High-Level Decision Making for Automated Highway Driving via Behavior Cloning

Lingguang Wang, Carlos Fernandez and Christoph Stiller

***Abstract*—Automated driving systems need to perform according to what human drivers expect in every situation. A different behavior can be wrongly interpreted by other human drivers and cause traffic problems, disturbances to other road participants, or in the worst case, an accident. In this paper, we propose a behavior cloning concept for learning high-level decisions from recorded trajectories of real traffic. We summarized and gave a clear definition of the main features that affect how humans make driving decisions. Some other approaches rely on complex neural networks where their decisions are impossible to understand. Due to the importance of the decision making module, we produce safe human-like behavior which is transparent to humans and easy to track. Furthermore, the learned policy is not overfitting to the limited training data and generalizes well to multi-lane scenarios with arbitrary speed limits and traffic density, which is strengthened by the successful application of merging policy on exiting. Simulation evaluations show that our learned policy is able to handle the intention uncertainty of surrounding agents, and provide human-like decisions in the sense of well-balanced behavior between efficiency, comfort, perceived safety, and politeness.**

***Index Terms*—Automated driving, decision making, Monte-Carlo Simulation, behavior cloning, Responsibility-Sensitive-Safety**

I. Introduction & State of the Art

AUTONOMOUS driving is a very complex task that was under research for decades. In the future, all vehicles will drive without human intervention and V2X communications will help to improve safety and optimize vehicle decisions based on precise knowledge of the surroundings. Nowadays, autonomous vehicles should drive under mixed traffic where human drivers are often difficult to predict. From human drivers' point of view, autonomous vehicles decisions should be similar to what humans would do but with safety as an essential requirement for all situations.

There are different approaches addressing the problem of human-like behavior generation. Many researchers focus on Reinforcement Learning (RL) trying to achieve intelligent driving behavior in a highly interactive environment [1], [2]. However, these approaches usually face some challenges. One is that the learned policy in simulated environments is hardly transferable to real-world environments. Moreover, designing a suitable reward function is not straightforward in practice. The author's preferences and personal experience often influence how they design their reward function. Therefore, proving that the learned policy is generally human-like is hard. Once the user wants to choose a different driving style, e.g. weight the reward terms differently, the policy should be trained from scratch again. Some recent approaches [3], [4] try to learn the value distribution instead of the value function and afterwards add α value in their approach to also tune the policy in real-time. However, adjusting multiple parameters online is still intractable.

Inverse Reinforcement Learning (IRL) [5]–[7] try to utilize the collected human demonstrations for learning a proper reward function, instead of engineering one, where imitating human driving style becomes possible. One well-known shortage of these approaches is to apply to high-dimensional problems with unknown dynamics. Some improved IRL approaches, e.g. Generative Adversarial Imitation Learning (GAIL) [8] and Adversarial Inverse Reinforcement Learning (AIRL) [9], are able to overcome this downside. For Instance, AIRL is able to learn a reward function and value function simultaneously and is demonstrated to perform well on high-dimensional tasks. However, as the learned value function is usually represented by a neural network to work on high-dimensional input space, knowing what exactly happened inside the network to understand why a certain decision is made is still not possible, since visualizing the data flow of the neural network is still a remaining challenge.

Almost all present Behavior Cloning (BC) approaches [10], [11] try to solve the autonomous driving problem end-to-end. They receive the RGB image of the front view as input and directly output control commands (acceleration, steering angle, etc.). The policy can be learned totally offline. With proper data augmentation, the BC approach can yield state-of-the-art driving behavior even in limited unseen environments. However, it still suffers from overfitting and performs badly in new scenarios with strongly interactive dynamic objects. In order to generalize enough in the target scenarios, an extremely huge amount of training data is needed, which should contain as little bias and variance as possible. Understanding the learned policy of these approaches is even more intractable as it covers the perception part as well.

Instead of an end-to-end BC approach, we propose one BC approach that only replaces the task of high-level decision making or behavior generation, which is in the middle of the whole autonomous driving pipeline, between perception stage and trajectory generation. The policy is highly modular, easy to parallelize and despite this paper is focused on highway scenarios, it can generalize to different scenarios.

This work is accomplished within the project "UNICARagil" (FKZ 16EMO0287) and the financial support from the Federal Ministry of Education and Research of Germany (BMBF) is acknowledged.

Lingguang Wang, Carlos Fernandez and Christoph Stiller are with the Institute of Measurement and Control Systems, Karlsruhe Institute of Technology (KIT), 76131 Karlsruhe, Germany, lingguang.wang, carlos.fernandez, stiller@kit.edu

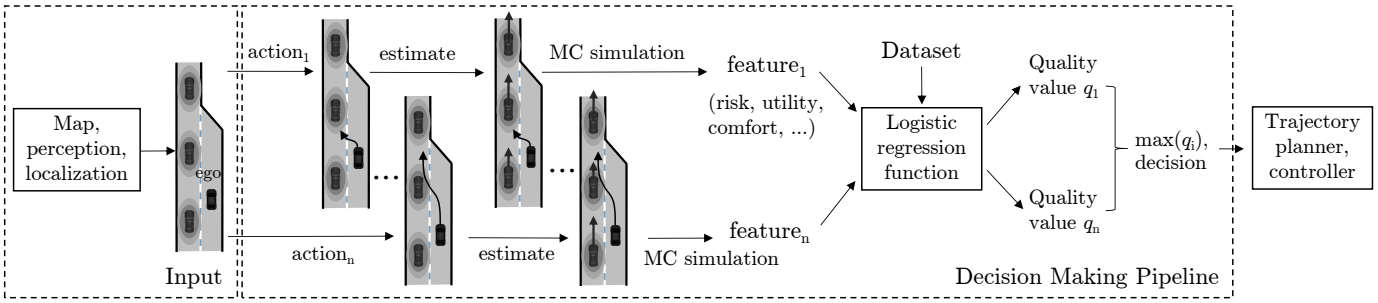


Fig. 1: Proposed decision making pipeline taking a merging scenario as an example.

The proposed decision making approach is presented in Fig. 1. As the input of our approach, we assume that the perception of the environment and the precise self-localization are done by other modules. Thus, the noisy states (location, velocity, size, etc.) of other traffic participants within certain sensing ranges are provided. Another important input is the High-Definition (HD) Map, e.g. lanelet2 [12], where the road topology and all the traffic rules are embedded. The output of our module is one high-level decision that is delivered to a trajectory planner, which connects to a low-level controller to complete the controlling of the vehicle.

At each decision step, we first generate possible reasonable high-level action candidates that fit the prior HD maps and fulfill the current traffic rules. For every action, several feature values (utility, comfort, risk, etc.) that characterize the action will be generated. In order to do that, we simulate the scene forward stochastically, given the uncertain states and estimation of intentions of surrounding agents, and assuming the ego vehicle follows the selected action. Performing a large number of Monte-Carlo (MC) simulations for each action, we understand how each decision affects the environment and other road users. Furthermore, how risk and other statistics change by doing a specific action is inferred.

There are other approaches that model the problem in the literature. One of the most common ones is POMDP [13], [14]. They can deal with the uncertainty of surroundings and strong interaction with the environment as well. They often utilize Monte-Carlo-Tree-Search to reveal the value function and solve POMDP. However, reward function engineering and solving POMDP in real-time are still remaining challenges. Instead of mapping the whole state space and action into a quality value with the value function, we try to map the feature values of each action to one single quality value with one logistic regression function, that represents how good the action is, similar to a cost function. By making use of the recorded trajectory, the ground-truth action that humans performed in the dataset can be revealed as well. The goal is to learn the weights of the logistic regression function which in the end outputs higher values for the human-like action and lower values for other actions. In this paper, we apply this approach and learn proper free lane change policy and merging policy in highway scenarios, thanks to the availability of the HighD [15], ExitD and Interaction Datasets [16]. In order to present the performance of our approach, we build a probabilistic simulator from scratch for highway scenarios and evaluate the policies there. Statistics show that our learned policies outperform the baseline policies with a large margin in most of the metrics.

To the authors' best knowledge, no work before has investigated high-level decision-making via behavior cloning. We summarize the following novelty and contributions of our approach:

- We systematically summarize and define possible features that could influence driving decisions, and integrate them in our framework.
- We estimate the feature values via MC Simulation, where the states and intention uncertainty of surrounding vehicles and possible reactions are considered.
- We propose one novel framework for learning human-like driving policy from real data that is able to incorporate prior knowledge from humans (HD-maps, traffic rules) and provide high-level decisions for subsequent modules.
- For lane change and merging on multi-lane roads, e.g. highway, we extend the Responsibility-Sensitive-Safety (RSS) concept [17] with additional realistic assumptions.

II. Backend Components for Policy

A. Relevant Features for Decision Making

One previous work [18] already summarizes existing features in state of the art. However, they mostly focus on relevant features for trajectory planning and not for high-level decision making. Therefore, we have adapted and summarized all the features in four groups which are specially important in behavior-level: *utility, ride comfort, perceived safety and politeness*. In order to make them comparable with each other, they are normalized between 0 and 1.

1) Utility: This feature group represents how soon the driver can reach his desired goal and how possible the desired maneuver can be accomplished. We summarize three *utilities*:

- U_1: How much progress can be achieved within a certain time.
- U_2: How much time the desired maneuver is needed.
- U_3: How possible the desired maneuver can be finished.

