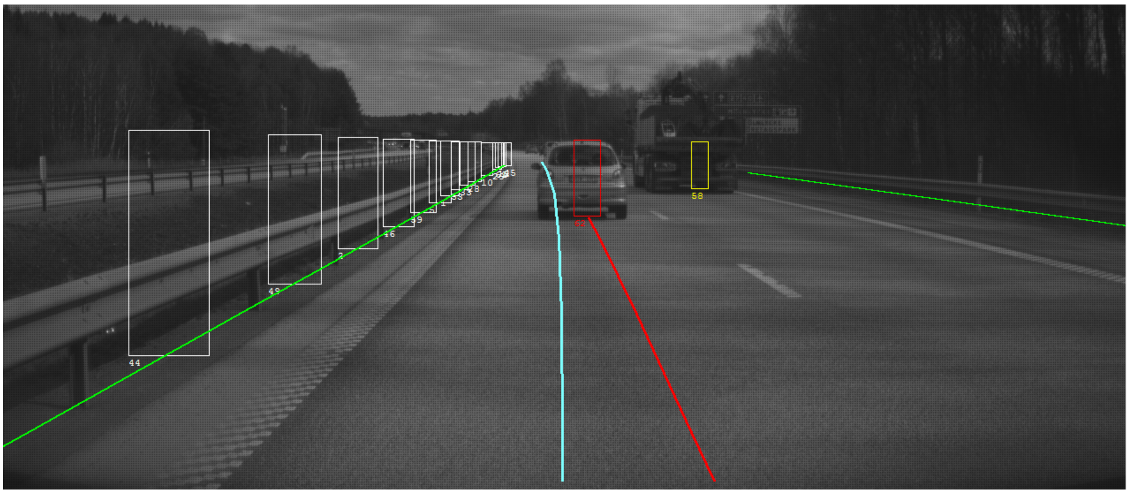
CHALMERS
UNIVERSITY OF TECHNOLOGY



# Radar Only Highway Target Lane Following

In collaboration with Delphi Automotive

Master's thesis in Systems and Control

JOHANNES GUNNARSSON
OSCAR PANTZARE

Department of Electrical Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2017

Master's thesis EX018/2017

# Radar Only Highway Target Lane Following

In collaboration with Delphi Automotive

JOHANNES GUNNARSSON
OSCAR PANTZARE

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Electrical Engineering
*Division of Systems and Control*
Chalmers University of Technology
Gothenburg, Sweden 2017

Radar Only Highway Target Lane Following
In collaboration with Delphi Automotive
JOHANNES GUNNARSSON
OSCAR PANTZARE



Supervisor: Martin X Larsson, Delphi Automotive
Examiner: Jonas Fredriksson, Department of Electrical Engineering

Master's Thesis EX018/2017
Department of Electrical Engineering
Division of Systems and Control
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: Video capture of active lane following, showing reference polynomial (red) and road prediction (light blue)

Typeset in LaTeX
Gothenburg, Sweden 2017

Radar Only Highway Target Lane Following
In collaboration with Delphi Automotive
Johannes Gunnarsson
Oscar Pantzare
Department of Electrical Engineering
Chalmers University of Technology

## Abstract

The thesis investigates in how a radar only target lane following trajectory can be designed in order to achieve both longitudinal and lateral control on highway driving. The algorithm presented in this report estimates the road in order to identify and validate possible target vehicle. Based on the target vehicle motion is a trail generated which the trajectory is based on. Experimental results demonstrate that it is possible to achieve a radar only trajectory for both longitudinal and lateral control.

Keywords: Radar-only, Lane following, Highway trajectory, Advanced driver assistance system

# Acknowledgements

We would like to thank following persons for their contributions to the project:

**Jonas Fredriksson**, for being our supervisor and examiner at Chalmers University of Technology.
**Martin X Larsson**, for being our supervisor at Delphi Automotive and supporting the project.
**Joachim Rukin**, for support, inspiration and guidance in the right direction.
**Adam Petterson**, for countless time giving support and help when we needed it.
**Maciej Różewicz**, for implementation of Volvo trucks guardrail tracker in Volvo Cars cads4 platform.
**Mats Björnerbäck**, for giving us the opportunity to do our master thesis at Delphi.
**Claes Tornevall**, for support with testing in vehicle.
**Daniel Pham**, for delivery of radar only embedded software.

Johannes Gunnarsson & Oscar Pantzare, Gothenburg, July, 2017

# Contents

# List of Figures

# List of Tables

# 1

# Introduction

The first advanced driver assistance systems (ADAS) released was adaptive cruise control (ACC) in 1995 in Japan and 1998 in Europe [1]. Since then the research and development of ADAS has been exponentially increasing. Today ADAS does not only support longitudinal control but also lateral control. Luxury car manufacturers offer lane keeping assistance (LKA), lane following (LF) and even autopilot systems with the help of sensor fusion systems. In most cases these sensor fusion systems are based on radar, vision and in some cases ultrasonic sensors [2]. These sensor fusion systems provide often information in 360 degrees around the vehicle that is used for feature function development of the ADAS.

Delphi Automotive Systems [3] (Delphi) is one of the worlds leading suppliers of hardware and software for ADAS. In collaboration with Volvo Cars and Mobileye, Delphi has pushed the limit of what is possible with ADAS. The sensor fusion system used by Delphi and equivalent companies are often expensive and not accessible for every car manufacturer. By limiting the use of sensors to a radar-only design, ADAS can become cheaper and more accessible.

Today *radar-only* systems are able to provide information for adaptive cruise control, forward collision warning and automatic emergency braking systems (AEB) [4]. But yet no system for lateral control has been implemented by only using radar sensors. If a radar-only system was able to assist the driver with both longitudinal and lateral control, highway assistant pilot system would become more affordable and available to more people and potentially save lives.

## 1.1 Purpose and goals

The purpose of this project was to investigate how a radar-only setup can be used in order to create a highway lane following trajectory for both longitudinal and lateral control. The goal of the project was to design a target lane following trajectory based on a radar-only platform. The aim was to generate a trajectory based on a snail trail of a selected target vehicle that will be used as a reference for the host vehicle. The target vehicle then provide information for both longitudinal and lateral control.

## 1.2 Use case

The trajectory algorithm have a limited operating range including both road type and velocity range. The road type is restricted to highway roads with no directly oncoming vehicles. The velocity range is from $70km/h$ to the recommended speed limits in the actual country. The target vehicle is also limited to be in the same lane as the host vehicle in order to be considered a reference vehicle. If the target vehicle is changing lane, leaving the highway or driving in a non reliable and illegal way the trajectory algorithm should inform the driver and give back control of the host vehicle. The intended use of the system is to assist the driver during highway driving. The system should not be considered an autonomous system and the driver should always superintend the vehicle and the traffic.

### 1.2.1 Specification

- The trajectory should stop following a target vehicle if it:
  - Changes lane
  - Exits highway
  - Drives in an unreliable and illegal way
- If another vehicle unexpectedly cuts in between the target vehicle and the host vehicle, the system should warn the driver and stop following the target vehicle. Depending on the situation the vehicle cutting in could be considered as a new target vehicle.
- The trajectory algorithm is limited to following traffic situations:
  - Highway driving
  - At least two lanes are present
  - No direct oncoming traffic
  - Host vehicle velocity above 70 km/h

## 1.3 Delimitations

- Only front sensing radar and host vehicle data are given as input to the trajectory algorithm.
- The project focuses only on the trajectory design, no lateral and longitudinal controller was designed or implemented. Some control theory was discussed in theory in order to understand how a controller could be designed.
- The performance was evaluated both by analysis in Matlab and by testing the system on specific routes in specific traffic situations.
- No interface or communication to the driver was designed or implemented implemented.

## 1.4 Thesis outline

This report includes six chapters besides the introduction chapter.

In chapter two the result from a background study is presented. The background includes initial decisions regarding hardware setup, software tools and verification method. The background also includes information about traffic regulations, properties of highway design and ethical and sustainable aspects of the project.

In chapter three the theory and methods that the project is based on is presented. The theory and methods chapter includes both general theory that is useful for understanding the report as well as specific mathematical algorithms that are implemented in the project.

In chapter four the design of the trajectory algorithm is presented. The design of the trajectory algorithm describes the method of the project and how the theory presented in chapter three is implemented in the trajectory algorithm.

In chapter five the result is presented. The result includes both result from simulations and real time testing in vehicle. The result section discusses the individual parts of the algorithm and the output of the algorithm.

In chapter six the discussion of the result is presented and in the final chapter seven is the conclusion presented.

# 2
# Background

In this chapter the outcome from the background study, including initial decisions and progress of the project, is presented. This includes description of hardware setup, software tools, traffic regulations and verification method.

## 2.1 Hardware setup

The platform for the project is Delphis and Volvo Cars cads4 project, where Delphi in collaboration with Mobileye assisted Volvo with a RaCam unit including both hardware and software. The RaCam is a single unit integrated sensor fusion system of radar and vision. With the RaCam, ADAS like ACC, LKA, AEB, Automatic Head Beam Control (AHBC), traffic sign recognition (TSR) and vehicle, pedestrian, animals and general object identification are possible. In this project only the radar data is considered and not the fused radar and vision data. The fused data can be used as a tool for verification and finding situations of interest, for example lane change of target vehicle.

The radar unit in the RaCam is an electronically scanning radar (ESR). The ESR includes one long range radar scan and one mid-range radar scan. The long range radar has a range of $200m$, a field of view of $\pm10°$ and accuracy of $\pm0.5m$. The mid range radar has a range of $60m$, a field of view of $\pm45°$ and accuracy of $\pm0.25m$. The field of view for both long and mid range scans are visualized in figure: 2.1. Both long and mid-range radar have a update rate of $50ms$.

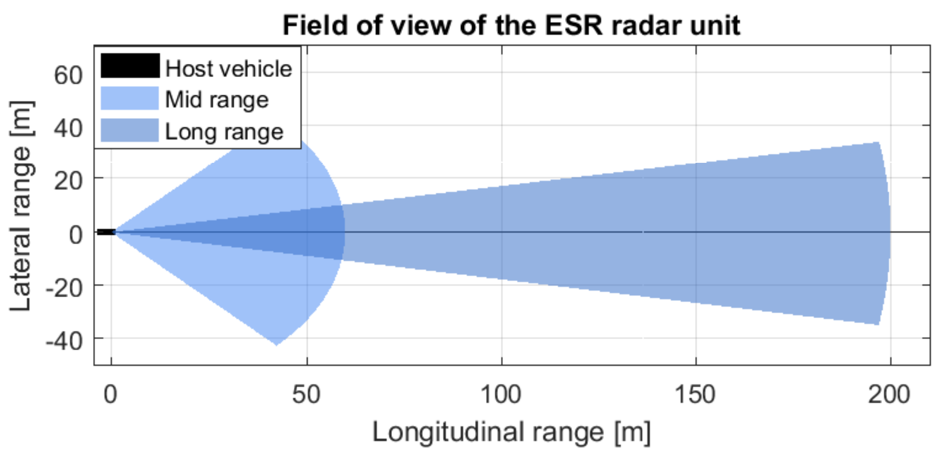



**Figure 2.1:** Field of view of the ESR radar unit.

## 2.2 Software tools

In order to analyze the data and develop algorithms, several software tools are needed.

Matlab was used for doing numerical analysis, data visualization and algorithm development. In order to implement the developed algorithms in the embedded software, Matlab Coder was used to convert the algorithms into C++ code. To make this possible the code developed in Matlab need to take the characteristics of the C++ language taken into account

DVtool (DataView) was used in order to verify functions developed in Matlab. DVtool visualizes selected data in different ways, e.g in the view of the driver (3D) or in a helicopter perspective (2D). DVtool can both be utilized to analyze logged data or live in the vehicle. An additional view in DVtool was implemented where the suggested trajectory is shown as a line on the road in order to easily overview the outcome of the trajectory algorithm.

TrackerPC will be used for the testing of the algorithms in reality without doing the implementation in the host vehicle. This is done by running the algorithms in real time on the computer while driving with the inputs taken from the host vehicle.

## 2.3 Road properties and traffic regulations

Information about properties and design of roads and traffic regulations should be considered in the trajectory design. Specifications like maximum road curvature and recommendation of distance to vehicle ahead is of special interest for the estimation of the road and design of the trajectory.

### 2.3.1 Road curvature

The government agency Swedish Transport Administration (In swedish: Trafikverket) has certain rules and requirements for new roads. Two of these are about road curvature and road curvature change. There is an inverse relation between curve radius and curvature,

$$C = \frac{1}{R} \tag{2.1}$$

where $C$ is the curvature of the road and $R$ is the radius of the curve.

The maximum allowed road curvature and road curvature change are speed limit dependent. In table: 2.1 a list of maximal allowed road curvature and curvature change for different speed limits is presented [5].

| Speed limit $[km/h]$ | Maximal Curvature $[1/m]$ | Maximal Curvature change $[1/m]$ |
|---|---|---|
| 60 | $7.1 \cdot 10^{-3}$ | $3.3 \cdot 10^{-3}$ |
| 80 | $2.5 \cdot 10^{-3}$ | $2 \cdot 10^{-3}$ |
| 100 | $1.4 \cdot 10^{-3}$ | $1.3 \cdot 10^{-3}$ |
| 110 | $1.1 \cdot 10^{-3}$ | $1.1 \cdot 10^{-3}$ |
| 120 | $0.8 \cdot 10^{-3}$ | $1 \cdot 10^{-3}$ |

**Table 2.1:** Maximal curvature and curvature change for different speed limits.

From the road curvature maximum lateral acceleration be calculated using the relation $a_y = Cv_x^2$, where $v_x$ is the longitudinal velocity. This gives:

| Speed limit $[km/h]$ | Maximal lateral acc $[m/s^2]$ | Maximal lateral jerk $[m/s^3]$ |
|---|---|---|
| 60 | 1.98 | $1.188 \cdot 10^{-4}$ |
| 80 | 1.23 | $1.28 \cdot 10^{-4}$ |
| 100 | 1.10 | $1.3 \cdot 10^{-4}$ |
| 110 | 1.04 | $1.331 \cdot 10^{-4}$ |
| 120 | 0.93 | $1.440 \cdot 10^{-4}$ |

**Table 2.2:** Maximal acceleration and acceleration change for different speed limits.

In order to have the maximal curvature as specified as in table: 2.1 the road is required to have banked turns with a slope of at least 4%, which is further described in section: 2.3.2.

### 2.3.2 Road banking

All roads are not perfectly flat, this is made by purpose in order to drain water from the road surface, reduce wear of tires and improve safety. Exactly how roads should be built is regulated by Trafikverket [5]. At a road section which is considered straight, the cross slope of the road should be 2.5% with the lowest point at the right side of the road. In a turn with high curvature, the cross slope of the road should be $> 4\%$ with the lowest point at the side which the road is turning towards.

If the turn is not banked with 4% or more, the maximally allowed curvature is reduced as shown in table: 2.3.

| Speed limit $[km/h]$ | Maximal Curvature $[1/m]$ |
|---|---|
| 60 | $0.67 \cdot 10^{-3}$ |
| 80 | $0.4 \cdot 10^{-3}$ |
| 100 | $0.26 \cdot 10^{-3}$ |
| 110 | $0.22 \cdot 10^{-3}$ |
| 120 | $0.18 \cdot 10^{-3}$ |

**Table 2.3:** Maximal curvature in curves without sufficent banking.

### 2.3.3 Distance to vehicles ahead

The distance to a vehicle ahead should according to the law of Sweden, Trafikförordningen (1998:1276) 3:2 [6], always be adapted so that there is no risk of collision if the car ahead slows down or stops. The recommendation from Trafikverket is the "three second rule" which means that one needs to keep a distance of three seconds to the vehicle ahead [7]. However this is only a recommendation and driving with a headway below three seconds will not be considered as violation of Trafikförordningen (1998:1276) 3:2.

The three second rule is most likely not applicable in a lane vehicle following system. For example, if driving in 120 km/h this gives a longitudinal distance of 100m. This is a distance that in high density traffic situations will lead to cut-ins between the host and target vehicle. Headway between $1 - 2s$ is more likely applicable for a lane following system which corresponds to a distance of $33 - 66m$ at $120km/h$. The distance to the vehicle ahead should however never be shorter than that the vehicle in a safe way can inform the driver to take back steering if so needed.

## 2.4 Trajectory design

Given that a target vehicle satisfies the requirements, a trajectory should be generated based on the target vehicle motion. The longitudinal reference should be given by a selected longitudinal distance between the host vehicle and target vehicle, selected by the driver of the host vehicle. The lateral reference should be based on a snail trail of the target generated and represented as coefficients of a third degree polynomial. Based on the radar trail and eventual information about obstacles in the path a path plan algorithm optimizes the future path. This path will be generated as a lateral trajectory for the host vehicle. The reference should include:

- Selected longitudinal distance between the host and the target vehicle.
- Longitudinal distance, velocity and acceleration of the target vehicle.
- Coefficients for a third degree polynomial representing the lateral future path.

If a target vehicle no longer satisfies the requirements, the trajectory should be reset.

## 2.5 Verification method

The verification method includes both offline and online (real time overview) hardware in the loop test. Offline verification can be achieved by comparing the trajectory algorithm output with radar and vision fusion output. For online hardware in the loop test the trajectory algorithm is implemented in a trackerPC. Together with implementation of a new view in DVtool the output of the trajectory could be monitored in real time.

