



Modelling the Level of Trust in a Cooperative Automated Vehicle Control System

Thomas Rosenstatter

Embedded and Intelligent Systems, 30 credits

Halmstad University, September 18, 2016

Thomas Rosenstatter: *Modelling the Level of Trust in a Cooperative Automated Vehicle Control System*, © September 18, 2016

SUPERVISOR:
Cristofer Englund

EXAMINERS:
Antanas Verikas
Slawomir Nowaczyk

LOCATION:
Halmstad, Sweden

ABSTRACT

Vehicle-to-Vehicle communication is the key technology for achieving increased perception for automated vehicles where the communication allows virtual sensing with the use of sensors placed in other vehicles. In addition, this technology also allows recognising objects that are out-of-sight. This thesis presents a Trust System that allows a vehicle to make more reliable and robust decisions. The system evaluates the current situation and generates a Trust Index indicating the level of trust in the environment, the ego vehicle, and the other vehicles. Current research focuses on securing the communication between the vehicles themselves, but does not verify the content of the received data on a system level. The proposed Trust System evaluates the received data according to sensor accuracy, behaviour of other vehicles, and the perception of the local environment. The results show that the proposed method is capable of correctly identifying various situations and discusses how the Trust Index can be used to make more robust decisions.

Nobody should start to undertake a large project. You start with a small trivial project, and you should never expect it to get large. If you do, you'll just overdesign and generally think it is more important than it likely is at that stage. Or worse, you might be scared away by the sheer size of the work you envision. So start small, and think about the details. Don't think about some big picture and fancy design. If it doesn't solve some fairly immediate need, it's almost certainly over-designed. And don't expect people to jump in and help you. That's not how these things work. You need to get something half-way useful first, and then others will say "hey, that almost works for me", and they'll get involved in the project.

— Linus B. Torvalds [1]

ACKNOWLEDGEMENTS

I would like to thank my thesis supervisor Cristofer Englund from the Halmstad University for giving me his advice and providing me with ideas from other perspectives.

As this thesis was strongly related to the Grand Cooperative Driving Challenge 2016, I would also like to thank the entire team Halmstad and our team leaders Wojciech Mostowski and Maytheewat Aramrattana for their support. I also want to thank all the companies who sponsored us with equipment or software.

Thank you.

Thomas Rosenstatter

Contents

List of Figures	ix
List of Tables	xi
Listings	xii
Acronyms	xiii
1 INTRODUCTION	1
1.1 Goal and Approach	2
1.2 Contribution	4
1.3 Thesis Outline	4
2 RELATED WORK	7
2.1 Sensor Fusion	8
2.1.1 Kalman Filter	9
2.1.2 Extended Kalman Filter	11
2.1.3 Particle Filter	13
2.1.4 R�sum�	14
2.2 Trust in Vehicular Ad-Hoc Networks	15
2.2.1 Security Challenges	15
2.2.2 Trust Establishment	16
2.2.3 Trust and Reputation Models	17
2.3 Situation Awareness	21
2.4 Summary	21
3 DATA ACQUISITION	23
3.1 Linear Model	25
3.2 Non-linear Model	27
3.3 Summary	31
4 CONCEPT	33
4.1 Trust System	35
4.1.1 Trust Index	36
4.1.2 Trust Index TI_{ego}	37
4.1.3 Trust Index TI_{mio}	40
4.1.4 Trust Index TI_{env}	43
4.1.5 Trust Index TI_{vi}	45
4.2 V2V Perception	46
4.3 Decision Making	47
4.4 Summary	48
5 IMPLEMENTATION	49
5.1 Architecture	50
5.1.1 Sensor Fusion	50
5.1.2 Trust System	52

5.1.3	V2V Perception	56
5.1.4	Decision Making	58
5.2	Information Flow	59
5.3	Summary	60
6	EXPERIMENTS	61
6.1	Scenarios	61
6.2	Accomplishment	63
6.3	Results	64
6.3.1	Left Lane	64
6.3.2	Right Lane	66
6.3.3	Unreliable Geographical Position	67
6.3.4	Comparison of the Behaviour Identifier	68
7	CONCLUSION AND FUTURE WORK	71
7.1	Future Work	72
A	LINEARISED KALMAN FILTER EQUATIONS	73
B	ARCHITECTURE	75
B.1	State Machine	77
	BIBLIOGRAPHY	81

List of Figures

Figure 2.1	Technique of centralised and decentralised filtering.	9
Figure 2.2	Operation cycle of a Kalman Filter.	11
Figure 2.3	Operation cycle of an Extended Kalman Filter.	13
Figure 2.4	Categorisation of trust establishment techniques.	16
Figure 2.5	Entity-based trust management.	19
Figure 3.1	Sensor Environment with Vehicle-to-Vehicle communication.	23
Figure 3.2	Linear Sensor Fusion model.	25
Figure 3.3	Activity Diagram for the linear model.	27
Figure 3.4	Distance fusion with a Kalman Filter.	28
Figure 3.5	Activity Diagram for the non-linear model.	31
Figure 4.1	Factors influencing the awareness of a vehicle.	34
Figure 4.2	Composition of the Data Acquisition module.	35
Figure 4.3	Data calculated by the Trust System.	36
Figure 4.4	Composition of TI_{ego}	39
Figure 4.5	Factors describing the behaviour of a vehicle.	40
Figure 4.6	Composition of TI_{mio}	44
Figure 4.7	Types of environment in which a vehicle is driving.	45
Figure 4.8	Illustration of the map and the classification of the surrounding vehicles.	47
Figure 5.1	The GCDC competition car from team Halmstad.	49
Figure 5.2	The trunk of the competition car with its devices.	50
Figure 5.3	Class diagram of the KF	51
Figure 5.4	Class diagram of the non-linear model.	52
Figure 5.5	Class diagram of trust indices.	53
Figure 5.6	Observed interaction between the Most Important Object and other vehicles.	54
Figure 5.7	Illustration of the dynamic angle range calculation.	57
Figure 5.8	System Architecture of the GCDC car.	60
Figure 6.1	Phases of the GCDC highway scenario.	62
Figure 6.2	Map where the highway scenario took place.	63
Figure 6.3	Experiment I: Highway scenario starting from the left lane.	65
Figure 6.4	Experiment II: Highway scenario starting from right lane.	67

Figure 6.5	Unreliable/Inaccurate geographical position.	68
Figure 6.6	Approaches to detect speed fluctuation.	69
Figure B.1	The overall system architecture of the GCDC car including LCM message names.	75
Figure B.2	Class diagram of the Trust System.	76
Figure B.3	State machine for the GCDC scenario 1.	77
Figure B.4	State machine for the GCDC scenario 2.	78
Figure B.5	State machine for the GCDC scenario 3.	79
Figure B.6	State machine for going back to manual mode.	79

List of Tables

Table 3.1	Relevant measurements in a Vehicular Ad-Hoc Network environment.	24
Table 4.1	Listing of the proposed trust indices.	38
Table 4.2	Description of certain trust levels.	38

Listings

Listing 1	Pseudo statement for the pairing index i	54
Listing 2	Pseudo statement for the pairing index ii	55

ACRONYMS

ACC	Adaptive Cruise Control
AI	Artificial Intelligence
CACC	Cooperative Adaptive Cruise Control
CAM	Cooperative Awareness Message
CAN	Controller Area Network
CC	Cruise Control
DA	Data Acquisition
DENM	Decentralised Environmental Notification Message
DOP	Dilution of Precision
ECU	Electronic Control Unit
EKF	Extended Kalman Filter
ENU	East, North, and Up
ETSI	European Telecommunications Standards Institute
GCDC	Grand Cooperative Driving Challenge
GPS	Global Positioning System
HDOP	Horizontal DOP
HLC	High-Level Control
HMI	Human Machine Interface
iCLCM	i-GAME Cooperative Lane Change message
i-GAME	Interoperable GCDC AutoMation Experience
KF	Kalman Filter
LCM	Lightweight Communications and Marshalling
LLA	Longitude, Latitude, and Altitude
LLC	Low-Level Control
MAS	Multi-Agent System
MIO	Most Important Object
MLC	Mid-Level Control
OPC	Organisation Pace Car
PDOP	Position DOP
PF	Particle Filter

RSU	Road Side Unit
SF	Sensor Fusion
STOM	Safe-To-Merge
TI	Trust Index
TS	Trust System
VANET	Vehicular Ad-Hoc Network
VDOP	Vertical DOP
V2I	Vehicle-to-Infrastructure
V2V	Vehicle-to-Vehicle
V2X	Vehicle-to-Everything

INTRODUCTION

Yearly more than 1.2 million people die in road accidents, making it to the globally leading cause of death [2]. The increasing number of vehicles on public roads and the goal to reduce the environmental pollution combined with the technological progress leads to research and development in the area of autonomous and cooperative driving. Davila et al. [3] published a paper in context of the SATRE project where they investigate the benefits of platooning systems. Their conclusion is that platooning is safer and for the reason that the vehicle control system is fully autonomous, vehicle dynamics are optimised as well. Autonomous vehicles also increase the comfort for the passengers. This thesis is introducing a Trust System (TS) that evaluates the current situation and thus supports the vehicle control system in decision making. Software has also become the major area of innovation within a vehicle, more than 80 percent of the novelty is achieved by computer systems and their software [4]. The telematic systems of Daimler and Kia use the Internet for exchanging vehicle status information and automatic calls to the emergency service number in case an accident occurred [5]. Publicly known projects, such as the Self-Driving Car Project¹ from Google, demonstrate the technological progress over the past years.

Autonomous driving aims to perceive the environment with the own sensors of the vehicle, e. g. Global Positioning System (GPS), radar, lidar, and camera. Due to the high price of long range and wide angle sensors, and the limitation of proximity sensors to only detect line-of-sight objects, cooperative driving turned out to be a reasonable alternative to expensive sensors with the advantage of perceiving also out-of-sight objects via wireless exchange of local information [6]. Further, cooperative driving can be a key technology to increase traffic safety and efficiency [7].

One vital application for Vehicle-to-Vehicle (V2V) communication is platooning or Cooperative Adaptive Cruise Control (CACC). However, following only the vehicle in front by measuring the speed of the vehicle and distance to the ego vehicle does not significantly increase the efficiency of the entire platoon. With the use of wireless communication to exchange sensor information, time gaps of less than one second can be achieved [8]. Another field of application is a cooperative interaction to safely cross an intersection.

¹ See: <https://www.google.com/selfdrivingcar/> (2016/06/12)

The proposed system described in this thesis has been designed for the Grand Cooperative Driving Challenge (GCDC)² 2016 and has been tested and evaluated in the course of this competition. The GCDC is the result of the Interoperable GCDC AutoMation Experience (i-GAME)³ project, supported by the European Commission. i-GAME combines research and demonstration for an interoperable exchange of messages between vehicles and infrastructure.

The GCDC describes three scenarios that are performed in a cooperative and automated manner. These scenarios are performed with the support of V2V communication. Scenario 1 describes a cooperative approach for automated merging on a two lane highway, when one lane becomes closed because of a roadwork. In the second scenario three vehicles enter a T-intersection at the same time and with the same velocity. The two vehicles on the main road allow the vehicle coming from the perpendicular road to enter the main street in a collaborative way, so that none of the vehicles needs to stop. In Scenario 3, the emergency scenario, an emergency vehicle informs the traffic participants about its appearance and the other vehicles cooperatively create space for the emergency vehicle [9, 10].

1.1 GOAL AND APPROACH

The goal of this project is to design, develop and test a framework for a cooperative and automated vehicular system that allows the control system of the vehicle to perform more robust decisions based on a Trust Index (TI) of other vehicles according to their behaviour and data provided via V2V and Vehicle-to-Infrastructure (V2I) communication. For safety reasons, a robust system should be developed to be able to handle situations even though the received data is wrong or inaccurate [5].

V2V communication offers new ways of designing the system of a vehicle, but how can one ensure that the data accuracy is sufficient? Or that the behaviour of the other vehicles is perfect or good enough? Many researches consider the hardware layer for securing the communication between vehicles. Securing the communication with methods, e. g. use of certificates, are described in [11]. These methods assure that the messages have not been modified during the transmission, they do not guarantee the accuracy of the data and proper handling of errors made by other vehicles.

The overall topic of this thesis is to:

Investigate how to design a framework for a cooperative and automated vehicle that can perform more robust decisions based on trust and awareness?

² See: <http://gcdc.net/en/> (2016/06/12)

³ See: <http://gcdc.net/en/i-game> (2016/06/12)

and it is split into two more specific research questions:

- *Can Artificial Intelligence (AI) be used to create trust between vehicles based on their current and historical performance?*
- *Can trust improve the situation awareness in order to perform more robust decisions?*

To design a framework that performs more robust decisions based on trust and awareness, the system of the ego vehicle needs to analyse the data provided by the other vehicles, their driving behaviour and their reaction to misbehaving vehicles on the road. Further, it is necessary to analyse the vehicle's own performance to make it comparable to the others. Evaluating the vehicles according to their reliability is not enough for making more robust decisions. It is inevitable to keep track of the environment and adapt to it. For instance is the GPS position on rural streets more accurate than in cities due to the reflection of the electromagnetic waves. The TI can be designed as an absolute value with a certain range, but it is also possible to plan a TI that describes the performance of the other vehicle relative to the ego vehicle. Besides the driving behaviour, the TI can consider the current packet loss of V2V messages and the environment, as well.

The first subquestion about applicability of AI to create a trust measurement is answered within this thesis. AI offers a large variety of algorithms that can be used for this purpose. The two relevant major fields are *Probabilistic Reasoning over Time* and *Learning Algorithms*, such as classification with the use of a Support Vector Machine or Neural Networks. The Kalman Filter (KF) is an algorithm belonging to the first mentioned area. This filter tries to estimate the new states of a given model by taking the observations over time into account. Models used with this filter are required to follow a linear Gaussian distribution. In case that the model cannot be assumed to be linear Gaussian distributed, the Extended Kalman Filter (EKF) can be used [12]. One notable benefit of using one of the above mentioned filter techniques is the possibility to fuse the received sensor data in order to create a better estimation of the vehicle's position [13].

Creating a reliable TI of the surrounding vehicles and the ego vehicle to make more robust decisions is the goal of this thesis. To solve this issue, the first step is to investigate the applicability of filters that are based on *Probabilistic Reasoning over Time*. Using this results as basis allows the further investigation of how the TI can be assigned. The following design and implementation of the system is going to be tested and evaluated in this thesis.

1.2 CONTRIBUTION

So far, many publications focus on how to establish trust between agents or vehicles within a Vehicular Ad-Hoc Network (VANET). The authentication of nodes is essential for Vehicle-to-Everything (V2X) communication as to be ready to counter attacks and other malicious actions. The set up of a trustworthy connection between the nodes at the communication level is the base for further security related applications, such as a TS that evaluates the sensor accuracy and the behaviour of the vehicles. The different types of trust establishing techniques are explained in [11]. Further, Zhang explains in [14] current techniques to model trust. None of these methods deal with the sensor quality nor the behaviour of the vehicles or nodes. They are only evaluating if the node sends the correct events, e.g. slippery road or traffic jam.

Considering the V2V information as a new virtual sensor and estimating a better accuracy using Sensor Fusion (SF) has been described in [15]. The result of this publication has been verified within a simulation environment.

This thesis' contribution is the design of a TS that creates a TI by considering the quality of the sensor information provided by the other traffic participants, their behaviour, and the environment itself. The use of this TS allows the decision-making controller to make more robust decisions when interacting with a specific vehicle. Further, this TS has been tested in a Volvo S60 during the GCDC 2016 in Helmond, the Netherlands.

1.3 THESIS OUTLINE

Chapter 2 describes the existing research in the area of SF, trust in VANETs, and situation awareness. The SF algorithms, such as the KF, EKF, and Particle Filter (PF) are briefly described in Section 2.1. Current research in the area of establishing trust in VANETs, and methods to model trust are summarised in Section 2.2. Section 2.3 discusses the impact of automation on situation awareness.

Chapter 3 gives details about the sensor data used for generating the Trust Index, and the linear and non-linear relation between these measurements. The linear model, presented in Section 3.1, is using a KF that fuses the distance measured by the radar and the distance calculated with the use of the geographical position of both vehicles. Section 3.2 explains the EKF for improving the position accuracy of the vehicle by considering the measurements of inertial sensors.

The concept of the developed TS, the representation of the V2V perception, and the decision making algorithm are shown in Chapter 4. Section 4.1 provides detailed information of the generated TI. Further, Section 4.2 explains the algorithm for identifying the surrounding

vehicles. The decision making, described in [Section 4.3](#), has been implemented as a state machine.

The overall architecture of the vehicle's system is presented in [Chapter 5](#). [Section 5.1](#) focuses on the implementation of the proposed [TS](#), the [SF](#), and the perception of the surrounding vehicles. The flow of the information exchanged between the software modules is explained in [Section 5.2](#).

[Chapter 6](#) presents the behaviour of the [TS](#) when performing the [GCDC](#) highway scenario. The possible impacts on the decision making are further discussed in this chapter. [Chapter 7](#) concludes the thesis and points out future research based on the introduced [TS](#).

RELATED WORK

This chapter defines the problem that is being solved by this thesis and the current studies in this area in order to clarify this thesis' innovation. The novelty of this project is about how to create trust between vehicles based on the information gathered via V2V communication and the on-board sensors of the vehicle. It is important to point out that this thesis does not consider security mechanisms on the physical layer. The focus is on the verification of the provided data by using AI in combination with the observations of the ego vehicle.

Trust to the other vehicles is an important component of inter-vehicle communication. Knowing about how much trust you can put in the others data is an essential task for building a robust system that relies on V2V communication. The problem of considering the other vehicle's data for making decisions is that one cannot trust all vehicles to the same extent. Due to different GPS devices, different accuracies of the provided sensor data, and different control systems, it is important to distinguish between various levels of trust.

There are already message sets for V2V communication that exchange the basic information between vehicles and infrastructure. The Cooperative Awareness Message (CAM) is providing information such as the vehicle type, geographical position, lane number, yaw rate, velocity, and acceleration [16, 17]. This message set is essential for inter-vehicular communication and it already provides information about the accuracy of the measurement. However, relying only on the provided accuracy details does not ensure that the given data follows that accuracy, for the reason that the precision is highly dependent on the current environment. Decentralised Environmental Notification Message (DENM) is a message set focussing on V2I communication. This message set is mostly used for Cooperative Road Hazard Warning applications to warn vehicles using event codes, e. g. road work warning, emergency vehicle approaching, and slippery road [17, 18]. The i-GAME specific protocol that has been developed and tested at the GCDC is named i-GAME Cooperative Lane Change message (iCLCM). This message contains scenario-specific fields and additional information for cooperative driving, e. g. time headway, bearing, range rate, and target longitudinal acceleration [19].

Section 2.1 describes the SF and filtering methods that have been considered for this thesis. A comparison of the introduced methods can be taken from Section 2.1.4. The subsections about the different

The time headway is the time gap to the vehicle in front [19].

filters functions as a short explanation, a detailed description can be taken from the sources used in these subsections. Current research, concerning trust and awareness, is covered in [Section 2.2](#). The security challenges in VANETs is described in [Section 2.2.1](#). The next step after identifying the challenges in V2V communication, is authentication. [Section 2.2.2](#) lists the different methods to establish trust on the system level with the focus on authentication. Current research concerning trust models for VANETs and other areas is given in [Section 2.2.3](#). Situation awareness becomes more important the higher the level of automation of a vehicle is. Current methods on how to increase situation awareness can be taken from [Section 2.3](#). A brief summary can be taken from [Section 2.4](#).

2.1 SENSOR FUSION

The aim of SF is to combine the sensor information of independent observations to achieve a better estimation compared to the value of only one sensor. Hall and Llinas compare in [20] the use of multi-sensor data fusion with the human ability to combine several senses. For instance are human beings and animals capable of combining the visual and acoustic perception to see around a corner or through vegetation [20].

The SF can be categorised according to [21] in three different techniques: the centralised fusion, the autonomous/distributed fusion, and the hybrid fusion. These techniques have different approaches of where the filtering of the raw sensor data is going to be performed. The first step in all cases is the preprocessing of the data, which means the conversion *“from sensor-based units and coordinates to convenient coordinates and units for central processing”* [20, p. 15]. [Figure 2.1a](#) illustrates the concept of centralised fusion, which means that the filtering of the raw data and the fusion is performed within the same module. The distributed fusion, illustrated in [Figure 2.1b](#), performs the filtering and the fusion in two different locations. The filter is applied to each observation variable independently and thus, the fusion process has only access to the estimates and their covariance. This technique has the advantage of a reduced information load that is transmitted to the core fusion process. When extracting features from an image, the transmission of the raw data uses more resources than the transmission of the estimates. The distributed fusion is for this kind of sensor data more suitable than the centralised fusion. The drawback of this technique is that the fusion process does only have access to the estimates and therefore, the result is typically less accurate compared to the centralised fusion. The third technique, the hybrid fusion, combines the aforementioned techniques and applies the fusion of the sensor data and the estimates dynamically [20, 21].

The result of SF is the state \hat{x}_i and the covariance estimation P_i . [Figure 2.1](#) illustrates a filter with the observations y_1 and y_2 as inputs

and the outputs \hat{x} and P . The centralised filter in Figure 2.1a receives the raw data of both sensors as input and computes also the SF. Figure 2.1b shows two independent filter systems for each observation. The output of these filters is the estimated state and the covariance estimation based on its source data. The fusion unit computes the final estimation based on all its \hat{x}_i and P_i received by the filters [21].

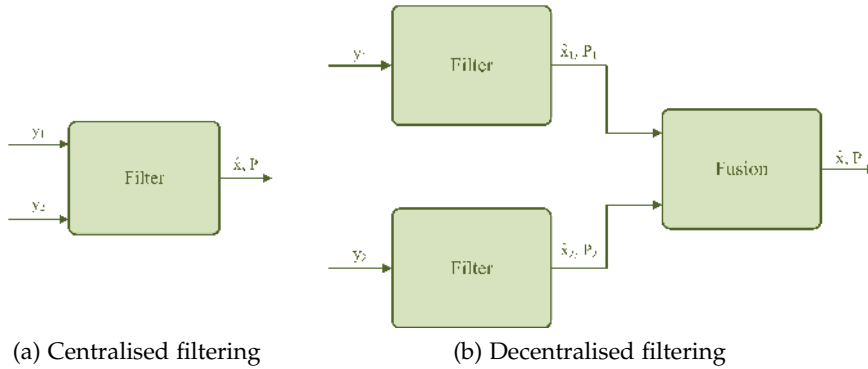


Figure 2.1: Technique of centralised and decentralised filtering. (reproduced from [21, p. 172])

Section 2.1.1 describes the concept of the KF. Moreover, Section 2.1.2 gives details about the EKF, which allows filtering of systems with a non-linear behaviour.

2.1.1 Kalman Filter

The KF was first introduced in the year 1960 by Rudolph E. Kalman [22]. This recursive filter offers an efficient way to estimate past, present and future states by minimizing the mean of the squared error. The small computational requirements and the use of recursion are the reason for the great success of the KF [23, 24].

A linear discrete-time controlled process with state $x \in \mathfrak{R}^n$ can be expressed by the linear stochastic difference equation

$$x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1}, \quad (2.1)$$

where

- A is a $n \times n$ matrix describing the relation between the previous time step $k - 1$ and the current time step k ,
- B is a $n \times l$ matrix relating to the control input $u \in \mathfrak{R}^l$,
- u is the optional control input, and
- w represents the process noise [13, 23].

The measurement $z \in \mathbb{R}^m$ is expressed as

$$z_k = Hx_k + v_k, \quad (2.2)$$

where

H is a $m \times m$ matrix relating from the state x to the measurement z and often called measurement model, and

v_k is the measurement noise [13, 23].

