicle’s body frame relates to the z-axis defined as the Up direction in the ENU-frame where \( \theta \) is the pitch angle and denotes the slope of the road. The relation between the velocity in the body system and the local ENU-system can be summarized as

\[
\begin{align*}
\dot{E} &= \dot{p}_E = v_E = v \cos(\psi + \beta) \cos(\theta), \\
\dot{N} &= \dot{p}_N = v_N = v \sin(\psi + \beta) \cos(\theta), \\
\dot{U} &= \dot{p}_U = v_U = v \sin(\theta),
\end{align*}
\]

where \( v \) is the magnitude of the velocity, defined as \( v = \sqrt{v_x^2 + v_y^2 + v_z^2} \).
Figure 3.1: Reference frame relationship: A vehicle's velocity given in the body frame and in relation to the ENU-frame.

Figure 3.2: Reference frame relationship: A vehicle's velocity given in the body frame and in relation to the ENU-frame.
3.2 Motion Models

An object’s motion can be described by a model, based on a set of mathematical equations. The basic approach is to set up constraints for how the object is allowed to behave and therefore governed by kinematics laws and the undescribed behaviour is modeled as random noise. A model’s dynamics is often described in continuous time, but when implemented in a computer, the model is rewritten into a discrete form because sensor measurement is typically obtained at discrete times.

### 3.2.1 Dead Reckoning Vehicle Model

One straight-forward motion model, referred to as the Dead Reckoning vehicle model, uses the measurements given in the vehicle’s coordinate frame and transforms it into a common coordinate frame, recall (3.1). This model is also similarly described by Andersson and Fjellström [2004] and is derived from a constant acceleration model [Gustafsson, 2012]. The model utilizes the position, velocity and acceleration as states in three directions, 

\[ x = \begin{bmatrix} p_x(t) & p_y(t) & p_z(t) & v_x(t) & v_y(t) & v_z(t) & a_x(t) & a_y(t) & a_z(t) \end{bmatrix}, \]

and specific vehicle parameters are not included. The model can be described as

\[
\dot{x}(t) = \begin{bmatrix} 0_{3\times3} & I_3 & 0_{3\times3} \\
0_{3\times3} & 0_{3\times3} & I_3 \\
0_{3\times3} & 0_{3\times3} & 0_{3\times3} \end{bmatrix} x(t) + \begin{bmatrix} 0_{3\times3} \\
0_{3\times3} \\
I_3 \end{bmatrix} w(t), \quad (3.2)
\]

where \( w(t) \) is process noise, \( I_3 \) is the identity matrix and \( 0_{3\times3} \) is a three by three zero matrix. The discrete state-space representation is defined as

\[
x(t + T) = \begin{bmatrix} I_3 & T I_3 & \frac{T^2}{2} I_3 \\
0_{3\times3} & I_3 & T I_3 \\
0_{3\times3} & 0_{3\times3} & I_3 \end{bmatrix} x(t) + \begin{bmatrix} \frac{T^3}{6} I_3 \\
\frac{T^2}{2} I_3 \\
T I_3 \end{bmatrix} w(t). \quad (3.3)
\]

Inserting (3.1) and deriving the body acceleration in similar manner as in (3.1), the predicted positions, velocities and accelerations are given as

\[
\begin{align*}
p_x(t + T) &= p_x(t) + T v(t) \cos(\psi(t)) \cos(\theta(t)) + \frac{T^2}{2} a \cos(\psi(t)) \cos(\theta(t)) \\
p_y(t + T) &= p_y(t) + T v(t) \sin(\psi(t)) \cos(\theta(t)) + \frac{T^2}{2} a \sin(\psi(t)) \cos(\theta(t)) \\
p_z(t + T) &= p_z(t) + T v(t) \sin(\theta(t)) + \frac{T^2}{2} a \sin(\theta(t)) \\
v_x(t + T) &= v_x(t) + T a_x(t) \\
v_y(t + T) &= v_y(t) + T a_y(t) \\
v_z(t + T) &= v_z(t) + T a_z(t) \\
a_x(t + T) &= a_x(t) \\
a_y(t + T) &= a_y(t) \\
a_z(t + T) &= a_z(t)
\end{align*}
\]

where the sideslip angle \( \beta \) and the process noise \( w(t) \) have been neglected.
3.2.2 Longitudinal Model

A more detailed model of a vehicle could be to describe the system with Newton’s second law of motion and define the internal and external forces acting on the system. This section describes the movement of a heavy duty vehicle (HDV) where the internal forces governed by the vehicle’s powertrain can be modeled according to Eriksson and Nielsen [2012] and Sahlholm [2011], see Figure 3.3. From Figure 3.3, a mathematical representation is set up according to Newton’s second law as

\[ F_{\text{engine}} - F_{\text{brake}} - F_{\text{airdrag}} - F_{\text{roll}} - F_{\text{gravity}} = m_t a, \]  
(3.5)

where \( F_{\text{engine}} \) is the driving force from the engine, \( F_{\text{airdrag}} \) is the air drag, \( F_{\text{roll}} \) is the rolling resistance, \( F_{\text{gravity}} \) is the gravitational force acting on the system, \( m_t \) is the total mass and \( a \) is the longitudinal acceleration. The braking torque, \( F_{\text{brake}} \), is hard to measure and also often zero and is therefore neglected in this model.

The equation describing the internal forces is defined as

\[ F_{\text{engine}} = \frac{i_t i_f \eta_t \eta_f}{r_w} T_e, \]  
(3.6)

where \( i_t \) is the conversation ratio and \( \eta_t \) is the efficiency constant for the transmission gear, \( i_f \) and \( \eta_f \) is the conversation ratio and efficiency constant for the final drive, respectively. \( T_e \) is the engine output torque and \( r_w \) denotes the wheel radius.

The model for the air drag is defined as

\[ F_{\text{airdrag}} = \frac{1}{2} c_d A_d \rho_a v^2, \]  
(3.7)

where \( c_d \) is the aerodynamic drag coefficient, \( A_d \) is the frontal area of the vehicle, \( \rho_a \) is the air density and \( v \) is the vehicle speed.

\[Figure 3.3: Longitudinal model: The forces acting on the vehicle.\]
3.2 Motion Models

The roll resistance is given by

\[ F_{\text{roll}} = m g c_r \cos(\theta), \]  

(3.8)

where \( m \) and \( g \) are the mass and gravity components, \( c_r \) is the rolling resistance coefficient and \( \theta \) denotes the pitch angle, the road slope.

The gravity model is modeled as

\[ F_{\text{gravity}} = m g \sin(\theta), \]  

(3.9)

and depending on the road slope this force either acts as a regressive or progressive force.

The total accelerated mass of the vehicle \( m_t \) is defined from specific vehicle properties together with the vehicle’s mass

\[ m_t = m + \frac{J_w + i_t^2 i_f^2 \eta_t \eta_f J_e}{r_w^2}, \]  

(3.10)

where \( J_w \) is the mass moment of inertia for the wheel, \( J_e \) is mass moment of inertia for the engine, \( r_w \) is the wheel radius, \( i_t, i_f \) denote the conversion ratio for the transmission and final drive and \( \eta_{t,f} \) denote the efficiency constant for the transmission and final drive, respectively.

Combining (3.5)-(3.10) results in

\[
\dot{v} = \frac{d v}{d t} = \frac{1}{m_t} (F_w - F_{\text{airdrag}}(v) - F_{\text{roll}}(\theta) - F_{\text{gravity}}(\theta)) = \\
= \frac{1}{J_w/r_w^2 + m + i_t^2 i_f^2 \eta_t \eta_f J_e} \left( F_w - F_{\text{airdrag}}(v) - F_{\text{roll}}(\theta) - F_{\text{gravity}}(\theta) \right) = \\
= \frac{r_w^2}{J_w + m r_w^2 + i_t^2 i_f^2 \eta_t \eta_f J_e} \left( i_t i_f \eta_t \eta_f \frac{T_e}{r_w} - \frac{1}{2} c_w A_d \rho_d v_x^2 - c_r m g \cos(\theta) - m g \sin(\theta) \right),
\]  

(3.11)

which can be rewritten as

\[ \dot{v} = \kappa_1 T_e - \kappa_2 v^2 - \kappa_3 \cos(\theta) - \kappa_4 \sin(\theta), \]  

(3.12)

where the constants are given by

\[
\kappa_1 = \frac{r_w i_t i_f \eta_t \eta_f}{J_w + m r_w^2 + i_t^2 i_f^2 \eta_t \eta_f J_e}, \quad \kappa_2 = \frac{1}{2} \frac{r_w^2 A_d \rho_d c_d}{J_w + m r_w^2 + i_t^2 i_f^2 \eta_t \eta_f J_e}, \quad \kappa_3 = \frac{c_r m g}{J_w + m r_w^2 + i_t^2 i_f^2 \eta_t \eta_f J_e}, \quad \kappa_4 = \frac{r_w^2 m g}{J_w + m r_w^2 + i_t^2 i_f^2 \eta_t \eta_f J_e}.
\]  

(3.13)

Equation (3.12) describes how the dynamics of a HDV affects the longitudinal acceleration which separates this model from the dead reckoning vehicle model.
### 3.2.3 Bicycle Model

The bicycle model is a commonly used representation for describing the dynamics of a ground vehicle [Bevly and Cobb, 2010]. In the model the wheels on the same axle are represented by a single tire and therefore the model assumes that the slip angles as well as the steer angles affecting both wheels are approximately the same. Also the effects of the suspension roll is small, which holds for most driving situations [Ryu, 2004]. The model equations are derived from the forces and moments acting on the vehicle and the model is depicted in Figure 3.4. The

![Bicycle Model: The forces acting on the vehicle.](image)

The slip angle $\beta$ can be assumed to be zero when driving at a low speed with minimal slip, but for higher speeds the vehicle will develop larger slip angles. With Newton’s second law as the starting point the lateral force, with the slip angle included and the moment around the center of gravity, can be written as

\[
F_y = m\ddot{y} = m(v_x\dot{\beta} + v_x\omega_\psi) = F_{yF} \cos \delta + F_{yR},
\]

\[
M = I_z\ddot{\psi} = aF_{yF} \cos \delta - bF_{yR},
\]  

(3.14)
where $v_x$ is the longitudinal velocity, $\beta$ is the sideslip angle, $\omega_\psi$ is the yaw rate, $I_z$ is the yaw moment of inertia and $F_{yF}, F_{yR}$ are the front and rear tire forces. $a$ and $b$ are distances from the center of gravity of the vehicle to the front and rear axles, respectively. The dynamic equations for the sideslip rate and yaw rate are of interest and can be rewritten using (3.14) as

$$\dot{\beta} = \frac{F_{yF} \cos(\delta) + F_{yR}}{mv_x} - \omega_\psi,$$

$$\dot{\omega}_\psi = \frac{aF_{yF} \cos(\delta) - bF_{yR}}{I_z},$$

where $m$ is the vehicle mass, $F_{yF}$ and $F_{yR}$ are the lateral tire forces, $\delta$ is the steering angle, $a$ and $b$ are distances to the front and rear axles from the center of gravity and $I_z$ is the moment of inertia about the vehicle’s yaw axis.

