High-level Decision Making under Safety Constraints for Autonomous Vehicles

Zur Erlangung des akademischen Grades eines

DOKTORS DER INGENIEURWISSENSCHAFTEN (Dr.-Ing.)

von der KIT-Fakultät für Maschinenbau
des Karlsruher Instituts für Technologie (KIT)
angenommene

DISSERTATION

von

M.Sc. Lingguang Wang

Tag der mündlichen Prüfung:	20.08.2024
Hauptreferent:	Prof. Dr.-Ing. Christoph Stiller
Korreferent:	Prof. Dr.-Ing. Steven Peters

Abstract

Future transportation systems serve as a crucial foundations for enhancing human productivity. Among these, automated driving has emerged as a vital technology with the potential to improve the comfort and efficiency of road traffic while freeing human hands from driving tasks.

It is anticipated that automated driving systems will coexist with human drivers on the road for decades before achieving full automation. As such, a key research goal is to develop autonomous vehicles that closely mimic human driving behavior, enabling passengers, other human drivers, and traffic participants to better understand and cooperate with these vehicles. Furthermore, ensuring provable safety is essential for the widespread acceptance of automated driving systems.

To develop driving behavior capable of handling generic traffic scenarios, existing approaches often frame the behavior planning problem as a sequential decision-making process aimed at maximizing expected future rewards. These methods frequently suffer from unrealistic reward functions and lack definitive proof of human-like behavior. To address these shortcomings, machine-learning techniques have been employed to derive driving policies from recorded human driving trajectories in real traffic. However, some of these approaches face runtime challenges, while others struggle to ensure compliance with traffic rules and safety regulations.

In this work, I introduce a high-level decision-making framework for autonomous vehicles, focusing on safety across diverse traffic situations and adherence to traffic rules. This method, enhancing an existing safety concept, takes into account factors like road types and occlusions, while also relaxing safety requirements to achieve more natural driving behavior. It involves

generating safe action options, simulating future traffic scenarios to assess outcomes, and using machine learning to analyze human driving data for decision-making. This results in actions that mirror human decision processes in complex driving environments.

My approach is evaluated through simulations of various scenarios, including parallel lanes and intersecting lanes. Simulation evaluations demonstrated that the learned policy outperformed the rule-based baseline approaches, producing more human-like behavior while balancing driving efficiency, comfort, perceived safety, and politeness.

Finally, the key part of the proposed approach has been successfully deployed on real experimental vehicles. Demonstrations have been conducted on a test track, as well as in regular on-road experiments.

Kurzfassung

Zukünftige Mobilitätskonzepte bilden die entscheidende Grundlage für die Steigerung der menschlichen Produktivität. Unter ihnen hat sich das automatisierte Fahren als eine wichtige Technologie etabliert, die darauf abzielt, den Komfort und die Effizienz des Straßenverkehrs zu verbessern und gleichzeitig die Hände der Fahrer von Fahraufgaben zu befreien.

Es ist absehbar, dass automatisierte Fahrsysteme für mehrere Jahrzehnte mit menschlichen Fahrern auf der Straße koexistieren werden, bevor sie vollständig automatisiert sind. Daher ist es ein wichtiges Forschungsziel, autonome Fahrzeuge zu entwickeln, die menschliches Fahrverhalten möglichst genau imitieren, damit Passagiere, andere Fahrer und Verkehrsteilnehmer diese Fahrzeuge besser verstehen und mit ihnen kooperieren können. Darüber hinaus ist die Gewährleistung nachweisbarer Sicherheit für die breite Akzeptanz automatisierter Fahrsysteme von entscheidender Bedeutung.

Um ein Fahrverhalten zu erzeugen, das für generische Verkehrsszenarien geeignet ist, stellen bestehende Ansätze das Verhaltensplanungsproblem häufig als sequenziellen Entscheidungsprozess dar, bei dem zukünftige erwartete Belohnungen maximiert werden. Ein wesentlicher Nachteil dieser Methoden besteht in der Schwierigkeit, geeignete Belohnungsfunktionen für realistisches und menschähnliches Verhalten zu finden. Um diese Mängel zu beheben, sind Techniken des maschinellen Lernens vorgeschlagen worden, um Fahrverhalten aus aufgenommenen menschlichen Trajektorien im realen Verkehr abzuleiten. Einige dieser Ansätze haben jedoch Laufzeitprobleme, während andere Schwierigkeiten haben, die Einhaltung von Verkehrsregeln und Sicherheitsvorschriften sicherzustellen.

In dieser Arbeit stelle ich einen Ansatz zur Verhaltensplanung für autonome Fahrzeuge vor, der im Hinblick auf die Sicherheit in verschiedenen Verkehrssituationen und die Einhaltung von Verkehrsregeln entwickelt wird. Diese Methode, die ein bestehendes Sicherheitskonzept verbessert, berücksichtigt Faktoren wie Straßentypen und Verdeckungen und lockert gleichzeitig Sicherheitsanforderungen, um ein natürlicheres Fahrverhalten zu erreichen. Sie beinhaltet die Erzeugung sicherer Aktionskandidaten, die Bewertung des Szenenfortgangs durch Vorwärtssimulation und die Verwendung von maschinellem Lernen zur Analyse von menschlichen Fahrdaten für die Entscheidungsfindung. Dies führt zu Verhalten, die menschliche Entscheidungsprozesse in komplexen Fahrumgebungen widerspiegeln.

Mein Ansatz wird in Simulationen verschiedener Szenarien bewertet, darunter parallele Fahrbahnen und sich kreuzende Fahrbahnen. Simulationsevaluationen zeigten, dass unsere erlernte Fahrstrategie im Vergleich zu regelbasierten Ansätzen eine menschenähnlichere Verhaltensweise aufweist und dabei die Effizienz und sicherheit, den Komfort und die Höflichkeit besser ausbalanciert.

Schließlich wurden die Schlüsselkomponente des vorgeschlagenen Ansatzes erfolgreich in echten Versuchsfahrzeugen implementiert. Demonstrationen wurden sowohl auf einer Teststrecke als auch bei Experimenten im Straßenverkehr durchgeführt.

Acknowledgements

Writing a PhD thesis is a long journey. I would like to express my sincere gratitude to all those who have contributed to my PhD journey.

First and foremost, I would like to thank my supervisor, Prof. Stiller, for his invaluable guidance and support throughout my PhD. The insights gained from our discussions in group meetings and scientific seminars were immensely inspirational.

My gratitude extends to Prof. Steven Peters for his role as co-examiner and the opportunity to collaborate with his team members, Dr. Homolla and Dr. Ackermann, in the UNICARagil project.

I am also deeply grateful to all my colleagues at MRT for their support. The exchanges we shared were crucial in refining my research ideas. A special note of thanks to Sahin, who placed his faith in me and recommended me to our institute. I am thankful to Carlos for his pivotal role in shaping my initial PhD concept, and to Maximilian, Danial, Christoph, Piotr, and Sahin for their enriching discussions. In project work, Martin's exceptional leadership was invaluable, and Eduardo's steadfast support and camaraderie were truly remarkable.

I am also thankful to Carlos, Sahin, Johannes, and Etienne for their insightful feedback on the drafts of this work.

Last but not least, I would like to express my heartfelt thanks to my family, especially my wife Sirui, and friends for their love, support and encouragement.

Thank you all!

Karlsruhe, February 2024 *Lingguang Wang*

Contents

Appendix

1 Introduction

The development of Autonomous Driving (AD) technology is a complex and multifaceted undertaking that involves numerous sub-modules. Over the past several decades, significant efforts have been devoted to researching and developing AD systems with the goal of enhancing the overall safety and efficiency of road traffic.

Section 1.1 first introduces the context and motivation of the work. Section 1.2 formulates the objectives. Afterward, Section 1.3 summarizes the contributions of the work.

1.1 Context and Motivation of the Work

AD can hardly be achieved solely through a single module. A reasonable division of the functionality of AD into sub-systems and sub-modules enables a structured and simplified realization of functional safety, as well as the opportunity for parallel development and testing. However, it is critical to establish proper interfaces between sub-systems and sub-modules in advance to ensure seamless integration and interoperability.

In the literature, it has been suggested in [Taş17] that an AD system can be subdivided into four sub-systems, namely *sensors*, *perception and scene understanding*, *behavior and motion planning*, and *vehicle control and actuation*. The *sensors* collect environmental data using devices like cameras, radars and lidars. *Perception and scene understanding* processes this data to identify vehicle's surroundings, classify obstacles, assess their movements, and understand their intentions. *Behavior and motion planning* then uses this information to

determine the vehicle's driving decisions and plan future trajectory, considering safety and traffic rules. Ultimately, *vehicle control and actuation* manages the vehicle's steering, acceleration, and braking functions to accurately follow the outlined trajectory.

The present work focuses on *behavior and motion planning*, specifically on the *behavior planning* (or *decision making*) aspect of AD. It generates high-level semantic decisions, such as changing lanes, yielding, or passing through intersections, rather than concrete time-state sequences (*trajectories*) which is managed by *motion planning* (or *trajectory planning*).

It is important to note that a decision-making module cannot assume perfect knowledge about the environment, as neither the *perception* module nor the *scene understanding* module can provide perfect states and predictions for traffic participants. With these uncertainties, the perceived safety of the passengers is more likely to be compromised, leading to increased driving risks. Slowing down is often considered a universal solution to minimize driving risks, but this may result in overly cautious behavior, thus affecting driving efficiency. Therefore, it is crucial for the decision maker to strike a balance between risk and utility.

Furthermore, decision making in AD becomes particularly challenging in mixed traffic situations where Autonomous Vehicles (AVs) coexist with human-driven vehicles and other traffic participants. From their perspective, the behavior of AVs should be similar to that of human drivers to enable normal interaction. A different behavior can lead to misinterpretation, causing traffic problems, disturbances to other road participants, or even accidents. Additionally, as the universal acceptance of autonomous driving systems relies on the provable safety and explainability of the output decision, decision-making systems that are opaque and lack interpretability of their outputs are not suitable for this purpose.

1.2 Objectives

Motivated by the numerous challenges associated with developing AD systems, the objective of this research is to propose a novel approach for high-level decision making in AVs that can make reasonable decisions under uncertain environment perception and scene understanding. The proposed approach should be versatile enough to be applied to various traffic scenarios without requiring significant modifications. Additionally, the output decision should remain at a semantic level, which can then be transferred to any trajectory planning and control module.

Prioritizing the safety of both the autonomous vehicle and other traffic participants, the approach must allow for formal safety verification under certain traffic rules and reasonable assumptions. The approach will also be designed to output decisions that are interpretable and explainable, so that human users and regulators can understand the system's decision-making process and ensure that it is behaving in a safe and responsible manner.

Furthermore, the proposed approach should make decisions that are similar to the patterns from human driving trajectories and should balance efficiency, comfort, and perceived safety in a human-like way. Ideally, the approach should allow for the acquisition of diverse driving policies from the training data to reflect different types of drivers. To achieve this adaptability, the proposed approach will be based on a machine learning framework, which will allow the system to learn from real-world driving data and adapt to new scenarios as they arise.

Overall, the proposed decision-making approach aims to address the key challenges of developing a safe, efficient, and adaptable AD system. By incorporating machine learning, formal safety verification, and interpretability, the proposed approach has the potential to significantly improve the safety and efficiency of autonomous driving systems, and enable the widespread adoption of this transformative technology.

1.3 Contributions

The contributions of this thesis are as follows:

- Proposing a comprehensive framework for learning human-like driving behaviors from recorded data, integrating human knowledge (e.g., traffic rules) for high-level decision-making. This framework ensures interpretability and traceability of decisions.
- Refining the safety concept from [Nau20a], extending the Responsibility-Sensitive Safety (RSS) framework [Sha17] to cover various traffic scenarios and intersection types, addressing occlusions and proximity challenges. Precise definitions for vehicle reachability and other limits enhance the safety model.
- The extended RSS safety concept is further relaxed in the case of a better perception and scene understanding, e.g. by the proposed method for temporal tracking of occluded road sections. Statistic analyses on the relaxed safety concept on real traffic data are performed as well.
- Introducing a new schema for representing and executing high-level actions in various scenarios, and proposing rule-based policies that are provable safe within the safety framework.
- To derive driving policies from real data, key features affecting driving decisions are identified. Their values are estimated via Monte-Carlo Simulation (MCS) that is able to handle uncertainties and occlusions.
- Developing a clustering approach with MCS-derived features to group training data, capturing varied driving styles and facilitating diverse policy learning.
- Creating a method to convert recorded trajectories into interactive simulations, enabling the evaluation of our approach in realistic settings.

2 Fundamentals and Related Work

Decision making is a cornerstone of autonomous systems, underpinning their ability to navigate complex environments, interact safely with other agents, and achieve designated objectives. However, the real-world scenarios in which these systems operate often present numerous uncertainties, arising from incomplete observations, unpredictable environments, or inherent system dynamics. This chapter delves into the theoretical foundations and practical approaches to decision making under such uncertainties, setting the stage for its application in the field of automated driving.

2.1 Fundamentals of Decision Making under Uncertainty

The art and science of making informed choices in the face of uncertainty have been a focal point of research across diverse domains, from economics to robotics. Decision making under uncertainty attempts to offer robust solutions that account for various sources of ambiguities, ensuring reliable and efficient system behavior. The following subsections dissect the principal methodologies for sequential decision making, decision making with state or model uncertainty, and the emerging field of learning decisions directly from demonstrations.

2.1.1 Sequential Decision Making

Sequential decision making involves making a series of decisions over time, often with the intent of optimizing a particular outcome. The Markov Decision Process (MDP) framework serves as a foundational model for representing these decision making processes.

Markov Decision Process (MDP)

A MDP is a mathematical structure, described by the tuple (S, A, T, R), where S represents a finite set of states, capturing all possible situations or configurations of the system in question. A denotes a finite set of actions, defining the different choices or maneuvers that can be executed in any given state. $T : S \times A \times S \to [0,1]$ is the state transition function. It provides the dynamics of the environment, stipulating the probability of moving from one state to another after choosing a specific action. The state transition occurs probabilistically, based on the present state and chosen action. This presupposition, which posits that the subsequent state relies solely on the current state and action, excluding any preceding states or actions, is termed the *Markov assumption*. $R : S \times A \times S \to \mathbb{R}$ is the reward function, which allots a numerical reward (or cost) for transitioning between states given an action.

Within the MDP framework, a policy π is defined as a mapping from states to actions, $\pi : S \to A$. In deterministic policies, each state is mapped to a specific action. In stochastic policies, π defines a probability distribution over actions for each state, $\pi : S \to \mathscr{P}(A)$, where $\mathscr{P}(A)$ is the set of all probability distributions over A.

The objective in an MDP is to determine an optimal policy π^* that maximizes the expected cumulative reward over time. The quality of a policy is quantified by its value function $V^\pi(s)$, which calculates the expected cumulative reward of starting in state s and following policy π.

Bellman Equation

The value function is a cornerstone of understanding the quality of decision making in MDPs. It quantifies the expected cumulative reward when starting from a particular state s and following a specific policy π.

Mathematically, for a policy π, the value function $V^{\pi}(s)$ is defined as:

$$V^{\pi}(s) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t R(s_t, \pi(s_t), s_{t+1}) | s_0 = s\right] \tag{2.1}$$

Where $\mathbb{E}$ is the expectation operator, and s_t denotes the state at time t. The discount factor γ (where $0 \leq \gamma < 1$) determines the present value of future rewards, with a lower γ giving more weight to immediate rewards and a higher γ emphasizing long-term gains.This function provides a measure of the worth of a state when adhering to policy π, considering both immediate and future rewards.

While each policy has its value function, some policies are better than others. The optimal value function $V^*(s)$ signifies the highest expected cumulative reward achievable from state s:

$$V^*(s) = \max_{\pi} V^{\pi}(s) \tag{2.2}$$

It represents the best performance that can be attained from a state, irrespective of the initial policy.

Obtaining $V^*(s)$ directly is often computationally challenging, especially for large state spaces. However, one can exploit the recursive nature of the value function to simplify the process. This brings us to the Bellman equation, which provides a relationship between the value of a state and the potential values of its successors.

The Bellman optimality equation is foundational for deriving the optimal value function and, subsequently, the optimal policy. It is given by:

$$V^*(s) = \max_{a \in A} \sum_{s' \in S} T(s, a, s')[R(s, a, s') + \gamma V^*(s')] \tag{2.3}$$

The term $\max_{a \in A}$ seeks the action that maximizes the expected value of the cumulative reward. The sum over s' accounts for all possible next states, weighting them by their transition probabilities. The reward $R(s, a, s')$ represents the immediate reward, and $\gamma V^*(s')$ captures the discounted future reward.

The beauty of the Bellman optimality equation lies in its recursive structure. By iteratively updating the value function estimates using this equation, one can converge to the optimal value function. In essence, the Bellman equation offers a systematic, recursive approach to deduce the optimal decision-making strategy in MDPs.

Solving MDPs

Several iterative algorithms aim to solve MDPs and obtain the optimal policy π^*:

- **Value Iteration (VI)**: VI successively refines the value of each state using the Bellman equation until a convergence criterion is met. With each iteration, the value function approximation improves, eventually converging to the optimal value function.
- **Policy Iteration (PI)**: PI alternates between assessing a fixed policy and enhancing it based on the current estimated value function. Although the number of possible policies is exponential in the number of states, PI often converges quickly for small problems. However, it can be expensive for large problems as the policy needs to be evaluated each iteration.

In essence, the MDP framework provides a systematic approach to address sequential decision-making challenges, offering the tools to find the best action strategy for any given system.

2.1.2 Decision Making with State Uncertainty

State uncertainty introduces a significant layer of complexity in the decision-making process. It emerges when an agent cannot perfectly perceive its environment due to noise, sensor limitations, or other sources of ambiguity. The Partially Observable Markov Decision Process (POMDP) serves as a foundational model to address decision making in scenarios riddled with state uncertainties.

Partially Observable Markov Decision Process (POMDP)

A POMDP is an extension of the MDP that incorporates the notion of observations. Formally, a POMDP is described by the tuple (S, A, T, R, O, Z), where O represents a finite set of observations that the agent can perceive. $Z : S \times A \times O \rightarrow [0,1]$ is the observation function, which provides the probability of observing $o \in O$ after taking action $a \in A$ and landing in state $s' \in S$.

Given the uncertainties in state observations, agents operating within POMDPs often maintain a belief $b(s)$, which is a probability distribution over all states in S. The set of all possible beliefs constitutes the *belief space*, denoted as B. With each action taken and subsequent observation received, the belief is updated to better reflect the current understanding of the environment. Several methods, such as the Kalman filter and particle filter, are employed to update these beliefs:

- **Kalman Filter:** Assuming the system and observations are linear and Gaussian, the Kalman filter offers an efficient way to update beliefs based on a series of measurements over time. It has several extensions to be applicable on non-linear systems as well.
- **Particle Filter:** For non-linear or non-Gaussian systems, particle filters use a set of samples (or particles) to represent the belief distribution and update it.

Offline vs. Online Belief State Planning

Offline methods aim to compute a policy for every possible belief in the belief space B beforehand. While these methods can produce policies that are immediately accessible during execution, they often face scalability issues, especially in large state spaces. In contrast, *online* methods compute actions on-the-fly based on the current belief. By focusing only on the immediate decision, online methods can navigate large state spaces more efficiently, and adapt to unforeseen scenarios or non-stationary environments better. While online methods offer several advantages, they also face some challenges.

- **Computational Time Constraints:** Online methods require real-time or near-real-time computation. For environments that necessitate rapid decisions, the agent might not have sufficient time to explore the belief space deeply, leading to suboptimal actions.
- **Limited Lookahead:** Due to time constraints, online methods might often consider only a limited horizon. This can cause them to miss long-term consequences of actions, making the decision process myopic.
- **Vulnerability to Model Imperfections:** Online methods rely heavily on the accuracy of the system's model for on-the-fly computations. Any imperfection in the model, such as inaccurate transition or observation probabilities, can greatly degrade the quality of decisions.

Despite these challenges, online belief state planning remains a valuable approach in many scenarios, especially when the state space is large or when the environment is dynamic. However, understanding these limitations is crucial to apply online methods effectively and to devise potential mitigations.

There are several methods for solving POMDPs online. For instance, *forward search* evaluates the consequences of possible actions by simulating trajectories from the current belief state. While effective in evaluating the consequences of actions by simulating trajectories, it has some significant challenges. For instance, the branching factor in forward search can lead to exponential growth in the number of trajectories as the depth of search increases. This can become

computationally intractable for deep searches or environments with many actions. In order to tackle the problems, *branch and bound* aims to prune the search tree by setting bounds on the potential value of trajectories. If a partial trajectory's upper bound is lower than the currently known best solution's lower bound, this trajectory can be pruned without further exploration. This approach reduces the search space significantly, making the process more efficient. Instead of considering all possible trajectories, *sparse sampling* selects a random subset to evaluate. This method, while potentially missing certain paths, drastically reduces the computational overhead. By adjusting the number of samples, one can trade-off between accuracy and computational efficiency.

Transitioning from traditional forward search methods, Monte Carlo Tree Search (MCTS) has emerged as a powerful tool for online belief state planning. By statistically sampling the belief space, MCTS focuses on more promising paths, leveraging the exploration-exploitation trade-off. Its tree-based structure allows for iterative deepening, addressing the short-horizon bias of conventional forward search. With its ability to scale gracefully with computational resources and provide anytime solutions, MCTS stands as a robust alternative in scenarios where forward search's disadvantages become pronounced.

In sum, when grappling with state uncertainty, methodologies derived from POMDPs offer robust tools to ensure agents make informed decisions, even when faced with partial or noisy observations.

2.1.3 Decision Making with Model Uncertainty

Model uncertainty arises when the agent lacks a complete understanding or model of the environment dynamics, leading to a need for exploration. Reinforcement Learning (RL) is a paradigm where agents learn by interacting with an environment and receiving feedback in the form of rewards. The primary goal in RL is to find a strategy or policy that maximizes the expected cumulative reward over time.

Model-based vs. Model-free Methods

In the RL framework, there are two main approaches: *model-based* and *model-free*. *Model-based* methods utilize a model of the environment's dynamics to plan and decide on actions, often allowing for efficient learning. However, acquiring an accurate model can be challenging in many real-world situations. In contrast, *model-free* methods do not rely on an explicit model of the environment. Instead, they learn a policy or value function directly from interactions with the environment, making them more versatile in the face of model uncertainty.

Q-learning

Focusing on model-free methods, one of the most fundamental algorithms is Q-learning. The central idea is to learn the Q-function, denoted $Q(s,a)$, which represents the expected cumulative reward from taking action a in state s and following the optimal policy thereafter. The Q-function is updated iteratively using the Bellman equation:

$$Q(s,a) \leftarrow Q(s,a) + \alpha \left(r + \gamma \max_{a'} Q(s', a') - Q(s,a) \right) \tag{2.4}$$

where α is the learning rate, r is the immediate reward after taking action a in state s, γ is the discount factor, and s' is the succeeding state after the action.

On-policy vs. Off-policy Methods

Model-free methods can be further categorized into *on-policy* and *off-policy* methods. *On-policy* methods, learn the value of the policy being used for exploration, i.e., the same policy is responsible for both learning and decision making. Off-policy methods, like Q-learning, learn the value of an optimal policy while behavior might be derived from another exploratory policy, allowing them to learn independently of the policy used to generate the data.

As the field of RL evolved, several advanced on-policy algorithms emerged, aiming to address challenges in stability, scalability, and efficiency. *Trust Region Policy Optimization* [Sch15] and *Proximal Policy Optimization* [Sch17] enforce a constraint on the policy update to ensure that changes aren't too drastic, promoting stable learning.

Besides Q-learning, *Actor-Critic* [Sut99] methods count as off-policy approaches as well, which decouple the policy and value function into two separate structures: an actor that proposes actions based on the learned policy and a critic that evaluates them based on the learned value function. This separation allows for more stable and efficient learning. *Deep Deterministic Policy Gradient* [Sil14] is an actor-critic method but is designed for continuous action spaces.

On-policy approaches are constantly adapting to their learned behavior, making them more responsive to changes but are often more expensive to train. In contrast, *off-policy* methods, can learn from past experiences or even hypothetical scenarios. This decoupling allows for greater training flexibility. However, a major challenge is reconciling the discrepancy between the target and behavior policies, requiring complex corrections to align state-action distributions, which, if not managed carefully, can lead to learning instability and divergence due to high variance or bias.

2.1.4 Learning Decisions from Demonstration

Learning from demonstration, often referred to as Imitation Learning (IL), is a paradigm where the learning agent seeks to emulate expert behavior by observing demonstrations. Unlike traditional RL, where agents learn from trial and error, imitation learning benefits from expert knowledge, potentially speeding up the learning process. IL introduces two additional notations. τ is a trajectory, which is a sequence of state-action pairs $\tau = \{(s_1, a_1), \ldots, (s_T, a_T)\}$. $\mathscr{D}$ represents a dataset of expert trajectories $\mathscr{D} = \{\tau_1, \tau_2, \ldots, \tau_N\}$.

Behavior Cloning (BC)

Behavior Cloning (BC) is a supervised learning approach where the agent learns a policy $\pi(a|s)$ that tries to replicate the expert's actions in observed states. Mathematically, BC seeks to maximize the likelihood of the observed expert actions given the states:

$$\pi^* = \arg\max_{\pi} \sum_{\tau \in \mathcal{D}} \sum_{(s,a) \in \tau} \log \pi(a|s) \tag{2.5}$$

However, BC has inherent shortcomings such as cascading errors, where small errors can compound over time as the agent diverges from the expert's trajectory, and poor generalization, where BC might not perform well in states not present in the demonstration dataset. To address these challenges, strategies like Dataset Aggregation (DAgger) [Ros11] have been proposed. DAgger involves iteratively collecting dataset from both the expert and the current policy to ensure the policy is trained on its distribution of states.

Inverse Reinforcement Learning (IRL)

Inverse Reinforcement Learning (IRL) is predicated on determining the underlying reward function that an expert might be optimizing, given their observed behaviors. Traditional IRL endeavors to find a reward function that makes the expert's behavior appear optimal. In contrast, Maximum Entropy IRL [Zie08] assumes the expert acts optimally with respect to some reward function but also behaves to maximize the entropy of their policy. This view suggests that the expert introduces randomness in their actions, reflecting a probabilistic manner in decision making.

Formally, the objective in Maximum Entropy IRL is given by:

$$\max_{R} \left(\sum_{\tau \in \mathcal{D}} p(\tau|R) - \lambda H(\tau) \right), \tag{2.6}$$

where R is the inferred reward function and τ is a trajectory consisting of state-action pairs. $p(\tau|R)$ denotes the probability of trajectory τ given the reward function R. It indicates how probable a particular trajectory is for a given reward function. Deriving this requires solving an MDP with reward function R to get a policy and then computing the likelihood of the trajectory under this policy. $H(\tau)$ is the entropy of the trajectory, signifying the unpredictability in the expert's behavior. λ is a parameter balancing the expert data fit and the policy's entropy. A higher λ emphasizes stochasticity, while a smaller one accentuates fitting expert demonstrations.

The goal in Maximum Entropy IRL is to deduce a reward function R that harmonizes these two components: aligning with expert trajectories while also encapsulating stochastic behavior.

Adversarial Methods

Generative Adversarial Imitation Learning (GAIL) [Ho16] presents an adversarial approach to imitation learning. The idea is to train a policy π and a discriminator concurrently. The discriminator, $D(s, a)$, tries to distinguish between the agent's and expert's trajectories. It is trained to maximize:

$$\mathscr{L}_D = \sum_{\tau \in \mathscr{D}} \log D(s, a) + \sum_{\tau' \sim \pi} \log(1 - D(s, a)), \tag{2.7}$$

while the policy, $\pi(a|s)$, minimizes:

$$\mathscr{L}_\pi = \mathbb{E}_{(s,a)\sim\pi}[\log(1 - D(s, a))]. \tag{2.8}$$

In GAIL, the policy learns to produce trajectories that the discriminator can not distinguish from the expert's, leading to imitation of the expert behavior.

To summarize, IL provides a path to harness expert demonstrations, sidestepping some challenges posed by traditional RL. Whether through direct replication in BC, reward inference in IRL, or adversarial frameworks like GAIL,

these methods present rich avenues for learning policies in complex environments, especially when exploration is costly or dangerous.

2.2 Decision Making for Automated Driving

Decision making under uncertainty is a fundamental concept that finds its application in various domains, including robotics. However, when it comes to AD, the decision-making process is nuanced and presents unique challenges. Unlike general decision making for robotics, AD operates in a dynamic environment with multiple unpredictable agents, such as human drivers, pedestrians, and other road users. This introduces additional complexities:

Human Interaction: AD systems must anticipate and understand human behavior, which cannot be directly measured, leading to high uncertainties in prediction [Hub17].

Dynamic Environments: Road conditions, traffic flow, and environmental factors can change rapidly, requiring the AD system to constantly adapt and make decisions in real-time.

Traffic Rule Compliance: AVs are required to obey various predefined traffic rules that has large influence on the decision-making process.

Safety Constraints: Unlike other robotic systems, any decision made by an AD system has direct implications on human safety, both inside and outside the vehicle. Especially critical is the fact that AVs usually drive at higher speed which makes safety more difficult to maintain. This necessitates the development of decision-making algorithms that prioritize safety above all else.

Given these challenges, various approaches have been proposed to address decision making for AD. In the subsequent sections, I will delve into *probabilistic planning approaches*, *RL*, and *imitation learning*, shedding light on the recent advancements and methodologies in these areas.

2.2.1 Probabilistic Planning Approaches

Probabilistic planning approaches aim to address the inherent uncertainties in the decision-making process for AD. By leveraging probabilistic models, these approaches provide a structured way to handle the unpredictable nature of the driving environment.

A notable approach in this domain is the use of MCTS. For instance, a cooperative combinatorial motion planning algorithm has been proposed that does not necessitate inter-vehicle communication, relying instead on MCTS [Len16]. This approach is particularly beneficial in scenarios like autonomous highway driving.

The challenge of understanding and predicting the intentions of surrounding vehicles has been formulated as a POMDP. The intention of other vehicles serves as hidden variables in this model, accounting for uncertainties stemming from noisy sensor data and the unobservable nature of human intentions [Hub17]. This POMDP formulation has been further extended to address complex driving scenarios. For instance, merging into narrow gaps in high traffic density urban environments requires considering interactions with other vehicles. By including surrounding drivers in the state space, a more interactive behavior can be realized, which is solved online using Monte Carlo sampling combined with an efficient A* rollout heuristic [Hub18].

Another challenge in urban driving is handling occlusions. A POMDP-based maneuver planner has been introduced that uses Monte Carlo sampling to generate possible future episodes. These episodes account for the uncertain behavior of known traffic participants and the existence probability of phantom vehicles in occluded areas [Hub19].

The concept of belief-based rewards has also been explored, where planning in POMDPs inherently gathers the necessary information to act optimally under uncertainties. By considering belief-based rewards, POMDP planning can be guided towards informative beliefs, combining the original reward with the expected information gain [Fis20]. Efficiency in sampling-based planning approaches has been enhanced by utilizing variants of Multi-Armed Bandit

heuristics. These heuristics make Lipschitz continuity assumptions on the outcomes of actions, improving the efficiency of such planning methods [Taş21].

Lastly, to address the curse of dimensionality in belief space planning, policies trained in belief space have been proposed as heuristics. These heuristics guide online belief space planning algorithms, offering a more efficient solution to the problem [Fis22].

In summary, probabilistic planning approaches offer a robust framework for decision making in AD, accounting for uncertainties presented by the dynamic driving environment. However, they still face some challenges:

- **Computational Complexity**: As the size of the state, action, or observation space increases, the computational requirements grow, making it intractable for real-time decision making in complex driving scenarios.
- **Non-parallelizable Tree Search**: Tree searches in POMDPs are inherently sequential and may not be easily parallelized, limiting the speed-up gains from modern multi-core processors.
- **Scalability Issues**: Real-world driving environments have huge number of scenarios and environmental conditions, making these approaches less practical for comprehensive AD systems.
- **Difficulty in Imitating Human Behavior**: As POMDPs still rely on the underlying reward function, replicate human-like driving behavior remains a challenge.

2.2.2 Reinforcement Learning

RL has emerged as a powerful tool for decision making in complex environments, offering the potential to learn optimal policies from interactions with the environment. In the context of AD, RL provides a framework to learn driving behaviors that can adapt to the dynamic and uncertain road environments. However, the application of RL to AD presents unique challenges and considerations, which have been addressed in various ways by the research community.

Solving Scenarios with Deep Reinforcement Learning

Deep RL combines the power of neural networks with RL to handle high-dimensional state spaces in driving scenarios. For instance, the on-ramp merge scenario, a challenging task due to the interactive nature of merging with highway traffic, has been tackled using deep RL techniques. A Long Short-Term Memory (LSTM) architecture was employed to model the interactive environment, feeding into a Deep Q-Network (DQN) to determine the optimal driving policy [Wan17]. Planning with high-level actions is incorporated in [Tri20] to increase the safety of the merging policy, allowing the learning component to be agnostic to the low-level control scheme. Passenger comfort is further increased by combining model predictive control with RL in [Lub21].

To ensure robust lane change decisions under observation uncertainties, an observation adversarial RL approach was proposed, optimizing lane change policies while keeping policy variations within bounds [He23]. Overtaking is also challenging as it requires long-horizon planning capability and is tackled in [Liu20] with DQN. Similarly, deep RL has been applied to navigate occluded intersections, surpassing the performance of heuristic approaches in task completion time and goal success rate [Ise18]. The challenge of negotiating behaviors at intersections has also been addressed using DQN, where the policy adapts vehicle speed based on observations of other vehicles' distances and speeds [Tra18].

Safety Considerations

Safety remains paramount in AD. A multi-agent RL approach was proposed to ensure both driving comfort and safety. This approach decomposes the problem into a *Policy for Desires* and trajectory planning with hard constraints. The introduction of an *Option Graph* further reduces the effective horizon, optimizing gradient estimation [Sha16]. Another approach combined RL with formal safety verification, ensuring only safe actions are chosen. This method achieved fast learning rates without causing collisions, outperforming rule-based systems [Mir18]. The challenge of navigating intersections was addressed using

a modular decision making algorithm, incorporating a safe RL algorithm with model-checking for safety guarantees and a belief update technique for robustness against perception errors [Bou19].

Incorporating risk measure in learning process is another way of improving safety performance. In order to address scalability and safety, a constrained RL approach was proposed, automating the trade-off between risk and utility without requiring reward parameter tuning in [Kam22]. A risk-aware DQN approach was introduced for navigating unsignalized occluded intersections, incorporating risk prediction into the Q-network for safer policies [Kam20].

Architectural Innovations and Scalability

Handling variable-sized inputs, especially in dynamic environments like AD, requires specialized architectures. The limitations of fully-connected neural networks and other established approaches were addressed by employing the structure of Deep Sets in off-policy RL, which showed better generalization to unseen situations [Hue19]. The combination of planning and deep RL has also been explored, using the AlphaGo Zero algorithm extended to a domain with a continuous state space, outperforming baseline methods [Hoe20]. Another approach combined RL with game theory, using a training curriculum based on level-k behavior, resulting in policies robust to model discrepancies [Bou20].

Offline RL and Incorporating Human Demonstrations

Offline RL, learning directly from offline datasets, is especially relevant for AD due to the feasibility of collecting real-world driving data. An offline RL benchmark for AD was introduced, deploying popular offline algorithms and analyzing their performance under different datasets [Fan22]. Human demonstrations offer valuable insights into safe and efficient driving behaviors. By incorporating human demonstrations into the RL-based decision making strategy, the safety of RL decisions was significantly improved, outperforming other learning-based strategies [Wu22].

Discussion

RL has shown promise in various domains, including games and robotics. However, deploying RL approaches in real-world AD systems presents several major challenges:

- **Safety Concerns**: RL agents learn by interacting with environments, which often involves making mistakes to learn optimal strategies. However, in real-world driving, mistakes can be catastrophic.
- **Sim-to-Real Transfer**: While training RL agents in simulators is safer and more scalable, transferring learned policies to real-world scenarios is non-trivial due to the reality gap. Simulated environments can not perfectly replicate real-world conditions.
- **Modeling Human Behavior**: A human-like driving behavior is hardly achievable just by designing a reward function.
- **Interpretable and Explainable AI**: For safety and regulatory reasons, it is crucial to understand and explain the decisions made by AD systems. RL policies, especially those derived from deep learning, can be black-boxes, making them hard to interpret.

2.2.3 Imitation Learning

In the context of AD, IL involves learning from human driving trajectories to make decisions that closely resemble human behavior. This section delves into the various methodologies in IL that have been developed for AD.

Behavior Cloning

BC for AD is often an end-to-end approach where the system learns to map observations directly to actions using supervised learning. The primary advantage of this method is its simplicity, as it directly learns from expert demonstrations without the need for reward engineering. Direct mapping from observations (e.g. input camera images) to actions abandons many intermediate modules as well. Success in complex scenarios, especially in lateral motion control [Sha18]

has been demonstrated. As seen in the NAVNet architecture [Sak19], it also possesses the ability to handle temporal dynamics and convolutional perceptual representations. However, according to [Cod19], end-to-end BC has the following shortcomings:

- Requiring huge amount of training data.
- Susceptibility to dataset bias and overfitting.
- Absence of causal modeling.
- Training instabilities that may hinder real-world deployment.