### 2.5.1 Test route

In order to check if the trajectory algorithm meets the requirement a test route is defined. The defined test route is between Delsjömotet and Flygplatsmotet on E4/40 in both directions. In figure: 2.2 the test route is illustrated. The test route includes sections with different speed limits, variation in curvature and number of lanes.

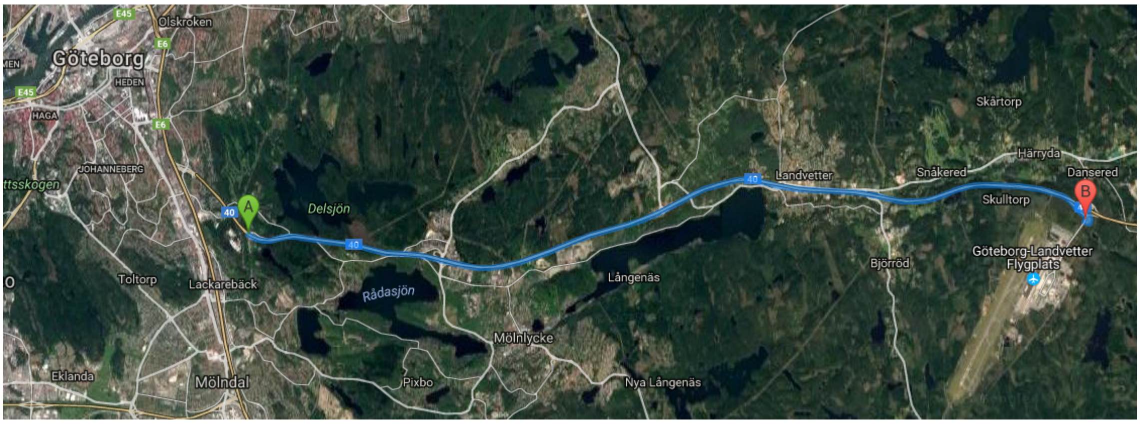



**Figure 2.2:** Illustration of test route, Delsjömotet marked as A and Flygplatsmotet marked as B.

# 3

# Theory and methods

In this chapter the theory and methods that the project is based on is presented. The theory and methods chapter includes both general theories that are useful for understanding the report and also specific mathematical algorithms that are implemented in the report. For simplified notation in this chapter, the time variable $t$ is removed from all time dependent variables.

## 3.1 Basic principle of the Radar

In general a radar unit consists of two major parts, a transmitter and a receiver. By comparing the received effect to the transmitted effect the distance to an object creating a reflection can be computed [8]. One registration of the receiver will be denoted as detection. By utilizing the Doppler effect, it is possible to determine whether a detection comes from a stationary or a moving object and the relative speed of the object can be obtained [9].

In a sensor fusion software the detections are processed into tracklets and objects. A tracklet is a single detection or multiple detections which are followed from the previous time step by using an extended kalman filter (EKF) with a constant velocity (CV) motion model. An object is a single tracklet or group of tracklets close to each other with a similar behavior. The properties of an object can be determined from the tracklets (length, width, speed, heading etc). All moving objects are considered rectangular shaped vehicles.

The ESR radar used in the RaCam does not have elevation measurement of the radar detections. This means that an overhead sign, a small steel plate on the ground or a parked car could be represented in the same way by the radar detection. This leads to obstacle avoidance being hard to achieve without visually identifying the object.

## 3.2 Vehicle motion

The coordinate system used is the Society of Automotive Engineers (SAE) coordinate system which is defined in figure: 3.1, where the origin is the position of the radar unit.

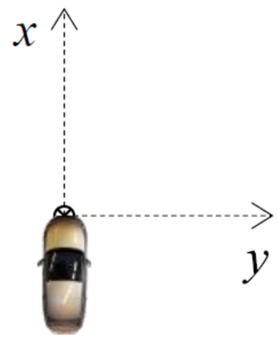



**Figure 3.1:** Illustration of the SAE coordinate system.

Depending on usage four different vehicle motion models are mentioned in the report:

- Constant velocity model
- Constant acceleration model
- Coordinated turn model
- One track model (two wheel vehicle motion model, bicycle model)

### 3.2.1 Constant velocity model

The most basic model of motion is the constant velocity model where the position and velocity represent the states of the model ($x = [p \quad v]^T$). The model assumes constant velocity and changes in velocity are modeled as disturbances [10]. In the one dimensional case the motion of an object can be described by:

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ \tilde{q} \end{bmatrix}. \tag{3.1}$$

### 3.2.2 Constant acceleration model

An extension of the constant velocity model is the constant acceleration model which adds another state for acceleration ($x = [p \quad v \quad a]^T$). The assumption now is that the acceleration is constant instead of the velocity. Similar to the constant velocity model changes in acceleration are modeled as a disturbance. In the one dimensional case the motion of an object can be described by:

$$\dot{x} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ \tilde{q} \end{bmatrix}. \tag{3.2}$$

### 3.2.3 Coordinated turn model

In order to accurately model a curve of a highway the non-linear coordinated turn model is a good option since it describes the motion along the edge of a circle. Noise is added to the velocity state $v$ and turning rate state $\omega$, in order to model changes in the curvature and speed.
The other states are:

- $p_x$ - representing the position of the tracked object along the x-axis
- $p_y$ - representing the position of the tracked object along the y-axis
- $\phi$ - representing heading of the tracked object

These states are defined as shown in figure 3.2.

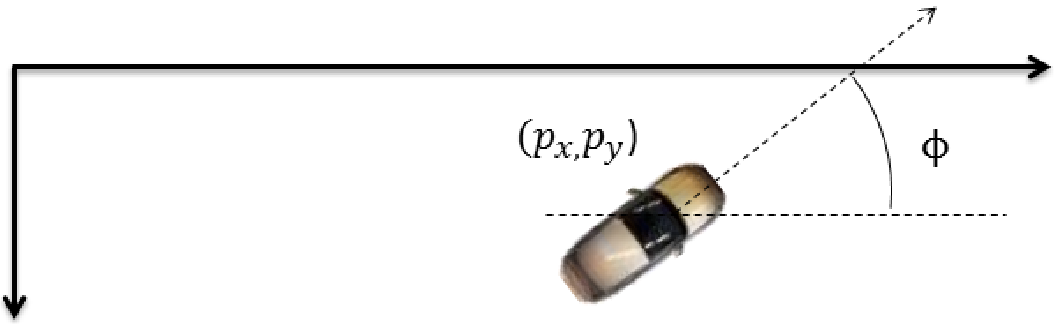



**Figure 3.2:** Illustation of coordinated turn model.

The state vector and transition equation are:

$$x = \begin{bmatrix} p_x \\ p_y \\ v \\ \phi \\ \omega \end{bmatrix} \qquad \dot{x} = \begin{bmatrix} v \cdot cos(\phi) \\ v \cdot sin(\phi) \\ a \\ \omega \\ \alpha \end{bmatrix} \tag{3.3}$$

where $a$ and $\alpha$ are the noise components added, this corresponds to $\tilde{q}$ in the constant acceleration and velocity models.

### 3.2.4 Transient one track model

The transient one track model is commonly used to describe lateral dynamics of a vehicle. Assuming one axle captures the most important phenomena of lateral dynamics this simplifies the calculation compared to using a two track model [11]. In figure: 3.3 a one track model is illustrated assuming no wind drag.

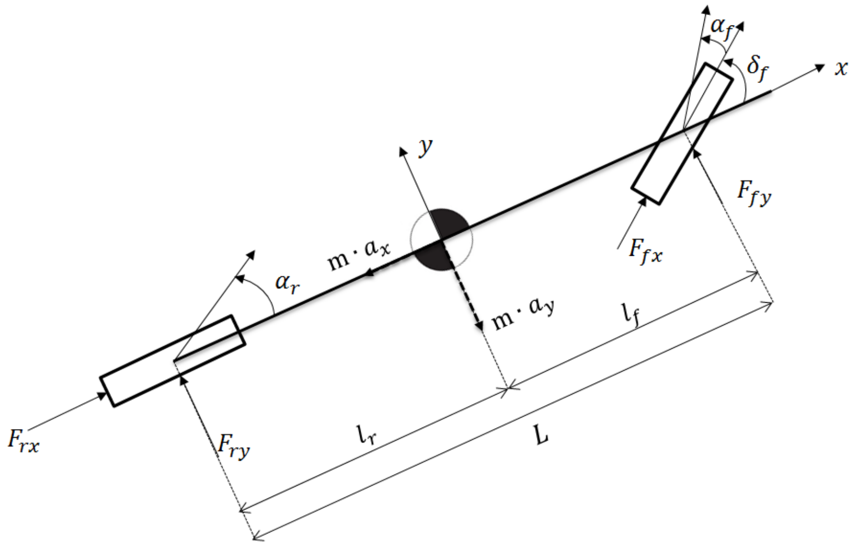



**Figure 3.3:** Free body diagram of the one track model including forces, steering and slip angles and fundamental dimensions.

Equilibrium equations given by the free body diagram:

$$\begin{aligned} m \cdot a_x &= F_{fx} \cdot \cos(\delta_f) - F_{fy} \cdot \sin(\delta_f) + F_{rx} \\ m \cdot a_y &= F_{fx} \cdot \sin(\delta_f) + F_{fy} \cdot \cos(\delta_f) + F_{ry} \\ J \cdot \dot{\omega}_z &= (F_{fx} \cdot sin(\delta_f) + F_{fy} \cdot cos(\delta_f)) \cdot l_f - F_{ry} \cdot l_r \end{aligned} \tag{3.4}$$

where $a_x$ and $a_y$ can be described by:

$$a_x = \dot{v}_x - \omega_z \cdot v_y \qquad a_y = \dot{v}_y + \omega_z \cdot v_x. \tag{3.5}$$

Following small angle assumptions are reasonable [11]:

- Small steering angle, $sin(\delta_f) \approx \delta_f$ and $cos(\delta_f) \approx 1$
- Small tire slip angle, $s_{fy} \approx \alpha_f$ and $s_{ry} \approx \alpha_r$
- Small body slip angles ($\beta_f$ and $\beta_r$), $\alpha_f = \beta_f - \delta_f$ and $\alpha_r = \beta_r$ where $\beta_f$ and $\beta_r$ are approximated according:

$$\beta_f = \frac{v_y + \omega_z \cdot l_f}{v_x} \qquad \beta_r = \frac{v_y - \omega \cdot l_r}{v_x} \tag{3.6}$$

The slip angles $\alpha_f$ and $\alpha_r$ are then given by:

$$\alpha_f = \delta_f - \frac{v_y + \omega_z \cdot l_f}{v_x} \qquad \alpha_r = \frac{v_y - \omega \cdot l_r}{v_x}. \tag{3.7}$$

Assuming linear tire model for cornering, where $C_f$ and $C_r$ is the cornering stiffness of front and rear wheels, the lateral forces is described by:

$$F_{fy} = -C_f \cdot s_{fy} \qquad F_{ry} = -C_r \cdot s_{ry}. \tag{3.8}$$

The equilibrium equations can now be rewritten as:

$$
\begin{aligned}
m \cdot \dot{v}_x &= m \cdot \omega_z \cdot v_y + C_f \frac{\omega_z \cdot l_f + v_y}{v_x} \delta_f \\
m \cdot \dot{v}_y &= C_r(\frac{\omega_z \cdot l_r - v_y}{v_x}) - C_f(\frac{\omega_z \cdot l_f + v_y}{v_x}) - m \cdot \omega \cdot v_x + C_f \cdot \delta_f \\
J \cdot \dot{\omega}_z &= -l_f \cdot C_f(\frac{\omega_z \cdot l_r + v_y}{v_x}) - l_r \cdot C_r(\frac{\omega_z \cdot l_r - v_y}{v_x}) + l_f \cdot C_f \cdot \delta_f.
\end{aligned}
\tag{3.9}
$$

For lateral dynamics the longitudinal equilibrium equation can be neglected and a state vector can be defined as: $\mathbf{x} = [y \quad \dot{y} \quad \omega \quad \dot{\omega}]^T$ where $\dot{y} = v_y$ which gives the state space equation:

$$
\dot{\mathbf{x}} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{C_f + C_r}{m \cdot v_x} & 0 & -\frac{c_f \cdot l_f - c_r \cdot l_r}{m \cdot v_x} - v_x \\ 0 & 0 & 1 & 0 \\ 0 & -\frac{c_f \cdot l_f - c_r \cdot l_r}{J \cdot v_x} & 0 & -\frac{c_f \cdot l_f^2 + c_r \cdot l_r^2}{J \cdot v_x} \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \\ \frac{c_f}{m} \\ 0 \\ \frac{l_f \cdot c_f}{J} \end{bmatrix} \delta_f. \tag{3.10}
$$

## 3.3 Polynomial fitting

Polynomial fitting is a method used to find a function describing a data set in the best possible way. The degree of the polynomial can vary depending on the application, but the maximum degree has to be at least one less than the size of the data set in order to find a unique polynomial for the data set. In order to find a third degree polynomial the data set must contain four or more points. To find the coefficients $p$, the following system of linear equations has to be solved [12]:

$$
Vp = y \tag{3.11}
$$

where $V$ is a matrix (the Vandermonde matrix), and $y$ and $p$ are vectors formed as:

$$
V = \begin{bmatrix} x_1^n & x_1^{n-1} & \dots & 1 \\ x_2^n & x_2^{n-1} & \dots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ x_m^n & x_m^{n-1} & \dots & 1 \end{bmatrix} \qquad y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_m \end{bmatrix} \qquad p = \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{bmatrix} \tag{3.12}
$$

where $m$ is the size of the data set and $n$ is the the degree of the requested polynomial and $x_i$ and $y_i$ is the data of interest.

## 3.4 Bayesian statistics and filtering

In this section Bayesian statistics and filtering is introduced. Two filtering and smoothing methods that utilize the concept of Bayesian statistics are also presented.

### 3.4.1 Basic Bayesian statistics

Bayesian statistics is a statistical inference framework that can be used for estimation, classification, detection and model selection [13]. In Bayesian statistics the unknown quantities are described as random. In order to estimate the state $x$ given a measurement $y$ the Bayesian method includes three key steps:

- **Modeling**
  Modeling what is known about $x$ and the measurement $y$ by utilizing the prior $p(x)$ and the density $p(y|x)$. The prior is defined as the probability distribution of the unknown parameter $x$ before observation. The conditional density $p(y|x)$ is defined by:

$$p(x, y) = p(y|x)p(x). \tag{3.13}$$

- **Measurement update**
  Combining the prior with the measurement is done in order to summarize what was known about $x$. This is known as the likelihood and denoted as $p(x|y)$.

- **Decision making**
  Decision making is done by minimizing the expected loss when calculating a posterior given the likelihood of $x$ and a loss function $C(x, \hat{x})$, where $\hat{x}$ is the estimation of $x$. The posterior can be calculated using Bayes' rule:

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)} \tag{3.14}$$

  and the optimal Bayesian decision is defined by:

$$\hat{x} = arg \min_{a} \mathbb{E}\{C(x, a)|y\}. \tag{3.15}$$

### 3.4.2 Basic Bayesian filtering

When the unknown parameter $x$ is time dependent the Bayesian statistics estimation approach can be modified in order to recursively, at every time step $k$, estimate the unknown parameter. By redefining $x$ and $y$ as $x_k$ and $y_k$ it is possible to recursively compute the posterior $p(x_k|y_{1:k})$ from $p(x_{k-1}|y_{1:k-1})$ where $y_{1:k} = [y_1 \quad y_2 \quad \dots \quad y_k]$ contains all data up to time $k$.

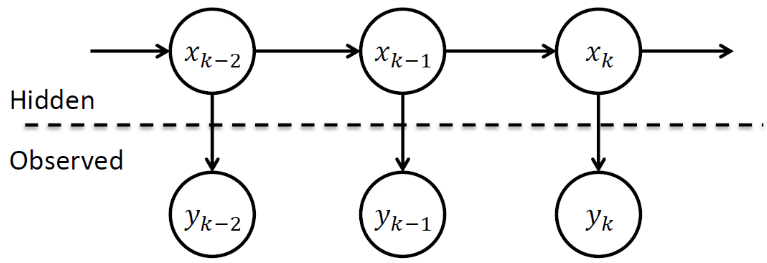



**Figure 3.4:** State space model described as a Bayesian network.

Introducing time-discrete state space model with the motion model $p(x_k|x_{k-1})$, the measurement model $p(y_k|x_k)$, assuming that $x_0 \propto p(x_0)$ and that:

$$p(x_k|x_{0:k-1}, y_{1:k-1}) = p(x_k|x_{k-1}) \tag{3.16}$$

and

$$p(y_k|x_{0:k}, y_{1:k-1}) = p(y_k|x_k). \tag{3.17}$$

Both $x_k$ and $y_k$ are stochastic processes and the assumption in equation: 3.16 that the future state only depends on the present state and not the past is recognized as the Markov property which implies $x_k$ is a Markov process.

