Finding uncertainty of sensor fusion in automotive driving

Authors:
Schadrack Kwizera
Jude Dilshan Jayasuriya

Supervisor: Sven Nordebo

Examiner: Sven-Erik Sandström

Date: 2022-May

Course code: 5ED36E, 30 hp

Subject: Electrical Engineering

Level: Master

Department of Physics and
Electrical Engineering
Faculty of Technology

Abstract

Human error has been the most common cause of car accidents. Advances in sensing and data fusion have made recent progress in autonomous vehicles that will increase the potential of drastically improving safety, efficiency, and cost of transportation.

In this thesis, we present an overview of finding the error probability of sensor fusion in automotive driving, and we will investigate the collision probabilities in automated vehicles.

In our study, we simulate automated driving systems in a virtual environment using real-world maps using MATLAB Automated Driving Toolbox, Simulink, and Roadrunner.

During the study, we will investigate different scenarios such as weather conditions, noise, lighting, and road conditions with an ‘ego-vehicle’ equipped with multiple sensors such as; lidar and vision sensors.

Contents

I. Introduction

1. History of autonomous vehicles

Historically automated vehicles have been in the robotics engineers' imagination for a long time. At the 1939 world's fair General Motors presented their vision of automated highways where cars would follow the road and maintain safe distances using automatic radio control in their Futurama exhibit [13, 11].

Though in recent car production, some autonomous vehicle features such as automated cruise control (ACC), lane keeping assist driver assistance (AFIL), adaptive lights (AFL), brake assist (BAS), brake assist systems warning of a vehicle in the blind spot (BLIS), and systems for monitoring driver fatigue (Driver Alert/Attention Assist) [3-5]. There are also systems supported by artificial intelligence, for example, for image analysis [5-9], but fully automated vehicles are not commercially available.

In recent years, self-driving cars have become among the actively discussed and researched topics due to the need for driving improvement in terms of safety, efficiency, and cost [3,4]. Within the complex driving environment caused by continuous increment of road congestion, the suggested solution is advance in sensing and data fusion.

Data collected from different sensors, with information like maps, are used to build models of the surrounding traffic scene and encode relevant aspects of the driving problem. These models allow the autonomous vehicle to plan how it will drive, optimizing comfort, safety, and progress towards its destination.

Lane-keeping system is an important technology used in autonomous vehicles in which vehicles can be run following the anticipant lane automatically based on the vehicle-mounted sensors [5]. To plan for the trajectory of the autonomous vehicle, we also require models of how other traffic participants are likely to move soon and all risks for the different potential autonomous vehicle trajectories.

2. Why autonomous vehicles

The autonomous vehicle has been a majors subject due to the different impacts that it will have on transportation in terms:

- Safety: Human error is estimated to cause at least 90% of vehicle accidents. As autonomous vehicles outperform human drivers in perception, decision-making and execution, adopting them may reduce or eliminate car accidents [6].

- Service: Autonomous vehicles will be an accurate solution for individuals unable/prohibited from driving for various reasons. And in countries with a high percentage of senior citizens like Japan and Italy, the need for bus drivers, truck drivers and taxi drivers is high. The autonomous vehicle will resolve that shortage in transportation at a low cost [6-8].

- Platooning: with autonomous vehicles Platooning will increase throughput through large vehicle density, advantages of platooning include higher density and the reduction of energy consumption. Because of reduced air drag due to the small distance between vehicles, empirical data from California shows that human drivers need about a 1.63-second gap between vehicles so, at the speed of 100km/h, this leaves around 11% of longitudinal length utilization and by using an autonomous vehicle that can decrease [6].

3. Classification of autonomous vehicle

The classification of autonomous vehicles according to the degree of involvement of human support relative to the car's functioning. Table.1 presents the driving level according to the International Society of Automotive Engineers (ISAE).

Table. 1 Autonomous driving levels according to the ISAE classification [13].

Level	Description
0	The control of the vehicle belongs to the driver. Even if the car notifies about the hazards, The driver is responsible for monitoring the environment and must be ready to take control.
1	The vehicle has some support for aspects of driving, e.g., steering or acceleration/braking. The driver is responsible for monitoring the environment and must be ready to take control.
2	Partial automation of the vehicle - the use of the system for both driving and speed control; the driver is responsible for the supervision and implementation of the remaining driving elements. The driver is responsible for monitoring the environment and must be ready to take control.
3	Conditional automation of the vehicle. The possibility of taking over the car control on all aspects of driving, assuming that the driver must be ready at any time to take control of the car.
4	High level of automation in a vehicle, the car can take control of all aspects of driving, even if the human driver does not respond to the call to take control.
5	Full automation of the vehicle - independent driving under all conditions

4. Challenges, risks, and limitations

Though autonomous vehicles are not commercially available, many studies have shown multiple risks, fear, and challenges on the subject,

Below, we listed some risks and costs.

In terms of safety, as mentioned in [2,14]:

- Crashes would be inevitable. Fatality and injury rates would likely increase at least from the short to medium term in the testing period of self-driving cars.
- System failures and sudden breakdowns could be fatal to both vehicle occupants and other road users.

In terms of economy, as mentioned in [15,16]:

- The self-driving capability could be expensive, resulting in a disparity in socio-economic access.
- Car-sharing and car-pooling services are heavily skewed; in favour of a few groups of people in cities, mainly young, highly educated, tech-savvy, and affluent. That may continue in the era of autonomous transport.
- Prioritizing investment in autonomous transport could negatively affect investment in existing transport services, especially public transit. Low-income groups and non-car users would be affected as a result.

In terms of security and privacy, as mentioned in [14, 15]:

- Mass communication, surveillance and data sharing pose potential threats to the right to privacy and individual freedoms.
- Passenger information and activities took while onboard self-driving cars may need recording.
- Data generated by the various models would be saved and shared.
- Autonomous cars could become lethal weapons in the hands of terrorist organizations, computer hackers, disgruntled employers, and hostile nations.

In terms of employment, as mentioned in [8,13-15]:

- Reduction in employment in various sectors of the transport industry such as haulage, private vehicle hire, drivers and technicians are inevitable consequences of automated driving.
- In general, low-skilled, routine labour in the private sector like cashiers, cleaners and drivers would be the most negatively affected by automation as it will take over their work.

5. Methodology

As the purpose of our study is to simulate the derivation of uncertainty-related outcomes of an automatic driving system. We did a systematic review of various papers, reports, and articles available on the subject online using a set of keywords such as autonomous vehicles, autonomous transport, connected vehicles, driverless vehicles, self-driving vehicles, physically challenged, disabled or disability, safety, privacy, estimation, probability, security and so on, then we divided our study into five main phases below:

- ❖ Development of the automatic driving system
- ❖ Simulate the system in real-world tracks with actors and collect data on variables
- ❖ Generate the probability distribution
- ❖ Take sample data from the probability distribution
- ❖ Estimate the probability of critical scenarios

However, this is a mass-scale project which requires extensive time and resources. In practice, those phases would need a team with different competencies. That's why we decided to use some alternative ways that will simplify the process and yet covers the expectation of this study.

II. Development of automatic driving system

We used Matlab® Automatic Driving Toolbox® (ATD) for this purpose. Instead of developing algorithms and systems, we used in-built test benches of Matlab ATD. To narrow the scope, we focused on one common driving scenario, the highway lane changing.

1. Highway driving experiment through simulation

We studied the designing and simulation process of lane changing manoeuvre by which we went through the developing process of the virtual world as we intended to use them during the simulation, and we went through the designing process of a motion planner and controller. Finally, the last part was about running the simulation and testing. There are three categories of the core competencies of the autonomous driving system: perception, planning and control [17].

