

# Risk-aware motion planning for automated vehicle among human-driven cars

Jin I. Ge<sup>1</sup>, Bastian Schürmann<sup>2</sup>, Richard M. Murray<sup>1</sup>, and Matthias Althoff<sup>2</sup>

**Abstract**—We consider the maneuver planning problem for automated vehicles when they share the road with human-driven cars and interact with each other using a finite set of maneuvers. Each maneuver is calculated considering input constraints, actuator disturbances and sensor noise, so that we can use a maneuver automaton to perform high-level planning that is robust against low-level effects. In order to model the behavior of human-driven cars in response to the intent of the automated vehicle, we use control improvisation to build a probabilistic model. To accommodate for potential mismatches between the learned human model and human driving behaviors, we use a conditional value-at-risk objective function to obtain the optimal policy for the automated vehicle. We demonstrate through simulations that our motion planning framework allows an automated vehicle to exploit human behaviors with different levels of robustness.

## I. INTRODUCTION

High-level motion planning for automated vehicles has seen significant improvements since the 2004 DARPA Grand Driving Challenge, and a multitude of methods have been proposed for automated agents to reach a target while avoiding the surrounding obstacles [1]. While reach-avoid problems formulated in continuous time and space can easily incorporate low-level effects like input constraints and disturbances, the corresponding numerical solutions often suffer from high computational loads. This prompted the use of motion primitives to shift a large portion of computation costs offline [2]. In particular, when motion primitives are designed with consideration of low-level dynamics, they can provide a formal guarantee ensuring robustness in high-level motion planning [3], [4].

While discrete planners based on graph search (e.g., A\* algorithm) and sampling (e.g., RRT\*) have been successfully used in self-driving demonstrations, reach-avoid problems remain challenging in dynamic environment, where a high level of uncertainty can arise due to interaction with human-driven cars on road. Previous studies often simplify such behavior uncertainties by assuming human drivers maintain constant speed or follow given trajectories. As a result, the corresponding planning strategy often has difficulties to adapt to aggressive or friendly human behaviors, and thus tends to be overly defensive in order to ensure overall safety.

\*This work is supported by NSF VeHiCal project (Grant Number 1545126) and by the European Commission UnCoVerCPS project (Grant Number 643921).

<sup>1</sup>Department of Computing and Mathematical Sciences, California Institute of Technology, Pasadena, CA 91107 USA. [jge@caltech.edu](mailto:jge@caltech.edu), [murray@cds.caltech.edu](mailto:murray@cds.caltech.edu)

<sup>2</sup>Department of Informatics, Technische Universität München, Boltzmannstr. 3, 85748 Garching, Germany. [bastian.schuermann@tum.de](mailto:bastian.schuermann@tum.de), [althoff@tum.de](mailto:althoff@tum.de)

However, when the interaction between the automated agent and humans is explicitly modeled, the automated agent is able to employ far less conservative strategies [5]. Such explicit human models are crucial for the implementation of automated vehicles in human-dominated traffic. While we may describe human decision-making concisely using cost functions obtained through inverse reinforcement learning, studies have pointed out that human decisions are often not optimal due to limited rationality [6]. Therefore, instead of using deterministic human models that follow optimal policies under given cost functions, we may consider probabilistic models that are able to regenerate diverse human behaviors and reflect human preferences through underlying probability distributions. To achieve this, we choose the control improvisation technique [7]–[9] that allows us to explicitly formulate and validate such probabilistic specifications.

Since the sojourn time of a human-driven vehicle around the automated car can be too short to collect sufficient amount of data for behavior learning, the cost function and probabilistic preferences of a human driver often rely heavily on a-priori parameters trained with ensemble driver data. Therefore, it is desirable to allow the automated vehicle to adjust its optimal strategy based on its confidence level regarding the accuracy of probabilistic human models. In particular, we consider a conditional value-at-risk objective function [10], which quantifies an upper bound on the mismatch of the probabilistic human model. Further, optimal policies under various modeling errors can be obtained through risk-aware decision-making [11].

The remainder of this paper is organized as follows: We first describe the construction of an action set for both human-driven and automated cars using motion primitives. Then, we describe the probabilistic human model using control improvisation, where the cost function can be learned from inverse reinforcement learning. Next, we formulate the reach-avoid problem into a Markov decision tree problem and obtain the robust optimal policy based on conditional value-at-risk objective. Finally, we present several case studies and conclude with a summary and an outlook.

## II. CONSTRUCTING MANEUVERS THROUGH MOTION PRIMITIVES

In this section, we construct vehicle maneuvers using motion primitives, in order to formulate the motion planning problem in terms of Markov decision processes. For

simplicity, we model each vehicle by a kinematic model

$$\begin{aligned} \dot{p}_x &= v \cos(\theta) \\ \dot{p}_y &= v \sin(\theta) \\ \dot{v} &= u_1 + w_1 \\ \dot{\theta} &= u_2 + w_2 \end{aligned} \quad (1)$$

where  $p_x$  and  $p_y$  are the longitudinal and lateral positions for the car's center of mass,  $v$  is its velocity, and  $\theta$  is its orientation.  $u_1$  and  $u_2$  are the acceleration and steering input, respectively, with corresponding additive disturbances  $w_1$  and  $w_2$ . In (1), we consider bounded uncertainty  $w = [w_1, w_2] \in W$  and limited input  $u = [u_1, u_2] \in U$ . We denote the state vector by  $x = [p_x, p_y, v, \theta]$ , and we denote a possible solution to (1) under an input trajectory  $u(\cdot)$ , an initial state  $x(0)$ , and a disturbance trajectory  $w$  by  $x(t|u(\cdot), x(0), w)$ .