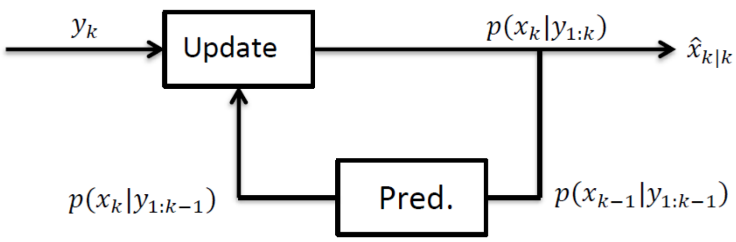



**Figure 3.5:** Block diagram illustrating the recursive estimation of the parameter x given the measurement y.

- **Prediction step**
  The prediction $p(x_k|y_{1:k-1})$ from $p(x_{k-1}|y_{1:k-1})$ is given by:

$$\begin{aligned} p(x_k|y_{1:k-1}) &= \int p(x_k, x_{k-1}|y_{1:k-1})dx_{k-1} \\ &= \int p(x_k|x_{k-1}, y_{1:k-1})p(x_{k-1}, |y_{1:k-1})dx_{k-1} \\ &= \int p(x_k|x_{k-1})p(x_{k-1}, |y_{1:k-1})dx_{k-1} \end{aligned} \tag{3.18}$$

  this is recognized as the Chapman-Kolmogorov equation.

- **Measurement update**
  The measurement update computation of $p(x_k|y_{1:k})$ from $p(x_k|y_{1:k-1})$ is given by:

$$\begin{aligned} p(x_k|y_{1:k}) &= p(x_k|y_k, y_{1:k-1}) \\ &= \frac{p(y_k|x_k, y_{1:k-1})p(x_k|y_{1:k-1})}{p(y_k|y_{1:k-1})} \\ &= \frac{p(y_k|x_k)p(x_k|y_{1:k-1})}{p(y_k|y_{1:k-1})}. \end{aligned} \tag{3.19}$$

This can be summarized as the filtering equation:

$$\begin{aligned} &p(x_k|y_{1:k-1}) = \int p(x_k|x_{k-1})p(x_{k-1}, |y_{1:k-1})dx_{k-1} \\ &p(x_k|y_{1:k}) \propto p(y_k|x_k)p(x_k|y_{1:k-1}). \end{aligned} \tag{3.20}$$

### 3.4.3 Kalman filtering

The filtering equation (equation: 3.20) is applicable for all types of filtering problems. However, the posterior distributions often have no analytical expression. The most important exception are linear and Gaussian models:

$$\begin{aligned} x_k &= A_{k-1}x_{k-1} + q_{k-1} \\ y_k &= H_k x_k + r_k \end{aligned} \tag{3.21}$$

where $A_{k-1}$ is the transition matrix and $H_k$ the measurement model matrix.

For linear and Gaussian models $p(x_m|y_{1:n})$, where $m < n$ is Gaussian for all $m$ and $n$, letting $\hat{x}_{m|n}$ be the mean and $P_{m|n}$ be the covariances of $p(x_m|y_{1:n})$ such that $p(x_m|y_{1:n}) = \mathcal{N}(x_m; \hat{x}_{m|n}, P_{m|n})$. The Kalman filer, recursively computes $\hat{x}_{k|k-1}$, $P_{k|k-1}$, $\hat{x}_{k|k}$ and $P_{k|k}$ for $k = 1, 2...$ [14].

The algorithm has two major steps, the prediction and update step. The prediction step uses the model of the system to estimate how the states of the system have changed since the last measurement. In the update step, the new measurement is compared to the predicted state and an optimal estimation is found based on the certainty of the model and the measured value (accuracy of the sensor).

- **Prediction step**

$$\begin{aligned} \hat{x}_{k|k-1} &= A_{k-1}\hat{x}_{k-1|k-1} \\ P_{k|k-1} &= A_{k-1}P_{k-1|k-1}A_{k-1}^T + Q_{k-1} \end{aligned} \tag{3.22}$$

- **Update step**

$$\begin{aligned} \hat{x}_{k|k} &= \hat{x}_{k|k-1} + K_k v_k \\ P_{k|k} &= P_{k|k-1} + K_k S_k K_k^T \end{aligned} \tag{3.23}$$

Where the Kalman gain $K_k$, the innovation $v_k$ and the innovation covariance $S_k$ are defined by:

$$\begin{aligned} K_k &= P_{k|k-1}H_k^T S_k^{-1} \\ v_k &= y_k - H_k\hat{x}_{k|k-1} \\ S_k &= H_k P_{k|k-1} H_k^T + R_k. \end{aligned} \tag{3.24}$$

### 3.4.4 Extended Kalman filtering

The Kalman filter presented in chapter: 3.4.3 gives the optimal solution for linear models but is not applicable for non-linear models. Instead the Extended Kalman filter is used for non-linear models [15].

The extended Kalman filter linearizes the non-linear models around the previous state estimation. A non-linear model:

$$\begin{aligned} x_k &= f_{k-1}(x_{k-1}) + q_{k-1} \\ y_k &= h(x_k) + r_k \end{aligned} \tag{3.25}$$

where $f_{k-1}(x_{k-1})$ is the motion model, $h_k(x_k)$ is the measurement model and $q_{k-1}$ and $r_k$ are Gaussian random variables with covariance $Q_{k-1}$ and $R_k$. Using Taylor series expansions $f_{k-1}(x_{k-1})$ is linearized around $x_{k-1|k-1}$ and $h_k(x_k)$ around $\hat{x}_{k|k-1}$. The linear approximation of the system (equation: 3.25) is given by:

$$\begin{aligned} x_k &\approx f(\hat{x}_{k-1|k-1}) + f'(\hat{x}_{k-1|k-1})(x_{k-1} - \hat{x}_{k-1|k-1}) + q_{k-1} \\ y_k &\approx h(\hat{x}_{k|k-1}) + h'(\hat{x}_{k|k-1})(x_{k-1} - \hat{x}_{k-1|k-1})r_k \end{aligned} \tag{3.26}$$

where $f'_{k-1}(x_{k-1})$ and $h'_k(x_k)$ are the Jacobian matrices of the motion and measurement model and $\hat{x}$ is the estimated state. The Kalman filtering equations can then be applied on the linearized system.

- **The prediction step** for the extended Kalman filter is then given by:

$$\begin{aligned} \hat{x}_{k|k-1} &= f(\hat{x}_{k-1|k-1}) \\ P_{k|k-1} &= f'(\hat{x}_{k-1|k-1})P_{k-1|k-1}f'(\hat{x}_{k-1|k-1})^T + Q_{k-1}. \end{aligned} \tag{3.27}$$

- **The update step** for the extended Kalman filter is then given by:

$$\begin{aligned} \hat{x}_{k|k} &= \hat{x}_{k|k-1} + K_k(y_k - h(\hat{x}_{k|k-1})) \\ P_{k|k} &= P_{k|k-1} - K_k S_k K_k^T \\ S_k &= h'(\hat{x}_{k|k-1})P_{k|k-1}h'(\hat{x}_{k|k-1})^T + R_k \\ K_k &= P_{k|k-1}h'(\hat{x}_{k|k-1})^T S_k^{-1}. \end{aligned} \tag{3.28}$$

### 3.4.5 Rauch-Tung-Striebel smoothing

Fixed interval smoothing methods such as the forward-backward smoothing algorithm Rauch-Tung-Striebel can be applied in order to smooth the filtering result of a linear system [16]. The forward step of the algorithm is the Kalman filtering defined in chapter: 3.4.3 and summarized below:

$$\begin{aligned} K_k &= P_{k|k-1}H_k^T S_k^{-1} \\ \hat{x}_{k|k} &= \hat{x}_{k|k-1} + K_k\left(y_k - \hat{x}_{k|k-1}\right) \\ P_{k|k} &= P_{k|k-1} + K_k S_k K_k^T. \end{aligned} \tag{3.29}$$

When the forward step is run over $k = 1, .., K$ and the $\hat{x}_{k|k}$, $P_{k|k}$, $\hat{x}_{k+1|k}$ and $P_{k+1|k}$ are stored in each time instance. The backward smoothing step starting at $k = K-1$ defined as:

$$G_k = P_{k|k}A_k^T P_{k+1|k}^{-1} \tag{3.30}$$

$$\hat{x}_{k|K} = \hat{x}_{k|k} + G_k\left(\hat{x}_{k+1|K} - \hat{x}_{k+1|k}\right) \tag{3.31}$$

$$P_{k|K} = P_{k|K} + G_k\left[P_{k+1|K} - P_{k+1|k}\right]G_k^T. \tag{3.32}$$

### 3.4.6 Extended Rauch-Tung-Striebel smoothing

The Rauch-Tung-Striebel smoothing algorithm presented in section: 3.4.5 is only applicable for the linear case. In a non-linear system (equation: 3.25), where an extended Kalman filter is utilized instead for filtering, the extended Rauch-Tung-Striebel smoothing algorithm is utilized for smoothing. The extended Rauch-Tung-Striebel algorithm is defined as [13]:

$$
\begin{aligned}
\hat{x}^{-}_{k+1|k} &= f(\hat{x}_{k|k}) \\
P^{-}_{k+1|k} &= F(\hat{x}_{k|k})P_{k|k}F^{T}(\hat{x}_{k|k}) + Q_k \\
G_k &= P_{k|k}F^{T}(\hat{x}_{k|k})(P^{-}_{k+1|k})^{-1} \\
\hat{x}^{s}_{k|K} &= \hat{x}_{k|k} + G_k(\hat{x}^{s}_{k+1|K} - \hat{x}^{-}_{k+1|k}) \\
P^{s}_{k|K} &= P_{k|k} + G_k(P^{s}_{k+1|K} - P^{-}_{k+1|k})G^{T}_{k}.
\end{aligned}
\tag{3.33}
$$

## 3.5 Road model estimation

In a previous master thesis at Delphi a road model was designed [17]. The model utilizes fused radar and vision data. It is possible to modify this in order to only use radar data. The inputs to the road model are:

- **Lane marker estimation**
  Lane marker estimation is made from vision data and the estimation includes four lane markers, the two closest to the host vehicle and the two next to them, if they are present. The lane markers are reported as a third-order polynomial $y^i_k = a^i_{0,k} + a^i_{0,k}x^1_k + a^i_{0,k}x^2_k + a^i_{0,k}x^3_k$ where $i = [1, ..., 4]$. The parameters $a_{0,k}$, $a_{1,k}$, $a_{2,k}$ and $a_{3,k}$ are the coefficients of the polynomial. Each polynomial has a maximal valid range defined as $x^i_{l,max,k}$.
- **Barrier estimation**
  Barrier estimation is made from stationary radar data and one barrier at each side can be reported. When a barrier is present and reported it is represented as a third-order polynomial $y^i_k = b^i_{0,k} + b^i_{0,k}x^1_k + b^i_{0,k}x^2_k + b^i_{0,k}x^3_k$ where $i = [1 \quad 2]$ for different barriers. $y_k$ is the lateral position and $x_k$ is the corresponding longitudinal position in the coordinate system in figure: 3.6. The barrier/guard-rail estimation has also a valid range $x^i_{b,max,k}$.
- **Host vehicle motion**,
  Host vehicle motion including host speed $v_{host,k}$ and current yaw rate $\dot{\phi}_k$ is assumed to have no uncertainties. The host vehicles side slip angle $\alpha_{slip,k}$ is also reported.
- **Moving vehicle observations**
  Moving vehicle observation are used under the assumption that other vehicles also drive along the road. Heading $\phi_k$ is the utilized measurement of receding vehicles. The measurement is only valid at the position of the observed vehicle, the longitudinal distance $x_k$ is also reported in order to do the update in the correct state(s).

The basic idea of the road model is to Kalman filter the sampled curvature in $n$ number of sampled points of the road. The Kalman state vector is defined as $x_k = [\phi_k \quad C_{0,k} \quad ... \quad C_{n,k}]$ where $C_{0,k}$ is the curvature in the sampled points $i = 1, 2, ..., n$ at time $k$. The distance between the samples is defined as $\delta = \frac{PredictionHorizon}{n}$. In the implementation the prediction horizon was defined as $200m$ and $n = 40$. In figure: 3.6 the concept of the sampled road curvature is illustrated.

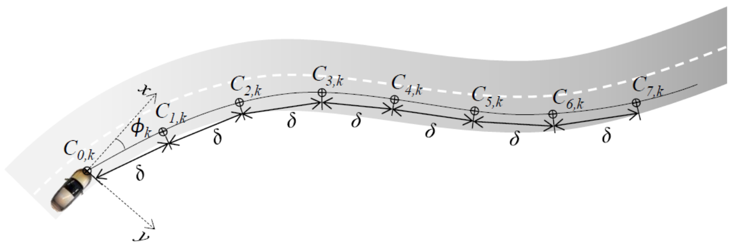



**Figure 3.6:** Illustration of the state representation in the local coordinate system.

Since the representation of lane markers and barriers estimation are both polynomials, three different measurement models are used for the four sources. The measurement models used are:

- **Host vehicle motion model**
  Since lateral control is provided by the driver it is assumed that the host vehicle heading is parallel to the road, any deviation is modelled as measurement noise. The measured $\alpha_{slip}$ equal to the heading state $\phi_k$. And similarly the calculated $C_{host} = d\psi_k / v_{host,k}$ is used as initial curvature $C_{0,k}$.
- **Lane and barrier polynomials**
  In order to utilize the information given, the polynomials are transformed to linear combinations of the state vector, e.g the polynomial shape is described by angles and curvatures.
- **Moving vehicle observations**
  The update of the angular measurement is done in a similar way as the angles calculated from the polynomials.

## 3.6 Path planning

In order to design a feasible trajectory, path planning can handle the decision making between following the radar trail of the target vehicle and avoiding obstacles in the path. Since only radar data is provided stationary objects are hard to give confidence. Therefore obstacles in the path will mainly be based on moving detections. Two popular path planning methods are based on Elastic Band Theory and Potential Fields.

### 3.6.1 Elastic Band Theory

The method of Elastic Bands minimizes the total energy of the elastic band spanning between the start and end point [18]. The band is modeled as a series of particles connected with springs. By definition the springs are in a relaxed state (energy equals to zero) when the band is connecting the two points in a straight line, and if the length of the band is increased an Internal Contraction Force is created by the springs. Obstacles is modeled as a potential repelling the particles, i.e. the force is higher if the band is closer to the obstacle than if it is far away.

### 3.6.2 Potential Fields Theory

The method of Potential Fields represents the space as a potential function, which have large values where there is an obstacle, and the free space has values corresponding to the distance to the nearest obstacle [19]. The value of the potential function corresponds to the cost of visiting that node. In order to find the best path the algorithm minimizes the sum of the cost of the visited nodes and the distance between them.

## 3.7 Basic stability and control theory

Vehicle control design is outside the scope of the project. However, in order to design a good trajectory some control theory understanding is important. Some basic stability and control theory, the definition of a linear quadratic regulator and how it can be implemented in a one track model as a lane centering controller is presented.

A linear system can be described by:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned} \tag{3.34}$$

where $x$ is the state vector and $u$ is the input vector, and $A$, $B$, $C$, $D$ are matrices and finally $y$ is the output vector. The system is input-output stable if all eigenvalues of $A$ are in the left half plane, not including the imaginary axis [20]. If some eigenvalues exists outside the left half plane the control signal $u$ can be selected to be state feedback:

$$u = -Lx. \tag{3.35}$$

If equation: 3.35 is included in equation: 3.34, then the resulting equation is:

$$\dot{x} = (A - BL)x \tag{3.36}$$

$L$ shall be chosen so that all eigenvalues of $(A - BL)$ are within the left half plane in order to make the system stable. If such a matrix $L$ exists the system is stabilizable [20].

Systems that cannot be expressed in the same form as in equation: 3.34 is considered to be a nonlinear system. A nonlinear system can be expressed as a linear system in a restricted range, and is considered to be stable around an equilibrium if all eigenvalues of the linear transition matrix $A$ have strictly negative real parts.

Typically a control system is constructed like in figure: 3.7, where the output is compared to the reference in order to find the error which is used to find the new control signal to the plant.