Utilizing the Pacejka tire model, also known as the magic formula [Pacejka and Besselink, 2008], the front and rear tire forces $F_{yF}, F_{yR}$ can be expressed as

$$F_{yF} = -C_\alpha f \alpha_f,$$

$$F_{yR} = -C_\alpha r \alpha_r.$$

The Pacejka tire model is used to model nonlinear lateral tire behavior [Salmon and Bevly, 2014] where the model assumes that the tire forces remain in the linear region of the tire and are proportional to the tire’s respective slip angle times the tire’s cornering stiffness $C_\alpha$, which is defined as the slope of the lateral force and slip angle [Bevly and Cobb, 2010]. A vehicle’s tire generates lateral force which makes it possible to control the direction of the vehicle. A tire’s contact area to the ground will deform in the direction of travel as the tire spins. This elasticity produces the lateral forces with a slip angle $\alpha_f$, which represents the angle between the tire’s direction of travel and its longitudinal axis. The angular front and rear tire slip can be described and simplified with a small angle assumption as

$$\alpha_f = -\delta + \arctan\left(\frac{v_y + a\omega_\psi}{v_x}\right) \approx -\delta + \frac{v_y + a\omega_\psi}{v_x},$$

$$\alpha_r = \arctan\left(\frac{v_y - b\omega_\psi}{v_x}\right) \approx \frac{v_y - b\omega_\psi}{v_x}.$$

In Anderson and Bevly [2005] a formula to estimate the front and rear cornering stiffness is presented. Using a steady state yaw rate and sideslip, which are functions of weight split and tire cornering stiffness, and if a known weight split is assumed, then the front and rear cornering stiffness can be estimated as

$$C_\alpha r = \frac{mv \omega_\psi}{(\frac{b}{a} + 1)\alpha_r} \approx \frac{mv \omega_\psi}{(\frac{b}{a} + 1)(\beta - \frac{ba\omega_\psi}{v})},$$

$$C_\alpha f = \frac{bC_\alpha \alpha_r}{a\alpha_f} \approx \frac{bC_\alpha \beta - \frac{br}{v}}{a(\beta + \frac{ar}{v} - \delta)}.$$
In general, the cornering coefficient for a single tire is often pre-determined from a tire rig. The determined value is then doubled to give the axle cornering stiffness \((C_{af} \text{ and } C_{ar})\) [Anderson and Bevly, 2005].

Combining (3.14)-(3.20), the state-space representation for the lateral bicycle model can be written as

\[
\begin{bmatrix}
\dot{\omega}_\psi \\
\dot{\beta}
\end{bmatrix} =
\begin{bmatrix}
-a^2C_{af} - b^2C_{ar} & C_{ar}b - aC_{af} \\
bC_{ar} - aC_{af} & -aC_{af} - C_{ar}
\end{bmatrix}
\begin{bmatrix}
\omega_\psi \\
\hat{\beta}
\end{bmatrix} + \begin{bmatrix}
aC_{af} \\
bC_{af}
\end{bmatrix} \frac{I_z}{mv_x} \delta. \quad (3.23)
\]

Equation (3.23) describes the dynamics of the yaw rate \(\omega_\psi\) and the slip angle \(\beta\). The lateral acceleration can also be derived and included in the dynamics of the model.

### 3.2.4 Four-Wheel Vehicle Model

In order to model the effects of roll, the previous model needs to be modified into a four-wheel vehicle model [Bevly and Cobb, 2010]. Roll will occur due to a different weight distribution on the tires. The tires on the same axle will experience a different vertical force due to this weight change and thus will affect the lateral and longitudinal forces acting on the tires [Salmon and Bevly, 2014]. Figure 3.5 depicts an overview of the vehicle model, where \(\omega_\psi\) is the yaw rate, \(\beta\) is the sideslip angle, \(\delta\) is the steering angle, \(v_x\) and \(v_y\) is the longitudinal and lateral velocity respectively, \(t_{f,r}\) is the axle width and \(a\) and \(b\) are the distances to the front and rear axles. By formulating Newton’s second law the forces in the lateral direction and the moments about the vehicle’s center of gravity is expressed as

\[
F_y = m\ddot{y} = (F_{yfL} + F_{yfR}) \cos \delta + (F_{xfL} + F_{xfR}) \sin \delta + F_{yrL} + F_{yrR}, \quad (3.24a)
\]

\[
M_{cg} = I_z \dot{\omega}_\psi = a \left( (F_{yfR} + F_{yfL}) \cos \delta + (F_{xfR} + F_{xfL}) \sin \delta \right)
- \frac{t_f}{2} \left( (F_{yfR} - F_{yfL}) \sin \delta - (F_{xfR} - F_{xfL}) \cos \delta \right)
- b \left( F_{yrR} + F_{yrL} \right) - \frac{t_r}{2} \left( F_{xrL} - F_{xrR} \right). \quad (3.24b)
\]

The vehicle fixed lateral acceleration due to sideslip and the effect from the yaw rate is given by

\[
\dot{y} = v \sin \beta \Rightarrow \ddot{y} = \ddot{v} \sin \beta + v \dot{\beta} \cos \beta + \omega_\psi v \cos \beta. \quad (3.25)
\]

Combining (3.24) and (3.25) results in an expression for \(\dot{\beta}\) as

\[
\dot{\beta} = \frac{(F_{yfR} + F_{yfL}) \cos \delta + (F_{xfR} + F_{xfL}) \sin \delta + F_{yrL} + F_{yrR}}{(mv \cos \beta)} - \frac{\dot{v} \tan(\beta)}{v} - \omega_\psi. \quad (3.26)
\]

The lateral forces are defined as

\[
F_{yfR} = -C_{afR} \alpha_{fR}, \quad F_{yfL} = -C_{afL} \alpha_{fL}, \quad F_{yrR} = -C_{arR} \alpha_{rR}, \quad F_{yrL} = -C_{arL} \alpha_{rL}, \quad (3.27)
\]
where $C_{\alpha f,r}$ is the lateral stiffness. The lateral tire slip can be defined as in Salmon and Bevly [2014] as

$$\alpha_{fR} = \delta + \frac{v_y + a \omega_y}{v_x + t_f \omega_y}, \quad \alpha_{rR} = \frac{v_y - b \omega_y}{v_x + t_f \omega_y},$$

$$\alpha_{fL} = \delta + \frac{v_y + a \omega_y}{v_x - t_f \omega_y}, \quad \alpha_{rL} = \frac{v_y - b \omega_y}{v_x - t_f \omega_y}.$$ (3.28)

The longitudinal forces are given by

$$F_{xfR} = K_{sfR} \dot{s}_{fR}, \quad F_{xfL} = K_{sfL} \dot{s}_{fL},$$

$$F_{xrR} = K_{srR} \dot{s}_{rR}, \quad F_{xrL} = K_{srL} \dot{s}_{rL},$$ (3.29)

where $K_{sfR, fL}$ is the longitudinal stiffness and $s$, the longitudinal slip is defined as

$$s = \frac{R_w}{v_x} - 1, \quad v_x > R_w,$$

$$s = 1 - \frac{v_x}{R_w}, \quad v_x < R_w.$$ (3.30)

When modeling vehicle roll, which is most noticeable during turns, a roll model could include several springs and dampers of the suspension. Figure 3.6 shows...
a two-state roll plane model taken from Solmaz et al. [2008], where $I_x$ is the roll mass moment of Inertia, $h_{CG}$ is the height from the ground to the center of gravity, $K_\phi$ is the roll stiffness, $C_\phi$ is the roll damping coefficient. The sum of the moments about the roll center gives

$$I_x \ddot{\phi} + C_\phi \dot{\phi} + K_\phi \phi = m h_{CG} (a_y \cos(\phi) + g \sin(\phi)), \quad (3.31)$$

with the assumption that the vehicle’s sprung mass rotates about a fixed point at the centerline of the lateral axis on the ground. If also assuming that $\dot{\phi}$ is constant during a time interval and small changes of the angles, the roll angle can be simplified and written on a state space form as

$$\begin{bmatrix} \dot{\phi} \\ \dot{\omega}_\phi \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ \frac{m h_{CG} - K_\phi}{I_x} & -\frac{C_\phi}{I_x} \end{bmatrix} \begin{bmatrix} \phi \\ \omega_\phi \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{m h_{CG}}{I_x} \end{bmatrix} a_y. \quad (3.32)$$
This chapter explains how the different sensors have been modeled. Also, the sensor characteristics have been investigated. The raw measurements obtained from the INS contain different forms of errors. These errors can for example consist of sensor noise, scale factor errors, bias instability from turn-on, quantization errors and bias drift. To compensate for these errors, a model describing both the true sensor output and the errors can be included in a Kalman filter [Shin, 2005].

Noise is labeled as the high-frequency component of the sensor output and can be seen as an additional signal resulting from interference with the sensor itself or other electronic equipment [Godha, 2006]. Unmodeled behavior of the sensors such as vehicle vibration is an example of a noise source [Bevly and Cobb, 2010]. Other unmodeled effects such as temperature variations will result in a drifting bias, where the measurement values will deviate from its true value. It is important to remove these biases because of the accumulation of errors when integrating the measurement values when relating the measurements with other states [Bevly and Cobb, 2010].

4.1 Accelerometer Model

The measurement data received from the accelerometer is in this thesis modeled as a mix of the true value, a slowly changing bias and random noise. An offset bias is determined by a calibration phase and is then removed from the raw measurements. Hence, this turn-on bias is not included in the model.

An accelerometer, which outputs measured specific force is modeled as in Godha [2006] as

\[ a = f_a + b_a + \varepsilon_a, \]

where \( f_a \) is the specific force, \( b_a \) is the slowly changing sensor bias and \( \varepsilon_a \) is random white Gaussian noise. The bias error \( b_a \) will vary from one turn-on to an-
other and should therefore be estimated with a Kalman filter. A model of the bias is described in Section 4.4.