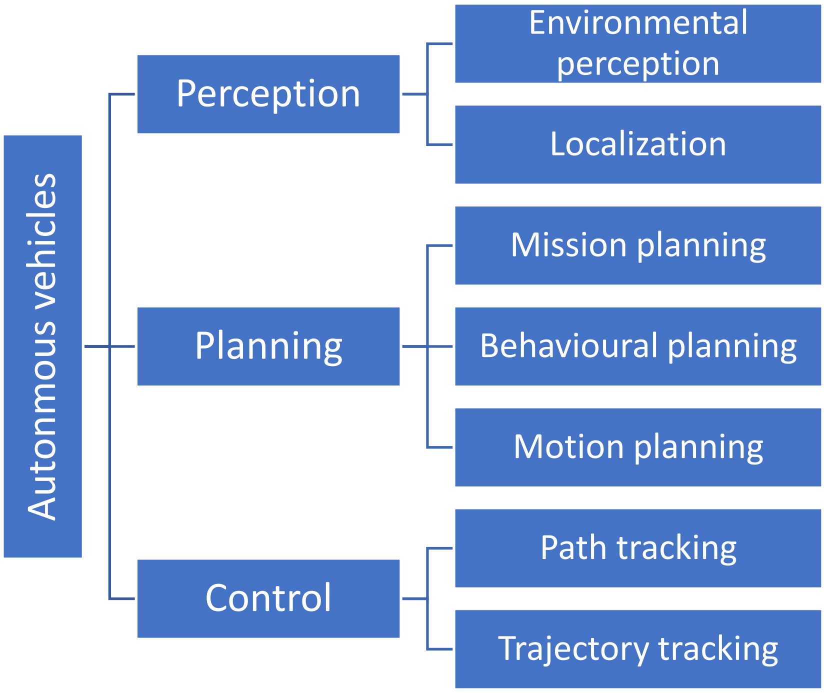


Figure 1: The core competencies of autonomous driving system [17].

1.1 Perception

Perception is the ability of an autonomous system to collect information and extract relevant knowledge from the environment.

Environmental perception is the ability of the vehicle to identify the position of obstacles, detect the signs and road markings, and categorize data by their semantic meaning.

Localization is the vehicle's ability to determine its position in the environment [17].

1.2 Planning

Typically, the vehicle should be able to move from a start location to a target destination in an optimized path by avoiding obstacles and assuring safety. The process of making decisions to achieve these goals is planning. In a broad sense, mission planning is the determination of the route between the start toward the destination. The vehicle takes a road considering the cost factors such as distance and traffic conditions.

Behavioural planning refers to the decision-making to ensure that vehicle achieves the mission planner's prescribed route by following road rules and safely interacting with other road users.

Motion planning is deciding on a sequence of actions over an incident like collision avoidance.

1.3 Control

Control is the ability of a vehicle to perform planned actions generated by the planning processes providing necessary inputs to the hardware [17]. Path following refers to following a predefined path which does not involve time as a constraint. The vehicle will reach the goal at whatever speed by following that path.

On the other hand, trajectory tracking involves time as a constraint. The vehicle has to be at a certain point at a particular time [18].

2. Selection of scenario

A scenario is a temporal sequence of scenes, whereby actions and events of the elements involved occur within this sequence [19]. Overtaking, vehicle cut-in and lane changing are examples of classes of scenarios.

There are three types of scenarios: complex, challenging, and critical [20]; most of the time, complex and challenging types cannot be separated [20]. However, critical scenarios are distinguished using matrices like time to collision. For our study, we selected a complex scenario type (lane changing), which may become a critical type of scenario once there is a collision.

Because safe lane changing requires correct behaviour prediction of actors other than ego vehicles, human-based driving and most current automated driving systems concentrate on the single-vehicle lane change with self-detective information. However, with the development of vehicle-to-vehicle (V2V) communication, it is possible to share information among multiple vehicles. Although cooperative lane changes are still a new area with more complicated scenarios, they will improve safety and lane-change efficiency [21]. For our study, we selected single-vehicle lane changing.

3. Highway lane change test bench

We used a test bench model developed by the MathWorks team to simulate an automated lane change manoeuvre system. This model involves; an automated driving toolbox, model predictive control toolbox, and navigation toolbox in MATLAB [22].

In this model, we can customize perception, planning and control competencies. This model uses five vision sensors and one radar sensor to detect other vehicles from the surrounding view of the ego vehicle. It uses a joint probabilistic data association (JPDA) based tracker to track the fused detections from these multiple sensors [22]; using these data, the lane change planner generates a feasible trajectory executed by the lane change controller. This test bench includes six major subsystems [23], as shown in Figure 2.

They are:

a. Scenario and environment
b. Lane change planner and planner configuration
c. Lane change controller
d. Vehicle dynamics
e. Metric assessment
f. Visualization

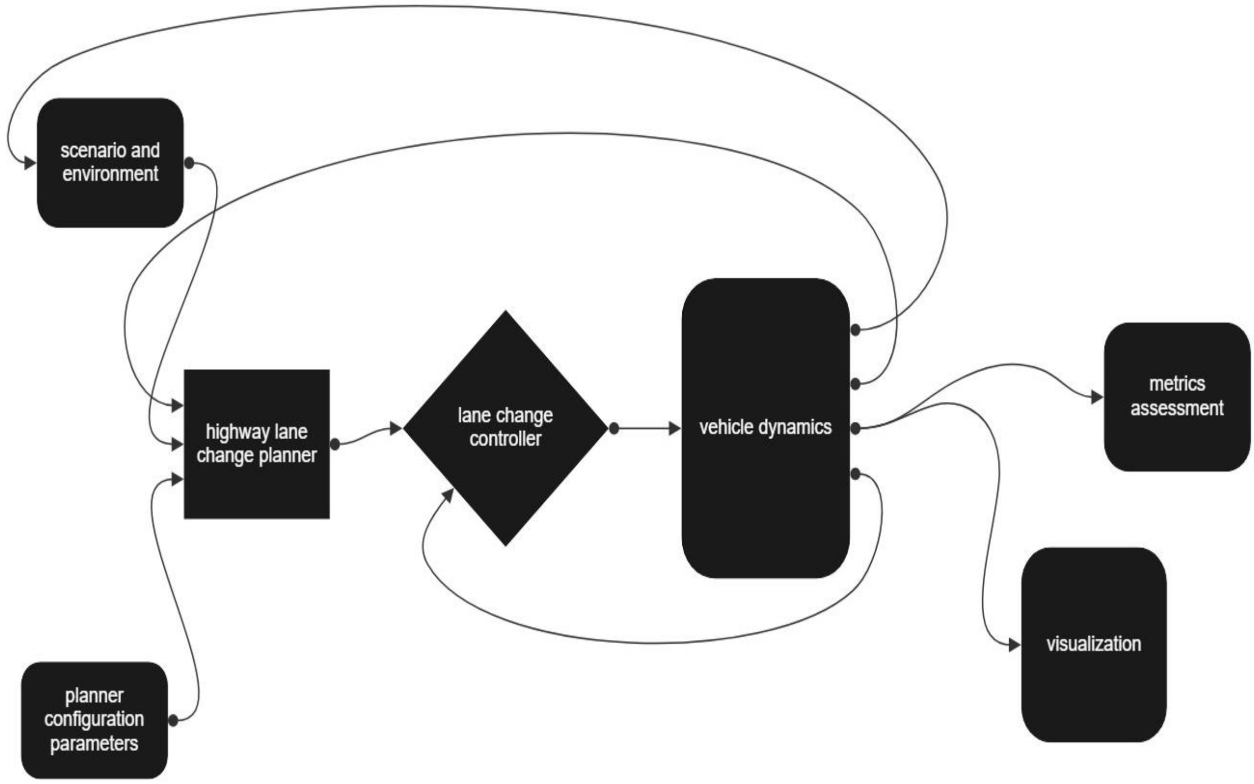


Figure 2: Test bench major subsystems [22].

❖ Scenario and environment subsystem

The scenario and environment subsystem read the map data from the base workspace and output information about lanes and reference paths [23]. This subsystem consists of scenario reader block and vehicle to world block. The scenario reader block reads in a driving scenario from the workspace.

This block takes in ego vehicle information to perform a closed loop simulation. It outputs ground truth about actors and lane boundaries in ego vehicle coordinates. The vehicle to world block converts target vehicle positions from vehicle coordinates to world coordinates.

❖ Highway lane change planner subsystem

The Highway Lane Change Planner Subsystem plays a role in this test model by determining the optimal path to change.

This subsystem checks for collisions and transforms the global coordinates to the Frenet coordinates.Also, it generates multiple possible trajectories that the ego vehicle can take.

Frenet Coordinates are an intuitive way of a road position representation than the traditional (x, y) Cartesian Coordinates. Frenet coordinates use variable ‘s’ and ‘d’ to describe the position of a vehicle in a reference path.

The ‘s’ coordinate represents the longitudinal displacement and is the distance along the path.

The ‘d’ coordinate represents the lateral displacement and is the side-to-side position on the road, as in figure 3.