U_1 can be relevant in all scenarios. For example, high U_1 could mean less deviation from the desired velocity of the driver or the speed limit. In scenarios where certain maneuvers are clearly defined, U_2 and U_3 are more interesting. As an example, in an on-ramp merging scenario, selecting a gap that requires less merging time (U_2) and is more probable to

success (U_3) is usually more important than cutting into the very first gap that is the fastest option (U_1).

In highway scenario, U_1 can be formulated as $1-|\frac{v}{v_{\text{des}}}-1|$ where v denotes the average velocity achieved by a maneuver or a trajectory and v_{des} represents the desired velocity of the driver. We use desired velocity instead of the speed limit because they often deviate from each other. For instance, the utility is considered not to be damaged when a driver drives a low-performance car with 100 kph on a lane with 120 kph speed limit, as 100 kph could fit best to his vehicle and his desire. In contrast, sport car drivers might drive close below punished speed violation limits, which in some countries is 1.1 times the speed limit. Even driving at the speed limit is considered to damage their utility. Note that for automated vehicles (ego), the desired velocity should be set within the legal range.

2) Ride comfort: In trajectory planning, jerk and acceleration in longitudinal and lateral direction are often to be punished in the cost function. However, in behavior generation, creating plans that are optimal w.r.t. jerk is not necessary. Therefore we only include longitudinal acceleration C_1 and lateral acceleration C_2 in decision making in general scenarios. In highway driving conditions that is focused in this paper, we pay attention particularly to longitudinal comfort $C_1 = 1-|\frac{a_{\text{l}}}{a_{\text{max}}}|$, where a_{l} is the average absolute longitudinal acceleration of a maneuver or a trajectory and a_{max} is the maximumly executable deceleration of the vehicle.

3) Perceived safety: Perceived safety is also treated as risk and it has several definitions in the state of the art depending on the scenario where it is applied. For right of way at intersections, the time that elapses between one vehicle leaving a conflict zone and another traffic participant entering this zone [19]. In follow driving condition, space headway d, time headway t_{TH} and time to collision t_{TTC} are often utilized for describing how safe the following vehicle is. Some other works compute risk probabilistically [20], where a probabilistic prediction of other traffic participants is assumed to be provided. Given the ego trajectory, the probability of collision can be computed. These approaches rely on an upstream prediction of others that is not depending on the ego vehicle. However, collision probability could also change depending on the reaction time of each driver. Therefore, computing real collision probability considering all possible reactions of the involved agents is almost computationally intractable.

Instead of labeling events as risky where a final collision can occur with a certain probability, we count the ones as risky where harsh reactions of the vehicle are needed to avoid collision, e.g. emergency brake or an emergency evade, disregard of whether the evasive reaction helps and whether a final collision occurs. The margin where the evasive reaction should be performed is when the RSS safety is broken. The risk R is then defined to be the probability that a harsh or an evasive reaction of the ego vehicle is needed to maintain RSS safety.

4) Politeness: Courtesy during driving is important to improve traffic smoothness and increase safety. An experienced driver focuses not only on their own benefit but behaves in a way such that others' comfort and utility are affected as little as possible. Polite behaviors can often prevent future risky events as well, e.g. yield for merging vehicles. We measure politeness P by looking at the utility U_1 and comfort C_1 of the vehicle that is influenced the most by the ego vehicle's action where $U_{1,i}$ and $C_{1,i}$ is the first type of utility and comfort of i-th vehicle. n is the total number of surrounding vehicles.

$$P_1 = \min_{i=1}^{n} U_{1,i}, P_2 = \min_{i=1}^{n} C_{1,i} \tag{1}$$

B. Action Space

At every decision step, the vehicle can choose between different action candidates. The detailed planning and execution of the requested maneuver will be done by subsequent modules. We differentiate the action candidates in free lane change scenario and merging/exit scenario, where the main difference is whether a mandatory target lane exists.

1) High-Level action classes: In free lane change scenarios, we define five semantic actions:

- a_1: Keep in the current lane
- a_2: Decelerate in the current lane
- a_3: Accelerate in the current lane
- a_4: Change lane to the left into the current gap
- a_5: Change lane to the right into the current gap

In merging and exiting scenarios, the number of possible actions is variable. We concentrate mainly on the possible gaps implicitly constructed by the vehicles on the target lane. We focus at maximumly four vehicles on the target lane which are longitudinally closest to the ego vehicle and within the sensing range of the onboard sensors. Therefore the number of actions is automatically limited to five. Four of them are merging in front of the target vehicles, and the last one is merging to the very last gap after the last target vehicle. The merging actions are assigned with notation a_{gap_i} where i denotes the gap number. Our proposed pipeline illustrated in Fig. 1 can tackle a variable number of candidate actions.

2) Proof-of-concept low-level control with customized IDM: For a proof-of-concept controlling of the vehicles, the actions that are used in this paper are decoupled in longitudinal and lateral directions. They are defined as $a_i = [a_{\text{lon}}, v_{\text{lat}}]$ where a_{lon} and v_{lat} denote the longitudinal acceleration and lateral velocity.

The Intelligent Driver Model (IDM) [21] generates the longitudinal acceleration $\dot{v}_{\text{IDM}}$ which is determined by

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

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

$$d^*\left(v,\Delta v\right) = d_0 + vT_{\text{d}} + \frac{v\Delta v}{2\sqrt{ab}} \tag{3}$$

The parameters to set are: 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 a function of the velocity difference Δv and the distance to the

front vehicle d. The action a_1 can directly utilize the output of IDM, while a_2 can be realized by reducing v_d and increasing T_d and a_3 the opposite.

This formula has several drawbacks according to [22], e.g. the output acceleration can be unlimited which is physically not realizable. For our application, it is only suitable for follow driving actions a_1, a_2 and a_3. For other actions where the ego vehicle should properly cut into the desired gap constructed by two vehicles, it is not applicable. Sometimes even both vehicles of the gap can be behind or in front of the ego vehicle, which results even in negative distances d. Furthermore, during longitudinal adjustment, a proper distance to the leading vehicle on the source lane is to be kept as well, i.e. more than one leading vehicle should be taken into account. In order to address these issues, we introduce some modifications from [22] and customize the IDM model. The modified $\dot{v}_\mathrm{IDM}$ can be defined by

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

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 our 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 as well via $a_\mathrm{lon} = \min\{\max\{\dot{v}_\mathrm{IDM}, -a\}, a\}$ to the range of $[-a, a]$. Fig. 2 illustrates two examples of how the customized IDM controls the ego vehicle to fit into different gaps that are moving with constant velocity.

The lateral velocity v_lat of the actions a_1, a_2 and a_3 are 0 as they do not involve lateral movement. 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 [23] where the positive sign of v_lat points to the target lane.

$$v_\mathrm{lat} = \begin{cases} \min\{0.17v, 0.8\frac{\mathrm{m}}{\mathrm{s}}\} & \text{RSS safe w.r.t. the gap} \\ -\min\{0.17v, 0.8\frac{\mathrm{m}}{\mathrm{s}}\} & \text{else} \end{cases} \tag{5}$$

The prerequisite for having lateral velocity is the longitudinal RSS safety w.r.t. the leading and following vehicle. The definition of RSS safety will be explained in the next chapter. Note that in each action, a fall-back longitudinal reaction is included. As soon as the RSS safety does not hold, e.g. by other vehicles cutting in front closely or because the merging lane is going to end[1], the emergency braking $a_\mathrm{emerg,decel}$ will overwrite the output of the customized IDM.

[1]Equivalent to an obstacle with velocity $0\frac{\mathrm{m}}{\mathrm{s}}$ standing at the end of merging lane

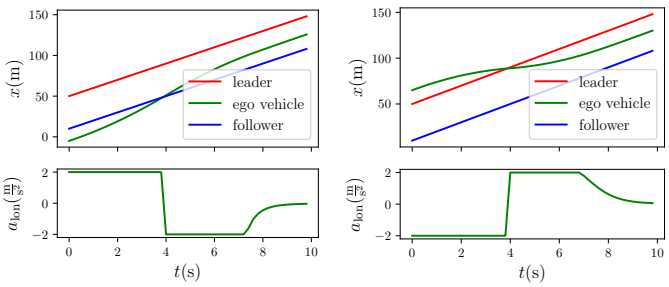


Fig. 2: Longitudinal position x and acceleration a_lon of the ego vehicle with the customized IDM. The leader and the follower of the target gap are moving with constant velocity. (a) The ego vehicle is initially behind the gap. (b) Ego vehicle is initially in front of the gap.

C. Safety consideration

According to RSS "common sense" rule 5, the cut-in from ego vehicle shall not be reck-less, i.e. a lane change or a merging cannot be performed with an arbitrarily small distance to the following vehicle in the target lane. Furthermore, during the lane change process, longitudinal RSS safety to the leading vehicle on the target lane and on the source lane shall not be harmed as well. We will first shortly revise the longitudinal RSS safety on a single lane which is comprehensively discussed in [24]. Afterward, we extend the RSS merging safety concept introduced in [25] to cover the scenario of merging into a gap of two vehicles. Finally, we introduce some additional reasonable assumptions to increase the feasibility of the RSS safety in reality.