### 4.2 Gyroscope Model

The gyroscope measures changes in the vehicle’s movement as well as angular rates due to Earth’s rotation. Due to the usage of low-cost MEMS gyroscope sensors, the effect from Earth’s rotation will be included in the sensor noise. A model of the gyroscope can be described as in Godha [2006] as

\[
\ddot{\omega} = \omega + b_g + \epsilon_g,
\]

where \(\omega\) is the true angular rotation, \(b_g\) is the sensor bias and \(\epsilon_g\) is sensor random noise.

### 4.3 Odometer Model

The vehicle’s velocity can be calculated using the wheel speed obtained from the vehicle’s CAN-bus. The most accurate measurement is obtained when using the speed from a wheel axle where the wheels are not driven by the engine namely free rolling wheels. If the wheel’s angular velocity is available as a measurement, the velocity can be modeled as in Lundquist et al. [2011] as

\[
v_x = \frac{R_{w,i} w_i + R_{w,j} w_j}{2},
\]

where \(R_{w,i}\) denotes the wheel radii of wheel \(i\) and \(w_i\) denotes the angular velocity of the wheel \(i\). Also, a virtual yaw rate can be calculated from the wheel’s angular velocity if the length of the wheel axle is known as

\[
\dot{\omega}_\psi = \frac{R_{w,i} w_i - R_{w,j} w_j}{L},
\]

where \(R_{w,i}\) denotes the wheel radii of wheel \(i\), \(w_i\) denotes the angular velocity of the wheel \(i\) and \(L\) is the length of the wheel axle.

The wheel radii can differ from their nominal values because of different air pressure in the wheels or different weight distribution of the vehicle. Hence, the wheel radii should be estimated with a Kalman filter. The actual wheel radii can be defined as the nominal value plus an unknown wheel radii error

\[
R_{w,i} = R_{nom,i} + \delta R_i.
\]

When including the wheel radii error, (4.3) and (4.4), can be reformulated as in Lundquist et al. [2011] to

\[
v_x = \frac{R_{w,i} w_i + R_{w,j} w_j}{2} = \frac{R_{nom,i} w_i + R_{nom,j} w_j}{2} + \frac{\delta R_i w_i + \delta R_j w_j}{2},
\]
and
\[ \dot{\omega}_\psi = \frac{R_{w,i}w_i - R_{w,j}w_j}{L} = \frac{R_{\text{nom},i}w_i - R_{\text{nom},j}w_j}{2} + \frac{\delta R_{i}w_i - \delta R_{j}w_j}{L}. \] (4.7)

The wheel radii error can be modeled as a random walk process
\[ \delta R_{i} = \varepsilon_r, \] (4.8)
where \( \varepsilon_r \) is random white Gaussian noise.

The longitudinal force acting on the tire will generate a motion which in turn produces longitudinal slip. Longitudinal slip must occur at the wheels in order for the vehicle to move. But if the road is slippery then the wheels will spin much faster but not create an equal forward motion and therefore result in a falsely calculated traveled distance. The slip of a wheel can be described as a percentage and is defined as in Bevly and Cobb [2010] as
\[ \% \text{ slip} = \frac{R_{w,i}w_i - v_x}{v_x}, \] (4.9)
where \( R_{w,i} \) is the radius of wheel \( i \), \( w_i \) is the wheel speed of the wheel \( i \) and \( v_x \) is the longitudinal velocity. The longitudinal slip can be described as a scale factor error \( s \), and modeled as
\[ z_{odo} = h(x) + \varepsilon = (1 + s)v_x + \varepsilon, \] (4.10)
where \( s \) is modeled as a random walk process and \( \varepsilon_s \) is random white Gaussian noise
\[ \dot{s} = \varepsilon_s. \] (4.11)

Sideslip occurs when the vehicle is turning quickly such that the track angle or path angle is different from the turning angle. This can also occur when there is low friction between the road and the wheels. One formula to determine the sideslip angle is given by
\[ \beta = \arctan \left( \frac{v_y}{v_x} \right) \approx \frac{v_y}{v_x}, \] (4.12)
where \( v_y \) and \( v_x \) are the lateral and longitudinal velocity, respectively.

If the path angle or direction of the velocity denoted \( \iota \) is given, then the sideslip angle can be determined by
\[ \dot{\beta} = \iota - \psi, \] (4.13)
where \( \psi \) is the vehicle yaw angle given by a gyroscope.
4.4 Sensor Characteristics

When using low-cost MEMS sensors, the slowly changing biases and scale factor errors must be modeled and compensated for. The choice of the model is dependent on the operation time, sensor performance, and working environment [Shin, 2005]. If the operation time is very short, then the errors can be treated practically as constants. If the INS should run for a very long time, then the behaviour of the sensor errors must be carefully investigated. The inertial sensor random errors can in general be expressed as: white noise, random constant, random walk or Gauss-Markov processes [Nassar, 2003].

Since measurements from accelerometers and gyroscopes are integrated to match the states in the Kalman filter, the white noise components are also integrated and this will increase the uncertainty of the velocity and attitude. Velocity random walk (VRW) and angular random walk (ARW) are the terms used to describe these effects [Shin, 2005]. Values of the VRW and the ARW are usually determined through sensor characteristic analysis. Thus, to investigate the behavior of the sensors long data sets have been collected.

4.4.1 Sensor Bias

A bias in an inertial sensor is defined as the average of the output over a specified time when the sensor is stationary [Godha, 2006]. The bias consist of two parts, a deterministic offset called a turn-on bias and a time varying bias-drift. The turn-on bias remains constant over a long time and is easily removed by a calibration phase measuring the mean. The bias-drift is the rate at which the error in a sensor accumulates with time and must be modeled stochastically since it is random by nature [Godha, 2006]. The random process is considered to be stationary, in other words time invariant, and can thus be assumed to be completely defined by its autocorrelation function [Nassar, 2003] which in discrete time is calculated as the autocorrelation sequence. This assumption can be made because by computing the autocorrelation of random data the dependency between the values at one time and the values at another time can be described [Nassar, 2003], where as white random noise cannot be compensated for, as there is no correlation between past and future values [Bevly and Cobb, 2010]. The finite discrete autocorrelation sequence is calculated as

\[ R_{bb}(m) = E[b(k)b(k + m)] = \frac{1}{N - m} \sum_{k=1}^{N-m} b(k)b(k + m), \quad (4.14) \]

where \( k \) is the sampling time and \( m \) is the sampling lag and \( b \) is the process. The Fourier transform of the ACS is called the Power Spectral Density (PSD) and is given by

\[ S_{bb}(e^{j\omega}) = \sum_{m=-\infty}^{\infty} R_{bb}(m)e^{-j\omega m} \quad (4.15) \]

and describes how the energy of the measurements in the time domain is distributed in the frequency domain [Nassar, 2003]. A white noise process has a
4.4 Sensor Characteristics

constant power spectral density (PSD) when the mean has been removed [Nassar, 2003] and the ACS of a stationary white noise process is determined by

\[ R_{bb}(m) = \frac{1}{2\pi} \int_{-\pi}^{\pi} S_{bb}(e^{j\omega})e^{j\omega m} d\omega = \frac{S_{bb}(0)}{2\pi} \int_{-\pi}^{\pi} e^{j\omega m} d\omega = S_{bb}(0)\delta(m), \quad (4.16) \]

where \( \delta \) is the dirac function in continuous time. The ACS of a white noise process will show one peak at lag = 0 and will be uncorrelated for all other lag values. Often, the computed autocorrelation sequence of an INS sensor data set does not follow the appearance of a white sequence process and instead tend to show a correlation between past and future values. To create a model which also output time-correlated noise, the random error component in the sensors can be modeled by passing white noise of zero mean through a certain shaping filter, i.e. a linear dynamic system, to yield an output of time-correlated noise [Nassar, 2003]. This will result in a model which will try to fit the actual residual error component of the inertial sensor.

A first-order Gauss-Markov process is a random process that can be generated from passing white noise through a shaping filter. A stationary Gaussian process that has an exponentially decaying autocorrelation is called a Gauss-Markov process and the equation is formulated as

\[ R_{xx}(\tau) = \sigma^2 e^{-\beta|\tau|}, \quad (4.17) \]

where \( \sigma^2 \) is the noise variance, \( \tau_i \) is the lags and \( \beta_i \) is the inverse of the process correlation time \( \tau_i \).

A Gauss-Markov process varies slowly with time and its most outstanding characteristic is that it can represent bounded uncertainties and is able to model a large number of physical processes [Shin, 2005]. This means that any correlation coefficient at any time lag is less or equal to the correlation at lag = 0, \( R(\tau) \leq R(0) \). Therefore, Gauss-Markov processes are used to model slowly varying sensor errors such as random biases and scale factors. Modeling the biases as a First order Gauss-Markov process results in the expression given in Godha [2006] and Nassar [2003] as

\[ \dot{b}_i = -\frac{1}{\tau_i} b_i + \eta_{bi}, \quad (4.18) \]

where \( \tau_i \) is a Gauss-Markov model parameter obtained by calculating the Auto Correlation Function of a static data-set. The noise \( \eta_{bi} = q_{bi}w(t), \) is a combination of Gaussian white noise and the spectral density \( q \), which can be calculated as \( q_{bi} = \sqrt{2\sigma_i^2/\tau_i} \) where \( \sigma_i \) is another parameter in the Gauss-Markov model determined by calculating the Auto Correlation Function.

In Figure 4.1 a draft over the ACF from an ideal first order Gauss-Markov process is shown. To find \( \tau = \tau_i \) at \( R_{xx}(\tau_i) \) first the value of \( \frac{1}{\tau_i} \sigma_i^2 \) is computed and then \( \tau_i \) corresponds to the value on the x-axis.
Measurements

To investigate the behavior of the sensors and to determine the random process parameters for modeling of the biases, the ACS calculated from a collected static data set of the inertial sensors has been studied. As already mentioned, it is important to first remove the mean. Also, if the inertial sensors suffers from high measurement noise, this should be removed using a low-pass or a band-pass filtering technique. Data received from an low-cost inertial sensor are often buried in high frequency measurement noise [Nassar, 2003] and the output from sensors mounted on and inside a vehicle will contain a combination of actual vehicle motion and noise which in turn will affect the resulting positioning error. Hence, care should be taken so that not actual vehicle motion is removed.