The two variables w_k and v_k standing for the process and measurement noise are independent from each other and have a normal probability distribution

$$\begin{aligned} p(w) &\sim N(0, Q) \text{ and} \\ p(v) &\sim N(0, R), \end{aligned} \quad (2.3)$$

where

Q is the process noise covariance matrix, and

R is the measurement noise covariance matrix [23].

The KF is split into two parts, the time update and the measurement update. The time update is used to calculate the *a priori* estimates of the state and the error covariance. In order to give feedback to the filter, the measurement update equations are used to achieve a better *a posteriori* estimate by combining new observations with the *a priori* estimate. A prediction of the state is performed during the time update and a correction is done with the use of the measurement update equations [23].

The time update equations can be taken from Equation 2.4. As previously stated, the time update predicts the *a priori* estimate based on the previous state information at time step $k - 1$. P_k^- is the *a priori* estimate error covariance and \hat{x}_k^- is the *a priori* estimate of state x at time k with having the knowledge before time step k [23].

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_{k-1} \quad (2.4a)$$

$$P_k^- = AP_{k-1}A^T + Q \quad (2.4b)$$

The measurement update equations are depicted in Equation 2.5. The first step in the measurement update is the calculation of the Kalman gain K , or blending factor. The Kalman gain is an $n \times m$ matrix that minimises the *a posteriori* error covariance. \hat{x}_k and P_k are the *a poste-*

riori estimates calculated using new observations - the measurement z_k . The matrix H is called the measurement model [23].

$$K_k = P_k^- H^T (H P_k^- H^T + R)^{-1} \quad (2.5a)$$

$$\hat{x}_k = \hat{x}_k^- + K_k (z_k - H \hat{x}_k^-) \quad (2.5b)$$

$$P_k = (I - K_k H) P_k^- \quad (2.5c)$$

Figure 2.2 illustrates the operation cycle of the KF. The time update is performed using the previous *a posteriori* estimates at time step $k - 1$. The *a posteriori* estimates are calculated in the measurement update and act as a feedback for the time update in the next step.

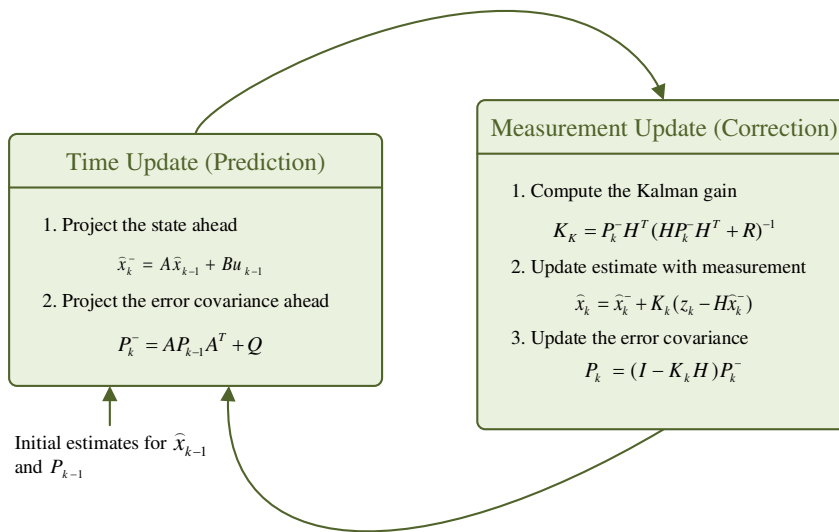


Figure 2.2: Operation cycle of a Kalman Filter (reproduced from [23, p. 6]).

2.1.2 Extended Kalman Filter

The **EKF**, which is described in Section 2.1.1, solves the problem of having a non-linear measurement relationship. This non-linearity is caused by the relation between the position and the velocity, and acceleration over time. A linear system can be estimated with the use of the **KF**. The solution provided by the **EKF** is linearising the estimation. The partial derivatives of the process and measurement functions are the key to estimate also non-linear connections [23].

The state x_k can be described as a function, the state-transition function, depending on the previous state x_{k-1} , the previous control input u_{k-1} , and the previous process noise w_{k-1} [23]. Equation 2.6 illustrates this change. The same applies for the measurement z_k shown

in Equation 2.2, what is described as a function h depending on x_k and v_k .

$$x_k = f(x_{k-1}, u_{k-1}, w_{k-1}) \quad (2.6)$$

For the reason that the noise w_k and v_k are in practice unknown, they can be set to 0. The EKF can be split as the KF into two groups. Equation 2.7 shows the equations representing the time update [23].

$$\hat{x}_k^- = f(\hat{x}_{k-1}, u_{k-1}, 0) \quad (2.7a)$$

$$P_k^- = A_k P_{k-1} A_k^T + W_k Q_{k-1} W_k^T \quad (2.7b)$$

A_k and W_k are the process Jacobians at time k and Q_{k-1} is the process noise covariance at the time $k - 1$. A Jacobian is a matrix containing the partial derivatives of a function, in this case the function f .

A is the process Jacobian matrix with respect to x

$$A_{[i,j]} = \frac{\partial f_{[i]}}{\partial x_{[j]}}(\hat{x}_{k-1}, u_{k-1}, 0), \text{ and} \quad (2.8)$$

W is the process Jacobian matrix with respect to w

$$W_{[i,j]} = \frac{\partial f_{[i]}}{\partial w_{[j]}}(\hat{x}_{k-1}, u_{k-1}, 0). \quad (2.9)$$

For the reason that all Jacobian matrices are time varying, they are written with the subscript k . Moreover, Equation 2.10 contains the measurement update equations of the EKF. To highlight that the EKF is a Bayes filter, one can write the estimated *a priori* error covariance P_k^- as $P_{k|k-1}$ and the *a posteriori* error covariance P_k as $P_{k|k}$ [23].

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + V_k R_k V_k^T)^{-1} \quad (2.10a)$$

$$\hat{x}_k = \hat{x}_k^- + K_k [z_k - h(\hat{x}_k^-, 0)] \quad (2.10b)$$

$$P_k = (I - K_k H_k) P_k^- \quad (2.10c)$$

H is the measurement Jacobian matrix with respect to x

$$H_{[i,j]} = \frac{\partial h_{[i]}}{\partial x_{[j]}}(\hat{x}_k^-, 0), \text{ and} \quad (2.11)$$

V is the measurement Jacobian matrix with respect to v

$$V_{[i,j]} = \frac{\partial h_{[i]}}{\partial v_{[j]}}(\hat{x}_k^-, 0). \quad (2.12)$$

Figure 2.3 illustrates the operation cycle of the EKF. The equations in this figure have been simplified for the reason of readability. The Jacobian matrices A, W, H and V are still time varying. Furthermore, the control input u_{k-1} has been ignored.

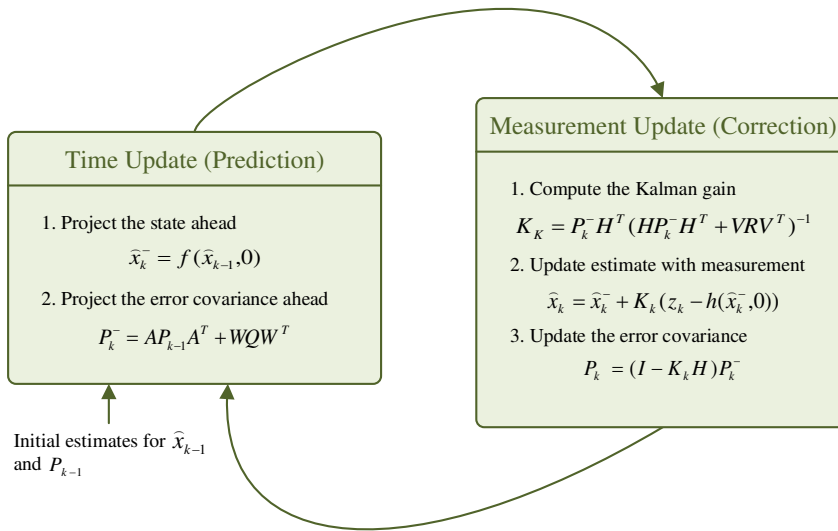


Figure 2.3: Operation cycle of an Extended Kalman Filter.
(reproduced from [23, p. 11])

2.1.3 Particle Filter

The PF is another solution for estimating the state of a non-linear system. A popular application for the PF is the position estimation of underwater vessels, ships, cars, and aircraft. The PF as well as the EKF belong to the field of non-linear/non-Gaussian Bayesian tracking. [21, 25].

The three steps of the PF algorithm are

1. Measurement update,
2. Resampling, and
3. Time update [21].

Choosing the distribution and the number of particles is the initial step when using a PF. The next step is to generate N particles following the chosen distribution with equal weights. In the measurement update, the weights are adjusted in relation to the variation of the predicted and observed measurements. Resampling is about generating N new samples depending on the new weights of the particles. The time update is comparable to the prediction in the KF, the new position of the particles is estimated with the use of a dynamic model that describes the action. A continuous execution of these three steps leads to closer particles and thus to a better estimate of the current state [21, 26].

Gustafsson describes in [26] several ways to tune the PF. One way is to adapt the resampling process. Resampling is necessary in order to avoid having only a few particles with a significant weight. On the other hand, resampling at every iteration increases the uncertainty. A possible solution for this problem is to perform the resampling only when it is necessary, for instance when the number of significant samples reaches a certain limit [26].

2.1.4 *Rèsumè*

A KF is the most common approach to filter a system that has a linear Gaussian distribution. When it comes to estimating a system state that does not follow a linear Gaussian distribution, more filters can be considered. The EKF and PF are the most commonly used filters for this kind of application. Both algorithms have their benefits and disadvantages that will be discussed in this section.

Systems with non-Gaussian densities, such as the bimodal distribution, may not be suitable for the EKF for the reason that the EKF linearises around the current mean and covariance. One major benefit of the EKF is the computational requirement. At each time step the EKF has to evaluate the function $f(x)$ and $h(x)$. In comparison, the PF needs to evaluate every particle, which results in N evaluations. Gustafsson describes that the PF is by a factor of $\frac{N}{n_x}$ more demanding than the EKF concerning the computation. The former described factor contains N , the number of particles, and n_x the number of size of the state [25, 26].

The computational savings of the EKF compared to the PF are important when it comes to real time measurements. In a VANET, messages are exchanged in a frequency of 10Hz or even higher such as the iCLCM message set, which is sent out every 40ms respectively with a

frequency of 25Hz (see [19]). Having only 10 other vehicles around leads to $(10 + 1) \cdot 25$ messages or position estimations per second.

2.2 TRUST IN VEHICULAR AD-HOC NETWORKS

Improving the situation awareness of automated vehicles is a highly interesting research area. With the increasing number of electronic components, called Electronic Control Units (ECUs), it becomes necessary to broaden the research in the area of situation awareness in order to develop safer and more reliable vehicles regarding self-driving functionalities. The vehicles are taking over more and more tasks from the driver, e.g. anti-lock braking and lane keeping assistance, in order to improve safety or to increase the comfortability for the passengers.

Miller et al. distinguish in [27] between six different levels of automation. In L0, the driver has full control over the vehicle and is thus able to control the brake, steering, throttle, and motive power. Vehicles with features such as anti-lock braking are classified as L1. Is the vehicle able to control two of the aforementioned functions, such as steering and braking, it falls into the category L2. It is necessary to point out, that a L2 vehicle only performs the automated action in certain situations and always with a driver that oversees the vehicle. L3 vehicles have more responsibilities for perceiving the environment and are in charge of alerting the driver when a change to manual control is necessary. Is a vehicle driving autonomously to a certain destination without any further input of the driver, it is a L4 vehicle. The last level, L5, is a fully autonomous vehicle that can drive without a driver [27].

This section briefly lists the security challenges and the current research about trust and awareness in an V2X environment. V2X communication expresses the ability of vehicles to communicate with their environment, by creating a VANET. The vehicle is not only able to communicate with other vehicles, it is also able to receive messages from Road Side Units (RSUs) about emergencies, road conditions and other safety related information.

2.2.1 Security Challenges

THREATS. As pointed out by Bijlsma et al. in [5], the safety and security threats in VANETs and inter-vehicular systems have become very important due to revealed security gaps in these systems. The motivation for attackers is identical to attacks on smartphones and personal computers. The two noticeable incentives for attacking a system are profit, and the urge for destruction and amusement. Espionage is not visible and thus it is not possible to give detailed information about the dissemination within IT [5].

SEPARATION. Software within a vehicle is used to enhance functionalities in two domains: technical improvements such as **CACC**, and infotainment services. Vehicle manufacturers have to separate these domains in order to ensure a safe system at any point of time, even if the system has been compromised [5].

RAPID CHANGE. Vehicles are moving with different speed and may thus be visible only for a few seconds. This highly dynamic environment causes topology changes in a high frequency. A **RSU** is stationary and has for that reason a velocity of 0 km/h. A car may drive with a speed of 200 km/h. These two extrema highlight the fast change of a **VANET** topology. Additionally, it is necessary to have a scalable vehicle communication system because the vehicles will receive a high amount of data in congested areas [14, 28].

DATA INTEGRITY. Data may be securely transmitted, but this does not guarantee the integrity of the received information. Applications, such as **CACC**, rely on the integrity of the data. A comparison of the received information with local sensors can help to identify incorrect data [5, 29].

2.2.2 Trust Establishment

Creating a trust relationship between vehicles is a prerequisite for **V2V** communication. Given the base of having the other vehicle identified or verified, allows a higher level verification of the vehicle's data and its behaviour. Patel et al. evaluate different approaches based on trust for secure routing in a **VANET** in [30]. The different concepts for establishing trust known from fixed networks are provided by Wex et al. in [11]. Wex et al. assess these approaches for their usability in **VANETs**. The two basic methods on how to set up trust are infrastructure-based and self-organising trust establishment. An illustration of the different techniques can be found in **Figure 2.4** [11].

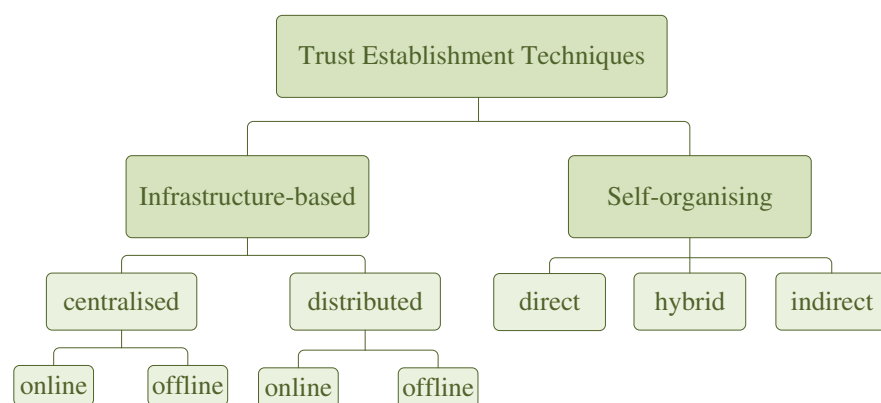


Figure 2.4: Categorisation of trust establishment techniques.
(reproduced from [11, p. 2])

INFRASTRUCTURE-BASED TRUST. This techniques mostly make use of certificates for authentication. The X.509 standard, see [31], is used for authentication of companies or private persons within the internet. Approaches exist that try to apply this standard to VANETs. Each entity is identified by a unique name that is certified by a trusted authority. An approach that does not require a globally unique name is *Simple Distributed Security Infrastructure*. Kerberos is another method to perform infrastructure based authentication. The major drawback of Kerberos is the required online certification procedure, which makes this method not suitable for VANETs. Other techniques for infrastructure-based trust, e.g. use of pseudonyms, digital credentials, and group signatures, are discussed in [11] and [32].

SELF-ORGANISING TRUST. This technique does not need a trusted third party and is further more independent from the need of a global knowledge. Trust between the vehicles or nodes is therefore dynamic and dependent on the timespan in which a vehicle is connected. This entails that a vehicle, which is longer connected, may have a higher trust than a vehicle that is only known for a short period of time. A *direct* self-organising trust establishment considers just the communication and interaction with the other nodes, while vehicles using the *indirect* method exchange their trust information about other nodes with each other (second hand information). A combination of the former mentioned methods is the *hybrid* technique. [11] provides a description and evaluation of self-organising methods, such as CONFIDENT, Terminodes, SPRITE, and LLSC [11, 32].

PRIVACY-PRESERVING TRUST ESTABLISHMENT. There exist several approaches to provide a privacy-preserving authentication in VANETs. The Situation-Aware Trust Architecture (SAT) is proposed in [33]. The cryptographic-based approach uses attributes for encryption. This method has the advantage that it also allows to create group policies that describe a group with certain attributes. Tajeddine et al. suggest in [34] a group-based model, that allows other nodes to evaluate trust of the entire group [33, 34].

2.2.3 Trust and Reputation Models

An authentication of the other participants like other vehicles or RSUs is the basis for V2V communication. The next step is the identification of faulty data caused by sensor errors or nodes sending malicious information. Due to safety requirements in vehicles, it is inevitable to recognise the previously mentioned cases and react properly. Existing methods from other areas, such as trust management in a Multi-Agent System (MAS), may be suitable for VANETs, as well. So far the focus was set on the integrity and confidentiality of messages within vehicular networks, the evaluation about the quality of the information has played a minor role. Section 2.2.1 lists the challenges in a VANET.

Because of the high velocity of the vehicles, the topology changes quickly and the time to react is short. Thus, it is important to be able to verify the received data in real-time [14].

Trust is a common measure in decentralised systems, such as a MAS. Aras et al. discuss in [35] the relation between trust and uncertainty. The identified sources for uncertainty in a MAS are among others the uncertainty in the observation, and the uncertainty when using second-hand information. The authors propose that the trust representation has to reflect the uncertainty [35].

MULTI-AGENT SYSTEMS. Agents are designed to cooperate with each other to achieve a common goal. This also applies to cooperative driving. The intersection scenario described in Chapter 1 requires a collaboration between all involved vehicles. As there are selfish drivers on the roads, there might be also selfish agents. For that reason it is necessary to create a trust metric that considers the interaction over a certain period of time. The more the agents performed a task with each other the more accurate is the trust model. A higher value of trust will result in a higher chance for a prospective interaction with this agent [14, 36].

TRAVOS (Trust and reputation in the context of inaccurate information sources) is shown in [36]. This proposed model counts the number of successful and unsuccessful interactions with a certain node (trustee). Based on this information the truster, the node which evaluates the trustee, generates a beta distributed probability density function with $\alpha = \#$ of successful outcomes +1 and $\beta = \#$ of unsuccessful outcomes +1. A detailed description of this method to manage trust can be taken from [36].

Zhang refers in [14] to several trust management methods suitable for MAS, but the problem of limited knowledge about the environment still exists. Many of the models shown by Zhang assume complete knowledge about the other agents. VANETs are highly dynamic and it might be the case that a vehicle never “meets” the other vehicle again, or knows the vehicle only for a few seconds [14].

Zhang differentiates between three categories of trust management frameworks in VANETs: entity-oriented, data-oriented, and combined trust models [14].

ENTITY-ORIENTED TRUST MODEL. These frameworks focus on the trustworthiness of the vehicles. Minhas’ et al. approach proposed in [37] distinguishes between role-based, experience-based, priority-based trust, and majority-opinion. The key properties of VANETs are shown in Figure 2.5. In VANETs one has to cope with the sparsity of data, which is caused by the dynamic network topology, and the fact that communication can be event-triggered. It is important to highlight, that the reports mentioned by Minhas et al. contain informa-

tion about the road conditions or events, e. g. slippery road or traffic jam [14, 37].

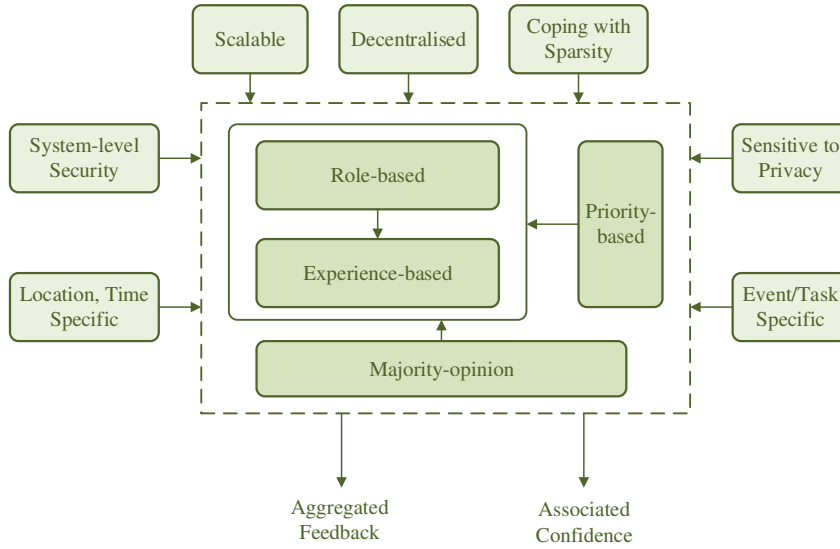


Figure 2.5: Entity-based trust management. (reproduced from [37, p. 3])

Role-based trust assigns a certain level of trust according to the identity of the node. A police car or an emergency vehicle have a higher trust level compared to an ordinary car. As in every approach for trust management, the trust in a certain vehicle is dependent on its experience with it. Minhas et al. combine these two trust models and create a priority-based matrix based on these results. The majority-opinion model considers the reports of all nodes and generates a value by considering also the experience and role of the vehicles [37].

Experienced-based trust values are in an interval of $(-1, 1)$. 1 entails fully trust in the vehicle, while -1 implicates absolute distrust. Minhas et al. propose the following scheme for updating a nodes experience-based trust value. $T_A(B) \in (-1, 1)$ describes the trust of vehicle A in node B within the given interval. Equation 2.13 shows the update procedure if node B’s information has been proven as being reliable [37].

$$T \leftarrow \begin{cases} T + \alpha \cdot (1 - T) & \text{if } T \geq 0 \\ T + \alpha \cdot (1 + T) & \text{otherwise} \end{cases}, \quad (2.13)$$

where

- T Experience-based trust value ($T_A(B)$)
- $0 < \alpha < 1$ Positive increment factor

In case that node B's information has been evaluated as wrong, Equation 2.14 is used to update the trust value of B.

$$T \leftarrow \begin{cases} T + \beta \cdot (1 - T) & \text{if } T \geq 0 \\ T + \beta \cdot (1 + T) & \text{otherwise} \end{cases}, \quad (2.14)$$

where

T Experience-based trust value ($T_A(B)$)
 $-1 < \beta < 0$ Negative decrement factor

Minhas et al. further propose a forgetting factor $\lambda(0 < \lambda < 1)$ that assigns a lower weight to the previous trust value respectively the past interactions. The time difference between the previous and the current interaction is expressed in t . Equation 2.15 shows the updated formula for calculating the experience-based trust value when the information is evaluated as reliable. The updated formula when the B's information is considered as wrong, is shown in Equation 2.16 [37].

$$T \leftarrow \begin{cases} \lambda^t \cdot (1 - \alpha) \cdot T + \alpha & \text{if } T \geq 0 \\ \lambda^{-t} \cdot (1 + \alpha) \cdot T + \alpha & \text{otherwise} \end{cases} \quad (2.15)$$

$$T \leftarrow \begin{cases} \lambda^t \cdot (1 - \beta) \cdot T + \beta & \text{if } T \geq 0 \\ \lambda^{-t} \cdot (1 + \beta) \cdot T + \beta & \text{otherwise} \end{cases} \quad (2.16)$$

The framework of Minhas et al. provides an entity-based trust model that considers the role of the vehicle and the past experience with the vehicles. Moreover, the authors have proposed a priority-based trust value that takes the afore mentioned values into account. The method based on experience of Minhas et al. has been described in more detail due to the relevance for this work. A complete explanation of the author's suggested framework can be taken from [37].