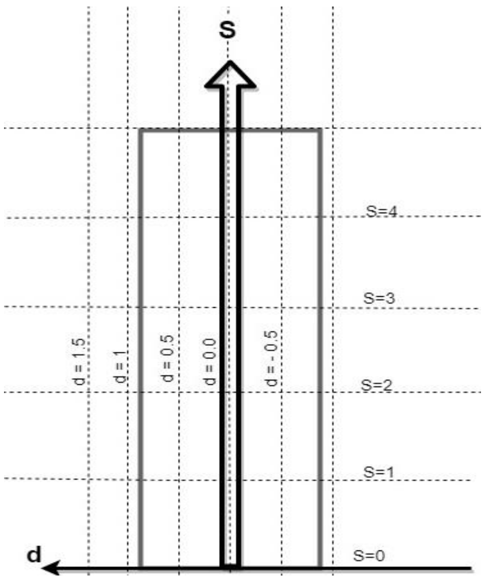


Figure 3: Representation of a reference path in Frenet coordinates on a road segment [22].

❖ planner block

Planner block evaluates cost values for all terminal states.

❖ Lane change controller

The lane change controller follows the reference trajectory selected by the highway lane change planner with a collision-free optimal trajectory as an input from the lane change planner subsystem.
An input to the controller is a reference point on the path, and another input is the longitudinal velocity that comes from the vehicle's dynamic subsystems.

The output from the controller feeds into the vehicle dynamics and makes this a closed loop system.
The lane change controller subsystem consists of a virtual lane centre, preview curvature, and path following controller blocks.
The path following the controller block keeps the vehicle travelling within a marked highway lane while maintaining a user-set velocity.

- ✓ The virtual lanes block creates a virtual lane from the path points. The controller must know the lateral deviation and relative yaw angle accordingly to the virtual lane.

- ✓ The preview curvature block converts trajectory to curvature input which is required because the ego vehicle needs to track the curvature while its longitudinal velocity is also changing.

- ✓ Path following for lane change first needs lateral control that keeps the ego vehicle travelling along the centre line of its lane by adjusting the steering of the ego vehicle and is called lane-keeping assist.

- ✓ Secondly, longitudinal control maintains a user-set velocity of the ego vehicle and is known as cruise control.

There are reasons to use adaptive model predictive control (MPC) instead of classical Proportional-Integral-Derivative controllers (PID). Tuning the Proportional-Integral-Derivative controllers (PID) for larger systems is challenging. Model predictive control (MPC) can handle multi-input multi-output allowing the controller to respond to data from various sensors. It can consider the constraints from these sensors, like slowing down the vehicle if a corner and stopping when a sign is detected. It is known as a preview capability.

A traditional predictive controller model is ineffective at handling the varying dynamics as it uses a constant internal plant model. Therefore, the adaptive model is preferred.

- ✓ The patch following controller, which is inside this subsystem, controls the lateral and longitudinal motion. Lateral control proposes the steering angle as longitudinal control suggesting longitudinal acceleration.

- ✓ Vehicle dynamics are responsible for the ego vehicle's longitudinal, lateral and yaw motion. This subsystem contains pre-built vehicle models that can be customized and parameterized. The outputs from this subsystem have the information about the position, velocity, yaw, yaw rate and the ego actor id.

❖ Matrices and visualization subsystems

Matrices and visualization subsystems are used to monitor the test bench parameters and visualize them.

III. Simulate the system in real world tracks with actors and collect data on variables

We used the RoadRunner component of Matlab and OpenStreetMap to use real-world maps in simulations [25]. We used a Swedish road precisely in jönköping. After performing a simulation for each time frame, data such as; velocity, acceleration, and position could be collected; with these data in the future, we could identify the events where two vehicles are close to each other to make the possibility for a collision. For such an event, we can calculate the time-to-collision (TTC). However, one simulation takes considerable time and computer power, and it was not practical to perform many simulations, such as 10,000, using our computers. Therefore, we used the highD dataset with the information on vehicle movement on a highway in Germany. Here we substitute the data which needs to be generated after many simulations in Matlab with this dataset.

1. RoadRunner

RoadRunner is an interactive editor that lets you design 3D scenes for simulating and testing automated driving systems. To customize Roadway scenes, you create region-specific road signs and markings. Then we insert signs, signals, guardrails, road damage, foliage, buildings, and other 3D models. RoadRunner provides tools for setting and configuring traffic signal timing, phases, and vehicle paths at intersections.

RoadRunner supports lidar point cloud visualization, aerial imagery, and GIS data. You can import and export road networks.

Below we presented some figures on what road design may look like in RoadRunner.

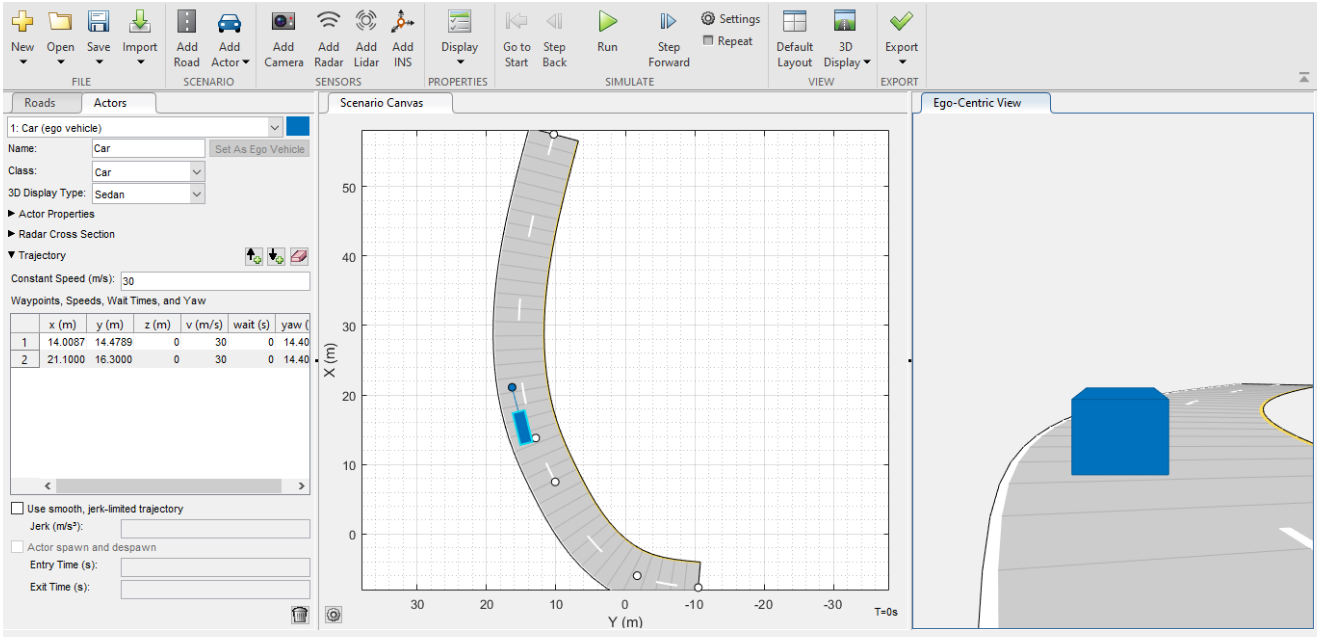


Figure 4: An example of a small road and one vehicle in RoadRunner.

Figure 5: A 3D view of an example of a small road and one vehicle in RoadRunner.

2. OpenStreetMap (OSM)

Crowdsourced mapping and citizen-driven spatial data collection are radically changing the relationship between traditional map production and those individuals and organizations that consume the data [25]. In the past, professionals created maps under the domain of the national mapping agencies. The evolution of online mapping tools, the access to high-resolution satellite imagery, and mobile devices with GPS for geotagging features empowered the citizens to produce maps. OpenStreetMap (OSM) is one of the most successful and cited examples [26]. This concept is called Volunteered Geographic Information (VGI).

The OpenStreetMap (OSM) project started in 2004. Now it has become the most famous VGI project. OpenStreetMap (OSM) can be of use in many areas such as Data Download Applications and Services, Education and Research Use of OSM, Disaster and Humanitarian OSM, Government and Industry Usage, Visualization of OSM Data, Software (OSM Editors, Routing Services, Vector Rendering, other services), Quality Assurance for OSM, and Games and Leisure [27].

In the project, we used a road in Jonkoping, Sweden (see Appendix).

