Driving Risk Models for Predicting, Planning and Warning

Vom Fachbereich
Elektrotechnik und Informationstechnik
der Technischen Universität Darmstadt
zur Erlangung des akademischen Grades
eines Doktor-Ingenieurs (Dr.-Ing.)
genehmigte Dissertation

von

M.Sc. Tim Puphal

geboren am 23. Mai 1991 in Frankfurt am Main

Referent:	Prof. Dr.-Ing. J. Adamy
Korreferent:	Prof. Dr. rer. nat. B. Sendhoff
Tag der Einreichung:	26. Januar 2021
Tag der mündlichen Prüfung:	21. April 2021

D17
Darmstadt 2021

Puphal, Tim - Driving Risk Models for Predicting, Planning and Warning
Darmstadt, Technische Universität Darmstadt
Jahr der Veröffentlichung auf TUprints: 2021

Verfügbar unter lediglich die vom Gesetz vorgesehenen Nutzungsrechte gemäß UrhG.

Abstract

Automated cars and driver assistance systems constantly progress in complementing the human user in many parts of the driving task. Prominent examples include car-following on a highway, blind spot monitoring, recommending safe lane changes or even navigating on urban streets. This current trend has mostly originated due to affordable perception sensors and the improved speed of computer chips.

However, for a wider acceptance of self-driving cars, there is still a need to prove safety in terms of accidents and near-critical encounters caused by a technical system. Essentially, humans want technologies in which the reasons behind actions and warnings are known. This understanding helps trust to be increased and allows the driver to deliberately take over control from the system. The ultimate goal is to provide generic and transparent planning algorithms with considered safety margins.

In this dissertation, the presented challenge is tackled by developing analytical driving risk models and applying them to the relevant automotive domains of prediction, planning and warning. The models predict motion of vehicles along paths and incorporate several risk types, e.g., from collisions to sharp turns. Hereby, risks are composed of probabilities and severities and improve the behavior selection of the vehicle.

The dissertation is divided into three parts. Firstly, existing risk models of related work are enhanced with real-world uncertainties that arise from vehicle dynamics, unknown future environment changes and possible behavior alternatives of other vehicles. Analyses using accident data and normal traffic data show that this model has, amongst others, a higher fidelity than state-of-the-art time indicators. Secondly, a novel planning approach is introduced, which minimizes situational risks and maximizes utility and comfort to obtain ego velocity profiles. In all the statistical simulations of car-following and intersection driving, the approach successfully realizes a proactive maneuver. The major novelty of this planner is the intelligent inclusion of priorities between interacting vehicles. Lastly, the dissertation is concluded by leveraging risk-based planners for online driver warning with different car sensor setups and test locations, which shows their real-time applicability. Specifically, and in practice, the time

predictions and low-risk trajectories are transformed into intuitive signal outputs for visualization to a driver.

To summarize, the proposed methods in this dissertation are based on fully transparent models with probabilistic formulations. This can be seen as a substantial contribution for the validation and advancement of intelligent robots; specifically, vehicles. Compared to simple reactive logics and data-driven machine learning methods, the approaches provide detailed information about the system's situation understanding and reasoning for motion planning.

Even if they are not used as driving support technologies themselves, they still could help to rate the driving proficiency and safety of other existing platforms or, rather, the human driver. The basis is always formed by an integrated risk calculation that is parametrized from recorded car encounters and average variations in car dynamics. In this way, we may come a step closer to the goal of zero crashes with fewer traffic jams on roads and comfortable travel.

Kurzfassung

Automatisierte Fahrzeuge und Assistenzsysteme entwickeln sich ständig weiter, um den Fahrer in vielen Fahraufgaben zu ergänzen. Bekannte Beispiele sind das Abstandhalten und Folgen von Autos auf Autobahnen, die Kontrolle von toten Winkeln, die Empfehlung sicherer Fahrspurwechsel oder die Navigation in städtischen Straßen. Diese Innovationen haben größtenteils ihre Ursache in den preiswerten Sensoren zur Perzeption und der verbesserten Rechenleistung von Computerchips.

Trotz der Möglichkeiten ist es zur breiten Akzeptanz von selbstfahrenden Automobilen unerlässlich, die Sicherheit in Bezug auf selbst verschuldete Unfälle oder kritische Begegnungen nachzuweisen. Die Gesellschaft bevorzugt im Wesentlichen Technologien, bei der die Ursachen hinter der ausgewählten Aktion oder Warnung erkennbar sind und verstanden werden. Das Verständnis hilft, Vertrauen aufzubauen und ermöglicht so dem Fahrer, die Kontrolle von einem aktiven System bewusst zu übernehmen. Das finale Ziel ist die Bereitstellung eines generischen und transparenten Algorithmus zur Planung mit harten Sicherheitsgrenzen.

In dieser Dissertation wird das vorgestellte Problem behandelt, indem Risikomodelle beim Fahren entwickelt und sie auf die relevanten Automobilthemen der Prädiktion, Planung und Warnung angewendet werden. Die Modelle prädizieren die Bewegung von Fahrzeugen entlang von Kartenpfaden und umfassen mehrere Typen von Risiken, wie Risiken von Kollisionen bis hin zu gefährlich scharfen Kurven. Hierbei, werden die Fahrrisiken beschrieben durch Wahrscheinlichkeiten und Unfallschwere und verbessern erfolgreich das Verhalten für das Fahrzeug.

Die Dissertation besteht insgesamt aus drei Teilen. Zuerst werden die bestehenden Modelle aus dem Stand der Technik mit realen Unsicherheiten erweitert, die in der Fahrdynamik, in unbekannten zukünftigen Umwelteinflüssen und in möglichen Verhaltensalternativen von anderen Fahrzeugen aufkommen. Unfall- und Verkehrsdaten zeigen, dass unter anderem das Risikomodell eine höhere Auflösung hat als gängige Zeitindikatoren. Als Nächstes wird eine neuartige Planungsmethode eingeführt, welche das Situationsrisiko minimiert und den Nutzen und Komfort maximiert, um ein eigenes Geschwindigkeitsprofil zu finden. In allen statistischen Simu-

lationen von Auffahren oder Kreuzungsfahrten, realisiert der Ansatz erfolgreich ein intelligentes Manöver. Die besondere Eigenschaft der Planungsmethode ist es, die Vorfahrten zwischen den interagierenden Autos intelligent zu berücksichtigen. Zuletzt wird der risikobasierte Planer online für eine Fahrerwarnung mit unterschiedlichen Sensorkonfigurationen und Testorten genutzt. Damit ist die Anwendbarkeit im echten Straßenverkehr aufgezeigt. Das Endsystem ist in der Lage, die Zeitprädiktionen und Trajektorien mit niedrigem Risiko in intuitive Signalausgänge zur Visualisierung in der Praxis umzuformen.

Zusammenfassend basieren die vorgeschlagenen Methoden auf vollständig transparenten Modellen mit probabilistischen Formulierungen. Dies kann als ein bedeutender Beitrag zur Validierung und zum Fortschritt von intelligenten Robotern, im Speziellen bei Fahrzeugen, angesehen werden. Im Vergleich zu reaktiven Logiken und datengetriebenem maschinellem Lernen bieten die Methoden detaillierte Informationen über die Fahrsituation mit direkter Verbindung zur Bewegungsplanung.

Wenn sie nicht als Techniken zur Assistenz verwendet werden, könnten sie dennoch helfen die Fahrerleistung oder Sicherheit von existierenden Plattformen zu bewerten. Möglich ist auch die Einschätzung der Fähigkeiten des Menschen beim Fahren. Die Grundlage bildet immer eine integrierte Risikoberechnung, die von elektronisch aufgezeichneten Fahrzeugbegegnungen und mittleren Variationen der Fahrzeugdynamik parametrisiert wird. So kommen wir dem Finalziel der Vermeidung von Unfällen, weniger Verkehrsstaus und komfortablem Fahren ein Stück näher.

Acknowledgments

This dissertation was financially supported by the Honda Research Institute Europe GmbH. During my PhD journey, I worked in the European Union project VI-DAS from the Horizon 2020 Research Program and in several internal Honda projects from the industry. Hereby, the cooperation with the Control Methods and Robotics Lab enabled me to stay in close contact with the research community of the Technical University of Darmstadt.

On this note, I would like to thank my supervisor Dr. Julian Eggert, who shaped me to become a researcher. He taught me the analytical skills for modeling physical and probabilistic systems, and patiently showed me how to properly analyze and understand research results. Likewise, I thank my advisor Dr. Thomas Weisswange. Due to his effort, I always gave my best and could concentrate on the PhD work. The feedback in meetings and discussions were invaluable for this success.

Next, I want to thank my university supervisors Prof. Jürgen Adamy and Dr. Volker Willert. Their tips on how to publish papers and draft a good PhD thesis were helpful in the process. I also like to thank for their patient effort on reading the bi-annual progress reports.

The work time was only fun and interesting because of the close working relations with my office colleagues from Honda. Among the most important have been Benedict Flade, Viktor Losing, Dennis Orth, Raphael Wenzel, Matti Krüger and Benjamin Metka. Benedict accompanied me in working with middleware softwares for integration and preparing numerous demonstrations, presentations and papers. I also enjoyed the many boulder gym visits with Viktor, Dennis and Benjamin and the short but funny Japanese classes with Raphael and Matti. From the university lab, I am thankful for advices of other PhD students, namely, Fabian Müller, Moritz Bühler, Karsten Kreutz, Florian Damerow and Stefan Klingelschmitt. The summer and Christmas parties were received well.

Simultaneously, at Honda, it was an honor working with all professional scientists, senior scientists and chief scientists: Dr. Bram Bolder, Antonello Ceravola, Dr. Jörg Deigmöller, Dr. Nils Einecke, Dr. Lydia Fischer, Dr. Frank Joublin, Dr. Malte Probst, Dr. Sven Rebhan and Dr. Nico Stein-

hardt. I hope I did not forget somebody, if so, thanks to you as well. In the list, I want to appreciate particularly my colleague and project manager Malte. He showed me how to properly use Python, SVN and Github and gave me insights on his versatile knowledge of computer science. Besides, the hard questions in the scientific seminars from everybody made me to think every aspect of my results through.

Another big thanks goes to the management team of Honda with Andreas Richter, Dr. Jens Schmuedderich and Prof. Bernhard Sendhoff. I am very grateful to have given the opportunity to do research and gain work experience within many projects in the company.

The structure and writing style of my introduction and conclusion chapters follow and are inspired by the dissertation of Andrej Karpathy [57] from Stanford University. Although I do not know him, I am thankful for his blog posts and great writing skills, effectively showing me the way to succeed in a PhD.

Last but not least, I owe the largest thanks to my fiancé Mayo Ishida and my parents Renate Puphal and Jörg Puphal, who supported me in the good as well as bad times during my PhD. You gave me the power to pursue further education and career prospects.

Contents

List of Symbols

Abbreviations

ACC	Adaptive Cruise Control
ADAS	Advanced Driver Assistance System
AV	Autonomous Vehicle
BTN	Brake Threat Number
CMS	Collision Mitigation System
FDM	Foresighted Driver Model
GIDAS	German In-Depth Accident Study
GNSS	Global Navigation Satellite System
HMI	Human-Machine Interface
HUD	Head-Up Display
IDM	Intelligent Driver Model
IIDM	Intersection Intelligent Driver Model
LDW	Lane Departure Warning
NGSIM	Next Generation Simulation
OSM	OpenStreetMap
PET	Post-Encroachment Time
PMM	Path-following Mixture Model
RM	Risk Maps
RNS	Risk Navigation System
ROPT	Risk Optimization Method
RRT	Rapidly-Exploring Random Tree
RSD	Risk Spot Detector
R-LDM	Relational Local Dynamic Map
SUMO	Simulation of Urban Mobility
TH	Time Headway
TP	Traffic Participant
TTC	Time-To-Collision
TTCE	Time-To-Closest-Encounter
TTX	Time-To-X
VI-DAS	Vision Inspired Driver Assistance System
WHO	World Health Organization

Notation

x	Single scalar
x_i or x_j	Set of similar scalars
$\bar{x}$	Mean value
$x_{\max}$	Maximal value
$x_{\min}$	Minimal value
x_{start}	Start value
x_{end}	End value
x_{th}	Threshold value
Δx	Difference value
$\dot{x}$	Derivative of variable x
X	Integrated value
$\mathbf{x}$	Vector
$\mathbf{x} \times \mathbf{y}$	Cross product
$\mathbf{X}$	Matrix
$f(t)$	Function of time
$f(s)$ or $f(t+s)$	Function of future time
$f(s;t)$	Future time function starting at current time
pmf	Probability mass function of variable
cdf	Cumulative distribution function of variable
$\lvert \cdot \rvert$	Absolute value
$\exp\{\cdot\}$ or $e^{(\cdot)}$	Exponential function
$\lVert \cdot \rVert$	Euclidean norm of vector
$\det \lvert \cdot \rvert$	Determinant of square matrix
$\langle \cdot \rangle$	Expected value of variable

Latin Uppercase Letters

$A(s;t)$	Time-accumulated event probability
A_{runs}	Cumulative distribution for, e.g., TH_{2D} in simulation
$B(t)$	Ego benefit over time t
$C(t)$	Trajectory cost over time
D_0	Damage offset for minor costs
$D_{1,2}$	Brownian motion diffusion slope for vehicle 1 or 2
D_c	Steepness constant in TTCE and Gaussian method
$D_{\text{coll}}(s;t)$	Car-to-car collision damage at prediction s
$E(s;t,\Delta t)$	Probability of any event (i.e., escape or critical)
FP	False-positive detections
FWHM	Full Width at Half Maximum value

$\hat{I}_{\text{event}}(\Delta t)$	Instantaneous event probability in interval Δt
$I_{s,1}$ and $I_{e,1}$	Start and end point of interaction in trajectory
I_z	Intersection zone in RNS
M_p	Number of ego paths
N_o	Number of other vehicles
N_p	Number of components in PMM
N_t	Number of ego trajectories
$O(t)$	Ego comfort over time
$P_{\text{crit}}(s;t,\Delta t)$	Critical probability P_E for Gaussian method
$P_{\text{coll}}(s;t,\Delta t)$	Probability that Gaussian overlap happens
$P_{\text{curv}}(s;t,\Delta t)$	Probability that acceleration is violated in curve
$P_E(s;t,\Delta t)$	Differential probability of critical event
$P(\lambda)$	Poisson distribution with time span λ
Q_{runs}	Probability distribution for simulation runs
$R(t)$	Maximal or accumulated risk over time
R_{SA}	Risk value of survival analysis
$\mathbf{R}$	Rotation matrix for Gaussians
$\hat{S}(t+s;t)$	Homogeneous survival probability
$S(t+s;t)$	Inhomogeneous probability
S_c	Factor in Gaussian function $f_c(x)$
T	Required time gap within IIDM
T_a	Smoothing window of acceleration filter
TH_{2D}	Two-dimensional Time Headway
T_{ext}	Interval for constant velocity in TH_{2D}
$\text{TH}_{\text{stable}}$	Time Headway once distance Δl is constant
$U(t)$	Ego utility over time
W	Window factor of filter for jerk

Latin Lowercase Letters

$a(t)$	Vehicle acceleration over time t
(a_a, s_a)	Assumed acceleration phase for another vehicle
(a_b, s_b)	Assumed braking phase for another vehicle
a_c	Constant deceleration
a_d	Acceleration of driver for stop maneuver
$\widetilde{a}_d$	Driver acceleration when crossing intersection
a_f	Acceleration for follower
$a_{f,2}$	Fixed acceleration for another vehicle
a_m	Constrain in acceleration for IIDM
$a_{p,n}$	Segment accelerations of snake profile

a_y	Lateral acceleration of vehicle
$b_{1,2,3,4}$	Criticality bins (1 - dangerous, 2 - offensive, 3 - uncomfortable and 4 - noticeable driving)
b_{des}	Desired deceleration of IIDM
b_s	Deceleration threshold for safety check
b^c	Driver-specific comfort factor for acceleration
b^d	Utility factor to reach desired velocity in ROPT or RM
b^j	Comfort factor for jerk
b^t	Travel utility factor for velocity > 0
c_i	Velocity uncertainty constant for vehicle i
d_b	Distance at balance point for IIDM
d_{back}	Distance to the back vehicle
d_{front}	Distance to the front vehicle
d_{proj}	Projection distance from vehicle to path
$d_{t:t+s}$	Vehicle distance sequence from time t until $t+s$
d_E	Distance at critical event
$d_{I,1}$	Ego distance to start of traffic intersection
d_P	Ego distance to crossing point of two paths
$e(t+s;t)$	Event density (probability of events per time unit)
$f_{1,2}(x)$	Gaussian function of possible 1D positions x
$f_c(x)$	Gaussian collision function for position
$f_{\text{PMM},i}(x)$	Gaussian function of PMM
$h_g(s)$	Gaussian filter over prediction s
i	Index for any vehicle
j	Index for other vehicles
j_1	Jerk of ego vehicle
k	Index for Gaussian component in PMM
k_{coll}	Damage increase factor for collision
k_{lon}	Slope of sigmoid function $\alpha_{\text{lon}}(s)$
$l_{1,2}$	Longitudinal position of ego and other vehicle
$(\text{lat}, \text{lon}, \text{alt})$	Geographic points from localization device
l_{blend}	Length of blending in lane change
l_c and k	Constants for path blending
Δl	Longitudinal vehicle distance
Δl_h	End of possible path within RNS
m_1	Operating mass of ego vehicle
m_f	Smoothing factor in PMM
n	Number of segments for snake profile
n_{gap}	Number of missed gaps
o	Time offset for selected trajectory in ROPT

p	Politeness factor for IIDM
$\mathbf{p}_{\text{blend}}(l)$	Blended path for lane change with length l
s	Future time
s_d	Duration of single ramp in double ramp profile
s_l	Length of segment in snake profile
s_E	Time until a collision occurs
t	Real time
Δt	Small time interval
t_d	Detection time of crash
t_{gap}	Time gap between two vehicles
t_E	Time of collision
v_0	Starting velocity
$v_{1,2}$	Velocity of ego (1) and another vehicle (2)
$v_{\text{curv},j}(s)$	Maximal curve velocity for another vehicle j
v_d	Desired velocity in ROPT
v_f	Velocity for following vehicle
$v_{f,2}$	Fixed velocity for another vehicle
$v_{\text{low},1}$ and $v_{\text{up},1}$	Lower and upper boundary for ego velocity
$v_{p,n}$	End velocities for n segments in snake profile
v_r^m	End velocity of ramp profile for trajectory m
(v_v, s_v)	End point of velocity and predicted time for fixed acceleration trajectory
(v_s, s_s)	End point for fixed stopping trajectory
v^h	End velocity within RM
v_{tar}	Target velocity in warning
$\mathbf{v}$	Velocity vector in x,y-direction
$w_{i,k}$	Weight factor for components of PMM
$w^*(l)$	Sigmoidal weight for path blending
(x, y, z)	Cartesian coordinates
$\mathbf{x}(t+s)$	Future trajectory function for times $t+s$
x_{rel}, y_{rel}	Ego-relative coordinates
$\mathbf{z}_{t:t+s}$	Predicted sequence of scene states

Greek Letters

α	Exponentiation factor in TTCE or Gaussian method
α_i	Orientation of Gaussians
α_{ego} and α_{other}	Orientation of ego and other vehicle
$\alpha_{\text{lon}}(s)$	Awareness discounting for longitudinal risks
β_{coll}	Steepness factor in collision uncertainty

$\Delta\gamma_s$	Difference angle at interaction start between vehicle pair
δ	Acceleration exponent in IIDM
ϵ	Offset constant in TTCE or Gaussian method
η_{coll}	Damage mid-point for collision
Θ	Angle between ego-relative and absolute coordinates
$\boldsymbol{\theta}$	Optimization parameters for ego trajectory
κ	Curvature of the road
λ_b	Average brake lag for standstill
λ_e	Average engine lag to reach $a_{\max}$
$\lambda_{p,0}$	Adjustable first lag in snake profile in ROPT
$\mu_{1,2}$	Mean positions of vehicles
μ_c	Mean collision point
$\mu_{\text{RSD},1}$	Mean of first velocity distribution from detected risks of RSD
$\boldsymbol{\mu}_i$	Mean position vector
σ	Generalized standard deviation
σ_0	Offset in positional uncertainty
$\sigma_{1,2}$	Position uncertainty in vehicles
σ_c	Normal collision deviation
$\sigma^2_{\text{lon},i}$	Longitudinal position variance for each vehicle i
$\sigma^2_{\text{lat},i}$	Lateral position variance
$\sigma_{l,i}(s)$	Velocity-dependent position deviation
σ_t	Deviation in detection time t_d
$\sigma_{v,i}$	Standard deviation of velocity
σ_R	Deviation in estimated maximal risk $R_{\max}$
$\sigma_{\text{RSD},1}$	Deviation of first velocity distribution from RSD
$\boldsymbol{\Sigma}_i$	Two-dimensional uncertainty matrix
$\hat{\tau}^{-1}$	Time-independent event rate
$\tau^{-1}(\mathbf{z}_t)$	State-dependent event rate
τ_0^{-1}	Constant escape rate
τ^{-1}_{crit}	Event rate of any criticality
τ^{-1}_{coll}	Instantaneous collision event rate
$\tau^{-1}_{\text{coll},0}$	Scale factor in collision rate
τ^{-1}_{curv}	Event rate for losing control in curves

1 Introduction

For many years now, one of the main goals in the field of modern robotics and automotive research has been the automation of the driving task. On average, over 1500 people die in traffic accidents worldwide per day [97]. Traffic fatalities remain one of the leading causes of death. The underlying hope is that technology is able to reduce this high accident rate and prevent near-critical encounters before they even occur. Endless traffic jams might be resolved if there are robot taxis. Also, we would have the freedom to utilize time for other activities, and young and old drivers are able to safely use vehicles at all.

Based on life experience, humans intuitively accomplish the driving task. Given the pictures of a dangerous and a safe car-to-car encounter in Figure 1.1, we can determine which car was at fault and what behavior would have avoided this criticality. In general, the car must keep a sensitive distance to the car in front or drive on the adjacent lane to overtake.

On this note, drivers have an intuitive sense of a car's motion and a feeling for driving risks. They can easily apply traffic rules in this process and determine the risk in different scenarios other than just highway driving. In summary, humans are extremely effective in understanding traffic. For machines, the driving task is, in contrast, not as easy to accomplish.

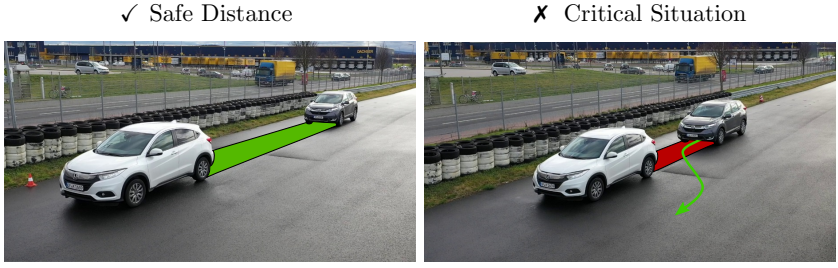

Figure 1.1: Human sense of collision risk. Left: Correct behavior of following vehicle as it keeps a safe distance. Right: Following vehicle creates a dangerous situation and should brake or overtake (see red bar and green trajectory).

1.1 Open Challenges

The ability to self-drive is based on a complicated series of "computing" and "acting" processes. Essentially, the computer must be enabled to sense the environment, make plans based on the situational predictions and act with its own steering wheel, pedals and brake.

Steps of Automated Driving

In a first sensing step, we need to measure the correct *geometrical distance* from the ego vehicle to any other obstacle. Here, we consider the ego car as the vehicle controlled by the system. A GNSS localization device can give us the ego position while perception sensors, such as the camera, radar and lidar, output the position of the other vehicles. In the end, these sensed objects with a rectangular shape in a joint coordinate system represent the traffic.[1] A different size of an object can subsequently change the distance values, and common small errors in the sensors induce inherent noise in the position or false detections.

The planning and acting steps are likewise significant for automated driving. There are many possibilities how the situation might evolve. The other car could make a lane change in the future (i.e., discrete *path choices*) or suddenly brake in the current lane (i.e., *behavior choices*). For two cars and more, the combination of possible situations becomes exponentially high. Accordingly, uncertainties arise not only from sensors, but also from the behavior of the vehicles and unknown environmental changes in the future time. The latter results from the fact that predictions in the far future generally become less accurate.

In this situational space, the planning algorithm must find a proactive and safe maneuver for one's own car that is, e.g., described via a *velocity profile along the map path*, the centerline of the road lane. Therefore, the driving task may involve the prediction and evaluation of different ego maneuvers based on situational risk to work out a single optimal trajectory.[2] Instead of controlling the car directly, the machine can finally also advise the human with recommendations or warnings via the outputs of the planner. This prediction and warning essentially add another dimension in the sense-plan-act cycle, pictured in Figure 1.2.

[1] In the rest of the dissertation, we assume self-driving with an underlying map. Otherwise, maps from perception sensors can be used as well.

[2] We define a trajectory to be a sequence of future vehicle positions with timestamps for each value.

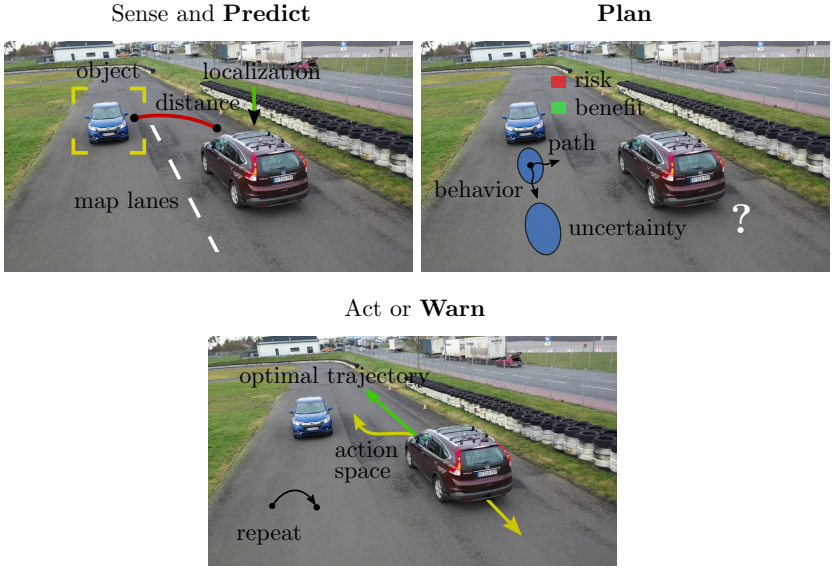

Figure 1.2: Sensing, planning and acting step corresponding to the prediction and warning dimensions. The driving task is complex for autonomous machines and centers around driving uncertainties and related risks.

Current State

Although the mentioned driving task is challenging for a machine, rapid advances in the automated driving industry are being made. Investments in start-ups and research projects of large manufacturers are increasing every year [49]. Due to the high computational power of today's processing units, small embedded computers are able to employ machine learning algorithms for object detection in real-time [141] and iterate over different situational developments with search algorithms [115].

The safety report of Waymo from the year 2018 is promising in that with machines few accidents may be happening per kilometer [130]. Their diverse simulation tests and fleet of car prototypes with the same hardware mark a step towards automated driving in inner-city environments. On the other hand, assistance systems on highways are successfully continuing to relieve stop-and-go and small acceleration tasks for humans in car-following scenarios [53]. These accomplishments show that self-driving is generally possible, in small scale or in specific situations.

Still, the majority of approaches are not transparent in terms of the reasoning behind their driving actions and are hard to evaluate with respect to performance and robustness. In order for humans to accept automated vehicles, the only choice is to prove that these vehicles create no or, at least, fewer accidents. Car manufacturers use scenario-based testing, hardware-in-the-loop or further validation methods to ease trust issues [131]. For instance, Tesla is also attempting to harness the data of their customers to improve the system incrementally [111]. Nevertheless, to completely solve the challenges, several million kilometers need to be driven in practice without inducing errors in the system's components or creating accidents [134]. This strict testing for the validation of self-driving functionalities is time-consuming and hardly possible.

Since there is no driving without dangerous encounters, missing validation is preventing legal and ethical entities in most countries from allowing driving on public roads for systems with higher automation levels (i.e., L2) [25]. Here, we can refer to the SAE automation standard with L1 meaning operation only by a human driver and L5 without even the supervision of the human driver [100]. Overall, there is lack of trust in the safety and transparency of these technologies.

1.2 Contribution

In this dissertation, novel technologies for automated cars and assistance systems are developed based on **driving risk models**.[3] For instance, an ambition is to detect vehicle-to-vehicle accidents early and categorize the criticality of current driving. Then, an optimal behavior can successfully be planned with exact values of the risks and benefits (utility and comfort), depending on the traffic situation.

The proposed risk technologies are always white-box models whose internal states are examinable, in contrast to black-box models. The target is to show that, with a wide range of applications in **simulation, on real data and field tests**, we may be able to resolve the validation issues and tackle current planning problems in automotive safety systems.

Specifically, the risk-based planner covers many situations with a unified set of formulas. The probabilities of several risk causes are therein considered, i.e., risk of collisions with other cars, curve risks for turning sharply and regulatory risks due to priority rules. This reduces the complexity of

[3] The bold expressions highlight the main attributes of the models described in this dissertation.

validation due to the **single system approach**. Both longitudinal following on highways and intersection scenarios in inner cities with different velocity ranges are handled. An example is taking turns at a junction, which includes curve and collision risks.

To develop this further, the risk formulas are easily extendable to other risk causes than the previously mentioned risks. In this dissertation, we will realize that if they are modeled as a probabilistic state-time equation, they could be integrated in the approach. The framework is thus transferable to other mobility domains, such as planes and satellites in aerospace or complex moving, legged robots. For these cases, the coordinate system should be adapted and the prediction of how the velocity evolves over a path differs.

Inherent Motivation

The aspirations to address self-driving cars with risk models can foremost be motivated by the ability to finely balance the risks with benefits. Most common assistance systems are tuned to be very conservative by keeping high distances or driving slowly. By knowing the quantitative impact of the risk based on changing behaviors, the model allows the possibility of going beyond conservative styles to sportive driver modes that focus more on utility or comfort than risk aspects. A **discrete margin on the safety** can still be ensured with a correct parametrization and penalties in the planning.

Conversely, one could classify the currently applied system or human driver by driving style. It will be explained in this dissertation that connecting a risk model based on probabilities and prediction uncertainties additionally allows human driving to be imitated. The personalized system adapts to the preferred style of the user.

An accurate probabilistic risk model with **uncertainty consideration for driving** can be motivated with more short-term applications as well. Quantitative risk analysis enables critical situations to be differentiated from near-critical situations. Besides the time of critical events, the model provides information on the experienced probability and severity with uncertainties in car position and velocity. In situations with multiple cars, we may even filter the obstacles that interact with the driver by analyzing the share of their influence on the risk.

Understanding which situations are frequently dangerous or where risk spots are on roads comprises rich meta-information for navigating in traffic. We might not only use the risk models for planning but may also assess the

performance of self-driving systems for comparison. Additionally, matching a planned maneuver with the actual human driving is powerful to find out what went wrong and **recommend a better option for the driver**. In other words, planning with risk is applicable for driver warning.

Facing Problems

As we could see, the modeled risks in this dissertation enable several applications, which can be summarized as: 1. the situational assessment (i.e., the sensing and predicting step of the driving task), 2. the planning of ego velocities (subsequent planning step) and 3. the support of drivers (action or warning step). This matches to the sense-plan-act cycle of automation, mentioned earlier. Nevertheless, trying to solve the driving task with only a set of analytic equations also poses challenges.

It is not feasible to model every dangerous situation by manually adding potential risk causes. Reinforcement learning, compare, e.g., [109], can learn safe motions via trial and error from large amounts of data and has enabled very complicated scenarios with interactions, including many cars, to be addressed. Still, the basis should always be an analytical model to ensure safety margins, whereby learned models may be added to the risk framework. Another criticism for the approach is that risk is parametrized and does not specify the real probabilities of an accident. The question for costs of a deadly crash is problematic. Ethically, we should not make such a determination objectively [114], but the business of insurances involves exactly this quantification.

Throughout the dissertation, the risk models are parametrized based on recorded driving data and simulation statistics to cover different uncertainties in the real world, and they are applied to driver support in real driving scenarios. The risk does not depict exact costs but tries to capture relative differences between low-risk situations and high-risk situations in a fine-grained way.

1.3 Related Work

The previous introductory Sections 1.1 and 1.2 highlighted that risk models can help to solve the latest driving problems in the car industry. Especially, including uncertainties in the prediction is crucial for taking the current systems to a next level. Hereinafter, we will survey important related work on self-driving technologies in more detail. At first, achieve-

ments of prevalent Advanced Driver Assistance Systems (ADAS) from the automotive industry and of Automated Vehicles (AV) from the IT industry are explained. Since ensuring the safety for ADAS and AV is a major challenge, we will compare their strategies in terms of validation. Finally, this section concludes with literature on previous risk models which the dissertation is based on.

1.3.1 Automotive and IT Industry

If we closely examine the automotive industry and its research community, so far, intelligent driving was realized with the help of a bottom-up strategy. Support technologies have been incrementally introduced and then further improved. The goal is to ultimately achieve fully self-driving cars. Out of this, a range of technologies came into being that tackle multiple aspects of the driving task.

Driver Support

A starting point for safety methods was marked by simple passive systems, which reduced collision outcomes with a seat belt, crumple zone or others. Afterwards, active systems were primarily provided in vehicles, avoiding collisions before they happen [6]. Prominent examples include technologies that are activated once a driver loses car control (e.g., electronic stability systems), and, more importantly, technologies that operate during normal driving [43]. For this dissertation, we will focus on the last type of ADAS and therefore review more examples in the following. Hereby, please note that not an extensive literature overview, but insights of the individual differences are shown.

The first advanced support system was Adaptive Cruise Control (ACC). ACC allows to automatically follow another leading car on the highway. As sensors, radar or camera devices were employed for this purpose [135]. Building on the initial achievements, Lane Departure Warning (LDW) additionally informs the user when the current driving lane is left by accident [54, 75]. At last, we should mention parking assists as well [12]. They are acting around unstructured environments and support drivers in car parking. Concretely, an augmented field of view from the surrounding is offered with, e.g., cameras.

In the course of time, such described solutions have been improved for superior functionality in driving situations. They were either developed to solve further driving situations or for higher levels of support. For instance,

stop and go assists [72] extended the ACC for car-following towards smaller speeds, and Collision Mitigation Systems (CMS) [9] provide warnings in order to perform a full stop in critical situations around urban intersections. With the help of perception sensors inside a parking garage, valet systems [102] can even steer cars into parking spots.

Discussion on Validation

Overall, common assistance technologies successfully facilitate conducting some parts of the driving task under human supervision. In this context, we align the basic systems to the known SAE levels of automated driving, see Table 1.1. While the solutions of LDW and CMS can be considered on the lowest SAE level, ACC products are categorized as technologies in SAE level one. Parking assists, stop and go assists, and parking garages, are eventually going in the direction of higher levels (L2, L3 and L4, respectively). Variants of these technologies are manifold in the automotive market.

However, ADAS solely solve a specific driving situation with system constraints. Combining them in a single unified framework for full automation (i.e., L5) seems complex and infeasible. Starting already on SAE level two, the system has to make independent decisions also in critical situations. Especially, when we consider the problem of ensuring safety in all scenarios, this strategy is unfavorable.

Therefore, in a subsequent main milestone, technical approaches tried to explicitly incorporate the motion prediction of vehicles and use this prediction knowledge to infer a safe action. Notably, an intelligent ver-

Table 1.1: The SAE levels assign different shares of the driving task to Driver (D) and Vehicle (V). We distinguish between situation monitoring, execution of plans and fallback solutions. Table was taken from [27].

SAE level	0	1	2	3	4	5
Degree of Automation	*None*	*Assisted*	*Partial*	*Conditional*	*High* (some modes)	*Full*
Examples	LDW, CMS	ACC	Parking assist	Stop and go assist	Parking garage	Robot taxi
Monitoring	D	D	D	→ V	V	V
Execution	D	→ D+V	→ V	V	V	V
Fallback	D	D	D	D	→ D+V	→ V

sion of ACC [104] detects neighboring vehicles that will cut into one's own driving lane, and brakes in advance to smoothly react on the environment. Normal ACC would only be able to reduce the speed when another vehicle is already on the lane. Further interesting studies apply map geometries from navigation systems and enhance the fuel-efficiency of the car in curves [90]. Before an upcoming turn, the engine management changes its characteristics for a seamless braking behavior. Instead of reactive logics, those systems can proactively support the driver.

In this dissertation, the aspiration of earlier works in ADAS are shared and prediction and uncertainty considerations are improved with driving risk models. It will be demonstrated that this leads to a holistic approach, in the sense that the support can be provided in one system for different driving situations. Risk models can help on all SAE levels and the systems are rendered more flexible and suitable for validation.

Self-Driving Cars

The seemingly difficult combination of ADAS led the IT industry attempt to solve the self-driving problem in a top-down strategy instead. Their aim was to directly achieve Automated Vehicles (AV) with SAE level five. In this context, the DARPA Grand and Urban Challenges [10, 11] generated a motivational spur for the fields of robotics and artificial intelligence. Participating test cars could self-drive in constrained environments. Equipped with high-cost sensor devices, an integrated software was developed in the sense-plan-act paradigm.

The authors of [115], for instance, detected obstacles with lidar sensors, planned driving paths via search algorithms and put plans into action with a velocity and steering controller. Similarly, the work from [35] used, in total, GNSS, radar and lidar for sensing, trajectory optimization for planning and a Model-Predictive Control (MPC) for acting. From these achievements, Waymo [130] was able to provide small fleets of robot taxis in certain road areas, which take passengers without a safety driver from a start to an end point. Although AV constantly have fewer disengagements, traffic accidents are not entirely preventable [68].

At this point, we want to analyze further work in research. Because of the breakthroughs in machine learning, from this time onward, many more car modules relied on data-driven approaches. An end-to-end architecture was trained to derive the applicable steering angle from front images, effectively following free single-lane streets [7]. The solution does not work well with other dynamic vehicles; therefore, discrete driving actions were

learned for, e.g., stopping and accelerating, using a Long-Short Term Memory (LSTM) network [138]. Simultaneously, reinforcement learning [109] gained popularity for imitating cooperative behaviors with trial and error in many simulations. However, deep learning is not convincing to improve on motion planning yet but can successfully be applied to detect objects within sensor processing [110].

A big issue for AV and, especially, for learning-based methods is that testing of safety is even harder, because they are becoming black-box models. One cannot see the internals and derive the behavior for unseen data. Related work connected areas in images, which are relevant for action [61], but there is no quantification of induced risk.

In this line of thought, the dissertation at hand draws inspirations from the common sense-plan-act steps of self-driving vehicles. Probabilistic risk models are developed that form a compromise between rule-based ADAS on the one side, and data-driven AV with machine learning on the other side. This represents a safety by design concept, as the underlying models relate to the accident costs. The different steps are integrated into a single system, similar to an end-to-end approach, but we can still look into its internal variables. For this reason, the system is defined as a white-box model.

1.3.2 Risk Maps

The related work section is now finalized by stating the difference of the proposed risk models to the previous work on special probabilistic risks. Particularly, we analyze research by F. Damerow, S. Klingelschmitt and J. Eggert on risk and prediction models [19–21, 26–28, 62]. This dissertation builds upon their achievements.

As one of the original publications, the foundations of a survival analysis for driving risks were introduced with [26]. This analysis is a popular choice in financial domains and uses a Poisson process for describing critical events. To the knowledge of the author, it was the first time used in the automotive domain. The works of [19, 20] successfully derived a motion planner out of the risk model, called Risk Maps (RM). RM search and visualize multiple ego trajectories with their risks to other encountered vehicles. Computationally less costly planners were additionally presented in, e.g., [28]. Both could tackle car-following and drive on intersection scenarios to some extent. For handling situations robustly, [21] then proposed the planning of motions for each other car and used them to infer an ego motion of its own, including all plans. In comparison, with the usage of

[62], the other future behaviors can also be predicted, in this process, with machine learning. Finally, an overview of risk-based planners was given in the literature [27].

Proposed Model

Throughout the dissertation, the listed works on driving risks are utilized and enhanced. Concretely, the dissertation applies risk models not only in planning but also in predicting (e.g., accident detection) and warning (i.e., driver support) tasks. The major theoretical contribution is the detailed modeling of driving uncertainties. Gaussian distributions model uncertainties in the position and velocity of the vehicles. Combined with the survival analysis, a general risk model is achieved that offers superior results and appealing practical properties.

Furthermore, motion planning is done in a more efficient manner. Parametrized trajectories are essentially found based on a risk-dependent cost function with non-linear optimizers. All proposed methods will be compared with baseline models and their performance and robustness improvement is shown. Hereby, a focus of this dissertation is the real-world application in test cars. The models will be employed in many simulations and with prototypes, having online sensors.

1.4 Outline

In summary, driving risk models are presented for the sense-plan-act cycle or rather the predict-plan-warn sequence of automated driving. The upcoming chapters are structured accordingly: we start with the predicting task in Chapter 2, continue with the main planning model in Chapter 3 and Chapter 4 finishes off with the warning application for the driver. All chapters can be read independently, however, reading them in order is recommended.

The main theme, which reappears in the dissertation, is the formulation of the risk models for self-driving cars or driver support systems. It is argued that these models improve on related work.

In detail, *Chapter 2* lays the foundation of the risk framework by introducing prediction and criticality evaluation over the continuous future time. Given a finite number of traffic participants, we can determine whether the current driving encounter (i.e., relative positions and velocities) leads to an accident or not. Additionally, based on a longer stream of situation data we can filter out at what point other vehicles may come

close to the ego driver. The content of this chapter is based on the IEEE journal article of Puphal, Probst and Eggert [92] and to some extent on work by Eggert and Puphal published in JSAE [29].

In *Chapter 3*, we go one step further and use the risk model with ego benefits as a scalar cost function for planning. The goal is to find a velocity profile along the likely map path that considers tactical options, such as to first brake and then accelerate again. It also incorporates prioritized vehicle-to-vehicle interactions. Using an optimization method, we can always find a solution that minimizes risk and maximizes utility and comfort for the situation from the ego perspective. The chapters are based on two conference papers from Puphal et al. [93, 94], which both were presented at IEEE conferences.

As the last component for automated driving, in *Chapter 4*, the risk-based planning for warning is realized as support systems in real recordings and car prototypes. We transform a computed low-risk trajectory into understandable goal states for the user and also attempt to tackle the research question of how to warn of upcoming risks with map data, traffic objects (e.g., crosswalks) and other vehicles. The purpose is to show that the risk models can work in real-time for real driving scenarios, such as driving at intersections or merging on a highway. Parts of the chapter are based on the IEEE conference paper of Puphal, Flade and Eggert [91] and on work from the EU project Vision-Inspired Driver Assistance System (VI-DAS) [125].

Chapter 5 ultimately summarizes the findings of this dissertation and explores future research areas that might benefit from using the methods.

2 Predicting Risks under Uncertainty

This chapter introduces the driving risk models which are used throughout the dissertation. As the first key component within the sense-plan-act cycle of automated driving, risk prediction allows for situational assessment.

We strive for a measure that can distinguish between safe and critical driving. In technical terms, risk is defined as the probability that a critical event might occur weighted by its severity, i.e., its potential consequences in terms of damages and injuries [120]. But, as already indicated, the main problem of risk is its predictive character involving vehicles' uncertainties, which spread over time and must be modeled in a sensible way.

For simplicity, only risks from collisions between interacting vehicles are considered first. These are the most complex risk sources. Later on, we will handle other risk sources as well (see Chapters 3 and 4). A focus is put on applications of accident detection and hazard classification, while comparing a novel method with state-of-the-art approaches. In more detail, the research questions tackled in this chapter are:

1. **Baseline Models**: Which common approaches exist for risk models? What are the key differences in these metrics? How and with which data can they best be evaluated?

2. **Uncertainty**: What uncertainties are occurring in normal driving and should be included in the risk measures? Is there an underlying measure that incorporates or generalizes over other approaches?

3. **Critical Situations**: Which traffic situations lead to collisions? Is car-following or intersection crossing more challenging to model with risks? More specifically, can specific driving categories be filtered depending on their risk level?

Hereby, the target is always to develop a solid theoretical risk model for driving, which is suitable for the validation and planning of self-driving systems. The various conducted experiments should support its superiority and holistic nature.

In the remainder of this chapter, we first elaborate on related work for general risk prediction. Then, single model-based risk measures are chosen and compared by evaluating their strengths and weaknesses qualitatively and by testing them on a set of car-to-car accident scenarios. Based on this evaluation, a novel risk model is proposed that probabilistically incorporates uncertainties as they arise in real-world conditions. We will make it robust for different driving conditions, such as turning at an intersection. Finally, this chapter is concluded by applying the models on classifying risks in normal driving.

2.1 Related Work

In previous related research work, numerous approaches for prediction and risk assessment have been introduced [67]. The major fields engaging in risk estimation methods are, e.g., the automotive industry [2, 13, 18, 34, 48, 52, 59, 71, 105, 117, 124, 127, 128, 132], robotics [26, 47, 65, 82], aviation [46, 79, 103] and aerospace technology [37], civil engineering [112], data science [7, 55, 62] and economics [142]. In such works, risks are often used for motion planning in a cost function and help to react on hazards.

Generally, it is possible to calculate risks by detecting hazardous driver intentions. For example, the steering behavior of vehicles is learnable using image streams under a convolutional neural network [7]. By comparing the measured with learned wheel angles, the deviation can be assumed to correlate with current risks. Due to the fact that solely the road structure influences the behavior, the resulting risk value indicates, however, risks from sharp turns and not collision risk.