DATA-ORIENTED TRUST MODEL. The analysis of the received data is in the focus of this kind of models. The data-centric approach of Raya et al. in [38] bases its trust on reports received from multiple devices. These reports contain information about events such as ice on the road or traffic jam. Each vehicle is from a certain type, which is the basis for the calculation of the default trustworthiness. The event- or task-specific trust is depending on a so-called trustworthiness function f . Dynamic factors are considered with a security status function and dynamic trust metric functions for each attribute of the report. Additionally, the method considers the distance between the reporter's location and the event location. Raya et al. calculates the trust value based on these factors. The authors compare different evaluation techniques to this method, such as weighted voting, Bayesian interference, and Dempster-Shafer theory. The Bayesian interference

is used to calculate the probability of a certain task given the new evidence [14, 38].

COMBINED TRUST MODEL. The distributed reputation model suggested in [39] is a representative of this class. The node generates its own opinion about the message content by also considering the forwarding node's direct or indirect reputation and appends it to the message before forwarding it to the other participants. The authors call this method *Opinion Piggybacking* [14, 39].

2.3 SITUATION AWARENESS

The environment is shifting from systems fully controlled by human beings to partly or fully automated systems. This change from a static to a dynamic environment causes a new way of situation awareness. In the area of modern work settings the out-of-the-loop performance problem has been identified. When working with fully automated systems, the operators tend to react much slower in case of an automation failure due to the passive monitoring of the system. One solution is to use varying levels of automation in order to keep the operators in the decision-making loop. This phenomenon can also be applied to the automated driving, a driver that just monitors the system will most probably react slower. For that reason it is essential to increase the situation awareness of the vehicle [40].

Kurmis et al. describe in [41] a system that increases the situation awareness of a vehicle based on the ego vehicle's on-board sensors, other vehicle's safety warning or road information messages, and road and government information gathered from **V2I** communication. The evaluation of the received data and the own vehicle's data is hereby performed using a utility function. The resulting utility is used by the Decision Support System which determines if the message is forwarded to the other traffic participants or not [41].

Röckl's work in [42] shows a design approach of a situation aware vehicle system that uses inference. The author supports his choice with a comparison of alternative situation models, such as formal logics, neural networks and data mining. Röckl describes the use of a **PF** in [15] that fuses the information received from other vehicles. A simulation of the designed **CACC** that relies on the proposed system for decision-making and perception is the result of Röckl's thesis [42].

2.4 SUMMARY

This section provides the status of research concerning fusion of sensor data and trust in **VANETs**. The three most commonly used filters for **SF** have been described and compared according to their area of application.

A system designed for **V2V** communication needs to address certain challenges that are listed in **Section 2.2.1**. The fast changing topology

of the VANET is one of the major issues that need to be addressed in the system architecture. It will happen that a vehicle “meets” the other vehicle only once in its lifetime. Moreover, it is possible that the data exchange lasts for only a few seconds for the reason that the vehicles are driving in the opposite direction. The way how to establish trust in such a highly dynamic environment has been also discussed. This thesis assumes the existence of an implemented method providing secure communication with authenticated vehicles at system level.

Section 2.2.3 presents different models of how to create trust. It gives details of how trust can be designed in a MAS and the current approaches for VANETs. The methods proposed for V2V environments only consider the information about events received from the other vehicles, they do not take the sensor accuracy and behaviour of the traffic participants into account. The experience-based algorithm shown by [37] provides the calculation of a TI including a forgetting factor that gives new information a higher weight. Trust has been chosen as a measure for the current situation because trust is considering various factors including the uncertainty of the observed information.

Situation awareness of autonomous vehicles is very important for the safety of the passengers. V2V communication enables the vehicles to perceive out-of-sight objects and to receive warnings about dangerous situations earlier. Röckl proposes in [42] the use of inference to increase the situation awareness of the vehicle.

Further, Röckl presents the use of a PF to fuse the information of the other traffic participants. The developed PF is used for a CACC within a simulation environment. Röckl’s approach does not generate a TI that gives information about the behaviour and sensor accuracy of the vehicle as proposed in this thesis. The PF is only used to get a better estimation of the other vehicle’s position. However, Röckl considers the uncertainty given by the PF to increase the situation awareness [15, 42].

DATA ACQUISITION

To offer new functionalities and safety features, today's cars use sensors, such as radar, ultrasonic, and camera. These sensors allow the vehicle to detect lanes, speed regulation signs, and even assist while parking. Vehicles designed for autonomous driving, such as the Google car¹, use laser, radar and cameras to provide a 360° object identification.

The results of the TS proposed in this thesis are evaluated using a test vehicle that provides data from the radar, built-in sensors, and V2X communication for perceiving the environment. Figure 3.1 illustrates the communication with the other vehicles. The GeoNetworking network architecture defined by the European Telecommunications Standards Institute (ETSI) in [43] is used for communication. The transmission on the physical layer is established using IEEE 802.11p [44], a standard for wireless vehicular communication.

The radar in combination with the geographic position of the ego and the preceding vehicle is used to achieve a more accurate estimation of the distance to the car in front and also to evaluate the data accuracy of the preceding vehicle.

The built-in sensors of the car are the measurements, such as velocity, acceleration, angle of the steering wheel, and yaw rate.

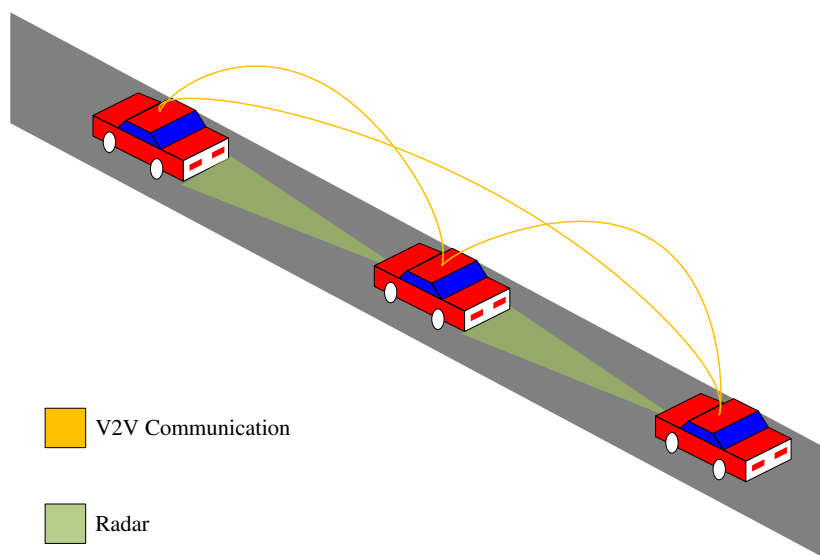


Figure 3.1: Sensor Environment with V2V communication.

¹ See: <https://www.google.com/selfdrivingcar/how/> (2016/06/12)

The on-board sensors of the vehicle build the basis for knowing the status of the ego vehicle. To be able to calculate a relative position to other vehicles, it is necessary to use a positioning via satellites such as GPS or Galileo. Many GPS devices provide additional information about the accuracy of the position, the dimensionless Dilution of Precisions (DOPs) values. Horizontal DOP (HDOP) and Vertical DOP (VDOP) characterise the accuracy of the horizontal or vertical position solution [45, 46].

Table 3.1 lists the the measurements including their resolution relevant for this thesis. For more detailed information about the exchanged information see the ETSI and the i-GAME specifications [17, 19]. HDOP and VDOP values are the only measurements in Table 3.1 that are not being exchanged with the other vehicles. Furthermore, the preceding vehicle will be named Most Important Object (MIO) according to [19]. Giving the other vehicles more information about oneself enables the design of more efficient manoeuvres, like a CACC that reacts smoother to speed changes using the desired acceleration of the MIO.

TYPE	MEASUREMENT	UNIT/RESOLUTION
on-board	speed over ground	0.01 m/s
	longitudinal acceleration	0.1 m/s ²
	desired long. acceleration	0.01 m/s ²
	lateral acceleration	0.1 m/s ²
	yaw rate	0.01 deg/s
Geographical position	latitude	0.1 μdeg
	longitude	0.1 μdeg
	heading	0.1 deg
	HDOP	—
	VDOP	—
with respect to preceding vehicle	bearing	0.002 rad
	range	0.01 m
	range rate	0.01 m/s
	time headway	0.01 s

Table 3.1: Relevant measurements in a VANET environment.

Section 3.1 describes how the KF is used. The SF of the distance to the preceding vehicle can be obtained by two sensors, the vehicle's front radar and the distance between two geographic positions. Due to the fact that this relation is linear Gaussian distributed, a KF can be used for fusing this data.

The relation between speed, acceleration, yaw rate, heading, and

position is non-linear. For that reason it is necessary to create a model that represents the relation between these measurements. The designed model used for SF is based on a EKF. A description of the model and the specific EKF for this thesis can be taken from Section 3.2.

3.1 LINEAR MODEL

Roads are curvy and therefore the radar might not detect the MIO while driving in a curve. It may also be the case that the geographical position is not precise enough and thus a second sensor to verify the distance is necessary. Moreover, a fusion of both distances can signify the accuracy of the other participant's position. These two applications are useful in order to increase traffic safety and to detect measurement errors or inaccuracies caused by the ego or the other vehicle. The evaluation of the position quality of the MIO can be additionally considered by the TS described in Section 4.1.

Moreover, this model includes security mechanisms to provide correct or at least safe information to the vehicle's system. Figure 3.2 illustrates the model for fusing the distance to the vehicle in front given two different sources. The geographical position can be the raw data provided by the GPS device or already a fused dataset. The *Distance Calculation* block calculates the ellipsoidal distance using the geographical positions of both vehicles and their length. The position of the vehicle is throughout this thesis defined as the geometrical centre of the vehicle.

For the reason that the data of both sensors might not be available at the same time due to environment changes, such as driving in a tunnel or on a curvy road, it is important to choose a proper variance for the data and select the data in respect to these circumstances. This procedure is done by the *Data Selection* shown in Figure 3.2.

The *Kalman Filter* module performs the SF considering the adapted standard deviation provided by the *Data Selection*. A more detailed description about the KF is shown in Section 2.1.1.

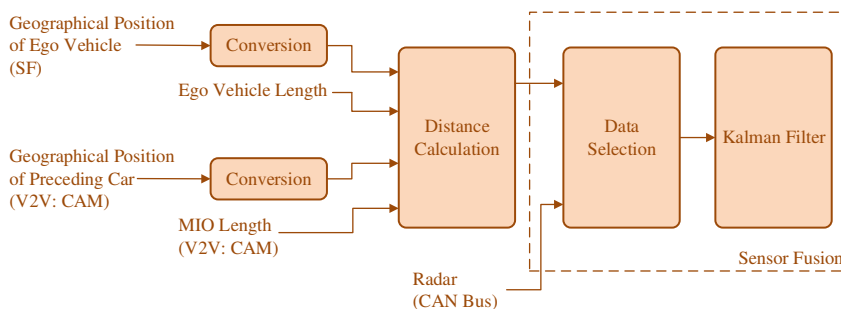


Figure 3.2: Linear SF model.

The *Data Selection* module will provide the **KF** with the necessary sensor data including their variances σ^2 . Simplifying [Equation 2.5b](#) the following allows a state estimation given two sensors and their standard deviation. The first step to obtain such a filter is to assume that the transformation matrix H is 1, which results in [Equation 3.1](#).

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - \hat{x}_k^-) \quad (3.1)$$

In case of having two sensors that provide their measurement in the same unit, the predicted value \hat{x}_k^- can be replaced with the second measurement. The outcome of this one-dimensional filter is a static **SF** of both sensors considering their variance [24].

$$x_{\text{fused}} = z_1 + K \cdot (z_2 - z_1), \quad (3.2)$$

where

z_1 measurement of sensor 1

z_2 measurement of sensor 2

K Kalman gain

The Kalman gain K is calculated using the variances σ^2 described in [Equation 3.3](#).

$$K = \frac{\sigma_1^2}{\sigma_1^2 + \sigma_2^2} \quad (3.3)$$

The variance of the fused measurement can be calculated according to [Equation 3.4](#).

$$\sigma^2 = (1 - K) \cdot \sigma_1^2 \quad (3.4)$$

When fusing the sensor measurements, it is important to highlight that the position of z_1 and z_2 in [Equation 3.2](#) influences the result. In case that the measurement of sensor 1 is more accurate ($\sigma_1 \ll \sigma_2$), it has to replace the estimate \hat{x}_k^- in [Equation 3.1](#).

Using the proposed filter allows to adapt the variance σ_i^2 of the sensors according to the current needs. Further, the *Data Selection* module decides whether to fuse the sensor data or to rely on only one sensor in case that one measurement is not available or the measurement is not accurate enough. Thus, use cases like driving in a tunnel or losing the radar target on a curve are safely considered. Assuming that the distance calculated using the geographical position is above or below a certain threshold compared to the radar information, the system will consider only the radar distance d_r . This proceeding ensures that the vehicle does not crash with the car in front when receiving a wrong position from the **MIO**.

A comparison of both distances over time can be used to classify the control behaviour of the preceding vehicle. It provides not only in-

formation about the position accuracy, it also shows the acceleration profile of the vehicle.

The activity diagram for the linear model is illustrated in Figure 3.3. It is going to be executed every time a geographical position update via CAM is received. Additionally, a map that is part of the perception of the TS described in Section 4.2 identifies the vehicle in front.

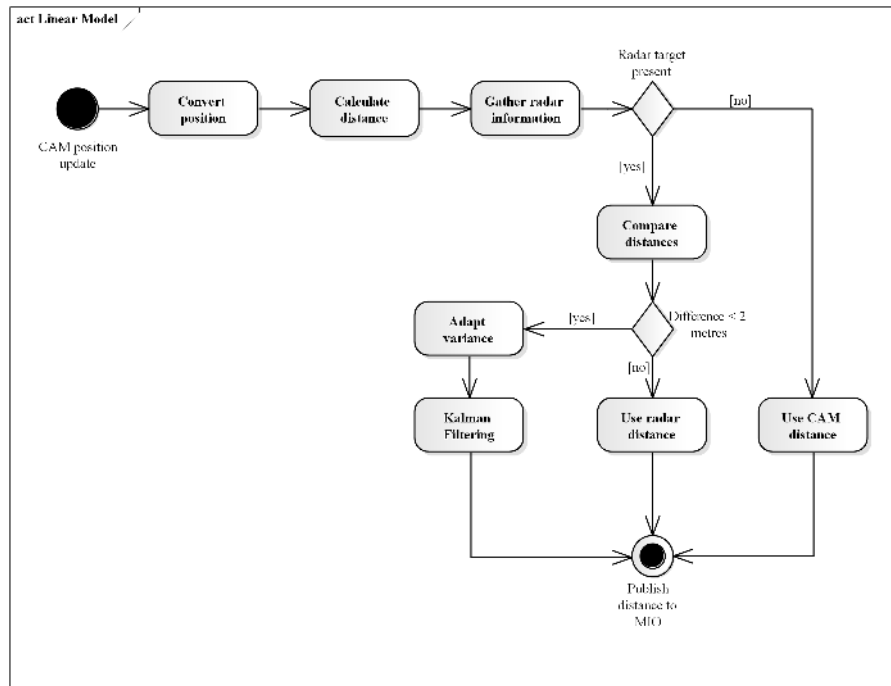


Figure 3.3: Activity Diagram for the linear model.

Figure 3.4 shows a simulated result of the proposed KF. The simulation has been performed with PreScan². The system has a highly accurate position and therefore a low variance. To force sensor errors, the ego vehicle was equipped with only one radar sensor, which causes a narrow view of the preceding vehicle and thus it is not able to detect a target on curves. Figure 3.4 illustrates that the proposed filter is able to compensate wrong sensor information, such as a loss of radar information. The former mentioned *Data Selection* module has not been used for this simulation. An evaluation of more complex experiments can be found in Chapter 6.

PreScan² has been provided by TASS International.

3.2 NON-LINEAR MODEL

Table 3.1 shows that the received information about the ego and the other vehicles have a certain relation with each other. For instance is the position of a vehicle dependent on its velocity, heading and yaw rate. Combining these sensor measurements in order to achieve a more accurate estimation of the position, a non-linear model based on the kinematic relations is needed. Section 2.1.2 describes the theo-

² See: <https://www.tassinternational.com/>

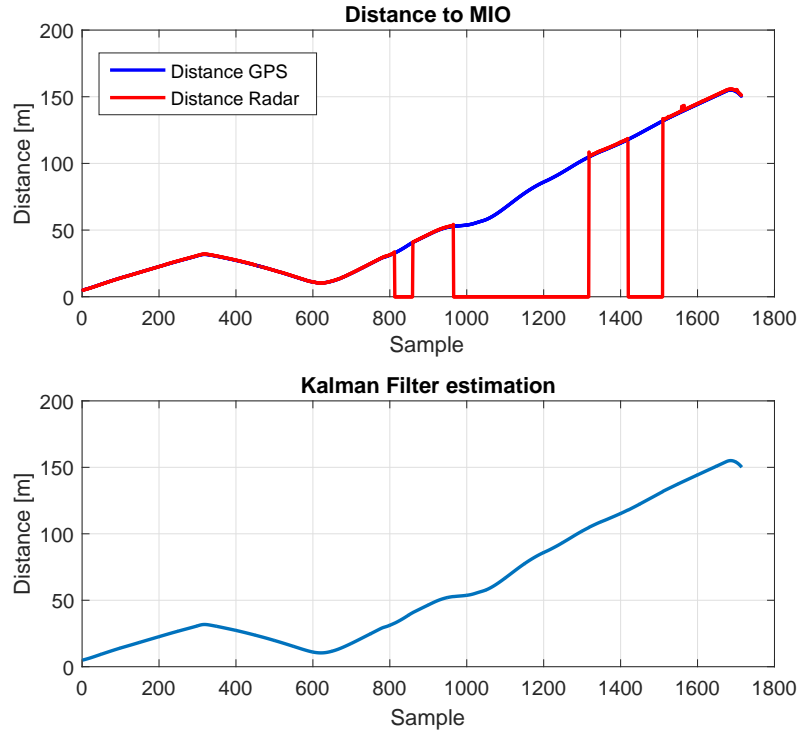


Figure 3.4: Distance fusion with a KF.

retical background for the [EKF](#) implemented for this thesis. Moreover, [\[13, 47\]](#) provide the [EKF](#) for vehicles using the position, acceleration, heading, yaw rate, and velocity of the vehicle.

The state vector x in [Equation 3.5](#) is taken from [\[47\]](#) and shows the input for the [EKF](#) at time k .

$$x_k = \begin{bmatrix} E_k[\text{m}] \\ N_k[\text{m}] \\ \varphi_k[\text{rad}] \\ v_k[\text{m/s}] \\ \dot{\varphi}_k[\text{rad/s}] \\ a_k[\text{m/s}^2] \end{bmatrix}, \quad (3.5)$$

where E_k and N_k are the east and north coordinates within a relative coordinate system in metre, φ_k is the heading to the north, v_k is the velocity, $\dot{\varphi}_k$ is the yaw rate, and a_k is the acceleration of the vehicle.

The [GPS](#) device provides the position in Longitude, Latitude, and Altitude ([LLA](#)) form. To use the position with the process model shown in [Equations 3.6, 3.7, 3.8, 3.9, 3.10, and 3.11](#), it is necessary to transform the [LLA](#) coordinates into a local coordinate system, such as East, North, and Up ([ENU](#)). [\[13\]](#) shows the transformation steps that are necessary for converting the coordinates from [LLA](#) to [ENU](#) and back-

wards. MATLAB's function *geodeticzenu* has been used for this conversion throughout this thesis.

The heading needs to be preprocessed as well. The GPS device provides the heading in compass format and thus, it has to be transformed to unit circle format in order to work with sine and cosine functions, and with the process model. Inaccuracies when driving with low speed have been directly corrected in the Data Acquisition module. The third manipulation is the correction of jumps from 0 deg to 360 deg. The other measurements, such as the velocity, acceleration, and the yaw rate, are provided by the internal sensors of the vehicle via the Controller Area Network (CAN) bus [13].

The motion model is the link from the previous state x_{k-1} to the current state x_k . The kinematic model taken from [13, 47] describes this relation between the previous and the next state. Equations 3.6, 3.7, 3.8, 3.9, 3.10, and 3.11 express this non-linear kinematic relation, the state-transition function f . The variable T in these equations is the sampling time, the time difference between time $k-1$ and k .

$$\begin{aligned} E_k = & E_{k-1} + \cos(\varphi_{k-1}) \left(v_{k-1} T + a_{k-1} \frac{T^2}{2} + \dot{a}_{k-1} \frac{T^3}{3} \right) - \dots \\ & \sin(\varphi_{k-1}) \left(v_{k-1} \dot{\varphi}_{k-1} \frac{T^2}{2} + (v_{k-1} \ddot{\varphi}_{k-1} + 2a_{k-1} \dot{\varphi}_{k-1}) \frac{T^3}{6} + \dots \right. \\ & \left. (\dot{a}_{k-1} \dot{\varphi}_{k-1} + a_{k-1} \ddot{\varphi}_{k-1}) \frac{T^4}{8} + \dot{a}_{k-1} \ddot{\varphi}_{k-1} \frac{T^5}{20} \right) \end{aligned} \quad (3.6)$$

$$\begin{aligned} N_k = & N_{k-1} + \sin(\varphi_{k-1}) \left(v_{k-1} T + a_{k-1} \frac{T^2}{2} + \dot{a}_{k-1} \frac{T^3}{3} \right) + \dots \\ & \cos(\varphi_{k-1}) \left(v_{k-1} \dot{\varphi}_{k-1} \frac{T^2}{2} + (v_{k-1} \ddot{\varphi}_{k-1} + 2a_{k-1} \dot{\varphi}_{k-1}) \frac{T^3}{6} + \dots \right. \\ & \left. (\dot{a}_{k-1} \dot{\varphi}_{k-1} + a_{k-1} \ddot{\varphi}_{k-1}) \frac{T^4}{8} + \dot{a}_{k-1} \ddot{\varphi}_{k-1} \frac{T^5}{20} \right) \end{aligned} \quad (3.7)$$

$$\varphi_k = \varphi_{k-1} + \text{sign}(v_{k-1}) \left(\dot{\varphi}_{k-1} T + \ddot{\varphi}_{k-1} \frac{T^2}{2} \right) \quad (3.8)$$

$$v_k = v_{k-1} + a_{k-1} T + \dot{a}_{k-1} \frac{T^2}{2} \quad (3.9)$$

$$\dot{\varphi} = \dot{\varphi}_{k-1} + \ddot{\varphi}_{k-1} T \quad (3.10)$$

$$a_k = a_{k-1} + \dot{a}_{k-1} T \quad (3.11)$$

The process model A is the Jacobian of the state-transition function f , it can be taken from the [Appendix A](#) in [Equation A.1](#). The measurement model H describes the relation between the state and the measurement. H can be an identity matrix, which means that the mapping of the state to the measurement z_k is linear. In case that an element of the state is a combination of two measurements, the measurement model is no longer an identity matrix. The model used for this thesis has a linear relation between the state and the measurements. For that reason, H is an identity matrix and can be seen in [Equation A.3](#).

The filter parameters, such as the measurement noise covariance matrix R , and the process noise covariance matrix Q , define the deviation of the measured data and the states. A heuristic analysis showed that Q , shown in [Equation A.4](#), is similar to the measurement noise covariance R . Due to the fact that the implemented [EKF](#) has to adapt to the environment changes, such as low accuracy of the [GPS](#) position or loss of the satellite connection, two measurement noise covariance matrices are used depending on the situation.