1) Longitudinal RSS safety on a single lane: The basic RSS safety is defined in a leader-follower setup on a single lane. It stresses that the follower should keep a minimum safe distance d_safe to the front vehicle that is related to the current velocity of both, and some other parameters, such as reaction time of the follower, maximum deceleration of the leader $a_\mathrm{max,decel}$, minimum deceleration of the follower $a_\mathrm{min,decel}$, etc. It ensures that the follower is guaranteed to not collide with its predecessor even in some reasonable worst case that is defined by the assumed parameters. The formulas are not repeated here but the readers are referred to [17] for details.

2) RSS safety for lane change: The following safety concept to ensure a smooth and not so conservative merging, which is suitable for lane change as well, is presented in [25]. The merging vehicle intends to accelerate with some minimum acceleration $a_\mathrm{min,accel}$, unless it already reaches the speed limit or cannot accelerate due to the front vehicle on the target and source lane. The prioritized following vehicle on the target lane reacts by decelerating with $a_\mathrm{soft,decel}$ after the merging vehicle is on the target lane plus a usual reaction time. 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 to the unlimited future. An analytical solution of the initially required safe distance $d_\mathrm{safe,init}$ to the prioritized following vehicle can be found in [25] when the merging vehicle can do a minimum acceleration $a_\mathrm{min,accel}$ during the whole time.

However, when taking the leading vehicle on the target and source lane into account, an analytical solution might not exist, since $a_{\text{min,accel}}$ of merging vehicle might be disturbed. In our extension, we utilize numerical simulation to check the merging safety into a gap. The merging vehicle is assumed to adjust its speed to the gap via the customized IDM in section II-B2 instead of constant acceleration. The leading vehicle on the target lane continues with constant velocity.

In order to see how realistic the safety concept is, we further examine in the real data how many percent of human drivers obey this rule. We use HighD and ExitD datasets for this purpose. By selecting a parameter set $a_{\text{soft,decel}} = -1.2\frac{\text{m}}{\text{s}^2}$ which is considered as in comfortable range in [26], the reaction time of the ego vehicle 0.4s (anticipated in [27]) and the reaction time for other vehicles 0.7s (argued in [28]), the Table I presents the percentage of safe lane changes in dataset with different $a_{\text{min,decel}}$ and $a_{\text{max,decel}}$.

TABLE I: 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 $a_{\text{max,decel}} \geq a_{\text{min,decel}}$ is assumed.

ratio of safe lane changes		$a_{\text{max,decel}}(\frac{\text{m}}{\text{s}^2})$			
		-6	-8	-10	-12
$a_{\text{min,decel}}(\frac{\text{m}}{\text{s}^2})$	-6	0.877	0.875	0.563	0.511
	-8		0.886	**0.761**	0.677
	-10			0.889	0.804
	-12				0.885
ratio of safe merges		$a_{\text{max,decel}}(\frac{\text{m}}{\text{s}^2})$			
		-6	-8	-10	-12
$a_{\text{min,decel}}(\frac{\text{m}}{\text{s}^2})$	-6	0.928	0.846	0.723	0.662
	-8		0.943	**0.909**	0.860
	-10			0.949	0.933
	-12				0.955

It can be seen that with the recommended parameter set $a_{\text{max,decel}} = -10\frac{\text{m}}{\text{s}^2}$ and $a_{\text{min,decel}} = -8\frac{\text{m}}{\text{s}^2}$ from [24], only 76.1% of all the lane changes in HighD dataset and 90.9% of all merges in ExitD dataset are RSS safe. The ratio of unsafe ones are significantly lower as the recorded accidents quote (which is 0 in the datasets). This is an indication that this RSS safety concept for lane change does not match human consensus somewhere.

3) Extended RSS safety for lane change with additional assumptions: The large number of violations of RSS safety in naturalistic traffic is shortly explained in [24]. It is claimed that an emergency deceleration of the predecessor during a lane change does not occur without any prior warning, e.g. the emergency brake of the pre-predecessor. Therefore, if two predecessors can be observed by the ego vehicle, the $a_{\text{max,decel}}$ of the predecessor can be reduced depending on how far the pre-predecessor is. In case that the pre-predecessor is extremely far or does not appear within the sensing range, the predecessor should not brake maximumly without any reason. 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 until the lane change is complete and all the vehicles maintain a stable state[2]. If so, the lane change action should be aborted immediately.

The authors of [24] suggest to lower the $a_{\text{max,decel}}$ of the predecessor to a fixed value (e.g. $-4\frac{\text{m}}{\text{s}^2}$) for a short period of time when the pre-predecessor can be observed. However, we propose to not assume less than $0.5a_{\text{max,decel}}$, which is $-5\frac{\text{m}}{\text{s}^2}$ when choosing $a_{\text{max,decel}} = -10\frac{\text{m}}{\text{s}^2}$. Note that only 6 of all 107613 trajectories in HighD dataset has higher than $-5\frac{\text{m}}{\text{s}^2}$ deceleration, all for avoiding crash to the front vehicle. Furthermore, depending on the distance of pre-predecessor, the possible acceleration of predecessor can vary instead of a fixed value. Therefore we propose the following safety concept: assuming the pre-predecessor[3] brake with $a_{\text{max,decel}}$, the needed deceleration for the predecessor not colliding with pre-predecessor is a_{need}, the predecessor will brake with $a'_{\text{max,decel}}$ which bounds a_{need} between $0.5a_{\text{max,decel}}$ and $a_{\text{max,decel}}$. The new safe distance d'_{safe} can be calculated correspondingly.

With the additional assumptions, we check again the safe lane change ratio in datasets and obtain Table II. The extended RSS safety yields much fewer violations. With the parameter set $a_{\text{max,decel}} = -10\,m/s^2$ and $a_{\text{min,decel}} = -8\,m/s^2$, the safe lane change and merging ratios increase from 76.1% and 90.9% to 92.1% and 97.5% respectively. Despite the extended RSS safety concept is still less optimistic than most human drivers, we think is a reasonable balance between safety and human consensus.

TABLE II: Percentage of RSS safe lane changes with additional assumption.

ratio of safe lane changes		$a_{\text{max,decel}}(\frac{\text{m}}{\text{s}^2})$			
		-6	-8	-10	-12
$a_{\text{min,decel}}(\frac{\text{m}}{\text{s}^2})$	-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		$a_{\text{max,decel}}(\frac{\text{m}}{\text{s}^2})$			
		-6	-8	-10	-12
$a_{\text{min,decel}}(\frac{\text{m}}{\text{s}^2})$	-6	0.990	0.945	0.876	0.811
	-8		0.993	**0.975**	0.947
	-10			0.995	0.987
	-12				0.995

We recommend using the extended RSS safety concept on scenarios where the precondition of the additional assumptions can be checked at a low cost (merging and extremely sparse traffic). For free lane change on multiple lanes, checking the preconditions against all the neighboring vehicles is almost intractable and we recommend applying RSS safety concept described in Section II-C2.

III. Behavior Cloning Approach

A. Features Acquisition via Monte-Carlo Simulation

In previous chapters, we formulated the features that are relevant to characterize how good an action is in many aspects. All candidate actions are safe in the sense of our extended RSS safety for cut-in. At least, autonomous vehicles shall not be blamed if they obey the RSS safety rule and accidents

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

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

happen. In this chapter, we explain how the distinct features can be obtained for each action, e.g. how convenient and risky merging into a certain gap will be.

Human drivers always have a complete picture of the whole driving scene in mind, and make decisions that are not optimal but reasonable considering all the possible evolutions of the scene. POMDPs try possible actions and simulate the reactions of the environmental agents by probabilistic driver models, in order to find the best policy given the state or belief space and provide online plans. However, they still suffer from complexity in large multi-agent settings, as the cardinality of the action and observation space grows exponentially in the number of agents. Instead of searching online action sequence that is well optimized, we regard our proposed semantic high-level action candidates that respect traffic rules and HD maps as suboptimal options. We claim that in real traffic, it is more realistic and sufficient to achieve a certain degree of convenience and safety, rather than to be optimal w.r.t. an engineered reward function. Therefore, we target the goal to select one from those suboptimal options. The output is not a fixed trajectory like open-loop planning approaches but a homotopy class, which provides online interactive plans similar to POMDP at a lower computational cost.

The decision is made depending on the features (risk, utility, comfort, etc.) associated with each action. In order to acquire those, we try to simulate the environment forward starting from the current sensed scene, where the ego vehicle consistently follows one of the actions, the surrounding agents are sampled from their states (position, velocity, acceleration, etc.) distributions and will act by sampling from their estimated probabilistic driver models and intention models. We will explain how the driver models and intention models are estimated in later chapters. For each action of the ego vehicle, the simulation will repeat enough times (MC simulation), where other agents behave randomly in order to reveal possible future developments. The results of all simulations will be summarized and averaged as features for each action, such as how often the ego vehicle is expected to fall back to emergency braking by trying one certain gap, or how fast the ego vehicle is expected to drive by changing lane to left.