3. HighD dataset

To get a highD dataset, remotely controlled aerial vehicles such as drones recorded German highway traffic. Computer vision algorithms extracted those trajectories.

By using neural networks, vehicles were detected and localized in every frame.

To smooth trajectories from those detections, they tracked vehicles over time. With Bayesian smoothing, the movement was smoothed [29].

IV. Generate the probability distribution

Within the last ten years, several projects have been dealing with collecting driving data recorded with onboard sensors[28]. EuroFOT in Europe, SHRP 2 in the United States, Next Generation SIMulation (NGSIM), KITTI, and Cityscapes are examples [29].

In the highD dataset we used, post-processed trajectories of cars and trucks from drone video recordings on German highways around Cologne in 2017 and 2018 [29], some key facts of the dataset are given below [30].

- In the dataset, they recorded around 110 500 vehicles.
- the total driven distance is 45 000 km,
- the total driven time is 447 h,

- 11 000 lane changes, 5 600 were completely performed in the observed area,
- 850 cut-in manoeuvres: THW from around 0.1 – 4 s with a distribution peak at 1s,
- proportion of 77% passenger cars, 23% trucks,
- 6 locations around Cologne (2 or 3 lanes per direction),
- 60 videos recorded at 25 fps and 4K resolution,
- the average video length is 17 min (a total of 16.5 h),
- the pixel resolution is 1 PBX = 10 cm,
- each vehicle is visible for a median of 14 s,
- the recording time was between 08 a.m. and 5 p.m.,
- sunny weather and low wind conditions during the recordings.

The data is in CSV files. The variables such as velocity, distance, acceleration, time-to-collision (TTC), and time headway (THW) are available for each recorded time frame. We developed the probability distribution function for time-to-collision (TTC) after a data analysis for the highD dataset. For our study, we considered the events where time-to-collision (TTC) is less than 200s to build the probability distribution.

1. Time to collision (TTC)

Finding a collision-free optimal trajectory estimation theory must be applied as in [1,2] is defined as a process of inferring the value of a quantity from indirect, inaccurate, and uncertain observations.

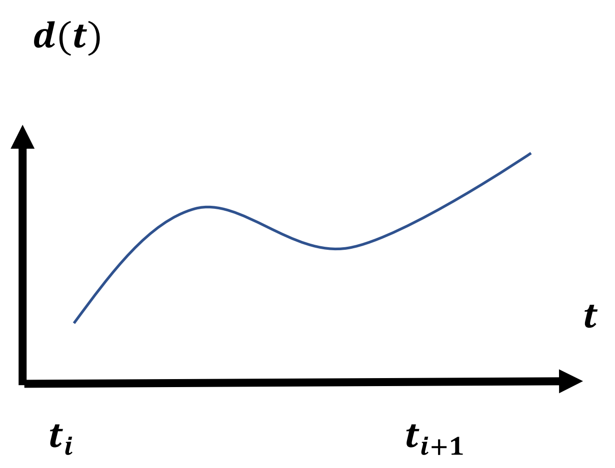
Estimators are into two classes:

Parameter estimators: deals with time-invariant system

State estimators: deals with time-variant system

For our project's sake, we will only consider a state estimator. In our project, we used estimation theory to see how an automated vehicle (V1) determine the velocity of the second vehicle (V2) in the front to make some decision such as increasing or decreasing the speed and lane changing.

❖ Measurement (input data)

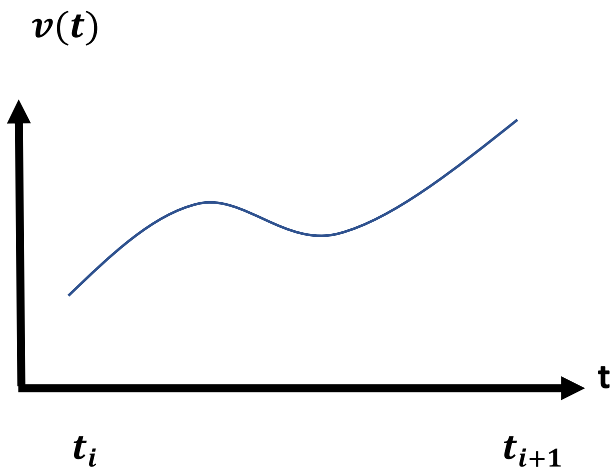


Input data

$time\ t_i \quad i = 1, \dots, 16000$

$distance\ d(ti) > 0 \quad i = 1, \dots, 16000$

$velocity\ v(ti) \geq 0 \quad i = 1, \dots, 16000$



Definitions

$$v(t) = \frac{d}{dt}\, d(t)$$

$$v(t) < 0 \qquad d(t)\ is\ decreasin$$

$$a(t) = \frac{d}{dt}\, v(t)$$

$$a(t) < 0 \qquad v(t)\ is\ decreasing$$

Statistical model

A. general case

Given a general case measurement model

$$\begin{cases} v(t_i) = & v + w_i \\ v(t_{i+1}) = & v + a\,(t_{i+1} - t_i\,) + w_{i+1} \\ & \quad\quad . \\ & \quad\quad . \\ & \quad\quad . \\ v(t_{i+I}) = & v + a\,(t_{i+I} - t_i\,) + w_{i+I} \end{cases} \tag{1}$$

where v is velocity, a is acceleration, t_i is time and w_i is zero mean uncorrelated Gaussian noise with variance σ^2

$$E\left(w_i w_j\right) = \sigma^2 \sigma_{ij} \tag{2}$$

$$\underbrace{\begin{pmatrix} v(t_i) \\ v(t_{i+1}) \\ \vdots \\ . \\ v(t_{i+I}) \end{pmatrix}}_{x} = \underbrace{\begin{pmatrix} 1 & 0 \\ 1 & t_{i+1} - t_i \\ . & . \\ . & . \\ . & . \\ 1 & t_{i+I} - t_i \end{pmatrix}}_{H} \underbrace{\begin{pmatrix} v \\ a \end{pmatrix}}_{\theta} + \underbrace{\begin{pmatrix} w_i \\ w_{i+1} \\ . \\ . \\ . \\ w_{i+I} \end{pmatrix}}_{w} \quad (3)$$

so that

$$x = H\theta + \mathrm{w}, \quad (4)$$

where

$$C = E\{ww^T\} = \sigma^2 I, \quad (5)$$

and $$C^{-1} = \frac{1}{\sigma^2}\mathrm{I}, \quad (6)$$

where x is a random measurement, θ is the unknown (deterministic) parameter to be estimated, and C is covariance matrix.

Based on the statistical model, as in [1] it can be shown that the Fisher information matrix I(θ) is given by

$$I(\theta) = H^T C^{-1} H = \frac{1}{\sigma^2}\, H^T H, \quad (7)$$

and Maximum Likelihood (ML) estimate $\hat{\theta}_{\mathrm{ML}}\,(x)$ is given by

$$\hat{\theta}_{\mathrm{ML}}(x) = (H^T C^{-1} H)^{-1}\, H^T C^{-1} x = (H^T H)^{-1} H^T x. \quad (8)$$

B. special case

❖ Two-point estimation of v and a

Given a special case measurement model

$$\begin{cases} v(t_i) = & v + w_i \\ v(t_{i+1}) = & v + a\,(t_{i+1} - t_i\,) + w_{i+1} \end{cases}, \tag{9}$$

where v is velocity, a is acceleration, t_i is time and w_i is zero mean uncorrelated Gaussian noise with variance σ^2

$$\underbrace{\begin{pmatrix} v(t_i) \\ v(t_{i+1}) \end{pmatrix}}_{x} = \underbrace{\begin{pmatrix} 1 & 0 \\ 1 & t_{i+1} - t_i \end{pmatrix}}_{H} \underbrace{\begin{pmatrix} v \\ a \end{pmatrix}}_{\theta} + \underbrace{\begin{pmatrix} w_i \\ w_{i+1} \end{pmatrix}}_{w}, \tag{10}$$

so that

$$x = H\theta + \mathrm{w}\,, \tag{11}$$

where x is a random measurement, θ is the unknown (deterministic) parameter to be estimated, and C is covariance matrix.

