Map building and gearshift optimization for articulated haulers

Magnus Saaf, Aseel Hana

Master’s thesis, February 2011

Mälardalen University

School of innovation, design and engineering
Map building and gearshift optimization for articulated haulers

This report was written by
Magnus Saaf, Aseel Hana

Supervisor
Ph.D Gianantonio Bortolin

Examiner
Professor Lars Asplund

Mälardalen University
School of innovation, design and engineering
Box 883, 731 23 Västerås
Sweden

http://www.mdh.se/idt

Release date: 2011-02-01
Category: Public
Edition: First
Comments: This report is part of the requirements for the degree of Master of Science in Robotics Engineering. The report represents 30 ECTS points.
Rights: © Magnus Saaf, Aseel Hana 2011
Abstract

Increasing environmental awareness influences many entities, spanning from the private individual to the largest companies in the world, to help preserve our available resources. For heavy duty truck manufacturers this commitment can be seen, for instance, in the many efforts made to increase fuel efficiency and thereby decrease fuel consumption. This master’s thesis presents a solution to the problem, where both time and fuel are saved thanks to better gearshift decision algorithms inside the automatic gearboxes of Volvo CE articulated haulers. The gearshift algorithms investigated are based on look-ahead information of upcoming road sections, which is obtained by iterative map building. The results were successful, but also indicate that more work should be carried out in this area.

Keywords: Iterative map building, Simultaneous localization and map building, Kalman filtering, Fuzzy logic, Heavy off-road vehicles, Gearshift strategies.
Preface

This master’s thesis report documents the results of 20 weeks of work carried out at Volvo CE in Eskilstuna. The completion of the report and the presentation of its contents are part of the requirements for obtaining the Master of Science in Robotics Engineering degree at Mälardalens University. The copyrights of this report belong to its authors.
Acknowledgements

First, we would like to thank Gianantonio Bortolin, who was our supervisor at Volvo CE. Your support, guidance and enthusiasm made the project easier in many ways. Special thanks to Ulf Andersson at LTU for lending the DGPS equipment, and sharing your experiences and knowledge of automotive articulated vehicles. We also express our gratitude to Robin Lilja and Peter Nicol for proof reading this report from a technical and linguistic point of view, respectively. Finally, our thanks to Sara Brandberg, for helping us produce the main part of the images used in the report.

I would like to thank my mother and my father for their immeasurable encouragement and support. You have both played great parts in the things I have accomplished so far in my life. For that, I will be forever grateful. Special thanks also to my friends and colleagues Robin Lilja and Robin Pettersson. You two made the years at MDH easier and more enjoyable, and I believe we all share experiences that we will never forget. Lastly, I would like to thank my master’s thesis partner and friend Aseel Hana for a, in many ways, fantastic time at Volvo.

Magnus Saaf

First of all I would like to thank my family for their support, trust and all the good things they have done for me. I also express my gratitude to my friend Dani Barkah, who also supported me all the time. Further, I am grateful to my friend Sahag Normanian for the great time at MDH, which made my studies easier. A big thank-you to my master’s thesis partner, colleague and friend, Magnus Saaf for the great time at Volvo.

Aseel Hana
Contents

List of Figures vii

List of Tables ix

1 Introduction 1
  1.1 Background .......................... 1
  1.2 Articulated haulers ............................ 1
  1.3 Problem statement ........................... 2
  1.4 Simulation software ........................... 3
  1.5 Report overview ........................... 3

2 Related work 4
  2.1 Look-ahead systems in heavy duty trucks ......................... 4
  2.2 Iterative map building ................................ 4
  2.3 Gearshift strategies ................................ 5
  2.4 Commonly used sensors ............................ 6

3 Positioning systems 7
  3.1 Dead reckoning systems and odometry .......................... 7
  3.2 Beacon based positioning systems ............................ 7
  3.3 Global Navigation Satellite System (GNSS) .................... 8
    3.3.1 WAAS/EGNOS .................................. 8
    3.3.2 DGPS ........................................ 8
    3.3.3 Ideal positioning ................................ 9
    3.3.4 Commercial usage ............................ 10

4 Available sensors and sensor modeling 11
  4.1 GPS receiver .................................. 11
  4.2 Tilt sensor .................................... 13
  4.3 Articulation sensor ................................ 15
  4.4 Tachometer .................................... 15

5 The Kalman filter 16
  5.1 Brief introduction ................................ 16
  5.2 Basic components ................................ 16
    5.2.1 State vector ................................ 17
    5.2.2 Dynamic model ................................ 17
    5.2.3 Observation model ............................ 17
  5.3 Discrete Kalman filter algorithm ............................ 18
## CONTENTS

5.4 Filter parameters and tuning ........................................ 19
5.5 Extended Kalman filter (EKF) .................................. 19

### 6 Map matching algorithms 21
6.1 Data storage ..................................................... 21
6.2 Data fusion algorithms .......................................... 22
   6.2.1 General data fusion algorithm ...................... 23
   6.2.2 Measurement modification algorithm ............ 23
6.3 Algorithm selection ............................................. 23

### 7 Vehicle modeling 24
7.1 Articulated hauler model ........................................ 24
   7.1.1 Engine .................................................. 24
   7.1.2 Torque converter ...................................... 25
   7.1.3 Transmission .......................................... 25
   7.1.4 Dropbox ............................................... 26
   7.1.5 Cardan shaft .......................................... 26
   7.1.6 Center differential ................................. 26
   7.1.7 Axle hub .............................................. 27
   7.1.8 Wheels ................................................ 27
   7.1.9 Chassis ................................................. 27
7.2 External forces .................................................. 28
   7.2.1 Air resistance force ................................. 28
   7.2.2 Rolling resistance force ......................... 28
   7.2.3 Gravitational pull ................................... 28
7.3 Track model ..................................................... 29
7.4 Driver .......................................................... 29
7.5 Vehicle motion .................................................. 29

### 8 Observer modeling 30
8.1 System model .................................................... 30
   8.1.1 Spatial sampling ..................................... 30
   8.1.2 Spatially sampled system model .................. 31
8.2 Extended Kalman filter implementation .................. 31

### 9 Gearshift strategies 34
9.1 Existing gearshift strategy ................................... 34
9.2 First approach - Workload prediction .................... 34
   9.2.1 Workload prediction .................................. 34
   9.2.2 Strategy I - "Simple" .............................. 35
9.3 Further development .......................................... 35
   9.3.1 Fuzzy logic ........................................... 36
   9.3.2 Strategy II - "Fuzzy" ............................... 38
9.4 Algorithm selection ............................................. 38

### 10 Simulation and results 39
10.1 Positioning and map matching ................................. 39
   10.1.1 Five runs with a full load on a level track .... 39
   10.1.2 Five runs with zero load on a level track ...... 42
   10.1.3 Five runs with full load on a hilly track ...... 44
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Volvo CE articulated hauler</td>
<td>2</td>
</tr>
<tr>
<td>3.1</td>
<td>Concept of dead reckoning</td>
<td>8</td>
</tr>
<tr>
<td>3.2</td>
<td>Differential GPS</td>
<td>9</td>
</tr>
<tr>
<td>3.3</td>
<td>Trilateration</td>
<td>10</td>
</tr>
<tr>
<td>4.1</td>
<td>Validation of the tilt sensor model</td>
<td>14</td>
</tr>
<tr>
<td>5.1</td>
<td>Discrete Kalman filter cycle</td>
<td>18</td>
</tr>
<tr>
<td>6.1</td>
<td>Third criterion for data fusion</td>
<td>22</td>
</tr>
<tr>
<td>7.1</td>
<td>Schematic of the articulated hauler’s powertrain</td>
<td>24</td>
</tr>
<tr>
<td>7.2</td>
<td>Schematic of external forces affecting the vehicle</td>
<td>28</td>
</tr>
<tr>
<td>9.1</td>
<td>Prediction horizon</td>
<td>35</td>
</tr>
<tr>
<td>9.2</td>
<td>Predicted work when simulating on a hilly track</td>
<td>35</td>
</tr>
<tr>
<td>9.3</td>
<td>Membership function</td>
<td>37</td>
</tr>
<tr>
<td>9.4</td>
<td>Fuzzy AND operator</td>
<td>38</td>
</tr>
<tr>
<td>9.5</td>
<td>Fuzzy OR operator</td>
<td>38</td>
</tr>
<tr>
<td>10.1</td>
<td>Vehicle position with a full load on a level track</td>
<td>40</td>
</tr>
<tr>
<td>10.2</td>
<td>Track slope with a full load on a level track</td>
<td>41</td>
</tr>
<tr>
<td>10.3</td>
<td>Coefficient of rolling resistance with a full load on a level track</td>
<td>41</td>
</tr>
<tr>
<td>10.4</td>
<td>Vehicle position with zero load on a level track</td>
<td>43</td>
</tr>
<tr>
<td>10.5</td>
<td>Track slope with zero load on a level track</td>
<td>44</td>
</tr>
<tr>
<td>10.6</td>
<td>Coefficient of rolling resistance with zero load on a level track</td>
<td>44</td>
</tr>
<tr>
<td>10.7</td>
<td>Vehicle position with a full load on a hilly track</td>
<td>45</td>
</tr>
<tr>
<td>10.8</td>
<td>Track slope with a full load on a hilly track</td>
<td>46</td>
</tr>
<tr>
<td>10.9</td>
<td>Coefficient of rolling resistance with a full load on a hilly track</td>
<td>47</td>
</tr>
<tr>
<td>10.10</td>
<td>Vehicle position with zero load on a hilly track</td>
<td>48</td>
</tr>
<tr>
<td>10.11</td>
<td>Track slope with zero load on a hilly track</td>
<td>49</td>
</tr>
<tr>
<td>10.12</td>
<td>Coefficient of rolling resistance with zero load on a hilly track</td>
<td>49</td>
</tr>
<tr>
<td>10.13</td>
<td>Track section used for gearshift strategy analyzing</td>
<td>50</td>
</tr>
<tr>
<td>10.14</td>
<td>Test results using existing gearshift strategy</td>
<td>51</td>
</tr>
<tr>
<td>10.15</td>
<td>Test results using the &quot;Simple&quot; gearshift strategy</td>
<td>52</td>
</tr>
<tr>
<td>10.16</td>
<td>Test results using the &quot;Fuzzy&quot; gearshift strategy</td>
<td>53</td>
</tr>
<tr>
<td>10.17</td>
<td>Test results using a tuned revision of the &quot;Fuzzy&quot; gearshift strategy</td>
<td>54</td>
</tr>
</tbody>
</table>
LIST OF FIGURES

A.1 Correlation between articulation angle and turning radius. . . . . 60
B.1 Local Tangent Plane coordinate system. . . . . . . . . . . . . . . 61
# List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>Garmin GPS 18 USB electrical characteristics</td>
<td>12</td>
</tr>
<tr>
<td>4.2</td>
<td>Garmin GPS 18 USB receiver performance</td>
<td>12</td>
</tr>
<tr>
<td>4.3</td>
<td>Kavlico TS905 tilt sensor electrical characteristics</td>
<td>14</td>
</tr>
<tr>
<td>4.4</td>
<td>Kavlico TS905 tilt sensor performance</td>
<td>14</td>
</tr>
<tr>
<td>10.1</td>
<td>Improvement of positioning accuracy due to iterative map building when simulating with a full load on a level track.</td>
<td>41</td>
</tr>
<tr>
<td>10.2</td>
<td>Improvement of positioning accuracy due to iterative map building when simulating with zero load on a level track.</td>
<td>44</td>
</tr>
<tr>
<td>10.3</td>
<td>Improvement of positioning accuracy due to iterative map building when simulating with a full load on a hilly track.</td>
<td>46</td>
</tr>
<tr>
<td>10.4</td>
<td>Improvement of positioning accuracy due to iterative map building when simulating with zero load on a hilly track.</td>
<td>49</td>
</tr>
<tr>
<td>10.5</td>
<td>Comparison between analyzed gearshift strategies</td>
<td>55</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

This chapter first introduces the reader to the background and aim of the project. Articulated haulers are thereafter explained at a basic level, and subsequently the problem of this thesis is stated. Finally, crucial software is described followed by a complete report overview.