MC simulations are associated with a maximum simulation horizon t_{max} and the features $[U_1, U_2, U_3, C_1, R, P_1, P_2]$ in Chapter II-A can be approximated. From our simulations, we obtain numerical values of the defined features and also other semantic information. On one hand, the feature values are averaged among all MC simulations following eq. 6 and eq. 7. Note that eq. 6 applies for C_1^*, P_1^* and P_2^* as well. On the other hand, when the action succeed, simulation time t_{finish} is recorded and t_{max} is used in case action has failed.

$$U_1^* = \frac{1}{n}\sum_{i=1}^{n} U_{1,i} \tag{6}$$

$$U_2^* = \frac{1}{n}\sum_{i=1}^{n} \left(\frac{t_{\text{finish},i}}{t_{\text{max}}}\right) \tag{7}$$

$$U_3^* = \frac{n_{\text{finish}}}{n} \tag{8}$$

$$R^* = \frac{n_{\text{emerg}}}{n} \tag{9}$$

Besides a "successful" maneuver, a fall-back or emergency maneuver can occur as well. The risk R^* can be approximated by eq. 9 where n_{emerg} denotes the number of simulations where the emergency maneuver of ego vehicle is triggered.

The biggest strength of MC simulation is that, unlike POMDP which usually builds search trees and needs many computational resources, each of the single MC simulations is independent of others and thus can be well parallelized in a multi-core system. In this way, this method can be run in real-time without problems and can be even more efficient with customized hardware. The accuracy of the feature approximation from MC simulation increases with the number of simulations. We tested the variance of the fall-back rate of selecting one certain gap in an example merging scenario related to the number of MC simulations. MC simulations with 100, 500, and 1000 repetitions will produce 0.09, 0.018, and 0.017 variances on the fall-back ratios, and the run-time on an 8-core laptop with 8 threads are 10ms, 50ms, and 100ms respectively. We take 500 repetitions as a good balance of run-time and accuracy.

The precondition of this approach is a well sensed and estimated probabilistic environment, which is discussed in the next chapter.

B. Basic Behaviors and Their Estimation

The input for the MC simulation is the perceived and estimated environment. We formulate several basic behaviors for highway driving and afterwards introduce the estimation of each behavior for MC simulation. Note that for all the behavior models, trucks will have a different parameter set as normal vehicles, e.g. they behave with less acceleration, are less prone to yield to merging attempts, and are less possible to perform lane change.

The basic behaviors that model the environment vehicles are required to be as simple as possible since the environment needs to be propagated in a large number of MC simulations. Too complicated behavior models demand too much computational resources that could slow down the MC simulation.

1) Follow lane behavior with yielding capability: For car following, the customized IDM can follow more than one leading vehicle. However, in usual highway driving, a more intelligent follow-lane behavior is required, which is able to behave politely for merging vehicles or other vehicles that show clear lane change desire (e.g. via indicator), and we call it extended IDM (E-IDM). As soon as a cut-in desire from another vehicle is detected, it computes a yielding motivation value m by a logistic regression function

$$m = \frac{1}{1 + e^{-\theta_{\text{Y}}^T f_{\text{Y}}}} \tag{10}$$

with the θ_{Y} to be the weight vector and $f_{\text{Y}} = [d, t_{\text{TH}}, \dot{t}_{\text{TH}}]$ to be the feature vector, where d denotes the distance between the ego vehicle and the merging vehicle, $t_{\text{TH}} = \frac{d}{v_{\text{main}}}$ is the time headway to the merging vehicle and $\dot{t}_{\text{TH}} = \frac{v_{\text{main}} - v_{\text{merge}}}{v_{\text{main}}}$

is the changing rate of time headway. v_{main} and v_{merge} are the velocities of the ego vehicle and the merging vehicle. The logistic regression function is trained with the Interaction dataset and ExitD dataset where in total 3320 vehicles are recorded to yield to a merging vehicle and 432 vehicles not. One threshold value $m_{\text{th}} = 0.5$ is introduced to control the willingness of yielding. If the vehicle decides to yield ($m > m_{\text{th}}$), it treats both the merging vehicle and the preceding vehicle in the current lane as target vehicles.

2) Estimation of E-IDM: For car following, the surrounding vehicles are assumed to obey the customized IDM model. Thus, the parameters for the customized IDM model need to be estimated either from experience or from the perception and tracking of the surrounding vehicles. If the ego vehicle intends to cut in or merge, the yielding intention of the relevant vehicles needs to be estimated as well.

For the customized IDM, we adopt the recommended IDM parameters estimated from HighD dataset in [29]. In order to simulate the estimation error, the parameters will be added with random noise in different repetitions of MC simulations. For acquiring a more accurate estimation of the parameters, the long-term observation of each vehicle is utilized as well, as long as the perception module is capable of. For example, if an agent is in a stationary car-following scenario longer than 3s, with an overall velocity variation of less than $1.5\frac{\text{m}}{\text{s}}$ and a time headway variation of less than 0.2s, we update the desired time headway with the observed one. Similarly, if a vehicle keeps its velocity (velocity variation of less than $1.5\frac{\text{m}}{\text{s}}$) longer than 3s, where the time headway to the front vehicle is larger than 3s, it will be used as the desired velocity. This can help recognize different types of drivers, e.g. drivers who prefer higher velocity and are not rule-compliant, and who prefer much lower velocity than the speed limit.

Estimating the yielding intention is similar to computing the yielding motivation. We use the same logistic function

$$P(\text{yielding}|\hat{f}_{\text{Y}}) = \frac{1}{1+e^{-\hat{\theta}_{\text{Y}}^T \hat{f}_{\text{Y}}}} \tag{11}$$

with a different feature vector $\hat{f}_{\text{Y}} = [a, d, t_{\text{TH}}, \dot{t}_{\text{TH}}]$, where another feature a is included, which is the acceleration of the vehicle whose yielding intention is estimated. The other features stay unchanged. The $\hat{\theta}_{\text{Y}}$ vector is retrained with the same data as in training the yielding behavior.

In MC simulation, the vehicle is expected to yield to a merge or cut-in attempt when $P(\text{yielding}|\hat{f}_{\text{Y}})$ is bigger than the same threshold 0.5, by treating it as one of the target vehicles in its customized IDM model.

3) Lane change behaviors and estimation of lane change intention: Vehicles driving on a multi-lane highway might change lanes to gain efficiency or to bring convenience to others. The Minimizing Overall Braking Induced by Lane changes (MOBIL) strategy [30] is used as the basic lane change behavior. This model makes lane change decisions with the goal of maximizing the acceleration of all the involved vehicles. The IDM model is used to compute the accelerations of surrounding vehicles. Then, a lane change is performed if

$$\tilde{a_{\text{e}}} - a_{\text{e}} + p((\tilde{a_{\text{n}}} - a_{\text{n}}) + (\tilde{a_{\text{o}}} - a_{\text{o}})) > a_{\text{th}} \tag{12}$$

where a_{e}, a_{n} and a_{o} are the accelerations of the ego vehicle, the following vehicle on the target lane, and the following vehicle on the source lane, if no lane change of the ego vehicle is performed. Correspondingly, the ones with tildes are their accelerations if the ego vehicle change lanes. The politeness factor p is included to control how much the acceleration gains and losses of other vehicles are valued. The left side of eq. 12 represents the overall acceleration gain a_{gain} which should be bigger than a threshold value a_{th} for a lane change. Note that if lane change is possible both to the left and right, it will be done in the direction where eq. 12 fulfills and whose a_{gain} is higher.

The MOBIL model can be combined with the E-IDM in order to be able to yield to cut-in attempts, which we call deterministic extended MOBIL (DE-MOBIL). If the vehicle has high motivation to yield ($m > m_{\text{th}}$), a_{e} will be computed regarding the vehicle with cut-in desire as front vehicle. Depending on eq. 12, it either does a lane change or decelerates.

In order to estimate the MOBIL model, the parameters p and a_{th} are calibrated from HighD datasets where 12380 lane changes are included. The MOBIL model has the highest accuracy over all the lane changes when $p = 0.9$ and $a_{\text{th}} = 0.5$ m/s^2. The estimation of DE-MOBIL is separated into two steps. Firstly, the E-IDM is estimated via the method in the previous section. In order to better predict and simulate the scene in MC simulation, the lane change probability of the vehicles on the main lanes should be estimated at each simulation step. A probabilistic lane change behavior can be induced from the DE-MOBIL, which is referred to as probabilistic extended MOBIL (PE-MOBIL). We define the net acceleration gain for keeping lane to be $a_{\text{k}} = 0$, and for changing lane to left and right to be $a_{\text{l}} = a_{\text{gain,l}} - a_{\text{th}}$ and $a_{\text{r}} = a_{\text{gain,r}} - a_{\text{th}}$. The probability mass of each option is

$$m_{\text{MOBIL}}(i) = \frac{e^{a_{\text{i}}}}{\sum_{j\in\{k,l,r\}} e^{a_{\text{j}}}}, \text{for } i \in \{k, l, r\} \tag{13}$$

The estimate of DE-MOBIL is also called probabilistic extended MOBIL (PE-MOBIL). When using PE-MOBIL only as a probabilistic policy, the lane change decisions can be output by sampling from the probability mass $m_{\text{MOBIL}}(i)$. However, when using it as an estimation, the movement of the vehicle is another important source of information showing whether it intends to do a lane change, besides estimating the lane change probability by computing the motivation of a lane change. We learn another logistic regression model for computing lane change probability mass m_{move} using the movements (signed lateral distance to the centerline d_{c} and the lateral velocity v_{lat} of the target vehicle) as the features.