$$H = \begin{pmatrix} 1 & 0 \\ 1 & T_i \end{pmatrix}, \qquad H^{-1} = \frac{1}{T_i}\begin{pmatrix} T_i & 0 \\ -1 & 1 \end{pmatrix}, \tag{12}$$

$$T_i = t_{i+1} - t_i. \tag{13}$$

Maximum Likelihood (ML) estimate $\hat{\theta}_{\mathrm{ML}}\,(x)$ is given by

$$\hat{\theta}_{\mathrm{ML}}\,(x) = \begin{pmatrix} \hat{v} \\ \hat{a} \end{pmatrix} = (H^T H)^{-1} H^T x = H^{-1} H^{-T} H^T x = H^{-1}\, x \tag{14}$$

$$= \frac{1}{T_i}\begin{pmatrix} T_i & 0 \\ -1 & 1 \end{pmatrix}\begin{pmatrix} v(t_i) \\ v(t_{i+1}) \end{pmatrix} = \frac{1}{T_i}\begin{pmatrix} T_i\, v(t_i) \\ v(t_{i+1}) - v(t_i) \end{pmatrix} \qquad (15)$$

$$= \begin{pmatrix} v(t_i) \\ \frac{v(t_{i+1})-v(t_i)}{T_i} \end{pmatrix}, \qquad (16)$$

and this result is physically reasonable as it can be seen clearly bellow

$$\begin{cases} \hat{v}(t_i) = v(t_i) \\ \hat{a}(t_i) = \frac{v(t_{i+1})-v(t_i)}{t_{i+1}-t_i} \end{cases} \text{, and can be done for } i = 1, \dots, 15999 \qquad (17)$$

and Fisher information is

$$I(\theta) = \frac{1}{\sigma^2}\, H^T H = \frac{1}{\sigma^2}\begin{pmatrix} 1 & 1 \\ 0 & T_i \end{pmatrix}\begin{pmatrix} 1 & 0 \\ 1 & T_i \end{pmatrix} \qquad (18)$$

then we got

$$I(\theta) = \frac{1}{\sigma^2}\begin{pmatrix} 2 & T_i \\ T_i & {T_i}^2 \end{pmatrix}, \qquad (19)$$

and

$$I^{-1}(\theta) = \frac{\sigma^2}{{T_i}^2}\begin{pmatrix} {T_i}^2 & -T_i \\ -T_i & 2 \end{pmatrix}. \qquad (20)$$

Because estimation error can result in the wrong decision of our automated vehicle, we used Cramer-Rao lower bound to determine the best estimation possible.

Cramer-Rao lower bound is given by

$$E(|\hat{v}(t_i) - v|^2) \geq [I^{-1}(\theta)]_{11} = \sigma^2, \tag{21}$$

$$E(|\hat{a}(t_i) - a|^2) \geq [I^{-1}(\theta)]_{22} = \frac{2\sigma^2}{{T_i}^2} \ . \tag{22}$$

1. Investigation on how Cramer-Rao lower bound depend on $\boldsymbol{T_i}$

As it can be seen the first element of the Cramer-Rao bound

$$CRLB(v) = [I^{-1}(\theta)]_{11} = \sigma^2 \ , \tag{23}$$

does not depend on T_i.

But the second element of the Cramer-Rao bound which is

$$CRLB(a) = [I^{-1}(\theta)]_{22} = \frac{2\sigma^2}{{T_i}^2}, \tag{24}$$

it does depend on T_i.

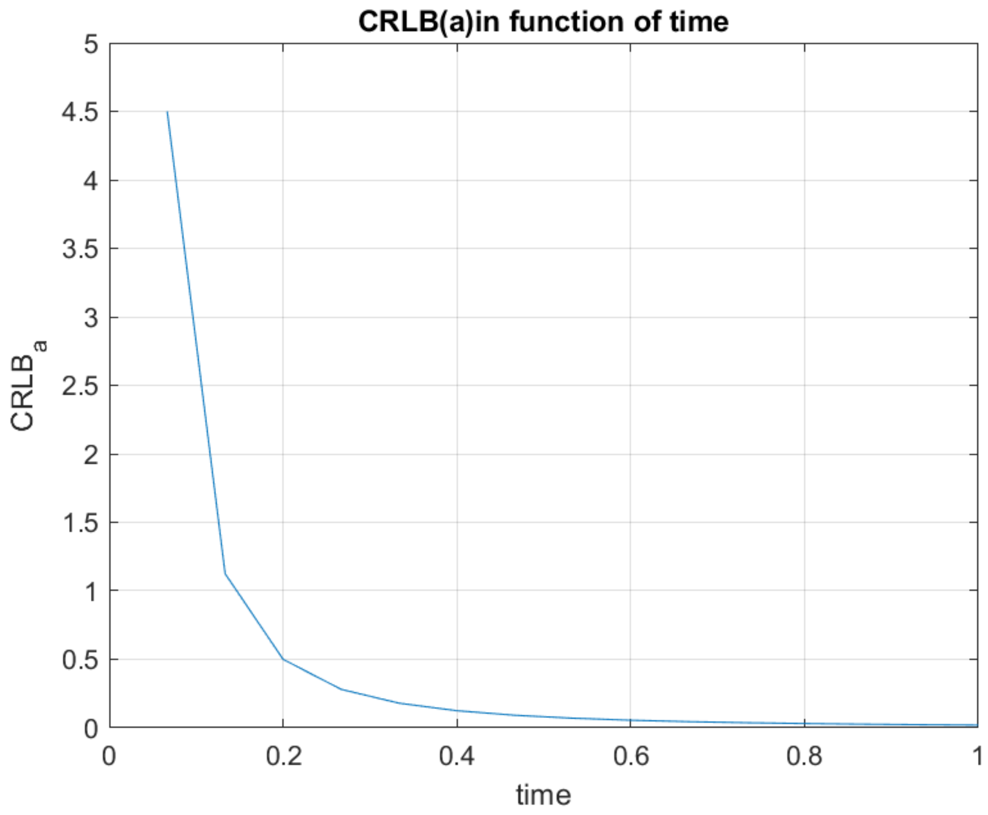


Figure 6: CRLB(a)

It can be difficult to know how accurate must the $\hat{a}$-estimate be to be useful for Time to collision (TTC).

2. Relative error

Let assume that we require that the relative error

$$\frac{\sqrt{E(|\hat{a}(t_i)-a|^2)}}{a} \leq 0.03\ (3\%) \quad (25)$$

we have

$$\frac{\sqrt{[I^{-1}(\theta)]_{11}}}{v} \leq 0.03 \quad (26)$$

And

$$\frac{\sqrt{[I^{-1}(\theta)]_{22}}}{a} \leq \ 0.03 \quad (27)$$

For good estimation:

- $$\frac{\sqrt{CRLB(v)}}{\hat{v}} \leq 0.03 \,, \tag{28}$$

which means

$$\frac{\sigma}{\hat{v}} \leq \ 0.03 \tag{29}$$

- $$\frac{\sqrt{CRLB(a)}}{\hat{a}} \leq 0.03 \,, \tag{30}$$

which means $$\frac{\sqrt{2}\sigma}{\frac{T_i}{\hat{a}}} \leq 0.03 \tag{31}$$

According to the assumption we made about relative error, for a good estimation $\frac{\sqrt{2}\sigma}{\frac{T_i}{\hat{a}}} \leq 0.03$