[Equation 3.12](#) and [Equation 3.13](#) show the measurement noise covariance matrix for the former described two cases. It can be easily seen that only the variance of the first three measurements, the east and north position, and the heading, change due to the fact that the variance of the vehicle's on-board sensors are independent of the geographical position of the vehicle. [Equation 3.12](#) presents a low variance of the geographical position, which indicates that this matrix is used when the [GPS](#) position is sufficiently accurate. In case the precision of the position and the heading decreases, [Equation 3.13](#) is being used. The chosen variances have been determined by considering the precision information of the manufacturer and heuristic tests.

$$R_1 = \begin{bmatrix} 0.1^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.1^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & (2\frac{\pi}{180})^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 10^{-4} & 0 & 0 \\ 0 & 0 & 0 & 0 & (0.1\frac{\pi}{180})^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.12)$$

$$R_2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & (10 \frac{\pi}{180})^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 10^{-4} & 0 & 0 \\ 0 & 0 & 0 & 0 & (0.1 \frac{\pi}{180})^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.13)$$

Figure 3.5 illustrates the processing chain, when receiving new position information. The **EKF** is applied to both, the data of the ego vehicle and the other vehicles. The time difference between two time steps can be found by considering the timestamp of the **CAM** or by using the internal timestamp of the measurement data. Furthermore, the Kalman gain K as well as the error covariance matrix P can be used for consideration of the accuracy of the data. An increasing Kalman gain indicates that the filter “trusts” the measurements to a wider extent than before [23].

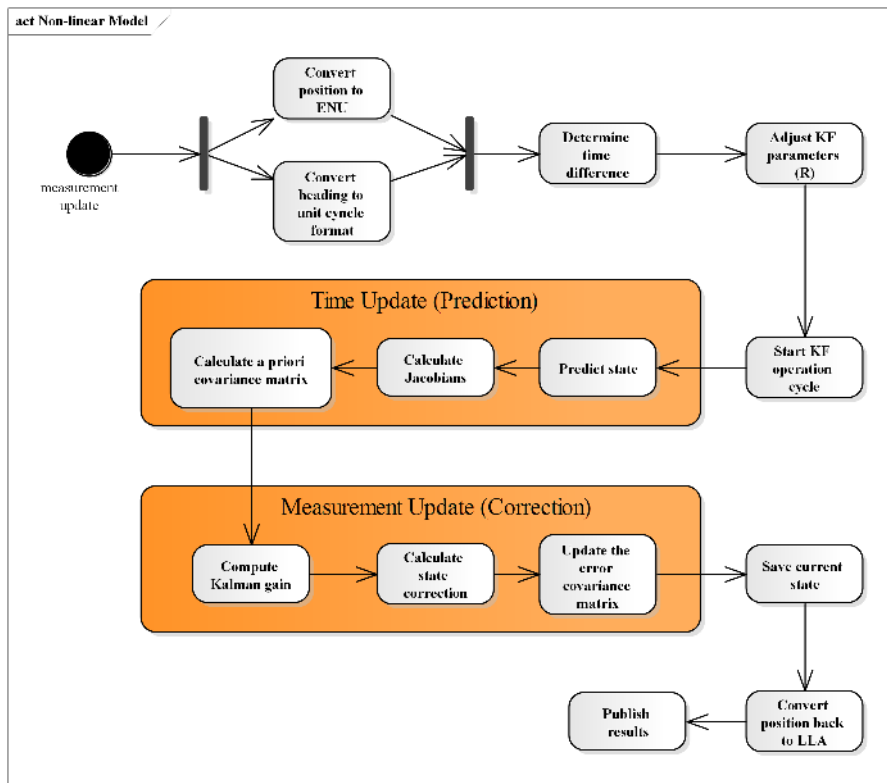


Figure 3.5: Activity Diagram for the non-linear model.

3.3 SUMMARY

The data relevant for this thesis is listed in Table 3.1. The introduction of a virtual sensor, the sensor information of the other vehicles, can

be used to improve road safety and efficiency with the use of cooperative platooning. Knowing the geographical position of the other vehicle without perceiving it with the own vehicle's sensors helps to increase situation awareness. This chapter described modelling the distance to the preceding vehicle with the use of the radar and the V2V data in [Section 3.1](#). The SF of the V2V data in order to have a better position estimation of the other vehicles has been described in [Section 3.2](#) by using a non-linear model. The data of the ego vehicle can also be applied to this model. Moreover, this non-linear model improves the efficiency of platooning and other cooperative interaction applications for the reason that this data is more reliable than the raw information received by the ego vehicle.

The non-linear model can be further used to observe the accuracy of the received sensor information by taking the Kalman gain into account. Knowing and observing the other vehicles, especially the preceding vehicle, is essential for increasing the awareness about the quality of the provided data. Both models rely on a KF which is briefly described in [Section 2.1](#).

CONCEPT

Many factors can influence the driving behaviour of a human driver. When it comes to highly automated vehicles, the vehicle needs to perceive the environment on its own. The higher the level of automation, the more important is the perception of the environment. Perception can be increased by using more and different types of sensors, but they do not enable the detection of out-of-sight objects or support an interaction framework with other vehicles. For that reason and the low cost, wireless communication between vehicles is considered as a promising addition to automated vehicles.

Automated and cooperative vehicles are introduced due to the need for traffic safety and efficiency [3], and reduction of air pollution. V2V communication is used to increase the line-of-sight with the goal to improve the vehicle performance. However, using other vehicles' information also brings safety and security concerns with it. As listed in Section 2.2, there are already proposed mechanisms to establish trust between each other. An evaluation of the other vehicle's sensor quality/precision or of its behaviour has not been in the focus yet. In an operational environment, the vehicle's might not provide equally reliable information and thus, a system that evaluates the received information is necessary.

The system of the vehicle needs to adapt to changing situations. It needs to take certain factors into consideration that require changes in the controller of the system. Therefore it is necessary that the system can rely on the information provided by the other vehicles, in order to make more robust decisions. The factors that may influence the vehicle's driving behaviour or decision making are illustrated in Figure 4.1.

SENSOR QUALITY. The knowledge about the precision/accuracy of the vehicles is important to make profound decisions based on this information. Sensor quality is essential when it comes to the data of other vehicles because the ego vehicle may not be able to verify the information with its own sensors. The CAM contains fields describing the accuracy of the measurements, but it might be the case that the sensor, for instance the position via GPS or Galileo, is highly dependent on the environment and thus the message field might not get sufficiently updated.

STATIC ENVIRONMENT. Most of our environment on the road does not change frequently, e. g. tunnels, bridges are also considered as

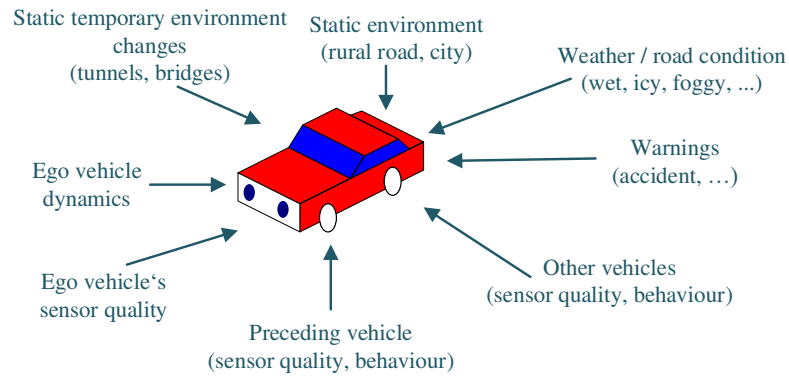


Figure 4.1: Factors influencing the awareness of a vehicle.

static. The environment has a strong impact on our sensors. For instance, the satellite connection of the [GPS](#) device is dependent on the environment, due to the reflection of electromagnetic waves within cities.

DYNAMIC ENVIRONMENT. Loss of messages can be an indicator of an environment change. During tests with a vehicle able to use [V2X](#) communication, it was experienced that trucks can block the communication between the preceding vehicle of the truck and the vehicle in the back, due to the physical properties of a truck. A research by Larsson [48] covers this topic of forwarding messages within a platoon of heavy duty vehicles. Identifying that there is a temporary communication loss caused by another vehicle is necessary in order to be aware of the current situation. Roadwork messages or other important warnings might be received late due to the lack of communication with vehicles or infrastructure in front of the truck.

BEHAVIOUR. The longitudinal and lateral controller design is depending on the manufacturer or even on the model of the vehicle. For that reason it has to be considered how the vehicle behaves or reacts to certain events. For example, a vehicle following another vehicle, that has a high fluctuation of the velocity, should be able to compensate the fluctuating speed to maintain string stability. A further indicator of the behaviour is the observation of the vehicles while they are interacting with each other.

[Section 4.1](#) describes the [TS](#) in more detail. The composition of the [TI](#) and how it can be used for decision making is explained in [Section 4.1.1](#). Perception is needed for increasing the situation awareness of the automated vehicle. The way of how the proposed system senses its environment is explained in [Section 4.2](#). The process of decision making and the chosen architecture is described in [Section 4.3](#).

4.1 TRUST SYSTEM

The introduction of this chapter illustrated the factors influencing a vehicle's driving behaviour in Figure 4.1. These factors should be considered for an autonomous vehicle using V2X communication in order to perform robust decisions and increase safety. The tasks of the TS are the perception of the environment, and the creation of a TI that indicates the trustworthiness of other vehicles, the trust in the environment and the trust in its own sensors. Verifying also the performance of the ego vehicle is an essential part of the TS in order to make the own performance comparable to the other vehicles.

The aim of using a TS is to have an entity in the software architecture of the vehicle that is responsible for the perception and evaluation of the current situation and traffic participants. This module offers the other components of the vehicle's system more detailed information about the performance of other vehicles to make their decision based on that information. The perception is not only the task of the TS, it complements the SF by creating a map of the vehicles that are surrounding the ego vehicle. Figure 4.2 illustrates the communication flow between the SF and the TS. The TS receives so-called trust information from the ego and cooperative SF in order to evaluate the other participants. The position accuracy values, HDOP and VDOP, are gathered from the GPS device and provided by the Data Acquisition for other modules.

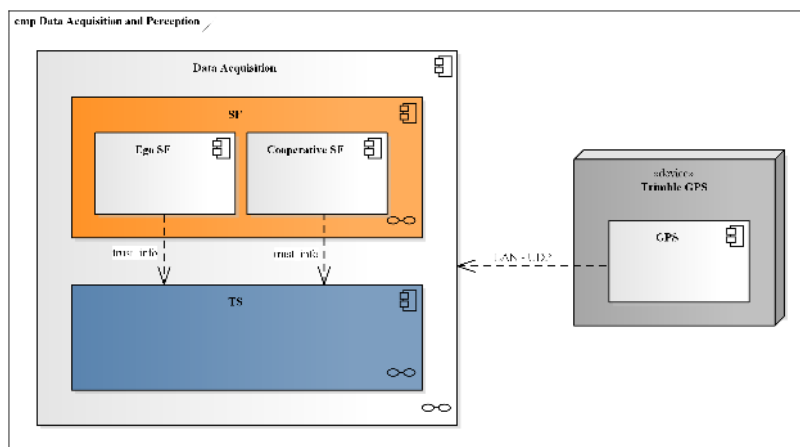


Figure 4.2: Composition of the Data Acquisition module.

The TS receives the information illustrated in Figure 4.1. The most significant output of the TS is the TI and a map that shows the surrounding vehicles. Since the TS already generates the map, it also calculates the measurements with respect to the MIO. The identification of the MIO is necessary for the TS in order to evaluate the trust in the preceding vehicle. This vehicle is the most important one for the reason that the behaviour of this vehicle is directly influencing the safety of the ego, and all other vehicles close to it.

Additional data generated by the **TS** is the trust in the environment, the trust in the other vehicles, and the trust in the ego vehicle itself. The evaluation of the ego vehicle is necessary in order to make it comparable to the other traffic participants. [Figure 4.3](#) depicts the data produced by the **TS**.

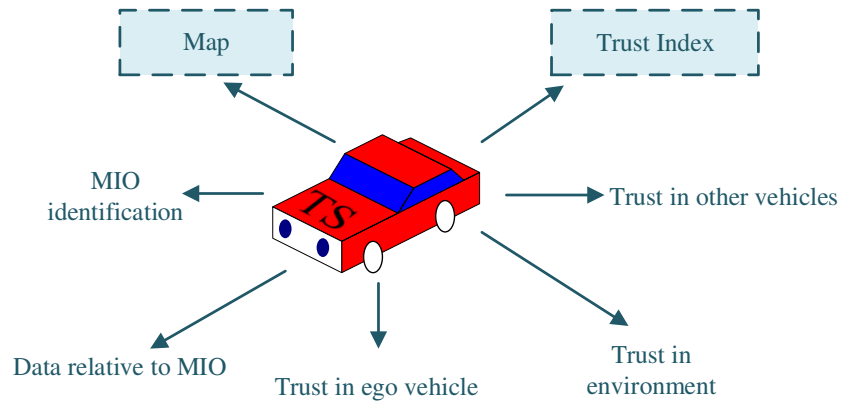


Figure 4.3: Data calculated by the **TS**.

TRUST INDEX. The **TI** is the major product of the **TS**, it indicates the overall trust level at the current point of time. Additionally, the **TS** calculates a separate **TI** for the environment, the ego vehicle, and the other vehicle. A more detailed description of the proposed **TI** can be taken from [Section 4.1.1](#).

MAP. The map is essential to increase the perception of the vehicle. It tells the **TS**, as well as the other modules, about the vehicles it is surrounded of. To increase situation awareness, all vehicles in front of the ego vehicle are vital, but for interacting with other vehicles in order to achieve a common goal, it is also necessary to know the vehicles on the other lanes and those behind. The identification of the **MIO** and the data relative to the preceding vehicle are a side product of the **TS**. The data calculated with respect to the **MIO** is listed in [Table 3.1](#) in [Chapter 3](#).

4.1.1 Trust Index

The **TI** has to be designed as a value within a certain interval that indicates the current overall trust combining the trust in the environment, in the vehicle itself, and the trust in the other vehicles.

An absolute **TI** has been chosen for the reason that such an index is easier to exchange with other vehicles in case of using a distributed database providing **TIs** of other vehicles. Sharing a **TI** that is relative to the behaviour of the ego vehicle makes it difficult to share with other vehicles, since the index is subjective.

*For instance, the identification of the vehicle behind on the right lane is necessary for the **GCDC 2016** highway scenario.*

In order to combine the different TI s to have one index that describes the overall trust in the environment and the vehicles, a weighted average is proposed. The formula for the weighted average TI can be taken from Equation 4.1.

$$\text{TI} = \frac{\sum_{n=0}^N w_n \cdot \text{TI}_n}{\sum_{n=0}^N w_n}, \quad (4.1)$$

where

- TI is the combined TI ,
- N is the total number of TI s,
- w_n is the weight for that specific TI , and
- TI_n is the index of a certain TI .

The weights for each of the TI s will be defined in Chapter 5. The defined TI s are listed in Table 4.1 and will be further named by their associated shorthand symbol.

All indices throughout this thesis are set to be in an interval of $(0, 1)$. Table 4.2 provides a literal description of a few TI values.

Having an index that contains information about the current environment, the ego vehicle, and the other vehicles can be used by the decision making algorithm to adapt its speed according to this information or to alert the driver that it is not safe to drive in automated mode. The driver needs to be aware of the situation as long as the vehicle is L3 on the scale for automated vehicles. L4 vehicles would benefit from this index by having one index that describes the overall situation. It has to be said that the system cannot only rely on one index, but it can be used as a basis to make a decision and consider the details in further steps.

4.1.2 Trust Index TI_{ego}

This TI indicates the trust of the ego vehicle in its own sensor data. Validating the own trust in the sensor quality is essential in order to provide information about the awareness. If the system is not able to trust its own sensor data to a certain extent, the system will not be able to perceive its environment correctly and thus the passenger's safety of the ego and the surrounding vehicles would be at risk. The EKF provides the Kalman gain K_k at each time step and reflects how much "trust" it puts into the predicted state x_k^- , respectively the new measurement z_k . The HDOP and VDOP values provided by the GPS device indicate the position accuracy.

Figure 4.4 illustrates the process of computing the TI_{ego} . The first step is to compute a partial TI that contains the GPS precision ac-

An increasing K_k indicates more "trust" in the measurement (see Section 3.2).

SYMBOL	DESCRIPTION
TI_{ego}	The trust in the ego vehicle according to sensor quality. The HDOP and VDOP values, and the EKF results can be considered for this index.
TI_{mio}	The trust in the preceding vehicle combining sensor quality and behaviour. The linear model explained in Section 3.1 provides additional knowledge about the sensor quality of a vehicle by comparing and fusing the radar distance d_r with the geographical distance. Moreover, a monitoring of the difference between these two distance values can lead to the evaluation of the controller behaviour.
TI_{env}	The trust in the environment, e.g. urban, rural, bridges, or tunnels. The HDOP and VDOP values can be integrated in this TI as well. Mapping these values to the geographical position, where they have been experienced, is one way to predict signal loss.
TI_{v_i}	The trust in vehicle v_i combining sensor quality and behaviour. This TI does not have a strong influence, for the reason that the verification of the measurement quality cannot be directly verified with a sensor of the ego vehicle.

Table 4.1: Listing of the proposed [TIs](#).

TRUST LEVEL	DESCRIPTION
0	Do not trust the system at all.
0.5	Neutral, the system does not have enough evidence to trust or distrust the system.
1	Fully trust, the system trusts the provided information entirely.

Table 4.2: Description of certain trust levels.

According to the former mentioned [DOP](#) values. These values can be infinitely large [49]. For that reason, it is necessary to set a limit that indicates a highly inaccurate measurement. This limit is set to 25, as a precise geographical position is an requirement for cooperative driving. The limit has been gained by heuristic analysis of the [DOP](#) values while driving under different conditions. The [VDOP](#) value is usually higher than the [HDOP](#) value because the signals received from the satellites are coming from above and not from all sides as with the horizontal coordinates. This is also the reason, why the vertical posi-

tion is more inaccurate. The Position DOP (PDOP) value is a combination of the former DOP values and used for further computation [49].

The PDOP value is calculated according to [49]. Equation 4.2 pro-

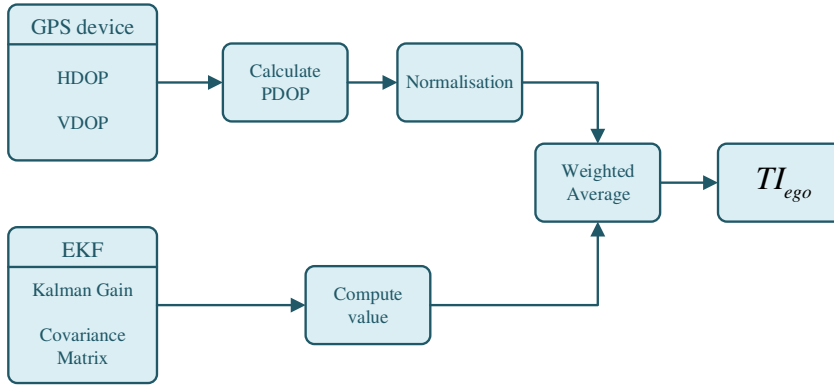


Figure 4.4: Composition of TI_{ego} .

vides the formulas for calculating TI_{ego} . The first step is to compute the PDOP value and normalise it with a maximum value of 25. Next, the Kalman gain has to be evaluated. This is done by calculating the mean of the diagonal of K_k , which contains the factor of “trust” for each measurement. The overall TI of the ego vehicle representing the quality of the sensors is achieved by applying a weighted average of the indices describing the EKF and the GPS accuracy.

Each measurement described in z_k has an equal weight for calculating TI_{ekf} described in Equation 4.2c. A higher weight for the position accuracy is not necessary for the reason that the quality of the GPS position itself is already considered in the TI_{gps} . Indeed, the position is also located within the EKF, but this position is estimated using the motion model described in Section 3.2.

$$PDOP = \sqrt{HDOP^2 + VDOP^2} \quad (4.2a)$$

$$TI_{gps} = \begin{cases} 1 & \text{if } PDOP > 25 \\ \frac{PDOP}{25} & \text{otherwise} \end{cases} \quad (4.2b)$$

$$TI_{ekf} = \frac{\sum_{i=1}^N K_k(i, i)}{N} \quad (4.2c)$$

$$TI_{ego} = \frac{w_{gps} \cdot TI_{gps} + w_{ekf} \cdot TI_{ekf}}{w_{gps} + w_{ekf}} \quad (4.2d)$$

4.1.3 Trust Index $TI_{m_{i0}}$

$TI_{m_{i0}}$ and TI_{v_i} are both combining trust in the measurement quality and trust in the behaviour of the other vehicle. A definition of correct behaviour or wrong behaviour is highly complex, since one can consider many factors that help to define behaviour itself. A survey of the factors expressing the behaviour of a vehicle is shown in this Section.

BEHAVIOUR. The general behaviour of vehicles can be described through observation and past experiences with the vehicle. Figure 4.5 illustrates the identified factors that can be used to conclude about the behaviour of a vehicle. The current velocity itself provides information about the cautiousness of the driver or automated system. An autonomous system should have the ability to perceive the environment to a certain extent to make correct decision about the speed. For instance should the system or the driver adapt its or his speed when it is raining heavily or if the road is slippery. Speed limits have to be obeyed, as well.

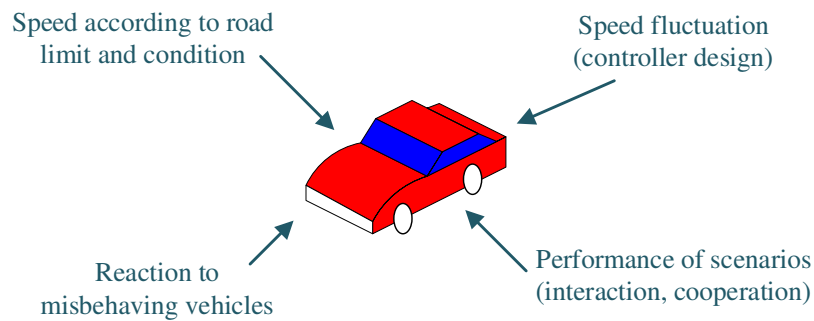


Figure 4.5: Factors describing the behaviour of a vehicle.

The system of other vehicles might do something wrong or unexpected due to a lack of perception. Vehicles have to react properly even if other traffic participants make a mistake. Observing misbehaving vehicles and the response to this actions by other vehicles helps to identify the robustness of the system of a vehicle.

The interaction with the surrounding vehicles is important for cooperative driving. The main requirement for cooperative driving is using a common interaction protocol. Especially the [GCDC 2016](#) has shown that the implementation of the interaction protocol may differ. It may be also the case that vehicles have problems with the interoperability and interpret the received data wrongly. Vehicles not performing according to the interaction protocol have to be identified and properly treated when executing a scenario together.

Additionally, the velocity profile of the vehicles can be observed. Are the vehicles driving continuously or is the speed fluctuating when

driving on a straight road. This fluctuation is an indicator for the implemented speed controller.

The **MIO** is identified via the map generated by the **TS**. An explanation of how the map is designed can be taken from [Section 4.2](#). Since the **MIO** is important for road safety and the sensor data can be compared to one of the ego vehicle's sensor, it offers a more distinct **TI** compared to TI_{v_i} .

DISTANCE. As described in [Section 3.1](#), a method to verify the transmitted position of the preceding car is to compare the distance measured by the radar with the distance calculated by using the geographical position of both vehicles. Monitoring the error of these two distances as shown in [Equation 4.3](#) over time allows one to figure out how the error is distributed. The knowledge of the mean and its deviation infer with the fluctuation of the speed and thus used as an indicator of the precision of a sensor.

$$e_d = \sqrt{(d_r - d_{gps})^2} \quad (4.3)$$

TIME HEADWAY. The proposed observation of the distance error over time has the drawback that the distance is not constant, but varies with speed. The distance to the vehicle in front can be measured with the time headway h . The desired distance required for the participation at the **GCDC 2016** is shown in [Equation 4.4](#) [50].

$$d_{des} = r + h \cdot v, \quad (4.4)$$

where

- d_{des} is the desired distance,
- r is the standstill distance (typically 6 metres),
- h is the time headway, and
- v is the current velocity.