Finally, the probabilities from two sources will be combined by using the Mixing Rule of Evidence Theory [31]

$$m(i) = w_1 m_{\text{MOBIL}}(i) + w_2 m_{\text{move}}(i) \tag{14}$$

with w_1 and w_2 to be the weighting of two information sources equally set to 0.5.

4) Merging behavior and its estimation: A heuristic merging behavior mentioned in [32] is considered. The gap is assumed to move with constant velocity and the merging vehicle

tries to catch the gap either by accelerating or decelerating constantly. We call it Closest Gap Merging Policy (CGMP).

The CGMP can be estimated probabilistically as well. First, the time that is estimated to approach i-th gap is computed to be t_i. The softmax function is again applied to generate the probability for approaching each gap using eq. 15. Merging decision will be evaluated tactically with 1s interval in MC simulation.

$$P(i) = \frac{e^{-t_i}}{\sum_{j=1}^{n} e^{-t_j}}, \text{for } i \in \{1, ..., n\} \tag{15}$$

C. Learning Decision via Behavior Cloning

In section III-B, the probabilistic environment is established, in which vehicles yield and change lanes following our behavior model. They react to changes in their surroundings. For the autonomous vehicle, the features for trying each action are obtained from the MC simulations by interacting with a probabilistic world. However, making decisions from these features is not straightforward. In order to be understandable to passengers and other human drivers, autonomous vehicles should behave as close to human drivers as possible, e.g. they should not be overly egoistic (weight utility and comfort too much), overly cautious (weight risk too much) or overly courteous (weight politeness too much). Therefore, the weighting of each feature should be best learned from trajectories of real traffic. We pursue a mapping from features of all actions to the action index, which is similar to a classification problem. We do not intend to use neural networks for this job, as we want to explicitly show the weighting for each feature in different scenarios to better understand the decision and prevent overfitting the limited data. Therefore, we utilize linear logistic regression to solve the problem. For each action a_i or a_{gap_i}, the regression function outputs one quality value q_i by

$$q_i = \frac{1}{1 + e^{-\theta_{\text{P}}^T f_i}} \tag{16}$$

where θ_{P} is the learned weight vector for a policy. $f_i = [U_1^*(a_i), U_2^*(a_i), U_3^*(a_i), C_1^*(a_i), R^*(a_i), P_1^*(a_i), P_2^*(a_i)]$ is the feature vector generated from MC simulation for action a_i. One training data is prepared as a pair of features $[f_1, f_2, ..., f_n]$ and a label i_{GT} that associates with the ground truth action humans performed. The goal is to optimize θ_{P} such that the quality value for the ground truth action $q_{i_{\text{GT}}}$ close to 1 and for other actions close to 0. For training, the cross-entropy loss is applied. During inference, the action with the highest q_i will be selected.

1) Learning merging and lane change behavior: We decide to separate the merging behavior from the lane change behavior, because humans might weight features differently in these two scenarios. Obtaining ground-truth action for merging scenarios is straightforward, as the gap that is accomplished in the end is assumed to be the ground-truth of the gap that is initially approached. For lane change, as we already know whether a lane change occurs in the trajectory, we assume a lane change decision is made when the lateral velocity is higher than $0.25\frac{\text{m}}{\text{s}}$ to the target direction. This value is larger than 98.1% of the lateral velocities in trajectories with no lane change. The lane change decision will continue until the ego vehicle is physically on another lane. For other frames, we assume the decision to be keeping lane a_1.

TABLE III: Learned weights for merging and lane change behavior.

	Utility			Comfort	Risk	Politeness	
	U_1^*	U_2^*	U_3^*	C_1^*	R^*	P_1^*	P_2^*
θ_{merge}	**0.5**	0.05	**-1.0**	0.05	**-0.7**	0.1	0.15
θ_{lc}	0.183	**0.3**	-0.15	0.1	**-0.367**	**1.0**	0.25

Note that we first discard the action candidates a_2 and a_3 during training and include them during inference, because the trajectories of a_1, a_2 and a_3 are similar, and matching the recorded trajectories to one of them becomes ambiguous. The training data is extracted from the datasets tactically with 1s interval. From HighD and ExitD datasets, we obtain 23154 valid[4] training frames for merging scenario and 253331 valid training frames for lane change (among which 55852 frames have lane change labels). They are split into 75% training set and 25% test set. Finally, the weight vectors for merging θ_{merge} and for lane change θ_{lc} are presented in Table III, where the weights are normalized between -1 and 1.

It is noticeable that human drivers prefer gaps that are prone to success and have less chance to fall back in merging. In free lane change, they care more about not impeding other vehicles.

Note that the lanes in ExitD dataset are not straight. Therefore, we perform our approach in Frenet-Frame w.r.t. the road centerline of the vehicle that is regarded as ego. Examples in the later chapters present the capability of our approach working on different road geometries.

The validation accuracy on the test set is 79.5% for lane change and 94.4% for merging. One explanation for the relatively higher accuracy for merging is that, in real traffic, human drivers have higher variance doing lane changes depending on which type of driver they are, but will usually pursue the first gap they see on the target lane for merging.

2) Risk bounded merging behavior: For evaluation, we include another merging behavior where we do not allow arbitrarily high fall-back rate R^* (risk). Same as before, the action with the highest quality value q_i is selected, but after the ones with higher than the risk threshold R^*_{th} are discarded. In case of the risk for all actions is higher than R^*_{th}, merging to the very last gap becomes the decision. We set the threshold to be $R^*_{\text{th}} = 0.2$, which is higher than the risk of 3.9% of all the merging decisions from humans.

IV. Evaluation

As highway scenarios are often associated with high velocities and high risk, it is better to have a thorough evaluation in simulation before putting the behaviors on a test vehicle. We developed a flexible and modular simulation environment for highway scenarios and evaluated our behaviors there. Firstly, the simulation environment will be shortly explained. Then we will evaluate the behaviors in some specific challenging scenarios as well as on massively generated random traffics.

[4]The first and last two seconds of each trajectory are abandoned as the vehicle is at the border of the field-of-view of the drone. Frames where no vehicle is on the target lane for merging are regarded as invalid as well.

A. Simulation Environment

Some existing simulators (like SUMO [33]) are able to simulate highway scenarios, but do not support customized behavior for other agents. BARK [34] is another benchmark for behavior evaluation, but it is not specifically designed for highway scenarios, and is not easy to configure for our purpose. Therefore, we developed our customized simulation by imitating the concept of BARK. The simulation allows a manual design of the road network (arbitrary number of main lanes, merging lanes and exit lanes) with arbitrary parameters (shape, width, length, speed limit, etc.). On each lane, agents with any manual designed behaviors (IDM behavior, MOBIL lane change behavior, learned lane change behavior, CGMP, learned merging behavior, etc.) with arbitrary parameters (IDM parameter, RSS parameter, yielding parameter, MOBIL parameter, etc.) can be initiated. After running the simulation, each agent is able to sense its surrounding environment with a predefined range and move with its customized behavior. In this way, agents with extreme behavior can be simulated, e.g. with non-realistic IDM parameters where the desired time headway is only 0.3s or RSS parameter with $a_{\text{max,decel}} = -0.5\frac{\text{m}}{\text{s}^2}$. Furthermore, including more than one agent with our learned lane change or merging policy is possible as well. To best imitate the real traffic, vehicles on the most right main lane and the merging lane are partly (30%) initialized as trucks, which have larger geometry and different behavioral parameters as normal vehicles. Other vehicles are assumed to be able to perceive the geometry of the trucks and have a different estimation of their behavior models. The simulation is equipped with proper visualization and the history of all the agents can be recalled. The validity of the simulation is checked visually in a large part of the scenarios in the datasets before being applied in our evaluation.

For evaluation, one or two autonomous agents are initialized with our learned behavior, and the other agents should have different behaviors depending on the lanes that they locate. We apply the PE-MOBIL policy for the agents on the main lanes, and probabilistic CGMP for agents on the merging lane. Note that the parameters of these non-autonomous agents in the simulator are randomized and not known to the autonomous agents. The autonomous agents can only estimate their intention and parameters (explained in Section III-B) and initiate the MC simulation. In this way, error of estimation is introduced as estimating the real world.

B. Evaluation for On-Ramp Merging

We compare three policies for merging scenarios: the CGMP, the learned merging policy (LMP), the risk bounded merging policy (RBMP).

1) Challenging scenarios: The biggest challenge for merging in dense traffic is to recognize the cooperative intention of the vehicles on the other lane and select the proper gap. We present one example in Fig. 3 where the two blue agents merge with CGMP, LMP and RBMP respectively. Fig. 3(a) presents three moments of driving with CGMP. The LMP and RBMP generate a similar behavior which is shown by Fig. 3(b). As can be seen in Fig. 3(a), the first merging vehicle with the CGMP insists on merging in front of the red agent and finally has to fall back and stop, because it is not able to estimate the yielding intention of the red agent. However, with LMP and RBMP, the vehicles are able to finish merging safely (Fig. 3(b)). By analyzing the feature values of a_{gap_1} (merging before the red vehicle) and a_{gap_2} (merging after the red vehicle) with Fig. 3(c), we observe that a_{gap_1} is at the beginning with around 86% success rate U_3^* and 12% fall-back rate R^*. However, as the intention of the red vehicle becomes more clear and the merging lane is closer to the end, the fall-back rate is increasing. The decision finally switches to a_{gap_2} which is all the time safer.