1.1 Background

Today’s increasing environmental awareness influences many entities spanning from the private individual to the largest companies in the world, to help preserve our available resources [1]. For heavy duty truck manufacturers this commitment can be seen, for instance, in the many efforts made to increase fuel efficiency and thereby decrease fuel consumption. Fuel efficiency can today be increased in a number of ways. One of them is to use look-ahead information [2]. Such systems download road information of their current area (often using a GPS receiver and a wireless internet connection) and then estimate what the road ahead of them looks like. Knowledge of upcoming road sections makes it possible to chose a gearshift strategy that not only saves fuel, but also causes less wear to the truck itself [3].

1.2 Articulated haulers

An articulated hauler is basically a dump truck designed to operate off-road. Both dump trucks and articulated haulers are built for transporting large amounts of material from one location to another, but there are situations where it is advantageous to take a shorter route over rough terrain. The articulated hauler consists of two main parts, a tractor and a fixed mounted trailer. Instead of turning the front wheels as on most of today’s cars, steering is obtained from the articulation. This provides good off-road mobility with less stress on the frames and smaller turning radii. The steering wheel controls two hydraulic cylinders that act around the hinge between the tractor and the trailer, adjusting the yaw angle. Since this hinge allows free rotation around the roll axis, the trailer is not affected by the tractor driving over a bump causing it to roll, and vice versa.
CHAPTER 1. INTRODUCTION

The engine and the driveline constitute a motor vehicle’s powertrain. There are different definitions of what a driveline is, but in this thesis the driveline definition includes the clutch, transmission, shafts, differentials and wheels. Such a definition explains the very purpose of the driveline - To transfer torque and angular velocity from the engine to the wheels. Articulated haulers use a torque converter and a lockup instead of a clutch. Together with automatic transmission, this leads to direct economic and environmental benefits such as increased fuel efficiency. Another benefit is driving force while shifting, also referred to as powershift. The three axles between each wheel pair and the axle between the tractor and the trailer are all equipped with differentials that can either be locked or unlocked. In locked mode, a differential ensures both wheels have equal rotational speed but not necessarily equal torque. In slippery terrain it is not unusual to have different traction under each wheel, and a locked differential might increase the total traction of the vehicle. For comparison, a situation where the vehicle is turning in terrain with high traction requires different rotational speed of the wheels in a wheel pair in order to avoid tire wear and stress to the axle. To achieve this the driver needs to set the differentials to unlocked mode, allowing the wheels to spin at different rotational speeds. Differentials are controlled either manually, or automatically by the current automatic traction control (ATC) system.

1.3 Problem statement

This thesis is about making smart gear shift decisions based on look-ahead information of upcoming road sections. Much work in this particular area has already been done, and many vehicles use these systems for cruise control and gear shift decisions today [3] [4]. For example, a long distance lorry can download existing detailed maps of its current location using a mobile Internet connection, and therefore also information about the upcoming road (Information is stored as maps in a GPS device, or downloaded via satellite.). However, information such as road surface characteristics and topology is seldom available in off-road terrain like quarries, gravel pits or mines (where articulated haulers operate),
thus making look-ahead information based decisions difficult [3]. One of the two main problems examined in this thesis is how to obtain this crucial information. By taking advantage of the fact that articulated haulers in general drive several times along the same track, road information recorded on previous runs can be used to improve later performance. The Volvo CE model A35E is equipped with numerous sensors providing vital information that can be used to estimate road properties. By complementing existing sensors with a GPS receiver, road property estimation could be improved. The other main problem examined in this thesis is about using anteriorly recorded road information to make clever gear shift decisions. Some work in this field has already been carried out, and the aim of this thesis work is to make further advances.

1.4 Simulation software

The program package MATLAB & SIMULINK was used for simulation, calculation and analysis in this thesis work. SIMULINK is a powerful tool for both design and simulation of model based dynamic systems. Furthermore it provides a graphical user interface and a design block library. Volvo CE has developed a customized model library for SIMULINK named VSim+, which was also used in this thesis work.

1.5 Report overview

This report consists of twelve chapters. Chapters 1-2 give the reader an introduction to articulated haulers and an overview of the related work done in the area. In chapters 3-6 positioning systems are reviewed along with a brief introduction to the sensors used in this thesis, how they are modeled and how they can be the subjects of sensor fusion. Chapters 8-9 describe how the vehicle and its observer are modeled. Finally, chapter 10 introduces the reader to gearshift strategies, and chapters 11-12 present the results and summarize the work.
Chapter 2

Related work

This chapter discusses a number of different topics closely related to the work in this thesis. The topics mentioned are look-ahead systems in heavy duty trucks, iterative map building techniques and gearshift strategies. Lastly, the most commonly used sensors in these types of applications are listed.

2.1 Look-ahead systems in heavy duty trucks

Numerous articles on this topic can be found and read today. Most of them concern look-ahead systems in heavy duty trucks or other vehicles that normally drive on existing roads. It is also common that they suggest methods including a GPS receiver for obtaining look-ahead information used for speed control, gear shift decisions or other systems. One usual look-ahead method is to determine the vehicle’s position with a GPS receiver, match the position against an existing road and provide the system (speed control system for instance) with information on relevant data (speed limits) of upcoming road segments. Unfortunately, the quality and availability of maps holding such relevant data is today still quite poor. However, this kind of information will perhaps become both better and cheaper in the future. More on this can be found in [14]. Another method for obtaining look-ahead information is to let multiple vehicles collect relevant road data using GPS receivers and other sensors, and then forwarding this data to a dynamic map database through a wireless Internet connection. This technique enables the obtaining of look-ahead information on upcoming road segments from other vehicles that have passed the same track or road before. A similar solution can be read about in [15]. A third method is to collect data from several runs along the same route and merge the data into a better approximation for every run [3].

2.2 Iterative map building

As mentioned in the previous section, iterative map building is a technique exploiting numerous runs along one specific route for enhancing the precision of the map iteratively. It is important to keep in mind that a map is a set of data representing features of some kind at the locations included in the map. The features stored may vary greatly depending on the purpose of the map,
ranging from topology and vegetation to earth fertility and population density. For instance, relevant map data for a look-ahead speed control system could be the speed limit combined with road slope at each mapped location. The role of the map in this thesis work is to provide the gearshift strategy system with useful information, leading to better gearshifts. Iterative map building is a major issue in this thesis work, and a lot of work in the area has been carried out already. In [26], an iterative map building technique for estimating highway road inclination is presented. The technique proposes data fusion from multiple observations in order to increase road inclination estimate accuracy iteratively. A similar technique is used in this thesis work and will be discussed more in the map matching algorithms chapter. Closely related theory is also discussed in [24], where estimated positions and vehicle trajectory characteristics are matched against nearby roads. Here, the locations of nearby roads are considered static and serve as reference values. However, there are big similarities between this road matching theory and, for instance, the data fusion theory presented in [26]. Another theory that also includes map building is the highly reputable **Simultaneous localization and map-building (SLAM)** algorithm. The SLAM problem asks if it is possible for a mobile robot to be placed at an unknown location in an unknown environment and for the robot to incrementally build a consistent map of this environment while simultaneously determining its location within this map. A solution to the SLAM problem has been seen as a "holy grail" for the mobile robotics community as it would provide the means to make a robot truly autonomous. The solution to the SLAM problem has been one of the notable successes of the robotics community over the past 15 years. SLAM has been formulated and solved as a theoretical problem in a number of different forms. SLAM has also been implemented in a number of different domains from indoor robots, to outdoor, underwater, and airborne systems. At a theoretical and conceptual level, SLAM can now be considered a solved problem. However, substantial issues remain in practically realizing more general SLAM solutions and notably in building and using perceptually rich maps as a part of a SLAM algorithm [28]. More about a solution to the SLAM problem can be read in [30]. Furthermore, optimization of the SLAM algorithm can be read about in [29]. The optimized SLAM algorithm is, for instance, suitable for real-time applications.

### 2.3 Gearshift strategies

Automatic gearboxes have become more and more common in heavy trucks. Two reasons for this are improved driver comfort and driving economy. However, in order to achieve better driving economy, the automatic gearbox must be able to make equally good (or better) gearshift decisions as an experienced human driver. Since the gearbox is obviously unable to read the road ahead like the driver does, it has to rely on other systems providing information to base its gearshift decisions on. As long as this information is correct, work carried out on the subject [14] shows that improvement of gearshift strategies can indeed save both time and fuel.
2.4 Commonly used sensors

The most commonly used and important sensor in these systems today is the GPS receiver. It can provide other systems with position information of sufficient accuracy, and it does not suffer from error growth with time like dead reckoning (DR) systems do. However, the GPS signal can be distorted or even lost in tunnels or under roofs. Therefore, systems consisting of a GPS receiver in combination with a DR system are common, and can be considered fairly redundant. A tachometer is a helpful instrument for estimating a vehicle’s speed. It is designed to measure the rotational speed of a shaft or disk. Thus, if the rotational speed of the wheel is known, the translational speed of the vehicle can easily be calculated. Moreover, longitudinal acceleration sensors and compasses are often used for approximating the road slope and the heading, respectively [3].
Chapter 3

Positioning systems

This chapter briefly describes and reviews a few of the most common positioning systems available and used today. The first section covers positioning using dead reckoning systems and odometry. Beacon based positioning systems are then reviewed, followed by a more comprehensive survey of satellite navigation systems.

3.1 Dead reckoning systems and odometry

Methods where the current position is derived from a known starting position together with information about turning directions and travelled distance are commonly referred to as dead reckoning methods. As long as the system has access to this information, it can perform its task independent of other systems or references. However, one drawback is that all dead reckoning systems suffer from error growth as time passes. Algorithms or filters such as the Kalman filter can reduce this error, and are therefore frequently found in dead reckoning systems. There are several ways to implement dead reckoning systems. For instance, course information can be obtained from a simple magnetic compass. It is also possible to use more advanced techniques such as fibre optic gyroscopes or gyrocompasses. Another method of implementation is the inertial navigation system (INS), where accelerometer and gyro data are used to determine position by integration. INS implementations are usually fast and therefore suitable for real time applications, but they also suffer from error growth with time. Hence, GPS systems can be used for persistent calibrations. Systems that model change in position based on wheel speed differences of the vehicle are usually referred to as odometry systems, and a master’s thesis at Scania CV [5] covers this topic in detail.

3.2 Beacon based positioning systems

Beacon based positioning systems are similar to GNSS (which will be covered later), in the sense that they both estimate global position by acquiring relative position to objects with a known global position. These systems offer accurate positioning with no time cohering error growth, but are obviously also dependent on the actual presence of beacons along the track on which the vehicle travels.
3.3 Global Navigation Satellite System (GNSS)

Global navigation satellite system (GNSS) is the umbrella term for systems that use satellites for positioning and navigation. The first system, TRANSIT, was launched in the years 1959-1964 and included seven satellites. A decade later, the GPS system project was initiated, and the work continued for about 20 years before the system was fully developed [6]. The GPS system uses 24 satellites in comparison to its ancestor TRANSIT’s seven [7].

