Department of Mechanical Engineering
Dynamics and Control Research Group

Vehicle State Estimation by Fusing Kinematic Sensors with Radar and Data from V2V Communication

Automotive Technology
Master's Thesis

B.J. (Bart) van de Sande
1017995
DC 2023.078

Supervisors:
dr. ir. A.A.J. Lefeber
ir. R. de Haan

Eindhoven, October 2023

This report was made in accordance with the TU/e Code of Scientific Conduct for the Master thesis

Abstract

During platooning, a vehicle is followed autonomously by one or multiple vehicles. The vehicle states of the leading vehicle are required to be accurately estimated to ensure proper reference inputs for the motion controllers of the following vehicle. The vehicles used in this project are equipped with onboard kinematic sensors, forward-facing radar and V2V communication. State information of the leading vehicle can be determined from the V2V communication and/or the forward-facing radar measurements. Three vehicle state estimation systems are developed that estimate the vehicle states of the leading and following vehicles in a platoon depending on what information sources are available. When only the V2V communication is available, an EKF with a unicycle model is used to increase the sampling frequency of the received V2V messages containing the vehicle states of the leading vehicle. Using discrete dynamics the estimated vehicle states are compensated for the delay in communication. The vehicle states of the following vehicle are estimated by fusing measurements from onboard kinematic sensors with GPS measurements. When the V2V communication is unavailable, the state estimation of the leading vehicle is done by using an EKF that determines unicycle model states from radar measurements. When the V2V communication is active and the radar detects the leading vehicle, the EKF and discrete dynamics are used to determine the states of the leading vehicle without delay. This is fused with the radar measurements and estimated following vehicle states in an EKF with two unicycle models.

In simulation, using both the V2V communication and radar measurements poses a trade-off, where some of the noise on the radar measurements is introduced to the estimation. However, this results in the most accurate absolute vehicle state estimation out of the three methods. Estimating the states of the leading vehicle from the V2V communication results in a more accurate estimation of the velocity and heading than when only the radar sensor is utilized. The estimation of the relative vehicle distance and velocity is more accurate when using only the radar sensor compared to using both the radar sensor and the V2V communication. Because both the absolute and relative vehicle states are required for the vehicle motion controllers, using radar measurements and V2V communication yields the highest-performing state estimation. Full-scale experiments are conducted and the vehicle state estimation systems are tested with the measurement data. The accuracy of the system cannot be determined from the experiments due to the absence of ground truth data.

The main recommendation is to experimentally validate the developed systems by using an RTK-GPS sensor to gather ground truth data. By comparing the estimated vehicle states of both vehicles to the ground truth data, the performance and accuracy of the vehicle state estimation system in different situations can be assessed experimentally.

Acknowledgments

First of all, I would like to express my sincere gratitude towards dr. ir. Erjen Lefeber and ir. Redmer de Haan for the boundless effort of supervising me during my project. Erjen, thank you for providing me with feedback and critical questions regarding my progress. This gave me new insights and helped me to view my problems from a different perspective. Redmer, thank you for getting me acquainted with the vehicles used for testing and their software. Your clever suggestions and comments contributed greatly to the quality of my project. Furthermore, I would like to thank both of you for the sympathy I received during our meetings when I ran into personal issues.

I would like to express my appreciation towards the Generaal Majoor de Ruyter van Steveninckkazerne and Patrick Schmitz for giving me the opportunity to perform full-scale experiments on the training facility. This ensured a controlled test environment to perform the experiments in a safe manner.

A big thanks to Gerard and Erwin from the AT lab for assisting with the experiments and maintaining the test vehicles. Without them, the experiments would not have been possible.

I would like to thank everyone who has been working in the AT lab for the past 9 months for the enjoyable atmosphere and for thinking along. I cherish the fun times and the open discussions about our projects. Prabhat, I am thankful for your brainstorming with me and for your insights on object tracking. Aron, thank you for proofreading sections of my report and putting in the effort to give concise and useful feedback.

Lastly, I would like to thank my family and friends for supporting me emotionally. Conducting this thesis was harsh at times, but your unconditional support helped me through it. Thanks for giving me opportunities to relax and caring for me.

Nomenclature

Accents

$\dot{}$	First derivative
$\hat{}$	Extended Kalman Filter estimate
^{T}	Transpose

Abbreviations

ACC	Adaptive cruise control
CACC	Cooperative adaptive cruise control
CAM	Cooperative awareness message
CoG	Centre of gravity
DD	Discrete dynamics
EKF	Extended Kalman Filter
ENU	East North Up
ESKF	Error state Kalman Filter
FoV	Field of view
GNN	Global Nearest Neighbour
GNSS	Global Navigation Satellite System
GPS	Global positioning system
IMU	Inertial measurement unit
KF	Kalman Filter
M2M	Measurement-to-measurement
MIO	Most Important Object
NED	North East Down
T2T	Track-to-track
UKF	Unscented Kalman Filter
V2V	Vehicle-to-vehicle

Sub- and superscripts

x	Longitudinal
y	Lateral
z	Vertical
0	Initial
1	Leading vehicle
2	Following vehicle
m	Measurement
c	Received from communication
r	Relative
rad	Radar measurement
exp	Expected
cor	Corrected

Matrices

A	Linearized system matrix
B	Input system matrix
C	Linearized measurement matrix
I	Identity matrix
K	Kalman gain
P	Kalman covariance
Q	Process noise covariance
R	Measurement noise covariance
S	Residual covariance

Vectors and reference frames

$\vec{e}_0$	Inertial reference frame
$\vec{e}_1$	Relative reference frame fixed to vehicle 1
$\vec{e}_2$	Relative reference frame fixed to vehicle 2
$\vec{p}_1^0$	Absolute position of vehicle 1
$\vec{p}_2^0$	Absolute position of vehicle 2
$\vec{p}_1^2$	Relative position of vehicle 1 in the reference frame fixed to vehicle 1

Symbols

m	Vehicle mass	kg
I_{zz}	Vehicle yaw inertia	kgm^2
l	Wheelbase	m
l_r	Distance from rear axle to radar	m
a	Length from front wheel to centre of gravity	m
b	Length from rear wheel to centre of gravity	m
$C_{\alpha f}$	Cornering stiffness of the front axle	N/deg
$C_{\alpha r}$	Cornering stiffness of the rear axle	N/deg
$F_{y,f}$	Lateral tire force of the front wheel	N
$F_{y,r}$	Lateral tire force of the rear wheel	N
v_x	Longitudinal velocity	m/s
v_y	Lateral velocity	m/s
v_0	Initial velocity	m/s
a_x	Longitudinal acceleration	m/s^2
a_y	Lateral acceleration	m/s^2
a_z	Vertical acceleration	m/s^2
t	Time	s
α_f	Front wheel slip angle	rad
α_r	Rear wheel slip angle	rad
δ	Steering angle	rad
ψ	Heading angle	rad
$\dot{\psi}$	Yaw rate	rad/s
x	First coordinate of inertial vehicle position	m
y	Second coordinate of inertial vehicle position	m
ω_x	Roll velocity	rad/s
ω_z	Yaw velocity	rad/s
k	Timestep	-
f	Unicycle model system equations	-
c	Measurement system equations	-
$\tilde{y}$	Measurement residual	-
T_s	Sampling time	s
u	Input signal	-
h	Communication delay	s
x	System state	-

$v_{g,rad}$	Ground speed of radar object	m/s
d_{pos}	Position deviation	m
d_{vel}	Velocity deviation	m/s
x_r	Relative longitudinal distance	m
y_r	Relative lateral distance	m
$v_{x,r}$	Relative longitudinal velocity	m/s
$v_{y,r}$	Relative lateral velocity	m/s
σ_x	Relative longitudinal distance covariance	m
σ_y	Relative lateral distance covariance	m
σ_{vx}	Relative longitudinal velocity covariance	m/s
σ_{vy}	Relative lateral velocity covariance	m/s
$x_{r,exp}$	Expected relative longitudinal distance	m
$y_{r,exp}$	Expected relative lateral distance	m
$v_{x,r,exp}$	Expected relative longitudinal velocity	m/s
$v_{y,r,exp}$	Expected relative lateral velocity	m/s
α_{exp}	Expected radar angle	rad
r_{exp}	Expected radar range	m

Contents

Chapter 1

Introduction

1.1 Project Context and Problem Definition

The number of personal vehicles on the road is ever increasing [1], and as a result congestion and the number of accidents increase [2]. Furthermore, human errors are the most common cause for car accidents [3]. Autonomous driving has the potential to significantly reduce both congestion and the number of accidents [4], by removing the error-prone human driver. The distance between vehicles can be decreased when vehicles are driven autonomously compared to when driven by a human driver [5]. This will increase traffic flow, and thus decrease congestion. This makes automated driving a highly relevant research area.

To participate in the research field of autonomous driving, the TU/e started the i-CAVE (integrated Cooperative Automated VEhicles) project in 2015, which ended in 2021. To perform full-size experiments the TU/e has equipped several vehicles with hardware and software required for driving autonomously. The vehicles which are used for this project are Renault Twizies. The Renault Twizy is a small electric vehicle able to carry two people. On these vehicles, an inertial measurement unit (IMU), a Global Navigation Satellite System (GNSS), and a front-facing radar have been mounted. These vehicles are also equipped with communication hardware, such that vehicle-to-vehicle (V2V) communication is available. Currently, the TU/e has three Renault Twizies to perform platooning experiments with, which makes it possible to perform special experiments like autonomously merging between two vehicles.

Platooning is an application of autonomous driving where multiple vehicles form a closely packed formation with short inter-vehicle distances. A leading vehicle is followed autonomously by one or multiple following vehicles. For the following vehicles, the position and other vehicle states of the leading vehicle must be known accurately to ensure correct references for the driving controllers. The position and motion of the leading vehicle are measured with multiple sensors with different characteristics such as range and accuracy, and have different update frequencies.

Sensor measurements always show undesired inaccuracies. Measurement errors can be detected by including redundancy in the sensor set. Sensor fusion is described as "the combining of sensory data or data derived from sensory data such that the resulting information is in some sense better than would be possible when these sources were used individually" [6]. It is argued that using multiple types of sensors for redundant measurements is beneficial for negating the spatial and temporal limitations of individual sensors, sensor deprivation, and sensor imprecision and uncertainty. The main objective of sensor fusion systems is to take measurements from one or multiple sensors to estimate or infer one or more quantities of interest [7]. For a sensor fusion system to function, three main components are required consisting of one or multiple sensors that measure an observable quantity, one or multiple models that relate the sensor output to the quantity of interest, and an estimation algorithm that combines model and measurement data.

In this project, a situation is investigated where two vehicles are driving in a platoon. The following vehicle receives information about the vehicle states of the leading vehicle from the V2V communication and the measurements done by the forward-facing radar. With onboard kinematic sensors and GNSS measurements, the vehicle state of the following vehicle can be determined. This results in multiple sources of information for estimating the vehicle states of both vehicles.

1.2 Research Objectives

The main research goal is formulated as:

> ***"Develop an algorithm that fuses measurements from forward-looking sensors with onboard GNSS and IMU sensors, and V2V communication data for vehicle state estimation for the leading and following vehicles in a platoon."***

From this main goal, several sub-goals are formulated:

- The algorithm should be able to fuse measurements from different sensors which are taken in different reference frames at different sample frequencies.
- The algorithm should be able to match the correct radar object data to the communicated vehicle information.
- When either the communication is interrupted or the radar sensor does not provide data, the algorithm should still provide an estimation of the vehicle states.

1.3 Thesis Outline

The used vehicle fixed reference frame, Kalman Filtering and the vector notation used in this report are discussed in Chapter 2. An overview of literature is shown in Chapter 3. In Chapter 4, the test vehicles and their equipment required for driving autonomously are discussed. The simulation model that is used to develop and test the sensor fusion system offline is explained in Chapter 5. Chapter 6 discusses the algorithm that estimates the vehicle states of both vehicles. Chapter 7 discusses the full-scale experiments conducted to gather measurement data to test the different systems on. The method developed to determine the correct radar object data based on state information from both vehicles is discussed in Chapter 8. The application of the vehicle state estimation algorithm on experimental data is discussed in Chapter 9. In Chapter 10, the conclusions and recommendations are discussed.

Chapter 2

Preliminaries

This chapter introduces the Kalman Filter (KF) and the reference frames and the vector notations used in this thesis. First, the KF and the underlying equations are introduced. Next, the reference frames and the notation used to denote them are discussed. The vectors indicating the position of the vehicles in different reference frames are introduced. At last, the conversion between the different reference frames is discussed.

2.1 Kalman Filtering

The KF is a stochastic method that uses a mathematical model for filtering signals with statistical and systematical errors. It was introduced by Kalman and Bucy in 1960 [8]. A KF is used to fuse data from measurements, by providing a maximum likelihood estimate of the parameter. It can also be used to relate inputs from multiple sensors to internal states of a model when the dependencies are linear. Figure 2.1 shows the basic concept of Kalman filtering. The KF algorithm consists of three steps. Firstly, based on the previous state estimation and

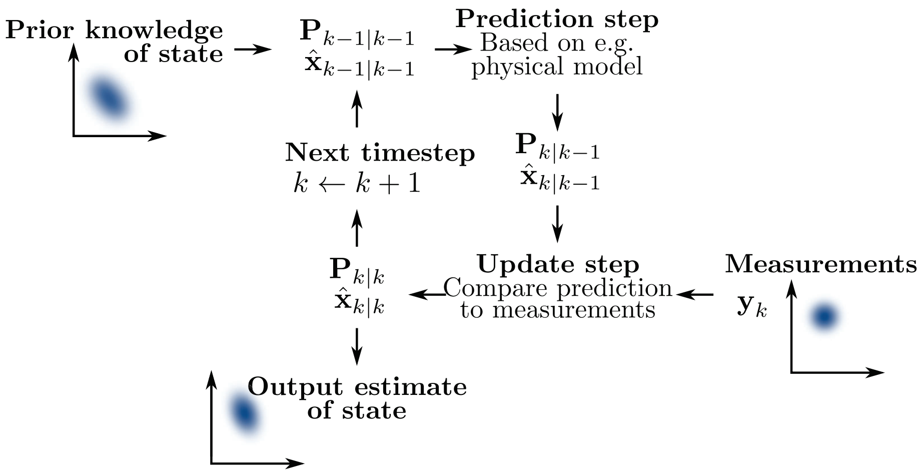


Figure 2.1: Overview of the concept of Kalman Filtering [9]

its covariance the KF predicts the system states of the next time step. Once the measurements are received, this prediction is compared to the measurements. Based on the Kalman covariance, P, the predicted states and the measurements a new estimate of system states is calculated.

The standard KF is only applicable for linear systems [8]. To apply Kalman filtering in non-linear systems, an Extended Kalman Filter (EKF) is used. The principle of this filter is similar but involves linearization of the non-linear system [10]. During the prediction step, the system state matrix is linearized using a first-order Taylor approximation. During the comparison of the measurement and determining the Kalman gain, the measurement matrix is linearized using a first-order Taylor approximation. When the non-linearities in the system are large, the EKF becomes inaccurate due to the linearization using a first-order approximation [11].

The system state estimate at the current time step k is $\hat{x}_{k|k}$. In this notation, the first subscript indicates the time instance of the prediction step, and the second subscript indicates the time instance of the measurements which are used. Firstly, the EKF calculates a prediction step based on the system equations f, the previous estimation of system states and the inputs. The state estimation for the current time without incorporating the current measurements, $\hat{x}_{k|k-1}$, is calculated by evaluating the system evolution equations in the last state estimate with the current input as

$$\hat{x}_{k|k-1} = f(\hat{x}_{k-1|k-1}, u_k) \tag{2.1}$$

where $\hat{x}_{k-1|k-1}$ is the state estimation done at time step $k-1$, and u_k is the current input. To calculate the predicted covariance estimate, $P_{k|k-1}$, the system and measurement equations have to be linearized. The linearized system at time k, A_k, is given by

$$A_k = \frac{\partial f}{\partial x}|_{\hat{x}_{k-1|k-1}, u_k}. \tag{2.2}$$

The linearized measurement matrix, C_k, is calculated using

$$C_k = \frac{\partial c}{\partial x}|_{\hat{x}_{k|k-1}}, \tag{2.3}$$

where c is the measurement function. The predicted Kalman covariance estimate, $P_{k|k-1}$ is calculated using

$$P_{k|k-1} = A_k P_{k-1|k-1} A_k^T + Q, \tag{2.4}$$

where $P_{k-1|k-1}$ denotes the Kalman covariance at the previous time instance and Q is the process noise covariance. This matrix describes the uncertainty in the system state equations. The measurement residual, $\tilde{y}_k$, is calculated using

$$\tilde{y}_k = y_k - c(\hat{x}_{k|k-1}), \tag{2.5}$$

where y_k describes the incoming measurements at the current time step k. The residual covariance S_k is calculated using

$$S_k = C_k P_{k|k-1} C_k^T + R, \tag{2.6}$$

where R is the measurement noise covariance, which describes the uncertainty in the incoming measurements. The Kalman gain K_k is calculated using

$$K_k = P_{k|k-1} C_k^T S_k^{-1}, \tag{2.7}$$

where S_k^{-1} denotes the inverse of matrix S_k. The new state estimation, $\hat{x}_{k|k}$, is calculated using

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k \tilde{y}_k. \tag{2.8}$$

Lastly, the current Kalman covariance is calculated using

$$P_{k|k} = (I - K_k C_k) P_{k|k-1}. \tag{2.9}$$

Every time step that the real-time application is executed, a state estimation is calculated using the previously discussed equations.

2.2 Reference Frames and Vectors

In this report, the vector notation as discussed in [12] is used. A reference frame consisting of three orthogonal axes is described as $\vec{e}^q$, where q describes the origin of the reference frame. Reference frame $\vec{e}^0$ is the inertial reference frame fixed to earth. The reference frames $\vec{e}^1$ and $\vec{e}^2$ are fixed to the leading and following vehicle respectively. A position vector is denoted as $\vec{p}_w^q$, where q describes the reference frame in which the vector is expressed, and w describes the object which is indicated by the vector.

Figure 2.2 shows the situation that is investigated in this project. Vectors $\vec{p}_1^0$ and $\vec{p}_2^0$ denote the positions of the leading and following vehicle respectively. Vector $\vec{p}_1^2$ denotes the position of the leading vehicle in the reference frame of the following vehicle. These vectors and their time derivatives correspond to the relevant vehicle states, which consist of the position, velocity, acceleration, heading angle, and yaw rate. This report uses the East North Up (ENU) inertial reference frame so that the x-axis of the inertial frame, $\vec{e}_1^0$, is pointing to the East as in Figure 2.2. The y-axis of the inertial frame, $\vec{e}_2^0$, points towards the North. To have a right-handed axis system, the z-axis of the inertial frame, $\vec{e}_3^0$, points upwards. The heading angle of a vehicle is 0 rad when the vehicle is driving towards the East, and a positive rotation is defined as counterclockwise.

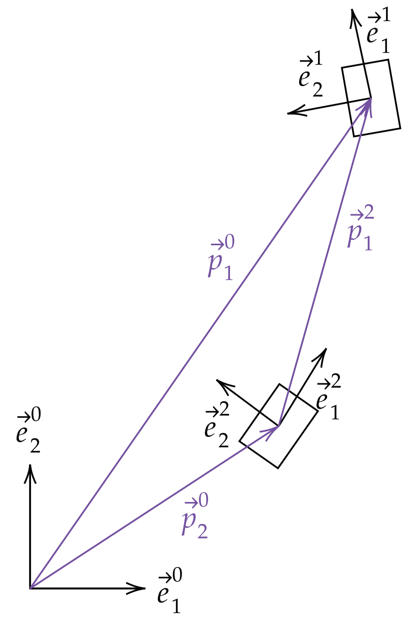


Figure 2.2: Representation of the investigated situation with two vehicles in 2D space and their position vectors.

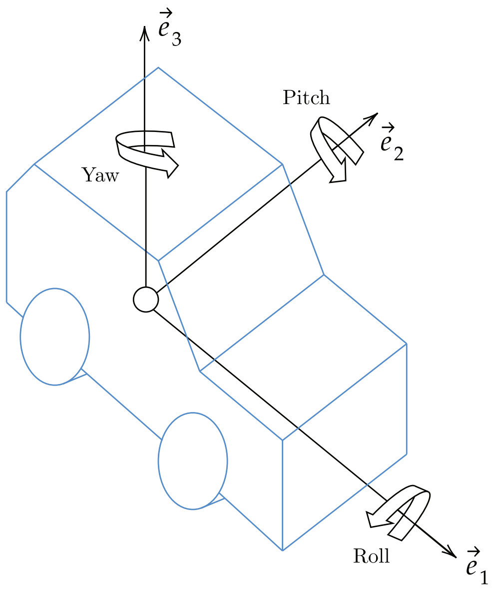


Figure 2.3: Representation of the ISO 8855 vehicular coordinate system.

2.2.1 ISO 8855 Axes System

The ISO 8855 reference frame is used to define the vehicle fixed reference frames. ISO 8855 is an international standard defining a vehicular axes system. The origin of the three-axis system is fixed to the centre of gravity of the vehicle [13]. Figure 2.3 shows this axes system fixed to a vehicle.

The first vector of the vehicle-fixed reference frame of vehicle 1, $\vec{e}^{\,1}_1$, points in the driving direction of the vehicle. The second axis of a vehicle-fixed reference frame, $\vec{e}^{\,1}_2$, points to the left of the vehicle. At rest, these vectors are parallel to the ground. The third vector of the vehicle-fixed reference frame points upwards, so a counter-clockwise rotation is denoted as positive. The axes system remains fixed to the vehicle when it pitches or rolls so that either or both the $\vec{e}^{\,1}_1$ and $\vec{e}^{\,1}_2$ axes no longer are parallel to the ground.

2.2.2 NED Axes System

In the software on the Renault Twizy the North East Down (NED) reference frame is used. This is an inertial reference frame where the first axis points North, the second axis points East, and the third axis points down. In this reference frame, a positive rotation is clockwise. The heading angle of 0 rad is defined to the North.

2.2.3 Coordinate System Conversion

A position vector in the inertial NED frame $\vec{p}^{NED}$ can be expressed in the ENU frame using

$$\vec{p}^0 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \vec{p}^{NED}, \tag{2.10}$$

where $\vec{p}^0$ is the same position vector expressed in the ENU frame. The heading angle of a vehicle expressed in the NED frame ψ_{NED} is converted to the heading angle in ENU frame ψ_{ENU} using

$$\psi_{ENU} = (2\pi) - (\psi_{NED} - \frac{1}{2}\pi), \tag{2.11}$$

to account for the difference in headings at 0 rad and the different rotation directions.

To convert a position vector in the inertial ENU frame to the same vector expressed in the reference frame fixed to vehicle 2, a coordinate transformation is used. This coordinate transformation from a position vector in the inertial reference frame $\vec{p}^0$ to a position vector in the vehicle fixed reference frame to the following vehicle, $\vec{p}^2$, is expressed as

$$\vec{p}^2 = \begin{bmatrix} \cos(\psi_2) & \sin(\psi_2) & 0 \\ -\sin(\psi_2) & \cos(\psi_2) & 0 \\ 0 & 0 & 1 \end{bmatrix} \vec{p}^0 \tag{2.12}$$

where ψ_2 is the heading angle of vehicle 2. Here it is assumed that the vehicles move in a 2D plane.

2.3 Summary

This chapter describes the basic principles of Kalman filtering. The vector notation and reference frame which are used in this report are discussed. The ISO 8855 vehicle fixed reference frame is explained and the NED and ENU reference frames are elaborated. Lastly, the conversions between the NED, ENU and vehicle fixed reference frames are discussed.

Chapter 3

Literature Study

In this chapter, an overview of conducted research in the field of vehicle state estimation is given. First, an overview of what sort of sensors, models and algorithms are used in vehicle state estimation systems is given. Secondly, research on tracking objects from radar measurements is discussed. Lastly, several vehicle motion controllers and their required reference inputs are discussed.

3.1 Vehicle State Estimation

Autonomously driven vehicles use software to determine and control the motion of the vehicle. This software consists of controllers that determine the desired motion of the vehicle, and what the throttle, brake, and steering inputs should be. For these motion controllers, the reference is crucial as controllers cannot give proper outputs when controlling towards an incorrect reference. Because of this, the estimation of the states of both vehicles is of great significance.

The tracking of other vehicles can be done in a vehicle fixed reference frame, or the inertial reference frame. Using radar measurements, V2V communication, and global positioning system (GPS) measurements the tracking of vehicles in the inertial frame is improved compared to only using GPS measurements [14]. Here, a situation with driving on a three-lane road with three neighbouring vehicles is considered. Multiple methods are compared with differential GPS combined with V2V and ultra-wideband radio ranging providing the most accurate position tracking. Track-to-track (T2T) association is used to match the data coming from different vehicles, and a KF is used to fuse the data coming from various sensors.

In [15], radar sensor measurements are fused with IMU and accelerometer data when GNSS measurements are not available. Here both the vehicle fixed frame and the inertial frame are considered. For data fusion, an EKF is used where the model states consist of an extended unicycle car model. This unicycle car model has extra states regarding the vertical position and velocity, and the velocities in the inertial frame are also taken as states. In this paper, only the localization of the host vehicle is determined using sensor fusion as no other vehicles are considered. It is concluded that drift occurs when the GNSS signal is lost for a longer amount of time.

In [16], a KF is used to fuse GPS and IMU data. A Wiener process acceleration model is used to simulate the vehicle in the inertial frame. Based on fuzzy logic, the sensor fusion algorithm switches between inputs when the GPS sensor gives invalid measurements. It is argued that a KF is an attractive sensor fusion tool because of the additive nature of the update step. This algorithm also can be extended easily to incorporate other types of measurements such as a radar sensor. For vehicle motion controllers only the location of the vehicle in the inertial frame is not enough, as the heading and velocity are also required. Therefore, the Wiener acceleration process model cannot be used for the estimation of vehicle states. Extending this model to include the velocity and heading would make using the Wiener acceleration process redundant, as a normal state for the acceleration can be used in that case.

IMU and GPS data are fused using a smoothed error state Kalman Filter (ESKF) in [17]. Here a non-linear vehicle model, that contains information regarding the location, speed, angular velocity and acceleration, is used in the ESKF. Rauch-Tung-Striebel smoothing ensures less noise in the results. This is a form of smoothing where all the previous set of states are incorporated to enhance the localization accuracy and robustness. Due to its nature of using all previous state estimates, this smoothing requires a lot of computation power. Therefore due to computational constraints, this method is undesired.

In [18], a single-track vehicle model extended with kinematic body motion equations for road banking and inclination is used for vehicle state estimation through sensor fusion. A KF is used to fuse sensors commonly mounted on production cars to estimate 3D vehicle velocity and vehicle pitch and roll. The roll and pitch states of a vehicle are not relevant for autonomously following a vehicle. Furthermore, the inputs to the kinematic single-track model are not available for both vehicles. A single-track model also requires vehicle parameters that may not be available in a heterogeneous platoon. Therefore, using a single-track model is not possible.