Inverse Reinforcement Learning

IRL seeks to deduce the latent reward structures that experts implicitly follow, offering a deeper understanding of the motivations behind expert decisions. This approach has been applied in various contexts within the field of AD.

Some research has delved into the hierarchical nature of driving decisions, encompassing both discrete and continuous choices. For instance, a probabilistic prediction approach based on hierarchical IRL has been proposed, which has shown promise in scenarios like ramp-merging [Sun18].

Personalizing AD to individual preferences is a growing area of interest. The Personalized Adaptive Cruise Control (ACC) system learns driver-specific car-following preferences using model-based maximum entropy IRL [Zha22]. Additionally, leveraging naturalistic human driving data, models have been proposed that focus on discrete latent driving intentions, offering a more realistic representation of human driving behavior [Hua20, Xu23]. In the field trajectory planning, the exploration of cost functions in IRL has been pivotal in understanding their suitability for mimicking human behavior across various scenarios [Nau20b]. Furthermore, the integration of BC with IRL, as proposed in a preprocessing framework for expert examples, has shown potential in enhancing the quality of expert demonstrations, leading to more accurate reward functions [Li21].

Ensuring safety in IRL is paramount. Research like Constrained Soft Reinforcement Learning has addressed maximum entropy IRL in constrained environments, emphasizing the importance of safety [Fis21]. Cooperative trajectory planning methods, combined with Maximum Entropy IRL and MCTS, have been proposed to learn reward models that mimic expert cooperative trajectories [Kur22]. Notably, DriveIRL has demonstrated the real-world potential of IRL in dense urban traffic scenarios [Pha23].

Adversarial Methods

Adversarial methods in IL involve a game-theoretic approach where a policy is trained to imitate expert behavior while a discriminator attempts to differentiate between the policy's actions and those of the expert.

GAIL stands as a fundamental approach in adversarial IL for driver behavior modeling. It has showcased significant robustness compared to direct BC [Sac22]. Variations of GAIL, such as those augmented with BC, have been proposed for urban driving scenarios, emphasizing the method's versatility and efficiency in diverse contexts [Kar21]. Another notable advancement is the use of GAIL for modeling driver behavior in multi-agent settings, highlighting the method's capability to capture complex interactions among multiple agents [Bha23].

Adversarial IL's adaptability is evident in its ability to learn effectively from imperfect demonstrations. By leveraging confidence scores, methods like two-step importance weighting IL and GAIL with imperfect demonstration and confidence have been developed to address challenges posed by sub-optimal demonstrations [Wu19]. Furthermore, novel techniques to enhance performance in the presence of imperfect demonstrations are introduced in [Hu23], underscoring the method's resilience.

To enhance the safety, the Safety-Aware Hierarchical Adversarial Imitation Learning [Jam23] approach employs hierarchical adversarial IL tailored for urban environments, demonstrating the method's potential in handling complex urban driving scenarios.

Discussion

In summary, while BC offers a direct and intuitive approach to imitation learning, it can face challenges in generalization and robustness. IRL, with its focus on underlying reward structures, provides a deeper understanding but can be computationally intensive. Adversarial Methods, especially GAIL, strike a balance by leveraging game-theoretic principles to achieve robustness and generalization, making them promising for real-world applications. However, safety is not always guaranteed and traffic rules can hardly be integrated in these approaches, despite special hierarchical safety design [Jam23].

2.3 Safety in Decision Making for Automated Driving

Safety is paramount in the field of AD. As vehicles become increasingly autonomous, the decision-making processes guiding their actions must be robust, reliable, and above all, safe. Ensuring safety in decision making is not just about preventing collisions; it is about instilling trust in passengers, other road users, and the wider public. This trust hinges on the vehicle's ability to make decisions that are not only technically sound but also align with human expectations and traffic norms. From navigating complex urban environments to handling high-speed highway scenarios, the vehicle's decision-making system must be equipped to assess risks, anticipate uncertainties, make informed choices that prioritize safety, and have a fall-back plan [Wan20c]. In the subsequent sections, I delve deeper into the intricacies of risk assessment and the verification of safety, two critical pillars that uphold the safety standards in AD decision making.

2.3.1 Risk Assessment

Risk assessment is a crucial component in the decision-making process for AD. It quantifies the potential dangers associated with a given action or trajectory,

enabling the system to make decisions that prioritize safety. Traditional metrics, often referred to as Time-To-X metrics, have been widely used to evaluate the risk of potential collisions [Lef14]. These include:

Time to Collision (TTC): Defined as the time it would take for two vehicles to collide if they continue at their current speeds and on their current paths. Mathematically, it is given by:

$$\mathrm{TTC} = \frac{d}{v_{\mathrm{ego}} - v_{\mathrm{obstacle}}} \tag{2.9}$$

where d is the distance between the ego vehicle and the obstacle, and v_{ego} and v_{obstacle} are their respective velocities.

Time to React (TTR): Represents the time available for the vehicle to take an evasive action before a potential collision. It can be derived from various sensor data and contextual information.

Time Headway (THW): The time it would take for the ego vehicle to reach the position of the vehicle in front, given by:

$$\mathrm{THW} = \frac{d}{v_{\mathrm{ego}}} \tag{2.10}$$

While these metrics provide a foundational understanding of collision risks, they might not capture the complexities and uncertainties inherent in real-world driving scenarios, especially in dynamic and occluded environments. Advanced approaches have emerged to address these challenges.

A group of studies focuses on the concept of predictive risk maps or safety envelopes [Dam15, Pie18, Ber22, Pie19], which evaluate future behavior alternatives in terms of predictive risks. For instance, the work in [Dam15] introduces predictive risk maps that indicate the risk associated with a certain ego-car trajectory at different predicted times. Similarly, the idea of navigating based on occupancy risk is explored in [Pie18], where the density and motion of objects are mapped to an occupancy risk, allowing agents to adjust their interactions based on chosen risk levels. Parameters for quantifying the risk

levels are learned from real-world datasets [Pie18] and this approach is further improved in [Pie19].

Another significant direction in risk assessment is the emphasis on probabilistic motion prediction. The study in [Kim18] presents an algorithm that assesses collision risks for local path candidates by predicting the motion of surrounding vehicles based on lane probabilities. This concept of lane-based risk assessment is further explored in [Woo18, Kum18], where advanced adaptive cruise control systems and lane merging frameworks are proposed. These systems aim to minimize collision risks by predicting the intentions and future actions of surrounding traffic participants.

A few studies have delved into the analytic computation of collision probabilities, e.g. [Phi19, Alt21], gains time efficiency compared to MCS, while some others [Wan20b] approximates long-term collision risk by inferring from the risks of short-term MCSs. The work in [Fre20] takes a different approach by deriving a risk approximation framework directly in continuous time, addressing the limitations of discrete-time dynamics.

Lastly, the challenge of real-time risk assessment, especially in scenarios with occlusions, is addressed in [Yu20, Taş18, McG19, Wan20a]. These studies emphasize the importance of considering visibility and interactions in risk assessment, proposing methods that ensure collision-free navigation even in highly occluded environments under certain assumptions.

In conclusion, while traditional risk metrics provide a baseline for understanding collision risks, the dynamic nature of real-world driving scenarios necessitates the development of advanced risk assessment methodologies. The integration of predictive risk maps, probabilistic motion prediction, and occlusion-aware techniques, as highlighted in the aforementioned studies, paves the way for safer and more efficient AD systems.

2.3.2 Verification of Safety

Safety verification is a cornerstone in the development and deployment of AD systems. While probabilistic collision risk assessments provide insights into

potential hazards, they cannot offer formal guarantees of safety. This is where formal safety verification approaches come into play, ensuring that the decision-making processes of AVs adhere to rigorous safety standards, even in complex and uncertain environments. There are mainly two research directions in providing formal safety verification: *reachable-set analysis* and *RSS-based safety analysis*.

Reachable-set Analysis

Reachable-set analysis [Alt14, Alt16] is a powerful tool that provides over-approximations of all possible states a system can reach, ensuring that an AV's decisions remain within safe bounds. The work by [Alt16] introduces a method that over-approximates all possible occupancies of surrounding traffic participants over time, allowing for formal guarantees of collision-free maneuvers. Similarly, the study in [Son18] presents a worst-case analysis of the TTR, providing a deterministic upper bound for this critical metric. Addressing the challenges posed by occlusions, [Orz18] introduces a safety verification method that uses the vehicle's field-of-view and a map to identify potentially hidden obstacles, ensuring safety even in scenarios with limited visibility.

RSS Safety and Extensions

The RSS model, introduced by [Sha17], provides a formalized approach to safety assurance, focusing on defining and verifying safe states for AVs. This model has inspired several extensions and refinements. For instance, [Orz19] integrates RSS-motivated safe-states with reachable-set analysis to ensure safety in complex traffic scenarios. The research in [Nau21] delves into the parameters of RSS, aiming to strike a balance between safety and traffic flow. Further enhancing the RSS model, [Sid22] presents a framework that calculates minimum safe inter-vehicular distances for various control policies. The work by [Has23] introduces a goal-aware extension of RSS, utilizing program logic to handle complex planning scenarios. Lastly, addressing the challenges posed by perception uncertainties, [Ber22] offers a probabilistic approach on top of

the original deterministic RSS approach to calculate safety envelopes, allowing for adjustable safety and performance based on chosen risk levels.

Discussion

The strength of reachable-set analysis lies in its ability to account for the worst-case scenarios, ensuring that no potential state is overlooked. However, this approach can often be overly conservative, potentially leading to inefficient driving behaviors and hindering the smooth integration of AVs into existing traffic flows. On the other hand, the RSS model, emphasizes the responsibility of safety and adherence to traffic rules, instead of ensuring absolute safety. This results in a less conservative and more practical safety framework, better suited for dynamic and unpredictable traffic environments. Furthermore, the adaptability and extensibility of the RSS model, as evidenced by its various refinements and extensions to different scenarios, make it a more scalable solution.

3 High-level Decisions under Safety Constraints

Following the objectives of this study and an overview of the state-of-the-art approaches, this chapter delves into the specifics of my proposed method. Prior to making any decision, it is of paramount importance for the decision-making system to ensure safety and comfort of passengers, as well as comply with traffic regulations and avoid collisions with other vehicles and obstacles. The primary contribution of this section is the formulation of safety requirements for a decision process that integrates diverse traffic rules and uncertainties.

To begin, an overview of my overall decision-making process is provided in Section 3.1, in which the inputs and outputs of the pipeline are briefly introduced, and simplifications and assumptions are discussed. Subsequently, in Section 3.2, I present a comprehensive discussion of the underlying safety concept for different types of roads, intersections, and possible occlusions. In order to avoid overly cautious behavior, safety constraints can be gradually relaxed as perception and scene understanding capabilities improve, as discussed in Section 3.3. Finally, in Section 3.4, I propose a representation of high-level actions for various scenarios and derive rule-based policies that are provably safe based on the aforementioned safety concept.

3.1 Model Assumptions and Overview of the Approach

AD systems typically rely on sensors to perceive the environment. However, due to physical limitations of the sensors, the resulting measurements can be

noisy and the environment only partially observed. As a result, the perception module of AD systems can only provide probabilistic distributions that represent possible ranges of object states, rather than revealing ground-truth states. My proposed decision-making module is assumed to receive these uncertain states of surrounding agents and obstacles, with given distributions. Regarding agents in occluded roads, further assumptions will be introduced in later chapters.

In addition to reactive decisions, it is desirable for AD systems to estimate hidden states of surrounding traffic participants, such as intended routes or lane change probabilities, in order to make proactive decisions. This corresponds to the *scene understanding* or *prediction* discussed in Section 1.1. For evaluation of my approach, I propose proof-of-concept *scene understanding* approaches in this work, which can be replaced by any upstream modules. Depending on the scenarios and traffic rules, the intentions of other agents can be dependent or independent of the ego vehicle's decision. For example, in an on-ramp merging scenario, the willingness of relevant vehicles to cooperate with the merging vehicle may depend on its approaching style. In contrast, at intersections where vehicles' priority must be ensured, crossing decisions are typically made independently of non-prioritized vehicles. Therefore, I suggest selecting an interactive behavior modeling or a single-shot intention estimation depending on the specific scenario.

Many researchers strive to achieve AD without relying on pre-existing High-Definition (HD) maps by using powerful perception modules to reconstruct the environment, e.g. via perceived road boundaries, traffic signs, and traffic lights [Wen21, Mey18]. However, this approach is challenging, and in my work, I assume the availability of a HD map that contains all necessary information about traffic rules and road layout, such as the lanelet2 framework [Pog18]. Additionally, I assume the availability of a self-localization module that provides the state of the ego vehicle with negligible uncertainty.

The goal of the decision-making module is to provide a safe, efficient and comfortable high-level decision that is afterwards processed and refined by a trajectory planner. A feasible way is to frame the decision into planning targets and constraints. However, in order to ensure at least one local optimum from the

planner, the kinematic (e.g. maximum turning radius) and dynamic constraints (e.g. maximum longitudinal acceleration) of the vehicle need to be accounted for beforehand. Assuming a successful generation of the trajectory by the planner, negligible control errors are expected as well.

My work assumes that the AV is equipped with sensors that provide a 360° coverage of the surrounding environment within a certain distance range. Any obstacle that has a similar height to the AV, such as other vehicles and walls, is assumed to limit the Field of View (FoV) of the AV and create occlusions, inside of which no object is perceivable anymore. However, existing objects within the FoV are assumed to be detected. Dealing with possible ghost objects [Tas20], i.e. false positives, is possible but is not within the scope of this work.

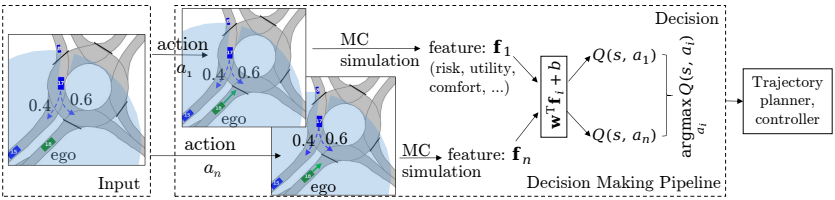


Figure 3.1: Overview of the decision making pipeline.

The comprehensive methodology of my approach, depicted in Figure 3.1, proceeds as follows: in a given scenario, multiple high-level action candidates are conceived, conforming to the HD map and traffic regulations. Each high-level action undergoes a safety audit based on the criteria delineated in Section 3.2. Assuming the ego vehicle chooses an action a_i from the current state s, and adheres to a policy that elects a_i with a probability of $\pi(a_i|s) = \pi(a_i) = 1$ that is independent of the future states, I employ MCS to generate prospective rollouts. These rollouts involve repeated queries of estimated intentions or interactive behavioral models (introduced in later chapters) of perceived traffic participants, accounting for all occlusion possibilities. After extensive rollouts for each policy (action), certain features can be ascertained which provide insight into the merits of implementing that specific policy (action). A linear function, harboring unknown parameters (a vector of weights $\mathbf{w}$ and bias b), is used to estimate the action value (also known as Q-value)

$$Q(s, a_i) = \mathbf{w}^\top \mathbf{f}(s, a_i) + b = \mathbf{w}^\top \mathbf{f}_i + b \tag{3.1}$$

for every candidate action a_i under the present state s, derived from their features $\mathbf{f}_i$. The final decision corresponds to the action yielding the maximal Q-value. Since all actions have the same bias b, I eliminated it, i.e. $Q(s, a_i) = \mathbf{w}^\top \mathbf{f}_i$. Ultimately, the aim is to update the weights $\mathbf{w}$, such that the Q-value of the most human-like action surpasses other actions.

The rationale behind opting for feature vectors to represent the Q-value and learn the weights, rather than directly learning the policy distribution $\pi(a|s)$ based on human driving trajectories as per conventional behavioral cloning approaches, lies in the incomplete coverage of the states in the dataset. Even if the distribution $P(a|s)$ can be trained reliably for states encompassed in the provided dataset, the resulting policy would be untrained in other states. Utilizing a feature function facilitates generalization to unseen states, given that the feature space dimensionality is significantly smaller than the state space [Koc22].

3.2 Ensuring Safety with Extended Responsibility-sensitive Safety (RSS)

One of the key challenges for universal acceptance of AVs by the public is driving without putting human lives at risk. The goals of convenience and safety are often perceived as contradictory, where maximizing one could potentially damage the other [Wan23a, Wan23b]. As public roads are shared with numerous other traffic participants, such as vehicles, cyclists, and pedestrians, AVs must not only drive by maximizing their own objectives, such as high speed, but also take into account the risk they pose to others. However, AVs should at least achieve a certain level of efficiency to accomplish their goals within an acceptable time. Assuming worst-case behaviors of all traffic participants could result in the ego vehicle driving at extremely low speed or even standing still. Moreover, as stated in [Sha17], guaranteeing that the ego vehicle will

never be involved in a collision is intractable in real traffic, regardless of the speed. Therefore, reducing the speed is not an ideal way to ensure safety.

After reviewing many approaches for safety verification in Section 2.3.2, such as RSS [Sha17] and reachable-set analysis [Alt14], I propose the following objectives for an ideal safety verification approach in real road traffic:

1 Safety needs formal guarantees. As argued in [Nau20a], any on-road or simulation-based safety validation with huge kilometers is undesired, because any code change in the planning pipeline needs a complete reevaluation that is extremely expensive.

2 Safety verification approaches need to be real-time capable, to prevent the vehicle from entering an unsafe state before completing the verification of its output behavior.

3 Instead of guaranteeing that the ego vehicle will not be involved in a collision, the guarantee is never to cause a collision. This can be achieved by incorporating traffic rules that introduce fixed driving rules, such as traffic lights, and the sense of priority, such as the right-of-way rule. However, some traffic rules[1] cannot be mathematically expressed in a machine-readable format. Thus, the necessary translation of traffic rules into mathematical formulations can be proposed to enable their use in safety-critical applications.

4 While operating within the bounds of traffic regulations, safety verification should additionally aim at achieving an overall smooth traffic flow but not overly-conservative driving behaviors.

Motivated by similar objectives, [Nau20a] extends the RSS concept [Sha17] and includes the consideration of traffic rules at intersections with crossing and merging traffic, and parallel lanes. In this work, I further enhance this approach

[1] For example, “An [nicht besonders geregelten] Kreuzungen und Einmündungen hat die Vorfahrt, wer von rechts kommt.”(§8 I StVO)

by considering: a wider range of traffic participants, occlusions, the safety challenges posed by close consecutive intersections, and other limiting factors, such as the maximum reachability of vehicles.

In this work, I assume a prior HD map that embeds all traffic rules and road geometries. My safety concept is mainly based on a lane-wise road topology, where lateral safety is ensured by avoiding collisions with lane boundaries when driving on a single lane that does not interfere with any other lane. I first review longitudinal safety on a single lane in Section 3.2.1, then extend to parallel lanes in Section 3.2.2, where lane changes occur, and to intersecting lanes in Section 3.2.3, where priority rules apply. In addition, I consider the impact of sensor occlusions and limited FoV in Section 3.2.4.

3.2.1 RSS Safety for Single Lane

The RSS concept defines basic longitudinal safety in a leader-follower setup on a single lane, as presented in [Sha17]. It emphasizes the need for the follower to maintain a minimum safe distance from the leader, which ensures that the follower will not collide with the leader even in the following worst-case scenario: The predecessor, previously traveling with a velocity of v_{lead}, decelerates with the maximum deceleration of $a_{\text{max,dcc,obj}}$ until it comes to a complete stop. During a reaction time of ρ_{ego}, the ego vehicle, which was previously traveling at v_{ego}, accelerates with a maximum acceleration of $a_{\text{max,acc,ego}}$, and then brakes with a assured deceleration of $a_{\text{max,dcc,ego}}$.

I propose to deviate from this assumed worst case and do not assume the acceleration phase of ego vehicle during the reaction time. In reality, during usual car-following, the ego vehicle will not apply $a_{\text{max,dcc,ego}}$ unless the distance to the front vehicle is sufficiently larger than its desired distance headway. Another case where the ego might execute $a_{\text{max,dcc,ego}}$ is when the traffic starts to flow and the leading vehicle and ego vehicle accelerate both with their maximum capabilities. However, in this case, the ego vehicle is usually highly concentrated and the reaction time is smaller than the average human performance, which balances the more traveled distance during the reaction phase with $a_{\text{max,dcc,ego}}$. Therefore, I assume the ego vehicle maintains its previous

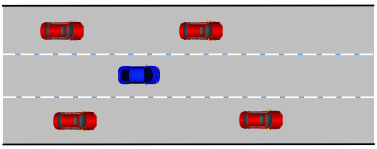
velocity v_{ego} during the reaction phase. With this assumption, the minimum safe distance can be computed as

$$d_{\text{safe}} = \max\left((v_{\text{ego}} - v_{\text{lead}})\rho_{\text{ego}} + \frac{v_{\text{ego}}^2}{-2a_{\text{max,dcc,ego}}} - \frac{v_{\text{lead}}^2}{-2a_{\text{max,dcc,obj}}}, 0\right) \quad (3.2)$$

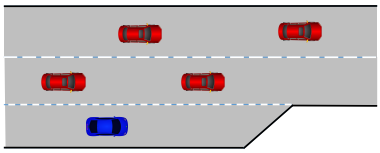
In some cases, e.g. due to a very large velocity of the predecessor, the required safe distance can be negative. The max() function prevents it and generates a non-negative safety distance.

3.2.2 RSS Safety for Parallel Lanes

In this section, I present the safety approach in the context of parallel lanes for unidirectional traffic, which permit lane changes between them, typically marked by dashed lines. Parallel lanes can be classified into two types - multi-lane configurations, wherein drivers have the freedom to change lanes, and entrance lanes, which require mandatory merging, as depicted in Figure 3.2.



(a) Parallel lanes where lane changes are possible



(b) Parallel lanes where a mandatory merging for the blue vehicle is demanded.

Figure 3.2: Two types of parallel lanes where lane change and merging safety needs to be ensured.

A safety concept for cut-ins is presented in [Nau20a], where a constant acceleration $a_{\text{max,acc,ego}}$ of the cut-in vehicle is assumed. However, this is not always

possible and thus I propose the following safety rule for lane change and merging: Upon completion of merging[1], the merging vehicle should accelerate with some assumed acceleration $a_{\text{max,acc,ego}}$, which might be limited due to other factors[2]. From the moment of completed merging, a reaction time of ρ_{obj} is allotted to the prioritized following vehicle on the target lane, which then decelerates with $a_{\text{soft,dcc,obj}}$. The merging is considered as safe and not significantly impeding the prioritized vehicle, when their distance is not less than d_{safe} from the merging time until a stable state.

An analytical solution of the initially required safe distance of the merging vehicle to the prioritized following vehicle $d_{\text{safe,follow,init}}$ can be found in [Nau20a] with the assumption that the merging vehicle can keep accelerating with $a_{\text{max,acc,ego}}$ after merging. However, in the presence of aforementioned factors, $a_{\text{max,acc,ego}}$ of merging vehicle is not always guaranteed, and thus an analytical solution might not exit.

I propose to utilize numerical simulation to examine the safety of the merging or lane change, in case that an analytical solution is not available. After finishing lane change, I simulate the following scenario forward with a small time interval[3]: The leading vehicle on the target lane continues with constant velocity, the following vehicle on the target lane decelerates with $a_{\text{soft,dcc,obj}}$ after a reaction time of ρ_{obj}, and the merging or lane change vehicle accelerates with its maximum possible acceleration $a_{\text{pos,acc,ego}} < a_{\text{max,acc,ego}}$ such that $d_{\text{safe,lead}}$ to the leading vehicle on the target lane is not violated. A safe and non-disruptive merging or lane change can be characterized by the preservation of a minimum safe distance, denoted as $d_{\text{safe,follow}}$, between the ego vehicle and the following vehicle on the target lane from the moment of lane change until a critical time point[4].

[1] Merging or lane change is considered as complete, when the vehicle has left the lane with a certain proportion, e.g. $\frac{1}{2}$ of the vehicle's geometry.

[2] Such as reaching the speed limit or the presence of a leading vehicle on the merging lane.

[3] Such that vehicles do not pass through each other's geometry between two time steps.

[4] The time point where the distance between the ego vehicle and the following vehicle on the target lane increases faster than the required safe distance.

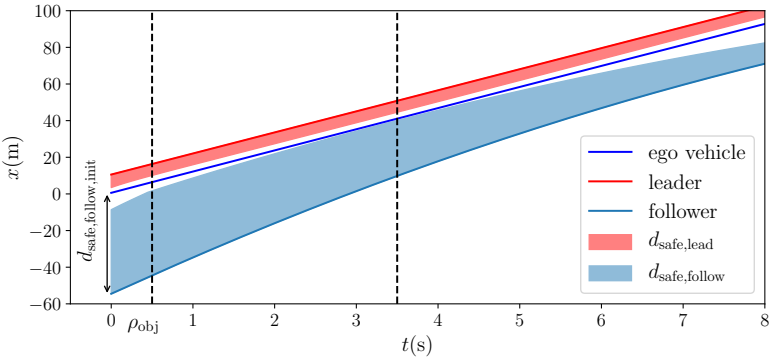


Figure 3.3: An example of numerical simulation to check lane change or merging safety.

Figure 3.3 illustrates an example of the numerical simulation with 0.2 s time interval of a critical merging or lane change. The ego vehicle finishes lane change at time 0 at $x = 0\,\text{m}$, where the leading vehicle and the following vehicle on the target lane initially locate at 15 m and −55 m. From this time point, the follower on the target lane is assumed to decelerate with $a_{\text{soft,dcc,obj}} = -1.2\,\frac{\text{m}}{\text{s}^2}$ after a reaction time $\rho_{\text{obj}} = 0.5\,\text{s}$. The initial distances after lane change to the leader and follower are apparently bigger than d_{safe} at time 0. However, the lane change is still critical, because the simulated distance to the following vehicle at around 3.5 s just meets the required safe distance $d_{\text{safe,follow}}$. Therefore, $d_{\text{safe,follow,init}} > d_{\text{safe,follow}}$ is initially demanded, in order to not force the following vehicle on the target lane to brake more than $a_{\text{soft,dcc,obj}}$. In this example, $t = 3.5\,\text{s}$ is the critical time point.

To assess the practical feasibility of the proposed safety concept, I conduct a thorough evaluation on real-world driving data to quantify the percentage of human drivers who adhere to this rule. For this purpose, I leverage the HighD [Kra18] and ExitD [Moe22] datasets. To facilitate a fair comparison, I select a parameter set comprising of $a_{\text{soft,dcc,obj}} = -1.2\,\frac{\text{m}}{\text{s}^2}$ (considered a comfortable value in [Hob77]), $\rho_{\text{ego}} = 0.5\,\text{s}$ (anticipated for AVs in [Xu21]), and $\rho_{\text{obj}} = 0.7\,\text{s}$ (recommended in [Mar16]). The results of this analysis are presented in Table 3.1, which outlines the percentage of safe lane changes observed in the datasets for different combinations of $a_{\text{max,dcc,ego}}$ and $a_{\text{max,dcc,obj}}$.

Table 3.1: Percentage of RSS safe lane changes (from 12380 lane changes in HighD dataset) and merges (from 4604 on-ramp merges in ExitD dataset) with certain deceleration parameters. Both datasets are recorded on German highways. Note that $a_{\text{max,dcc,obj}} \leq a_{\text{max,dcc,ego}}$ is assumed.

Ratio of safe lane changes		$a_{\text{max,dcc,obj}}(\frac{\text{m}}{\text{s}^2})$			
		-6	-8	-10	-12
$a_{\text{max,dcc,ego}}(\frac{\text{m}}{\text{s}^2})$	-6	0.877	**0.775**	0.563	0.511
	-8		0.886	**0.761**	0.677
	-10			0.889	**0.804**
	-12				0.885
Ratio of safe merges		$a_{\text{max,dcc,obj}}(\frac{\text{m}}{\text{s}^2})$			
		-6	-8	-10	-12
$a_{\text{max,dcc,ego}}(\frac{\text{m}}{\text{s}^2})$	-6	0.928	**0.846**	0.723	0.662
	-8		0.943	**0.909**	0.860
	-10			0.949	**0.933**
	-12				0.955

Upon analyzing the results presented in Table 3.1, it becomes evident that with the setting where the maximum possible deceleration of the leader $a_{\text{max,dcc,obj}}$ is $-2\ \frac{\text{m}}{\text{s}^2}$ more than the follower $a_{\text{max,dcc,ego}}$ [1], only around 80% of all lane changes in the HighD dataset and around 90% of all merges in the ExitD dataset satisfy the RSS safety criterion. Notably, the percentage of unsafe lane changes is significantly lower than the number of recorded accidents, which were found to be zero in both datasets. This discrepancy suggests that there may be a mismatch between the proposed RSS safety concept and human driving consensus. It is argued in [Nau21] that the majority of human RSS violations can be attributed to the additional assumptions that humans tend to apply while driving. This intriguing observation warrants further investigation, which will be carried out in detail in the forthcoming Section 3.3.

[1] It is suggested to assume that the leader has more braking capability as the follower to have a more strict safety condition.

3.2.3 RSS Safety for Intersecting Lanes

While the preceding subsection addressed longitudinal safety in the context of parallel lanes, the present subsection focuses on the safety of lanes that intersects. To this end, [Nau20a] proposes a classification scheme that distinguishes between three distinct traffic patterns - *crossing*, *merging*, and *diverging*. In cases of *diverging* traffic, wherein vehicles originating from the same lane continue on different lanes, the longitudinal safety principles discussed in Section 3.2.1 are deemed adequate. In contrast, I revise the safety approach for *crossing* and *merging* traffic scenarios, and subsequently introduce my proposed enhancements. It is important to note that my analysis focuses solely on scenarios governed by priority-based traffic rules and does not take into account specialized traffic rules, such as those pertaining to zipper merging or intersections controlled by traffic lights.

Safety for Merging and Crossing Conflict Zones

The term of a *conflict zone* serves as a fundamental basis for discussions related to safety in intersecting lanes, which is defined as the region of overlap between lanes that may result in potential conflicts. Figure 3.4 presents one example at an unsignalized intersection, where the ego vehicle should give way to the oncoming prioritized vehicles. Two of their possible routes intersect with the route (green dashed line) of the ego vehicle, resulting in a crossing and a merging conflict zone.

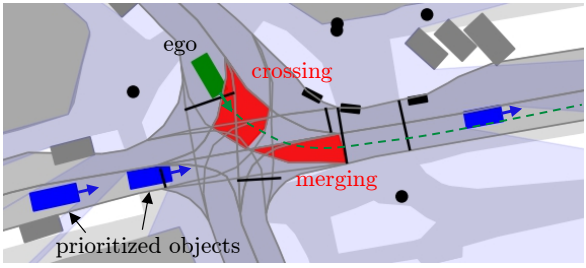


Figure 3.4: Example of crossing and merging conflict zones at a right-before-left intersection.

I reformulate the proposal in [Nau20a] with additional terms and supplemental constraints for better scalability. For a crossing conflict zone, the safety of the non-prioritized vehicle (ego) can be ensured when at least one of the conditions C_1 and C_2 is held:

- C_1: Ensure to be able to stop before the conflict zone.
- C_2: Ensure to safely pass the conflict zone.

To satisfy C_1, the ego vehicle needs to be able to stop before the conflict zone with less than its executable deceleration $a_{\text{max,dcc,ego}}$. There are two ways to satisfy C_2, which are:

- $C_2(a)$: At the time ego vehicle enters the conflict zone with its *maximum reachability*, e.g. maximum acceleration $a_{\text{max,acc,ego}}$ until the allowed speed limit v_{limit}, the prioritized vehicle is at sufficient distance, i.e. its required deceleration to stop in front of the conflict zone is acceptable, e.g. less than its $a_{\text{soft,dcc,obj}}$.
- $C_2(b)$: Ego vehicle can guarantee to have left the conflict zone with its *maximum reachability* for a predefined Time of Zone Clearance (TZC) $t_{\text{TZC,min}}$, before the prioritized vehicle can enter it with its *maximum reachability*, e.g. $a_{\text{max,acc,obj}}$ until its maximum velocity $v_{\text{max,obj}}$.

The TZC denotes the duration between the departure of the first vehicle from the conflict zone and the arrival of the second vehicle into it. Note that the maximum velocity $v_{\text{max,obj}}$ of other vehicles should be set to a realistic value which represents expected speeding (e.g. 110% of the speed limit v_{limit}) [Orz18].

In the case of merging conflict zones, C_1 still applies. However, the conflict zone cannot be fully traversed, as the length of overlap between two merging lanes can be unlimited. Once the merging is complete, the ego vehicle becomes the leading vehicle for the prioritized vehicle. If the prioritized vehicle can maintain a safe RSS following distance to the ego vehicle with only a slight deceleration, the merging maneuver is deemed non-impeding and courteous. As such, I propose C_3 as a replacement for C_2, which is essentially a rephrased version of the safety rule for lane change in Section 3.2.2:

- C_3: From the moment that the ego vehicle enters the conflict zone, as determined by its *maximum reachability*, until the critical time point, the prioritized vehicle maintains RSS safe following distance $d_{\text{safe,follow}}$ to the ego vehicle, with no more than $a_{\text{soft,dcc,obj}}$ deceleration performed.

Note that the *maximum reachability* of the ego vehicle and prioritized vehicles in C_2 and C_3 may be subject to various limiting factors, which will be discussed in subsequent sections. The computation of the critical time point was already discussed in Section 3.2.2. Although $t_{\text{TZC,min}} = 0\,\text{s}$ is sufficient to ensure safety, it may still be perceived as threatening by drivers of the prioritized vehicles. As a result, I must identify an acceptable value for $t_{\text{TZC,min}}$ to ensure that no overreactions occur on the part of the prioritized vehicles. In order to accomplish this, I conducted an analysis of 4057 crossing scenarios and the associated TZCs using data from the inD dataset [Boc20]. The TZC was computed for each vehicle based on the assumptions outlined in $C_2(b)$ when C_1 is no longer applicable. My findings indicate that 81.4% of vehicles crossed with a TZC of more than 0.5 s and 61.6% with more than 1 s. To not have overly conservative safety conditions, I select $t_{\text{TZC,min}} = 0.5\,\text{s}$. The RSS parameters employed for the AV are presented in Table A.2 in the column "normal".

Safety for Cyclists and Pedestrian Crossing

Pedestrians are only considered for intersecting-lane scenarios, e.g. at zebra crossing, but not in parallel-lane scenarios[1]. Otherwise, they are responsible for their own safety if the ego vehicle reacts properly, e.g. brakes in time.

I do not implement specific RSS safety rules for cyclists. Instead, cyclists driving on vehicle lanes are treated as vehicles but with different RSS parameters, such as maximum deceleration $a_{\text{max,decel}}$, due to their distinct dynamics. Cyclists located on walkways are treated as normal pedestrians.

[1] Exceptions are e.g. pedestrian zones, where the ego vehicle is traveling with sufficiently low velocity such that stopping with small braking distance is possible where consideration of RSS is not needed.

Zebra crossing belongs to crossing conflict zones, where C_1 and C_2 apply, prioritizing pedestrians instead of vehicles. As pedestrians have different dynamics, I reformulate C_2:

- $C_{2,p}$: Ego vehicle can guarantee to have left the conflict zone with its *maximum reachability* for a predefined TZC $t_{TZC,min}$, before the prioritized pedestrians can enter it with their *maximum reachability*.

The *maximum reachability* of pedestrians is defined as the ability to accelerate to a maximum velocity of $v_{max,p}$ with no delay and other limitations, taking into account their high mobility. In the inD dataset, the maximum recorded velocity of pedestrians is 15.07 $\frac{m}{s}$, which is considered an outlier. Therefore, I select $v_{max,p} = 5.02 \frac{m}{s}$ at the 99th percentile. This value is significantly higher than the 85th percentile speed of 2.39 $\frac{m}{s}$ reported in another study [Jai14].

To allow a better overview of the proposed RSS safety rules, Figure 3.5 illustrates the examination process for a single conflict zone.

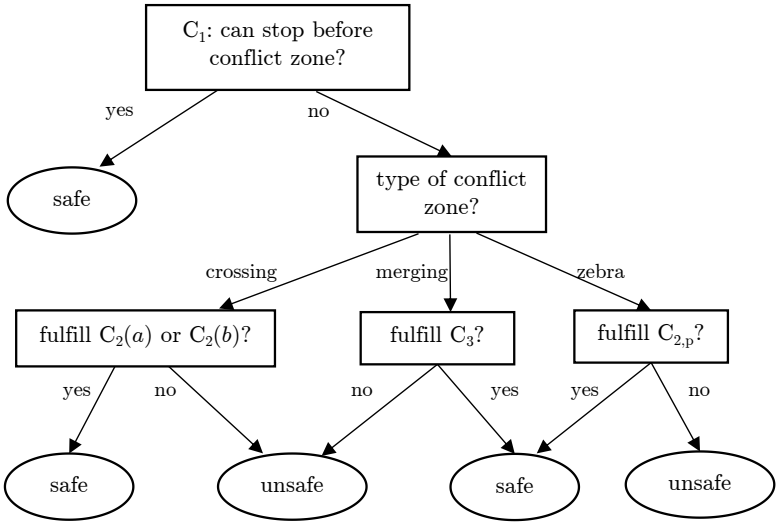


Figure 3.5: Overview of the RSS safety decision graph for a single conflict zone.