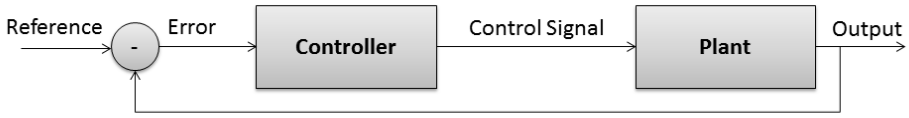



**Figure 3.7:** Basic control system.

### 3.7.1 Continuous time Linear Quadratic Regulator

In this section a Linear Quadratic Regulator for a continuous-time linear system is defined [21]. A continuous-time linear system, where $t \in [t_0, t_1]$ can be defined by:

$$\dot{x} = Ax + Bu \qquad x(0) = x_0. \tag{3.37}$$

With the cost function defined as:

$$J = \frac{1}{2}\int_0^\infty (x^T Q_x x + u^T Q_u u)dt = \frac{1}{2}\int_0^\infty V(x,u)dt \tag{3.38}$$

where $Q_x \in \mathbb{R}^{n\times n}$, $Q_x^T = Q_x$, $Q_u \in \mathbb{R}^{p\times p}, Q_u^T = Q_u > 0$ represent the constant weights. By minimizing $J$ over an infinite window poles can be optimally allocated. The optimization problem:

$$J^* = \min_u J. \tag{3.39}$$

Subjected to the system dynamics and boundary conditions with the input variable u is then solved by following Langrangian:

$$L(x,u,\lambda) = V + \lambda^T(Ax + Bu - \dot{x}) = x^T Q_x x + u^t Q_u u + \lambda^T(Ax + Bu - \dot{x}) \tag{3.40}$$

where $\lambda$ is an auxiliary variable. To derive the optimum value, the Euler-Lagrange equation is applied which in state space gives:

$$\begin{bmatrix} \dot{x} \\ \dot{\lambda} \end{bmatrix} = \begin{bmatrix} A & -BQ_u^{-1}B^T \\ -Q_x & -A^T \end{bmatrix} \begin{bmatrix} x \\ \lambda. \end{bmatrix} \tag{3.41}$$

The optimal input $u*$ expressed as a function of the co-state $\lambda$ and is desired to be a function of the state $x$. Therefore the optimal $\lambda$ is assumed to be:

$$\lambda^* = Px^*. \tag{3.42}$$

Then the Euler-Langrange equation can be rewritten as:

$$\dot{x} = Ax = -BQ_u^{-1}B^T Px\dot{\lambda} = \dot{P}x^* + P[Ax - BQ_u^{-1}BPx]. \tag{3.43}$$

Rearranging the terms gives:

$$(\dot{P} + PA + A^T P + Q_x - PBQ_u^{-1}B^T P)x = 0. \tag{3.44}$$

To guarantee that the equation holds for any arbitrary states $x$, the time varying transformation matrix $P$ has to satisfy the following nonlinear matrix differential equation (differential Riccati equation (DRE)):

$$\dot{P} + PA + A^T P + Q_x = PBQ_u^{-1}B^T P. \tag{3.45}$$

Considering the stationary solution of the DRE we obtain an algebraic matrix equation:

$$\bar{P}a + A^T\bar{P} + Q_x = \bar{P}BQ_u^{-1}B^T\bar{P} \tag{3.46}$$

where $\bar{P} = lim_{t\to\infty}P$ denotes the steady-state value.
The optimal control is then given by:

$$u^* = -Q_u^{-1}B^T\bar{P}x^* = -\bar{K}x^* \tag{3.47}$$

where $\bar{K}$ is the LQ gain. The optimal feedback equation is then given by:

$$\dot{x}^* = (A - B\bar{K})x^*, \quad x(0) = x_0. \tag{3.48}$$

### 3.7.2 Linear quadratic regulator for lane centering

Linear quadratic regulator is commonly used for lane centering [22]. Furthermore a one track model (chapter: 3.2.4) or bicycle model with neglected roll motion is acknowledged to be sufficiently accurate for the purpose of control design.

The LQR problem utilize desired yaw rate and a magnetic road reference system for lane following control [22]. The magnetic road reference system represents the future path of the road which can be replaced with a lane center polynomial given by host sensors and tracking.

The state vector $x = [e_y \quad \dot{e}_y \quad e_\psi \quad \dot{e}_\psi]$ for state feedback control for lane centering, where $e_y$ is lateral offset, $\dot{e}_y$ is lateral rate, $e_\psi$ is yaw rate and $\dot{e}_\psi$ is yaw error rate [23].

By letting $\delta$ be the steering angle input and utilizing the three degree lane center polynomial defined as:

$$y = a_0 + a_1x + a_2x^2 + a_3x^3 \tag{3.49}$$

the desired yaw rate $\dot{\psi}_{des}$ can be determined by road curvature given by:

$$\frac{1}{R} \approx \frac{1}{2 \cdot a_2} \tag{3.50}$$

and vehicle speed $v_x$ which gives:

$$\dot{\psi}_{des} = \frac{v_x}{R} \approx \frac{v_x}{2 \cdot a_2}. \tag{3.51}$$

Given the three degree lane polynomial, measured yaw rate $\dot{\psi}$, velocity $v_x$ and integrating the desired yaw rate all states of the state vector become observable.

- **Lateral offset**, $e_y = a_0$
- **Lateral rate**, $\dot{e}_y = v_x \cdot (tan^{-1}a_1)$
- **Yaw error**, $e_\psi = \tan^{-1} a_1$
- **Yaw rate error**, $\dot{e}_\psi = \dot{\psi} - \frac{v_x}{2a_2}$

Utilizing the one track model defined in chapter: 3.2.4 the vehicle dynamics for lane centering becomes [23]:

$$\begin{bmatrix} \dot{e}_y \\ \ddot{e}_y \\ \dot{e}_\psi \\ \ddot{e}_\psi \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{C_f+C_r}{mv_x} & \frac{C_f+C_r}{m} & \frac{l_fC_f+l_rC_r}{mv_x} \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{l_fC_f+l_rC_r}{J\cdot v_x} & \frac{l_fC_f-l_rC_r}{J} & \frac{l_f^2C_f+l_r^2C_r}{J\cdot v_x} \end{bmatrix}}_{A} \begin{bmatrix} e_y \\ \dot{e}_y \\ e_\psi \\ \dot{e}_\psi \end{bmatrix} + \underbrace{\begin{bmatrix} 0 \\ \frac{C_f}{m} \\ 0 \\ \frac{l_f \cdot c_f}{J} \end{bmatrix}}_{B_1} \delta + \underbrace{\begin{bmatrix} 0 \\ -\frac{C_f \cdot l_f - C_R \cdot l_r^2}{m \cdot v_x} \\ 0 \\ -\frac{C_f \cdot l_f^2 - C_R \cdot l_r^2}{J \cdot v_x} \end{bmatrix}}_{B_2} \dot{\psi}_{des} \tag{3.52}$$

Defining the state feedback law:

$$\delta = -Kx. \tag{3.53}$$

The eigenvalues of the closed loop matrix $A - B_1K$ can then be optimally placed by the state feedback (LQ gain $\bar{K}$) calculated according to chapter: 3.7.1. The closed loop system using this state feedback controller becomes:

$$\dot{x} = (A - B_1\bar{K})x + B_2\ \dot{\psi}_{des}. \tag{3.54}$$

### 3.7.3 Sliding mode control for longitudinal control

For longitudinal control it is convenient to have the requested acceleration as control output $u$ [24]. Measurement of the range $x_1$ and range rate $x_2$ between the host and the target vehicle is required in order to use the proposed controller, and also the acceleration of the target $\ddot{x}_{target}$. The task of the controller is to control the desired distance $d_0$ to the target vehicle. This is achieved by choosing the control parameters based on the range in relation to $d_0$ and range rate in relation to the speed thresholds ($v_{fallback}$ and $v_{catchup}$). This gives four zones, each one corresponding to one set of control parameters. The control law is given by:

$$\begin{aligned} \sigma(x) &= \alpha_1 x_1 + \alpha_2 x_2 \\ \ddot{x} = u &= -\frac{1}{\alpha_2}(-\alpha_1 x_1 - \alpha_2 \ddot{x}_{target} + \mu \cdot sign(\sigma(x))) \end{aligned} \tag{3.55}$$

where $\alpha_1$, $\alpha_2$ and $\mu$ are the control parameters, chosen depending on range and range rate.

### 3.7.4 Constant time gap controller for longitudinal control

The constant time gap control is a method developed for autonomous control of road vehicles which ensures stability [23]. The controller has two modes, speed control and spacing control (or headway control). The control law of the spacing controller is defined by:

$$\ddot{x} = u = -\frac{1}{h}(\dot{x} + \lambda\delta) \tag{3.56}$$

where $h$ is the time gap to the preceding vehicle and

$$\delta = x + h\dot{x}. \tag{3.57}$$

As long as $\lambda > 0$, $\delta$ is expected to converge to zero.

The switches between the two control modes is defined in figure: 3.8. When the host vehicle is driving faster than the preceding vehicle the range rate is negative and the gap between the vehicles is decreasing. When the gap is small enough headway control is activated. Similarly, if the preceding vehicle is speeding up the speed control is activated when the red diagonal line is passed.

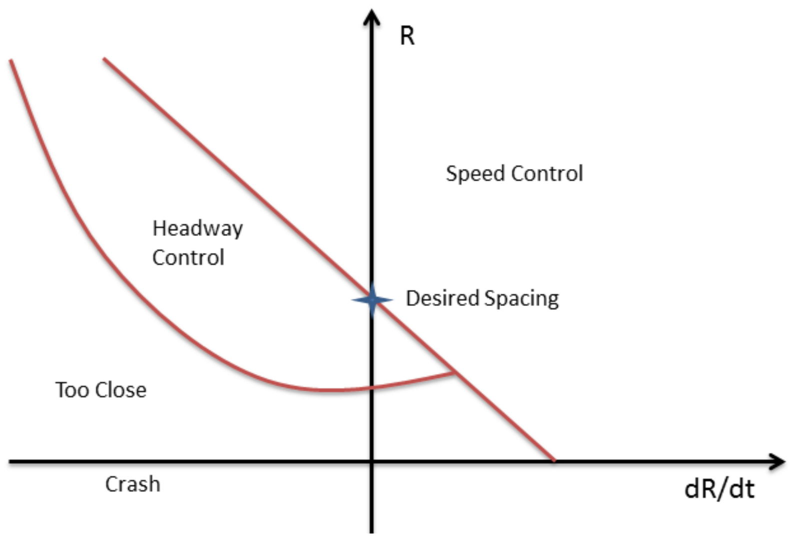



**Figure 3.8:** Range (R) - Range rate (dR/dt), diagram for constant time gap control policy.

# 4

# Design of trajectory algorithm

In this chapter the method of the trajectory algorithm is described. The main idea of the algorithm is described in the pseudocode below and in the following subchapters each part of the method is described in detail. The algorithm is developed in Matlab then autocoded to C++ and implemented in an offline resim and an embedded software in the host vehicle.

```
Input: sensor data (4.1), host data, trajectory
    If on highway
        run Road model (4.2) estimation
        run Target selection (4.3)
        If target selected
            If target not initialized
                If first instance of target
                    Add current target information
                else
                    Motion update (4.4.1) previous collected target information
                    Add current target information
            If target initialized
                Motion update previous collected target information
                Add current target information
                Filter collected radar trail(4.4.2)
                Path planning(4.5)
                return Trajectory
        else
            Reset Trajectory
    else
        Reset Trajectory
return: Trajectory
```

## 4.1 Sensor information

The sensor information utilized are from the front radar unit and from the host vehicle sensors. The radar data is fused and given as objects, tracklets and barrier estimations. No further object or tracklet tracking are considered.

### 4.1.1 Host vehicle sensors

The availible sensors of the host vehicle are:

- **Current host speed**, $v_{host}$, is measured with high accuracy.
- **Current host yaw rate**, $\dot{\psi}_{host}$, is measured with high accuracy.
- **Current roll rate**, $\dot{\phi}_{host}$, is measured with low accuracy.

Both speed and yaw rate are considered as control inputs in the road estimation process. The measurement of roll rate has a constant offset, this measurement is used to find the roll angle of the host vehicle which is used to validate the state update in the road estimation process. The average offset is deducted in order to make the roll rate measurement more reliable.

### 4.1.2 Barrier/guardrail observations

The reported observation of the barrier/guardrail is an estimation of the barrier/guardrail. This estimation is made from stationary detections along the road reported by the radar. If a barrier/guardrail is detected this is reported as a third degree polynomial. The lateral position $y$ depends on four coefficients ($a_k$, where k = [0, 1, 2, 3]) and the longitudinal distance $x$ from the car, it is calculated as:

$$y = a_0 + a_1 x + a_2 x^2 + a_3 x^3. \tag{4.1}$$

One barrier/guardrail on each side can be detected. The measurement comes with a confidence estimation, which is either 0 or 1, and a valid range. In figure: 4.1 the guardrails are presented as an overlay in the video.

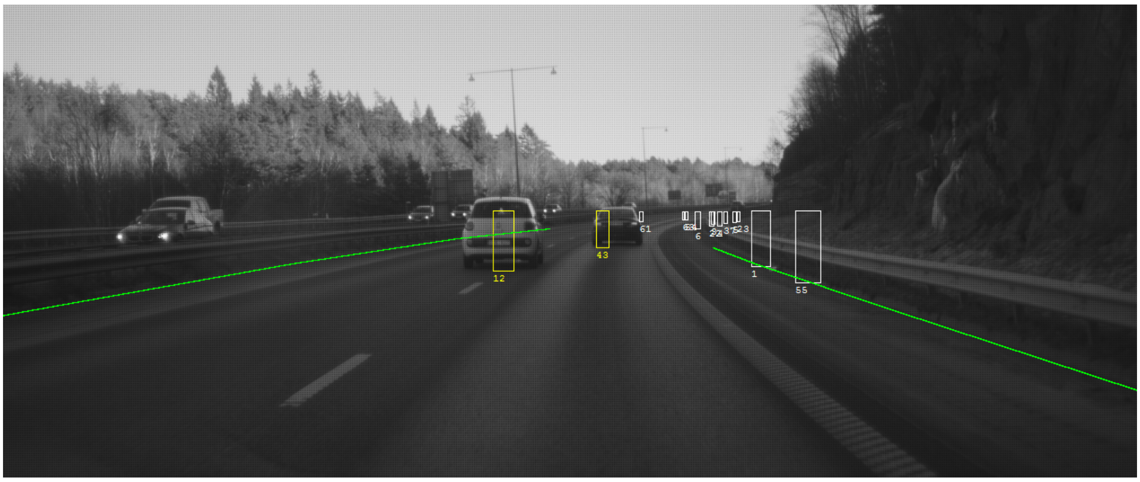


**Figure 4.1:** Detected guardrails, marked as green lines.

### 4.1.3 Receding vehicle observations

The observed receding vehicles are included in the outcome of the radar-only fusion tracking. The fusion tracker filters the moving detections and tracklets from the radar and tracklet tracker. Each vehicle reported has the following properties:

- Position
- Velocity
- Acceleration
- Speed Over Ground
- Heading $\phi$
- Heading Rate $\dot{\phi}$
- Standard deviation of
  - Position measurement
  - Speed Over Ground measurement
  - Heading measurement

All measurements are expressed in the SAE coordinate frame, presented in figure: 3.1.

## 4.2 Road model

The road model was developed for longitudinal control and therefore based on host lateral motion [17]. The modifications from the road model is that it only utilizes radar input. When removing the vision input some adjustments in the implementation needs to be done, especially regarding lane estimation when lane markers are lost. The lane width is assumed to be a function of speed in order to cover for the missing data. The state vector of the road model is:

$$x = \begin{bmatrix} \phi_k & C_{0,k} & C_{1,k} & \dots & C_{10,k} \end{bmatrix}^T . \tag{4.2}$$

Beside the modification of the lane estimation there are some other changes implemented in the road model, these include:

- When the guardrail outlier tracker or complementary guardrail tracker classifies guardrail as an outlier, a simplified tracker based on the previous road prediction is used.
- Roll angle measurements are used to limit maximum curvature in the road model.
- Functions for classifying receding vehicles as outliers when they are:
  - Identified to entering or exiting highway.
  - Identified as ghost vehicles outside the road barriers.
- Increase noise if in radar shadow by vehicle in front.
- Reduce host motion dependency.

- If target vehicle is present it is removed from the update step of receding vehicles.

The output of the road model is a third degree polynomial:

$$y(x) = a_1 \cdot x + a_2 \cdot x^2 + a_3 \cdot x^3 \tag{4.3}$$

that estimates the lateral position ($y$) of the road for a given longitudinal distance ($x$). The polynomial is limited by a maximum range depending on the quality of the measurements. By taking the derivative of the output polynomial:

$$\dot{y}(x) = a_1 + 2 \cdot a_2 \cdot x + 3 \cdot a_3 \cdot x^2 \tag{4.4}$$

the heading of the road for a given longitudinal distance can be estimated. Assuming that the derivative of the road is small, the heading of the road can be approximated as:

$$\phi(x) = tan^{-1}(\dot{y}(x)) \approx \dot{y}(x). \tag{4.5}$$

### 4.2.1 Guardrail outlier

Guardrails are considered an outlier if any threshold value is violated by the reported polynomial. The reported polynomial needs to have an offset ($a_0$) large enough, so intersection with the host is avoided. Since highway driving entails low curvatures the coefficients $a_1$, $a_2$ and $a_3$ are restricted to be small, and the range of the reported guardrail shall be large to make sure that the reported guardrail follows the road.

### 4.2.2 Complementary guardrail tracker

If a guardrail is considered as outlier or not reported at all, a complementary guardrail tracker tries to find a new guardrail. The complementary guardrail tracker utilizes the mid-range scan, which gives more accurate radar measurement up to $50m$. The complementary guardrail tracker also utilizes the previously calculated road estimation. The guardrail is then calculated in the following way:

- A set of potential guardrail tracklets are first found by identifying tracklet IDs for all tracklets classified as stationary and to the left side of host if the left guardrail is calculated and to the right of host if the right guardrail is calculated.
- In order to find potential starting points for the guardrail, the new set of tracklet IDs are looped over and the closest longitudinal tracklet that have at least a lateral offset of $1.5m$ is saved. Since highways often have two guardrails separating the two directions of the road the closest longitudinal point could belong to the guardrail of the other side. This is prevented by the possibility to save the two closest longitudinal points if they have a lateral offset of 2m between each other.