3.3.1 WAAS/EGNOS

WAAS (Wide Area Augmentation System) is a highly accurate positioning system developed in North America for use in civil aviation applications. The system uses a network of ground based reference stations to collect information about small variations in the GPS satellites’ signals. This information is routed to geostationary satellites and from there returned to WAAS compatible GPS receivers on earth. The GPS receivers use the information to correct the position estimate, resulting in improved accuracy. Since the ground based reference stations are only to be found in North America, WAAS compatible GPS receivers do not provide more accurate position estimates in the rest of the world. However, there are similar systems in other regions. One of them is the European equivalent, EGNOS (European Geostationary Navigation Overlay Service) [3].

3.3.2 DGPS

DGPS, or differential GPS, is a system very similar to WAAS/EGNOS, in the sense that they all use information about satellite signal variations to correct the position estimate. WAAS/EGNOS can be considered modern satellite based DGPS systems. The term DGPS was earlier often used to refer to systems where
the satellite signal variation messages were sent over FM radio from ground stations with known positions. The idea is that the ground station (used as a reference) and the GPS receiver suffer from the same signal variations, thus the GPS receiver's position estimate can be corrected by simply subtracting the error since both the true and the measured location of the ground station is known. The use of FM radio broadcasting is particularly suitable in offshore applications because of the increased range of the radio waves [8].

![Image of geostationary reference station and mobile GPS receiver](image)

Figure 3.2: The geostationary reference station and the mobile GPS receiver are considered to suffer from the same position measurement error. This error is calculated in the reference station and broadcasted to mobile receivers by FM radio (as shown in the figure) or via satellite.

### 3.3.3 Ideal positioning

The GPS system’s 24 satellites are organized in six different 12-hour orbital paths spaces, so that at least five satellites are in view from every point on the globe at any given time. The satellites continuously transmit military and civilian navigation data on two L-band frequencies at 1575.42 MHz and 1227.6 MHz, respectively. Five monitor stations and four ground antennas located around the world gather data on each satellite’s exact position. The system then relays this information to the master control station at Schriever Air Force Base in Colorado, which provides overall coordination of the network and transmits correction data to the satellites. Each satellite emits radio signals that a receiver - a miniature device installed on a vehicle or carried by hand - uses to estimate the satellite’s location as well as the distance between the satellite and the receiver. Given one satellite’s location and the distance to it, the possible position of the receiver is limited to be somewhere on the surface of an imaginary sphere with the satellite as origo and a radius equal to the distance between receiver and satellite. By adding the same information from another satellite, the possible position of the receiver is again limited to the surface of a second imaginary sphere, thus the receiver is located somewhere in the intersection of the two spheres - a circle. Intersection of yet another imaginary sphere leaves only two points in the three dimensional space as candidates for the true location of the receiver. In many applications the surface of the earth itself can serve as the fourth sphere and the thereby strictly determine the receiver’s location.
other case a minimum of four satellites is needed. This technique is commonly known as trilateration [10].

Figure 3.3: The two spheres A and B intersect and form a circle in the three dimensional space. A third sphere, C, intersect with the circle and define the only two points located at all three sphere surfaces.

3.3.4 Commercial usage

Before the year 2000 a noise was added to the GPS signal by the US military. The signal noise caused positional errors of 0-70 meters. This was an intentional degradation of the system in order to keep both civil and potential enemy accuracy level of positioning down. However, this noise could easily be compensated for when the DGPS was introduced. As potential enemies could side-step the signal noise and use the GPS system to its full extent anyway, president (now former president) Bill Clinton ordered removal of the added noise, as it only caused trouble for civil users at that point [8]. This was the first step in the GPS modernization program. As time passed, more signals have been added to improve the GPS system. One example is the L2C, which provides civil users with more robust signal reception in places with previously poor reception. The final step in the modernization program is the GPS III program, which plans a complete update of the GPS system [9].
Chapter 4

Available sensors and sensor modeling

This chapter gives a brief presentation of both the performance and the models of the most important sensors used in this thesis work.

4.1 GPS receiver

The GPS receiver used in this project was the Garmin GPS 18 USB. It is a small and compact receiver (61 mm in diameter and 19.5 mm in height) that interfaces to a computer with an available USB port. Drivers are available for use on Windows computers. Macintosh and Linux drivers are not available from Garmin. The product includes an embedded receiver and an antenna, and supports multiple satellite tracking (up to 12 satellites). It also includes the capability of using the WAAS differential GPS system. A few important features are listed below.

- USB interface.
- Tracks and uses multiple satellites for fast, accurate positioning and velocity estimates.
- DGPS capability using real-time WAAS corrections yielding position errors of less than three meters.
- Compact design.
- Non-volatile memory.
- Waterproof design (withstands continuous exposure to water)

Due to these features, the Garmin GPS 18 USB receiver was a good, but still quite cheap, choice for this type of application, where the environment itself often aggravates the use of fragile and cumbersome hardware [12].
Table 4.1: Garmin GPS 18 USB electrical characteristics

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input voltage</td>
<td>4.4-5.5 Vdc</td>
</tr>
<tr>
<td>Input current</td>
<td>110 mA @ 5.0 Vdc</td>
</tr>
<tr>
<td>Operating temperature</td>
<td>-30°C to +80°C</td>
</tr>
<tr>
<td>Storage temperature</td>
<td>-40°C to +90°C</td>
</tr>
</tbody>
</table>

Table 4.2: Garmin GPS 18 USB receiver performance

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acquisition time (hot)</td>
<td>~1 second</td>
</tr>
<tr>
<td>Acquisition time (warm)</td>
<td>~38 seconds</td>
</tr>
<tr>
<td>Acquisition time (cold)</td>
<td>~45 seconds</td>
</tr>
<tr>
<td>Reacquisition time</td>
<td>&lt; 2 seconds</td>
</tr>
<tr>
<td>Update rate</td>
<td>1 Hz</td>
</tr>
<tr>
<td>Positioning accuracy</td>
<td>&lt; 15 meters, 95% typical</td>
</tr>
<tr>
<td>Velocity accuracy</td>
<td>0.1 knot RMS steady state</td>
</tr>
<tr>
<td>WAAS positioning accuracy</td>
<td>&lt; 3 meters, 95% typical</td>
</tr>
<tr>
<td>WAAS velocity accuracy</td>
<td>0.1 knot RMS steady state</td>
</tr>
</tbody>
</table>

The Garmin GPS 18 USB receiver was modeled in Simulink as an ideal sensor disturbed by a noise component. The noise added to the signal is assumed to be Gaussian distributed with a mean of zero. Due to the fact that GPS receiver accuracy improves with a larger number of visible satellites [10], the noise variance was modeled to be dependent on the number of satellites in view in order to achieve this behavior. The GPS receiver signals are given by

\[
\varphi(n) = x_\varphi + u_\varphi(n) \tag{4.1}
\]

\[
\lambda(n) = x_\lambda + u_\lambda(n) \tag{4.2}
\]

\[
z(n) = x_z + u_z(n) \tag{4.3}
\]

\[
v(n) = x_v + u_v(n) \tag{4.4}
\]

where \(x\) is the ideal value, \(\varphi\) the latitude, \(\lambda\) the longitude, \(z\) the altitude, \(v\) the velocity, \(u\) the noise function and \(n\) the number of visible satellites. The reason why there are four different noise functions is that the horizontal accuracy (latitude and longitude) differs from the altitude accuracy. This is also the case with velocity accuracy which differs from both horizontal and altitude accuracy.
4.2 Tilt sensor

The Volvo CE model A35 is equipped with a Kavlico TS905 tilt sensor for measuring the hauler’s longitudinal gradient. The sensor is designed for applications such as: road construction, machine tools, agricultural vehicles, container handling, belt operations and hydraulic lift systems, where tough and high vibration environments are common. Some important features are listed below [11].

- Variable angular range.
- EMI/RFI/ESD protection.
- Temperature compensated.
- Over-voltage & reverse polarity protection.
- Shock & vibration tolerant.

The TS905 has a broad operating temperature range, but with varying response times depending on temperature. However, the response time is assumed to be fast in this thesis work. In conformity with the GPS receiver model, the TS905 was modeled to output the true value (voltage) with a noise component added to it. The true value is estimated with the following equation provided by the manufacturer [3].

\[
V = 2.5 + 0.08 \left( \alpha + \arctan \left( \frac{a \cos \alpha}{g - a \sin \alpha} \right) \right) \frac{180}{\pi} \tag{4.5}
\]

Here, \( V \) is the voltage output, \( \alpha \) the slope, \( a \) the acceleration and \( g \) the gravitational acceleration. The tilt sensor uses an accelerometer to estimate the longitudinal gradient. Due to this fact it is evident that the signal is disturbed not only by electrical noise, but also a larger noise component generated by acceleration. The voltage output (including noise) was therefore modeled to be dependent on acceleration according to

\[
V(\alpha, a) = x_V(\alpha, a) + u_V(a) \tag{4.6}
\]

\[
u_V(a) = ka \tag{4.7}
\]

where \( x_V \) is the ideal value (eq. 4.5) and \( u_V \) the noise function (Remaining arguments are defined as before.). The model was validated by comparing its output to the output from the tilt sensor mounted in the vehicle. During these tests, acceleration was estimated by differentiating the vehicle’s velocity. However, gearshifts can occasionally cause rather extreme acceleration that affects this approach negatively. Therefore a low-pass filter was implemented in the final tilt sensor model design.
Table 4.3: Kavlico TS905 tilt sensor electrical characteristics

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input voltage</td>
<td>$5.0 \pm 0.25$ Vdc</td>
</tr>
<tr>
<td>Output voltage</td>
<td>0.5 - 4.5 Vdc</td>
</tr>
<tr>
<td>Input current</td>
<td>5 mA MAX</td>
</tr>
<tr>
<td>Output impedance</td>
<td>100 $\Omega$</td>
</tr>
<tr>
<td>Operating temperature</td>
<td>-30$^\circ$C to +85$^\circ$C</td>
</tr>
<tr>
<td>Storage temperature</td>
<td>-40$^\circ$C to +100$^\circ$C</td>
</tr>
</tbody>
</table>

Table 4.4: Kavlico TS905 tilt sensor performance

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Angular ranges</td>
<td>-20$^\circ$ to +20$^\circ$ through -60$^\circ$ to +60$^\circ$</td>
</tr>
<tr>
<td>Response time (10 - 90% of span @ +85$^\circ$C)</td>
<td>0.5 seconds</td>
</tr>
<tr>
<td>Response time (10 - 90% of span @ +25$^\circ$C)</td>
<td>1.0 seconds</td>
</tr>
<tr>
<td>Response time (10 - 90% of span @ -30$^\circ$C)</td>
<td>3.0 seconds</td>
</tr>
<tr>
<td>Vibration</td>
<td>10 G's peak sinusoidal (10 - 2000 Hz)</td>
</tr>
</tbody>
</table>

Figure 4.1: Validation of the tilt sensor model. The expected tilt sensor output (data from the actual tilt sensor), is compared to the estimated output from the model (eq. 4.5).
4.3 Articulation sensor

The articulation sensor (a potentiometer mounted in the articulation point) measures the angle between the tractor and the trailer in the horizontal plane. This feature is not included in all Volvo CE articulated haulers, but it is possible to build in, and there are plans for making it standard. High precision angle measurements of this kind are fairly easy to achieve. Hence, the articulation sensor was modeled to act as an ideal sensor, always providing the observer with a noise-free value.