Stop Line, Regulatory Element and its Conflict Zones

I utilize lanelet2 maps where traffic rules are well-defined and encoded in Regulatory Elements (REs) [Pog18]. In the case of unsignalized intersections, right-of-way REs govern traffic rules and determine which vehicles have priority. Each right-of-way RE comprises the following components:

- *Stop line*: before which the yielding vehicle is recommended to stop, such that no conflict zone is impeded.
- *Traffic sign* (optional): e.g. yielding sign, stop sign, etc.
- *Yielding lanelet*: vehicles tending to pass through this lanelet should yield to prioritized traffic participants.
- *Right-of-way lanelets*: traffic participants that are possible to pass one of the right-of-way lanelets have priority over the yielding vehicles.

Using information obtained from the right-of-way RE, I can easily identify the prioritized traffic participants and their corresponding conflict zones in the scene. I accomplish this by iterating over all perceived traffic participants and checking whether they possess a possible *route*[1] that can traverse one of the right-of-way lanelets. If such a *route* exists, the traffic participant has priority. Corresponding conflict zones can be generated by calculating the overlapping area between all their possible *routes* and the ego vehicle's *route*.

For example, in Figure 3.4, the left turning lanelet of the ego vehicle is designated as the yielding lanelet, while the oncoming lanelets of the west and south arms serve as the right-of-way lanelets. If a vehicle approaches from the south arm, it will create two additional conflict zones with the ego vehicle.

Figure 3.6 illustrates the ego vehicle coming from the same direction but driving different routes, where the right-of-way lanelets and possible conflict zones differ considerably. When the ego vehicle plans to go straight (left figure), only the oncoming lanelet of the west arm will be the right-of-way lanelet, resulting

[1] A *route* means all the lanelets that can be used to a destination. They can be connected by a generic sequence of lane changes and successors.

in three possible conflict zones. If it tends to turn right (right figure), it is not required to yield to any other vehicles.

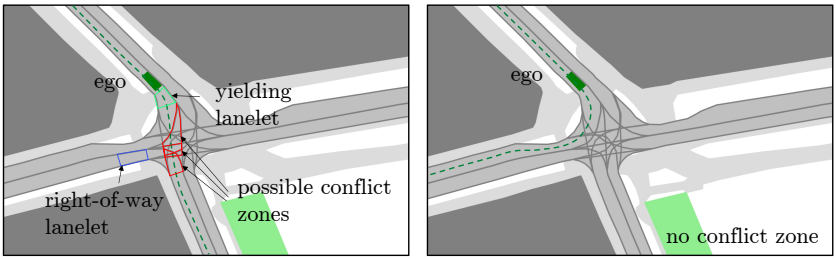


Figure 3.6: Possible conflict zones depending on the ego route.

Supplemental Constraints for Maximum Reachability of the Ego Vehicle

When checking if the ego vehicle can safely pass a conflict zone with C_2, $C_{2,p}$ and C_3, the term *maximum reachability* is introduced, which refers to the ability to accelerate to the maximum speed. I identify three important factors that can limit the *maximum reachability* of the ego vehicle.

The first factor is the presence of a leading vehicle. If the leading vehicle is possible to come to a complete stop at or shortly after the conflict zone, the time for the ego vehicle to reach or leave the conflict zone may become infinite. In such cases, none of C_2 (including $C_2(a)$ and $C_2(b)$), $C_{2,p}$ and C_3 can be satisfied.

The second factor is the presence of two consecutive stop lines or REs. In the example of Figure 3.4, there is one zebra crossing right after the conflict zones. If safely passing the pedestrian crossing is not guaranteed, the *maximum reachability* of the ego vehicle is additionally limited by "being able to stop before the zebra crossing", when assessing safety for the two red conflict zones. This ensures that when the ego vehicle is doing its best to reach or pass the red conflict zones to satisfy C_2, $C_{2,p}$ and C_3, its speed is still low enough to allow it to stop before the next pedestrian crossing.

The third factor is the physical and comfort limitation. The former includes friction limit which reduces the acceleration ability. Without this information

explicitly provided, the assumption is that the ego vehicle can always achieve its maximum longitudinal acceleration $a_{\text{max,acc,ego}}$. However, due to comfort reasons, the maximum velocity of the ego vehicle can be limited by the maximum lateral acceleration $a_{\text{max,acc,lat,ego}}$ on curvy roads.

In summary, the three supplemental constraints for *maximum reachability* of the ego vehicle are formulated as follows:

- If ego vehicle has a leading vehicle, the *maximum reachability* is limited by keeping the RSS safety distance $d_{\text{safe,lead}}$ to the leading vehicle, while the leading vehicle decelerates with its maximum deceleration $a_{\text{max,dcc,obj}}$.
- If the next stop line or RE is close to the current one, and safely passing all the conflict zones of the next stop line is not guaranteed, the *maximum reachability* is limited by being able to safely stop before the first conflict zone of the next stop line with $a_{\text{max,dcc,ego}}$.
- The *maximum reachability* is limited by not exceeding the velocity that reaches its maximum lateral acceleration $a_{\text{max,acc,lat,ego}}$.

RSS Safety for Traversing a Regulatory Element

With the term RE, I introduce the RSS safety condition for the non-prioritized vehicle traversing through a RE as follows: The safety of the non-prioritized vehicle (ego) can be ensured when at least one of the conditions $C_{1,\text{reg}}$ and $C_{2,\text{reg}}$ is held:

- $C_{1,\text{reg}}$: Ensure to be able to stop before the first conflict zone of the RE.
- $C_{2,\text{reg}}$: Ensure to safely pass all conflict zones of the RE simultaneously.

Figure 3.7 presents the examination of RSS safety for traversing a stop line or RE that has multiple conflict zones.

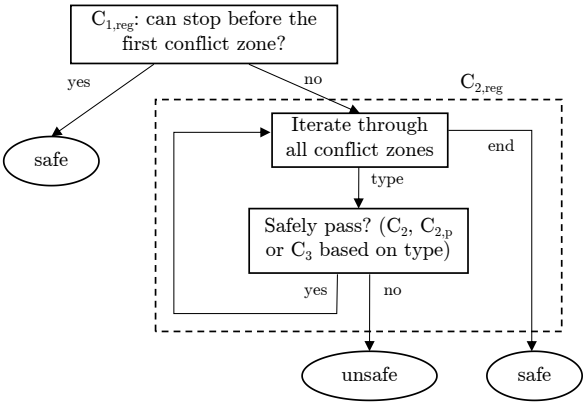


Figure 3.7: Overview of the RSS safety decision graph for a stop line or RE.

3.2.4 RSS Safety under Occlusions

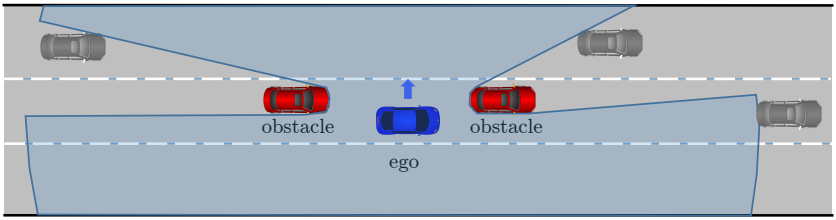
When driving on a single lane, the ego vehicle needs to be able to stop fully within its visible range d_{vis}, by assuming a static obstacle just behind the visible range. This additionally limits its maximum velocity $v_{\text{max,vis,ego}}$ besides the speed limit v_{limit}. An analytical solution of $v_{\text{max,vis,ego}}$ is given in [Nau20a]

$$v_{\text{max,vis,ego}} = a_{\text{max,dcc,ego}}\rho_{\text{ego}} + \sqrt{a_{\text{max,dcc,ego}}\rho_{\text{ego}}{}^2 - 2a_{\text{max,dcc,ego}}d_{\text{vis}}} \quad (3.3)$$

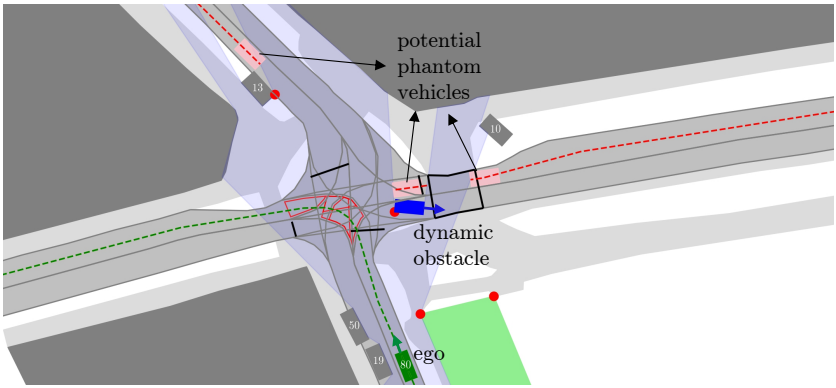
When performing lane changes or merging on parallel lanes, RSS safety verification requires observing the leading vehicle on the ego lane and target lane, and the following vehicle on the target lane, as proposed in Section 3.2.2. However, in some cases, the FoV may be occluded, resulting in potential dangerous lane changes. In such scenarios, worst-case phantom vehicles can be assumed to be located at the critical sensing edge, as shown in Figure 3.8a. In this case, a leading phantom vehicle with $0\,\frac{\text{m}}{\text{s}}$ velocity and a following phantom vehicle with $v_{\text{max,obj}}$ velocity can be assumed on the left lane at the sensing edge. However, with this worst-case assumption, the lane change may become almost

intractable. Therefore, it is recommended to perform lane changes and merging with a sufficient FoV on the target lane to ensure safety.

When approaching intersections under occlusion, prioritized traffic participants may be hidden. In [Orz18], it is proposed to over-approximate the possible states in occluded road sections and assume worst-case phantom vehicles located at the sensing borders. These phantom vehicles are assumed to travel with $v_{\text{max,obj}}$ and create several possible conflict zones for the RE. A crossing or pass decision can only be made when $C_{2,\text{reg}}$ is fulfilled, taking all the phantom prioritized vehicles into account. The resulting behavior is provably safe even in the worst-case scenario, but may be overly conservative [Wan21].



(a) Occlusions on parallel lanes that affect lane change and merging.



(b) Occlusions at intersections that create potential phantom vehicles on prioritized road section.

Figure 3.8: Occlusions on parallel-lanes and intersecting lanes.

Although occlusions can greatly limit the driving behavior that aims for comfort and convenience, human drivers have the ability to reason and speculate about what might be possible in such occlusions, allowing them to maintain safety without overreacting or significantly reducing their speed. To address occlusions in a more effective and less overly conservative manner, I will introduce an approach in the next subsection which was published in [Wan21].

3.3 Relaxing Safety Constraints with Better Perception and Scene Understanding

In pursuit of objective 4 in Section 3.2, I aim to develop a safety concept that maximizes the convenience of non-prioritized vehicles while not compromising the safety of prioritized traffic participants, resulting in a better traffic flow. To achieve this goal, my enhanced RSS safety concept is further relaxed when there is a better perception and scene understanding capability. This is particularly evident in cases where a larger visible distance d_{vis} is available, as it directly alleviates the constraints on the velocity limit $v_{\text{max,vis,ego}}$. However, I also consider other aspects that require a certain level of reasoning in order to further relax other safety constraints.

3.3.1 Visible Pre-predecessor

On a single lane, the ego vehicle shall not collide with its predecessor by assuming the worst-case maximum deceleration of it, as discussed in Section 3.2.1. However, as discovered in [Nau21] in the real data, RSS safety is violated by human drivers much more than the number of recorded traffic accidents. It is argued that most human RSS violations can be explained by the assumption, that an emergency deceleration of the predecessor occurs for other reasons. For example, human drivers rely on the behavior of the pre-predecessor to infer whether a deceleration of the predecessor is likely to occur. If the pre-predecessor is detected and tracked, human drivers can assume that a deceleration of the predecessor is likely caused by a deceleration of the pre-predecessor.

Moreover, if the pre-predecessor is very far away or does not appear within the sensing range, executing emergency brake by the predecessor without any reason is even against the regulation [Müh72] as it largely hinders the traffic flow. Therefore, the $a_{\text{max,dcc,obj}}$ assumption of the predecessor can be reduced depending on the distance to the pre-predecessor, which would further reduce the required safety distance between the vehicles.

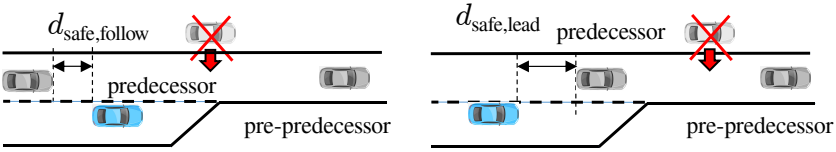


Figure 3.9: Examples of scenarios where the RSS safe distances $d_{\text{safe,follow}}$ and $d_{\text{safe,lead}}$ can be reduced by assuming predecessor brake less than $a_{\text{max,dcc,obj}}$ with visible pre-predecessor.

Figure 3.9 presents two example scenarios where RSS safety distances $d_{\text{safe,follow}}$ and $d_{\text{safe,lead}}$ can be reduced. In the left one, the ego vehicle (blue) is treated as predecessor and is assumed to brake less than $a_{\text{max,dcc,obj}}$ because the pre-predecessor is visible by the following vehicle on the target lane. The same explanation goes for the right one.

Note that this assumption can only be hold with a precondition: no other obstacles, e.g. pedestrians or vehicles from other lanes, can suddenly enter between predecessor and pre-predecessor (as shown by the red arrows in Figure 3.9) until the lane change is complete and all the vehicles restore a stable state[1]. If so, the lane change action should be aborted immediately.

With a visible pre-predecessor and the precondition met, I reduce the assumption of the maximum deceleration of the predecessor to $0.5a_{\text{max,dcc,obj}}$, which corresponds to $-5\,\frac{\text{m}}{\text{s}^2}$ for $a_{\text{max,dcc,obj}} = -10\,\frac{\text{m}}{\text{s}^2}$. This value is not overly optimistic since only 6 out of all 107613 trajectories in the HighD dataset exhibit a larger deceleration. Moreover, the possible deceleration of the predecessor can vary depending on the distance to the pre-predecessor. Therefore, I propose an

[1] Every vehicle that is involved in the lane change restore their usual RSS safe distance d_{safe}

enhancement to the RSS safety distance in Section 3.2.1: Assuming the pre-predecessor[1] brake with $a_{\text{max,dcc,obj}}$ until a full stop, the needed deceleration for the predecessor not colliding with pre-predecessor is a_{need}, the predecessor will brake with $a'_{\text{max,dcc}}$ which is a bounded value of a_{need} between $0.5a_{\text{max,dcc,obj}}$ and $a_{\text{max,dcc,obj}}$. The new safe distance d'_{safe} can be calculated correspondingly with Equation (3.2).

With the additional assumptions, I have re-examined the safe lane change ratio in the datasets and presented the results in Table 3.2. The relaxed RSS safety leads to significantly fewer violations when compared to Table 3.1. With the parameter set of $a_{\text{max,dcc,obj}} = -10\,\frac{\text{m}}{\text{s}^2}$ and $a_{\text{max,dcc,ego}} = -8\,\frac{\text{m}}{\text{s}^2}$, the safe lane change and merging ratios have increased from 76.1% and 90.9% to 92.1% and 97.5%, respectively. Even though the number of violations is still lower than the number of accidents, I posit that this new approach strikes a reasonable balance between safety and human consensus.

Table 3.2: Percentage of RSS safe lane changes (from 12380 lane changes in HighD dataset) and merges (from 4604 on-ramp merges in ExitD dataset) with certain deceleration parameters. Both datasets are recorded on German highways. Note that $a_{\text{max,dcc,obj}} \leq a_{\text{max,dcc,ego}}$ is assumed.

Ratio of safe lane changes		$a_{\text{max,dcc,obj}}(\frac{\text{m}}{\text{s}^2})$			
		-6	-8	-10	-12
$a_{\text{max,dcc,ego}}(\frac{\text{m}}{\text{s}^2})$	-6	0.974	0.875	0.796	0.728
	-8		0.976	**0.921**	0.873
	-10			0.976	0.944
	-12				0.977
Ratio of safe merges		$a_{\text{max,dcc,obj}}(\frac{\text{m}}{\text{s}^2})$			
		-6	-8	-10	-12
$a_{\text{max,dcc,ego}}(\frac{\text{m}}{\text{s}^2})$	-6	0.990	0.945	0.876	0.811
	-8		0.993	**0.975**	0.947
	-10			0.995	0.987
	-12				0.995

[1] In case it is out of sensing range, assuming one at the sensing border.

I recommend to use this approach for scenarios where the preconditions of the additional assumptions can be checked at a low cost (e.g., merging into one-lane road). However, for free lane change on multiple lanes, checking the preconditions against all neighboring vehicles is almost intractable. In such cases, I suggest applying the approach described in Section 3.2.1.

3.3.2 Tracking of Occlusions

In Section 3.2.4, I discussed the trade-off between assuming worst-case prioritized vehicles in occlusions to ensure provable safety and producing efficient driving behavior. However, human drivers possess the ability to reason about possible traffic participants in occlusions by utilizing their prior knowledge about the street and continuously observing changes in the FoV while moving forward. I propose an approach in this subsection to replicate this human-like intelligence, which was published in [Wan21].

I initially introduce the approach for reasoning about states of occluded vehicles, but it can be easily adapted for application to other types of traffic participants. By reducing the state intervals in occlusions from worst case, the safety constraints of the RSS framework in Section 3.2.4 can be relaxed, allowing for more expedient traversal of all the conflict zones of one RE.

State Set and Subset

Two mild assumptions are involved for simplification. Firstly, the vehicles are assumed to move along the centerline, since the longitudinal distance to the conflict zone is the one that affects the RSS safety directly. However, the concept is applicable for more dimensions such as lateral position and orientation, in order to model more traffic participants like pedestrians and cyclists. Given the longitudinal position along the centerline of the route, the global coordinates of the vehicle can be retrieved by utilizing the lanelet2 map. The second mild assumption is, no vehicle drives backward, i.e. with velocity lower than $0\,\frac{\mathrm{m}}{\mathrm{s}}$.

With those assumptions, the state of a vehicle on a certain route is represented as the longitudinal position s and the velocity v. The state set of the vehicle

is then a two-dimensional space defined by $\mathcal{S} = \{(s, v)\}$ with $s \in [s_1, s_2]$ and $v \in [0\ \frac{\text{m}}{\text{s}}, v_{\text{max,obj}}]$. One subset $\mathbf{S}$ can be arbitrary part of the set $\mathbf{S} \in \mathcal{S}$, which can be represented by $\mathbf{S} = \{(s, v)\}$ with $s \in [\underline{s}, \overline{s}]$ and $v \in [\underline{v}, \overline{v}]$. Each subset $\mathbf{S}$ contains a part of possible states of vehicles on the lane. One example is shown in Figure 3.10.

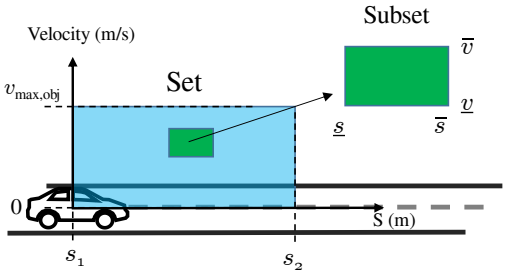


Figure 3.10: Example of set and subset.

When the target lane is occluded, I can over-approximate all the possible vehicles' states in the occlusions by creating several sets $[\mathcal{S}_1, \mathcal{S}_2, ...]$ on each of the occluded sections of the lane in v-s-space.

Operations on Subsets

I first define three geometric operations for a subset $\mathbf{S}$: *Grow*, *split* and *merge*.

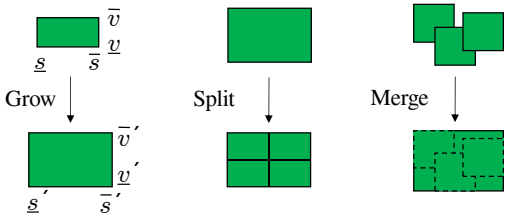


Figure 3.11: *Grow*, *split* and *merge* operations for subsets.

Grow: After a certain time period t, one subset $\mathbf{S}$ expands its region and propagate its $\{\underline{v}, \overline{v}, \underline{s}, \overline{s}\}$ to $\{\underline{v}', \overline{v}', \underline{s}', \overline{s}'\}$, by following the vehicle dynamics,

i.e. acceleration limits $a_{\text{max,dcc,obj}}$ and $a_{\text{max,acc,obj}}$. One example is shown in Figure 3.11. The new limits are computed following

$$\overline{v}' = \begin{cases} \overline{v} + ta_{\text{max,acc,obj}} & (\overline{v} + ta_{\text{max,acc,obj}} \leq v_{\text{max,obj}}) \\ v_{\text{max,obj}} & (\overline{v} + ta_{\text{max,acc,obj}} > v_{\text{max,obj}}) \end{cases} \tag{3.4}$$

$$\underline{v}' = \begin{cases} \underline{v} + ta_{\text{max,dcc,obj}} & (\underline{v} + ta_{\text{max,dcc,obj}} \geq 0) \\ 0 & (\underline{v} + ta_{\text{max,dcc,obj}} < 0) \end{cases} \tag{3.5}$$

$$\overline{s}' = \begin{cases} \overline{s} + \overline{v}t + \frac{1}{2}a_{\text{max,acc,obj}}t^2 & (\overline{v} + ta_{\text{max,acc,obj}} \leq v_{\text{max,obj}}) \\ v_{\text{max,obj}}t - \frac{(v_{\text{max,obj}} - \overline{v})^2}{2a_{\text{max,acc,obj}}} & (\overline{v} + ta_{\text{max,acc,obj}} > v_{\text{max,obj}}) \end{cases} \tag{3.6}$$

$$\underline{s}' = \begin{cases} \underline{s} + \underline{v}t + \frac{1}{2}a_{\text{max,dcc,obj}}t^2 & (\underline{v} + ta_{\text{max,dcc,obj}} \geq 0) \\ \underline{s} - \frac{\underline{v}^2}{2a_{\text{max,dcc,obj}}} & (\underline{v} + ta_{\text{max,dcc,obj}} < 0) \end{cases} \tag{3.7}$$

Split: One subset can be *split* into several subsets if they can cover the same region as the original subset, as shown in Figure 3.11.

Merge: As the reverse of *split*, several subsets can also be substituted by one subset, if it covers all their regions. However, this might introduce some over-approximation of the state intervals, as illustrated in Figure 3.11.

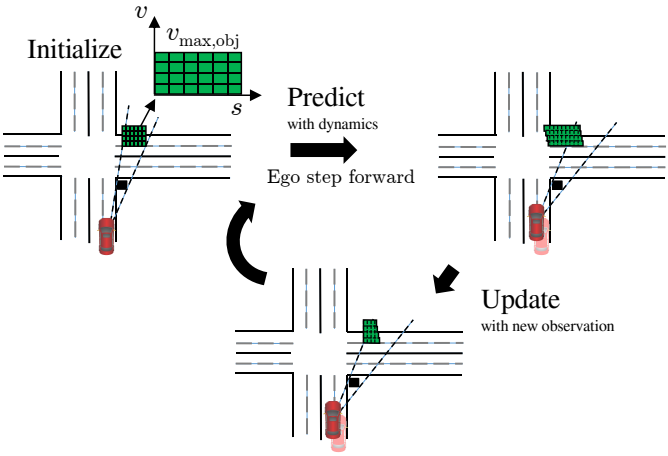


Figure 3.12: The pipeline of tracking subsets.

Tracking Subsets in Occlusions

The objective of this approach is to track potential hidden vehicles in occlusions, thereby relaxing the worst-case assumption and using this information for behavior planning. The pipeline (as depicted in Figure 3.12) begins with initializing subsets on newly observed occlusions. By utilizing the aforementioned operations to predict the subsets and updating them with new observations, they can be tracked in a closed loop. At each planning step, the updated subsets can be used to generate phantom vehicles and verify the safety constraints of the RSS framework.

Initialize: As depicted in Figure 3.12, when the ego vehicle observes an occlusion on the prioritized lane, a state set $\mathcal{S} = (s, v)$ with $s \in [s_1, s_2]$ and $v \in [0\,\frac{\text{m}}{\text{s}}, v_{\text{max,obj}}]$ can be initialized in v-s-space. $\mathcal{S}$ encompasses all the potential states of the vehicles in the initial occlusions. Then, the initial set $\mathcal{S}$ is split into subsets with a discretization size of $\Delta v = 0.2\,\frac{\text{m}}{\text{s}}$ and $\Delta s = 0.2\,\text{m}$, each of which covers a portion of the possible states in the occlusions.

Predict: The prediction step is described in detail in Figure 3.13. After one time step, the ego vehicle moves to a slightly different position. Meanwhile,

all the initialized subsets with a size of Δv and Δs should also *grow* according to the vehicle dynamics, albeit with different scales. For instance, subsets A and B in Figure 3.13 will have different sizes according to Equation (3.4) to Equation (3.7) after *growing*. The union of all *grown* subsets will no longer be rectangular, but rather similar to the green region in the upper right figure of 3.13. The reason for this distorted shape is that the subsets in the upper part will move faster than the subsets in the bottom, due to their overall higher velocity. After *growing*, the number of subsets remains the same, but they differ in size and overlap with each other. I then employ the *split* operation for all the subsets with the discretization size of Δv and Δs. For example, subsets A and B will be *split* into four subsets each. By doing so, the total number of subsets increases exponentially. To prevent memory and computational issues, I limit the number of subsets by *merging* the *split* subsets using a strategy displayed in the bottom left of Figure 3.13. The rectangular envelope of all subsets can be discretized with Δv and Δs, which results in M×N grids in v-s-space. The *split* subsets are *merged* into one if their centers lie within the same grid, for instance, the two blue subsets in Figure 3.13 will be *merged* into one. After the entire **prediction** step, I obtain another set of subsets, with a maximum number of M×N, each of which has a maximum size of $2\Delta v \times 2\Delta s$, as illustrated in the upper right figure of Figure 3.12.

Update: As the ego vehicle already has a new FoV after one time step, the predicted subsets should be updated according to the new observation. One subset will be removed once any part between its $\underline{s}$ and $\overline{s}$ is exposed in the FoV. As Δs is much smaller than the length of a real vehicle and a subset has maximum $2\Delta s$ size after prediction, it is guaranteed that a vehicle covered by a subset is visible, once the subset itself is visible.

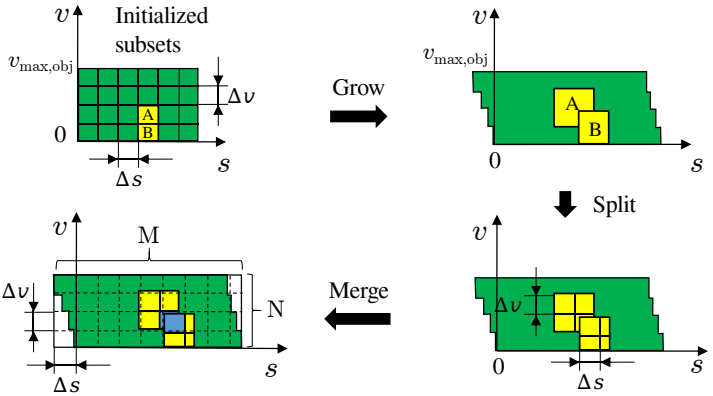


Figure 3.13: Predicting subsets.

Losing Track on Borders of Occluded Lane Sections

Up until this point, I have only considered cases where both ends of the occluded section are under tracking. This ensures that no new vehicle from outside can enter the occluded lane section. However, when one end of the occluded section connects with the invisible world, it is assumed that new subsets with a velocity interval of $[0\ \frac{\text{m}}{\text{s}}, v_{\text{max,obj}}]$ will enter the occlusion from the connecting point. The new subsets are predicted and updated in the same manner as the existing subsets. One example is depicted in Figure 3.14.

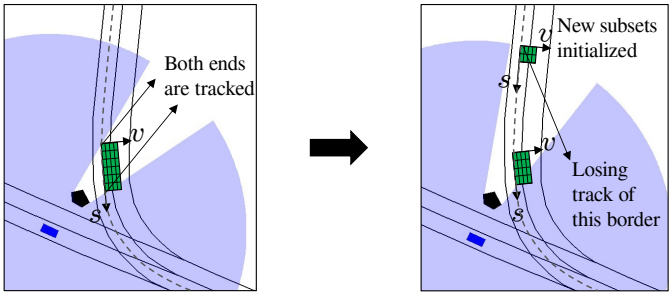


Figure 3.14: Losing track on borders of occluded lane sections

Application on RSS Safety Verification

In this chapter, this approach is utilized for verifying RSS safety under occlusions. Once the state intervals of possible hidden prioritized vehicles are over-approximated by this approach, the *maximum reachability* of the possible hidden vehicles can be determined. The evaluation in [Wan21] shows that a significantly less conservative behavior can be achieved when occlusions can be tracked with this approach.

3.3.3 Limited Reachability of Prioritized Vehicles

As discussed in Section 3.2.3, the *maximum reachability* of the ego vehicle can be restricted under several conditions. Similarly, the *maximum reachability* of the prioritized vehicles can also be limited. The former enhances the safety requirements, whereas the latter weakens them.

Limitation by Comfort

Prioritized vehicles may not be able to reach their maximum velocity $v_{\text{max,obj}}$ due to high curvature of the roads, as depicted in Figure 3.15. In this case, a reasonable maximum lateral acceleration $a_{\text{max,acc,lat,obj}}$ can be assumed for prioritized vehicles.

Ideally, the $a_{\text{max,acc,lat,obj}}$ should be the maximum lateral acceleration recorded in real traffic. In [Rey01], it was demonstrated that human drivers can tolerate different maximum lateral accelerations at different velocities. However, for simplicity, I only consider a single value at all velocities. The maximum recorded lateral acceleration in the inD and rounD datasets is $7.83\,\frac{\text{m}}{\text{s}^2}$, which matches the results of the experiments in [Rey01], which found a maximum lateral acceleration of approximately $8\,\frac{\text{m}}{\text{s}^2}$ at a speed of $10\,\frac{\text{m}}{\text{s}}$ for various participants. Therefore, I set $a_{\text{max,acc,lat,obj}} = 7.83\,\frac{\text{m}}{\text{s}^2}$.

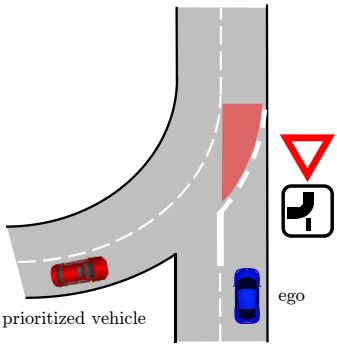


Figure 3.15: The *maximum reachability* of the prioritized vehicle is limited by the maximum lateral acceleration caused by the curvature of the road.

Limitation by Traffic Rule

Prioritized vehicles may be stopped by traffic rules for an unknown period of time, such as when facing traffic lights or pedestrian crossings, as illustrated in Figure 3.16. If the ego vehicle can perceive this information reliably, I propose the following assumptions for the *maximum reachability* of prioritized vehicles:

- With a pedestrian blocking the route of the prioritized vehicle, it is assumed that the pedestrian will pass with its maximum velocity $v_{\text{max,p}}$ and the prioritized vehicle will accelerate with $a_{\text{max,acc,obj}}$ such that it can pass the pedestrian tightly.
- With a red traffic light blocking the route of the prioritized vehicle, it is assumed that the traffic light will switch to yellow and then green immediately (unless the ego vehicle observes the beginning of the red traffic light and a minimum duration can be assumed in this case), and the prioritized vehicle will continue its velocity within the reaction time ρ_{obj} and then accelerate with $a_{\text{max,acc,obj}}$ until reaching $v_{\text{max,obj}}$.

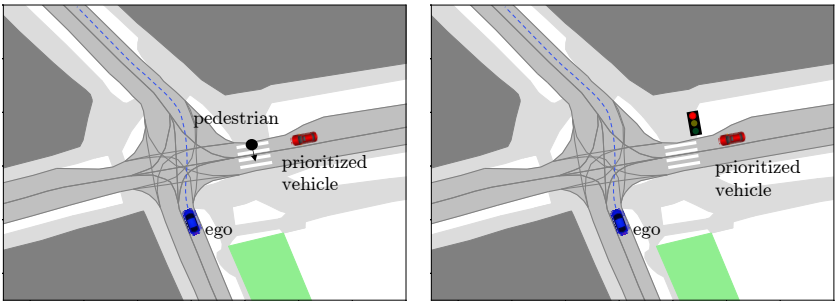
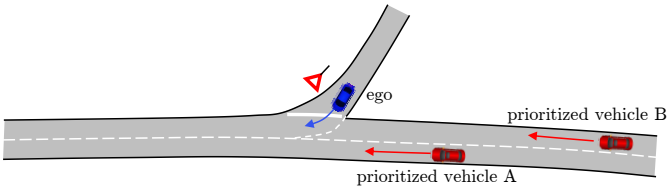


Figure 3.16: The *maximum reachability* of the prioritized vehicle is limited by a crossing pedestrian or a red traffic light, when the ego vehicle tends to follow the blue dashed route.

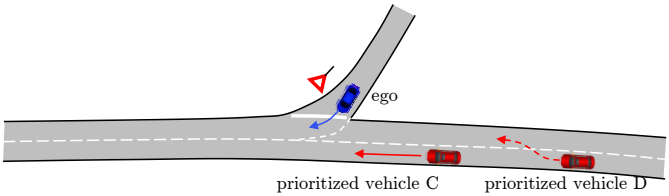
Other obstacles could also block prioritized vehicles, such as another stationary vehicle with emergency lights activated. However, as an overtaking maneuver may be performed to avoid such obstacles, they are not regarded as legal and definite blockage.

Limitation by Conscious Decision

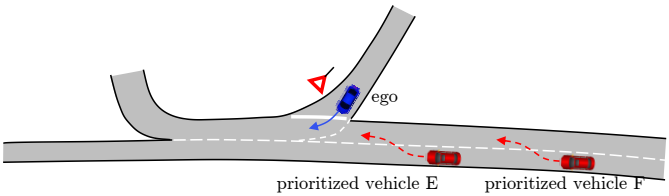
Prioritized vehicles are assumed to make conscious decisions that maximize their utility and minimize their risk while allowing for the convenience of non-prioritized vehicles, if feasible.



(a) The ego vehicle must ensure RSS safety with the prioritized vehicle B. It is not necessary to assess RSS safety for vehicle A, as a conscious decision for it is to not change lanes to the right.



(b) It is evident that a conscious decision for vehicle C is to stay in the current lane, while for vehicle D, it may be to change lanes to the right to overtake. Thus, ensuring RSS safety with vehicle D is necessary.



(c) When prioritized vehicles have a compelling reason to change lanes, such as continuing along their route, RSS safety must be ensured for all of them.

Figure 3.17: Examples of prioritized vehicles making conscious decisions that affect the RSS safety of the ego vehicle.

In the examples illustrated in Figure 3.17a, the ego vehicle should yield to prioritized vehicles A and B. However, for prioritized vehicle A, changing lanes to the right could put it at risk if the ego vehicle suddenly moves in, without gaining any benefit. Therefore, a conscious decision for vehicle A to maintain smooth traffic flow for all vehicles is to stay in the current lane. In this scenario, ensuring RSS safety with vehicle A is not necessary. However, in Figure 3.17b, changing lanes to the right for vehicle D may be beneficial if overtaking vehicle C allows for a higher velocity. In this case, ensuring RSS safety with vehicle D

is necessary. In Figure 3.17c, both prioritized vehicles may have a compelling reason to change lanes if their routes necessitate it. In this situation, ensuring RSS safety for both vehicles is required. Furthermore, aside from the example depicted in Figure 3.17c, the prioritized vehicles may have other reasons to change lanes, such as parking on the right side if allowed.

As a result, I introduce the following RSS common sense to additionally limit the *maximum reachability* of prioritized vehicles in order to allow an overall smooth traffic:

- If changing to the lane that may conflict with the non-prioritized vehicle is not advantageous (such as gaining velocity, continuing its route, pulling over, etc.), the prioritized vehicle is not assumed to do so and may be excluded when assessing RSS safety.

I do not cover all possibilities of this rule since the term “advantageous”can be scenario-dependent. Instead, I leave this rule open to allow concrete mathematical formulations depending on scenarios.

3.4 High-level Actions and Rule-based Policies

Following a comprehensive discussion on RSS safety, I propose high-level action spaces for various scenarios and introduce safe rule-based policies based on these actions in this section.

Driving behind a leading vehicle in a single lane typically does not require high-level decision making and can be efficiently addressed using contemporary ACC functionality. As a result, my proposed high-level action representations are solely intended for scenarios that require semantic decisions. Similarly to Section 3.2, I divide these scenarios into two categories: parallel lanes and intersecting lanes. The former involve decisions such as lane changes or merging. The latter include scenarios where a decision does not require any intelligence, e.g. at intersections controlled by traffic lights. Rather, I focus solely on unsignalized intersections controlled by priority rules, but decisions can be ambiguous. Such scenarios include unsignalized intersections, unprotected left

turns, and roundabouts, where semantic decisions like slowing down, passing, or stopping are possible.

Subsequently, I present proof-of-concept low-level executions of the proposed actions that generate longitudinal and lateral control signals, primarily relying on the Intelligent Driver Model (IDM) [Tre00]. Although the low-level planner can be replaced by any trajectory planner and controller, it adequately serves my evaluation purposes. Following this, I introduce several rule-based policies based on the proposed RSS-safe actions. They will serve as baseline policies in subsequent chapters.