- Based on the found potential starting point the previous road estimation is then utilized in order to find tracklets that could potentially belong to the guardrail. If two potential starting points are identified the algorithm starts with the tracklet that has the lowest lateral offset to the host vehicle. Looping over the set of potential guardrail tracklets defined as all tracklets that have a lateral offset less than $1m$ from the previous road estimation (lateral shift to start in the found potential starting point) at the longitudinal distance from host of each tracklet saved.
- If more than 5 tracklets are classified as belonging to the guardrail a Kalman filtration is done using a constant velocity model as motion model. This generates a smooth set of tracklets that are used in a third degree polynomial fitting for calculating the new set of coefficients describing the guardrail. The Kalman filtering of stationary tracklets is described in detail in chapter: 4.2.3.

### 4.2.3 Kalman filtering of stationary tracklets

The Kalman filter utilizes a motion update step in the prediction and since tracklets from guardrails are stationary some adjustments to the data is required. The first step is to sort the tracklets to have the closest longitudinal tracklet first and the longitudinal tracklet farthest away last. The motion model used is a constant velocity model presented in chapter: 3.2.1 with the state vector $\mathbf{X} = [x \quad y \quad \dot{x} \quad \dot{y}]$. The state transition matrix $A_{k-1}$ is defined as:

$$A_{k-1} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.6}$$

and model noise matrix $Q_{k-1}$ is defined as:

$$Q_{k-1} = \begin{bmatrix} \frac{\Delta t^3}{3} & 0 & \frac{\Delta t^2}{2} & 0 \\ 0 & \frac{\Delta t^3}{3} & 0 & \frac{\Delta t^2}{2} \\ \frac{\Delta t^2}{2} & 0 & \Delta t & 0 \\ 0 & \frac{\Delta t^2}{2} & 0 & \Delta t \end{bmatrix} Qc \tag{4.7}$$

where $Q_c$ is the noise covariance found by tuning.

The initial state is $\mathbf{X_0} = [x_0 \quad y_0 \quad \dot{x}_0 \quad \dot{y}_0]$, $x_0$ and $y_0$ is given by longitudinal and lateral position of the longitudinal closest tracklet. $\dot{x}$ is set to the host velocity and $\dot{y}$ is set to be zero, these assumptions are done since guardrails usually are relatively straight. Since the filtration is done over stationary detections a simulated sampling time $\Delta t$ is calculated during each iteration. The simulated sampling time is calculated by taking the longitudinal difference between the previous and current tracklet divided by the host velocity:

$$\Delta t = \frac{x_{k+1} - x_k}{v_{host}}. \tag{4.8}$$

Measurement model matrix $H_k$ is a constant matrix defined as:

$$H_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}. \tag{4.9}$$

Measurement noise matrix $R_k$ is defined as:

$$R_k = \begin{bmatrix} R_x & 0 \\ 0 & R_y \end{bmatrix} \tag{4.10}$$

where $R_i$ (i $= x,\ y$) is then measurement uncertainties reported by the tracklet fusion.
The Kalman algorithm is derived according to chapter: 3.4.3 where the final algorithm is:

$$\begin{aligned} \hat{x}_{k|k-1} &= A_{k-1}\hat{x}_{k-1|k-1} + q_{k-1} \\ P_{k|k-1} &= A_{k-1}P_{k-1|k-1}A_{k-1}^T + Q_{k-1} \end{aligned} \tag{4.11}$$

$$\begin{aligned} S_k &= H_k P_{k|k-1} H_k^T + R_k \\ v_k &= y_k - H_k \hat{x}_{k|k-1} \\ K_k &= P_{k|k-1} H_k^T S_k^{-1} \end{aligned} \tag{4.12}$$

$$\begin{aligned} \hat{x}_{k|k} &= \hat{x}_{k|k-1} + K_k v_k \\ P_{k|k} &= P_{k|k-1} + K_k S_k K_k^T \end{aligned} \tag{4.13}$$

### 4.2.4 Roll angle utilization

In the prestudy it was concluded that roads are banked (see section: 2.3.2), and the angle of road banking combined with the speed limit of the road determine the maximum allowed curvature of the road. In order to be able to measure the banking angle of the road, it is assumed that the roll angle of the car is equal to the banking angle of the road.

The roll angle of the host vehicle combined with the host vehicle speed are used to find the maximum allowed curvature. If a state update results in a curvature state that is larger than the maximum allowed curvature, the measurement noise is increased by 20% and the calculations are repeated.

### 4.2.5 Identification of vehicles entering and exiting highway

If a receding vehicle is driving on the ramp of the highway a measurement update based on its position and heading of the vehicle will decrease the precision of the road estimation due to the fact that the vehicle is not driving on the highway as expected. This is illustrated in figure: 4.2 where a white van is entering the highway from the right side.


**Figure 4.2:** Receding vehicle entering the highway in front of the host vehicle.

In order to detect if a vehicle is entering or exiting the highway the vehicles lateral acceleration and lateral velocity are checked. A vehicle entering the highway has a high lateral velocity while a vehicle exiting the highway has a high lateral acceleration. If the acceleration and velocity are not considered reasonable, the vehicle is not used to update the road estimation.

### 4.2.6 Identification of ghost vehicles

Ghost targets appear when a detection of a vehicle is reflected in an other surface on the way back to the host vehicle. This can for example be the flat surface in the guardrail, which induces an error in both angle and range which changes the properties of the perceived vehicle. This results in a perceived vehicle driving on the side of the road with similar speed as the true vehicle, but in another position and with a different heading. An example of a ghost vehicle is shown in figure: 4.3, where the perceived vehicle is outside the guardrail.


**Figure 4.3:** Vehicle outside of guardrail identified as a ghost target.

If a vehicle is completely inside the road barriers it is most likely a true vehicle, if not the vehicle is considered to be a ghost vehicle and is therefore not used to update the road estimation.

### 4.2.7 Identification of vehicles in radar shadow

If the detection is reflected in the road surface instead of the guardrail, a vehicle can be perceived even when it is out of sight. This is exemplified in figure: 4.4, where a vehicle is hidden behind the vehicle just in front of the host vehicle. When an object is not visible the number of detections is generally low and the reliability of the measurement is therefore also low.


**Figure 4.4:** Receding vehicle identified to be in radar shadow from another receding vehicle.

If a vehicle is not visible to the host the reported noise of the measurement is increased.

## 4.3 Target selection

The target selection includes two phases, when target is active and when no target is active.

When no target is active the target selection method can utilize more information about the host vehicle since no trajectory is generated and the driver has control of the vehicle. Since the driver is in control of the vehicle the condition that the host vehicle is driving along the lane is assumed to be true.

When the target is active the host vehicle is driving according to the reference generated from the target, and for that reason it is not possible to assume that the host is driving along the true lane. The target heading is now compared to the prediction of the road in order to detect undesirable driving actions taken by the target vehicle, such as lane changes or highway exits.

In the pseudo code below the main idea of the target selection is summarized:

```
If no target is active
    Loop over all receding vehicles
        If vehicle is within longitudinal interest
            Calculate estimated road lateral position at vehicle
                If target is within predicted lane
                    Vehicle is considered as target
if target is active
    Compare heading of road estimation at target with heading of target
    If heading difference is within threshold
        If target lat and long acceleration is within threshold
            Keep vehicle as target
            Calculate target confidence
            Increase target tracking age
        else
            Reset target
    else
        Reset target
```

### 4.3.1 Identification of potential target in host lane

In order to find a potential target vehicle it is assumed that the host vehicle is driving in the middle of its own lane. To check if the host vehicle is driving in its lane along the road and not changing lane or leaving the highway, the target selection algorithm is also limited by a maximum allowed host yaw rate. This will not remove all situations when host vehicle is not driving in the middle of its own lane, but it removes some of the times when the host could be changing lane. Since the target identification is based on that the host vehicle is driving in the middle of the lane this assumption has to be done.

At each time instance if no target is identified the algorithm is looping over all receding vehicles and if a vehicle is within maximum allowed longitudinal target range and not been considered as a target vehicle during the last $3s$, the vehicle is considered as a potential target. The predicted lateral position and heading of the road at the vehicle ($x_{vehicle}$) is then estimated. The lateral position is calculated by equation: 4.3 and the heading by equation: 4.4. The vehicle is then considered a possible target if the following requirements are satisfied:

- Difference between the predicted lateral position of the road at the vehicle ($y_{road}$) and the measured lateral position of the vehicle ($y_{vehicle}$) is within the limit.
- Difference between predicted heading of the road at the vehicle ($\phi_{road}$) and the measured heading of the vehicle ($\phi_{vehicle}$) is within the limit.
- No other vehicle closer that satisfies the requirements.

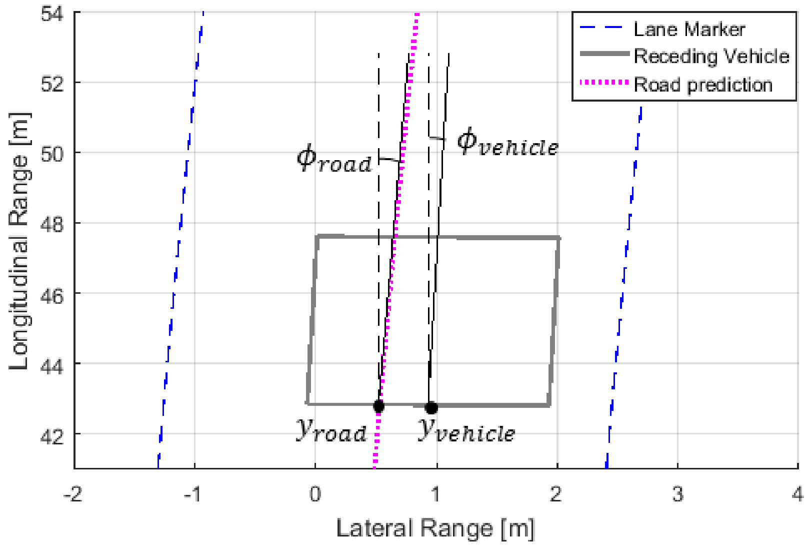



**Figure 4.5:** Definition of the lateral positions $y_{road}$ and $y_{vehicle}$ and the headings $\phi_{road}$ and $\phi_{vehcile}$ used in the target selection.

If the requirements are satisfied and the vehicle is considered as a potential target the object ID, lateral position, longitudinal position, heading and speed of the target are saved and the track age of the target is set to one.

### 4.3.2 Verification of target driving in lane

When a target is identified the lateral position of the host vehicle can no longer be utilized in order to verify that the target vehicle drives according to the defined requirements. Instead the heading of the target is compared with heading of the road prediction and heading of other receding vehicles in order to verify that the target keeps driving in the host lane.

If one or more of the following requirements are fulfilled the target is still considered a target.

- The lowest heading difference between target and any other receding vehicle within longitudinal distance of $100m$ is within the limit.
- The heading difference between the target and the road prediction at the distance of the target, at the current time instance or in the five previous time instances are within limit.

The target vehicle still needs to be inside the maximum allowed longitudinal distance in order to still be considered a target. If none of the requirements are fulfilled for three consecutive time instances, the target is no longer considered as driving in the lane or in a suitable way and is released as target.

### 4.3.3 Target confidence

Target confidence is calculated based on the heading difference between the measurement of targets heading and the heading of road estimation at the longitudinal distance of the target. The heading of the road estimation is given by the derivative of the road estimation polynomial (equation: 4.4).

The highest confidence is defined as one and the lowest confidence is zero. The confidence is calculated based on the six last heading differences between the target vehicle and road estimation. The confidence is calculated as the percentage of how many of the six last heading differences that meet the requirement. If all six heading differences meets the requirements the confidence is one, if none of the heading difference meets the requirements the confidence is set to zero. Zero confidence is allowed for three consecutive time instances according to the target selection algorithm (chapter: 4.3.2) before the target is released.

## 4.4 Target trail

The basic idea of the target trail includes:

- Saving the last 150 detections of the target or as many as possible if target is newly detected, saved detections include: lateral position, longitudinal position, heading, heading rate
- Motion update the saved target information based on host motion.
- When host vehicle reaches the first saved detection of target the initializing phase is over and target trail is possible to generate
  - Kalman filter the detections of the target trail.
  - Rauch-Tung-Striebel smoothing the filtered data.
  - Calculate third degree polynomial of target vehicle trail.

### 4.4.1 Motion update of target information

In order to save the detections of the target trail, the detections are considered fixed points on the road that the host vehicle drives over. In every time instance the detections need to be moved according to the host vehicle motion. This update is done according to:

$$
\begin{aligned}
x_{k+1} &= x_k - v_k \cdot cos\left(\frac{\dot{\psi}_k \Delta_t}{2}\right) \\
y_{k+1} &= y_k - v_k \cdot sin\left(\frac{\dot{\psi}_k \Delta t}{2}\right) \Delta t - x_k \cdot sin\left(\frac{\dot{\psi}_k \Delta t}{2}\right)
\end{aligned}
\tag{4.14}
$$

where $v_k$ is the host speed, $\dot{\psi}_k$ is the host yaw rate and $\Delta t$ is the time from last sample.

### 4.4.2 Kalman filtering of target trail

The stored trail is filtered by an extended Kalman filter with coordinated turn motion model. The state vector $x$ and coordinated turn motion model are defined as:

$$x = \begin{bmatrix} p_x \\ p_y \\ v \\ \phi \\ \omega \end{bmatrix} \qquad \dot{x} = \begin{bmatrix} v \cdot cos(\phi) \\ v \cdot sin(\phi) \\ a \\ \omega \\ \alpha \end{bmatrix}. \tag{4.15}$$

The coordinated turn motion model discretization ($f_{k-1}$) is [25]:

$$f_{k-1} = \begin{bmatrix} p_x + \frac{2v}{\omega} sin(\frac{\omega\Delta t}{2}) cos(\phi + \frac{\omega\Delta t}{2}) \\ p_y + \frac{2v}{\omega} cos(\frac{\omega\Delta t}{2}) sin(\phi + \frac{\omega\Delta t}{2}) \\ v \\ \phi + \omega\Delta t \\ \omega \end{bmatrix} \tag{4.16}$$

since $\phi$ and $\omega$ are small during highway driving and the sampling time $\Delta t$ also is small, assuming small angles ($sin(\theta) = \theta$ and $cos(\theta) = 1$) is reasonable and the updated discretization ($f^*_{k-1}$) is then:

$$f^*_{k-1} = \begin{bmatrix} p_x + v\Delta t \\ p_y + v\Delta t\phi + v\omega\Delta t^2 \\ v \\ \phi + \omega\Delta t \\ \omega \end{bmatrix}. \tag{4.17}$$

The updated discretization function $f^*_{k-1}$ depends on the state vector and is non-linear. According to the extended Kalman filter the non-linearities is eliminated by linearization in each time instance around the previous estimated state $\hat{x}_{k-1}$. The Jacobian ($f'_{k-1}$) of $f^*_p(x)$ is then: presented in:

$$f'_{k-1} = \begin{bmatrix} 1 & 0 & \Delta t & 0 & 0 \\ 0 & 1 & \Delta t\phi_{k-1} + \omega_{k-1}\Delta t^2 & v_{k-1}\Delta t & v_{k-1}\Delta t^2 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{4.18}$$

Transition covariance matrix $Q_{k-1}$ is defined as:

$$Q_{k-1} = G\sigma G^T \tag{4.19}$$

and is calculated in each iteration and since the component:

$$G = \begin{bmatrix} \frac{\Delta t^2 cos(\phi_{k-1})}{2} & 0 \\ \frac{\Delta t^2 sin(\phi_{k-1})}{2} & 1 \\ \Delta t & 0 \\ 0 & \frac{\Delta t^2}{2} \\ 0 & \Delta t \end{bmatrix} \tag{4.20}$$

depends on the state vector and is nonlinear [25]. $\sigma$ is defined as:

$$\sigma = \begin{bmatrix} \sigma_v^2 & 0 \\ 0 & \sigma_\omega^2 \end{bmatrix} \tag{4.21}$$

where $\sigma_\omega$ and $\sigma_v$ are parameters for tuning of the filter.

Measurement model is linear and matrix $H_k$ is a constant matrix defined as:

$$H_k = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{4.22}$$

The measurement noise matrix $R_k$ is constant and defined as:

$$R_k = \begin{bmatrix} R_x & 0 & 0 & 0 \\ 0 & R_y & 0 & 0 \\ 0 & 0 & R_v & 0 \\ 0 & 0 & 0 & R_\omega \end{bmatrix} \tag{4.23}$$

where $R_i$ (i = $x$, $y$, $v$, $\omega$) is the measurement uncertainties reported by the fusion output.