4.4 Tachometer

A tachometer is a sensor measuring angular speed. Volvo CE articulated haulers can have up to four tachometers mounted along the driveline [13]. By measuring the angular speed of the outgoing drive shaft while having information on the current gear ratio (including wheel size), vehicle velocity can be estimated. However, zero wheel slip must obviously be presupposed using this approach.
Chapter 5

The Kalman filter

The Kalman filter was the selected observer model in this thesis since previous work in the same field [3] utilised the filter with fine results. Kalman filtering is also a well documented and powerful algorithm. Extensive research with Kalman filters in autonomous navigation applications as provided satisfying results. The implementation and computational requirements are also suitable for articulated haulers. This chapter gives an introduction to the Kalman filter and its computational origins.

5.1 Brief introduction

In 1960, Rudolf E. Kalman published his famous paper describing a recursive solution to the discrete-data linear filtering problem. Since that time, due in large part to advances in digital programming, the Kalman filter has been subject to extensive research and application, particularly in the area of autonomous or assisted navigation [20]. The Kalman filter is a recursive mathematical method that provides computational means to estimate the state of a process, while minimizing the mean of the squared root error. The filter allows estimations of past, present and future states, even when the precise behavior of the modeled system is unknown. This makes the Kalman filter very powerful. The filter consists of two steps: prediction and correction. In the prediction step, the filter estimates the state of the system based on the system’s dynamic model. Subsequently, the state is in some way measured in order to get a correction of the predicted estimate. This procedure is continuously repeated at each time step, explaining why the Kalman filter is referred to as recursive. Moreover, both the prediction and observation of the process are considered to be suffering from white noise [19].

5.2 Basic components

It is possible to roughly divide the Kalman filter into three basic components: the state vector, the dynamic model and the observation model.
5.2.1 State vector
The state of the dynamic system is represented by the state vector. Its variables cannot be measured directly, but they are potentially inferred from values that are measurable. For instance, a car traveling on a straight line could have its state defined by the position and the velocity. In general the state vector is represented as
\[ \hat{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}. \]

The state vector holds two different sets of values in each time step cycle. The first set consists of a priori (predicted) values and the second set consists of a posteriori (corrected) values after a measurement is made. Henceforth, a priori values and a posteriori values will follow the annotations \( X^- \) and \( X^+ \), respectively.

5.2.2 Dynamic model
The state vector transformation over time is described by the dynamic model, and it is usually represented by a system of differential equations.
\[
\dot{x}(t) = \frac{d}{dt} x(t) = f(x(t), m(t)) \tag{5.1}
\]

In the linear case this can be rewritten as
\[
\dot{x}(t) = F \cdot x(t) + n(t), \tag{5.2}
\]
where \( x(t) \) is the state vector, \( n(t) \) the dynamic noise, which is usually assumed to be white noise, and \( F \) the constant dynamic matrix.

5.2.3 Observation model
The relationship between the state and the measurements is represented by the observation model. Measurements can be described by a system of equations that are linear if the system model is linear (depending on the system variables). The measurements, or observations, are often made at discrete time steps [19].
\[
l(t_i) = h(x(t_i), v(t_i)) \tag{5.3}
\]

In vector form this system becomes
\[
l(t_i) = H \cdot x(t_i) + w(t_i), \tag{5.4}
\]
where \( l(t_i) \) represents the observations of \( x(t_i) \) at time \( t_i \), \( w(t_i) \) is the measurement noise and \( H \) is the observation matrix.
5.3 Discrete Kalman filter algorithm

As already mentioned, the Kalman filter uses a form of feedback control when estimating the process state. It predicts the state at some time and obtains feedback in the form of noisy measurements. The a priori estimates constitute not only estimates of the process state itself, but also an estimate of the state error covariance. Both estimates are thereafter improved in the measurement update phase, becoming a posteriori estimates.

\[ x_k^- = A x_{k-1}^- + B u_{k-1} \]  
\[ P_k^- = A P_{k-1}^- A^T + Q \]

In these equations, \( A \) is the state transition matrix that relates the state at the previous time step \( k-1 \) to the state at the current time step \( k \) in the absence of both control function and process noise. The matrix \( B \) relates the optional control input to the state \( x \), and \( Q \) is a matrix representing the process noise covariance. Corresponding calculations of the a posteriori estimates of process state and error covariance are shown below.

\[ K_k = P_k^- H^T (H P_k^- H^T + R)^{-1} \]  
\[ x_k = x_k^- + K_k (z_k - H x_k^-) \]  
\[ P_k = (I - K_k H) P_k^- \]

Here, the matrix \( H \) relates the process state to the measurement \( z \), and \( R \) is a matrix representing the measurement noise covariance. \( K \) is called the Kalman gain and can be considered to be the blending factor of predicted and measured values. It can be shown that \( K \) minimizes the a posteriori error covariance [19]. More information on this can be found in [16] [17] [18]. (5.7) can be rewritten as

\[ K_k = \frac{P_k^- H^T}{H P_k^- H^T + R} \]
and by inspection it is obvious that

\[
\lim_{R_k \to 0} K_k = H^{-1} \quad (5.11)
\]

\[
\lim_{P_k \to 0} K_k = 0. \quad (5.12)
\]

In other words, as the measurement error covariance approaches zero, the Kalman gain weights the measurements more heavily. Alternatively, as the a priori estimate error covariance becomes smaller, the measurements are weighted less heavily.

### 5.4 Filter parameters and tuning

When implementing the Kalman filter, the measurement noise covariance matrix \( R \) is generally determined prior to operation of the filter. This is possible due to the fact that the process needs to be measured while operating the filter anyway, and by making off-line sample measurements the error covariance could be determined beforehand. The process noise covariance \( Q \) is usually more difficult to determine since possibilities to directly observe the estimated process are rare. However, far from all processes demand a precise determination of \( Q \), due to the fact that a relative poor process model can produce acceptable results as long as enough uncertainty is injected via \( Q \). If a precise determination of \( Q \) is desirable, another Kalman filter can be used for off-line tuning of the first filter’s parameters. Such a process is generally referred to as system identification. Under conditions where \( Q \) and \( R \) are constant, both estimation error covariance \( P_k \) and Kalman gain \( K_k \) will stabilize quickly and remain constant. In such cases the parameters can be pre-computed by running the filter off-line. Unfortunately, the measurement error seldom remains constant, making such a technique useless [19].

### 5.5 Extended Kalman filter (EKF)

The Kalman filter addresses the problem of estimating the state of a discrete-time controlled process that is governed by a linear stochastic difference equation. Numerous processes are, however, non-linear, and sometimes even the measurement relationship to the processes is non-linear. Such situations require adjustments to the standard Kalman filter. One solution is a Kalman filter that linearizes about the current mean and covariance, generally referred to as an extended Kalman filter (EKF). Akin to a Taylor series, estimation is linearized around the current estimate using the partial derivatives of the process and measurement functions. Calculations of a priori process state and error covariance estimates for the EKF are shown below.

\[
x^-_k = f(x_{k-1}, u_{k-1}, 0) \quad (5.13)
\]

\[
P^-_k = A_k P_{k-1} A_k^T + W_k Q_{k-1} W_k^T \quad (5.14)
\]
Here, \( f \) is the non-linear function that relates the state at the previous time step \( k-1 \) to the state at the current time step \( k \). \( A \) is the Jacobian matrix of partial derivatives of \( f \) with respect to \( x \)

\[
A_{i,j} = \frac{\partial f_{[i]}}{\partial x_{[j]}},
\]

(5.15)

and \( W \) is the Jacobian matrix of partial derivatives of \( f \) with respect to the process noise \( w \)

\[
W_{i,j} = \frac{\partial f_{[i]}}{\partial w_{[j]}}.
\]

(5.16)

Moreover, Kalman gain and \textit{a posteriori} values for the EKF are calculated as

\[
K_k = P_k^{-1} H_k^T (H_k P_k^{-1} H_k^T + V_k R_k V_k^T)^{-1}
\]

(5.17)

\[
x_k = x_k^- + K_k (z_k - h(x_k^-, 0))
\]

(5.18)

\[
P_k = (I - K_k H_k) P_k^-.
\]

(5.19)

The non-linear function \( h \) relates the process state to the measurement of it. \( H \) is the Jacobian matrix of partial derivatives of \( h \) with respect to \( x \)

\[
H_{i,j} = \frac{\partial h_{[i]}}{\partial x_{[j]}},
\]

(5.20)

and \( V \) is the Jacobian matrix of partial derivatives of \( h \) with respect to the measurement noise \( v \)

\[
V_{i,j} = \frac{\partial h_{[i]}}{\partial v_{[j]}},
\]

(5.21)
Chapter 6

Map matching algorithms

When the articulated hauler visits an already known section of the track, data from past observations are merged with data from the current observation, in order to increase track data accuracy iteratively. This chapter includes a description of how road data is stored, and a comparison between two algorithms for merging data. Lastly, the reason for the final choice between these two algorithms is given.

6.1 Data storage

As mentioned in the previous chapter, the Kalman filter was selected as the vehicle observer model. The filter’s state vector contains a number of states (more on this in the observer model chapter), including track data of great importance. This track data is stored at given intervals, together with its corresponding covariance matrix (also estimated by the Kalman filter), forming an entity that can be used as a map. The road data is stored as a set of vectors on the form

\[ m_i = \begin{bmatrix} x \\ y \\ z \\ \alpha \\ \varphi \\ C_r \end{bmatrix}, \]

where \( x, y \) and \( z \) represent the Cartesian coordinates in the LTP (Local Tangent Plane) space, \( \alpha \) the inclination, \( \varphi \) the horizontal direction and \( C_r \) the coefficient of rolling resistance. Storage of horizontal direction data is beneficial from two aspects. If it is assumed that the hauler always passes a given track section along the same line (either back or forth), the horizontal direction data can be merged with data from previous runs. The other aspect is that inclination data from runs in one direction can be merged with data from runs in the opposite direction, but one must remember to change the sign of the inclination data. Corresponding adjustment needs to be done with the horizontal direction data as well. Obviously, no adjustments to the P-matrix (covariance matrix) are needed in such situations.
6.2 Data fusion algorithms

The main issue with data fusion algorithms is to extract the data to be merged. In this thesis, data representing the current state estimate is conditionally merged with the most suitable state vector from the set of vectors constituting the map. Thus, if the most suitable state vector stored fulfills certain requirements, it is used for merging. One master’s thesis [24] gives a solution to the problem of deciding whether the requirements are fulfilled or not, and another [3] provides a slight modification. Given the position data $x$, $y$ and $z$ together with the horizontal direction data $\phi$, the following criterions were derived:

\[ \min(||\text{pos}_e - \text{pos}_s||) < D_p, \text{pos}_s \in \Psi \]  
\[ \min(|\phi_e - \phi_s|) < D_d, \phi_s \in \Psi. \]  

Here, $\text{pos}_e$ represents estimated position, $\text{pos}_s$ represents a position stored in the map $\Psi$, and $\phi_e$ and $\phi_s$ represent estimated and stored horizontal directions, respectively. Given the threshold values $D_p$ and $D_d$, data fusion is performed if and only if both criteria hold. In addition, a third criterion was specified during this thesis work. As the hauler approaches a stored position $\text{pos}_s$, no data fusion is performed until the angle formed by the vehicle trajectory and the vector $u = [\text{pos}_s - \text{pos}_e]$ is close to $90^\circ$. 