The time headway used in [Equation 4.4](#) is within the range of 0.6 to 1 second [50]. Caused by the fact, that the time headway indicates the time gap to the vehicle in front, one can calculate the current time headway and monitor this value over time. This method considers the speed of the vehicle and thus does not change abruptly when changing the velocity. [Equation 4.5](#) shows the calculation of the time headway.

$$h = \frac{d}{v}, \quad (4.5)$$

where

- d is the distance to the vehicle in front, and
- v is the current velocity of the ego vehicle.

Monitoring the time headway takes also the velocity of the MIO into account. A problem using this method is that the time headway might not change significantly due to the reaction of the ego vehicle's controller. The controller of the test vehicle reacts for safety reasons to deceleration much faster than to accelerations.

The velocity of the vehicle is sent via CAM to the surrounding vehicles.

SPEED. The stability of the speed of the MIO can also be considered by analysing the speed profile. A condition for using the velocity profile of the MIO is a second sensor that is able to measure the speed of the preceding vehicle. A front radar is able to detect the distance to an obstacle in front and the range rate. The range rate is defined in [51] as the rate of the change of the distance to the obstacle as defined in Equation 4.6a.

$$\dot{R} = \frac{dR}{dt} \quad (4.6a)$$

$$v_{mio} = v_{ego} + \dot{R}, \quad (4.6b)$$

where

R is range or distance to the detected obstacle,

\dot{R} is range rate, and

v is the velocity of the MIO or ego vehicle.

Equation 4.6b shows the calculation of the preceding vehicle's speed considering the speed of the ego vehicle and the range rate \dot{R} of the MIO. The range rate in m/s provides information about the change of the distance in metre. The formula is only valid if the preceding vehicle is moving in the same direction as the ego vehicle. This requirement is fulfilled when driving on a straight road. The calculation is not valid when driving in curves.

INTERACTION. The communication between the vehicles during a scenario can be observed as long as they are within the range of the ego vehicle. The highway scenario, scenario 1, of GCDC 2016 relies on the correct pairing of the vehicles in order to perform a safe merge from one lane to the other. The pairing procedure is described in the i-GAME deliverable 3.2 [19]. Observing a wrong pairing of the MIO, or another vehicle around the ego vehicle, can be used for consideration in both, the TI_{mio} and the TI_{v_i} . Prospective scenarios for cooperative driving may have better measures to be considered in a TI.

The interaction during a scenario is described in the i-GAME proposal [19].

Another measure to validate the correct implementation of the interaction protocol is the observation of the Safe-To-Merge (STOM) message. Vehicles that have paired up correctly are making a gap relative to their pairing partner. As soon as the vehicle making the

gap for the merging vehicle decides that the gap is large enough to merge, it sends out a **STOM** message to tell the merging vehicle that it can safely merge. Observing the created gap when the **STOM** message is sent, can be used to verify the trust in the vehicle when interacting with each other.

RECOMMENDED SPEED. The speed of the preceding vehicle indicates how safe it is driving. Considering the weather and the road conditions in combination with the knowledge about the current speed limit allows to conclude a recommended speed that should be driven. The information about the road condition can be measured with the own vehicle or by using the received road information from a **RSU** that warns vehicles about dangerous situations. With regards to **AI**, an inference engine can be implemented to make a decision about the recommended speed that should be driven.

REACTION TO MISBEHAVING VEHICLES. The robustness of the vehicle's control system is important and should be considered in the **TI** computation for each vehicle. Is the vehicle not able to react to misbehaving vehicles properly, the life of the passengers in the vehicle itself and the surrounding vehicles might be at risk. Analysing the behaviour of all vehicles and evaluating the reaction of the surrounding vehicles needs significant computational power and different types of sensors are necessary.

Figure 4.6 illustrates the measures that can be considered for the TI_{mio} . The partial **TI** representing the interaction experience with the **MIO** evaluates the correct pairing and sending of **STOM**. Both results are summarized as one index using weighted averaging. The degree of the speed fluctuation is mapped to a predefined value. Taking into account the factors weather, the road condition, and the current speed limit allows the computation of the recommended speed using inference. A comparison of the recommended speed and the speed of the preceding vehicle is mapped to a partial **TI**. The forth partial **TI** is the reaction to misbehaving vehicles. A weighted average is used to combine these four indices to calculate the TI_{mio} .

The factors illustrated in Figure 4.5 have been identified as potential influences of this **TI**. Further, Figure 4.6 shows the composition of TI_{mio} . For the reason that the test vehicle was not able to perceive all the mentioned measures, such as the detection of the speed limit, road condition, and the observation of the reaction to misbehaving vehicles, this work focuses on the following three factors: **EKF** result of the non-linear model, the speed fluctuation, and the performance when interacting with other vehicles.

4.1.4 Trust Index TI_{env}

The environment is categorised into static and dynamic. The static environment does not change frequently, such as in urban or rural

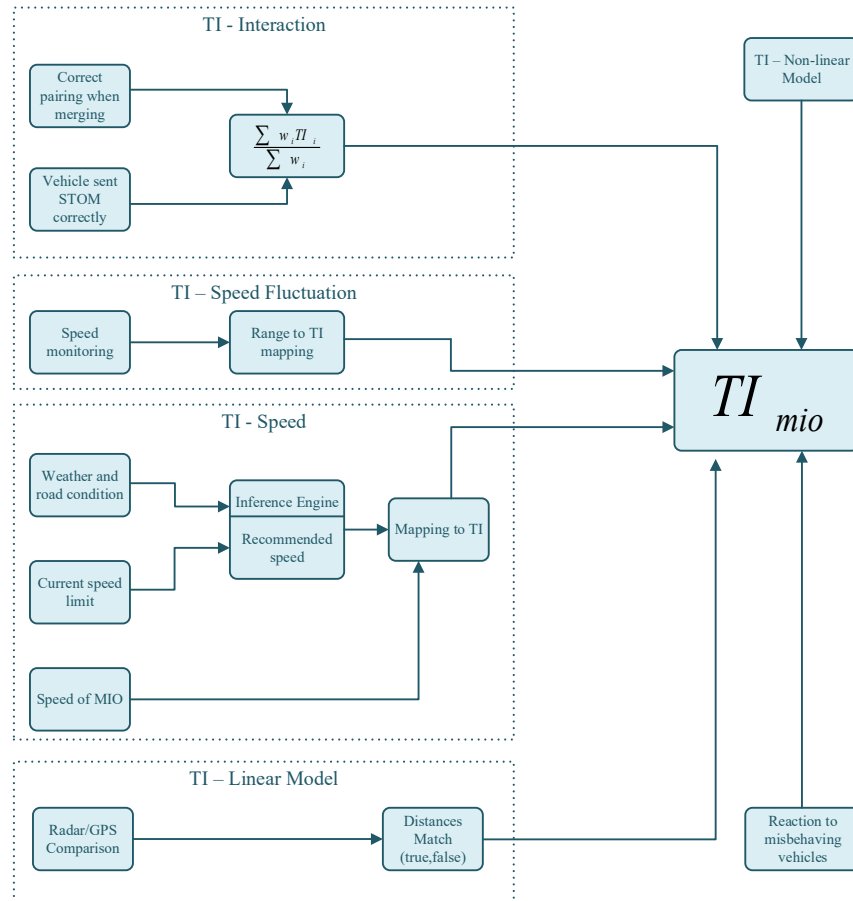


Figure 4.6: Composition of TI_{mio} .

areas, bridges, or tunnels. Dynamic environment changes are for instance the lack of communication to vehicles that are in front of a heavy duty vehicle. Due to the physical properties of trucks, they are blocking the wireless communication to the vehicles in front or behind of the truck. Research about algorithms deciding what should be forwarded by the truck in order to enable the communication for all vehicles within a certain range is ongoing. Larsson proposes in [48] a performance centric forwarding algorithm.

There are methods on how to decide whether the vehicle is currently in a city or on a rural road. An approach to achieve this task is the matching with a street map that evaluates the environment and informs the system about the current environment of the vehicle. Further, it is possible to learn about the environment by monitoring the [PDOP](#) value provided by the [GPS](#) device. A combination of both approaches integrates learned experience about the environment with preexisting knowledge.

[Figure 4.7](#) shows the different types of environments that have been identified. High buildings close to each other such as in a city cause

reflections of the electromagnetic waves from the satellites and thus the GPS position becomes inaccurate. Tunnels or bridges cause a temporary position loss. When driving on rural roads the position is accurate as long as the signals are not disturbed by large trees that surround the road. The index describing the environment according

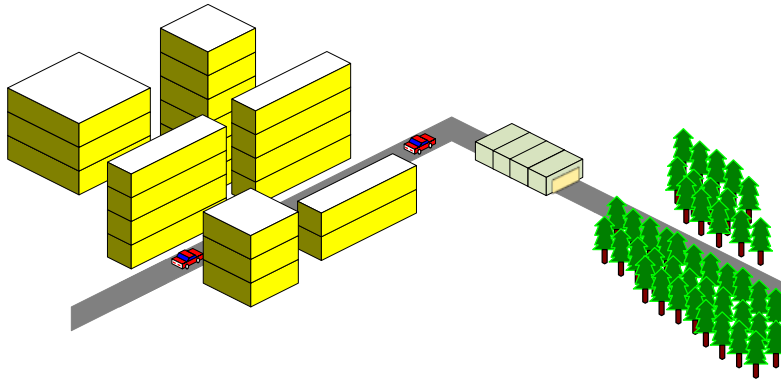


Figure 4.7: Types of environment in which a vehicle is driving.

to a street map and the index providing information about the experienced position accuracy can be combined with a weighted average. The TI_{env} presented in this thesis uses the experienced $PDOP$ value. The mapping of the learned accuracy is done with circles of a certain size, e.g. 5 metre. Further, it has to be highlighted, that the TI_{env} maps the experienced $PDOP$ value to a certain level between 0 and 1 which refers to the expected accuracy of the position for the reason, that the precision of the position is in relation to the environment.

Another approach of computing the TI_{env} is to create a scale that considers also other dangerous situations or situations where an increased awareness is necessary. Such a situation is city traffic. Within a city, the vehicle has to drive more carefully because of the reduced distance to the other vehicles and the presence of pedestrians on the pavement.

4.1.5 Trust Index TI_{vi}

This TI is quite similar to TI_{mio} . A system with enough resources for computation can observe and evaluate the interaction of the surrounding vehicles with each other. Moreover, more sensors and different types of sensors are needed in order to evaluate the precision of the other vehicle's information.

Another challenge is the calculation of a TI considering observations made by other vehicles. More sensors can increase the perception of the ego vehicle, but they can only be used to a certain extent. When it comes to the evaluation of vehicles that are not surrounding the ego vehicle, the system has to rely on the measurements from

other vehicles. The position of a vehicle can be compared with the distance to the **MIO** transmitted from the vehicle behind. For the reason that the ego vehicle cannot fully trust this measurement, the **TI** of the vehicle behind has to be considered in the **TI** calculation of the target vehicle.

4.2 V2V PERCEPTION

The different levels of automation are described in Section 2.2.

The knowledge of the vehicle is strongly dependent on its level of automation. If it is a fully autonomous vehicle, the system needs to detect the lanes, obstacles and other vehicles, pedestrians, traffic lights, speed limits, and road conditions. The system used to evaluate the concept of this thesis has a longitudinal controller, the lateral interaction with the car has to be done manually. This restriction decreases the number of sensors needed in order to be able to drive automatically. The perception of the test vehicle provides the data gathered from the built-in radar, and the data from the other vehicles received via **V2V** communication.

A map of the surrounding vehicles is essential for fulfilling the interaction of the scenarios described in [19]. The system needs to identify the vehicle on the front right/left and also the vehicle behind it on the right lane. To generate a map that fulfils this requirement with the available sensors, the system has to rely on the vehicles position to a certain extent, because the vehicle itself is only able to detect a vehicle in front by using the built-in radar.

The identification of the surrounding vehicle's position can be determined by considering the lane in which the vehicle is currently driving or by computing the relative angles to the vehicles with the shortest distance. Both approaches are computed and further on combined with each other in order to achieve a more robust identification of the vehicle's relative position.

LANE. The **CAM** as well as the **iCLCM** message set contain a field describing the lane in which the vehicle is currently driving. The **CAM** lane is an integer in the range of -1 to 14 . -1 refers to a position outside of the road, 0 is the hard shoulder and 1 is the outermost driving lane [17]. By taking the lane position of the vehicle into account, the algorithm is more robust against misidentified vehicles for the reason that this algorithm is also applicable in curves.

ANGLES. Considering only the relative angles of the vehicles and their distance provides a more robust identification of vehicles that are sending a wrong or no lane ID. The downside of only using this algorithm for identifying the relative position of the surrounding vehicles is the wrong classification of vehicles when driving in a curve.

A method that combines both of the proposed algorithms takes advantage of both. For the reason that the vehicles experienced in the [GCDC 2016](#) mostly sent the correct lane ID, this algorithm is used as a basis. In case a vehicle is identified at another position by the map that considers only the angles and the distance, this field is also updated in the overall map. This approach makes it possible that one vehicle is identified at two places, for instance front-left and front. The fault tolerance of this method will be further discussed in [Chapter 6](#).

[Figure 4.8](#) illustrates the ego vehicle and the relative position description of the surrounding vehicles. The first letter contains the information if the car is in the front of the vehicle or behind it, while the other letter indicates left or right. The blue areas indicate the angle range that is used for classifying the position of the vehicles. A more detailed description of both algorithms and the integration in the system is shown in [Chapter 5](#).

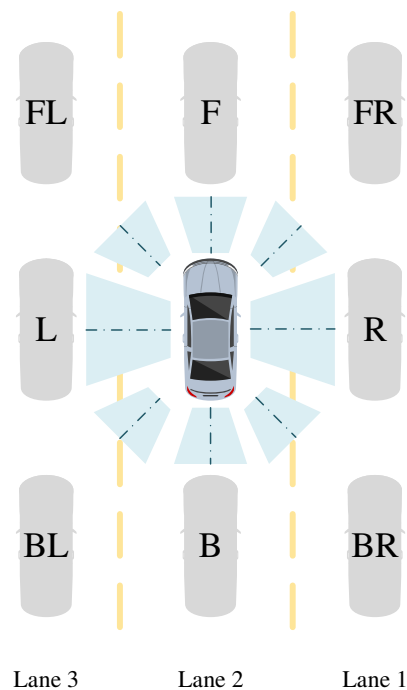


Figure 4.8: Illustration of the map and the classification of the surrounding vehicles.

4.3 DECISION MAKING

The high-level decisions, such as following a certain vehicle, pairing with a vehicle, or merging, are realised as states of a finite state machine. The system itself will be manually triggered with a Human Machine Interface (HMI) to start automated driving or it automatically triggers new states in case of an interaction between the vehicles. It needs to be highlighted that the [HMI](#) is asking the driver for confir-

mation when it comes to decisions such as lane changing. The confirmations are for safety reasons and were required by the [GCDC 2016](#). Moreover, this kind of confirmations increase the situation awareness of the driver, because the system is not a L4 or L5 autonomous vehicle.

The states have conditions that need to be fulfilled in order to trigger a new state. The conditions to make a transition to a new state are to some extent defined by the [GCDC 2016](#) interaction protocol. Additionally, the conditions contain checks that consider the trustworthiness of the information provided by the vehicle, as well as its behaviour.

A more robust decision can be achieved by taking the proposed [TI](#) into account. This index tells the decision making module the reliability of the current information by considering the described factors. The [TI](#) has a bigger influence to the decision of the vehicle with a higher level of automation due to the ability to perceive the environment to a greater extent and the increased control of the vehicle.

4.4 SUMMARY

This Chapter provided the design concept for a system that is able to make more robust decisions based on trust in the own vehicle, the surrounding vehicles, and the environment. Factors that are influencing the awareness of the system of a vehicle have been identified.

Further, the desired output of the [TS](#) has been determined. A combination of the trust in the environment, the own sensors of the vehicle, and the vehicle in front are described as the [TI](#) to increase situation awareness of the system. A detailed explanation of the different types of [TI](#), as well as their calculation can be taken from [Section 4.1.1](#). The fourth partial [TI](#), the TI_{v_i} , expressing the other vehicles visible to the system of the vehicle, has been described but will not be implemented due to missing computation resources and the lack of sensors that verify their transmitted data.

Additionally, the concept of how to perceive the surrounding vehicles has been described in [Section 4.2](#). The use of two different approaches of how to identify the relative position of the surrounding vehicles has been chosen in order to increase the robustness of wrong transmitted lane IDs, but also to withstand curves where the identification using only the angles most probably fails.

The decisions are made according to a state machine that builds up on the interaction protocol for the three [GCDC 2016](#) scenarios. In addition, the triggers for making a transition to a new state are considering the [TI](#) provided by the [TS](#).

IMPLEMENTATION

This chapter describes the architecture of the designed system for creating a **TI** that can be used to make more robust decisions based on trust. The following paragraphs briefly describe the setup of team Halmstad's **GCDC** car in order to comprehend design choices. [Section 5.1](#) provides a detailed explanation of the implemented **SF**, **TS** and decision making module. Additionally, [Section 5.2](#) describes the overall architecture of the **GCDC** 2016 system as well as the internal communication between the different modules in more detail.

Java¹ has been chosen as the primary programming language for the system of the **GCDC** competition car of team Halmstad. For that reason, also the trust modules have been implemented in this programming language. The communication between the modules of the system has been achieved with Lightweight Communications and Marshalling (**LCM**)². The benefit of using **LCM** is the ease of replaying log data and the real time observation of the exchanged data.

The competition car, a Volvo S60, has been provided by Volvo Car Corporation. [Figure 5.1](#) shows the competition car with its various antennas for the differential **GPS** and the wireless communication on the roof. Additionally to the antennas, there are two lights indicating whether the vehicle is in automated (green light) or manual (red light) mode.



Figure 5.1: The **GCDC** competition car from team Halmstad (photo taken at the RDW Lelystad, the Netherlands).

The components of the car are shown in [Figure 5.2](#). The power is provided via a 12V to 220V power converter which is attached to

¹ See: <https://java.com/en/> (2016/07/01)

² See: <https://lcm-proj.github.io/> (2016/07/01)

*The MicroAutoBox
has been provided by
Fengco Real Time
Control AB.*

an uninterruptible power supply that is further connected to all devices. The communication between the devices is established with a wireless router. The V2X communication with the other vehicles and infrastructure is provided by the Alix System Board alix2d3³. In order to have a certain accuracy of the geographical position in the range of $\pm 10\text{cm}$, a differential GPS device is needed. The dSPACE MicroAutoBox⁴ is used as the interface between the system and the car. Furthermore, the MicroAutoBox executes the low-level longitudinal controller. The system of the competition vehicle is executed on a computer, which is located in the front passenger seat.

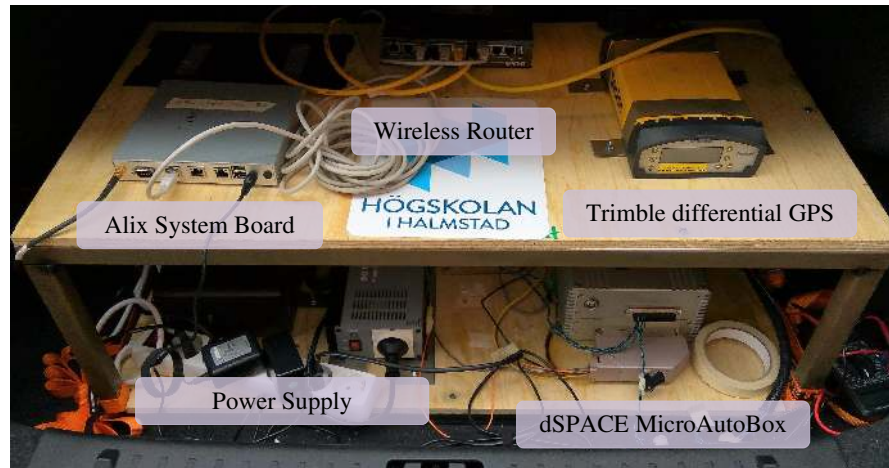


Figure 5.2: The trunk of the competition car with its devices.

5.1 ARCHITECTURE

This Section describes the architecture of each module in more detail. An overview of the communication between the modules and the overall GCDC system architecture is shown in Section 5.2. Segments of the class diagram of the TS are provided within the text below. A detailed class diagram of the TS can be taken from Appendix B in Figure B.2.

5.1.1 Sensor Fusion

The SF is split into the linear and non-linear model. The linear model is integrated in the TS because this model requires both, the measurements from the ego vehicle and the preceding vehicle. The Cooperative SF applies the non-linear model to the V2V data from the other vehicles and the Ego SF applies this model to the measurements from the ego vehicle.

The class *Kalman Filter* is designed to have an option for adapting the variance of the two sensors in order to be able to perform the SF when having an inaccurate geographical position. For that reason,

³ See: <http://pcengines.ch/alix2d3.htm> (2016/07/02)

⁴ See: <https://www.dspace.com/en/pub/home.cfm> (2016/07/02)

the method *feedSensorData* of the filter has four parameters, where two are the measurements and the other two parameters are logical values that indicate whether the measurement is accurate or imprecise.

The linear model is converged within the **TS** that executes the *feedSensorData* method from the **KF**. It updates the distance estimation every time it receives a new measurement from the radar. The same trigger applies for the measurements with respect to the preceding vehicle. The *Distance Calculation* and the *Data Selection* blocks, shown in Figure 3.2, are implemented within the **TS** and the *Kalman Filter* block is realised as a separate class. The setter for the variance have been implemented in order to be able to change the variance of the sensors during runtime. Figure 5.3 shows the class diagram of the **KF** and the **TS**. A more detailed explanation of the **TS** can be found in Section 5.1.2. The functionality of the superclass *LCMService* is described in Section 5.2.

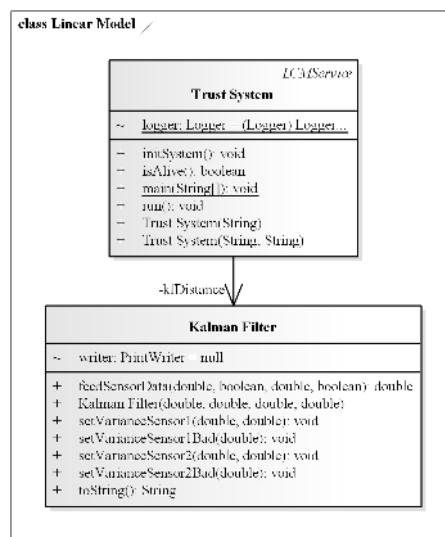


Figure 5.3: Class diagram of the **KF**.

The Cooperative and Ego **SF** are two separate classes running as individual threads on the system of the competition car. Both **SF** classes have an object of type *Filter*, which acts as a wrapper for the implemented **EKF**. This **EKF** class provides the corresponding functions to the equations described in Section 2.1.2 and Section 3.2. Additionally, the **EKF** class is able to switch the measurement noise covariance matrix *R* to emphasise the accuracy of the **GPS**. The change of the *R* matrix is necessary because the sensor data of the **GPS** device is provided in a lower rate and the measurement is thus more inaccurate.

The method to perform the **SF** is named *feedSensorData* as in the linear model. This method not only returns the fused sensor data, it also returns the calculated Kalman gain, which is necessary for calculating the **TI** of the vehicles. In addition, the wrapper class *Filter*

performs the conversion of the geographical position and the correction of the heading as described in [Section 3.2](#). The class diagram of the afore mentioned classes can be taken from [Figure 5.4](#). The reason for having two constructors of the Cooperative and Ego SF is that an additional configuration is necessary for simulating the entire system of the car.