Data sets from an accelerometer and a gyroscope were recorded during five hours and their ACS were computed. In Figure 4.2, the autocorrelation sequence generated from the accelerometer used at Flowscape has been computed without any preprocessing and by investigating the figure, the appearance is close to a white noise sequence, with just one high peak. In Figure 4.3 the Power Spectrum Density (PSD) has been calculated on the ACS generated from the accelerometer measurements. As can be seen there is a lot of high frequency noise that affect the sensor. In Figure 4.4 the collected data set for the gyroscope has been used to generate the autocorrelation. A similar apperance to a white noise sequence can be seen for the pitch and yaw axis whereas the roll measurement is more similar to Figure 4.1. The computed Power Spectrum Density (PSD) from the ACS generated from the gyroscope measurements can be seen in Figure 4.5. In the gyroscope data, there seems to be a lot of high frequency noise, similar to the accelerometer data.
4.4 Sensor Characteristics

Figure 4.2: Computed ACS for a static measurement of the accelerometer.

Figure 4.3: Computed PSD for a static measurement of the accelerometer.
To see the effect when removing the highest frequencies in the ACS a Low-Pass filter was used on the five hour static data set. A butterworth-filter with a cut off
4.4 Sensor Characteristics

frequency chosen to $2 \cdot 10^{-3}$ hertz was used to filter the data with the aim to avoid removing any actual vehicle motion. However, possibly could another method for noise decomposition generate even better results. Figure 4.6 and 4.7 show how the signal energy decreases after removing the high frequency components.

Figure 4.6: Computed PSD for a static measurement of the accelerometer with LP-filtering.
Figure 4.7: Computed PSD for a static measurement of the gyroscope with LP-filtering.

Removing some of the high frequency components results in a similar appearance of a Gauss Markov-process as described above. In Figure 4.8 and 4.9 the autocorrelation has again been computed for the accelerometer and gyroscope, but with the highest frequencies removed.
4.4 Sensor Characteristics

Figure 4.8: Computed ACS for a static measurement of the accelerometer with LP-filtering.

Figure 4.9: Computed ACS for a static measurement of the gyroscope with LP-filtering.
When the high frequencies are removed from the ACS, Figure 4.8 and Figure 4.9 starts to resemble the appearance of the ideal first order Gauss Markov-process. In the same manner as in Figure 4.1, the value of the variance $\sigma^2$ is given when $\tau = 0$. The computation of $\frac{1}{2} \sigma^2$ results in a value on the y-axis which corresponds to the value for $\tau$ on the x-axis. The parameters calculated from these data sets are summarized in Table 4.1.

<table>
<thead>
<tr>
<th></th>
<th>$\sigma^2$</th>
<th>$\tau$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$acc_x$</td>
<td>36.98</td>
<td>$2.83 \cdot 10^4$</td>
</tr>
<tr>
<td>$acc_y$</td>
<td>43.2</td>
<td>7795</td>
</tr>
<tr>
<td>$acc_z$</td>
<td>22.65</td>
<td>$2.582 \cdot 10^4$</td>
</tr>
<tr>
<td>$gyr_x$</td>
<td>0.03129</td>
<td>$1.542 \cdot 10^4$</td>
</tr>
<tr>
<td>$gyr_y$</td>
<td>0.004576</td>
<td>$1.87 \cdot 10^4$</td>
</tr>
<tr>
<td>$gyr_z$</td>
<td>0.0009507</td>
<td>$1.368 \cdot 10^4$</td>
</tr>
</tbody>
</table>

*Table 4.1: Table over the Gauss-Markov parameters.*
Vehicle State Estimation

Measurements received from a GNSS can be combined with the INS measurements using different approaches. Two commonly used methods are the utilization of an error-state estimator and a model-based estimator. An error-state estimator, also known as a complementary filter uses a set of equations called mechanization equations [Godha, 2006]. The mechanization equations convert the measurements received from the accelerometer and the gyroscope. By forming a difference between the received position from a GNSS and the received position from an INS an estimate of the error is obtained, which can be used to correct the position received from the INS. The other method is called a model-based estimator or a total-state implementation. Using the vehicle models described in Chapter 3, an estimated position is computed from the differential equations describing the vehicle dynamics. Measurements received from both a GNSS and an INS are used to correct the estimated position.

The GNSS integration can be implemented differently in both of the mentioned methods. Two possible ways are by utilizing a loosely coupled system or a tightly coupled system. In a loosely coupled system, the GNSS and the INS operate independently from each other. This method is suboptimal however, since the Kalman filter loses information of the correlation between the calculated position from the GNSS and the position from the INS. In a tightly coupled system, the measurement update are modified to incorporate pseudo range and carrier phase measurements from the GNSS. The estimated position from the GNSS can then be corrected by utilizing information from the INS resulting in a more precise system.
5.1 Error-state Kalman Filter

An error-state Kalman filter is a Dead Reckoning integration approach which is set up by describing the INS through a set of equations formulating the dynamics of the inertial sensors. These equations are used to convert the acceleration and angular rate measurements received from the sensors into a position and are often called mechanization equations [Godha, 2006].

The position given by the GNSS receiver is combined with the position from the INS by forming the difference from each system [Zhao, 2011]. The difference is given as a measurement to a Kalman filter which estimates the errors. Feedback is optimal for low-cost MEMS sensors [Shin, 2001] and the estimated errors are therefore used to correct the raw INS measurement output [Godha, 2006].

![Figure 5.1: The integration between GPS and INS in a loosely coupled scheme.](image)

The method results in two different state space representations since there is no interaction between the INS and GNSS other than in the measurement model. Figure 5.1 shows an overview of the implementation scheme. The first state vector is used in the system model which describes the dynamics of the inertial sensors and is only used during time updates. In Figure 5.1 the system model is the black box denoted INS mechanization equations. The second state vector describes the dynamics for the error states and is used to update the Covariance matrix $P$ and during the measurement update. The state space vector for the system model contains position, velocity and attitude. The state space vector for the error dynamics contains the errors in position, velocity, attitude as well as biases for the accelerometer and gyroscope.

The state vector containing the error states are obtained by linearizing the inertial sensors around a nominal state. However, the measurements received from the sensors have to be very accurate, otherwise the linearization will not be valid [Shin, 2001].

**Mechanization Equations**

To receive the position, velocity and attitude from the INS, three steps are formulated. The first step is to subtract the turn-on bias included in the measurement output from the sensors. These biases can be obtained during an initialization phase when the vehicle is stationary.
The second step is to use the gyroscope output data to form a transformation matrix from the body frame to the local navigation frame. This is done by integrating the body sensed rotation rates obtained from the gyroscope. A high-quality gyroscope can measure the angular rates due to Earth’s rotation as well as the rotation of the vehicle. Therefore, by subtracting Earth’s rotation the actual vehicular rotation rate is given, transformed into the local navigation frame.

The third step is to rotate the measured specific force from the body frame to the local navigation frame using the calculated transformation matrix. The acceleration can also be considered as a combination of the sensed gravity and the coriolis effect as well as the actual vehicular generated force. Thus, the gravity force and Coriolis effect have to be removed before integrating the acceleration into velocity and position.

Summarizing the algorithm described above results in a set of inertial navigation equations which can be expressed as

\[
\dot{x}^n = \begin{bmatrix}
\dot{p}^n \\
\dot{v}^n \\
\dot{\psi}^n
\end{bmatrix} = \begin{bmatrix}
C^n_b f^b - (2\Omega_{ie}^n + \Omega_{en}^n)v^n + \gamma^n \\
R^n_b (\Omega_{ib}^b - \Omega_{in}^b)
\end{bmatrix},
\]

(5.1)

where \( \Omega \) represents the skew symmetric matrix form of the vector \( \omega \). For a detailed derivation, see Shin [2001].

Figure 5.2 depicts an overview of a flowchart of the mechanization equations which also include Earth’s rotation, recall Section 2.5.1.

The INS equations described in (5.1) are perturbed to form the error dynamics. A perturbation method linearizes nonlinear differential equations systems [Britting, 2010] and that is the same as applying the Taylor series expansion and retaining only the constant and linear terms [Shin, 2005]. The error dynamics stated in (5.2) is a modified version from the perturbation of the INS mechanization equations presented in Shin [2001],

\[
\begin{bmatrix}
\delta \dot{p}^n \\
\delta \dot{v}^n \\
\delta \dot{\psi}^n
\end{bmatrix} = \begin{bmatrix}
C^n_b \delta f^b + [f^n \times] \delta \psi^n + \Omega_{e} \delta v^n \\
\Omega_{e} \delta v^n - [\omega_{in}^n \times] \delta \psi^n - R^n_b \delta \omega_{ib}^b
\end{bmatrix},
\]

(5.2)
where the $\delta v^n$ is the perturbed velocity, $\delta f^b$ is the perturbed specific force, $\delta \Psi$ is the perturbed attitude and $\delta \omega^b$ is the perturbed angular rate.

The error state space vector is defined as

$$\delta x(t) = \begin{bmatrix} \delta p^{nT} & \delta v^n T & \delta \Psi^n T & \delta b^b_T & \delta b^g_{nT} \end{bmatrix},$$

where the last two states, $\delta b^b_T$ and $\delta b^g_{nT}$, denote the acceleration and gyroscope biases, respectively.

The linearization of the error dynamics is formulated as

$$\dot{\delta x}(t) = \frac{\partial f}{\partial x} \delta x(t) + \frac{\partial f}{\partial u} \delta u(t) = F(t)\delta x(t) + G(t)\delta u(t),$$

where $F$ is the dynamic matrix, $\delta x$ is the error state vector, $G$ is a shaping matrix which transforms the measurements to the correct navigation frame and $\delta u$ consist of the errors in the measurements, $\delta u = [\delta f^b, \delta \omega^b]^T$.

The $F$ and $G$ matrices used in (5.4) are expressed as

$$F = \begin{bmatrix} \Omega_v & C^n_b & 0_{3x3} & 0_{3x3} \\ 0_{3x3} & I_3 & 0_{3x3} & 0_{3x3} \\ 0_{3x3} & 0_{3x3} & \Omega_e & -[\omega^b_{in}] \\ 0_{3x3} & 0_{3x3} & 0_{3x3} & \Omega_e \end{bmatrix}, \quad G = \begin{bmatrix} 0_{3x3} & 0_{3x3} \\ C^n_b & 0_{3x3} \\ 0_{3x3} & -R^n_b \\ 0_{3x3} & 0_{3x3} \end{bmatrix},$$

and $\Omega_v$, $\Omega_e$ as well as $[\omega^b_{in}]$ and $[f^n]$ matrices used in (5.2) are expressed as

$$\Omega_v = \begin{bmatrix} -\frac{vE \tan(\theta) + vH}{N+h} & -2\omega \sin(\phi) & \frac{vN \tan(\theta) + vE}{N+h} + 2\omega \cos(\phi) \\ 2\omega \sin(\phi) + \frac{vE \tan(\theta)}{M+h} & -\frac{vH}{M+h} & -\frac{vN \tan(\theta)}{N+h} + \frac{vE}{M+h} \end{bmatrix},$$

$$\Omega_e = \begin{bmatrix} 0 & -1 \\ \frac{1}{N+h} & 0 \\ 0 & 0 \end{bmatrix},$$

$$[f^n] = \begin{bmatrix} f_x & -f_z & f_y \\ 0 & f_z & -f_x \\ -f_y & f_x & 0 \end{bmatrix}, \quad [\omega^b_{in}] = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix}.$$

The last two matrices, $[f^n]$ and $[\omega^b_{in}]$, present the measurements on a skew-symmetric form.