![Diagram](Image)

Figure 6.1: Third criterion for data fusion. The angle $\beta$ formed by the vehicle trajectory and the vector $u = [\text{pos}_s - \text{pos}_e]$ should be close to $90^\circ$ before data fusion is performed. This prevents data fusion with stored locations that are already passed, or not yet reached.

This was shown to be a necessary modification to the original requirements for data fusion since the map itself only holds information of the spatial domain, and not the temporal domain. Simplified, road data should not be merged with data associated with, for instance, a position already passed. Once the data fusion requirements are fulfilled, numerous algorithms can be used in the actual data fusion process. Two data fusion algorithms were investigated and analyzed in this thesis work; the well documented General data fusion algorithm and the experimental Measurement modification algorithm.
6.2.1 General data fusion algorithm
The algorithm is described in [25] and is especially suitable for Kalman filter applications. Given the current state estimate \( x_e \), a stored state \( x_s \) and their covariance matrices \( P_e \) and \( P_s \), data fusion is performed by

\[
P_f = (P_s^{-1} + P_e^{-1})^{-1}
\]

(6.3)

\[
x_f = P_f (P_s^{-1} x_s + P_e^{-1} x_e).
\]

(6.4)

Unfortunately, the method requires independent measurements. Since the vehicle’s state estimate is based on repeated measurements from the same sensors, measurement independence is not achieved. This will over time lead to an underestimation of the covariance matrix \( P_f \), causing new estimates to have less and less effect [25]. In rapidly changing environments such as quarries, where articulated haulers often operate, the exact opposite behavior might be desirable. A solution to the problem is therefore presented in [26], where new estimates are weighted higher, and old estimates are successively forgotten. By setting the rate at which old estimates are forgotten, a tradeoff situation between the accuracy of the map and the map’s ability to change is introduced. In this thesis work, a-forgetting rate of 0.5 was chosen, meaning that a new estimate is weighted equally highly as the stored state. This was done by assigning \( P_s \) the value of \( P_e \).

6.2.2 Measurement modification algorithm
The algorithm is an experimental method tested in this thesis work. It is based on modification of the Kalman filter inputs. The Kalman filter uses measurements from the GPS-receiver in its second stage of the Kalman cycle. As the GPS-receiver is modeled to have a quite high error covariance, the Kalman filter weights its measurements accordingly. The idea of the Measurement modification algorithm is to use stored position data as fake GPS-receiver measurements with relatively low error covariance instead of the authentic GPS receiver measurements. Thus, the Measurement modification algorithm replaces the GPS receiver measurements with location data from the map. Hence, the algorithm emulates a far better GPS-receiver in situations when data from previous runs is available.

6.3 Algorithm selection
A test case with multiple runs along one track was simulated using both map matching algorithms discussed. The Measurement modification algorithm gave indications of good performance, but also tended to be rather unstable. Analysis of the General data fusion algorithm showed a satisfying performance without any unstable behavior. Due to the forgetting rate of old estimates, positioning accuracy increased with the number of runs along the track, but finally converged to a lower bound. In addition to this, it is well documented in comparison with the experimental Measurement modification algorithm. Hence, the General data fusion algorithm was selected for this thesis work. Details of the simulation results are listed in the simulations and results chapter.
Chapter 7

Vehicle modeling

This chapter describes how the simplified state-space model (which is used in the Kalman filter) of the articulated hauler’s driveline is implemented. The library package VSim+ includes more detailed models of both transmission and engine. Section 7.1 presents how the articulated hauler and a few of its most crucial parts are modeled. Thereafter follows a short description of how external forces such as air resistance and the track itself act on the vehicle (Section 7.2 and section 7.3). The final three sections discuss the driver model, the kinetics of the vehicle and the system modeling.

7.1 Articulated hauler model

The vehicle model described in this thesis embraces the hauler’s powertrain. Powertrain is the umbrella term for all the parts from the engine to the wheels. Hence, the entity referred to as the powertrain is in this thesis defined as the engine and the driveline.

<table>
<thead>
<tr>
<th>Engine</th>
<th>Torque converter</th>
<th>Transmission</th>
<th>Dropbox</th>
<th>Wheel</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_{comb,e}$</td>
<td>$T_{tc}$</td>
<td>$T_{dbx}$</td>
<td>$T_{cs}$</td>
<td>$T_w$</td>
</tr>
<tr>
<td>$T_{fric,c}$</td>
<td>$\omega_e$</td>
<td>$\omega_{tc}$</td>
<td>$\omega_{dbx}$</td>
<td>$\omega_W$</td>
</tr>
</tbody>
</table>

Figure 7.1: Schematic of the articulated hauler’s powertrain.

7.1.1 Engine

A Volvo CE model A35E articulated hauler has a turbocharged diesel engine that has its maximum torque around 1200 rpm [3]. To simplify the model, the engine is considered as a function that generates torque. Thus, internal parts
are neglected in this thesis work. Let $J_e$ and $\dot{\omega}_e$ be the engine mass moment of inertia and crankshaft angular acceleration, respectively, then Newton’s second law of motion gives:

$$J_e \dot{\omega}_e = T_e - T_{tc},$$  \hspace{1cm} (7.1)  

where $T_e$ is the engine torque and $T_{tc}$ is the torque converter’s outgoing torque. It should be mentioned that the engine torque $T_e$, is the engine’s net torque derived from:

$$T_e = T_{comb,e} - T_{fric,e},$$  \hspace{1cm} (7.2)  

where $T_{comb,e}$ is the torque produced by internal combustion and $T_{fric,e}$ is the torque lost due to friction.

### 7.1.2 Torque converter

All Volvo CE articulated haulers are equipped with torque converters. A torque converter is a fluid coupling that transfers rotational power, often from the engine to the driveline of a vehicle. Automobiles with automatic gearboxes are often equipped with torque converters. Moreover, Volvo CE articulated haulers have a lockup system, in order to avoid pumping losses by physically linking the pump and turbine within the torque converter. In this thesis the lockup is assumed to be engaged, which implies modeling of a "stiff" torque converter:

$$T_{tc} = T_t$$  \hspace{1cm} (7.3)  

$$\omega_c = \omega_{tc},$$  \hspace{1cm} (7.4)  

where $T_{tc}$ and $\omega_{tc}$ are the transferred torque and angular speed, respectively.

### 7.1.3 Transmission

Depending on the model, Volvo CE articulated haulers are equipped with either six or nine gears. The simulations made in this thesis assume a gearbox with six gears. The gearbox is modeled as follows:

$$T_t i_t \eta_t = T_{dbx}$$  \hspace{1cm} (7.5)  

$$\omega_{tc} = i_t \omega_t,$$  \hspace{1cm} (7.6)  

where $i_t$ and $\eta_t$ are the gear depending conversion ratio and the representation of gearbox energy losses, respectively.
7.1.4 Dropbox
A dropbox is a mechanical coupling that distributes torque and angular velocities from one input axle to two output axles. In this work, the dropbox is assumed to be ideal (no losses due to friction). Hence, it is modeled with the equations

\[ T_{\text{dbx}} = T_{\text{cs}} \]  
\[ \omega_t = \omega_{\text{dbx}}, \]  

where \( T_{\text{dbx}} \) is the combined dropbox torque in both output axles, \( \omega_{\text{dbx}} \) is the angular velocity, and \( T_{\text{cs}} \) is the cardan shaft torque.

7.1.5 Cardan shaft
A cardan shaft is a mechanical component used to route angular velocity and torque. Normally these shafts are manufactured to be somewhat flexible, but in this work the cardan shaft is modeled to be stiff. This gives the cardan shaft model equations

\[ T_{\text{cs}} = T_{\text{cd}} \]  
\[ \omega_{\text{dbx}} = \omega_{\text{cs}}, \]  

where \( T_{\text{cd}} \) is the torque in the center differential and \( \omega_{\text{cs}} \) the angular velocity of the cardan shaft.

7.1.6 Center differential
The articulated hauler’s center differential distributes power between different axles. In the model used in this thesis work, all three axles are included. The center differential model equations are given by

\[ T_{\text{cd}}i_{\text{cd}} = T_{\text{hub}} \]  
\[ \omega_{\text{cs}} = i_{\text{cd}}\omega_{\text{cd}}, \]  

where \( T_{\text{hub}} \) is the combined torque in all three hubs, \( i_{\text{cd}} \) the center differential conversion ratio and \( \omega_{\text{cd}} \) the angular velocity of the center differential.
7.1.7 Axle hub
The axle hub is the mechanical component on which the wheel is mounted. By letting $T_w$ and $\omega_{hub}$ denote wheel torque and angular velocity of the hub, respectively, the torque of the hub $T_{hub}$ and center differential angular velocity $\omega_{cd}$ are given by

$$T_{hub} = i_{hub} T_w$$

$$\omega_{cd} = i_{hub} \omega_{hub},$$

where $i_{hub}$ is the conversion ratio of the hub. Obviously, hub inertia is neglected in this thesis work.

7.1.8 Wheels
Let $J_w$ be the wheel inertia and $T_{long}$ be the longitudinal mass of inertia and torque induced by the vehicle resisting forces. This gives the wheel model equations

$$J_w \dot{\omega}_w = T_w - k_b B - T_{long}$$

$$\omega_{hub} = \omega_w,$$

where $k_b$ is a constant and $B \in [0, 1]$ is the normalized breaking force. $T_{long}$ and the wheel radius $r_w$ give the resulting longitudinal friction force at the wheel $F_w$ according to

$$F_w = \frac{T_{long}}{r_w}.$$

Moreover, vehicle velocity $v$ and wheel angular velocity relate according to

$$v = r_w \omega_w,$$

assuming the gear is not in neutral. Using the model equations in this section, it can be shown that vehicle velocity $v$ depends on engine angular velocity $\omega_e$ as follows:

$$v = \frac{r_w}{i_{tcd} i_{hub}} \omega_e.$$

In addition to the assumptions already made, it is important to keep in mind that this only holds as long as the wheels do not loose their grip on the ground, causing them to slide.

7.1.9 Chassis
A finished model of an articulated hauler chassis was chosen from the VSim+ toolbox. As no analyzing or research regarding the chassis is part of the problem statement in this thesis work, chassis modeling will not be discussed.
7.2 External forces

The vehicle is modeled to be affected by three external forces: air resistance, rolling resistance and gravitational pull.

Figure 7.2: Schematic of external forces affecting the vehicle. The three most important forces are air resistance force $F_{air}$, rolling resistance force $F_{roll}$ and gravitational pull $F_g$.

7.2.1 Air resistance force

Air resistance force $F_{air}$ is dependent on the front area of the vehicle $A_a$ and the vehicle speed $v$ according to

$$F_{air} = \frac{1}{2} C_w A_a \rho_a v^2,$$  \hspace{1cm} (7.20)

where $\rho_a$ is the air density and $C_w$ the dimensionless drag coefficient. Note that the air resistance force increases by the square of vehicle velocity. Hence, this force should not be neglected at higher velocities, especially not when the front area $A_a$ is large, as is the case with articulated haulers.

7.2.2 Rolling resistance force

Rolling resistance force $F_{roll}$ is caused by the friction between the wheels and the road. It is modeled as

$$F_{roll} = C_r F_N = C_r m g \cos \alpha,$$ \hspace{1cm} (7.21)

where $C_r$ is the coefficient of rolling resistance, $F_N$ the normal force, $\alpha$ the road slope, $g$ the gravitational acceleration and $m$ the mass of the vehicle.