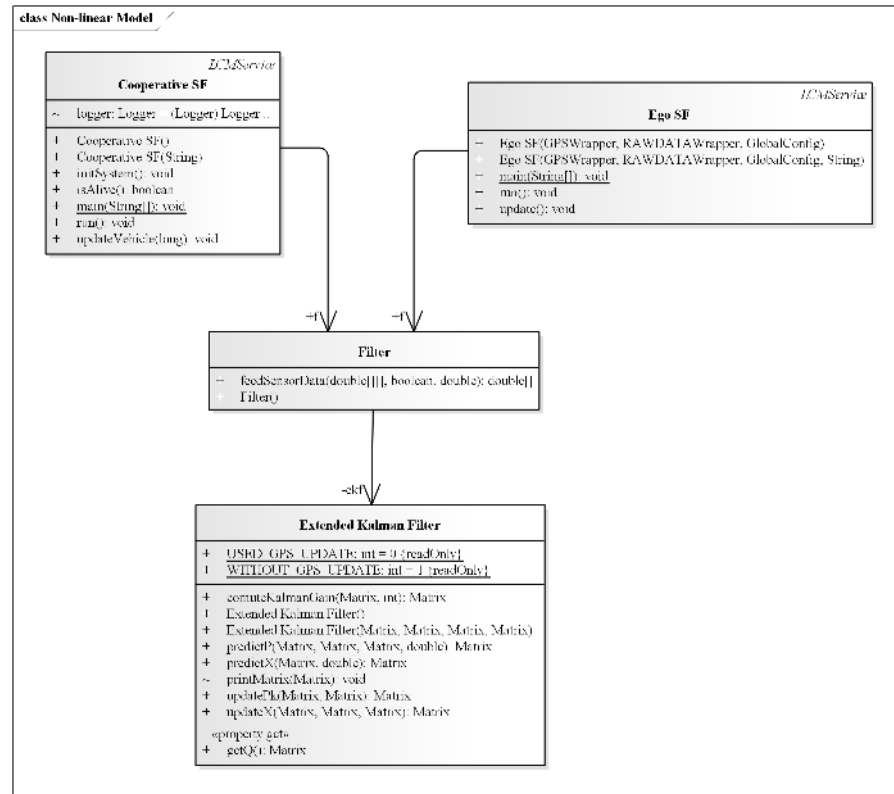


Figure 5.4: Class diagram of the non-linear model.

An explanation of how the sensor measurements are received by the SF and TS modules is provided in [Section 5.2](#).

5.1.2 Trust System

The TS consists of one service class that is receiving and sending information to the entire system. It generates the map of the surrounding vehicles as shown in [Section 4.2](#). Moreover, the TS is responsible for updating the TI by considering the received measurement data, such as the distance to the preceding vehicle, the geographical position of the ego vehicle including the HDOP and VDOP values, the data received via V2V communication, and the trust information sent by other modules of the vehicle's system. This trust information provides details about correct pairing according to the interaction protocol and the Kalman gain calculated by the Ego SF.

First, the generation of the TIs will be described in more detail. Then, the generation of the map including the two algorithms for

identifying the surrounding vehicles is explained. At the end of this section an overview of the **TS** is shown.

The **TI** is split into three separate **TIs** that are explained in [Section 4.1](#). Each of the sub-**TIs** provides methods to update the information and to compute the **TI** of the partial index. The class **TI** is a wrapper class that contains objects of all three sub-indices and provides methods to calculate the overall **TI** of the system. [Figure 5.5](#) shows the class diagram of the **TI** classes. Their functionality is described below.

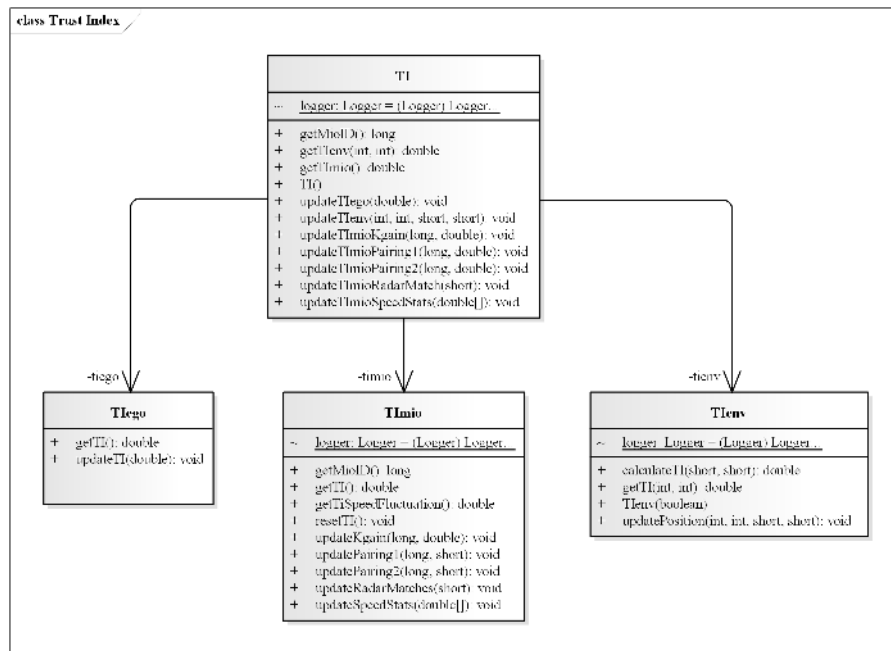


Figure 5.5: Class diagram of the **TIs**.

TI_{ego}. This **TI** considers the **DOP** values that give information about the position accuracy, received by the **GPS** device and the Kalman gain of the non-linear model. This data is produced by the Ego **SF** and sent as trust information to the **TS**. The **TS** updates the **TI_{ego}** according to this information and computes the overall **TI** when receiving trust information updates.

TI_{mio}. The calculation of the **TI** about the preceding car considers

- the Kalman gain of the non-linear model of the **MIO**,
- the interaction with other vehicles,
- the radar matching according to the linear model, and
- the speed fluctuation of the vehicle.

The index representing the Kalman gain is using the average of the sensor's Kalman gain value. The interaction index of the **TI_{mio}** can have three different values, $ti_{int} = \{0, 0.5, 1\}$. The value 0.5 shows that only one method to verify the pairing had a positive result. The

verification of the pairing is performed on the left lane, the merging lane. Correct behaviour, as defined in the interaction protocol, is identified on the right lane by observing the **MIO** when sending the **STOM** message. Figure 5.6 illustrates the two cases, where the ego vehicle is on the left or on the right lane.

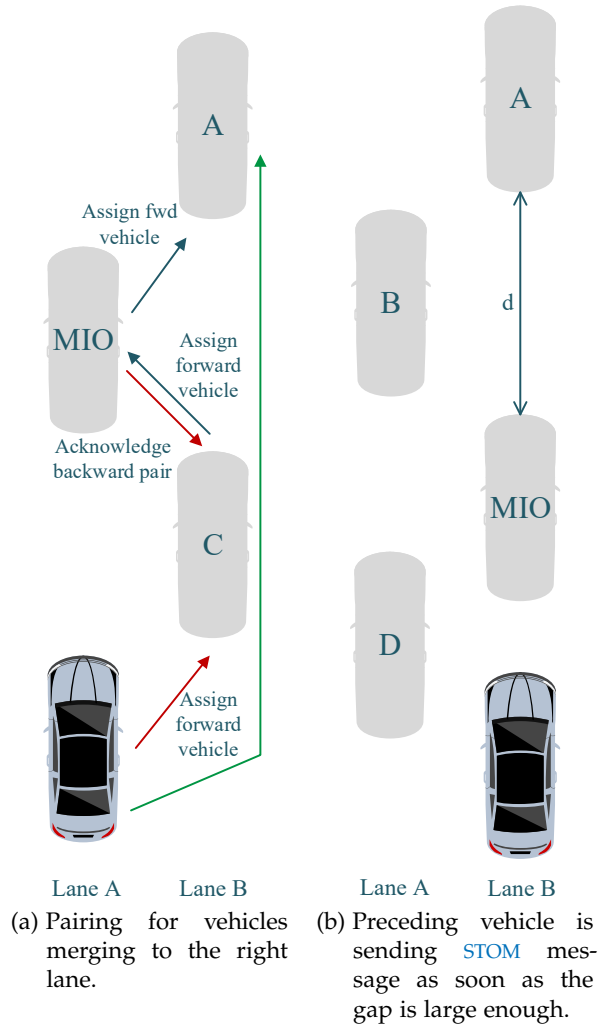


Figure 5.6: Observed interaction between the **MIO** and other vehicles.

Figure 5.6a illustrates two ways of verifying the pairing of the **MIO**. The red arrows illustrate the first approach. The **MIO** has to acknowledge its backward pair. The literal description of this statement can be written as follows: *The forward partner of the ego vehicle has set the **MIO** of the ego vehicle as its forward pair and the **MIO** of the ego vehicle has set the forward pair of the ego vehicle as its backward pair.* Listing 1 shows a pseudo statement where the second term is used to make sure that the forward partner of the ego vehicle (vehicle C) has initiated the correct pairing with the **MIO** of the ego vehicle.

Listing 1: Pseudo statement for the pairing index i

```
Ego.fwd == Ego.mio.fwd && Ego.fwd.fwd == Ego.mio.bwd;
```

The second pairing index is illustrated with the green arrow. This check is comparing the pairing to vehicle A in [Figure 5.6a](#). Literally the check can be described as: *The mio of the forward partner of the ego vehicle is the forward partner of the mio of the ego vehicle*. [Listing 2](#) shows the pseudo statement for this check.

Listing 2: Pseudo statement for the pairing index ii

```
Ego.fwd.mio == Ego.mio.fwd;
```

[Figure 5.6b](#) shows the verification of the interaction, when the ego vehicle is in lane B and the MIO is sending a STOM message for vehicle B. The calculation for the minimum distance is shown in [Equation 5.1](#) and predefined by the GCDC organisation. The STOM message should be sent to the forward partner of this vehicle as soon as this distance has been reached. The distance between the MIO and the preceding vehicle of the MIO (its MIO) is calculated with the use of the geographical position provided by both vehicles via V2V communication.

$$d \geq (2 \cdot f \cdot (r + h \cdot v) + L), \quad (5.1)$$

where

- d is the distance to the car in front,
- f is the scaling factor (set to 0.7),
- r is the standstill distance,
- h is the time headway, and
- L is the length of the merging vehicle (vehicle B in [Figure 5.6b](#)).

The radar match is a binary value that indicates whether the distance calculated with the use of the geographical position of both cars and the radar matches or the difference of the two distances is above a certain limit.

[Section 4.1.3](#) describes three different approaches to detect a speed fluctuation of the vehicle in front by considering the information provided by the radar and the V2V communication. The implemented algorithm uses the range rate \dot{R} and the velocity of the ego vehicle. Further, the other two approaches will be compared to this method in [Chapter 6](#).

The class, *MioStatistics*, is designed for providing the necessary statistical information and has been implemented in a way that it saves the last 1000 values which have a time difference of 500ms. The class offers methods to add new values and to compute the mean and standard deviation. Monitoring the speed with a rate of 2Hz and storing

the last 1000 observations gives statistical information about the past 8.3 minutes.

The mapping follows according to the level of the calculated standard deviation of the velocity of the observed vehicle. The method of using \hat{R} is only possible when the vehicle is in front of the ego vehicle for the reason that the range rate can only be observed with the radar when the vehicle is in front of it.

TI_{env} . The environment can be evaluated by considering the weather, road condition, and the position accuracy. As pointed out in [Section 4.1.4](#), the competition car does not have the sensors for evaluating the two first mentioned factors. For that reason, the TI_{env} relies on the [DOP](#) values.

Since the system saves the observations of the corresponding area, it is able to learn about its environment. If the system has already information about the environment, it uses this data to generate the trust index of the environment by looking at the prospective areas and the corresponding [TI](#).

TI_{v_i} . As pointed out in [Section 4.1.1](#), the computational power requirement and the lack of various different types of sensors to perceive the other vehicles with a second sensor is restricting the calculation of TI_{v_i} tremendously for the system of the test vehicle.

For that reason, the TI_{v_i} has been restricted in this work by computing the [TI](#) of the non-linear model described in [Section 3.2](#) of the forward pair within the [GCDC](#) scenario 1.

The class [TS](#) has a has-a relationship to [TI](#) and the *MioStatistics* class. Moreover, the *Statistics* class serves as a superclass for *MioStatistics* and provides the basic functions for calculating the mean and standard deviation out of a list of values or measurements.

5.1.3 V2V Perception

Next to the computation of the [TI](#), the [TS](#) is also computing a map of the surrounding vehicles and performs the calculation of measurements with respect to the preceding vehicle. As discussed in [Section 4.2](#), the map uses two different approaches in order to make a more robust identification of the surrounding vehicles, the neighbours.

MAP ANGLES. A list of all vehicles around the ego vehicle is received by the [TS](#). The first step is to filter out all vehicles that are not going into the same direction. The information about the direction is gathered via the [CAMs](#) that contain the heading. Next, all vehicles that are more than 100 metres away from the ego vehicle are sorted out because they are too far away for interacting with them. The calculated relative angle to the vehicles is used to identify the relative position of the vehicles. The vehicle with the shortest distance to the ego vehicle of each group, e. g. front-left, is chosen as a surrounding vehicle.

The range for the angles is calculated dynamically by considering the distance to the vehicle and the length or width of the vehicle itself. This dynamic angle range is calculated with the use of [Equation 5.2](#). Furthermore, [Figure 5.7](#) illustrates the parameters for calculating the dynamic angle range. For vehicles in front or behind, the algorithm uses the lane width as parameter. The remaining angle ranges are used to identify the other vehicles, e. g. on the front-right.

$$\alpha = \arctan \frac{\text{view width}/2}{d} = \arctan \frac{L/2}{d}, \quad (5.2)$$

where

- α is half of the angle range,
- view width is the length of the object that should be detected, and
- d is the distance to the object.

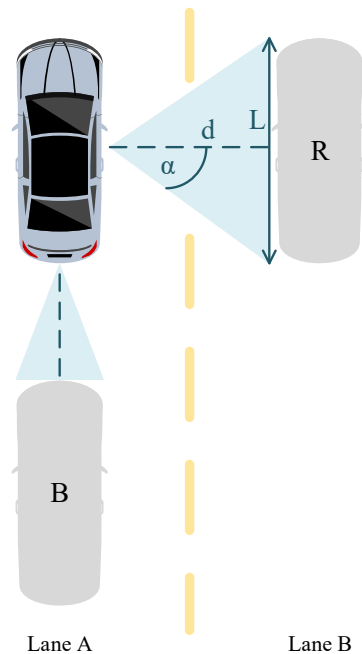


Figure 5.7: Illustration of the dynamic angle range calculation.

MAP LANES. The vehicles are filtered by direction and distance to the ego vehicle, such as in the map that uses only the relative angles. The next step is to put the vehicles into a list of vehicles corresponding to their lane. The relative position of the vehicles is calculated by using a predefined angle range for each position. Vehicles with the shortest distance to the ego vehicle within their category, e. g. front-left (see [Figure 4.8](#)), are chosen for the map.

NEIGHBOURS. The *Neighbours* class is combining both map types in order to have a robust identification of the vehicles. For the reason that it was experienced in the [GCDC 2016](#) that some vehicles did not transmit the correct lane, the map considering only the relative angles needs to be taken into account. *Neighbours* takes the map that uses the lanes as base and adds vehicles from the map considering the angles only in case that the system did not identify a vehicle at this location yet. A detailed class diagram of the [TS](#) can be found in [Figure B.2](#).

5.1.4 Decision Making

A state machine has been designed for making decisions based on the interaction protocol defined in [\[19\]](#). The communication with the driver is performed via a [HMI](#) that provides necessary information about the location of the vehicle, the surrounding vehicles and the current state of the state machine. Furthermore, the driver is able to interact with the [HMI](#) in order to set parameters, e.g. lane ID, the location of the centre of the intersection, the radius, and the time when the vehicle has to enter the competition zone, or to confirm actions. The confirmation of actions is required by the [GCDC 2016](#). It is used for safety reasons and to increase the situation awareness of the driver.

The [TI](#) generated by the [TS](#) is sent to the High-Level Control ([HLC](#)), which represents the decision making module. The [HLC](#) uses this information about the [TI](#) in order to make the driver aware of problems, such as the pairing cannot be performed due to a misbehaving vehicle or a vehicle with faulty sensor data.

[Section B.1](#) depicts the state diagrams for each scenario as well as the state flow for going from automated mode back to manual driving. [Figure B.3](#) illustrates the states for scenario 1, the highway scenario. Goal of this scenario is the automated merging of the vehicles on the left lane to the right lane. Each scenario starts in *STANDBY* and goes over to the scenario specific states as soon as a scenario has been chosen. The state machine splits into two branches due to the different interaction when being in lane A or B.

The state machine of scenario 2, the intersection scenario, is shown in [Figure B.4](#). A more detailed description of the scenarios can be found in [\[19\]](#). Entering all this information is only for test purposes at the [GCDC](#), this information is planned to be broadcasted by the [RSUs](#).

The emergency vehicle scenario, scenario 3, informs the driver as soon as a [CAM](#) or [DENM](#) containing the emergency notification is received and makes a smooth transition to manual driving with the driver's consent. The corresponding state diagram can be taken from [Figure B.5](#).

Changing from automated to manual mode is performed in a safe

way. The vehicle increases the gap to the vehicle in front, to correspond to the reaction time of a human driver. Figure B.6 illustrates this state flow.

5.2 INFORMATION FLOW

Figure 5.8 illustrates the entire system of the test vehicle including the communication between the different devices. The HLC represents the decision making algorithm and the Mid-Level Control (MLC) converts the high level commands, such as *follow vehicle with ID 120 with a distance of 10 metres*, into commands for the Low-Level Control (LLC). The LLC is executed on a separate real time device, the MicroAutoBox that performs the longitudinal control of the vehicle and provides the interface to the CAN bus of the car. Further, the roof lights, see Figure 5.8, indicating autonomous or manual mode and an emergency button to cut off the connection to the CAN bus of the vehicle, are controlled by the MicroAutoBox and a custom-built circuit board. The HMI, a tablet, is communicating via wireless LAN to the system of the vehicle and enables the interaction with the driver. V2V communication is established by the Alix cards that forward the packets to the communication module of the system and sends the encoded messages from the system to the other vehicles. The GPS device is directly communicating with the Data Acquisition module via UDP. If the communication is not further defined, such as LAN - UDP, information is exchanged via LCM messages.

All classes that are exchanging LCM messages have the class LCM-Service as superclass. This class provides the basic functionality to send and receive LCM messages on predefined channels. Additionally, this class offers a heartbeat function in order to identify a crashed module. Due to the fact that LCM uses UDP as transport protocol, the system had to be designed to be resistant against packet loss.

The Cooperative SF receives all CAMs and iCLCMs from the other vehicles from the Communication module as *dirty* messages. The system classifies the unmodified data received by other vehicles as *dirty*. It manipulates the lane information in case it contradicts to the platoon ID field, which is located in the iCLCM. Additionally, the Cooperative SF executes the EKF for the preceding vehicle and the forward partner, and sends the Kalman gain via a trust info message to the TS. All other modules of the system are listening to the *clean* CAM and iCLCM sent by this module. The Ego SF executes the non-linear model of the ego vehicle and generates the CAM.

The HLC module is running the state machine. It receives the *clean* data of the other vehicles, the interaction messages from the HMI, and the TI from the TS. The information used for the calculation of the interaction index for the TI_{mio} is generated by this module. Control messages sent to the MLC are dependent on the current state and the environment.

The platoon ID field starts counting from the most left lane, while the lane ID starts on the outermost right lane.

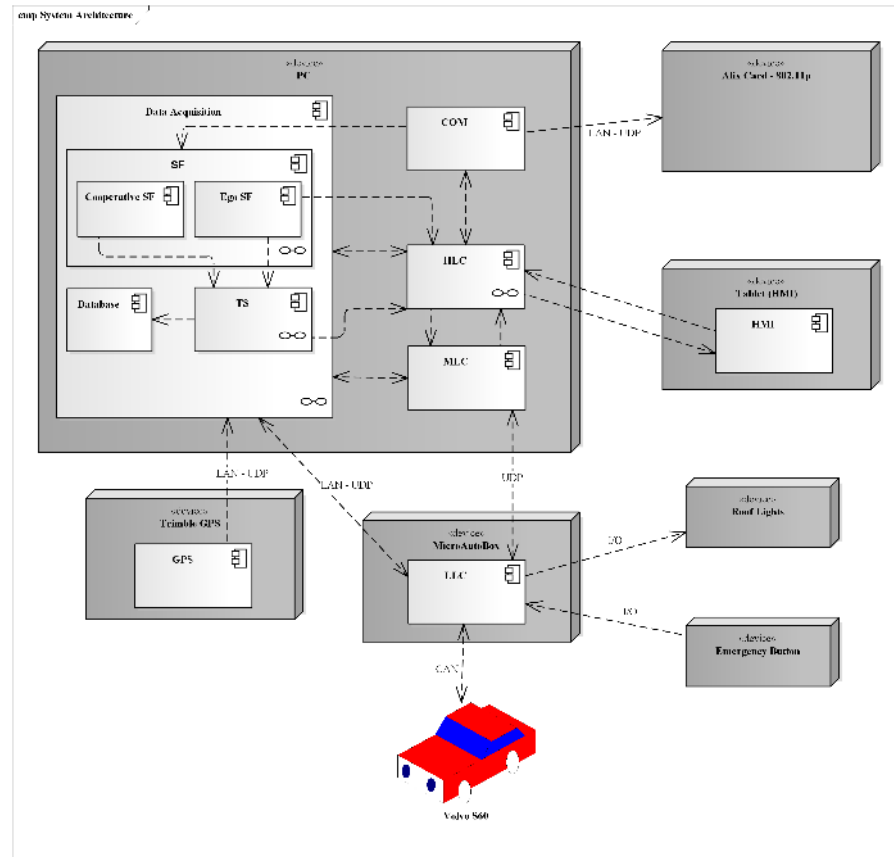


Figure 5.8: System Architecture of the GCDC car.

The **TS** updates the **TI** according to the received trust information, the observations about the environment, and the **MIO**. The map is generated in a separate thread and published as a map **LCM** message.

5.3 SUMMARY

The functionality of the competition car has been taken into consideration when developing the system. The car provided mechanisms to control the acceleration of the vehicle but no mechanism to control the lateral steering. Furthermore, the vehicle provided the distance to the vehicle in front with the use of the built-in radar.

The decision making is mostly based on the interaction with the other vehicles and the distance to the preceding vehicle. The **TI** has been used for verifying the current situation and to alert the driver in case that the system would not be able to perform properly.

The implemented **SF**, and **TS** offer the basis for creating a **TI** that is distributed to the other components of the system of the vehicle. The algorithms proposed for computing each individual **TI**, such as the TI_{mio} , and TI_{ego} , can be extended by adding a new different **TI**. The proposed algorithms can be extended to estimate trust for other sensors if available.

EXPERIMENTS

The [GCDC 2016](#) was judged based on the three scenarios: the highway scenario, the intersection scenario, and the emergency vehicle scenario. The developed [TS](#) is evaluated with data from the first mentioned scenario, because this scenario relies the most on [V2V](#) communication of different vehicles. The intersection scenario only requires the data provided by the Organisation Pace Car ([OPC](#)), and the third scenario is basically listening to emergency messages.

6.1 SCENARIOS

The highway scenario took place on the highway A270 in the Netherlands. The vehicles in the right lane, lane B, were driving with a speed of 60 km/h and the vehicles on the left lane, lane A, were driving at 80 km/h. Due to speed limitations of some participating vehicles, this scenario was split into two heats. The former described velocities are used for the high-speed heat. The low-speed heats have a speed of 45 km/h on lane A and 40 km/h on lane B.

The highway scenario is split into four phases, the pace making, the parallel pairing, the sequential pairing, and the merging phase. [Figure 6.1](#) illustrates the former mentioned steps of the highway scenario [19].