TABLE IV: Statistics for merging on random traffic with different merging policies. (avg. = average).

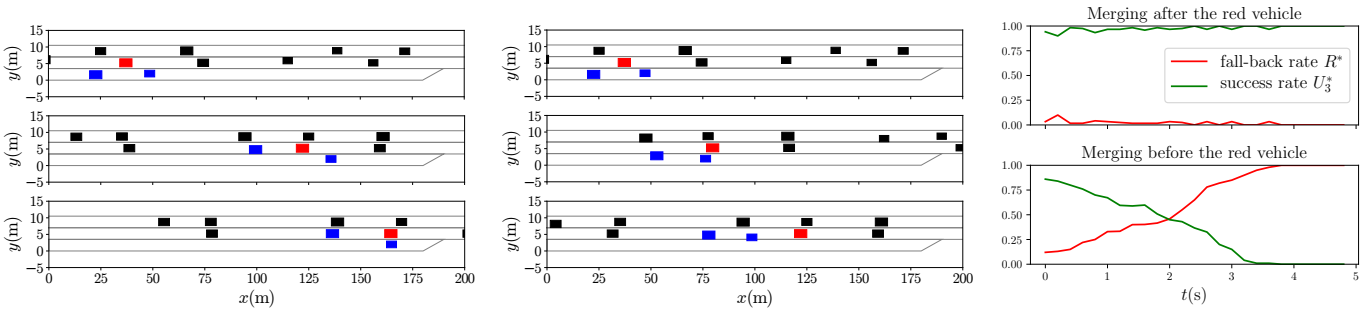
		avg. merging time (s)	$n_{\text{fall-back}}$
$t_{\text{HW}} \sim \mathcal{U}(0.8, 1.4)s$	CGMP	8.142	181
	LMP	**6.212**	22
	RBMP	6.582	**11**
$t_{\text{HW}} \sim \mathcal{U}(1.2, 2.0)s$	CGMP	5.314	124
	LMP	**4.482**	19
	RBMP	4.591	**9**

2) Evaluation on random traffic: We generate 500 random scenes with two main lanes on the left side and one merging lane on the right side. The lanes are attached randomly with one of the three speed limits 60 kph, 80 kph and 100 kph, which results in different lengths of the merging lane. Random agents are generated on the main lanes with two different densities, which are represented by the time headway between vehicles that follow two uniform distributions $\mathcal{U}(0.8, 1.4)$s and $\mathcal{U}(1.2, 2.0)$s. To best simulate a real traffic scene, the agents are attached with randomized size, states, internal parameters of their behavior model, etc. Two vehicles with 1s time headway will be initialized on the merging lane with the same merging policy. For each of the three policies, the evaluation will be done once through the exact same 500 initial scenes to make the results comparable. However, how the scene evolves will be affected by the selected policy. Finally, some statistics are summarized in Table IV after all 1000 merges.

Apparently, merging in denser traffic leads to more fall-backs and more merging time. It is also noticeable that the LMP and RBMP allow significantly faster merging and lead to much fewer fall-backs than the baseline policy CGMP. By bounding the risk (RBMP), the merges are slightly slower but produce fewer fall-backs than LMP. It is difficult to judge which one between LMP and RBMP is better as it raises the topic of which level of risk humans accept and how the user would sacrifice efficiency to feel safer. The authors do not conclude on that but leave the decisions for users.

C. Evaluation for Highway Exit

We regard the exit behavior as a similar behavior of merging, where one target lane exists and the lane change intention is indicated early, such that other vehicles have the chance to yield and open the gap. Therefore, all the three merging policies (CGMP, LMP and RBMP) are suitable for exiting and are evaluated in random traffic as well. However, more than one merge can be needed if the ego vehicle is on the most left



(a) t=1s, 5s, 7s (b) t=1s, 3s, 5s (c)

Fig. 3: An example of merging scenario where the blue rectangles are merging agents, and others are surrounding agents. (a) Merging agents follow CGMP. The first one has to fall back by trying merging in front of the red agent. (b) Merging agents follow LMP and RBMP and finish merging successfully by merging after the red agent. (c) The fall-back rate and success rate of two candidate actions for the first merging agent in (b) at different time steps.

TABLE V: Statistics for exiting on 1000 random traffics with different merging policies and starting positions. (avg. = average, acc. = accomplish).

		avg. acc. time (s)	$n_{\text{fall-back}}$
200m before exit ends	CGMP	12.145	73
	LMP	**11.905**	37
	RBMP	11.935	**36**
500m before exit ends	CGMP	11.857	24
	LMP	**11.482**	5
	RBMP	11.591	**3**

lane, but the merging policy can be executed subsequently. The only modification for using LMP and RBMP for exiting is that, when building the MC simulation, the ego lane is assumed to end shortly[5], and maximumly until the end of the exit opening. With this heuristic, the ego vehicle will not always pursue the very first gap that is perceived because it produces a certain fall-back rate R^*.

The map is built with three main lanes and one exit lane on the right side. Random vehicles will be generated on the map with the density of $t_{\text{HW}} \sim \mathcal{U}(1.2, 1.8)$s. We initiate the ego vehicle on the most left lane at two distances 200m and 500m before the exit ends, each with 1000 simulations. Note that the exiting is regarded as accomplished when the ego vehicle is able to locate at the exit lane before the exit opening ends. Table V provides the simulation results.

The same pattern as in merging can be observed, that LMP and RBMP are in general better than CGMP. By bounding the risk (RBMP), the exits are slightly slower but produce fewer fall-backs as LMP. The earlier the ego vehicle starts merging to the right, the less risky the exit will be. Therefore, it is recommended to start exiting with sufficient reserve if the route is known beforehand from the map.

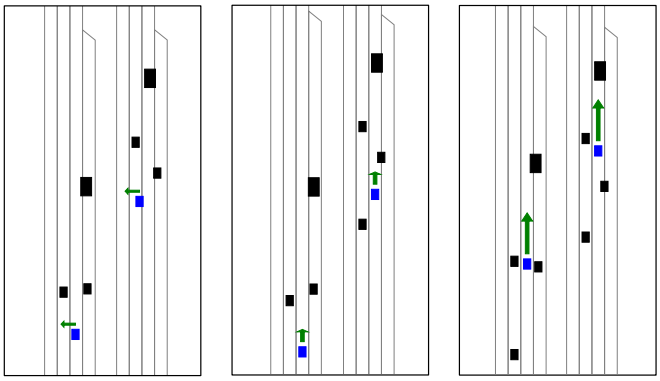
D. Evaluation for Free Lane Change

Highway driving is tackled by Adaptive Cruise Control (ACC) systems since decades, which helps follow the leading vehicle safely and turn the wheels smoothly. However, the goal of highway driving should not only be safe follow driving, but achieving maximum efficiency while maintaining safety, which is not a simple task. Lane changes, proper acceleration and deceleration should be performed at the proper time to gain efficiency, while preventing potential risky situations and not affecting others negatively. Evaluations are firstly done on some challenging scenarios and afterward on random traffic. Three policies are compared, the E-IDM, the DE-MOBIL and our learned lane change policy (LLCP).

1) Challenging scenarios: Typical free lane changes are performed when the ego vehicle desires higher speed but is blocked by the slow-driving vehicle in front, and at the same time, the other lanes are free. We do not focus on this basic scenario but more challenging ones, where some of them can be covered by MOBIL as well but some not. Fig. 4 illustrates three scenarios where the autonomous vehicles controlled by the LLCP perform different courteous behavior to let the merging vehicle cut in easier. In Fig. 4(a), a lane change to the left is possible by which the velocity of the ego vehicle is damaged at least. In scenarios where a lane change is not possible (Fig. 4(b) and Fig. 4(c)), it performs either a deceleration a_2 or an acceleration a_3 depending on the situations to allow a smoother merging. The DE-MOBIL can perform a deceleration or a lane change as well depending on the overall acceleration gain, assuming the merging desire are known to the ego vehicles. However, the drawbacks of DE-MOBIL are obvious. Firstly, it can not perform acceleration when needed. Furthermore, it focuses only on a few vehicles of the traffic but ignores others that can potentially influence the MOBIL set-up.

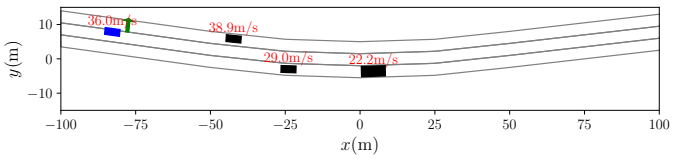
We further evaluated the LLCP on some other challenging scenarios which the E-IDM and DE-MOBIL can not tackle at the conceptual level. In Fig. 5(a), the ego vehicle with LLCP does a lane change to the left a_4 to prevent a potentially risky situation where the vehicle behind the truck might suddenly perform an overtake without prior indication. The probability of this happening is not low as the vehicle is approaching the truck with high relative velocity. We perform MC simulations for keeping lane a_1 and changing lane to left a_4 for the current moment. The R^* and U_1^* of a_1 are 0.11 and 0.79, but a_4 produces 0 and 0.86 respectively and is therefore safer and faster. Without receiving an explicit lane change indication

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

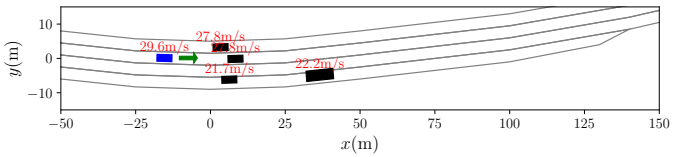


(a) Lane change a_4 for merging vehicles (b) Decelerate a_2 for merging vehicles (c) Accelerate a_3 for merging vehicles

Fig. 4: The autonomous agent (blue rectangles) with LLCP performs different courteous behavior (represented by green arrows) to enable a smoother merging of other merging vehicles.