The Renault Twizies used in this thesis have a specific sensor set consisting of an IMU, an accelerometer, a GNSS receiver, and a radar sensor. Furthermore, a radar sensor can be used to determine more information regarding the leading vehicle than purely the position. Therefore, to incorporate all these sensors and estimate the relevant vehicle states of both the leading and following vehicles, a new solution has to be proposed.

3.2 Radar Tracking

When utilizing the radar sensor, only the measurement data regarding the leading vehicle should be considered. The radar sensor detects multiple objects in its field of view (FoV), and therefore the correct data should be selected. This can be done using the information received from the V2V communication between the vehicles.

The Most Important Object (MIO) from the radar data is determined based on assumptions regarding the relative location to the host vehicle in [19]. In this research, multiple objects are tracked using sensor fusion. Radar measurements are fused with data from onboard sensors on the host vehicle using an Unscented Kalman Filter and Global Nearest Neighbour data association. This creates multiple objects being tracked in space relative to the host vehicle. In this research, no V2V communication is utilized, and a MIO is always selected.

Different strategies to associate radar detections with information received from different V2V sources are tested in [20]. Measurement-to-measurement (M2M) association was found to be error-prone, but straightforward. A T2T association algorithm is presented, which yields better results. This algorithm only uses a GPS sensor to track the host vehicle and uses a KF to filter the radar data. This results in shortcomings in determining the track of the host vehicle, and excessive processing required for filtering the radar data.

Multiple V2V senders are matched to multiple radar objects to acquire extra information about the surrounding objects in [21]. Tests have been executed with M2M and T2T association algorithms. Associating the data with T2T was found to yield better results. In this research only a single V2V sender is present, rendering this method not usable.

Therefore, a new solution has to be proposed where the V2V communication is utilized to determine whether the radar sensor detects the leading vehicle. In this situation, only a single V2V sender is considered, and the leading vehicle may or may not be detected by the radar sensor. When the radar fails to detect the leading vehicle, vehicle state estimation should rely solely on V2V communication.

3.3 Vehicle Motion Controllers

Currently, it is not yet defined what motion controllers will be used on the Twizies. To make the platform suitable for a wide range of different controllers, it needs to be investigated what types of controllers can be implemented, and what types of reference inputs these controllers require. In this section, different longitudinal and lateral vehicle motion controllers are investigated. Longitudinal controllers determine the amount of throttle or braking required for following the reference, whereas lateral controllers determine the steering wheel position.

For longitudinal control in platooning either Adaptive Cruise Control (ACC) or Cooperative Adaptive Cruise Control (CACC) is used. The main objective of these controllers is to make vehicles follow the preceding vehicle at a desired distance [22]. To do so, the relative longitudinal positions, velocities, and accelerations are required. For CACC the desired longitudinal acceleration is used to anticipate the motion of the preceding vehicle. The desired acceleration of the preceding vehicle can be used to control the acceleration of the following vehicle using feed-forward control. The required longitudinal acceleration to achieve the desired following distance is the output of the ACC and CACC controllers. These controllers use a body-fixed reference frame to the ego vehicle.

An example of lateral control is the pure pursuit controller. Here a point on a target path is taken, and with the assumption of kinematic steering the desired steering angle is calculated based on geometry [23]. The look-ahead controller is a different type of lateral controller. This controller takes the position, velocity, heading angle, and yaw rate of the leading vehicle as inputs [24]. The Stanley controller follows a path based on a lateral error and heading error [25]. To ensure stability yaw rate damping and steering damping have been added to the control law. This lateral controller requires the position, heading angle, yaw rate, and velocity as inputs. Longitudinal and lateral controllers are implemented in [26]. Here CACC is used for longitudinal control, and a path following controller is used for lateral control. The lateral controller takes the absolute position and heading angles of the leading and following vehicle and converts these to the vehicle-fixed frame of the following vehicle. Most lateral vehicle controllers use a body-fixed frame to the ego vehicle, comparable to the longitudinal controllers. Because lateral vehicle control can be implemented in various ways and its final implementation is not known as of yet, the sensor fusion system should put out all relevant vehicle states that may be used by the different types of lateral controllers. Therefore, the position and velocity of both vehicles should be available in both the relative and inertial reference frames. The heading angle and yaw rate of both vehicles should also be available in the inertial reference frame. Furthermore, the lateral acceleration of both vehicles should be available in the inertial reference frame.

3.4 Summary

This chapter describes different implementations of sensor fusion for vehicle state estimation and tracking. The most used type of fusion algorithm is the KF or a variation of it such as the EKF or the ESKF, in which multiple types of models can be used. For radar matching, T2T association is preferred over other methods because it performs better. The position and velocity of both vehicles in the relative and inertial reference frame are required as outputs to be compatible with multiple types of vehicle motion controllers. The yaw rate and heading angle, and the longitudinal and lateral accelerations of the vehicles should also be available from the vehicle state estimator.

Chapter 4

Twizy Platform

The Renault Twizies used for full-scale experiments are equipped with extra devices for autonomous driving. Actuators for steering and braking are installed. The throttle pedal signal is simulated when the autonomous system is driving, which also allows for regenerative braking. The vehicle is equipped with an Advantech ARK-3520P real-time computer [22]. This PC runs an application on the Simulink RealTime operating system. The application runs at 100 Hz and is used to process measurements and control the actuators required for autonomous driving. The sensor set of the vehicles has been extended to accommodate autonomous driving, and a communication module has been installed on the vehicle to allow for V2V communication. The real-time PC communicates to the sensors and actuators using CAN buses. The communication, sensors and software are explained in the following sections.

4.1 Communication

All Twizies have been equipped with a router device for wireless communication between vehicles. The ITS G5 standard for V2V communication is used, which is a protocol for communication between vehicles using wifi [27]. The communicated messages contain information regarding the vehicle states such as position and heading, and information regarding the vehicle dimensions. The messages are composed with the Cooperative Awareness Message (CAM) protocol [28]. The relevant contents of this protocol are listed below:

- The vehicle position represents the geographical location of the vehicle's rear axle by denoting its latitude, longitude, and altitude, which are expressed in micro degrees. Different communication fields are used to express the vehicle's position in the NED frame.
- The heading with regards to the WGS84 North in degrees. The value is scaled between 0 to 3600, with 900 denoting east, 1800 denoting south, and 2700 denoting west. The resolution is 0.1°.
- The speed is expressed as a scalar which denotes the absolute velocity of the centre of gravity of the vehicle with respect to the inertial frame in metres per second. Therefore this value will always be positive, and will not have a direction.
- Drive direction. When this value is set to 0 the vehicle is moving in the forward direction. It is set to 1 when the vehicle is driving backwards.
- Longitudinal acceleration of the centre of mass of the empty vehicle expressed in metres per second squared according to ISO 8855. A positive value denotes an increase in velocity whereas a negative value denotes a decrease in velocity. This quantity is measured in the longitudinal direction of the local reference frame fixed to the sending vehicle.
- Lateral acceleration of the centre of mass of the empty vehicle expressed in metres per second squared according to ISO 8855. Acceleration to the left-hand side of the driver is denoted as positive.
- Vertical acceleration of the centre of mass of the empty vehicle expressed in metres per second squared according to ISO 8855. Acceleration downwards is expressed as a positive value.
- Curvature. This value denotes the radius of the current vehicle trajectory in reciprocal metres according to ISO 8855. The value is set to 0 when the vehicle is driving straight. The curvature is positive when a turn to the left is made.
- Yaw rate in radians per second according to ISO 8855. This denotes the rotation around the centre of mass of the empty vehicle, with a clockwise rotation being denoted by a negative value.
- Path history. This consists of a list of up to 40 path points of the previously driven path of the vehicle. These path points are described with a latitude, longitude, and attitude from a specific reference point in micro degrees.

Usually, CAM messages are sent with a frequency of 1 to 10 Hz depending on the number of vehicles using the wireless communication [22]. However, since only two vehicles are using the wireless communication the message transfer frequency has been increased to 25 Hz. The wireless communication messages are received with a constant delay of 20 ms [22].

4.2 Sensors

For autonomous driving, accurately knowing the states of the ego-vehicle and the vehicle that is being followed is required. The sensor set of the Renault Twizies is extended with an IMU, a GNSS receiver, an odometer, and a forward-facing radar to improve the vehicle state estimation capabilities.

IMU

The longitudinal, lateral, and vertical acceleration, and roll and yaw rate are measured by a Bosch MM5.10 IMU. This sensor operates at 100 Hz [29], and communicates the measurements to the real-time PC via CAN bus. The sensor is mounted close to the centre of gravity of the vehicle underneath the driver seat [22]. Internally, the sensor filters the measurement signals with a low-pass filter of 15 Hz.

GNSS

The vehicles have been equipped with an u-blox EVK-M8T GNSS sensor to get an absolute position measurement. This sensor measures the longitude, latitude, and altitude of the vehicle with respect to the WGS84 standard in degrees [30]. The sensor communicates to the real-time PC using a serial RS-232 interface with an update frequency of 5 Hz and has an output delay of 0.18 s [22].

Odometer

The Twizy is equipped with an odometer that measures the rotational velocity of the rear axle before the differential. Because of this, the average rotational velocity of the rear wheels is measured. The speed of the vehicle can be determined using the gear ratio, the radius of the wheels, and the rotational velocity of the axle.

Radar

A Bosch MMRevo14 is mounted on the front of the vehicle. This sensor measures the relative distance and velocity in longitudinal and lateral directions [31]. The radar can detect up to 32 objects and has two sets of antennas. The main antenna can detect objects up to 160 m with a small FoV angle. The elevation antenna has a range of up to 36 m and has a larger FoV angle. The FoV angle is therefore range dependent and is shown in Table 4.1. The radar sensor classifies the detected objects in the following categories: pedestrian, motorcycle, car, construction element, and unknown [22]. The cycle time of the radar is 60 ms, and the measurements are communicated to the real-time PC via CAN.

Horizontal field of view [°]	Range [m]
± 6	160
± 9	100
± 10	60
± 25	36
± 42	12

Table 4.1: Horizontal radar FoV as a function of range [31].

The update frequency, accuracy, and range of the different sensors are shown in Table 4.2.

Sensor	Range	Accuracy	Frequency
Radar	0.36 - 160 m $\pm$ 10° (60 m) $\pm$ 25° (36 m) $\pm$ 42° (12 m)	0.12 m 0.11 m/s $\pm$ 0.3 °	16.7 Hz
GNSS	-	2.0 m 0.05 m/s $\pm$ 0.3 ° heading accuracy	5 Hz
IMU	$\pm$ 4.2 g 0-163 °/s	0.01 g 0.1 °/s	100 Hz

Table 4.2: The range, accuracy and measurement frequency of the IMU [29], GNSS sensor [30], and radar [31].

4.3 Overview of Autonomous Driving System

Figure 4.1 shows the overview of the current implementation which is used for autonomous driving on the Renault Twizies. The GNSS, IMU, and odometer are used for tracking the states of the host vehicle. The

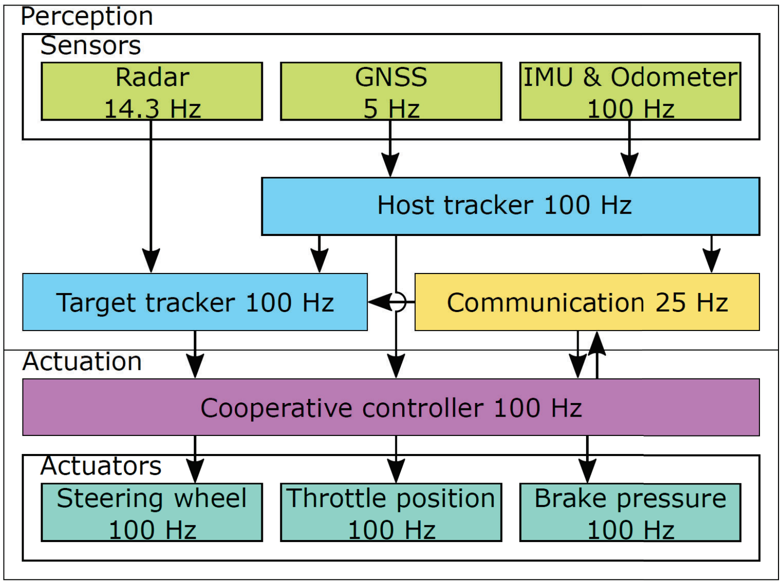


Figure 4.1: Overview of the autonomous driving system [32].

target is tracked using the outputs of the host-tracker, the radar sensor, and the communicated information. The cooperative controller uses the vehicle state information of the host and the target, and the communicated information. At last, the cooperative controller calculates the steering, brake and throttle inputs.

4.3.1 Host Tracking

Not all kinematic states can be determined from direct sensor measurements, such as the heading angle. The Host-tracking block is developed to estimate the relevant kinematic states and their covariances [22]. This is done with an ESKF, which uses the integrated information from the motion sensors and compares this to the GNSS location whenever the GNSS sensor gives an update. The radar sensor is not used when determining the vehicle state of the following vehicle. When the distance between the vehicles is measured, this can be used to correct errors in the GNSS measurements done on both vehicles.

4.3.2 Target tracking

In the Target-tracking block the vehicle states of the preceding vehicle are determined. The current solution has the following shortcomings. Currently, the target tracker consists of two parts where the V2V communication and the radar measurements are used to determine the vehicle states of the leading vehicle separately. With the V2V communication and the output of the Host-tracking, the relative distance and velocity between both vehicles are determined without considering the communication delay. From the V2V communication, it is also determined whether the leading vehicle is in the FoV of the radar sensor. When the preceding vehicle is detected, the relative distance and velocity between the vehicles are also calculated from the radar measurements. Moreover, only the relative distance and position between the vehicles are determined. The absolute vehicle states of the leading vehicle, such as velocity, acceleration and position, are directly taken from the V2V communication without considering the communication delay.

4.3.3 Cooperative controller

The cooperative controller determines the motion setpoints of the vehicle. Based upon the target tracking, host tracking and communication, the cooperative controller determines setpoints for the steering wheel, throttle position, and brake pressure to control the longitudinal and lateral motion of the vehicle.

For longitudinal control currently cruise control, ACC and CACC are implemented and operational. When driving, one of these three can be selected to control the longitudinal motion of the vehicle. The cruise controller tries to follow a velocity set point. With ACC a velocity set point is followed, but when an object or vehicle is detected in front the vehicle is slowed down. While using ACC the vehicle replicates the velocity of the leading vehicle. CACC uses the desired acceleration from the leading vehicle that is sent over V2V. The cruise control, ACC and CACC control the longitudinal motion of the vehicle by giving a throttle input to the vehicle. For lateral control, a pure pursuit controller is implemented currently. This pure pursuit controller is not tested and validated and therefore the lateral control is not operational.

4.4 Proposed Changes to the System

In this project, the host and target trackers are improved, as the current host and target trackers have shortcomings. Incorporating the radar sensor increases the accuracy of the host tracker because the radar measures the relative distance and velocity between the vehicles directly. This is more accurate than deriving the relative distance and velocity between the vehicles from the GNSS and IMU measurements. The current system is also not able to determine the vehicle states when the V2V communication is not available. It tracks an MIO but does not derive vehicle states from that. Based on whether the V2V communication is active and the radar sensor detects the leading vehicle the host and target trackers are adjusted. When only the V2V communication is active, the vehicle states of the leading vehicle will be determined purely from the communicated information. When the radar sensor detects the leading vehicle, but the V2V communication is inactive, the target tracking is done using solely the radar sensor measurements. When both are active, the leading and following vehicle states will be estimated by fusing the onboard sensors of the vehicle with the radar sensor. This combines the host and target tracker when the radar sensor detects the leading vehicle.

4.5 Summary

The Renault Twizies used as a testing platform for this project are equipped with router devices to enable V2V communication. Vehicle state information is communicated between vehicles at 25 Hz with a constant delay of 0.02 s. An IMU, GNSS, Odometer and Radar sensor have been mounted to the Renault Twizies to enhance the vehicle state estimation capabilities. Currently, separate host and target trackers are used to determine the vehicle states of the leading and following vehicles. In this project, the host and target trackers are combined.

Chapter 5

Simulation Environment

To develop the sensor fusion system without requiring the test vehicles, a simulation environment has been set up. In this simulation, two vehicles, simulated with a dynamic single-track model, drive a predetermined track. Using sensor models, data can be generated without requiring full-scale experiments with vehicles which increases the speed of the development. In contrast to experiments, ground truth vehicle state information is available directly from the single-track models. In simulation, different scenarios can be considered without safety concerns. Moreover, the simulation is a controlled environment in which certain effects, such as vehicle trajectories or sensor inaccuracies, can be isolated. The overview of the simulation environment model is shown in Figure 5.1.

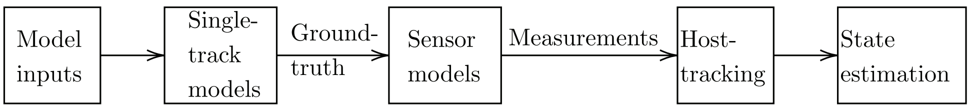


Figure 5.1: Systematic overview of the Simulation Environment model.

The model inputs define the path of the vehicles. The single-track models take the inputs to determine ground truth vehicle state data. Using sensor models realistic measurements are mimicked. These measurements are used by the state estimation algorithm. The following sections describe the various components of the Simulation Environment model.

5.1 Single-Track Model

The dynamic single-track vehicle model is shown in Figure 5.2. The model uses linear tire forces, relying on the assumption that during normal driving the tire grip is never saturated. A dynamic single-track model is used to replicate an offset between the heading angle of the vehicle and the angle of the vehicle trajectory. This offset angle is the vehicle side slip angle β. The inputs of the single-track model consist of the longitudinal acceleration, a_x, and the steering angle δ. The longitudinal velocity, v_x, is determined using

$$v_x(t) = v_{x,0} + \int_0^t a_x(t)\, dt, \tag{5.1}$$

where $v_{x,0}$ is the initial velocity and t denotes the time. The front wheel slip angle α_f is calculated using

$$\alpha_f = \delta - \arctan(\frac{v_y + a\dot{\psi}}{v_x}), \tag{5.2}$$

where a is the length from the front wheel to the centre of gravity (CoG), v_y is the lateral vehicle velocity, and $\dot{\psi}$ is the yaw rate. The rear wheel slip angle α_r is calculated using

$$\alpha_r = -\arctan(\frac{v_y - b\dot{\psi}}{v_x}), \tag{5.3}$$

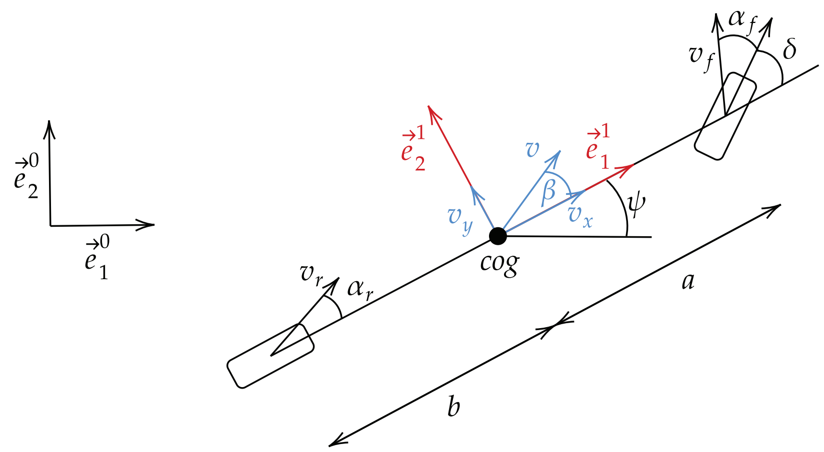


Figure 5.2: Schematic of the dynamic single-track vehicle model.

where b is the length from the rear wheel to the CoG of the vehicle. The front and rear lateral tire forces, $F_{y,f}$ and $F_{y,r}$, are calculated with

$$F_{y,f} = \alpha_f C_{\alpha f} \tag{5.4}$$

$$F_{y,r} = \alpha_r C_{\alpha r}, \tag{5.5}$$

where $C_{\alpha f}$ and $C_{\alpha r}$ are the cornering stiffness of the front and rear axles respectively. The single-track model is given by

$$\begin{cases} \dot{v}_y = \frac{1}{m}(F_{y,f}\ \cos(\delta) + F_{y,r}) - v_x\ \dot{\psi} \\ \ddot{\psi} = \frac{1}{I_{zz}}(a\ F_{y,f}\ \cos(\delta) - b\ F_{y,r}) \\ \dot{x} = v_x\ \cos(\psi) - v_y\ \sin(\psi) \\ \dot{y} = v_x\ \sin(\psi) + v_y\ \cos(\psi), \end{cases} \tag{5.6}$$

where $\dot{v}_y$ denotes the change of lateral velocity, $\ddot{\psi}$ denotes the yaw acceleration, ψ denotes the heading angle, $\dot{x}$ and $\dot{y}$ denote the change of position with respect to the inertial reference frame, m denotes the vehicles mass and I_{zz} denotes the yaw moment inertia. The parameters used in the single-track model are shown in Table 5.1. The wheelbase is known from manufacturer information and the mass of the vehicle includes the driver. The cornering stiffness of both axles and the inertia are estimated using rules of thumb [33]. For standard vehicles, the cornering stiffness of the front axle is lower than that of the rear axle to induce understeering behaviour. The yaw inertia of a vehicle is estimated with a rule of thumb using

$$\frac{I_{zz}}{ml^2} \approx 0.24. \tag{5.7}$$

The location of the CoG of production vehicles is estimated using

$$0.35 < \frac{a}{l} < 0.48. \tag{5.8}$$

For simplicity, this fraction is chosen to be equal to 0.40 giving a value for a. The length from the CoG to the rear axle is calculated using

$$b = l - a. \tag{5.9}$$

The two vehicles are simulated using equal single-track models. The inputs given to vehicle 1 are delayed by 2 seconds for vehicle 2, so vehicle 2 follows vehicle 1 with a following time of 2 seconds. This results in two sets of vehicle states, $[x_1, y_1, v_{x1}, a_{x1}, \psi_1, \dot{\psi}_1]$ for the leading vehicle, and $[x_2, y_2, v_{x2}, a_{x2}, \psi_2, \dot{\psi}_2]$ for the following vehicle.

Parameter	Symbol	Value
Mass	m	530 kg
Wheelbase	l	1.686 m
Cornering stiffness front axle	$C_{\alpha f}$	30000 N/deg
Cornering stiffness rear axle	$C_{\alpha r}$	50000 N/deg
Inertia	I_{zz}	331 kgm^2

Table 5.1: Single-track model parameters.

5.2 Sensor Models

The vehicle states are known precisely in simulation. In a practical situation, this is not the case because sensor measurements are not perfect. Sensors can have offsets and inaccuracies, and measurements are prone to noise. To simulate sensors, ground truth vehicle states are fed through sensor models. How these sensor models are created by replicating measurement data and using manufacturer information is discussed in this section. The simulated measurement results are given to the Host-tracking block to replicate the inputs of the state estimation in a real-life situation.

During standstill, the measured accelerations and angular velocities of the vehicle should be 0, except for the vertical acceleration which should be -9.81 m/s. Figure 5.3 shows sensor noise and bias on measurements from the Bosch MM5.10 IMU during standstill. The acceleration in the vertical direction is normalized by adding the gravitational constant. By taking the mean of the sensor output during this period, the bias is found. The

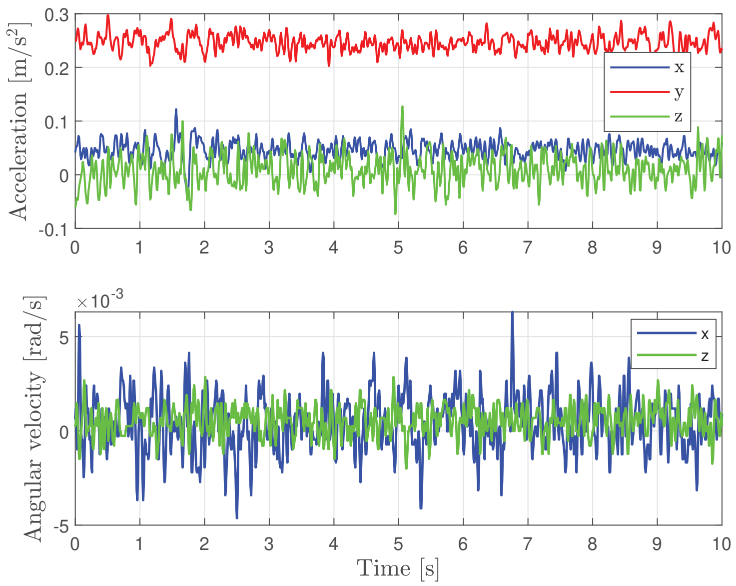


Figure 5.3: IMU measurements during standstill with normalized vertical acceleration.

noise term is found by taking the standard deviation of the signal. The characteristics of the sensor output for one specific experiment are shown in Table 5.2. After restarting the vehicle and sensors, the biases on the IMU measurements can vary.

Measurement	Symbol	Noise variance	Bias
Longitudinal acceleration	a_x	0.0159 m/s^2	0.0460 m/s^2
Lateral acceleration	a_y	0.0152 m/s^2	0.3976 m/s^2
Vertical acceleration	a_z	0.0289 m/s^2	0.0090 m/s^2
Roll velocity	w_x	0.0015 rad/s	0.0005 rad/s
Yaw velocity	w_z	0.0008 rad/s	0.0004 rad/s

Table 5.2: IMU sensor measurement characteristics for one specific experiment.

Setting	Value
Latitude reference	0.897935330867767°
Longitude reference	0.095802396535873°
Altitude reference	0 m
Horizontal position accuracy	2.0 m
Vertical position accuracy	2.0 m
Velocity accuracy	0.1 m/s
Decay factor	0.999

Table 5.3: Settings for the GPS sensor Simulink block.

The IMU sensor is modelled in simulation by adding the bias and white noise with the magnitudes shown in Table 5.2 to the ground truth vehicle state information. Because a single-track model does not consider vehicle roll, the measured roll angle of the vehicle is set to 0 rad.

The GNSS sensor is simulated using the GPS block from the Navigation Toolbox [34]. This block takes the x and y-position and the x and y-velocity in the inertial reference frame of the vehicle as inputs. The outputs of the GPS block are the latitude, longitude, and altitude, the velocities in the navigation system, and the heading. The model uses a reference position in latitude longitude altitude frame, equal to the reference position used in Host-tracking. The accuracy of the position and velocity measurements are taken from manufacturer information [30]. The decay factor determines the ratio between white noise and a slowly wandering error for the simulated sensor error. The standard decay factor from the toolbox is taken. The used settings are showing in Table 5.3.