As an alternative, a Bayesian network is employed in [62] to classify car behaviors into typical maneuvers, i.e., drive straight or turn. Training the network with accident data could lead, e.g., to maneuver detections of vehicles violating traffic rules. Nevertheless, risks cannot be identified for situations that are not in the dataset.

Those risks on the behavioral level are based on machine learning algorithms, and their accuracy is highly influenced by the data. Since freely available data for risky driving is sparse, model-based risks are promising. These models are white-box systems, which are more favorable for the validation of driving systems. Risk models can be divided into time-based, probabilistic and shape-based risks, which are described in the following. In the end, we decide on three risk measures to be examined closely in the theoretical Section 2.2.

Time-based Risk Indicators

Risk measures based on time heuristics consider future vehicle trajectories with kinematic models. The Time-To-Collision (TTC) [124] is one example of Time-To-Event (TTX) indicators and defined as the deterministic time until the trajectories of two vehicles intersect. For the prediction, a constant velocity model is implicitly given in the equations. Variants of this TTC incorporate different velocity profiles, such as constant acceleration models [132].

Since TTC only considers vehicle dynamics for longitudinal following, its equations have been extended for 2D operation, e.g., in [128]. To overcome the collision assumption in TTC and to have realistic values in non-crash scenarios, the orientation of the vehicles is taken into account. Both drawbacks of TTC are also handled with the Time-To-Closest-Encounter (TTCE) [19] and, similarly, the concept of Kamm's circle [127] is used as positional uncertainty to derive a "Worst-TTC" as the maximal risk.

To even consider kinematic constraints of the acting vehicles, Time-To-Brake (TTB) and Time-To-Steer (TTS), compare [48], analyze the time remaining for an emergency braking or steering maneuver to still avoid the longitudinal crash. Depending on the actual behavior of the vehicle (constant velocity, braking or steering), TTB and TTS will render the risk better or worse than TTC.

A related indicator of time-based risks is lastly the Time Headway (TH), which describes the time until a collision in car-following, taking into account ego kinematics only. Post-Encroachment Time (PET) [2] generalizes this notion to intersections. Due to the simple calculation and intuitive interpretation of TH as well as TTC, the driving assistance function Adaptive Cruise Control (ACC) controls TH, and most Collision Mitigation Systems (CMS) take TTC into account. Here, ACC could already be verified with a formal method to be stable on highways [71].

Probabilistic Risk Prediction

Alongside these time indicators, probabilistic risk prediction presumes critical events to be dependent on probability distributions. Risks can thus incorporate uncertainties in the prediction.

Gaussian methods, such as [37, 82], exemplarily model predicted trajectories of moving entities with spatial normal distributions and calculate their overlap as a collision probability. A more sophisticated method is [65], in which the distribution variances are separated into longitudi-

nal and lateral components and instead of integrating analytically, Monte Carlo simulation gives an approximation of the overlap. In this process, the Gaussian parameters can also be taken from the covariance matrix of a Kalman Filter. With this in mind, [52] extrapolates Constant Yaw Rate and Acceleration (CYRA) and [47] employs a Particle Filter with Dirac delta functions as a distribution instead.

For vehicles, planes and satellites, it is reasonable to assume the movement along fixed paths due to their inflexible steering. In contrast, amongst others, mobile robots and pedestrians have very variable actions in space [59]. This leads to larger driving uncertainties. To determine the common measurement uncertainties of cars, [13] inspected the errors in position, velocity, heading and acceleration from real sensors and found parameter values for describing multi-modal Gaussians.

For this dissertation, the most important risk model constitutes the survival analysis from [26]. Specifically, with Poisson processes, the probability of remaining accident-free is calculated out of the mean time between critical events, which itself is dependent on the cars' future trajectories. The risk measure follows as the complement of the corresponding survival function and allows to consider not only collision risk but also, for instance, the risk for losing control in curves.

Opposed to calculating an analytical solution for event probabilities, Monte Carlo strategies [105] are widespread. They approximate risks by sampling positions from distribution functions and comparing the number of collisions with misses. This is practicable when the direct solution is complex but requires high computational costs for reliable estimates. For example, [65] apply Monte Carlo for simulating collisions of wheeled robots with specific shapes. Similarly, [103] improve the convergence time by importance sampling in three-dimensional ranges of motion.

Risks from Car or Road Shapes

Finally, if the future behavior is known or easier to estimate, collisions can be checked discretely as well. We distinguish between online models for driver support, which calculate the risks before they happen and ex-post models for analyzing risks on roads after they have occurred.

In one work, traffic participants are projected onto map data paths and predicted longitudinally using prior knowledge (e.g., stopping at a stop line or driving at constant velocity) [34]. Afterwards, a possible crash is determined with intersection checks of geometrical shapes around the assumed positions. Closely related, [18] detect deviations from assumed paths with

a Gaussian process, as well as intentions (e.g., brake or accelerate) via Hidden Markov Models (HMM), and [46] create in the context of Dynamic Probabilistic Risk Assessment (DPRA) discrete cells in spatio-temporal state-space to trace back faulty maneuvers or system behaviors.

As mentioned, critical situations can be analyzed after their occurrence with ex-post scores for the purpose of, e.g., traffic flow management.

Approaches in this direction include kriging techniques [112], which allow creating incident heat maps by extrapolating accident ratios from road locations with data to segments where no data is available. Similarly, [55] derive risk spots on map data from recorded acceleration and jerk profiles of cars, and airplane risk zones are constructed with the probability of one point colliding into another volume using the rice formula in [79]. Furthermore, the authors of [142] optimized the structure and parameters of a Bayesian network from accident data causes (weather, inattention, lane change, etc.).

Despite of all these works, a theoretically grounded and generalizable risk measure for automated driving is missing. In this dissertation, the focus is hereby put on time-based and probabilistic risks from [19, 26, 82]. In particular, we derive the model-based risk measures TTCE, Gaussian method and survival analysis and compare them in simulations with the purpose of analyzing their properties for an application on driving data. Each measure is representative of a broader family, and we examine their ability to consider uncertainties. As it turns out, the most sophisticated model combines the Gaussian method with the survival analysis and includes extensions for realistic uncertainty handling.

2.2 Theory of Risk Models

This section outlines the properties of a suitable risk measure and describes the general steps within a framework for its calculation. It then gives a detailed mathematical introduction of the three chosen risk measures [19, 26, 82], highlighting their differences.

As a starting point, we consider dynamic collision scenarios (i.e., a traffic scene with two traffic participants TP1, TP2) at an arbitrary moment in time t. Beginning at t, the target is to estimate the risk of a critical event that could happen at a future time $t+s$, i.e., which is at temporal distance s in the future. We assume the events to be disruptive and to have no duration, so that they can be fully characterized by their time $t+s$, if they happen at s into the future.

Since most of the commonly used risk measures do not address severity explicitly, we will concentrate on risk as an event occurrence probability. However, severity will be included in the argumentation for motion planning in Chapter 3 in a straightforward way. An indicator for risk is the probability function $P_E(s;t,\Delta t)$ that a critical event will happen during an interval of size Δt *around a future time* $t+s$. As a probability, it should be positive and appropriately normalized so that $P_E(s;t,\Delta t) \in [0,1]$. With a collision of time t_E (*time until* a collision occurs s_E), if a collision is imminent (at $s_E \to 0$), it should be $\lim_{s\to 0} P_E(s;t,\Delta t) \to 1$ and if no collision ever occurs, it should be $\lim_{s\to\infty} P_E(s;t,\Delta t) \to 0$.

A compact risk measure $R(t)$ would comprise, for each point in time t, the entire **accumulated expected future risk** contained in $P_E(s;t,\Delta t)$, $s \in [0,\infty]$. There are several ways to gain such a measure, e.g., by extracting the maximal expected risk

$$R(t) := \max_{s>0} P_E(s;t,\Delta t) \tag{2.1}$$

or using an appropriately accumulated risk

$$R(t) := \sum_{s=0}^{\infty} P_E(s;t,\Delta t) \quad . \tag{2.2}$$

Please remind that in both equations the severity of risk is neglected.

The latter is a cleaner form because it comprises the *full future* event probability, but it requires a more careful derivation due to proper normalization considerations. Additionally, heuristically motivated risk measures often directly estimate $R(t)$ without $P_E(s;t,\Delta t)$ (see Section 2.2.1). Again here, it should be guaranteed that $R(t) \in [0,1]$ and $R(t) \to 1$ for a collision at $s_E \to 0$ and $R(t) \to 0$ for no collision at any time.

Framework for Risk Prediction

A risk prediction framework consists of three components, depicted in Figure 2.1. In a first step, a prediction of how the situation will evolve in the future is performed. In our notation, designating $\mathbf{z}$ as the state vector of a scene, the predicted sequence of future scene states is given by $\mathbf{z}_{t:t+s}$. The prediction can hereby be modeled at different levels of detail depending on the geometry, kinematics and interaction. For example, a low-level prediction would treat vehicles as dynamic entities and use constant acceleration, velocity or turn-rate assumptions in kinematic equations. On the next level, road geometries could be taken into account to constrain the paths on which vehicles can drive. The outcome of the prediction is a

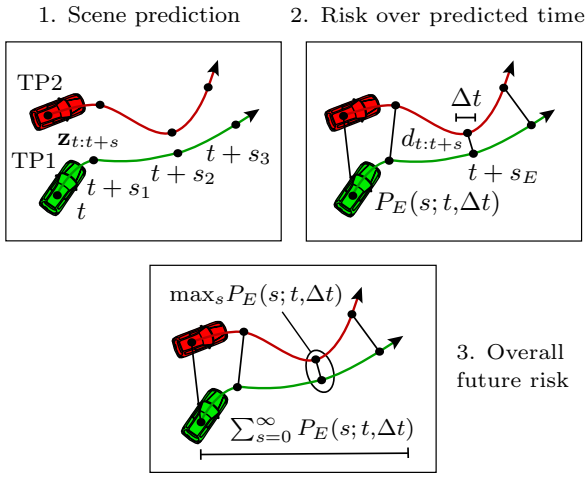

Figure 2.1: Risk framework for collision prediction between two traffic participants. We predict the driving scene in 1, derive risk indicators (e.g., distance for collisions) over future time in 2 and either take the maximum or compute an integration to obtain a single risk value in 3.

selected set of possible future trajectories for each involved vehicle.

In a second step, the time evolution of the scene $\mathbf{z}_{t:t+s}$ is evaluated in terms of criticality by extracting features that are indicative of risk. For collision risk, the predicted trajectories are compared for each point in predicted time s to obtain the spatio-temporal proximity $d_{t:t+s}$ between the vehicle trajectories. This leads to an **instantaneous risk** function or to features, e.g., of time to and distance at the point of maximal risk, resp. the time until the event s_E and the predicted proximity d_{t+s_E} at that time. Afterwards, the event probability $P_E(s;t,\Delta t)$ can be calculated from the instantaneous risk function. The third step comprises retrieving the risk measure in the form of a **scalar risk function** $R(t)$ according to Equation (2.1) or (2.2).

Now, we begin by describing the traditional heuristic Time-To-Event (TTX) type measures, in particular Time-To-Contact, which is extended towards 2D operation to retrieve the Time-To-Closest-Encounter (TTCE). The second introduced risk measure models position uncertainty with a Gaussian distribution and uses spatial occupancy probabilities for collision risks. Finally, we derive a risk measure based on so-called "survival" conditions, which includes uncertainty in the driving history.

2.2.1 Time-To-Closest-Encounter

The family of TTX-based risk measures represents proximal safety indicators based on the time left until a critical event. Particularly, the well-known TTC is defined as the time remaining until two vehicles will engage in a collision if they continue driving along the same path according to some prediction models [124].

A common assumption is that both traffic participants (TPs) drive with constant velocities $v_{1,2}$. In this case, if they start driving at time t with longitudinal positions $l_{1,2}$, the time of collision/time of critical event will be at $s_E = \mathrm{TTC} = -\Delta l/\Delta v$, whereby $\Delta l := l_1 - l_2$ and $\Delta v = v_1 - v_2$. However, TTC is rather limited for complex scenes, because (i) it is only applicable to 1D scenarios due to the specified longitudinal variables in the equation and (ii) it presumes that a collision will happen, so that near-crash cases cannot be evaluated. Several extensions of TTC to 2D have been developed [128] but generally lack justification from theoretical and empirical side.

For deriving a risk measure with TTC, the heuristic assumption is made that the overall risk of a predicted collision at a time s_E into the future decreases with increasing temporal distance to the event if nothing changes. When there is more time left until the incident, there is a higher chance for other things to happen, which might lead to a different future evolution of the scene and to an avoidance of the event. An in-depth explanation is described later in this section. The most straightforward approach is then to calculate s_E in order to get the risk $R(t)$ according to

$$R_{\mathrm{TTC}}(t) \sim \left[\frac{1}{s_E}\right]^{\alpha}$$

with $\alpha > 0$. To avoid divergence and to fulfill the normalization conditions from the last section, we introduce a small constant ϵ and a steepness constant D_c so as to retrieve

$$R_{\mathrm{TTC}}(t) := \left[\frac{\epsilon}{\epsilon + D_c\, s_E}\right]^{\alpha}. \tag{2.3}$$

Critical events like passing-by at high velocity cannot be captured by Equation (2.3). A logical refinement is given by considering not only collision events but generally the future events of the highest criticality as a temporal reference. In the case of collision risk, this leads to the time of closest proximity $s_E = \mathrm{TTCE}$ (see Figure 2.2). We split the TTCE-dependent risk into two factors: One that handles the temporal decay and

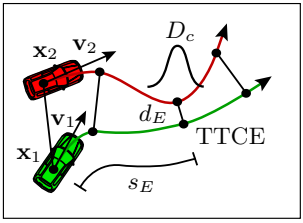

Figure 2.2: Collision risk prediction with TTCE. The time where vehicles come the closest to each other is a heuristic indicator that correlates with the collision probability.

another one that accounts for the increased spatial collision danger in case of high proximity. In this way, we obtain

$$R_{\mathrm{TTCE}}(t) := \left(\frac{\epsilon}{\epsilon + D_c\, s_E}\right)^{\alpha} \exp\left\{-\frac{d_E^2}{2\sigma^2}\right\} \tag{2.4}$$

with the Euclidean distance $d_E := d(t + s_E)$ between the vehicles at the moment of the critical event and a Gaussian variance σ^2.

Relation to Driving Uncertainty

In case of a collision, we would obtain $d_E = 0$, and the spatial term reduces to 1 so that we get back to $R_{\mathrm{TTC}}(t)$. In case of a near-crash incident, the spatial term can be used to quantify how critical the incident was in terms of spatial proximity, accounting for uncertainty in the predicted positions. As a further modification, we model the fact that the uncertainty in positions increases with larger prediction times s_E (see also Section 2.2.2). This means that for events which lie further away in the future, we will consider larger σ with $\sigma(t + s_E) := D_c\, s_E$, so that

$$R_{\mathrm{TTCE}}(t) := \underbrace{\left(\frac{\epsilon}{\epsilon + D_c\, s_E}\right)^{\alpha}}_{\text{temporal uncertainty}} \underbrace{\exp\left\{-\frac{d_E^2}{2D_c\, s_E}\right\}}_{\text{spatial uncertainty}}. \tag{2.5}$$

With a prediction model for the 2D trajectories $\mathbf{x}_{1,2}(t + s)$ of two TPs, we can directly calculate $s_E = \mathrm{TTCE}$ as well as d_E at that time. In case of constant velocity assumptions, i.e.,

$$\mathbf{x}_{1,2}(t + s) = \mathbf{x}_{1,2}(t) + \mathbf{v}_{1,2}\, s,$$

we gain for the time course of distances

$$d(t + s) = \|\Delta\mathbf{x}(t) + \Delta\mathbf{v}\, s\| = \sqrt{[\Delta\mathbf{x}(t)]^2 + 2\Delta\mathbf{x}\Delta\mathbf{v}\, s + (\Delta\mathbf{v})^2\, s^2}$$

which has its minimum at the predicted time

$$s_E = -\frac{\Delta\mathbf{x}(t)\,\Delta\mathbf{v}}{\|\Delta\mathbf{v}\|^2}$$

and which reduces to TTC in the longitudinal case. The distance of closest proximity is finally

$$d_E = \sqrt{[\Delta\mathbf{x}(t) + \Delta\mathbf{v}\, s_E]^2} = \sqrt{\left\{\Delta\mathbf{x}(t) - \frac{\Delta\mathbf{v}\,[\Delta\mathbf{x}(t)\,\Delta\mathbf{v}]}{\|\Delta\mathbf{x}\|^2}\right\}^2}$$

$$= \sqrt{\left[\frac{(\Delta\mathbf{x}(t)\times\Delta\mathbf{v})\times\Delta\mathbf{v}}{\|\Delta\mathbf{v}\|^2}\right]^2} = \|\Delta\mathbf{x}(t)\|\,|\sin[\angle(\Delta\mathbf{x}(t),\Delta\mathbf{v}]|. \qquad (2.6)$$

As a result, using d_E from Equation (2.6) and inserting it into the spatial term of Equation (2.4), the outcome is a risk measure that generalizes TTC to the 2D as well as to non-collision cases. Figure 2.2 summarizes TTCE and its input values.

2.2.2 Gaussian Method

As a second family of collision risk indicators, we consider approaches based on spatial occupancy probabilities [82]. For this purpose, the normalized probability densities for the spatial positions of two traffic participants (TPs) indexed 1,2 are described by Gaussian functions

$$f_{1,2}(x) = \frac{1}{\sqrt{2\pi\sigma_{1,2}^2}}\,\exp\left[-\frac{(x-\mu_{1,2})^2}{2\sigma_{1,2}^2}\right]$$

with the mean positions $\mu_{1,2}$ and variances $\sigma_{1,2}$.[1]

A collision at a position x then occurs if both TPs coincide at the same position. Consequently, a way to quantify the likelihood of a coincidence/collision at a common position x is

$$f_c(x) := f_1(x) f_2(x).$$

where f_1 and f_2 are the Gaussians of TP1 and TP2. Since their product is again a Gaussian function, we get

$$f_c(x) = \frac{S_c}{\sqrt{2\pi\sigma_c^2}}\,\exp\left[-\frac{(x-\mu_c)^2}{2\sigma_c^2}\right]$$

[1] The univariate distribution is described, but generalizations to cover 2D positions are analog.

with the single terms of

$$\frac{1}{\sigma_c^2} = \frac{1}{\sigma_1^2} + \frac{1}{\sigma_2^2}, \mu_c = \frac{\mu_1\sigma_1^2 + \mu_2\sigma_2^2}{\sigma_1^2 + \sigma_2^2} \text{ and}$$
$$S_c = \frac{1}{\sqrt{2\pi(\sigma_1^2 + \sigma_2^2)}} \exp\left[-\frac{(\mu_2 - \mu_1)^2}{2(\sigma_1^2 + \sigma_2^2)}\right].$$

The Gaussian position probability densities encompass the final positions of all possible trajectories that lead to points x at a certain moment in time $t+s$. The probability that the first TP, driving along its trajectory, is hit by the second TP is eventually given by spatially integrating $f_c(x)$ over all positions where the first TP can be:

$$P_E(s; t,\Delta t) \sim \int_{\infty} f_c(x)\, dx = S_c. \tag{2.7}$$

From Collision Probability to Risk Measure

So how do we get from the collision probability (2.7) to a risk measure as a function of the time t similar to the one introduced in Equation (2.4)? It is assumed that the moving TPs follow a trajectory which undergoes certain variations in speed and heading. This accounts for mean positions through time $\mu_{1,2}(t+s)$ with spatial uncertainties $\sigma_{1,2}(t+s)$. For $\sigma_{1,2}(t+s)$, a simple Brownian motion diffusion model with a linear increase of uncertainty starting at $\sigma_{1,2}(t) = 0$ is used according to

$$\sigma_{1,2}^2(t+s) := D_{1,2}\, s. \tag{2.8}$$

When we insert Equation (2.8) into (2.7), we obtain

$$P_E(s; t,\Delta t) \sim \frac{1}{\sqrt{2\pi D_c\, s}} \exp\left\{-\frac{[d(t+s)]^2}{2D_c\, s}\right\}$$

with a joint diffusion constant $D_c := D_1 + D_2$ and an "expected" distance $d(t+s) := \|\mu_2(t+s) - \mu_1(t+s)\|$. Furthermore, to satisfy the normalization requirements, we add a small constant ϵ to the first term and thus gain

$$P_E(s; t,\Delta t) \sim \left(\frac{\epsilon}{\epsilon + D_c\, s}\right)^{\alpha} \exp\left\{-\frac{[d(t+s)]^2}{2D_c\, s}\right\} \tag{2.9}$$

with constant $\alpha = 1/2$.

Equation (2.9) depicts the probability that two TPs will be at the same position within a future interval around $[t+s]$, starting at t and assuming Gaussian distributed positions around the TPs mean positions $\mu_{1,2}$. It can be seen that for larger prediction horizons s the overall collision likelihood decreases, because of the larger uncertainty in the TPs positions. The square-root dependency $\alpha = 1/2$ is hereby a direct consequence of the Gaussian approach and the diffusion spread of the spatial probability densities. Nevertheless, in a scene with allowed and non-allowed areas (road/non-road) where every time a TP leaves the road area, it will already deviate from its "normal" prediction so that the collision probability will exhibit a faster decay with s.

From Equation (2.9), we can now define a risk measure. As a conservative approach, we extract the maximal encountered future event probability as a risk indicator with

$$s_E := \text{argmax}_s \, P_E(s; t, \Delta t)$$

and retrieve for the Gaussian risk indicator

$$R_{\text{Gauss}}(t) := P_E(s_E; t, \Delta t)$$

or rather

$$R_{\text{Gauss}}(t) := \left(\frac{\epsilon}{\epsilon + D_c \, s_E} \right)^{\alpha} \exp\left\{ -\frac{d_E^2}{2\sigma^2} \right\}, \tag{2.10}$$

whereas d_E is again the Euclidean distance at the moment of the critical event $d_E = d(t + s_E)$.

In Figure 2.3, the growing Gaussian position distributions are pictured, whereas the overlap between both Gaussians at the distance d_E is taken for the risk $R_{\text{Gauss}}(t)$. Equation (2.10) has the same form as Equation (2.5). However, the difference is the calculation of s_E, which we estimate here via the maximized $P_E(s; t, \Delta t)$ and which occurs directly with TTCE in Section 2.2.1. For a small diffusion constant $D_c \to 0$ or imminent collisions

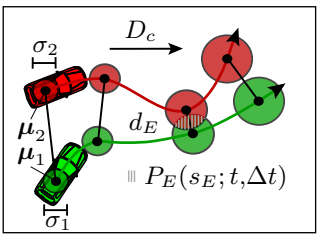

Figure 2.3: Collision risk prediction with the Gaussian method. Positional uncertainties are visualized around the vehicles, whereby the Gaussian overlap is taken as collision probability.

with $s_E \to 0$, the two approaches become equivalent and thus have same output values.

2.2.3 Survival Analysis

In Sections 2.2.1 and 2.2.2, we have seen two standard risk measures that try to tackle some challenges in prediction. In particular, dealing with uncertainty over time and space, as well as a generalization to 2D, enables the use of risk measures in a broader range of situations. The two measures could be aligned in a general framework for risks. However, neither TTCE nor the Gaussian method provide a solid theoretical explanation, but rather motivated heuristics.

In this section, a risk measure is described based on a grounded approach for the statistics of rare events and first passage time problems. This risk measure is also able to deal with uncertainties, but differently to the previously presented ones, it provides full interpretability in terms of (normalized) probabilities, and it considers additionally the situation history.

Accident occurrences are sometimes modeled as a thresholding process based on a Poisson-like event probability [26]. For an exemplary vehicle, in a sufficiently small time interval of size Δt, the so-called **instantaneous event probability** is characterized by an event rate $\hat{\tau}^{-1}$ (units: events/sec) according to

$$\hat{I}_{\text{event}}(\Delta t) := \hat{\tau}^{-1}\Delta t \quad . \tag{2.11}$$

The term instantaneous event probability denotes that this probability does not (yet) take the history into account.

The **survival probability** function $\hat{S}(t+s;t)$ indicates the probability that the vehicle will survive from t until $t+s$, i.e., that it will not be engaged in an event like an accident. From Equation (2.11), we can directly derive the survival probability after a small time interval Δt if the survival probability at t' was $\hat{S}(t';t)$

$$\hat{S}(t'+\Delta t;t) = \hat{S}(t';t)\,[1-\hat{\tau}^{-1}\Delta t]$$

so that with the starting condition $\hat{S}(t;t)=1$ we get

$$\hat{S}(t+s;t) = \exp\{-\hat{\tau}^{-1}s\}, \tag{2.12}$$

which describes the **homogeneous survival probability** for constant rates $\hat{\tau}^{-1}$.

The real risk event modeling occurs by a proper parameterization and variation of a time-varying $\hat{\tau}^{-1}(t)$ resp. its state-dependent analogous function $\tau^{-1}(\mathbf{z}_t)$ with $\hat{\tau}^{-1}(t) := \tau^{-1}(\mathbf{z}_t)$. In $\tau^{-1}(\mathbf{z}_t)$, we include all the risk factors with the context information in the state vector $\mathbf{z}_t$. Correspondingly, in dangerous situations, the event rate is higher than in harmless situations. For temporally varying $\hat{\tau}^{-1}(t)$, Equation (2.12) modifies to

$$\hat{S}(t+s;t) = \exp\{-\int_0^s \hat{\tau}^{-1}(t+s')\,ds'\} \quad .$$

The states have been left out here for the simplicity of the given derivations. If included back, we acquire the state-dependent survival probability function

$$S(t+s;t,\mathbf{z}_{t:t+s}) = \exp\{-\int_0^s \tau^{-1}(\mathbf{z}_{t+s'})\,ds'\} \tag{2.13}$$

which defines the probability that a vehicle survives during $[t,t+s]$ without being involved in a critical event and which depends on the entire state vector sequence $\mathbf{z}_{t:t+s}$.

From Survival Probability to Risk Measure

To quantify the risk of an accident between the ego car and another car, the instantaneous **collision event rate** $\tau_{\text{coll}}^{-1}(\mathbf{z})$ is modeled by

$$\tau_{\text{coll}}^{-1}(\mathbf{z}) = \tau_{\text{coll},0}^{-1}\, e^{-\beta_{\text{coll}}\,|\mathbf{x}_1-\mathbf{x}_2|}$$

with constants $\tau_{\text{coll},0}^{-1}$ and β_{coll}. While the scaling factor $\tau_{\text{coll},0}^{-1}$ is chosen so that $\hat{I}_{\text{event}}(\Delta t, \mathbf{z})$ of each other car approaches 1 at collision, the steepness factor β_{coll} is used to model the position uncertainty originating from several possible sources like sensor inaccuracy, state prediction errors, unexpected driver behavior or unknown vehicle sizes. With β_{coll}, closer proximity leads to higher $\tau_{\text{coll}}^{-1}(\mathbf{z})$ and, accordingly, to a reduced probability of surviving.

The approach is consistently extensible to different types and sources of risk by using a composite event rate like

$$\tau^{-1}(\mathbf{z}) := \tau_0^{-1} + \tau_{\text{coll}}^{-1}(\mathbf{z}) + \tau_{\text{curv}}^{-1}(\mathbf{z}) + ... = \tau_0^{-1} + \tau_{\text{crit}}^{-1}(\mathbf{z}),$$

which can comprise the terms related to critical events, such as vehicle-to-vehicle collisions τ_{coll}^{-1}, losing control in curves τ_{curv}^{-1} and other rates. The

escape rate τ_0^{-1} plays a special role: It contains all (unknown and non-critical) "escape" events which might cause the current prediction to be invalid so that critical events lying further away in the future cannot occur anymore. For instance, if we assume constant velocity in the prediction and the collision will occur at TTC = 10 s, each disturbance, voluntary or involuntary action away from the constant velocity assumptions that occur within $[t, t + \mathrm{TTC}]$, will prevent the collision from happening.

Figure 2.4 illustrates τ_0^{-1} as trajectory alternatives changing $\mathbf{z}_{t:t+s}$ and the collision event rate $\tau_{\mathrm{coll}}^{-1}$, which are both used in the survival analysis.

By multiplying the instantaneous event probability from Equation (2.11) with the survival probability from Equation (2.13), the **event probability** can now be calculated as

$$E(s;t,\Delta t,\mathbf{z}_{t:t+s}) = I_{\mathrm{event}}(\Delta t,\mathbf{z}_{t+s})\, S(t+s;t,\mathbf{z}_{t:t+s}).$$

This is the probability that any type of event (escape event or critical) will happen in an interval of length Δt around $t + s$, given a state vector history $\mathbf{z}_{t:t+s}$ if we start observations at t and no event happened during $[t,t+s]$. Correspondingly, the event density (i.e., the probability of events per time unit) after a time s starting at t is given by

$$e(t+s;t,\mathbf{z}_{t:t+s}) := E(s;t,\Delta t,\mathbf{z}_{t:t+s})/\Delta t = \tau^{-1}(\mathbf{z}_{t+s})\, S(t+s;t,\mathbf{z}_{t:t+s})$$

and the total **time-accumulated event probability** (i.e., the probability that any of the events will happen during $[t,t+s]$) by

$$A(s;t,\mathbf{z}_{t:t+s}) := \int_0^s e(s';t,\mathbf{z}_{t:t+s'})\, ds'.$$

As a risk measure, only the time-accumulated probability of *critical events* should be applied. If we separate the critical events from the escape events (which avoid the accidents), we find out that the event density

$$e(s;t,\mathbf{z}_{t:t+s}) = \tau^{-1}(\mathbf{z}_{t+s})\, S(t+s;t,\mathbf{z}_{t:t+s}) =$$

$$[\tau_0^{-1} + \tau_{\mathrm{crit}}^{-1}(\mathbf{z}_{t+s})]\, S(t+s;t,\mathbf{z}_{t:t+s}) := e_0(s;t,\mathbf{z}_{t:t+s}) + e_{\mathrm{crit}}(s;t,\mathbf{z}_{t:t+s})$$

also separates into two respective terms. Consequently, for the time-accumulated event probability

$$A(s;t,\mathbf{z}_{t:t+s}) = A_0(s;t,\mathbf{z}_{t:t+s}) + A_{\mathrm{crit}}(s;t,\mathbf{z}_{t:t+s})$$

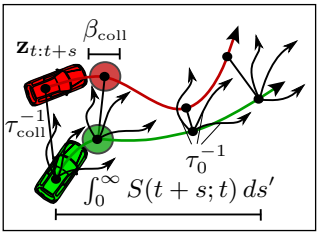

Figure 2.4: Collision risk prediction with survival analysis. The escape behaviors are visualized with arrows that allow to survive a possible accident. Over the future time, the probability of surviving increases.

holds true. The first term quantifies the overall future probability that critical events will be avoided during $[t, t+s]$, whereas the second term expresses the future probability of getting involved in a critical event during the same time span.

Since we are interested in the estimation of the time-accumulated future risk, we use

$$P_E(s;t,\Delta t) := e_{\text{crit}}(s;t,\mathbf{z}_{t:t+s})\Delta t$$

as the time-resolved differential risk measure. The integral risk measure then is

$$R(t) = A_{\text{crit}}(\infty;t,\mathbf{z}_{t:t+\infty}).$$

It can be shown that the time-accumulated event probability integrates to 1, which means

$$A(\infty;t,\mathbf{z}_{t:t+\infty}) = 1.$$

This is a sensible and necessary normalization condition because, for an infinite future, a critical event will happen with certainty 1.

Using this information, we arrive at

$$R_{\text{SA}}(t) = 1 - A_0(\infty;t,\mathbf{z}_{t:t+\infty}) = 1 - \underbrace{\tau_0^{-1}\int_0^\infty S(t+s;t,\mathbf{z}_{t:t+s'})\,ds'}_{\text{Overall escape probability}} \quad (2.14)$$

for our final risk measure. Figure 2.4 illustrates at the bottom the procedure of integrating the survival function $S(t+s;t)$ over the predicted time s to retrieve the risk $R_{\text{SA}}(t)$.

In summary, Equation (2.14) contains the overall probability of escaping a future critical event and its complement quantifies the overall probability of engaging in a future critical event. The equation is well-behaved in

the limiting cases. It automatically approaches 0, if there are no critical events present, because the overall escape probability approaches 1. For an imminent critical event, the escape probability reaches 0 as there is no time left for any type of escape events in terms of avoidance behavior or similar. The normalized function behavior is one of the core benefits of the survival analysis.

2.3 Model Comparison

For a quantitative comparison, the presented risk measures were tested on real crash cases taken from the German In-Depth Accident Study (GIDAS) dataset [32]. The GIDAS Pre-Crash-Matrix contains reconstructed trajectories of two TPs involved in an upcoming collision for longitudinal and intersection scenarios on average t = -5.5 sec ahead.

In order to quantify the risk for near-crash cases, we additionally change the course of the scene evolution. In the longitudinal car-following examples, the path of one TP is shifted laterally in such a way that the minimal distance is $d_E = 7\,\mathrm{m}$ instead of $d_E = 0\,\mathrm{m}$. There is no collision anymore, but a close passing. In the intersection example, we use for one TP the Foresighted Driver Model (FDM) [28], which changes the velocity profile $v(t)$ along the given path of the dataset. Accordingly, the TP decelerates and lets the other TP pass to avoid an accident.

At every timestep t in the simulation, a constant velocity model is used for both TPs to predict the distance $d_{t:t+s}$ for future times s. The prediction horizon is set to $s_h > 5.5\,\mathrm{sec}$ so that the time of crash or near-crash s_E is within the prediction interval. The sequence $d_{t:t+s}$ represents the input for all three risk measures. In this way, they have the same prerequisites and if the real trajectory violates the constant velocity assumption with an acceleration $a(t) \neq 0\,\mathrm{m/sec^2}$, all risk measures are equally impacted. Furthermore, we select the parameter D_c of TTCE and the Gaussian method and τ_0^{-1} as well as β_{coll} for the survival analysis such that for the maximal risk value of each $R_{\max} > 0.5$ holds true in the near-crash case.

Figure 2.5 shows the simulation view of a longitudinal crash and near-crash case starting from time t = -5.3 sec until the point of maximal criticality $t_E = 0\,\mathrm{sec}$ and plots of the risk measures $R_{\mathrm{TTCE}}(t)$, $R_{\mathrm{Gauss}}(t)$ and $R_{\mathrm{SA}}(t)$. Equally, Figure 2.6 summarizes the simulation results for an intersection scenario with the interval $t \in$ [-5.2 sec, 0 sec].

In both scenarios, $R_{\mathrm{SA}}(t)$ approaches the value 1 faster in the crash case and has a lower $R_{\max}$ in the near-crash case. Compared to $R_{\mathrm{TTCE}}(t)$, the

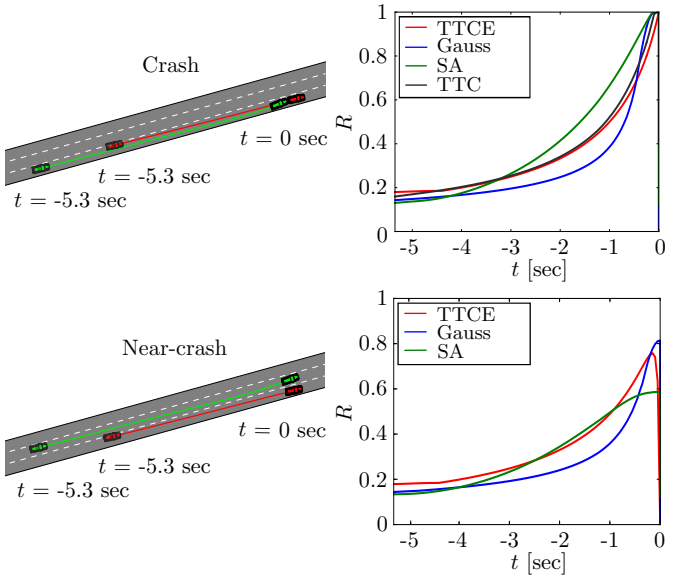

Figure 2.5: Comparison of risk measures for a longitudinal scenario with simulation view and their respective curves plotted. Top: Crash case. Bottom: Near-crash case. See Table 2.1 for a list of the measures' shortcuts.

performance of $R_{\text{Gauss}}(t)$ is higher, but its robustness lower. The reason lies in the square-root dependency $\alpha = 1/2$ in $R_{\text{Gauss}}(t)$ as opposed to $\alpha = 1$, which makes its curve shape flatter at $t < t_E$ and steeper close to t_E. Lastly, only for the longitudinal crash case $R_{\text{TTC}}(t)$ can be calculated with the help of Equation (2.3). It resembles $R_{\text{TTCE}}(t)$, but has slightly higher values because of a different parametrization of temporal uncertainty D_c.

Statistics for Multiple Scenarios

After demonstrating the general behavior of the risk measures, they are now applied on a set of 42 scenarios. The set consists of seven longitudinal and seven intersection scenarios from GIDAS, which have not only a crash and near-crash case, but also a non-crash case.

In the longitudinal examples, we move the path of one TP laterally so that $d_E = 12\,\text{m}$ is reached for a non-crash-case, compared to 7 m in the previously introduced near-crash case. In the intersection samples, we set constant velocities for both TP with the result that they now pass the

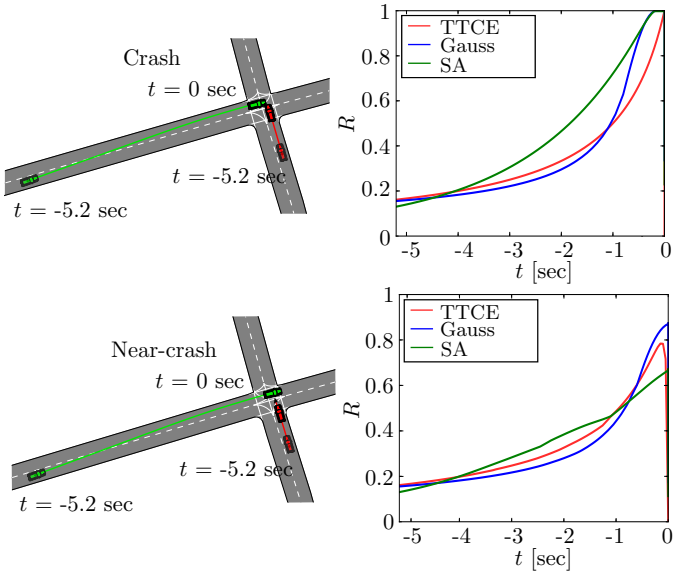

Figure 2.6: Comparison of risk measures for an intersection scenario. Instead of shifting the path laterally, we let the other car change the velocity to avoid the accident in the near-crash case. Top: Crash case. Bottom: Near-crash case.

intersection $|\Delta t| = 2\,\text{sec}$ away from each other. In Figure 2.7, the distance sequences $d(t)$ are pictured for all 42 test instances, which range from 120 m to 0 m. The pictured graphs $d(t)$ should not be confused with the predicted distances $d_{t:t+s}$, which act as the input of the risk measure. The distances decrease from a starting value in between $d = [20\,\text{m}, 120\,\text{m}]$ at time $t \approx -6\,\text{sec}$ to distances $d < 15\,\text{m}$ at the critical time $t = 0$.

When a risk measure crosses a threshold of $R_{\text{th}} = 0.7$, we define it as having detected a crash. An optimal risk measure has an early detection time t_d of the crash in the crash cases and has no false-positive detections FP in the near- and non-crash cases, which occurs if $R_{\max} > R_{\text{th}} = 0.7$ takes effect. Hereby note that the threshold of $R_{\text{th}} = 0.7$ is a compromise between low t_d and low FP, and that false negatives FN are generally zero in the crash cases.

In Table 2.1, the averaged values of t_d and $R_{\max}$ and the accumulated FP/N of each risk measure are listed for the 42 test samples. Furthermore, the variances σ_t and σ_R show the spread of the results. As anticipated from the last section, the survival analysis has the highest $|t_d|$ and the low-

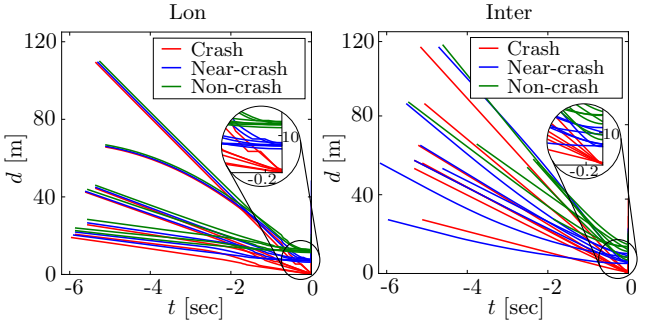

Figure 2.7: Variance in distance sequences of all scenarios of the extended GIDAS dataset. Distances for short times are highlighted in the plots. Left: Longitudinal scenario. Right: Intersection scenario.

Table 2.1: Statistics of risk measures for extended GIDAS dataset. The used shortcuts for the measures are Time-To-Closest-Encounter (TTCE), Gaussian method (Gauss), Survival Analysis (SA) and Time-To-Collision (TTC). A non-crash case was added in the analysis.

		Crash		Near-crash		Non-crash	
		Lon	Inter	Lon	Inter	Lon	Inter
TTCE	t_d [sec]	-0.47	-0.45	–	–	–	–
	σ_t [sec]	0.05	0.01	–	–	–	–
	$R_{\max}$	–	–	0.78	0.75	0.62	0.63
	σ_R	–	–	0.02	0.08	0.01	0.08
	FP/N	–	–	7/7	4/7	*0/7*	2/7
Gauss	t_d [sec]	-1.36	-0.85	–	–	–	–
	σ_t [sec]	0.75	0.39	–	–	–	–
	$R_{\max}$	–	–	0.84	0.82	0.53	0.57
	σ_R	–	–	0.03	0.11	0.03	0.16
	FP/N	–	–	7/7	6/7	*0/7*	2/7
SA	t_d [sec]	**-1.46**	**-1.14**	–	–	–	–
	σ_t [sec]	0.48	0.23	–	–	–	–
	$R_{\max}$	–	–	0.63	0.63	0.35	0.42
	σ_R	–	–	0.04	0.12	0.02	0.15
	FP/N	–	–	**0/7**	**3/7**	*0/7*	**0/7**
TTC	t_d [sec]	-0.76	–	–	–	–	–
	σ_t [sec]	0.16	–	–	–	–	–

est $R_{\max}$ and FP/N. The Gaussian method exhibits a larger $|t_d|$ than TTCE, but in the near-crash case, $R_{\max}$ and FP/N are smaller in TTCE.

In general, a crash is detected later and there are more false detections in intersection compared to longitudinal scenarios. That is because a trajectory with $a(t) \neq 0\,\mathrm{m/sec^2}$ results in a more parabolic curve of $d(t)$ in intersections (see Figure 2.7), which in turn causes higher errors in $d_{t:t+s}$ due to the constant velocity assumption used for prediction. Finally, TTC is included as a standard method in the automotive community to measure collision risk. However, in the evaluated cases the performance of the developed three risk measures has proven to be better and they work in a broader range of scenarios, i.e., in intersection scenarios as well as in near- and non-crash cases.[2]

2.4 Proposed Risk Model

So far, we saw an overview of the common risk metrics and compared their performance on accident detection tasks. The survival analysis proved to be superior to other methods, while tackling problems from uncertainties in driving. At this point, we want to understand in detail how the consideration of the future poses challenges for risk assessment.

On one hand, the involved processes which cause criticalities are inherently uncertain. Automated vehicles encounter non-linearities (behavior interaction and feedback loops), sensor inaccuracies (false positive or false negative detections), unknown environment parameters (occlusions or missing map data) as well as unobservable facts (drivers' state of mind) [27]. On the other hand, the variability of traffic situations leads to multiple dimensions in the probability of dangerous events:

1. Different types of risks might arise during the scene's progress. Typical risks are vehicle-to-vehicle collisions, loss of control in a curve or rule violations, e.g., not considering an intersection priority.

2. Possible subset of entities that are involved in critical events. For collision risks, this implies the pairwise consideration of traffic participants.

[2] As a remark, in Table 2.1, TTC has a higher (meaning better) $|t_d|$ compared to TTCE. This might seem unexpected, because we derived TTCE to be a generalization of TTC. However, the TTCE parametrization was optimized to work in all cases (crash, near-crash and non-crash), whereas the TTC implementation specializes on longitudinal crash cases only.

3. For a particular risk type and a subset of concerned entities, multiple evolutions of the scene (e.g., turn left, go straight, turn right or lane change) create distinct events.

4. For a scene, a countless number of critical events can happen at various predicted states (position, velocity, acceleration, etc.).

In this section, a novel and probabilistic situation risk model is derived based on the survival analysis and extended to naturally integrate sensory, temporal and behavioral uncertainties as they arise in real scenarios.

The model is applied and tested on naturalistic driving data of a multi-lane boulevard with several intersections, enabling the visualization of road criticality maps. For this reason, we can call the model Risk Spot Detector (RSD). The underlying measure will be used throughout the dissertation: for the motion planning in Chapter 3 and the warning system in the subsequent Chapter 4. As a comparison, TTC and the simpler metric Time Headway (TH) are also benchmarked in the classification task. The presented RSD can account better for low but existing risk in normal driving on multi-lane segments and intersections with dense traffic.

2.4.1 Gaussian Method with Survival Analysis

We assume the predicted scene of two traffic participants TP1 and TP2 as given. The state vector $\mathbf{z}_{t:t+s}$ describes their trajectories over predicted times, i.e., t, $t+s_1$, $t+s_2$, $t+s_3$ and so on. Paths are provided from map geometries and the positions are retrieved by calculating the traveled distance from an assumed velocity profile.