The system described in (5.4) is given on continuous time form and is rewritten in a discrete time form as

$$\delta x_{k+1} = F_k \delta x_k + w_k,$$

where $F_k$ is the state transition matrix and $w_k$ is the process noise. The state transition matrix $F_k$ is given by

$$F_k \approx I + FT_s = \begin{bmatrix} I_3 & T_s f^n_s & 0_{3x3} & 0_{3x3} \\ 0_{3x3} & I_3 + T_s \Omega_v & T_s [f^n] & 0_{3x3} \\ 0_{3x3} & T_s \Omega_e & I_3 - T_s [\omega^b_{in}] & 0_{3x3} \\ 0_{3x3} & 0_{3x3} & 0_{3x3} & I_3 \end{bmatrix}.$$
and the process noise covariance matrix $Q_k$ can be written as
\[ Q_k \approx GQG^T T_s. \] (5.8)

The observation is given by
\[ z_k = I_3 \left( p_{\text{INS}}^n - p_{\text{GPS}}^n \right). \] (5.9)

The other state space vector used in the system model, formulated from the INS equations (5.1), is defined as
\[ x(t) = \begin{pmatrix} p_n^T & v_n^T & \Psi_n^T \end{pmatrix}^T. \] (5.10)

The state space model for the INS-dynamics is given by
\[ \dot{x}(t) = F_{\text{INS}}(t)x(t) + G_{\text{INS}}(t)u(t), \] (5.11)
where $F_{\text{INS}}$ is the dynamic matrix, $G_{\text{INS}}$ is a shaping matrix which transforms the measurements to the correct navigation frame and $u$ is the input
\[
F_{\text{INS}}(t) = \begin{pmatrix}
0_{3 \times 3} & I_3 & 0_{3 \times 3} \\
0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\
0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3}
\end{pmatrix},
G_{\text{INS}}(t) = \begin{pmatrix}
0_{3 \times 3} & 0_{3 \times 3} \\
C_n^b & 0_{3 \times 3} \\
0_{3 \times 3} & R_n^b
\end{pmatrix}. \] (5.12)

The $F_{\text{INS}}$ and $G_{\text{INS}}$ matrices in a discrete form is given by
\[
F_{k,\text{INS}} = \begin{pmatrix} I_3 & T_s I_3 & 0_{3 \times 3} \\
0_{3 \times 3} & I_3 & 0_{3 \times 3} \\
0_{3 \times 3} & 0_{3 \times 3} & I_3
\end{pmatrix},
G_{k,\text{INS}} = \begin{pmatrix} 0_{3 \times 3} & 0_{3 \times 3} \\
C_n^b & 0_{3 \times 3} \\
0_{3 \times 3} & R_n^b
\end{pmatrix}. \] (5.13)

The INS will operate in prediction mode and receives inputs from the accelerometer and gyroscope. The system is supported through feedback of the estimated errors from the latest measurement update from the error model.

If new GPS data becomes available, a measurement update is carried out:
\[
K_k = P_{k-1} H_k^T (H_k P_{k-1} H_k^T + R_k)^{-1}
\]
\[
\begin{bmatrix}
\delta \hat{x}_k \\
\delta \hat{u}_k
\end{bmatrix} =
\begin{bmatrix}
0_{9 \times 1} \\
0_{6 \times 1}
\end{bmatrix}
+ K_k (y_k - H_k \begin{bmatrix} \hat{x}_{k-1} \\
0_{6 \times 1}\end{bmatrix})
\]
\[
\hat{x}_k = \hat{x}_{k-1} + \delta \hat{x}_k
\]
\[
P_k = P_{k-1} - K_k H_k P_{k-1}
\] (5.14)

If no GPS data:
\[
\hat{u}_{k-1} = \hat{u} + \delta \hat{u}_{k-1}
\]
\[
\hat{x}_{k+1} = F_{k,\text{INS}} \hat{x}_k + G_{k,\text{INS}} \hat{u}
\]
\[
\delta \hat{u}_k = F_k (10:15,10:15) \delta \hat{u}_{k-1}
\]
\[
Q_k = GQG^T T_s
\]
\[
P_k = F_k P_{k-1} F_k^T + Q_k
\] (5.15)

This algorithm is the work of [Blomfeldt and Haverstad, 2008].
5.2 Model-based Kalman Filter

The model-based Dead Reckoning approach uses the velocity and angular measurements to directly calculate the next position, recall Section 3.1, where the measurements are assumed to be constant during a sample period. Using knowledge of vehicle dynamics together with the GNSS generally results in nonlinear system and requires an EKF implementation.

The vehicle dynamic models described in Chapter 3 are utilized to form the motion model. The measurement models derived from Chapter 4 describe the characteristics and the relationship to the vehicle’s state. A flowchart over the input signals can be seen in Figure 5.3.

![Figure 5.3: Block diagram over the integration of the subsystems for the model-based Kalman filter.](image)

The differential equations for the motion models are based on the first DR vehicle
5.3 Data Generation

Two data sets have been used throughout the thesis. First, the two different implementation techniques, error-state and model-based, were investigated using a data set retrieved from Scania CV AB. This data was collected using an accurate system, which is a combined Inertial Navigation and GNSS system developed by Oxford Technical Solutions. The result using this system can be seen in Chapter 6.

The components used at Flowscape AB are of MEMS-type and since the error-state model implementation works best when using fairly accurate measurements, which is often not feasible with low-cost sensors, the model-based filter was chosen for the latter part of the work. Thus, the second data set was collected using the low-cost sensors provided by Flowscape AB and later simulated on the different types of vehicle models and where the ground truth was collected using a high precision GPS from Trimble.

Ground Truth

The documented position accuracy of the GPS receiver R8 from Trimble AB is approximately 3 cm in all directions. To validate the output data from the GPS receiver three tests were performed. A Site Positioning System total station (SPS) from Trible AB was also rented. A SPS is a laser used to measure distances which has a documented position accuracy of 4 mm in tracking mode. The GPS receiver as well as a prisma used to interact with the laser were installed on the roof of a car. The prisma and GPS receiver could not be installed on the same location and
it is only the lateral deviation that has been studied. The tests were performed on a road section with open sky and the result can be seen in Figure 5.4. Studying the deviation when driving at a constant speed around 80 km/h, the documented 3 cm error matches the test results.

Figure 5.4: Plot over the lateral deviation of the GPS receiver compared to the SPS

5.4 Filter Implementation

The integration is separated into two parts, the initial static part and the dynamic process part. During initialization, the vehicle is stationary to determine the initial orientation and sensors noise biases before the vehicle starts to move. During the dynamic process, the filter will first wait for an initial position received from the GPS.

Initial Alignment

Initialization alignment of the angles roll and pitch can be obtained from a steady-state measurement of the accelerometer which under static conditions gives a measurement of the sensed gravity [Godha, 2006],

$$\phi = \text{sign}(a^b_z) \arcsin \left( \frac{a^b_x}{g} \right),$$  \hspace{1cm} (5.18)

$$\theta = -\text{sign}(a^b_z) \arcsin \left( \frac{a^b_y}{g} \right).$$  \hspace{1cm} (5.19)
An initial estimate of the yaw angle can be determined by sensing Earth’s rotation rate under stationary conditions. The roll- and pitch angle, obtained during the initialization phase are rotated to the horizontal frame using

$$\omega_{ib}^h = \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi) & \cos(\phi) \end{bmatrix} \omega_{ib}^b \quad (5.20)$$

and then the yaw angle can be computed as

$$\psi = -\arctan\left(\frac{\omega_{ib}^h)_x}{(\omega_{ib}^h)_y}\right).$$

(5.21)

However, this method might not work if the gyroscope bias and noise exceeds Earth’s rotation rate. Therefore, another approach should be carried out when computing the heading from low-cost MEMS. The initial yaw angle can instead be estimated using a magnetometer. By calculating the initial pitch and roll angles, the measured magnetic field vector can be projected on a horizontal plane parallel to Earth’s surface using [Wang, 2006],

$$\begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix}_H = R_y(\theta)R_x(\phi) \begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix},$$

(5.22)

where $M_x$, $M_y$ and $M_z$ are the magnetometer measurements given in the body frame, $M_{xH}$ and $M_{yH}$ are the projected magnetometer measurements on the horizontal plane.

The angle, $\alpha$, between the vehicle’s forward axis and the magnetic North is computed using a lookup table to minimize processing time [Wang, 2006],

$$\alpha = \begin{cases} 90^\circ, & M_{xH} = 0, M_{yH} < 0, \\ 270^\circ, & M_{xH} = 0, M_{yH} > 0, \\ 180^\circ - \arctan\left(\frac{M_{yH}}{M_{xH}}\right) \times \frac{180}{\pi}, & M_{xH} < 0, \\ -\arctan\left(\frac{M_{yH}}{M_{xH}}\right) \times \frac{180}{\pi}, & M_{xH} > 0, M_{yH} < 0, \\ 360^\circ - \arctan\left(\frac{M_{yH}}{M_{xH}}\right) \times \frac{180}{\pi}, & M_{xH} > 0, M_{yH} > 0. \end{cases}$$

(5.23)

Finally, the vehicle heading $\psi$ can be determined by adding or subtracting a declination angle $\nu$ to correct for true North where the declination angle is determined from a lookup table based on the geographic location or by using a model over the magnetic field, $\psi = \alpha + \nu$.

Last, if the vehicle is in motion, the vehicle heading can be calculated using

$$\psi = \arctan\left(\frac{v_E}{v_N}\right),$$

(5.24)

where $v_E$ is the velocity in the east direction and $v_N$ is the velocity in the north direction obtained from the GPS respectively.
Dynamic Process

The EKF uses measurements from the accelerometer and gyroscope as well as the wheel speed received from the vehicle during the measurement update. During the time update the motion model receives information about current gear, current gear ratio and if the wheel is braking or shifting gear which is given as an input from CAN. These measurements are stored in a vector called input vector.