The odometer measures the rotational velocity of the rear axle. With the gear ratio and the wheel radius, the velocity of the vehicle can be calculated. Figure 5.4 shows the velocity measurement of the odometer during constant velocity driving. It can be seen that this velocity measurement does not show a white noise

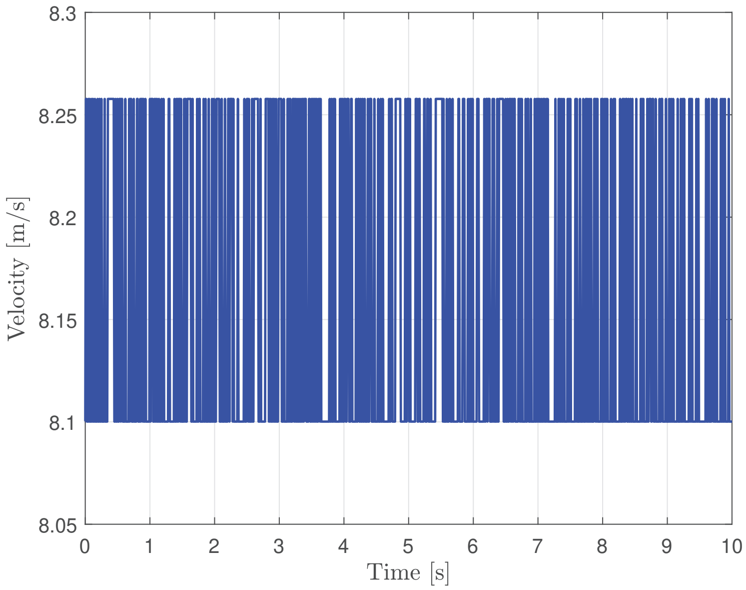


Figure 5.4: Odometer measurement during constant velocity driving.

characteristic such as the IMU, but instead oscillates between two values. The sensor has a resolution of slightly over 0.15 m/s. This is modelled by adding noise with a variance of 0.1 times the resolution to the ground truth velocity and applying the same resolution to the signal in the simulation.

The Bosch MMRevo14 radar sensor takes position and velocity measurements relative to its own position. Because the sensor is mounted to vehicle 2, the sensor takes measurements in the vehicle fixed reference frame of the following vehicle. To model the radar sensor, the radar sensor measurements have to be calculated from the ground truth data. Figure 5.5 shows both vehicles 1 and 2, and their position vectors in inertial frame $\vec{e}^0$, $\vec{p}_1^0$ and $\vec{p}_2^0$, respectively. The position vector of a vehicle points to the middle point on the rear axle. The vector $\vec{p}_1^2$ denotes the position of the leading vehicle in the relative reference frame fixed to the following vehicle.

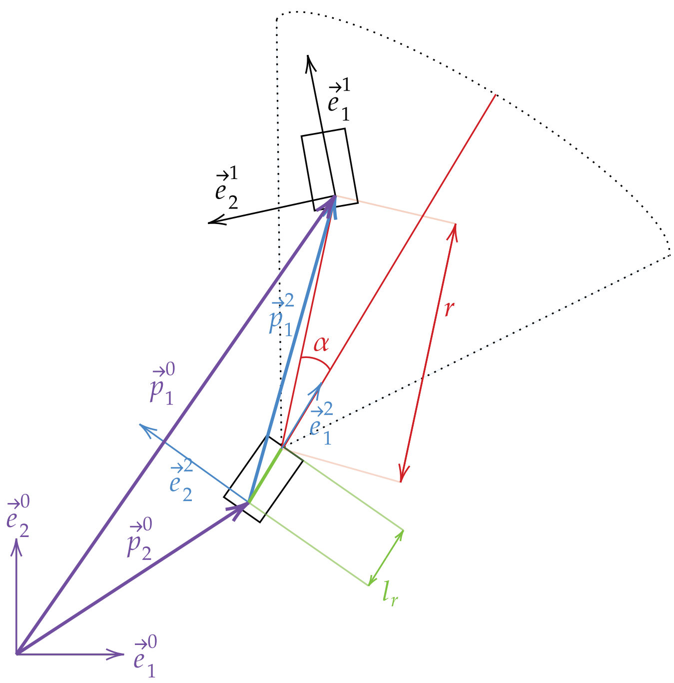


Figure 5.5: Representation of the used reference systems and the radar measurement.

Measurement	Symbol	Noise variance
Relative longitudinal distance	x_r	0.12 m
Relative lateral distance	y_r	0.20 m
Relative longitudinal velocity	$v_{x,r}$	0.11 m/s
Relative lateral velocity	$v_{y,r}$	0.20 m/s

Table 5.4: Radar measurement characteristics [31].

Radar measurements are calculated from the ground truth vehicle state data using the coordinate system transformation discussed in Section 2.2.3. The radar measurement consists of the relative longitudinal and lateral distances and velocities. In simulation, the radar only detects the other vehicle. Therefore, no radar matching algorithm is applied in the simulation environment. The relative longitudinal distance, x_r, is calculated using

$$x_r = \cos(\psi_2)(x_1 - x_2) + \sin(\psi_2)(y_1 - y_2). \tag{5.10}$$

The relative lateral distance, y_r, is calculated using

$$y_r = -\sin(\psi_2)(x_1 - x_2) + \cos(\psi_2)(y_1 - y_2). \tag{5.11}$$

The relative longitudinal velocity, $v_{x,r}$, is determined by taking the time derivative of the relative longitudinal distance. This gives

$$\begin{aligned} v_{x,r} = &-\dot{\psi}_2 \sin(\psi_2)(x_1 - x_2) + \cos(\psi_2)(\cos(\psi_1)v_{x1} - \cos(\psi_2)v_{x2}) \\ &+ \dot{\psi}_2 \cos(\psi_2)(y_1 - y_2) + \sin(\psi_2)(\sin(\psi_1)v_{x1} - \sin(\psi_2)v_{x2}). \end{aligned} \tag{5.12}$$

The relative lateral velocity, $v_{x,r}$, is by taking the time derivative of the relative lateral distance, giving

$$\begin{aligned} v_{y,r} = &-\dot{\psi}_2 \cos(\psi_2)(x_1 - x_2) - \sin(\psi_2)(\cos(\psi_1)v_{x1} - \cos(\psi_2)v_{x2}) \\ &- \dot{\psi}_2 \sin(\psi_2)(y_1 - y_2) + \cos(\psi_2)(\sin(\psi_1)v_{x1} - \sin(\psi_2)v_{x2}). \end{aligned} \tag{5.13}$$

White noise is added to the radar measurements to model inaccuracies in the radar sensor outputs. The noise terms are shown in Table 5.4. The sensor accuracy is specified in range and angle. Because of this, the accuracy of lateral measurements is dependent on the longitudinal distance. During simulations, the longitudinal distance remains below 40 m, which results in the maximal lateral error specified in Table 5.4.

The sensor measurements are an input to the Host-tracking algorithm. This algorithm has the following inputs: the longitudinal, lateral, and vertical accelerations, the yaw and pitch rates, the North, East, and Down positions and velocities, the latitude, the longitudinal velocity, and whether the GPS sensor has given an update. The Host-tracking algorithm uses a model based on quaternions, which is an alternative method to represent the rotation of a frame in 3-dimensional space. The input matrix of the model consists of a set of noises and uses the accelerations in the inertial frame to interpolate and correct the GPS measurements between updates. The GPS measurements are sampled at 5 Hz, and the other sensors are put in at 100 Hz. The Host-tracking block gives the estimated vehicle states at 100 Hz. In the simulation, the communication between the vehicles is replicated by delaying the outputs of the Host-tracking block of vehicle 1 with the communication delay of 0.02 s. The signals are also resampled to a frequency of 25 Hz to imitate the sending frequency.

5.3 Simulation Results

To demonstrate the outputs of the modelled sensors, the Host-tracking algorithm, and the ground truth vehicle state data, a simulation is done where the single-track models drive in a circular path. The inputs for this are a constant steering angle of 5° and a constant velocity of 10 m/s. The simulation time is adjusted so that 2 full circles are driven.

The errors of the simulated GPS sensor are shown in Figure 5.6. It can be seen that the errors show a saw tooth pattern. This occurs because the GPS sensor produces an output at a frequency of 5 Hz, whereas the ground truth data is available at 100 Hz, so the ground truth changes while the GPS output remains constant. Because of this, the initial error between the signals when a new GPS update is received is smaller than just before a new GPS update is received. The errors for both the position measurements are of similar magnitude, which is also the case for the measurement errors in the velocities. The heading angle of the vehicle shows a negative error with a saw tooth pattern. This is explained by the constant rotation of the vehicle to the left. Because of this, the error is always negative just before a new GPS measurement arrives because the vehicle has rotated more in between the GPS measurement updates.

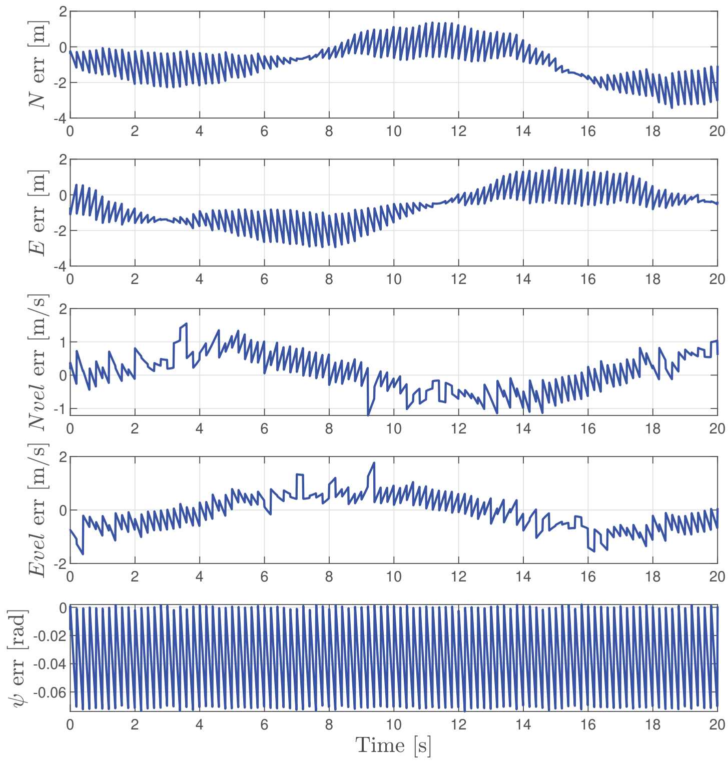


Figure 5.6: Measurement errors of the simulated GPS sensor.

Figure 5.7 shows the GPS measurements, the Host-tracking estimation and the ground truth in the inertial frame for a section of the driven circle. The GPS and Host-tracking outputs are not equal to the ground truth and vary between the two laps. The GPS measurement shows a path that is not smooth because of the

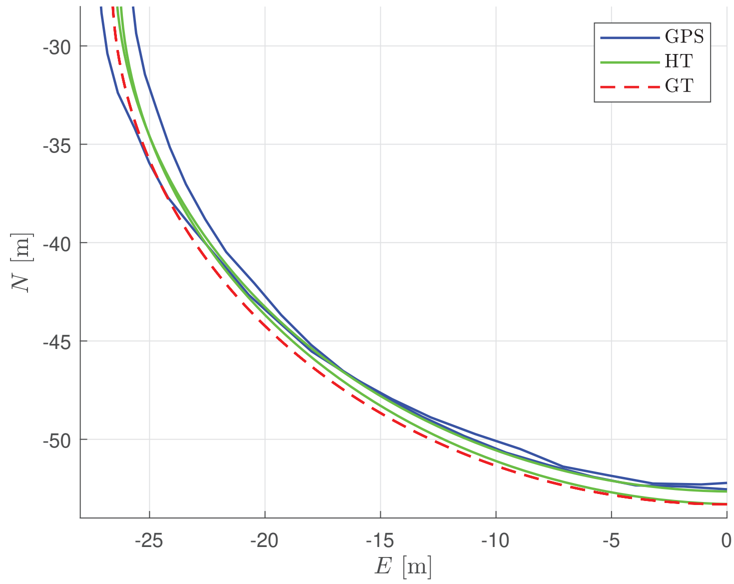


Figure 5.7: Part of the driven path in the inertial frame with ground truth, GPS output and Host-tracking output when driving two laps on a circular path.

lower sampling frequency of the GPS sensor. The Host-tracking output is smooth due to its higher sampling frequency. It can also be seen that the GPS measurement shows a larger deviation from the ground truth than the Host-tracking output because the Host-tracking algorithm corrects the GPS errors with the kinematic sensors.

The errors of the individual sensors and the Host-tracking algorithm when driving in the circular path are shown in Figure 5.8. It can be seen that Host-tracking filters the position, velocity, and heading data. The longitudinal acceleration and yaw rate outputs of Host-tracking are equal to the sensor measurements because Host-tracking uses these signals to estimate the other states and does not filter these signals. The maximal position error of the Host-tracking is smaller than the error in the GPS measurement for both the North and East coordinates. The velocity error of Host-tracking is very small at 0.005 m/s. In this simulation, the vehicle drives with a constant velocity, which makes the velocity estimate more accurate than for a varying velocity. Because the yaw rate of the vehicle is constant, the heading also is easier to estimate for the Host-tracking algorithm. Because of this, the maximal estimation error is 0.0016 rad, which is 0.09°.

The ground truth radar measurements and the radar outputs with inaccuracies are shown in Figure 5.9. The radar measurements remain constant during the simulation because both vehicles drive exactly the same path with a constant delay. The relative velocities between the vehicles remain 0 m/s because the vehicles do not move relatively to each other. The radar error shows the low-frequency white noise which is added to the measurement signal.

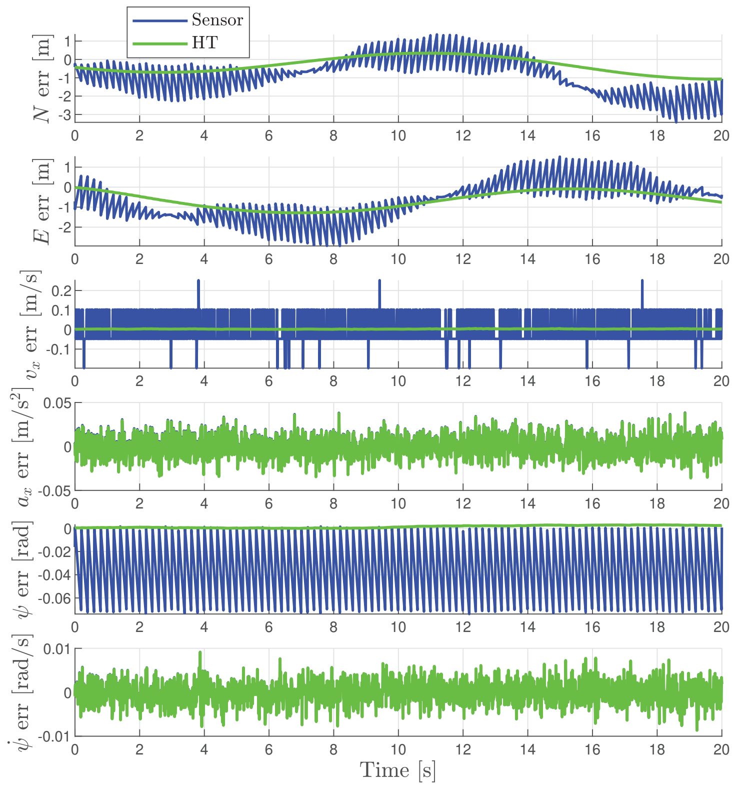


Figure 5.8: The error between the ground truth and the modelled sensors and Host-tracking in simulation with a circular path.

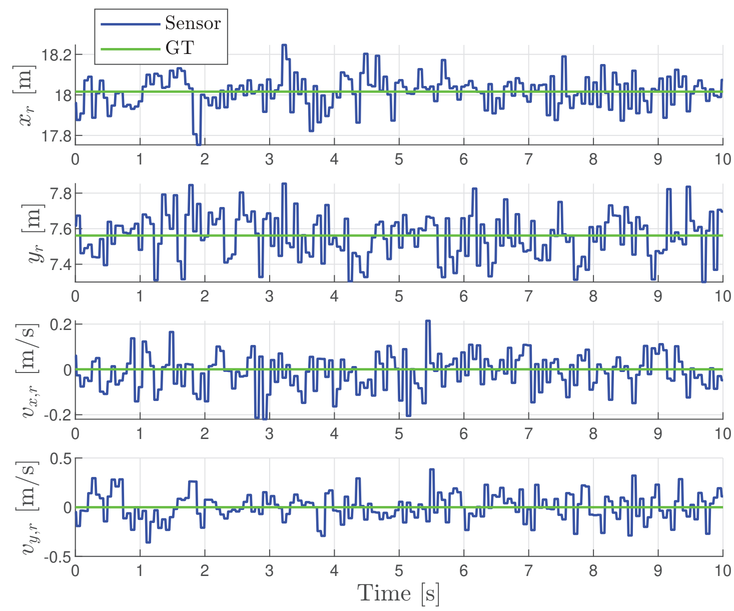


Figure 5.9: Comparison between the ground truth radar measurements and the modelled radar sensor for a circular path.

5.4 Summary

A simulation environment, containing two single-track models is constructed to allow for offline testing. Ground truth vehicle state data is calculated directly with the single-track models. Using manufacturer information and experimental data, sensor models are created. These sensor measurements are fed through the Host-tracking algorithm. Simulations are executed to determine the errors of the Host-tracking algorithm. Host-tracking estimates the position with an accuracy of 0.77 m when driving in a circle with a constant velocity and steering angle. The heading angle is estimated with an accuracy of 0.09 $^{\circ}$, and the velocity estimate is accurate up to 0.005 m/s when the vehicles drive in a circle with constant velocity.

Chapter 6

Vehicle State Estimation in Simulation

The primary objective of the vehicle state estimation system is that the required vehicle states, which are taken as reference inputs of the vehicle motion controllers, are available at 100 Hz without delay. These outputs include the position of the vehicles in absolute and relative reference frames, the relative and absolute velocity, the longitudinal and lateral acceleration, the yaw rate and the heading angle of both vehicles.

The system runs on the following vehicle, and therefore the output of the Host-tracking algorithm is taken directly to have the states of vehicle 2 at 100 Hz without delay. The states of vehicle 1 are communicated through V2V at 25 Hz with a delay of 0.02 s. The radar sensor mounted to the following vehicle measures the relative longitudinal and lateral positions and velocities between the vehicles when the leading vehicle is detected at 16.7 Hz. It is assumed that the radar sensor does not have a delay.

Three possible situations can occur depending on whether the radar sensor detects the leading vehicle and the V2V communication is active. Because the inputs of the state estimation system change depending on this, the state estimation is divided into three parts:

- Situation 1: Only the V2V communication is active
- Situation 2: Only the radar sensor is active
- Situation 3: Both the V2V communication and the radar sensor are active

When no V2V messages are received, and the radar sensor does not detect the leading vehicle, the states of the leading vehicle cannot be estimated. This means that the vehicle is not able to follow its predecessor autonomously in this situation. A fallback system should be developed for this case, but that is not within the scope of this research project.

6.1 Situation 1: Only Communication

Figure 6.1 shows the strategy for when only V2V communication is available.

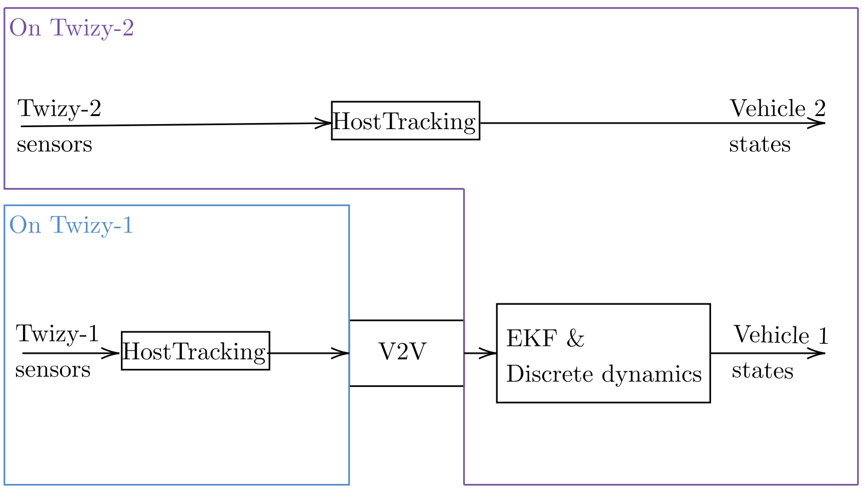


Figure 6.1: Overview of the state estimation system when only the V2V communication is active.

The vehicle states of the leading vehicle are estimated locally using Host-tracking. Through the V2V communication, the estimated states are communicated to the following vehicle. To estimate the states of the first vehicle onboard the following vehicle, an EKF is used in combination with discrete dynamics (DD). The EKF is used to estimate the vehicle states from the time at which the V2V message is sent. This means that the EKF estimates the vehicle states from the delay period in the past. The EKF is also used to resample the 25 Hz input signal to a 100 Hz output signal. To compensate for the V2V communication time delay DD is used. The vehicle states of vehicle 2 are estimated directly from the onboard sensors by the Host-tracking algorithm.

6.1.1 Extended Kalman Filter

A sensor fusion algorithm requires a model to describe the system's behaviour [6]. The different sensor measurements are compared to this model to determine the vehicle states. A vehicle model needs to be chosen from which the required vehicle information required for the motion controllers discussed in Section 3.3 can be derived. Because the preceding vehicle can be a different type of car, the model should not contain vehicle parameters. For this purpose, the unicycle model is chosen as it does not include any vehicle parameters. The unicycle model is a simple kinematic method for describing a vehicle's motion and is shown in Figure 6.2. The

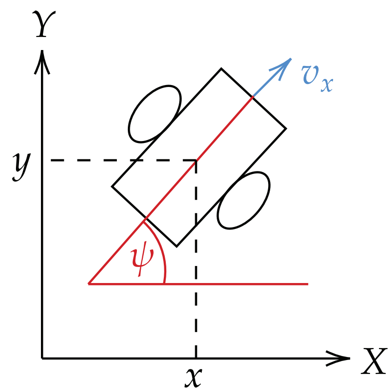


Figure 6.2: Representation of the unicycle model with position coordinates x and y, velocity v_x and heading angle ψ.

unicycle model describes a vehicle whose single wheel cannot steer. The full vehicle is rotated to steer. The model does not incorporate forces, which makes this a kinematic model. The inputs of the model are the longitudinal acceleration a_x and the yaw rate $\dot{\psi}$. The state evolution system of the unicycle model, f, is described as

$$f = \begin{cases} \dot{x} = v_x \, cos(\psi) \\ \ddot{y} = v_x \, sin(\psi) \\ \dot{v}_x = a_x \\ \dot{\psi} = \omega_z, \end{cases} \tag{6.1}$$

where x and y are the vehicle position in the inertial reference frame, v_x is the longitudinal velocity, and ψ is the heading angle of the vehicle. By using this vehicle model, the situation is simplified to 2D where elevation changes are not taken into account. This could result in large estimation errors when vehicles are driven on roads with high inclination angles. Argument (t) is left out for readability. The measurements of the system are linear and can be written as

$$c = \begin{bmatrix} x_m \\ y_m \\ v_{x,m} \\ \psi_m \end{bmatrix}, \tag{6.2}$$

where subscript m denotes the measurement of a specific state. State system f and measurement system c are used in the EKF discussed in Section 2.1 to estimate the states of the unicycle model. The input vector u consists of the longitudinal acceleration and yaw rate of the vehicle.

The covariance matrices for the process noise Q and the measurement noise R are tunable parameters which influence the performance of the EKF [35]. When the process noise covariance matrix is made small, the

EKF values the model evolution more. This results in a smoother output of the EKF, but it also results in a slower responding EKF. If the values in the process noise covariance matrix are increased, the EKF trusts the measurements more. Due to the noisy characteristic of measurements, this means that the output of the EKF shows more noise. It is assumed that the noise terms in the measurement are not correlated. Because of this, the measurement and process noise matrices only have entries on the diagonal. The initial process noise matrix for this situation, $Q_{0,1}$, is determined by taking the maximum change in a specific state under normal conditions. The maximal velocity of a Renault Twizy is 80 km/h [36], therefore maximal change in either the x or y-position is 22.2 m in one second. The maximal change in velocity is determined by the vehicle's achievable acceleration. Risky driving results in larger accelerations in the vehicle. During normal driving, the longitudinal acceleration does not exceed 0.4 g, which is around 4 m/s^2 [37]. Assuming no side slip in a vehicle, at 10 m/s with a lateral acceleration of 4 m/s the vehicle's yaw rate is 0.4 rad/s. The initial process noise matrix is expressed as

$$Q_{0,1} = \begin{bmatrix} 22.2T_s & 0 & 0 & 0 \\ 0 & 22.2T_s & 0 & 0 \\ 0 & 0 & 4T_s & 0 \\ 0 & 0 & 0 & 0.4T_s \end{bmatrix}, \tag{6.3}$$

where T_s is the sample time of 0.01 s. The initial measurement noise matrix for when only communication is used, $R_{0,1}$, is determined by the maximal errors of the measurements. From the Host-tracking simulation, the maximal position error was found to be 1 m in the ideal case. The velocity and heading angle errors were negligible but are taken to be 0.1 m/s and 0.01 rad initially. The initial measurement noise matrix is written as

$$R_{0,1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0.1 & 0 \\ 0 & 0 & 0 & 0.01 \end{bmatrix}. \tag{6.4}$$

By feeding the EKF undelayed information at 25 Hz and comparing its outputs to the ground truth the performance can be asserted. By tuning the noise matrices a balance is found between responsiveness, noise and error.