Finally the prediction step can be formulated as:

$$\begin{aligned} \hat{x}_{k|k-1} &= f^*(\hat{x}_{k-1|k-1}) \\ P_{k|k-1} &= f'(\hat{x}_{k-1|k-1})P_{k-1|k-1}f'(\hat{x}_{k-1|k-1})^T + Q_{k-1} \end{aligned} \tag{4.24}$$

and the update step as:

$$\begin{aligned} \hat{x}_{k|k} &= \hat{x}_{k|k-1} + K_k(y_k - h(\hat{x}_{k|k-1})) \\ P_{k|k} &= P_{k|k-1} - K_k S_k K_k^T \\ S_k &= H(\hat{x}_{k|k-1})P_{k|k-1}H(\hat{x}_{k|k-1})^T + R_k \\ K_k &= P_{k|k-1}H(\hat{x}_{k|k-1})^T S_k^{-1}. \end{aligned} \tag{4.25}$$

### 4.4.3 Extended Rauch-Tung-Stribel smoother of target trail

In order to smooth the extended Kalman filtered data the non-linear backward smoothing method extended Rauch-Tung-Stribel is used. The radar trail data of length $k = K$ and $\hat{x}_{k|k}$, $P_{k|k}$, $\hat{x}_{k+1|k}$ and $P_{k+1|k}$ are stored in each iteration. The extended Rauch-Tung-Stribel algorithm is presented in section 3.4.6. The backward smoothing step starting at $k = K - 1$ is then given by:

$$\begin{aligned} \hat{x}^-_{k+1|k} &= f(\hat{x}_{k|k}) \\ P^-_{k+1|k} &= F(\hat{x}_{k|k})P_{k|k}F^T(\hat{x}_{k|k}) + Q_k \\ G_k &= P_{k|k}F^T(\hat{x}_{k|k})(P^-_{k+1|k})^{-1} \\ \hat{x}^s_{k|K} &= \hat{x}_{k|k} + G_k(\hat{x}^s_{k+1|K} - \hat{x}^-_{k+1|k}) \\ P^s_{k|K} &= P_{k|k} + G_k(P^s_{k+1|K} - P^-_{k+1})G_k^T. \end{aligned} \tag{4.26}$$

## 4.5 Path planning and Trajectory design

For lateral control no further path planning is done besides the filtering and smoothing of the radar trail of the target vehicle. The filtered and smoothed radar trail is generated with the help of polynomial fitting as a third degree polynomial. The polynomial works as the lateral reference for the host vehicle and would be able to be applied to a regular lane centering controller that utilizes a lane centering polynomial. Since no elevation estimation of stationary detections is given they cannot be used in a path planning algorithm. When no elevation estimation is available an overhead sign above the road and a stationary vehicle on the road could be reported in the same way, therefore stationary detections cannot be used for path planning.

For longitudinal control only the selected distance between the host vehicle and target vehicle is needed for the design of the trajectory. This utilizes the information about the longitudinal position, velocity and acceleration of the target vehicle.

The output of the trajectory algorithm then includes:

- Coefficients for a third degree polynomial describing the lateral future path
- Selected longitudinal distance between the host and the target vehicle
- Distance to target vehicle
- Longitudinal velocity of target
- Longitudinal acceleration of target

Where the polynomial is used for lateral control and the desired distance and the information of the target is used for longitudinal control.

# 5

# Result

In this chapter the result of the algorithm is presented. In figure: 5.1 and 5.2 the output of the trajectory algorithm is illustrated. The data used for the result is collected from a public highway.

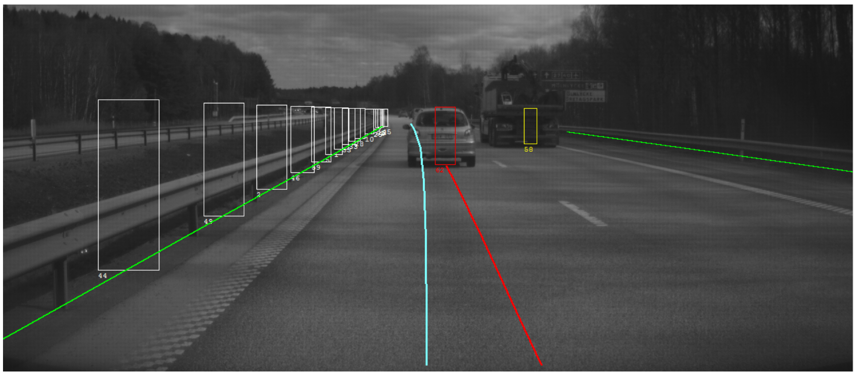


**Figure 5.1:** Main video view in DVtool, illustrates the road prediction in blue, target trail in red and guardrails in green.

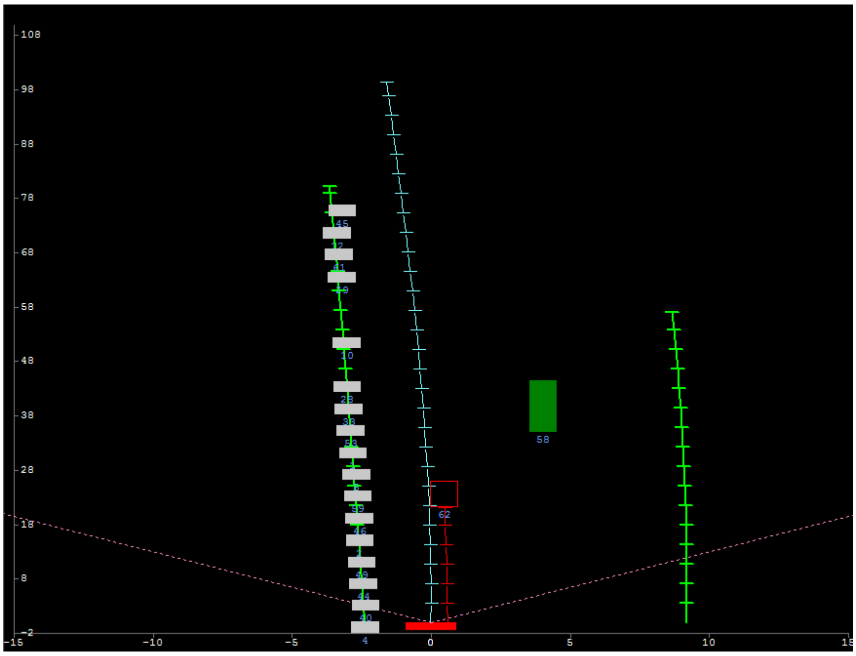


**Figure 5.2:** Plan view in DVtool, illustrates the road prediction in blue, target trail in red and guardrails in green.

## 5.1 Evaluation method

The trajectory algorithm consist of different parts which are all independent of each other and important for the final output of the trajectory algorithm. Therefore the result of the trajectory algorithm is analyzed both part by part and the trajectory output. All results are from the pre-defined test route in chapter: 2.5.1.

Two important measurements of the result are:

- **Availability**, measured in percentage of time.
- **Performance**, measured as the root-mean-square error (RMSE) of the estimation compared to the ground truth data.

The RMSE is defined by:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2} \tag{5.1}$$

where $N$ is the number of samples, $\hat{y}$ is the estimation and $y$ is the ground truth data. The ground truth data used to evaluate the performance are radar-vision fusion output and a road geometry reference.

- **The radar-vision fusion output** in the cads4 project by Delphi was developed in collaboration with Mobileye. This fusion output is utilized by Volvo Cars in the SPA-platform for both lateral and longitudinal control.
- **The road geometry reference** represents the true path of the road. An accurate method to do this is to utilize GPS and measure the position at each time step. Similar to the previous master thesis at Delphi [17] no GPS data was available in the hardware setup. Instead a road geometry reference is calculated based on the measured speed and yaw rate of the host vehicle. The road geometry estimates the movement in a global coordinate system. This method is know as dead-reckoning and calculated in the same way as in the in the previous master thesis at Delphi [17]. The calculation of the road geometry reference is derived in chapter: 5.1.1.

### 5.1.1 Generation of ground truth

The movement of the host vehicle in the vehicle local coordinate system is defined by:

$$\begin{aligned}\Delta x_{host,k} &= v_k \cos\left(\frac{\dot{\psi}_k dt}{2}\right) dt \\ \Delta y_{host,k} &= v_k \sin\left(\frac{\dot{\psi}_k dt}{2}\right) dt\end{aligned} \tag{5.2}$$

where $v_k$ is the velocity of the host vehicle and $\dot{\psi}_k$ is the yaw rate and $dt$ is the sampling time.

The movement in the global coordinate system is defined by:

$$\begin{bmatrix}\Delta x_k \\ \Delta y_k\end{bmatrix} = \begin{bmatrix}\cos(\psi_k + \alpha_{slip,k}) & -\sin(\psi_k + \alpha_{slip,k}) \\ \sin(\psi_k + \alpha_{slip,k}) & \sin(\psi_k + \alpha_{slip,k})\end{bmatrix} \begin{bmatrix}\Delta x_{host,k} \\ \Delta y_{host,k}\end{bmatrix} \tag{5.3}$$

and the heading $\psi$ is defined by

$$\psi_k = \psi_{k-1} + \dot{\psi}_k dt. \tag{5.4}$$

## 5.2 Road model

The performance of the road model is determined by evaluating the estimated road prediction, the performance of the guardrail estimation and the outlier functions of receding vehicles.

The data used in the analysis of the road model is from two laps on the pre-defined test route. Since host lane change impact the road prediction and ground truth road geometry reference no lane change was made by the host vehicle during the collection of the data. On the first lap the host vehicle drove in the right lane and on the second lap the host vehicle drove in the left lane. The total length of the dataset is 44min long and covers 70km of driving.

### 5.2.1 Reported guardrails and complementary guardrail tracker

The outcome of the complementary guardrail tracker is shown in figure: 5.3. The guardrail in black is the reported guardrail by the Delphi fusion tracker, the guardrail has been considered an outlier and does not meets the requirement for guardrail filter in the road model. The guardrail in blue is the complementary calculated guardrail and the set of tracklets estimated to belong to the guardrail are marked with circles.

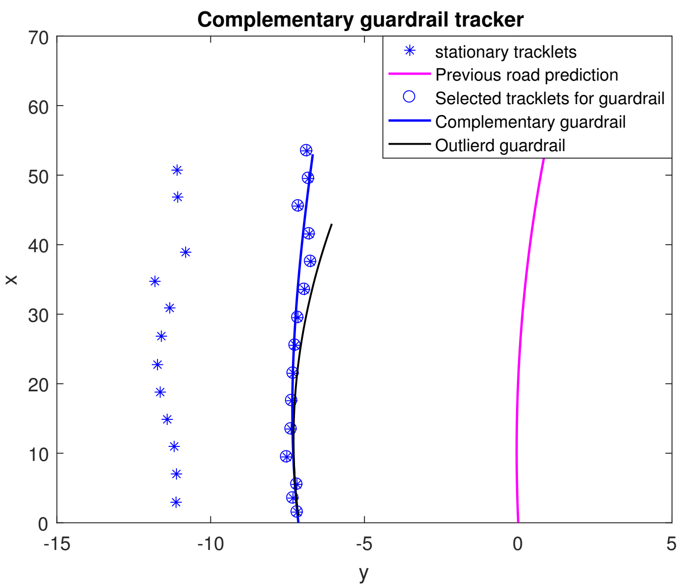



**Figure 5.3:** Outlier guardrail (in black) and complementary calculated guardrail (in blue) based on the previous road prediction (in pink).

In figure: 5.4 the performance of stationary tracklet filter is shown. The filtered tracklets are shown in blue and unfiltered tracklets in orange.

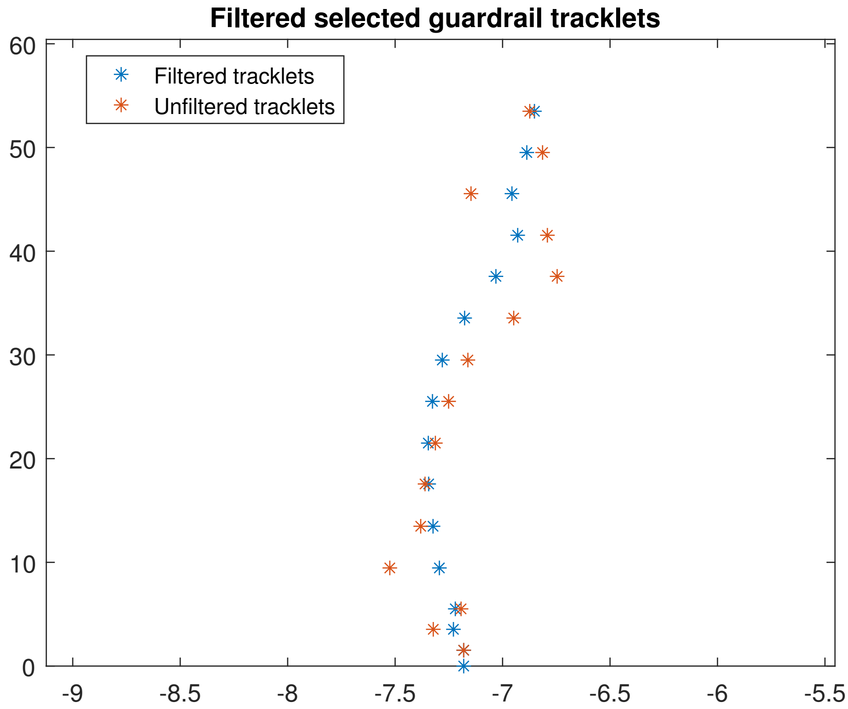



**Figure 5.4:** Kalman filtration of stationary tracklets estimated to belong to the guardrail.

Both availability and performance are important measurements of the guardrail. In most cases it can be assumed that the guardrail is following the direction of the road. Therefore the road geometry reference defined in chapter: 5.1.1 is used as ground truth data.

The result is evaluated as individual guardrails and as a merged guardrail, which is the actual guardrail used in the road model. The availability and the performance are given at different ranges from the host vehicle. In figure: 5.5 the RMS error for merged guardrail with and without complementary guardrail tracker are shown as a function of range. In the same figure the availability of the merged guardrail with and without complementary guardrail tracker is shown as function of range.

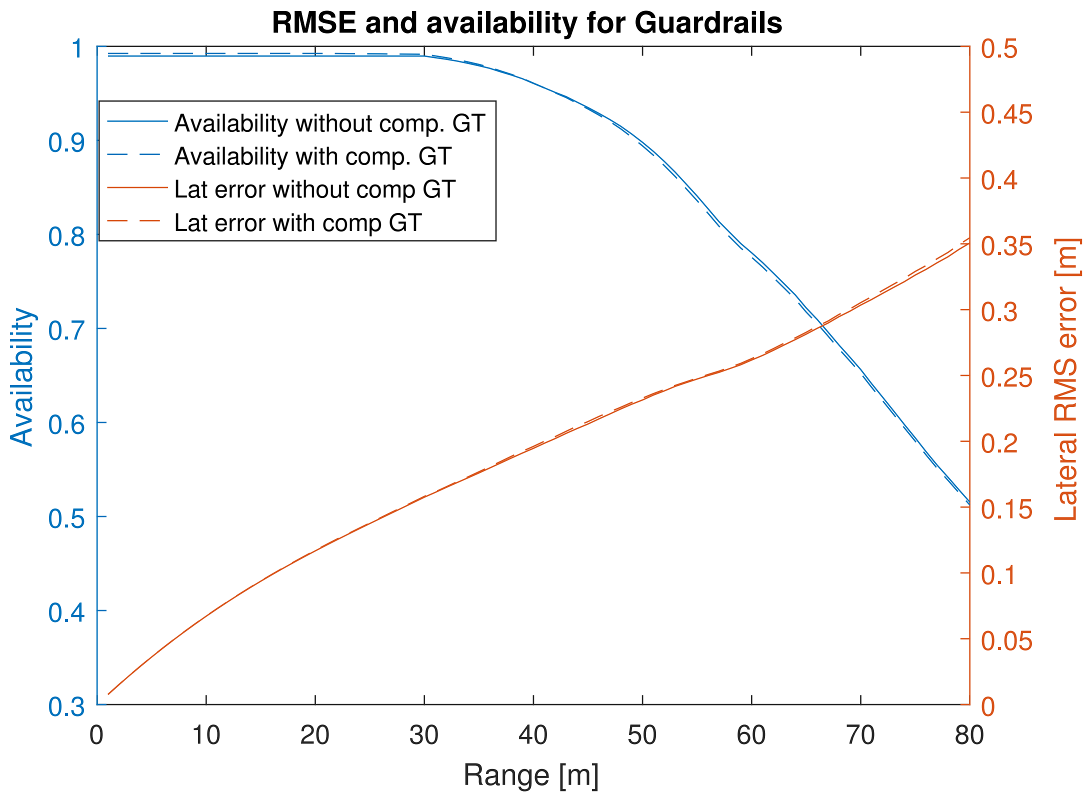



**Figure 5.5:** Comparison of the RMS error (in red) and availability (in blue) with and without complementary guardrail tracker.