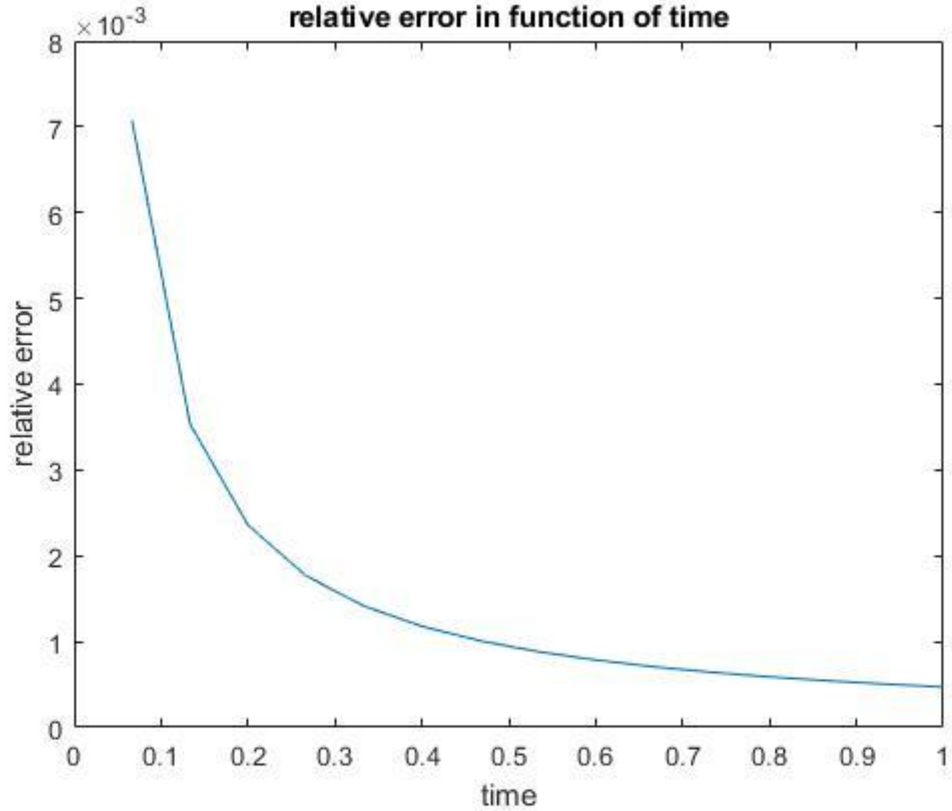


Figure 7: Relative error

❖ **Multi-point estimation**

Let consider 16754 incidents of 22539 vehicles. For TTC = T_i , time to collision at time t_i, based on measurements $d(t_i)$ and $v(t_i)$.

Model:

$$d(t) = d(t_i) + v(t_i)t = 0\,, \tag{32}$$

$$t = -\frac{d(t_i)}{v(t_i)}, \tag{33}$$

note that

$$v(t_i) < 0, \text{ otherwise } t = \infty, \tag{34}$$

$$d(t_i) > 0 \ \text{ (No collision).} \tag{35}$$

The histogram of the time to collision (TTC) generated in MATLAB is given in figure 8.

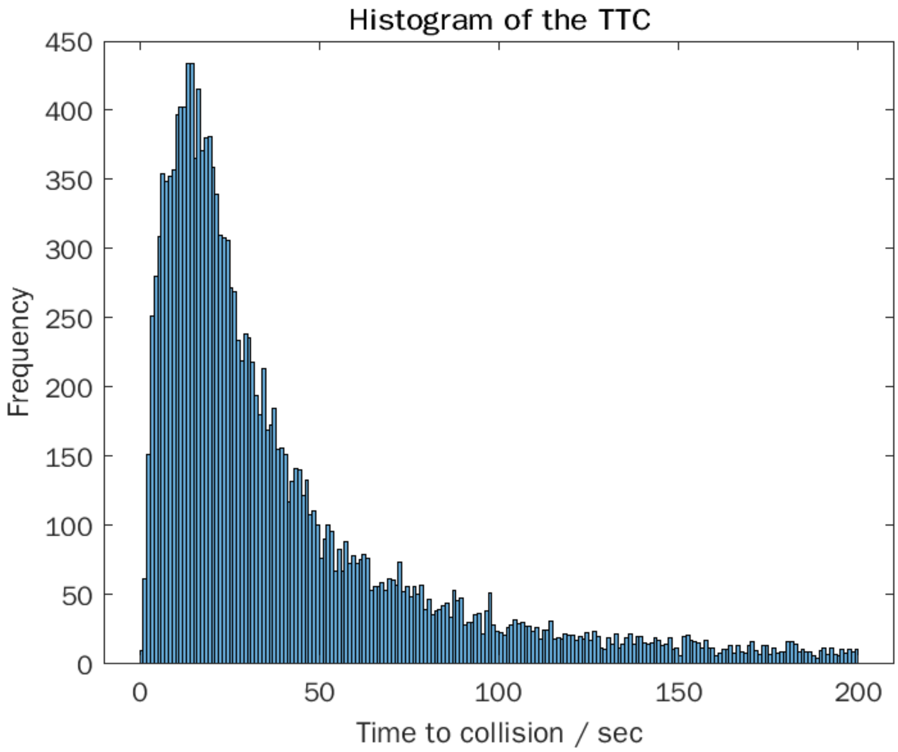


Figure 8: Histogram of the incidents where time-to-collision (TTC) < 200 s

The probability density function in the histogram is not continuous. The Kernel density estimation is for solving the problem of discontinuity. In a non-parametric system, the kernel density method estimates the distribution density function of the scenario parameters [31].

The formula of Kernel density estimation is:

$$\hat{p}(x) = \frac{1}{hN} \sum_{i=1}^{N} K\left(\frac{x - x_i}{h}\right) \tag{36}$$

where $\hat{p}(x)$ is the estimated distribution density function; $K(x)$ is the kernel function, and the Gaussian kernel function is selected; N_k is the sample number of the distribution to be estimated; x is the random variable; x_i is the sample; and h is the bandwidth, $h > 0$ [31]. The probability distribution function generated in MATLAB is given in figure 9.

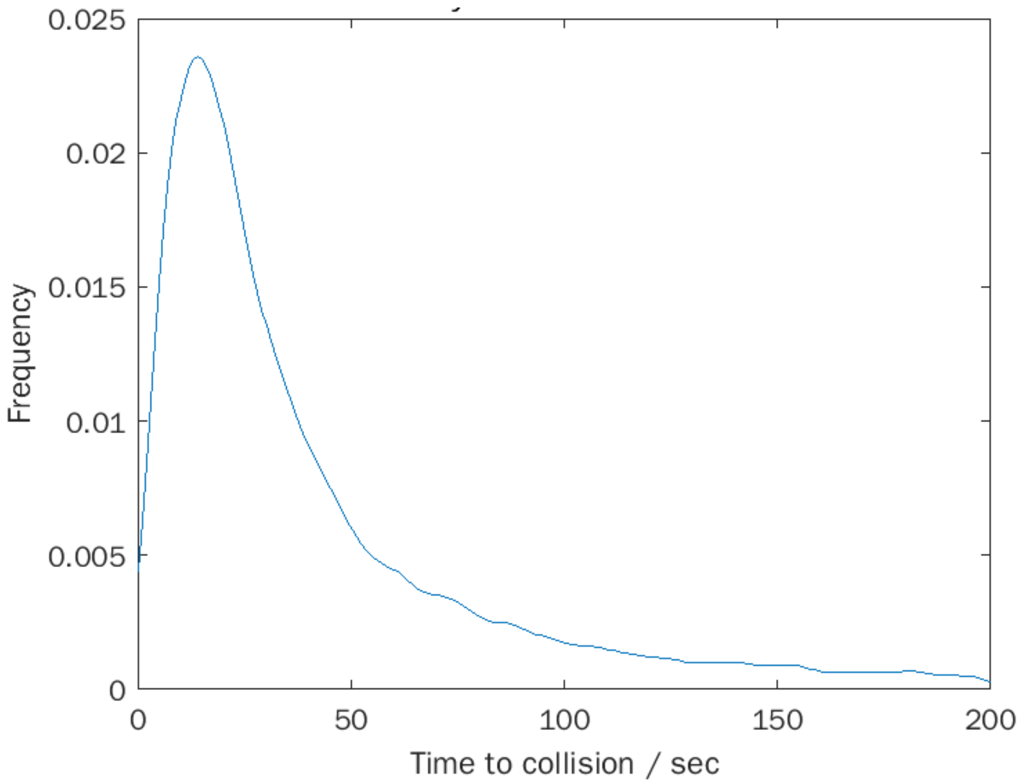


Figure 9: probability distribution of the TTC

V. Estimate the probability of critical scenarios

Critical scenario metrics include time-to-collision (TTC), braking time, expected deceleration, and so on [31]. For this study, we selected TTC to define the critical events.

Where TTC is the collision time, d_a is the distance between the ego-vehicle and the preceding vehicle, and ΔV is the relative speed of the ego-vehicle and the preceding vehicle [31].

We found two sources which propose two different matrices. Xia Q and team propose the matrix given in table 2 [31] while Benmimoun M and team propose the matrix given in table 3 [32].

Table 2: Critical scenario matrix based on time-to-collision (TTC) by Xia Q and team [31].

TTC range / sec	**Condition**
TTC <= 0	Collision condition
0 < TTC < 0.5	Pre-collision condition
0.5 < TTC < 2.5	Dangerous condition
2.5 < TTC	Safe condition

Table 3: Critical scenario matrix based on TTC by Benmimoun and team [32].