3.4.1 Parallel Lanes

In parallel lanes, the vehicle can make longitudinal and lateral decisions in a combinatorial manner. Parallel lanes can be further categorized into scenarios that involve free lane changes and mandatory merging, as illustrated in Figure 3.2. It is important to note that mandatory merging not only encompasses on-ramp merging scenarios, but also cases where the ego vehicle is pursuing a clear target lane, such as when it intends to exit the highway and must merge into the right adjacent lanes. I define distinct high-level action classes for free lane changes and mandatory merging.

High-level Actions for Free Lane Change

In free lane change scenarios, I define five semantic actions:

- $a_{\text{free},1}$: Keep lane and maintain a regular car-following style
- $a_{\text{free},2}$: Keep lane and use a more conservative car-following style
- $a_{\text{free},3}$: Keep lane and use a more aggressive car-following style
- $a_{\text{free},4}$: Change lane to the left into the current gap
- $a_{\text{free},5}$: Change lane to the right into the current gap

High-level Actions for Mandatory Merging

In mandatory merging scenarios, I prefer actions that have a semantic meaning, such as possible gaps implicitly constructed by vehicles on the target lane, instead of longitudinal and lateral accelerations or jerks, as implemented in some RL-based approaches [Fis22]. To limit the action space, I account for a maximum of four vehicles on the target lane that are longitudinally closest to the ego vehicle and within the FoV. Consequently, the number of actions is limited to five, with four of them involving merging in front of the target vehicles, and the fifth one involving merging into the very last gap after the last target vehicle. The merging actions are denoted as $a_{\text{merge},i}$, where i refers to the gap number. My decision-making pipeline, as depicted in Figure 3.1, is capable of addressing a variable number of gaps, in case of less than four target vehicles.

Low-level Action Executions

Once a high-level decision $a_{\text{free},i}$ or $a_{\text{merge},i}$ is made, it is decoupled into a longitudinal acceleration a_{lon} and lateral velocity v_{lat} for the purpose of proof-of-concept control of the vehicle.

The IDM generates the longitudinal acceleration $\dot{v}_{\text{IDM}}$

$$\dot{v}_{\text{IDM}} = a\left(1 - \left(\frac{v}{v_{\text{d}}}\right)^4 - \left(\frac{d^*(v, \Delta v)}{d}\right)^2\right) \tag{3.8}$$

where d^* is the desired distance to the vehicle ahead, which is defined by

$$d^*(v, \Delta v) = d_0 + vT_{\text{d}} + \frac{v\Delta v}{2\sqrt{ab}}. \tag{3.9}$$

The parameters that need to be set include the maximum acceleration (a), desired velocity (v_{d}), minimum accepted distance (d_0), desired time gap (T_{d}), and desired deceleration (b). The output acceleration is determined by the velocity difference (Δv) and the distance to the vehicle in front (d).

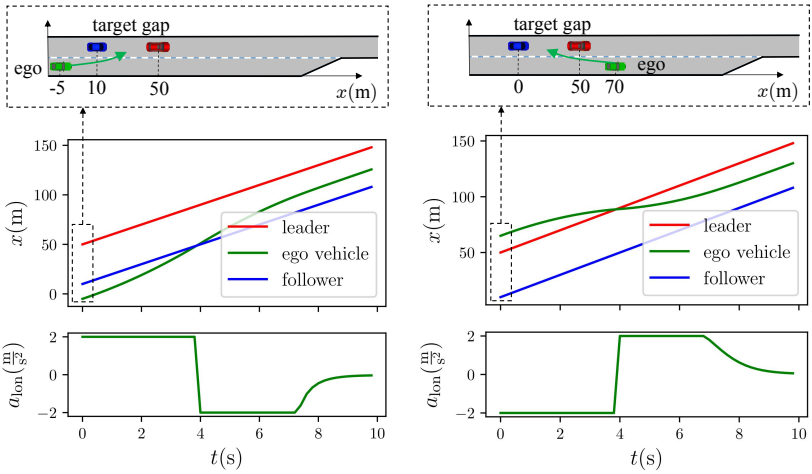
The action $a_{\text{free},1}$ can directly utilize the output of IDM, while $a_{\text{free},2}$ can be realized by reducing v_{d} and increasing T_{d} (e.g. by 10%), whereas the opposite can be taken for $a_{\text{free},3}$.

For other actions ($a_{\text{free},4}, a_{\text{free},5}, a_{\text{merge},i}$) where lane changes or merging are involved, this formula can not be directly applicable. The reason is that the ego vehicle should fit into a gap that is constructed by two vehicles, i.e. the leading and following vehicles on the target lane, rather than following a single vehicle. It could even be possible that both vehicles of the gap are behind or ahead of the ego vehicle, resulting in negative distances d. Furthermore, during longitudinal adjustment, it is necessary to maintain a proper distance from the leading vehicle on the source lane as well, i.e., multiple leading vehicles need to be taken into account simultaneously. In order to address these issues, I introduce several modifications and customize the IDM model, which I refer to as the Intelligent Driver Model for Merging (MIDM) in later chapters. The $\dot{v}_{\text{IDM}}$ generated by MIDM can be formulated as

$$\dot{v}_{\text{IDM}} = a\left(1 - \left(\frac{v}{v_{\text{d}}}\right)^4 - \max_{i=1}^{n}\left(\frac{d^*\left(v, \Delta v_{\text{f}_i}\right)}{g(d_{\text{f}_i})}\right)^2 + \left(\frac{d^*\left(v_{\text{b}}, \Delta v_{\text{b}}\right)}{g(d_{\text{b}})}\right)^2\right) \tag{3.10}$$

where $g(d) = \max\{\delta, d\}$ is the bounded distance with δ to be a small number (e.g. $1e^{-10}$) to prevent numerical errors. The Δv_{b}, Δv_{f_i}, d_{b}, d_{f_i} are velocity differences and distances to the following vehicle of the gap and the i-th leading vehicle. Note that for leading vehicles, the distance d_{f_i} is positive when the vehicle is in front of the ego vehicle. For the following vehicle, d_{b} is positive when it is behind the ego vehicle. In this case, there are two possible leading vehicles, one on the target lane and one on the source lane. Finally, the output longitudinal acceleration will be bounded via $a_{\text{lon}} = \min(\max(\dot{v}_{\text{IDM}}, a_{\text{max,dcc,ego}}), a_{\text{max,acc,ego}})$ to the range of $[a_{\text{max,dcc,ego}}, a_{\text{max,acc,ego}}]$.

Figure 3.18 illustrates two examples of how the MIDM controls the ego vehicle to fit into different gaps that are moving with constant velocity. Parameters used in Equation (3.10) are listed in Table A.1.



(a) The ego vehicle is initially behind the gap.

(b) The ego vehicle is initially in front of the gap.

Figure 3.18: Longitudinal position x and acceleration a_{lon} of the ego vehicle with MIDM. The leader and the follower of the target gap are moving with constant velocity. Parameters $a_{\text{max,dcc,ego}}$ and $a_{\text{max,acc,ego}}$ are set to $-2\,\frac{\text{m}}{\text{s}^2}$ and $2\,\frac{\text{m}}{\text{s}^2}$ in this example (from [Wan23a], ©2023 IEEE).

The lateral velocity v_{lat} of the actions $a_{\text{free},1}$, $a_{\text{free},2}$ and $a_{\text{free},3}$ are $0\,\frac{\text{m}}{\text{s}}$. For the other actions, the non-holonomic kinematics of the autonomous car are taken into account by constraining v_{lat} via a maximum side slip angle similar as proposed in [Hub18] where the positive sign of v_{lat} points to the target lane.

$$v_{\text{lat}} = \begin{cases} \min\{0.17v, 0.8\,\frac{\text{m}}{\text{s}}\} & \text{RSS safe w.r.t. the gap} \\ -\min\{0.17v, 0.8\,\frac{\text{m}}{\text{s}}\} & \text{else} \end{cases} \tag{3.11}$$

The prerequisite for having lateral velocity is RSS safety to the leading and following vehicle. Additionally, each action includes a fallback longitudinal reaction. As soon as the RSS safety is violated, e.g. by other vehicles cutting in front closely or because the merging lane is going to end[1], the emergency braking $a_{\text{min,decel}}$ will overwrite the output of the MIDM.

[1] Equivalent to an obstacle with velocity 0m/s standing at the end of merging lane.

Follow lane behavior with yielding capability

Multi-lane driving requires a more sophisticated policy than MIDM when vehicles on other lanes require cooperation from the ego vehicle during a lane change. This necessitates that the ego vehicle behaves courteously towards lane-changing or merging vehicles that clearly communicate their desire to change lanes (e.g., via the use of turn signals). To address this, I introduce the Cooperative Intelligent Driver Model (CIDM). When a cut-in desire from another vehicle is detected, the ego vehicle calculates a yielding motivation value m using a logistic regression function

$$m = \frac{1}{1 + e^{-\theta_{\mathrm{Y}}^{T} f_{\mathrm{Y}}}} \tag{3.12}$$

with the θ_{Y} to be the weight vector and $f_{\mathrm{Y}} = [d, t_{\mathrm{TH}}, \dot{t}_{\mathrm{TH}}]$ to be the feature vector, where d denotes the distance between the ego vehicle and the merging vehicle, $t_{\mathrm{TH}} = \frac{d}{v_{\mathrm{main}}}$ is the time headway to the merging vehicle and $\dot{t}_{\mathrm{TH}} = \frac{v_{\mathrm{main}} - v_{\mathrm{merge}}}{v_{\mathrm{main}}}$ is the changing rate of time headway. v_{main} and v_{merge} are the velocities of the ego vehicle and the merging vehicle. The model is trained with the Interaction dataset [Zha19] and ExitD dataset [Moe22] where in total 3320 vehicles are recorded to yield to a merging vehicle and 432 vehicles not. The learned parameters are presented in Table A.3 together with modified versions for different driving styles. To control the willingness to yield, I introduce a threshold value $m_{\mathrm{th}} = 0.5$. If the vehicle decides to yield ($m > m_{\mathrm{th}}$), it treats both the merging vehicle and the preceding vehicle in the current lane as target vehicles and calculates its acceleration using Equation (3.10).

Rule-based Lane Change Policy and Merging Policy

After introducing the action classes and their low-level executions for multi-lane scenarios, different rule-based policies are proposed utilizing these actions.

For free lane change, the Minimizing Overall Braking Induced by Lane changes (MOBIL) strategy [Kes07] can be used as one rule-based lane change policy.

This model makes lane change decisions with the objective of maximizing the acceleration of all the involved vehicles. The IDM model is used to calculate the accelerations of the surrounding vehicles. Then, a lane change is performed if

$$\tilde{a}_{\mathrm{e}} - a_{\mathrm{e}} + p * ((\tilde{a}_{\mathrm{n}} - a_{\mathrm{n}}) + (\tilde{a}_{\mathrm{o}} - a_{\mathrm{o}})) > a_{\mathrm{th}} \tag{3.13}$$

where a_{e}, a_{n}, and a_{o} represent the accelerations of the ego vehicle, the following vehicle on the neighbor lane, and the following vehicle on the original lane, respectively, assuming no lane change of the ego vehicle is performed. Correspondingly, the ones with tildes represent their accelerations if the ego vehicle changes lane. The politeness factor p is included to control how much the acceleration gains and losses of other vehicles are valued. The left side of Equation (3.13) represents the overall acceleration gain a_{gain}, which must be greater than a threshold value a_{th} for a lane change to occur. Note that if a lane change is possible in both directions, it will be performed in the direction where Equation (3.13) is satisfied and whose a_{gain} is higher.

This MOBIL policy is unable to choose between $a_{\mathrm{free},1}$, $a_{\mathrm{free},2}$, and $a_{\mathrm{free},3}$. Therefore, when applying this policy, $a_{\mathrm{free},2}$ and $a_{\mathrm{free},3}$ are not considered. The politeness factor p and the acceleration threshold a_{th} are presented in Table A.4, along with modified versions for different driving styles.

The MOBIL model can be combined with the CIDM in order to perform a lane change for cut-in attempts, which I call Cooperative Minimizing Overall Braking Induced by Lane changes (CMOBIL). If the vehicle has high motivation to yield ($m > m_{\mathrm{th}}$), a_{e} will be computed additionally taking the vehicle with cut-in desire as one of the front vehicles. Depending on Equation (3.13), it either performs a lane change or decelerates.

In mandatory merging scenarios, a rule-based merging policy based on heuristics is described in [Nau19]. In this approach, each gap is assumed to move with constant velocity, and the merging vehicle tries to reach the gap by either accelerating or decelerating at a constant rate. The gap selection heuristic involves selecting the gap that can be reached earliest in space along the target lane. This policy is called Closest-Gap Merging Policy (CGMP).

3.4.2 Intersecting Lanes

In the case of intersecting lanes, specifically unsignalized intersections, I propose longitudinal high-level decisions, which can then be transformed into either a longitudinal recommended velocity profile or a desired acceleration. The assumption is that the AV moves along the centerline of the route, without any lateral decision making involved. Obstacle avoidance within the lane, fine speed control and comfort maximization, etc. are supposed to be accomplished within the subsequent trajectory planner.

High-level Actions and Rule-Based Policies

I introduce three basic high-level actions for intersections.

- *Stop* ($a_{\text{inter,1}}$): Stopping before the first conflict zone (or the stop line).
- *Pass* ($a_{\text{inter,p}}$): Passing the conflict zones with *maximum reachability*.
- *Squeeze* ($a_{\text{inter,s}}$): Carefully advancing with minimum velocity.

Figure 3.7 depicts a simplest RSS-safe policy for passing intersections. This policy guarantees that either stopping before the first conflict zone or passing all conflict zones is satisfied. However, in scenarios where no intersection between the two conditions can be found, the ego vehicle can become trapped in a deadlock situation where traversing from one to the other with only *stop* and *pass* is not possible, especially in cases with strong occlusions. In such situations, I introduce a third action, *squeeze*, which allows the ego vehicle to slowly approach or even enter the conflict zone to gain more visibility, for instance, with a velocity of $1\,\frac{\text{m}}{\text{s}}$. This action is considered RSS-safe as well. I refer to this simple policy as the first basic intersection policy (B1).

Extended Actions and Advanced Policies

It has been observed that in certain scenarios of the datasets, human drivers are able to find a smoother transition between stopping (C_1) and passing

($\{C_2, C_{2,p}, C_3\}$), such as intersections with slight occlusion. Instead of *stopping* before the conflict zones with constant deceleration, until $\{C_2, C_{2,p}, C_3\}$ is satisfied to switch to *pass*, they try to decelerate less at the beginning. This allows them to switch to *pass* before they must execute a harsh brake to avoid violating C_1. To replicate this behavior, I can introduce another action that is similar to *stop* but with less deceleration at the beginning and more deceleration as the vehicle approaches the conflict zones. Conversely, there are also scenarios where the ego vehicle slows down more at the beginning. This can signal the vehicle's cooperative stopping intention to other traffic participants.

There are potentially infinite variations of approaching or stopping styles, but for the simplicity of the action space, I introduce only two additional actions: *fast approach* ($a_{\text{inter},2}$) and *early stop* ($a_{\text{inter},3}$) in addition to the existing *stop* action. The low-level implementation of these actions is discussed in the next subsection. With these new actions, I can create more rule-based policies such as B2, which replaces *stop* in B1 with *fast approach*, and B3, which replaces *stop* in B1 with *early stop*. Furthermore, more advanced and human-like policies can be developed, such as dynamically selecting different approaching styles at each decision step rather than sticking to a single one.

The rule-based policies B1, B2 and B3 are visualized in Figure 3.19.

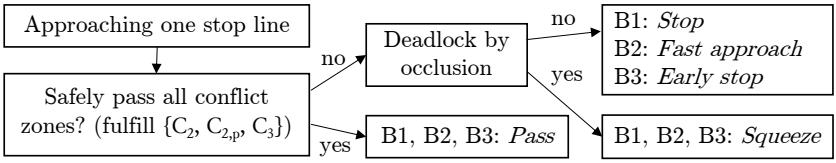


Figure 3.19: Flow charts for the policies B1, B2 and B3 (from [Wan23b], ©2023 IEEE).

Low-level Action Executions

IDM is again utilized to generate longitudinal acceleration $a_{\text{lon}} = \dot{v}_{\text{IDM}}$ for all approaching actions, where the stop line or the first conflict zone is regarded as an additional virtual obstacle with 0 velocity. In order to generate different approaching styles *fast approach*, *stop* and *early stop*, I add another parameter α

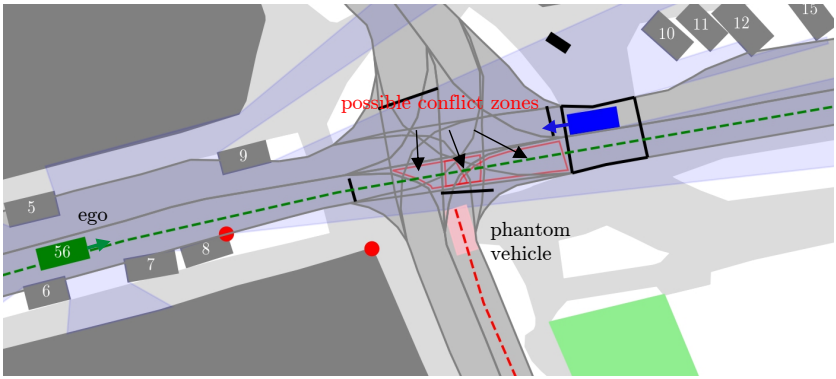
into the original formulation, which is reformulated as Intelligent Driver Model for Intersection (IIDM)

$$\dot{v}_{\mathrm{IDM}} = a\left(1 - \left(\frac{v}{v_{\mathrm{d}}}\right)^4 - \alpha\left(\max\left(\frac{d^*\left(v, \Delta v_{\mathrm{f_{lead}}}\right)}{d_{\mathrm{f_{lead}}}}, \frac{d^*\left(v, \Delta v_{\mathrm{f_{sl}}}\right)}{d_{\mathrm{f_{sl}}}}\right)\right)^2\right) \tag{3.14}$$

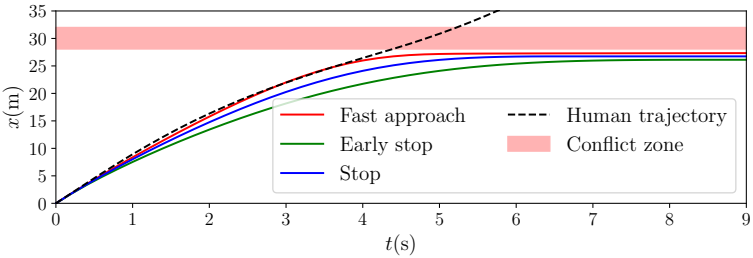
where $d^*(v, \Delta v) = d_0 + vT_{\mathrm{d}} + \frac{v\Delta v}{2\sqrt{ab}}$ is the desired distance to the vehicle ahead. The parameters to set are similar to Equation (3.8). α aims to control the contribution of the leading vehicle ($\mathrm{f_{lead}}$) or virtual obstacle from stop line ($\mathrm{f_{sl}}$) to the overall acceleration and is 1 in the original IDM formulation. In my work, α is set to 0.5 for *fast approach*, 1 for *stop* and 2 for *early stop*.

Figure 3.20 depicts an occluded intersection, where a human driver is not decelerating with the typical IDM deceleration to stop before the conflict zones. Instead, a smoother transition between *stop* and *pass* is observed, which is both safe and comfortable. Among the three predefined actions, the *fast approach* trajectory matches the human driver's trajectory the best and should be selected in this case by a human-like policy.

In the case of the *pass* action, the longitudinal acceleration should be determined by the *maximum reachability* of the ego vehicle before leaving all conflict zones. This concept is discussed in detail in Section 3.2.3. As for the *squeeze* action, the longitudinal acceleration can be obtained by using a speed P-controller that aims to maintain a velocity of $1\,\frac{\mathrm{m}}{\mathrm{s}}$.



(a) Ego vehicle approaching an occluded intersection



(b) Human approaching profile and generated action profiles

Figure 3.20: The ego vehicle is approaching an occluded intersection with potential phantom vehicles. The y coordinate of (b) is the longitudinal distance along the green dashed route of the ego vehicle in (a). One recorded human driver tries to approach with the black trajectory. My IIDM generates three different profiles for the three approaching actions (from [Wan23b], ©2023 IEEE).

4 Learning Driving Policies from Naturalistic Trajectories

The objective of this chapter is to determine the most human-like, convenient, comfortable, and least risky high-level decision from the action candidates proposed in Section 3.4, which already comply with the RSS safety constraints. As stated in Section 3.1, the aim is to characterize all actions using feature vectors. Utilizing recorded trajectories, the goal is to update the weights w of the linear function in Figure 3.1, in order to maximize the Q-value of the more human-like actions over other actions.

In this chapter, I first categorize various relevant features for decision making in Section 4.1. I then introduce an approach in Section 4.2 to estimate the features under a probabilistic environment. Subsequently, I optimize the Q-function using human driving trajectories, and finalize the learned policies for different scenarios in Section 4.3. To imitate diverse driving styles of humans, I perform driving style analysis and induce stylized policies in Section 4.4.

4.1 Relevant Features for Decision Making

A previous study [Nau20b] has already provided a summary of the existing features in the current state of the art. However, these features are primarily pertinent to trajectory planning rather than high-level decision making. In trajectory planning, optimizing for comfort and smoothness of the trajectory is often a critical objective, which can be achieved by minimizing the input jerk sequences [Zie15]. Conversely, for behavior-level decision making, the emphasis is more on high-level features, such as anticipated progress and perceived safety.

I classify all the features into four categories that are deemed important in high-level decision making: *utility, ride comfort, perceived safety,* and *politeness*. The features are represented numerically using scalars, for which mathematical formulations, either based on a resulting trajectory from an action, or based on other estimation methods, have been developed in Section 4.2. To enable comparison among the different features, they have been additionally normalized between 0 and 1 using the established formulations.

4.1.1 Utility

In the context of MDP, *utility* refers to the accumulated future rewards discounted over time, which can incorporate a wide range of attributes that impact driving behavior [Kno21], e.g. progress towards the destination, time spent driving, collision avoidance, etc. However, I adopt a different interpretation of the term *utility*, using it specifically to represent how progress towards the destination can be achieved through a specific action. Furthermore, I subdivide *utility* into three concrete features:

- U_1: How quickly the overall progress can be made by the action.
- U_2: How soon the action (e.g. merging into a gap) can be completed.
- U_3: How likely the action (e.g. merging into a gap) can be achieved.

A high U_1 indicates that the vehicle is able to maintain a speed closer to the desired velocity. The calculation of U_1 is based solely on the vehicle's velocity, and can be formulated as follows:

$$U_1 = 1 - \left| \frac{1}{n} \sum_{i=1}^{n} \frac{v_i}{v_{\text{des}}} - 1 \right| \tag{4.1}$$

where $v_1, v_2, \ldots, v_n$ denotes the sequence of velocities resulting from the given action, and v_{des} represents the desired velocity of the driver. This formulation also penalizes velocities that exceed v_{des}. It is worth noting that the desired velocity reflects the driver's preference rather than the speed limit. However,

for automated vehicles, the desired velocity should be set to the speed limit v_{limit} or the speed recommendation of the road in order to comply with legal requirements and avoid disrupting traffic flow.

In scenarios where explicit maneuvers are required, such as lane changes, merging, approaching intersections, the U_2 and U_3 features play a more significant role. They cannot be directly estimated from a resulting trajectory but are rather semantic-level information. For instance, when merging onto a highway, choosing a gap that requires less merging time (U_2) and is more likely to succeed (U_3) is usually a more attractive option than cutting into the very first gap that seems to be the fastest option (U_1). At intersections, U_2 and U_3 describe how quickly and how likely it is to cross the intersection. In Section 4.2, I will explain how to estimate U_2 and U_3.

4.1.2 Ride Comfort

Human comfort in AVs are characterized by naturality, disturbances, apparent safety, and motion sickness, as suggested in [Elb15]. Naturality and motion sickness are difficult to quantify using a single trajectory and are therefore not considered in this work. Apparent safety, on the other hand, is accounted for in other proposed features and is not included in the ride comfort category.

In trajectory planning, disturbances such as jerk and acceleration in the longitudinal and lateral directions are often considered to affect driving comfort and are penalized in the cost function [Zie15, Bur18]. However, in high-level decision making, optimizing plans with respect to jerk is not essential. Instead, I only take longitudinal and lateral acceleration into account.

In parallel-lane scenarios, the output decision is decoupled into longitudinal acceleration and lateral velocity by my low-level execution $a_{\text{free},i}$,$a_{\text{merge},i} = [a_{\text{lon}}, v_{\text{lat}}]$, see Section 3.4.1. The lateral acceleration at the i-th time step $a_{\text{lat},i}$ is derived by $a_{\text{lat},i} = \frac{v_{\text{lat},i} - v_{\text{lat},i-1}}{\Delta t}$.

At intersections, the output decision consists of only one longitudinal acceleration value a_{lon} along the centerline of the route. The lateral acceleration can be inferred from the velocity v and the curvature of the centerline κ via $a_{\text{lat}} = v^2\kappa$.

Assuming we have a sequence of $\{a_{\text{lon},1}, a_{\text{lon},2}, \dots, a_{\text{lon},n}\}$ and $\{a_{\text{lat},1}, a_{\text{lat},2}, \dots, a_{\text{lat},n}\}$ from a trajectory, the comfort feature C is computed by

$$C = 1 - \frac{1}{n} \sum_{i=1}^{n} \frac{\sqrt{a_{\text{lon},i}^2 + a_{\text{lat},i}^2}}{\left|a_{\text{max,dcc,obj}}\right|} \tag{4.2}$$

4.1.3 Perceived Safety and Driving Risk

Perceived safety is also treated as risk and has several definitions in the literature, depending on the scenario. I have reviewed numerous approaches that aim to compute collision risk in Section 2.3.1, which typically rely on upstream predictions of other vehicles that do not involve the ego vehicle. However, collision probability can also depend on the reaction time and style of each driver, making it computationally intractable to compute by considering all possible reactions of the agents involved. Moreover, as collisions are rare occurrences in real-world traffic, validating the collision probability is challenging.

Instead of relying on collision probability, I count events as risky, where evasive reactions of the ego vehicle are essential to maintain RSS safety, such as emergency braking or evasion. If a collision still occurs, the ego vehicle is not considered responsible, in accordance with the principles of RSS. The margin where the evasive reaction should be initiated is when the RSS safety is violated or about to be violated. Driving risk is then defined as the probability of taking evasive reactions, to restore or maintain RSS safety. Accordingly, I introduce two categories of driving risk: *emergency risk* R_1 and *fall-back risk* R_2.

The *emergency risk* represents the probability of emergency situations where the RSS safety is violated, either passively (e.g., by intruding traffic participants who disregard RSS) or actively (e.g., by violating $C_{1,\text{reg}}$ and $C_{2,\text{reg}}$ simultaneously). In such cases, a "proper response", such as braking with $a_{\text{max,dcc,ego}}$, needs to be performed. For example, during free driving, other vehicles on adjacent lanes might disregard the RSS safety and suddenly cut in front of the ego vehicle, forcing the ego vehicle to respond appropriately and execute a harsh

brake, resulting in a non-zero *emergency risk*. With this definition, one may argue that the *emergency risk* is always greater than 0 as long as there is a vehicle on the adjacent lane ahead. While this is true, the *emergency risk* is expected to be extremely low in most situations and should not affect my normal driving. In order to minimize the possibility of being involved in traffic accidents (even if not responsible), thresholds for the *emergency risk* can be proposed but is not in the scope of this work.

The *fall-back risk* represents the probability of switching to the fall-back plan, where the RSS safety is on the verge of being violated. The fall-back plan is still RSS-safe, but considered risky because the driver or passengers may feel endangered. For example, in a highway on-ramp merging scenario, a fall-back plan may be a failed merge followed by a harsh stop at the end of the merging lane. At an occluded intersection, when the vehicle approaches without caution (e.g. with *fast approach*), hoping that no prioritized vehicle is behind the occlusion so that switching to *pass* is possible soon, but one suddenly appears, the uncomfortable final part of the trajectory has to be executed. In this work, a *fallback* is defined as a deceleration over a threshold $a_{\text{fallback,dcc}}$, such as $a_{\text{fallback,dcc}} = 0.8 a_{\text{max,dcc,ego}}$.

I list some situations where R_1 and R_2 may be greater than 0. $R_1 > 0$ when

- Traffic participants behave beyond RSS assumptions (e.g. decelerate more than $a_{\text{max,dcc,obj}}$ or drive over $v_{\text{max,obj}}$).
- Traffic participants violate traffic rules (e.g. take way, cut in, or cross disregarding RSS safety requirements).
- Perception results are associated with high uncertainty (e.g. ghost objects or extremely large estimation error).

$R_2 > 0$ when

- The ego vehicle has an incorrect estimation of the turning, routing, or cooperation intentions of other traffic participants.
- The ego vehicle estimates the uncertainty in occlusions overly optimistic.

It is difficult to eliminate the *emergency risk* (R_1) entirely since adversarial behaviors cannot be avoided by AVs. However, it can be minimized by sacrificing

the *utility* and estimating the worst-case scenarios. The primary goal of this work, however, is not to eliminate R_1, but rather to ensure that the autonomous vehicles do not cause accidents. In the event of high R_1, the autonomous vehicles must adhere to traffic rules and be prepared for any emergency situation. Furthermore, it is assumed that the perception system operates reasonably well, and therefore, the third point is not considered in this work.

The *fall-back risk* R_2 can be eliminated, e.g. by always following *stop* or *early stop* actions, and can also be reduced without compromising *utility* too much, e.g. by having a better prediction module that generates a more accurate estimation of the environment. However, as the consequence of switching to *fallback* is not as severe as an emergency situation, human drivers often risk the *fallback* to be more efficient. For instance, when crossing a familiar intersection with occlusions where oncoming prioritized vehicles are rarely encountered from experience, they may tolerate one harsh brake in 100 fast crossings, rather than 100 soft brakes, among which 99 are unnecessary. In this example, the low probability of vehicles coming out of occlusion will reduce R_2 and allow for a more efficient crossing. The goal is to find a balance between *utility* and *driving risk* with recorded human driving trajectories and imitate how humans compromise between both.

R_1 and R_2 are high-level features that cannot be directly computed for an action, similar to U_2 and U_3. In order to estimate them, I propose an approach that will be explained in detail in Section 4.2.

4.1.4 Politeness

Courteous behavior during driving is important and can have a positive impact on overall traffic flow. When drivers exhibit courteous behaviors, it can reduce the likelihood of traffic jams and bottlenecks, leading to a more efficient use of road space. A skilled driver not only considers their own benefit but also behaves in a way that minimizes the impact on the comfort and utility of other drivers. The ability to plan suitable courteous behaviors is crucial for the public acceptance of autonomous systems. Moreover, human drivers and other traffic

participants expect cooperative behavior from autonomous vehicles. For instance, pedestrians crossing a zebra crossing often check for approaching vehicles even if they have priority. This is known as a two-step crossing as proposed in [Jai14]. If the vehicles are far away or decelerating early, pedestrians feel safe and proceed to cross. Conversely, if vehicles are not slowing down, pedestrians usually wait and observe their behavior. If a vehicle demonstrates an explicit courteous behavior, such as slowing down early, it encourages pedestrians to cross earlier, reducing the crossing time for both parties and increasing the utility of all traffic participants.

Moreover, cooperative behavior can also create a less stressful driving experience for all road users, avoiding potential hazard situations, for example yielding to merging vehicles early to avoid risky and aggressive cut-ins.

The level of *politeness* exhibited by the ego vehicle can be quantified by computing the average utility U_1 and average comfort C of the n surrounding traffic participants. Specifically, I define $P_1 = \frac{1}{n}\sum_{i=1}^{n} U_{1,i}$ and $P_2 = \frac{1}{n}\sum_{i=1}^{n} C_i$, where $U_{1,i}$ and C_i represent the utility and comfort of the i-th object, respectively. The trajectories of the surrounding traffic participants can be obtained through my proposed MCS approach, which will be elaborated in detail in Section 4.2.

4.2 Feature Estimation via Monte-Carlo Simulation (MCS)

Human drivers often consider the entire driving environment and make decisions that are not necessarily optimal but reasonable, taking into account all possible evolutions of the scene. However, finding optimal solutions in large multi-agent settings using methods such as MCTS can be complex due to the exponential growth of the action and observation space with the number of agents. To address this challenge, I propose semantic high-level action candidates that comply with traffic rules and RSS safety as suboptimal options. My approach focuses on achieving a certain degree of convenience and safety rather than to be optimal with an engineered reward function, which I believe is more

practical and achievable in real traffic scenarios. The goal is to select the most appropriate option from these choices.

The decision-making process relies on the features associated with each action, such as risk, utility, and comfort. To obtain these features, I use MCS to simulate the environment forward from the current scene, where the ego vehicle follows one of the high-level actions, and the surrounding agents react based on either a fixed prediction or an interactive driver model. The ego vehicle does not change its action unless the maneuver of the action is complete (e.g. completed lane change) or in case of a fall-back reaction (e.g. returning back to source lane in merging, or brake to stop before conflict zones). The simulation also includes sampling phantom vehicles from occlusions. Each episode of MCS can result in a different future due to the randomness in the environment model, which will be discussed in the next subsections. To obtain accurate feature values through MCS, the environment model should closely resemble reality. For each possible action candidate, the simulation is repeated with a sufficient number of episodes N, after which the feature values can be computed based on the simulation histories. To make decisions in real-time, each episode of MCS is limited by a simulation horizon $t_{\text{mcs,max}}$, with a discrete time step Δt, resulting in $m = \frac{t_{\text{mcs,max}}}{\Delta t}$ time steps.

4.2.1 Feature Estimation

In order to acquire U_1 (average speed), C (comfort), P_1 and P_2 (politeness), the trajectories of the ego vehicle and other traffic participants need to be available. U_2 (maneuver completion time), U_3 (maneuver success rate), R_1 (emergency risk) and R_2 (fall-back risk) are rather values that can not be estimated from the trajectories, but from other semantic information of the MCSs. I introduce the tag $*$ for the estimated feature vector $\mathbf{f}^*_{a_i} = [U^*_{1,a_i}, U^*_{2,a_i}, U^*_{3,a_i}, C^*_{a_i}, R^*_{1,a_i}, R^*_{2,a_i}, P^*_{1,a_i}, P^*_{2,a_i}]$ for a certain action $a_i \in \mathscr{A}$. $\mathscr{A}$ could be $\{a_{\text{free},1}, a_{\text{free},2}, a_{\text{free},3}, a_{\text{free},4}, a_{\text{free},5}\}$, $\{a_{\text{merge},1}, a_{\text{merge},2}, ..., a_{\text{merge},n}\}$ or $\{a_{\text{inter},1}, a_{\text{inter},2}, a_{\text{inter},3}\}$ depending on the scenarios.

After each MCS, the trajectories of all agents can be recorded, and U_1, C, P_1 and P_2 are computed once for this episode according to the equations in Section 4.1.

I propose to compute U^*_{1,a_i}, $C^*_{a_i}$, P^*_{1,a_i} and P^*_{2,a_i} by averaging U_1, C, P_1 and P_2 over all the episodes.

From one MCS, not only numerical values, i.e. U_1, C, P_1 and P_2, are obtained, but also other semantic information, such as whether a maneuver succeeds or a fall-back action has been executed. If the maneuver succeeds in i-th episode, the simulated completion time $t_{\text{mcs,finish},i}$ is recorded. Otherwise, $t_{\text{mcs,max}}$ is utilized for $t_{\text{mcs,finish},i}$. As a result, U^*_{2,a_i} (average normalized completion time of maneuver a_i) and U^*_{3,a_i} (success rate of maneuver a_i) can be formulated as follows

$$U^*_{2,a_i} = \frac{1}{N} \sum_{i=1}^{N} \left(\frac{t_{\text{mcs,finish},i}}{t_{\text{mcs,max}}} \right) \tag{4.3}$$

$$U^*_{3,a_i} = \frac{n_{\text{mcs,finish}}}{N} \tag{4.4}$$

where $n_{\text{mcs,finish}}$ represents the number of episodes where the desired maneuvers are completed. Similarly, R^*_{1,a_i} and R^*_{2,a_i} represent the ratios of episodes where the RSS safety is violated the ego vehicle has executed an emergency brake, and where a fall-back plan have been executed.

4.2.2 Modeling State Uncertainty

As outlined in Section 3.1, it is assumed that uncertainties are present in the states (e.g. position, velocity, acceleration) of both the ego vehicle and traffic participants. The localization and perception module provide established distributions for these uncertainties.

Upon initiating a fresh episode of MCS, the states of all traffic participants are stochastically sampled from their distributions. Consequently, in each episode of MCS, all agents within the scenario may commence with varying positions and velocities, among other factors, albeit within a confined scope.

4.2.3 Modeling Surrounding Vehicles

In MCS, the primary traffic participants modeled are human-operated vehicles. They are simulated to exhibit diverse behavior models, driving patterns, and navigational goals. Additionally, they are assumed to possess comprehensive knowledge of traffic regulations and an accurate perception of their surroundings. To preserve efficiency, the behavior models ought to be uncomplicated, facilitating rapid querying, since executing the environment across numerous MCSs can impose considerable computational demands. Consequently, more intricate learning-based behavior models, such as those founded on neural networks, are excluded from this study.