7.2.3 Gravitational pull

In heavy vehicles like Volvo CE articulated haulers, gravitational pull will certainly be the largest external force affecting the vehicle. The gravitational pull $F_g$ is a function of road slope $\alpha$ and is calculated by

$$F_g = m g \sin \alpha.$$ \hspace{1cm} (7.22)

Depending on the model, a fully loaded Volvo CE articulated hauler can weigh up to 60 metric tons. Thus, the gravitational pull $F_g$ becomes very large as road slope increases.
CHAPTER 7. VEHICLE MODELING

7.3 Track model

It is troublesome trying to model a realistic track for the purpose of this thesis work, since articulated haulers often drive in off-road terrain away from standard roads. However, data was collected from authentic runs on a real off-road track and was then processed, in order to extract road curvature, slope and other important features.

7.4 Driver

The driver is modeled as a simple PI-controller, originating from existing models developed at Volvo. Development of the driver model has not been part of this thesis work.

7.5 Vehicle motion

Using the wheel model combined with the models of external forces and Newton’s second law of motion, it is possible to derive an expression for the vehicle’s translational kinetics:

\[ m_t \frac{dv}{dt} = F_w - F_{air} - F_g - F_{roll}. \]  

(7.23)

where \( m_t \) is the vehicle’s total mass, including inertial mass [3].
Chapter 8

Observer modeling

This chapter gives a brief presentation of the system model used in this thesis work. There is also a short description of the Kalman filter that serves as the vehicle state observer model.

8.1 System model

As mentioned in the map matching algorithms chapter, a number of quantities, such as vehicle position, pitch, horizontal direction and track rolling resistance are found in the Kalman filter’s state vector. In addition to this, vehicle velocity is also included. A time dependent state vector is then defined as

\[
\hat{x}(t) = \begin{bmatrix} x(t) \\ y(t) \\ z(t) \\ v(t) \\ \alpha(t) \\ \varphi(t) \\ C_r(t) \end{bmatrix},
\]

where \(x\), \(y\) and \(z\) represent the vehicle’s position in the LTP coordinate system, \(v\) the velocity, \(\alpha\) the vehicle’s pitch, \(\varphi\) the vehicle’s heading in the horizontal plane and \(C_r\) the coefficient of rolling resistance.

8.1.1 Spatial sampling

However, a time dependent state vector is of no use when trying to merge road data from several different runs. The model needs to be spatially sampled, in order to obtain estimates at specific spatial locations more easily [3].
8.1.2 Spatially sampled system model

Distance dependent state and measurement vectors can be defined as

\[
\begin{bmatrix}
\hat{x}_k \\
\hat{y}_k \\
\hat{z}_k \\
\hat{v}_k \\
\hat{\alpha}_k \\
\hat{\phi}_k \\
C_{r,k}
\end{bmatrix} =
\begin{bmatrix}
x_{GPS,k} \\
y_{GPS,k} \\
z_{GPS,k} \\
v_{Tach,k} \\
\alpha_{Tilt,k} \\
\phi_{GPS,k} \\
C_{r,k}
\end{bmatrix},
\]

where \( GPS, Tach \) and \( Tilt \) denote that the measurements come from the GPS receiver, the tachometer and the tilt sensor, respectively. By using Euler’s integration method, the state transition equations are given by

\[
\begin{align*}
\hat{x}_k &= x_{k-1} + \Delta s \cos \alpha_{k-1} \cos \phi_{k-1} \\
\hat{y}_k &= y_{k-1} + \Delta s \cos \alpha_{k-1} \sin \phi_{k-1} \\
\hat{z}_k &= z_{k-1} + \Delta s \sin \alpha_{k-1} \\
\hat{v}_k &= v_{k-1} + \frac{\Delta s}{v_{k-1}} \left( F_w - F_{air} - F_{roll} - F_g \right) \\
\hat{\alpha}_k &= \alpha_{k-1} \\
\hat{\phi}_k &= \phi_{k-1} + \Delta s \left( \frac{\cos \alpha_{k-1}}{l^2 \sin \Phi + (l_1 + l_2 \cos \Phi) \left( \frac{\cos \Phi}{\sin \Phi} \right)} \right) \\
C_{r,k} &= C_{r,k-1},
\end{align*}
\]

where \( m_t, \Delta s \) and \( \Phi \) denote total mass (including inertial mass), step length and the vehicle’s articulation angle respectively. To clarify the math behind the state transition equations it should be mentioned that \( x, y, z \) are updated by presupposing linear translation. Velocity \( v \) is updated by adding the current acceleration times elapsed time to the old value. Equations 7.1-7.23 show the computational origins of \( F_w, F_{air}, F_{roll}, F_g \) and \( m_t \), that are used in (8.4). The road inclination \( \alpha \) is, in line with the coefficient of rolling resistance \( C_r \), inheriting its previous value as there is no easy way of predicting a posterior value. Horizontal direction \( \phi \) is updated according to the radius of the vehicle’s circular trajectory shown in appendix A.

8.2 Extended Kalman filter implementation

Since the vehicle model is non-linear, a standard Kalman filter observer model is not suitable. Hence, the extended Kalman filter is used in this thesis work.
By inspection it is obvious that the extended Kalman filter’s state equation can be described as

\[
\begin{bmatrix}
    x_k \\
    y_k \\
    z_k \\
    v_k \\
    \alpha_k \\
    \varphi_k \\
    C_{r,k}
\end{bmatrix}
= \begin{bmatrix}
    x_{k-1} + \Delta s \cos \alpha_{k-1} \cos \varphi_{k-1} \\
    y_{k-1} + \Delta s \cos \alpha_{k-1} \sin \varphi_{k-1} \\
    z_{k-1} + \Delta s \sin \alpha_{k-1} \\
    v_{k-1} + \frac{\Delta s}{v_{k-1}} \left( F_{w} - F_{air} - F_{roll} - F_g \right) \\
    \alpha_{k-1} \\
    \varphi_{k-1} + \Delta s \left( \frac{\cos \alpha_{k-1}}{C_{r,k-1}} \right) \\
    C_{r,k-1}
\end{bmatrix}
+ \begin{bmatrix}
    u^x_k \\
    u^y_k \\
    u^z_k \\
    u^v_k \\
    u^\alpha_k \\
    u^\varphi_k \\
    u^C_k
\end{bmatrix}.
\]

The two variables \( l_1 \) and \( l_2 \) represent the distances from the articulation point to the front wheel axle and the point in between the two rear wheel axles, respectively. Consequently, the Jacobian matrix of partial derivatives becomes

\[
A = \begin{bmatrix}
    1 & 0 & 0 & 0 & \frac{\partial f_1}{\partial \alpha_{k-1}} & \frac{\partial f_2}{\partial \varphi_{k-1}} & 0 \\
    0 & 1 & 0 & 0 & \frac{\partial f_1}{\partial \alpha_{k-1}} & \frac{\partial f_2}{\partial \varphi_{k-1}} & 0 \\
    0 & 0 & 1 & 0 & \frac{\partial f_2}{\partial \alpha_{k-1}} & \frac{\partial f_2}{\partial \varphi_{k-1}} & 0 \\
    0 & 0 & 0 & 1 & \frac{\partial f_3}{\partial C_{r,k-1}} & 0 & 0 \\
    0 & 0 & 0 & 0 & \frac{\partial f_3}{\partial C_{r,k-1}} & 1 & 0 \\
    0 & 0 & 0 & 0 & 0 & 1 & 0 \\
\end{bmatrix},
\]

where

\[
\frac{\partial f_1}{\partial \alpha_{k-1}} = -\Delta s \sin \alpha_{k-1} \cos \varphi_{k-1} \quad (8.8)
\]

\[
\frac{\partial f_1}{\partial \varphi_{k-1}} = -\Delta s \cos \alpha_{k-1} \sin \varphi_{k-1} \quad (8.9)
\]

\[
\frac{\partial f_2}{\partial \alpha_{k-1}} = -\Delta s \sin \alpha_{k-1} \sin \varphi_{k-1} \quad (8.10)
\]

\[
\frac{\partial f_2}{\partial \varphi_{k-1}} = \Delta s \cos \alpha_{k-1} \cos \varphi_{k-1} \quad (8.11)
\]

\[
\frac{\partial f_3}{\partial \alpha_{k-1}} = \Delta s \cos \alpha_{k-1} \quad (8.12)
\]

\[
\frac{\partial f_3}{\partial \varphi_{k-1}} = 1 - \frac{\Delta s}{v^2_{k-1}} \left( F_{w} - F_{air} - F_{roll} - F_g \right) \quad (8.13)
\]

\[
\frac{\partial f_4}{\partial \alpha_{k-1}} = \frac{mg \Delta s}{v_{k-1} m_t} \left( C_{r,k-1} \sin \alpha_{k-1} - \cos \alpha_{k-1} \right) \quad (8.14)
\]

\[
\frac{\partial f_4}{\partial C_{r,k-1}} = -\frac{mg \Delta s \cos \alpha_{k-1}}{v_{k-1} m_t} \quad (8.15)
\]

\[
\frac{\partial f_5}{\partial \alpha_{k-1}} = \frac{\Delta s}{v_{k-1}} \left( \frac{v_{k-1} \sin \alpha_{k-1}}{l_2 \sin \Phi + (l_1 + l_2 \cos \Phi) \left( \frac{\sin \Phi}{\sin \Phi} \right)} \right) \quad (8.16)
\]
Here, $m$ represents the vehicle’s total mass, and $g$ the gravitational acceleration. The Kalman filter’s measurement equation can be described by

$$ y_k = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_k \\ y_k \\ z_k \\ v_k \\ \alpha_k \\ \phi_k \end{bmatrix} + \begin{bmatrix} z_k^x \\ z_k^y \\ z_k^z \\ z_k^\alpha \\ z_k^\phi \end{bmatrix}. $$

One issue regarding the measurement update step in the Kalman filter is that different sensors have different sample times. For example, the tachometer provides the system with new measurements at a rate of 10 Hz, while the GPS-receiver only provides new measurements at a rate of 1 Hz. This problem was solved by letting the "slower" sensors hold their old values until new values arrive. However, older values are weighted less heavily in the sensor fusion process. Moreover, it should be mentioned that the error covariance matrix $P$ was initialized with small values along its diagonal.
Chapter 9

Gearshift strategies

The purpose of the map building part in this thesis is to enable prediction of track property data ahead of the vehicle. This chapter discusses one approach to such property prediction. Furthermore, two algorithms for acting based on track property information ahead are reviewed.

9.1 Existing gearshift strategy

The current gearshift strategy used in Volvo CE articulated haulers [31] aims at determining favorable gearshift points under particular circumstances. In most combustion engines (the diesel engine of the Volvo CE articulated hauler included), the maximum possible torque delivered is highly dependent on engine speed. The engine is relatively weak at lower speeds, has its best performance around 1200 RPM, and starts to get weaker again as the engine speed increases even more. Roughly speaking, the engine speed should not be too low or too high, as these situations affects fuel efficiency and performance negatively\(^1\). Another issue is the lockup system. If the engine speed is too low, the lockup system disengages and causes a major drop in fuel efficiency. Lots of work has already been carried out in the automatic gearshift strategy area. For instance, [14] proposes a solution for highway applications, while [4] reviews corresponding techniques in off-road situations. The solution in this thesis work is influenced by both.

9.2 First approach - Workload prediction

Two important properties of a track are its gradient and topological features. Based on information about these properties, combined with the vehicle’s total mass, a prediction of the workload that lies ahead can be made.