6.1.2 Discrete Dynamics

The information required to estimate the current vehicle states of the leading vehicle is not available due to the communication delay. To estimate the vehicle states without delay, a state prediction has to be done using the delayed vehicle states with DD. By assuming that the inputs of the unicycle model remain equal during this delay period, a projection can be made. This assumption relies on the fact that the acceleration and yaw rate of a vehicle change slowly during normal driving conditions [38]. By discretizing the system and integrating the states the state projection can be made [39]. For a linear discrete system with sampling time k, the states x after delay h can be calculated using

$$x(k+h) = e^{Ah}x(k) + \int_0^h e^{As}ds\ Bu(k), \tag{6.5}$$

where A is the state flow matrix, B is the input matrix and u is the input. For non-linear systems, this cannot be used but the partial integral of the state flow f is taken over time. The estimation of current unicycle states is given as $[\hat{x}, \hat{y}, \hat{v}_x, \hat{\psi}]$, and is calculated from the output states of the EKF derived from the communication $[x_c, y_c, v_{x,c}, a_{x,c}, \psi_c, \dot{\psi}_c]$. The current estimation of the x-position $\hat{x}$ is calculated using

$$\hat{x} = x_c + (a_c\cos(\psi_c) + a_c\cos(\psi_c + h\dot{\psi}_c) - v_{x,c}\dot{\psi}_c\sin(\psi_c) + (a_ch + v_{x,c})\dot{\psi}_c\sin(\psi_c + h\dot{\psi}_c))/\dot{\psi}_c^2, \tag{6.6}$$

where h is the delay period. The current estimation of the y-position $\hat{y}$ is calculated using

$$\hat{y} = y_c + (v_c\dot{\psi}_c\cos(\psi_c) - (a_ch + v_{x,c})\dot{\psi}_c\cos(\psi_c + h\dot{\psi}_c) + a_c(-\sin(\psi_c) + \sin(\psi_c + h\dot{\psi}_c)))/\dot{\psi}_c^2. \tag{6.7}$$

The current velocity estimation $\hat{v}_x$ is calculated using

$$\hat{v}_x = v_{x,c} + a_{x,c}h. \tag{6.8}$$

The current heading angle estimation $\hat{\psi}$ is calculated using

$$\hat{\psi} = \psi_c + \dot{\psi}_ch. \tag{6.9}$$

When the yaw rate is close to zero, the x and y-position estimates become infinite due to the division by the yaw rate. Therefore, when the yaw rate is minimal, it is assumed that the orientation angle of the vehicle remains constant. Because of this, the estimated x and y-positions are expressed as

$$\hat{x} = x_c + (v_{x,c} + \frac{a_{x,c}h}{2})\cos(\psi_{x,c}), \tag{6.10}$$

and

$$\hat{y} = y_c + (v_{x,c} + \frac{a_{x,c}h}{2})\sin(\psi_{x,c}), \tag{6.11}$$

Discrete Dynamics Results

To evaluate the best-case performance, the DD is evaluated on ground truth data from the simulation environment. The ground truth data is delayed by 0.02 s to replicate the V2V communication. Also, the delayed signals are compared with the ground truth data to investigate how large the error is when no compensation is done for the communication delay. This error is defined as the difference between the estimation and the ground truth data. Figure 6.3 shows the trapezoidal acceleration and constant steering input used for this simulation.

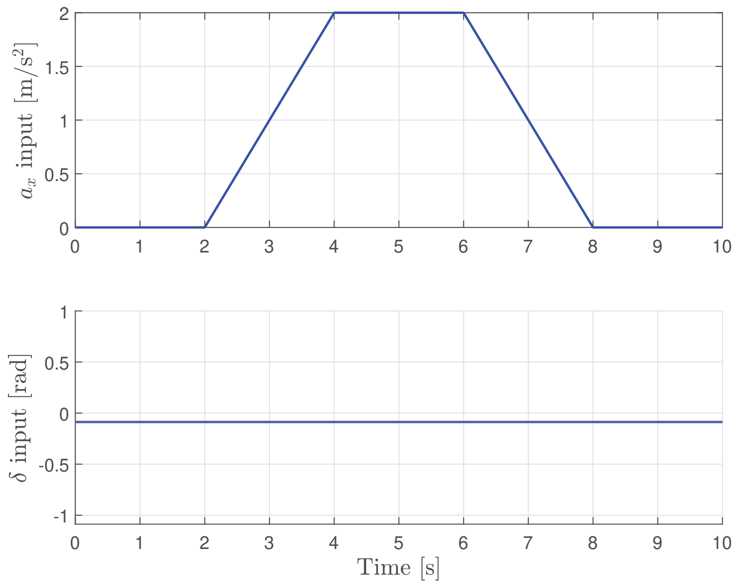


Figure 6.3: Constant steering and trapezoidal acceleration vehicle model inputs.

Using the delayed vehicle state information and the DD the correction factor is calculated for the current vehicle states. Figure 6.4 shows the correction required to compensate for the time delay. It can be seen that the compensation for the position follows a sinusoidal pattern. While driving parallel to either of the reference frame axes, the vehicle motion is only corrected in the direction of that axis. This only happens at rotation intervals of 0.5 π. When the vehicle is oriented in a different direction, both of the position coordinates have to be compensated. The profile of the velocity compensation represents the acceleration input. Due to the increase of velocity the yaw rate correction increases, because the heading angle of the vehicle changes more during the delay period.

The communicated vehicle states and the projected vehicle states using the DD are shown in Figure 6.5. The error of the communicated vehicle states is larger than the compensated vehicle states. It can seen that the acceleration input profile is represented in the error of the communicated velocity. The yaw angle shows a negative error for the communicated states due to the rotation of the vehicle to the left. The errors apparent in the communicated vehicle states are equal to the vehicle's motion during the delay period. The DD are able to compensate for the delay period with the constant steering angle and trapezoidal acceleration inputs, which are near constant.

Because this method of compensating for delay relies on the assumption that the vehicle inputs remain equal during the delay period, simulations with constantly varying inputs are executed. For this, sinusoidal acceleration and steering input are generated. The steering input has an amplitude of 5° and a frequency of 0.33 Hz. The acceleration input has an amplitude of 1 m/s^2 and has a frequency of 0.5 Hz. The inputs are

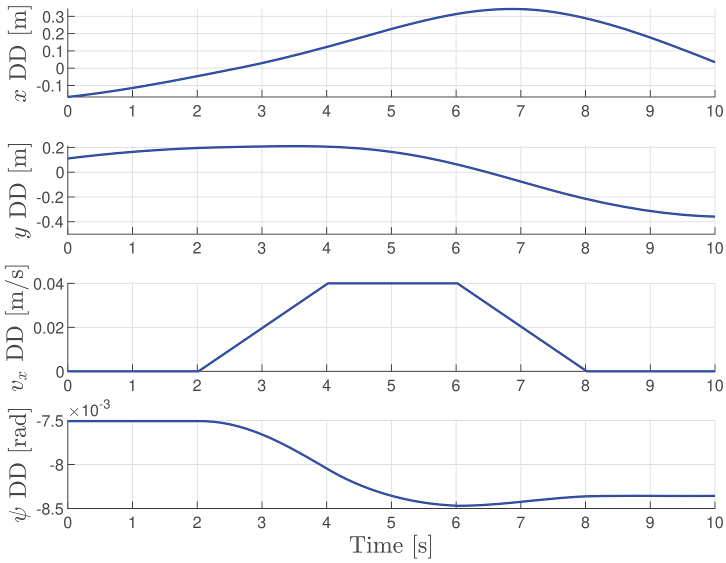


Figure 6.4: State compensation for delay for constant steering with trapezoidal acceleration input.

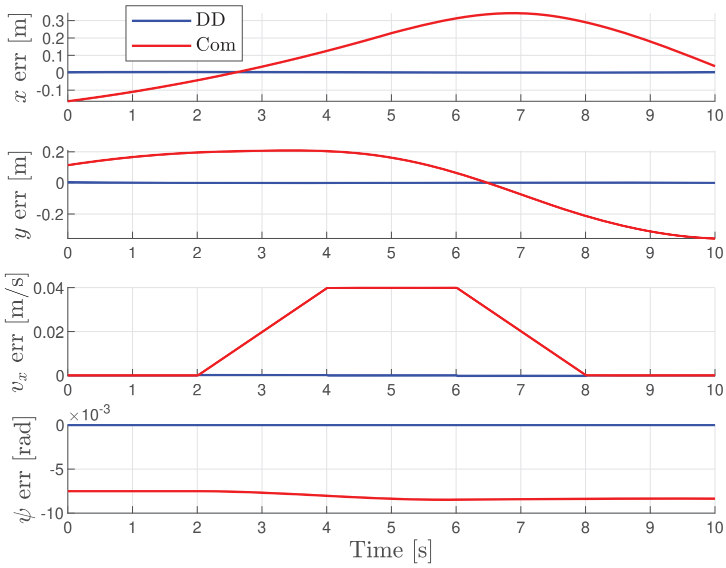


Figure 6.5: Error between the communicated vehicle states, the projected vehicle states, and the ground truth vehicle states for simulation with a constant steering and trapezoidal acceleration input.

shown Figure 6.6. Figure 6.7 shows the required compensation for the delayed vehicle states with constantly varying inputs. It can be seen that for the constantly varying inputs the compensations calculated with the DD also constantly vary. The inputs consist of two sinusoidal signals with different frequencies and the outputs of the discrete dynamics have the shape of two summed sinusoidal signals. The compensations for the position coordinates and velocity are smaller than for the constant steering input, but the compensation for the heading angle is larger.

Figure 6.8 shows the errors for both the communicated and projected vehicle states for the constantly varying inputs. For the constantly varying inputs, the maximal position error is 0.21 m for the communicated states. For the projected states, the position error is smaller than 0.01 m. Whereas the maximal velocity and heading error of the communicated states are 0.01 m/s and is 0.01 rad respectively, the errors for the compensated states are again negligible. From these results, it can be concluded that the assumption that the inputs remain constant during the delay period is valid because the delay period is small. When the delay period is increased,

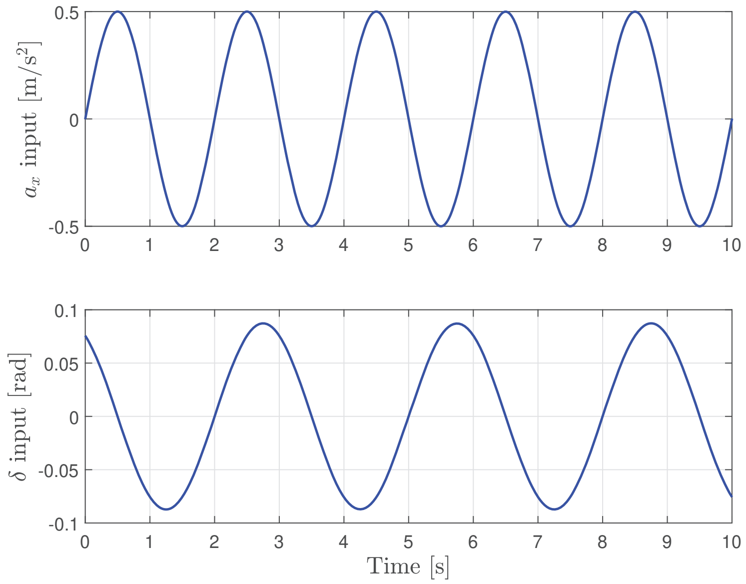


Figure 6.6: Constantly changing steering angle and acceleration vehicle model inputs.

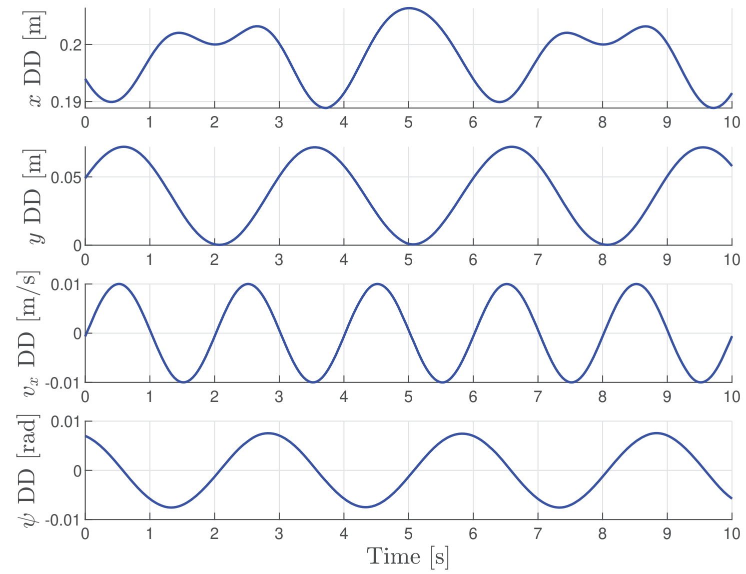


Figure 6.7: State compensation for delay for constantly varying steering and acceleration input.

the compensations calculated with the DD are less accurate and larger errors are present in the estimations. When the input signals have a larger change over time, the error of the estimation done with the DD becomes larger.

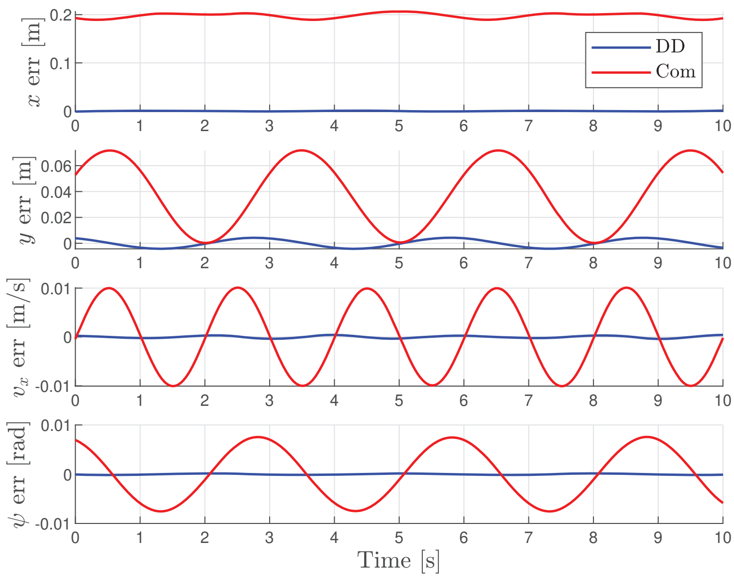


Figure 6.8: Error between the communicated vehicle states, the projected vehicle states, and the ground truth vehicle states for constantly varying inputs.

6.2 Situation 2: Only Radar

Figure 6.9 shows the proposed strategy which is used when only the radar sensor is active. Here, the vehicle

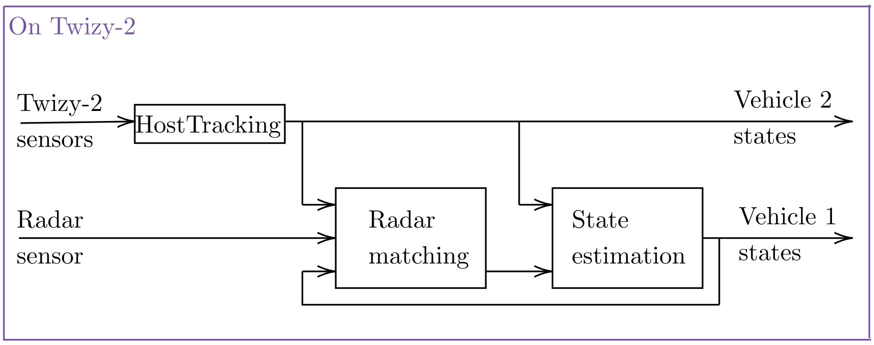


Figure 6.9: Overview of the state estimation system when only the radar sensor is active.

states of the following vehicle are estimated by Host-tracking. Radar matching uses the vehicle states of the following vehicle and the previous states of the leading vehicle to determine which of the radar detections is the leading vehicle. In simulation, the radar sensor only detects the leading vehicle. Therefore, the radar matching is omitted in the simulation. The EKF takes the states of the following vehicle and the radar object data, which is sampled at a lower frequency, as inputs to estimate the states of the leading vehicle at 100 Hz.

For this situation, the same EKF as described in Section 6.1.1 is used. The unicycle model takes the longitudinal acceleration and yaw rate as inputs, and when the V2V communication is not operational these values are not available. Therefore, the inputs to the unicycle model are set to 0. Due to the implementation of the radar measurements, the measurement equations c are changed to

$$c = \begin{bmatrix} x_r \\ y_r \\ v_{x,r} \\ v_{y,r} \end{bmatrix}, \tag{6.12}$$

where the radar measurements are described as in (5.10), (5.11), (5.12), and (5.13). This gives the measurements a non-linear relation to the system states. The measurement matrix is linearized using (2.3), and due to the non-linearity this is not an identity matrix. The radar measurements arrive at 16.7 Hz, so the time between the measurement updates is increased.

The initial process noise matrix is not changed compared to when only communication is used because the same unicycle model is used. The measurement noise covariance matrix is changed, because in this case radar measurements are used. The radar sensor accuracies as described in Table 5.4 are used. This results in the initial measurement noise matrix for when only the radar sensor is used, $R_{0,2}$, to be

$$R_{0,2} = \begin{bmatrix} 0.12 & 0 & 0 & 0 \\ 0 & 0.2 & 0 & 0 \\ 0 & 0 & 0.11 & 0 \\ 0 & 0 & 0 & 0.2 \end{bmatrix}. \tag{6.13}$$

In this situation, no delay is apparent for the EKF measurement inputs. Therefore, the noise matrices are tuned by comparing the outputs of the EKF to the ground truth directly without adjusting the inputs.

6.3 Situation 3: Communication and Radar

Figure 6.10 shows the strategy which is used when the radar sensor detects the leading vehicle and the V2V communication is operational. The Host-tracking algorithm is used on both vehicles to determine the vehicle

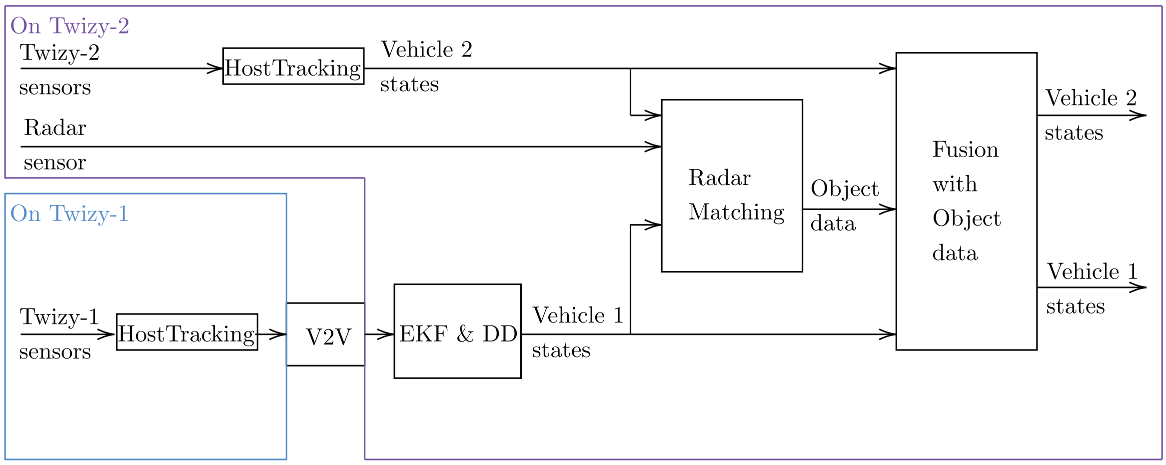


Figure 6.10: State estimation system when the V2V communication and radar sensor are active.

states. The state information of vehicle 1 is sent using the V2V communication and the EKF and DD are used to get non-delayed state information for vehicle 1. Based on the states of both vehicles, the radar matching algorithm selects a radar object. When a radar object is selected, another EKF is used to fuse the vehicle states of both vehicles with the radar object data. This is done because the radar sensor is much more accurate than the GPS sensor to determine the position of the vehicle.

The EKF and DD combination used in this situation is equal to the algorithm which is described in Section 6.1. The second EKF used to fuse the vehicle state with the radar measurements uses the same EKF algorithm. Because this EKF fuses the system states of both vehicles, the state equations are extended to include both vehicles. The system state equations are given by

$$f = \begin{cases} \dot{x}_1 = v_{x,1} \ \cos(\psi_1) \\ \ddot{y}_1 = v_{x,1} \ \sin(\psi_1) \\ \dot{v}_{x,1} = a_{x,1} \\ \dot{\psi}_1 = \omega_{z,1} \\ \dot{x}_2 = v_{x,2} \ \cos(\psi_2) \\ \ddot{y}_2 = v_{x,2} \ \sin(\psi_2) \\ \dot{v}_{x,2} = a_{x,2} \\ \dot{\psi}_2 = \omega_{z,2}, \end{cases} \tag{6.14}$$

where subscript 1 denotes the states of vehicle 1, and subscript 2 denotes the estimated states of vehicle 2. The outputs of the Host-tracking for vehicle 2 and the DD for vehicle 1 are treated as measurements together with the radar sensor outputs. The measurement function, c, is described as

$$c = \begin{bmatrix} x_{m,1} \\ y_{m,1} \\ v_{x,m,1} \\ \psi_{m,1} \\ x_{m,2} \\ y_{m,2} \\ v_{x,m,2} \\ \psi_{m,2} \\ x_r \\ y_r \\ v_{x,r} \\ v_{y,r} \end{bmatrix}. \tag{6.15}$$

Because the state space is increased from 4 to 8 states, the process noise matrix is 8-by-8. The initial process noise matrix, $Q_{0,3}$ is composed as

$$Q_{0,3} = \begin{bmatrix} Q_{0,1} & 0 \\ 0 & Q_{0,1} \end{bmatrix}. \tag{6.16}$$

The initial measurement noise matrix $R_{0,3}$ is composed by taking the measurement noise matrices from situation 1 and situation 2 and combining them as

$$R_{0,3} = \begin{bmatrix} R_{0,1} & 0 & 0 \\ 0 & R_{0,1} & 0 \\ 0 & 0 & R_{0,2} \end{bmatrix}. \tag{6.17}$$

These initial matrices are tuned with the same method as discussed in Section 6.2.

6.4 State Estimation Results

Two sets of input signals are constructed to determine ground truth data from the single-track models. Realistic sensor measurements are generated using the sensor models discussed in Section 5.2. These sensor measurements are the input of the three developed vehicle state estimation systems. In this section, the three systems are validated and compared in simulation.

The first set of single-track model inputs consists of a steering input of 0° and a trapezoidal acceleration input and is shown in Figure 6.11. The initial velocity of the vehicles is set to 10 m/s.

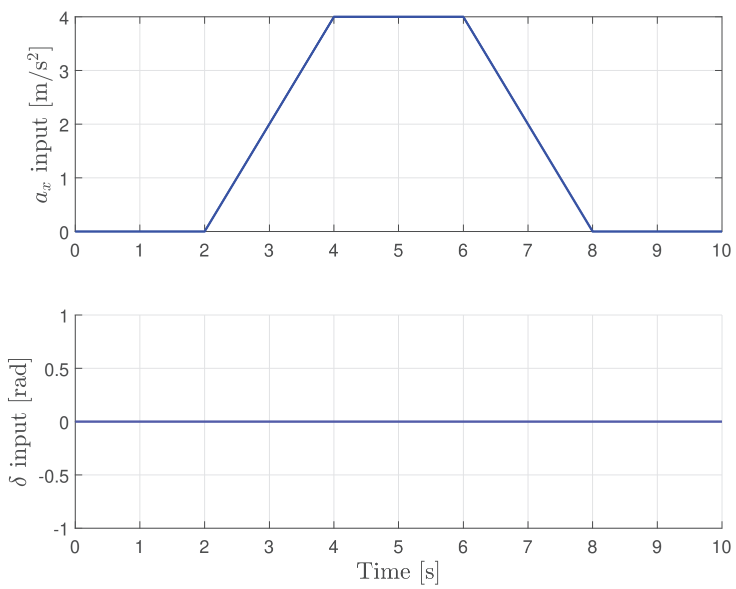


Figure 6.11: Inputs for the single-track models for a straight line acceleration

The estimation error is determined by comparing the ground truth vehicle data to the estimated vehicle states. Figure 6.12 shows the estimated states of the leading vehicle with the three different estimation algorithms to compare the performance. It can be seen that the estimation error for the x-position is the smallest

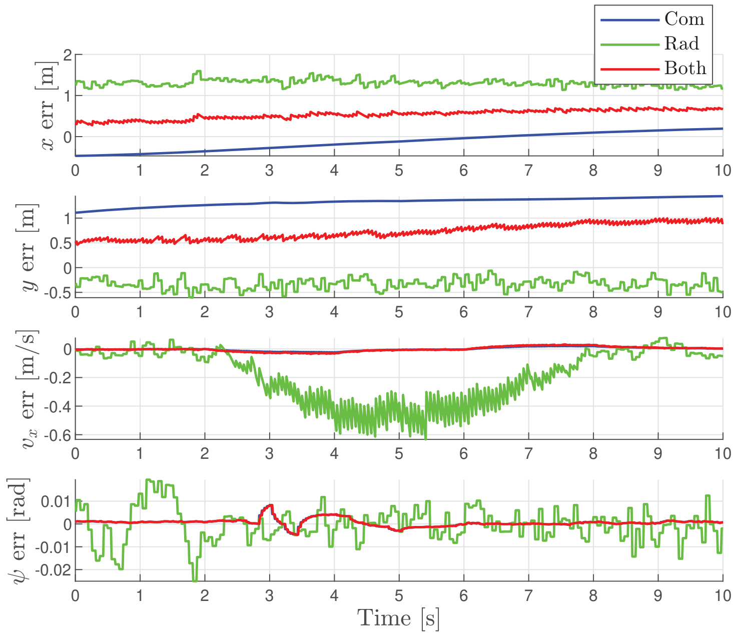


Figure 6.12: Estimation errors of vehicle 1 states for the three different methods for straight line driving with acceleration input.

when using both the V2V communication and the radar sensor. This signal shows more noise than the estimation done by purely the communication, but less noise than the estimation done with only the radar sensor. The error of the radar estimation is comparable to the estimation with both communication and radar but has coincidentally the opposite sign. The error of the communication-based estimation is the largest for the y-position. The velocity estimation for the methods with only communication and both communication and radar are comparable. The velocity estimation done by the radar sensor shows a larger error and has more noise. It has a negative error because the model assumes the vehicle is not accelerating. The heading estimation done with the radar shows more noise and a larger error than the estimate from the other methods, but the errors of the other two methods are similar.

Figure 6.13 shows the estimated vehicle states for vehicle 2 for driving in a straight line with the different estimation algorithms. Because the Host-tracking algorithm is used to determine the states of vehicle 2 for the methods where only the radar or the communication is active, these provide equal results. It can be seen that fusing the communication with the radar sensor measurements results in a smaller x-position error and changes the sign of the y-position error. The magnitude of the y-position error remains equal, but noise is apparent on the position estimations where both the radar and communication are used. For all methods, the velocity error is small with a maximum of 0.005 m/s. The heading angle error derived from the estimation with both the radar and communication follows the Host-tracking estimation but when a radar update is received an error peak is formed.

Figure 6.14 shows the errors in relative vehicle states for all three estimation methods for straight-line driving. The state estimation based on the radar sensor shows the smallest error because the radar directly measures the relative vehicle positions and velocities. The method where only communication is used results in the largest errors for the relative x and y-positions and distance. The method with both the radar sensor and the communication shows a trade-off, where the absolute states are estimated more accurately, but the relative vehicle states are estimated slightly worse than when using purely the radar sensor. The relative angle between the vehicles is noisy for the method with both radar and communication but remains smaller than 1.5°. This noise on the angle occurs because the estimated heading angle of the following vehicle has peaks.