The filter is running at 10 Hz and GPS positions are received at 1 Hz. The inertial sensors are also sampled at 10 Hz and the data given by CAN is received at approximately 20 Hz. If a new measurement from CAN is not available the previous measurement will instead be given because some signals received from CAN are used in the prediction step in the vehicle models.

For the longitudinal vehicle model consideration has to be taken when the vehicle is braking or shifting gear since this is a behavior not described by the motion model. The unknown behavior is dealt with by increasing the process noise matrix $Q$. Hence, when the vehicle is braking or shifting gear the uncertainty for some of the states is increased, see Appendix A.

Covariance Mapping

Since the Kalman filter is implemented within the ENU-frame, the measurements are transformed from the body frame to the ENU-frame. The measurement noise covariance matrix has to the transformed as well and this is done by using the transformation matrices described in Chapter 2 as

$$ R_{\text{sensor},\text{ENU}} = C^n_b R_{\text{sensor},\text{body}} C^b_n. $$

(5.25)

Out-of-sequence Measurements

The filter receives GPS updates every second, but these measurements are delayed about 0.3 seconds due to data processing. To update the correct state when a GPS measurement is available the states and covariances already computed from the INS measurements are therefore stored between GPS measurements. The internal clock time, the state vector $x$, the input vector to the motion model and the error covariance matrix $P$ are stored each INS update. When a measurement from the GPS becomes available, the filter is rolled back to the time just before the GPS update. Then a time update is performed followed by a measurement update with the position update from the GPS data. Now the filter is "fast forwarded" to current time by doing time updates and measurement updates with the stored INS data.
In this chapter the result from the different implementations is presented. First, the error-state and the model-based implementation are compared using data from OxTS. Second, the vehicle models derived in Chapter 3 are implemented in a model-based Kalman filter and utilized together with data collected from low-cost sensors.

The performance of the different vehicle models has been evaluated with data collected from highways. The highway drive was conducted in a HDV between Södertälje and Järna in Sweden. Results have been obtained from a loosely-coupled GNSS solution in both the error-state implementation as well as the model-based implementation. The generated data was analysed in MATLAB.

### 6.1 Implementation Comparison

The Dead Reckoning vehicle model was utilized in the model-based filter when comparing the performance to the error-state filter, see Figure 6.1. The trajectory for the error-state implementation fluctuates around the trajectory of the ground truth and was hard to tune which could derive from linearization problems. The trajectory of the DR model is very smooth and manage to accurately follow the trajectory of the ground truth. A difference between the implementations is that the error-state implementation uses an accelerometer which is integrated twice to obtain a position whereas the model-based implementation uses the vehicle speed, which is integrated only once to obtain a position. The model-based implementation seems to be less sensitive to inaccurate sensor data compared to the error-state implementation. The model-based implementation has been utilized during the following investigations.
6.2 Performance Evaluation

The results from this section were gathered using low-cost sensors together with the model-based Kalman filter implementation developed in a system written in C++. The test vehicle was a Scania truck with a mass of 38090 kg and the test drive was conducted on the highway between Södertälje and Järna. The data was collected using a real-time embedded system and logged to be able to re-simulate the test drive later. The filter has afterwards been tuned to fit the ground truth data. All except the four-wheel vehicle model were implemented within the real-time system. Also, the bicycle model performs unsatisfactory due to inaccurate vehicle parameters. Figure 6.2 displays the trajectory using the bicycle model with sensor data from OxTS compared to the ground truth. The trajectory of the bicycle model resembles the trajectory of the ground truth, but the bicycle model does not respond quickly enough to changes resulting in a poor fit to the ground truth, after extensively tuning of the filter. The result was even worse when using data collected from the low-cost sensors.
6.2 Performance Evaluation

Figure 6.2: A comparison of the bicycle model and the true trajectory.

Figure 6.3 shows how well the first Dead Reckoning vehicle model and the longitudinal model can estimate the true trajectory during a highway drive between Järna and Södertälje without GPS corrections. The trajectory of the bicycle model is also included. At first, the trajectories from the DR vehicle model and the longitudinal model seem to be fairly similar but when also considering the altitude, depicted in Figure 6.4, there is a clear performance difference between the models. The most noticeable result is the dependency of a correctly estimated set of vehicle parameters for the bicycle model. The bicycle model performs poorly even after a considerable amount of tuning.
**Figure 6.3:** Comparison of the DR vehicle model, the longitudinal model and the bicycle model during a highway drive between Järna and Södertälje.

**Figure 6.4:** The trajectories of the DR vehicle model, the longitudinal model and bicycle model given in 3D. The altitude substantially differs between the DR vehicle model and the longitudinal model.

When correcting the position with a GPS, all of the models align with the ground truth depicted in Figure 6.5. Now, there is not a significant difference between the trajectories. In Figure 6.6 the altitude is corrected but with large fluctuations occurring during GPS outages. However, the amount of error is correlated to a
slightly incorrect synchronization, when updating the position with the GPS, and an obvious conclusion can not be drawn. A deeper analysis addressing the GPS synchronization issue is given in Chapter 7.

Figure 6.5: The performance of the DR vehicle model, the longitudinal model and the bicycle model using GPS corrections.
In Figure 6.7 a new highway drive has been conducted, driving in the opposite direction. The trajectories are again plotted without GPS corrections. The difference between the models is slightly larger than before which is even more distinguishable when considering the altitude, see Figure 6.8.
6.2 Performance Evaluation

Figure 6.7: Comparison of the DR vehicle model, the longitudinal model and the bicycle model during a highway drive between Södertälje and Järna. The longitudinal model manages to closely follow the trajectory of the ground truth throughout the drive whereas the DR vehicle model begins to slightly deviate from the middle of the drive.

Figure 6.8: The trajectories of the DR vehicle model, the longitudinal model and the bicycle model given in 3D.

Again, using corrections from a GPS improves the result so that the models manage to follow the true trajectory correctly, depicted in Figure 6.9. As seen previously when correcting the positions from a GPS, small deviations is present in
the 3-D plot, Figure 6.10, and this is correlated to an incorrect synchronization of the GPS and INS systems, which is further discussed in Chapter 7.

Figure 6.9: The performance of the DR vehicle model, the longitudinal model and the bicycle model using GPS corrections.
During these tests several GPS outages occurred as a result of bridges and signs. One incident from both test drives is depicted in Figure 6.11. The first plot shows the estimated trajectory during a GPS outage when driving from Järna to Södertälje whereas the second plot shows the estimated trajectory during a GPS outage driving from Södertälje to Järna.
Figure 6.11: The above plot depicts a three second GPS outage during a highway drive between Järna and Södertälje. The DR vehicle model and the longitudinal model both deviates about a meter. The second plot shows a ten second GPS outage during a highway drive between Södertälje and Järna where the longitudinal model is closer to the ground truth than the DR vehicle model. The bicycle model manages to fit extraordinary well to the ground truth during the GPS outage but the general performance of this model is poor and this result should be treated with caution.

The absolute position error in East-, North- and Up-directions from both test drives is given in Figure 6.12 and Figure 6.13 during a specific time interval. The absolute error is calculated as the smallest distance between a position from the different vehicle models and a line connecting two GPS positions closest to the estimated position. The absolute position error is plotted during a road section with GPS data available. The deviation from the ground truth is very small in Figure 6.13 where the syncing of the GPS and INS is better than in Figure 6.12.

During the test drive between Järna and Södertälje a total of five GPS outtages occurred and three during the test drive between Södertälje and Järna. The three GPS outages which occurred when driving between Södertälje and Järna are given in a more detailed plot in Figure 6.14 where the start of the GPS outage is marked with an arrow. It is hard to use the position error plot given in Figure 6.12 for a detailed evaluation since the result is affected by a GPS syncing issue, which leads to a larger error than in reality.
Figure 6.12: The absolute position error when driving between Järna and Södertälje during a time interval with GPS corrections.

Figure 6.13: The absolute position error when driving between Södertälje and Järna during a time interval with GPS corrections.
A comparison of the attitudes between the vehicle models for the drive between Järna and Södertälje is given in Figure 6.15. Since the road was mostly flat, there should not be much fluctuation of the roll and pitch angles which also can be seen in the result, where they stay around zero. In Figure 6.16 the attitude estimate is given from the drive in the opposite direction. The most interesting observation is the estimated pitch-angle for the longitudinal model which stays close to zero. The pitch angle affects the position whereas the roll angle does not.
Figure 6.15: A comparison of the attitude from the test drive between Järna and Södertälje.

Figure 6.16: A comparison of the attitude from the test drive between Södertälje and Järna.
7 Conclusions and Future Work

This chapter discusses the result and the conclusions that can be drawn. The second part suggests areas worth investigating in for future work.

7.1 Conclusions

There is a balance between choosing a simple or a complicated model. In this case, the well-known bicycle model was found to be dependent on a correct set of estimated vehicle parameters. The performance was unsatisfactory and the model seems to be sensitive to errors in the parameters. The reason for not investigating further into this model, as well as the four-wheel vehicle model which is the extension of the bicycle model, is because of the difficulty to know the center of gravity in a HDV. Three parameters are dependent on the center of gravity; \( a \), the distance from the center of gravity to the front wheel axis, \( b \), the distance from the center of gravity to the rear wheel axis and \( I_z \), the yaw moment of inertia. The center of gravity changes constantly during a drive in a Heavy Duty Vehicle and these parameters will not stay static. The whole body of an HDV is also damped with springs and is therefore disconnected from the chassis which also complicates the process.

While the Dead Reckoning vehicle model gives a reasonable result since it manages to follow the ground truth, it is not as accurate as the longitudinal vehicle model considering the performance without GPS corrections. The longitudinal vehicle model is also quite suitable for the highway drive since a highway is considerably straight with few sharp turns.

The longitudinal vehicle model used most of the CAN-signals, engine torque, current gear, current gear ratio and if the vehicle is braking or shifting gear. These signals could directly be used in the vehicle model where as the bicycle model, only used the information from the steering wheel angle. This CAN-signal does
not directly transfer the actual angle of the wheel to the indicated steering wheel angle, which also added to the inability of correctly estimate the trajectory when using the bicycle model. The bicycle model should not however be ruled out, but needs more investigation.

The main conclusion from the results is the issues with the GPS integration. The GPS position data was not timestamped, but the general frequency of new updates was known. Hence, the syncing has been carried out manually which has affected the results. Comparing the Dead Reckoning vehicle model and the longitudinal model when not using corrections from a GPS, the results clearly show that the longitudinal model performs better. When adding corrections from the GPS, however, the syncing problem is affecting the outcome. Therefore, the performance during GPS outages may be better than presented here.