In table: 5.1 the availability of the individual guardrails and merged guardrails with and without complementary guardrail tracker is presented. The availability is given for four different longitudinal ranges from the host vehicle.

| | $> 0$m | $> 20$m | $> 40$ m | $> 80$m |
|---|---|---|---|---|
| Left GR without complementary GT | 0.9940 | 0.9914 | 0.9718 | 0.5562 |
| Left GR with complementary GT | 0.9940 | 0.9921 | 0.9716 | 0.5550 |
| Right GR without complementary GT | 0.7501 | 0.7177 | 0.5970 | 0.2649 |
| Right GR with complementary GT | 0.7501 | 0.7222 | 0.6008 | 0.2641 |
| **Merged GR without complementary GT** | 0.9897 | 0.9897 | 0.9604 | 0.5166 |
| **Merged GR with complementary GT** | 0.9925 | 0.9925 | 0.9609 | 0.5122 |

**Table 5.1:** Availability of guardrail with and without complementary guardrail tracker for given longitudinal range.

In table: 5.2 the RMS error of the individual guardrails and merged guardrails with and without complementary guardrail tracker is presented. The RMS error is given at three different longitudinal ranges from the host vehicle.

| | 20m | 40 m | 80m |
|---|---|---|---|
| Left guardrail without complementary GT | 0.2450 | 0.3432 | 0.5959 |
| Left guardrail with complementary GT | 0.2013 | 0.2882 | 0.5781 |
| Right guardrail without complementary GT | 1.6899 | 0.6916 | 0.7678 |
| Right guardrail with complementary GT | 1.6757 | 0.6588 | 0.7565 |
| **Merged guardrail without complementary GT** | 0.1173 | 0.1983 | 0.3669 |
| **Merged guardrail with complementary GT** | 0.1179 | 0.2007 | 0.3695 |

**Table 5.2:** RMSE of guardrail with and without complementary guardrail tracker for given longitudinal range.

### 5.2.2 Filtering of receding vehicle

The RMS error for heading of the receding vehicles are compared to the heading of the road at the distance of the receding vehicle in order to determine the performance of the filters. The heading of the road is given from the derivative of the road geometry reference calculated in section: 5.1.1. The filtering of receding vehicles includes:

- Vehicle entering highway
- Ghost vehicles
- Shadowed vehicle

In table: 5.3 the RMS error of the heading of receding vehicles with and without filtering are presented. In the table the result of only using vehicle entering highway filtering and ghost vehicles filtering is also presented.

| | RMSE for heading |
|---|---|
| Without filter | 0.0356 |
| With ghost filter | 0.0282 |
| With entering highway filter | 0.0247 |
| With both ghost and entering highway filter | 0.0238 |

**Table 5.3:** RMSE for heading, comparing vehicle heading to road heading, using both ghost and entering highway filter.

The filter of shadow vehicles does not remove the measurement, instead it increases the noise of the measurement since the measurement is not as reliable as non shadow vehicles. In table: 5.4 the RMS error of the heading of receding vehicle with and without shadow filtering are presented.

| | Without filter | With shadowed vehicles filter |
|---|---|---|
| RMSE for heading | 0.0356 | 0.0317 |

**Table 5.4:** RMSE for heading, comparing vehicle heading to road heading.

### 5.2.3 Road estimation

The road estimation represents the predicted future path of the host vehicle therefore the road geometry reference is suitable to use as ground truth data. The RMS error of the road estimation is shown as a function of range in figure: 5.6. In table: 5.5 the RMS error is given for three different longitudinal ranges.

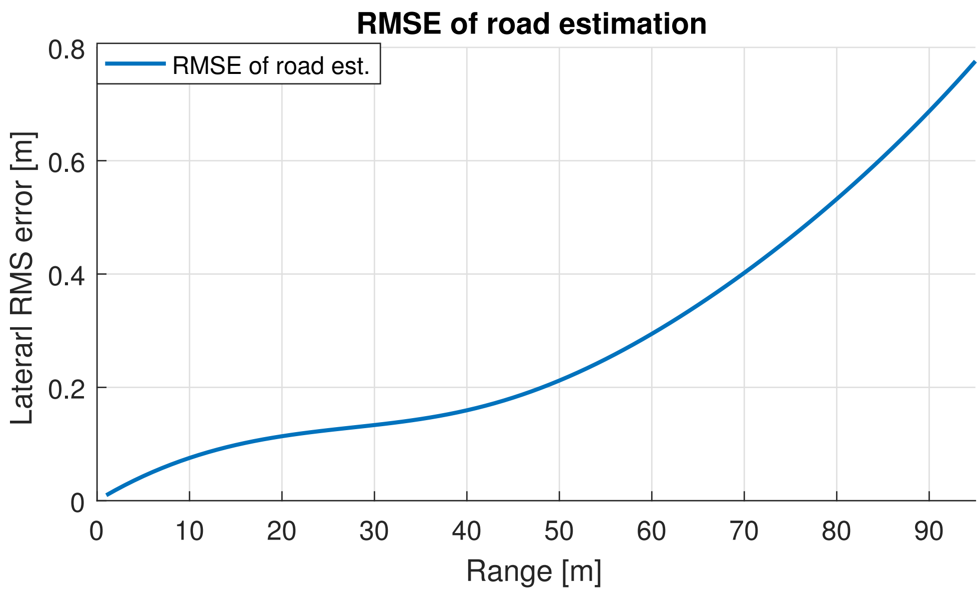



**Figure 5.6:** RMSE of road estimation as function of range.

| | 20m | 40 m | 80m |
|---|---|---|---|
| Road prediction | 0.1138 | 0.1595 | 0.5323 |

**Table 5.5:** RMSE of road prediction for given longitudinal ranges.

## 5.3 Target selection and analysis

By utilizing lane identification of receding vehicles from vision output the availability of active targets can be analyzed. Finding all instances a receding vehicle is in the same lane as the host vehicle according to the ground truth data and within the longitudinal maximum range for target requirements. The availability of the targets can then be analyzed by comparing the ground truth with reported targets by the target selection algorithm.

The dataset used for target selection includes the dataset used for validation of the road model and selected lane change logs. The total length of the data set is 60 min.

| | Availability |
|---|---|
| Radar only active target | 0.9027 |
| **Radar only valid target** | 0.8211 |
| Fusion active target | 0.9603 |
| **Fusion valid target** | 0.9192 |

**Table 5.6:** Availability of active target.

### 5.3.1 Target lane change identification

Target lane change identification is verified by utilizing lane classification from vision data as ground truth. The dataset used for the lane change analysis includes selected lane change logs. The total number of lane changes in this dataset is 58. In figure: 5.7 the time instance error between when the target selection algorithm releases the target is compared to the ground truth. Negative time instance error means that the target selection algorithm releases the target before ground truth identifies the lane change.

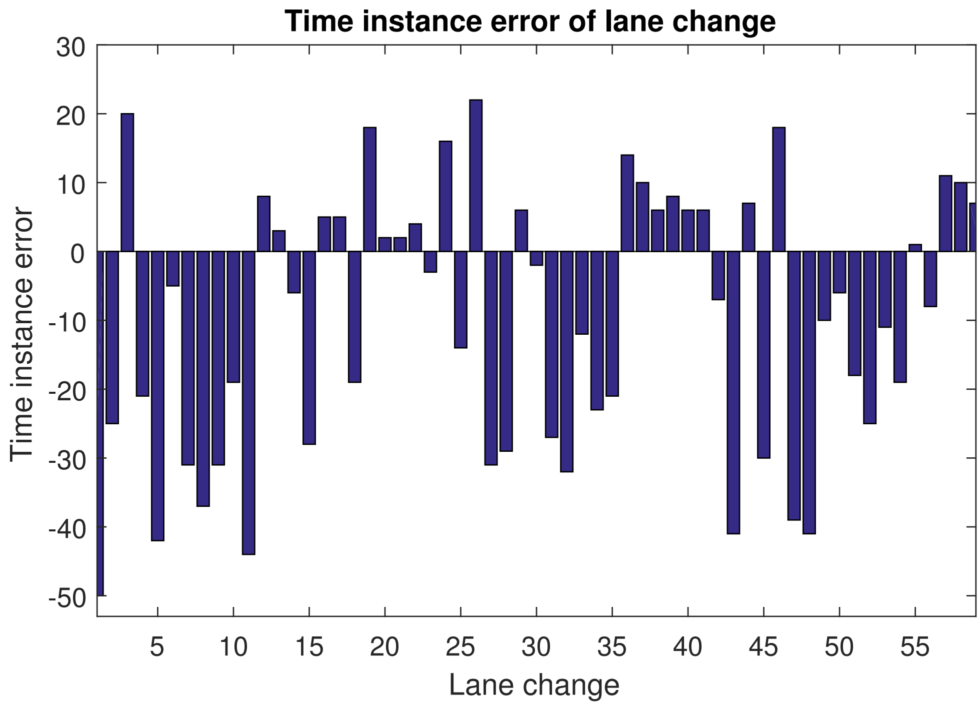



**Figure 5.7:** Time instance error for all 58 lane changes in the dataset.

| Time instance error | $< 0$ | $< 10$ | $< 20$ | $< 30$ |
|---|---|---|---|---|
| Result [%] | 59.32 | 84.75 | 99.61 | 100 |

**Table 5.7:** Performance of the target selection for lane change.

Sampling time is in average $0.03s$ which gives following translation to time in seconds:

| Time instance | 10 | 20 | 30 |
|---|---|---|---|
| Time $[s]$ | 0.3 | 0.6 | 0.9 |

**Table 5.8:** Time instance to seconds.

## 5.4 Target trail filtering

Filtering of radar trail is essential and easy to overview graphically in each iteration, as shown in figure: 5.8.

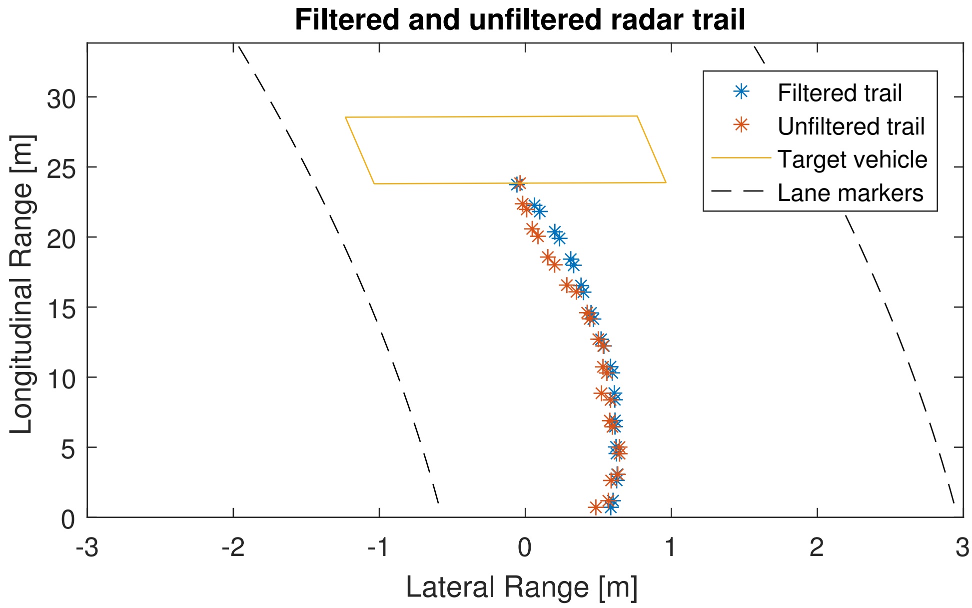



**Figure 5.8:** Illustration of extended Kalman filtering of target trail.

The result is measured as the RMS error between all points in the trail with and without filtering compared to the road geometry reference. The road geometry reference is not always representing how the target vehicle in front is driving. But in most cases the target in front is likely to drive in the path of the road geometry reference as long as it drives in the lane. The RMS error at different ranges are presented in table: 5.9.

| | 15m | 30 m | 50m |
|---|---|---|---|
| Trail without filtering [m] | 0.2633 | 0.3180 | 0.4797 |
| Trail with filtering [m] | 0.0929 | 0.1697 | 0.268 |

**Table 5.9:** RMSE of target trail with and without filtering.

And in figure 5.9 the RMS error for both filtered and unfiltered trail is described as a function of the range.

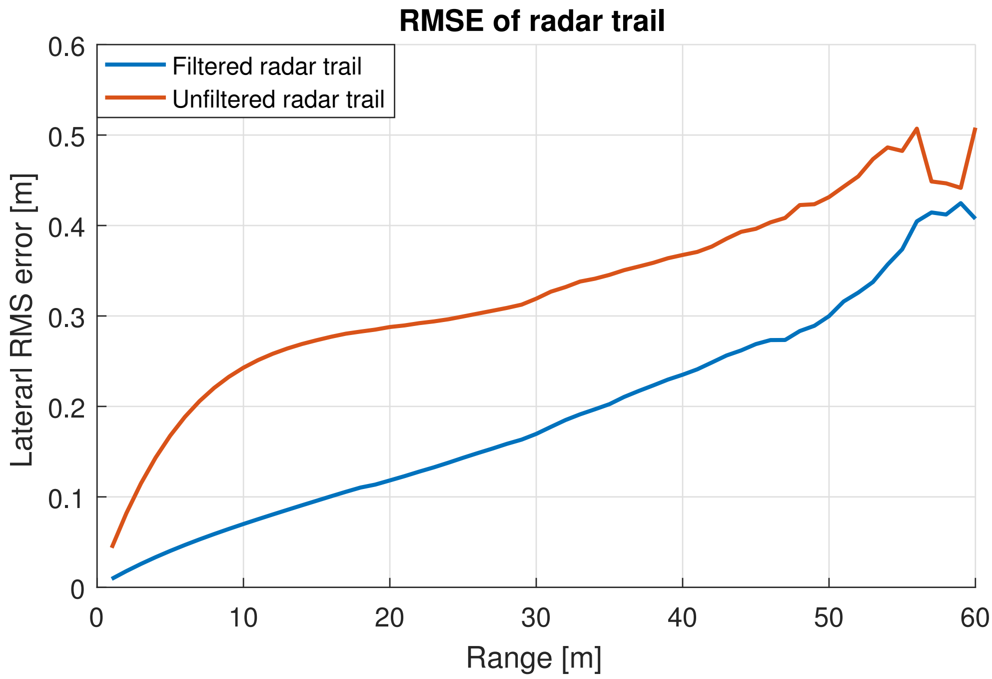



**Figure 5.9:** RMS error of filtered and unfiltered target trail.

## 5.5 Trajectory output

The final output of the trajectory algorithm is verified using the road geometry reference. The trajectory output is also compared to how a lane vision system and the trajectory algorithm on radar-vision fusion object would perform. Comparison of the performance between the trajectory algorithm on radar only and radar-vision fusion shows if the results are affected by the algorithm or unstable radar only tracking.

The outcome of the trajectory algorithm is also analyzed between development environment and embedded environment in order to validate that the performance is the same in simulations and embedded testing.

### 5.5.1 Verification of result in development environment and embedded code

In figure: 5.10 a comparison of the output from Matlab and the embedded C++ resim is shown. The coefficient compared is the $a_0$ parameter in the radar trail which represents the lateral offset of the trail to origin.

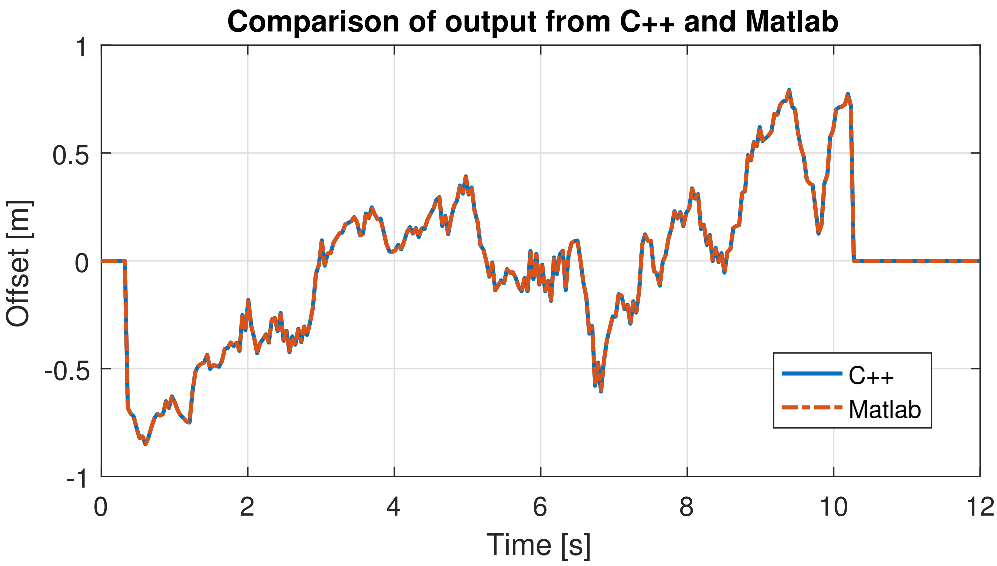



**Figure 5.10:** Comparison of coefficient $a_0$ of the trajectory polynomial from C++ and Matlab.

### 5.5.2 Trajectory output

The trajectory output is given as a third degree polynomial calculated by using a polynomial fitting of the filtered trail points. The availability of the output is the same as given from target selection in chapter: 5.3.

In figure: 5.11 the RMSE of the Trajectory output (radar only trail) is shown as a function of range. In the figure the RMSE of the vision lane centering and Trajectory output of fusion data (fusion trail) is also shown. In table: 5.10 the RMS error is presented at given ranges for the trajectory output with both radar only and fusion and the lane center polynomial given by the vision system.