(a) A forsighted lane change



(b) Decelerate instead of changing lane to right

Fig. 5: Two challenging scenarios that can be tackled by LLCP but not DE-MOBIL. 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 the vehicle behind the truck, the DE-MOBIL can not perform this foresighted behavior. Fig. 5(b) demonstrates another scenario where the ego vehicle is blocked by a slow driving vehicle in front. Another slow-moving vehicle exits on the left lane as well which makes a lane change to the left not beneficial. The DE-MOBIL will output a lane change to the right where the ego vehicle is not blocked by any other vehicles. However, this is a potentially risky decision as two merging vehicles on the merging lane could finish merging at any time. If so, the right lane becomes crowded and the ego vehicle needs to brake more. The MC simulation outputs $R^*(a_5) = 0.34, U_1^*(a_5) = 0.68$ for changing lane to right a_5 and $R^*(a_2) = 0, U_1^*(a_2) = 0.77$ for deceleration a_2. As a result, a_2 is a better option according to LLCP.

2) Evaluation on random traffic: We evaluate the three policies in random simulated traffics as well. There are some differences to the random traffics for merging policies. The map contains three main lanes and one merging lane. Two vehicles will be initiated on the merging lane driving with probabilistic CGMP. On the main lanes, vehicles will be generated with random time headway $t_{\text{HW}} \sim \mathcal{U}(1.4, 2.2)$s. Only one of the vehicles on the main lanes is randomly defined as an autonomous vehicle and drives with the LLCP, which will be substituted by E-IDM and DE-MOBIL for the same initial scene. In total 1500 scenes are generated and will be simulated twice. In the first round, all the vehicles have recommended parameters with small randomness. The second round is more challenging, where 20% of the vehicles are assigned with inappropriate IDM and RSS parameters (desired time headway $T_{\text{d}} = 0.3$s and the maximum deceleration of others $a_{\text{max,decel}} = -0.5\frac{\text{m}}{\text{s}^2}$), which results in close-to-crash lane changes and merges. If so, the vehicle behind has to execute an emergency brake (fall-back).

Table VI presents statistical results. In total, LLCP generates much less lane changes than DE-MOBIL, but still provides overall higher U_1, C_1, P_1 and P_2. If we consider lateral acceleration in comfort measure as well, LLCP will be even more comfortable and stable. Another noticeable highlight of the LLCP is that it produces significantly safer driving behavior with much fewer fall-backs. In the first round where others drive safely, the E-IDM has reasonable 0 fall-backs because others do not do unsafe lane change in front of the autonomous vehicle. However, the DE-MOBIL has strangely 11 fall-backs. We reproduced the simulations and discovered the reason. At the time where the ego vehicle tends to change lane to another lane, another vehicle on the third lane starts a lane change towards the same lane as well. The lane changes from both vehicles are initially safe, but as soon as they appear on the target lane at the same time, the one behind becomes unsafe if they are too close, which results in an emergency brake. In this case, it is ambiguous which vehicle to blame. Note that for our free lane change evaluation, we apply the RSS safety rule without extension in Section II-C2 as recommended. However, this edge case is not covered by it and could initiate further investigations, but is not the scope of this paper. However, our LLCP can prevent this risky situation and has 0 fall-backs, because it recognizes the lane change intention of others and can abort their lane change in the early stage. In the second round with aggressive vehicles, it appears that even keeping lane can be sometimes risky as well, because unsafe cut-ins from others can not be prevented. However, with our LLCP, the vehicle can change to other safer lanes or decelerate a_2 as soon as the intention of the aggressive cut-ins is recognized.

V. Conclusions And Outlook

In this work, we proposed a behavior cloning concept for learning high-level decisions from recorded trajectories of real traffic, unlike most previous works that focus on end-to-end behavior cloning for controlling. We summarized and gave a clear definition of the main features that affect how humans make driving decisions. The features are acquired via MC simulation, which receives the uncertain states and estimates of the driver models from surrounding agents as inputs. Two important goals of this work are on one side producing human-like behavior, on the other side making the decision understandable and transparent to humans. Thus, we adopt one logistic function to output the final decision

TABLE VI: Statistics for evaluation on random traffic with different lane change policies. (avg. = average)

		number of lane changes	$n_{\text{fall-back}}$	avg. U_1 of ego	avg. C_1 of ego	avg. P_1	avg. P_2
0% of agressive vehicles	E-IDM	0	**0**	0.901	0.828	0.683	0.422
	DE-MOBIL	1220	11	0.905	0.809	0.684	0.426
	LLCP	158	**0**	**0.911**	**0.831**	**0.687**	**0.429**
20% of agressive vehicles	E-IDM	0	26	0.896	0.815	0.69	0.417
	DE-MOBIL	1288	23	0.901	0.799	0.691	0.417
	LLCP	199	**6**	**0.907**	**0.818**	**0.693**	**0.425**

and recover the weights of all features using the real data. The validation accuracy using the test data shows a human-close decision. On the other side, tracing back the decision is not complicated. At undesired decisions, developers can either check the MC simulations to see whether the estimated environment and resulting features make sense, or inspect the logistic function to examine whether the weighting is inappropriate. Furthermore, the learned policy is not overfitting to the limited training data but generalizes to multi-lane scenarios with arbitrary speed limits and traffic density, which is strengthened by the successful application of merging policy on exiting. Evaluation results from simulated random traffic demonstrate the superiority of our approach over the rule-based baseline policies, e.g. an overall better performance on safety, efficiency and politeness, even in scenarios with abnormal driving behavior from other vehicles.

Another highlight of this approach is that the design of the pipeline is highly modular. The estimation of the environment can be substituted by other prediction modules that are able to estimate the behavior of other agents given the ego vehicle's action. As the output of our module is the high-level decision, i.e. the gap to merge, any low-level trajectory planner can be adopted since the constraints for the planner are clearly given.

For lane change and merging, we extend the RSS safety concept with additional assumptions and evaluate the feasibility of this safety rule in real traffic data. The fewer violations show that the proposed rule is not overly cautious. Therefore, we propose to integrate these assumptions as additional common sense into the RSS safety concept.

Due to the limited length of this paper, we did not comprehensively evaluate on how the weighting of each feature affects the driving style. Changing the weighting around the learned values is expected to affect the driving style, e.g. pursue faster gap but more prone to fall-back by weighting the utility more and the risk less. After having a thorough understanding of the weighting, the possibility of providing several pre-defined driving styles or even tuning them online can be provided to the user for different preferences. Furthermore, we would like to implement another leaning-based approach, e.g. RL-based, and compare with our approach.

In the future, we plan to apply the same concept for inner-city scenarios and generate high-level decisions according to the same features. More uncertainties should be taken into account, e.g. uncertain decision of vehicle crossing intersection or doing a turn, uncertain existence of objects in occlusion, etc. Furthermore, edge cases for RSS safety concept need to be exhaustively analyzed and it needs to be extended to cover the inner-city scenarios as well, which is already discussed by some previous works, e.g. [35]. The final goal is to generate high-level decisions for traversing intersections, zebra crossing, etc. efficiently, safely, and when needed politely.

REFERENCES

[1] B. R. Kiran, I. Sobh, V. Talpaert, P. Mannion, A. A. A. Sallab, S. K. Yogamani, and P. Pérez, "Deep reinforcement learning for autonomous driving: A survey," *CoRR*, vol. abs/2002.00444, 2020. [Online]. Available: https://arxiv.org/abs/2002.00444

[2] D. Kamran, C. F. Lopez, M. Lauer, and C. Stiller, "Risk-aware high-level decisions for automated driving at occluded intersections with reinforcement learning," *CoRR*, vol. abs/2004.04450, 2020. [Online]. Available: https://arxiv.org/abs/2004.04450

[3] D. Kamran, T. Engelgeh, M. Busch, J. Fischer, and C. Stiller, "Minimizing safety interference for safe and comfortable automated driving with distributional reinforcement learning," *CoRR*, vol. abs/2107.07316, 2021. [Online]. Available: https://arxiv.org/abs/2107.07316

[4] W. Dabney, G. Ostrovski, D. Silver, and R. Munos, "Implicit quantile networks for distributional reinforcement learning," *CoRR*, vol. abs/1806.06923, 2018. [Online]. Available: http://arxiv.org/abs/1806.06923

[5] S. Sharifzadeh, I. Chiotellis, R. Triebel, and D. Cremers, "Learning to drive using inverse reinforcement learning and deep q-networks," *CoRR*, vol. abs/1612.03653, 2016. [Online]. Available: http://arxiv.org/abs/1612.03653