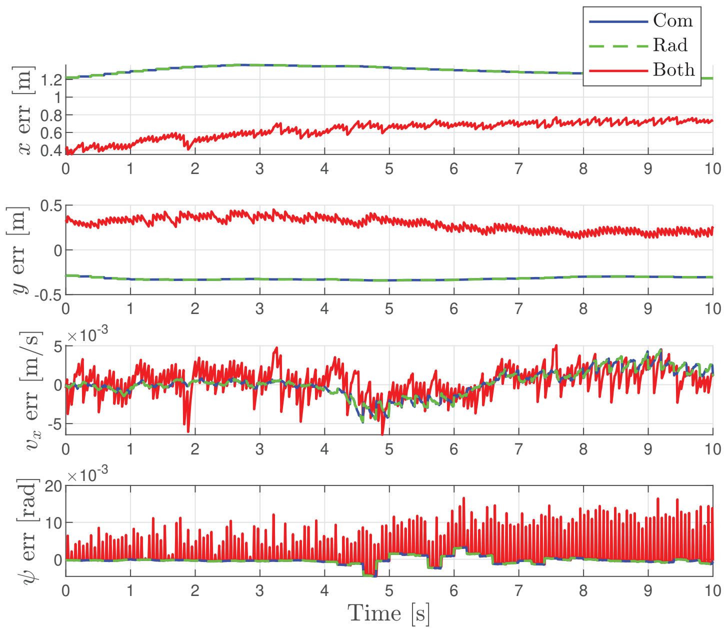


Figure 6.13: Estimation errors of vehicle 2 states for the three different methods for straight line driving with acceleration input.

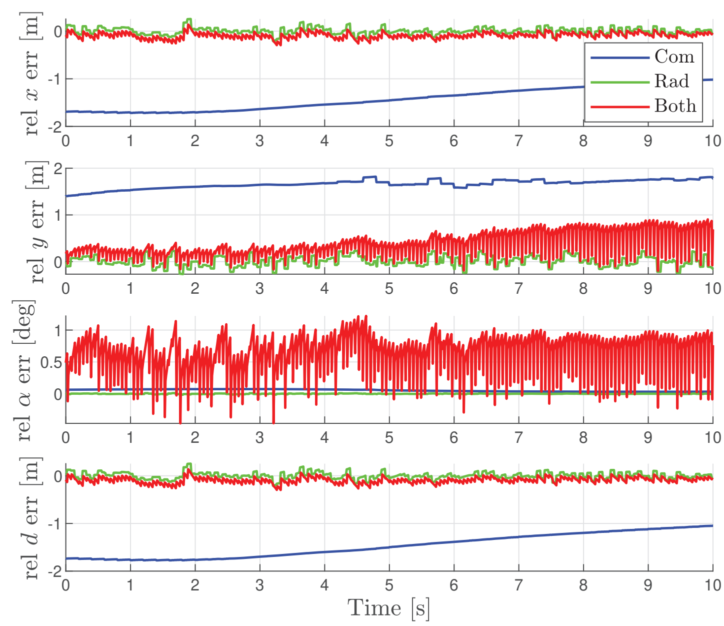


Figure 6.14: Estimation errors in the relative vehicle states for straight line driving with different state estimation methods.

The second set of inputs is selected to result in a circular path. The single-track vehicle inputs consist of a constant velocity of 10 m/s with constant steering input and are shown in Figure 6.15.

The estimation errors of the different algorithms for vehicle 1 states when driving in a circle with a constant velocity are shown in Figure 6.16. The results are similar to those for driving in a straight line. When the radar sensor and V2V communication are used, the position errors are the smallest, but some noise is put on the signal compared to the method where only communication is used. The velocity and heading angle errors for the methods with communication only and where communication and radar are used are similar. The velocity and heading estimations when only the radar sensor is used show more noise and a larger error. The heading estimation when only the radar is used also shows a bias. This is explained because the EKF used to fuse the

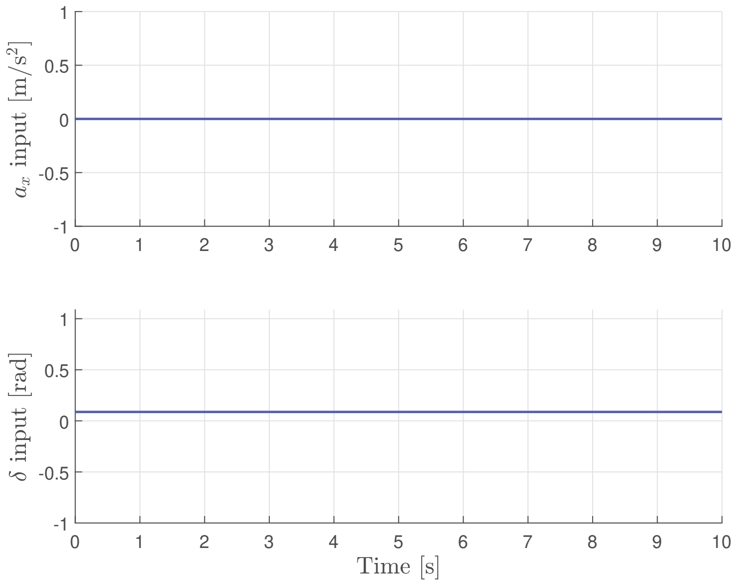


Figure 6.15: Inputs of the single-track models for a circle with constant velocity.

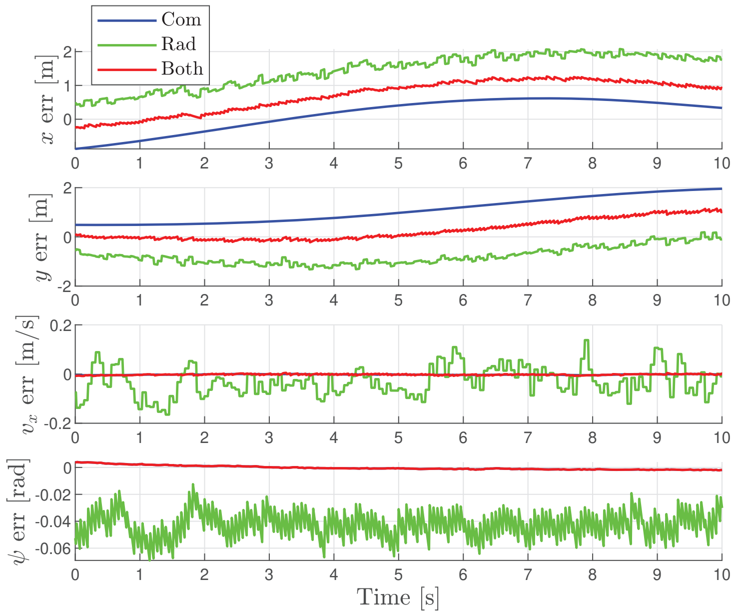


Figure 6.16: Estimation errors of vehicle 1 states for the three different methods for driving in a circle with a constant velocity.

radar measurements assumes the vehicle has a yaw rate of 0 rad/s.

Figure 6.17 shows the estimation errors for vehicle 2 for the different methods when driving in a circle. The results for the methods where either the communication or radar sensor is used are equal. The method where both are utilized results in a smaller error for the position coordinates. The velocity errors for all methods are negligible. With only the radar sensor or communication, the heading error can be estimated without error for vehicle 2 when driving in a circle. When information from the radar and communication is fused, peaks form in the heading error when the radar measurement arrives.

Figure 6.18 shows the estimation errors in relative states for the different estimation methods when driving in a circle with a constant velocity. For driving in a circle, using only the radar sensor gives the most accurate results for the relative vehicle states. It can be seen that using only the communicated vehicle information results in a large error. Using both provides results which are close to only using the radar sensor. The estimated angle between the vehicles is noisy for the simulation where both are used and shows a larger error of up to 2.1°

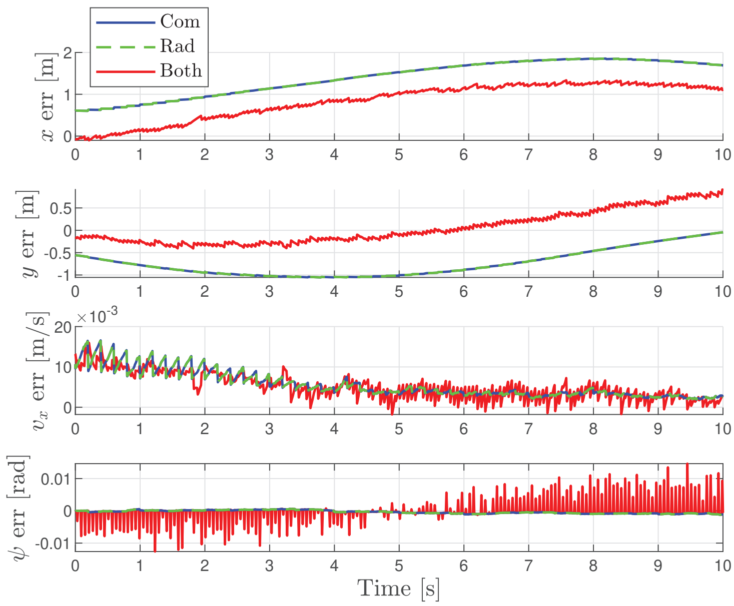


Figure 6.17: Estimation errors of vehicle 2 states for the three different methods for driving in a circle with a constant velocity.

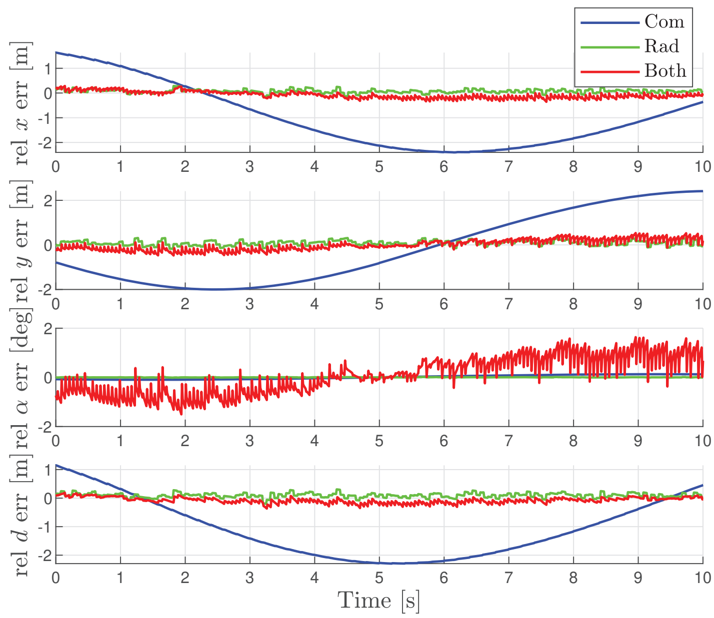


Figure 6.18: Estimation errors in the relative vehicle states for driving in a circle with different state estimation methods

compared to the other methods.

Table 6.1 shows the maximal estimation errors for the different methods for straight-line driving with acceleration. Table 6.2 shows the maximal estimation errors for the different methods when driving in a circle. The maximal errors of the absolute vehicle positions are similar between the methods where either the communication or the radar is used. The absolute errors of the position estimates are the smallest when the data from the communication is fused with radar measurements. When the vehicle states are determined solely from the communication, the absolute and relative vehicle positions are estimated with large errors. The heading and velocity of vehicle 1 are estimated with a large error when only the radar is used, compared to the other methods. When the communicated states are fused with the radar measurements, a trade-off is made. The absolute and relative vehicle states are estimated more accurately, but some of the noise from the radar measurements is introduced to the estimated vehicle state data. Because the vehicle motion controllers discussed in Section 3.3

potentially use both absolute and relative vehicle states, the estimation when both the communication and radar sensor are used has the highest performance.

Situation	Vehicle	Max. x error [m]	Max. y error [m]	Max. v_x error [m/s]	Max. ψ error [rad]
Communication	1	0.66	1.41	0.02	0.008
	2	1.35	0.35	0.004	0.005
Radar	1	1.55	0.58	0.62	0.022
	2	1.35	0.35	0.004	0.005
Both	1	0.77	0.98	0.03	0.007
	2	0.60	0.48	0.005	0.020

Table 6.1: Maximal estimation errors for the three different situations when driving in a straight line with trapezoidal acceleration

Situation	Vehicle	Max. x error [m]	Max. y error [m]	Max. v_x error [m/s]	Max. ψ error [rad]
Communication	1	0.99	2.02	0.005	0.005
	2	1.90	1.02	0.02	0.002
Radar	1	2.01	1.23	0.17	0.066
	2	1.90	1.02	0.02	0.002
Both	1	1.13	0.97	0.015	0.005
	2	1.20	0.67	0.02	0.018

Table 6.2: Maximal estimation errors for the three different situations when driving in a circle with constant velocity

Because the sensors are modelled using random noises with a specified variance, the accuracy of the vehicle state estimation deviates between different simulations with equal settings. This is shown in Appendix B. The simulation results which are presented in this section represent the average performance of the state estimation systems. This is determined by running 10 simulations and selecting the simulation which corresponds to the mean of the error of all simulations.

6.5 Conclusion and Recommendations

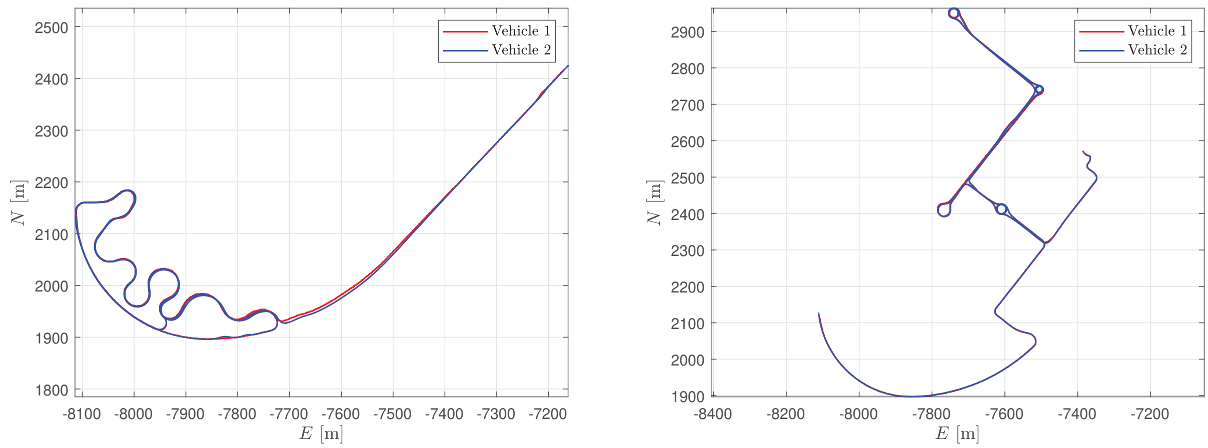
Depending on whether V2V communication and radar measurements are available three different vehicle state estimation algorithms have been developed and discussed. Using only V2V communication gives estimated states which are smooth, but have a large error of up to 1.9 m in the absolute reference frame and up to 2.2 m in the relative reference frame. Estimating the velocity and heading using only the V2V communication yields better results with smaller errors of 0.02 m/s and 0.01 rad. Using only the radar sensor to determine the vehicle states of the leading vehicle relies on the state estimation of the following vehicle. Therefore, the errors which are formed during the vehicle 2 state estimation are also apparent in the estimated vehicle 1 states in the inertial frame. Using only the radar sensor results in noise on the output signals. The absolute position is estimated with an accuracy of up to 2.0 m and the relative position between the vehicles is estimated with an accuracy of up to 0.4 m. The velocity estimation using the radar sensor has a maximal error of 0.62 m/s. Using both the communication and the radar sensor results in a result which lies between the separate methods. The absolute vehicle states are estimated with smaller errors than with either communication or radar sensor, with a maximal error of 1.2 m and a velocity error of 0.03 m/s. The relative state estimation is improved compared to using only the communicated information but performs slightly worse than using only the radar sensor resulting in a maximal error in relative distance of 0.5 m compared to 0.4 m when only using the radar sensor. Because both the absolute and relative vehicle states are required for the vehicle motion controllers, the estimation when both the radar and communication are active gives the highest performance.

In this current implementation, the state estimation is heavily reliant on the Host-tracking algorithm. During simulation, it was found that the Host-tracking algorithm also has its shortcomings. With more time a new on-board vehicle state estimation system could be developed, improving the performance of the algorithms discussed in this chapter.

Chapter 7

Full-scale Experiments

Full-scale experiments with 2 Renault Twizies are conducted at the Generaal Majoor de Ruyter van Steveninck-kazerne, a military training facility for driving and traffic participation with military vehicles. The driven paths of both vehicles for two test runs are shown in Figure 7.1. The first dataset was acquired while driving on a



(a) Dataset 1 **(b)** Dataset 2

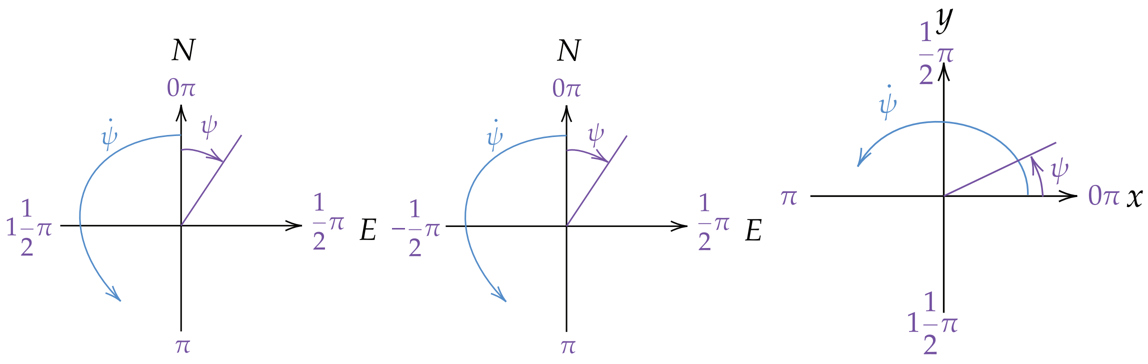
Figure 7.1: The driven path for both vehicles during two different test runs.

windy road. The vehicles were manually steered, and standard cruise control was used to drive with a constant velocity. During this, the cruise control velocity of the following vehicle was altered manually to ensure a safe following distance. For acquiring the second dataset, the vehicles were driven in a city-like environment with 90° corners and roundabouts without cruise control. Since the vehicles were not driven fully autonomously, the following times between the vehicles are not constant. Also because the vehicles were steered by humans, the paths of both vehicles are not equal. Due to the limitation of having to drive on specific roads, the vehicle trajectories are not similar to the paths driven in simulations. Because of this, comparing the simulations with the measurements cannot be done qualitatively. The V2V communication and radar sensors were operational during these tests.

On the PC in Twizy-2 all the required signals of the real-time model were logged, including the sensor measurements, the communicated information from Twizy-1, and the Host-tracking outputs. No logfile was saved on the Twizy-1 PC due to reliability issues. Because of this, the only information available from Twizy-1 are the delayed V2V communication messages logged on Twizy-2.

The NED frame is used on the Twizies while in simulation the ENU frame is used with the vehicle fixed reference frame specified by ISO8855. These reference frames are discussed in Chapter 2. A representation of the positive yaw rate direction, the heading angle and the axes of the inertial frame are shown in Figure 7.2. It can be seen that the yaw rate and heading angle have opposite signs in the test data, which is odd. The yaw rate is expressed in the same direction for the simulation and testing data. The vehicle state estimation system uses the heading angle definition as used in the simulation. To express the heading angle from the measurement data ψ_{data} to the ENU frame used in simulation ψ_{sim}, it is converted using

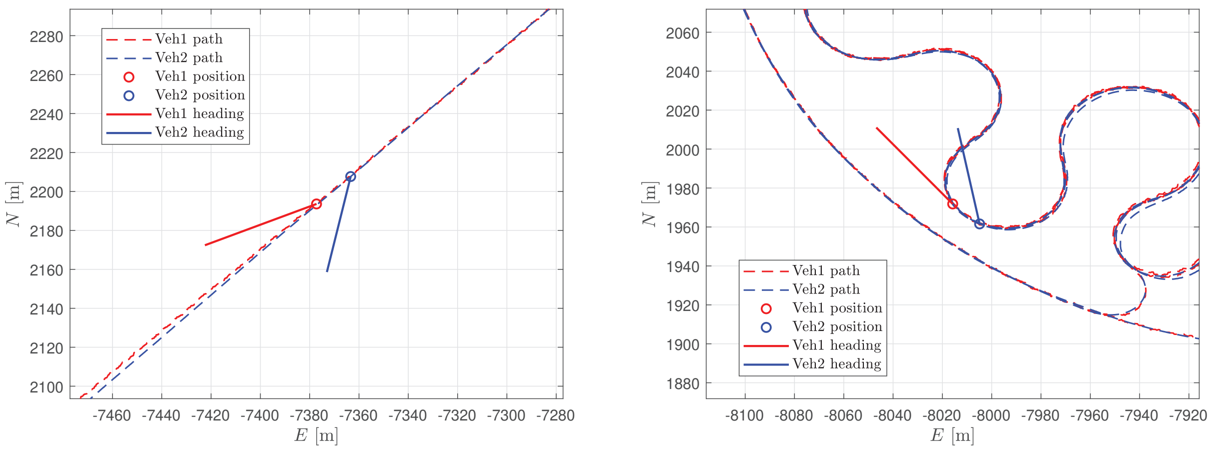
$$\psi_{sim} = (2\pi) - (\psi_{data} - \frac{1}{2}\pi). \tag{7.1}$$



(a) Twizy-1 data **(b)** Twizy-2 data **(c)** Simulation

Figure 7.2: Representation of the heading angle ψ, yaw rate $\dot{\psi}$ and inertial axes in measurement data and simulation.

The heading angle measured by the GPS sensor sometimes shows large deviations. Figure 7.3 show vehicle positions, path and heading angle measurements by the GPS sensor. It can be seen that the measured heading



(a) Instance 1 **(b)** Instance 2

Figure 7.3: Two instances where a large error in the measured heading angle is apparent.

angle of both vehicles shows a large deviation from the driven path in Figure 7.3a. The driven path is close to straight, so the vehicle slip angles in this situation are near 0°. The heading angle of vehicle 1 in Figure 7.3b is pointed inwards of the curve. During the experiment, the vehicles were driven under normal circumstances, and therefore the actual vehicle heading cannot be equal to what the GPS sensor measured.

During the process of running the different vehicle state estimation algorithms on the experimental data, it was found that large estimation errors occur once the heading angle makes a jump due to being expressed in an interval of 2π. This occurs because the EKF linearizes the system equations around an operating point, and once a jump of 2π is made in the heading angle this linearization has a large error. Therefore, the heading angle of the experimental data is not expressed in an interval of 2π, but can get arbitrarily large once the vehicle continues to rotate in the same direction.

Summary

This chapter describes the full-scale experiments executed to gather measurement data for experimental validation of the radar matching and the vehicle state estimation systems. Two test runs on different types of roads are described. The difference between the coordinate systems of the simulations and measurements, and the conversion between them are discussed. Lastly, large errors in the heading measurements of the GPS sensor are discussed.

Chapter 8

Radar Object Matching

Twizy-2 is equipped with a Bosch Mid-range radar sensor that can detect up to 32 objects and operates with a cycle time of 60 ms [31]. The radar sensor measures the relative distance r and angle α to an object, as shown in Figure 8.1. The relative distance and angle are internally converted to the relative longitudinal and lateral distances, x_{rad} and y_{rad}, and with onboard processing the relative longitudinal and lateral velocities, $v_{x,rad}$ and $v_{y,rad}$, are calculated. The measurements together with their covariances, are communicated over CAN.

The Bosch radar sensor is capable of separating objects when the distance between the objects is larger than 0.72 m, the velocity difference is larger than 0.66 m/s or the measured angle has a difference larger than 7°. Because of this, multiple of the 32 radar detections may be points on the leading vehicle. As the radar sensor is mounted to Twizy 2, the measurements are taken in the vehicle fixed reference frame $\vec{e}^2$. The FoV angle is dependent on the range and is given in Table 4.1

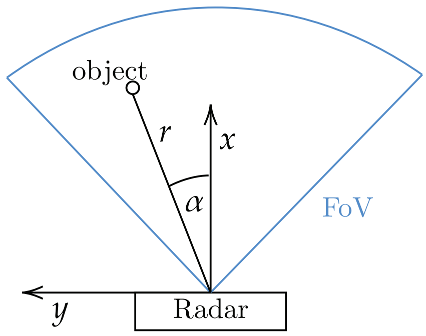


Figure 8.1: Representation of the radar FoV, measurement angle α and distance r, and the relative reference frame.

Even though the radar sensor measures the relative distance and velocity, the accuracy is much higher at 0.12 m compared to a GNSS sensor with an accuracy of over 2 meters. Therefore, utilizing the radar sensor can improve the vehicle state estimation accuracy, but only when detecting the leading vehicle. The 32 radar objects are compared to the vehicle states of the leading vehicle to assess whether the vehicle is detected. Figure 8.2 shows the approach of the radar object matching algorithm. This algorithm should be robust so that the amount of false positives and false negatives are minimized. Firstly, based on the estimated states of both vehicles, the expected radar measurement is calculated. Using the expected radar measurements, the vehicle states, and the radar object data several checks are performed to reject the objects which cannot be the preceding vehicle. Lastly, out of the remaining objects, the closest object to the expected radar measurement is selected with the assumption that rarely multiple of the 32 radar detections are the leading vehicle. The steps of the algorithm are explained in the following sections.

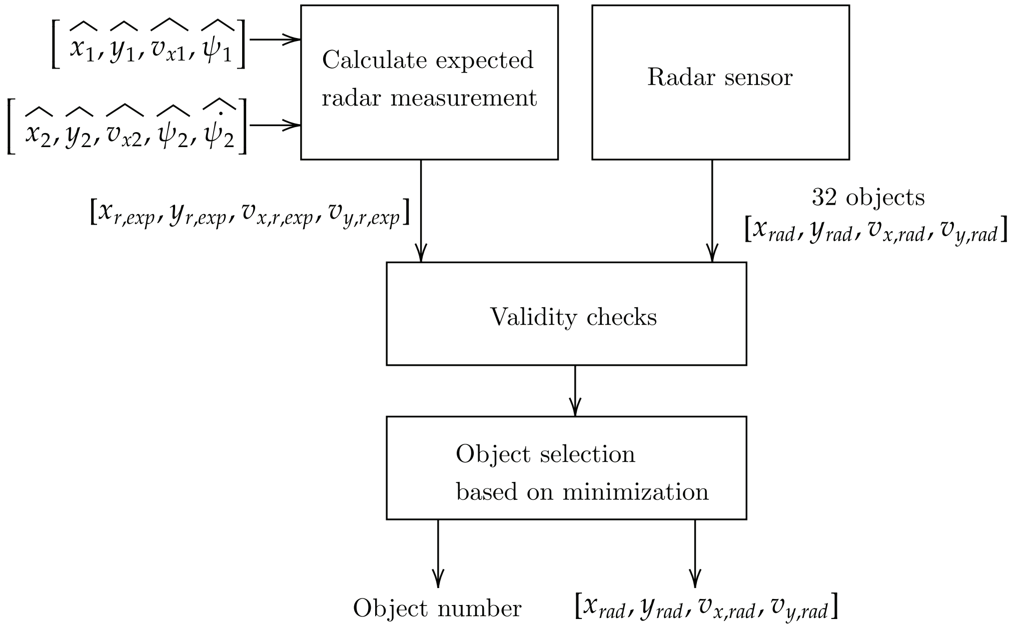


Figure 8.2: Approach of the radar object matching algorithm.