9.2.1 Workload prediction

The dynamics of the articulated hauler suggest that track information too far ahead (for instance, 1 km) is of no interest. Thus, a prediction horizon was

\(^1\) Gianantonio Bortolin
set to 30 m. In other words, the prediction algorithm calculates the change in energy of position over the next 30 m of the track.

Figure 9.1: The workload is defined as the change in energy of position when driving 30 m forward starting at the current position.

Figure 9.2: An example of the predicted work as a function of time when traveling with full load on a hilly track.

9.2.2 Strategy I - ”Simple”

As a first approach, the gearshift points calculated by the existing gearshift strategy were modified depending on the predicted workload ahead of the vehicle. This modification was achieved by adding an offset to the original gearshift points following the crude rule set

\[ W_{\text{pred}} > W_{ub} \Rightarrow O_{su} = 0, O_{sd} = 100 \]  
\[ W_{\text{pred}} < W_{lb} \Rightarrow O_{su} = -50, O_{sd} = 0. \]  

Here, \( W_{\text{pred}} \) represents the predicted work, and \( W_{ub} \) and \( W_{lb} \) the upper and lower work bounds for predicted work, respectively. \( O_{su} \) denotes the offset added to the shift up points, and \( O_{sd} \) denotes the offset added to the shift down points. In this thesis work, \( W_{ub} = 5 \cdot 10^5 \text{ J} \) and \( W_{lb} = -5 \cdot 10^5 \text{ J} \).

9.3 Further development

In certain situations it might be the case that the vehicle can climb steep hills relatively easy. For instance, if the vehicle travels at high speed, it also possesses a lot of kinetic energy (due to its weight). In fact, the kinetic energy possessed by the vehicle increases quadratically with increasing speed. Thus, the gearshift
point offsets were set to be a function of predicted work and vehicle velocity rather than predicted work alone. However, different velocities and workloads can be combined in an infinite number of ways, forcing the combining domain to be narrowed. One well documented method for solving such problems is fuzzy logic.

9.3.1 Fuzzy logic

Fuzzy logic is a computer science approach that aims to mimic the way a human brain solves problems, thinks and reasons. The idea is to use natural language terms instead of quantitative terms in the reasoning process. It is formally defined as a form of knowledge representation, suitable for notions that cannot be defined precisely, but which depend upon their context. Since it enables computerized devices to reason and solve problems more like humans, it comes in handy when conventional logic fails. Fuzzy logic uses linguistic variables, and because of this, it can deal with almost any proposition expressed in natural language. Linguistic variables are variables whose values can be expressed or represented in natural language words or sentences. In general, relations between multiple linguistic variables can be expressed on terms of fuzzy if-then rules. Fuzzy logic can therefore provide approximate problem solving and reasoning, once the meanings of relevant propositions are determined [27].

Example

This is a simple but illuminating example of temperature control with air conditioning inside a house using fuzzy logic. Consider the fuzzy rule set

\[
\begin{align*}
R1: & \quad \text{IF temperature IS hot THEN set AC to max} \\
R2: & \quad \text{IF temperature IS warm THEN set AC to medium} \\
R3: & \quad \text{IF temperature IS cold THEN turn off AC}
\end{align*}
\]

which make perfect sense for humans. However, the problem is to determine whether the linguistic variable temperature is hot, warm or cold. In fuzzy logic, the variable temperature will always be hot, warm and cold at the same time, but to different extents. Membership functions are often used when determining to what extent a linguistic variable belongs to a certain value. This value will in the example above provide the corresponding rule with a certain “credibility”, giving a hint on what action to take.
Figure 9.3: A membership function for indoor temperature. The function tells to what extent a particular temperature is *hot*, *warm* or *cold*.

**Logical operators**

If a rule is of the form

\[
RX: \text{IF } S_1 \text{ IS } P_1 \text{ AND } S_2 \text{ IS } P_2 \text{ THEN } A_1,
\]

it must be determined to what extent both \( S_1 \) is \( P_1 \) and \( S_2 \) is \( P_2 \). In fuzzy logic, the logical AND operator is defined as

\[
\min(\text{member}(S_1, P_1), \text{member}(S_2, P_2)),
\]

where \( \text{member} \) is the membership function, and \( \min \) is the minimum value function. Similarly, the logical OR operator is defined as

\[
\max(\text{member}(S_1, P_1), \text{member}(S_2, P_2)).
\]
9.3.2 Strategy II - "Fuzzy"

Velocity and predicted work were combined in the fuzzy rule set

R1: IF $W_{\text{pred}}$ IS $h$ AND vel IS ($l$ OR $m$) THEN $O_{su} = 0$, $O_{sd} = 95$
R2: IF $W_{\text{pred}}$ IS $h$ AND vel IS $h$ THEN $O_{su} = 0$, $O_{sd} = 25$
R3: IF $W_{\text{pred}}$ IS $m$ AND vel IS $l$ THEN $O_{su} = 0$, $O_{sd} = 25$
R4: IF $W_{\text{pred}}$ IS $m$ AND vel IS ($m$ OR $h$) THEN $O_{su} = 0$, $O_{sd} = 0$
R5: IF $W_{\text{pred}}$ IS $l$ AND vel IS $h$ THEN $O_{su} = -25$, $O_{sd} = 0$
R6: IF $W_{\text{pred}}$ IS $l$ AND vel IS ($l$ OR $m$) THEN $O_{su} = -15$, $O_{sd} = 0$

where $l$, $m$ and $h$ represent the linguistic values low, medium and high, respectively. Moreover, the triangular membership function was used and analyzed with several different base points and peak locations.

9.4 Algorithm selection

All three gearshift strategies (including the existing gearshift strategy) were tested and reviewed with interesting results.
Chapter 10

Simulation and results

This chapter lists, analyzes and compares simulation results obtained from the Simulink model of the vehicle.

10.1 Positioning and map matching

The first part of this master’s thesis includes investigation of the possibilities with iterative map building. This section presents simulation results from four different test cases. Positioning, track slope estimation and coefficient of rolling resistance estimation are the entities investigated more closely.

10.1.1 Five runs with a full load on a level track

In the first test case, five runs around a level oval-shaped track were simulated with a continuously fully loaded trailer.

Position

All three position estimates follow the true values (obtained from DGPS) very closely. The mean spatial error is 1.73 m in the first run, but decreases to a final error of 1.44 m after five iterations. There are, however, outliers among the position estimates which can be considered maximum errors. The outliers cause worst case estimations with errors of ~3 m.
Figure 10.1: Vehicle position as a function of time when simulating with a full load on a level track.

**Slope**

Track slope estimation shows very satisfying results, with a mean error of less than 0.3° after five iterations. The estimate fluctuations during the first 15 s are caused by relatively high acceleration and several gearshifts (which affect the tilt sensor).
CHAPTER 10. SIMULATION AND RESULTS

Table 10.1: Improvement of positioning accuracy due to iterative map building when simulating with a full load on a level track.

<table>
<thead>
<tr>
<th>Iteration</th>
<th>Mean spatial error</th>
<th>Relative improvement from Run I</th>
</tr>
</thead>
<tbody>
<tr>
<td>Run I</td>
<td>1.73 m</td>
<td>N/A</td>
</tr>
<tr>
<td>Run II</td>
<td>1.65 m</td>
<td>4.40 %</td>
</tr>
<tr>
<td>Run III</td>
<td>1.42 m</td>
<td>18.0 %</td>
</tr>
<tr>
<td>Run IV</td>
<td>1.44 m</td>
<td>16.5 %</td>
</tr>
<tr>
<td>Run V</td>
<td>1.43 m</td>
<td>17.0 %</td>
</tr>
</tbody>
</table>

Figure 10.2: Track slope as a function of time when simulating with a full load on a level track.

Rolling resistance coefficient

The coefficient of rolling resistance estimation converges to \(~0.04\), which is also the true value. Frequent gearshifts during the first 15 s disturb the estimation and cause greater uncertainty.

Figure 10.3: Coefficient of rolling resistance as a function of time when simulating with a full load on a level track.
10.1.2 Five runs with zero load on a level track

As a second test case, five runs along the same oval-shaped track were simulated, but with a continuously empty trailer.

Position

All three position estimates are within a reasonable error margin. The results are worse than in the case with a fully loaded trailer, but still quite good. The mean spatial error was 4.62 m in the first run, and after five iterations it shrunk to 3.43 m. One interesting finding among the results is that the outliers cause maximum error estimates of ~4.7 m. Thus, not much more than the mean error from the first run.
Slope

The slope estimation converges to a level as good as in the test case with a fully loaded trailer. A smaller total mass however enables higher acceleration, causing the estimate fluctuations during the first 15 s to be greater.

Figure 10.4: Vehicle position as a function of time when simulating with zero load on a level track.
Table 10.2: Improvement of positioning accuracy due to iterative map building when simulating with zero load on a level track.

<table>
<thead>
<tr>
<th>Iteration</th>
<th>Mean spatial error</th>
<th>Relative improvement from Run I</th>
</tr>
</thead>
<tbody>
<tr>
<td>Run I</td>
<td>4.62 m</td>
<td>N/A</td>
</tr>
<tr>
<td>Run II</td>
<td>4.33 m</td>
<td>6.38 %</td>
</tr>
<tr>
<td>Run III</td>
<td>3.93 m</td>
<td>15.0 %</td>
</tr>
<tr>
<td>Run IV</td>
<td>3.60 m</td>
<td>22.2 %</td>
</tr>
<tr>
<td>Run V</td>
<td>3.43 m</td>
<td>25.8 %</td>
</tr>
</tbody>
</table>

Figure 10.5: Track slope as a function of time when simulating with zero load on a level track.

**Rolling resistance coefficient**

Simulating with an empty trailer seems to have little affect on the coefficient of rolling resistance estimation compared to when the trailer was fully loaded.

Figure 10.6: Coefficient of rolling resistance as a function of time when simulating with zero load on a level track.

### 10.1.3 Five runs with full load on a hilly track

In order to get a fairer view of the Kalman filter’s and map matching algorithm’s performance, five runs were also simulated on a hilly track with two steep slopes.
The trailer was fully loaded during all five runs in the first test.

**Position**

Again, all three position estimates follow their corresponding reference value very well. The results are, however, somewhat worse than when simulating on a level track. A mean spatial error of $2.53\,\text{m}$ was registered from the first iteration, and after five iterations it shrank to a final value of $2.13\,\text{m}$. In conformity with the first test case, the maximum error was $\sim 3\,\text{m}$.

![Graphs of vehicle position as a function of time](image)

Figure 10.7: Vehicle position as a function of time when simulating with a full load on a hilly track.
Table 10.3: Improvement of positioning accuracy due to iterative map building when simulating with a full load on a hilly track.

<table>
<thead>
<tr>
<th>Iteration</th>
<th>Mean spatial error</th>
<th>Relative improvement from Run I</th>
</tr>
</thead>
<tbody>
<tr>
<td>Run I</td>
<td>2.53 m</td>
<td>N/A</td>
</tr>
<tr>
<td>Run II</td>
<td>2.55 m</td>
<td>-0.91 %</td>
</tr>
<tr>
<td>Run III</td>
<td>2.22 m</td>
<td>12.3 %</td>
</tr>
<tr>
<td>Run IV</td>
<td>2.06 m</td>
<td>18.4 %</td>
</tr>
<tr>
<td>Run V</td>
<td>2.13 m</td>
<td>15.7 %</td>
</tr>
</tbody>
</table>

**Slope**

Even on a hilly track, with more gearshifts and acceleration changes, the slope estimation showed very good results. The mean error was a little larger (=0.6°) compared to the results from the level track simulations. A maximum error of 5° was registered.