One of the differences between the DR vehicle model and the longitudinal model is the estimated pitch angle. Since this angle directly affects the position estimate, the model that can correctly estimate the pitch angle will also perform better. As can be seen in Figure 6.16, the estimated pitch angle for the longitudinal model is closer to zero which is likely to be correct since the highway between Södertälje and Järna is mostly flat. The dynamics of the longitudinal model describes the longitudinal acceleration, where the pitch angle is included. Since the position is affected by the velocity and acceleration, one possible explanation could be that the longitudinal model can more accurately describe the acceleration than the DR vehicle model, which in reality should be the case.

The results suggest that the Dead Reckoning vehicle model and the longitudinal model are both able to estimate the trajectory during GPS outages for an average time of two seconds. The DR vehicle model is able to estimate the trajectory within 15 cm for approximately 1.5-2 seconds before the estimation error grows too large. The longitudinal model performance is slightly better and is able to estimate the position within 15 cm for 1.5-2.5 seconds before the estimation error grows too large. This equals to, driving at the maximum allowed speed at 80 km/h, a position within 15 cm during a distance of approximately 40 meters.

7.2 Future Work

The choice of vehicle models could be further investigated and especially the bicycle model. The biggest issue when using the bicycle model is, as mentioned, the approximated vehicle parameters. These should be correctly estimated for the used vehicle which is needed for a better performance.

The performance is very much affected by the output of the low-cost MEMS, where further investigation of the sensor noise could also improve the position accuracy. In this thesis the sensor biases were modeled with a Gauss-Markov process where the Gauss-Markov parameters were computed from a long static data set. The static data set was analysed and the highest noise components were removed using a Low-Pass filter but another approach for example a wavelet decomposition could further improve the result when calculating the parameters used in the Gauss-Markov process.
7.2 Future Work

The GPS was perhaps the biggest issue and is needed for a correct path estimation. The output from the GPS used in this thesis was only positions, but a GPS is also able to calculate the velocity through a method called doppler shift. Next, using a dual-channel GPS receiver instead of a single-channel GPS, which was done in this thesis, enables the user to also separate the vehicle’s own roll from the declination of the road as described in Ryu [2004]. It is also possible to measure roll and yaw. Utilizing roll and yaw measurements from the GPS for calibration of the gyro biases should improve the heading estimation.

The INS and GPS were not installed in the same place, which also affects the result. There are methods to compensate for this using for example the lever-arm effect, but this was not utilized in this thesis due to limited time and could perhaps improve the result.

A proper fault-detection should also be incorporated such that outliers are neglected but is left to future work.

As a conclusion, the syncing issue when integrating the GPS and INS is of utmost importance to receive a working system should be carried out properly.
Appendix
Appendix

In this chapter three of the vehicle models from Chapter 3 are given in discrete time. Also initial values for the error covariance matrix $P$, measurement noise covariance matrix $R$ and process noise covariance matrix $Q$ are given. Last, the used vehicle parameters are presented. For an review of the notations see page ix.

A.1 Vehicle Models

The mathematical equations for the Dead Reckoning vehicle model, the longitudinal vehicle model and the bicycle model are summarized in this section.
Dead reckoning vehicle model

\[ E_{k+1} = E_k + T_s v_k \cos(\psi_k) \cos(\theta_k) + \frac{T_s^2}{2} a_k \cos(\psi_k) \cos(\theta_k) + \frac{T_s^3}{6} w_k \]

\[ N_{k+1} = N_k + T_s v_k \sin(\psi_k) \cos(\theta_k) + \frac{T_s^2}{2} a_k \sin(\psi_k) \cos(\theta_k) + \frac{T_s^3}{6} w_k \]

\[ U_{k+1} = U_k + T_s v_k \sin(\theta_k) + \frac{T_s^2}{2} a_k \sin(\theta_k) + \frac{T_s^3}{6} w_k \]

\[ v_{x,k+1} = v_{x,k} + T_s a_{x,k} + \frac{T_s^2}{2} w_k \]

\[ v_{y,k+1} = v_{y,k} + T_s a_{y,k} + \frac{T_s^2}{2} w_k \]

\[ v_{z,k+1} = v_{z,k} + T_s a_{z,k} + \frac{T_s^2}{2} w_k \]

\[ a_{x,k+1} = a_{x,k} + T_s w_k \]

\[ a_{y,k+1} = a_{y,k} + T_s w_k \]

\[ a_{z,k+1} = a_{z,k} + T_s w_k \]

\[ \phi_{k+1} = \phi_k + T_s \omega_\phi + \frac{T_s^2}{2} w_k \]

\[ \theta_{k+1} = \theta_k + T_s \omega_\theta + \frac{T_s^2}{2} w_k \]

\[ \psi_{k+1} = \psi_k + T_s \omega_\psi + \frac{T_s^2}{2} w_k \]

\[ \omega_\phi_{k+1} = \omega_\phi_k + T_s w_k \]

\[ \omega_\theta_{k+1} = \omega_\theta_k + T_s w_k \]

\[ \omega_\psi_{k+1} = \omega_\psi_k + T_s w_k \]

\[ s_{k+1} = s_k + T_s w_k \]

\[ \beta_{k+1} = \beta_k + T_s w_k \]

\[ \delta_{rw,k+1} = \delta_{rw,k} + T_s w_k \]

\[ \delta_{lw,k+1} = \delta_{lw,k} + T_s w_k \]

\[ b_{acc,x,k+1} = (1 - \frac{T_s}{\tau_{ax}}) b_{acc,x,k} + T_s q_{ax} w_k \]

\[ b_{acc,y,k+1} = (1 - \frac{T_s}{\tau_{ay}}) b_{acc,y,k} + T_s q_{ay} w_k \]

\[ b_{acc,z,k+1} = (1 - \frac{T_s}{\tau_{az}}) b_{acc,z,k} + T_s q_{az} w_k \]

\[ b_{gyr,x,k+1} = (1 - \frac{T_s}{\tau_{gx}}) b_{gyr,x,k} + T_s q_{gx} w_k \]

\[ b_{gyr,y,k+1} = (1 - \frac{T_s}{\tau_{gy}}) b_{gyr,y,k} + T_s q_{gy} w_k \]

\[ b_{gyr,z,k+1} = (1 - \frac{T_s}{\tau_{gz}}) b_{gyr,z,k} + T_s q_{gz} w_k \]
Longitudinal vehicle model

\[ E_{k+1} = E_k + T_s v_k \cos(\psi_k) \cos(\theta_k) + \frac{T_s^2}{2} a_k \cos(\psi_k) \cos(\theta_k) + \frac{T_s^3}{6} w_k \]

\[ N_{k+1} = N_k + T_s v_k \sin(\psi_k) \cos(\theta_k) + \frac{T_s^2}{2} a_k \sin(\psi_k) \cos(\theta_k) + \frac{T_s^3}{6} w_k \]

\[ U_{k+1} = U_k + T_s v_k \sin(\theta_k) + \frac{T_s^2}{2} a_k \sin(\theta_k) + \frac{T_s^3}{6} w_k \]

\[ v_{x,k+1} = v_{x,k} + T_s (\kappa_1 T_{e,k} - \kappa_2 v_k^2 - \kappa_3 \cos(\theta_k) - \kappa_4 \sin(\theta_k)) + \frac{T_s^2}{2} w_k \]

\[ v_{y,k+1} = v_{y,k} + T_s a_{y,k} + \frac{T_s^2}{2} w_k \]

\[ v_{z,k+1} = v_{z,k} + T_s a_{z,k} + \frac{T_s^2}{2} w_k \]

\[ a_{x,k+1} = a_{x,k} - 2 T_s \kappa_2 v_k + T_s w_k \]

\[ a_{y,k+1} = a_{y,k} + T_s w_k \]

\[ a_{z,k+1} = a_{z,k} + T_s w_k \]

\[ \phi_{k+1} = \phi_k + T_s \omega_{\phi} + \frac{T_s^2}{2} w_k \]

\[ \theta_{k+1} = \theta_k + T_s \omega_{\theta} + \frac{T_s^2}{2} w_k \]

\[ \psi_{k+1} = \psi_k + T_s \omega_{\psi} + \frac{T_s^2}{2} w_k \]

\[ \omega_{\phi,k+1} = \omega_{\phi,k} + T_s w_k \]

\[ \omega_{\theta,k+1} = \omega_{\theta,k} + T_s w_k \]

\[ \omega_{\psi,k+1} = \omega_{\psi,k} + T_s w_k \]

\[ s_{k+1} = s_k + T_s w_k \]

\[ \beta_{k+1} = \beta_k + T_s w_k \]

\[ \delta_{rw,k+1} = \delta_{rw,k} + T_s w_k \]

\[ \delta_{lw,k+1} = \delta_{lw,k} + T_s w_k \]

\[ b_{acc,x,k+1} = (1 - \frac{T_s}{\tau_{ax}}) b_{acc,x,k} + T_s q_{ax} w_k \]

\[ b_{acc,y,k+1} = (1 - \frac{T_s}{\tau_{ay}}) b_{acc,y,k} + T_s q_{ay} w_k \]

\[ b_{acc,z,k+1} = (1 - \frac{T_s}{\tau_{az}}) b_{acc,z,k} + T_s q_{az} w_k \]

\[ b_{gyr,x,k+1} = (1 - \frac{T_s}{\tau_{gx}}) b_{gyr,x,k} + T_s q_{gx} w_k \]

\[ b_{gyr,y,k+1} = (1 - \frac{T_s}{\tau_{gy}}) b_{gyr,y,k} + T_s q_{gy} w_k \]

\[ b_{gyr,z,k+1} = (1 - \frac{T_s}{\tau_{gz}}) b_{gyr,z,k} + T_s q_{gz} w_k \]
Bicycle model

\[ E_{k+1} = E_k + T_s v_k \cos(\psi_k + \beta_k) \cos(\theta_k) + \frac{T_s^2}{2} a_k \cos(\psi_k + \beta_k) \cos(\theta_k) + \frac{T_s^3}{6} w_k \]

\[ N_{k+1} = N_k + T_s v_k \sin(\psi_k + \beta_k) \cos(\theta_k) + \frac{T_s^2}{2} a_k \sin(\psi_k + \beta_k) \cos(\theta_k) + \frac{T_s^3}{6} w_k \]

\[ U_{k+1} = U_k + T_s v_k \sin(\theta_k) + \frac{T_s^2}{2} a_k \sin(\theta_k) + \frac{T_s^3}{6} w_k \]

\[ v_{x,k+1} = v_{x,k} + T_s a_x + \frac{T_s^2}{2} w_k \]