I: PACE MAKING. At the beginning, the [OPCs](#) are bringing the vehicles into the right position. As soon as the vehicles are correctly positioned, a roadwork message is sent to all participants. A roadwork message implicates a roadwork on a certain lane (lane A) and the reduction of all vehicles' speed to 40 km/h.

II: PARALLEL PAIRING. A merge request triggers the so-called *B2A* pairing, which means that the vehicles on the right lane are setting their forward pair, the vehicle on the front-left, in the [iCLCM](#) message. This pairing is performed in parallel by all vehicles. Additionally, their forward partner acknowledges the pairing by setting them as its backward pair. When the *B2A* pairing is done, the vehicles in lane B create a gap so that their forward partner can merge in front of it.

III: SEQUENTIAL PAIRING. After a certain time, the lead vehicle on the left lane pairs up with the vehicle on the front-right and creates a gap. The front-right vehicle is identified as the [MIO](#) of the backward pair. As soon as the gap is large enough (see [Equation 5.1](#)), the back-

ward pair of the lead vehicle sends out a **STOM** message indicating that the gap is large enough for the lead vehicle to merge.

IV: MERGING DONE. After merging, the lead vehicle adapts its parameters to the right platoon and the new lead vehicle on the left lane can start with the *A2B* pairing.

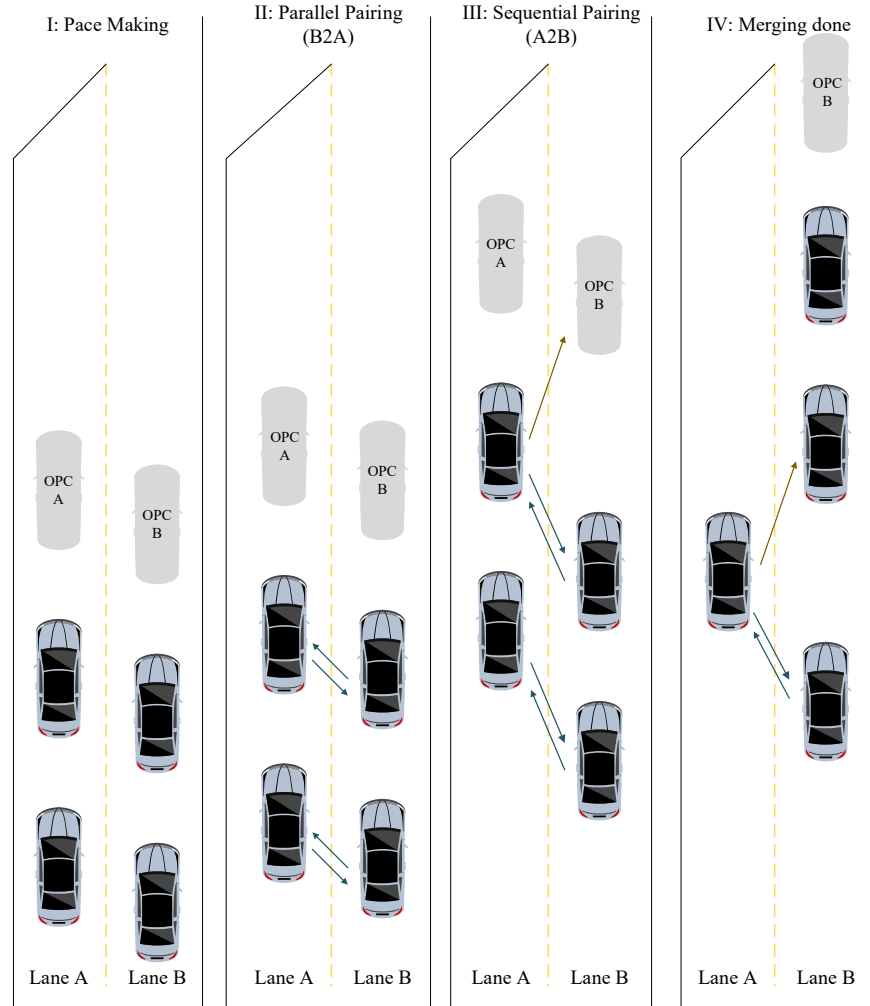


Figure 6.1: Phases of the **GDC** highway scenario (reproduced from [19, p. 8]).

A heavy duty vehicle in front of the ego vehicle may cause a lack of communication with the vehicles in front of the truck. A research describing message forward algorithms for trucks is shown in [48]. During the competition, one case has been experienced, where the communication to all vehicles in front of two trucks on both lanes was blocked. The only available information about the vehicles in front is the **MIO** information transmitted by the trucks. With this information, the system of the vehicle has the knowledge that another vehicle is in front of its **MIO** with a certain distance. Other information, such as the preceding vehicle of the **MIO** of the truck cannot be gathered and thus

an increased situation awareness is necessary for a correct evaluation of the situation.

6.2 ACCOMPLISHMENT

The highway scenario of the **GCDC** is the focus of this thesis for generating a **TI** and thus making a robust system by considering the **TI** in decision making. The competition took place in Helmond, the Netherlands, in May 2016. The data gathered during this competition has been logged according to the requirements of the competition shown in [50] and with the use of *lcm-log*. The generated **LCM** logs can be played with the *lcm-logplayer*. Moreover, the logs have been created for each module individually as a separate file containing only the received or transmitted data. For that reason, one is able to run the log of the **TS** containing all information the **TS** received. Figure 6.2 shows the track on Google Maps¹ of the highway A270 between Eindhoven and Helmond that has been used for the highway scenario of the **GCDC**.

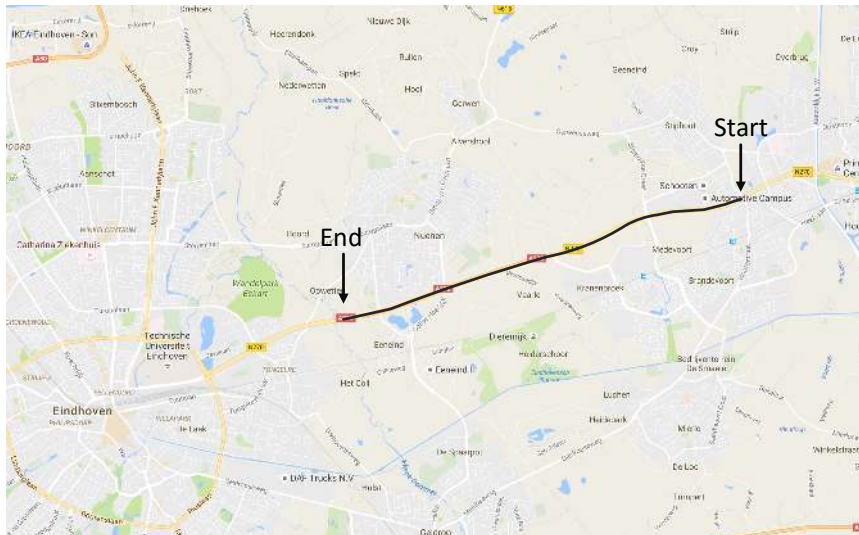


Figure 6.2: Map where the highway scenario took place (adapted from Google Maps¹).

The algorithm of the **TS**, as well as the methods of the other modules depicted in Figure 5.8, can be changed and afterwards fed with the same data from the competition, due to the use of **LCM**. Thus, the results presented in this thesis contain real data of vehicles driving in a platoon and interacting with each other. This approach offers the modification and verification of the implemented algorithms with the communication and interaction data from the competition.

The use of logged data and not simulated data for evaluating the system has been chosen for several reasons. First, the competition

¹ <https://www.google.at/maps/>

data offers a big variety of different devices with different accuracies or functionalities. Second, the implemented control system of the vehicle is unique for each vehicle and thus closer to reality. Third, the sensor data is already noisy and does not need to be manipulated as within a simulation. Additionally, the recorded data also includes the V_2V communication delay due to reflections of the signal.

6.3 RESULTS

The results of this thesis are shown as graphs illustrating the behaviour of the TI during different heats and different placements of the vehicle within the two platoons. The partial TIs are depicted in the same figure to show the composition and highlight the influences of the partial TIs . The generation of the TI has been simplified due to the limited performance of the system of the competition car. For reasons of presentability, the TIs have been filtered with a moving average filter from the previous twelve seconds.

6.3.1 Left Lane

The vehicles starting in the left lane have the task to acknowledge the backward partner during the parallel pairing. As soon as the vehicle is in the first place and the forward partner has been identified, the vehicle has to create a gap with respect to this vehicle. When receiving the $STOM$ message from the backward pair, the gap is large enough and the vehicle can perform the merging. After the merge, the vehicle has to adapt to the new platoon and set its new MIO .

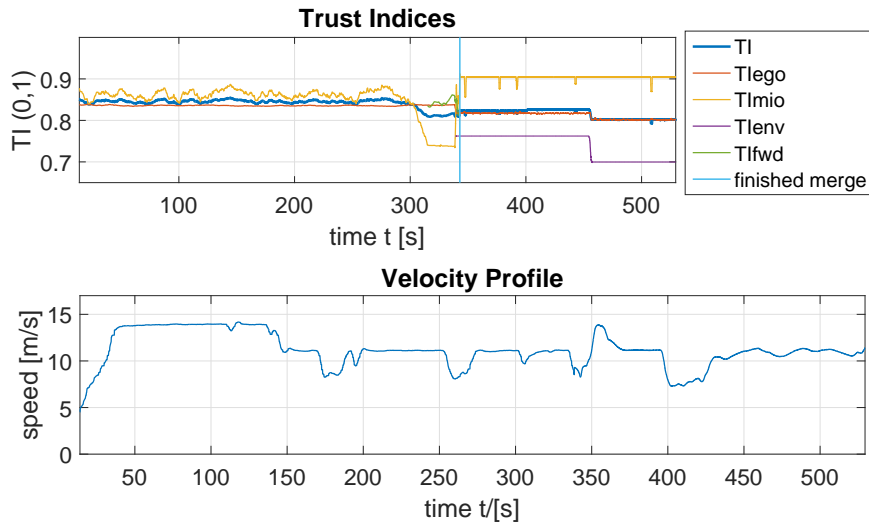
Figure 6.3 illustrates the TI and the speed profile of the test vehicle during a low-speed heat. The speed profile is shown for the sake of completeness and omitted in further figures. The behaviour of the TI_{ego} is stable with a value at around 0.84 until the vehicle merges. The TI_{ego} decreases as soon as the TS has more information about the accuracy of the geographic position, which is shown by the appearance of the TI_{env} that considers also the $PDOP$ value gathered from the GPS device. The late appearance of the $PDOP$ value is caused by the GPS device that did not provide this information before.

Comparing the TI_{mio} before and after the merge shows that there are different vehicles in front of the ego vehicle. It also indicates that the MIO after the merge is more reliable compared to the preceding vehicle before the merge. The vertical blue line indicates the moment when the merge to the right lane has been finished. It can be also seen that the TI_{mio} is decreasing while the merge is being performed because the system cannot detect the correct MIO during that phase and thus the radar match is also negative.

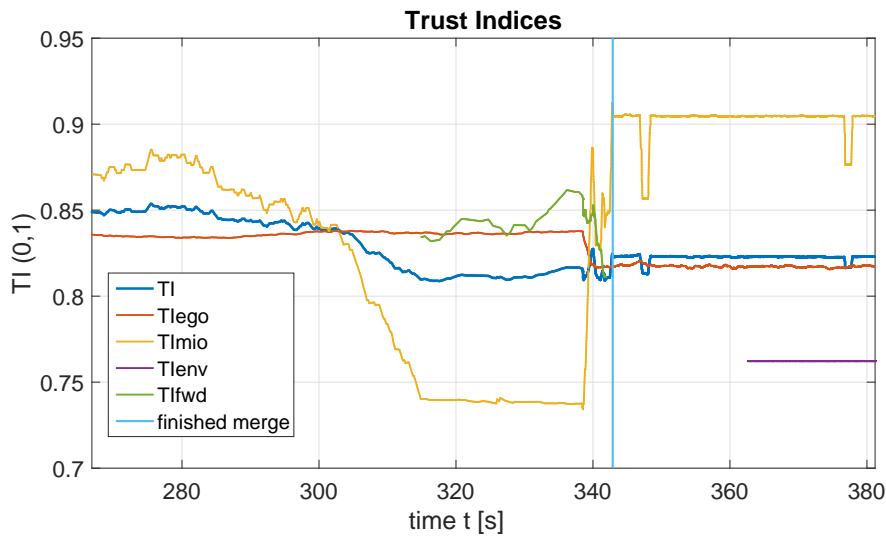
The forward partner is only set within the B_2A and merging phase. TI_{fwd} describes the trust in the forward partner and is for that reason only calculated within these phases.

The appearance of the TI_{env} after the merge in Figure 6.3a shows

the influence of knowledge about the environment on the TI . The TI_{env} is calculated by using the current PDOP value or an experienced TI from the area close to the vehicle.



(a) TI and speed profile.



(b) TI during the merge.

Figure 6.3: Experiment I: Highway scenario starting from the left lane.

A detailed graph of the TI during the merge can be taken from [Figure 6.3b](#). It shows the TI_{fwd} in more detail and also illustrates the behaviour of the TI_{mio} during the merge.

The decreased TI during the merge can be used to tell the decision making module that it has to increase its situation awareness. However, the lane change has to be treated as a special case, since the car has to keep its speed in order not to interfere with the interaction protocol. Furthermore, the trust in the new preceding vehicle is sta-

ble with a value of around 0.9. This information can be used by the decision making algorithm to decrease the time headway or the distance to the preceding vehicle for the reason that this vehicle provides highly reliable data.

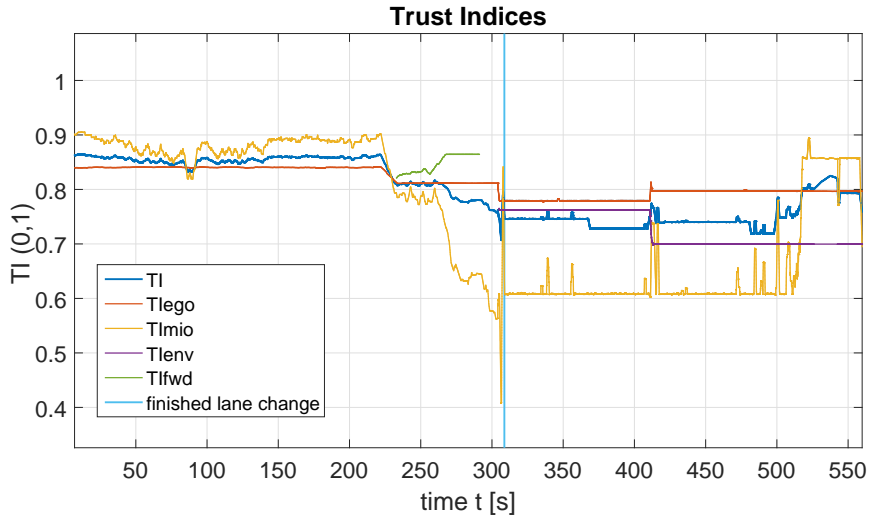
6.3.2 Right Lane

In the highway scenario, when the vehicle starts in the right lane, it does not need to change the lane. It creates a gap with respect to its forward partner on the left lane and sends out a **STOM** message as soon as the gap is large enough for the forward partner to merge.

Figure 6.4 depicts the composition and behaviour of the **TI** and the TI_{mio} . The TI_{ego} shown in **Figure 6.4a** is also stable and changing when the **PDOP** value occurs. The behaviour of the TI_{fwd} behaves similarly to the one presented in **Figure 6.3a**. First, the TI_{fwd} is lower and increases with the experience gathered about the vehicle. It shows also the same behaviour during the merge. As soon as the vehicle, in this case the forward partner, changes the lane, the TI_{mio} is decreasing significantly due to the change of the ego vehicle's preceding car. Further, the new **MIO** has a lower **TI** compared to the previous one. This is caused by a continuous misidentification of the **MIO**. Due to this misidentification, the observed values with the radar and the reported data via **V2V** communication does not match and the **TI** decreases. The reason for this continuous misidentification is a low accuracy of the received geographical position. It has been experienced that the new preceding vehicle has provided delayed position data and thus the system could not identify it correctly.

Figure 6.4b depicts the partial **TIs** used to calculate the TI_{mio} . For a better illustration of the **TIs**, the same moving average filter has been applied. This filter also causes the effect that the **TI** of the radar is not binary. The forward partner is getting the new **MIO** in the highway scenario. For the reason that the TI_{fwd} only uses the non-linear model for calculating this index, the TI_{mio} does not necessarily match with the TI_{fwd} . The behaviour of the TI_{mio} is strongly influenced by the result of the radar match. During the merge and afterwards, the measured radar distance differs from the distance using the geographical position. The evaluation of the radar match is performed within the linear model of the system of the car as described in **Section 3.1**.

The identification of measurement mismatches by comparing the own sensor data of the ego vehicle with the reported data is important for situation awareness. The implemented controller did not have any problems with following this vehicle in a platoon because the system relies, for safety reasons, on the radar information. The decreased TI_{mio} tells the other modules of the system that the reported data can not be fully trusted.



(a) Behaviour of the TI

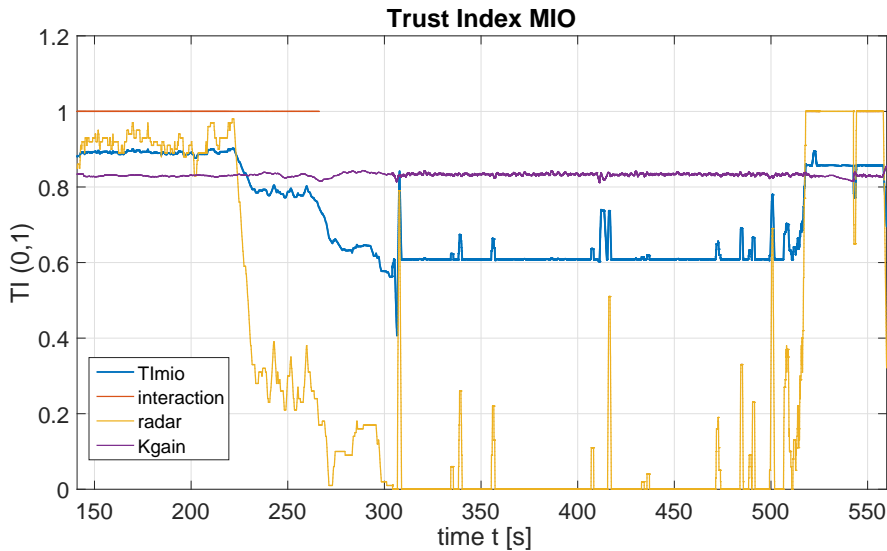
(b) Composition of TI_{mio} .

Figure 6.4: Experiment II: Highway scenario starting from right lane.

The graph shows that the trust in the new MIO after the merge is about 0.6. Comparing the overall TI of the system highlights that the situation awareness has to be increased. A possible reaction to this behaviour can be an increased time headway or distance to the preceding vehicle.

6.3.3 Unreliable Geographical Position

The proposed system has to rely to a certain extent on the geographical position provided by the other vehicles because the system is only able to verify the position of the preceding vehicle and the non-linear model is not able to improve the position of highly inaccurate posi-

tion data or unreliable data. One case was experienced in the course of the [GCDC](#) where the intended forward partner provided unreliable position information.

The introduced [TS](#) is able to improve the geographical position of a vehicle by considering the inertial sensor information, but it cannot improve the position of highly inaccurate measurements that provide changes in the position from ± 80 metres. [Figure 6.5](#) illustrates such a situation. The position of the competition car is plotted in blue as a reference to the correct position. The red curve shows the position of the forward partner.

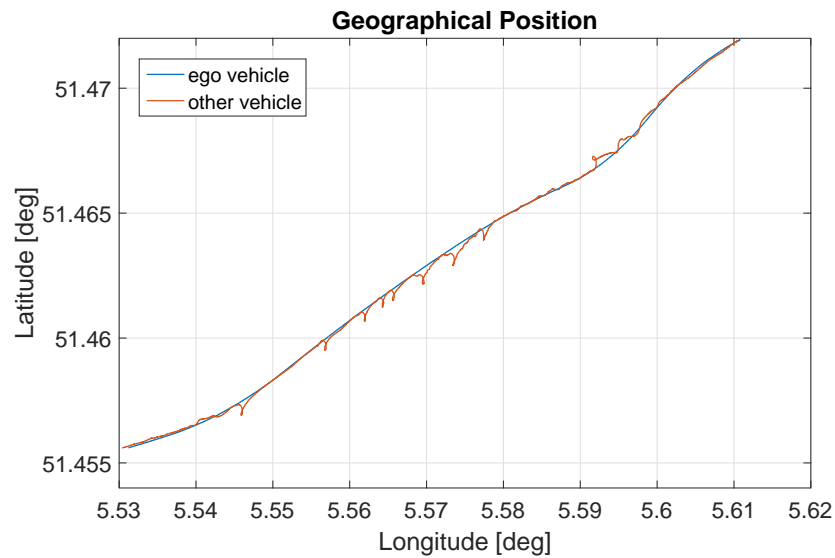


Figure 6.5: Unreliable/Inaccurate geographical position

With the current setup of the proposed [TS](#), vehicles that provide unreliable position, cannot be correctly identified. For computational reasons only vehicles that behave accurately are monitored, i.e. vehicles with large fluctuations in position can not be followed with the use of the linear model.

6.3.4 Comparison of the Behaviour Identifier

[Section 4.1.3](#) describes three approaches that can be used to detect speed fluctuation of the [MIO](#). These three approaches, the distance, time headway, and the speed, have been statistically evaluated using the *MioStatistics* class described in [Section 5.1.2](#).

[Figure 6.6](#) shows the standard deviation of the monitored measurements and the velocity of the observed vehicle. It is shown that the monitoring of the distance to the preceding vehicle is not a good measure to identify the speed fluctuation of the [MIO](#). As pointed out in [Section 4.1.3](#), the variation of the time headway is not significant enough, for the reason that the controller of the competition car reacts

to the speed changes immediately in case that the safety distance is violated.

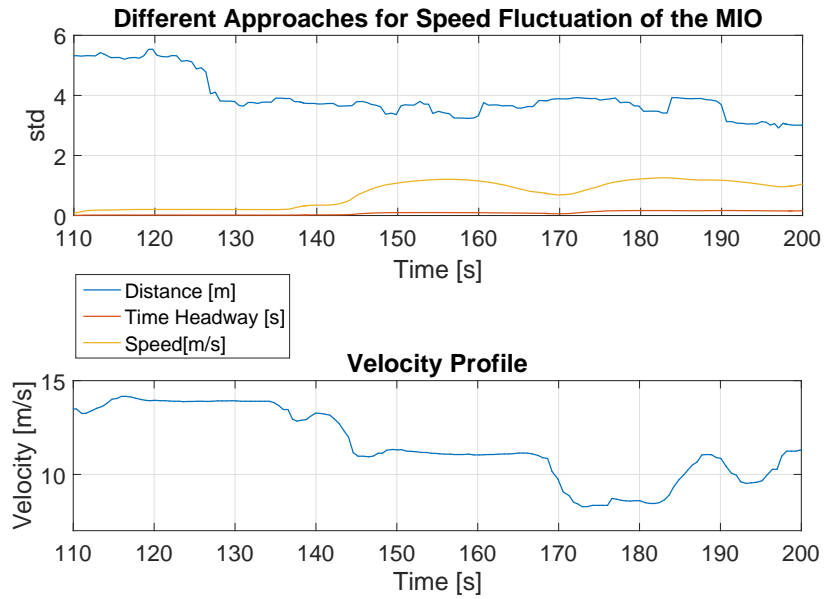


Figure 6.6: Approaches to detect speed fluctuation.