8.1 Expected Radar Measurement

The situation described in Figure 5.5 is used but with the estimated vehicle states as ground truth data is not available when the algorithm is running on the Twizy PC. The positions of both vehicles are expressed in the inertial frame, whereas the radar sensor measures in the vehicle fixed frame. Therefore, the expected radar measurements are calculated by a coordinate system conversion from the inertial reference frame to the relative reference frame fixed to vehicle 2 using the estimated vehicle states of the leading vehicle $[\hat{x}_1, \hat{y}_1, \hat{v}_{x1}, \hat{\psi}_1]$ and the vehicle states of the following vehicle $[\hat{x}_2, \hat{y}_2, \hat{v}_{x2}, \hat{\psi}_2, \hat{\dot{\psi}}_2]$. The expected relative longitudinal distance, $x_{r,exp}$, is calculated using

$$x_{r,exp} = \cos(\hat{\psi}_2)(\hat{x}_1 - \hat{x}_2) + \sin(\hat{\psi}_2)(\hat{y}_1 - \hat{y}_2) - l_r, \tag{8.1}$$

where $[\hat{x}_1, \hat{y}_1]$ and $[\hat{x}_2, \hat{y}_2]$ represent the estimated positions of the vehicles in the inertial frame, $\hat{\psi}_2$ is the estimated heading angle of the following vehicle, and l_r is the length from the rear axle of the following vehicle to the radar sensor as shown in Figure 5.5 and is equal to 1.5 m. It is necessary to compensate for the radar being mounted to the front of the vehicle because the vehicle's position is indicated by the point between the rear wheels. The expected relative lateral distance, $y_{r,exp}$, is calculated using

$$y_{r,exp} = -\sin(\hat{\psi}_2)(\hat{x}_1 - \hat{x}_2) + \cos(\hat{\psi}_2)(\hat{y}_1 - \hat{y}_2). \tag{8.2}$$

By taking the time derivative of the relative lateral and longitudinal distances, the relative lateral and longitudinal velocities are determined. The expected relative longitudinal velocity, $v_{x,r,exp}$, is expressed as

$$\begin{aligned} v_{x,r,exp} = &-\hat{\dot{\psi}}_2 \sin(\hat{\psi}_2)(\hat{x}_1 - \hat{x}_2) + \cos(\hat{\psi}_2)(\cos(\hat{\psi}_1)\hat{v}_{x1} - \cos(\hat{\psi}_2)\hat{v}_{x2}) \\ &+ \hat{\dot{\psi}}_2 \cos(\hat{\psi}_2)(\hat{y}_1 - \hat{y}_2) + \sin(\hat{\psi}_2)(\sin(\hat{\psi}_1)\hat{v}_{x1} - \sin(\hat{\psi}_2)\hat{v}_{x2}), \end{aligned} \tag{8.3}$$

where $\hat{\dot{\psi}}_2$ is the estimated yaw rate of the following vehicle, $\hat{\psi}_1$ is the estimated heading angle of the leading vehicle, and $\hat{v}_1$ and $\hat{v}_2$ are the estimated velocities of vehicle 1 and 2 respectively. The expected relative lateral velocity, $v_{y,r,exp}$, can be calculated using

$$\begin{aligned} v_{y,r,exp} = &-\hat{\dot{\psi}}_2 \cos(\hat{\psi}_2)(\hat{x}_1 - \hat{x}_2) - \sin(\hat{\psi}_2)(\cos(\hat{\psi}_1)\hat{v}_{x1} - \cos(\hat{\psi}_2)\hat{v}_{x2}) \\ &- \hat{\dot{\psi}}_2 \sin(\hat{\psi}_2)(\hat{y}_1 - \hat{y}_2) + \cos(\hat{\psi}_2)(\sin(\hat{\psi}_1)\hat{v}_{x1} - \sin(\hat{\psi}_2)\hat{v}_{x2}). \end{aligned} \tag{8.4}$$

These expected measurement values can be compared to the 32 radar objects to determine whether a radar detection is the leading vehicle.

8.2 Validity Checks

To determine whether the leading vehicle is potentially detected by the radar, several validity checks are performed using multiple metrics. Using the estimated vehicle states, it is estimated whether the leading vehicle is located within the radar FoV. The expected radar angle, α_{exp}, is calculated using

$$\alpha_{exp} = \arctan\left(\frac{y_{r,exp}}{x_{r,exp}}\right). \tag{8.5}$$

The expected radar range r_{exp} is calculated using

$$r_{exp} = \text{sign}(x_{r,exp})\sqrt{x_{r,exp}^2 + y_{r,exp}^2}. \tag{8.6}$$

Note that the sign of the expected relative longitudinal distance determines the sign of the expected radar range. This results in a negative value when vehicle 1 is estimated to be behind vehicle 2. Comparing the estimated range and angle to the specified radar FoV, it is decided whether the leading vehicle is within the FoV. A margin of 10% is taken on the range and FoV angle to account for errors in the estimated vehicle states.

Static objects are denied by comparing the measured relative longitudinal velocity of the radar object to the vehicle's velocity. The ground velocity of all detected objects, $v_{g,rad}$, is calculated using

$$v_{g,rad} = v_{x,rad} + v_2. \tag{8.7}$$

If the ground speed of a radar object is below a threshold, the object is deemed static. This threshold is set to 2 m/s lower than the longitudinal velocity of vehicle 2. By testing different values on measurement data, 2 m/s was found to give the best trade-off between rejecting false measurements without rejecting the correct measurements. When the V2V communication is operational, the ground speed of the radar objects is also compared to the velocity of the leading vehicle.

Lastly, for each of the 32 radar objects the deviation from the expected measurement is compared to a threshold. This threshold is experimentally determined for each of the distance calculation methods which are discussed in the next section. A radar object should pass all checks to be considered as potentially the leading vehicle by the algorithm.

8.3 Object Selection

Using the Euclidean, Manhattan, and Mahalanobis distances the expected radar measurements are compared to the radar objects that passed the previously discussed checks. The three different methods are shown in Figure 8.3. The Euclidean distance describes the distance between two points as the straight line between them. The Manhattan distance between two points is calculated by summing the distances over two orthogonal axes [40]. The Mahalanobis distance between two points is expressed as the number of times the measurement covariance fits in the square of the distance. The Euclidean method is chosen because it is the most obvious choice when determining the distance between two points. The Manhattan method is chosen for its straightforward and linear approach. The Mahalanobis method is implemented because it uses the measurement covariances when calculating the distance.

The Euclidean distance between the measured object positions and the expected radar measurement, d_{pos}, is expressed as

$$d_{pos} = \sqrt{(x_r - x_{r,exp})^2 + (y_r - y_{r,exp})^2}, \tag{8.8}$$

where x_r and y_r are the relative longitudinal and lateral distance measurements of 32 objects. This results in an array of 32 distances for each of the radar objects. Similarly, the difference between the measured object velocity and the expected radar measurement, d_{vel}, is calculated using

$$d_{vel} = \sqrt{(v_{x,r} - x_{x,r,exp})^2 + (v_{y,r} - v_{y,r,exp})^2}, \tag{8.9}$$

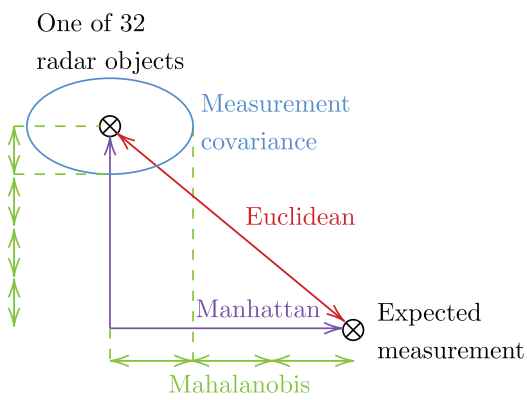


Figure 8.3: Representation of the Euclidean, Manhattan, and Mahalanobis distance calculation methods.

where $v_{x,r}$ and $v_{y,r}$ are the relative longitudinal and lateral velocity measurements of 32 objects.
Using the Manhattan distance, the position and velocity distances are expressed as

$$d_{pos} = |x_r - x_{r,exp}| + |y_r - y_{r,exp}|, \tag{8.10}$$

and

$$d_{vel} = |v_{x,r} - x_{x,r,exp}| + |v_{y,r} - v_{y,r,exp}|. \tag{8.11}$$

The position distance using the Mahalanobis method is expressed as

$$d_{pos} = (x_r - x_{r,exp})\sigma_x^{-1}(x_r - x_{r,exp})^T + (y_r - y_{r,exp})\sigma_y^{-1}(y_r - y_{r,exp})^T, \tag{8.12}$$

where σ_x and σ_y denote the covariance on the relative longitudinal and lateral distances respectively. The velocity difference using the Mahalanobis method is expressed as

$$d_{vel} = (v_{x,r} - v_{x,r,exp})\sigma_{vx}^{-1}(v_{x,r} - v_{x,r,exp})^T + (v_{y,r} - v_{y,r,exp})\sigma_{vy}^{-1}(v_{y,r} - v_{y,r,exp})^T, \tag{8.13}$$

where σ_{vx} and σ_{vy} are the covariance on the relative longitudinal and lateral velocities respectively.

8.4 Results

To compare the three methods, the two data sets discussed in Chapter 7 are used. Figure 8.4 shows the relative longitudinal distance of the chosen radar object compared to the expected value based on the estimated vehicle states when the validity checks are removed. In this case, the object is selected by the smallest deviation from the expected value. Therefore, the algorithm always finds an object. It can be seen that between $t = 142$ s and $t = 180$ s, a large deviation of 12 m between the chosen object and the expected measurement occurs. This also occurs between $t = 233$ s and $t = 255$ s. When the leading vehicle is not detected by the radar sensor another object is selected, causing this deviation. Simply picking the closest radar object in these instances results in many false positives for all three distance calculation methods.

After the implementation of validity checks, tuning the methods resulted in a fraction of accepted measurements for all methods between 79 and 83 %. During this tuning, the thresholds for the maximum allowed distance are set for each method individually. To have a fair comparison, the allowed distance from the expected measurement is retuned for all three methods to ensure that the same percentage of radar measurements is accepted. Figure 8.5 shows the relative longitudinal distance for the three different distance calculation methods with the validity checks. It can be seen that the number of false positives decreases, as between $t = 152$ s and $t = 180$ s no measurements are accepted where previously wrong measurements were selected. This also occurs between $t = 233$ s and $t = 255$ s. All methods are able to track the expected measurements. However, at $t = 600$ s the Manhattan method results in false positives with a distance of 40 m from the expected measurements.

Ideally, ground truth vehicle state data is available to determine the performance of the different methods. Since this is not the case, the different methods are compared to the expected radar measurements based on the estimated states. Because the amount of accepted measurements is equal for all methods, the average

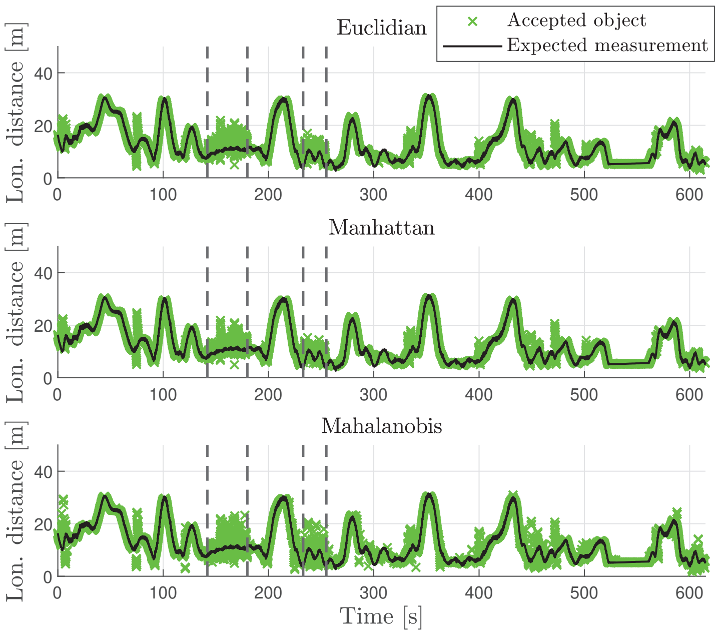


Figure 8.4: Selected radar object based on proximity for different distance methods compared to the expected measurement.

Matching method	Fraction [-]	$e_{p,Eucl.}$ [m]	$e_{v,Eucl.}$ [m/s]	$e_{p,Manh.}$ [m]	$e_{v,Manh.}$ [m/s]
Euclidean	0.814	1.57	2.42	1.89	2.98
Manhattan	0.814	1.64	2.36	1.96	2.91
Mahalanobis	0.814	1.63	2.34	1.95	2.89

Table 8.1: Performance of the three radar matching methods.

deviation from the accepted objects to the estimated value is taken to compare the performance. This is done by calculating both the Euclidean and Manhattan errors between the accepted radar objects and the expected radar measurement. The Euclidean position error, $e_{p,Eucl}$ is calculated using

$$e_{p,Eucl} = \frac{\sum_{t=1}^{t_{max}} \sqrt{(x_{obj}(t) - x_{r,exp}(t))^2 + (y_{obj}(t) - y_{r,exp}(t))^2}}{t_{max}}, \tag{8.14}$$

where x_{obj} and y_{obj} are the relative positions of the accepted object at a specific time step t. The Euclidean velocity error, $e_{v,Eucl.}$, is calculated using

$$e_{v,Eucl.} = \frac{\sum_{t=1}^{t_{max}} \sqrt{(v_{x,obj}(t) - v_{x,r,exp}(t))^2 + (v_{y,obj}(t) - v_{y,r,exp}(t))^2}}{t_{max}}, \tag{8.15}$$

where $v_{x,obj}$ and $v_{y,obj}$ are the relative velocities of the accepted object. In equal fashion, the Manhattan errors are determined using

$$e_{p,Manh.} = \frac{\sum_{t=1}^{t_{max}} |x_{obj}(t) - x_{r,exp}(t)| + |y_{obj}(t) - y_{r,exp}(t)|}{t_{max}}, \tag{8.16}$$

and

$$e_{v,Manh.} = \frac{\sum_{t=1}^{t_{max}} |v_{x,obj}(t) - v_{x,r,exp}(t)| + |v_{y,obj}(t) - v_{y,r,exp}(t)|}{t_{max}}. \tag{8.17}$$

Table 8.1 shows the errors of the different radar matching methods. It can be seen that for both metrics the Euclidean matching results in a lower position error, but a larger velocity error than both the Manhattan and the Mahalanobis methods. The Mahalanobis method performs slightly better than the Manhattan method. Figure 8.6 shows the three different matching methods with the same thresholds as shown previously used on a different data set. It can be seen that for this data set the three methods perform comparably. However, the Mahalanobis method has false positives. The performance of the different methods is shown in Table 8.2. It can be seen that the Euclidean method has the lowest fraction of accepted detections, and the position errors are

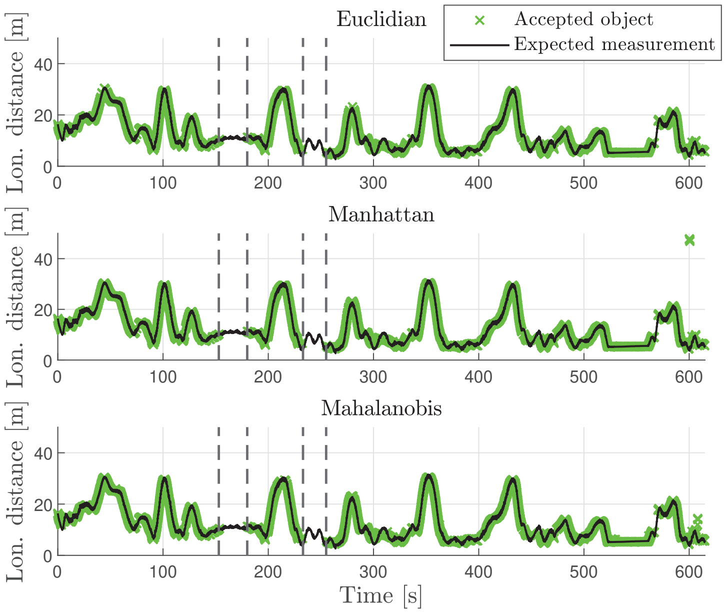


Figure 8.5: Accepted radar object for different distance methods compared to the expected measurement.

the smallest of all three methods. The Manhattan and Mahalanobis result in a smaller velocity error, as was the case for the first data set. To determine the best-performing method, all radar states were considered. Only the results for the longitudinal distance are shown here because it represents the results of the other states. The full results with all the radar states for both data sets are shown in Appendix A.

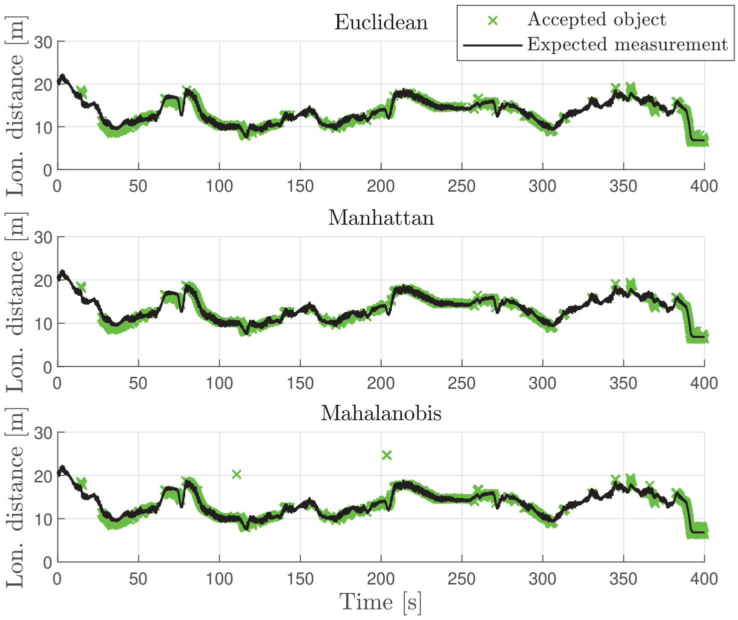


Figure 8.6: Accepted radar objects for a different set of measurement data.

Matching method	Fraction [-]	$e_{p,Eucl.}$ [m]	$e_{v,Eucl.}$ [m/s]	$e_{p,Manh.}$ [m]	$e_{v,Manh.}$ [m/s]
Euclidean	0.540	2.25	1.94	2.59	2.33
Manhattan	0.594	2.50	1.73	2.81	2.04
Mahalanobis	0.577	2.43	1.90	2.79	2.27

Table 8.2: Performance of the three radar matching methods on a different data set.

The Mahalanobis and Manhattan methods resulted in clear outliers compared to the expected values. This did not occur for the Euclidean method, deeming this method the most robust. Furthermore, the radar sensor is more accurate in determining the relative position than the relative velocity. Therefore, giving a more accurate position is desired over a more accurate velocity. Because of this, the Euclidean method is chosen for a final implementation.

Figure 8.7 shows all the radar states for the Euclidean method. It can be seen that a constant deviation in

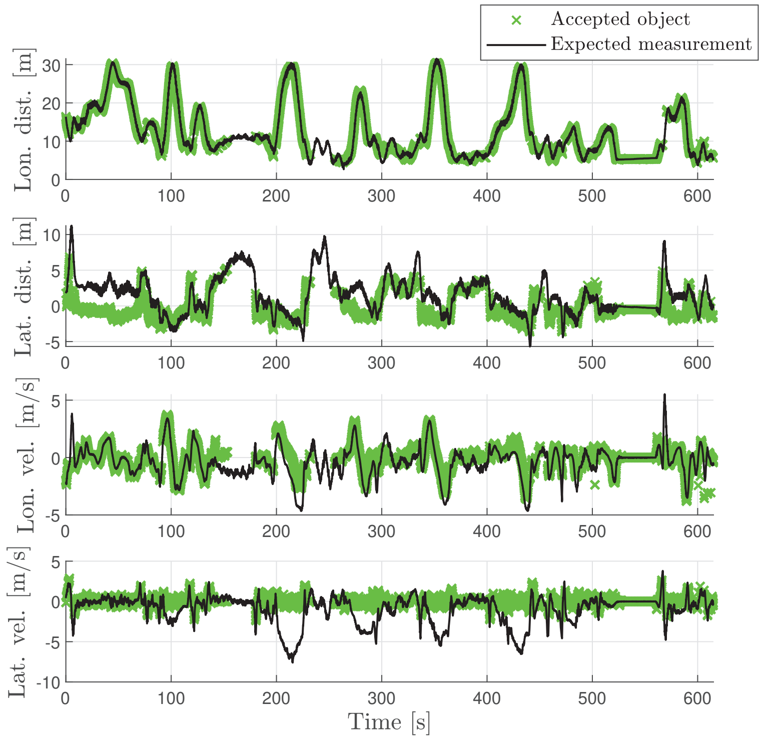


Figure 8.7: Accepted data for all radar states for Euclidean method.

lateral distance is apparent from $t = 20$ s and $t = 70$ s. This again occurs from $t = 520$ s and $t = 550$ s. This constant offset in relative lateral distance is caused because the zero point of the radar angle measurement is not aligned sufficiently with the vehicle's axis. This can occur because the radar is calibrated poorly, or the current mounting of the radar is not structurally proper. A constant error in the heading angle measurement of the GPS sensor could also cause this. While the tracking for the longitudinal distance and velocity is accurate, the tracking for the lateral velocity is not accurate, where errors of up to 6 m/s are visible.

Figure 8.8 shows which object is selected over time. When object 0 is selected, no object is accepted. It can be seen that the radar matching algorithm selects different radar objects over time. The longest period where the selected radar number stays constant is 60 seconds. Around $t = 150$ s and $t = 450$ s it can be seen that the selected radar object changes quickly between different numbers.

To investigate whether the large deviation in relative lateral position is the cause of improper mounting or calibration of the radar, the radar measurements are corrected during the post-processing of the data. An extra term α_o is defined which describes the offset angle between the calibration and mounting of the radar. To correct the radar measurements, the distance to an object is calculated using

$$r = \sqrt{x_r^2 + y_r^2}. \tag{8.18}$$

The radar angle to an object α is calculated using

$$\alpha = \arctan(\frac{y_r}{x_r}). \tag{8.19}$$

The radar measurement angle to an object when the radar is mounted properly, α_{cor}, is calculated using

$$\alpha_{cor} = \alpha + \alpha_o. \tag{8.20}$$

The corrected lateral and longitudinal, $x_{r,cor}$ and $y_{r,cor}$, measurements of the radar are calculated using

$$x_{r,cor} = r\cos(\alpha_{cor}), \tag{8.21}$$

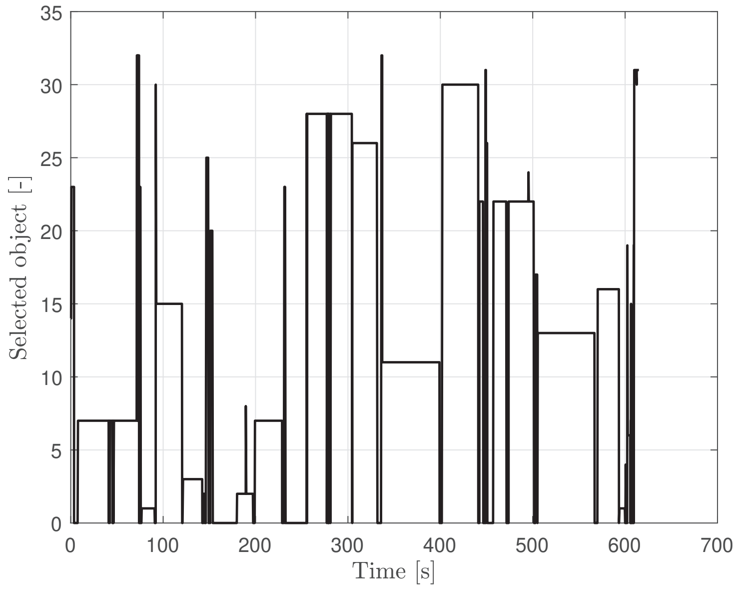


Figure 8.8: Selected radar object over time.

and

$$y_{r,cor} = r \sin(\alpha_{cor}). \tag{8.22}$$

For the lateral and longitudinal velocity measurements, the same recalibration can be done where in the equations above the relative position measurements are replaced with the relative velocity measurements.

After adjusting the correction angle for the radar in increments of 0.1° and determining the Euclidean error between the expected and corrected radar measurements, it was discovered that the radar sensor had an offset of 8.1°. Due to the small angle, the longitudinal measurements are less affected by this correction as compared to the lateral measurements. Implementing this correction led to a 28.7% reduction in the deviation of the position between the expected and actual radar measurements for dataset 1 and a 28.6% reduction for dataset 2. Additionally, the velocity deviation decreased by 36.8% for dataset 1 and by 50.5% for dataset 2.

8.5 Conclusions and Recommendations

An algorithm to select the correct radar object based on estimated vehicle states has been developed. Based on validity checks, a decision is made whether one of the measured objects is potentially the preceding vehicle. From the objects that passed the checks, the object closest to the expected value is chosen. Three different methods for determining the closest object have been implemented and tested. The Euclidean method results in the best performance since position data is considered more important than velocity data. After tuning the algorithm a different data set was used to validate the robustness of the algorithm. Here also the Euclidean method gave the best performance. The average error to the expected measurement is 1.57 m and a velocity error of 2.42 m/s for the first data set, and 2.25 m and a velocity error of 1.94 m/s for the second data set. It was found that the radar sensor was mounted with an offset of 8.1°. Correcting for this decreased the position and velocity deviations between the expected and actual measurements up to 28.7% and 50.5% respectively.

The current algorithm does not take into account the possibility of having more than one radar object detected on the leading vehicle, which is one of its shortcomings. Currently, there is also no correction for which part of the vehicle is detected. The position of the vehicle is defined as the location of the rear axle, whereas the radar sensor detects an arbitrary point on the rear of the vehicle. During this project, the utilization of the RTK-GPS system was not possible. With the RTK-GPS, the point on the vehicle which is detected can be determined. Also, it can be determined whether multiple radar objects denote different points on the same vehicle. It was found that currently the radar calibration or mounting causes a discrepancy between the expected lateral distance and the measured lateral distance. This has an impact on the distance thresholds set for the acceptance of the measurements. With proper radar mounting and calibration, the performance of the radar matching system is expected to be further increased. Lastly, the covariances on the state estimates and the radar measurements are not taken into account when selecting a radar object because the Mahalanobis method performed worse than the Euclidean method. Further research could provide a different method of incorporating this, resulting in a better radar matching algorithm.