Figure 4 shows an example of the motion planning problem where the automated car (blue) moves from an initial state A to the final state D. Such a task can be planned by concatenating a sequence of lane-following and lane-change maneuvers. Different maneuver sequences such as  $(AC_2, C_2D)$ ,  $(AB_1, B_1C_1, C_1D)$ , and  $(AB_2, B_2D)$  may be chosen based on the motion of human-driven cars nearby (red and grey cars). Therefore, we define a motion primitive as a tuple

$$\mu = \left( v_0, x_f, S_0, u(\cdot|x(0), w), R_{[0, \Delta t]}(S_0, W) \right) \quad (2)$$

where  $\Delta t$  is the length of time for the motion primitive,  $v_0$  is the initial speed,  $x_f$  is the desired end state at  $\Delta t$ , and  $u(\cdot|x(0), w)$  is an input generator given the initial state  $x(0)$  and disturbance  $w$ . The initial set

$$S_0 = [-e_x, e_x] \times [-e_y, e_y] \times [v_0 - e_v, v_0 + e_v] \times [-e_\theta, e_\theta], \quad (3)$$

describes the state uncertainty around the nominal starting point  $(0, 0, v_0, 0)$  in the local coordinate system, where  $e_x$ ,  $e_y$ ,  $e_v$ , and  $e_\theta$  are the bounds for position, speed and orientation measurement noise. Based on the end state  $x_f$  and time horizon  $\Delta t$ , we may obtain the nominal trajectory and nominal input for the motion primitive, as shown by the black curves in Fig. 2. However, given the bounded disturbances and uncertainties, in order to ensure robustness around the nominal trajectory, we need to obtain an input generator  $u(t|x(0), w)$  for  $t \in [0, \Delta t]$ , with its corresponding reachable set over-approximated by

$$R_{[0, \Delta t]}(S_0, W) \supseteq \{x(t|u(\cdot), x(0), w) \mid t \in [0, \Delta t], x(0) \in S_0, w \in W\}, \quad (4)$$

where  $u(\cdot)$  is the input trajectory generated using (2). Based on the reachable set (4), we define the occupancy set

$$\mathcal{O}_{[0, \Delta t]}(S_0, W) \supseteq \{\Gamma(x) \mid x \in R_{[0, \Delta t]}(S_0, W)\}, \quad (5)$$

where the function  $\Gamma(x) : \mathbb{R}^4 \rightarrow \mathcal{P}(\mathbb{R}^2)$  maps the vehicle's state  $x$  to its occupancy area, where  $\mathcal{P}(\mathbb{R}^2)$  denotes the power set in the  $(p_x, p_y)$ -plane, see [3], [4] for details.

Note that compared with classic motion primitive settings such as [2], the occupancy calculation (5) under disturbances

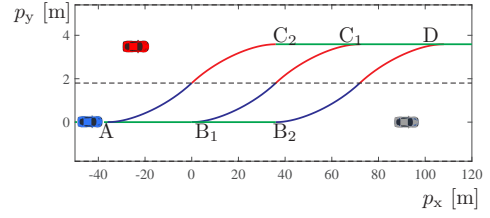


Fig. 1. Hierarchical motion planning for the automated car (blue) by concatenating lane-following and lane-changing maneuvers. Different sequences such as  $(AC_2, C_2D)$ ,  $(AB_1, B_1C_1, C_1D)$ , and  $(AB_2, B_2D)$  may be chosen based on the motion of human-driven cars nearby (red and grey).

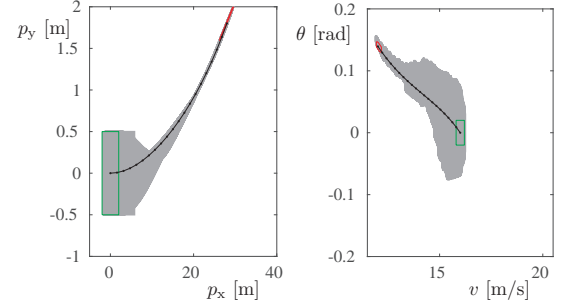


Fig. 2. A motion primitive used in left-turning maneuvers, see the blue curves in Fig. 1. The nominal trajectories are shown in solid curves while the shaded areas mark the reachable set  $R_{[0, \Delta t]}(S_0, W)$ . The green and red rectangles mark the initial and final sets, respectively.