\[ v_{y,k+1} = v_{y,k} + T_s a_y + \frac{T_s^2}{2} w_k \]

\[ v_{z,k+1} = v_{z,k} + T_s a_x + \frac{T_s^2}{2} w_k \]

\[ a_{x,k+1} = a_{x,k} + T_s w_k \]

\[ a_{y,k+1} = a_{y,k} + T_s w_k \]

\[ a_{z,k+1} = a_{z,k} + T_s w_k \]

\[ \phi_{k+1} = \phi_k + T_s \omega_\phi + \frac{T_s^2}{2} w_k \]

\[ \theta_{k+1} = \theta_k + T_s \omega_\theta + \frac{T_s^2}{2} w_k \]

\[ \psi_{k+1} = \psi_k + T_s \omega_\psi + \frac{T_s^2}{2} w_k \]

\[ \omega_{\phi,k+1} = \omega_{\phi,k} + T_s w_k \]

\[ \omega_{\theta,k+1} = \omega_{\theta,k} + T_s w_k \]

\[ \omega_{\psi,k+1} = (1 + T_s \frac{-a^2 C_{af} - b^2 C_{ar}}{I_z v_{x,k}}) \omega_{\psi,k} + T_s (\frac{b C_{ar} - a C_{af}}{I_z} - \omega_{\psi,k} + \frac{a C_{af}}{I_z} \delta_k + w_k) \]

\[ s_{k+1} = s_k + T_s w_k \]

\[ \beta_{k+1} = (1 + T_s \frac{-C_{af} - C_{ar}}{m v_{x,k}^2}) \beta_k + T_s ((\frac{b C_{ar} - a C_{af}}{m v_{x,k}^2} - 1) \omega_{\psi,k} + \frac{a C_{af}}{m v_{x,k}^2} \delta_k + w_k) \]

\[ \delta_{rw,k+1} = \delta_{rw,k} + T_s w_k \]

\[ \delta_{lw,k+1} = \delta_{lw,k} + T_s w_k \]

\[ b_{acc,x,k+1} = (1 - \frac{T_s}{\tau_{ax}}) b_{acc,x,k} + T_s q_{ax} w_k \]

\[ b_{acc,y,k+1} = (1 - \frac{T_s}{\tau_{ay}}) b_{acc,y,k} + T_s q_{ay} w_k \]

\[ b_{acc,z,k+1} = (1 - \frac{T_s}{\tau_{az}}) b_{acc,z,k} + T_s q_{az} w_k \]

\[ b_{gyr,x,k+1} = (1 - \frac{T_s}{\tau_{gx}}) b_{gyr,x,k} + T_s q_{gx} w_k \]

\[ b_{gyr,y,k+1} = (1 - \frac{T_s}{\tau_{gy}}) b_{gyr,y,k} + T_s q_{gy} w_k \]

\[ b_{gyr,z,k+1} = (1 - \frac{T_s}{\tau_{gz}}) b_{gyr,z,k} + T_s q_{gz} w_k \]
A.2 Covariance Matrices

Initial Error Covariance Matrix

\[
\begin{pmatrix}
P(p_x, p_x) &= 1000 \\
P(p_y, p_y) &= 1000 \\
P(p_z, p_z) &= 1000 \\
P(v_x, v_x) &= 10 \\
P(v_y, v_y) &= 10 \\
P(v_z, v_z) &= 10 \\
P(a_x, a_x) &= 10 \\
P(a_y, a_y) &= 10 \\
P(a_z, a_z) &= 10 \\
P(\phi, \phi) &= 0.01 \\
P(\theta, \theta) &= 0.01 \\
P(\psi, \psi) &= 0.01 \\
P(\omega_\phi, \omega_\phi) &= 0.03 \\
P(\omega_\theta, \omega_\theta) &= 0.03 \\
P(\omega_\psi, \omega_\psi) &= 0.001 \\
P(s, s) &= 1 e^{-6} \\
P(\beta, \beta) &= 1 e^{-6} \\
P(\delta_{ru}, \delta_{ru}) &= 1 e^{-4} \\
P(\tilde{\delta}_{ru}, \tilde{\delta}_{ru}) &= 1 e^{-4} \\
P(b_{acc,x}, b_{acc,x}) &= 0.25 \\
P(b_{acc,y}, b_{acc,y}) &= 0.25 \\
P(b_{acc,z}, b_{acc,z}) &= 0.25 \\
P(b_{gyr,x}, b_{gyr,x}) &= 4 e^{-4} \\
P(b_{gyr,y}, b_{gyr,y}) &= 5 e^{-4} \\
P(b_{gyr,z}, b_{gyr,z}) &= 6 e^{-4}
\end{pmatrix}
\]

Measurement Noise Covariance Matrices

\[
R_{gyro} = \begin{bmatrix}
0.5 & 0 & 0 \\
0 & 0.04 & 0 \\
0 & 0 & 0.06
\end{bmatrix}
\]

\[
R_{acc} = \begin{bmatrix}
0.02 & 0 & 0 \\
0 & 0.04 & 0 \\
0 & 0 & 0.3
\end{bmatrix}
\]

\[
R_{odo} = \begin{bmatrix}
0.5 & 0 \\
0 & 0.0001
\end{bmatrix}
\]
### Process Noise Covariance Matrix

\[
\begin{pmatrix}
Q(p_x, p_x) &= 4.5 \\
Q(p_y, p_y) &= 1 \\
Q(p_z, p_z) &= 1 \\
Q(v_x, v_x) &= 9 \\
Q(v_y, v_y) &= 1 \\
Q(v_z, v_z) &= 1 \\
Q(a_x, a_x) &= 4 \\
Q(a_y, a_y) &= 4 \\
Q(a_z, a_z) &= 4 \\
Q(\phi, \phi) &= 0.1 \\
Q(\theta, \theta) &= 0.1 \\
Q(\psi, \psi) &= 0.1 \\
Q(\omega_{\phi}, \omega_{\phi}) &= 1 \\
Q(\omega_{\theta}, \omega_{\theta}) &= 1 \\
Q(\omega_{\psi}, \omega_{\psi}) &= 0.8 \\
Q(s, s) &= 5\times 10^{-7} \\
Q(\beta, \beta) &= 1\times 10^{-3} \\
Q(\delta_{rw}, \delta_{rw}) &= 1\times 10^{-4} \\
Q(\delta_{uw}, \delta_{uw}) &= 1\times 10^{-4} \\
Q(b_{acc,x}, b_{acc,x}) &= 0.25 \\
Q(b_{acc,y}, b_{acc,y}) &= 0.25 \\
Q(b_{acc,z}, b_{acc,z}) &= 0.25 \\
Q(b_{gyr,x}, b_{gyr,x}) &= 3\times 10^{-3} \\
Q(b_{gyr,y}, b_{gyr,y}) &= 3\times 10^{-3} \\
Q(b_{gyr,z}, b_{gyr,z}) &= 3\times 10^{-3}
\end{pmatrix}
\]

### Process Noise Covariance Matrix during brake or gear shift

The other values are not changed

\[
\begin{pmatrix}
Q_b(p_x, p_x) = 50 \\
Q_b(p_y, p_y) = 50 \\
Q_b(p_z, p_z) = 50 \\
Q_b(v_x, v_x) = 10 \\
Q_b(v_y, v_y) = 10 \\
Q_b(v_z, v_z) = 10 \\
Q_b(a_x, a_x) = 10 \\
Q_b(a_y, a_y) = 10 \\
Q_b(a_z, a_z) = 10
\end{pmatrix}
\]
A.3 Vehicle Parameters

Mass, $m = 38210 \text{ [kg]}$

Wheel inertia, $J_w = 65.1 \text{ [kgm]}$

Engine inertia, $J_e = 3.5 \text{ [kgm]}$

Airdrag coefficient, $C_d = 0.7$ [-]

Cross section area, $A_a = 10.0 \text{ [m]}$

Density of air, $\rho_a = 1.125 \text{ [kg/m]}$

Roll resistance coefficient, $c_r = 0.009 \text{ [m/s]}$

Final gear ratio, $i_f = 2.71$ [-]

Final gear efficiency, $\eta_f = 0.98$ [-]

Transmission ratio for the actual gear, $i_t =$ Received from CAN [-]

Transmission efficiency, $\eta_t =$ Given by a look-up table [-]

Engine Torque, $T_e =$ Received from CAN [Nm]

Distance from Center of Gravity to front axis, $a = 4.0 \text{ [m]}$

Distance from Center of Gravity to rear axis, $b = 3.5 \text{ [m]}$

Cornering stiffness front, $C_{\alpha f} = 120000 \text{ [N/rad]}$

Cornering stiffness rear, $C_{\alpha r} = 420000 \text{ [N/rad]}$

Yaw moment of inertia, $I_z = 5000 \text{ [kgm]}$
Bibliography


Upphovsrätt

Detta dokument hålls tillgängligt på Internet — eller dess framtida ersättare — under 25 år från publiceringsdatum under förutsättning att inga extraordinära omständigheter uppstår.

Tillgång till dokumentet innebär tillstånd för var och en att läsa, ladda ner, skriva ut enstaka kopior för enskilt bruk och att använda det oförändrat för icke-kommersiell forskning och för undervisning. Överföring av upphovsrätten vid en senare tidpunkt kan inte upphäva detta tillstånd. All annan användning av dokumentet kräver upphovsmannens medgivande. För att garantera äktheten, säkerheten och tillgängligheten finns det lösningar av teknisk och administrativ art.

Upphovsmannens ideella rätt innefattar rätt att bli nämnd som upphovsman i den omfattning som god sed kräver vid användning av dokumentet på ovan beskrivna sätt samt skydd mot att dokumentet ändras eller presenteras i sådan form eller i sådant sammanhang som är kränkande för upphovsmannens litterära eller konstnärliga anseende eller egenart.

För ytterligare information om Linköping University Electronic Press se förlagets hemsida http://www.ep.liu.se/

Copyright

The publishers will keep this document online on the Internet — or its possible replacement — for a period of 25 years from the date of publication barring exceptional circumstances.

The online availability of the document implies a permanent permission for anyone to read, to download, to print out single copies for his/her own use and to use it unchanged for any non-commercial research and educational purpose. Subsequent transfers of copyright cannot revoke this permission. All other uses of the document are conditional on the consent of the copyright owner. The publisher has taken technical and administrative measures to assure authenticity, security and accessibility.

According to intellectual property law the author has the right to be mentioned when his/her work is accessed as described above and to be protected against infringement.

For additional information about the Linköping University Electronic Press and its procedures for publication and for assurance of document integrity, please refer to its www home page: http://www.ep.liu.se/

© David Enberg