Car Following

In a car-following scenario on a singular lane, it is assumed that neighboring vehicles adhere to the IDM with predetermined parameters. Optionally, their IDM parameters may be fine-tuned based on long-term observations, provided the perception and scene comprehension modules possess the requisite capabilities. For example, if a vehicle maintains its velocity (velocity fluctuation of less than $1.5\,\frac{\mathrm{m}}{\mathrm{s}}$) for over 3 s, with a time headway exceeding 3 s to the preceding vehicle, the current velocity is employed as the desired velocity v_{des}. More sophisticated approaches to estimate the IDM parameters online can also be found [Kre22].

Cooperative Yielding to Merging vehicles

As the ego vehicle (AV) or other vehicles execute a mandatory merge and approach a gap, it is crucial to observe relevant vehicles on the target lane to evaluate their cooperativeness and determine whether to proceed with the initially identified gap.

Estimating the yielding intention is similar to computing the yielding motivation in Equation (3.12). I use the same logistic function

$$P(\text{yielding}|\hat{f}_{\text{Y}}) = \frac{1}{1 + e^{-\hat{\theta}_{\text{Y}}^{T}\hat{f}_{\text{Y}}}} \tag{4.5}$$

with a different feature vector $\hat{f}_{\text{Y}} = [a, d, t_{\text{TH}}, \dot{t}_{\text{TH}}]$, where another feature a is included, i.e. the acceleration of the vehicle whose yielding intention is estimated. The other features remain unaltered. The $\hat{\theta}_{\text{Y}}$ vector is retrained with the same data used for training the CIDM. The resulting $\hat{\theta}_{\text{Y}}$ is presented in Table A.7.

For initialization of the MCSs, the yielding intention of the target vehicle is sampled from the initial $P(\text{yielding}|\hat{f}_{\text{Y}})$. In essence, if a target vehicle exhibits $P(\text{yielding}|\hat{f}_{\text{Y}}) = 0.4$, it will be initialized with a cooperative intention in 40% of the episodes of MCSs and a non-cooperative intention in other episodes. During a single MCS, the yielding intention will be reevaluated every 1 second. A new yielding intention is deemed cooperative when $P(\text{yielding}|\hat{f}_{\text{Y}}) > 0.5$. Once the yielding intention is classified as cooperative, the target vehicle either conducts a lane change or decelerates based on Equation (3.13) with the CMOBIL model.

Mandatory Merging

Vehicles operating on a mandatory merging lane are presumed to adhere to a probabilistic merging model. This model yields a list of probabilities for all gaps that the merging vehicle approaches. The CGMP serves as the foundation, which can also be expressed probabilistically. Initially, the time required to complete the approach to the i-th gap is calculated as $t_{\text{merge},i}$, with the method presented in [Nau19]. Subsequently, the probability corresponding to the gap being approached by the vehicle is inferred by assuming a Boltzmann distribution, as illustrated in Equation (4.6). Merging intentions are assessed tactically at 1-second intervals within the MCS.

$$P_{\text{merge},i} = \frac{e^{-t_{\text{merge},i}}}{\sum_{j=1}^{n} e^{-t_{\text{merge},j}}} \tag{4.6}$$

Free Lane Change

Evaluating the lane change probability of nearby vehicles is crucial when conducting MCS for multi-lane roads. Lane change intention can be estimated using two approaches: model-based and movement-based [Wis17]. The former calculates the motivation for a lane change using a deterministic lane change model, such as MOBIL, while the latter estimates lane change probability based on vehicle states, e.g. lateral velocity, lateral distance to the centerline.

For the model-based approach, I estimate the lane change probability utilizing the acceleration gain from Equation (3.13) of MOBIL. I designate the net acceleration gain for keeping the current lane as $\Delta a_{\mathrm{k}} = 0$, and for changing lanes to the left and right as $\Delta a_{\mathrm{l}} = a_{\mathrm{gain,l}} - a_{\mathrm{th}}$ and $\Delta a_{\mathrm{r}} = a_{\mathrm{gain,r}} - a_{\mathrm{th}}$, respectively. The probability mass for each option is computed by

$$m_{\mathrm{MOBIL},i} = \frac{e^{\Delta a_{\mathrm{i}}}}{\sum_{j \in \{\mathrm{k,l,r}\}} e^{\Delta a_{\mathrm{j}}}}, \text{ for } i \in \{\mathrm{k,l,r}\} \tag{4.7}$$

In the movement-based approach, I devise another logistic regression model to calculate the lane change probability mass $m_{\mathrm{move},i}$, employing the movements (signed lateral distance to the centerline d_{c} and the lateral velocity v_{lat} of the target vehicle) as features.

Ultimately, the probabilities derived from both approaches are combined using the Mixing Rule of Evidence Theory [Sen02]

$$P_i = w_1 m_{\mathrm{MOBIL},i} + w_2 m_{\mathrm{move},i}, \text{ for } i \in \{\mathrm{k,l,r}\} \tag{4.8}$$

with w_1 and w_2 to be equally set to 0.5.

Intersection Behavior

Within the MCS, as surrounding vehicles approach an unsignalized intersection, they adhere to one of the rule-based yielding policies (B1, B2, and B3)

outlined in Section 3.4.2. Given that all these policies comply with RSS safety requirements, they should not result in collisions during MCSs.

Behavior Models for Different Driving Styles

In real-world scenarios, roads are populated by drivers exhibiting various driving styles. To simulate this diversity, I predefine three driving styles (*aggressive*, *normal*, *defensive*) based on aggressiveness, each associated with a distinct set of parameters. For example, *aggressive* agents prefer shorter time headways and higher desired velocities during car-following with IDM, exhibit lower probabilities of yielding to merge attempts with CIDM and CMOBIL, and have a smaller threshold a_{th} for lane changes with MOBIL. The parameters for all driving styles are summarized in Appendix A.1. Furthermore, when approaching an intersection, agents are assigned different rule-based yielding policies (B1, B2, or B3) according to their aggressiveness levels. Upon initializing the MCS, each vehicle is assigned an aggressiveness level, and subsequently, a specific parameter set and behavior model. Ideally, the aggressiveness level should be determined by tracking each vehicle's history, which can be achieved with an external module outside the scope of my work.

Trucks are assumed to be accurately classified by the perception module due to their significantly larger size. They typically exhibit less dynamic driving behavior and slower desired speeds but are more aggressive in terms of reduced cooperativeness concerning yielding intentions. Consequently, I introduce a separate parameter set specifically for trucks as well.

Routing Intention Estimation

Each vehicle has distinct goals, which are unknown to the ego vehicle without supplementary information (such as indicators). The goal is represented by a global route, comprising a sequence of lanelets on the map. An example is depicted in the left figure of Figure 4.1. As input for the MCS, the probabilities $\{P(r_i), \dots, P(r_I)\}_{v_n}$ of each vehicle v_n following all possible routes

$\{r_i, \dots, r_I\}_{v_n}$ for $I \in \mathbb{N}$ are necessary. In the absence of external prediction modules, equal probabilities are assigned. Upon initializing an episode of MCS, all vehicles are randomly assigned to one of the routes sampled from their probability distributions.

I employ a basic routing prediction method [Pet13] that generates routing probabilities by matching the vehicle's state distribution to the centerline of each route. Initially, the Mahalanobis distance $d(v_n, r_i)$ between the vehicle v_n and the route r_i is calculated, followed by the induction of probability by assuming a Boltzmann distribution.

$$P(r_i)_{v_n} = \frac{e^{-d(v_n, r_i)}}{\sum_{j=1}^{I} e^{-d(v_n, r_j)}}, \text{for } i \in \{1, \dots, I\} \tag{4.9}$$

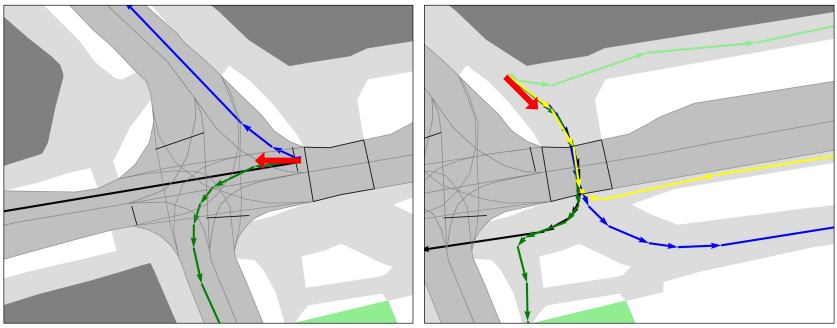


Figure 4.1: Different route options from lanelet2 map for vehicles (left) and pedestrians (right) given the pose (red arrow) (from [Wan23b], ©2023 IEEE).

In general, any prediction module (e.g., [Que18]) capable of providing the same information can be employed, enhancing the modularity of my method and rendering it prediction-agnostic.

4.2.4 Modeling Surrounding Pedestrians and Cyclists

In MCS, pedestrians may have multiple potential routes as well, and the estimation is conducted using the same method as for vehicles, as depicted in the right figure of Figure 4.1. When no zebra crossing is ahead, pedestrians are assumed to move constantly with the detected velocity following one of the routes. However, when they are closer to the zebra crossing than a threshold d_{min} but not yet on it, they will begin making crossing decisions. Once on the zebra crossing, they are simulated to cross straight ahead with a velocity sampled from a uniform distribution $\mathcal{U}(0.5v_{\text{max,p}}, v_{\text{max,p}})$.

I assume an interactive behavior model for pedestrians attempting to cross zebra crossings. The fundamental idea is that pedestrians tend to start crossing with a higher probability when the traffic is clear but will hesitate when the street is busy, or vehicles are driving fast and do not exhibit decelerating intentions explicitly, as discovered in this research [Sch22]. I fit a logistic regression model for predicting the crossing probability of pedestrians at zebra crossings using the inD dataset

$$P_{\text{cross}} = \frac{1}{1 + e^{-(\theta_{\text{p}}^T f_{\text{p}} + b_{\text{p}})}} \tag{4.10}$$

with θ_{p} as the weight vector, b_{p} the bias, and $f_{\text{p}} = [a_{\text{v}}, a_{\text{v,need}}, v_r]$ the feature vector, where a_{v} denotes the current acceleration of the closest oncoming vehicle to the zebra crossing, $a_{\text{v,need}}$ is its required acceleration to stop before the zebra crossing, and $v_r = 1 - \frac{v}{v_{\text{limit}}}$ is its normalized speed to the speed limit. The results of θ_{p} and b_{p} can be found in Table A.6. A crossing decision is made when $P_{\text{cross}} > 0.5$. Before stepping onto the zebra crossing, pedestrians will update their decision every 1 second of simulation time by computing P_{cross} again.

Cyclists are also recorded in the datasets. I do not introduce specific behavior models or intention estimation methods for cyclists but assign either pedestrian-like or vehicle-like behavior models. Cyclists perceived to be on the walkway will be modeled with pedestrian behaviors. Those driving on vehicle lanes

will be treated similarly to vehicles, albeit with cyclist-specific IDM and RSS parameters based on the dynamics of bicycles.

4.2.5 Modeling of Abnormal Behaviors

In Section 4.1.3, various scenarios are identified where traffic participants exhibit abnormal behaviors, resulting in emergency responses from the ego vehicle ($R_1 > 0$). To obtain a realistic estimation of R_1 (emergency risk), these behaviors must be incorporated into the MCS as well. I examined several instances of non-compliant traffic rule behaviors and quantified their occurrences in the datasets. The findings are presented in Table 4.1.

Table 4.1: Traffic rules non-compliant behaviors and their occurrence rate.

Type	Datasets	Total cases	Occurrence rate (%)
Exceeding 20% speed limit	InD, rounD	19671 vehicles	4.8
Exceeding RSS acceleration/deceleration	InD, rounD	19671 vehicles	0.01
Taking way disrespect RSS safety	InD, rounD	11081 intersections	17.1
Unexpected pedestrian crossing	InD	3093 pedestrians	0

In addition to assigning the three aggressive levels to vehicles, I also allocate abnormal behaviors to vehicles initialized in the MCS based on their actual occurrence rates in the datasets. This can be done, for example, by assigning an abnormal RSS parameter or a high desired speed, among other factors. It is worth to mention that no unexpected pedestrian crossings[1] were recorded in the datasets. Nonetheless, I still assign 0.1% of pedestrians in MCSs to deviate from their optional routes and potentially cross the street unexpectedly. Ideally,

[1] Those that occur outside of zebra crossings and resulted in other vehicles braking more than $a_{\text{soft,dcc,obj}}$.

this probability can be calculated by monitoring pedestrian movement through external modules.

4.2.6 Sampling of Phantom Vehicles from Occlusions

In MCS, vehicles are expected to approach and cross intersections following rule-based policies that are RSS safe, even in the presence of occlusions. However, simulating the FoV for every agent in the scene is computationally infeasible. Thus, I assume that other vehicles possess a perfect perception of the environment and make informed decisions. For the ego vehicle, the FoV polygon is simulated forward as it moves, taking into account currently perceived static obstacles and simulated dynamic obstacles (excluding pedestrians). With the limited FoV, the ego vehicle can only *pass* when the safety condition described in Section 3.2.4 is satisfied.

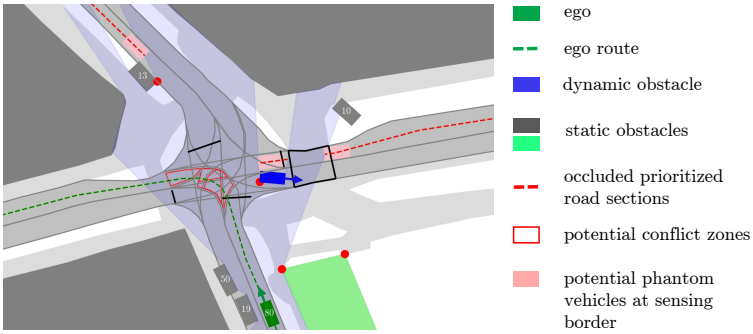


Figure 4.2: Possible phantom vehicles on occluded prioritized road sections.

During a MCS where the ego vehicles follows *fast approach*, it might not need to switch to fallback when no vehicle is simulated to emerge from occlusion. However, in reality, if the ego vehicle behaves the same way and expect a smooth transition to *pass* based on its anticipation from MCSs, it may be forced to fall back when vehicles appear from occlusion. To account for this potential fallback probability, phantom vehicles also have to be sampled from occluded road sections. The concept involves sampling based on the perceived traffic

density. To reduce the computational burden of MCS, phantom vehicles are only sampled from occluded sections of prioritized roads.

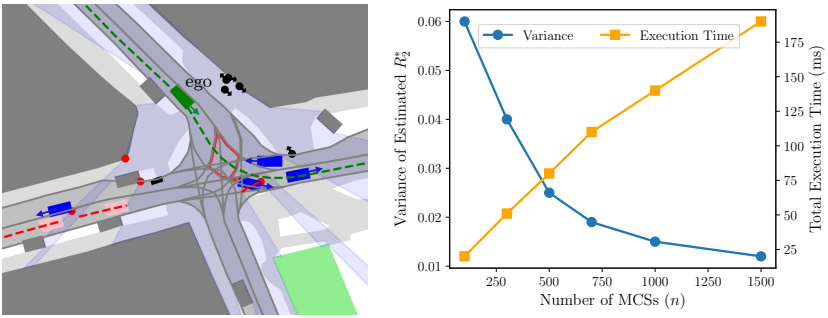
Figure 4.2 illustrates an example of possible occluded prioritized routes under the FoV, where phantom vehicles need to be sampled in MCS. For doing this, the traffic density $g = \frac{N_{\text{veh}}}{L}$ is first computed, where N_{veh} represents the number of the perceived dynamic vehicles in the scene (including the ego vehicle), and L is the total length of roads in all directions covered by the FoV. The expected number of phantom vehicles on each occluded section will be $n_{i,\text{exp}} = gl_i \in \mathbb{R}$ where l_i represents the length of the i-th section. The actual number of sampled vehicles is $n_{i,\text{sample}} = \lfloor \max(\mathcal{N}(n_{i,\text{exp}}, 1), 0) \rfloor$ and $\lfloor \rfloor$ is a floor function. After $n_{i,\text{sample}}$ is decided, phantom vehicles are sampled on the occluded sections with the following rules:

- They must keep at least 0.5 s time distance to both the already existing phantom vehicles on the section, and the visible vehicles outside of the section. If not possible, no new vehicle is sampled.
- They are initialized with a velocity following a uniform distribution $\mathcal{U}(0.8v_{\text{limit}}, 1.2v_{\text{limit}})$.
- Their behavior models and other parameters are allocated the same way as the visible vehicles.

4.2.7 Run-time Evaluation

In comparison to MCTS that is often used for solving POMDPs, which generally builds search trees and is not readily parallelizable, each individual MCS operates independently from others, enabling parallelization within a multi-core system. The feature values converge with an increasing number of MCSs. I examined the relationship between accuracy and speed in relation to the number of episodes for the MCS. This evaluation was conducted in a representative urban driving scenario featuring five pedestrians, one cyclist, and four nearby vehicles, along with occlusions generated by static and dynamic obstacles, as depicted in Figure 4.3a. The accuracy of the MCS can be characterized by the variance of the estimated feature values (e.g., estimated fall-back rate R_2^*)

across multiple runs, each with an identical number of episodes. As the number of episodes grows, the variance decreases. Conversely, the run-time will expand as the number of episodes increases. I performed the experiment on a laptop equipped with a Core-i7 8th-Gen Intel CPU and 8 threads, evaluating episodes in parallel. The outcomes are displayed in Figure 4.3b. I deem 500 episodes to be an appropriate balance between run-time and accuracy. The parameters for configuring the MCS can be found in Table A.5. It is worth mentioning that my approach can be further enhanced with customized hardware, such as CPU with numerous cores, or even Graphics Processing Units (GPUs).



(a) A typical urban driving scene.

(b) Variance and execution time with the number of MCSs.

Figure 4.3: Run-time and accuracy evaluation in a representative scenario.

4.3 Learning Policies from Datasets

As illustrated in Figure 3.1, following the approximation of features for each action using MCSs, a linear function with parameters w is employed to generate Q-values for the action. To learn a policy that balances feature values akin to human drivers—neither excessively egoistic (overemphasizing utility and comfort) nor overly cautious (prioritizing risk too much)—it is first necessary to determine human decision preferences. A neural network is not used to represent the Q-function because the aim is to explicitly present the weighting for each feature, thereby better understanding the decision and avoiding overfitting to the limited data.

Since humans may value features differently for lane changes, mandatory merging, and intersections, separate datasets are employed to learn distinct sets of weights for different scenarios. In total, three sets of weights are generated for free lane change, merging, and intersection policies. The highD [Kra18] and exitD [Moe22] datasets are used for the first two policies, while the InD [Boc20] and rounD [Kra20] datasets are applied for learning intersection behavior. The data generated for the intersection policy is divided into a 50% training set and a 50% test set. However, all data for parallel-lane policies are used for training, as evaluation is conducted in separate simulations rather than on the traffic scenes of the datasets.

4.3.1 Generation of Training Data

Each training data point d in the training dataset $\mathscr{D}$ consists of the data $[f^*_{a_1}, \dots, f^*_{a_n}]$ and the label $[P_{a_1}, \dots, P_{a_n}]$. Here, n represents the number of action candidates and $a_i \in \mathscr{A}$. The probability or preference P_{a_i}, reflecting how human drivers select each of the actions a_i, can be estimated based on their recorded trajectories. Distinct estimation methods are employed depending on the scenarios under consideration.

Intersecting Lanes

For unsignalized intersections, where only longitudinal decisions are made, matching the ground-truth trajectories of human drivers to the action candidates $a_{\text{inter},i}$ can be ambiguous. Consequently, I adopt a probabilistic matching approach rather than a deterministic one.

From MCSs, the average velocity profiles $\overline{V}_{a_i} = [\overline{v}_{a_i,t_0}, \overline{v}_{a_i,t_1}, \dots, \overline{v}_{a_i,t_{\text{mcs,max}}}]$ can be obtained, where $\overline{v}_{a_i,t_j} = \frac{1}{N}\sum_{k=1}^{N} v_{a_i,k,t_j}$, and v_{a_i,k,t_j} is the velocity of the ego vehicle at t_j time step of k-th MCS by following the action a_i.

For every valid frame[1] of every recorded vehicle in the dataset, N episodes of MCSs take place for each of the three actions, producing three distinct average velocity profiles. Afterward, an error ϵ_{a_i} between the ground-truth velocity profile $V_{\text{gt}} = [v_{\text{gt},t_0}, \dots, v_{\text{gt},t_{\text{mcs,max}}}]$ of the vehicle and the three generated velocity profiles $\overline{V}_{a_i}$ is computed with

$$\epsilon_{a_i} = \frac{1}{m} \sum_{t_j} \left| v_{\text{gt},t_j} - \overline{v}_{a_i,t_j} \right| \tag{4.11}$$

where m is the number of the time steps in a MCS episode. The probability P_{a_i} is estimated with

$$P_{a_i} = \frac{e^{-\epsilon_{a_i}}}{\sum_{a_j} e^{-\epsilon_{a_j}}}, a_i, a_j \in \mathcal{A} \tag{4.12}$$

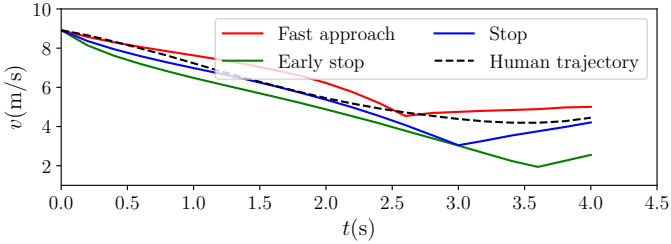


Figure 4.4: Example of the average velocity profiles from MCSs and the human-driven trajectory at 1 training frame (from [Wan23b], ©2023 IEEE).

As an illustration, I perform the MCSs with $t_{\text{mcs,max}} = 4\,\text{s}$ in the scenario depicted in Figure 4.2. The average velocity profiles for the three actions from MCSs, along with the ground-truth human trajectory, are recorded and displayed in Figure 4.4. The turning points of the three velocity profiles represent the transition between slowing down and *pass*, where the simulated FoV covers a sufficient portion of the prioritized lanes and safety conditions are met.

[1] Frames are valid when the vehicle in the ground-truth trajectory has not passed all intersections.

Utilizing Equation (4.12), the matched probabilities of the three actions are $P_{a_{\text{inter},2}} = 0.635$ (*fast approach*), $P_{a_{\text{inter},1}} = 0.364$ (*stop*), and $P_{a_{\text{inter},3}} = 0.001$ (*early stop*).

There are four intersections and three roundabouts containing in total 56 recordings of ca. 30 minutes in inD and rounD datasets. I evaluated 37016 valid frames and generated training data of the same size.

Parallel Lanes

In parallel-lane scenarios, acquiring ground-truth actions is straightforward. For mandatory merging, the ultimately achieved gap is assumed to be the ground-truth gap initially targeted. For lane changes, since I already know whether a lane change occurs in the trajectory, I assume a lane change decision is made when the lateral velocity exceeds 0.25m/s in the target direction. This value is greater than 98.1% of the lateral velocities in trajectories without lane changes. The lane change decision persists until more than half of the ego vehicle's geometry locates on another lane, i.e. $P_{a_{\text{free},4}} = 1$ or $P_{a_{\text{free},5}} = 1$. For other frames, I assume the decision is one of $a_{\text{free},1}$, $a_{\text{free},2}$ and $a_{\text{free},3}$, where $P_{a_{\text{free},4}} = P_{a_{\text{free},5}} = 0$.

I apply the same probabilistic matching method to match the ground-truth velocity profile to one of the velocity profiles of $a_{\text{free},1}$, $a_{\text{free},2}$ and $a_{\text{free},3}$, resulting in three probabilities $P_{a_{\text{free},1}} + P_{a_{\text{free},2}} + P_{a_{\text{free},3}} = 1$.

Training data is tactically extracted from the datasets with 1 s interval. Too small time interval results in similar states and outputs. From HighD and ExitD datasets, I obtain 23154 valid[1] training frames for the merging scenario and 253331 valid training frames for lane changes (22.05% of the frames have lane change labels).

[1] The first and last two seconds of each trajectory are discarded as the vehicle is at the edge of the FoV of the drone. Frames where no vehicle is on the target lane for merging are also considered invalid.

4.3.2 Loss Function and Learned Policies

I use softmax cross-entropy loss $\mathscr{L}$ for back-propagation and updating the parameters w, which is formulated for the entire training dataset as

$$\begin{aligned}\mathscr{L} &= -\sum_{d\in\mathscr{D}}\sum_{a_i\in\mathscr{A}} P_{a_i}\log\left(\frac{e^{-Q_{a_i}}}{\sum_{a_j\in\mathscr{A}} e^{-Q_{a_j}}}\right)\\ &= -\sum_{d\in\mathscr{D}}\sum_{a_i\in\mathscr{A}} P_{a_i}\log\left(\frac{e^{-\mathbf{w}^\top\mathbf{f}^*_{a_i}}}{\sum_{a_j\in\mathscr{A}} e^{-\mathbf{w}^\top\mathbf{f}^*_{a_j}}}\right)\end{aligned} \tag{4.13}$$

Consequently, the learned weights for the three scenarios (free lane change, mandatory merging, and intersection) are presented in Table A.8. Analogous to the feature values, the absolute values of the weights are also normalized between 0 and 1. In this manner, the human preferences for each feature are comparable and representative across different policies.

With these learned weights, I formulate three policies for different scenarios, i.e. Learned Merging Policy (LMP), Learned Lane Change Policy (LLCP) and Learned Intersection Policy (LIP).

For evaluation purposes, I incorporate an additional merging behavior where I do not permit arbitrarily high fall-back risk, approximated by the estimated fall-back rate R_2^* from MCS. As before, the action with the highest quality value Q is chosen, but only after those with fall-back risk higher than the threshold $R^*_{2,\text{th}}$ have been discarded. If the fall-back risks for all actions exceed $R^*_{2,\text{th}}$, the decision will be to merge into the very last gap. I set the threshold at $R^*_{2,\text{th}} = 0.2$, which is higher than the risk of 3.9% of all human merging decisions. This policy is termed Risk-Bounded Learned Merging Policy (RBLMP).

4.4 Learning Policies for Diverse Driving Styles

Incorporating the possibility for users of AD systems to select driving styles that closely resemble their own preferences enhances the overall user experience and fosters wider acceptance of autonomous vehicles. By offering a range of predefined driving policies, the AV can cater to the diverse needs of individual drivers, thereby making the technology more competitive in the market. This feature acknowledges the inherent variability in human driving behavior, and adapts the autonomous vehicle's decision-making process to better align with the expectations and comfort levels of its users.

The importance of providing customizable driving styles lies in the fact that drivers differ significantly in terms of risk tolerance, politeness, and preference for efficiency or comfort. An aggressive driver, for instance, may prioritize utility and be willing to accept higher risks, while a cautious driver may emphasize politeness and comfort over other factors. By allowing users to select from a range of predefined driving policies, autonomous vehicles can emulate the unique characteristics of human driving behavior, resulting in a more intuitive and personalized driving experience.

Furthermore, the flexibility to adjust driving styles online while the vehicle is in operation enables users to adapt the AV's behavior according to changing road conditions, traffic situations, or personal preferences. This feature not only enhances the vehicle's adaptability and responsiveness but also empowers users with a greater sense of control and trust in the system, which is essential for the successful integration of autonomous vehicles into daily life.

The driving style of the learned policy can be reflected in the weights assigned to the features. For example, an aggressive driver might assign more weight to utility and less to risk, while a cautious driver might prioritize politeness and comfort over other features. In the previous section, all training data for a given scenario were used to train a single policy, thus obtaining a universal set of weights. However, this data may encompass human drivers with different styles and driving preferences, such as both aggressive and cautious drivers. Consequently, this section proposes a method for clustering the training data

into several subsets that represent diverse driving preferences and deriving distinct sets of weights for each.

4.4.1 Clustering of Training Data

I introduce three driving styles depending on the level of aggressiveness. Each recorded trajectory ξ_i in the datasets belongs to one driver and all the training data in this trajectory $d \in \xi_i$ should be assigned to the same style.

First, I compute an average feature vector $\mathbf{f}_{\xi_i}$ for the trajectory ξ_i using

$$\mathbf{f}_{\xi_i} = \frac{1}{n} \sum_{d \in \xi_i} \sum_{a_j \in \mathcal{A}} \mathbf{f}^*_{a_j} P_{a_j} \tag{4.14}$$

where n is the number of training data in this trajectory. To recall the notation, $\mathbf{f}^*_{a_j}$ represents the estimated eight-dimensional feature vector from MCS for executing a_j at the data point d of the trajectory. P_{a_j} denotes the estimated probability of human driver executing a_j at this time point. This feature vector describes an average behavior of a trajectory ξ_i.

I argue that aggressiveness is a one-dimensional feature, where I can employ Principal Component Analysis (PCA) to reduce the eight-dimensional feature vector $\mathbf{f}_{\xi_i}$ to a single dimension. Subsequently, I apply k-means clustering to the reduced one-dimensional feature for all trajectories in the dataset, creating three clusters C_1, C_2, and C_3. It is not yet clear to which aggressive level the trajectories in each cluster belong. However, this information can be retrieved after training three policies using the trajectories in the three clusters, by looking at the resulting weights on different features. As mentioned before, the policy that weights utility more and risk less will be categorized to aggressive style; the same logic applies conversely.

Each cluster contains a specific number of trajectories (e.g., $\xi_i \in C_k$) and their corresponding training data $d \in \xi_i$. However, not all trajectories in a cluster may clearly belong to that cluster, particularly those at the margins of other clusters, which need to be excluded when training a stable and stylized policy.

To address this issue, I adopt a probabilistic assignment to each cluster, where three probabilities $P(\xi_i \in C_1)$, $P(\xi_i \in C_2)$, and $P(\xi_i \in C_3)$ are utilized. These probabilities are computed using:

$$P(\xi_i \in C_k) = \frac{e^{-d(\xi_i, C_k)}}{\sum_{j=1}^{3} e^{-d(\xi_i, C_j)}} \tag{4.15}$$

where $d(\xi_i, C_k)$ represents the distance of one trajectory ξ_i to the cluster C_k, which can be obtained with

$$d(\xi_i, C_k) = \frac{1}{n_{C_k}} \sum_{\xi_j \in C_k} d(\xi_i, \xi_j) = \frac{1}{n_{C_k}} \sum_{\xi_j \in C_k} d(\mathbf{f}_{\xi_i}, \mathbf{f}_{\xi_j}) \tag{4.16}$$

n_{C_k} is the number of trajectories in the cluster C_k and $d(\mathbf{f}_{\xi_i}, \mathbf{f}_{\xi_j})$ is the Euclidean distance of two feature vectors of two trajectories.

In each cluster, I eliminate the trajectories with the lowest 20% probability of belonging to that cluster ($P(\xi_i \in C_k) < 0.2$), and retain the remaining 80% of trajectories for training the policy.

4.4.2 Learned Stylized Policies

Upon clustering and filtering the datasets, I obtain three clusters of trajectories for all three scenarios, i.e. free lane change, merging and intersection. The number of remaining valid trajectories and their training data frames for the three scenarios are displayed in Table 4.2.

Table 4.2: Number of remaining valid trajectories and training frames. 1k = 1000.

	Free lane change			Merging			Intersection		
	C_1	C_2	C_3	C_1	C_2	C_3	C_1	C_2	C_3
Trajectories	19k	10k	4k	2.5k	953	361	6.4k	4.2k	1.2k
Frames	76k	41k	13k	8.9k	3.1k	1.2k	24k	15k	4.5k

Utilizing the same strategy and loss function as in Equation (4.13), I could train in total three policies for each scenario. Observations reveal that the weights obtained from training with the second cluster, C_2, are fast identical to those derived from utilizing the entire dataset (shown in Table A.8). This phenomenon can be attributed to the fact that the behavior typified by the second level of aggressiveness closely aligns with what is considered *normal* behavior. This similarity suggests that the universal policy, which is trained across the full dataset, effectively neutralizes the influence of both *aggressive* and *defensive* data, resulting in comparable weights.

Consequently, in each scenario, I derive an additional two policies corresponding to the *aggressive* and *defensive* categories, utilizing the remaining two clusters. The weights for all such stylized policies are subsequently calculated and detailed in Table A.9. It is important to note that the assignment of *aggressive* or *defensive* to each policy depends on the weights to specific features. For instance, the weight attributed to the *aggressive* merging policy for utility U_1 (average velocity) stands at 0.7, significantly higher than that for a defensive policy, which is 0.4. This aggressive policy also assigns a markedly lower weight to the fall-back risk R_2 (-0.4) in comparison to the defensive policy (-0.7), indicating a higher tolerance for fall-back rates. The policies thus formulated are designated as Learned Aggressive Merging Policy (LAMP), Learned Defensive Merging Policy (LDMP), Learned Aggressive Lane Change Policy (LALCP), Learned Defensive Lane Change Policy (LDLCP), Learned Aggressive Intersection Policy (LAIP), and Learned Defensive Intersection Policy (LDIP).

5 Evaluation

As outlined in Section 1.2, the objective of this research is to develop a high-level decision-making approach for uncertain environments. While prioritizing RSS safety, the resulting decisions should emulate human-like patterns, striking a balance between efficiency, comfort, risk, and courtesy. Since conducting extensive on-road tests is infeasible, I propose evaluating my approach in close-to-real simulation environments. To showcase the potential of each policy comprehensively, evaluations are performed in various scenarios specifically designed for the relevant policies. Additionally, case studies are conducted to visualize the trajectories of the output behaviors and demonstrate the capabilities of my approach in detail.

The evaluation scenarios are divided into parallel-lane scenarios in Section 5.1 and intersecting-lane scenarios in Section 5.2, aligned with developed policies. The former encompasses mandatory merging, exiting from main roads, and free lane change scenarios. The latter includes situations such as unsignalized intersections, zebra crossings, and roundabouts, with and without occlusions. Lastly, in Section 5.3, I implement real-time capable software and integrate it into our automated driving pipeline, testing it in both a closed-loop simulation and on our experimental vehicle.

5.1 Evaluation on Parallel-lane Scenarios

I divide the parallel-lane scenarios into on-ramp merging, free lane change, and freeway exiting. To evaluate the policies for these scenarios, I developed a versatile and modular simulation environment. First, I provide a brief overview

of the simulation setup. Subsequently, I assess the behaviors in specific challenging scenarios as well as on randomly generated traffic situations. The latter approach enables the derivation of various statistics and metrics, such as the frequency of successful merges and fallback occurrences.

5.1.1 Evaluation Simulation

While existing simulators (e.g., SUMO [Lop18]) can simulate highway or multi-lane scenarios, implementing customized behavior for other agents is not straightforward. Consequently, I developed my customized simulation. This simulation allows for the manual design of road networks (including an arbitrary number of main lanes, merging lanes, and exit lanes) with customizable parameters (shape, width, length, speed limit, etc.). Agents exhibiting any designed behavior with various parameters can be initialized on each lane. After running the simulation, each agent can sense its surrounding environment within a predefined range and move according to its customized behavior. The simulation features appropriate visualization and allows for recalling agents' history, in order to highlight interesting showcases.

Lane-based Behavior Models

For evaluation purposes, one or two autonomous agents are initialized with my learned behaviors (LLCP, LMP, or RBLMP), while other agents follow the rule-based policies (detailed in Section 3.4.1) depending on their respective lanes. Environmental agents are assigned the CMOBIL policy on main lanes and the CGMP policy on merging lanes. These agents will switch their behavior model upon transitioning to a different lane type. For example, a merging agent initialized with CGMP will adopt the CMOBIL policy upon completing the merge and entering the main lanes.

Initialization of Parameters

Parameters for IDM, CMOBIL, and RSS of non-autonomous agents in the simulator are randomized and hidden from the autonomous agents. Autonomous agents can only estimate the intention, behavior models, and parameters of other agents before initiating the MCS, as explained in Section 4.2.3. This introduces an estimation error, which inevitably exists when estimating real-world situations. Moreover, agents exhibiting abnormal behavior can be simulated, such as those with unrealistic and risky IDM parameters (e.g., a desired time headway of only 0.3 s) or RSS parameters with $a_{\text{max,dcc,obj}} = -0.5\,\frac{\text{m}}{\text{s}^2}$. To closely emulate real traffic, vehicles on the rightmost main lane and merging lane are partially (30%) initialized as trucks, which possess larger geometries and distinct behavioral parameters. However, autonomous agents are assumed to be able to classify these trucks and estimate their behavior differently for MCS, following Section 4.2.3.

Reproducible Random Traffic

To ensure a fair comparison of different policies, a large number of random traffic situations and their road corridors are generated and saved in configuration files. These files can be loaded by the simulation for each policy, producing identical initial scenes. Reloading only the states of the environmental agents is not enough, as their hidden parameters are also randomized (e.g. IDM, CMOBIL, and RSS parameters). To reproduce the exact same parameters when comparing different policies, random seeds that are utilized to generate these parameters for each environmental agent are stored in the configuration files as well. However, the subsequent development of the traffic scene will diverge depending on the autonomous agents' behaviors.

5.1.2 Compared Policies and Metrics

I compare my learned policies with the rule-based baseline policy for each parallel-lane scenario and introduce various metrics derived from the massively simulated random traffic.