The speed of the MIO has been monitored by using the speed of the competition car and the rate of change of the distance to the preceding vehicle. This measurement does not rely on the V2V communication, it uses the competition car's sensors. The graph in Figure 6.6 illustrates that the standard deviation is low at the beginning and increases with the decrease of the speed. The behaviour around second 170 shows that the standard deviation decreases until second 170 due to a constant speed, but increases again, due to an additional increase of the speed.

CONCLUSION AND FUTURE WORK

Automated situation awareness is a fundamental functionality that enables decision making for automated and cooperative vehicles. Unreliable V2V communication makes it necessary to evaluate the provided information according to integrity, authenticity, accuracy of the received data, and reliability of the information. Current state-of-the-art researches focus on the establishment of trust between road users to provide integrity of the data, and authenticity. This thesis focuses on a model for evaluating the reliability of the provided data received via V2V communication from other vehicles. The introduced TS perceives its environment and generates a TI indicating the systems' level of trust in the sensor readings and their reliability at any given time instant. The TI is within a range of 0 and 1, and takes several factors into account. These factors are the environment itself, the ego vehicle, and the other vehicles, especially the preceding vehicle.

The evaluation of the sensor accuracy of the ego vehicle and the other vehicles is performed with a linear and non-linear model. The linear model describes the relation between the distance measured with a radar and the distance based on the provided geographical position. A kinematic model of the vehicle in combination with an EKF has been implemented in the non-linear model. The TS generates the TI based on various factors that can influence the situation.

The results of the TS are illustrated and discussed. The behaviour of the TI in various situations shows the correct identification of situations, where the preceding vehicle has a lower reliability. A discussion about the possible influences of the proposed TS in decision making is also given.

This thesis investigates *how to design a framework for a cooperative and automated vehicle that can perform more robust decisions based on trust and awareness*. The thesis describes the identified factors that influence a vehicle's decision making. These factors have been combined in order to provide a TI that can be used to support the vehicle's decision making system.

The first research question this thesis attempts to answer: *Can AI be used to create trust between vehicles based on their current and historical performance?* has been investigated in this thesis. It is shown how an EKF, of the category *Probabilistic Reasoning over Time*, can be used for estimating the reliability of the other vehicle's measurements.

The second research question: *Can trust improve the situation awareness in order to perform more robust decisions?* is only answered to a certain extent in this thesis. The introduced **TS** and its **TI** estimate the trust in the current situation and the results show that the **TI** reacts properly to situation changes.

7.1 FUTURE WORK

The results based on the **GCDC** 2016 highway scenario show that the **TS** correctly reacts to situation changes. The factors used to evaluate the current situation can be extended by adding more different types of sensors in order to increase the perception of the ego vehicle. An increased perception of the ego vehicle enables the **TS** to verify the received data of the surrounding vehicles in more ways in order to generate a more precise **TI**. The ego vehicle used to evaluate the proposed system only provided access to the measured radar distance to the vehicle in front, since the system needs two sources of information (**V2V** and e.g. radar) to verify the accuracy, the system was not able to estimate a **TI** of the surrounding vehicles.

Future research should include experiments with an automated vehicle that has a higher level of automation and more proximity sensors. This would give a more complete **TI** of the situation awareness and it would also give an indication of the computational complexity of the **TS**.

This thesis presents the behaviour of the **TI** and the partial **TIs**. It highlights that the importance of each of the partial **TIs** changes in various situations, e.g. during the merging. To adapt to such situations, a neural network can be used to adapt the weights for each **TI** dynamically. A neural network that has the partial **TIs** as input can be trained to weigh the inputs depending on the different situations.

Overall, the weighing of the partial **TIs** has to be further investigated. Performing more tests in an environment with cooperative and automated vehicles combined with an enhanced decision making algorithm can lead to a better insight on how the **TIs** have to affect the vehicle's decision making.

LINEARISED KALMAN FILTER EQUATIONS

These equations are completing the [EKF](#) described in [Section 3.2 \[13\]](#).

PROCESS JACOBIAN MATRIX A

$$A = \begin{bmatrix} 1 & 0 & A_{1,3} & A_{1,4} & A_{1,5} & A_{1,6} \\ 0 & 1 & A_{2,3} & A_{2,4} & A_{2,5} & A_{2,6} \\ 0 & 0 & 1 & 0 & \text{sign}(v)T & 0 \\ 0 & 0 & 0 & 1 & 0 & T \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{A.1})$$

$$A_{1,3} = -\sin(\varphi)(vT + a\frac{T^2}{2}) - \cos(\varphi)(v\dot{\varphi}\frac{T^2}{2} + 2a\dot{\varphi}\frac{T^3}{6}) \quad (\text{A.2a})$$

$$A_{1,4} = \cos(\varphi)T - \sin(\varphi)(\dot{\varphi}\frac{T^2}{2}) \quad (\text{A.2b})$$

$$A_{1,5} = -\sin(\varphi)(v\frac{T^2}{2} + 2a\frac{T^3}{6}) \quad (\text{A.2c})$$

$$A_{1,6} = \cos(\varphi)\frac{T^2}{2} - \sin(\varphi)(2\dot{\varphi}\frac{T^3}{6}) \quad (\text{A.2d})$$

$$A_{2,3} = \cos(\varphi)(vT + a\frac{T^2}{2}) - \sin(\varphi)(v\dot{\varphi}\frac{T^2}{2} + 2a\dot{\varphi}\frac{T^3}{6}) \quad (\text{A.2e})$$

$$A_{2,4} = \sin(\varphi)T + \cos(\varphi)(\dot{\varphi}\frac{T^2}{2}) \quad (\text{A.2f})$$

$$A_{2,5} = \cos(\varphi)(v\frac{T^2}{2} + 2a\frac{T^3}{6}) \quad (\text{A.2g})$$

$$A_{2,6} = \sin(\varphi)\frac{T^2}{2} + \cos(\varphi)(2\dot{\varphi}\frac{T^3}{6}) \quad (\text{A.2h})$$

MEASUREMENT JACOBIAN MATRIX H

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{A.3})$$

PROCESS NOISE COVARIANCE MATRIX Q

$$Q = \begin{bmatrix} 0.1^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.1^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & (10\frac{\pi}{180})^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 10^{-4} & 0 & 0 \\ 0 & 0 & 0 & 0 & (0.1\frac{\pi}{180})^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{A.4})$$

B.1 STATE MACHINE

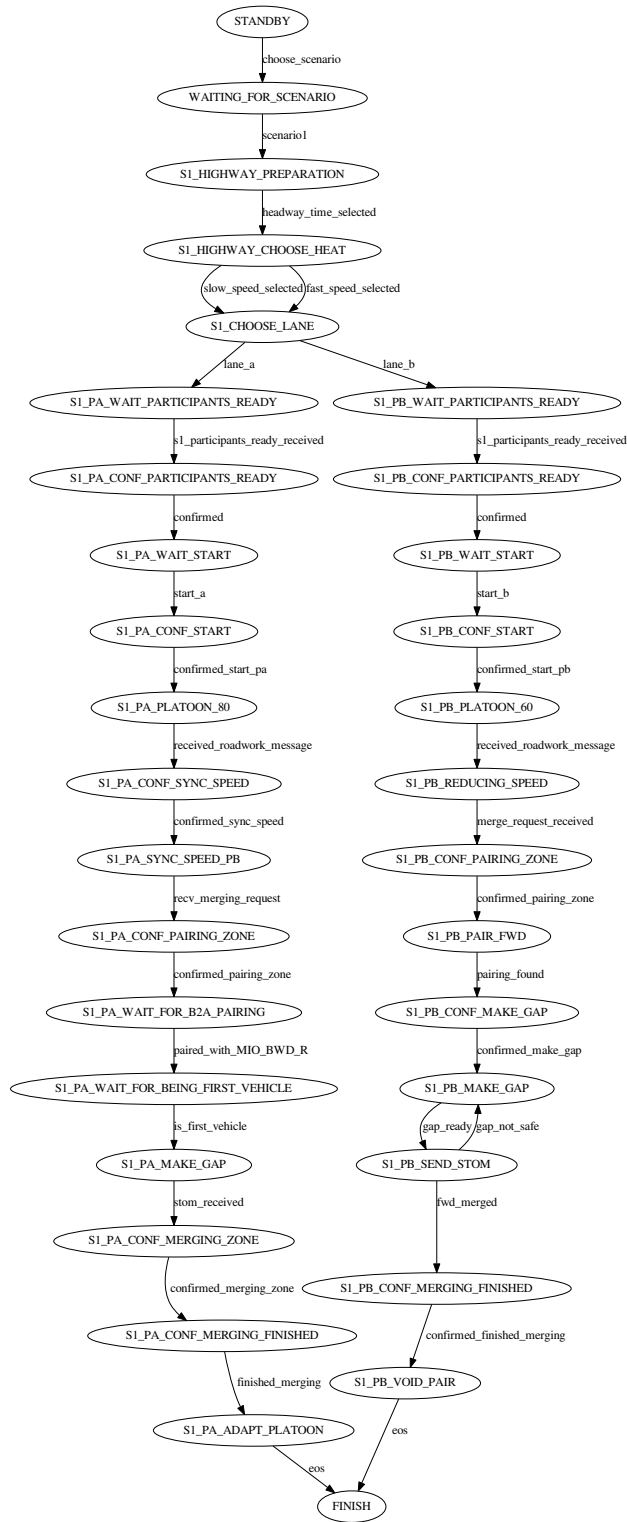


Figure B.3: State machine for the GCDC scenario 1.

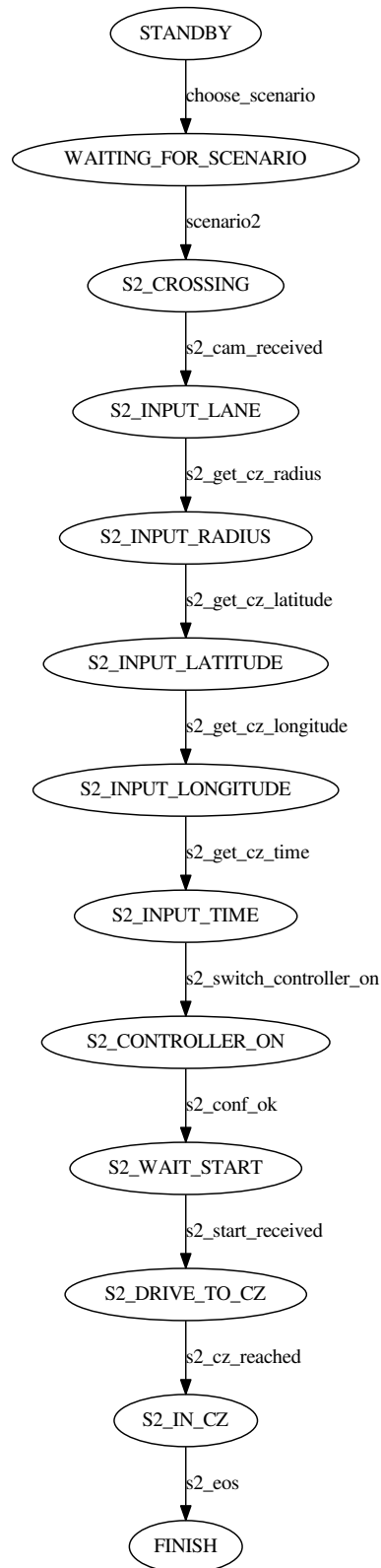


Figure B.4: State machine for the **GCDC** scenario 2.

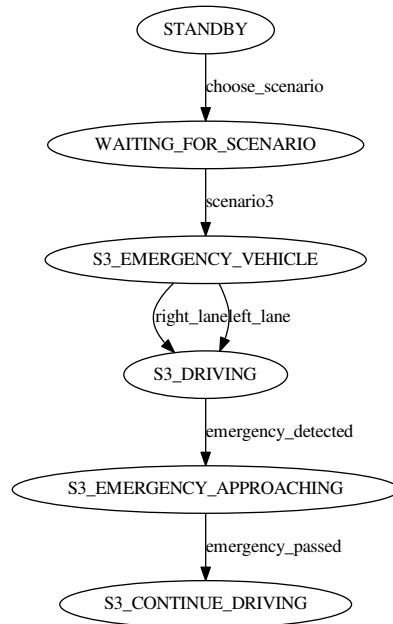


Figure B.5: State machine for the GCD scenario 3.

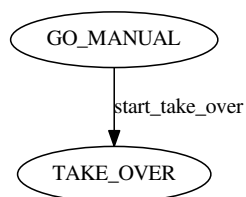


Figure B.6: State machine for going back to manual mode.

BIBLIOGRAPHY

- [1] L. B. Torvalds, "Desktop Market has already started, Preston St. Pierre of Linux Times interviews Linus Torvalds." An online version is available at <http://web.archive.org/web/20050404020308/http://www.linuxtimes.net/modules.php?name=News&file=article&sid=145> (18.06.2016), 2004.
- [2] World Health Organization (WHO), *Global Status Report on Road Safety 2015*. Geneva: WHO Press, 2015.
- [3] A. Davila and M. Nombela, "Platooning - safe and eco-friendly mobility.," *SAE Technical Paper 2012-01-0488*, 2012.
- [4] R. N. Charette, "This Car Runs on Code." An online version is available at <http://spectrum.ieee.org/transportation/systems/this-car-runs-on-code> (18.06.2016), 2009.
- [5] T. Bijlsma, S. de Kievit, J. van de Sluis, E. van Nunen, I. Passchier, and E. Luijck, *Security Challenges for Cooperative and Interconnected Mobility Systems*, pp. 1–15. Cham: Springer International Publishing, 2013.
- [6] S.-W. Kim, Z. J. Chong, B. Qin, X. Shen, Z. Cheng, W. Liu, and M. H. Ang, "Cooperative perception for autonomous vehicle control on the road: Motivation and experimental results," *IEEE International Conference on Intelligent Robots and Systems*, pp. 5059–5066, 2013.
- [7] S. W. Kim, B. Qin, Z. J. Chong, X. Shen, W. Liu, M. H. Ang, E. Frazzoli, and D. Rus, "Multivehicle Cooperative Driving Using Cooperative Perception: Design and Experimental Validation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 2, pp. 663–680, 2015.
- [8] J. Ploeg, B. T. Scheepers, E. van Nunen, N. van de Wouw, and H. Nijmeijer, "Design and experimental evaluation of cooperative adaptive cruise control," *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, no. June 2016, pp. 260–265, 2011.
- [9] E. Semsar-Kazerooni, A. Morales Medina, and H. H. Bengtsson, "DEL150330 i-GAME D2.1 Interaction Protocol," tech. rep., i-game gcdc, 2015.

- [10] C. Englund, L. Chen, J. Ploeg, E. Semsar-Kazerooni, A. Voronov, H. H. Bengtsson, and J. Didoff, "The grand cooperative driving challenge 2016: boosting the introduction of cooperative automated vehicles," *IEEE Wireless Communications*, vol. 23, pp. 146–152, August 2016.
- [11] P. Wex, J. Breuer, A. Held, T. Leinmuller, and L. Delgrossi, "Trust Issues for Vehicular Ad Hoc Networks," *VTC Spring 2008 - IEEE Vehicular Technology Conference*, pp. 2800–2804, 2008.
- [12] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. Pearson Education, 2 ed., 2003.
- [13] N. Magnusson and T. Odenman, "Improving absolute position estimates of an automotive vehicle using GPS in sensor fusion," Master's thesis, Chalmers University of Technology, 2012.
- [14] J. Zhang, "A survey on trust management for VANETs," *Proceedings - International Conference on Advanced Information Networking and Applications, AINA*, pp. 105–112, 2011.
- [15] M. Röckl, J. Gacnik, and J. Schomerus, "Integration of Car-2-Car Communication as a Virtual Sensor in Automotive Sensor Fusion for Advanced Driver Assistance Systems," in *Proceedings. FISITA*, no. 1, (Munich, Germany), pp. 2008–2008, Springer Automotive Media, 2008.
- [16] ETSI (European Telecommunications Standards Institute), "Vehicular Communications ; Basic Set of Applications ; Part 2 : Specification of Cooperative," *ETSI TS 102 637-2*, vol. 1, pp. 1–18, 2011.
- [17] ETSI (European Telecommunications Standards Institute), "Intelligent Transport Systems (ITS); Users and applications requirements; Part 2: Applications and facilities layer common data dictionary," *ETSI TS 102 894-2*, vol. 1, pp. 1–78, 2013.
- [18] ETSI (European Telecommunications Standards Institute), "Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 3: Specifications of Decentralized Environmental Notification Basic Service," *ETSI TS 102 637-3*, vol. 1, no. 1, pp. 637–3, 2010.
- [19] J. van de Sluis, L. Chen, and L. Garcia-Sol, "DEL_i-GAME_D3.2 Proposal for extended message set for supervised automated driving," tech. rep., i-game gcdc, 2015.
- [20] D. L. Hall and J. Llinas, "An introduction to multisensor data fusion," *Proceedings of the IEEE*, vol. 85, no. 1, pp. 6–23, 1997.
- [21] F. Gustafsson, *Statistical Sensor Fusion*. Studentlitteratur, 2 ed., 2010.

- [22] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Journal of Basic Engineering*, vol. 82, no. 1, p. 35, 1960.
- [23] G. Welch and G. Bishop, "An Introduction to the Kalman Filter," *In Practice*, vol. 7, no. 1, pp. 1–16, 2006.
- [24] R. Faragher, "Understanding the basis of the kalman filter via a simple and intuitive derivation [lecture notes]," *IEEE Signal Processing Magazine*, vol. 29, no. 5, pp. 128–132, 2012.
- [25] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.
- [26] F. Gustafsson, "Particle Filter Theory and Practice with Positioning Applications," *IEEE A&E Systems Magazine*, vol. 25, no. 7, pp. 53–81, 2010.
- [27] D. Miller, A. Sun, and W. Ju, "Situation Awareness with Different Levels of Automation," *2014 IEEE International Conference on Systems, Man, and Cybernetics*, pp. 688–693, 2014.
- [28] E. Schoch, F. Kargl, and M. Weber, "Communication patterns in vanets," *IEEE Communications Magazine*, vol. 46, pp. 119–125, November 2008.
- [29] F. Kargl, P. Papadimitratos, L. Buttyan, M. Müter, E. Schoch, B. Wiedersheim, T. V. Thong, G. Calandriello, A. Held, A. Kung, and J. P. Hubaux, "Secure vehicular communication systems: implementation, performance, and research challenges," *IEEE Communications Magazine*, vol. 46, pp. 110–118, November 2008.
- [30] H. Vasudevan, A. R. Joshi, N. M. Shekokar, N. J. Patel, and R. H. Jhaveri, "International conference on advanced computing technologies and applications (icacta) trust based approaches for secure routing in vanet: A survey," *Procedia Computer Science*, vol. 45, pp. 592 – 601, 2015.
- [31] International Telecommunication Union: Telecom Standardization Sector, "International Standard ISO/IEC 9594-8 Recommendation ITU-T X.509," 2012.
- [32] S. Soni, "Reliable Trust Computation model in Vehicular ad-hoc Network," *American Journal of Advanced Computing*, vol. II, no. 1, pp. 7–11, 2015.
- [33] X. Hong, D. Huang, M. Gerla, and Z. Cao, "SAT: Situation-aware trust architecture for vehicular networks," *IEEE Communications Magazine*, no. November, pp. 31–36, 2008.

- [34] A. Tajeddine, A. Kayssi, and A. Chehab, "A privacy-preserving trust model for vanets," in *Computer and Information Technology (CIT), 2010 IEEE 10th International Conference on*, pp. 832–837, June 2010.
- [35] H. Aras, C. Beckstein, S. Buchegger, P. Dittrich, T. Hubauer, F. Klan, B. Knig-Ries, and O. Wolfson, "Uncertainty and trust," in *Univ. of Illinois, Dagstuhl Seminar Proceedings*, vol. 8421, 2009.
- [36] W. T. L. Teacy, J. Patel, N. R. Jennings, and M. Luck, "TRAVOS: Trust and reputation in the context of inaccurate information sources," *Autonomous Agents and Multi-Agent Systems*, vol. 12, no. 2, pp. 183–198, 2006.
- [37] U. F. Minhas, J. Zhang, T. Tran, and R. Cohen, "Towards expanded trust management for agents in vehicular ad-hoc networks," *International Journal of Computational Intelligence Theory and Practice (IJCITP)*, vol. 5, no. 1, 2010.
- [38] M. Raya, P. Papadimitratos, V. D. Gligor, and J. P. Hubaux, "On Data-Centric Trust Establishment in Ephemeral Ad Hoc Networks," *IEEE INFOCOM 2008*, pp. 1912–1920, 2008.
- [39] F. Dötzer, L. Fischer, and P. Magiera, "VARS: A vehicle ad-hoc network reputation system," *Proceedings - 6th IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks, WoWMoM 2005*, pp. 454–456, 2005.
- [40] F. Durso and A. Sethumadhavan, "Situation awareness: Understanding dynamic environments," *Human Factors: The Journal of the ...*, vol. 50, no. 3, pp. 442–448, 2008.
- [41] M. Kurmis, A. Andziulis, D. Dzemydiene, S. Jakovlev, M. Voznak, and G. Gricius, "Cooperative Context Data Acquisition and Dissemination for Situation Identification in Vehicular Communication Networks," *Wireless Personal Communications*, vol. 85, no. 1, pp. 49–62, 2015.
- [42] M. Röckl, *Cooperative Situation Awareness in Transportation*. PhD thesis, Leopold-Franzens-University Innsbruck, 2009.
- [43] ETSI (European Telecommunications Standards Institute), "Intelligent Transport Systems (ITS); Vehicular Communications; GeoNetworking; Part 3: Network architecture," *ETSI TS 102 636-3*, vol. 1, pp. 1–23, 2010.

- [44] IEEE, "Ieee standard for information technology – telecommunications and information exchange between systems – local and metropolitan area networks – specific requirements – part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications – amendment 6: Wireless access in vehicular environments," *IEEE Std 802.11p-2010 (Amendment to IEEE Std 802.11-2007 as amended by IEEE Std 802.11k-2008, IEEE Std 802.11r-2008, IEEE Std 802.11y-2008, IEEE Std 802.11n-2009, and IEEE Std 802.11w-2009)*, pp. 1–51, Jul 2010.
- [45] E. Kaplan and C. Hegarty, *Understanding GPS: Principles and Applications*. Artech House, 2005.
- [46] R. Yarlagadda, I. Ali, N. Al-Dhahir, and J. Hershey, "Gps gdop metric," *IEE Proceedings - Radar, Sonar and Navigation*, vol. 147, pp. 259–264, Oct 2000.
- [47] L. Danielsson, *Tracking and radar sensor modelling for automotive safety systems*. PhD thesis, Chalmers University of Technology, 2010.
- [48] M. Larsson, *Methods to Improve V2V Communications in Platoons of Heavy Duty Vehicles*. No. 17, Lund: Halmstad University Press, 2016.
- [49] R. B. Langley, "Dilution of Precision," *GPS World*, vol. 10, no. May, pp. 52–59, 1999.
- [50] H. Salunkhe, H. H. van Huysduynen, B. Nijssen, and J. Terken, "DEL_i-GAME_D7.1 GCDC judging criteria and their evaluation (Final)," tech. rep., i-game gcdc, 2016.
- [51] B. S. Bourgeois, "Using Range and Range Rate for Relative Navigation," tech. rep., Naval Research Laboratory Marine, Mississippi, 2007.

COLOPHON

This document was typeset using the typographical look-and-feel `classicthesis` developed by André Miede. The style was inspired by Robert Bringhurst's seminal book on typography "*The Elements of Typographic Style*". `classicthesis` is available for both \LaTeX and \LyX :

<http://code.google.com/p/classicthesis/>

Happy users of `classicthesis` usually send a real postcard to the author, a collection of postcards received so far is featured here:

<http://postcards.miede.de/>

Final Version as of September 18, 2016 (`classicthesis`).



PO Box 823, SE-301 18 Halmstad
Phone: +35 46 16 71 00
E-mail: registrator@hh.se
www.hh.se