[6] C. You, J. Lu, D. Filev, and P. Tsiotras, "Advanced planning for autonomous vehicles using reinforcement learning and deep inverse reinforcement learning," *Robotics and Autonomous Systems*, vol. 114, pp. 1–18, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0921889018302021

[7] M. Wulfmeier, D. Rao, D. Z. Wang, P. Ondruska, and I. Posner, "Large-scale cost function learning for path planning using deep inverse reinforcement learning," *The International Journal of Robotics Research*, vol. 36, no. 10, pp. 1073–1087, 2017. [Online]. Available: https://doi.org/10.1177/0278364917722396

[8] J. Ho and S. Ermon, "Generative adversarial imitation learning," *CoRR*, vol. abs/1606.03476, 2016. [Online]. Available: http://arxiv.org/abs/1606.03476

[9] J. Fu, K. Luo, and S. Levine, "Learning robust rewards with adversarial inverse reinforcement learning," *CoRR*, vol. abs/1710.11248, 2017. [Online]. Available: http://arxiv.org/abs/1710.11248

[10] F. Codevilla, E. Santana, A. M. Lopez, and A. Gaidon, "Exploring the limitations of behavior cloning for autonomous driving," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[11] W. Farag and Z. Saleh, "Behavior cloning for autonomous driving using convolutional neural networks," in *2018 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT)*, 2018, pp. 1–7.

[12] F. Poggenhans, J.-H. Pauls, J. Janosovits, S. Orf, M. Naumann, F. Kuhnt, and M. Mayr, "Lanelet2: A high-definition map framework for the future of automated driving," 11 2018, pp. 1672–1679.

[13] W. Zhan, C. Liu, C. Chan, and M. Tomizuka, "A non-conservatively defensive strategy for urban autonomous driving," in *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, Nov 2016, pp. 459–464.

[14] C. Hubmann, N. Quetschlich, J. Schulz, J. Bernhard, D. Althoff, and C. Stiller, "A pomdp maneuver planner for occlusions in urban scenarios," in *2019 IEEE Intelligent Vehicles Symposium (IV)*, 2019, pp. 2172–2179.

[15] R. Krajewski, J. Bock, L. Kloeker, and L. Eckstein, "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.

[16] W. Zhan, L. Sun, D. Wang, H. Shi, A. Clausse, M. Naumann, J. Kümmerle, H. Königshof, C. Stiller, A. de La Fortelle, and M. Tomizuka, "INTERACTION dataset: An international, adversarial and cooperative motion dataset in interactive driving scenarios with semantic maps," *CoRR*, vol. abs/1910.03088, 2019. [Online]. Available: http://arxiv.org/abs/1910.03088

[17] S. Shalev-Shwartz, S. Shammah, and A. Shashua, "On a formal model of safe and scalable self-driving cars," *CoRR*, vol. abs/1708.06374, 2017. [Online]. Available: http://arxiv.org/abs/1708.06374

[18] M. Naumann, L. Sun, W. Zhan, and M. Tomizuka, "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.

[19] M. Naumann and C. Stiller, "Towards cooperative motion planning for automated vehicles in mixed traffic," *CoRR*, vol. abs/1708.06962, 2017. [Online]. Available: http://arxiv.org/abs/1708.06962

[20] L. Wang, C. F. Lopez, and C. Stiller, "Realistic single-shot and long-term collision risk for a human-style safer driving," in *2020 IEEE Intelligent Vehicles Symposium (IV)*, 2020, pp. 2073–2080.

[21] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," *Physical Review E*, vol. 62, no. 2, p. 1805–1824, Aug 2000. [Online]. Available: http://dx.doi.org/10.1103/PhysRevE.62.1805

[22] S. Albeaik, A. Bayen, M. T. Chiri, X. Gong, A. Hayat, N. Kardous, A. Keimer, S. T. McQuade, B. Piccoli, and Y. You, "Limitations and improvements of the intelligent driver model (idm)," 2021.

[23] C. Hubmann, J. Schulz, G. Xu, D. Althoff, and C. Stiller, "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.

[24] M. Naumann, F. Wirth, F. Oboril, K. Scholl, M. S. Elli, I. Alvarez, J. Weast, and C. Stiller, "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.

[25] M. Naumann, "Probabilistic motion planning for automated vehicles," Ph.D. dissertation, Karlsruher Institut für Technologie (KIT), 2020.

[26] L. L. Hoberock, "A Survey of Longitudinal Acceleration Comfort Studies in Ground Transportation Vehicles," *Journal of Dynamic Systems, Measurement, and Control*, vol. 99, no. 2, pp. 76–84, 06 1977. [Online]. Available: https://doi.org/10.1115/1.3427093

[27] X. Xu, X. Wang, X. Wu, O. Hassanin, and C. Chai, "Calibration and evaluation of the responsibility-sensitive safety model of autonomous car-following maneuvers using naturalistic driving study data," *Transportation Research Part C: Emerging Technologies*, vol. 123, p. 102988, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0968090X21000231

[28] G. Markkula, J. Engström, J. Lodin, J. Bärgman, and T. Victor, "A farewell to brake reaction times? kinematics-dependent brake response in naturalistic rear-end emergencies," *Accident Analysis Prevention*, vol. 95, pp. 209–226, 2016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0001457516302366

[29] R. P. Bhattacharyya, R. Senanayake, K. Brown, and M. J. Kochenderfer, "Online parameter estimation for human driver behavior prediction," *CoRR*, vol. abs/2005.02597, 2020. [Online]. Available: https://arxiv.org/abs/2005.02597

[30] A. Kesting, M. Treiber, and D. Helbing, "General lane-changing model mobil for car-following models," *Transportation Research Record*, vol. 1999, no. 1, pp. 86–94, 2007. [Online]. Available: https://doi.org/10.3141/1999-10

[31] K. SENTZ and S. FERSON, "Combination of evidence in dempster-shafer theory." [Online]. Available: https://www.osti.gov/biblio/800792

[32] M. Naumann, H. Königshof, and C. Stiller, "Provably safe and smooth lane changes in mixed traffic," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, 2019, pp. 1832–1837.

[33] P. A. Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wiessner, "Microscopic traffic simulation using sumo," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, 2018, pp. 2575–2582.

[34] J. Bernhard, K. Esterle, P. Hart, and T. Kessler, "BARK: open behavior benchmarking in multi-agent environments," *CoRR*, vol. abs/2003.02604, 2020. [Online]. Available: https://arxiv.org/abs/2003.02604

[35] P. Orzechowski, K. Li, and M. Lauer, "Towards responsibility-sensitive safety of automated vehicles with reachable set analysis," 11 2019, pp. 1–6.

Lingguang Wang studied Automotive engineering in Shanghai, China and Karlsruhe, Germany, and received a B. Sc. from Tongji University, Shanghai, China and M. Sc. from Karlsruhe Institute of Technology, Karlsruhe, Germany in 2015 and 2018. He is currently pursuing a Ph.D. degree within the Institute of Measurement and Control Systems at Karlsruhe Institute of Technology, Germany, supervised by Prof. Dr.-Ing. Christoph Stiller. His research interests include risk assessment, motion planning and high-level decision making for autonomous driving.

Carlos Fernandez studied Computer Science Engineering at University of Alcala (Spain). He received the B. Sc. degree in 2008 and the M.Sc. in Advanced Electronics Systems and Intelligent Systems in 2010. His PhD was awarded with the highest mark CUM LAUDE in 2016 and it was focused on computer vision applied to intelligent transportation systems and autonomous driving under the supervision of Prof. Dr. Miguel Angel Sotelo. Since 2017 he is group leader at Institute of Measurement and Control Systems at Karlsruhe Institute of Technology, Germany. He is reviewer of IEEE conferences and journals and his research interests include smart infrastructure, perception methods and trajectory prediction for autonomous driving.

Christoph Stiller received the Electrical Engineering degree from Aachen, Germany, and Trondheim, Norway, and the Diploma and Dr. Ing. degrees from Aachen University of Technology, Aachen, Germany, 1988 and 1994, respectively. He became a Postdoctoral Scientist with the INRS Telecommunications, Montreal, QC, Canada In 1994. In 1995, he joined the Corporate Research and Advanced Development of Robert Bosch GmbH, Hildesheim, Germany. In 2001, he became a chaired Professor at Karlsruhe Institute of Technology, Germany. In 2010, he spent three months by invitation at CSIRO in Brisbane, Australia. In 2015, he was a Guest Scientist for five months with the Bosch RTC and Stanford University, Palo Alto, CA, USA. He served as Editor-in-Chief of the IEEE Intelligent Transportation Systems Magazine (2009-2011) and as Associate Editor for the IEEE Transactions on Image processing (1999-2003), for the IEEE Transactions on Intelligent Transportation Systems (2004-2015), for the IEEE Intelligent Transportation Systems Magazine (2012-ongoing) and as Senior Editor for the IEEE Transactions on Intelligent Vehicles (2015-ongoing). His Autonomous Vehicle AnnieWAY was a finalist in the Urban Challenge 2007 and the winner and second winner of the Grand Cooperative Driving Challenge 2011 and 2016, respectively. In 2013, he collaborated with Daimler on the automated Bertha Benz Memorial Tour.