Chapter 9

Vehicle State Estimation on Measurement Data

This chapter describes the application of the three vehicle state estimation systems discussed in Chapter 6 on the experimental data discussed in Chapter 7. To gain object-level radar data of the leading vehicle the radar matching algorithm discussed in Chapter 8 is applied to the experimental data. These radar measurements are corrected for the offset between the calibration angle and the mounting.

9.1 Situation 1: Communication only

Due to missing the logged data on Twizy-1, recreating the vehicle states through the EKF with DD as discussed in Section 6.1 can be done, but no conclusions can be drawn on the accuracy. Therefore, to test the system when only communication is active, the vehicle state estimates of Host-tracking on Twizy 2 are delayed for 0.02 s and resampled at 25 Hz to simulate the communication. This is fed through the EKF and DD, and the outputs are compared to the undelayed Host-tracking outputs of vehicle 2. Ground truth data of vehicle states is not available, and therefore comparing to the undelayed Host-tracking output is the only measure of performance that can be evaluated.

Figure 9.1 shows the errors between the undelayed Host-tracking output of vehicle 2, the simulated communicated vehicle states, the EKF output and the output of the DD. It can be seen that for experimental data the delayed communication and the EKF result in errors of up to 0.25 m in both position coordinates. These errors in the position are decreased by applying the DD. The error of the velocity estimation is comparable for the three methods because cruise control is used. By applying the DD the heading angle errors are also decreased. The state estimations done with the EKF and the EKF with DD show an error peak at the beginning of the data set. This occurs because the EKF initializes at that moment and requires time to settle.

Figure 9.2 shows the estimation errors for a different dataset. It can be seen that the maximal position error is decreased. The velocity signals are comparable. The maximal heading angle error is decreased from 0.015 rad to 0.01 rad. For experimental data of vehicle 2, the EKF and DD decrease the error that is a result of the lower sampling frequency and delay of the replicated V2V communication.

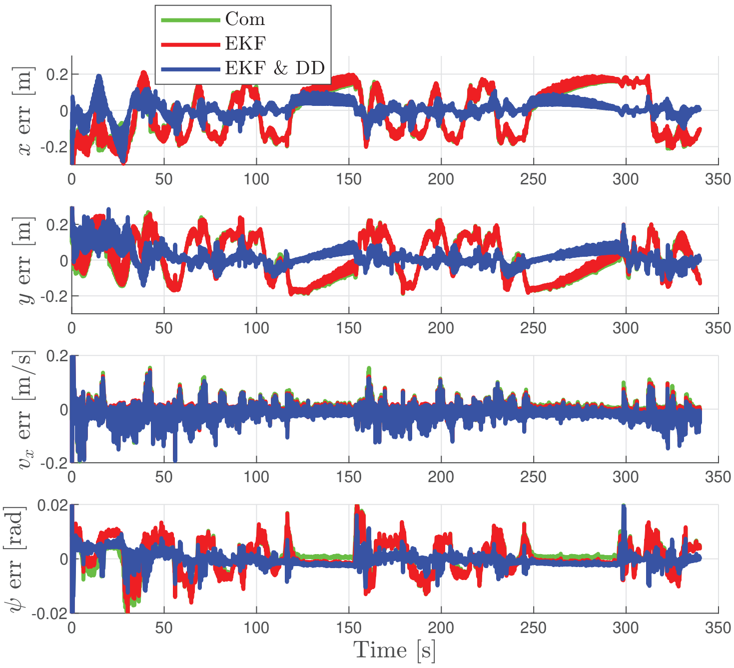


Figure 9.1: Error between the delayed vehicle states, EKF and discrete dynamics outputs for experimental data from test run 1 of vehicle 2.

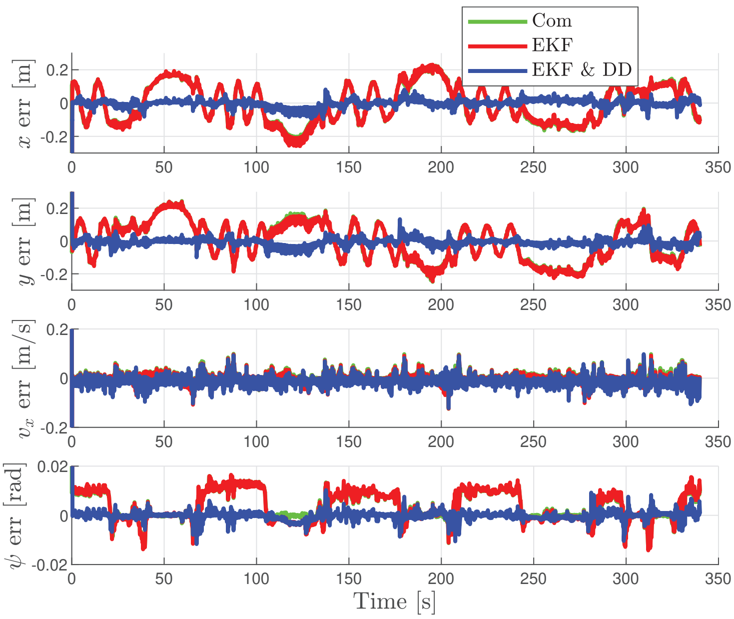


Figure 9.2: Error between the delayed vehicle states, EKF and discrete dynamics outputs for experimental data from test run 2 of vehicle 2.

9.2 Situation 2: Radar only

The V2V communication was operational during all of the testing runs. However, to simulate the V2V communication being defective only the radar data is used to estimate the vehicle states of the leading vehicle. The results of the state estimation done based on the radar measurements are compared to the communicated vehicle state information of the leading vehicle.

Switching between the different algorithms when either the communication or radar sensor becomes inactive has not been implemented due to time constraints. Therefore, a part of the first dataset is used where the selected radar object remains constant. Figure 9.3 shows the error between the through V2V communication received Host-tracking output of vehicle 1 and the state estimation when using the radar measurements. It can

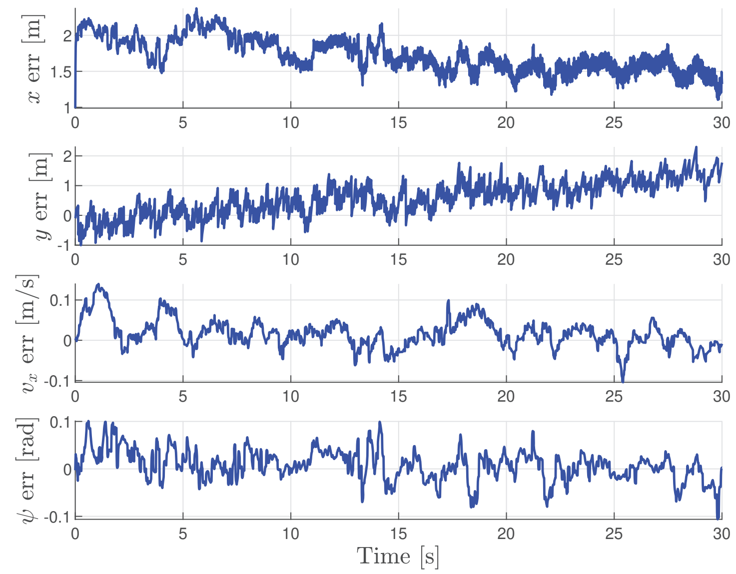


Figure 9.3: Error between the estimated vehicle states using the radar sensor and the communicated vehicle states of the leading vehicle for dataset 1.

be seen that estimating the vehicle positions with purely the radar sensor results in large differences with the Host-tracking estimations of up to 2.35 m for both the x and y-positions. The estimations done with the radar are noisier than the Host-tracking outputs, resulting in noisy error signals.

Figure 9.4 shows the large Euclidean distance between the position estimation using the radar sensor and the communicated position. It can be seen that the maximal difference between the communicated position

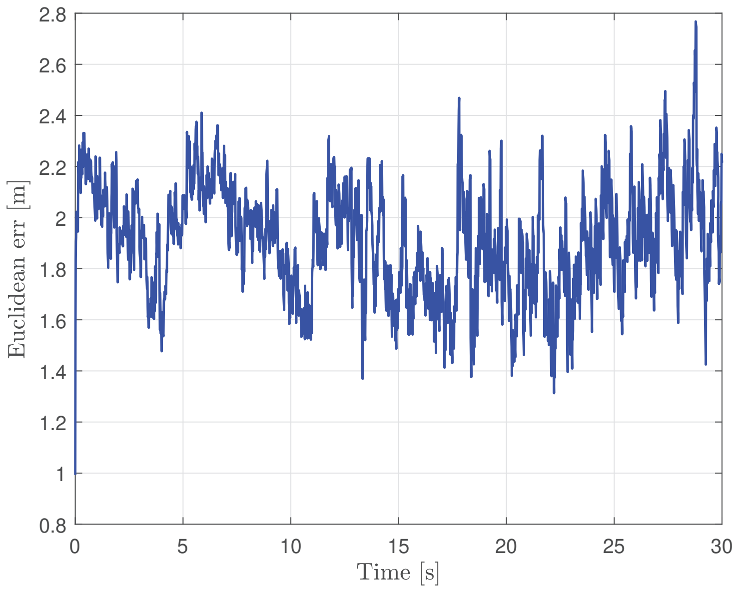


Figure 9.4: Euclidean position error between the state estimation using the radar sensor and the communicated vehicle position for dataset 1.

and the estimated position equals 2.77 m. The average position difference is 1.95 m.

Figure 9.3 shows the error between the state estimation and the Host-tracking output of vehicle 1 for the second dataset. The magnitudes of the positional errors are similar to Figure 9.3. However, the error in the velocity is larger at 0.6 m/s. The heading angle shows a constant offset of -0.2 rad during the first 20 seconds

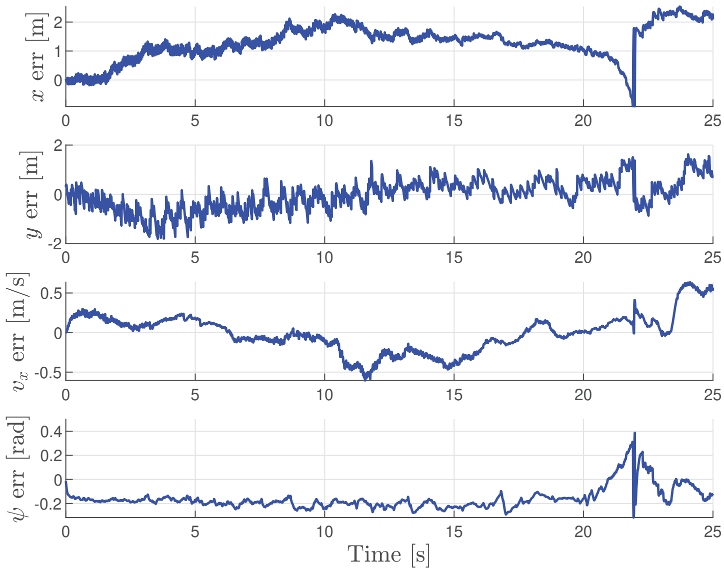


Figure 9.5: Error between the estimated vehicle states using the radar sensor and the communicated vehicle states of the leading vehicle for dataset 2.

of this dataset. A top view of both vehicles on the driven path of this data set at t = 5 s is shown in Figure 9.6. Here it can be seen that both vehicles are not aligned with their path, and have a deviation to opposing sides.

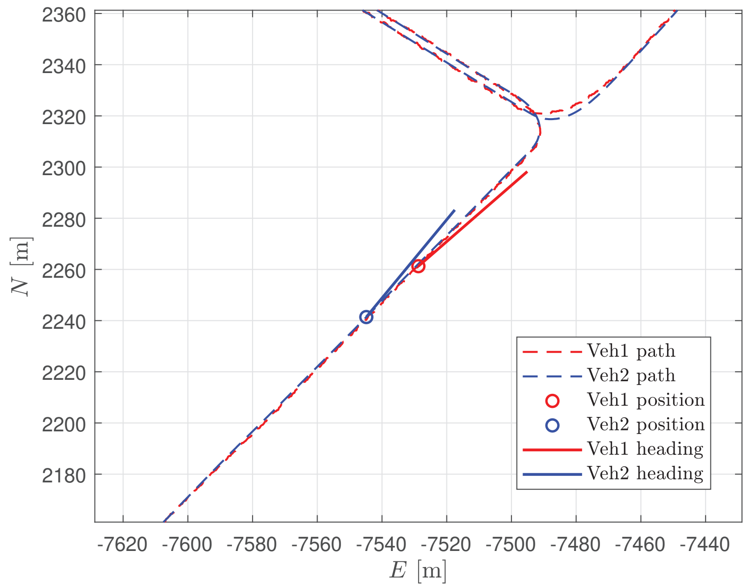


Figure 9.6: Position and heading of both vehicles at t = 5 s in Figure 9.5.

The heading angle estimations of the Host-tracking algorithm on both vehicles have a difference of 0.2 rad while the vehicles are driving in the same straight path. This is equal to the constant error of 0.2 rad in the heading angle error shown in Figure 9.5.

Figure 9.7 shows the position error between the state estimation using the radar sensor and the communicated position for dataset 2. The maximal position error is 2.75 m which is comparable to the maximal difference for the first data set. The average position difference is smaller at 1.45 m.

Due to the lack of ground truth data, no conclusions can be drawn on the performance of the state estimation system when only the radar data is used. The differences between the communicated and estimated states are within the error margins of the Host-tracking algorithm. It cannot be determined which set of estimated states

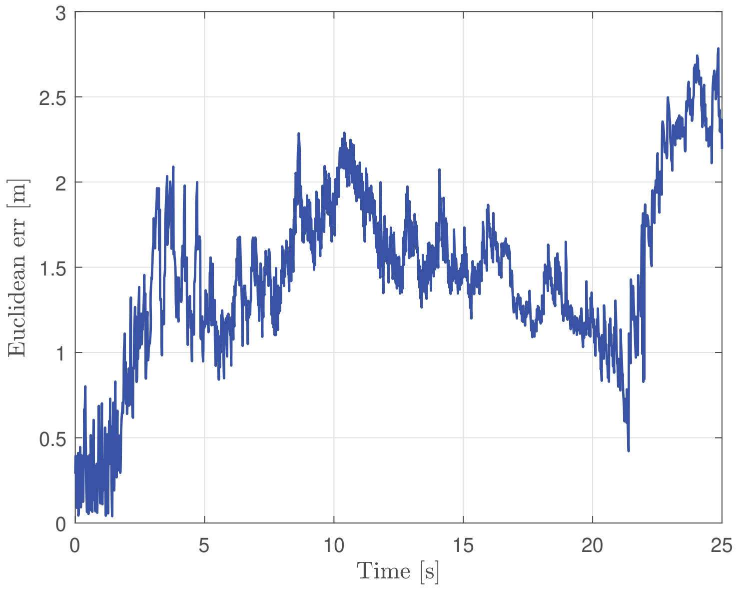


Figure 9.7: Euclidean position error between the state estimation using the radar sensor and the communicated vehicle position for dataset 2.

represents the actual vehicle states better. However, the estimation when solely the radar sensor is used results in a nosier signal than the Host-tracking algorithm. This is consistent with the simulations.

9.3 Situation 3: Communication and Radar

The errors between the Host-tracking outputs of both vehicles and the state estimation when both the communication and the radar sensor are used, are shown in Figure 9.8. Here the communicated states of vehicle 1 are fitted with an EKF and DD are used to compensate for the communication delay. The Host-tracking outputs of vehicle 2 and the outputs of the DD are fused with the selected radar object data. The yaw rate of vehicle 1 was not included in the V2V communication during the testing. Therefore it is reconstructed by numerical differentiation of the heading angle. It can be seen that the positions of both vehicles are altered by fusing with the radar sensor. The estimated position deviates up to 1.4 m from the Host-tracking estimate. The maximal error in the velocity is 0.11 m/s for vehicle 1 and 0.06 m/s for vehicle 2. The heading angle difference is minimal, at 0.005 rad for vehicle 1 and 0.012 rad for vehicle 2.

Figure 9.9 shows the error between the Host-tracking outputs and the fusion algorithm for dataset 2. It can be seen that the x-position errors are mirrored around 0. This occurs because the radar sensor is used to change the position of both vehicles equally. The velocity error for this dataset is larger at 0.23 m/s for vehicle 1 and 0.11 m/s for vehicle 2. The heading angle errors are small, but a peak occurs in the heading angle estimation of vehicle 1 at t = 21 s. This is caused by a large error in the heading angle estimation of the Host-tracking algorithm due to a poor GPS measurement at that time.

The estimation differences lie within the accuracy range of the Host-tracking algorithm. Because of this, it cannot be determined whether the Host-tracking estimation or the estimation where the radar measurements are fused with V2V communication is more accurate.

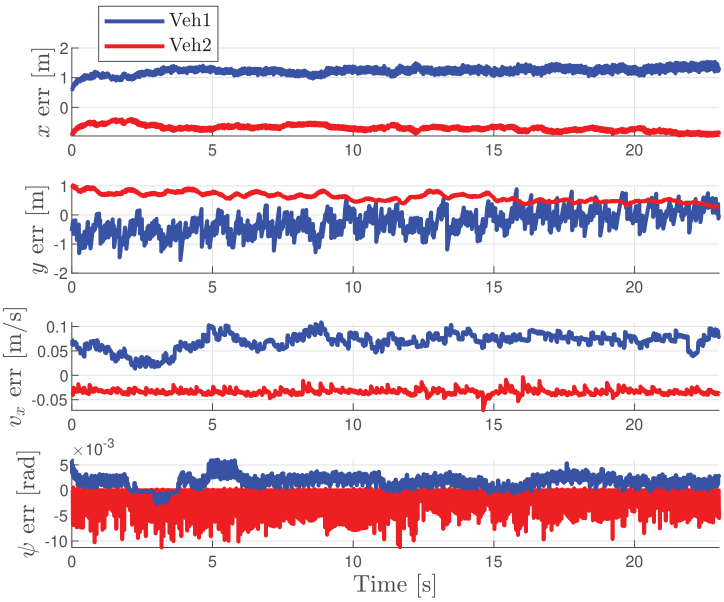


Figure 9.8: Error between the Host-tracking outputs compared to the vehicle state estimation where V2V communication data is fused with radar measurements for dataset 1.

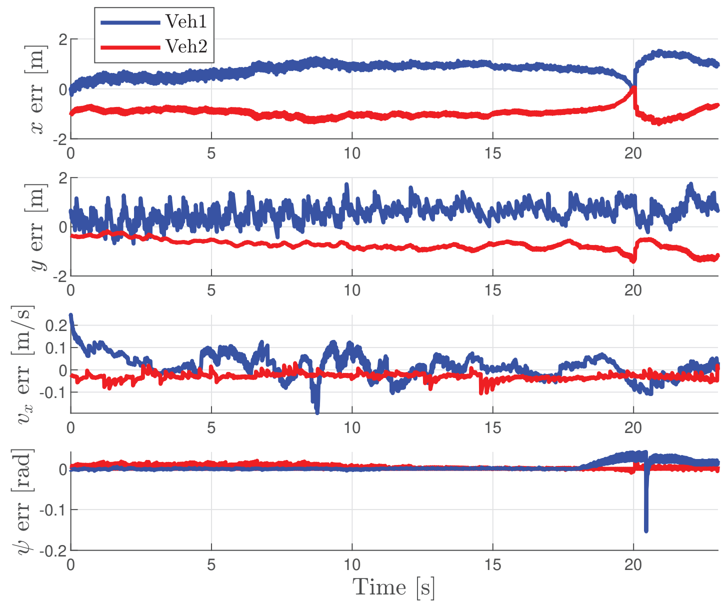


Figure 9.9: Error between the Host-tracking outputs compared to the vehicle state estimation where V2V communication data is fused with radar measurements for dataset 2.

9.4 Conclusion and Recommendations

The performance of the vehicle state estimation algorithm in the different configurations cannot be determined from the experiments due to the absence of ground truth data. The logging of the Twizy-1 data failed, and therefore a workaround is required to determine the yaw rate for the situation where only communication is active. Large measurement errors are found as the heading angle of the vehicles is not parallel to the straight vehicle path at times. The outputs of the vehicle state estimation systems were compared to the Host-tracking outputs for both vehicles. When only communication is active, the EKF with DD corrects the communication delay. Determining the unicycle states from the radar measurements results in a large deviation with the output of the Host-tracking of up to 2.35 m, 0.6 m/s and 0.2°. It cannot be determined whether the Host-tracking or the radar-based state estimation yields more accurate results, but the radar-based estimation results in more noise on the estimated states. When the V2V communication and the radar sensor detects the leading vehicle, the Host-tracking outputs are adjusted to comply with the radar measurements.

With ground truth data available, the accuracy of the vehicle state estimation algorithms can be determined. The accuracy of the Host-tracking algorithm can also be experimentally determined in this way. The switching between the 3 situations is not implemented due to time constraints. For this, switching between KFs should be investigated for smooth transitions when the situation changes. When lateral control is operational on the Twizies, fully autonomous experiments can be executed. During this, the following distances and paths of the vehicles can be controlled more accurately than with human drivers. This increases the validity of comparisons between experiments and simulations.

Chapter 10

Conclusion and Recommendations

The goal of this project is to develop an algorithm that fuses measurements from forward-looking sensors with onboard GNSS and IMU sensors, and V2V communication data for vehicle state estimation in platooning. Depending on whether the V2V communication is operational and the radar sensor detects the leading vehicle, three situations can occur: Only the communication is active, only the radar sensor detects the leading vehicle, or both the V2V communication is operational and the radar detects the leading vehicle. To complete the research objectives, three vehicle state estimation systems are developed for these situations, that estimate the vehicle states of the leading and following vehicles at 100 Hz without delay. Measurements taken in the absolute reference frame and the vehicle fixed reference frame of the following vehicle are fused when the V2V communication is active and the radar detects the leading vehicle. Simulations are executed to analyze the performance of the vehicle state estimation in the three situations. Full-scale experiments are conducted to test the vehicle state estimation systems on experimental data. Additionally, a method to select the radar object which belongs to the leading vehicle based on the estimated vehicle states is developed.

10.1 Conclusions

- When only the V2V communication is available, an EKF based on a unicycle model is used to increase the sampling frequency of the V2V communication to 100 Hz to match the frequency of the real-time model on the onboard PC. Discrete dynamics are used to compensate for the delayed reception of the messages to determine the vehicle states of the leading vehicle. For this, it is assumed that the inputs of the unicycle model remain constant during the delay period of 0.02 s. The vehicle states of the following vehicle are estimated using the Host-tracking algorithm.
- Estimating the states of the leading vehicle is done using solely radar measurements when the V2V communication is not available. The states of the following vehicle are determined by fusing data from the onboard kinematic and GNSS sensors with Host-tracking. The relative distance and velocity between the vehicles is determined with an EKF which uses a unicycle model to fuse the radar measurements. The absolute vehicle position of the leading vehicle is determined by taking the absolute position of the following vehicle and adding the inter-vehicle distances and velocities.
- When the communication and radar sensor are active, Host-tracking is used to determine the vehicle states of the following vehicle. From the V2V communication, the vehicle states of the leading vehicle are determined with the EKF and discrete dynamics. An EKF filter with two unicycle models is used to fuse the vehicle state information with the radar measurements to increase the accuracy of the estimated relative distance and velocity between the vehicles.
- In simulation, where the vehicles are modelled with a dynamic single-track model, the accuracy of the state estimation when only V2V communication is available is 1.8 m in the inertial reference frame. The velocity and heading angle are estimated with an accuracy of 0.02 m/s and 0.01 rad. The inter-vehicle distance is estimated with a maximum error of 2.2 m. When only the radar sensor is used to determine the states of the leading vehicle, the absolute position estimation is accurate up to 1.5 m. The velocity is estimated with a maximal error of 0.62 m/s and the accuracy of the relative distance between both vehicles is 0.4 m. When both are active, the absolute position estimation is accurate up to 0.8 m. The maximal error in velocity is 0.03 m/s and the relative vehicle distance has a maximal error of 0.5 m.
- Estimating the states of the leading vehicle with the radar sensor results in more noise on the estimated states than when V2V communication is used. The state estimation is less accurate for the heading and velocity when only the radar measurements are used. The relative vehicle distance and velocity are more

accurately estimated when the radar is used compared to when only V2V communication is used. Using both the V2V communication and the radar sensor results gives the best performance, yielding more accurate absolute vehicle state estimation than when only the radar is used, and more accurate relative vehicle state estimation than when the communication is used. Using radar measurements to correct the relative position and velocity between the vehicles improves the position estimation in the inertial frame.

- Full-scale experiments were executed where Renault Twizies equipped with extra sensors and a wireless communication link were driven manually. The developed vehicle state estimation systems were tested on the experimental data. The accuracy and performance of the vehicle state estimation systems cannot be determined from the experiments due to the absence of ground truth data. However, the differences in the estimated states from the different methods are within the accuracy of the Host-tracking algorithm.
- A method is developed to select a radar object based on estimated vehicle states of both the leading and following vehicles. The expected radar measurement is calculated based on the estimated vehicle states, and this is compared to the actual radar measurements using the Euclidean distance. Using validity checks, no radar object is selected when it is unlikely that the radar sensor detects the leading vehicle.

10.2 Recommendations

- The main recommendation is to validate the state estimation results from the experiments through the utilization of the RTK-GPS. Due to its high accuracy, the data gathered with the RTK-GPS can be used as ground truth. Using this, the performance of the Host-tracking algorithm, and the vehicle state estimation systems for the three different situations can be determined experimentally.
- When either the V2V communication messages are no longer received, or the radar sensor does not detect the leading vehicle, the sources of information change. When this happens, a switch should be made between the different vehicle state estimation systems. Currently, this is not implemented due to time constraints. Further research could implement this switching and integrate this into the software which runs on the Twizy PC.
- When the vehicles are driven autonomously during experiments, the following distances and driven paths can be controlled more precisely than with human drivers. This results in more similarity between the simulations and measurements, yielding the opportunity to compare the measurements to the simulations. This can point out whether the simulation method represents the real-life situation, and whether the performance of the vehicle state estimation system is consistent between experiments and simulations.
- The vehicles which were used for performing experiments are prototypes. These vehicles are built specifically for running experiments and only three have been produced. Because of this, reliability issues are common on both the software and hardware sides of the vehicle. This resulted in losing experimental data of the leading vehicle, which required a workaround. Redoing these experiments with proper data logging on the leading vehicle would omit the necessity of the workaround.
- The radar matching currently considers one radar object to be the leading vehicle at a time. With the object separation of the radar sensor, multiple radar detections could be the leading vehicle. Further research can provide a method to use multiple radar detections when the leading vehicle is detected multiple times. Due to the absence of ground truth data, it is currently not known which part of the vehicle is detected by the radar sensor. The vehicle's position is defined by the location of the rear axle, and by detecting an arbitrary point on the rear of the vehicle a measurement error is introduced. Furthermore, V2V communication is currently required to determine what radar object is selected. Once a radar object is selected and the communication becomes unavailable, the previous radar object measurement can be used for the state estimation and selection of a radar object. Once the system initializes without the V2V communication, the radar matching algorithm cannot select a radar object because no vehicle state estimation data is available. For this specific case, a method to select a radar object based on where the leading vehicle is expected to be should be developed. A deviation of 8.1° is found in the calibration of the radar sensor. During the post-processing of the experimental data, this was corrected. When the vehicle state estimation systems run on the vehicles in real time, the radar should be calibrated correctly to ensure accurate measurements.
- The developed system heavily relies on the output of the Host-tracking algorithm. Estimation errors in the Host-tracking algorithm will propagate towards the vehicle state estimation system. Improving the accuracy of the Host-tracking algorithm therefore increases the accuracy of the vehicle state estimation system which is developed in this thesis.
- In further research the state estimation system could be extended to a 3D situation. The current solution simplifies the 3D world to a 2D plane where vehicles cannot move vertically. This could result in estimation errors when the vehicles drive on a road with a high inclination angle.