The basic and main idea of the novel risk model is that we utilize the *probability output of the Gaussian method* $P_E(s;t,\Delta t)$ for the collision rate $\tau_{\text{coll}}^{-1}(\mathbf{z}_{t:t+s})$ in the survival analysis (see Figure 2.8). The Gaussian method ultimately allows to model the positional uncertainties for TP1 and TP2 separately, which gives freedom in the parametrization. Additionally, we still have the *advantageous property of the survival analysis* to integrate probabilities for achieving an overall future risk value $R(t)$.

In detail, we start with the formulation of the Gaussian method. From now on, we will call the collision probability $P_{\text{coll}}(s;t,\Delta t)$. Equation (2.7) is the basis and is given by

$$P_{\text{coll}}(s;t,\Delta t) \sim \frac{1}{\sqrt{2\pi(\sigma_1^2+\sigma_2^2)}} \exp\left\{-\frac{(\mu_2-\mu_1)^2}{2(\sigma_1^2+\sigma_2^2)}\right\} \tag{2.15}$$

with the single mean positions μ_1 and μ_2 of the TPs and their uncertain-

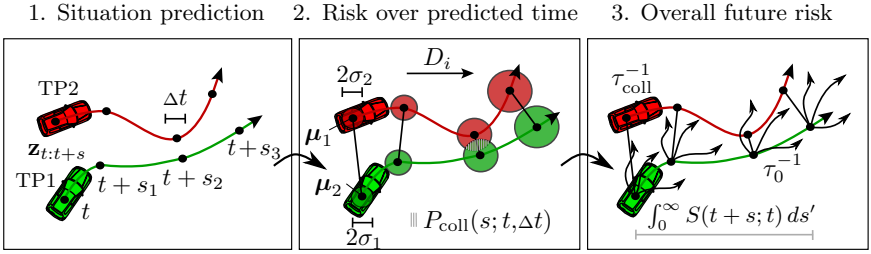

Figure 2.8: Combination of Gaussian method and survival analysis for risk prediction. Thereby, a risk measure is proposed that can flexibly include uncertainties in the position, velocity and future time.

ties σ_1 and σ_2. Cars are assumed to be indexed with $i = 1{,}2$. Accordingly, the variance σ_i^2 grows with a Brownian model depending on the future time via

$$\sigma_i^2(t+s) := \sigma_{0,i}^2 + D_i s. \tag{2.16}$$

Here, a starting value $\sigma_{0,i}^2$ was added to the increase factor D_i.

The inhomogeneous Poisson process is defined by the total event rate $\tau^{-1}(\mathbf{z}_{t:t+s})$, which characterizes the mean time between events and consists of a critical event rate τ_{crit}^{-1} and the escape rate τ_0^{-1} (any type of influences or behavioral options that contribute mitigating resp. "escaping" from critical events)

$$\tau^{-1}(\mathbf{z}_{t:t+s}) = \tau_0^{-1} + \tau_{\text{crit}}^{-1}. \tag{2.17}$$

Before, collision risk was represented by the event rate τ_{coll}^{-1} for one TP pair. However, situations in which the ego car interacts with several other TPs j and other types of risks, such as the risk of losing control in curves τ_{curv}^{-1}, may be included likewise with

$$\tau_{\text{crit}}^{-1} = \sum_j \tau_{\text{coll},j}^{-1} + \tau_{\text{curv}}^{-1}.$$

The novel idea is that collision probabilities of the Gaussian method $P_{\text{coll}}(s;t,\Delta t)$ are evaluated for each vehicle pair and then summed up for the overall collision event rate τ_{coll}^{-1}. In comparison, in Section 2.2, the rate τ_{coll}^{-1} was modelled with a simpler distance-based function. We thus set the event rate as the probability over a small time Δt, which leads to

$$\tau_{\text{coll},j}^{-1}(\mathbf{z}_{t:t+s}) = P_{\text{coll},j}(s;t,\Delta t)/\Delta t.$$

The survival function $S(s;t,\mathbf{z}_{t:t+s})$ then indicates the probability that the ego vehicle will not be engaged in an event like an accident from t until $t+s$. Combining (2.17) with the survival function, one can derive a probability density for general events

$$p_E(s;t,\mathbf{z}_{t:t+s}) = \tau_0^{-1} S(s;t,\mathbf{z}_{t:t+s}) + \tau_{\text{crit}}^{-1} S(s;t,\mathbf{z}_{t:t+s})$$

and we obtain the overall future risk, this time, as the integral over all predicted times of the critical events

$$R(t) = \int_0^\infty \tau_{\text{crit}}^{-1}(\mathbf{z}_{t:t+s}) S(s;t,\mathbf{z}_{t:t+s})\, ds. \tag{2.18}$$

Due to numerical reasons, in the application, the temporal integration is capped with a fixed prediction horizon s_h.

Figure 2.8 matches the novel model to the general three steps of predictive risks. Besides a first prediction step, it employs the Gaussian method for the second step (i.e., risk over predicted time) and the survival analysis for the third step (i.e., overall future risk).

2.4.2 Extensions for Normal Driving

The Gaussian method was seamlessly included in the survival analysis. In the following, we will look at how driving uncertainties can be improved with the Gaussian method. This is especially helpful in normal driving with low but existing collision probabilities.

The Gaussian probability densities allow formulating collision uncertainty as a function of spatial uncertainties $\sigma_i^2(t+s)$ from the involved TPs. The value of $\sigma_{0,i}^2$ quantifies **measurement uncertainty** and can be different for each TP. It reflects the uncertainty at current times $s=0$.

For future times $s>0$, the parameter D_i specifies **prediction uncertainty**. Subsequently, the survival analysis normalizes the event probabilities related to all TPs and risk types and for the time s_E until a critical event

$$\lim_{s_E \to 0} R(t) \to 1 \text{ and } \lim_{s_E \to \infty} R(t) \to 0$$

holds true. A constant escape rate τ_0^{-1} reduces thereby $S(s;t,\mathbf{z}_{t:t+s})$ over the predicted time and introduces an accumulating event neglection effect (events in the more distant future are considered to a lesser extent). Similarly, if high τ_{crit}^{-1} occurs early, for all times afterwards, $S(s;t,\mathbf{z}_{t:t+s})$ is diminished. RSD thus takes **historical uncertainty** correctly into account (future risks that arise after another critical event are further reduced).

As shown in Section 2.3, longitudinal following and intersection crash cases are detected by the survival analysis earlier than by the Gaussian method alone or TTC (at least 1.1 sec before the critical event actually happens). At the same time, it has a considerably lower number of false-positive detections for near- and non-crash cases.[3]

Nevertheless, to enhance the predicted risks in normal driving, we need to account for further uncertainties precisely. For this purpose, the refined positional uncertainty modeling of the Gaussian method is made use of and extensions are introduced that improve the risk model in three special situations: close passing of other TPs in 1. longitudinal segments, 2. when stopping in front of as well as 3. turning at intersections.

2D Gaussians

Because the TPs are predicted to drive along predefined paths, we assume that $\sigma_i^2(t+s)$ must be extended to two dimensions for handling longitudinal and lateral influences. In this way, we obtain ellipses specified by an uncertainty matrix $\boldsymbol{\Sigma}_i^{'}$ around the mean position vector $\boldsymbol{\mu}_i$.

$$\boldsymbol{\mu}_i = \begin{bmatrix} \mu_{x,i} \\ \mu_{y,i} \end{bmatrix}, \quad \boldsymbol{\Sigma}_i^{'} = \begin{bmatrix} \sigma_{\text{lon},i}^2 & 0 \\ 0 & \sigma_{\text{lat},i}^2 \end{bmatrix}$$

The top of Figure 2.9 shows the 2D Gaussians for one point in time $t+s$. The relative orientations to the absolute x,y-coordinate system are indicated with α_i. To retrieve the product of the corresponding Gaussian functions $\mathbf{f}_i$, the longitudinal and lateral uncertainties $\boldsymbol{\Sigma}_i^{'}$ have to be transformed with

$$\boldsymbol{\Sigma}_i = \mathbf{R}\boldsymbol{\Sigma}_i^{'}\mathbf{R}^T \text{ and } \mathbf{R} = \begin{bmatrix} \cos\alpha_i & -\sin\alpha_i \\ \sin\alpha_i & \cos\alpha_i \end{bmatrix}.$$

Equation (2.15) can then be rewritten in 2D to

$$P_{\text{coll}}(s;t,\Delta t) = \det|2\pi(\boldsymbol{\Sigma}_1 + \boldsymbol{\Sigma}_2)|^{-\frac{1}{2}} * \exp\{-\frac{1}{2}(\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1)^T(\boldsymbol{\Sigma}_1 + \boldsymbol{\Sigma}_2)^{-1}(\boldsymbol{\mu}_2 - \boldsymbol{\mu}_1)\}.$$

Without incorporating the orientation of the TPs, a longitudinal scenario of two TPs passing closely with a constant lateral offset from time t_{start} to t_{end} (see bottom of Figure 2.9) produces the same high risk as an in-

[3] In the experiments, we used an event threshold of $R(t) > 0.7$ for upcoming accidents. Decreasing the threshold would lead to higher sensitivity of RSD for accidents.

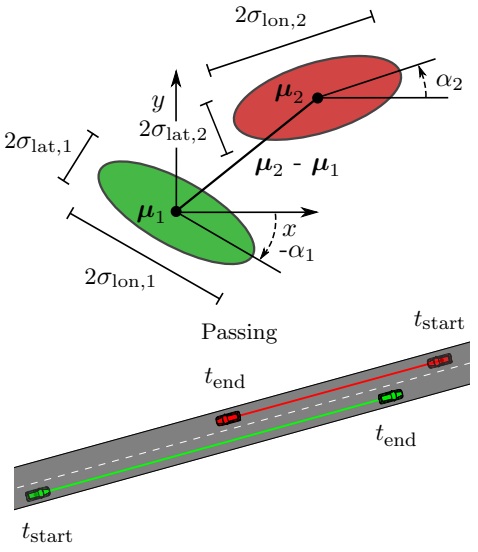

Figure 2.9: Top: Schema and variables of the 2D Gaussian ellipses. Bottom: Harmlessly passing another TP on a straight road, in which original 1D Gaussians would wrongly detect risks.

tersection scenario of two TPs passing closely with a 90° angle. In contrast, elongated 2D Gaussians rate longitudinal passing as more safe.

Position Uncertainty by Velocity Variance

Over the predicted time, $\sigma_i(t+s)$ grows proportionally to $\sqrt{s}$ according to Equation (2.16). We extrapolate the kinematics of the current state to retrieve trajectories, but the velocities of the TPs are not influencing the uncertainty prediction. After a prediction step of size Δs, their longitudinal position on the path l_i is shifted by Δl_i according to

$$l_i(s+\Delta s) = l_i(s) + \Delta l_i = l_i(s) + v_i(s)\Delta s$$

with velocities v_i.[4] When we additionally assume a longitudinal velocity uncertainty according to, e.g., a normal distribution with variance $\sigma_{v,i} = \langle v_i \rangle c_i$, the increase of spatial uncertainty is determined by the velocity

[4] Remark: we set the current time $t = 0\,\text{sec}$ and look only at the increment in the predicted time s.

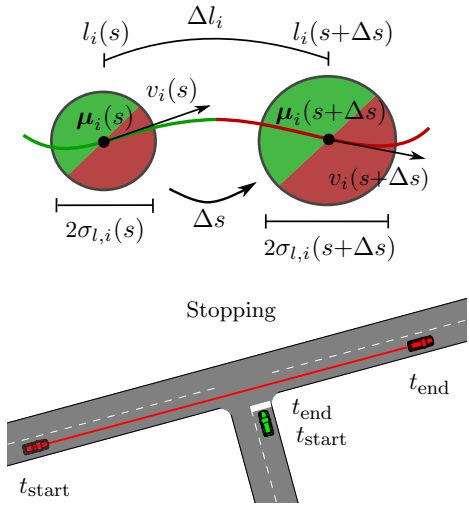

Figure 2.10: Top: Gaussian growth description for prediction. Bottom: Securely stopping at the intersection, while another TP crosses. Without considering velocity in the growth, these scenarios would not be reproduced realistically.

uncertainty factor c_i. For a discrete step in prediction time we then get

$$\sigma_{l,i}(s + \Delta s) := \sigma_{l,i}(s) + c_i v_i(s) \Delta s,$$

$$\sigma_{l,i}(s = 0) = \sigma_{0,i}.$$

Hereby, $\sigma_{l,i}(s)$ has slopes of $\sigma'_{l,i}(s)$ that depend on $c_i v_i(s)$.

Especially in scenarios with extreme velocities (i.e., $v_i < 5\,\text{m/sec}$ and $v_i > 15\,\text{m/sec}$), the previous Brownian model leads to over- or underestimation of uncertainties and thus the contained risk.

Figure 2.10 outlines the change in the width of the Gaussians $2\sigma_{l,i}(s)$ and a waiting TP at a T-intersection while another TP is crossing during the time interval $[t_{\text{start}}, t_{\text{end}}]$. Although the situation can be categorized as safe, Brownian position uncertainty would lead to large risk areas around the standing green car position. On the contrary, the velocity propagation yields constant small $\sigma_{l,i}(s)$ for the stopped car and this situation is seen as non-critical. For this reason, we will use the new propagation model.

Path-following Mixture Model

Large $\sigma_i(t + s)$ at prediction times $s \gg 0$ might unintentionally cover

opposite lanes when a TP is turning. Therefore, we include geometry sensitive Path-following Mixture Models (PMM), allowing curved Gaussian shapes for position uncertainties [70].

First, we split the Gaussian with $2\sigma_i$ into N_p smaller Gaussians of $2\sigma_{i,k}$, as drawn in the upper half of Figure 2.11. The smaller Gaussian components at $\mu_{i,k}$ are spread to the right and left from μ_i, while intersecting close to their Full Width at Half Maximum (FWHM) value.[5] The composition heuristics can be summarized as

$$\sigma_{i,k} = \frac{m_f \mathrm{FWHM}}{N_p}\sigma_i,$$

$$\mu_{i,k} = \mu_i + \frac{2k}{m_f \mathrm{FWHM}}\sigma_{i,k}$$

$$\text{with } k = 0,..., \pm\frac{N_p - 1}{2} \text{ and } \mathrm{FWHM} = 2\sqrt{2\ln 2}.$$

The constant m_f ensures that the constructed PMM does not contain additional local minima and that it has a smooth shape.[6]

At last, each function of the component $f_{i,k}$ is summed up and weighted with the factor $w_{i,k}$ to reduce the deviation from the former Gaussian f_i, which leads to

$$w_{i,k} = f_i(\mu_{i,k})\frac{f_i(\mu_i)}{\sum_k f_i(\mu_{i,k}) f_{i,k}(\mu_i)},$$

$$f_{\mathrm{PMM},i}(x) = \sum_k w_{i,k} f_{i,k}(x).$$

In $w_{i,k}$, the collective peaks of the components $\sum_k f_{i,k}(\mu_{i,k})$ are scaled to match the desired original heights $f_i(\mu_{i,k})$. The PMM composition heuristics achieves similar reconstruction errors as parameter optimization approaches for $f_{\mathrm{PMM},i}(x)$ and $N_p > 13$, however, providing a fast and direct calculation.

In the lower half of Figure 2.11, one TP takes a sharp curve and another TP crosses the intersection. During the situation occurring from t_{start} until t_{end}, the resulting phantom risks from the elongated Gaussians are avoided with the PMM because the smaller Gaussians cling along the curved path.

[5] At the same time, we require one element with $\mu_{i,k} = \mu_i$ and thus an uneven number N_p.

[6] For $m_f > 1$, the mixture components get closer to each other and have higher $\sigma_{i,k}$.

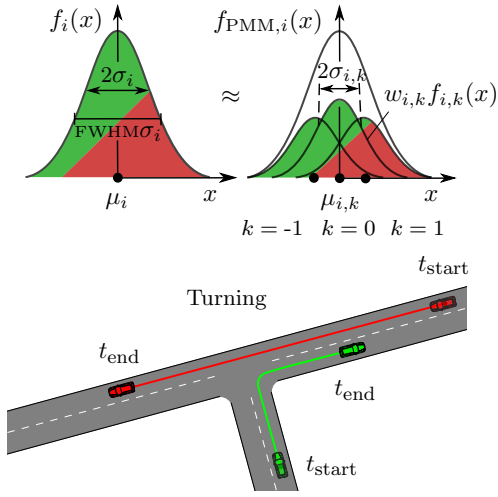

Figure 2.11: Top: Partition of Gaussian into multiple components. Bottom: Safely turning at intersections with an oncoming TP. The mixture model can more accurately model collision risks with curve segments in the prediction.

2.5 Behavior Extrapolation

The target is to apply the developed risk model from the last Section 2.4 for criticality classification. Specifically, the resulting Risk Spot Detector (RSD) is compared against the common Time Headway (TH) and Time-To-Collision (TTC). For longitudinal collisions, TH and TTC are able to quantify risks in terms of the time until a critical event happens. In this section, we analyze their prediction assumption and see how RSD can be used to cover risks in more general terms and, concretely, to integrate the assumptions from both TH and TTC.

Time Headway and Time-To-Collision

While an ego vehicle is driving with longitudinal velocity v_1 along a path and following another vehicle, TH [117] describes the time until the ego car travels from the current longitudinal position l_1 to the current longitudinal position of the other car l_2 according to

$$\text{TH} = \frac{-\Delta l}{v_1} \text{ with } \Delta l = l_1 - l_2. \tag{2.19}$$

The average human reaction time was found to be around $t_r \approx 1\,\text{sec}$ [132]. By keeping $\text{TH} > t_r$, once the other entity brakes at l_2, the ego entity has some time left to apply an appropriate deceleration to mitigate or even avoid a collision.

TH can be seen as a risk measure. Its inverse 1/TH is assumed to correlate with the collision probability given that the follower drives with constant v_1 in combination with the front vehicle suddenly stopping at l_2. By contrast, TTC [124] assumes that both cars continue driving with constant longitudinal velocities v_1 and v_2 and calculates the time until a collision occurs when $l_1 = l_2$,

$$\text{TTC} = \frac{-\Delta l}{\Delta v}, \text{where } \Delta v = v_1 - v_2. \tag{2.20}$$

The left and middle part of Figure 2.12 visualize a longitudinal car following scenario and the designated collision points from TH and TTC. For forward driving, a valid TTC only exists for $-\Delta l > 0$ and $\Delta v > 0$. In this case, $v_1 > \Delta v$ holds so that TH always overestimates potential risks as compared to TTC (i.e., $1/\text{TH} > 1/\text{TTC}$).

Behavior Uncertainty

TH and TTC approximate critical event probabilities based on the point of maximal criticality by deriving the collision time with simple kinematic equations. If the hypothetical accident does not occur as for non-longitudinal scenarios, there is no risk. Some attempts have been made to make TTC more flexible with, e.g., extrapolating constant deceleration for the obstacle [132]. Analogously, the required longitudinal deceleration of the follower can be compared with its maximal possible value to calculate Brake Threat Numbers [8].

Nevertheless, Risk Spot Detector (RSD) has the beneficial property of estimating a continuous, differential, probabilistic risk along the entire future predicted time, see Equation (2.18). Furthermore, it is valid independently of the predicted behavior and thus can be used with the same prediction assumptions in TH or TTC, but also for completely arbitrary paths and velocity profiles. With RSD, the velocity-dependent position uncertainty $\sigma_{l,i}(s)$ allows to systematically incorporate **behavior uncertainty**. For constant velocity assumptions, this results in the front part of the probability densities being fed by acceleration and the back part by deceleration behaviors. The middle of the probability densities around $\boldsymbol{\mu}_i$ constitutes constant average velocity.

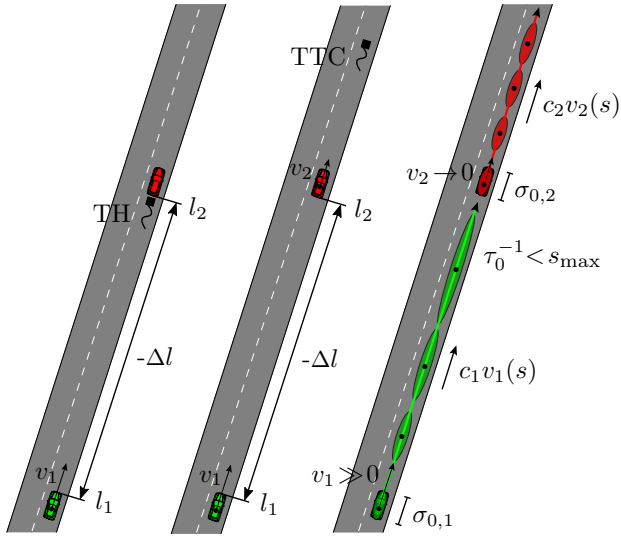

Figure 2.12: Left: Kinematic variables for TH. Middle: Additional velocity for the other car with TTC. Right: Parameter setting of behavior uncertainty in RSD that probabilistically includes both assumptions of TH and TTC.

We set $s_h = 12\,\text{sec}$ and $\tau_0^{-1} = 3\,\text{sec}$ to be capable of incorporating other TPs at least TH $< 5\,\text{sec}$ far away. Followingly, the motion planner Risk Optimization Method (ROPT) [94] was employed to parametrize $\sigma_{0,i}$ and c_i. ROPT utilizes RSD in a cost function to find safe behaviors through traffic. For longitudinal and intersection scenarios, sanity checks showed whether ROPT holds reasonable distance thresholds to other TPs in various v_i settings. Increasing c_i lets ROPT keep longer distances. In addition, typical accelerations at $s = 0$ and velocity changes after $s = 3\,\text{sec}$ in real driving recordings [121] were looked at and their standard deviations fitted to c_i. In both procedures, $6\sigma_{0,i} = 4\,\text{m}$ was obtained as the average TP length[7] and $c_i = 0.1$.

The variance increase of the Gaussians continuously changes over the predicted time according to $v_i(s)$. In the example of Figure 2.12 (see right part), the velocity of the follower v_1 is larger than v_2 of the front TP. In other words, the ellipses grow faster for TP1 than for TP2. It was found

[7] Equations (2.19) and (2.20) are both extended to consider the car sizes by changing Δl to $\Delta l^* = \Delta l + 4\,\text{m}$.

that the fitted parameter setting covers the average behavior uncertainty of normal driving statistics well, resulting in a mixture of the sudden stop assumption from TH and the constant velocity prediction from TTC.

In the RSD formalism, positional uncertainties between TPs are not explicitly correlated. Aside from this, the constant velocity prediction suggests unawareness of the TPs from each other. However, the risk calculation incorporates a mutual influence in TPs by the spread of the Gaussian velocity distributions and thus assumes that vehicles can take on velocities which deviate from constant velocity (e.g., a follower that accelerates towards a vehicle in front which brakes and vice versa). Furthermore, since RSD is agnostic to the type of trajectories, additional interactions can be incorporated by modeling specific motion patterns.

2.6 Experiments

For the experimental evaluation of the performance of RSD, the naturalistic dataset NGSIM (Next Generation Simulation) [121] was used. The task for RSD is to visualize experienced criticality levels on road maps. The NGSIM dataset consists of five traffic study areas in the US, in which cameras are mounted on high buildings.

With computer vision techniques, the positions of all vehicles were detected with an accuracy of ($\Delta x = 0.6\,\mathrm{m}, \Delta y = 1.2\,\mathrm{m}$), whereby Δx and Δy are errors in Cartesian coordinates. In particular, Lankershim Boulevard is suitable for testing the robustness of RSD. It has multiple intersections (4 with traffic lights, 2 with priority), wide and narrow road structures (2, 3 and 4 lanes), inner-city and highway stretches (velocities in the range $v = 0 - 20\,\mathrm{m/sec}$), a long curve (making up 20% of the total section) and dense traffic (1,000 vehicles in 15 minutes). As 96% of the TPs are cars, it is reasonable to neglect the particular masses and sizes (3.5% trucks and buses, and 0.2% motorbikes).

For the risk analysis, we first load and normalize the trajectories [101]. By employing an exponential moving average filter forwards and backwards, we can afterwards separately smooth the positions p, velocities v and accelerations a. Both v and a are computed through differentiation of p, which results in additional noise. Consequently, we set different smoothing widths $T_p = 10\,\mathrm{sec}$, $T_v = 20\,\mathrm{sec}$ and $T_a = 80\,\mathrm{sec}$ to compensate the noise increase, as it was similarly done in [113].

As a next step, we successively assume each car to take the role of an ego car and extract all other cars in the same time interval. In the simulation,

the real driven position sequence is used as the underlying paths for the trajectory prediction of RSD, TH and TTC. In this way, we already know in advance intentions of the cars (e.g., lateral lane changes or turning at an intersection). Because of fixed velocity extrapolations (i.e., constant velocity or sudden stop), their longitudinal behavior is, however, assumed to be unknown. Risks can only come from wrongly predicted velocities of the cars along their paths.

Figure 2.13 shows the probability mass function pmf and cumulative distribution function cdf with a histogram representation for the occurring v and a as well as distances $-\Delta l$ and relative velocities Δv to the front vehicle.[8] More than 25% of the time, the vehicles stand in traffic or in front of an intersection. The velocity distribution is bimodal, with a high peak at $v \approx 0\,\mathrm{m/sec}$ and a broader peak around $v = 12\,\mathrm{m/sec}$.

Overall, vehicles do not excessively brake or accelerate. The acceleration distribution has its mean at $\mu_a = 0\,\mathrm{m/sec^2}$ with a standard deviation of $\sigma_a = 0.4\,\mathrm{m/sec^2}$. When there is a front vehicle, most $-\Delta l$ lie around $\mu_{\Delta l} = 5\,\mathrm{m}$ of an approximated log-normal distribution. Only 5% have lower values than $\mu_{\Delta l}$ and they happen at small v. In contrast, Δv corresponds approximately to a logarithmic distribution with $\mu_{\Delta v} = 1\,\mathrm{m/sec}$, whereby big Δv takes place in any of the velocity intervals.

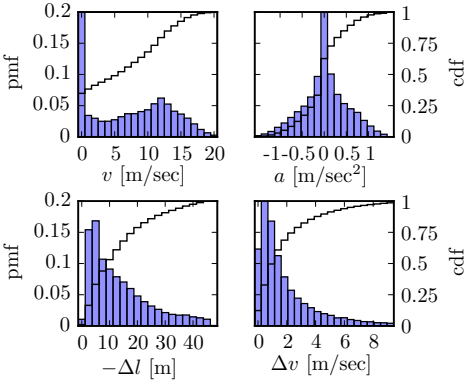

Figure 2.13: Variance of kinematics in data extracted from NGSIM (Lankershim Boulevard). We analyze single velocities and acceleration as well as distances and relative velocities in car-followings.

[8] Looking, e.g., at the histogram for v, pmf represents the share of data points within an interval of 1 m/sec and cdf is the successive sum of pmf values from left to right.

2.6.1 Results

In the following, we will analyze the qualitative and quantitative differences between the risk measures based on RSD, TH and TTC.

The experimental goal is to reason about which driving behaviors may create potentially fatal outcomes and to find the corresponding street areas on which they are likely to appear. For this purpose, four criticality cases are defined (dangerous, offensive, uncomfortable, noticeable) and for TH set accordingly to $b_1 = [0\,\text{sec}, 0.5\,\text{sec}]$, $b_2 = [0.5\,\text{sec}, 1\,\text{sec}]$, $b_3 = [1\,\text{sec}, 2\,\text{sec}]$ and $b_4 = [2\,\text{sec}, 4\,\text{sec}]$.[9]

The bins are colored red, yellow, cyan and blue and every calculated TH value of the NGSIM trajectories with a discretization of $\Delta t = 0.1$ s is sorted into them.[10] For RSD and TTC, we then fill the bins consecutively with the most to least hazardous events until they contain the same amount of events as for TH. The boundaries of the bins arise automatically from the corresponding first and last data points sorted into each bin. Eventually, criticality maps are created by plotting the color with the highest risk of RSD, TH and TTC at each road point and we analyze the velocity distributions in their bins.

The outcome for TH is depicted in Figure 2.14. TH classifies dangerous and "offensive" risks at velocity intervals around a Gaussian with $\mu_{\text{TH}} = 12\,\text{m/sec}$ and $\sigma_{\text{TH}} = 2.5\,\text{m/sec}$. When the ego car is following another car with high v and relatively low $|\Delta l|$, the sudden stop prediction causes strong worst cases. These situations are often found in the top right curvy segment. Moreover, uncomfortable and noticeable TH appear also for cars braking from moderate v approaching another car waiting close to an intersection. Due to the discontinuity of Equation (2.19), TH cannot evaluate risks for $v \to 0$.

Now, we concentrate on TTC, which depends on Δv. Mainly for cars approaching the lowest large intersection (East=70 m, North=120 m) with big Δv, TTC detects risks in the red and yellow bins (see Figure 2.15). The bins for TTC are computed as $b_{1,\text{TTC}} = [0\,\text{sec}, 2.15\,\text{sec}]$ and $b_{2,\text{TTC}} = [2.15\,\text{sec}, 5.76\,\text{sec}]$, which lies in the values $1 - 5\,\text{sec}$ commonly used for Collision Mitigation Systems (CMS). Alongside the logistic distribution of $\mu_{\text{TTC}} = 3\,\text{m/sec}$ and $\sigma_{\text{TTC}} = 3\,\text{m/sec}$, the cyan bin includes the critical car following scenario of TH. But since $b_{3,\text{TTC}} = [5.76\,\text{sec}, \infty]$, the data

[9] As references, in ACC the minimal TH is 1 sec and the maximal 3 sec. Furthermore, TH = 2 sec is recommended on US highways.

[10] Approximately 40% of the data points are in the bins, the rest 60% reflect safe behaviors with no noticeable risk at all.

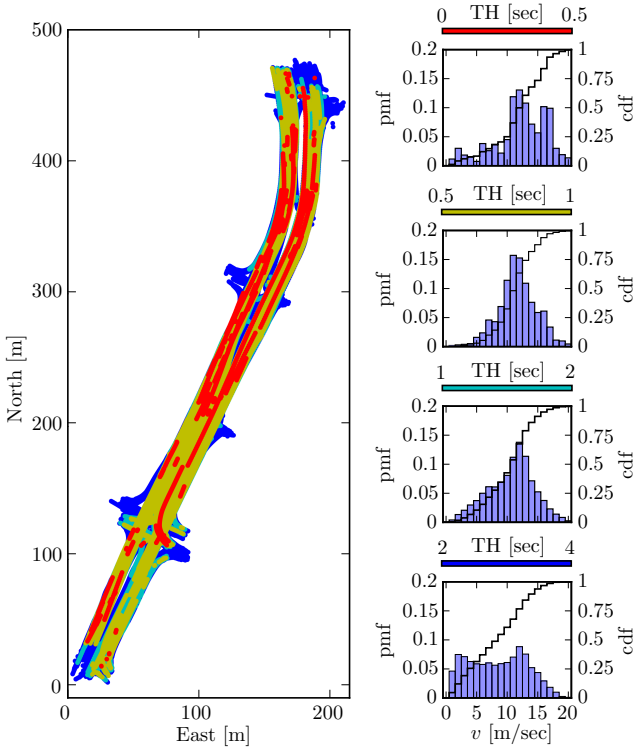

Figure 2.14: Criticality map and velocity histograms for Time Headway (TH). We differentiate between four risk levels (dangerous until noticeable driving) with velocity pmf and cdf shown in the diagrams of the right column. One can see that TH mostly detects tailgating on highways.

points are too few. TTC cannot detect car following situations with small distances, since $\Delta v \approx 0$ in these cases. On that account, the last blue bin includes arbitrary v in NGSIM. This can be seen by comparing the bin's distribution shape with the overall pmf of v from Figure 2.13.

Whereas TH and TTC only capture frontal longitudinal collision risks, RSD can capture all possible collision risks. However, for a fair comparison, RSD is initially applied on the front TP only. The parametrization of $\sigma_{0,i}$ and c_i defined in Section 2.5 is used for both the ego vehicle and other TP. As R is normalized to [0,1] and represents a probability, the first bin in Figure 2.16 starts at $R = 1$ and the last ends at $R \approx 0$. The thresholds

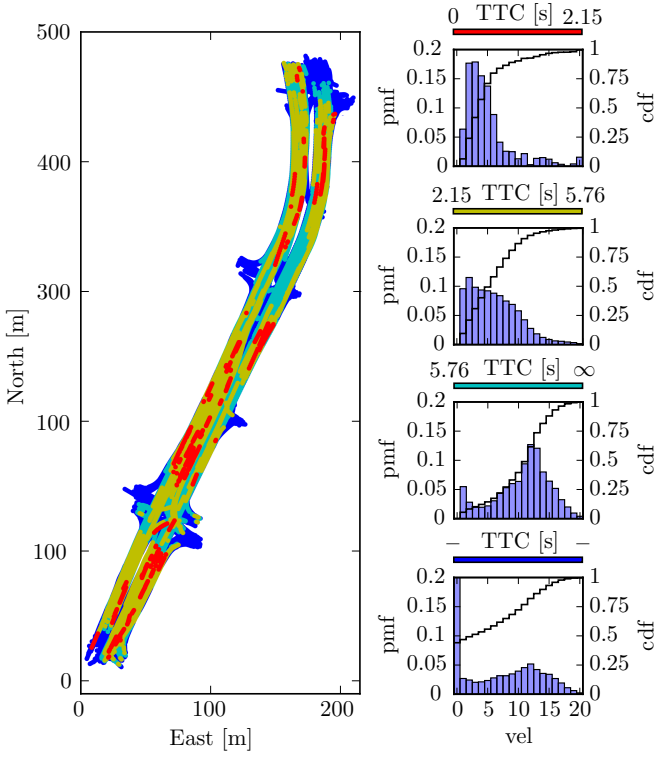

Figure 2.15: Criticality map and velocity histograms for the Time-To-Collision (TTC) metric. TTC captures well braking risks around intersections. These are in the dangerous red bins, including mostly driving with lower velocities.

are as follows: $b_{1,\mathrm{RSD}} = [1, 0.39]$, $b_{2,\mathrm{RSD}} = [0.39, 0.17]$, $b_{3,\mathrm{RSD}} = [0.17, 0.01]$ and $b_{4,\mathrm{RSD}} = [0.01, 0.002 \cdot 10^{-4}]$. To be compatible with TTC, we base the RSD on a constant velocity prediction. Also, similarly to TTC, RSD detects risk zones around intersections from existent Δv in combination with decreasing $|\Delta l|$ at $\mu_{\mathrm{RSD},1} = 4\,\mathrm{m/sec}$ in bins $b_{1,\mathrm{RSD}}$ and $b_{2,\mathrm{RSD}}$.

In addition, RSD incorporates acceleration and deceleration behavior through the velocity uncertainty. The bins $b_{3,\mathrm{RSD}}$ and $b_{4,\mathrm{RSD}}$ show a similar distribution of velocities as in TH with $\mu_{\mathrm{RSD},2} = 12\,\mathrm{m/sec}$. If we would set c_i to higher values, the abrupt stop prediction from TH could also be incorporated into RSD. This would shift the critical following incidents to $b_{1,\mathrm{RSD}}$ and $b_{2,\mathrm{RSD}}$ as well as the intersection approaching to $b_{3,\mathrm{RSD}}$ and

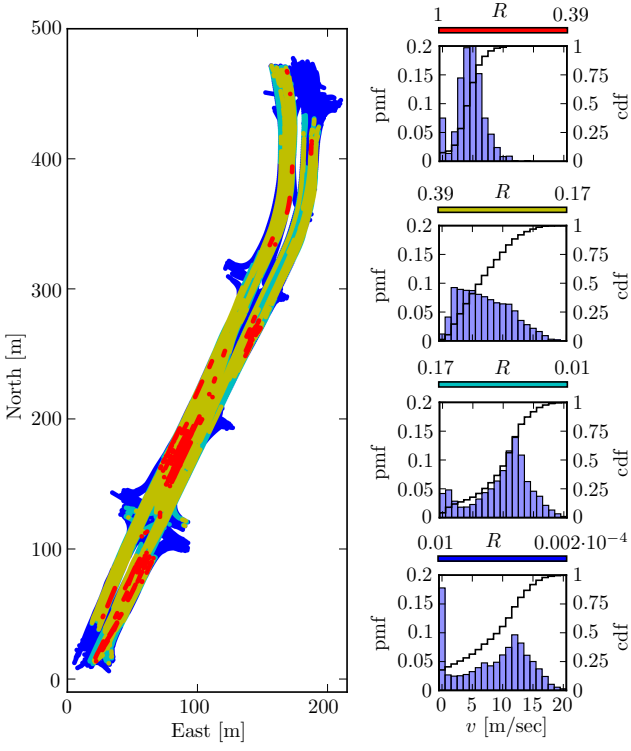

Figure 2.16: Criticality map and velocity histograms of the Risk Spot Detector (RSD) applied for front car, which is based on the novel risk model. RSD identifies intersection braking and highway tailgating. These are the same situation categories, which were detected separately from TH and TTC.[11]

$b_{4,\mathrm{RSD}}$. In general, constant velocity is, however, statistically more realistic (see the peak of the distribution at $\mu_a = 0\,\mathrm{m/sec^2}$ in Figure 2.13).

Traffic Situation Risk

We discovered that TH is able to filter larger amounts of critical points than TTC into the bins, but arranges risks in a different order of criticality levels. Besides, TTC is not precise for the uncomfortable and noticeable

[11] Note the difference in the boundaries of the last two bins to TTC in Figure 2.15. RSD is capable of having low but existent risk values, while TTC cannot distinguish all tailgating incidents from safe behaviors.

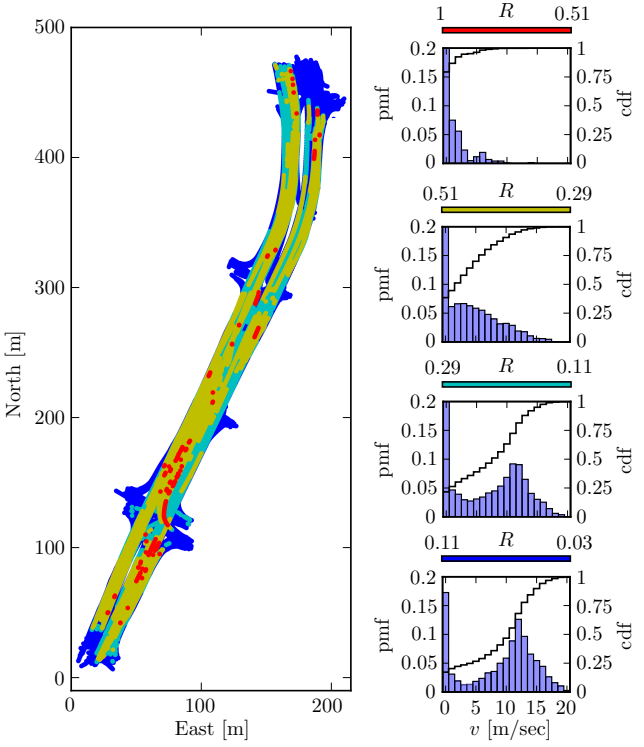

Figure 2.17: Criticality map and velocity histograms of RSD for all surrounding cars. Now, RSD has more localized hot spots and additionally describes in each bin the relation to traffic density. This is depicted in the velocity ranges that are close to zero.

bins. RSD can label the existing two types of risk causes into reasonable criticalities. To evaluate complete traffic situation risk, we consider in RSD all other TPs within a sensor range of $r = 50\,\text{m}$. Figure 2.17 shows that RSD filters out correctly the critical TPs on the same path in the front and back. No errors emerge from passing TPs during straight driving and turning at intersections. The criticality map has analogous hazard zones at the curve segment (East = 180 m, North = 400 m). This is achieved with elliptic 2D Gaussians that bend along the curve when using the PMM.

A finding of great importance is that the red and yellow risk spots in the proximity of intersections are more localized. Overall, we observe that RSD is more selective and has fewer false positives than the other risk

indicators. Since the criticality maps always display the highest detected risk at each spot, and since by construction the same total number of incidents per bin appear in all criticality maps, a smaller area covered by red spots implies that many cases fall onto a single spot.

In that sense, the RSD criticality map, taking all surrounding cars into consideration, is the most specific one in terms of risk localization. Interestingly, we find a large portion of the most critical cases having small or no velocity (distribution at $\mu_{\mathrm{RSD},3} = 0\,\mathrm{m/sec}$ with $\sigma_{\mathrm{RSD},3} = 0.5\,\mathrm{m/sec}$ is observable). RSD rates risks as most critical in $b^*_{1,\mathrm{RSD}} = [1, 0.51]$ when standing in a traffic jam with TPs to the sides and in front, while another TP comes from the back with large $-\Delta v$.

In the other three bins $b^*_{2,\mathrm{RSD}} = [0.51, 0.29]$, $b^*_{3,\mathrm{RSD}} = [0.29, 0.11]$ and $b^*_{4,\mathrm{RSD}} = [0.11, 0.03]$, this relation is carried on and increases the respective boundary values compared to $b_{2,\mathrm{RSD}}$, $b_{3,\mathrm{RSD}}$ and $b_{4,\mathrm{RSD}}$. It describes the generic notion of higher collision probabilities for denser traffic. In the bin $b^*_{2,\mathrm{RSD}}$ the TTC distribution, and in $b^*_{3,\mathrm{RSD}}$ and $b^*_{4,\mathrm{RSD}}$ the TH distribution are superposed in a mixture of distribution.

2.7 Conclusion

In this chapter, we could see how driving risk models are developed and applied to situation assessment. In detail, the introduced measures included the heuristic risks TH, TTC and TTCE as well as the probabilistic measures Gaussian method and the survival analysis. All were modeled to deal with temporal and spatial uncertainty and to fulfill the normalization requirements for a general framework of collision risks.

In simulations of longitudinal and intersection scenarios, the survival analysis resulted in the highest performance in terms of early detection time in the crash cases as well as robustness with fewer false-positive detections in near- and non-crash cases. The Gaussian method and TTCE had similar accuracy, which is reasonable since their underlying equations could be shown to be also similar.

Afterwards, a novel risk model was proposed that combines the advantages of the Gaussian method and the survival analysis. This model will repeatedly be used and analyzed in the planning and warning chapters of this dissertation. The survival analysis is a theoretically justified risk model that separates events into critical and escape events and calculates a risk measure by integrating the probability to survive over the predicted time. In combination with the Gaussian method, driving uncertainties can

be more precisely modeled as they appear in real-world situations.

An application on criticality classification revealed that the model is able to cover different hazard categories: 1. dynamic stop and go in heavy traffic, 2. approaching with moderate velocity a car standing in front, and 3. keeping low distance with high speed to another car. On the contrary, TH and TTC are not able to extract the points on the criticality map to the same extent, but rather cover special cases.

As a summary, the research questions in the introduction of this chapter have been answered with the following contributions:

- **Uncertainty**: Novel risk framework for unifying and adapting state-of-the-art risk models to consider driving uncertainties [29].
- **Baseline Models**: Comparison of the conventional approaches in the task of accident detection (crash versus near-crash and no-crash cases).
- **Uncertainty**: Novel risk model that combines the advantages of the Gaussian method with the survival analysis. Furthermore, the positional uncertainties were extended for real-world situations [92].
- **Critical Situations**: Experiments with the application field of criticality classification in normal driving for the validation of automated cars.

For further information about the used datasets in this chapter (i.e., GIDAS and NGSIM), please refer to Appendix 6.1. In Appendix 6.2, it is shown that also a combination of TH and TTC cannot produce risk spots with similar qualities as the new risk model. Additionally, the performance decrease is analyzed if the uncertainty extensions are not applied.

In the next Chapter 3, an explanation is given of how the developed risk models can be used to plan intelligent ego motions in different situations and ensure hard thresholds for risks. The unique attribute of this risk-based planning is the single system approach that has accurate uncertainty considerations for driving.

3 Planning Generic and Tactical Motions

In this chapter, we turn towards the second component for self-driving cars: planning of intelligent motions in dynamic driving environments.

Concretely, we want to plan generic motions that handle a wide range of situations. They should be proactive in the sense that the action of others is incorporated (i.e., interaction-awareness). This allows the smooth reaction to changes in the environment. The motions should also be tactical. Hereby, tactical driving is defined as the planned ability to consider more than one action consecutively. For instance, a tactical motion could consist of initial acceleration, followed by traveling at constant velocity or even braking. The full action capacity of the ego car's in-motion intelligence is thus used.

Currently, assistance systems enable semi-autonomous driving at low velocities in parking areas or at high velocities on highways, but yet, not for moderate velocities in complex inner-city situations [6]. Systems for these situations are developed separately, mostly reactive and not predictive, and do not adapt to the driver's needs. Especially, when confronted by an intersection, the interplay of vehicles plays a dominant role and a wide range of risks is faced (e.g., collision, curve and traffic rules). This requires driving systems that employ behavior prediction for risk estimation and subsequently plan safe behaviors. Multiple scenarios must be considered and different risk sources holistically incorporated.

The emerging research questions that are solved with a risk-based planner in this chapter are:

1. **Generic Plans**: What is necessary for a planner to work not only on highways, but also in intersection scenarios? Could the generic risk of this dissertation help in this process?

2. **Driving Strategy**: What increase in driving proficiency can be achieved with predictive optimization instead of simple reactive related planners? Is the optimizer able to ensure no collisions in a wide range of driving?

3. **Interaction-awareness**: How is interaction-aware driving defined and what is necessary to model this aspect in planning? Could the situation complexity be avoided by an efficient implementation of the prediction?

The target is to control vehicles with planned velocities along a map path. Therefore, this chapter does not address path planning but rather velocity planning. However, path constraints are relaxed in Chapter 4 by considering path options discretely (e.g., turn left, turn right or drive straight). This efficiently supports the driver in real scenarios.