Incident level	**Thresholds due to Time headway (THW)**			**Thresholds due to Time to collision (TTC)**	
	THW [s]	**Relative velocity**	Or	**TTC [s]**	**Status brake light [–]**
Level 1	0.5	> 20		1.75	Off
	0.35	> 10		-	-
Level 2	0.35	> 20		< 1	On
Level 3	-	-		< 1	Off

Capturing the velocity and distance data for several lane-changing events in MATLAB simulation was a time and resource-consuming task. Therefore, we decided to use the collected dataset from other researchers to replace the thousand-plus simulations.

1. Estimation for probability

The probability of a critical scenario is then calculated. The probabilities according to the critical scenario matrix is given below.

Table 4: Probability according to the critical scenario.

TTC range / sec	**Condition**	**Probability respectively to condition**
TTC <= 0	Collision condition	0.00000
0 < TTC < 0.5	Pre-collision condition	5.97×10^{-5}
0.5 < TTC < 2.5	Dangerous condition	0.00781
2.5 < TTC	Safe condition	0.99212

❖ **Velocity and acceleration (estimated) data**

Model:

$$d(t) = d(t_i) + \hat{v}(t_i) + \hat{a}(t_i)\frac{t^2}{2} = 0, \quad (37)$$

$$\begin{pmatrix} \frac{d}{dt}d(t) = \hat{v}(t_i) \\ \frac{d^2}{dt^2}d(t) = \hat{a}(t_i) \end{pmatrix}, \quad (38)$$

complete the squares

$$\left(t + \frac{\hat{v}(t_i)}{\hat{a}(t_i)}\right)^2 = \frac{\hat{v}^2(t_i)}{\hat{a}^2(t_i)} - 2\frac{d(t_i)}{\hat{a}(t_i)}, \quad (39)$$

$$t = -\frac{\hat{v}(t_i)}{\hat{a}(t_i)} \pm \sqrt{\frac{\hat{v}^2(t_i)}{\hat{a}^2(t_i)} - 2\frac{d(t_i)}{\hat{a}(t_i)}}, \quad (40)$$

we have two solutions t_1 and t_2.

- If t is complex value, there is no solution → put $t = \infty$

$$\left(\frac{\hat{v}^2(t_i)}{\hat{a}^2(t_i)} < 2\frac{d(t_i)}{\hat{a}(t_i)}\right), \tag{41}$$

- If $t_1 < 0$ and $t_2 < 0$, there is no solution → put $t = \infty$
- If $t_1 < 0$ and $t_2 > 0$ → put $t = t_2$

- If $t_2 < 0$ and $t_1 > 0$ → put $t = t_1$

- If $t_1 > 0$ and $t_2 > 0$ → put $t = \min(t_1, t_2)$

VI. Conclusion

By all indications, it is no longer a matter of if, but of when, autonomous vehicles will be on our highways and city streets, and in numbers sufficient to make a difference to the operational performance of our transportation networks. Through our project we have been able to realize the challenges which automated vehicle are facing and we found that most of technical challenges maybe overcome by improvement in sensor function fusion and state estimation and but most of emotions and economical challenges can be overcome through better planning before shifting to the automated vehicles.

VII. Bibliography

[1] Steven M. Kay, “Cramer-Rao Lower Bound” in *Fundamentals of statistical signal processing: Estimation theory*, New Jersey, USA: prentice-Hall. Inc, 1993, Ch. 3, pp. 27-62.

[2] Yoriyoshi Hashimoto, *et al.,*”A probabilistic model of pedestrian crossing behavior at signalized intersections for connected vehicles,” *Transportation Research Part C: Emerging Technologies*, Vol.71, Jul.2016, pp. 164-181, doi.org/10.1016/j.trc.2016.07.011.

[3] D. B. O. Boesl and B. Liepert, “4 Robotic Revolutions – proposing a holistic phase model describing future disruptions in the evolution of robotics and automation and the rise of a new Generation ‘R’ of Robotic Natives,” *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 1262-1267, doi: 10.1109/IROS.2016.7759209.

[4] Á. Takács, I. Rudas, D. Bösl and T. Haidegger, “Highly Automated Vehicles and Self-Driving Cars [Industry Tutorial],” in *IEEE Robotics & Automation Magazine*, vol. 25, no. 4, pp. 106-112, Dec. 2018, doi: 10.1109/MRA.2018.2874301.

[5] Y. Xu, B. Y. Chen, X. Shan, W. H. Jia, Z. F. Lu and G. Xu, “Model predictive control for lane keeping system in autonomous

vehicle," *2017 7th International Conference on Power Electronics Systems and Applications – Smart Mobility, Power Transfer & Security (PESA)*, 2017, pp. 1-5, Doi: 10.1109/PESA.2017.8277758.

[6] Sun, Yuchao & Olaru, Doina & Smith, Brett & Greaves, Stephen & Collins, Andrew. "Road to autonomous vehicles in Australia: A comparative literature review," Nov.2016.

[7] Acheampong, Ransford A, "Literature review on the social challenges of autonomous transport," Ap.2018, Doi: 10.13140/RG.2.2.31845.50406.

[8] Bansal, P., Kockelman, K.M. "Are we ready to embrace connected and self-driving vehicles? A case study of Texans," in *Transportation* **45,** Nov.2018, pp. 641–675, doi.org/10.1007/s11116-016-9745-z

[9] Prateek Bansal, Kara M. Kockelman, "Forecasting Americans' long-term adoption of connected and autonomous vehicle technologies," in *Transportation Research Part A: Policy and Practice*, Vol.95, Oct. 2017, pp 49-63, doi.org/10.1016/j.tra.2016.10.013.

[10] Adriano A. *et al.*, “Automated Vehicles and the Rethinking of Mobility and Cities,” in *Transportation Research Procedia,* Vol. 5, 2015, pp. 145-160, doi.org/10.1016/j.trpro.2015.01.002.

[11] Jurgen, Ronald. “Autonomous Vehicles for Safer Driving, SAE International”, *ProQuest eBook Central*, 20 https://ebookcentral-proquest-com.proxy.lnu.se/lib/linne-ebooks/detail.action?docID=5341854.

[12] Jean-François B. *et al..,* “The social dilemma of autonomous vehicles,” in *science*, Vol.352, no.6293, Jun. 2016, pp. 1573-1576, doi:10.1126/science.aaf2654.

[13] J. M. Anderson, *et al.,*” Vehicl*e* Technology: A Guide for Policymakers”. Santa Monica, CA: Rand Corporation, 2014.

[14] Thomopoulos, N., Givoni, M. “The autonomous car—a blessing or a curse for the future of low carbon mobility? An exploration of likely vs. desirable outcomes”. *Eur J Futures Res* **3,** 14 (2015). https://doi.org/10.1007/s40309-015-0071-z.

[15] Cavoli, Clemence.*et al.,*” Social and behavioural questions associated with Automated Vehicles. A Literature Review,” *department of transport*, Jan 2017.

[16] Mingyang HAO, Toshiyuki YAMAMOTO, “Shared Autonomous Vehicles: A Review Considering Car Sharing and Autonomous Vehicles,” Asian Transport Studies, Vol. 5, no 1, pp. 47-63, March 2018, doi.org/10.11175/eastsats.5.47

[17] M. H. Ang, *et al.*, ”Perception, Planning, Control, and Coordination for Autonomous Vehicles,” *Machines,* vol. 5, no. 1, 2017.

[18] I. Sanchez, A. D’Jorge, A. Ferramosca, G. Raffo och A. H. Gonzlez, ”Path Following and Trajectory Tracking Model Predictive Control using Artificial Variables for Constrained Vehicles,” i 2019 XVIII Workshop on Information Processing and Control (RPIC), Salvador, Brazil, 2019.

[19] S. Ulbrich, T. Menzel, A. Reschka, F. Schuldt och M. Maurer, ”Defining and Substantiating the Terms Scene, Situation, and Scenario for Automated Driving,” I 2015 IEEE 18th International Conference on Intelligent Transportation Systems, Gran Canaria, Spain, 2015.

[20] T. Ponn, M. Breitfuß, X. Yu och F. Diermeyer, ”Identification of Challenging Highway-Scenarios for the Safety Validation of Automated Vehicles Based on Real Driving Data,” I 2020 Fifteenth

International Conference on Ecological Vehicles and Renewable Energies (EVER), Germany, 2020.