For on-ramp merging, LMP, LAMP (aggressive), LDMP (defensive), and RBLMP (risk-bounded) are evaluated against the baseline CGMP. In this scenario, average merging time and number of fallback occurrences n_{fallback} serve as metrics.

For free lane change, the learned policies LLCP, LALCP (aggressive), and LDLCP (defensive) are compared with the rule-based policies CIDM and CMOBIL. CIDM only allows yielding for other cut-in vehicles by decelerating, not lane change. In this scenario, the number of lane changes n_{lc}, n_{fallback}, average U_1 (speeding), and average C (comfort) of the ego vehicle are considered significant metrics.

I treat freeway exiting as a scenario similar to on-ramp merging, where one target lane exists and the lane change intention is signaled early enough for other vehicles to cooperate and create a gap. As such, all merging policies are suitable and evaluated. If more than one merge is required (e.g., if the ego vehicle is on the leftmost lane), the merging policy can be executed repeatedly. The only modification when applying the learned policies for exiting is that, during MCS construction, the ego lane is assumed to end shortly[1] and no later than the end of the exit opening. With this heuristic, the ego vehicle will not always pursue the first perceived gap due to the risk associated with my assumed lane ending. Important metrics for this scenario are similar to those for on-ramp merging, including average exiting time and n_{fallback}.

5.1.3 Evaluation on Generated Traffic

On-Ramp Merging

I generate 500 random scenes with two main lanes on the left side and one merging lane on the right side. The lanes are randomly assigned one of the three speed limits: $60\,\frac{\text{km}}{\text{h}}$, $80\,\frac{\text{km}}{\text{h}}$, and $100\,\frac{\text{km}}{\text{h}}$, resulting in varying merging lane lengths. Random agents are generated on the main lanes with two different densities, represented by the time headway t_{HW} between vehicles following two

[1] The remaining distance is just enough for the ego vehicle to stop fully with $-2\,\frac{\text{m}}{\text{s}^2}$

uniform distributions $\mathcal{U}(0.8\,\text{s}, 1.4\,\text{s})$ and $\mathcal{U}(1.2\,\text{s}, 2.0\,\text{s})$. Two vehicles with a 1 s time headway are initialized on the merging lane, with the same merging policy. Each of the three policies is evaluated once across the exact same 500 initial scenes to ensure comparability of results. Figure 5.1 presents an example of the merging evaluation simulation.

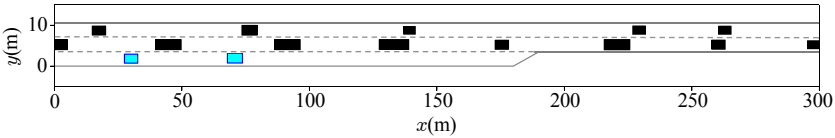


Figure 5.1: An example of the merging evaluation scene. The two blue rectangles represent two merging vehicles with the same policy. On the main lanes, the bigger black rectangles represent trucks, and others are random vehicles.

Finally, some statistics are summarized in Table 5.1 after all 1000 merges.

Table 5.1: Statistics for merging on 500 random traffics.

	Policies	merging time (s)	n_{fallback}
$t_{\text{HW}} \sim \mathcal{U}(0.8\,\text{s}, 1.4\,\text{s})$	CGMP (rule-based)	8.142	181
	LMP (learned policy)	**6.212**	22
	LAMP (aggressive style)	**6.091**	23
	LDMP (defensive style)	**6.235**	19
	RBLMP (risk-bounded)	6.582	**11**
$t_{\text{HW}} \sim \mathcal{U}(1.2\,\text{s}, 2.0\,\text{s})$	CGMP	5.314	124
	LMP	**4.482**	19
	LAMP	**4.461**	17
	LDMP	**4.498**	**11**
	RBLMP	4.591	**9**

As expected, merging in denser traffic results in more fallbacks and increased merging time. Notably, all learned policies enable significantly faster merging and yield far fewer fallbacks than the baseline policy CGMP. By bounding the risk, merges are slightly slower but produce fewer fallbacks than without risk constraints. Applying different driving styles, the average merging time and n_{fallback} fluctuate within a small range. A more aggressive driving style leads

to reduced merging time without sacrificing risk (n_{fallback}). In contrast, a more cautious behavior lowers risk at the cost of slower merging. The reason why the slightly more aggressive merging style LAMP does not substantially increase the failure rate of merging might be that, during merging in dense traffic, approaching a gap aggressively can also enhance the chance of success, as other vehicles are more likely to open the gap.

Free-way Exiting

The map is constructed with three main lanes and one exiting lane on the right side. Random vehicles are generated on the map with a density of $t_{\text{HW}} \sim \mathcal{U}(1.2\,\text{s}, 1.8\,\text{s})$. I initiate the ego vehicle on the far left lane at two distances: 200 m and 500 m before the exit opening ends, with 1000 simulations each. Note that exiting is considered successful when the ego vehicle is positioned in the exit lane before the exit opening ends. Figure 5.2 illustrates an example of the exiting scene in evaluation simulation and Table 5.2 presents the simulation results.

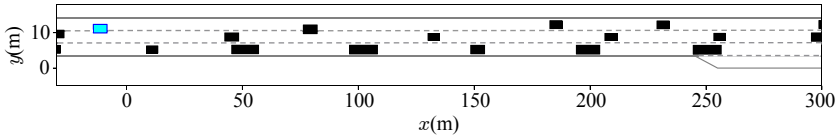


Figure 5.2: An example of the exiting scenario. The blue vehicle is initialized randomly on one of the two left main lanes and tries to merge to the exiting lane. Distance to the exiting lane is 200 m in this example.

A similar pattern to merging is observed, with learned policies generally outperforming CGMP. By bounding risk, exits are slightly slower but produce fewer fallbacks. The earlier the ego vehicle begins merging to the right, the less risky the exit will be. Therefore, it is recommended to initiate exiting with sufficient buffer if the route is known in advance from the map.

Table 5.2: Statistics for exiting on 1000 random traffics.

	Policies	exiting time (s)	n_{fallback}
200 m remaining distance	CGMP (rule-based)	12.145	73
	LMP (learned policy)	**11.905**	37
	LAMP (aggressive style)	**11.857**	39
	LDMP (defensive style)	**11.913**	**35**
	RBLMP (risk-bounded)	11.935	**36**
500 m remaining distance	CGMP	27.857	24
	LMP	**26.462**	5
	LAMP	**26.433**	6
	LDMP	**26.502**	**4**
	RBLMP	26.561	**3**

Free Lane Change

The random traffic settings for evaluating lane change policies have some differences compared to those for merging policies. The map contains three main lanes and one merging lane. Two vehicles are initiated on the merging lane, driving with CGMP. On the main lanes, vehicles are generated with random time headway $t_{\text{HW}} \sim \mathcal{U}(1.4\,\text{s}, 2.2\,\text{s})$. Only one vehicle on the main lanes is randomly selected as an autonomous agent and drives with the compared lane change policies. In total, 1500 scenes are generated and simulated twice. In the first round, all vehicles have recommended parameters with minor randomness. The second round is more challenging, with 20% of vehicles assigned inappropriate IDM and RSS parameters (desired time headway $T_{\text{d}} = 0.3\,\text{s}$ and maximum deceleration of others $a_{\text{max,dcc,obj}} = -0.5\,\frac{\text{m}}{\text{s}^2}$), resulting in risky follow-driving behaviors, close-to-crash lane changes, and merges. If this occurs, the vehicle behind must execute an emergency brake, which counts as a fallback as well. Figure 5.3 illustrates an example of the free lane change scene in evaluation simulation and Table 5.3 presents the statistical results.

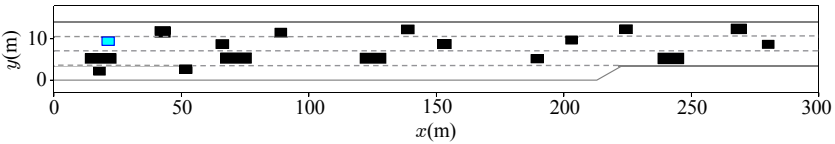


Figure 5.3: An example of the free lane change scenario. The blue vehicle is initialized randomly on one of the three main lanes.

Table 5.3: Statistics for evaluation on 1500 random traffics. (avg. = average)

	Policies	n_{lc}	n_{fallback}	avg. U_1	avg. C_1
0% abnormal agents	CIDM	0	**0**	0.901	0.828
	CMOBIL	1220	11	0.905	0.809
	LLCP (learned policy)	280	**0**	**0.911**	**0.829**
	LALCP (aggressive style)	263	**0**	**0.916**	0.817
	LDLCP (defensive style)	270	**0**	0.906	**0.843**
20% abnormal agents	CIDM	0	26	0.896	0.815
	CMOBIL	1288	23	0.905	0.799
	LLCP	277	**7**	**0.906**	**0.815**
	LALCP	252	9	**0.911**	0.803
	LDLCP	261	**6**	0.902	**0.828**

Overall, the learned policies of all styles lead to significantly fewer lane changes than CMOBIL but still provide higher velocity and comfort. Another noticeable advantage of the learned policies is that they result in substantially less risky driving behavior with far fewer fallbacks. In the first round, where others drive safely, CIDM has a reasonable 0 fallbacks because others do not perform unsafe cut-ins. However, CMOBIL has 11 fallbacks. After reproducing the simulations, I discovered that this occurs when the ego vehicle intends to change lanes and another vehicle on the third lane starts a lane change towards the same lane. The lane changes of both vehicles are initially safe, but as soon as they appear on the target lane simultaneously, the one behind becomes unsafe. In this case, it is unclear which vehicle is at fault. Note that for the free lane change evaluation, I apply the RSS safety rule in Section 3.2.2 without relaxation as recommended. However, this edge case is not covered and could

prompt further investigation, which is beyond the scope of this work. Nonetheless, the learned policies can prevent this risky situation and have 0 fallbacks, as they recognize the lane change intentions of others and can abort their lane change early on. In the second round with aggressive agents, it appears that even maintaining lane (CIDM) can be risky due to unsafe cut-ins from others. However, the learned policies enable the vehicle to change to safer lanes or execute deceleration action $a_{\text{free},2}$ as soon as the aggressive cut-in intention is recognized, leading to far fewer fallbacks.

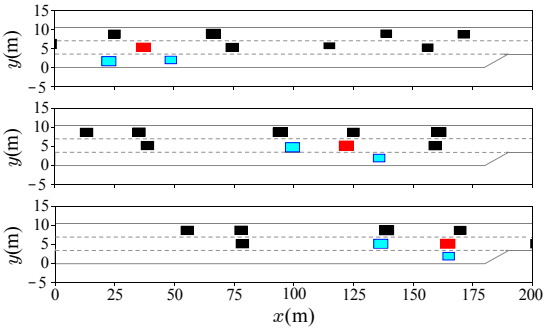
As expected, a more aggressive driving style (LALCP) achieves higher utility but lower comfort than LLCP, with fewer lane changes. In contrast, a more cautious style (LDLCP) increases comfort by slightly compromising velocity.

5.1.4 Challenging Scenarios

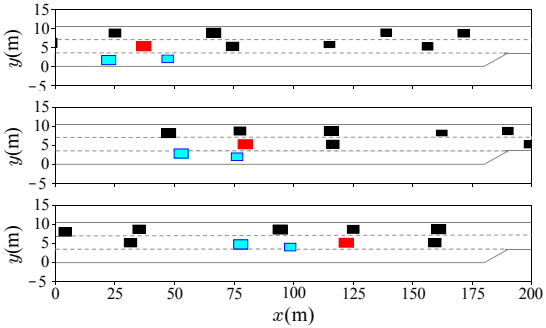
On-Ramp Merging

The primary challenge of merging in dense traffic is recognizing the cooperative intentions of vehicles on other lanes and selecting the appropriate gap to attempt merging.

I present an example in Figure 5.4, where two blue agents try to merge to the main lane. Figure 5.4a shows three moments of driving with CGMP, while Figure 5.4b with LMP and RBLMP that generate the same output behavior. In Figure 5.4a, the first merging vehicle using CGMP insists on merging in front of the red agent and eventually has to fall back and stop, as it cannot estimate the red agent's yielding intention. In contrast, with LMP and RBLMP, the autonomous agents can successfully complete the merging process (Figure 5.4b). At the beginning, the red vehicle's cooperation intention is unclear, so the first autonomous agent tries to merge in front. However, as the simulation proceeds and the merging lane nears its end, the fallback probability for merging in front increases. Consequently, the decision switches to merging behind the red vehicle just in time.



(a) $t = 1\,\text{s}, 5\,\text{s}, 7\,\text{s}$



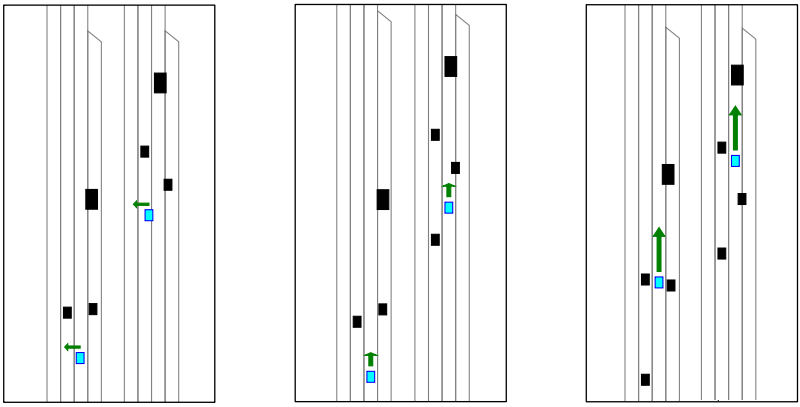
(b) $t = 1\,\text{s}, 3\,\text{s}, 5\,\text{s}$

Figure 5.4: An example of merging scenario where the blue rectangles are merging agents, and others are surrounding agents. (a) Merging agents follow CGMP. (b) Merging agents follow LMP and RBLMP.

Free Lane Change

Typical free lane changes occur when the ego vehicle desires a higher speed but is blocked by slower vehicles in front, while other lanes are free. Both CMOBIL and LLCP can achieve this behavior through their design. Therefore, I select some more challenging scenarios that demonstrate the superiority of LLCP over CIDM and CMOBIL.

Figure 5.5 showcases three scenarios in which autonomous vehicles governed by LLCP demonstrate courteous behavior, facilitating smoother cut-ins for environmental merging vehicles. In Figure 5.5a, a left lane change minimally impacts the utilities of both the ego vehicle and the merging vehicle. In situations where a lane change is not feasible (Figure 5.5b and Figure 5.5c), LLCP either decelerates ($a_{\text{free},2}$) or accelerates ($a_{\text{free},3}$), to enable more seamless merging. CMOBIL can also execute deceleration or lane change, depending on the total acceleration gain, provided the ego vehicles are aware of the merging intentions. Nonetheless, CMOBIL is incapable of performing acceleration, as depicted in Figure 5.5c.



(a) Lane change ($a_{\text{free},4}$) for merging vehicles.

(b) Decelerate ($a_{\text{free},2}$) for merging vehicles.

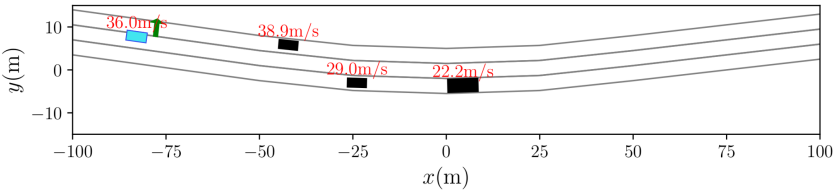
(c) Accelerate ($a_{\text{free},3}$) for merging vehicles.

Figure 5.5: The autonomous agent (blue rectangle) with LLCP performs different courteous behavior (represented by green arrows) to enable a smoother merging of other vehicles.

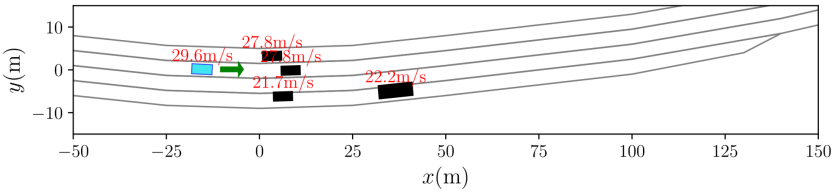
I conducted further evaluations of the LLCP in various challenging scenarios, which the CIDM and CMOBIL are unable to address. In Figure 5.6a, the ego vehicle employing LLCP executes a left lane change to mitigate the risk of a vehicle behind the truck abruptly overtaking without prior indication. The likelihood of this event is not negligible, as the vehicle approaches the truck at a high relative speed. I perform MCSs for maintaining the current lane ($a_{\text{free},1}$) and changing to the left lane ($a_{\text{free},4}$) at this moment. The R_2^* and U_1^* values for

$a_{\text{free},1}$ are 0.11 and 0.79, whereas $a_{\text{free},4}$ yields 0 and 0.86, making it a safer and faster option. Lacking an explicit lane change signal from the vehicle behind the truck, the CMOBIL is incapable of executing this proactive maneuver.

Figure 5.6b presents an additional scenario in which the ego vehicle is impeded by a slow-moving vehicle ahead. Concurrently, another slow vehicle occupies the left lane, rendering a left lane change unproductive. The CMOBIL suggests a right lane change, which yields the greatest acceleration gain. Nevertheless, this decision carries potential risks, as the two vehicles on the merging lane could complete their merge anytime, necessitating increased braking by the ego vehicle. The MCS generates $R^*_{2,a_{\text{free},5}} = 0.34, U^*_{1,a_{\text{free},5}} = 0.68$ for a right lane change and $R^*_{2,a_{\text{free},2}} = 0, U^*_{1,a_{\text{free},2}} = 0.77$ for deceleration. Consequently, based on the LLCP, $a_{\text{free},2}$ is considered a superior alternative.



(a) A foresighted lane change



(b) Decelerate instead of changing lane to right

Figure 5.6: Two challenging scenarios that can be tackled by LLCP but not CMOBIL. The ego vehicle is shown in blue and others in black. The velocities are attached to the rectangles. One truck is represented by a slightly bigger black rectangle (from [Wan23a], ©2023 IEEE).

5.2 Evaluation on Intersecting-lane Scenarios

To demonstrate the superiority of the LIP over rule-based policies, I first evaluate them in an interactive simulation built upon datasets. I further showcase the generalization and scalability of my approach by evaluating it on a new roundabout from the Interaction dataset [Zha19], which was not utilized for training. I classify evaluation scenarios into three categories: unsignalized intersections with mild occlusions due to static obstacles, intersections with severe occlusions, and roundabouts. I perform quantitative evaluations and present interesting case studies comparing the policies in the end.

5.2.1 Evaluation Simulation

Existing simulators and benchmark tools include Carla [Dos17], which offers realistic environment representation and sensor simulation but faces challenges in integrating lanelet2 maps and requires significant effort to design a low-level control interface. CommonRoad [Alt17] serves as another benchmark for evaluating planning algorithms, providing simulations that are partially recorded from real traffic and partially hand-crafted to create hazardous situations. However, it mainly focuses on evaluating motion planning algorithms' cost functions and it is less relevant for my high-level behavior generation approach. BARK [Ber20] aims to provide a realistic, interactive simulation environment initiated from datasets with reactive surrounding agents following several predesigned behavior models. Its design best meets my requirements. However, it does not support lanelet2 maps, and its benchmark function is not yet fully released. P3IV [Tas21] supports lanelet2 format and is able handle occlusions and uncertainty of objects, but it is not aimed for learning-based methods.

Evaluating behavior models in entirely offline datasets has the advantage of realistic agent movement based on recorded trajectories. However, the drawback is that they do not react to the automated ego vehicle, which deviates from the ground truth starting from the second simulation step. Following BARK's concept, I construct a similar simulation using the test data of the inD and rounD datasets, and Interaction datasets that is not used for training.

After running the simulation, one vehicle in the scene is designated as the ego vehicle, following a specific driving policy and replacing its original trajectory. Other agents behave according to their recorded trajectories but the control will be overtaken by AD policies once any of the following conditions are met:

- The distance to its front agent is less than the RSS-safe distance computed using relaxed RSS parameters.
- Starting to cross the intersection if the crossing is not RSS-safe according to relaxed RSS parameters.

Overridden AD agents are randomly assigned an aggressive level, and their behavior models and parameters are initialized according to Section 4.2.3.

Due to the limited FoV of the recording drone, new agents might spawn from the scene edge, potentially colliding with overridden agents. I do not spawn these new agents if any of the following conditions are met:

- The spawned position is already occupied by other agents.
- The distance of the spawning position to its front or following agent is less than the RSS safe distance computed using relaxed RSS parameters.

Relaxed RSS parameters enable more aggressive driving, for example, assuming a larger $a_{\text{max,dcc,ego}}$ and a smaller $t_{\text{TZC,min}}$, among other factors. These parameters are presented in Table A.2.

With these modifications, the simulation closely approximates reality while populating the environment with reactive agents that strive to avoid collisions and adhere to traffic rules. To validate my simulation, I compared it to one that only replays the offline datasets for other vehicles. Each vehicle in the datasets is treated as the ego vehicle once and follows the learned policy, while others either behave reactively or strictly follow their recorded trajectories. After processing the entire test dataset, I recorded the number of resulting collisions between all agents. On average, the number of collisions for simulating one AD agent is reduced from 2.3 to 0.05 by introducing my modifications. The remaining collisions primarily result from edge cases, such as the front vehicle not being clearly identified as they drive in the center of two lanes, or bounding boxes from the original trajectories having slight overlap due to inaccurate

dataset pre-processing. Despite these limitations, my simulation is considered sufficiently realistic, as on average, only 9.5% of agents in the scene have been overridden, and the majority of other agents continue to follow their trajectories.

5.2.2 Compared Policies and Metrics

I compare six policies, five of which have already been explained, including rule-based ones (B1 and B2) and learned ones for different driving styles (LIP, LAIP, and LDIP). To demonstrate the impact of the prediction module on the quality of the learned policy, I employ a superior proof-of-concept routing prediction module instead of Equation (4.9) and apply LIP to it, resulting in the sixth policy, i.e., Learned Intersection Policy with Better Prediction (LIPBP). Since all the ground-truth routes of the surrounding agents are known, their future 3 s of points on the ground-truth routes with a 0.5 s interval are used for matching to their possible routes, resulting in significantly better prediction accuracy. Mathematical details are omitted which are similar to Equation (4.9). My objective is not to provide an exceptional prediction module but to demonstrate how a good one can enhance planning quality.

I evaluate the following four metrics:

- *Mean Distance Error (MDE)* to the ground-truth trajectory: to show the human-likeness of each policy.
- *Average velocity*: proportional to the inverse of average crossing time, but includes the parts after crossing the intersection and is more representative of the overall utility.
- *Fall-back ratio*: how often does the ego agent need to switch to *fallback*.
- *Velocity gain of the traffic*: how the average velocity of traffic flow is improved compared to policy B1.

To ensure a fair comparison of the policies, the simulation time for all evaluated policies of a vehicle is equal, which is the duration of the ground-truth trajectory.

5.2.3 Evaluation on Test Scenarios

Intersections with Mild Static Occlusion

In the inD dataset, there are three intersections where occlusions caused by static obstacles are not severe enough to hinder driving. A total of 458 vehicles are evaluated, all of which encountered at least one yielding intersection.

Table 5.4: Statistics for simulation evaluation at intersections with mild static occlusion.

Policies	MDE (m)	Average velocity ($\frac{m}{s}$)	Fall-back ratio (%)	Velocity gain of the traffic ($\frac{m}{s}$)
B1 (normal stop)	**8.46**	6.19	**5.02**	0
B2 (fast approach)	10.51	**6.91**	17.14	-0.02
LIP (learned policy)	9.94	6.74	9.72	0.003
LIPBP (better prediction)	10.25	**6.81**	9.06	0.002
LAIP (aggressive style)	9.06	**6.81**	15.9	**0.03**
LDIP (defensive style)	**8.57**	6.32	**5.9**	0.002

Quantitative outcomes are displayed in Table 5.4. B1 demonstrates the most human-like behavior, albeit with a relatively low mean velocity. As it employs *stop* for the approaching action, it exhibits the lowest fall-back ratio. The non-zero fall-back ratio for B1 arises from instances where vehicles start being recorded close to an intersection and maintain high speeds, while the B1 policy has already switched to *pass* with satisfied RSS safety conditions, triggering fallbacks. Conversely, B2 (*fast approach*) attains the highest overall velocity but suffers from the largest fall-back ratio, making it the least human-like policy. Furthermore, traffic flow is negatively affected.

Both LIP and LIPBP present similar performance characteristics. They exhibit more human-like behavior and achieve substantially lower fall-back ratios than B2, accompanied by a minor decrease in velocity. Considering politeness within the features leads to an enhanced overall traffic flow. Utilizing superior prediction results in a reduced fall-back ratio and improved mean velocity.

An aggressive learned driving style, LAIP, showcases a behavior that lies between LIP and B2, with higher utility than LIP but more fall-backs, akin to B2. In contrast, LDIP embodies behavior more closely aligned with B1.

Intersections with Severe Static Occlusion

In the inD dataset, there exists an intersection where buildings and parked vehicles severely occlude all arms of the intersection, as depicted in Figure 4.2. Additionally, a pedestrian crossing on the west arm of the intersection, marked with a black polygon, requires the ego vehicle to yield to pedestrians. A total of 824 vehicles are evaluated, each having encountered at least one yielding intersection or pedestrian crossing. Quantitative results can be found in Table 5.5.

Table 5.5: Statistics for simulation evaluation for intersections with severe static occlusion.

Policies	MDE (m)	Average velocity ($\frac{m}{s}$)	Fall-back ratio (%)	Velocity gain of the traffic ($\frac{m}{s}$)
B1 (normal stop)	9.98	5.33	**2.5**	0
B2 (fast approach)	**9.38**	**5.86**	30.3	**0.09**
LIP (learned policy)	9.51	5.66	**7.3**	0.06
LIPBP (better prediction)	9.55	5.63	**6.6**	0.07
LAIP (aggressive style)	**9.05**	**5.83**	28	**0.08**
LDIP (defensive style)	9.63	5.56	**3.6**	0.05

Under conditions of severe occlusion, B1 emerges as the least human-like policy, achieving the lowest velocity. In contrast, B2 is the fastest and most human-like policy, but with a substantial 30.3% fall-back ratio, resulting in a highly unpleasant driving experience. Both LIP and LIPBP considerably reduce the fall-back ratio while slightly increasing MDE and reducing velocity. Policies that attain higher average velocities also facilitate smoother traffic flow. Employing a superior prediction module assists in reducing the fall-back ratio by better estimating the turning intentions of prioritized vehicles. Different driving styles, such as LAIP and LDIP, exhibit expected patterns in average velocity and fall-back ratio.

Roundabouts

In the rounD dataset, there are three roundabouts where the streets are predominantly clear in all directions. 4282 vehicles are evaluated in these scenarios.

The quantitative results are provided in Table 5.6. The simulation outcomes are similar to the intersections with severe static occlusions. The learned policies attain average velocities and manage risk at varying levels, depending on their driving styles. Utilizing an improved prediction module results in enhancements across most performance metrics.

Table 5.6: Statistics for simulation evaluation for roundabouts.

Policies	MDE (m)	Average velocity ($\frac{m}{s}$)	Fall-back ratio (%)	Velocity gain of the traffic ($\frac{m}{s}$)
B1 (normal stop)	6.40	5.41	**0.8**	0
B2 (fast approach)	**6.31**	**5.65**	24.2	**0.025**
LIP (learned policy)	6.36	5.53	**4.0**	0.004
LIPBP (better prediction)	**6.31**	5.59	**4.8**	0.013
LAIP (aggressive style)	**6.33**	**5.63**	20.1	**0.04**
LDIP (defensive style)	6.37	5.46	**2.2**	0.002

Unseen Roundabout in Interaction Dataset

The learned policies demonstrate improved performance across various scenarios within the datasets, albeit on the same intersections as the training data. To showcase the generalization of my approach, I select the roundabout *DR_DEU_Roundabout_OF* from the Interaction Dataset and apply the same quantitative evaluation. A total of 552 vehicles are evaluated, and the results are presented in Table 5.7. The statistics exhibit a pattern similar to that of the three roundabouts in the inD dataset, where the learned policies effectively maximize the ego vehicle's utility while maintaining a reasonably low fall-back ratio.

The evaluation results on the unseen roundabout demonstrate that my approach is map-agnostic. As the MCSs operate directly on the map and consider traffic

participants as input without abstracting any information, the estimated features maintain comparable accuracy for any unseen intersection. Exceptions may arise when performing MCSs on new scenarios where traffic participants exhibit significantly different behaviors (e.g., in different countries), potentially causing substantial errors in the behavior modeling described in Section 4.2.

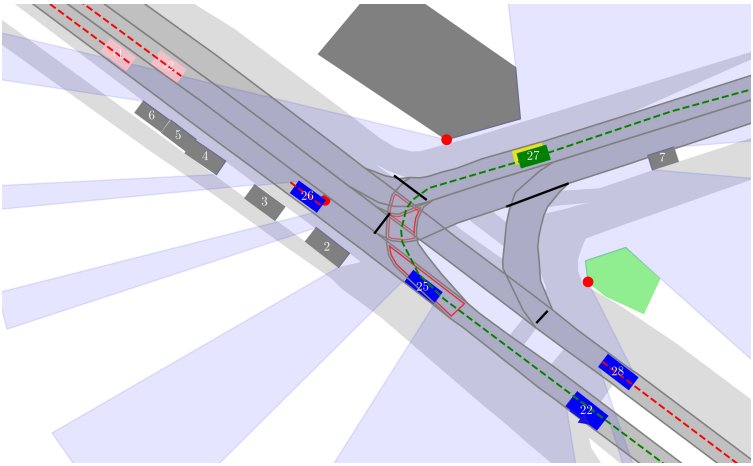
Table 5.7: Statistics for simulation evaluation for an unseen roundabout in Interaction dataset.

Policies	MDE (m)	Average velocity ($\frac{\text{m}}{\text{s}}$)	Fall-back ratio (%)	Velocity gain of the traffic ($\frac{\text{m}}{\text{s}}$)
B1 (normal stop)	6.58	6.83	**0.4**	0
B2 (fast approach)	**6.13**	**6.94**	13.4	**0.03**
LIP (learned policy)	**6.02**	**6.92**	**0.9**	0.01
LIPBP (better prediction)	**6.09**	**6.93**	**1.3**	0.02
LAIP (aggressive style)	**6.22**	**6.91**	10.3	0.01
LDIP (defensive style)	6.08	6.87	**0.2**	0.005

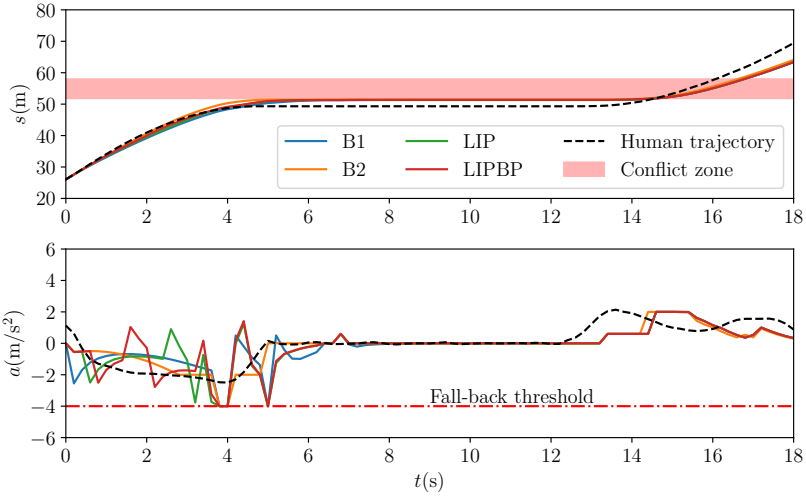
5.2.4 Case Study

I examine several representative scenarios and analyze the driving history of each driving policy, including *s*-profiles (longitudinal distance along the route) and *a*-profiles (acceleration). To avoid cluttering the charts, I do not present the profiles of LAIP and LDIP.

Figure 5.7 illustrates a complex unprotected left turn with multiple potential conflict zones, where potentially occluded and some visible vehicles have priority. The approaching accelerations of the policies differ, but all policies successfully navigate the intersection with similar *s*-profiles after the intersection is clear.



(a) Unprotected left turn.



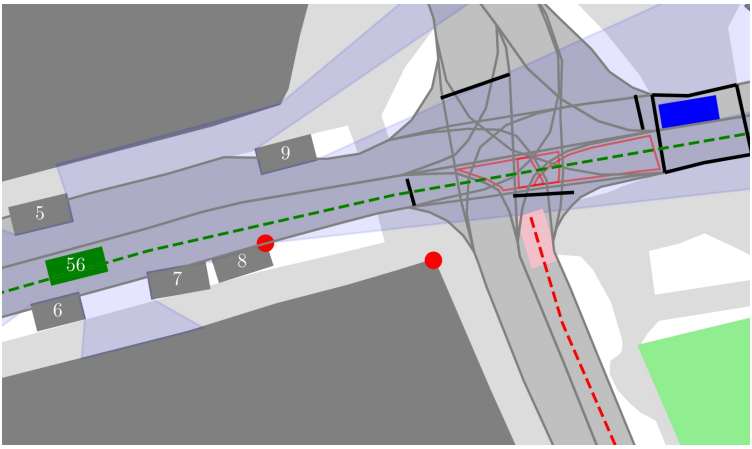
(b) s-profiles and a-profiles of different policies

Figure 5.7: The ego vehicle (green) tries to finish an unprotected left turn. The yellow polygon in the background visualizes the ground-truth position of the ego vehicle (from [Wan23b], ©2023 IEEE).

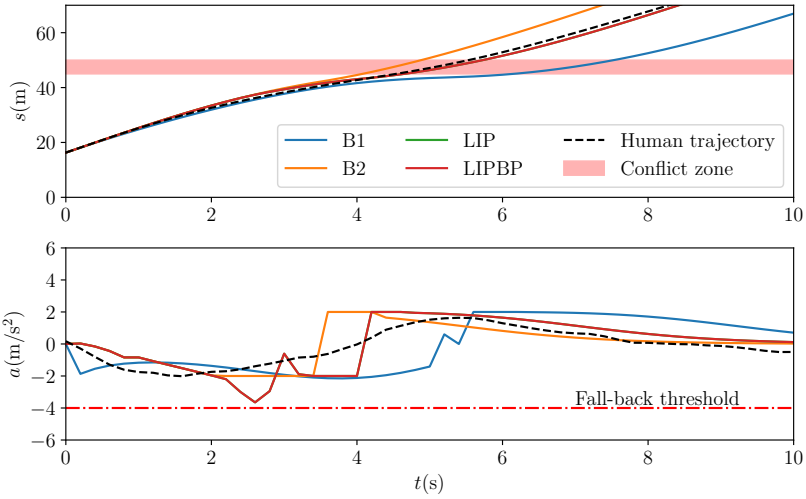
Figure 5.8 presents a scenario where the intersection is not severely occluded, but cautious driving is still required. Approaching relatively fast (with B2, LIP, and LIPBP) enables a smooth transition between deceleration and *pass*. With B1, the ego vehicle's velocity is significantly reduced. B2 attains a higher speed than even the human trajectory, which is efficient but may cause passengers to feel endangered. LIP and LIPBP generate the most human-like trajectories.

Figure 5.9 depicts a scenario at the same intersection, but with the ego vehicle approaching from the north arm. The prioritized lane (west arm) is further occluded by parked cars. B2 once again achieves the highest utility but must execute a fallback since stopping in front of the conflict zones is no longer guaranteed and *pass* is not safe either. B1 decelerates more but manages to pass the conflict zones without a fallback. LIP and LIPBP exhibit the same behavior, decelerating more than B1 initially but accelerating earlier than B1. As a result, B1, LIP, and LIPBP achieve nearly the same utility, slightly lower than the human, but all without a fallback. The human driver does not slow down excessively at the outset but intrudes into the conflict zones more aggressively than the *squeeze* action.

Figure 5.10 demonstrates the behaviors at consecutive stop lines, where the first is a pedestrian crossing and the second requires yielding to prioritized cyclists. After the ego vehicle detects the pedestrian, LIP and LIPBP execute an *early stop* action and decelerate more than B1 and B2 to demonstrate cooperative intention towards the pedestrian. The rationale for executing *early stop* is that, in MCSs, early deceleration encourages the pedestrian to choose the *cross* decision based on the pedestrian behavior models in Section 4.2.4. Consequently, the pedestrian can cross the zebra more quickly, clearing the conflict zone earlier and enabling the ego vehicle to pass sooner as well. This approach is expected to increase both the ego vehicle's and pedestrian's utilities. Note that B2 leads to a fallback again, as the conflict zone is not cleared soon enough. Subsequently, all four policies traverse the second stop line and its conflict zones similarly to the human trajectory.

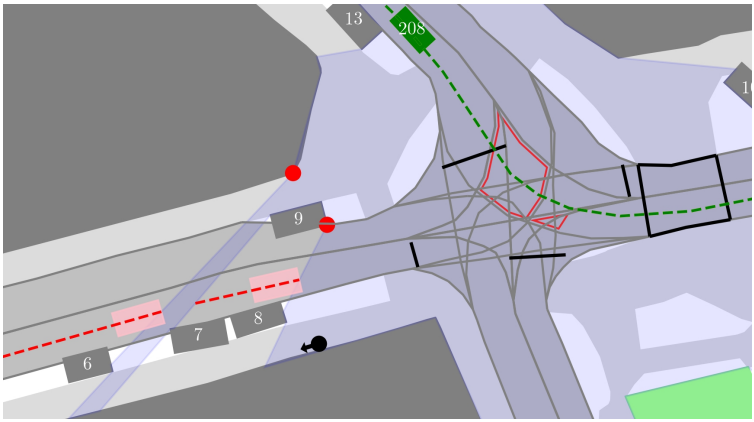


(a) Intersection with ordinary occlusion.