The following begins now with an evaluation of the relevant literature. A simple and common driver model is described in detail and its functionality extended for intersections. Using the novel risk model from the last Chapter 2, a new risk-based planner is then proposed. Experiments show that the presented planner is superior against state-of-the-art driving technologies and the planning framework is further improved to consider interaction-intensive situations, e.g., if there are given priorities between vehicles. Finally, analytical and statistical evaluations highlight the resulting generalizability of the approach.

3.1 Related Work

Motion planning involves breaking down a movement task into smaller consecutive maneuvers, which satisfy the constraint to be without collision, and optimizes the movement to be smooth and comfortable. Surveys of planning methods were conducted in [81], [58] and [107].

Very common planners are hereby traffic simulators. They can be divided into microscopic and macroscopic driving models. While microscopic models, such as the prominent Intelligent Driver Model (IDM), treat each cars' dynamics separately, macroscopic models look at indicators, such as traffic density, flow and average velocity with fluid dynamic equations [23]. A subclass of microscopic models are so-called cellular models, which discretize the time as well as space and thus work well for larger road networks [74].

Nevertheless, both types of traffic simulators are, in the end, only used for higher-level investigations. In contrast, common driver assistance systems in the market, e.g., Adaptive Cruise Control (ACC), control single time-based risks. In the case of ACC, the Time Headway (TH) is kept inside a valid range and always above a critical threshold.

More sophisticated and predictive planners are common in research for

self-driving cars. We classify the remaining state-of-the-art planning methods into three types: search-based methods, trajectory optimization and interaction-aware planning. They are subsequently described.

Search-based Planning

Search methods incrementally construct solutions in the search space, particularly, for path planning. Standard procedures find a path through static obstacle maps, but do not consider dynamic entities over time. For example, [15] combine task planning with spatial exploration to detect free areas with circle-shaped spatial probability propagation. For this reason, Optimal Reciprocal Collision Avoidance (ORCA) is applied in [123] to search the velocity instead of position space in all directions and retrieve a collision-free velocity vector. Similarly, the authors of [56] connect velocities along obstacle tangent points in the path-time space under acceleration bounds.

By comparison, graph-based searches overlay a grid on the driving space and at each grid point, the vehicle is allowed to move to adjacent grid points as long as they are free of obstacles. A set of actions are discretized and with A* [115], the shortest path can be found from the start to a goal grid point. Other search methods, such as Rapidly-Exploring Random Tree (RRT) [20], sample different points in the driving space and connect those in which no collision is detected. For high-dimensional problems, A* is too inefficient and RRT approximates a solution path.

Trajectory Optimization

We can consider the most promising approaches for generic and predictive planning to be methods with trajectory optimization.

In e.g. [35], velocity profiles of trapezoidal shapes are optimized with Model Predictive Control (MPC) and then evaluated against their proximity to dynamic obstacles, their smoothness and speed. Supplementary, [77] evaluates the costs of the ego car with risk and utility. As a risk indicator, the Time-To-Collision (TTC) zone is used. The crossing of intersections can also be learned based on potential fields and the Levenberg-Marquardt method [1]. Especially for T-intersections, [80] obtain personal critical gaps with Maximum Likelihood estimation and give out recommendations of safe gaps ahead.

MPC represents the planned trajectory with a dynamic function and a safe trajectory is found by minimizing a cost function [35]. Closely related,

potential fields consider the vehicle as a particle in a force field that combines attraction to the goal and repulsion from obstacles [1]. The collision-free trajectory can then be calculated using equations similar to electrostatics. In each of the cases, the convergence to a locally optimal trajectory is fast. However, the ability to find a globally optimal trajectory cannot be ensured, unless an appropriate initial guess is provided.

Reward-based algorithms are also optimizers by nature. They define that the vehicle can choose between different short actions in each position state. The results of the actions are thereby uncertain. When the vehicle reaches the target, it receives positive and if it collides, it gets negative rewards. Markov Decision Processes (MDP) are an example of a reward-based framework [108]. On the contrary, in state machines [40], whole discrete possible driving maneuvers are represented as a state. If specific conditions are fulfilled, the machine switches from one state to another, leading to a reactive behavior plan.

Interaction-aware Planning

The last family of planners are interaction-aware models. These planners consider not only the ego's risk, utility or comfort, but also costs from other cars, in order to find a solution for the complete traffic situation.

As mentioned, vehicle control along identical or parallel lanes is well established in the automotive industry. Still, the focus especially shifts from collision-free to likewise beneficial plans. For example, in platooning [40], the minimized cost functional includes the distance to all front vehicles to obtain more stable car-followings. Traffic light assists [119] create fuel-saving traverses during experienced phase switches (green, orange and red). When taking curves, [69] apply a proactive deceleration with driver models, and the course of lane changes are interpolated using Bezier curves in [95].

For crossing lanes with varying angles, possible driver intentions and ways of interaction become extensive. As a result, at an intersection, right-of-way matrices [33] typically fill each lane relation with passing orders and safe maneuvers are then coordinated between the actors via if-then transition of defined driving states. While doing so, a fuzzy system [66] could dynamically alter single entries of the priorities, e.g., on behalf of emergency vehicles. In contrast to conventional heuristics, [42] also employ priority graphs to construct continuous trajectories with safe gaps and [86] iterate through priority schemes for realizing orders even when each vehicle has to yield to another vehicle.

Recent research in cooperative planning takes the plans of other cars into account to locate global solutions. In [36], priority-based approaches are evaluated as most efficient, but they cannot handle all scenarios. As a comparison, Monte Carlo tree search [136] is used for lane merging with semantic vehicle-lane relationships, and the authors of [108] also tested MDP under mixed observability for unsignalized intersections. These methods lead to sensible ordering behaviors for specific scenarios which are implicitly influenced from the learned policy.

In conclusion, the planners in driving research are manifold and promising solutions are investigated. They are either model-based with analytic equations or leverage machine learning via extensive data. However, it should be emphasized that most approaches concentrate on path planning and consider velocity planning in a rather simple fashion. The focus lies on specially designed motions for single complex scenarios.

The proposed risk-based planner of this dissertation falls into the category of trajectory optimizers. With cycles of trajectory prediction and evaluation with the risk model, velocity profiles are planned. Specifically, intelligent motions are achieved that are generic, tactical and interaction-aware. The approach is compared in experiments with an improved version of the IDM, a traffic simulator for intersections.

3.2 Intersection IDM

This section describes the Intelligent Driver Model (IDM) as a common approach for velocity planning. We will modify the planner to be able to work in car-following and intersection scenarios. Consequently, the model is usable as a baseline in this dissertation.

The IDM [118] is a popular traffic model describing the dynamics of a leading and following car pair, driving along the same straight path. Its differential equation outputs a safe and efficient acceleration profile $\dot{v}_f$ for the follower

$$\dot{v}_f = \underbrace{a_m \left(1 - \left(\frac{v_f}{v_c}\right)^{\delta}\right)}_{\text{free term}} - \underbrace{a_m \left(\frac{d_b + v_f T}{d} + \frac{v_f(v_f - v_l)}{2d\sqrt{a_m b_{\text{des}}}}\right)^2}_{\text{interaction term}}. \tag{3.1}$$

In general, the following vehicle approaches the cruising velocity v_c with a maximum acceleration a_m and acceleration exponent δ in the free term. Nevertheless, once another vehicle is in front at a distance d with velocity v_l, the interaction term becomes dominant and reduces $\dot{v}_f$ to reach a

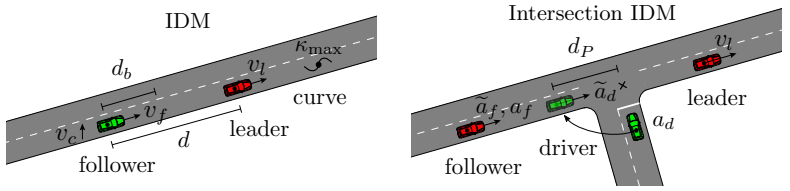

Figure 3.1: The IIDM is utilized as a baseline for velocity planning. Left: Planning with IDM and corresponding variables. Right: Extension to IIDM for intersections by assuming a hypothetical driver on the target lane.

balance point, defined by the minimum distance d_b and time headway T. The contained reaction time can be set with the desired deceleration b_{des}.

Similar to [69], we additionally account for curves in the path by filtering the maximal curvature $\kappa_{\max}$ in a curve segment $[\kappa_{\text{start}}, \kappa_{\text{end}}]$. The resulting maximal lateral acceleration a_y leads to an altered comfort velocity, which can be written as

$$v_c = \sqrt{a_y/\kappa_{\max}}, \text{ if } \kappa_{\text{start}} > \kappa_{\text{th}} \text{ and } \kappa_{\text{end}} < \kappa_{\text{th}}. \tag{3.2}$$

Hence, on sharp curve segments exceeding the threshold κ_{th}, the follower will converge to lower v_f due to the free term. In Figure 3.1, the variables of the IDM for a car-following scenario are given.

Extension for Intersections

Since the IDM is suitable for the simulation of longitudinal scenarios, it is often used in freeways. To improve the lateral applicability, lane-change decisions have been modeled in further research [60]. Here, the accelerations from the IDM are compared between a driver to the leader and a follower to the driver (i.e., we thus have a follower-driver-leader order). These computations are done for the current as well as the adjacent lane. If the acceleration gain of the driver is higher than the loss of his followers, a lane change is executed.

Henceforth, we align this idea for the crossing of intersections, where other cars are also influencing the ego car from a lateral direction.

First, we find the crossing point between the current and other path and obtain the distance d_P of the driver to the point. At segments close to the intersection, the position of the driver is projected and shifted along d_P, onto the other path. Then, we locate leading and following cars around this "hypothetical" driver.

The IDM equations are taken for the driver relative to the leader and the follower relative to the driver to retrieve $\widetilde{a}_d$ and $\widetilde{a}_f$, respectively. For the case of the current path, we assume constant acceleration for the follower a_f and a stopping maneuver for the driver a_d. If the incentive criterion

$$\underbrace{\widetilde{a}_d - a_d}_{\text{driver}} + p \underbrace{(\widetilde{a}_f - a_f)}_{\text{follower}} > \Delta a_{\text{th}} \tag{3.3}$$

is not fulfilled, the driver performs a_d. Otherwise the driver crosses the intersection by using $\widetilde{a}_d$. The politeness factor p and threshold Δa_{th} serve to parametrize the needed advantage at which the driver will pass in front of the following vehicle.

With the incentive criterion alone, it is possible that the follower crashes into the driver for small p. We therefore simultaneously check the following safety criterion

$$\widetilde{a}_f \geq b_s \tag{3.4}$$

with a safe deceleration b_s. Overall, the resulting method is the Intersection IDM (i.e., in short IIDM). In this respect, Figure 3.1 also gives a summary of the improved version with its projection idea and the involved cars, applied on a T-intersection scenario.

3.3 Proposed Risk Optimization Method

As an introduction, we defined an extended version of the IIDM for velocity planning in the previous section. Hereinafter, the novel planner is presented, which optimizes risks with parametrized velocity functions. The planner is called Risk Optimization Method (ROPT) [94].

At each point in time during the simulation, ROPT receives information about the current state of the traffic environment. This includes latest measured position and velocity of all participants as well as their associated future paths, extracted from map data. After the planning of velocities, ROPT executes the motion solution and starts again.

In the motion planning of ROPT, we target to predict an optimal *velocity profile for the ego car* along the path (see Figure 3.2). This implies that we do not determine the acceleration for the next time step, but we calculate a velocity profile, respectively, trajectory for a future time horizon. Initially, ROPT extrapolates one trajectory for each other car over the time horizon. For such cars, ROPT uses a constant velocity assumption. In comparison, we will see in Section 3.5 that improved models are also leveraged, e.g.,

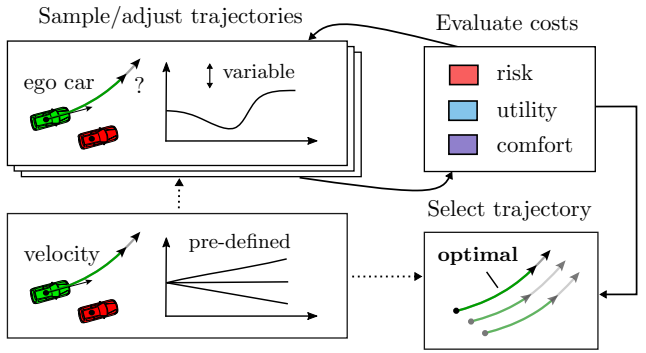

Figure 3.2: Risk Optimization Method (ROPT). The planner generates multiple trajectories for the ego car and optimizes them based on risk, utility and comfort. A single, optimal trajectory should be the output of ROPT.

constant acceleration and variants. After the trajectory prediction of other vehicles, the main steps of ROPT can be applied, illustrated in Figure 3.2. The steps of ROPT are the following:

- Sampling of trajectories for the car that is controlled by ROPT, i.e., named the ego agent.
- Evaluating costs, which includes the induced risks from other cars as a cost factor.
- Selecting one trajectory out of set of trajectories which has minimal cost and which will be executed.

In detail, for the trajectory sampling step, the planner creates two types of trajectories. These are trajectories that are further optimized and fixed, pre-defined trajectories. For the optimized trajectories, we can use velocity profiles with variable parameters, changing the shape of the profile. In that respect, the planner switches between adjusting the parameter values and evaluating a cost function that rates the trajectory.

The cost evaluation step for each ego trajectory then includes besides the *integral risk* (caused by both collisions and curves), also the *utility* (distance travelled) and *comfort* (strength and frequency of a velocity change). They are both cost factors, which are ego-centric and have to be balanced with the risk factor.

In the end, one of all the previously sampled trajectories is selected in ROPT as the best trajectory. Note that, in this step, the final optimized

trajectories are compared with the fixed trajectories. Since an optimization algorithm is applied, the first and second step within ROPT are intertwined. The steps are repeated until a cost optimum is found.

In the next sections, the reasons as well as implementation of the velocity profiles are explained first and, subsequently, we integrate the risk model of this dissertation with the formulas for utility and comfort. The goal is to develop a generic planner for motions.

3.3.1 Trajectory Optimization

For scenarios with only one dominant risk source (i.e., curve driving with curve risk or longitudinal following and crossing with collision risk), basic trajectory sampling methods are usually sufficient. Accordingly, in previous research, the Foresighted Driver Model (FDM) was developed in [28]. It samples, using gradient descent, a single acceleration and deceleration profile and balances risk with utility to find correct ego behaviors.

However, lane merge-ins and turning at intersections are more complex scenarios, in which the planned velocity profile must obey maximum curve speed and match the speed of the traffic flow. The FDM would include the risks in planning but converge to the local minimum from the curve risk. It is necessary to generate a tactical velocity profile that successfully considers multiple risk sources. In a different approach, velocity paths were incrementally constructed through risk graphs with Rapidly-Exploring Random Trees (RRT) [20]. RRT was shown to work in intersections with multiple other cars, but the stepwise evaluation makes it harder to constrain the convergence of the solution to specific simple trajectories.

With ROPT, problems of many local minima are circumvented by employing two strategies. On the one hand, we will optimize parametrized velocity profiles that enable to change the plans over the predicted time. Consequently, we can perform combinations of acceleration and deceleration phases. On the other hand, we sample a set of trajectories in which one is likely falling into the global minima. Global minimality cannot be ensured, but it is shown in practice, that this assumption holds true.

Double Ramp Profiles

The chosen parametrized velocity profiles consist of two consecutive acceleration and deceleration "ramps" (see left-hand side of Figure 3.3). Ramps are linear segments where the velocity is changed. A **double-ramp profile** is described by the end velocities of the first and the second ramp,

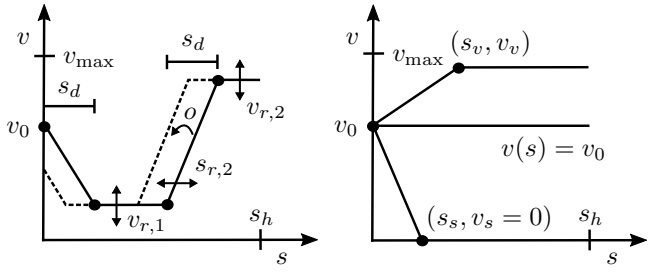

Figure 3.3: The two types of ego trajectories in ROPT, allowing the planning of generic and robust solutions. Left: Optimization of velocity ramps for predicted times. Right: Fixed standard trajectories.

$v_{r,1}$ and $v_{r,2}$, and the start time of the second ramp $s_{r,2}$. Each ramp has a fixed duration $s_d = 2.5\,\text{sec}$, whereas the first ramp starts with the current ego velocity v_0.

Note hereby that we are talking about the predicted time s. We, consequently, have a complete velocity profile $v(s)$ that is planned for the future of the ego car until a prediction horizon of s_h.

On top of this, the Nelder-Mead optimization algorithm [78] is leveraged for the optimization of the parameters. The optimizer is a downhill simplex type of method, which does not require gradient information for the costs. In each of the optimization iterations, the simple double-ramp velocity profile is converted into a trajectory, which, in turn, is evaluated for risk and ego benefits. Trajectories that violate **constraints on velocity or acceleration** are penalized with soft constraints. As the Nelder-Mead optimization is a local search, it depends on the initial value of the parameters. Hence, ROPT simultaneously optimizes N_t trajectories, starting from different initial values $v_{r,1}^m = v_{r,2}^m = v_r^m$, with $m \in 1, \ldots, N_t$ and v_r^m evenly spaced in $[2\,\text{m/sec}, v_{\max}]$.

If trajectory m was chosen in the previous time step, the optimization continues with its previous parameters. Additionally, it shifts the beginnings and ends of all ramps by an offset o that corresponds to the total time that the trajectory has been selected.[1]

Supplementary to the optimized trajectories, ROPT samples three **fixed trajectories** (see right-hand side of Figure 3.3): a constant speed trajectory with $v(s) = v_0$, a stopping trajectory with the end point of $(s_s, v_s = 0)$

[1] Once a trajectory has been active for times equal to the ramp duration ($o = s_d$), we insert a new ramp after ramp two.

and an acceleration trajectory to the end velocity $v_v < v_{\max}$ at the predicted time s_v. Such trajectories represent safe fallbacks, in the case that the optimized plans do not lead to proper solutions. This can occur from either not performing enough iterations to fine-tune the solution or from local minima issues in the cost landscape.

In the next section, the cost evaluation for the set of ego trajectories is outlined. For this purpose, we modify the risk prediction from Chapter 2 within ROPT. We enhance the equations to properly account for collision as well as curve risks. Furthermore, the severity part within risks is introduced. The final cost function for planning consists of those risks and own utility and comfort.

3.3.2 Trajectory Evaluation

In general, risk is the probability of a critical event multiplied with its possible damage outcome. In this context, the probability function $P_{\text{crit}}(s;t,\Delta t)$ for critical events is defined from the current time t around the future $t+s$ during an interval of size Δt. In Section 2.2.3, this critical probability was referred to as $P_{\text{E}}(s;t,\Delta t)$. Now, the risk measure $R(t)$ is the accumulated risk contained in $P_{\text{crit}}(s;t,\Delta t)$. By also weighting $P_{\text{crit}}(s;t,\Delta t)$ with the predicted damage of the event $D_{\text{crit}}(s;t,\Delta t)$, we lastly obtain risk as the expected severity.

ROPT combines the Gaussian method [37] for the estimation of critical event probabilities with the statistical survival analysis [26] to gain the overall risk. In the following, we consider the recurring general and simple situation evolution of this dissertation: an ego car (in green) encountering another car (in red), indexed with $i=1,2$, as depicted in Figure 3.4.

Details of Risk Prediction

Trajectories are predicted with assumptions, but cars will not strictly follow these trajectories. In reality, they undergo variations in position, speed and heading, which we defined as uncertainties (see Section 2.4.2). For this reason, we model positions with a normal distribution f_i. As we predict the cars to drive along map paths, the longitudinal uncertainty along the path is moreover defined to be higher than the lateral uncertainty. In this way, we obtain *2D ellipses* that are specified by an uncertainty matrix $\boldsymbol{\Sigma}_i^{'}$ around mean positions $\boldsymbol{\mu}_i$.

A collision occurs if two vehicles coincide at the same position, which is analog to the product of the two Gaussian functions $f_{\text{coll}} = f_1 f_2$. The

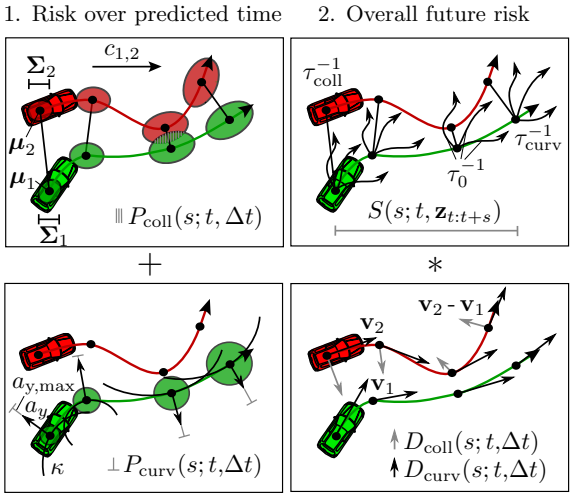

Figure 3.4: The planner is based on the novel risk model of this dissertation. Left: Gaussian method for collision and curve probability. Right: Survival analysis with damage terms, which are introduced in the end of this section.

collision probability is eventually given by spatially integrating f_{coll} over all positions, i.e.,

$$P_{\text{coll}}(s;t,\Delta t) = \det|2\pi(\boldsymbol{\Sigma}_1+\boldsymbol{\Sigma}_2)|^{-\frac{1}{2}} * \exp\{-\frac{1}{2}(\boldsymbol{\mu}_2-\boldsymbol{\mu}_1)^T(\boldsymbol{\Sigma}_1+\boldsymbol{\Sigma}_2)^{-1}(\boldsymbol{\mu}_2-\boldsymbol{\mu}_1)\}.$$

Due to temporal uncertainty, the deviation of real trajectories from the predicted trajectory differs with increasing prediction times. As a reminder, the current state of cars is extrapolated with kinematics to obtain trajectories. Their longitudinal position on the path l_i is increased with Δl_i, depending on the velocities v_i via

$$l_i(s+1) = l_i(s) + \Delta l_i = l_i(s) + v_i(s)\Delta s.$$

For the current time, $t = 0$ is imposed. The growth of spatial uncertainty can thus be derived from a velocity uncertainty factor c_i using

$$\sigma_{i,v}(s+1) := \sigma_{i,v}(s) + c_i v_i(s)\Delta s.$$

In this process, the variable Δs describes the prediction step size.

For the ego car, the **probability to drive off at curves** is formulated similarly. Here, we assume 1D circles with uncertainty σ_1 and look at the lateral acceleration

$$a_y(s) = \sqrt{\kappa(s) v_1(s)},$$

which is influenced by the curvature of the road κ. We then compare a_y with its maximal possible value $a_{\mathrm{y,max}}$ from vehicle dynamics constraints to obtain

$$P_{\mathrm{curv}}(s;t,\Delta t) = \frac{1}{\sqrt{2\pi\sigma_1^2}} \exp\left\{-\frac{\max(a_{\mathrm{y,max}} - |a_y|,0)^2}{2\sigma_1^2}\right\}. \tag{3.5}$$

If $|a_y|$ approaches $a_{\mathrm{y,max}}$, the probability $P_{\mathrm{curv}}(s;t,\Delta t)$ will accordingly increase. Note that on the right side of Equation (3.5), we did not explicitly write the dependency of the variables to the times t, s and Δt. This curve risk can also be easily extended to include longitudinal accelerations a_x, rendering it in a more realistic manner.

Next, in the survival analysis, accident occurrences are modeled as a thresholding process based on Poisson-like event probabilities.

A Poisson process is, as we know, influenced by a situation state-dependent event rate $\tau^{-1}(\mathbf{z}_{t:t+s})$, which models the mean time between events and consists of a total critical event rate $\tau_{\mathrm{crit}}^{-1}$ and a risk-avoiding general escape rate τ_0^{-1}. For $\tau_{\mathrm{crit}}^{-1}$, we consider collision risks represented by event rates $\tau_{\mathrm{coll}}^{-1}$ and risks of losing control in curves $\tau_{\mathrm{curv}}^{-1}$, which leads to[2]

$$\tau^{-1}(\mathbf{z}_{t:t+s}) = \tau_0^{-1} + \tau_{\mathrm{crit}}^{-1} = \tau_0^{-1} + \tau_{\mathrm{coll}}^{-1} + \tau_{\mathrm{curv}}^{-1}$$

with the included terms of

$$\tau_{\mathrm{crit}}^{-1}(\mathbf{z}_{t:t+s}) = P_{\mathrm{crit}}(s;t,\Delta t)/\Delta t,$$
$$P_{\mathrm{crit}}(s;t,\Delta t) = P_{\mathrm{coll}}(s;t,\Delta t) + P_{\mathrm{curv}}(s;t,\Delta t).$$

The core part of the risk model is the survival function. Concretely, it indicates the likelihood that the vehicle will "survive" accident events in the time interval $[t,t+s]$, in compliance with

$$S(s;t,\mathbf{z}_{t:t+s}) = \exp\{-\int_0^s \tau^{-1}(\mathbf{z}_{t:t+s'})\, ds'\}.$$

It has been empirically shown that statistics of human injury in traffic accidents depend on the acting velocity vectors $\mathbf{v}_i$ and has the form of a

[2] For simplicity, we describe here only collision events for a single car pair. Several car pairs may be added straightforwardly in the equations.

logistic function [16]. We postulate that the damage of the involved cars has the same qualitative relationship. **Car-to-car collision** and **curve control loss damage** is given by

$$D_{\text{coll}}(s;t,\Delta t) = \frac{D_{\text{max,coll}}}{1+\exp\{k_{\text{coll}}(\|\mathbf{v}_2-\mathbf{v}_1\|-\eta_{\text{coll}})\}},$$

$$D_{\text{curv}}(s;t,\Delta t) = \frac{D_{\text{max,curv}}}{1+\exp\{k_{\text{curv}}(\|\mathbf{v}_1\|-\eta_{\text{curv}})\}},$$

where the parameter k is the damage increase factor, η the damage midpoint and $D_{\max}$ the maximal damage. As a result, we acquire the overall risk engaging in a future critical event by temporally integrating the term of probabilities, damages and survival function

$$R(t) = \int_0^\infty (\tau_{\text{coll}}^{-1} D_{\text{coll}} + \tau_{\text{curv}}^{-1} D_{\text{curv}}) S \, ds.$$

Utility and Comfort Prediction

Now, the cost functional of ROPT is described. Drivers naturally do not only minimize risks but maximize own benefit as well. We consider benefit as the needed time to arrive at the goal and the comfort of the travel. The former is defined by the ego velocity course v_1 and the latter takes the acceleration and jerk profile a_1 and j_1 into account. In ROPT, we weight the components with driver-specific constants b^t, b^c and b^j and compute the integral future benefit with

$$B(t) = \int_0^\infty (b^t|v_1| - b^c|a_1| - b^j|j_1|) S \, ds. \tag{3.6}$$

For higher s, we also consider the **survival function** S in the evaluation. In this way, high-risk situations result in lower $B(t)$. This is a big difference to other planning approaches in the literature. Risk is dampening the benefits over time, which makes the planning smoother and more robust towards noise.

Finally, ROPT evaluates the cost function for each generated trajectory

$$C(t) = R(t) - B(t)$$

and executes the trajectory with the lowest $C(t)$. For comparability, $R(t)$ and $B(t)$ have to be transformed into the same unit. Since the severity factors in $R(t)$ require a monetarization, we use € for $C(t)$.

3.4 Comparison with Intersection IDM

For the simulations, IIDM (i.e., Intersection Intelligent Driver Model) and ROPT (i.e., Risk Optimization Method) face the complex task of planning merge-ins at T-intersections and their performances are compared with a statistical analysis. Merge-in support systems must estimate the chances that a gap between two consecutive vehicles can be taken successfully.

Experiments were conducted using an own customized version of the software Simulation of Urban Mobility (in short, SUMO) [5]. With SUMO, we can create arbitrary path combinations and then apply the planners to drive multiple vehicles. Start and end states of the cars are selectable in the simulation. In this sense, ideal environment conditions are assumed, in which position and velocity of the involved cars are synchronized and known without sensor errors.

Figure 3.5 illustrates the ego car waiting at the stop line, while other cars pass from left to right. The road traffic is modeled as a Poisson distribution $P(\lambda)$ with different intervals λ, and the number of missed gaps n_{gap} is counted. For continuous intervals λ, we add an offset with uniform random noise in the range of $[-0.5\,\text{sec}, 0.5\,\text{sec}]$. If the gap size t_{gap} is large enough, the ego car needs to increase v until the maximum curve velocity and subsequently accelerate even further to match the velocity of the traffic

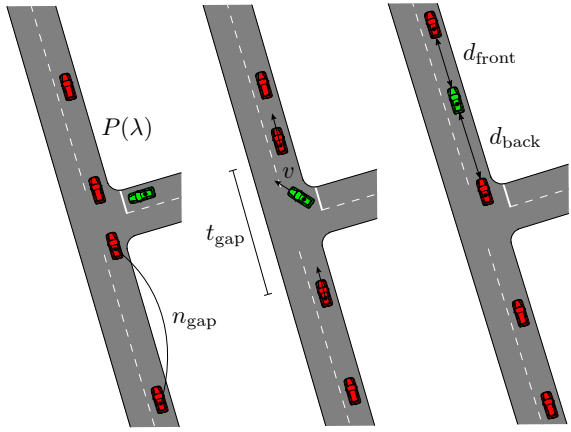

Figure 3.5: Time lapse of T-intersection merge-in scenario and relevant evaluation variables. The ego car (green) must wait to find a safe gap in the traffic flow (red) in order to take the turn successfully. This task is difficult due to the risk caused by the cars and the curve.

flow. The angle of the right turn with 90 ° and the constant traffic speed with $v_t = 10\,\mathrm{m/sec}$ are chosen in a way that makes single-ramp trajectories only feasible for very large gaps of more than 100 m.

Besides systematically changing λ, which results in different traffic densities, we modify the politeness factor p of IIDM and the travel benefit b^t for ROPT to control the driving behavior. For every model and traffic setting, we then run several simulations (ca. 200) to ensure statistical significance. Once the ego car advances past the stop line, the merge-in procedure starts and the encountered distances to the front d_{front} and rear vehicle d_{back} are recorded. In the evaluation, we look at the mean minimal distances $\bar{d}_{\mathrm{back,min}}$ and $\bar{d}_{\mathrm{front,min}}$, which indicate risk, as well as the mean utility indicators $\bar{n}_{\mathrm{gap}}$ and $\bar{t}_{\mathrm{gap}}$.

Results

Figure 3.6 plots the distances $\bar{d}_{\mathrm{back,min}}$ and $\bar{d}_{\mathrm{front,min}}$ for IIDM with $p \in [0.5, 4]$ and also ROPT with $b^t \in [0.1\,€/\mathrm{km}, 10\,€/\mathrm{km}]$ for different $\lambda \in [2\,\mathrm{sec}, 5\,\mathrm{sec}]$. As expected, $\bar{d}_{\mathrm{back,min}}$ decreases with lower λ for both models. Likewise, it declines with lower p, or higher b^t. The ego car enters smaller gaps, where the rear vehicle approaches more closely due to its excess speed. More importantly, the lower bound for $\bar{d}_{\mathrm{back,min}}$ over all runs (purple line) is always 0 in the case of the IIDM. Merge-ins cannot be performed properly, with accidents happening in 37 % of the runs. Here, a crash or near-crash case is assumed, if the distance between two cars is less than 1 m.

The reason can be identified in the simulation: At time step t_{start}, IIDM decides to merge, based on the incentive criterion of Equation (3.3). However, due to the restricted curve speed, it cannot accelerate quickly enough to be in front of the follower. At a later time step t_{end}, the criterion becomes invalid and a brake is executed. If the resulting stopping position is in the way of the crossing traffic, an accident occurs. In contrast, the lower bound of ROPT stays above 15 m. ROPT does not expect any deceleration but assumes constant velocity for the follower and yields consistently safe behavior.

Regarding $\bar{d}_{\mathrm{front,min}}$, only runs with safe mergers are listed for IIDM (otherwise there is no notion of a back car). While $\bar{d}_{\mathrm{front,min}}$ is approximately the same for IIDM and ROPT with different λ, it also descends with smaller p or bigger b^t. In ROPT, bigger b^t lead to less weight on risk and earlier merge-in begins. With IIDM, d_{front} is usually constant. Equation (3.3) does not consider the front car. Nevertheless, due to un-

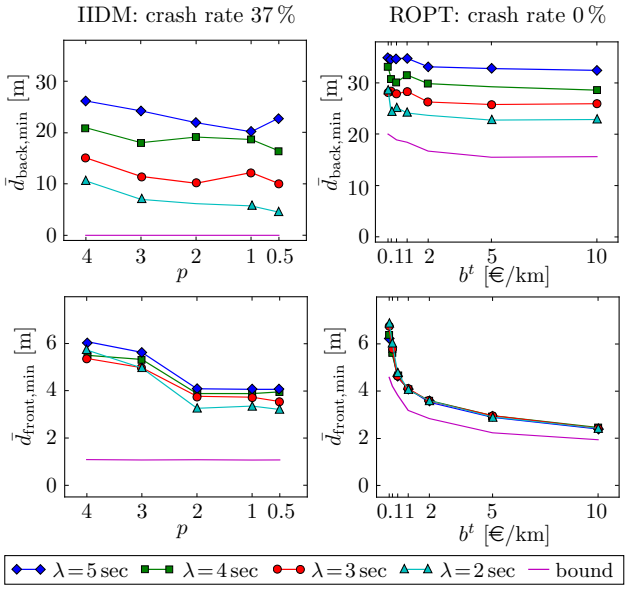

Figure 3.6: Comparison of IIDM and ROPT with risk indicators. We analyze the average experienced distances to the front and back car on the intersection for different traffic densities and model parametrizations. In ROPT, no collision occurred during any run.

successful merge-in attempts at previous gaps, it is possible that the ego car will have already advanced into the intersection.

Figure 3.7 pictures $\bar{n}_{\text{gap}}$ and $\bar{t}_{\text{gap}}$ with the same parameter variations. Both models choose earlier gaps (less $\bar{n}_{\text{gap}}$), based on willingness of driving into lower $\bar{t}_{\text{gap}}$, with decreasing p or increasing b^t. In general, IIDM has sharper slopes, because of its risk-proneness. For dense traffic settings (small λ), the models need to wait longer with higher $\bar{n}_{\text{gap}}$ and the taken gap has lower $\bar{t}_{\text{gap}}$. In total, $\bar{t}_{\text{gap}}$ lies in between $4-7$ sec which are realistic critical gaps [80].

Further Discussion

We saw that IIDM is unable to cope with merge-in scenarios. It implicitly predicts the behavior of the driver and leading vehicle but lacks a proper prediction of the follower. For this reason, we extend the IIDM further. In the spirit of ROPT, we explicitly extrapolate trajectories of other cars

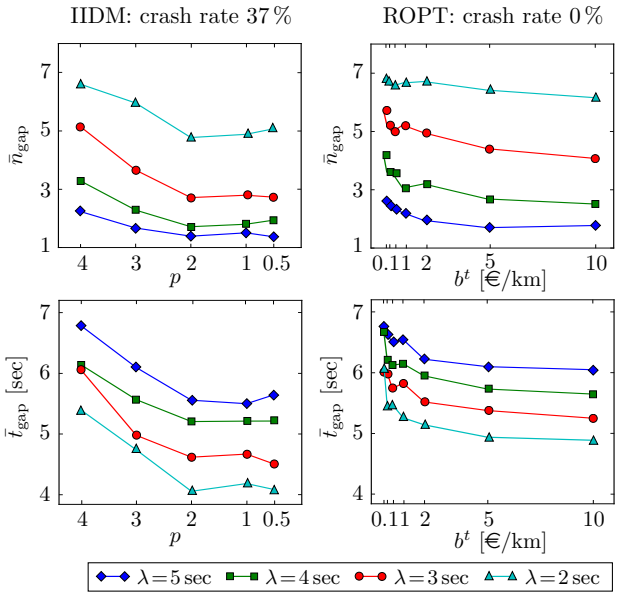

Figure 3.7: Comparison of IIDM and ROPT with utility indicators. Average taken number of gap and gap size show that both models achieve similar utility gains in the merge-in.

with constant velocity and the ego trajectory using Equations (3.1) and (3.2). At every predicted step, the criterions (3.3) and (3.4) are evaluated. Only if both hold true for the complete horizon, a merge-in is started. The outputs of the predictive IIDM are shown in Figure 3.8. The lower bound for $\bar{d}_{\text{back,min}}$ approaches 14 m for $p \to 0.5$, because b_s is set to $-2\,\text{m/sec}^2$ in the safety criterion. The courses of $\bar{d}_{\text{front,min}}$ are constant 7 m for different λ. Independent of p and λ, a safe behavior is achieved. Additionally, $\bar{n}_{\text{gap}}$ and $\bar{t}_{\text{gap}}$ are shifted to higher values, similar to ROPT.

Still, ROPT distributes the available t_{gap} more to d_{front} than d_{back}, while keeping absolute risks low (i.e., no crash happens) for varying b^t and λ.

As opposed to this, IIDM is not able to cap the risk with its heuristics. The predictive IIDM generates safe behavior, but different p can only change its influence on d_{back}. This is undesirable, because during merge-ins an entering car can generally react easier to unexpected behaviors of a front car than a back car. Moreover, neither versions of IIDM can derive a continuous acceleration course. They only switch between braking or

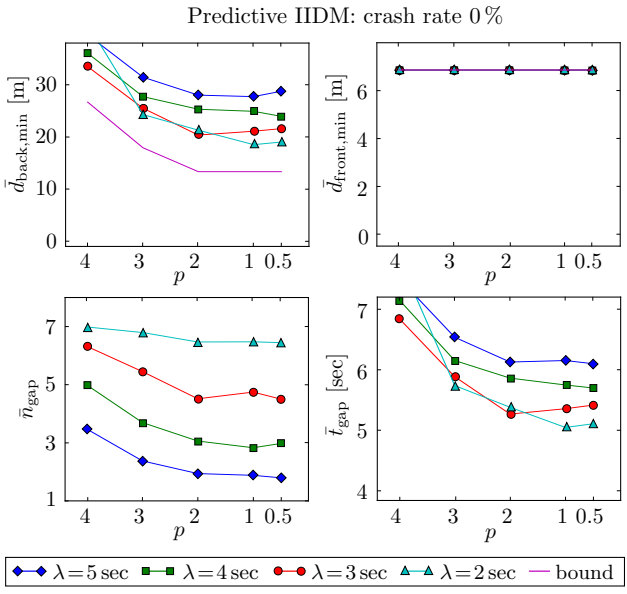

Figure 3.8: Predictive IIDM with risk and utility indicators. By checking the merge-in criterions of the IIDM in an explicit prediction, a crash rate of 0 % can be achieved as well.

following the front car. In contrast, ROPT possesses an explicit risk-utility tradeoff (i.e., making better use of t_{gap}).

For intersections of arbitrary geometry and traffic constellations, ROPT works out-of-the-box due to its instantaneous adaptation using a holistic predictive risk model. For instance, a crossing of X-intersections could also be handled in the predictive IIDM with additional go/no-go decisions. However, lateral risks are only considered shortly before the intersection and not afterwards. In complex interactions between ego and other cars, the combination of heuristics will potentially lead to unsafe behavior.

3.5 Considering Traffic Rules in Planning

To recap, we have now derived two velocity planners along driving paths: the basic IIDM and a novel risk-based planner. The termed Risk Optimization Method (ROPT) turned out to create more generic and tactical motions, which can be parametrized to obtain a distinct driving style. In

what follows, we will build upon ROPT. Particularly, the approach is improved to handle scenarios, where traffic rules can support the predictive planning and enable interaction-aware solutions.

Research Motivation

At intersections (e.g., Y-, T- and X-junctions) or highway merging (entering plus leaving ramps and overtaking), the driving task is frequently simplified with traffic codes for prioritization [76].

For instance, for cars, there are rights-of-way, stop lines or traffic lights. Even in simpler longitudinal scenarios, traffic participants should follow the direction of travel, keep on one side for multi-track streets and obey speed limits. These regulatory risks not only define who goes first, but also constrain the agents in their choice of actions and thereby make driving safer. Intentions become transparent and accessible for other agents so that they can be potentially leveraged for motion planning to improve the finding process of predictive solutions.

Earlier works model rules with state machines [11], ordering suitable maneuvers (keep distance, drive inside, etc.) based on current kinematics of vehicle pairs. For crossroads, the system should give way or come to a stop when an obstacle takes precedence. This works reliably in normal driving, in critical conditions, however, it may generate reactive solutions that solely center on safety, effectively neglecting driver comfort. For dealing robustly with a variable interplay of cars, it is therefore better to control behaviors dynamically using prediction-evaluation cycles that directly incorporate priority in the behavior finding procedure.

Summary

This section addresses the research problem of asymmetry in driving interactions. To this end, an underlying *planning framework* will be presented by looking more closely at the trajectory optimization problem. The chosen velocity profiles are enhanced to give ROPT more freedom and to fully optimize the comfort for the ego driver. An order assignment allows the identification of the *vehicle-to-vehicle interaction* (i.e., front, back, right and left geometric relations). Depending on the arising priority, ROPT then changes *the prediction model of other cars*.

Large-scale simulations highlight that the risk-based planner successfully approaches, crosses and leaves uncontrolled intersections with varying behaviors and taken paths of encountered traffic participants. In summary,

ROPT is hence able to achieve robust interaction-aware driving motions. This is considered to be a major success of the given dissertation.

3.5.1 Planning Framework

Motion planning in structured environments is essentially tackled by searching the velocity space v over the future time s. Structured environments are driving spaces where cars can follow map geometries. As depicted in Figure 3.9, the risk-based planner receives input signals of positions $\mathbf{x}_i$, velocities v_i and of given map paths for a green ego car and N_o other red cars.[3] Without prior knowledge, other trajectories are predicted on their respective paths with constant velocity up to a prediction horizon s_h. This aligns to the description in Section 3.3.

In the main planning step, parameters $\boldsymbol{\theta}$ from multiple velocity profiles v^m are optimized for the ego agent. Therefore, an alternation is performed between adjusting $\boldsymbol{\theta}$ and evaluating risks $R(t)$, utility $U(t)$ and comfort $O(t)$ of the arising dynamic scene for time t. Once a defined cost threshold is satisfied for each sample, the trajectory v^m with the lowest cost is chosen and executed within a time step Δt to obtain the accelerations a_i and jerks r_i. In doing so, the simulator updates other vehicles from measured fixed trajectories or may also control them with their own planners. By applying planners to each car, we can simulate action and reaction relations with the planning framework.

Optimization Problem

For driving scenarios with more than one risk source, the cost functional is non-convex. Hills and valleys with diverging sizes can occur in the costs. To avoid local minima, velocity shapes with high degrees of freedom are necessary. We can take arbitrary velocity profiles and optimize them based on the risk models. Instead of employing double ramps as velocity curves, we now choose so-called **"snake" velocity profiles** for allowing more flexibility in the planning. We select $n = 4$ segments, having fixed length $s_l = 2.5\,\text{sec}$, but variable end velocities $v_{p,n}$.

On the left of Figure 3.10 one snake profile is plotted, whereby p stands for one parameter in the parameter set $\boldsymbol{\theta}$. The snake shape leads to more interaction-awareness and thus to better tactical maneuvers. Because $v(s)$

[3] A traffic situation consists in this way of $N_o + 1$ participants, indexed with i. Hereby, the other cars are subscripted with j.

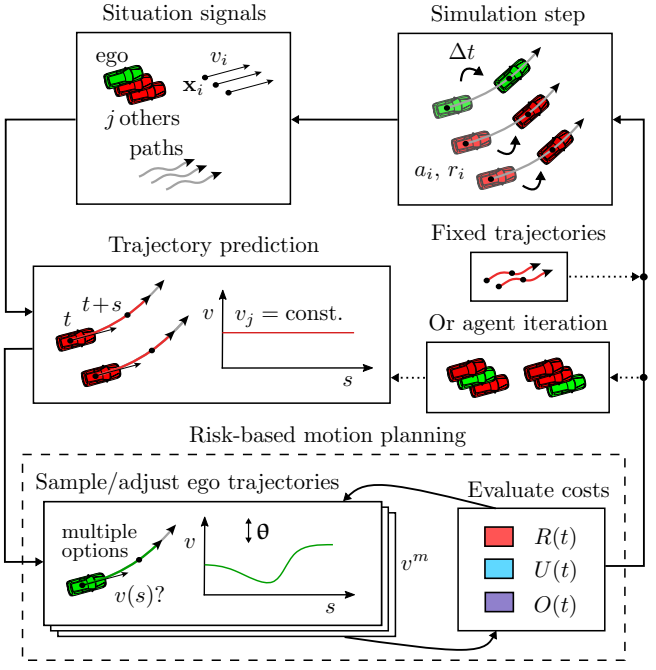

Figure 3.9: Concept of predictive velocity optimization. The assessment of situation costs follows an adaptation of ego velocity profiles in repetitive cycles. For further details of the optimization problem, please refer to the text.

is discontinuous, we moreover introduce an **adjustable first lag** $\lambda_{p,0}$ in the acceleration a_0. Figure 3.10 shows on the right that the following ramp transitions are also smoothed with a Gaussian filter, denoted as h_g.