[21] Y. Luo, G. Yang, M. Xu, Z. Qin och K. Li, ”Cooperative Lane-Change Maneuver for Multiple Automated Vehicles on a Highway,” Automotive Innovation, vol. 2, p. 157–168, 2019.

[22] The MathWorks Inc., ”Highway Lane Change,” The MathWorks Inc., 2021. [Online]. Available: https://se.mathworks.com/help/driving/ug/highway-lane-change.html.

[23] The MathWorks Inc., ”Autonomous Navigation for Highway Driving | Design and Simulate a Lane Change Maneuver System,” MATLAB, 28 Oct 2021. [Online]. Available: https://www.youtube.com/watch?v=DwlyVmIfiRQ.

[24] Franz Pucher, ”Trajectory Planning in the Frenet Space,” Franz Pucher, 25 August 2018. [Online]. Available: https://fjp.at/posts/optimal-frenet/.

[25] L. See, J. Estima, A. Pődör, J. J. Arsanjani, J.-C. L. Bayas och V. R, ”Sources of VGI for Mapping,” I *Mapping and the Citizen Sensor*, London, Ubiquity Press, 2017, p. 13–35.

[26] H. Fan, B. Yang, A. Zipf och A. Rousell, ”A polygon-based approach for matching OpenStreetMap road networks with regional transit authority data,” *International Journal of Geographical Information Science,* vol. 30, nr 4, pp. 748-764, 2016.

[27] P. Mooney och M. Minghini, ”A Review of OpenStreetMap Data,” I *Mapping and the Citizen Sensor*, London, Ubiquity Press, 2017, pp. 37-59.

[28] R. Eenink, Y. Barnard, M. Baumann, X. Augros och F. Utesch, ”UDRIVE: the European naturalistic driving study,” I *Proceedings of Transport Research Arena*, 2014.

[29] R. Krajewski, J. Bock, L. Kloeker och L. Eckstein, ”The highD Dataset: A Drone Dataset of Naturalistic Vehicle Trajectories on German Highways for Validation of Highly Automated Driving Systems,” I *IEEE 21st International Conference on Intelligent Transportation Systems (ITSC)*, Maui, Hawaii, USA, November 2018.

[30] F. Kruber, J. Wurst, S. Chakraborty och M. Botsch, ”Highway traffic data: macroscopic, microscopic and criticality analysis for capturing relevant traffic scenarios and traffic modeling based on the highD data set,” *arXiv: Signal Processing,* vol. 1903.04249, 2019.

[31] Q. Xia, Y. Chai och H. Lv, ”Research on Accelerated Testing of Cut-In Condition of Electric Automated Vehicles Based on Monte Carlo Simulation,” *Sustainability,* p. 13, 2021

[32] M. Benmimoun, F. Fahrenkrog, A. Zlocki och L. Eckstein, ”Detection and Classification of Critical Incidents by Means of Vehicle Data,” *ATZ worldwide,* vol. 114, p. 60–66 , 2012.

[33] M. H. Ang, D. Rus, Y. H. Eng, M. Meghjani, X. Shen, X. Du, H. Andersen och S. D. Pendleton, ”Perception, Planning, Control, and Coordination for Autonomous Vehicles,” *Machines,* vol. 5, nr 1, 2017.

[34] I. Sanchez, A. D’Jorge, A. Ferramosca, G. Raffo och A. H. Gonzlez, ”Path Following and Trajectory Tracking Model Predictive Control using Artificial Variables for Constrained Vehicles,” I *2019 XVIII Workshop on Information Processing and Control (RPIC),* Salvador, Brazil, 2019.

[35] S. Ulbrich, T. Menzel, A. Reschka, F. Schuldt och M. Maurer, ”Defining and Substantiating the Terms Scene, Situation, and Scenario for Automated Driving,” I *2015 IEEE 18th International Conference on Intelligent Transportation Systems,* Gran Canaria, Spain, 2015.

[36] T. Ponn, M. Breitfuß, X. Yu och F. Diermeyer, ”Identification of Challenging Highway-Scenarios for the Safety Validation of Automated Vehicles Based on Real Driving Data,” I *2020 Fifteenth International Conference on Ecological Vehicles and Renewable Energies (EVER),* Germany, 2020.

[37] Y. Luo, G. Yang, M. Xu, Z. Qin och K. Li, ”Cooperative Lane-Change Maneuver for Multiple Automated Vehicles on a Highway,” *Automotive Innovation*, vol. 2, p. 157–168, 2019.

[38] The MathWorks Inc., ”Highway Lane Change,” The MathWorks Inc., 2021. [Online]. Available: https://se.mathworks.com/help/driving/ug/highway-lane-change.html.

[39] The MathWorks Inc., ”Autonomous Navigation for Highway Driving | Design and Simulate a Lane Change Maneuver System,” MATLAB, 28 Oct 2021. [Online]. Available: https://www.youtube.com/watch?v=DwlyVmIfiRQ.

[40] e. a. Yoriyoshi Hashimoto, ”probabilistic model of pedesrian crossing behavior at signalized intersextions for connected vehicles,” *transportation research part c: emerging techonologies,* vol. 71, pp. 164-181, 2016.

Appendix

Real world map in Matlab.

As part of simulation real world map must be included to study how automated vehicle may react. For our study a road in Jönköping precisely Ekhagen was used, we generated a Matlab code for the road design with geographic coordinates.

```
Function scenario = createDrivingScenario()
% createDrivingScenario Returns the drivingScenario defined in the Designer
% Construct a drivingScenario object.
Scenario = drivingScenario(‘GeographicReference’, [57.7795 14.1874 0], ...
    ‘VerticalAxis’, ‘Y’);
% Add all road segments
roadCenters = [1243.137 -189.7562 -0.1236781;
    1250.246 -188.1947 -0.1250182];
laneSpecification = lanespec([1 1]);
road(scenario, roadCenters, ‘Lanes’, laneSpecification, ‘Name’, ‘114325524’);

roadCenters = [2002.674 -478.3759 -0.3315872;
    2006.574 -507.0973 -0.3350277;
```

```
    2009.648 -526.3857 -0.3375552;
    2011.914 -548.6593 -0.3401442;
    2016.26 -565.3297 -0.3429684;
    2018.525 -571.1312 -0.3441994;
    2022.98 -580.8073 -0.3464811;
    2030.079 -592.6985 -0.3498246;
    2050.768 -625.8439 -0.359592;
    2065.966 -650.5054 -0.3669514;
    2083.1 -677.7275 -0.375344;
    2094.69 -697.2341 -0.3812323;
    2109.162 -720.1584 -0.3885361;
    2128.144 -751.7674 -0.3984716;
    2148.835 -785.7699 -0.4094887;
    2156.119 -797.3601 -0.4133787;
    2161.526 -806.0554 -0.4162972;
    2178.435 -833.2104 -0.4255241;
    2191.657 -853.2616 -0.4326924];
laneSpecification = lanespec([1 1]);
road(scenario, roadCenters, ‘Lanes’, laneSpecification, ‘Name’, ‘Ekhagsringen’);
roadCenters = [2075.265 -230.3991 -0.3409658;
    2069.655 -255.4944 -0.3401028;
    2066.482 -263.5038 -0.3394021;
    2062.219 -270.2887 -0.3383096;
    2058.926 -276.583 -0.3375179;
```

```
    2050.353 -289.6404 -0.3353421;
    2045.353 -297.8734 -0.3341197;
    2039.432 -307.6216 -0.3326906;
    2030.438 -323.6751 -0.3306219;
    2019.511 -342.2912 -0.3281325;
    2008.71 -363.8696 -0.325924;
    2003.85 -377.5376 -0.3251928;
    1999.529 -400.4827 -0.325239;
    1998.272 -419.9404 -0.3260966;
    1999.451 -446.7363 -0.328285;
    2002.674 -478.3759 -0.3315872];
laneSpecification = lanespec([1 1]);
road(scenario, roadCenters, ‘Lanes’, laneSpecification, ‘Name’,
‘Ekhagsringen’);
roadCenters = [2228.57 -432.3825 -0.4030546;
    2106.429 -454.3332 -0.3631718;
    2038.058 -473.7696 -0.3424251;
    2007.345 -477.5717 -0.3329917;
    2002.674 -478.3759 -0.3315872];
laneSpecification = lanespec([1 1]);
road(scenario, roadCenters, ‘Lanes’, laneSpecification, ‘Name’,
‘Stockrosgatan’)
```