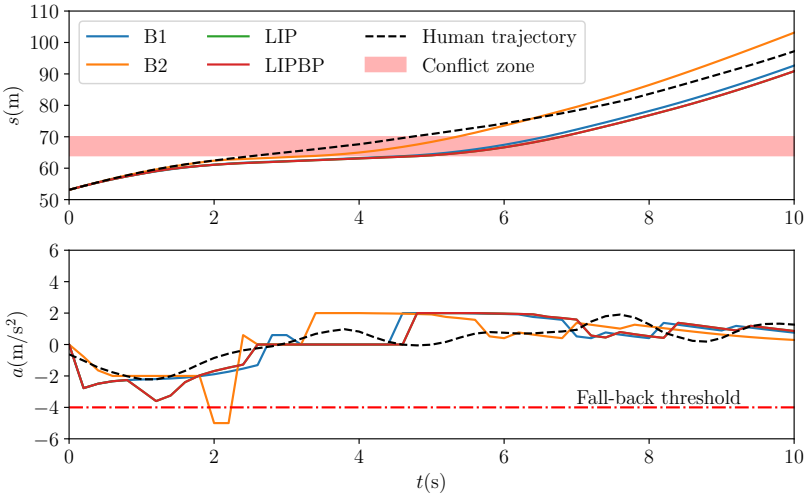


(b) s-profiles and a-profiles of different policies

Figure 5.8: The ego vehicle (green) is approaching an intersection with ordinary occlusion. The profiles of LIP and LIPBP overlap (from [Wan23b], ©2023 IEEE).

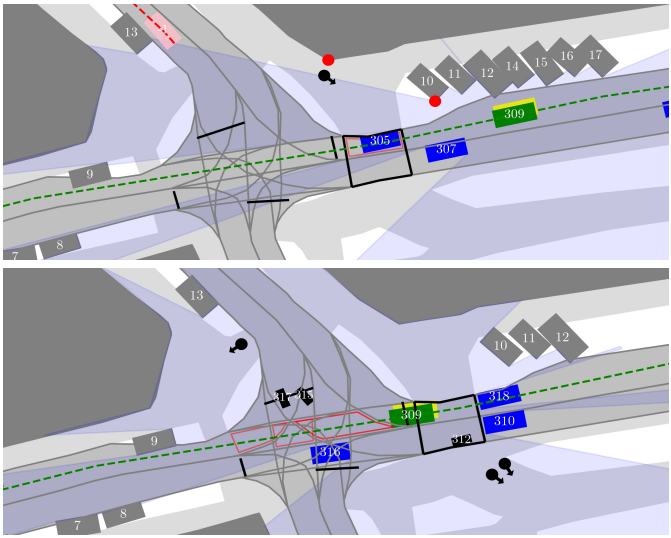


(a) Intersection with severe occlusion.

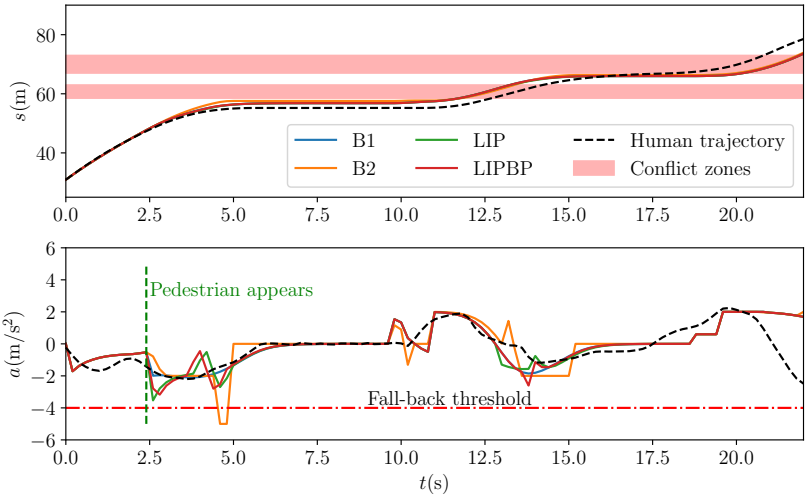


(b) s-profiles and a-profiles of different policies

Figure 5.9: The ego vehicle (green) is approaching an intersection with severe occlusion. The profiles of LIP and LIPBP overlap (from [Wan23b], ©2023 IEEE).

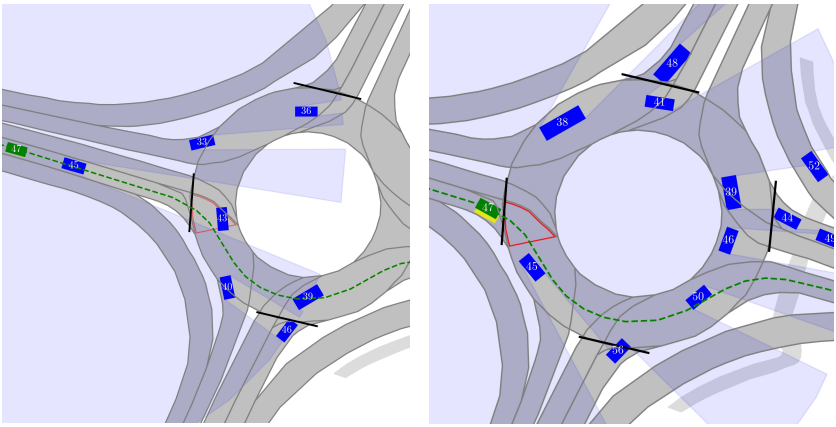


(a) Intersection with consecutive stop lines.

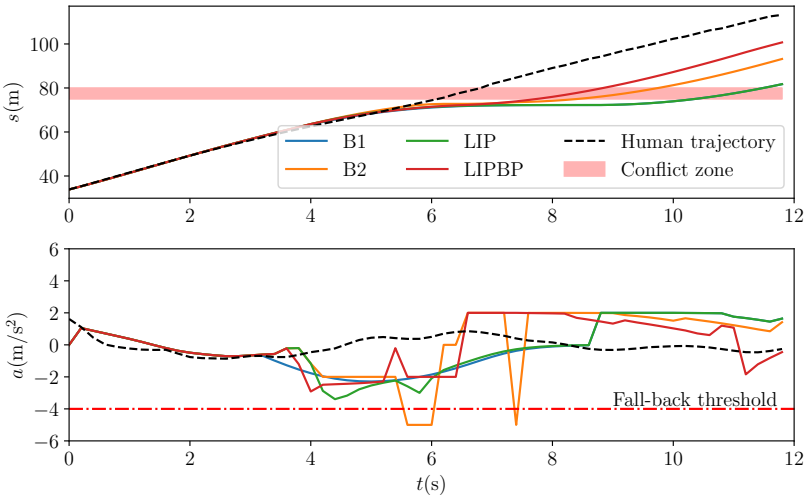


(b) s-profiles and a-profiles of different policies

Figure 5.10: The ego vehicle (green) is approaching consecutive stop lines. The first one is a pedestrian crossing, and the second one is a yielding (from [Wan23b], ©2023 IEEE).



(a) Roundabout.



(b) s-profiles and a-profiles of different policies

Figure 5.11: The ego vehicle (green) is approaching a roundabout. The vehicle with id 38 is recorded to exit the roundabout via the west exit (from [Wan23b], ©2023 IEEE).

Figure 5.11 displays a roundabout scenario. The human driver approaches the roundabout with only slight deceleration and enters with minimal hesitation. This is likely because the intention of the vehicle with id 38 is well-estimated by humans, for example, through indicator signals. The basic policy B1 and the learned policy LIP exhibit similar behavior, as they are uncertain about the exiting intention of id 38. Consequently, they must come to a full stop to ensure safety. B2 attempts to approach quickly, but safety is not guaranteed until the exiting behavior of id 38 is confirmed, leading to a fallback. However, with an improved prediction module and better intention estimation, LIPBP adjusts its velocity such that entering the roundabout becomes safe earlier and without a fallback.

In summary, B1 demonstrates the most conservative driving behavior, but it is the least risky one. B2, being the opposite of B1, maximizes the utility of the ego vehicle but at the cost of the most frequent fallbacks. The learned policy LIP strikes a good balance between utility, risk, and overall traffic flow. It achieves similar human-likeness, utility, and traffic flow as B2, but with significantly fewer fallbacks. By incorporating a better prediction module (LIPBP), the fallback ratio is further reduced without affecting other metrics.

5.3 On-vehicle Implementation and Testing

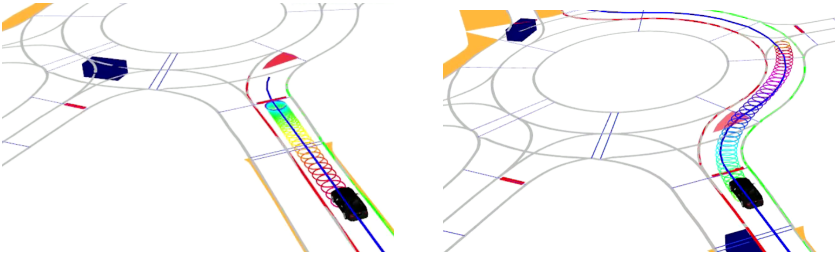
To demonstrate the applicability of my proposed approach, the core components are implemented in C++ using the middleware Robot Operating System (ROS) [Qui09] for communication with other modules and integrated into an automated driving pipeline. Owing to time constraints, only the baseline policy B1 and the learned policy LIP are ultimately developed, which can handle driving at unsignalized intersections and roundabouts, along with safety verification through RSS and MCS employing the lanelet2 library [Pog18].

The modularity of my approach is evidenced by its successful integration into the arbitration-graph-based decision-making pipeline [Orz21]. The necessary inputs are detailed in Section 3.1. The output high-level decision is converted into a customized corridor message, comprising left and right boundaries, a

reference centerline, and a suggested speed profile along the path. Utilizing this interface, an optimization-based trajectory planner and a subsequent controller from [Zie14] are employed to generate the desired trajectory and control commands.

5.3.1 Simulation Testing

The development and preliminary evaluation of behavior planning and control can be effectively carried out using CoInCar-Sim [Nau18]. This simulation relies on input from a simulated perception and localization system. For control purposes, a realistic vehicle model is implemented, incorporating characteristics from an actual car. The simulation is also based on ROS, allowing for seamless transfer of tested software from CoinCar-Sim to a real vehicle.



(a) A stopping trajectory is planned for the prioritized vehicle in roundabout.

(b) An accelerating trajectory is planned when the first vehicle has left the roundabout and the second one is far away to the conflict zone.

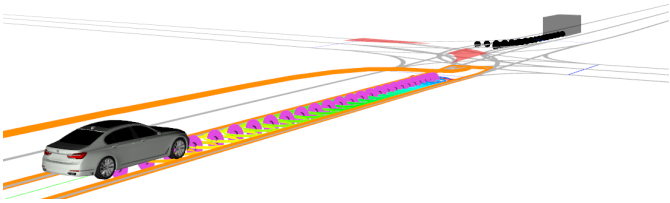
Figure 5.12: Two screeshots of the yielding scenario at a roundabout in the CoInCar-Sim. The planned trajectory is visualized with the colored circles at equidistant times. The left, right boundaries and the reference line of the driving corridor are illustrated by the red, green and blue lines. The red polygon represents the potential conflict zone between the route of the ego vehicle and prioritized vehicles.

Integration of Baseline Behavior

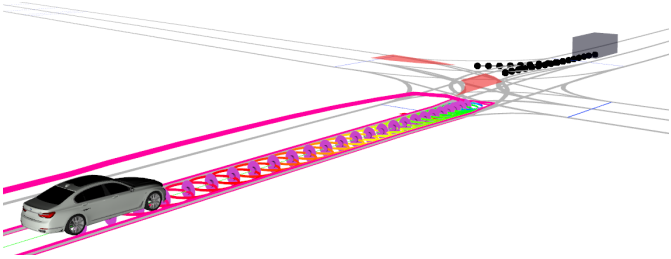
Firstly, some basis functionalities, e.g. conflict zone extraction and RSS safety assessment, are integrated into the behavior pipeline. Subsequently, the baseline behavior B1 is implemented within the arbitration graph.

Figure 5.12 demonstrates a yielding scenario at a roundabout, serving as an example for testing my implementation. As the ego vehicle approaches the roundabout (Figure 5.12a), it perceives a prioritized vehicle. It is not possible to *pass* the conflict zone without violating RSS safety, resulting in a *stopping* decision, which is then translated into a stopping trajectory. Once the first prioritized vehicle exits the roundabout and the second one remains far from the conflict zone, *passing* the conflict zone is deemed safe according to RSS verification. Consequently, an accelerating trajectory is generated to represent the *pass* decision, as depicted in Figure 5.12b.

Integration of Learned Yielding Behavior



(a) Stop before the conflict zone by B1 with *stop* action.



(b) Stop before the conflict zone by LIP with *fast approach* action.

Figure 5.13: Different actions allow for different approaching styles and speed profiles. The planned speed profiles along the future path are illustrated by the colored curves above the ego vehicle. The *fast approach* action decelerates later than the *stop* action.

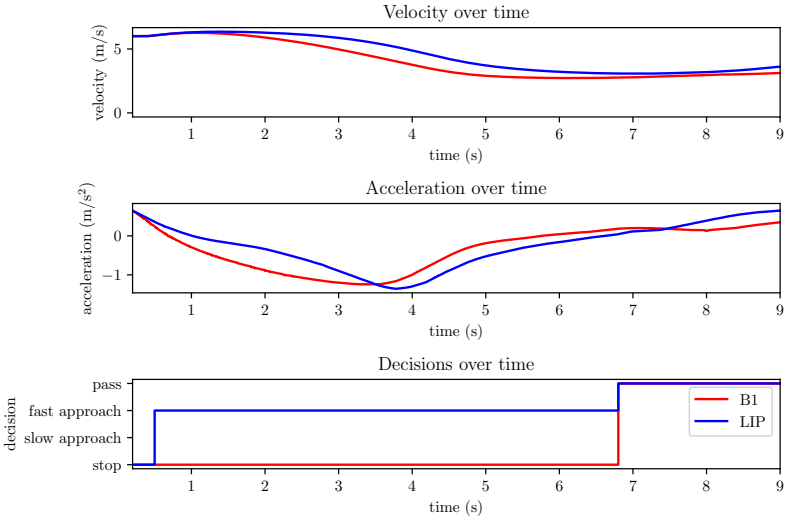


Figure 5.14: Comparison of LIP and B1.

After extensively testing the baseline behavior B1 across various scenarios and intersections, the MCS that demands a multi-threading programming and the learned yielding policy LIP are prepared for integration into the behavior pipeline. I first test the LIP in comparison to the B1 at an intersection, where the ego vehicle (white one) plans to turn left and the object (gray box) is set to turn right, resulting into two possible conflict zones as the route of the object is initially unknown to the ego vehicle. Figure 5.13 presents the scene. Since the object holds precedence over the ego vehicle, the latter should halt and yield passage to the former. Nevertheless, the velocity and proximity to the conflict zones may allow for a more comfortable deceleration method rather than abruptly stopping at the stop line.

As depicted in Figure 5.14, the LIP possesses the capability to opt for the *fast approach* action, which presents less deceleration at the beginning than the *stop*

action and a similar maximum deceleration overall. This allows for faster and smoother transition to *pass* action with less velocity reduction.

5.3.2 On-road Testing

The baseline yielding policy B1, along with RSS safety verification, was successfully showcased in two different events. At the time of the events, the learned yielding behavior was not ready and therefore, not demonstrated.

Demonstration On Experimental Vehicle

The first event is the demonstration day of Intelligent Vehicle Symposium 2022 in Aachen, on the Aldenhoven Testing Center track using our experimental vehicle *Joy*, as depicted in Figure 5.16. The baseline policy B1 is validated by interacting with other human-driven vehicles in the corresponding scenarios.

Figure 5.15: Our experimental vehicle *Joy*. *Photo by the Institute of Measurement and Control Systems (MRT), Karlsruher Institute of Technology (KIT).*

The track encompasses three types of unsignalized intersections, including a roundabout (the same one as in Figure 5.12), an unprotected left turn, and a merge into a prioritized lane.

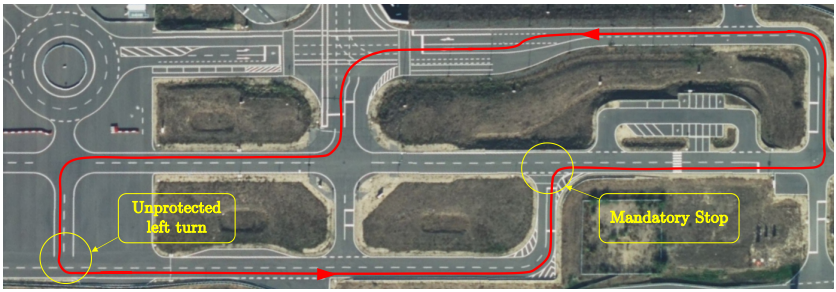


Figure 5.16: Demonstration track (red) in Aldehoven testing center. Imagery © 2023, map data © 2023 GeoBasis-DE/BKG (©2009)

Demonstration On Concept Vehicles of UNICARagil

On the final event of the *UNICARagil* project, the autonomous operation of four conceptual vehicles, namely *AutoSHUTTLE*, *AutoTAXI*, *AutoELF*, and *AutoCARGO*, is demonstrated on the Aldenhoven Testing Center track. Each vehicle navigates through distinct tracks, dynamically interacting with various vehicles driven by humans (except for *AutoCARGO*, which operates only in standstill in the final event). The yielding policy B1, in conjunction with an enhanced mandatory stopping policy[1], has been implemented across these vehicles. The visual representation of the vehicles can be found in Figure 5.17, with the specific tracks they followed and the scenarios encountered depicted in Figure 5.18 and Figure 5.19.

[1] The ego vehicle only commences the assessment of RSS safety and traversal through conflict zones subsequent to coming to a complete halt.

Figure 5.17: Concept vehicles of the project *UNICARagil*. From left to right are *AutoSHUTTLE*, *AutoTAXI*, *AutoELF*, and *AutoCARGO*. *Photo by the author.*

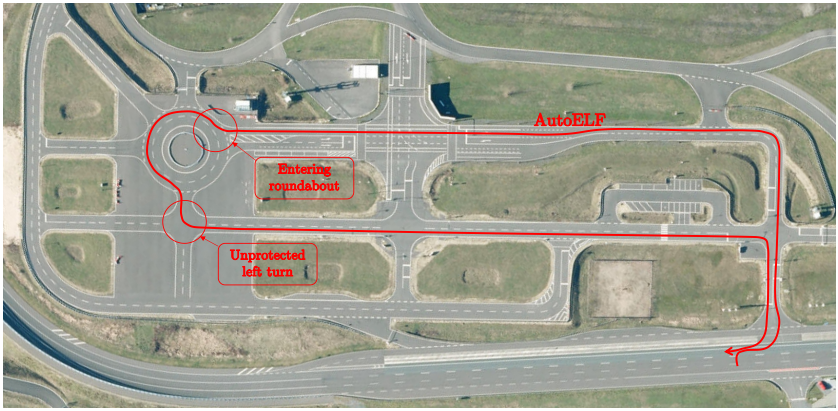


Figure 5.18: Track for *AutoELF* (red) in Aldehoven testing center. Imagery ©2023, map data ©2023 GeoBasis-DE/BKG (©2009)

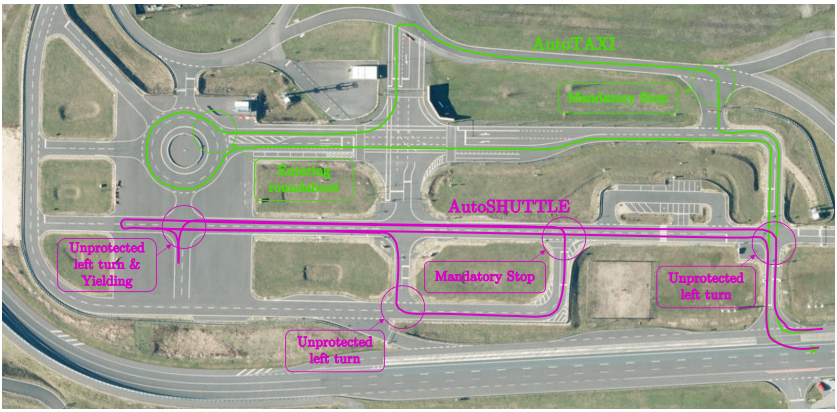


Figure 5.19: Tracks for *AutoTAXI* (green) and *AutoSHUTTLE* (pink) in Aldehoven testing center. Imagery ©2023, map data ©2023 GeoBasis-DE/BKG (©2009)

Although the learned policy LIP and its accompanying MCS were not available for the demonstration, subsequent efforts have resulted in their successful implementation. These components have undergone rigorous testing within the CoInCar-Sim framework and are presently poised for utilization on the *Joy* platform. Specifically, their application is prepared for deployment on a local track situated in the city of Karlsruhe.

6 Conclusions and Future Directions

6.1 Conclusions

This work presents a novel approach for generating high-level decisions for AVs that can effectively handle diverse uncertainty sources stemming from imperfect perception and scene understanding modules. While adhering to formal safety verification based on RSS, the approach manages to mimic behavior patterns observed in human-driven trajectories, striking a balance between efficiency, comfort, perceived safety, and politeness in a human-like manner. The output decision remains at the semantic level, ensuring independence from subsequent trajectory planning and control modules. Notably, the pipeline does not require any black-box systems, allowing for explainable and traceable output decisions in case of observed undesired behavior.

Safety verification is paramount in AD. This work adheres to the principles of RSS and introduces additional extensions grounded on traffic rules and right-of-way stipulations for various scenarios, as outlined in Section 3.2. Examples include parallel lanes and intersections. These augmentations address diverse conflict zones, traffic participants, and occlusions, while also accounting for critical limiting factors such as the ego vehicle's restricted reachability.

Owing to the stringent constraints imposed by safety requirements, decision-making efficiency is frequently constrained. To avoid excessive conservatism and adopt a more human-like approach to safety, the enhanced RSS concept is further relaxed in the presence of improved perception and scene understanding capabilities, as detailed in Section 3.3. This includes occlusion tracking, for example. To validate this relaxed safety concept, statistical analysis is carried

out on real-world traffic data across a range of scenarios. Appropriate representations of high-level actions are proposed under safety constraints in Section 3.4, with several rule-based policies serving as a baseline built upon safe high-level actions.

The objective of this work is to identify the decision that humans would choose in the same situation from all available action candidates. To this end, I propose characterizing each action using various features that encompass different aspects, as described in Section 4.1, such as utility, comfort, risk, and politeness. Human-like behavior is achieved implicitly by optimizing the weighting of these features such that the Q-value of the action closest to the implicitly inferred ground-truth action is maximized. To estimate the feature values of each action, considering diverse environmental uncertainties, MCS is employed to simulate potential future scene progressions, with behavior models and prior predictions of surrounding traffic participants iteratively queried. A probabilistic environmental model is proposed in Section 4.2. In addition to emulating a universal human driving style in Section 4.3, I identify distinct human driving styles, cluster the datasets, and learn various policies from each dataset group in Section 4.4. This enables the preparation of predefined driving styles and even online behavior tuning to cater to users' preferences.

Evaluations were conducted in customized simulation environments, one featuring randomly generated traffic and the other using recorded trajectories with reactive agents. Simulation results demonstrate the applicability of my approach in typical urban and highway driving scenarios, as well as its superiority over rule-based policies, exhibiting enhanced performance in human-likeness, safety, efficiency, and politeness. Additionally, on-vehicle implementation and driving experiments showcase the potential and scalability of my approach in real-world driving situations.

My approach boasts several notable features. One is its highly modular design, accepting uncertain perception results in any form as input. The output consists of high-level decisions, which can be converted into low-level control commands by any trajectory planning module. Another advantage is the ease of tracing back the resulting decision, facilitated by examining the MCS or the

Q-value from the linear function. Moreover, this approach is applicable to any scenario, provided that RSS safety and high-level actions are defined a prior.

6.2 Future Directions

The proposed approach serves as a foundation for several promising future research directions. First, its application can be extended to additional scenarios, such as overtaking, navigating narrow roads with oncoming traffic, or situations with specific traffic rules like zipper merging.

Moreover, moving beyond the basic behavior modeling and routing prediction for surrounding agents employed in this work, advanced strategies can further enhance the performance of the approach. Scene understanding and prediction using machine learning techniques with realistic traffic data will yield more reliable environmental modeling, enabling improved MCS for determining accurate feature values.

In terms of perceived environmental input, it is important to consider not only the uncertain states of traffic participants but also their classes and existence probabilities. Class distinctions affect behavior modeling, while uncertain existence alters the risk level associated with each action. These uncertainties also impact RSS safety, which currently assumes a deterministic world with certain agent states when generating safety requirements. A more practical approach for real-world applications would treat agent states as probabilistic or truncated distributions, incorporating state bounds when constructing worst-case setups for RSS. Recent research has already been started in this field [Ber22].

Finally, some real-life scenarios necessitate explicit cooperative signals to resolve deadlock situations, such as when AVs from all four arms of a right-before-left intersection arrive simultaneously. Without one vehicle relinquishing priority and conveying cooperation through machine-readable signals (e.g., flashing headlights as human drivers do), no vehicle can proceed. Other AVs should be capable of interpreting these signals and properly accounting for the cooperative agent in their RSS safety assessments, such as ignoring it when examining right-of-way safety.

Bibliography

[Alt14] Althoff, M. and Dolan, J. M.: “Online Verification of Automated Road Vehicles Using Reachability Analysis”. In: *IEEE Transactions on Robotics* 30.4 (2014), pp. 903–918. doi: 10.1109/TRO.2014.2312453 (cit. on pp. 27, 33).

[Alt16] Althoff, M. and Magdici, S.: “Set-Based Prediction of Traffic Participants on Arbitrary Road Networks”. In: *IEEE Transactions on Intelligent Vehicles* 1.2 (2016), pp. 187–202. doi: 10.1109/TIV.2016.2622920 (cit. on p. 27).

[Alt17] Althoff, M.; Koschi, M. and Manzinger, S.: “CommonRoad: Composable benchmarks for motion planning on roads”. In: *2017 IEEE Intelligent Vehicles Symposium (IV)*. 2017, pp. 719–726. doi: 10.1109/IVS.2017.7995802 (cit. on p. 113).

[Alt21] Altendorfer, R. and Wilkmann, C.: “A New Approach to Estimate the Collision Probability for Automotive Applications”. In: *Automatica* 127.C (May 2021). doi: 10.1016/j.automatica.2021.109497. url: https://doi.org/10.1016/j.automatica.2021.109497 (cit. on p. 26).

[Ber20] Bernhard, J.; Esterle, K.; Hart, P. and Kessler, T.: “BARK: Open Behavior Benchmarking in Multi-Agent Environments”. In: (Oct. 2020), pp. 6201–6208. doi: 10.1109/IROS45743.2020.9341222. url: https://ieeexplore.ieee.org/document/9341222/ (visited on 04/26/2022) (cit. on p. 113).

[Ber22] Bernhard, J.; Hart, P.; Sahu, A.; Schöller, C. and Cancimance, M. G.: “Risk-Based Safety Envelopes for Autonomous Vehicles

Under Perception Uncertainty". In: *2022 IEEE Intelligent Vehicles Symposium (IV)*. 2022, pp. 104–111. doi: 10.1109/IV51971.2022.9827199 (cit. on pp. 25, 27, 137).

[Bha23] Bhattacharyya, R.; Wulfe, B.; Phillips, D. J.; Kuefler, A.; Morton, J.; Senanayake, R. and Kochenderfer, M. J.: "Modeling Human Driving Behavior Through Generative Adversarial Imitation Learning". In: *IEEE Transactions on Intelligent Transportation Systems* 24.3 (2023), pp. 2874–2887. doi: 10.1109/TITS.2022.3227738 (cit. on p. 23).

[Boc20] Bock, J.; Krajewski, R.; Moers, T.; Runde, S.; Vater, L. and Eckstein, L.: "The inD Dataset: A Drone Dataset of Naturalistic Road User Trajectories at German Intersections". In: *2020 IEEE Intelligent Vehicles Symposium (IV)*. 2020, pp. 1929–1934. doi: 10.1109/IV47402.2020.9304839 (cit. on pp. 41, 92).

[Bou19] Bouton, M.; Nakhaei, A.; Fujimura, K. and Kochenderfer, M. J.: "Safe Reinforcement Learning with Scene Decomposition for Navigating Complex Urban Environments". In: *2019 IEEE Intelligent Vehicles Symposium (IV)*. 2019, pp. 1469–1476. doi: 10.1109/IVS.2019.8813803 (cit. on p. 20).

[Bou20] Bouton, M.; Nakhaei, A.; Isele, D.; Fujimura, K. and Kochenderfer, M. J.: "Reinforcement Learning with Iterative Reasoning for Merging in Dense Traffic". In: *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*. 2020, pp. 1–6. doi: 10.1109/ITSC45102.2020.9294338 (cit. on p. 20).

[Bur18] Burger, C. and Lauer, M.: "Cooperative Multiple Vehicle Trajectory Planning using MIQP". In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. 2018, pp. 602–607. doi: 10.1109/ITSC.2018.8569776 (cit. on p. 75).

[Cod19] Codevilla, F.; Santana, E.; Lopez, A. and Gaidon, A.: "Exploring the Limitations of Behavior Cloning for Autonomous Driving". In: *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019, pp. 9328–9337. doi: 10.1109/ICCV.2019.00942 (cit. on p. 22).

[Dam15] Damerow, F. and Eggert, J.: “Balancing risk against utility: Behavior planning using predictive risk maps”. In: *2015 IEEE Intelligent Vehicles Symposium (IV)*. 2015, pp. 857–864. doi: 10.1109/IVS.2015.7225792 (cit. on p. 25).

[Dos17] Dosovitskiy, A.; Ros, G.; Codevilla, F.; Lopez, A. and Koltun, V.: “CARLA: An Open Urban Driving Simulator”. In: *Proceedings of the 1st Annual Conference on Robot Learning*. 2017, pp. 1–16 (cit. on p. 113).

[Elb15] Elbanhawi, M.; Simic, M. and Jazar, R.: “In the Passenger Seat: Investigating Ride Comfort Measures in Autonomous Cars”. In: *IEEE Intelligent Transportation Systems Magazine* 7.3 (2015), pp. 4–17. doi: 10.1109/MITS.2015.2405571 (cit. on p. 75).

[Fan22] Fang, X.; Zhang, Q.; Gao, Y. and Zhao, D.: “Offline Reinforcement Learning for Autonomous Driving with Real World Driving Data”. In: *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*. 2022, pp. 3417–3422. doi: 10.1109/ITSC55140.2022.9922100 (cit. on p. 20).

[Fis20] Fischer, J. and Tas, Ö. S.: “Information Particle Filter Tree: An Online Algorithm for POMDPs with Belief-Based Rewards on Continuous Domains”. In: *Proceedings of the 37th International Conference on Machine Learning*. Ed. by III, H. D. and Singh, A. Vol. 119. Proceedings of Machine Learning Research. PMLR, 13–18 Jul 2020, pp. 3177–3187. url: https://proceedings.mlr.press/v119/fischer20a.html (cit. on p. 17).

[Fis21] Fischer, J.; Eyberg, C.; Werling, M. and Lauer, M.: “Sampling-based Inverse Reinforcement Learning Algorithms with Safety Constraints”. In: *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2021, pp. 791–798. doi: 10.1109/IROS51168.2021.9636672 (cit. on p. 23).

[Fis22] Fischer, J.; Bührle, E.; Kamran, D. and Stiller, C.: “Guiding Belief Space Planning with Learned Models for Interactive Merging”. In: *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*. 2022, pp. 2542–2549. doi: 10.1109/ITSC55140.2022.9922488 (cit. on pp. 18, 63).

[Fre20] Frey, K. M.; Steiner, T. J. and How, J. P.: “Collision Probabilities for Continuous-Time Systems Without Sampling”. In: *Robotics: Science and Systems XVI* XVI (June 2020), p. 1. doi: 10.15607/RSS.2020.XVI.019. url: https://hdl.handle.net/1721.1/137189.2 (cit. on p. 26).

[Has23] Hasuo, I.; Eberhart, C.; Haydon, J.; Dubut, J.; Bohrer, R.; Kobayashi, T.; Pruekprasert, S.; Zhang, X.-Y.; Pallas, E. A.; Yamada, A. et al.: “Goal-Aware RSS for Complex Scenarios via Program Logic”. In: *IEEE Transactions on Intelligent Vehicles* 8.4 (2023), pp. 3040–3072. doi: 10.1109/TIV.2022.3169762 (cit. on p. 27).

[He23] He, X.; Yang, H.; Hu, Z. and Lv, C.: “Robust Lane Change Decision Making for Autonomous Vehicles: An Observation Adversarial Reinforcement Learning Approach”. In: *IEEE Transactions on Intelligent Vehicles* 8.1 (2023), pp. 184–193. doi: 10.1109/TIV.2022.3165178 (cit. on p. 19).

[Ho16] Ho, J. and Ermon, S.: “Generative Adversarial Imitation Learning”. In: *Proceedings of the 30th International Conference on Neural Information Processing Systems*. NIPS’16. Barcelona, Spain: Curran Associates Inc., 2016, pp. 4572–4580 (cit. on p. 15).

[Hob77] Hoberock, L. L.: “A Survey of Longitudinal Acceleration Comfort Studies in Ground Transportation Vehicles”. In: *Journal of Dynamic Systems, Measurement, and Control* 99.2 (June 1977), pp. 76–84. doi: 10.1115/1.3427093. eprint: https://asmedigitalcollection.asme.org/dynamicsystems/article-pdf/99/2/76/5778098/76_1.pdf. url: https://doi.org/10.1115/1.3427093 (cit. on p. 37).

[Hoe20] Hoel, C.-J.; Driggs-Campbell, K.; Wolff, K.; Laine, L. and Kochenderfer, M. J.: "Combining Planning and Deep Reinforcement Learning in Tactical Decision Making for Autonomous Driving". In: *IEEE Transactions on Intelligent Vehicles* 5.2 (2020), pp. 294–305. doi: 10.1109/TIV.2019.2955905 (cit. on p. 20).

[Hu23] Hu, W.; Li, X.; Hu, J.; Kong, D.; Hu, Y.; Xu, Q.; Liu, Y.; Song, X. and Dong, X.: "A Safe Driving Decision-Making Methodology Based on Cascade Imitation Learning Network for Automated Commercial Vehicles". In: *IEEE Sensors Journal* 23.11 (2023), pp. 11285–11295. doi: 10.1109/JSEN.2023.3256704 (cit. on p. 23).

[Hua20] Huang, Z.; Lv, C. and Wu, J.: "Modeling Human Driving Behavior in Highway Scenario using Inverse Reinforcement Learning". In: *CoRR* abs/2010.03118 (2020). arXiv: 2010.03118. url: https://arxiv.org/abs/2010.03118 (cit. on p. 22).

[Hub17] Hubmann, C.; Becker, M.; Althoff, D.; Lenz, D. and Stiller, C.: "Decision making for autonomous driving considering interaction and uncertain prediction of surrounding vehicles". In: *2017 IEEE Intelligent Vehicles Symposium (IV)*. 2017, pp. 1671–1678. doi: 10.1109/IVS.2017.7995949 (cit. on pp. 16, 17).

[Hub18] Hubmann, C.; Schulz, J.; Xu, G.; Althoff, D. and Stiller, C.: "A Belief State Planner for Interactive Merge Maneuvers in Congested Traffic". In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. 2018, pp. 1617–1624. doi: 10.1109/ITSC.2018.8569729 (cit. on pp. 17, 65).

[Hub19] Hubmann, C.; Quetschlich, N.; Schulz, J.; Bernhard, J.; Althoff, D. and Stiller, C.: "A POMDP Maneuver Planner For Occlusions in Urban Scenarios". In: *2019 IEEE Intelligent Vehicles Symposium (IV)*. 2019, pp. 2172–2179. doi: 10.1109/IVS.2019.8814179 (cit. on p. 17).

[Hue19] Huegle, M.; Kalweit, G.; Mirchevska, B.; Werling, M. and Boedecker, J.: “Dynamic Input for Deep Reinforcement Learning in Autonomous Driving”. In: *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2019, pp. 7566–7573. doi: 10.1109/IROS40897.2019.8968560 (cit. on p. 20).

[Ise18] Isele, D.; Rahimi, R.; Cosgun, A.; Subramanian, K. and Fujimura, K.: “Navigating Occluded Intersections with Autonomous Vehicles Using Deep Reinforcement Learning”. In: *2018 IEEE International Conference on Robotics and Automation (ICRA)*. 2018, pp. 2034–2039. doi: 10.1109/ICRA.2018.8461233 (cit. on p. 19).