In this section, ROPT uses the non-gradient Powell's optimization method [88], which iteratively fits for $\mathbf{\theta}$ a quadratic function to three evaluation points and finds the function's vertex. Soft constraints are set with penalizations for exceeding the minimal/maximal values $v_{\max}$, $\lambda_{\min}$, $a_{\min}$ and $a_{\max}$. Altogether, the optimization problem can be formulated as

$$\min_{\mathbf{\theta}} \underbrace{f(v_{p,1}, v_{p,2}, v_{p,3}, v_{p,4}, \lambda_{p,0})}_{\text{decision variables } \mathbf{\theta}} = \min_{\mathbf{\theta}} \underbrace{R(t) - U(t) - O(t)}_{\text{fitness function } f},$$
$$\text{subject to } v_{p,n} \leq v_{\max}, \;\; \lambda_{p,0} \geq \lambda_{\min}, \;\; a_{\min} \leq a_{p,n} \leq a_{\max}$$

with segment accelerations $a_{p,n}$. A suitable ego maneuver is usually at-

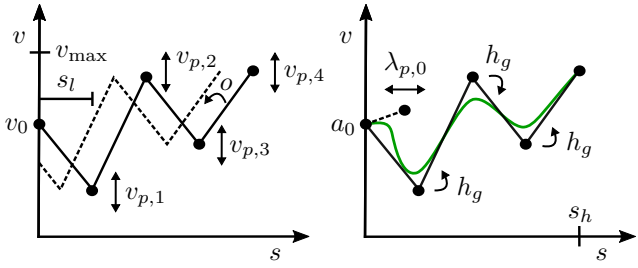

Figure 3.10: Left: Parameters and shift of velocity snake. Right: Lag implementation and corner smoothing. For targeting interaction-aware planning, we increase the freedom for ROPT to optimize the comfort for the ego driver.

tained in less than 30 evaluations. If not, we force the termination after a fixed evaluation number.

Besides the optimized snakes, we still sample the three fixed trajectories, from Section 3.3, in our improved implementation of ROPT. All trajectories are evaluated in terms of their fitness, after which one is selected for behavior execution. Here, we introduce a hysteresis so that a switch to a different trajectory v^m is done when the risk $R(t)$ of the new trajectory is relatively and absolutely smaller for a set period of time.

In conclusion, the given optimization represents a multi-objective problem (i.e., minimize risk and maximize utility and comfort). Additionally, the optimization is constrained in the velocity, acceleration and actuation lag. We have a stochastic model with uncertainties from the risks, and, lastly, the cost function is non-convex. Local minima are circumvented by sampling multiple ego profiles. In the upcoming sections, extensions for ROPT to handle interaction-aware driving situations will be further described.

Smoothing Snake Function

If we assume instantaneous actuation with fixed direct velocity points, as it was done in Section 3.3, we may create trajectories that are unfeasible in real vehicles. The effective jerk $r(s)$ from $v(s)$, however, requires a continuous velocity curve. For this reason, we extrapolate the initial acceleration a_0 for the time $\lambda_{p,0}$ and blend its velocity line with the old ramp $(v_0,v_{p,1})$ according to

$$v(s) = v_0 + (\lambda_{p,0} - s)a_0 + \frac{s}{s_l}(v_{p,1} - v_0),$$

whereby $s \in [0, \lambda_{p,0}]$. Afterwards, $v(s)$ is convoluted for the complete prediction interval s_h with a Gaussian function

$$h_g(s) = N_f(\sigma_s^2, \mu_s = 0).$$

We set the variance σ_s^2 and use $\mu_s = 0$ to achieve further smoothness of the overall velocity curve without overshooting. As the derivatives $a(s)$ and $r(s)$ are numerically recalculated after the smoothing steps for $v(s)$, errors from the Gaussian smoothing are prevented.

By optimizing $\lambda_{p,0}$, ROPT is allowed to influence the course of the jerk $r(s)$ (i.e., gradual actuation). Limits for $\lambda_{p,0}$ must also be enforced when high-risk situations occur. The average brake lag for sudden deceleration from 0 to $a_{\min}$ amounts to $\lambda_b = 0.4\,\text{sec}$, and engine acceleration to $a_{\max}$ takes $\lambda_e = 0.8\,\text{sec}$.[4] With this in mind, we qualify the minimal lag threshold $\lambda_{\min}$ depending on the acceleration a_0 as

$$\text{if } a_0 \geq 0\text{: } \lambda_{\min} = \frac{a_0}{a_{\max}}\lambda_e, \quad \text{else: } \lambda_{\min} = |\frac{a_0}{a_{\min}}|\lambda_b.$$

Compared to employing continuous polynomials from the start, our modified snake behaves smoothly and does not require the solution of a linear equation system to map $\boldsymbol{\theta}$ to the function shape. In risky scenarios, ROPT focuses on minimizing risks, while still considering utility and comfort. Especially in non-risky scenarios, ROPT is even able to fully concentrate on beneficial behaviors.

Risk and Utility Modifications

As alternative to the basic risk-based planner from Section 3.3, we also change the risk and utility terms in the costs. These form a new damage model that includes kinetic energy and an additional term for reaching a desired velocity.

Since the conveyed kinetic energy in a casualty is proportional to the operating masses m_i and velocity vectors $\mathbf{v}_i$, we use for collision and curve damage the formulas

$$D_{\text{coll}}(s; t, \Delta s) = D_0 + \frac{m_1 m_j}{2(m_1 + m_j)}\|\mathbf{v}_2 - \mathbf{v}_1\|^2$$

$$D_{\text{curv}}(s; t, \Delta s) = D_0 + \frac{1}{2}m_1\|\mathbf{v}_1\|^2$$

[4] In contrast, the action of taking the foot off the brake or gas pedal has immediate effect on the car.

with an offset D_0. Anytime a crash is not possible conditional to kinodynamics of the cars, D_{coll} is set to 0. By taking the prominent survival probability S that the ego entity will not be engaged in an event during $[t, t+s]$ via

$$S(s; t, \mathbf{z}_{t:t+s}) = \exp\{-\int_o^s \tau^{-1}(\mathbf{z}_{t+s'})\, ds'\},$$

and by calculating the corresponding event rates τ_{coll}^{-1} and τ_{curv}^{-1} to the damages, we can obtain again $R(t)$ as

$$R(t) = \int_0^\infty (\sum_j \tau_{\text{coll},j}^{-1} D_{\text{coll},j} + \tau_{\text{curv}}^{-1} D_{\text{curv}}) S\, ds.$$

Here, the collision risks are summed up for each other car j in the traffic situation. As a side note, a straightforward numerical calculation of the integral is sufficient with small Δs, e.g., we utilize 0.05 sec.

The benefits are composed of utility and comfort for the ego car. For the utility, we not only take the overall needed travel time affected by the ego velocity v_1, but also define the deviations to a desired velocity v_d as a contribution to utility. The components are weighted with driver-specific constants b^t and b^d to retrieve

$$U(t) = \int_0^\infty (b^t |v_1| + b^d |v_1 - v_d|) S\, ds.$$

Comfort returns are granted, if the ego behavior does not change abruptly (i.e., ego acceleration $a_1 \approx 0$) and the approach to planned a_1 is slow (i.e., ego jerk $j_1 \approx 0$). We can weight comfort costs with the constants b^c and b^j, which are given in Equation (3.6).

3.5.2 Prediction Under Priority

We incorporated ROPT in a planning framework that allows to predict and optimize velocities for cars in a general driving situation. But what happens exactly in interaction-intensive situations, where some cars are required to be considered as having priority over other cars?

In an accident, at least one car is responsible by law for damage costs [109]. For example, during a following scenario, the rear vehicle is held liable, if it fails to keep safe distances to a leading vehicle. To the contrary, for cars involved in a head-on collision, both drivers are considered to be at

fault. Around intersections with traversing paths, priorities allow the responsibility to be shifted to the driver who has to yield. A requirement for such longitudinal and lateral priority-dominated situations is that the considered superior entity (either leading or prioritized car) does not brake or speed up unreasonably. Otherwise in law, the share of blame and costs is divided again among the involved parties.

To implement asymmetry in interactions, situations were treated in related research as discrete awareness or non-awareness entity combinations [21]. Here, awareness is the behavior adaptation of one vehicle dependent on another vehicle. By iterating over each and superposing the inherent risks, an optimal trajectory was constructed. However, for ROPT, a more **computationally efficient way** is to only focus on the likely situations based on priorities. The other unlikely situations could be included in this approach as well but are neglected for this purpose.

In ROPT, we thus a) categorize the path relation between vehicles and match them to legal rights-of-way (e.g., front-before-back, right-before-left) and b) appropriately modify behavior-relevant prediction models of other cars (i.e., altering the influence on own risk and calculating different trajectories). In this way, we will consider only the likely other car behavior. The prediction and planning tasks are addressed hereafter.

Order Assignment

A generic driving scene of two traffic participants, defined as TP1 and TP2 with i = 1,2, is illustrated in Figure 3.11. In the context of geometrical interaction, we trail corridors, having a width c_w, from their current longitudinal position l_1 and l_2 until the trajectory end. The zone of interaction is subsequently given where both corridors interfere. We project start and end points to each path and get separate boundaries $I_{s,1}$, $I_{e,1}$ for TP1 and $I_{s,2}$ and $I_{e,2}$ for TP2. The points outline the area of interaction.

In the longitudinal driving case, in other words car-following, let one or both TPs be in the interaction zone at the moment in time t. Comparing the positions l_i allows now to assign TP2 being in front or in the back to TP1. In total, we can write

$$l_1 \in [I_{s,1}, I_{e,1}] \wedge l_2 \in [I_{s,2}, I_{e,2}] \rightarrow$$

$$\text{front: } l_1 < l_2, \text{ back: } l_1 > l_2.$$

For the lateral case, i.e., intersection crossing or lane changes, trajectories meet in the future. When we look at the difference angle $\Delta\gamma$ of the inter-

action start $I_{s,1}$ and $I_{s,2}$, TP2 is to the right or left depending on its value in compliance with the predictive logics

$$\angle I_{s,1} I_{s,2} = \gamma_{s,1} - \gamma_{s,2} = \Delta\gamma_s,$$
$$\text{right: } \Delta\gamma_s \in (0, \pi), \text{ left: } \Delta\gamma_s \in (\pi, 2\pi).$$

Possible interaction types for TP1, if the car crosses an X-intersection on a straight path, are also summarized in Figure 3.11. Besides TP1 and TP2 driving on the same path, the trajectory of TP2 can intersect, be curved before or after and merge with trajectory of TP1. For *front-before-back*, TP2 is defined as superior in front and as inferior in back relations. Analogously, *right-before-left* determines TP2 as superior for right and inferior for left contexts. In other countries with left-before-right, the order assignment is switched, and the same formulas may be applied.

Trajectories for Other Cars

On crowded public roads, we concentrate on the main cars around which have right-of-way. Since these represent the most critical cars. The remaining cars are only considered, if they also come critically close.

In this sense, we also include and model this awareness accounting process. ROPT weights the collision risk $\tau_{\text{coll},j}^{-1}$ of inferior obstacles with a monotonically decreasing function. Regarding longitudinal interactions,

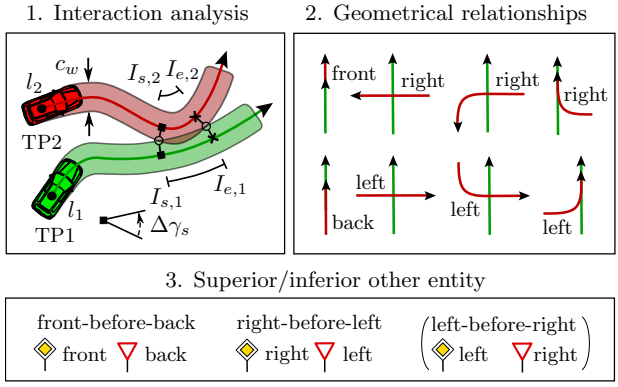

Figure 3.11: Individual steps for regulatory risk estimation on the basis of spatial path corridors. From the interaction area, we can identify what geometrical relationship is present (front, back, right and left).

a sigmoid function $\alpha_{\text{lon}}(s)$ is used with the slope k_{lon} and midpoint s_{lon}, which leads to the two terms

$$\alpha_{\text{lon}}(s) = 1 - \frac{1}{1 + \exp\{k_{\text{lon}}(s - s_{\text{lon}})\}},$$

$$\tau^{*-1}_{\text{coll},j}(\mathbf{z}_{t:t+s}) = \alpha_{\text{lon},j}(s)\tau^{-1}_{\text{coll},j}(\mathbf{z}_{t:t+s}). \tag{3.7}$$

In the end, we obtain a modified collision risk $\tau^{*-1}_{\text{coll},j}$ for single vehicles. The equations for $\alpha_{\text{lat}}(s)$ are the same, whereby the chances that the other car perceives us is lower in intersections and parameters k_{lat} and s_{lat} is set to higher values. The discounting function is plotted for longitudinal and lateral interactions in Figure 3.12 (left-hand side).

Another extension which is applied for the prediction model of other cars, are delayed acceleration patterns. Without priority knowledge, vehicles are extrapolated with constant velocity from Section 3.5.1. In addition to decreasing the awareness, ROPT predicts delayed accelerations in the lateral situation. If the other car is superior, the ego planner assumes first constant velocity for a time period of s_0, an acceleration phase (s_a, a_a) and lastly steady velocity up to s_h. The case differentiation follows as

$$v(s) = \begin{cases} v_0, & \text{for } s \in [0, s_0), \\ v_0 + a_a(s - s_0), & \text{for } s \in [s_0, s_a], \\ v_0 + a_a(s_a - s_0), & \text{for } s \in (s_a, s_h]. \end{cases}$$

Hereby, the values of a_a depend on the active velocity v_0 (i.e., we apply $a_a = 0$ for $v_0 = v_{\max}$ and linear growth to $a_a = a_{\max}$ when $v_0 = 0$). This is

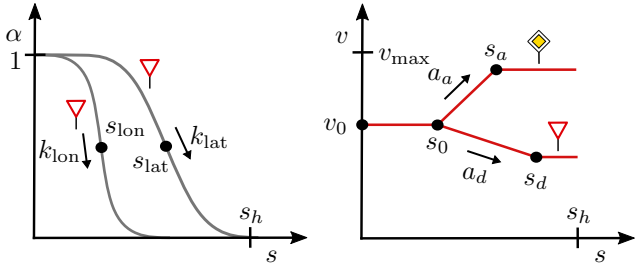

Figure 3.12: Depending on the priority, we change the risk prediction. Left: Change in collision risk another car imposes on ego car with sigmoidal weighting function. Right: Acceleration and deceleration assumptions for other entity.

based on the intuitive notion and empirical findings that cars at intersections tend to accelerate stronger starting from lower velocities at standstill. By comparison, if the other car has superior priority, ROPT uses a longer deceleration phase (s_b, a_b), i.e., braking with constant a_b.

In a last step, we additionally clip the velocities $v_j(s)$ to be higher than 0 and lower than the maximal curve velocity $v_{\text{curv},j}(s)$ and allowed limit $v_{\max}$ with

$$v_j(s) = \max(v_j(s), 0),$$
$$v_j(s) = \min(v_j(s), v_{\text{curv},j}(s), v_{\max}).$$

The altered velocity patterns from Figure 3.12 (right-hand side) lead to better predictions of other vehicles, given that they behave according to the traffic rules. It renders the predictions more realistic. Because of the delay s_0 in combination with the risk discounting parameters s_{lon} and s_{lat}, ROPT is in short-times even robust against moderately wrong assumptions. Each should be set that no crash happens for any acceleration or deceleration maneuver.

More detailed predictions can be achieved by considering environment conditions (just accelerating in interaction zone and coming to halt at stop line), participant types (e.g., motorbike or truck) and occurring situation class (highway versus inner city). These are potential further inputs for the planner ROPT.

3.6 Experiments

In the simulations, it is intended to show that the Risk Optimization Method (ROPT) of this dissertation can handle a wide array of vehicle-to-vehicle interactions, which typically occur at intersection crossings. The planned solution should be especially compliant with priority rules.

For this reason, we begin in Section 3.6.1 by analyzing one vehicle pair during dynamic followings before or after intersections as well as during passing behaviors within interaction areas. The effect of the altered prediction models from ROPT will be shown quantitatively. Secondly, in Section 3.6.2, we randomize the possible paths for the two cars with test statistics to establish the robustness of ROPT, in terms of risk and comfort.

As it turns out, the optimization compensates non-priority-compliant other behavior with adequately elevated jerks, while still maintaining adequate safety. To the knowledge of the author, this was not done before in related research works.

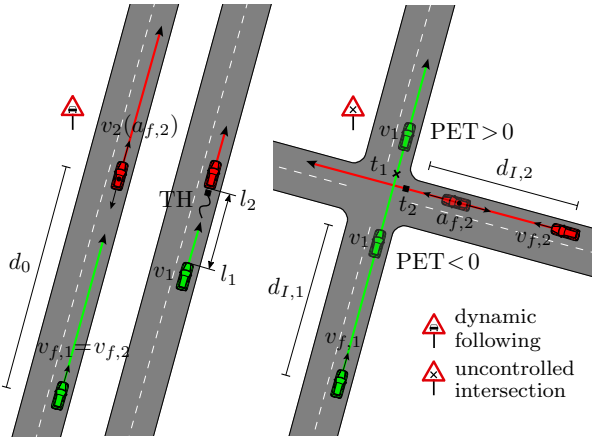

Figure 3.13: Left: Initial plus final conditions in a following scenario under front-before-back priority (case ego following). Right: Two possible scene evolutions for an intersection with right-before-left (case other priority).

3.6.1 Results

Both regarded basic scenarios are pictured in Figure 3.13: longitudinally driving behind a leading Traffic Participant (TP) in front and an uncontrolled intersection having a second TP to the right. Uncontrolled intersections are intersections, where priorities manage the traffic. Furthermore, we reproduce the cases that TP2 is in the back or approaching from the left. In each case, we vary for TP2 the initial velocity $v_{f,2}$ within 0 and 15 m/sec. After a time of 1 sec, a deceleration/acceleration $a_{f,2}$ is applied in the range from -3 to 3 m/sec^2 for the duration of 3 sec.

The challenge for ROPT is then to adapt TP1 (ego car, green) to the fixed actions of TP2 (other car, red), while considering the regulations front-before-back and right-before-left. Concerning the longitudinal environment, ROPT starts at a distance $d_0 = 50$ m to TP2 and with equal speed $v_{f,1} = v_{f,2}$. A soft road limit of $v_{\max} = 20$ m/sec is applied. For the intersection instance, beginning offsets until the path corridors overlap are chosen as $d_{I,1} = d_{I,2} = 40$ m and the velocity parameters of ROPT, i.e., $v_{f,1}$ and desired velocity $v_{d,1}$, are set to 10 m/sec.

In the evaluation, we particularly look at the indicators Time Headway (TH) [117] and Post-Encroachment Time (PET) [2], which depend on the kinematics of the vehicles. Both are risk measures that show intuitively

the time to a collision hazard. They can be formulated as

$$\mathrm{TH} = \frac{-\Delta l}{v_1} \text{ with } \Delta l = l_1 - l_2, \tag{3.8}$$
$$\mathrm{PET} = -\Delta t \text{ with } \Delta t = t_1 - t_2.$$

The events t_1 and t_2 indicate in PET when the ego entity leaves and when the obstacle enters the interaction zone, respectively. On that account, ROPT can either pass in front with PET > 0 or behind with PET < 0. Critical values are close to 0. However, we can see PET in the evaluations rather as a utility indicator, than a risk indicator. For TH, we extract the stable value $\mathrm{TH}_{\mathrm{stable}}$ once a constant longitudinal distance Δl is maintained between the cars.[5]

To complete the utility assessment of ROPT, we eventually capture the lower boundary $v_{\mathrm{low},1}$ and upper boundary $v_{\mathrm{up},1}$ from the executed velocity course v_1 of the planner.

Dynamic Following

An agent controlled with symmetric risk calculations to the front and back would react sensitively to all surrounding vehicles, even to following cars, e.g., in the case of tailgating. Due to the risk discounting with Equation (3.7), it is harder for a back car to negatively influence ROPT. The car is still not entirely ignored, since non-reaction results in legal blame.

In detail, the contour plots within Figure 3.14 sort the measured $\mathrm{TH}_{\mathrm{stable}}$ and extrema of v_1 into colored bins, for each other trajectory $(v_{f,2},\, a_{f,2})$.[6] In the case that the ego vehicle is following (compare the top row), another vehicle in front with priority yields proactive ego behaviors. When the other vehicle brakes down, ROPT uses unaltered collision risks and converges to moderate $\mathrm{TH}_{\mathrm{stable}} \approx 2\,\mathrm{sec}$ for large $v_{f,2}$ and $|a_{f,2}|$. If the other trajectory is a stopping trajectory, the minimum end speed $v_{\mathrm{low},1}$ becomes $0\,\mathrm{m/sec}$ in ROPT, and thus $\mathrm{TH}_{\mathrm{stable}}$ exceeds $5\,\mathrm{sec}$. Unlimited $\mathrm{TH}_{\mathrm{stable}}$ are moreover also carried on, when TP2 is moving away with $a_{f,2} > 0$. ROPT is therefore able to retain the varying beginning velocity, more specifically, $v_{\mathrm{low},1} \in [0\,\mathrm{m/sec}, 15\,\mathrm{m/sec}]$.

In contrast, the ego vehicle is leading and has priority in the bottom row of Figure 3.14. As it turns out, the planner allows lower $\mathrm{TH}_{\mathrm{stable}}$

[5] Equation (3.8) counts, if the ego car follows another vehicle. For the inverted case, the indices in TH are swapped.

[6] The grid step size amounts to $\Delta_{x,y} = 0.5$ with linear interpolations.

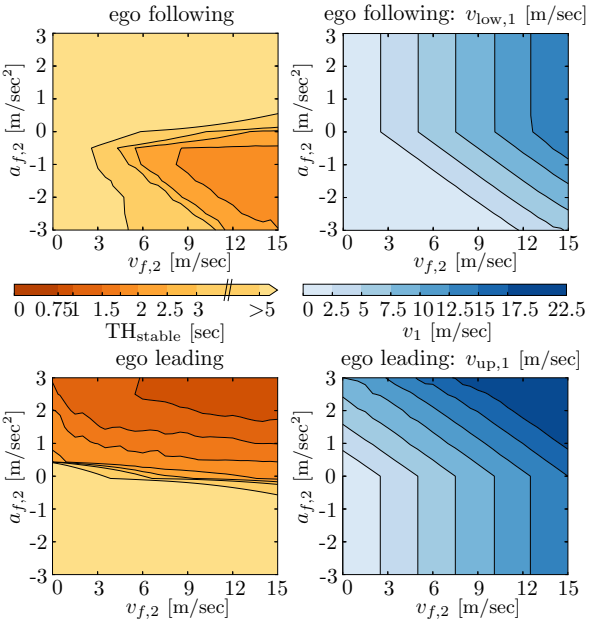

Figure 3.14: Indicators of ROPT behavior (minimal and maximal velocity in blue) and its interplay with another car (i.e., headway in red) for range of fixed other actions (varying initial speed and acceleration). The priority-dependent awareness horizon leads to lower headways for back vehicles.

until 1 sec to the back. If $a_{f,2}$ is positive, higher $v_{f,2}$ decrease the value of $\text{TH}_{\text{stable}}$. At the same time, the final maximum velocity $v_{\text{up},1}$ matches the accelerating follower with $v_{f,2} + 3\,\text{sec} \cdot a_{f,2} < 22.5\,\text{m/sec}$. For negative $a_{f,2}$, ROPT is not influenced by the decelerating obstacle (i.e., $\text{TH}_{\text{stable}}$ from 3 sec upwards) and delivers steady velocity (i.e., $v_{\text{up},1} = v_{f,1}$ applies).

Intersection Passing

Simple go/no-go decisions cannot ensure optimal driving for lateral priorities. Another car on the right with priority might be far away or decelerating so that the ego utility is neglected. More importantly, cars on the left without priority, which do not respect the right-of-way, may create arbitrary risk or discomfort peaks. Via the delayed acceleration patterns from Section 3.5.2, ROPT is able to continuously weigh the benefits with risks for passing a rule-based intersection first or second.

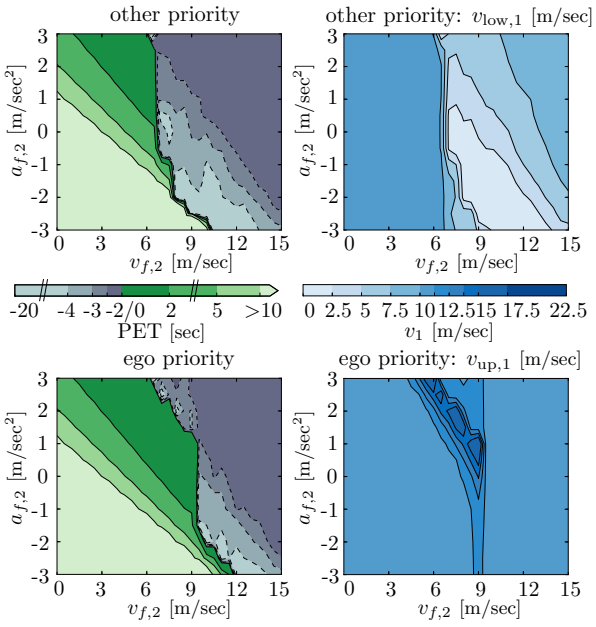

Figure 3.15: Results for ROPT with delayed acceleration patterns, in which we analyze the encroachment time (green - positive values, grey - negative values) and velocities. Top: Another car coming from right. Bottom: Car coming from left. On an intersection, ROPT accelerates more often when having priority.

Figure 3.15 visualizes the isolines of PET, $v_{\text{low},1}$ or $v_{\text{up},1}$ for the same variations of the other actions $v_{f,2}$ and $a_{f,2}$. While there is more area of PET < -2 sec (i.e., ROPT driving second) when another car has priority, the condition PET > 0 dominates (i.e., ROPT driving first) when the ego car has priority. ROPT correctly accelerates more often and tries to drive first when having priority.

The reason for this can be well observed in the ego velocity v_1. For instance, in the top row, TP2 has priority and ROPT brakes frequently having $v_{\text{low},1}$ under 2.5 m/sec. Additionally, accelerating back to the desired $v_{d,1}$ takes more time with PET $\in [-3\,\text{sec}, -20\,\text{sec}]$. The velocity growth prediction of TP2 leads to cautious ego behaviors. Still, overtaking is established in small other speeds $v_{f,2}$. For $v_{f,2} \to 0$, the other car does not interfere with the ego car and PET > 10 sec holds.

Vice versa, the bottom plots show that a prioritized ROPT is inclined

to accelerate with $v_{\mathrm{up},1}$ as far as 17.5 m/sec because it assumes the obstacle will stop. If the encountered vehicle disobeys (e.g., $a_{f,2} \approx 3\,\mathrm{m/sec}^2$ and $v_{f,2} \approx 7\,\mathrm{m/sec}$), ROPT will at some point halt and give way. The situations are still safe but create the strongest behavior change.

3.6.2 Statistical Evaluations

For the large-scale experiment, the unsignaled intersection is extended below with statistical conditions for the simulation. Altogether, we randomize path geometries, agents' starting states and the priority compliance of the other participant. This enables us to further analyze hazards and jerk caused or rather avoided by ROPT from car-to-car passings.

Simulation Setup

While we reduce the driving limit to fixed $v_{\max} = 8.5\,\mathrm{m/sec}$, each individual run has different angles between the four roads and random lane widths. Moreover, the start and destination roads for the ego and other vehicle are stochastic.[7] Both cars subsequently start with sampled velocities $v_{f,1}$ and $v_{f,2}$ from 3 m/sec until 8.5 m/sec having a set distance $d_{I,1} = d_{I,2} = 45\,\mathrm{m}$ to the intersection edge. Here, the desired cruising velocity $v_{d,1}$ for ROPT is always equal to $v_{\max}$ and $v_{d,2}$ of the other participant is also randomized including higher speeds $\leq 10\,\mathrm{m/sec}$. We finally vary the compliance of the second car. In 50 % of the experiments, TP2 is attentive, which means respecting the priority. However, in the other 50 %, TP2 is inattentive and ignores TP1's priority as long as the center-to-center distance d_2 is above 10 m. This results in particularly challenging situations, if TP1 has priority and assumes the obstacle will yield.

Opposed to Section 3.6.1, the simulation applies now a full multi-agent planning. As above, ROPT steers the ego car. In addition, the other car is controlled dynamically: it possesses the same cost function to evaluate trajectories but exploits a simpler mechanism for creating candidate trajectories (i.e., no full optimization from Section 3.5.1). In each time step, the other vehicle directly constructs differing acceleration/deceleration profiles and selects the best one among them.

After each run, the simulated driving scene is evaluated. For computing risk levels, we introduce a measure termed two-dimensional headway $\mathrm{TH}_{2\mathrm{D}}$ that expands TH to account for lateral distances.

[7] A prerequisite is that the start roads are distinct, and their paths intersect. The situation will therefore correspond to the basic lateral types depicted in Figure 3.11.

With the help of constant velocity extrapolation, $\mathrm{TH}_{2\mathrm{D}}$ essentially indicates the time when vehicle pairs will occupy or have occupied the same space. In detail, $\mathrm{TH}_{2\mathrm{D}}$ is obtained by first taking the bounding box for each agent consisting of four corners at the current step t. In the following calculation, we enlarge this box with length of $v_t \cdot \frac{T_{\mathrm{ext}}}{2}$ in both directions along the agent's path (here, v_t represents the velocity of the participant at t and T_{ext} is the extrapolation interval). This means that the resulting shape can bent around corners and is not convex. We lastly define $\mathrm{TH}_{2\mathrm{D}}$ to be the minimum T_{ext} where the shapes of the two cars overlap.

Alongside the $\mathrm{TH}_{2\mathrm{D}}$, the maximal value of the ego jerk course $r_{\max,1}$ is recorded. As a reference, most passengers rate a jerk of up to $3\,\mathrm{m/sec}^3$ as acceptable [87], and in emergency trajectories, jerks above $6\,\mathrm{m/sec}^3$ are common [4]. To neglect comfort reduction because of high frequency motion, we filter beforehand peaks in r_1 with rolling means and a window factor of $W = 0.5\,\mathrm{sec}$.[8]

Robustness Discussion

More than 2000 simulations were executed in total, involving the described settings. Figure 3.16 outlines the measured statistics for $\mathrm{TH}_{2\mathrm{D}}$ and $r_{\max,1}$. We initially focus on the ex-post risks.

The left side of Figure 3.16 renders cumulative distributions A_{runs} for $\mathrm{TH}_{2\mathrm{D}}$. Regardless of TP2 following priority (top) or violating (bottom) priority, $\mathrm{TH}_{2\mathrm{D}}$ is larger than 1 sec in at least 85 % of runs and $> 0.5\,\mathrm{sec}$ in all runs. In the cases when $0.5\,\mathrm{sec} < \mathrm{TH}_{2\mathrm{D}} \leq 1\,\mathrm{sec}$, ROPT rightfully exerts its priority and the other vehicle crosses right behind. Decreasing values of $\mathrm{TH}_{2\mathrm{D}}$ are a consequence of the parametrization for TP2. It has a higher set constant escape rate τ_0, which in effect leads to shorter prediction horizons and more aggressive planning. However, the main observation is that the trajectories of ROPT are always safe. ROPT must have compensated the incompliance of the other car, therefore we now look more closely into the behavior of ROPT.

Probability distributions Q_{runs} of the maximum jerk $r_{\max,1}$, encountered by ROPT, are given in Figure 3.16 on the right.

If the other vehicle obeys right-of-way, $r_{\max,1}$ is below $2\,\mathrm{m/sec}^3$ in almost any situation (approximately 99 %), i.e., the ride feels comfortable. ROPT is robust against the intersection geometry or differences in taken starting and desired speeds from TP2. Rising $r_{\max,1}$ solely appear when TP2 has

[8] Note that the moving average filter is not used within ROPT and only reduces outliers from $r_{\max,1}$ for the evaluation.

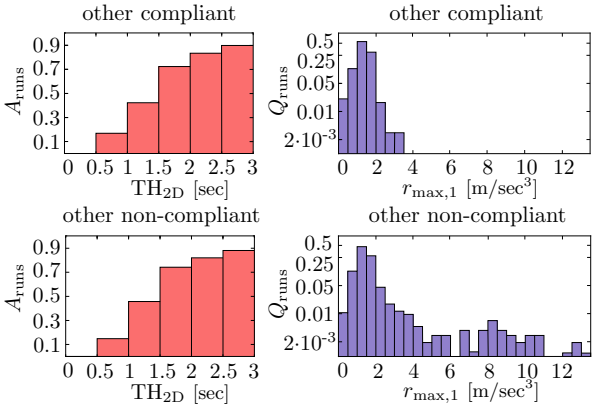

Figure 3.16: Robustness of ROPT in diverse stochastic conditions during intersection crossings, e.g., priority violation of another car. Left: Cumulative histogram for two-dimensional headway. Right: Probability histogram for maximum ego jerk (note the log scale).

a counteracting behavior with non-compliance to right-of-way. In such instances, ROPT must compensate others' negligence by accepting larger $r_{\text{max},1}$. Usually, it reacts by either clearing the intersection earlier, by accelerating away, or by making a full brake to let TP2 pass in front. The latter can produce in $< 1\,\%$ highest $r_{\text{max},1}$ with up to $13\,\text{m/sec}^3$. Nevertheless, even for other inattentive participants, the jerk of ROPT is low to moderate for $90\,\%$ of the situations and ROPT is able to smoothly adjust its behavior.

3.7 Conclusion

For this chapter, we examined planning of driving behaviors in dynamic environments. The main contribution is a novel risk-based planner that alternates between trajectory generation and evaluation in an optimization. Via the sampling of velocity profiles for the ego car, the integral risk, utility and comfort for the arising situation are evaluated. Hereby, risks are optimized based on the driving risk model from Chapter 2.

By comparison, the Intelligent Driver Model (IDM) was also extended for intersection scenarios to obtain the IIDM. In simulations of right turning at T-intersections, the risk-based approach ensures no collisions and

distributes risks evenly. IIDM has qualitatively similar results, but can yield crash cases, because of an if-then type of behavior.

In the second part of this chapter, a planning framework was presented to predictively plan dynamic velocity curves under rights-of-way. After geometrically determining the trajectory relationship between vehicle pairs, the corresponding collision risk may be discounted if other cars are longitudinally inferior (i.e., for cars to the back). In lateral traffic situations, other participants are simultaneously assumed to prototypically decelerate when inferior (e.g., participants on the left) or accelerate when superior (e.g., participants on the right).

Analytical experiments finally demonstrated that the method is able to comply with priority rules. With a path randomization, it was also shown that wide ranges of car-to-car interactions can be handled. It is argued that this is a major advantage to other planning methods.

Overall, this chapter addressed the research questions for planning with the listed contributions:

- **Generic Plans**: Novel velocity planner that optimizes risks, utility and comfort to achieve generic and tactical motions [94].
- **Driving Strategy**: Extensive simulations of merging at intersections, in which the planner did not produce any collisions and was superior to related planners (even predictive planners).
- **Interaction-awareness**: Novel planning framework, which changes the risk prediction of other cars for interaction-aware motions under given traffic rules [93].
- **Driving Strategy**: Analytical and statistical experiments on car-following and intersection crossing that show robust functionality.

Please refer to Appendix 6.3 and 6.4 of this dissertation for details. Specifically, attributes of the risk-based planner are outlined with Appendix 6.3, by showing graphs of its cost function. Distinct driving styles can be rendered, if the weights in the costs are changed. In addition, Appendix 6.4 compares the speed and performance of optimizers for planning.

In the subsequent Chapter 4, the risk models and planners of this dissertation will be leveraged on vehicle prototypes. The target is to show the applicability of the approaches for real-world driving.

4 Warning in Real-Time Applications

In this chapter, we will look closely at the acting or rather driver warning step in self-driving cars, the last part of a sense-plan-act system.

One of the major causes of accidents when driving in public is human error. According to the World Health Organization (WHO) [137], drivers encounter difficult situations due to inexperience, recklessness and fatigue, which leads to, e.g., speeding or maintaining inappropriate distances from other cars. Furthermore, they often have a partially limited field of view (i.e., caused by roadside objects) and fail to see driving points of interest. Reaction times to an object that appears can thus be critical. Lastly, anticipating the behavior of other drivers is sometimes complex for humans. The deviation of real situation and resulting ego action may generate unavoidable collisions.

We tackle the listed driving problems by proactively warning the driver using risk models. For this purpose, the methods of this dissertation are applied to test vehicles with real-time sensors. Additionally, a Human-Machine Interface (HMI) will be presented. Visualizing the safe behaviors from the system could potentially ease the prominent trust issue of self-driving cars. Our aim is therefore to provide warnings and reasoning for the warnings so that they are more likely to be followed.

Altogether, this chapter revolves around the following research questions for risk warning:

1. **Situation Data**: How should the analyzed situation data be structured efficiently and transparently? On the same note, how does the HMI then visualize impending risks to the driver?

2. **Prediction**: Which further steps must be implemented in a working system? How can map geometries improve the prediction?

3. **Risk Applications**: What are suitable real-time applications for the risk models? Apart from signals from sensors with high accuracy, can we use the methods with low-cost sensors?

The basis is given by the proposed risks models and planners from Chapter 2 and Chapter 3, respectively. As we will see, stored maps are crucial for bringing approaches from simulation into the real world. Here, an aim is to provide support several seconds before a critical event, which is one of the main differences as compared to other HMI solutions.

To introduce the topic, related work in risk warning is elaborated on the next section. Two possible applications are shown. We start with a risk-based driving navigation, which warns against high-level risks within a HMI environment. The system architecture, adaptations of the risk models and field tests are outlined. For the final application, we leverage the novel risk model of this dissertation and provide trajectory-level motion recommendations. This allows us to obtain a sophisticated lane change support. Risk graphs give warnings about dynamic collisions and are visualized in a real test vehicle.

4.1 Related Work

Current warning systems consist of the following processing steps: a) retrieval of risk indicators, b) comparison of actual versus planned behavior and c) on-demand activation of the HMI. Such categorization can be found in, e.g., the work of [63]. The sensor signals and saved map data are often utilized for obtaining risks for a driving vehicle, whereby the HMI is based on visual, audio or tactile modalities.

In particular, in this context, approaches for trajectory predictions are highlighted because they represent the basis for the functionality of future risk warning. In risk extraction, the prediction of a trajectory with high accuracy is beneficial.

As we already know, model-based methods in trajectory prediction use kinematics, describing a vehicle's motion directly with movement variables (i.e. position, velocity, acceleration and turn rate). Trajectories for future time steps are obtained by extrapolating the car states beyond the latest time, while assuming constant inputs. The simplest kinematic models are Constant Velocity (CV) or Constant Turn Rate and Velocity (CTRV) [106]. Enhanced prediction methods will be described below.

Trajectory Prediction

One improvement addresses the uncertainty that is considered for the predicted cars. An obvious way of doing this is to, e.g., overlay some Gaussian

distribution within the kinematic model of a Kalman Filter [52]. Normally, the spatial distribution of the predicted position is unknown. For that reason, Monte Carlo simulations [65] can provide tools for approximating this distribution. The idea is to randomly sample input values for the kinematic model to generate a set of trajectories.

For map-based methods, path geometries are leveraged in the prediction. In the automotive industry, support systems estimate paths from sensor measurements inside the vehicle. In contrast, self-driving cars heavily rely on pre-recorded maps (for instance, compare [35]). Prior knowledge, i.e., map data, is significant in helping to predict traffic situations for longer timespans and more robustly into the future. For this reason, in [84], self-driving cars are localized on a global map to retrieve the set of possible outgoing paths from lane geometries. Longitudinal behavior along paths can then be computed accordingly with the kinematic models.

Finally, with machine learning, we may train algorithms to predict trajectories from past measured states or estimate the maneuver intention of a driver (i.e., turn left, right or drive straight) using features, such as the distances to an intersection. On this note, the authors of [24] reproduce, for instance, trajectories in lane changes with Graph Neural Networks (GNN). Due to the vast behavioral options for vehicles, trajectory prediction with fail safe backups are also shifting into focus and are applied on top of the normal approaches [83].

Behavior-Relevant Risks

Now, we turn to the steps of current warning systems. A detailed outline of risk extraction and behavior inference is given, with a focus on real-time applicability.

In research, risk analysis becomes increasingly omnipresent. As we could see in Chapter 2, risk methods can be sorted into discrete time indicators, probabilistic risks and learned risks. For intuitiveness, time measures (see, e.g., [38] for Time-To-Brake and variants) are more advantageous. However, as already stated, neglecting uncertainty may not reflect real driving. In probability considerations [99], tradeoffs must be found between accurate and computationally inexpensive models. Lastly, learning the risks [129] is accurate for specific complex situations but results in reduced performance for situations that are not included in the training data.

With online data, traffic risk extraction can be achieved with the help of outside sensors, in particular, camera, lidar and radar. For example, the authors of [73] identify construction signs and vehicles in a front-mounted

camera. For headway control or some crash warnings, [39] also compute the Time-To-Collision (TTC). Fusing the camera with radar then allows noise to be reduced in the object velocity estimation. Fusion of different sensors is necessary for enabling robust automated vehicles.

In the two mentioned approaches, sensors operate in relative coordinate systems and all inputs are received on-the-fly. In comparison to this, ego sensors (e.g., a GNSS plus inertial measurement unit) can also be aligned with a global reference for leveraging map data. A real-time setup is shown in [139]. Their HMI displays pop-up symbols for speed limits or occluded crosswalks. Similarly, [90] employ road-level maps to calculate curvatures of the street for saving up fuel via an engine management.

For behavior recommendation, we can eventually refer back to the traffic models from Chapter 3. These models can easily be utilized for diverse applications due to their low computational costs. The Intelligent Driver Model (IDM) [118] allows the following of preceding cars using kinematic equations. With the lateral extension of MOBIL [60], lane changes are considered as well. The output of safe target velocities could be used, e.g., for simple support systems. Nevertheless, a major problem is that these models are not holistic and do not sufficiently regard uncertainties.

HMI for Warning

In the final warning stage, behavior-relevant risk information is communicated to the user in order to improve driving safety. While most warning systems apply various visual monitors, audible and tactile actuators are still a topic of research and not as widespread.

In this context, the work of [3], for instance, enhances head-up displays by sound to shorten the drivers' response time. On top of that, [22] also investigate light strips for spatially distant information, and alarms for immanent critical entities. In regard to navigation devices, tactile impulses on a car seat [14] can be applied to turning maneuvers, or mobile belts [64] vibrate according to the relative position of obstacles. During the process, the priority of elements is assigned by varying intensities of pressure.

Audible and tactile communication channels are intuitive for the driver. The driver requires more time to process the visual modality. However, it employs animations and color coding for conveying urgency and more importantly, in handovers between a machine and a human, visual context is vital for understanding the situation. This is due to the fact that humans mainly rely on visual perception when driving [85]. Thus, more complex situation information can be conveyed visually.

Regarding visual HMIs, symbolic displays in the cockpit (warnings with text or icons) have been thoroughly investigated in user studies, e.g., compare [85]. In their conclusion, the workload and effort to read traffic information can increase on a 3D display compared to a 2D display. Bird's-eye views are also rated as clearer and more comfortable than first person views. In order to provide a 360 ° analysis, [133] recently introduced safety rings which bend to indicate the position of dangerous vehicles. Similarly, lateral lane change and longitudinal speed recommendation can be visually conveyed in an effective way [44].

To summarize, there is promising applied research on the topics of trajectory prediction, risk estimation, motion planning and human-machine interaction. For this dissertation, map-based risk models under driving uncertainties are leveraged. Using the sampling of trajectories, a safe, future motion for the ego car is retrieved. Hereby, the warning is done on a visual HMI. This is unique compared to related work, which are not based on a separate prediction stage. The benefits of risk warnings will come forth in real prototype applications of the next sections.

4.2 Application of Risk Navigation

The following section deploys risks on the problem of intelligent automotive navigation. Concretely, we give out warnings (if necessary) along the driving route with a map database. Technical inputs and outputs of such a safety technology will be outlined.

A traffic situation, involves, additionally to traffic participants, the map structure and road markings. Altogether, they form networks of geometrical data. Thereon, multiple vehicles are then moving forward, while trying to keep the risk of accidents low. In urban environments, such as an intersection, the combination and interaction of traffic elements is diverse. Drivers must focus on elements that affect their behavior.

To support this analysis, the Risk Navigation System (RNS) is proposed. RNS is based on a local dynamic map that constitutes a central hub for saved *road data* and updates from *sensor measurements*. We reduce the situation complexity by extracting explicit paths within a *virtual horizon*, which also can provide further driving information. Consequently, variants of *Time-To-X indicators* from Chapters 2 and 3 are applied. These allow the detection of critical situations and recommendations of safe behavior. The end system provides navigation information to the user from a starting position to an urban goal. Once the users are not behaving in a risk-appro-

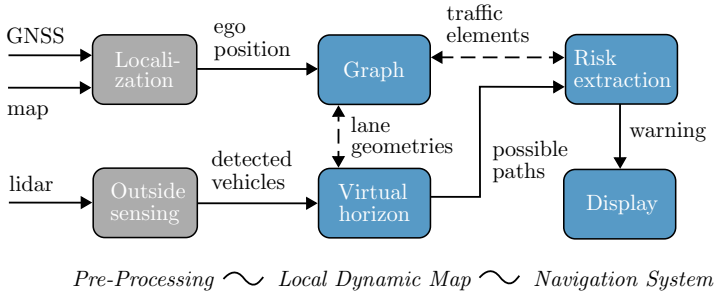

Figure 4.1: Concept of the Risk Navigation System (RNS), designed for driver support at intersections. We can use the local dynamic map for the retrieval of possible paths. With high-level risks, the driver is finally warned.

priate way, warnings are displayed which convey both the positional and temporal features of dynamic driving risks.