Bibliography

[1] Centraal Bureau voor de Statistiek, "Hoeveel personenauto's zijn er in Nederland?" 10 2022. [Online]. Available: https://www.cbs.nl/nl-nl/visualisaties/verkeer-en-vervoer/vervoermiddelen-en-infrastructuur/personenautos

[2] A. Retallack, A. &Ostendorf, "Current understanding of the effects of congestion on traffic accidents," *International journal of environmental research and public health, 16(18), 3400*, 2019.

[3] K. Bucsuházy, E. Matuchová, R. Zůvala, P. Moravcová, M. Kostíková, and R. Mikulec, "Human factors contributing to the road traffic accident occurrence," *Transportation Research Procedia*, vol. 45, pp. 555–561, 2020, transport Infrastructure and systems in a changing world. Towards a more sustainable, reliable and smarter mobility.TIS Roma 2019 Conference Proceedings. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2352146520302192

[4] W. Brenner and A. Herrmann, *An Overview of Technology, Benefits and Impact of Automated and Autonomous Driving on the Automotive Industry*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2018, pp. 427–442. [Online]. Available: https://doi.org/10.1007/978-3-662-49275-8_39

[5] B. Friedrich, *The Effect of Autonomous Vehicles on Traffic*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 317–334. [Online]. Available: https://doi.org/10.1007/978-3-662-48847-8_16

[6] W. Elmenreich, "An introduction to sensor fusion," *Vienna University of Technology, Austria*, vol. 502, pp. 1–28, 2002. [Online]. Available: https://www.researchgate.net/profile/Wilfried-Elmenreich/publication/267771481_An_Introduction_to_Sensor_Fusion/links/55d2e45908ae0a3417222dd9/An-Introduction-to-Sensor-Fusion.pdf

[7] S. S. R. Hostettler, "Lecture notes on basics of sensor fusion," Uppsala University, Sweden & Aalto University, Finland, 2020. [Online]. Available: https://users.aalto.fi/~ssarkka/pub/basics_of_sensor_fusion_2020.pdf

[8] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME–Journal of Basic Engineering*, vol. 82, no. Series D, pp. 35–45, 1960.

[9] Sep 2023. [Online]. Available: https://en.wikipedia.org/wiki/Kalman_filter

[10] M. I. Ribeiro, "Kalman and extended Kalman filters: Concept, derivation and properties," *Institute for Systems and Robotics*, vol. 43, no. 46, pp. 3736–3741, 2004.

[11] F. Gustafsson and G. Hendeby, "Some relations between extended and unscented Kalman filters," *IEEE Transactions on Signal Processing*, vol. 60, no. 2, pp. 545–555, 2012.

[12] N. van der Wouw, "Multibody dynamics — lecture notes," Eindhoven University of Technology, 2016.

[13] *International Standard ISO 8855 — Road vehicles — Vehicle dynamics and road-holding ablility — Vocabulary*, ISO, Geneva, Switzerland, 2011.

[14] I. A.-S. A. Hossain, Md. A. Elsafiey, "Cooperative vehicle positioning with multi-sensor data fusion and vehicular communications," *Wireless Networks*, vol. 25, no. 3, pp. 1403–1413, 2019, 9th IFAC Symposium on Advances in Automotive Control AAC 2019. [Online]. Available: https://link.springer.com/article/10.1007/s11276-018-1772-6

[15] E. Dawson, M. A. Rashed, W. Abdelfatah, and A. Noureldin, "Radar-based multisensor fusion for uninterrupted reliable positioning in GNSS-denied environments," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 12, pp. 23 384–23 398, 2022.

[16] F. Caron, E. Duflos, D. Pomorski, and P. Vanheeghe, "GPS/IMU data fusion using multisensor Kalman filtering: introduction of contextual aspects," *Information Fusion*, vol. 7, no. 2, pp. 221–230, 2006. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S156625350400065X

[17] Y. Yin, J. Zhang, M. Guo, X. Ning, Y. Wang, and J. Lu, "Sensor fusion of GNSS and IMU data for robust localization via smoothed error state Kalman filter," *Sensors*, vol. 23, no. 7, 2023. [Online]. Available: https://www.mdpi.com/1424-8220/23/7/3676

[18] V. Rodrigo Marco, J. Kalkkuhl, J. Raisch, W. J. Scholte, H. Nijmeijer, and T. Seel, "Multi-modal sensor fusion for highly accurate vehicle motion state estimation," *Control Engineering Practice*, vol. 100, p. 104409, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S096706612030068X

[19] V. Khattar, "Multi-object tracking using sensor fusion," PdEng thesis, Mechanical Engineering, Technische Universiteit Eindhoven, Oct. 2020. [Online]. Available: https://pure.tue.nl/ws/portalfiles/portal/173550145/2020_11_19_ASD_track_ASD_Khattar_V_100.pdf

[20] Q. Chen, T. Roth, T. Yuan, J. Breu, F. Kuhnt, M. Zöllner, M. Bogdanovic, C. Weiss, J. Hillenbrand, and A. Gern, "Dsrc and radar object matching for cooperative driver assistance systems," in *2015 IEEE Intelligent Vehicles Symposium (IV)*, 2015, pp. 1348–1354.

[21] T. Yuan, T. Roth, Q. Chen, J. Breu, M. Bogdanovic, and C. A. Weiss, "Track-to-track association for object matching in an inter-vehicle communication system," in *Signal and Data Processing of Small Targets 2015*, O. E. Drummond and R. D. Teichgraeber, Eds., vol. 9596, International Society for Optics and Photonics. SPIE, 2015, p. 959609. [Online]. Available: https://doi.org/10.1117/12.2187108

[22] F. Hoogeboom, "Safety of automated vehicles: design, implementation, and analysis," Ph.D. dissertation, Mechanical Engineering, Technische Universiteit Eindhoven, Oct 2020. [Online]. Available: https://pure.tue.nl/ws/portalfiles/portal/162391928/20201027_Hoogeboom.pdf

[23] S. Campbell, "Steering control of an autonomous ground vehicle with application to the darpa urban challenge," *Massachusetts Institute of Technology. Dept. of Mechanical Engineering*, January 2007. [Online]. Available: https://dspace.mit.edu/handle/1721.1/42301

[24] A. Bayuwindra, "Look-ahead tracking controllers for integrated longitudinal and lateral control of vehicle platoons," Ph.D. dissertation, Mechanical Engineering, Technische Universiteit Eindhoven, Oct. 2019, phd thesis. [Online]. Available: https://pure.tue.nl/ws/portalfiles/portal/136571512/20191030_Bayuwindra.pdf

[25] G. M. Hoffmann, C. J. Tomlin, M. Montemerlo, and S. Thrun, "Autonomous automobile trajectory tracking for off-road driving: Controller design, experimental validation and racing," in *2007 American Control Conference*, 2007, pp. 2296–2301.

[26] E. Lefeber, J. Ploeg, and H. Nijmeijer, "A spatial approach to control of platooning vehicles: Separating path-following from tracking," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 15 000–15 005, 2017, 20th IFAC World Congress. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S240589631733495X

[27] *ETSI EN 302 663, Intelligent Transport Systems (ITS); Access layer specification for Intelligent Transport Systems operating in the 5 GHz frequency band*, etsi, Sophia Antipolis, France.

[28] *ETSI EN 302 637-2, Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 2: Specification of Cooperative Awareness Basic Service*, etsi, Sophia Antipolis, France.

[29] *Acceleration Sensor MM5.10*, Bosch, Abstatt, Germany. [Online]. Available: https://www.bosch-motorsport.com/content/downloads/Raceparts/Resources/pdf/Data%20Sheet_68903691_Acceleration_Sensor_MM5.10.pdf

[30] *EVK-M8T Evaluation Kit*, u-blox, Thalwil, Switzerland. [Online]. Available: https://content.u-blox.com/sites/default/files/products/documents/EVK-M8T_UserGuide_%28UBX-14041540%29.pdf

[31] *Mid-range radar sensor (MRR) for front and rear applications*, Bosch, Abstatt, Germany. [Online]. Available: https://fccid.io/ANATEL/00850-13-03745/Manual-para-refer/F235B5AA-6AEA-408E-A9F5-4D1C40CC1EB9/PDF

[32] W. Scholte, "Control for cooperative merging maneuvers into platoons," Ph.D. dissertation, Mechanical Engineering, Technische Universiteit Eindhoven, Sep. 2022. [Online]. Available: https://pure.tue.nl/ws/portalfiles/portal/212613665/20220913_Scholte_hf.pdf

[33] I. Besselink, "Vehicle dynamics - 4AT000 - lecture notes," 2020.

[34] The MathWorks Inc., "Navigation Toolbox: Version 2.4 (R2023a)," Natick, Massachusetts, United States, 2023. [Online]. Available: https://nl.mathworks.com/products/navigation.html

[35] Y. Bulut, D. Vines-Cavanaugh, and D. Bernal, "Process and measurement noise estimation for Kalman filtering," in *Structural Dynamics, Volume 3*, T. Proulx, Ed. New York, NY: Springer New York, 2011, pp. 375–386.

[36] Renault Group, "All there is to know about Renault Twizy," 12 2023. [Online]. Available: https://www.renaultgroup.com/en/news-on-air/news/all-there-is-to-know-about-renault-twizy/

[37] B. G. Simons-Morton, M. C. Ouimet, Z. Zhang, S. E. Klauer, S. E. Lee, J. Wang, P. S. Albert, and T. A. Dingus, "Crash and risky driving involvement among novice adolescent drivers and their parents," *American Journal of Public Health*, vol. 101, no. 12, pp. 2362–2367, 2011, pMID: 22021319. [Online]. Available: https://doi.org/10.2105/AJPH.2011.300248

[38] A. Jahangiri, V. Berardi, and S. Ghanipoor Machiani, "Application of real field connected vehicle data for aggressive driving identification on horizontal curves," *IEEE Transactions on Intelligent Transportation Systems*, vol. PP, pp. 1–9, 12 2017.

[39] K. J. Aström and B. Wittenmark, *Computer Controlled Systems: Theory and Design 3rd ed.* Hoboken, NJ: Prentice Hall, 1997.

[40] F. E. Szabo, in *The Linear Algebra Survival Guide*, F. E. Szabo, Ed. Boston: Academic Press, 2015, pp. 234–252. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780124095205500217

Appendix A

Radar Matching Results

This appendix shows the expected and selected radar measurements for all four relative measurements of the radar matching algorithm discussed in Chapter 8. The three developed methods are tested on two data sets discussed in Chapter 7. Figure A.1 shows the results for all radar measurements of the three distance calculation methods for the first data set. Figure A.2 shows the full results for the second data set. It can be seen that the results for the other radar measurements reflect the results of the relative longitudinal distance discussed in Chapter 8, as the Manhattan and Mahalanobis methods show outliers. It can also be seen that the lateral measurements are less comparable to the expected measurements than the longitudinal results. The offset in the lateral distance appears for all three methods in both datasets, proving that the cause lies with the radar sensor itself instead of the handling of the measurements. This offset is caused by an offset between the radar calibration angle and the mounting as discussed in Chapter 8.

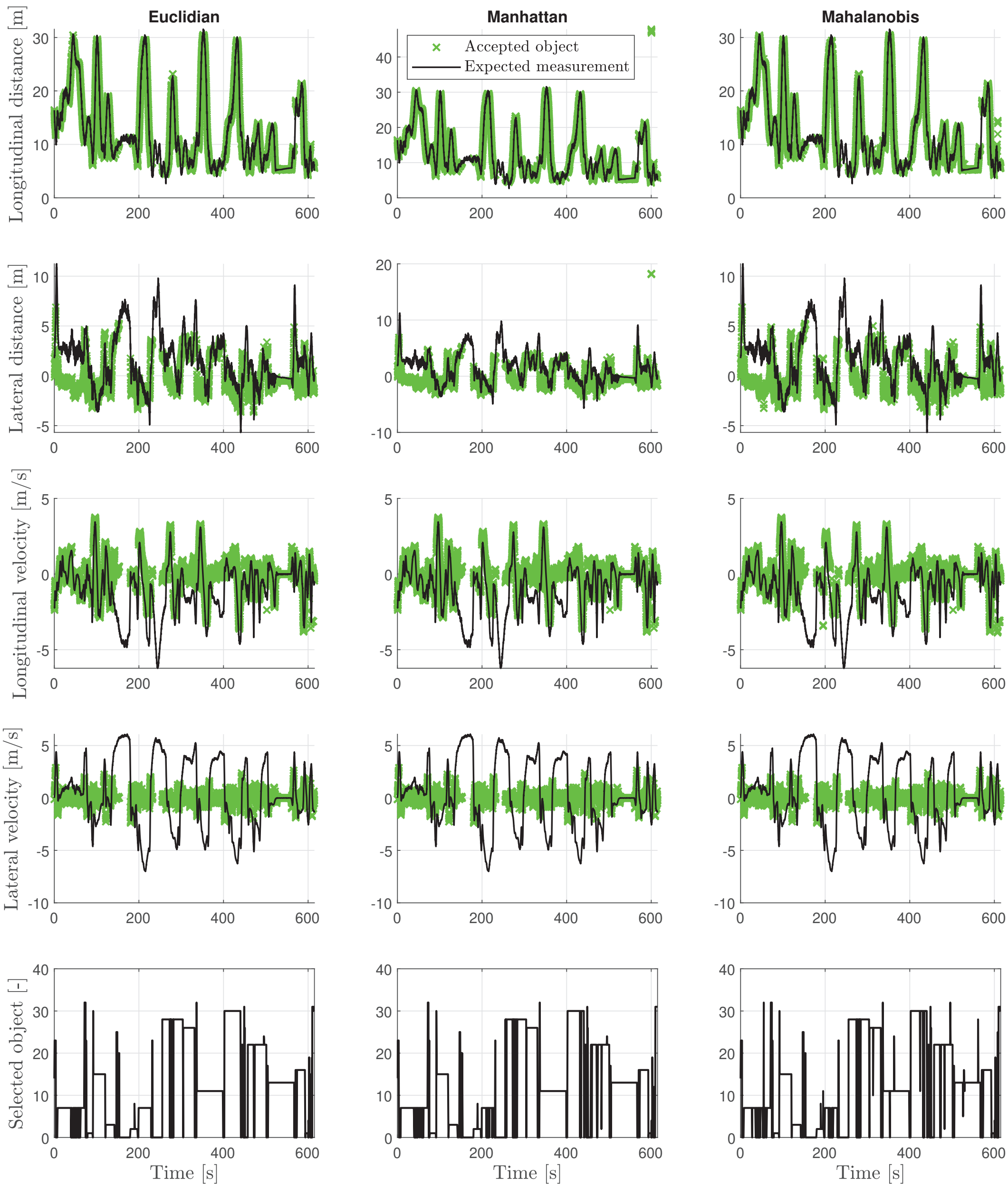


Figure A.1: The selected radar measurements compared to the expected measurements for all methods for data set 1.

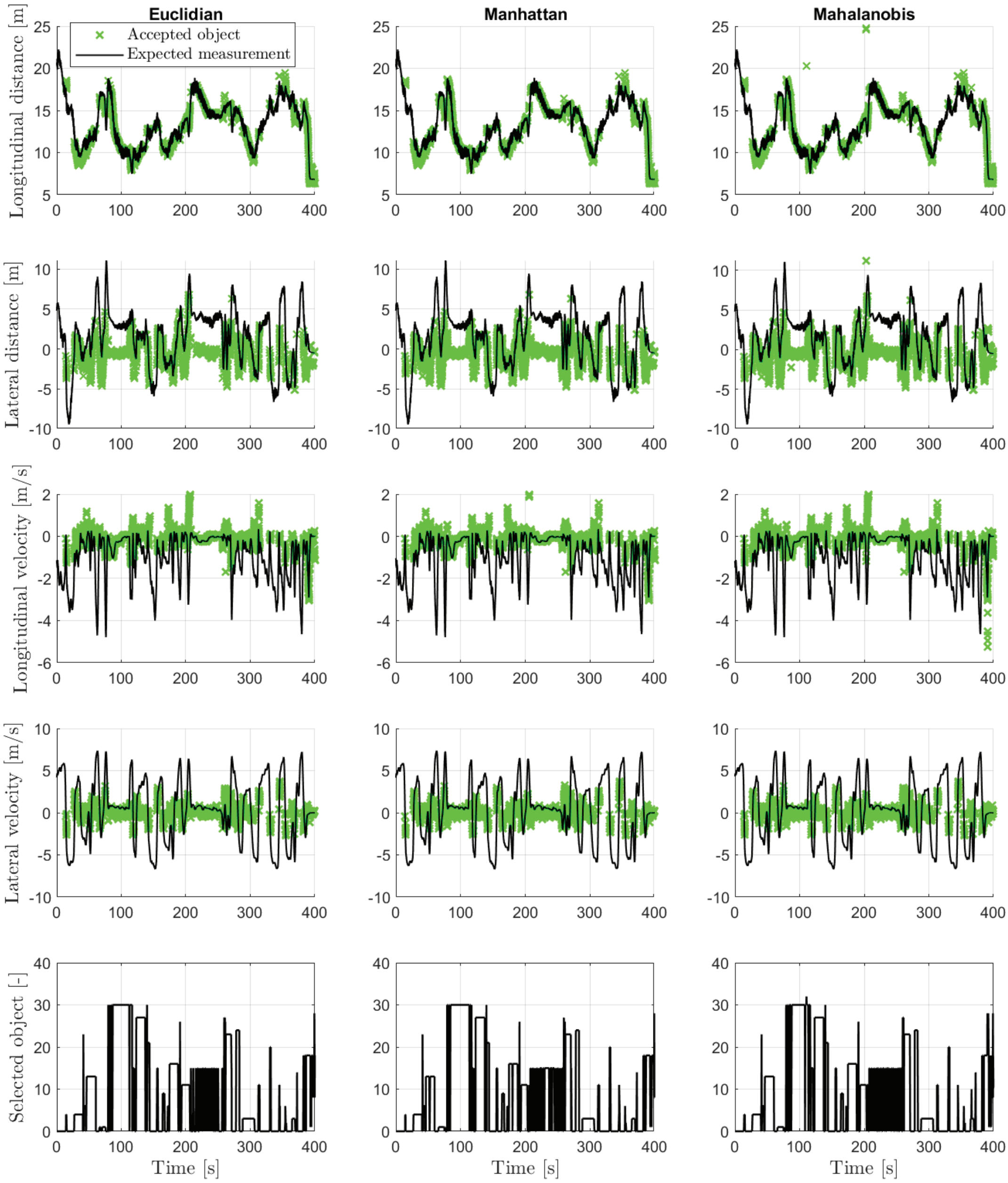


Figure A.2: The selected radar measurements compared to the expected measurements for all methods for data set 2.

Appendix B

Influence of Sensors Models with Randomized Noise

To demonstrate the influence of the randomized sensor models on the results, five simulations are performed where the radar sensor and the V2V communication are active. The sensors are simulated using equal parameters in each simulation. The vehicle states are estimated using the state estimator discussed in Section 6.3. The EKF measurement and process noise matrices, and the inputs to the single-track models also remain equal during the different simulations. The noises of the sensors use a randomized seed to have different noises on all sensors. Figure B.1 shows the estimation error for the vehicle 1 states for the different simulations. It can

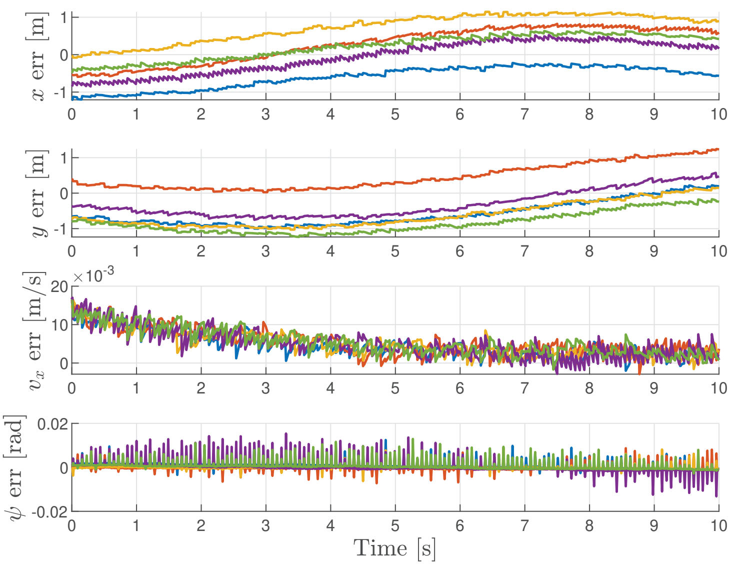


Figure B.1: Errors for vehicle 1 state estimates for multiple simulations when driving in a circle with constant velocity with randomized sensors.

be seen that the magnitudes of the x and y-position errors vary between runs. For some runs the maximal x-position error is 0.5 m, whereas for a different run, this error goes up to 1.1 m. This is an increase of 120 %. For the different simulations, the maximal error in y-position estimation is between 0.6 m and 1.2 m. For the velocity estimation, the largest and smallest error magnitudes are 0.010 and 0.013 m/s. The estimation error magnitudes for the heading angle are similar between the different simulations, with changes of 0.001 rad. The large difference in position estimations is explained because the GPS measurement can have an error of up to 2 m. The Host-tracking block relies heavily on this GPS sensor, and different GPS measurements therefore result in large differences in the outputs. The sensors measuring the velocity and yaw rate have better accuracies at 0.125 m/s and 0.1°. Therefore, randomizing the noises for these sensors has less effect on the overall results. The radar sensor mainly influences the noise which is placed upon the estimated states. It can be seen that this changes for the different simulations, but does not change large deviation caused by the GPS sensor.

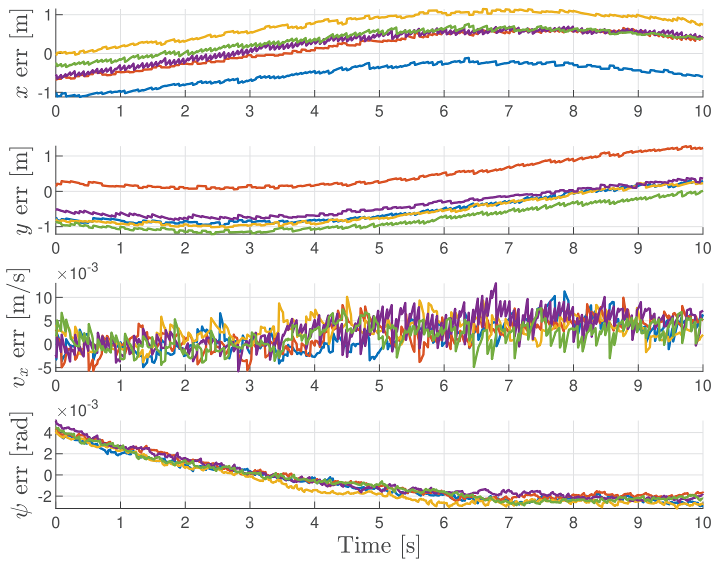


Figure B.2: Errors for vehicle 2 state estimates for multiple simulations when driving in a circle with constant velocity with randomized sensors.

Figure B.2 shows the state estimations of vehicle 2 with randomized sensors for driving in a circle with constant velocity. For the state estimates of vehicle 2, the same error magnitude changes are seen between the different simulations. The x and y-position magnitudes change up to 0.6 m. For the velocity and heading angle the impact of the randomized sensors is smaller, as was the case for vehicle 1.

Figure B.3 shows the relative error for the different simulations with randomized sensor noises for driving in a circle with a constant velocity.

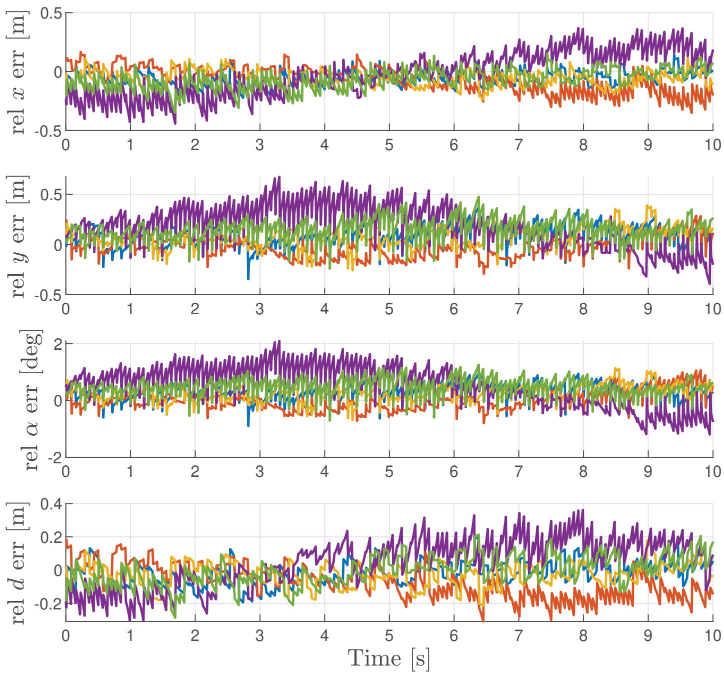


Figure B.3: Relative vehicle state errors for multiple simulations when driving in a circle with constant velocity with randomized sensors.

The maximal relative longitudinal position error varies between 0.35 and 0.15 m for the different simulations. This is an increase of 133%. For the relative lateral position error, the maximal error varies between 0.6 m and 0.4 m. For the relative angle between the vehicles, the maximal error is 2° for the simulation with the largest error and 0.9° for the simulation with the smallest error, which is a difference of 122%.

It can be concluded that the method used for simulating sensors with random noise seeds has a large impact on the accuracy of the state estimation algorithm. The magnitude of the estimation errors between different simulations with the same settings can vary up to 120% for the position estimates. For the relative vehicle states the error magnitude between different simulations varies up to 133% for relative distances and up to 122% for the relative angle between the vehicles.