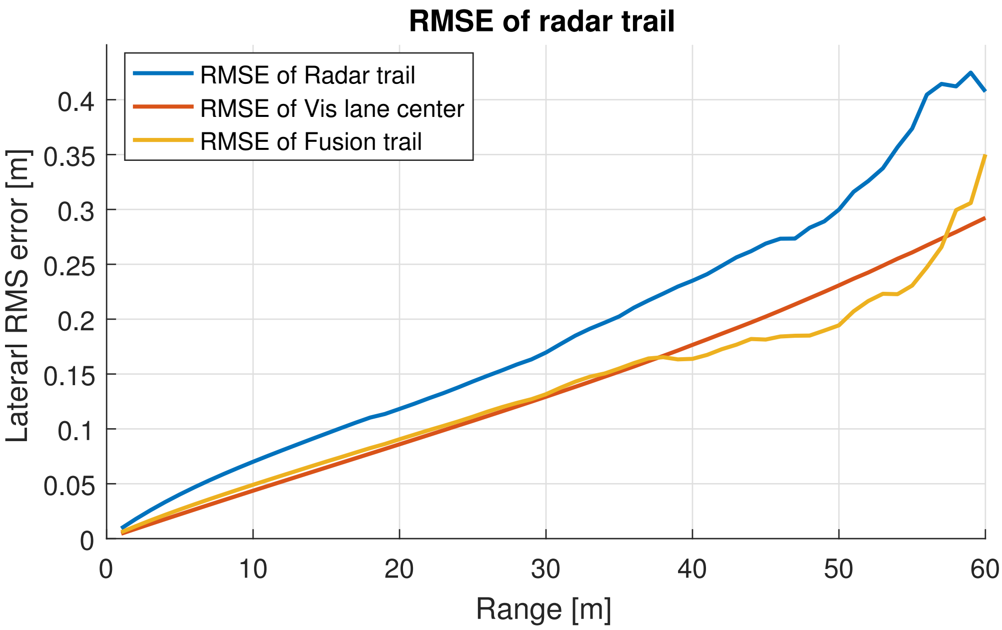



**Figure 5.11:** RMS error of radar only target trail, fusion target trail and vision lane ceneter.

| | 15m | 30 m | 50m |
|---|---|---|---|
| Radar only trail compared to road geometry reference | 0.0957 | 0.1696 | 0.2998 |
| Vision lane center compared road geometry reference | 0.0648 | 0.1293 | 0.2308 |
| Fusion trail compared road geometry reference | 0.0702 | 0.1316 | 0.1944 |

**Table 5.10:** RMSE of lateral trajectory output for given longitudinal range. Ground truth data given by road geometry reference.

# 6
# Discussion

In this chapter the discussion of the result is presented. The discussion includes reflection over the result, the utilization of the trajectory algorithm and future work.

## 6.1 Reliability of ground truth data

The chosen method for generating ground truth data has some disadvantages since the yaw rate is used to estimate the true path of the car. The yaw rate measurement does not always represent the movement of the vehicle accurately. In a banked curve the yaw rate will not be large enough to represent the true movement of the vehicle, instead the roll will angle increase. The result of this is that in a curve with a high banking angle, the estimated path of the curve is smaller than the true path of curve. The advantage of this method is that both ground truth and the measured data is in the same coordinate frame and hence are possible to compare. The error is however considered to be small and the advantage outweighs the disadvantage.

## 6.2 Result

The discussion of the result includes both a discussion of the final output of the trajectory algorithm but also step by step of the algorithm. The trajectory algorithm is based on three major steps, road estimation, target selection and trajectory design.

The road estimation is utilized in order to find and validate the target in the target selection. Based on a valid target a trail is generated that the trajectory design is based on. Every step of the algorithm is important for the final outcome, therefore all three steps are included in the discussion.

### 6.2.1 Road model

The road model estimation method was based on a previous proven working road model [17]. Some modification was however implemented in order to adjust the road model for longitudinal control, lateral control and radar only mode. The changes implemented in the road model include:

- **Guardrail outlier**
  The guardrail outlier was implemented in order to recognize and remove false reported guardrails from the update step in the road model. Guardrails are the

most reliable measurement in radar only systems and it is therefore important to secure that the used measurement is correct.

- **Implementation of complementary guardrail tracker**
In order to increase the availability of guardrails a complementary guardrail tracker was implemented. The complementary guardrail is used when a guardrail is reported as an outlier or not reported at all. The complementary guardrail tracker utilizes the previously calculated road in order to identify stationary tracklets belonging to the guardrail. This could be considered as a self-feeding error if the previous road estimation was inaccurate leading to the selection of false stationary tracklet for guardrail that will then be used in the update of the road model. This is however a behavior that has not been identified and the complementary guardrail is sparingly used. The complementary guardrail was first implemented in a stage when the regular guardrail tracker was performing worse and the complementary guardrail was more indispensable. When the regular guardrail tracker was replaced with a guardrail tracker from another Delphi project the complementary guardrail tracker to some extent became superfluous.

- **Utilization of roll angle measurements**
The roll angle is utilized together with knowledge about how roads are designed. With the knowledge about the roll angle and banking properties of the road the maximum curvature is limited in the update step. If the roll angle exceeds the threshold the noise of the measurement in the update step is increased. But due to the low accuracy of the measurement the full potential of this step cannot be reached.

- **Filtering of receding vehicle**
In order to increase reliability of receding vehicle measurements filtering of receding vehicle was implemented. The heading measurement of the receding vehicle used in the update step is an important measurement and it is therefore important to use vehicles that are driving on the highway and not entering, leaving or are falsely reported. In the same time the receding vehicles measurements are not as reliable in radar only fusion compared to radar and vision fusion. This creates the problem of identifying the bad measurements and removing them but it is also the reason for why this is essential.

- **Increase noise of host motion**
The road model was developed for longitudinal control and therefore based on host lateral motion [17]. Since the goal of the project is to achieve both longitudinal and lateral control the host motion dependency was reduced by increasing the noise in the update step of host motion. The increase is considered large enough to remove a non-desired closed loop behavior.

- **Remove target vehicle from receding vehicles**
Since the trajectory is based on a trail generated from the target vehicle, the target is removed from the set of receding vehicles used in the update step.

This leads to that the trajectory that the host will follow is independent of the road model that is only used in order to verify the target.

In figure: 5.5 the performance of the merged guardrail is presented. Looking in more detail at the result in table: 5.2 it is clear that the results for the left and merged guardrail are noticeable better than the right guardrail. It can be concluded that the high errors found while comparing the right guardrail to the road is due to that the guardrail does not represent the road in all highway situations. On the right side there are both ramps and exits which the guardrail follows. At these sections of the road the assumption that the guardrail follows the road is not correct, this is reflected in the measured performance.

The performance of the merged guardrail is better than both the left and right guardrail. The performance of the guardrail is overall impressively accurate. It can be concluded that the used guardrail in the update step, that have passed the guardrail outlier and complementary guardrail tracker is highly reliable. The high availability of the merged guardrail is also noticeable, at least one guardrail is almost always reported.

In figure: 5.6 the result of the final output of the road model is shown. The road estimation performs impressively well for a radar only system. The road estimation on average is always within half a lane width from the true road up to maximum estimation range. Stable and reliable road estimation is crucial for the design of the trajectory and the results show that the road estimation meets these requirements more than enough.

### 6.2.2 Target selection

Depending on if a target exists or not the target selection method is based on two different approaches.

When no target is available the lateral position of both the host and target vehicle can be utilized. Since no target is active no trajectory is generated and the driver is in control of the host vehicle. The target selection utilizes the longitudinal and lateral position and heading of the receding vehicles compared to the road estimation in order to identify a receding vehicle as a target vehicle.

When a target exist the host vehicle motion is dependent on the target vehicle. The trajectory is designed as a trail from the host vehicle and therefore lateral position cannot be used in order to validate that a target vehicle is driving according to the requirements or not.

The most important requirement of the target selection algorithm is to identify when a target changes lane and release it as a target when it does. In figure: 5.7 the time instance error is compared to the vision identification of lane change illustrated. The reason for why the target selection algorithm tends to release the target before the vision identification is due to two reasons. The algorithm is tuned to release a target too early rather than too late. Another reason is that when a target is starting a lane change the heading compared to the road is changed before the target vehicle has left the lane. Since the target selection is based on heading and not lateral position this is a reasonable behavior to identify.

The trade-off in the target selection algorithm is between high availability and sensitive tuning for identifying lane change. In table: 5.6 the availability of a target given that vision has reported a vehicle in a host lane within region of interest of being a target. Besides the sensitive tuning for identifying lane change the noise in the heading measurement is the reason for why the availability isn't higher. By comparison a radar-vision fusion system has a more stable heading measurement and therefore remarkably higher availability. This proves that the availability would be increased with development of radar hardware and more advanced tracklet tracking. However, the availability in the radar only system is still above 80% which can be considered as reasonable.

### 6.2.3 Trajectory

The final output of the trajectory includes:

- Coefficients for three degree polynomial describing the lateral future path
- Selected longitudinal distance between the host and the target vehicle
- Distance to target vehicle
- Longitudinal velocity of target
- Longitudinal acceleration of target

According to the theory chapter regarding control theory (3.7) the output from the trajectory is sufficient enough for both longitudinal and lateral control. The reference for longitudinal is not presented in the result since the reference is only given by the properties of the target vehicle directly reported by the fusion.

In figure: 5.11 the result of the reference for lateral control is presented. The performance of the radar only trajectory is compared with the performance of a fusion trail and the performance of a vision lane center polynomial. Since the vision lane center polynomial is the same output Volvo Cars uses for lateral control it is relevant and interesting to compare the trajectory to the vision output.

The performance of the lateral reference is considered to be good. The reason for why the fusion trail is performing better than the radar only is because of more accurate heading estimation of the target vehicle. The heading estimation is used to translate the target motion into the trail and with more accurate heading estimation the translation to the trail becomes more accurate. The most interesting result is the error around $15-30$m in front of the host vehicle. In a lateral lane center control the lateral offset is given at a pre-defined distance in front of the host vehicle, often around $15 - 30$m. The exponential error increase after 45m is not that relevant. Below 30m the difference between the radar only trajectory and vision lane center is less than 5cm which must be considered more than good enough. The difference between the performance of the fusion trail and vision lane center are negligible. This indicates that with a more accurate heading estimation a radar only can meet the performance of a lane center polynomial.

In figure: 5.10 the output in Matlab and C++ is verified to be exactly the same. This secures that the analysis in the development environment corresponds to the performance in the embedded software.

## 6.3 Utilization of radar only trajectory for highway driving

The reason for why a radar only solution for both longitudinal and lateral control can be hard to understand. The research regarding autopilot likely systems are in most cases based on advanced fusion systems with radars and cameras in 360 degrees around the vehicle. However, every car companies name are not Tesla or Volvo and can't afford expensive fusion sensor systems. This is why radar only systems are important, both hardware and software are developing rapidly and are cheaper than a radar-vision fusion system.

Being able to offer an affordable autopilot like system on a radar only platform would be a game changer for the automotive industry and highway driving. The result in this report proves that a radar only highway target lane following system is possible. Maybe the availability and reliability with today's radar solutions is not good enough but with the next generation of radar hardware and more advanced tracking algorithms of objects radar only highway target lane following could be achieved.

## 6.4 Utilization of target trail polynomial

Another important finding in the project is the possibility and utilization of a target trail and not only in a radar only set up. Highway pilot systems in fusion based system are mainly based on the output vision lane center polynomial. However, vision output is not always reliable. For example when there is snow, dirt on the road or heavy rain the vision system can have a hard time to report the lane markers. In these situations a target trail polynomial could serve an important purpose and replace the vision marker for the lateral control algorithm. The target trail polynomial could also serve a purpose as a second opinion to the lane center polynomial. If the target trail can verify that the lane center polynomial is correct the confidence of the highway pilot systems can be increased.

## 6.5 Future work

As mentioned previously in the discussion there are many interesting aspects of a radar only highway target lane following trajectory to further investigate. Both development of radar hardware, object tracking and development of the trajectory algorithm would increase the performance.

- **Radar hardware development**
  Implementation of the trajectory algorithm in the next generation radar hardware with higher accuracy and resolution of detection would increase the performance of the trajectory algorithm. With reliable elevation estimation of objects, stationary object could also be considered in the path planning.
- **Object tracking**
  With better radar only object tracking the performance of the trajectory algorithm would increase. This is shown by comparing how the performance of the algorithm is with radar-vision fusion compared to radar only. The heading estimation is an particularly important measurement which several key steps of the algorithm are based on.
- **Decrease dependency of host motion in road model**
  In this project the host vehicle motion was reduced in the road model, however reducing the host vehicle dependency more or even removing it would increase the independency between the road estimation and lateral control reference.
- **Closed Loop stability evaluation**
  In this project there were no possibilities to try the trajectory algorithm in closed loop system. The next natural step is to test the algorithm in a car with a steering unit in order to evaluate the closed loop behavior.
- **Integration in fusion based highway pilot system**
  Further investigation on how a target trail based trajectory could be utilized as a second opinion in a radar-vision fusion system with lane markers.
- **Driver communication and interface**
  In this thesis no focus has been on how the trajectory algorithm could be implemented as a product in a production vehicle. This would require an investigation on how the communication and interface to the driver would be designed. The interface between the driver and vehicle is extremely important in order to utilize the trajectory algorithm in a safe way.

# 7 Conclusion

This project concludes that a radar only system can be utilized in order to design a highway target lane following trajectory. If not with today's generation of radar hardware, at least with the next generation of radar hardware and object tracking. Longitudinal control is already possible with today's radar only systems. In this project it is shown that also lateral control is possible to achieve based on target lane following.

The algorithm presented in this report estimates the road in order to find and validate a suitable target vehicle. Based on the target vehicle a snail trail is generated which becomes the lateral reference for the host vehicle. It is shown that it is possible to identify if the target vehicle is driving in the same lane as the host vehicle and identify when the target vehicle is changing lane without seeing the lanes. The trajectory output is then based on the properties of the target vehicle and the generated target trail.

The algorithm developed in this project is not comprehensive enough to be considered an autopilot system. But the algorithm is enough to assist the driver with both longitudinal and lateral control on the highway. The longitudinal reference is given by the properties of the target vehicle and the desired distance to the target vehicle by the driver. The lateral reference is given by the target trail polynomial which is possible to replace with a vision lane center polynomial in a regular lane centering controller. The utilization of the trajectory algorithm can either be an active or passive system. It could be utilized as a passive system that assists the driver if it predicts that the driver is not following the road. It could also be utilized as an active system that controls the vehicle while the driver is monitoring the driving and is ready to assist the system if needed. This would also require a good interface to the driver to communicate when target lane following is possible and when it is not possible and the driver should take back control of the vehicle.

# Bibliography


[1] L. Xiao ,F. Gao , "A comprehensive review of the development of adaptive cruise control systems," *Vehicle System Dynamics*, 2010.

[2] Tesla Motors, "Description of autopilot system," https://www.tesla.com/presskit/autopilot#autopilot, accessed: 2017-01-12.

[3] Delphi Automotive Systems, "Delphi home page," http://www.delphi.com/, accessed: 2017-05-23.

[4] ——, "Delphi adaptive cruise control," http://www.delphi.com/docs/default-source/old-delphi-files/7705dce6-d83a-45b7-b378-bd3f732c76e0-pdf, accessed: 2017-01-20.

[5] Trafikverket, "Krav för vägars och gators utformning," Trafikverket, 781 89 Borlänge, June 2015.

[6] Näringsdepartementet, "Trafikförordning," Näringsdepartementet, Stockholm, September 1998.

[7] körkortonline.se, "Körning på landsväg," https://korkortonline.se/teori/landsvag/#avstand, accessed: 2017-01-26.

[8] D. K. Barton, *Radar equations for modern radar.* Boston, Mass: Artech House, 2013;2012;.

[9] A. M. Colman, *Doppler effect.* Oxford University Press, 2015.

[10] L. Svensson, "Designing a motion model, an introduction," 2016.

[11] B. Jacobson, C. U. of Technology, C. tekniska högskola, V. E. Department of Applied Mechanics, V. D. Autonomous Systems, and F. Institutionen för tillämpad mekanik, Fordonsteknik och autonoma system, "Vehicle dynamics compendium for course mmf062; edition 2016," 2016.

[12] MathWorks, "Polyfit," https://se.mathworks.com/help/matlab/ref/polyfit.html, accessed: 2017-04-18.

[13] S. Särkkä and C. U. (e-book collection), *Bayesian Filtering and Smoothing.* Cambridge: Cambridge University Press, 2013, vol. no. 3;no. 3.;.

[14] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Journal of Basic Engineering*, vol. 82, no. 1, p. 35, 1960.

[15] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, 2004.

[16] H. E. Rauch, C. T. Striebel, and F. Tung, "Maximum likelihood estimates of linear dynamic systems," *AIAA Journal*, vol. 3, no. 8, 1965.

[17] A. Petterson, V. Svensson, "Road geometry estimation for longitudinal vehicle control," Master's thesis, Chalmers University of Technology, 2015.

[18] S. K. Gehrig and F. J. Stein, "Elastic bands to enhance vehicle following." IEEE, 2001, pp. 597–602.

[19] Y. K. Hwang and N. Ahuja, "A potential field approach to path planning," *IEEE Transactions on Robotics and Automation*, vol. 8, no. 1, pp. 23–32, 1992.

[20] T. Glad and L. Ljung, "Control theory - multivariable and nonlinear methods." Taylor & Francis, 2000, pp. 43–322.

[21] K. J. Åström and R. M. Murray, *Feedback systems: an introduction for scientists and engineers*, illustrat ed. Princeton, N.J: Princeton University Press, 2008;2009;.

[22] W. Chee and M. Tomizuka, "Unified lateral motion control of vehicles for lane change maneuvers in automated highway systems." 1997.

[23] R. Rajamani, "Vehicle dynamics and control," 2012.

[24] S. Solyom and O. Nordin, "Method and apparatus for adaptive cruise control in a road vehicle," 2016.

[25] M. Roth, G. Hendeby, and F. Gustafsson, "Ekf/ukf maneuvering target tracking using coordinated turn models with polar/cartesian velocity." International Society of Information Fusion, 2014, pp. 1–8.