In Figure 4.1, the pipeline of the warning system is pictured. The main focus lies on the boxes indicated in blue (i.e., the local dynamic map, risk extraction and display). With real-time recordings, combining GNSS[1] and lidar sensors, RNS is shown to be effective in analyzing and visualizing risks from collisions, curves and traffic rules. The prediction ability of the driver is potentially improved. An ambition of RNS is to express driving situations with flexible environmental models because *generalized driver support* needs clear and efficient data management functions.

Therefore, we will analyze, in detail, how the data is stored in the local dynamic map. The corresponding virtual horizon for data retrieval is an important tool, which allows risk models to be applied to a real-world implementation.

Relational Local Dynamic Map

As helpful meta-information lies in the connectivity of environment entities, we employ a Relational Local Dynamic Map (R-LDM) that consists of so-called nodes, attributes and relations [30]. Effective queries of relevant data are ensured with a graph instead of a table database.

The R-LDM graph stores the traffic information in four layers based on their temporal dynamicity. Maps are represented as nodes with additional shape, orientation or type data, saved as attributes. At the lowest layer,

[1] Universally accepted term for Global Navigation Satellite System (e.g., GPS, Galileo or GLONASS).

the street geometry presents one of the most prominent data types. **Static data** is stored in varying levels of granularity, ranging from detailed lanes via half roads to the whole road. In this context, half roads are the sum of all lanes pointing in the same direction. Furthermore, the road subgraph consists of alternating segments and junctions, while all junctions at one location form an intersection.

The second layer comprises **quasi-static data** using a scale of several days. This implies information of traffic lights, roadside infrastructure, traffic signs and construction sites. On the subsequent third layer, **transient data** is stored and changes on hours scale or shorter (i.e., states of traffic lights, car density or temporary road conditions). Finally, the fourth layer of the R-LDM contains highly **dynamic data** about states of traffic participants, including vehicles and pedestrians. Sensed dynamic objects are connected to specific nodes in the lower layers which can be used for risk extraction. Thus, the R-LDM is maintained in real-time.

Figure 4.2 illustrates the four mentioned layers of the R-LDM and gives some examples for each in the top right-hand corner.

With regard to lower layers, available map databases, such as the crowd-sourced OpenStreetMap[2] (OSM), can be applied. Here, OSM mainly consists of navigational information represented in the form of road centerlines and single intersection points. For this reason, we add the topological information on lane-level and estimate the actual lane geometry under assumptions of a specific lane width. Aside from this, stop lines are manually added and linked to incoming lanes, while the positions of traffic lights, crosswalks or buildings are similarly extracted from OSM.

4.2.1 Virtual Horizon

An advantage of graph structures, with their connected nodes, is the wide scope of queries, with regard to path finding, routing or tree search. It is straightforward to extract relevant situations. In particular, the knowledge concerning upcoming driving paths for cars is important in risk extraction. When considering start and end points in the map, we can query **sequences of nodes** (segments and junctions) along the route by traversing graph relations.

Figure 4.2 also shows an intersection layout. Therein, lane nodes (denoted as :LaneSegment and :LaneJunction) and traffic light nodes (:TrafficLight) are obtained, while the graph is explored based on current position

[2]See www.openstreetmap.org for further explanation.

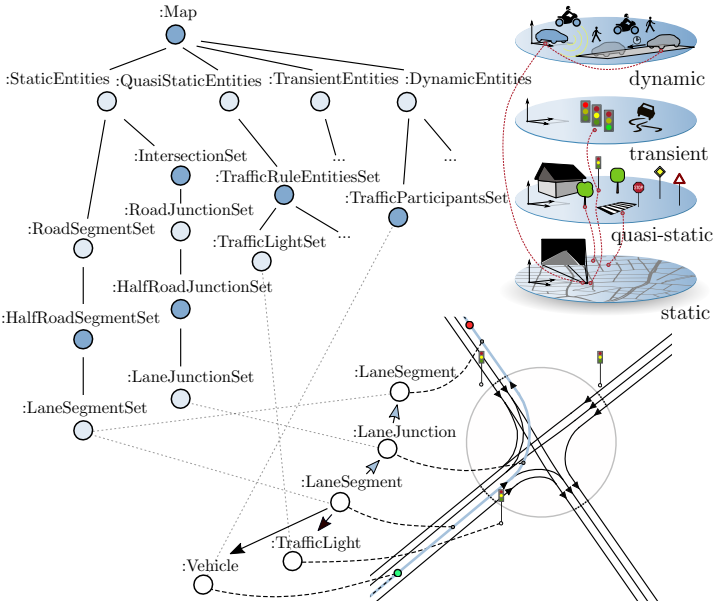

Figure 4.2: Nodes hierarchy and interconnection graph structure within the Relational Local Dynamic Map (R-LDM) for an intersection with traffic lights. The figure was taken from the publication [91].

and orientation of a selected vehicle (:Vehicle). We store relevant data in the attributes of the individual nodes. We are able to query any processed information that has been acquired online (e.g., from sensor data) or offline (e.g., from the OSM database).

Overall, the graph concept is accordingly based on layers (:StaticEntities, :QuasiStaticEntities, :TransientEntities, :DynamicEntities) and granularity, as shown on the top left part of Figure 4.2. In order to successfully calculate risks, not only one but several **possible paths** have to be queried. In particular, a virtual horizon can be determined, which extracts predicted paths for the vehicles in the complete traffic scene.

To illustrate the virtual horizon, Figure 4.3 addresses a scenario with two approaching cars. First, we map the GNSS-based ego vehicle position (in green) onto the center of the corresponding lane element. Then, we retrieve other cars from lidar (in red) up to a certain distance using R-LDM queries. More stored elements, for instance, quasi-static data with crosswalks can be equally obtained.

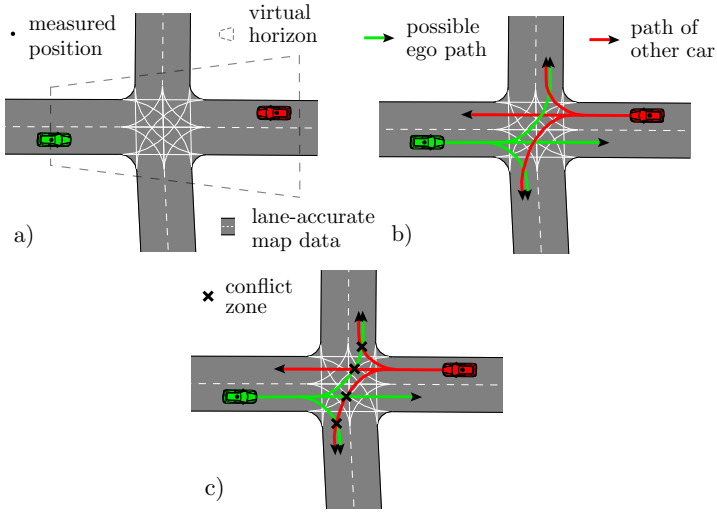

Figure 4.3: Steps of virtual horizon shown for collision risk. Top Left: Filtered relevant cars and their positions. Top Right: Retrieval of possible paths. Bottom: Map-based risk extraction. Crosses indicate path intersections.

By scanning the lane nodes for all cars, concatenating their polylines and ordering branches (straight, left and right, etc.) in a tree-like structure, we obtain different paths. In Figure 4.3, each car has three paths. The root of this tree is the current position, and its length increases with the number of segments traversed. Using the virtual horizon, we can, in the end, directly estimate the simple risks. Crossings of the driving paths reflect conflict zones where accidents are likely to happen. They are unique and critical for the given road.

4.2.2 Collision, Curve and Regulatory Risks

After querying the R-LDM for upcoming map structures, traffic elements and sensed objects, we predict future trajectories from the vehicle kinematics and evaluate their concrete risk time-courses. Proper safe maneuvers or velocities for recommendation are related to behavior planning. For this purpose, we divide risks into three generic types: 1) collision risk, 2) curve risk and 3) regulatory risk, similar as for the introduced risks in Chapters 2 and 3. These risk types are combined to support the driver. Hereinafter, the formulation of the different risks is described one by one.

Collision Risk

Alongside the coarse conflict zones from the last Section 4.2.1, risk metrics of a quantitative collision probability can be obtained. In a first step, paths must be transformed into spatio-temporal trajectories, e.g., using constant velocity models. For a vehicle-to-vehicle encounter (indexed with 1,2), this gives us position sequences $\mathbf{x}_1(s)$ and $\mathbf{x}_2(s)$ over the predicted time s with assumed constant velocities v_1 and v_2, compare Figure 4.4.

The distance between the vehicle pair represents the pointwise trajectory difference and can be written as

$$d(s) = \|\mathbf{x}_2(s) - \mathbf{x}_1(s)\| \,.$$

For time risks, we next consider the time of maximal criticality along the predicted trajectories. In other words, we filter out the minimal distance, which is the event distance d_E and get the corresponding Time-To-Closest-Encounter (TTCE) as s_E. For details about the calculation of d_E and s_E, please refer to Section 2.2.1. The two indicators are, in summary, retrieved with

$$d_E = \min\{d(s)\} \text{ and } s_E = \operatorname{argmin}\{d(s)\}.$$

Collision risk corresponds to the inverse of TTCE with $1/s_E$ and is only considered if the condition $d_E < d_{\min}$ holds true. In this way, the minimal distance value $d_{\min}$ defines some kind of risk "sensitivity".

Once there are multiple trajectory pairs as in Figure 4.3, we would iterate over newly introduced combinations $\mathbf{x}_{ij}$ (i.e., variation of ego paths i and other paths j) and extract the highest risk by $\max\{1/s_{E,ij}\}$. Therefore, in this method, we capture the time and space of passing vehicles.

Curve Risk

From quantitative risks, we can likewise infer recommended behaviors as a target velocity, which minimizes criticality and maximizes utility. This represents essentially a harder problem, since the driver should understand the warning as well as interpret and process the system's advice. Velocity planning for minimizing risks from sharp curves is explained now.

We consider a situation in which the ego user is driving with velocity v_0 at the current time t. A turn segment with curvatures $[\kappa_{\text{start}}, \kappa_{\text{end}}]$ is defined in the extracted path, if a thresholding condition

$$\kappa_{\text{start}} > \kappa_{\text{th}} \text{ and } \kappa_{\text{end}} < \kappa_{\text{th}}$$

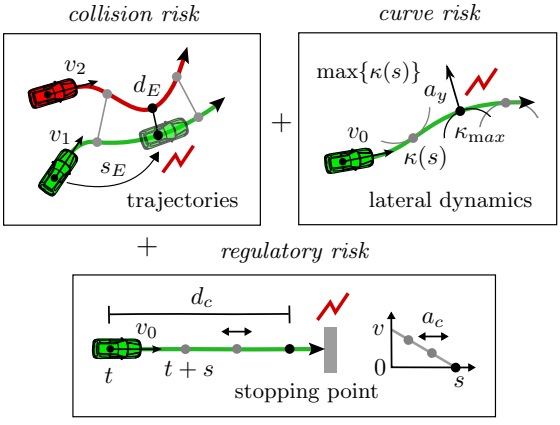

Figure 4.4: Three risk types of RNS with time as well as space metrics (collision, curve and regulatory risks). In this section, for simplicity, we will focus on time-based risk metrics.

becomes valid. Here, the threshold κ_{th}, similar to $d_{\min}$, lets us adjust the turn detection along the future profile $\kappa(s)$. In steep curves, the vehicle is prone to leave its lane. For higher velocities v_0, increased lateral acceleration is exerted depending on κ. When a_y depicts the lateral acceleration limit that the vehicle is able to follow and $\kappa_{\max} = \max\{\kappa(s)\}$ is the maximal curvature in the segment, we can derive a maximal target velocity

$$v_{\text{tar}} = \sqrt{a_y / \kappa_{\max}}. \tag{4.1}$$

In Equation (4.1), the variable v_{tar} indicates the speed that should be maximally reached at the curve. This is similar to the IDM velocity planning for curves (see Section 3.2). For low risks, we should thus move slower than v_{tar}, which leads to $v_0 < v_{\text{tar}}$ as a condition.

Regulatory Risk

Quasi-static elements of a stop line, traffic light or crosswalk create risks at rule violation. The reason behind this circumstance is that generally not obeying the norms can lead to unexpected and critical situations. One can think of a "virtual" risk exerting on the ego vehicle, when it is driving through prohibited zones. We can only drive safely because vehicles follow the rules.

In this sense, we want to stop directly in front of the traffic data. Mathematically, we assume a soft braking trajectory with constant deceleration a_c to the stopping point in the distance d_c along the ego path. With the future time s, we initially obtain the basic kinematic equation

$$d = \frac{a \cdot s^2}{2}. \tag{4.2}$$

Substituting s with the term v_{tar}/a_c and inserting $a = a_c$ as well as $d = d_c$ in Equation (4.2) ultimately results in

$$v_{\text{tar}} = \sqrt{2a_c \cdot d_c}, \tag{4.3}$$

whereby v_{tar} is the target velocity at which it can stop at the intersection with certainty. Equation (4.3) reminds, to some extent, of the IDM interaction term in Section 3.2. Here, a reaction time t_r can be added for long distances d_c using the formula $v_{\text{tar}} \approx \sqrt{2a_c \cdot d_c} - a_c \cdot t_r$.

Figure 4.4 illustrates the corresponding variables for the three types of risk from this section. At this point, it should be stressed that this approach employs explicit trajectory prediction along the map paths from the R-LDM. Specifically, we assume a constant velocity for the prediction in the collision case, an acceleration or deceleration trajectory in the curve case and a smooth braking trajectory to a fixed position for the regulatory case. We eventually filter and consider time points in the trajectory for the risks with Time-To-X indicators.

4.3 Visualization on Demand

In the preceding Section 4.2, we could see the process of storing the three main components of traffic situations. These were map structures, regulatory objects (i.e., stop lines or traffic lights) and dynamic obstacles (cars, bicycles, etc.). Furthermore, we understood how we can extract in their interaction relevant risks with time metrics. Based on the environment data and trajectory evaluation, ways of communicating the driving situation and risks are presented.

The warning system is achieved with a visual display (see Figure 4.5). We use a flexible renderer that visualizes data in Cartesian coordinates.

Nodes of the R-LDM have a range of potential attributes, such as the 3D position or the geometrical shape of objects. In the renderer, we can always visualize static and quasi-static data that are in the field of view from the ego driver. For this reason, a local 3D model is generated by converting

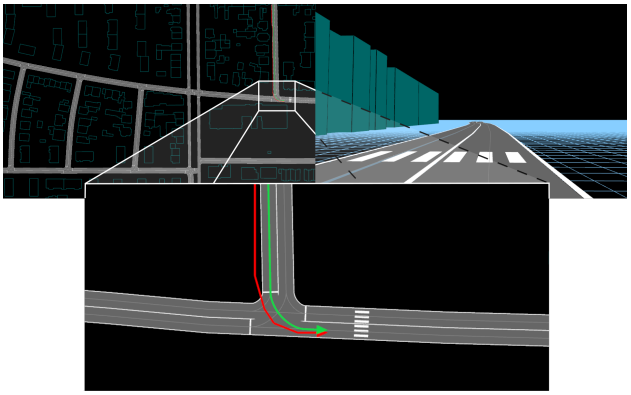

Figure 4.5: Rendered road network from two perspectives with the ego position being projected on the navigation route. The system can visualize, e.g., streets and their lanes, building outlines and vehicle driving paths.

the geographic points with (lat, lon, alt) into the ego-relative Cartesian coordinates of (x, y, z). Figure 4.5 shows, as an example, a map section in bird's-eye view, highlighting a single intersection. The first-person view of a car approaching a crosswalk is also depicted.

Dynamic data is constantly added to this static view. The zoomed-in excerpt from the map structure is given at the bottom of Figure 4.5, which includes a recorded GNSS trace (pictured in red). We project the trace onto the connected lane center (green). The virtual horizon and its possible paths are retrieved, as in the last Section 4.2.1. We can now update and move the excerpt using the actual position from the GNSS to obtain a live simulation. If driving risks are detected by the system, design elements are overlaid on this display. Here, we distinguish between proactive elements and short-term warning elements.

Proactive Support

Spatial as well as spatio-temporal relations are essential for intuitive risk-averse driver support. Further sources of information are a cause, likelihood and severity of potential risk [31].

The next logical step for RNS is the choice of suitable design elements. In this process, we assume that we know from the navigation route where the ego vehicle is driving (i.e., the ego path). From now on, the ego vehicle

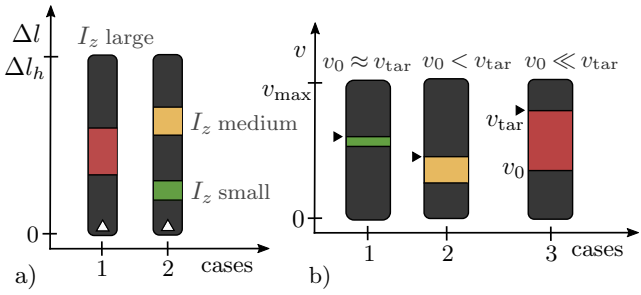

Figure 4.6: Chart elements for proactive support. With the hazard route (left), we can use distance to warn about hazardous zones on the ego path. The velocity scale (right) shows the recommended behavior to avoid risks. Color coding: green - low risk, yellow - moderate risk and red - high risk.

has only a single path for the risk estimation. Yet, for surrounding cars, all possible other paths are considered.

The so-called *hazard route* in Figure 4.6 is a concept that consists of a scale portraying distances to an upcoming risk element. Furthermore, the geometrical area or length of risks is considered. Risk is thus measured with respect to the ego path, ranging from the current position $\Delta l = 0\,\mathrm{m}$ to the end of the path Δl_h. In doing so, the length Δl_h can be chosen according to individual preferences. The larger Δl_h is, the earlier a driver can be warned. However, the more uncertain is also the warning (i.e., false positives could occur).

At an upcoming intersection, risk is defined by the section of the path that lies within the junction. Since risk corresponds to exposition time, we encode the path part from the intersection I_z with colors, ranging from green for short intersections to red for long ones. Figure 4.6 a) gives two examples of the hazard route. The left bar shows a large intersection (e.g., multi-lane four-way stop) in the vicinity and the right bar has a small and consecutive medium junction. This emphasizes that we may include more than one intersection in the warnings.

The *velocity scale*, Figure 4.6 b), is a second chart element which qualifies the difference between the current velocity of the vehicle v_0 and the target velocity v_{tar} from the trajectory evaluation of Section 4.2.2. The scale shows possible velocity values, from standstill $v = 0\,\mathrm{m/sec}$ to a maximal velocity $v_{\max}$. Depending on the velocity difference $|v_0 - v_{\mathrm{tar}}|$, the situation is rated as safe with $v_0 \approx v_{\mathrm{tar}}$ (green, left), as dangerous with, e.g., $v_0 < v_{\mathrm{tar}}$ (yellow, middle) to critical with $v_0 \ll v_{\mathrm{tar}}$ (red, right). The same holds true

for the opposite circumstances, when the actual ego velocity is larger than the recommended speed, i.e., $v_0 > v_{\text{tar}}$. This velocity scale is employed for curve or regulatory risks. Moreover, we may set a speed limit as the target velocity v_{tar}, once there is no risk ahead.

Short-Term Warning Elements

In order to emphasize the criticality of a situation, further intuitive warning elements, as, e.g., pop-up signs and lane colorings, are proposed. The following elements augment the proactive elements.

Explicit symbols indicate risk causes accompanied by the event time for collisions (s_E), distances from the risk spot for turns (i.e., right curve with d_r and left curve with d_l) or stopping distances for crosswalks (d_c). In Figure 4.7 a), the *pop-up signs* are pictured. The different signs are just a selection and more risk causes can be added. Their purpose is to clarify the reason for the warning and give information that can be understood by humans, instead of technical outputs.

Finally, we can highlight lane parts or positions according to corresponding risks. In the instance of curve and regulatory risk, the lane is colored from the ego position up to the point of maximal risk. For collision risk, we mark the point of the closest encounter as a red cube. An illustration of the *colored lane* is depicted in Figure 4.7 b) for regulatory risk, induced from a stop line. Again, the color is defined by the deviation $|v_0 - v_{\text{tar}}|$. It shows the therein considered navigation route with length Δl_h and another unlikely path at an intersection.

Note that the visualization of warnings only occurs if the risks are actually present. Altogether, the RNS provides a variety of tools for analyzing and circumventing critical driving situations, while trying not to overload the driver's awareness.

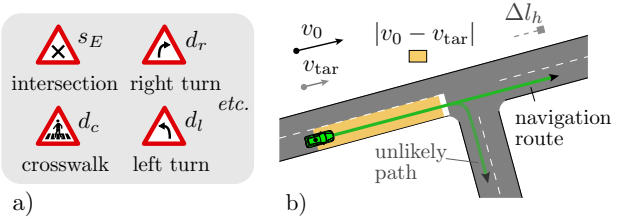

Figure 4.7: Short-term warning elements. Selected pop-up warnings (left) indicate the risk source, e.g., risks from a crosswalk. Additionally, a colored lane (right) depicts the spatio-temporal ego behavior relation to the risk.

4.4 Field Tests

In the evaluation of the complete Risk Navigation System (RNS) from Figure 4.1, the system is tested on real-world driving recordings employing the middleware RTMaps.[3] The used mobile platform is a modified Honda CR-V car, equipped with an OXTS localization device[4] and six Ibeo Lux lidars allowing 360 ° perception.[5] Additionally for debugging, a front-facing camera provides images. In the current implementation, RNS runs with a frequency of 10 Hz while employing the library OpenGL for environment rendering in the HMI.

Multiple recordings have been acquired from areas around Offenbach am Main, Germany. The inputs required by the system are positions, velocity estimates and angle measurements of both the ego car and surrounding vehicles. Regarding localization of the ego vehicle, we project the GNSS signal onto its connected navigation route. Therefore, the route of the user needs to be known in advance. In this sense, we replay the recordings and save the position trace. Alternatively, a filter could be added that selects the most likely ego path. Throughout the experiments, the R-LDM is queried for risks-related data around the current position.

Qualitative Outputs

In three intersection cases, we assess the support of RNS for approaching, turning and crossing tasks. Qualitative outputs of the RNS visualizations are shown as follows below. Furthermore, output values of distances d_c, d_r, d_l and times s_E as well as driven v_0 and target v_{tar} velocity are provided. If necessary, these quantitative values can be turned off (using a slim mode of RNS) in order to reduce the user's workload.

The first test run is on an X-shaped junction with another oncoming vehicle. Chronologically organized snapshots of the scenario are given in Figure 4.8, with front camera images on top and the synchronized RNS output on the bottom. RNS marks the recorded ego trace with red lines and the projected position as a green arrow tip. For the prediction horizon, we set $\Delta l_h = 50\,\text{m}$. Other detected cars are visualized as yellow dots, while the other most critical vehicle is depicted in blue, displaying its possible paths as well.

In the initial image, the ego car is approaching the intersection. As can

[3] For the RTMaps software, please consult www.intempora.com.
[4] Specifications of OXTS gear are described in www.oxts.com.
[5] Details of Ibeo sensors can be found on www.ibeo-as.com.

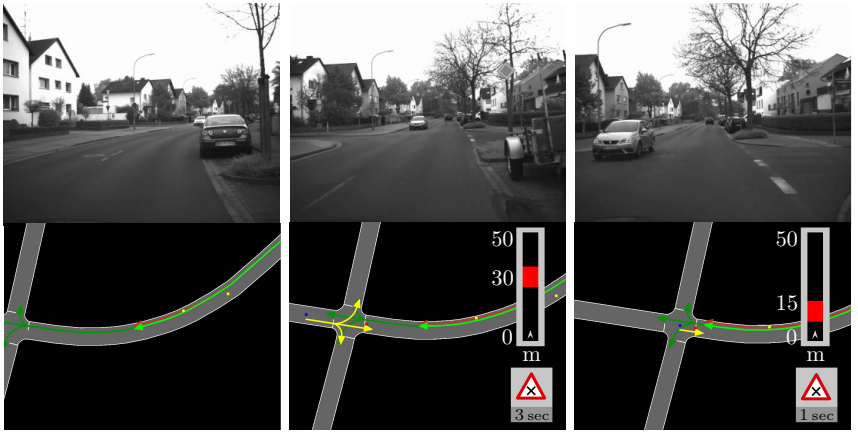

Figure 4.8: Collision risk example. At intersections, RNS computes TTCE for ego and other vehicles' paths and informs about the critical encounter point (red line: recorded trace, green line: trace projected on navigation path). Top: Front camera images. Bottom: Screen layout of system.

be seen, the junction lies at a distance of 30 m in the next picture. The RNS visualizes the point of closest encounter and indicates the event time $s_E = 3$ sec under the given traffic sign. At the time of the third picture, we nearly passed the other car with $s_E = 1$ sec and are close to the intersection, i.e., distance $d_I = 15$ m. When the vehicle passes the intersection layout, a critical situation is no longer present. The scenario shows how RNS can inform the driver about possible situations and improves the prediction capability.

The second test represents a T-junction with a sharp and critical right turn (see Figure 4.9). In this context, $v_{\text{tar}} = 4$ m/sec describes the velocity the driver should have adopted when reaching the curve. Since the vehicle exceeds v_{tar}, the velocity scale turns red. However, due to the appropriate behavior of the ego driver, i.e., reduction of the speed v_0, the scale changes to yellow on the second image from the left. Here, the distance to the turn d_r decreased simultaneously from 40 m to 10 m. When arriving at the curve, the driver now matches the RNS target velocity $v_0 = v_{\text{tar}}$, denoted in green. This example shows how users could leverage recommendations from curve risks, using the velocity scale of the RNS.

In the third experiment, a pedestrian intends to use a crosswalk and the car driver ignores the related regulatory risk. It should be considered that

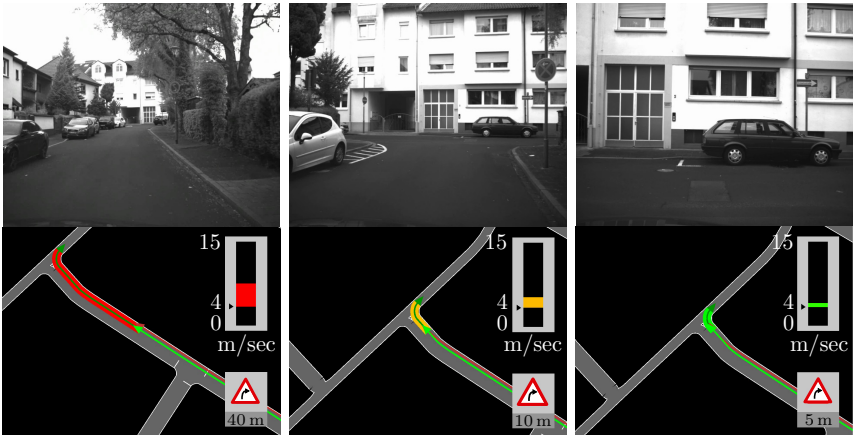

Figure 4.9: Curve risk example. RNS detects the sharp turn and recommends decreasing the velocity, which the driver is shortly after abiding to. The velocity scale goes from the color red to orange and green (i.e., due to difference of actual to desired velocity).

this crosswalk does not have traffic lights, as it may often be the case, e.g., in Germany. Since the R-LDM stores traffic elements, we can warn the driver already $d_c = 90\,\text{m}$ in advance with a pop-up symbol. In the following sequence of Figure 4.10, the driver keeps his velocity v_0 nearly constant, while the suggested stopping trajectory v_{tar} changes from 12 m/sec with a green warning, via 7 m/sec with a yellow velocity scale to 3 m/sec at 10 m distance to the waiting pedestrian. Due to the deviation between driver and RNS, the warning became critically red in the end. In turn, we are able to guide the driver's awareness towards relevant risks.

Performance Discussion

Overall, the given results underline the potential of RNS to proactively support the driver. We successfully utilized a risk-based navigation on the strategical level with time horizons of multiple seconds. Its visual display allowed us to consistently analyze the situational points of interest. This may potentially increase the drivers' trust in the support system because they can see meta-information from the underlying calculation.

The applicability of the concept was demonstrated in urban scenarios with single-lane streets. However, when handling complex roads, accurate

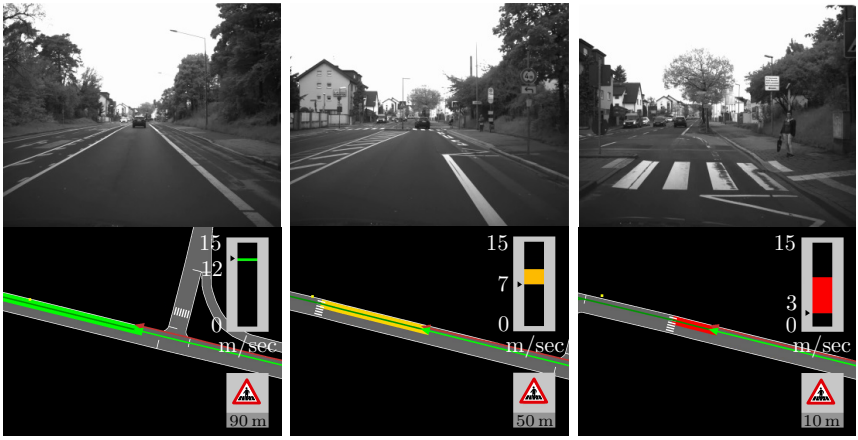

Figure 4.10: Regulatory risk example. In this situation, the driver does not let a prioritized pedestrian pass first on the crosswalk. RNS warns of traffic objects, displays the distance and colors the ego lane according to the risks.

lane-level localization becomes more important. On one side, the ego car must be projected onto the correct lane, which puts demands on the lateral localization precision. On the other side, obstacles that are sensed relative to an ego car must be assigned to their own proper lane, which adds further requirements to both lateral error and ego orientation estimation.

Risk models and planning problems become complex in these lane change situations, specifically, also due to the motions with high uncertainty from the cars. This problem is tackled by applying the novel risk model presented in this dissertation.

4.5 Application of Risk Maps

The previous risk application focused on the storage and retrieval of risk-related data in driving. Path constraints were relaxed by computing alternative driving paths of vehicles. We could see how an integrated HMI can display warnings of several risks. Hereby, the exploited time indicators, i.e., Time-To-Event (TTX) indicators, represent a baseline for risks.

Going even one step further back, the combination of a Gaussian method and survival analysis for collision risk predictions was proposed in Chapter 2. It is suited for motion planning in situations with multiple interact-

ing cars and has advantages over TTX measures. In simulations of Chapter 3, we showed that this method works robustly and does not produce collisions. For the application in this section, we modify the approach and apply the *theoretical risk framework* on the test vehicle for *driver warning* in forced lane changes or mergings.

System Overview

Lane changes are principally forced due to, e.g., ending lanes, construction areas or further parked vehicles. Figure 4.11 depicts such a driving scenario, see upper part. The green ego car must drive safely onto the neighboring lane, while two red cars must be considered. It can perform a lane change and accelerate, or stay on its lane and then brake. However, at some point, the car needs to change the lane. This scenario is deemed complex because of the potentially large sensor noise and prediction space.

By developing the planner called Risk Maps (RM), based on the novel risk from Section 2.4, we may efficiently find *lane change timings*. From a set of predictions, a single trajectory with minimal costs (tradeoff between risks and ego utility with comfort) is selected. In this process, a crucial module is again the Relational Local Dynamic Map (R-LDM) that helps to extract map paths, fuse sensor information and prepare situations. Es-

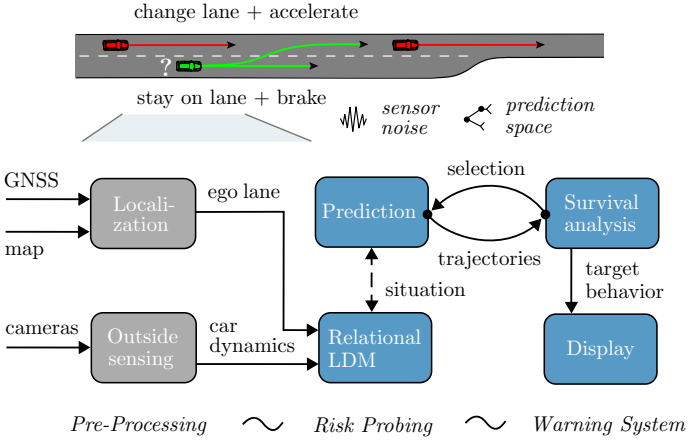

Figure 4.11: The target is to employ the Risk Maps (RM) technology in a real prototype for situations with dynamic lane changes. Hereby, we focus on the system blocks marked in blue.

sentially, the target is to move from simulation into real world. We focus on improving *uncertainty-awareness* and *transparency of the system*.

This time, the prototype setup consists of an inexpensive GNSS and several camera sensors. Figure 4.11, see bottom part, summarizes the system architecture with localization and outside modules.[6] In real experiments, RM is demonstrated to run on the test vehicle and create warning outputs on a HMI. The resulting system is able to recommend target velocities as well as path choices (e.g., go left), separated in gap and no-gap variations. As the main novelty, the visualization of uncertainties is enabled in a risk graph for future times.

Hereafter, the section is split into two components: R-LDM and RM. Concerning the R-LDM, we examine the static layer and its connection to the dynamic layers more in-depth than in Section 4.2. RM highlights the novel risk model of this dissertation for real-time planning and experiments round off the real application. For this analysis, driving recordings from a demonstration in the ITS European Congress[7] are taken.

4.5.1 Fusion of Driving Situations

A clear data structure of driving situations (paths, vehicles and their trajectories) is a prerequisite for an effective application of warning systems. In this context, an interconnected world model is beneficial as central storage, since it can fuse data of map geometries and sensor measurements.

We shared this aspiration and therefore have constructed the R-LDM as a graph, which saves data in layers depending on their variability. On the bottom layer, static map data of the driving environment is stored. This includes spatial geometries that are afterwards enriched by the dynamic entities, e.g., traffic participants, from higher graph layers. The R-LDM has data nodes, where data is stored as attributes. As relationships between nodes are a main component of graphs, connections between those nodes are lastly realized as relations (e.g., temporal or semantic).

Static Layer of R-LDM

The possible paths a driver will likely take are of interest, particularly, in the case of driving risk models. In Section 4.2, we composed a related graph layout and understood how to query single nodes over their relationships.

[6] This warning system was realized in the EU project Vision Inspired Driver Assistance System (VI-DAS).

[7] For details, see https://2019.itsineurope.com/demonstrations.

In this context, the different stored path attributes are now examined in the static layer of the R-LDM.

Within the R-LDM database, the center line of a driving lane serves as an approximation for paths. In order to conceive structures of lanes, we store roads as subgraph patterns consisting of one central node (described with labels :LaneSegment). Underneath, straight lane nodes alternate with junction nodes, where lanes are crossing. Our detailed attached property nodes are as follows:

- Polygonal lines, in other words, polylines, for the left and right lane boundary (i.e., the :LeftBoundary and :RightBoundary nodes),
- The **center polyline of lanes** (main :Centerline node), describing in combination with the lane boundaries the drivable area,
- Types of lane markings, the driving direction for the road plus related road angles and, finally,
- Other information, like road curvature.

Hereby, property choices depend on the application. In lane change situations, these listed properties can be considered appropriate. There are no limits for further information, but the omission of information can also be beneficial, since the computing time is reduced if, for instance, only center lines and not full geometries are saved.

However, when dealing with **multi-path roads**, the assumption of driving on the exact center line does not always hold true. Polylines describing an average driving path are therefore required. Driving paths must be refined. In the R-LDM, paths can be derived from publicly available sources (e.g., OSM), as in Section 4.2. For higher accurate maps, alternatively, we generate paths by parsing image data, such as satellite images, and refine them with recordings from previous drives.

Sensor Processing

We have successfully stored map geometries in the static layer of a graph. Subsequently, sensor data is prepared and fused with these maps. We use two techniques to process sensors: 1) GNSS and camera outputs are transformed into joint Cartesian coordinates, and 2) the ego and other detected vehicles are aligned on their paths. Figure 4.12 summarizes the steps with real-world sensors. In this way, we can predict trajectories along the driving paths.

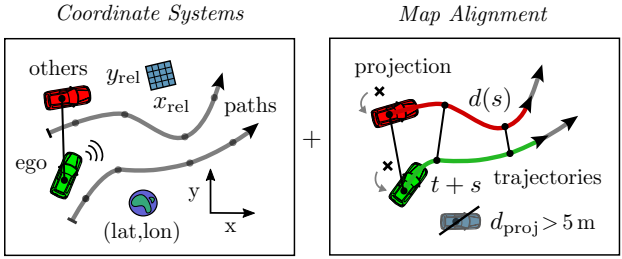

Figure 4.12: Pre-processing steps for map-based planning in R-LDM. Left: Car signals are transformed into the same global reference frame. Right: For motion predictions, we project the sensed vehicles on paths.

We assume ego positions in (lat, lon) to be given by the GNSS sensor. By employing an equirectangular projection with a radius of the earth r_e and set projection origin of the latitude lat_0, the geographic points (lat, lon) may be converted to Cartesian coordinates (x, y) via

$$\begin{aligned} \text{x} &= r_e \cos \text{lat}_0 \cdot \text{lon}, \\ \text{y} &= r_e \cdot \text{lat}. \end{aligned} \tag{4.4}$$

For Equation (4.4), the parameter lat_0 should be chosen in a way that the point is close to the midpoint of the map. Note that, in practice, we will use a more sophisticated but similar transverse **Mercator projection**.

The detection of vehicles in cameras requires a transformation of their positions as well, i.e., from the ego-relative body frame $(x_{\text{rel}}, y_{\text{rel}})$ into the global frame (x, y). For this purpose, we use a **rotation transformation** with the angle Θ between both systems. The standard, right-handed orientation of this frame is imposed. Regarding flat surfaces, we obtain for the positions of other vehicles

$$\begin{aligned} \text{x} &= \cos\Theta \cdot x_{\text{rel}} - \sin\Theta \cdot y_{\text{rel}}, \\ \text{y} &= \sin\Theta \cdot x_{\text{rel}} + \cos\Theta \cdot y_{\text{rel}}, \end{aligned}$$

whereby the ego position is afterwards added as an offset.

Overall, these transformations allow us to align the ego car on the closest (x, y)-point of a path, which is determined by the localization module. Other vehicles are projected to their own nearest path segment. However, we only take into account the vehicles that are close to a path and filter out the ones with distances of $d_{\text{proj}} > 5\,\text{m}$ away. Such geofencing limits position errors from camera sensors. In the final goal of the R-LDM graph,

the aligned vehicles on the paths are then fused into a driving situation. The upcoming section deals with the modifications of a R-LDM based on dynamic data.

Dynamic Layers

Lastly, driving situations are saved within the R-LDM in terms of single trajectory combinations of vehicles. This enables motion planners to reduce the relevant velocity space. In comparison to a path, which contains spatial points in a 2D-space, we say that a trajectory depicts the **driven sequence of a vehicle** and has timestamps $t + s$ for each point.

Concretely, once signals are acquired from online sensors, in the R-LDM, we continuously create a related object node and attach it to the lane nodes from previous sections. Specifically, the connection will be tagged with a :Contains relation, whereby the colon indicates a graph property. Object nodes may be of the three last R-LDM layers, which represent :QuasistaticEntity, :TransientEntity and :DynamicEntity. For cars or further vehicles (i.e., a :DynamicEntity), trajectories are then predicted by assuming fixed dynamics. We store the sequence of position (x, y), and, if available, of distance d, velocity v, acceleration a and jerk j.

As the objects are distinguished by the R-LDM layers, it is eventually possible to query a specific node when following not only lane-lane, but also object-lane and object-object connections. Figure 4.13 illustrates the interfaces between the static layer and the three dynamic layers. In the figure, world variables are mainly ego situations and the situations from

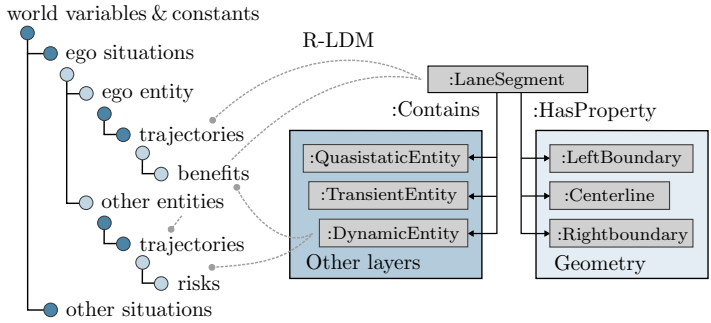

Figure 4.13: Data connections between the static (lane geometries) and dynamic layers (quasistatic, transient and dynamic data, combined with trajectories) within a relational graph.

the perspective of other vehicles. For each situation, we list: trajectories, ego benefits and risks for the driving environment. Effectively, the R-LDM helps to structure white-box model data that is leveraged for risk-based planning.

4.5.2 Probing with Risk Maps

In this section, the novel driving risk model from Chapter 2 is integrated with the idea of optimization, or rather the sampling, of ego trajectories, in comparison with Chapter 3. The focus lies on improving the *computation speed* for real-time applicability of the resulting Risk Maps (RM).

Using the R-LDM, we were able to fuse measured data into driving situations. However, assessing a multi-lane situation for planning thoroughly and, at the same time, in a fast manner remains very challenging. The reason lies mostly in the inherent combinatorics from these situations. With map data, the driving space was constrained and we differentiated between paths on the one hand, and trajectories on the other hand. Still, a *single path and trajectory* that is safe and beneficial for the ego driver needs to be found.

Therefore, so-called "probing" inside RM will be leveraged. We consider probing as the prediction of an ego motion and its evaluation of induced risks. By *intelligently probing* adequate numbers of fixed trajectories along paths in RM, motions will be efficiently planned. For this purpose, the trajectory variation with RM is first defined and we continue with the cost evaluation. Section 4.5.3 finalizes the description of RM with the extension of path planning, which ultimately allows to warn the driver.

Trajectory Variation

Hereinafter, we will analyze, as usual, the instance of a car-to-car encounter with, e.g., collision risks, depicted in Figure 4.14. For every ego trajectory variation h, constant speed is assumed for the other car (i.e., $v = \text{const.}$). Technically, we first sample N_t velocity profiles $v(s)$ on the path that start at the current velocity v_0 and end at the planned velocities v^h in the future time s, equidistantly sampled inside $v \in [0, v_{\max}]$. Accordingly, we get

$$v^h = \frac{h}{N_t - 1} \cdot v_{\max} \text{ with } h \in 0, ..., N_t - 1.$$

To reach those end velocities v^h, accelerations a^h are afterwards calculated that are maximal with $a_{\max}$ in case of an ego trajectory that corresponds

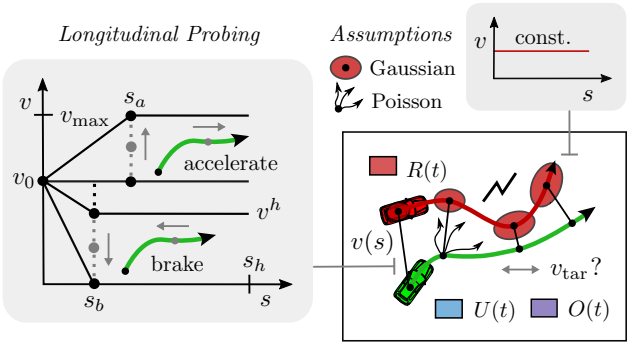

Figure 4.14: Trajectory prediction with RM. Left: Ego trajectories are planned with different acceleration and braking strengths, see gray arrows. Right: Evaluation of the risks, utilities and comfort for constant velocity of another car.

to $v_{\max}$. For the cases ending in a full stop $v^h = 0\,\text{m/sec}$ of the ego vehicle, acceleration becomes minimal with $a_{\min}$. Note that $a_{\min}$ represents a negative acceleration (i.e., braking motion). In total, this leads to

$$\text{if } v^h \geq v_0\text{: } a^h = a_{\max} \cdot \frac{v^h - v_0}{v_{\max} - v_0},$$

$$\text{else if } v^h < v_0\text{: } a^h = a_{\min} \cdot \frac{v_0 - v^h}{v_0}.$$

The intervals with either an acceleration or braking phase are executed for predicted durations of $s \in [0, s_a]$ or $s \in [0, s_b]$, respectively. We formulate the included times as

$$s_a = \frac{v_0}{a_{\max}} \text{ and } s_b = |\frac{v_0}{a_{\min}}|.$$

Hereby, a steady velocity is assumed in the subsequent interval $[s_a, s_h]$ and $[s_b, s_h]$. The parameter s_h defines the time horizon of the achieved velocity profile and is set according to the task. Figure 4.14 (on the left) depicts the longitudinal probing and the parameters for v and s.

In the presented planner, a target velocity $v^h = v_{\text{tar}}$ from the RM set is selected based on the explicit tradeoffs between risks $R(t)$ and benefits, which are further divided into the utility $U(t)$ and comfort $O(t)$, see Figure 4.14 on the right. In this process, at least $N_t \geq 3$ ego trajectories have to be sampled so that RM can choose between an acceleration and a braking option. Altogether, RM could potentially represent a fast planner applicable to avoid accidents.

The following will now recap these mentioned costs to select an optimal motion. Using the survival analysis, we include probabilistic assumptions of Gaussian uncertainty (visualized as 2D, red ellipses) and of Poisson processes (escape arrows for the ego car). However, this section represents a summary of the approach. For details, please refer to Chapter 2.

Survival Analysis

We cannot presume that predictions of other cars, e.g., constant velocity, will be followed in reality. With the novelty of this dissertation, trajectory positions were thus modeled using 2D Gaussians, which grow over the future time. In Section 2.4.2, collision probabilities are given by the **Gaussian overlaps** of car pairs. An analogical formulation is likewise utilized for the probability to drive off in curves. We modeled 1D Gaussians for the ego car and analyzed the exerted lateral acceleration.