[Jai14] Jain, A.; Gupta, A. and Rastogi, R.: “Pedestrian crossing behaviour analysis at intersections”. In: *International Journal for Traffic and Transport Engineering* 4.1 (2014), pp. 103–116 (cit. on pp. 42, 79).

[Jam23] Jamgochian, A.; Buehrle, E.; Fischer, J. and Kochenderfer, M. J.: “SHAIL: Safety-Aware Hierarchical Adversarial Imitation Learning for Autonomous Driving in Urban Environments”. In: *2023 IEEE International Conference on Robotics and Automation (ICRA)*. 2023, pp. 1530–1536. doi: 10.1109/ICRA48891.2023.10161449 (cit. on pp. 23, 24).

[Kam20] Kamran, D.; Lopez, C. F.; Lauer, M. and Stiller, C.: “Risk-Aware High-Level Decisions for Automated Driving at Occluded Intersections with Reinforcement Learning”. In: *2020 IEEE Intelligent Vehicles Symposium (IV)*. Las Vegas, NV, USA: IEEE Press, 2020, pp. 1205–1212. doi: 10.1109/IV47402.2020.9304606. url: https://doi.org/10.1109/IV47402.2020.9304606 (cit. on p. 20).

[Kam22] Kamran, D.; Simão, T. D.; Yang, Q.; Ponnambalam, C. T.; Fischer, J.; Spaan, M. T. and Lauer, M.: “A Modern Perspective on Safe Automated Driving for Different Traffic Dynamics Using Constrained Reinforcement Learning”. In: *2022 IEEE 25th International Conference on Intelligent Transportation Systems*

(ITSC). 2022, pp. 4017–4023. doi: 10.1109/ITSC55140.2022.9921907 (cit. on p. 20).

[Kar21] Karl Couto, G. C. and Antonelo, E. A.: "Generative Adversarial Imitation Learning for End-to-End Autonomous Driving on Urban Environments". In: *2021 IEEE Symposium Series on Computational Intelligence (SSCI)*. 2021, pp. 1–7. doi: 10.1109/SSCI50451.2021.9660156 (cit. on p. 23).

[Kes07] Kesting, A.; Treiber, M. and Helbing, D.: "General Lane-Changing Model MOBIL for Car-Following Models". In: *Transportation Research Record* 1999.1 (2007), pp. 86–94. doi: 10.3141/1999-10. eprint: https://doi.org/10.3141/1999-10. url: https://doi.org/10.3141/1999-10 (cit. on p. 66).

[Kim18] Kim, J. and Kum, D.: "Collision Risk Assessment Algorithm via Lane-Based Probabilistic Motion Prediction of Surrounding Vehicles". In: *IEEE Transactions on Intelligent Transportation Systems* 19.9 (Sept. 2018), pp. 2965–2976. doi: 10.1109/TITS.2017.2768318 (cit. on p. 26).

[Kno21] Knox, W. B.; Allievi, A.; Banzhaf, H.; Schmitt, F. and Stone, P.: "Reward (Mis)design for Autonomous Driving". In: *CoRR* abs/2104.13906 (2021). arXiv: 2104.13906. url: https://arxiv.org/abs/2104.13906 (cit. on p. 74).

[Koc22] Kochenderfer, M.; Wheeler, T. and Wray, K.: Algorithmus for decision making. The MIT Press, 2022 (cit. on p. 32).

[Kra18] Krajewski, R.; Bock, J.; Kloeker, L. and Eckstein, L.: "The highD Dataset: A Drone Dataset of Naturalistic Vehicle Trajectories on German Highways for Validation of Highly Automated Driving Systems". In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. 2018, pp. 2118–2125. doi: 10.1109/ITSC.2018.8569552 (cit. on pp. 37, 92).

[Kra20] Krajewski, R.; Moers, T.; Bock, J.; Vater, L. and Eckstein, L.: "The rounD Dataset: A Drone Dataset of Road User Trajectories at Roundabouts in Germany". In: *2020 IEEE 23rd International*

Conference on Intelligent Transportation Systems (ITSC). 2020, pp. 1–6. doi: 10.1109/ITSC45102.2020.9294728 (cit. on p. 92).

[Kre22] Kreutz, K. and Eggert, J.: "Fast online parameter estimation of the Intelligent Driver Model for trajectory prediction". In: *2022 IEEE Intelligent Vehicles Symposium (IV)*. 2022, pp. 758–765. doi: 10.1109/IV51971.2022.9827115 (cit. on p. 82).

[Kum18] Kumar, A. V. S. S. B.; Modh, A.; Babu, M.; Gopalakrishnan, B. and Krishna, K. M.: "A Novel Lane Merging Framework with Probabilistic Risk based Lane Selection using Time Scaled Collision Cone". In: *2018 IEEE Intelligent Vehicles Symposium (IV)*. 2018, pp. 1406–1411. doi: 10.1109/IVS.2018.8500652 (cit. on p. 26).

[Kur22] Kurzer, K.; Bitzer, M. and Zöllner, J. M.: "Learning Reward Models for Cooperative Trajectory Planning with Inverse Reinforcement Learning and Monte Carlo Tree Search". In: *2022 IEEE Intelligent Vehicles Symposium (IV)*. 2022, pp. 22–28. doi: 10.1109/IV51971.2022.9827031 (cit. on p. 23).

[Lef14] Lefèvre, S.; Vasquez, D. and Laugier, C.: "A survey on motion prediction and risk assessment for intelligent vehicles". In: *ROBOMECH Journal* 1.1 (2014), p. 1. doi: 10.1186/s40648-014-0001-z. url: https://doi.org/10.1186/s40648-014-0001-z (cit. on p. 25).

[Len16] Lenz, D.; Kessler, T. and Knoll, A.: "Tactical cooperative planning for autonomous highway driving using Monte-Carlo Tree Search". In: *2016 IEEE Intelligent Vehicles Symposium (IV)*. 2016, pp. 447–453. doi: 10.1109/IVS.2016.7535424 (cit. on p. 17).

[Li21] Li, D. and Du, J.: "Maximum Entropy Inverse Reinforcement Learning Based on Behavior Cloning of Expert Examples". In: *2021 IEEE 10th Data Driven Control and Learning Systems Conference (DDCLS)*. 2021, pp. 996–1000. doi: 10.1109/DDCLS52934.2021.9455476 (cit. on p. 22).

[Liu20] Liu, T.; Huang, B.; Deng, Z.; Wang, H.; Tang, X.; Wang, X. and Cao, D.: “Heuristics-oriented overtaking decision making for autonomous vehicles using reinforcement learning”. In: *IET Electrical Systems in Transportation* 10.4 (2020), pp. 417–424. doi: https://doi.org/10.1049/iet-est.2020.0044. eprint: https://ietresearch.onlinelibrary.wiley.com/doi/pdf/10.1049/iet-est.2020.0044. url: https://ietresearch.onlinelibrary.wiley.com/doi/abs/10.1049/iet-est.2020.0044 (cit. on p. 19).

[Lop18] Lopez, P. A.; Behrisch, M.; Bieker-Walz, L.; Erdmann, J.; Flötteröd, Y.-P.; Hilbrich, R.; Lücken, L.; Rummel, J.; Wagner, P. and Wiessner, E.: “Microscopic Traffic Simulation using SUMO”. In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. 2018, pp. 2575–2582. doi: 10.1109/ITSC.2018.8569938 (cit. on p. 102).

[Lub21] Lubars, J.; Gupta, H.; Chinchali, S.; Li, L.; Raja, A.; Srikant, R. and Wu, X.: “Combining Reinforcement Learning with Model Predictive Control for On-Ramp Merging”. In: *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*. 2021, pp. 942–947. doi: 10.1109/ITSC48978.2021.9564954 (cit. on p. 19).

[Mar16] Markkula, G.; Engström, J.; Lodin, J.; Bärgman, J. and Victor, T.: “A farewell to brake reaction times? Kinematics-dependent brake response in naturalistic rear-end emergencies”. In: *Accident Analysis & Prevention* 95 (2016), pp. 209–226. doi: https://doi.org/10.1016/j.aap.2016.07.007. url: https://www.sciencedirect.com/science/article/pii/S0001457516302366 (cit. on p. 37).

[McG19] McGill, S. G.; Rosman, G.; Ort, T.; Pierson, A.; Gilitschenski, I.; Araki, B.; Fletcher, L.; Karaman, S.; Rus, D. and Leonard, J. J.: “Probabilistic Risk Metrics for Navigating Occluded Intersections”. In: *IEEE Robotics and Automation Letters* 4.4 (2019), pp. 4322–4329. doi: 10.1109/LRA.2019.2931823 (cit. on p. 26).

[Mey18] Meyer, A.; Salscheider, N. O.; Orzechowski, P. F. and Stiller, C.: “Deep Semantic Lane Segmentation for Mapless Driving”. In:

2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). 2018, pp. 869–875. doi: 10.1109/IROS.2018.8594450 (cit. on p. 30).

[Mir18] Mirchevska, B.; Pek, C.; Werling, M.; Althoff, M. and Boedecker, J.: "High-level Decision Making for Safe and Reasonable Autonomous Lane Changing using Reinforcement Learning". In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. 2018, pp. 2156–2162. doi: 10.1109/ITSC.2018.8569448 (cit. on p. 19).

[Moe22] Moers, T.; Vater, L.; Krajewski, R.; Bock, J.; Zlocki, A. and Eckstein, L.: "The ExiD Dataset: A Real-World Trajectory Dataset of Highly Interactive Highway Scenarios in Germany". In: Aachen, Germany: IEEE Press, 2022, pp. 958–964. doi: 10.1109/IV51971.2022.9827305. url: https://doi.org/10.1109/IV51971.2022.9827305 (cit. on pp. 37, 66, 92).

[Müh72] Mühl, W. and Karl, R.: Kommentar. Berlin, Boston: De Gruyter, 1972. doi: doi:10.1515/9783110892352. url: https://doi.org/10.1515/9783110892352 (cit. on p. 49).

[Nau18] Naumann, M.; Poggenhans, F.; Lauer, M. and Stiller, C.: "CoInCar-Sim: An Open-Source Simulation Framework for Cooperatively Interacting Automobiles". In: *2018 IEEE Intelligent Vehicles Symposium (IV)*. 2018, pp. 1–6. doi: 10.1109/IVS.2018.8500405 (cit. on p. 127).

[Nau19] Naumann, M.; Königshof, H. and Stiller, C.: "Provably Safe and Smooth Lane Changes in Mixed Traffic". In: *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. 2019, pp. 1832–1837. doi: 10.1109/ITSC.2019.8917461 (cit. on pp. 67, 83).

[Nau20a] Naumann, M.: "Probabilistic Motion Planning for Automated Vehicles". PhD thesis. Karlsruher Institut für Technologie (KIT), 2020. 161 pp. doi: 10.5445/KSP/1000126389 (cit. on pp. 4, 33, 35, 36, 39, 40, 46).

[Nau20b] Naumann, M.; Sun, L.; Zhan, W. and Tomizuka, M.: "Analyzing the Suitability of Cost Functions for Explaining and Imitating Human Driving Behavior based on Inverse Reinforcement Learning". In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. 2020, pp. 5481–5487. doi: 10.1109/ICRA40945.2020.9196795 (cit. on pp. 22, 73).

[Nau21] Naumann, M.; Wirth, F.; Oboril, F.; Scholl, K.; Elli, M. S.; Alvarez, I.; Weast, J. and Stiller, C.: "On Responsibility Sensitive Safety in Car-following Situations - A Parameter Analysis on German Highways". In: *2021 IEEE Intelligent Vehicles Symposium (IV)*. 2021, pp. 83–90. doi: 10.1109/IV48863.2021.9575420 (cit. on pp. 27, 38, 48).

[Orz18] Orzechowski, P. F.; Meyer, A. and Lauer, M.: "Tackling Occlusions & Limited Sensor Range with Set-based Safety Verification". In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. 2018, pp. 1729–1736. doi: 10.1109/ITSC.2018.8569332 (cit. on pp. 27, 40, 47).

[Orz19] Orzechowski, P. F.; Li, K. and Lauer, M.: "Towards Responsibility-Sensitive Safety of Automated Vehicles with Reachable Set Analysis". In: *2019 IEEE International Conference on Connected Vehicles and Expo (ICCVE)*. 2019, pp. 1–6. doi: 10.1109/ICCVE45908.2019.8965069 (cit. on p. 27).

[Orz21] Orzechowski, P. F.; Burger, C.; Lauer, M. and Stiller, C. In: *at - Automatisierungstechnik* 69.2 (2021), pp. 171–181. doi: doi:10.1515/auto-2020-0099. url: https://doi.org/10.1515/auto-2020-0099 (cit. on p. 126).

[Pet13] Petrich, D.; Dang, T.; Kasper, D.; Breuel, G. and Stiller, C.: "Map-based long term motion prediction for vehicles in traffic environments". In: *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*. 2013, pp. 2166–2172. doi: 10.1109/ITSC.2013.6728549 (cit. on p. 86).

[Pha23] Phan-Minh, T.; Howington, F.; Chu, T.-S.; Tomov, M. S.; Beaudoin, R. E.; Lee, S. U.; Li, N.; Dicle, C.; Findler, S.; Suarez-Ruiz, F. et al.: "DriveIRL: Drive in Real Life with Inverse Reinforcement Learning". In: *2023 IEEE International Conference on Robotics and Automation (ICRA)*. 2023, pp. 1544–1550. doi: 10.1109/ICRA48891.2023.10160449 (cit. on p. 23).

[Phi19] Philipp, A. and Goehring, D.: "Analytic Collision Risk Calculation for Autonomous Vehicle Navigation". In: *2019 International Conference on Robotics and Automation (ICRA)*. 2019, pp. 1744–1750. doi: 10.1109/ICRA.2019.8793264 (cit. on p. 26).

[Pie18] Pierson, A.; Schwarting, W.; Karaman, S. and Rus, D.: "Navigating Congested Environments with Risk Level Sets". In: *2018 IEEE International Conference on Robotics and Automation (ICRA)*. 2018, pp. 5712–5719. doi: 10.1109/ICRA.2018.8460697 (cit. on pp. 25, 26).

[Pie19] Pierson, A.; Schwarting, W.; Karaman, S. and Rus, D.: "Learning Risk Level Set Parameters from Data Sets for Safer Driving". In: *2019 IEEE Intelligent Vehicles Symposium (IV)*. 2019, pp. 273–280. doi: 10.1109/IVS.2019.8813842 (cit. on pp. 25, 26).

[Pog18] Poggenhans, F.; Pauls, J.-H.; Janosovits, J.; Orf, S.; Naumann, M.; Kuhnt, F. and Mayr, M.: "Lanelet2: A high-definition map framework for the future of automated driving". In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. 2018, pp. 1672–1679. doi: 10.1109/ITSC.2018.8569929 (cit. on pp. 30, 43, 126).

[Que18] Quehl, J.; Hu, H.; Wirges, S. and Lauer, M.: "An Approach to Vehicle Trajectory Prediction Using Automatically Generated Traffic Maps". In: *2018 IEEE Intelligent Vehicles Symposium (IV)*. 2018, pp. 544–549. doi: 10.1109/IVS.2018.8500535 (cit. on p. 86).

[Qui09] Quigley, M.; Conley, K.; Gerkey, B. P.; Faust, J.; Foote, T.; Leibs, J.; Wheeler, R. and Ng, A. Y.: "ROS: an open-source

Robot Operating System". In: *ICRA Workshop on Open Source Software*. 2009 (cit. on p. 126).

[Rey01] Reymond, G.; Kemeny, A.; Droulez, J. and Berthoz, A.: "Role of Lateral Acceleration in Curve Driving: Driver Model and Experiments on a Real Vehicle and a Driving Simulator". In: *Human Factors* 43.3 (2001). PMID: 11866202, pp. 483–495. doi: 10.1518/001872001775898188. eprint: https://doi.org/10.1518/001872001775898188. url: https://doi.org/10.1518/001872001775898188 (cit. on p. 57).

[Ros11] Ross, S.; Gordon, G. and Bagnell, D.: "A Reduction of Imitation Learning and Structured Prediction to No-Regret Online Learning". In: *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. Ed. by Gordon, G.; Dunson, D. and Dudík, M. Vol. 15. Proceedings of Machine Learning Research. Fort Lauderdale, FL, USA: PMLR, Nov. 2011, pp. 627–635. url: https://proceedings.mlr.press/v15/ross11a.html (cit. on p. 14).

[Sac22] Sackmann, M.; Bey, H.; Hofmann, U. and Thielecke, J.: "Modeling Driver Behavior using Adversarial Inverse Reinforcement Learning". In: *2022 IEEE Intelligent Vehicles Symposium (IV)*. 2022, pp. 1683–1690. doi: 10.1109/IV51971.2022.9827292 (cit. on p. 23).

[Sak19] Saksena, S. K.; B., N.; Hegde, S.; Raja, P. and Vishwanath, R. M.: "Towards Behavioural Cloning for Autonomous Driving". In: *2019 Third IEEE International Conference on Robotic Computing (IRC)*. 2019, pp. 560–567. doi: 10.1109/IRC.2019.00115 (cit. on p. 22).

[Sch15] Schulman, J.; Levine, S.; Abbeel, P.; Jordan, M. and Moritz, P.: "Trust Region Policy Optimization". In: *Proceedings of the 32nd International Conference on Machine Learning*. Ed. by Bach, F. and Blei, D. Vol. 37. Proceedings of Machine Learning Research. Lille, France: PMLR, July 2015, pp. 1889–1897. url: https://proceedings.mlr.press/v37/schulman15.html (cit. on p. 13).

[Sch17] Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A. and Klimov, O.: “Proximal Policy Optimization Algorithms”. In: *CoRR* abs/1707.06347 (2017). arXiv: 1707 . 06347. url: http://arxiv.org/abs/1707.06347 (cit. on p. 13).

[Sch22] Schmitt, P.; Britten, N.; Jeong, J.; Coffey, A.; Clark, K.; Kothawade, S. S.; Grigore, E. C.; Khaw, A.; Konopka, C.; Pham, L. et al.: “Can Cars Gesture? A Case for Expressive Behavior Within Autonomous Vehicle and Pedestrian Interactions”. In: *IEEE Robotics and Automation Letters* 7.2 (2022), pp. 1416–1423. doi: 10.1109/LRA.2021.3138161 (cit. on p. 87).

[Sen02] Sentz, K. and Ferson, S.: “Combination of evidence in Dempster-Shafer theory”. In: (2002) (cit. on p. 84).

[Sha16] Shalev-Shwartz, S.; Shammah, S. and Shashua, A.: “Safe, Multi-Agent, Reinforcement Learning for Autonomous Driving”. In: *CoRR* abs/1610.03295 (2016). arXiv: 1610 . 03295. url: http : //arxiv.org/abs/1610.03295 (cit. on p. 19).

[Sha17] Shalev-Shwartz, S.; Shammah, S. and Shashua, A.: “On a Formal Model of Safe and Scalable Self-driving Cars”. In: *CoRR* abs/1708.06374 (2017). arXiv: 1708.06374. url: http://arxiv.org/abs/1708.06374 (cit. on pp. 4, 27, 32–34).

[Sha18] Sharma, S.; Tewolde, G. and Kwon, J.: “Behavioral Cloning for Lateral Motion Control of Autonomous Vehicles Using Deep Learning”. In: *2018 IEEE International Conference on Electro/Information Technology (EIT)*. 2018, pp. 0228–0233. doi: 10.1109/EIT.2018.8500102 (cit. on p. 21).

[Sid22] Sidorenko, G.; Fedorov, A.; Thunberg, J. and Vinel, A.: “Towards a Complete Safety Framework for Longitudinal Driving”. In: *IEEE Transactions on Intelligent Vehicles* 7.4 (2022), pp. 809–814. doi: 10.1109/TIV.2022.3209910 (cit. on p. 27).

[Sil14] Silver, D.; Lever, G.; Heess, N.; Degris, T.; Wierstra, D. and Riedmiller, M.: “Deterministic Policy Gradient Algorithms”. In: *Proceedings of the 31st International Conference on Machine*

Learning. Ed. by Xing, E. P. and Jebara, T. Vol. 32. Proceedings of Machine Learning Research 1. Bejing, China: PMLR, 22–24 Jun 2014, pp. 387–395. url: https://proceedings.mlr.press/v32/silver14.html (cit. on p. 13).

[Son18] Sontges, S.; Koschi, M. and Althoff, M.: "Worst-case Analysis of the Time-To-React Using Reachable Sets". In: *2018 IEEE Intelligent Vehicles Symposium (IV)*. 2018, pp. 1891–1897. doi: 10.1109/IVS.2018.8500709 (cit. on p. 27).

[Sun18] Sun, L.; Zhan, W. and Tomizuka, M.: "Probabilistic Prediction of Interactive Driving Behavior via Hierarchical Inverse Reinforcement Learning". In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. Maui, HI, USA: IEEE Press, 2018, pp. 2111–2117. doi: 10.1109/ITSC.2018.8569453. url: https://doi.org/10.1109/ITSC.2018.8569453 (cit. on p. 22).

[Sut99] Sutton, R. S.; McAllester, D.; Singh, S. and Mansour, Y.: "Policy Gradient Methods for Reinforcement Learning with Function Approximation". In: *Proceedings of the 12th International Conference on Neural Information Processing Systems*. NIPS'99. Denver, CO: MIT Press, 1999, pp. 1057–1063 (cit. on p. 13).

[Taş17] Taş, Ö. Ş.; Hörmann, S.; Schäufele, B. and Kuhnt, F.: "Automated vehicle system architecture with performance assessment". In: *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*. 2017, pp. 1–8. doi: 10.1109/ITSC.2017.8317862 (cit. on p. 1).

[Taş18] Taş, Ö. Ş. and Stiller, C.: "Limited Visibility and Uncertainty Aware Motion Planning for Automated Driving". In: *2018 IEEE Intelligent Vehicles Symposium (IV)*. 2018, pp. 1171–1178. doi: 10.1109/IVS.2018.8500369 (cit. on p. 26).

[Tas20] Tas, Ö. S. and Stiller, C.: "Tackling Existence Probabilities of Objects with Motion Planning for Automated Urban Driving". In: *CoRR* abs/2002.01254 (2020). arXiv: 2002.01254. url: https://arxiv.org/abs/2002.01254 (cit. on p. 31).

[Tas21] Tas, Ö. S.: P3IV: Probabilistic Prediction and Planning for Intelligent Vehicles Simulator. https://github.com/fzi-forschungszentrum-informatik/P3IV. 2021 (cit. on p. 113).

[Taş21] Taş, Ö. Ş.; Hauser, F. and Lauer, M.: "Efficient Sampling in POMDPs with Lipschitz Bandits for Motion Planning in Continuous Spaces". In: *2021 IEEE Intelligent Vehicles Symposium (IV)*. Nagoya, Japan: IEEE Press, 2021, pp. 1081–1088. doi: 10.1109/IV48863.2021.9575303. url: https://doi.org/10.1109/IV48863.2021.9575303 (cit. on p. 18).

[Tra18] Tram, T.; Jansson, A.; Grönberg, R.; Ali, M. and Sjöberg, J.: "Learning Negotiating Behavior Between Cars in Intersections using Deep Q-Learning". In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. 2018, pp. 3169–3174. doi: 10.1109/ITSC.2018.8569316 (cit. on p. 19).

[Tre00] Treiber, M.; Hennecke, A. and Helbing, D.: "Congested traffic states in empirical observations and microscopic simulations". In: *Physical Review E* 62.2 (Aug. 2000), pp. 1805–1824. doi: 10.1103/physreve.62.1805. url: http://dx.doi.org/10.1103/PhysRevE.62.1805 (cit. on p. 62).

[Tri20] Triest, S.; Villaflor, A. and Dolan, J. M.: "Learning Highway Ramp Merging Via Reinforcement Learning with Temporally-Extended Actions". In: *2020 IEEE Intelligent Vehicles Symposium (IV)*. 2020, pp. 1595–1600. doi: 10.1109/IV47402.2020.9304841 (cit. on p. 19).

[Wan17] Wang, P. and Chan, C.-Y.: "Formulation of Deep Reinforcement Learning Architecture toward Autonomous Driving for On-Ramp Merge". In: *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*. Yokohama, Japan: IEEE Press, 2017, pp. 1–6. doi: 10.1109/ITSC.2017.8317735. url: https://doi.org/10.1109/ITSC.2017.8317735 (cit. on p. 19).

[Wan20a] Wang, L.; Fernandez, C. and Stiller, C.: "Generating Efficient Behavior with Predictive Visibility Risk for Scenarios with Occlusions". In: *Proc. IEEE Intl. Conf. Intelligent Transportation*

Systems. Rhodes, Greece, Sept. 2020, pp. 1–7. doi: 10.1109/ITSC45102.2020.9294403 (cit. on p. 26).

[Wan20b] Wang, L.; Fernandez, C. and Stiller, C.: "Realistic Single-Shot and Long-Term Collision Risk for a Human-Style Safer Driving". In: *2020 IEEE Intelligent Vehicles Symposium (IV)*. Las Vegas, USA, June 2020, pp. 2073–2080. doi: 10.1109/IV47402.2020.9304541 (cit. on p. 26).

[Wan20c] Wang, L.; Wu, Z.; Li, J. and Stiller, C.: "Real-Time Safe Stop Trajectory Planning via Multidimensional Hybrid A*-Algorithm". In: *Proc. IEEE Intl. Conf. Intelligent Transportation Systems*. Rhodes, Greece, Sept. 2020, pp. 1–7. doi: 10.1109/ITSC45102.2020.9294291 (cit. on p. 24).

[Wan21] Wang, L.; Burger, C. and Stiller, C.: "Reasoning about Potential Hidden Traffic Participants by Tracking Occluded Areas". In: *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*. 2021, pp. 157–163. doi: 10.1109/ITSC48978.2021.9564584 (cit. on pp. 47, 48, 51, 57).

[Wan23a] Wang, L.; Fernandez, C. and Stiller, C.: "High-Level Decision Making for Automated Highway Driving via Behavior Cloning". In: *IEEE Transactions on Intelligent Vehicles* 8.1 (2023), pp. 923–935. doi: 10.1109/TIV.2022.3169207 (cit. on pp. 32, 65, 112).

[Wan23b] Wang, L.; Fernandez, C. and Stiller, C.: "Learning Safe and Human-Like High-Level Decisions for Unsignalized Intersections From Naturalistic Human Driving Trajectories". In: *IEEE Transactions on Intelligent Transportation Systems* 24.11 (2023), pp. 12477–12490. doi: 10.1109/TITS.2023.3286454 (cit. on pp. 32, 69, 71, 86, 93, 120, 122–125).

[Wen21] Wen, M.; He, F.; Yue, Y.; Zhang, J.; Zhu, H. and Wang, D.: "Target-Driven Mapless Navigation for Self-Driving Car". In: *2021 IEEE International Conference on Unmanned Systems (ICUS)*. 2021, pp. 505–511. doi: 10.1109/ICUS52573.2021.9641134 (cit. on p. 30).

[Wis17] Wissing, C.; Nattermann, T.; Glander, K.-H.; Hass, C. and Bertram, T.: “Lane Change Prediction by Combining Movement and Situation based Probabilities”. In: *IFAC-PapersOnLine* 50.1 (2017). 20th IFAC World Congress, pp. 3554–3559. doi: https://doi.org/10.1016/j.ifacol.2017.08.960. url: https://www.sciencedirect.com/science/article/pii/S2405896317314337 (cit. on p. 84).

[Woo18] Woo, H.; Ji, Y.; Tamura, Y.; Kuroda, Y.; Sugano, T.; Yamamoto, Y.; Yamashita, A. and Asama, H.: “Advanced Adaptive Cruise Control Based on Collision Risk Assessment”. In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. 2018, pp. 939–944. doi: 10.1109/ITSC.2018.8569759 (cit. on p. 26).

[Wu19] Wu, Y.-H.; Charoenphakdee, N.; Bao, H.; Tangkaratt, V. and Sugiyama, M.: “Imitation Learning from Imperfect Demonstration”. In: *Proceedings of the 36th International Conference on Machine Learning*. Ed. by Chaudhuri, K. and Salakhutdinov, R. Vol. 97. Proceedings of Machine Learning Research. PMLR, Sept. 2019, pp. 6818–6827. url: https://proceedings.mlr.press/v97/wu19a.html (cit. on p. 23).

[Wu22] Wu, J.; Huang, W.; Boer, N. de; Mo, Y.; He, X. and Lv, C.: “Safe Decision-making for Lane-change of Autonomous Vehicles via Human Demonstration-aided Reinforcement Learning”. In: *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*. 2022, pp. 1228–1233. doi: 10.1109/ITSC55140.2022.9921872 (cit. on p. 20).

[Xu21] Xu, X.; Wang, X.; Wu, X.; Hassanin, O. and Chai, C.: “Calibration and evaluation of the Responsibility-Sensitive Safety model of autonomous car-following maneuvers using naturalistic driving study data”. In: *Transportation Research Part C: Emerging Technologies* 123 (2021), p. 102988. doi: https://doi.org/10.1016/j.trc.2021.102988. url: https://www.sciencedirect.com/science/article/pii/S0968090X21000231 (cit. on p. 37).

[Xu23] Xu, C.; Zhao, W.; Wang, C.; Cui, T. and Lv, C.: “Driving Behavior Modeling and Characteristic Learning for Human-like Decision-Making in Highway”. In: *IEEE Transactions on Intelligent Vehicles* 8.2 (2023), pp. 1994–2005. doi: 10.1109/TIV.2022.3224912 (cit. on p. 22).

[Yu20] Yu, M.-Y.; Vasudevan, R. and Johnson-Roberson, M.: “Risk Assessment and Planning with Bidirectional Reachability for Autonomous Driving”. In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. 2020, pp. 5363–5369. doi: 10.1109/ICRA40945.2020.9197491 (cit. on p. 26).

[Zha19] Zhan, W.; Sun, L.; Wang, D.; Shi, H.; Clausse, A.; Naumann, M.; Kümmerle, J.; Königshof, H.; Stiller, C.; La Fortelle, A. de et al.: “INTERACTION Dataset: An INTERnational, Adversarial and Cooperative moTION Dataset in Interactive Driving Scenarios with Semantic Maps”. In: *arXiv:1910.03088 [cs, eess]* (2019) (cit. on pp. 66, 113).

[Zha22] Zhao, Z.; Wang, Z.; Han, K.; Gupta, R.; Tiwari, P.; Wu, G. and Barth, M. J.: “Personalized Car Following for Autonomous Driving with Inverse Reinforcement Learning”. In: *2022 International Conference on Robotics and Automation (ICRA)*. 2022, pp. 2891–2897. doi: 10.1109/ICRA46639.2022.9812446 (cit. on p. 22).

[Zie08] Ziebart, B. D.; Maas, A.; Bagnell, J. A. and Dey, A. K.: “Maximum Entropy Inverse Reinforcement Learning”. In: *Proceedings of the 23rd National Conference on Artificial Intelligence - Volume 3*. AAAI’08. Chicago, Illinois: AAAI Press, 2008, pp. 1433–1438 (cit. on p. 14).

[Zie14] Ziegler, J.; Bender, P.; Schreiber, M.; Lategahn, H.; Strauss, T.; Stiller, C.; Dang, T.; Franke, U.; Appenrodt, N.; Keller, C. G. et al.: “Making Bertha Drive—An Autonomous Journey on a Historic Route”. In: *IEEE Intelligent Transportation Systems Magazine* 6.2 (2014), pp. 8–20. doi: 10.1109/MITS.2014.2306552 (cit. on p. 127).

[Zie15] Ziegler, J.: “Optimale Bahn- und Trajektorienplanung für Automobile”. German. PhD thesis. 2015. 118 pp. doi: 10.5445/IR/1000057846 (cit. on pp. 73, 75).

Acronyms

ACC	Adaptive Cruise Control
AD	Autonomous Driving
AV	Autonomous Vehicle
BC	Behavior Cloning
CGMP	Closest-Gap Merging Policy
CIDM	Cooperative Intelligent Driver Model
CMOBIL	Cooperative Minimizing Overall Braking Induced by Lane changes
DAgger	Dataset Aggregation
DQN	Deep Q-Network
FoV	Field of View
GAIL	Generative Adversarial Imitation Learning
GPU	Graphics Processing Unit
HD	High-Definition

IDM	Intelligent Driver Model
IIDM	Intelligent Driver Model for Intersection
IL	Imitation Learning
IRL	Inverse Reinforcement Learning
LAIP	Learned Aggressive Intersection Policy
LALCP	Learned Aggressive Lane Change Policy
LAMP	Learned Aggressive Merging Policy
LDIP	Learned Defensive Intersection Policy
LDLCP	Learned Defensive Lane Change Policy
LDMP	Learned Defensive Merging Policy
LIP	Learned Intersection Policy
LIPBP	Learned Intersection Policy with Better Prediction
LLCP	Learned Lane Change Policy
LMP	Learned Merging Policy
LSTM	Long Short-Term Memory
MCS	Monte-Carlo Simulation
MCTS	Monte Carlo Tree Search
MDE	Mean Distance Error

MDP	Markov Decision Process
MIDM	Intelligent Driver Model for Merging
MOBIL	Minimizing Overall Braking Induced by Lane changes
PCA	Principal Component Analysis
PI	Policy Iteration
POMDP	Partially Observable Markov Decision Process
RBLMP	Risk-Bounded Learned Merging Policy
RE	Regulatory Element
RL	Reinforcement Learning
ROS	Robot Operating System
RSS	Responsibility-Sensitive Safety
THW	Time Headway
TTC	Time to Collision
TTR	Time to React
TZC	Time of Zone Clearance
VI	Value Iteration

A Appendix

A.1 Parameters of Different Driving Styles

Table A.1: IDM parameters for different driving styles.

	Aggressive	Normal	Defensive	Trucks	Unit
a	2.5	2	1.5	1	m/s^2
d_0	1.5	2	3	5	m
T_d	1.2	1.5	2	1.5	s
b	-3	-2	-1.5	-1	m/s^2

Table A.2: RSS parameters for different driving styles and the relaxed RSS parameter.

	Aggressive	Normal	Defensive	Trucks	Relaxed	Unit
ρ_{ego}	0.3	0.4	0.5	0.6	0.2	s
ρ_{obj}	0.5	0.7	1	0.8	0.5	s
$a_{max,dcc,ego}$	-6	-8	-6	-4	-8	m/s^2
$a_{max,dcc,obj}$	-6	-10	-10	-6	-6	m/s^2
$a_{max,acc,ego}$	2.5	2	1.5	1	3	m/s^2
$a_{max,acc,obj}$	2	3	3.5	3	2	m/s^2
$a_{max,acc,lat,ego}$	5	4	3	2	5	m/s^2
$a_{max,acc,lat,obj}$	6	7.83	9	7.83	6	m/s^2
$a_{soft,dcc,obj}$	-2.5	-2	-1.5	-2	-3	m/s^2
$t_{TZC,min}$	0.3	0.5	0.8	0.5	0.3	s

Table A.3: Parameters for computing yielding motivation.

Styles	θ_Y
Aggressive	[0.08, 1.4, -5]
Normal	[0.1, 1.8, -4.8]
Defensive	[0.15, 2, -4.5]
Trucks	[0.08, 1.4, -5]

Table A.4: MOBIL parameters for different driving styles.

	Aggressive	Normal	Defensive	Trucks	Unit
p	0.5	0.9	1.0	0.5	s
a_{th}	0.3	0.5	0.7	0.3	s

A.2 Parameters for MCS

Table A.5: Parameters for MCS setup.

$t_{mcs,max}$	12 s
N	500
Δt	0.3 s

Table A.6: Parameters for estimating pedestrian crossing intention.

θ_p	[0.75, -0.5, -1.5]
b_p	-2

Table A.7: Parameters for estimating yielding intentions of prioritized vehicles during merging

$\hat{\theta}_Y$	[0.08, 1.4, -5, -1.1]

A.3 Learned Weights of All Policies

Table A.8: Learned weights w for merging, free lane change and right-of-way intersection scenarios.

	Utility			Comfort	Risk		Politeness	
	U_1	U_2	U_3	C	R_1	R_2	P_1	P_2
$\mathbf{w}_{\text{merge}}$	0.5	-1	0.05	0.05	-0.7	-0.5	0.1	0.15
$\mathbf{w}_{\text{free}}$	0.183	-0.15	0.3	0.1	-0.367	-0.34	1.0	0.25
$\mathbf{w}_{\text{inter}}$	1	-0.95	0.88	0.08	-0.16	-0.5	0.16	0.16

Table A.9: Learned stylized (aggressive and defensive) weights for merging, free lane change and right-of-way intersection scenarios.

	Utility			Comfort	Risk		Politeness	
	U_1	U_2	U_3	C	R_1	R_2	P_1	P_2
$\mathbf{w}_{\text{merge,agg}}$	0.7	-1	0.08	0.02	-0.5	-0.4	0.06	0.12
$\mathbf{w}_{\text{merge,def}}$	0.4	-0.8	0.04	0.9	-1	-0.7	0.15	0.2
$\mathbf{w}_{\text{free,agg}}$	0.25	-0.23	0.5	0.06	-0.27	-0.3	1.0	0.14
$\mathbf{w}_{\text{free,def}}$	0.14	-0.12	0.2	0.15	-0.45	-0.4	1.0	0.33
$\mathbf{w}_{\text{inter,agg}}$	1	-0.93	0.95	0.03	-0.12	-0.12	0.12	0.04
$\mathbf{w}_{\text{inter,def}}$	1	-0.79	0.66	0.23	-0.44	-0.8	0.25	0.79