![Track slope as a function of time when simulating with a full load on a hilly track.](image)

**Rolling resistance coefficient**

The coefficient of rolling resistance estimation is a bit noisy during the hill climb, but a trend value close to the reference value of 0.04 can be extracted.
10.1.4 Five runs with zero load on a hilly track

As a final test, five runs were simulated on the same hilly track, but with a continuously empty trailer.

Position

Once again the position estimates are very close to their reference values. Compared to the results from the simulations with a fully loaded trailer, the estimations are a little worse. The first run resulted in a mean spatial error of 3.48 m, and the fifth run in a mean spatial error of 2.80 m. The maximum error was ~5 m.
Figure 10.10: Vehicle position as a function of time when simulating with zero load on a hilly track.

Slope

The results from track slope estimation were, like positioning estimate results, somewhat worse than in the case with a fully loaded trailer. While the maximum errors were approximately equal, the mean error from this test was calculated to be $\sim 0.8^\circ$. 
Table 10.4: Improvement of positioning accuracy due to iterative map building when simulating with zero load on a hilly track.

<table>
<thead>
<tr>
<th>Iteration</th>
<th>Mean spatial error</th>
<th>Relative improvement from Run I</th>
</tr>
</thead>
<tbody>
<tr>
<td>Run I</td>
<td>3.48 m</td>
<td>N/A</td>
</tr>
<tr>
<td>Run II</td>
<td>3.15 m</td>
<td>9.64 %</td>
</tr>
<tr>
<td>Run III</td>
<td>3.27 m</td>
<td>6.17 %</td>
</tr>
<tr>
<td>Run IV</td>
<td>2.91 m</td>
<td>16.4 %</td>
</tr>
<tr>
<td>Run V</td>
<td>2.80 m</td>
<td>19.7 %</td>
</tr>
</tbody>
</table>

Figure 10.11: Track slope as a function of time when simulating with zero load on a hilly track.

**Rolling resistance coefficient**

In line with the test with a fully loaded trailer, the coefficient of rolling resistance estimation is noisy. However, the trend value is very close to the reference value of 0.04.

Figure 10.12: Coefficient of rolling resistance as a function of time when simulating with zero load on a hilly track.
10.2 Gearshift strategies

The gearshift strategies compared and analyzed in this thesis are Volvo CE’s existing gearshift strategy for articulated haulers, the workload predictive “Simple” strategy and the workload/velocity-combining “Fuzzy” strategy. To test the concepts of the different gearshift strategies, the hilly track section from the map matching tests was used for gearshift strategy analyzing. This track section consists of a level straight, a steep climb, and then a downslope. Since the gearshift strategies analyzed are based on workload prediction, a hilly track will help in pointing out the concept each gearshift strategy is supposed to offer. Obviously, the consequences of applying a gearshift strategy will be exaggerated using this approach. This is due to the fact that a more level track will not provoke gearshift strategy actions to the same extent.

![Figure 10.13: LTP z-coordinate as a function of time when simulating a run on the track section used for gearshift strategy analyzing.](image-url)
10.2.1 Existing gearshift strategy

As the articulated hauler starts climbing the hill, gravitational pull forces its automatic gearbox to shift down. The 1st gear is engaged after $\sim 25$ s. However, at this point the engine speed has already reached critical levels of $\sim 600$ RPM, causing a massive torque drop and a lockup disengage. The hauler continues its course and makes it over the hill in 116.4 s, with a total fuel consumption of 1.358 kg.

![Figure 10.14: Test result using existing gearshift strategy.](image-url)
10.2.2 Strategy I - "Simple"

By applying the "Simple" gearshift strategy, the situation where the engine speed dropped below desired levels can be avoided. The predicted work was well over the threshold value even before the hauler started the hill climb, resulting in a faster gearshift sequence (4th to 1st). In addition to avoiding low RPM, a lockup disengage was not needed. Furthermore, the shift up sequence was faster in the downhill section of the track, due to a prediction of a negative workload. This resulted in RPM levels a bit lower than with the existing gearshift strategy. By using the "Simple" gearshift strategy, the entire track section was covered in 114.2 s, with a total fuel consumption of 1.348 kg.

![Graph of selected gear and engine speed](image.png)

Figure 10.15: Test result using the "Simple" gearshift strategy.
10.2.3 Strategy II - "Fuzzy"

The "Fuzzy" gearshift strategy includes vehicle velocity in the derivation of $O_{sd}$ and $O_{su}$. As the articulated hauler starts climbing the hill, the predicted workload rises and the vehicle velocity drops. This results in an even faster gearshift sequence down to the 1st gear, and an engine speed drop of not more than 300 RPM. From this level the engine provides maximum torque. Similar to the "Simple" strategy, the faster shift up sequence in the downhill section helps avoid the highest engine speeds. By applying the "Fuzzy" gearshift strategy, the track section was covered in 113.9 s, with a total fuel consumption of 1.347 kg.

Figure 10.16: Test result using the "Fuzzy" gearshift strategy.
Tuning

The "Fuzzy" gearshift strategy can be altered in many ways. One way, for instance, is to change the fuzzy rule set. In this thesis work the membership functions were tuned. Tuning was achieved by altering the base points and peak locations of the different membership functions. The base points and peak locations determine how memberships are estimated since they uniquely quantify their corresponding membership function. Hence, even if the rule set is left untouched, altered membership functions can provoke a new, more desirable, behavior of the algorithm. One revision of the membership functions showed very interesting results, where the 1st gear could be completely avoided while climbing the hill. Gearshifts in articulated haulers take up to 1.5 s, and, by tuning the "Fuzzy" gearshift strategy, two gearshifts less were needed during the track section. This behavior saved even more time and fuel, since the entire track section was covered in 111.8 s with a total fuel consumption of 1.337 kg.

Figure 10.17: Test result using a tuned revision of the "Fuzzy" gearshift strategy.
10.3 Summary

The map matching algorithm and its implementation showed good results. Map accuracy reached levels sufficiently high for use as gearshift strategy input. An articulated hauler is a rather large vehicle often moving in rough environments, and the map being built is not intended to serve as navigational assistance of any kind. Thus, the map accuracy requirements are not very high. This simulation study shows that even a system utilizing a relatively cheap GPS receiver with quite bad performance can provide a gearshift strategy algorithm with useful data. Mapping performance is also better when simulating with a fully loaded trailer. This behavior most certainly comes from the fact that a smaller mass enables higher acceleration, causing higher noise in the tilt sensor. The gearshift strategy results imply that significant improvements can be made by providing the gearshift algorithm with data of upcoming track sections.

<table>
<thead>
<tr>
<th>Strategy</th>
<th>Gearshifts</th>
<th>Lockup disengages</th>
<th>Fuel used</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Existing</td>
<td>12</td>
<td>1</td>
<td>1.358 kg</td>
<td>116.4 s</td>
</tr>
<tr>
<td>”Simple”</td>
<td>12</td>
<td>0</td>
<td>1.348 kg</td>
<td>114.2 s</td>
</tr>
<tr>
<td>”Fuzzy”</td>
<td>12</td>
<td>0</td>
<td>1.347 kg</td>
<td>113.9 s</td>
</tr>
<tr>
<td>”Fuzzy tuned”</td>
<td>10</td>
<td>0</td>
<td>1.337 kg</td>
<td>111.8 s</td>
</tr>
</tbody>
</table>
Chapter 11

Discussion

This chapter summarizes the simulation results and presents the conclusions of the work. Also, some propositions for further research in this specific field are mentioned.

11.1 Conclusions

The first main part of this thesis work concerns iterative map building in completely unknown environments. Simulation results indicate significant improvement of positioning accuracy (15-25%), even after only five runs along a track. However, positioning accuracy does not improve at the same rate after five iterations. One plausible explanation for this is the tradeoff situation (between positioning accuracy and the map’s ability to change) introduced due to the forgetting rate of old map data. In line with the positioning results, estimation of slope and coefficient of rolling resistance proved to be more than good enough for their purpose. If higher positioning accuracy is desired, experiments made within the simulator imply that the cheapest and simplest way of improving the performance is to use a more accurate GPS receiver. The second main part of this thesis work is a study on whether previously gathered map data can serve as useful input for gearshift decision algorithms in automatic gearboxes. All three gearshift strategies investigated tended to improve the hauler’s performance from both a time and fuel saving aspect. The results from the test track show that over 4% of time and 1-2% of fuel can be saved by using a smarter gearshift strategy. Other benefits are that lockup disengages and frequent gearshifts can be avoided, causing less wear to the machinery.

11.2 Further work

This thesis work is a simulation study only. Thus, the first obvious extension of the work would be to implement the system in a real articulated hauler and review its behavior. Furthermore, the GPS receiver model developed has built-in support for taking into account the current number of satellites in view. In general, GPS receiver accuracy increases with the number of satellites and vice versa. This means that numerous interesting case studies could be made, where, for instance, zero satellites are in view. Such a situation might occur
when the hauler travels through a long tunnel, or inside a deep quarry or mine. One study could be to investigate to what extent overall positioning is affected when the GPS signal is lost, and whether there are any counteraction alternatives. Moreover, the gearshift strategies reviewed in this thesis work show great performance. This, however, is a huge field, and this thesis only scratches the surface of what there is to discover. In order to develop a gearshift strategy that fully imitates the behavior of an experienced driver, a breakthrough is probably needed in the area of artificial intelligence. Until then, studies using common existing technologies, such as artificial neural networks, could be made. However, it should be taken into consideration that advanced artificial intelligence may require large amounts of computational power, which articulated haulers lack. Therefore, a solution utilizing reinforcement learning is probably suitable\textsuperscript{1}. Moreover, an extension of the "Fuzzy" gearshift strategy could be to take the vehicle’s kinetic energy rather than its velocity into consideration when estimating to what extent the predicted work ahead will obstruct the vehicle. Such a solution would be independent of the load in the trailer. It is also possible to modify the way the gearshift point offsets are calculated. For instance, the offsets could be functions of the workload, rather than pre-determined values. Another interesting study could be made on whether it is possible to use look-ahead information in order to improve the routine that translates desired throttle (gas pedal) to actual throttle. For example, there might be situations in which the driver pushes the gas pedal to the bottom in order to go faster, but the significantly increased fuel consumption causes a total economic drawback. The possibilities with a combination of a better gearshift strategy and a better throttle routine would be very interesting to investigate.

\textsuperscript{1}Lars Asplund
Bibliography


Appendix A

Correlation between articulation angle and turning radius

$\Phi$

$1_l$

$2_l$

$1_r$

$2_r$

Figure A.1: Correlation between articulation angle and turning radius.

$$r_2 = \frac{l_1}{\sin \Phi} + \frac{l_2 \cos \Phi}{\sin \Phi} = \frac{l_1 + l_2 \cos \Phi}{\sin \Phi} \quad (A.1)$$

$$r_1 = l_2 \sin \Phi + r_2 \cos \Phi = l_2 \sin \Phi + \frac{l_1 + l_2 \cos \Phi}{\sin \Phi} \cos \Phi = l_2 \sin \Phi + (l_1 + l_2 \cos \Phi) \left( \frac{\cos \Phi}{\sin \Phi} \right) \quad (A.2)$$
Appendix B

LTP (Local Tangent Plane) coordinate system

Figure B.1: The LTP coordinate system is a Cartesian coordinate system where the z-axis is a normal vector to the surface it touches.