Finally, with the help of a **Poisson process**, probabilities could be integrated with damages into one scalar, i.e., risk value $R(t)$, for the current time t. In this process, probabilities are formulated as critical event rates τ_{crit}^{-1} and the added survival function S reduces the influence of risks occurring later in predicted times s. By distinguishing between collision rates τ_{coll}^{-1} and curve rates τ_{curv}^{-1}, risks $R(t)$ are summarized with

$$R(t) = \int_0^\infty (\sum_j \tau_{\text{coll},j}^{-1} D_{\text{coll},j} + \tau_{\text{curv}}^{-1} D_{\text{curv}}) S \, ds. \tag{4.5}$$

A contained index j in τ_{coll}^{-1} from Equation (4.5) considers multiple cars, while the damage models $D_{\text{coll},j}$ and D_{curv} have to be described in a desired accuracy. Please remind that variables $D_{\text{coll},j}$, D_{curv}, and S are functions of the current time t and the future time s, or $t + s$, respectively.

Since RM also optimize utility and comfort for the ego driver, the other two cost aspects will be shortly discussed as well. Utility $U(t)$ depends on the ego velocity v_1 (general required time to arrive at a goal) and a desired velocity v_d (the individual preferences). For decreasing effects of $U(t)$ at higher times s, we multiply the components with the survival function S and write

$$U(t) = \int_0^\infty (b^t |v_1| + b^d |v_1 - v_d|) S \, ds.$$

Aside from this, comfort $O(t)$ takes ego acceleration a_1 and jerk j_1 into account and results in less abrupt switches between selected behaviors in

the planner. The major target is still to avoid risks. Consequently, comfort is also reduced via the survival function S. For $O(t)$, we get

$$O(t) = \int_0^\infty -(b^c|a_1| + b^j|j_1|)S\,ds.$$

The importance of risk versus benefit can be tuned utilizing the weights b^t, b^d, b^c and b^j, in both $U(t)$ and $O(t)$. Again, please note that dependencies of v_1, a_1 and j_1 to the times $t+s$ have not been written out.

An advantage of explicit predictions within RM is the straightforward visualization of costs. RM contain rich meta-information. As an example, once we calculate the rates τ_{crit}^{-1} of $R(t)$ for the ego trajectories over future times $v(s)$, we can generate *predictive risk graphs*. Figure 4.15 plots such a graph. It depicts risk spots for two separate cars, which become visible if the ego car comes close to another car. The shape of the risk spots depends on the cars' Gaussian uncertainties. Additionally, the figure shows the two components of $U(t)$ and $O(t)$. While $a_{\min} < a < a_{\max}$ is valid, comfort costs are lower. Utility is highest for velocities that are close to the desired speed v_d.

In the end, we calculate the total costs $C(t) = R(t) - U(t) - O(t)$, which allows us to select the trajectory with minimal c^h among all sampled tra-

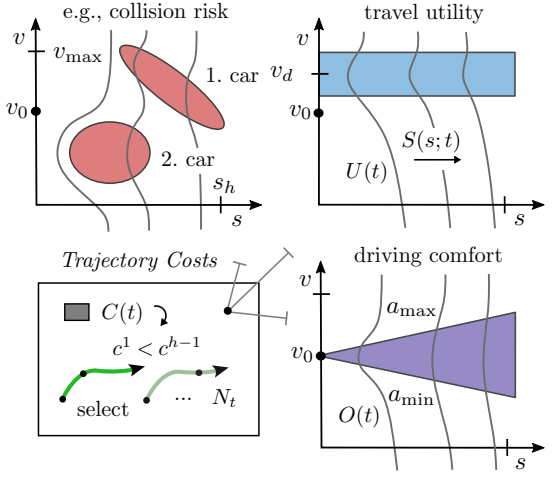

Figure 4.15: Visualizations of the driving risk (i.e., from collision and curve), utility and comfort costs. The trajectory costs are computed over the future time and then integrated with the survival analysis to obtain a single cost scalar.

jectories N_t. Particularly, in Figure 4.15, the first trajectory v^1, i.e., with $h=1$, was chosen because c^1 represents the smallest cost of all trajectories. With the cost graphs, we can intuitively indicate the underlying reason for this selection.

4.5.3 Path Planning

For the last module of the resulting warning system, this section outlines path planning. Building upon the ego trajectory variation and cost evaluation from RM, a novel path probing technique will be proposed. In this way, the use case of the introduction is solvable, see Figure 4.11. The driver has the support of the HMI, recommending a target motion.

As already mentioned, in large multi-lane roads (i.e., highways), a driver has distinct spatial options with the possibility to perform a lane change. In these instances, the lane change path and its start time and duration needs to be determined. The RM make **tactical decisions**, which are, e.g., lane changes for a future time (depicted in Figure 4.16).

In detail, we compute a path change that serves to blend between the centerline of the ego lane and the centerline of a neighboring lane. The blending begins at a longitudinal distance l_{start}, defined as

$$l_{\text{start}} = v_0 \cdot s_{\text{start}}.$$

Accordingly, this distance l_{start} depends on the current velocity v_0 and the future time s_{start}. The length of the blending interval l_{blend} is, in contrast, influenced by the current, lateral distance d_{path} between the two path options. Since the end of the segment l_{end} is composed of l_{start} and l_{blend}, we retrieve

$$l_{\text{end}} = l_{\text{start}} + l_{\text{blend}} = l_{\text{start}} + v_0 \sqrt{l_c \cdot d_{\text{path}}}. \tag{4.6}$$

In Equation (4.6), a parameter l_c defines the scale factor for the increase. Simply put, for higher l_c, the lane change would also take more time.

In a next step, we blend the ego path $\mathbf{p}_{\text{ego}}(l)$ into the neighboring path $\mathbf{p}_{\text{other}}(l)$ within the segment l_{blend}, using a sigmoid weighting term $w^*(l)$. Paths from Section 4.5.1 are initially resampled with evenly spaced points. The computed lane change path $\mathbf{p}_{\text{blend}}(l)$ then follows, with

$$\mathbf{p}_{\text{blend}}(l) = (1 - w^*(l)) \cdot \mathbf{p}_{\text{ego}}(l) + w^*(l) \cdot \mathbf{p}_{\text{other}}(l).$$

Generally, linear blending is achieved via the term

$$w(l) = \frac{l - l_{\text{start}}}{l_{\text{blend}}}, \text{with } l \in [l_{\text{start}}, l_{\text{end}}],$$

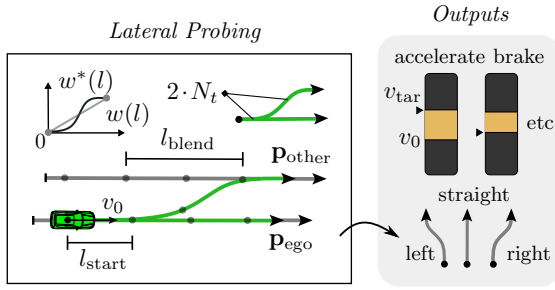

Figure 4.16: Signal outputs of RM. By also probing the costs for possible ego paths, we can plan tactical lane changes. Left: Blending of path options and their selection. Right: HMI with target velocity and lane change advice.

and we thus utilize the weight $w(l)$ in a sigmoidal weight function $w^*(l)$ with constant k to gain

$$w^*(l) = \frac{1}{1 + e^{-k(w(l)+0.5)}}.$$

Figure 4.16 (left box) depicts two prototypical paths. They are calculated with the aforesaid variables. Note that points in the path have to be sampled densely enough for a reasonable blending. With the **path blending**, we can initiate an immediate lane change by setting $s_{\text{start}} = 0\,\text{sec}$, while a tactical change is obtained with, e.g., $s_{\text{start}} = 2\,\text{sec}$. A utility offset u_l is necessary for the costs in order to incentivize the planner making lane changes. After all, RM choose the optimal path based on its costs.

Selection and Warning

For multiple paths, the path blending is done iteratively. With the total path number M_p, ultimately, $M_p \cdot N_t$ samples are generated in RM, since ego trajectories are varied on each path. We select a single trajectory with the lowest costs to obtain the target velocity v_{tar} as well as the target path $\mathbf{p}_{\text{tar}}$. In other words, we probe in longitudinal and lateral directions. The runtime of RM is constraint because we have a fixed sample size. This suits very well for real-time purposes. In the case of Figure 4.16, we get $M_p = 2$ and sample $2 \cdot N_{\text{t}}$ trajectories.

This planned solution is now transferred into a driver suggestion or warning in an HMI so that it is explained to the driver. We compare the planned behavior with the actual one to infer a warning.

As shown in Figure 4.16 (right-hand side), the developed HMI contains a *velocity scale* with the current v_0 and the safe velocity v_{tar}. Depending on the difference $|v_0 - v_{\text{tar}}|$, the driver needs to change its behavior (accelerate, brake, etc.). Furthermore, a *directional arrow* depicts the path choice, which can be either "left", "straight" or "right". The driver should make a lane change, i.e., left, right, or stay on the lane, i.e., straight motion. The figure illustrates some output examples of the HMI, which are derived from the planner.

4.6 Experiments

For the real application, we target to analyze the quantitative results of the system combination Risk Maps (RM) with the Relational Local Dynamic Map (R-LDM). Hereby, a demonstration took place within the frame of the ITS European Congress 2019 at Helmond, the Netherlands. The system was run live in front of scientific and non-scientific audiences.

The used car prototype is called Carlota and was provided by the Spanish research institute Vicomtech. Its sensor setup consists of a mid-range GNSS[8] and cameras. In more detail, the cameras are four Sekonix[9] cameras, which are combined with two Nvidia GPUs[10] and one Intel CPU. For this reason, Carlota can be seen as a low-cost test solution.

In total, the R-LDM serves to provide environmental data, i.e., paths for the risk calculation, at all times. Scenarios were recreated with three test drivers, maneuvering Carlota and two further vehicles. In each of those runs, RM supports to execute a safe lane change. The end of the road, indicated by cones, forces a lane change of the ego vehicle. In this context, the speed intervals for the tests ranged from 20 km/h to 40 km/h with a closed public highway stretch of 1.5 km length.

In what follows, we analyze and evaluate the given HMI concept, which is built on three components: a risk graph (top left of Figure 4.15), a velocity scale (Figure 4.16, top right) and lane change recommendations (Figure 4.16, bottom right). While demonstrating the system online, the velocity scale was displayed on an instrument cluster, inside the vehicle. For the lane change recommendation, LED stripes were used that were attached to the windshield and are visually conveying suggested driving

[8] The product model is given on the website https://www.u-blox.com/de/product/evk-8evk-m8.

[9] See the cameras on http://sekolab.com/products/camera.

[10] Please refer to https://www.nvidia.com/en-us/geforce/graphics-cards/rtx-2080-ti.

directions. Additionally, both elements, together with the risk graph, have been presented on a projector to the audience.[11]

The remainder of this section is structured accordingly: we first examine the data fusion of the sensors with the R-LDM as a knowledge hub for support systems. Afterwards, within Section 4.6.1, the results of RM and its HMI are described and discussed in regard of analyzing the lane change situations.

System Integration

Required inputs for RM are the ego state and its map relation. In a first step, we obtain a (visually) lane-matched ego position from a localization module. The ego vehicle is thus projected onto the lane center, determined by the localization module (i.e., the current ego lane). The R-LDM can now provide driving paths. As a second input of the demonstrated system, we need measured positions and velocities of other cars, see Section 4.5.1. In this pre-processing step, the four cameras (i.e., left, right, front and back) are used to detect surrounding obstacles in a 360° view. Specifically, detected bounding boxes of a YOLOv3 neural network [96] are projected from each camera image into a joint 3D world.

In the end, we fuse the pre-processed sensor signals with map data from the R-LDM, the graph-based environment representation. This can give us a driving situation for the current time.

At every timestep, $N_t = 21$ ego trajectories are sampled and predicted with RM (10 acceleration and 10 braking ramps with different acceleration/braking strength, and 1 constant velocity) to retrieve a situation for the R-LDM. Here, sampling is done for the current and parallel lane paths (i.e., $M_p = 2$). Similarly to Section 4.2, with the middleware RTMaps, we may ultimately integrate the software components of R-LDM and RM in a single system. RTMaps reads the sensor inputs and writes the final HMI outputs. The system could run with a frequency around $f = 10\,\text{Hz}$, which is suitable for automotive applications.

4.6.1 Demonstrations

In the tests conducted, the ego car drives on an ending lane and must take the gap between two vehicles. As we know, the driver could switch to its neighboring lane before or after the passing car (refer to Figure 4.11).

[11] Videos of the vehicle interior, the test area and the congress results can be found on YouTube: https://www.youtube.com/watch?v=8o3hT3H_gDU.

We will analyze examples of gap and no-gap situations below. These lane changes simultaneously include longitudinal and lateral spatial risks, which make them complex maneuvers.

Gap Example

Recordings have been replayed with three vehicles and, subsequently, we predict their trajectories. Figure 4.17 pictures the different situation snapshots with the ego velocity v_0 as well as distances d_1 and d_2 to the two other cars on the neighboring lane. Herein, the planned ego trajectory is colored green, with the predicted trajectories of surrounding vehicles colored red. On average, the other cars drive in the complete stream with constant velocity.[12]

A planned lane change is realized by blending the current path with the parallel path at a start time $s_{\text{start}} = 1\,\text{sec}$ and with a duration of 3 sec (for details, see Section 4.5.3). This aligns with usual lane change durations, e.g., according to [116]. For the visualizations, we choose a prediction horizon s_h of 6 sec but the actual risk is evaluated for $s_h = 12\,\text{sec}$. The car signals and possible paths are updated in the R-LDM on demand.

Figure 4.17 also illustrates the risk graph. The sampled ego trajectories are plotted as curves of velocity over the future time and the sum of collision and curve rates with τ_{crit}^{-1} are further visualized. The blue areas in this graph represent low probabilities between $[0\,\%/\text{sec}, 0.5\,\%/\text{sec}]$ and red areas are high values with $[0.5\,\%/\text{sec}, 1\,\%/\text{sec}]$ and also values $> 1\,\%/\text{sec}$. Finally, the single chosen trajectory is highlighted with green points. RM always tries to find a trajectory that bypasses the red hot spots. In this scenario, RM successfully judges the gap as sufficiently large. Changing the lane with moderate acceleration from 7 m/sec to $v_{\text{tar}} = 11\,\text{m/sec}$ presents the optimal maneuver. After the lane change, the message on the HMI changes from "go left" to "go straight" (i.e., the target path $\mathbf{p}_{\text{tar}}$). This example emphasizes the proactive support provided by RM.

No-Gap Example

In the gap scenario, we recommended to drive with constant velocity or to accelerate for performing the lane change. The front and back car are determining the gap. In this second scenario, the gap is too small to safely

[12] Note that the other trajectories' length also relates to a constant velocity, while the ego trajectories can incorporate deceleration or acceleration.

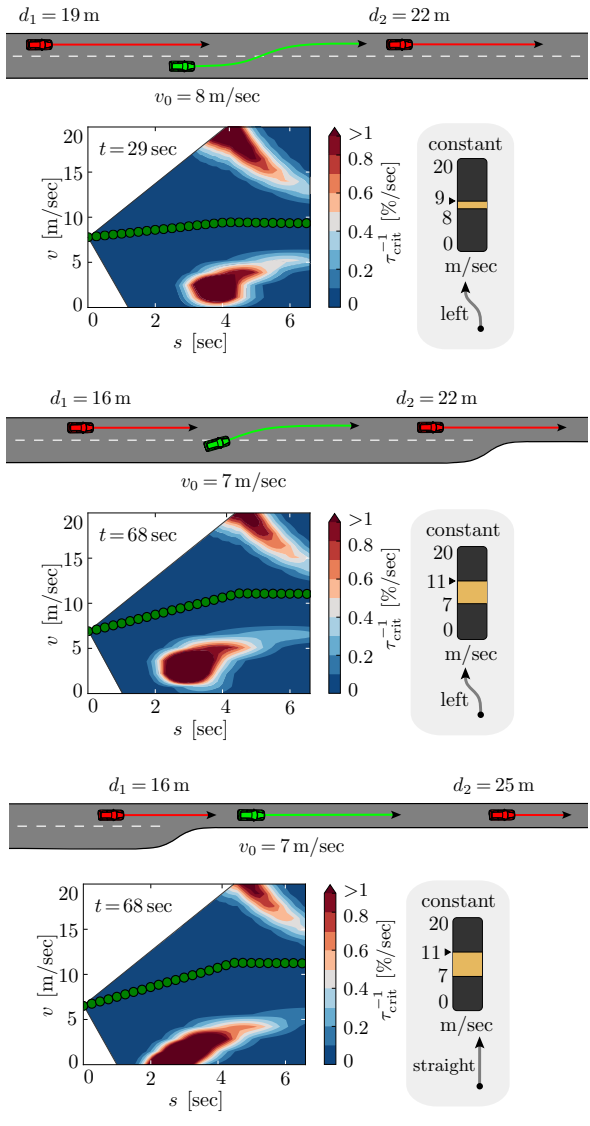

Figure 4.17: Application of RM and R-LDM in a gap scenario. Given are three snapshots from real-world recordings. The system signals to directly perform a lane change due to sufficient space between the neighboring vehicles. Top: Road layout and predictive situation. Bottom: HMI with risk visualizations. In the text, the reason and changes of the hot spots in the risk graph are explained.

conduct a lane change. Only after the two other vehicles have passed, RM signalizes that the driver could make a change.

Within Figure 4.18, the preceding car hinders the ego car to go on the target lane. Hereby, we want to analyze more closely the risk graph shown in the first scenario screenshot. If the ego vehicle brakes or drives constantly and makes the lane change, high risks are caused by the preceding vehicle. Due to the close lateral distance, a large risk spot is depicted in the bottom left of the graph. Secondly, if the ego car accelerates and changes lane, there is a high risk of colliding with the front vehicle, indicated by the other red area. The risk graph thus shows the reasoning behind the system's recommendation output: "brake" and "stay on the lane".

Actual driven speed v_0 and target speed v_{tar} are depicted in the velocity scale. As a reminder, the comfort velocity v_d determines the utility gain and can be chosen by the user. In Figure 4.18, the system penalizes velocities that deviate from $v_d = 10\,\text{m/sec}$. To find an ego trajectory in terms of risk versus comfort, we sanction high accelerations as well. RM, therefore, initially recommends to brake down to $4\,\text{m/sec}$ because of the ending lane. When the preceding vehicle has passed the ego vehicle, RM then correctly recommends to drive with constant velocity $6\,\text{m/sec}$ and v_{tar} is lastly accelerating until $9\,\text{m/sec}$. The ego driver's speed stays around $v_0 = 5\,\text{m/sec}$ and the lane change suggestion is followed.

Discussion on Results

As we could see, the combination of RM and R-LDM allowed for a reasonable support of the driver in lane change scenarios. Now, a short discussion concerning the results of the system are given. After showing insights into noise robustness with real-world sensors, the generalization for other driving situations is discussed.

Noise in the vehicles' position and velocity induced by the real camera and GNSS sensors can potentially hinder a faultless operation. In the system, we parametrized the weights in the planning (i.e., for risk, utility and comfort) to achieve robustness against such noise errors. As explained in Section 4.5.1, positions of all vehicles are projected to the closest path, derived from the R-LDM. With regard to risk computations, the noise did not affect the lane change advices of the experiments. Position noise that is constantly present is merely carried into RM in the future time, where the survival analysis compensates its influence.

However, the safe speed output from the planner can slightly fluctuate. This is due to the velocity noise in both the ego and other vehicles, which

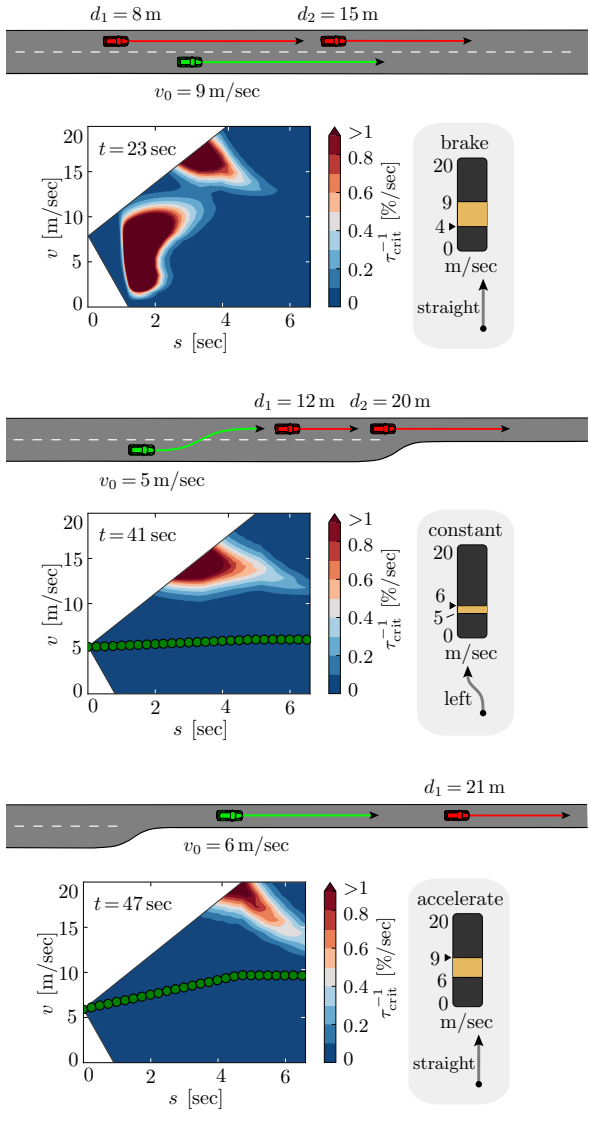

Figure 4.18: Behavior of RM and R-LDM in a no-gap scenario. Here, a forced lane change with acceleration is advised only after the preceding car passes. This represents a safe motion and the driver follows the warning. Note: The risk graphs always visualize the risks for the path with the lane change.

grows over the prediction time. Furthermore, a projection on wrong lanes induces the biggest potential errors. To tackle this noise propagation and impact of discrete errors, we introduce a hysteresis in RM. The hysteresis changes the outputs solely when the new selection shows lower risk values for at least 2 sec. The more RM is robust against sensor noise, the less it will proactively react on the changes in the environment. Lastly, a tradeoff was chosen between stable warnings versus better sensitivity.

To summarize, the given experiments showed that a map-based behavior support works online and has benefits in terms of its predictability and transparent risk model. The analyzed tests were conducted with velocities of 30 km/h. Nevertheless, other applications on Spanish city streets (i.e., 50 km/h limit) and German highway entrances (on average, 100 km/h) show the same functionalities and beneficial properties. Due to the generic characteristics of the models, the risk-based planner could be applied for lane change situations with varying velocities.

4.7 Conclusion

In this chapter, we learned about two novel driver warning systems, which make use of the risk models from this dissertation. In the first application, a risk navigation device in urban driving was developed. TTX indicators evaluate risks for a potential collision, sharp curve and rule violation, based on a local dynamic map that is queried for traffic objects.

The HMI included a renderer that constantly visualizes surrounding driving situations. Design elements then convey risks with spatio-temporal vehicle interactions for a proactive warning. Each of them has the same color coding, with green meaning safe to red meaning dangerous. In evaluations of real recordings, a successful functionality was shown when driving at intersections. Hereby, the used test vehicle was a high-cost platform with GNSS and lidar sensors.

In the second application, the real-time motion planner Risk Maps (RM) has been presented. It incorporates the survival analysis for risk prediction and plans velocities by efficiently sampling ramp curves.

Once the relevant cars are projected on stored map geometries, one can visualize risks for different behaviors (e.g., accelerating or decelerating). Additionally, RM was extended to recommend lane changes. In real tests, the warning functions of RM were shown for the purpose of lane change support. A live system presented the results in the frame of the VI-DAS project [125] on the ITS European Congress 2019. It could give detailed

advice about target speeds and lane changes, separated by gap and no-gap driving situations. Concerning the test car, a lower-cost GNSS and camera sensors was used, rather than expensive sensors.

To conclude, the research questions from this chapter were answered with the contributions of:

- **Situation Data**: Novel driver navigation device that warns of future risks with fused maps, other road information and sensors [91]
- **Risk Applications**: Real HMI application on long field tests with approaching, crossing and turning around urban intersections.
- **Prediction**: Novel modifications in RM for providing behavior recommendations when changing lanes. The planner has been integrated within a joint system from partners in the VI-DAS project.
- **Risk Applications**: Demonstration of proactive support for highways with three dynamically interacting vehicles, including complex lane changes.

For reproducibility and integrity of the results, Appendix 6.5, lists the most important parameters for all developed risk models, planners and warning systems. Compared to other sense-plan-act systems, they are considered to be more physically based. Appendix 6.6 also gives hardware descriptions for the test vehicles of this chapter.

This completes the technical chapters of this dissertation. In the subsequent and final chapter, the overall findings and an outlook for related future research topics are summarized.

5 Conclusion

The autonomous driving industry is in a transforming stage. The provided passive and active safety systems jumped from around six in 1995 (safety belt, electronic stability control, etc.) to more than 25 systems in 2020 (including adaptive cruise control and lane change support) [6]. Their stable performance often surpasses human capabilities in special sub-problems of driving. This can be, for example, observed in the time course of traffic fatalities [122], when they were introduced in each year [132].

As sales of assistance systems are steadily increasing, the demand seems to be high [17]. Furthermore, self-driving taxis were recently launched by Waymo as a hailing product without human safety drivers in Phoenix, USA in 2019 [41]. Huge advances in the perception sensors have enabled market penetration, especially, cost-efficient cameras are becoming powerful with on-device machine or deep learning. A wide realization of driving with computers in developed countries is coming close. Self-driving vehicles seem to be safer and more comfortable.

Nonetheless, there are still challenges left unsolved. The validation of autonomous vehicles requires billions of miles driven due to the lack of a general risk metric. Secondly, many opportunities are left untaken. The high computation power for embedded chips allow complex systems in the car that go beyond current simpler reactive driving in urban environments, which was not possible before. Here, please refer to the introductory Chapter 1 for details about the current state of self-driving cars.

Summary

In this dissertation, predictive and uncertainty-aware risk models were developed that further help to close the gap between the automotive market with the research community. It was shown that these models a) *are superior to common spatial or temporal indicators* in accident and critical situation classification for the validation of self-driving systems, b) *enable the generic planning of proactive velocities* for the own driving task and c) are desirable from a practical perspective, since they *are employed in real-time for warning* in complex scenarios.

The dissertation covered the aspects of theoretical contributions, of empirical experiments as well as of implementation and applications on hardware. This is considered as a substantial step for trust and motion planning in autonomous vehicles.

Individual Contributions

In Chapter 2, risk models were developed, which include an explicit spatio-temporal situation prediction with path and trajectory concepts. In total, the framework may incorporate driving uncertainties of position, time, velocity and possible behavior. With a novel proposed model, a higher performance, higher robustness and higher resolution could be finally achieved compared to state-of-the-art collision risks. The evaluations were based on samples of the GI-DAS and NGSIM dataset.

Chapter 3 introduced risk-based planning with optimization under velocity profiles. To minimize the cost function, curve risk and severity were added, and the system was allowed to increase utility (distance traveled) and comfort (low acceleration and jerk). By enabling multiple changes in the planning, generic and tactical solutions could be created. Concretely, the planner handled a variety of driving scenarios, namely, curve driving, longitudinal following and leading, intersection merging and crossing on intersections with stop lines and, most importantly, with car priorities. The simulations were based on SUMO, with an enhanced variation capability for the situations. In no cases were there car accidents and, where possible, the benefits were kept high. Changing weights in the costs allowed hereby to obtain different driver types.

In Chapter 4, the risk models and planners were implemented into real car prototypes. For this purpose, the framework has been adapted to deal with real-time signals from a map database and sensor platforms. The outputs from the models were then made accessible to drivers. Tests occurred in multiple locations: Offenbach am Main in Germany and Eindhoven in the Netherlands. On this note, both high-accurate, expensive GNSS and lidar as well as low-cost GNSS and pure camera approaches could be employed. Ultimately, the signals successfully visualized were, e.g., distances to regulatory risks from crosswalks, target safe velocities in sharp curves or path possibilities for cars encountered. In another application, lane change motions were also recommended. The comparison of the system's plan and the human behavior was essential for driver warning.

In conclusion, the proposed models are white-box models with differential equations and statistical variables. The model parameters were jointly

fitted to encompass realistic variations or uncertainties in driving. The variety of successful applications of the data in GI-DAS, NGSIM, SUMO and the car platforms showed that human behavior is well modeled by the separation of risk, utility and comfort, and the same prediction-reaction process might be implicitly carried out when driving.

This provides many benefits, because the parametrization follows physical properties and is done intuitively (e.g., with the positional uncertainty for measured sensor inaccuracies of detected objects). The prediction and warning components can be seamlessly integrated into the common sense-plan-act cycle of automation. If the system is not applied in planning, it additionally helps to rate other existing planning systems. The risk methods offer the possibility to understand why the system behaved erroneously or in a certain manner.

5.1 Further Outlook

Predictive driving and system validation advance the goals for personalized self-driving cars. Before we can provide them for every corner in the world, however, many challenges remain. Every country has its own road, weather and traffic conditions. But more importantly, in special scenarios, an ego-centric plan cannot solve the driving situation.[1] Instead, only if the participants drive altruistically and in a cooperative way, can the vehicles move forward. For this, a continuously adaptive agent is required.

Cooperative Driving

The remaining open research questions are illustrated with Figure 5.1. On the top left, an intersection with dense and heterogeneous traffic is given. Pedestrians, bicycles and cars are mixed together and are crossing on a small intersection. On the top right, we have a narrow passage with parked vehicles, constraining the driving space so that overtaking is necessary to move forward. The bottom picture shows a scenario with a traffic jam on the highway, where cars are trying to merge from the highway ramp. Due to the creativity and flexibility of the human mind, we solve these driving problems by adapting and integrating ourselves into the traffic flow, but the obstacles we have to manage for a working technical system are numerous and challenging.

[1] Remark: we will use the words situation, scenario and scene interchangeably to describe a traffic constellation.

Figure 5.1: Different types of complex scenarios, where human communication and cooperation is necessary to move forward. Images are not mine and, from left to right, were taken from [98] (Tuomas A. Lehtinen/Moment via Getty Images), [143] (Cruise) and [140] (iStock.com/kozmoat98).

In the following, several of the theoretical features from these cooperative situations with large shared driving spaces between the vehicles are described:

- Firstly, the usual options for an action are limited. Drivers are forced to have close vehicle encounters with slow speeds. This makes the situation, in turn, riskier in terms of probability, but less so in terms of the severity.
- For the same reason, no safe distances are and can be kept. Map lanes are not necessarily applicable anymore, because of, e.g., two-wheelers and stopped cars, which block the centerline of the road.
- Also, occlusion issues exist due to static, quasi-static or dynamic vehicles. The environment becomes partially observable and prediction capabilites are reduced in the planning. Probabilities of what will happen must be assumed near occlusions.
- There is no rule who drives first, because there is always another car that has priority. We must drive, e.g., boldly to merge into the traffic. Over time, cultural norms become valid, which are not written in law.

- Then, explicit and implicit communication becomes very important between human drivers. Information about the status and intent is given, either through our behavior, with hand signs, honking or another sound. This gives a different dimensionality to driving, with human science in the focus.
- In some situations, the deadlock cannot be resolved, because there was a misunderstanding and the cars now block each other's way. Nobody can drive on and we need to drive backwards and make space for the other car. It might be beneficial, if both follow the shared goal and at times, it could also be better, if only one car drives altruistically.
- If there are slower vehicles, such as bicycles, and vehicles that go fast, such as cars, we always might get to a point where overtaking is happening. Overtaking is a dangerous encounter, with oncoming cars on the other side and cars driving in the same direction. Here, we take higher speeds than usual to quickly manage the maneuver.
- Based on this, quite unexpected behaviors can come from other cars. They might drive on the sidewalk to let you pass or even overtake multiple cars in front, because the follower of the slow vehicle does not overtake over a long period of time. The path alternatives can be much more numerous and potentially in violation of a rule.
- To round off, the interaction is not between a maximum of three or four vehicles but can be between many more with different severity considerations (pedestrian-truck collision, etc.). The scene development and combinations of trajectories become very large and need to be handled.

The probabilistic risk models of this dissertation are a stepping stone towards generalized, transparent and intelligent driving (see Chapter 2). Particularly, the priority and change of prediction models in the planning process (see Chapter 3) tackle some of the aspects in cooperative problems. Likewise, the real-time probing techniques for path and velocity planning are helpful (see Chapter 4). Equipped with this framework, the agent will drive as safely as possible and likely merge into, pass or cross the other vehicles successfully. If other participants obey the rules and drive inside the lanes, we will even be able to plan proactively.

However, if they drive according to their own norms, a short-term reaction is the only possibility. We would not leverage the complete information of the traffic situation. In this sense, the risk-based planner can only optimize and improve what is modeled in the analytic cost function. The

capabilities of situation recognition and their iteration, shared spaces or conflicting goals as well as the human communication part would have to be extended further.

How can we enable the computer to find solutions for such a complex and large state-space, with the extreme influences of other entities on the own plan? On this note, the main research topics will be explained in the upcoming, last section.

Research Topics

It is important to recognize that the balance between **elaborating all the intentions** of the other cars and **only filtering the relevant** ones is important. Generally, motion can be divided into lateral path choices (driving straight and turning, overtaking a car in front or even driving backwards to park) and possible longitudinal velocity choices (from slight to strong braking and accelerating). In turn, both are executed in shared spaces for signaling which entity can go ahead first. We even signalize that we saw each other and will interact in the future. Usually, a classification algorithm, which is, for instance, learning-based [62], ranks these situations in terms of the likelihood that they occur.

To reduce the sheer complexity, one has to thus consider how to filter the critical participants based on their relevance to the ego driver and, also, to filter situations which have both a low risk and a low likelihood. Since humans solve deadlock situations with personal preferences and many past experiences and conventions, we might not be able to reach the same level of understanding in machines in short-term.

The second research problem involves the **incorporation of intention knowledge in the planning** cycle to improve navigation through interaction-intensive scenarios. Specifically, situation probabilities allow us to decide on the current optimal first-second relationship. When the agents drive in compliance with the prediction, we respond with a correct behavior. Once other cars change their behavior, the recognition model ranks another intention higher and again we react safely. An ideal system detects intentions in real-time and thus in a fast fashion.

In addition, considering the risks, utilities and comfort of the other cars in the ego cost function is an interesting new planning approach for cooperation. With the inference of this other view, we can minimize the complete traffic costs and explicitly decide whether to be friendly or drive ego-centered and drive first, taking the priority. The prediction confidence is here useful to change driving styles. High confidence relates to bold

driving, while a conservative approach is beneficial for low confidence. In this way, we can send the right signal back to other vehicles.

We need to combine a continuous **velocity planner with path optimization** for crowded traffic, i.e., choosing the lateral offsets along the map path. In this context, lateral path options have a higher impact for finding smart solutions. On the one hand, investigations of the dependency between the longitudinal and lateral plans are interesting. For example, planned overtaking with a sharp curve may require stronger braking. On the other hand, one might couple global search algorithms with low-level optimization. Especially for a path planner, we have to find discrete maneuvers, e.g., whether we overtake on the left or right. This maneuver is then improved by continuously fine-tuning the shape.

The lateral uncertainty in the risk plays a dominant role and for this risk model, Gaussians taking into account the shape of the real rectangular vehicle dimensions are required. At close proximities, the assumption of a point mass with an ellipse form is invalid. We saw, in this dissertation, the application of warning for driver support. For eventually transforming the working plan into real actuation, a longitudinal plus lateral controller with specific vehicle dynamics models needs to be implemented as well.

To conclude, all three areas of future work represent possible extensions for the framework in multi-party scenarios. With the generic, probabilistic and predictive risk models for handling complex driving, cooperative scenarios will also be easier to handle. The models provide some benefits compared to simpler reactive logics and other data-driven approaches, because they contain detailed information about the situation understanding and reasoning for motion planning.

I therefore hope that many ideas and findings of this dissertation can be reused to come a step closer of achieving the goal of zero crashes in all driving conditions with fewer traffic jams and comfortable travel.

6 Appendices

6.1 Datasets

Figure 6.1: Used datasets for risk prediction in Chapter 2. *Left*: German In-Depth Accident Database (i.e., GI-DAS), found in Section 2.3. The trajectories of all participants in a traffic situation, which has led to an accident, are hard to measure. GI-DAS tackled this problem by estimating ex-post trajectories from, e.g., braking marks and impact velocities. It was initiated in the year 1999 and contains, amongst others, the position and velocity streams of vehicles involved in accidents around Hannover and Dresden, Germany. There are longitudinal following crashes and different intersection crashes that amount to many examples. In the experiments, 14 cases were taken with 28 own variations to collect realistic near-crash and no-crash situations. *Right*: Next Generation Simulation (NGSIM), found in Section 2.6. In NGSIM, passing vehicles are detected with cameras that are mounted on a high building (see image). In the experiments, recordings from the Lankershim Boulevard have been used and two 15-minute streams (8:30 am to 8:45 am and 8:45 am to 9:00 am) analyzed, from an excerpt of 500 meters along the street in the Universal City neighborhood around Los Angeles, USA. They represent primarily congested conditions and are suited for microscopic driver behavior analysis. Note: The shown images are not mine and were taken from [126] (Volvo) and [121] (US Department of Transportation).

6.2 Criticality Maps

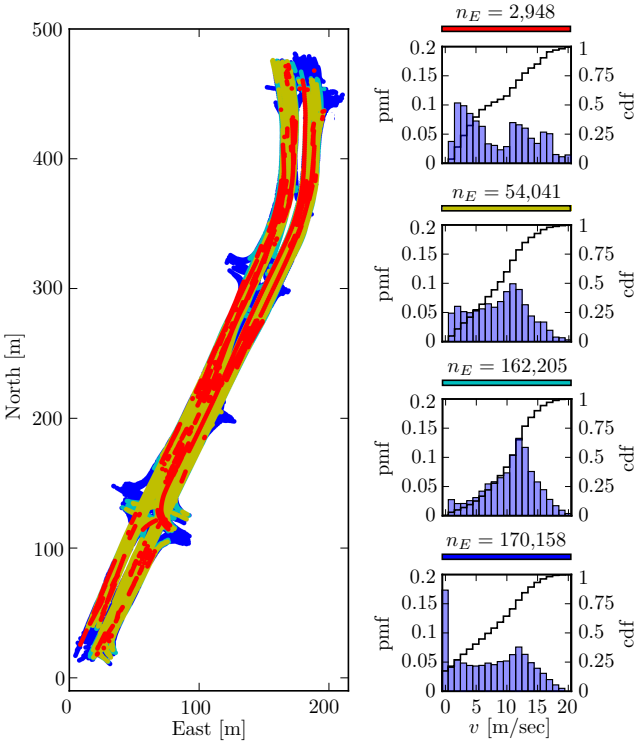

Figure 6.2: This plot shows the results of the hazard classification task from Section 2.6 for a mixture of Time Headway (TH) and Time-To-Collision (TTC). *Left*: Criticality map. *Right*: Velocity histograms. We add the sorted TH events with the TTC events for each hazard bin, which are colored as follows: dangerous in red, offensive in yellow, uncomfortable in cyan and noticeable in blue. One can see that the number of events n_E in the bins decrease with lower risk levels, similar to Heinrich's law [50]. The red bin contains about 3,000 events and the blue bin over 170,000 events. With this new time-based measure, we can detect more critical situations but it is also oversensitive to risk. Red risk spots are highlighted nearly in the complete traffic study area (around the bottom intersection and the long curvy segment on the top right). It is concluded that, with heuristic time metrics, the risk spots cannot be as localized as shown with the proposed novel risk model.

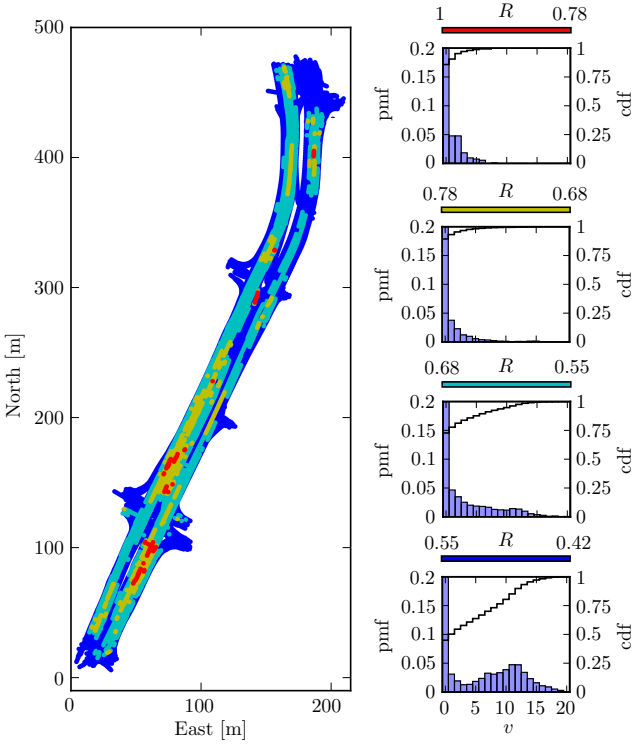

Figure 6.3: Results of the Risk Spot Detector (RSD) without uncertainty extensions. *Left*: Criticality map. *Right*: Velocity histograms. In this dissertation, the survival analysis was improved with extensions of 2D Gaussians, of uncertainty growth with velocity distributions and the Path-following Mixture Model (PMM). To see the achieved performance difference, in this figure, the hazard classification is shown without the extensions by merely employing 1D Gaussians in approximately the size of the car (i.e., 4 meters) for all predicted times. RSD cannot detect the three hazard categories anymore. Only the traffic density is propagated in all velocity bins (large bins in velocities around 0). Having low distances with high velocities and approaching a standing car with moderate velocity are not included. This parametrization reflects risks without the potential of critical other behaviors. With the proposed uncertainty modeling in Chapter 2, RSD therefore detects potential risks more realistically and proactively.

6.3 Pareto Front

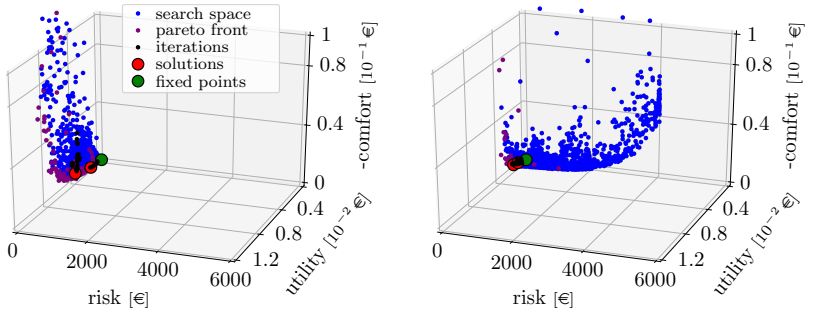

Figure 6.4: To understand the cost landscapes of the planner in Chapter 3, the quantitative ranges for risk, utility and comfort are given for different scenarios. These include situations with a single ego car, longitudinal situations with one or two other cars and intersection situations with vehicle's crossing or merging. One plot shows the costs for possible ego velocity profiles at one point in time in the simulation. The target is to see relative differences between the cost aspects and emphasize the possibility of having different solutions for the optimizer. For instance, risk is neglectable on a straight road with a single ego car (compare left plot). Here, we can, either accelerate (i.e., higher utility, but higher comfort costs) or stand still (i.e., utility and comfort are both 0). The pareto front then shows the points in between that are minimal for at least one objective. We can nudge the optimizer to end at a single point of the pareto front by setting weights of the objectives. In effect, this allows us to imitate different driver types, e.g., sportive when utility-focused or conservative when comfort-focused. As a reference, the costs are visualized from a conservative planner (iterations and final solutions) as well as from the fixed trajectories. The same parametrization was used in Section 3.6. *Left*: Free straight driving of ego car without other obstacles. *Right*: Free ego driving on a single upcoming sharp curve.

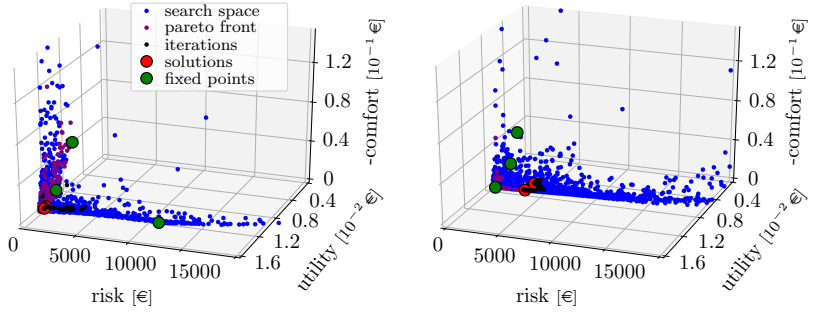

Figure 6.5: *Left*: Longitudinal following of car pair. *Right*: Following a front car and being followed by back car. When we drive behind another leading car, the behavior options are either to drive with the same velocity (i.e., low risk and low comfort costs) or to brake (i.e., lower risk with higher comfort costs). Moreover, with another car in the back, the ego braking behavior is defined by the back car. Note the different scales for risk, utility and comfort in comparison to Figure 6.4 and Figure 6.6. As already mentioned, the costs are given in euro and since the risk includes severity, the risk values are larger than one.

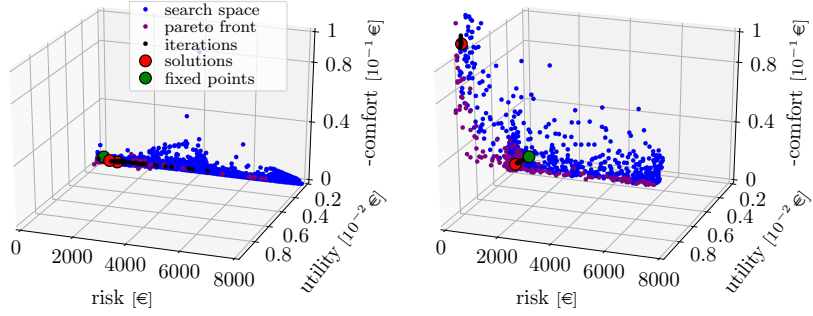

Figure 6.6: *Left*: Ego car is merging between two other cars at a T-junction. *Right*: Two cars crossing at an intersection. The simulations were done with the COBYLA optimizer [89] using two snake velocity profiles. For visualizing the action space, we also sample random velocity points from the snake profiles (blue points). Black points represent iterations and red dots the final solutions. The green dots are the fixed trajectories (i.e., constant velocity, smooth and hard braking). Finally, the points of the pareto front are highlighted in purple.

6.4 Optimizer Benchmark

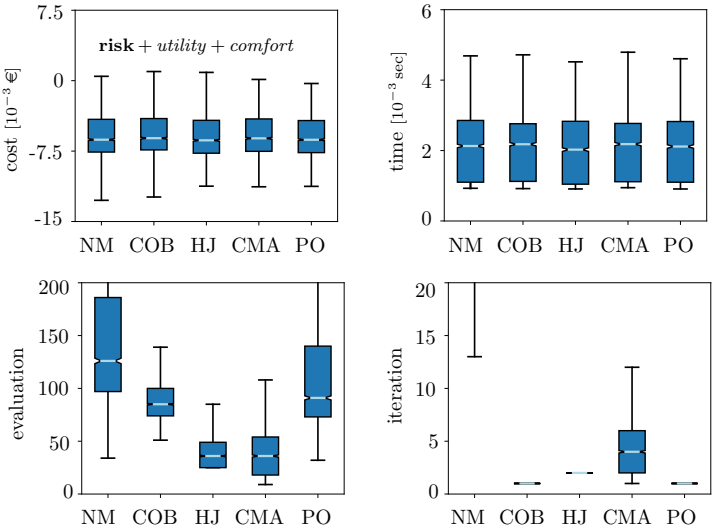

Figure 6.7: In this experiment, we explore more the cost landscape (risk versus utility and comfort) from Chapter 3 in terms of relations to the used optimizers. For a variety of starting velocities in curve driving and in longitudinal following, the median costs over all timesteps are analyzed. An optimization method computes the cost function in cycles and in one of those iterations, several function evaluations are performed. For example, the Nelder-Mead method (NM) does a function call for each corner point of a geometric simplex and then warps the object according to the found values. This is one iteration and is repeated. Besides the cost value and the number of evaluations and iterations, the plots show the times for one evaluation. The explored optimizers are listed in Table 6.1.

Table 6.1: Definition and sources of the selected optimizers.

Shortcut	Optimizer
NM	Nelder-Mead method [78]
COB	Constrained Optimization by linear approximation [89]
HJ	Hooke-Jeeves algorithm [51]
CMA	Covariance Matrix Adaptation evolution strategy [45]
PO	Powell's conjugate direction method [88]

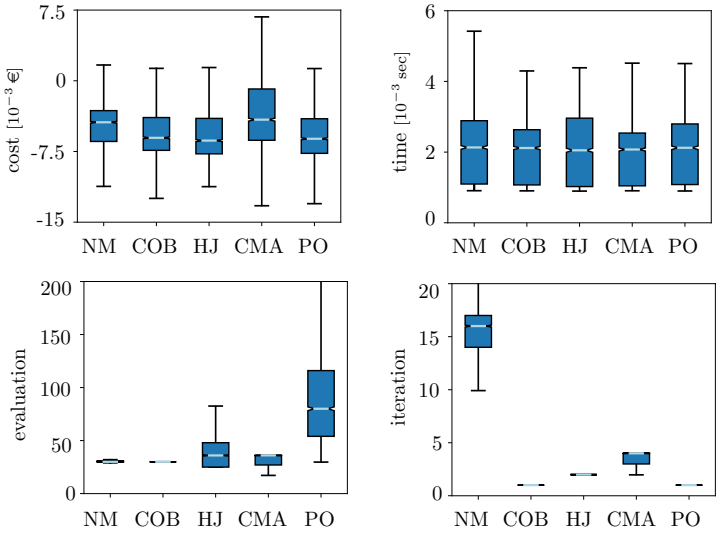

Figure 6.8: In comparison to Figure 6.7 with no constraint on evaluation numbers, we now set a maximum of 30 evaluations ($e_{\max} = 30$). The optimizers do not have the full time to tune the solution, and the costs are slightly higher. In general, evaluation numbers reduce to circa 30 for most of the optimizers. Only Powell's method (PO) has still more than 80 evaluations. This is due to its iteration strategy. PO is always doing a single iteration and has to finish this cycle with the heuristic constructions. Otherwise, the solution would be erroneous. The function time stays at 2 milliseconds for every optimizing method. Here, the Constrained Optimization (COB) is superior to the other methods, because it always keeps the maximum of 30. COB is especially suitable for real-time applications. Table 6.2 shows the different simulated longitudinal scenarios.

Table 6.2: Longitudinal driving scenarios in optimization tests.

Scenario	Description
Curve driving	In the simulation, the ego car drives through a $< 90\,^\circ$ curve with a desired velocity of 15 m/sec.
Vehicle following	Starting velocities of 1 m/sec, 5 m/sec, 10 m/sec and 20 m/sec. Additionally, the initial distances are changed so that the headway is < 2 sec and the ego car needs to brake.
Following and followed	Further back car that induces risks and the cars drive with 8 m/sec.

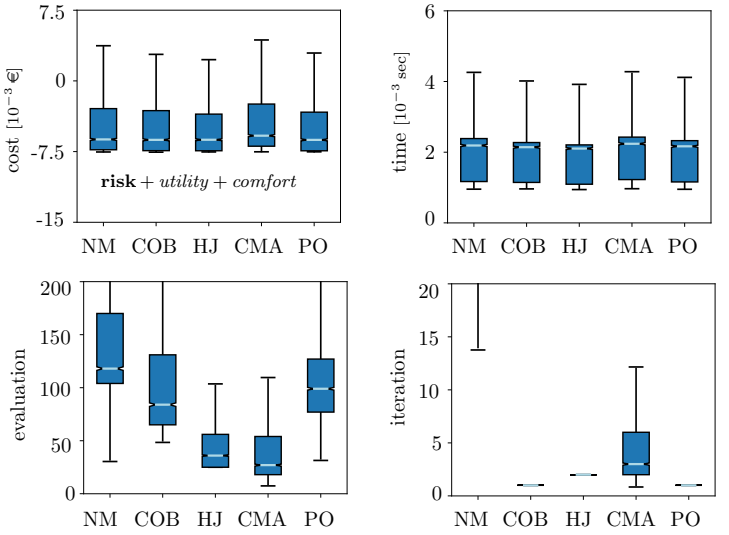

Figure 6.9: The next evaluation considers various intersection scenarios. These include gap taking scenarios and two cars crossing each other with different priority settings (see Table 6.3). Similar to the longitudinal cases from Figure 6.7, with no maximal evaluation set, Hooke-Jeeves (HJ) and Covariance Matrix Adaptation (CMA) have the least evaluations over all runs. All optimizers have enough time to fully find the minimal costs and the single function evaluation time is the same. Note that in the costs, the risks have the highest value range (1 € until several 10,000 €). Utility and comfort aspects lie in 10^{-1} and 10^{-2} intervals. The optimizers will thus quickly minimize the risk and then take time to finetune maximal utility and minimal comfort in the end.

Table 6.3: Lateral intersection scenarios in optimization tests.

Scenario	Description
Gap taking	The ego vehicle is standing in front of a stop line and merges between two other cars with velocities of 8 m/sec.
Crossing and no priority	Two cars are approaching a junction and the starting velocity is varied around 10 m/sec (i.e., driving) and 0 m/sec (i.e., standing). The ego car brakes, but also accelerates, if safe.
Crossing with priority	Again, the cars are driving over the intersection and we vary, whether they are dynamic or stopping in the beginning. The ego car mostly accelerates to cross.

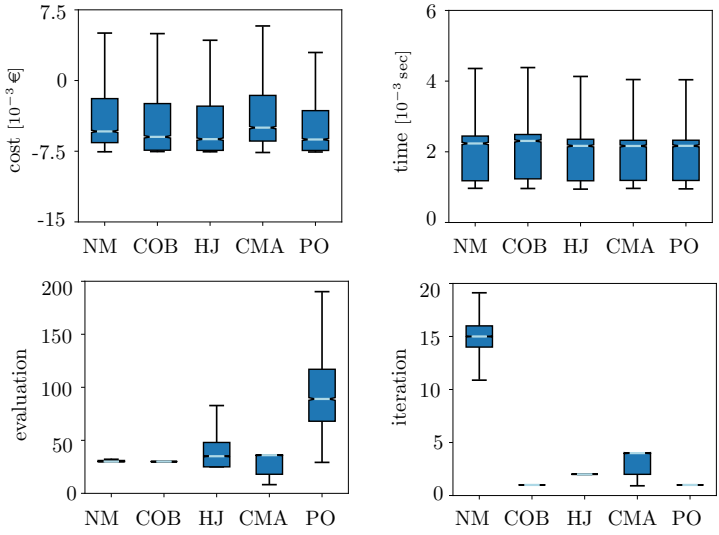

Figure 6.10: Intersection scenarios with maximal evaluation of 30. Here, as in the longitudinal scenarios of Figure 6.8, the optimizers have minor differences in the final cost values. The optimizer COB but also the NM and CMA method have small evaluation numbers around 30. However, COB outperforms the latter two, because it always keeps a fixed evaluation limit of 30. Iteration strategies differ for each method: NM has the highest iteration rate, followed by CMA, HJ and finally PO and COB. In all scenarios, the same risk model was used and this shows that the optimizers can handle different scenarios and keep the costs marginally small. Further optimizers, such as gradient-based optimizers numerically approximating the gradients, can be applied and compared as well. Note: The boxplot shows the median using light blue strokes, lower and higher quartile using the darker blue boxes and the minimum and maximum value using black whiskers.

6.5 Parametrization

Table 6.4: In this section, the main parameters are summarized for the risk models, planning and warning applications from Chapter 2, Chapter 3 and Chapter 4. In comparison to most other systems in automated driving, the end system has only 50 parameters that can be categorized into: uncertainty of risks (i.e., with 7 entries), severity of risks (having 5 entries), assumptions for prediction of other vehicles (with 9 shares), ego planning parameters (also 9 shares), driver type weights (4 items), vehicle dynamics (with 6 items), constants in the local dynamic map (containing 5 entries) and general simulation (again 5 entries). It should be stressed at this point that all constants in the models have physical properties with units m, sec and €. This allows their meaning and implications to be intuitively understood. Particularly, for vehicle uncertainty and predictions, we have a powerful tool to define behaviors and interactions in traffic scenarios. Most parameters are defined in Chapter 3, whereby the uncertainty parameters are introduced in Chapter 2. Chapter 4 gives the parameters for the R-LDM that allows the system to warn the human driver in real-world driving. The simulation parameters are set the same in every experiment.

Category	Variable & Value
Uncertainty	$\sigma_{0,\text{lon}} = 0.75\,\text{m}$, $\sigma_{0,\text{lat}} = 0.3\,\text{m}$, $c_i = 0.1$, $m_f = 1.5$, $N_p = 11$, $\sigma_{0,\text{curv}} = 0.15\,\text{m/sec}^2$ and $\tau_0^{-1} = 0.4\,1/\text{sec}$
Severity	$D_0 = 90\,€$, $m_i = 1000\,\text{kg}$, $D_{\text{max,curv}} = 10000\,€$, $k_{\text{curv}} = 0.7\,\text{sec/m}$ and $\beta_{\text{curv}} = 7\,\text{m/sec}$
Prediction others	$s_h = 12\,\text{sec}$, $k_{\text{lon}} = 5\,1/\text{sec}$, $s_{\text{lon}} = 1.5\,\text{sec}$, $k_{\text{lat}} = 3\,1/\text{sec}$, $s_{\text{lat}} = 2\,\text{sec}$, $s_a = 2.5\,\text{sec}$, $a_a = 3\,\text{m/sec}^2$, $s_b = 4\,\text{sec}$ and $a_b = -1.5\,\text{m/sec}^2$
Ego planning	$s_l = 2.5\,\text{sec}$, $e_{\max} = 30$, $m = 2$, $N_t = 21$, e.g., $s_v = 4\,\text{sec}$, $v_v = 10\,\text{m/sec}$, $s_s = 2\,\text{sec}$ for $v_0 = 14\,\text{m/sec}$, $t_{\text{start}} = 1\,\text{sec}$ and $l_c = 10\,\text{sec}^2/\text{m}$
Driver type	$b^t = 0.3\,€/\text{km}$, $b^d = 1.5\,€/\text{km}$, $b^c = 0.02\,€/(\text{km}\cdot\text{sec})$ and $b^j = 0.05\,€/(\text{km}\cdot\text{sec}^2)$
Vehicle dynamics	$a_{\min} = -7\,\text{m/sec}^2$, $a_{\max} = 3\,\text{m/sec}^2$, $\lambda_b = 0.4\,\text{sec}$, $\lambda_e = 0.8\,\text{sec}$, $a_{y,\max} = 7\,\text{m/sec}^2$, $v_{\max} = 25\,\text{m/sec}$ and, e.g., $v_d = 10\,\text{m/sec}$
Dynamic map	$\kappa_{\text{th}} = 0.001\,1/\text{m}$, $a_c = 0.5\,\text{m/sec}^2$, $t_r = 1.5\,\text{sec}$, $\Delta l_h = 100\,\text{m}$ and $d_{\text{proj}} = 5\,\text{m}$
Simulation	$\Delta t = 0.05\,\text{sec}$, $\Delta s = 0.05\,\text{sec}$, $w_{\text{lane}} = 4.5\,\text{m}$, $l_{\text{car}} = 4.5\,\text{m}$ and $w_{\text{car}} = 1.75\,\text{m}$.[13]

[13] Hereby, w_{lane} is the lane width, l_{car} the car length and w_{car} the car width.

6.6 Real Prototype

Figure 6.11: For the risk models in the first application of Chapter 4, compare Section 4.4, the test car named Honda ADAS Research Platform (HARP) was leveraged. HARP is based on a modified CR-V car model, and equipped with an OXTS localization and four lidar devices. Particularly, the Ibeo Lux lidars are built into the car body and also CAN data can be directly retrieved (ego speed, steering angle, etc.). Furthermore, a camera was utilized that is installed behind the windshield. To eventually obtain recordings from the ego perspective, the sensor data is read online with the middleware RTMaps on a StreamX machine. We required the position, velocity and angle signals of the ego and all other vehicles in the experiments. In comparison to the Carlota platform of the next Figure 6.12, HARP represents a more accurate, but also more costly, automated driving vehicle. Both prototypes were altogether shown to be applicable for the map-aware and probabilistic risk planners. This demonstrates the versatility of the approaches. *Left*: Image of HARP, showing the frontal view of the car. *Right*: Measurement units inside the trunk of the car. In the picture, connections of the sensors and computer are presented.

Figure 6.12: *Top*: Carlota test vehicle, employed for the second application in Section 4.6 of Chapter 4. Carlota was selected as a demonstrator in the European project VI-DAS that lasted from September 2016 until November 2019. For the purposes of real-time risk warning, besides sensor hardware, we required an intuitive Human-Machine Interface (HMI) setup. The HMI consisted of LED stripes that blink in specific patterns, and an instrument cluster and Head-Up Display. With the LED, the path choice (left, right or straight) have been shown and the displays visualized, amongst others, the current and target velocity. On the left, Carlota is pictured from the side. The right image shows the Sekonix cameras on a rack on top of its roof and the antenna from the u-blox GPS. *Bottom*: Final test track at ITS European Congress 2019. During the congress in June 2019, the road at the Automotive Campus was closed in Helmond, the Netherlands. A large projector showed the sensor and risk outputs so that the public audience was able to see Carlota and the algorithms live during lane changes from a booth at the roadside.

Figure 6.13: This work has been partially supported by the European Union's Horizon 2020 project VI-DAS, under the grant agreement number 690772.

7 Publications

Conference List

- T. Puphal, B. Flade, D. De Geus and J. Eggert. Proactive Risk Navigation System for Real-World Urban Intersections. In *IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2020.
- T. Puphal, M. Probst, M. Komuro, Y. Li, and J. Eggert. Comfortable Priority Handling with Predictive Velocity Optimization for Intersection Crossings. In *IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2019.
- T. Puphal, M. Probst, Y. Li, Y. Sakamoto, and J. Eggert. Optimization of Velocity Ramps with Survival Analysis for Intersection Merge-Ins. In *IEEE Intelligent Vehicles Symposium (IV)*, 2018.
- J. Eggert and T. Puphal. Continuous Risk Measures for ADAS and AD. in *JSAE Symposium on Future Active Safety Technology toward Zero Accidents (FAST-zero)*, 2017.
- J. Eggert, D. Aguirre Salazar, T. Puphal, and B. Flade. Relational Local Dynamic Maps for Driving Situation Analysis. In *JSAE Symposium on Future Active Safety Technology toward Zero Accidents (FAST-zero)*, 2017.
- F. Damerow, T. Puphal, Y. Li and J. Eggert. Risk-based Driver Assistance for Approaching Intersections of Limited Visibility. In *IEEE International Conference on Vehicular Electronics and Safety (ICVES)*, 2017.

Journals

- T. Puphal, M. Probst, and J. Eggert. Probabilistic Uncertainty-Aware Risk Spot Detector for Naturalistic Driving. *IEEE Transactions on Intelligent Vehicles (TIV)*, 2019.

- T. Puphal, B. Flade, M. Probst, V. Willert, J. Adamy and J. Eggert. Online and Predictive Warning System for Forced Lane Changes using Risk Maps. *IEEE Transactions on Intelligent Vehicles (TIV)*, submitted, 2021.
- J. Eggert and T. Puphal. Continuous Risk Measures for Driving Support. *JSAE International Journal of Automotive Engineering (IJAE)*, 2018.
- F. Damerow, T. Puphal, B. Flade, Y. Li, and J. Eggert. Intersection Warning System for Occlusion Risks using Local Dynamic Map. *IEEE Intelligent Transportation Systems Magazine (ITSM)*, 2018.

Patent Filing

- M. Probst, T. Puphal, and J. Eggert. Method for Assisting a Driver, Driver Assistance System, and Vehicle Including such Driver Assistance System. *EP19152513*, 2019.
- J. Eggert, T. Puphal, T. Sugino and F. Damerow. Risk-based Driver Assistance for Approaching Intersections of Limited Visibility. *EP18156467.5A. US15/895,325. JP2018023739A*, 2017.
- J. Eggert, T. Puphal, B. Flade and C. Wang. Warning System to Visualize Personal Risk Spaces for Driving Support. *EP21166216*, submitted, 2021.

Bibliography

[1] Y. Akagi and P. Raksincharoensak. Stochastic Driver Speed Control Behavior Modeling in Urban Intersections using Risk Potential-Based Motion Planning. In *Intelligent Vehicles Symposium*, 2015.

[2] B. L. Allen, B. T. Shin, and P. J. Cooper. Analysis of Traffic Conflicts and Collisions. In *Transportation Research Record*, 1978.

[3] P. Alves, J. Goncalves, R. J. F. Rossetti, E. Oliviera, and C. Olaverri-Moreal. Forward Collision Warning Systems Using Head-Up Displays. In *Intelligent Vehicles Symposium*, 2013.

[4] O. Bagdadi and A. Varhelyi. Jerky Driving – An Indicator of Accident Proneness. *Accident Analysis and Prevention*, 2009.

[5] M. Behrisch, L. Bieker-Walz, and et al. Sumo (Simulation of Urban MObility) - An Overview. In *Conference on Advances in System Simulation*, 2011.

[6] K. Bengler, K. Dietmayer, Färber, M. Maurer, C. Stiller, and H. Winner. Three Decades of Driver Assistance Systems - Review and Future Perspectives. *Intelligent Transportation Systems Magazine*, 2014.

[7] M. Bojarski, D. Del Testa, D. Dworakowski, B. Firner, and et al. End to End Learning for Self-Driving Cars. In *Conference on Computer Vision and Pattern Recognition*, 2016.

[8] M. Brännström, J. Sjöberg, and E. Coelingh. A Situation and Threat Assessment Algorithm for a Rear-End Collision Avoidance system. In *Intelligent Vehicles Symposium*, 2008.

[9] M. Brännström, J. Sjöberg, L. Helgesson, and C. Mikael. A Real-time Implementation of an Intersection Collision Avoidance System. In *World Congress for Federation of Automatic Control*, 2011.

[10] M. Buehler, K. Iagnemma, and S. Singh. The DARPA Grand Challenge - The Great Robot Race. *Springer Tracts in Advanced Robotics*, 2007.

[11] M. Buehler, K. Iagnemma, and S. Singh. The DARPA Urban Challenge - Automous Vehicles in City Traffic. *Springer Tracts in Advanced Robotics*, 2009.

[12] Z. Buyue, V. Appia, and et al. A Surround View Camera Solution for Embedded Systems. In *Conference on Computer Vision and Pattern Recognition*, 2014.

[13] P. Chang and C. Mertz. Monte Carlo Sampling Based Imminent Collision Detection Algorithm. In *International Conference on Transportation Information and Safety*, 2017.

[14] W. Chang, W. Hwang, and Y. G. Ji. Haptic Seat Interfaces for Driver Information and Warning Systems. *International Journal of Human-Computer Interaction*, 2011.

[15] C. Chen and et al. Combining Task and Motion Planning for Intersection Assistance. In *Intelligent Vehicles Symposium*, 2016.

[16] Y. Chen, J. Yang, and D. Otte. Load and Impact Conditions for Head Injuries in Car-to-Pedestrian and Car-to-Cyclist Accidents. In *Expert Symposium on Accident Research*, 2010.

[17] S. Choi, F. Thalmayr, D. Wee, and F. Weig. Advanced Driver-Assistance Systems: Challenges and Opportunities Ahead. *McKinsey & Company*, 2016. https://www.mckinsey.com.

[18] T. Christopher. Analysis of Dynamic Scenes: Application to Driving Assistance. *Institut National Polytechnique de Grenoble*, 2009.

[19] F. Damerow and J. Eggert. Predictive Risk Maps. In *Intelligent Transportation Systems Conference*, 2014.

[20] F. Damerow and J. Eggert. Balancing Risk against Utility: Behavior Planning using Predictive Risk Maps. In *Intelligent Vehicles Symposium*, 2015.

[21] F. Damerow and J. Eggert. Risk-Aversive Behavior Planning under Multiple Situations with Uncertainty. In *Intelligent Transportation Systems Conference*, 2015.

[22] A. Dettmann and A. C. Bullinger. Spatially Distributed Visual, Auditory and Multimodal Warning Signals - A Comparison. In *Human Factors and Ergonomics Society Europe Conference*, 2016.

[23] M. Di Francesco and et al. Rigorous Derivation of Nonlinear Scalar Conversation Laws from Follow-the-Leader Type Models via Many Particle Limit. *Archive for Rational Mechanics and Analysis*, 2015.

[24] F. Diehl, T. Brunner, M. Truong Le, and A. Knoll. Graph Neural Networks for Modelling Traffic Participant Interaction. In *Intelligent Vehicles Symposium*, 2019.

[25] S. Edelstein. Audi Gives Up on Level 3 Autonomous Driver-Assist System in A8. 2020. https://www.motorauthority.com/news.

[26] J. Eggert. Predictive Risk Estimation for Intelligent ADAS Functions. In *Intelligent Transportation Systems Conference*, 2014.

[27] J. Eggert. Risk Estimation for Driving Support and Behavior Planning in Intelligent Vehicles. *AT - Automation Technology*, 2018.

[28] J. Eggert, F. Damerow, and S. Klingelschmitt. The Foresighted Driver Model. In *Intelligent Vehicles Symposium*, 2015.

[29] J. Eggert and T. Puphal. Continuous Risk Measures for Driving Support. In *JSAE International Journal of Automotive Engineering*, 2018.

[30] J. Eggert, D. A. Salazar, T. Puphal, and B. Flade. Driving Situation Analysis with Relational Local Dynamic Maps (R-LDM). In *Symposium on Future Active Safety Technology*, 2017.

[31] M. J. Eppler. A Systematic Framework for Risk Visualization in Risk Management and Communication. *Risk Management*, 2009.

[32] C. Erbsmehl. Simulation of Real Crashes as a Method for Estimating the Potential Benefits of Advanced Safety Technologies. In *Technical Conference on the Enhanced Safety of Vehicles*, 2009.

[33] J. Erdmann and D. Krajzewicz. SUMO's Road Intersection Model. In *SUMO User Conference*, 2013.

[34] D. Ferguson, M. Darms, C. Urmson, and S. Kolski. Detection, Prediction, and Avoidance of Dynamic Obstacles in Urban Environments. In *Intelligent Vehicles Symposium*, 2008.

[35] D. Ferguson, T. M. Howard, and M. Likhachev. Motion Planning in Urban Environments. *Journal of Field Robotics*, 2008.

[36] C. Frese and J. Beyerer. A Comparison of Motion Planning Algorithms for Cooperative Collision Avoidance of Multiple Cognitive Automobiles. In *Intelligent Vehicles Symposium*, 2011.

[37] R. Garmier, B. Revelin, P. Legendre, and et al. Computation of a Collision Probability Based on a Gaussian Mixture Model of the TLE Accuracy. In *International Symposium on Space Flight Dynamics*, 2009.

[38] B. Gassmann, F. Oboril, C. Buerkle, S. Liu, S. Yan, and et al. Towards Standardization of AV Safety: C++ Library for Responsibility Sensitive Safety. In *Intelligent Vehicles Symposium*, 2019.

[39] I. Gat, M. Benady, and A. Shashua. A Monocular Vision Advance Warning System for the Automotive Aftermarket. *Society of Automotive Engineers Technical Paper*, 2005.

[40] A. Geiger, M. Lauer, F. Moosmann, B. Raft, and et al. Team AnnieWAY's Entry to the Grand Cooperative Driving Challenge 2011. *Transactions on Intelligent Transportation Systems*, 2012.

[41] S. Gibbs. Google Sibling Waymo Launches Fully Autonomous Ride-Hailing Service. *The Guardian*. https://www.theguardian.com.

[42] J. Gregoire, S. Bonnabel, and A. De La Fortelle. Priority-based Intersection Management with Kinodynamic Constraints. In *European Control Conference*, 2014.

[43] D. Gruyer, V. Magnier, and et al. Perception, Information Processing and Modeling: Critical Stages for Autonomous Driving Applications. *Annual Reviews in Control*, 2017.

[44] S. Habenicht, H. Winner, S. Bone, F. Sasse, and P. Korzenietz. A Maneuver-based Lane Change Assistance System. In *Intelligent Vehicles Symposium*, 2011.

[45] N. Hansen and A. Ostermeier. Convergence Properties of Evolution Strategies with the Derandomized Covariance Matrix Adaption: CMA-ES. In *Congress on Intelligent Techniques and Soft Computing*, 1997.

[46] M. Hejase, A. Kurt, T. Aldemir, and U. Ozguner. Quantitative and Risk-Based Framework for Unmanned Aircraft Control System Assurance. *Journal of Aerospace Information Systems*, 2017.

[47] D. Hennes, D. Claes, W. Meeussen, and K. Tuyls. Multi-Robot Collision Avoidance with Localization Uncertainty. In *International Conference on Autonomous Agents and Multiagent Systems*, 2012.

[48] J. Hillenbrand, K. Kroschel, and V. Schmid. Situation Assessment Algorithm for a Collision Prevention Assistant. 2005.

[49] D. Holland-Letz, M. Kässer, B. Kloss, and T. Müller. Start Me Up: Where Mobility Investments are Going. *McKinsey & Company*, 2019. https://www.mckinsey.com.

[50] E. Hollnagel. Safer Complex Industrial Environments: A Human Factors Approach. *CRC Press, Taylor & Francis Group*, 2009.

[51] R. Hooke and T. Jeeves. Direct Search Solution of Numerical and Statistical Problems. In *Journal of the Association for Computing Machinery*, 1961.

[52] A. Houenou, P. Bonnifait, and V. Cherfaoui. Risk Assessment for Collision Avoidance Systems. In *Intelligent Transportation Systems Conference*, 2014.

[53] T. Iijima, A. Higashimata, S. Tange, and et al. Development of an Adaptive Cruise Control System with Brake Actuation. *SAE Technical Paper*, 2000.

[54] S. Ishida and J. E. Gayko. Development, Evaluation and Introduction of a Lane Keeping Assistance System. In *Intelligent Vehicles Symposium*, 2004.

[55] M. Itoh, D. Yokoyama, M. Toyoda, and M. Kitsuregawa. Visual Interface for Exploring Caution Spots from Vehicle Recorder Big Data. In *International Conference on Big Data*, 2015.

[56] J. Johnson and K. Hauser. Optimal Acceleration-Bounded Trajectory Planning in Dynamic Environments Along a Specified Path. In *International Conference on Robotics and Automation*, 2012.

[57] A. Karpathy. Connecting Images and Natural Language. *Stanford University*, 2016.

[58] C. Katrakazas, M. Quddus, W.-H. Chen, and L. Deka. Real-Time Motion Planning Methods for Autonomous On-Road Driving. *Transportation Research Part C*, 2015.

[59] S. Kern, S. Zecha, and D. Westhofen. Intelligent Collision Risk Prediction Using Cooperative Sensor Technology. In *Praxis Conference Pedestrian Protection*, 2011.

[60] A. Kesting, M. Treiber, and D. Helbing. General Lane-Changing Model MOBIL for Car-Following Models. *Transportation Research Record Journal*, 2007.

[61] J. Kim and J. Canny. Interpretable Learning for Self-Driving Cars by Visualizing Causal Attention. In *International Conference on Computer Vision*, 2017.

[62] S. Klingelschmitt, M. Platho, H.-M. Groß, V. Willert, and J. Eggert. Combining Behavior and Situation Information for Reliably Estimating Multiple Intentions. In *Intelligent Vehicles Symposium*, 2014.

[63] P. M. Knoll. An Integrated HMI Concept for Driver Information and Driver Assistance Systems. In *Society for Information Display Symposium*, 2007.

[64] M. Krüger, C. B. Wiebel-Herboth, and H. Wersing. Approach for Enhancing the Perception and Prediction of Traffic Dynamics with a Tactile Interface. In *Automotive User Interfaces*, 2018.

[65] A. Lambert, D. Gruyer, and G. Saint Pierre. A Fast Monte Carlo Algorithm for Collision Probability Estimation. In *International Conference on Control, Automation, Robotics and Vision*, 2008.

[66] J.-H. Lee and H. Lee-Kwang. Distributed and Cooperative Fuzzy Controllers for Traffic Intersections Group. *Transactions on Systems, Man and Cybernetics, Part C*, 1999.

[67] S. Lefevre, D. Vasquez, and C. Laugier. A Survey on Motion Prediction and Risk Assessment for Intelligent Vehicles. In *ROBOMECH Journal*, 2014.

[68] S. H. Leilabadi and S. Schmidt. In-Depth Analysis of Autonomous Vehicle Collisions in California. In *Intelligent Transportation Systems Conference*, 2019.

[69] M. Liebner, F. Klanner, M. Baumann, C. Ruhhammer, and et al. Velocity-based Driver Intent Inference at Urban Intersections in the Presence of Preceding Vehicles. *Intelligent Transportation Systems Magazine*, 2013.

[70] J. K. Lin and P. Dayan. Curved Gaussian Models with Applications to the Modeling of Foreign Exchange Rates. In *Computational Finance Conference*, 1999.

[71] S. M. Loos, A. Platzer, and L. Nistor. Adaptive Cruise Control: Hybrid, Distributed, and Now Formally Verified. In *International Symposium on Formal Methods*, 2011.

[72] J.-J. Martinez and C. Canudas-de Wit. A Safe Longitudinal Control for Adaptive Cruise Control and Stop-and-Go Scenarios. *Transactions on Control Systems Technology*, 2007.

[73] S. Messelodi, C. M. Modena, M. Zanin, F. G. B. De Natale, F. Granelli, E. Betterle, and A. Guarise. Intelligent Extended Floating Car Data Collection. *Expert Systems with Applications*, 2008.

[74] A. Middleton, O. Biham, and D. Levine. Self Organization and a Dynamical Transition in Traffic Flow Models. *Physical Review A*, 1992.

[75] K. Mineta, K. Unuora, and T. Ikeda. Development of a Lane Mark Recognition System for a Lane Keeping Assist System. *SAE Technical Paper*, 2003.

[76] National Commitee on Traffic Laws and Ordinances. Department of Commerce, Bureau of Public Roads. Uniform Vehicle Code and Model Traffic Ordinance. 2000.

[77] M. Naumann and C. Stiller. Towards Cooperative Motion Planning for Automated Vehicles in Mixed Traffic. In *Conference on Intelligent Robots and Systems*, 2017.

[78] J. A. Nelder and R. Mead. A Simplex Method for Function Minimization. *The Computer Journal*, 1965.

[79] M. H. Nguyen and S. Alam. Airspace Collision Risk Hot-Spot Identification using Clustering Models. *Transactions on Intelligent Transportation Systems*, 2018.

[80] D. Orth, D. Kolossa, M. S. Paja, K. Schaller, A. Pech, and M. Heckmann. A Maximum Likelihood Method for Driver-Specific Critical-Gap Estimation. In *Intelligent Vehicles Symposium*, 2017.

[81] B. Paden, M. Cap, S. Z. Yong, D. Yershov, and E. Frazzoli. A Survey of Motion Planning and Control Techniques for Self-driving Urban Vehicles. *Intelligent Vehicles Journal*, 2016.

[82] S. Patil and et al. Estimating Probability of Collision for Safe Motion Planning under Gaussian Motion and Sensing Uncertainty. In *International Conference on Robotics and Automation*, 2012.

[83] C. Pek and M. Althoff. Computationally Efficient Fail-Safe Trajectory Planning for Self-driving Vehicles Using Convex Optimization. In *Intelligent Transportation Systems Conference*, 2018.

[84] D. Petrich and et al. Map-based Long Term Motion Prediction for Vehicles in Traffic Environments. In *Intelligent Transportation Systems Conference*, 2013.

[85] M. Plavsic and et al. Ergonomic Design and Evaluation of Augmented Reality Based Cautionary Warnings for Driving Assistance in Urban Environments. In *World Congress on Ergonomics*, 2009.

[86] M. G. Plessen and et al. Multi-automated Vehicle Coordination using Decoupled Prioritized Path Planning for Multi-Lane One- and Bi-Directional Traffic Flow Control. In *Conference on Decision and Control*, 2016.

[87] J. P. Powell and R. Palacin. Passenger Stability Within Moving Railway Vehicles: Limits on Maximum Longitudinal Acceleration. *Urban Rail Transit*, 2015.

[88] M. Powell. An Efficient Method for Finding the Minimum of a Function of Several Variables without Calculating Derivatives. *The Computer Journal*, 1964.

[89] M. Powell. A Direct Search Optimization Method that Models the Objective and Constraint Functions by Linear Interpolation. In *Advances in Optimization and Numerical Analysis*, 1994.

[90] H. Pu. Dynamic eHorizon with Traffic Light Information for Efficient Urban Traffic. *Advanced Microsystems for Automotive Applications*, pages 15 – 24, 2015.

[91] T. Puphal, B. Flade, D. De Geuss, and J. Eggert. Proactive Risk Navigation System for Real-World Urban Intersections. In *IEEE Intelligent Transportation Systems Conference*, 2020.

[92] T. Puphal, M. Probst, and J. Eggert. Probabilistic Uncertainty-Aware Risk Spot Detector for Naturalistic Driving. In *IEEE Transactions on Intelligent Vehicles*, 2019.

[93] T. Puphal, M. Probst, M. Komuro, Y. Li, and J. Eggert. Comfortable Priority Handling with Predictive Velocity Optimization for Intersection Crossings. In *IEEE Intelligent Transportation Systems Conference*, 2019.

[94] T. Puphal, M. Probst, Y. Li, Y. Sakamoto, and J. Eggert. Optimization of Velocity Ramps with Survival Analysis for Intersection Merge-Ins. In *IEEE Intelligent Vehicles Symposium*, 2018.

[95] X. Qian, I. Navarro, A. De La Fortelle, and F. Moutarde. Motion Planning for Urban Autonomous Driving using Bezier Curves and MPC. In *Intelligent Transportation Systems Conference*, 2016.

[96] J. Redmon and A. Farhadi. YOLOv3: An Incremental Improvement. In *arXiv*, 2018.

[97] H. Ritchie and M. Roser. Causes of Death. *Our World in Data*, 2020. https://ourworldindata.org/causes-of-death.

[98] G. Rodgers. Driving in Asia. *Tripsavvy*, 2019. https://www.tripsavvy.com/driving-in-asia-3987706. License acquired from Tuomas A. Lehtinen/Moment via Getty Images.

[99] M. Ruf, J. R. Ziehn, D. Willersinn, and et al. Comparison of Local vs. Global Optimization for Trajectory Planning in Automated Driving. In *10th Workshop Driver Assistance Systems*, 2015.

[100] SAE International. Taxonomy and Definitions for Terms Related to On-Road Motor Vehicle Automated Driving Systems. 2018.

[101] N. Saunier. Traffic Intelligence. https://bitbucket.org/Nicolas/trafficintelligence/wiki/Home.

[102] B. Schäufele and et al. Fully Automated Valet Parking in Underground Garages through External Positioning Information. In *World Congress on Intelligent Transport Systems*, 2014.

[103] E. Schmerling and M. Pavone. Evaluating Trajectory Collision Probability through Adaptive Importance Sampling for Safe Motion Planning. *Robotics: Science and Systems*, 2017.

[104] J. Schmüdderich and et al. A Novel Approach to Driver Behavior Prediction using Scene Context and Physical Evidence for intelligent ACC. In *Future Active Safety Technology Symposium*, 2015.

[105] M. Schreier, V. Willert, and A. Jürgen. Bayesian, Maneuver-Based, Long-Term Trajectory Prediction and Criticality Assessment for Driver Assistance Systems. In *Intelligent Transportation Systems Conference*, 2014.

[106] R. Schubert, E. Richter, and G. Wanielik. Comparison and Evaluation of Motion Models for Vehicle Tracking. In *Conference on Information Fusion*, 2008.

[107] W. Schwarting, J. Alonso-Mora, and D. Rus. Planning and Decision-Making for Autonomous Vehicles. *Annual Review of Control, Robotics, and Autonomous Systems*, 2018.

[108] V. Sezer, T. Bandyopadhyay, D. Rus, E. Frazzoli, and D. Hsu. Towards Autonomous Navigation of Unsignalized Intersections under Uncertainty of Human Driver Intent. In *Conference on Intelligent Robots and Systems*, 2015.

[109] S. Shalev-Shwartz, S. Shammah, and A. Shashua. On a Formal Model of Safe and Scalable Self-driving Cars. In *arXiv*, 2017.

[110] S. Sivaraman and M. M. Trivedi. Looking at Vehicles on the Road: A Survey of Vision-Based Vehicle Detection, Tracking, and Behavior Analysis. In *Transactions on Intelligent Transportation Systems*, 2013.

[111] Tesla. Vehicle Safety Report. 2019. https://www.tesla.com/Vehicle SafetyReport.

[112] L. Thakali, T. J. Kwon, and L. Fu. Identification of Crash Hotspots using Kernel Density Estimation and Kriging Methods: A Comparison. *Journal of Modern Transportation*, 2015.

[113] C. Thiemann, M. Treiber, and A. Kesting. Estimating Acceleration and Lane-Changing Dynamics Based on NGSIM Trajectory Data. *Journal of the Transportation Research Board*, 2008.

[114] J. J. Thomson. Killing, Letting Die, and the Trolley Problem. *The Monist*, 1976.

[115] S. Thrun, M. Montemerlo, S. Strohband, C. Dupont, G. Bradski, P. Mahoney, and et al. Stanley: The Robot that Won the DARPA Grand Challenge. *Journal of Field Robotics*, 2006.

[116] T. Toledo and D. Zohar. Modeling Duration of Lane Changes. *Transportation Research Record*, 2007.

[117] Transportation Research Board of the National Academies of Science. Highway Capacity Manual. 2010.

[118] M. Treiber, A. Hennecke, and D. Helbing. Congested Traffic States in Empirical Observations and Microscopic Simulations. *Physical Review E*, 2000.

[119] M. Treiber and A. Kesting. Automatic and Efficient Driving Strategies while Approaching a Traffic Light. In *Intelligent Transportation Systems Conference*, 2014.

[120] US Department of Homeland Security. Risk Management Fundamentals. 2011.

[121] US Department of Transportation. Next Generation Simulation (NGSIM). Traffic Analysis Tools. https://ops.fhwa.dot.gov/trafficanalysistools/ngsim.htm.

[122] US Department of Transportation. Traffic Safety Facts. Fatal Motor Vehicle Crashes: Overview. 2019.

[123] J. Van Den Berg, S. J. Guy, M. Lin, and D. Manocha. Optimal Reciprocal Collision Avoidance for Multi-Agent Navigation. In *International Conference on Robotics and Automation*, 1992.

[124] R. Van der Horst. Time-To-Collision as a Cue for Decision-Making in Braking. In *Vision in Vehicles III*, 1991.

[125] Vision-Inspired Driver Assistance System (VI-DAS). Project Financed by the European Union. 2019. See http://www.vi-das.eu and https://cordis.europa.eu/project/id/690772/de.

[126] T. Voelk. Crash Scene Investigations, With Automakers on the Case. *Volvo*, 2019. https://www.nytimes.com.

[127] W. Wachenfeld, P. Junietz, R. Wenzel, and H. Winner. The Worst-Time-To-Collision Metric for Situation Identification. In *Intelligent Vehicles Symposium*, 2016.

[128] J. Ward, G. Agamennoni, S. Worrall, and E. Nebot. Vehicle Collision Probability Calculation for General Traffic Scenarios Under Uncertainty. In *Intelligent Vehicles Symposium*, 2014.

[129] H. Watanabe, L. Tobisch, T. Laudien, J. Wallner, and G. Prokop. A Method for the Estimation of Coexisting Risk-Inducing Factors in Traffic Scenarios. In *Intelligent Vehicles Symposium*, 2019.

[130] Waymo. Safety Report. On the Road to Fully Self-Driving. 2018. https://storage.googleapis.com/sdc-prod/v1/safety-report/Safety

[131] T. A. Wheeler. Automotive Safety Validation in Simulation. PhD Thesis. *Stanford University*, 2018.

[132] H. Winner, S. Hakuli, F. Lotz, and C. Singer. *Handbook of Driver Assistance Systems: Basic Information, Components and Systems for Active Safety and Comfort*. Springer Vieweg, 2016.

[133] H. Winner, F. Lotz, E. Bauer, and et al. PRORETA 3: Comprehensive Driver Assistance by Safety Corridor and Cooperative Automation. *Handbook of Driver Assistance Systems*, 2016.

[134] H. Winner, W. Wachenfeld, and P. Junitz. Validation and Introduction of Automated Driving. *Automotive System Engineering II*, 2018.

[135] H. Winner, S. Witte, W. Uhler, and B. Lichtenberg. Adaptive Cruise Control System Aspects and Development Trends. *SAE Technical Paper*, 1996.

[136] P. Wolf, K. Kurzer, and et al. Adaptive Behavior Generation for Autonomous Driving using Deep Reinforcement Learning with Compact Semantic States. In *Intelligent Vehicles Symposium*, 2018.

[137] World Health Organization. World Report on Road Traffic Injury Prevention: Summary. 2004.

[138] H. Xu, Y. Gao, and et al. End-to-End Learning of Driving Models from Large-Scale Video Datasets. In *Conference on Computer Vision and Pattern Recognition*, 2019.

[139] G. Yahiaoui. SafetyNex - Onboard Driving Risk Assessment in Real-Time. https://www.safetynex.nexyad.net.

[140] M. Zetlin. Rude Drivers Who Merge at the Last Second. *Inc*, 2017. https://www.inc.com/minda-zetlin/rude-drivers-who-merge-at-the-last-second-are-doin.html. License from iStock.com/kozmoat98.

[141] Z.-Q. Zhao, P. Zheng, S.-T. Xu, and X. Wu. Object Detection with Deep Learning: A Review. *Transactions on Neural Networks and Learning Systems*, 2019.

[142] X. Zou and W. L. Yue. A Bayesian Network Approach to Causation Analysis of Road Accidents using Netica. *Journal of Advanced Transportation*, 2017.

[143] R. Zucker. How Self-Driving Cars Think: Navigating Double-Parked Vehicles. *Cruise*, 2019. https://medium.com/cruise/double-parked-vehicles-4f5ac8fc05a9.

Curriculum Vitae

Personal information	
Name	Tim Puphal
Email Address	timpuphal@hotmail.de
Date/Place of Birth	23. 05. 1991 in Frankfurt am Main

History	
from 09.2020	Robotics Scientist at Honda Research Institute Europe GmbH
2016 – 2020	Associate at Honda Research Institute Europe GmbH in Offenbach as well as external PhD student at Control Methods and Robotics Lab, Technical University of Darmstadt
2013 – 2016	Master of Science in Mechatronics, specialization: Automotive engineering, control theory and electrical drives, Technical University of Darmstadt
2014	Studies abroad in Tongji University (China) within scholarship program of Continental AG
2010 – 2013	Bachelor of Science in Mechatronics, Technical University of Darmstadt
2010	German Abitur from Georg-Büchner Gymnasium, Bad Vilbel