and initial uncertainties ensures the robustness of high-level motion planning against low-level effects. In particular, concatenation of  $\mu^A$  with  $\mu^B$  is possible if the final set of  $\mu^A$  is a subset of the initial set of  $\mu^B$  under a translational and rotational transformation  $F_A(\cdot)$ , that is,

$$S_f^A(S_0^A, W) \subseteq F_A(S_0^B), \quad (6)$$

where the over-approximation of the motion primitive's end set

$$S_f(S_0, W) \supseteq \{x(\Delta t|u(\cdot), x(0), w) \mid x(0) \in S_0, w \in W\}. \quad (7)$$

As an example, we show in Fig. 2 a motion primitive with time period  $\Delta t = 2$  [s], initial speed  $v_0 = 16$  [m/s], and desired end state  $x_f = [28[m], 1.8[m], 12[m/s], 0.14[rad]]$  with input constraints and disturbance bounds

$$\begin{aligned} u_1 &\in [-4, 4] [m/s^2], u_2 \in [-0.4, 0.4] [rad/s], \\ w_1 &\in [-0.5, 0.5] [m/s^2], w_2 \in [-0.02, 0.02] [rad/s]. \end{aligned} \quad (8)$$

The corresponding reachable sets are shaded grey in the  $(p_x, p_y)$ -plane and  $(v, \theta)$ -plane, while the corresponding nominal trajectories are plotted in dotted curves. The initial set  $S_0$  and the final set  $S_f(S_0, W)$  are marked by green and red rectangles, respectively. Note that this motion primitive and its orientation-wise mirror primitive constitute a lane-change maneuver in Fig. 1 (i.e.,  $AC_2, B_1C_1, B_2D$ ).

Therefore, a maneuver  $a$  can be defined for an automated or human-driven vehicle as a concatenation of motion primitives, that is,

$$a = (\mu_1, \dots, \mu_{N_\mu}), \quad (9)$$

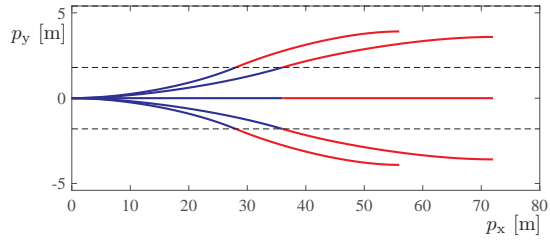


Fig. 3. One lane-keeping maneuver and four lane-changing maneuvers, where each maneuver consists of two motion primitives.

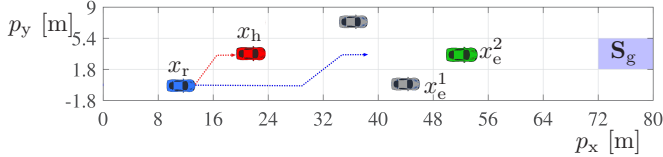


Fig. 4. Example of motion planning in a dynamic environment consisting of one automated car (blue), one interactive car (red), and two non-interactive cars (green and grey). While both interactive and non-interactive cars are human-driven, the latter does not respond to the motion of the automated car and is not included in the interactive framework. In order to move to the left lane, the automated car may choose to merge in front or behind the red vehicle, as marked by the blue and red arrows.

where  $N_\mu$  is the maximum number of motion primitives a maneuver contains, and  $\mu_{j-1}$  and  $\mu_j$  satisfy (6) for  $j = 2, \dots, N_\mu$ . While lane-keeping maneuvers on straight roads may have  $N_\mu = 1$ , lane-changing or lane-keeping maneuvers on curvy roads often have  $N_\mu > 1$ . For simplicity, in this paper we set  $N_\mu = 2$  for all maneuvers and assume the end orientation of each maneuver to be aligned with the direction of the road, as shown in Fig. 3.

### III. MANEUVER AUTOMATA MODELS

In this section, we describe the dynamics of both automated and human-driven vehicles, see Fig. 4. While the automated car is the only agent in the system that is controllable, we categorize the human-driven vehicles into the non-interactive cars and the interactive cars. The interactives (red) react to the intention of the automated car, while the motion of non-interactive vehicles (green and grey) is not influenced by the motion of the automated car (blue). Therefore, we construct Markov models for the motion of non-interactive vehicles that represent traffic conditions, and build maneuver automata to describe the interaction between the automated and interactive vehicles.

We denote the state vector for the human-robot system

$$X = [x_e, x_h, x_r], \quad (10)$$

with non-interactive vehicles (the environment)  $x_e = [x_e^1, \dots, x_e^{N_e}]$ , interactive vehicles (the humans)  $x_h = [x_h^1, \dots, x_h^{N_h}]$ , and the automated car (the robot)  $x_r$ . For the case shown in Fig. 4,  $N_e = 2$  and  $N_h = 1$ .

To reason about the discrete-time maneuver planning, we consider the state of each vehicle at the beginning or the end of a maneuver. Since each car's orientation is assumed

to be aligned with the direction of the lane at the end of a maneuver, for simplicity we may omit the orientation  $\theta$  in the discrete states and define

$$\begin{aligned} x_e^i &= [n_x^{e,i}, n_y^{e,i}, n_v^{e,i}], \\ x_h^i &= [n_x^{h,i}, n_y^{h,i}, n_v^{h,i}], \quad x_r = [n_x^r, n_y^r, n_v^r], \end{aligned} \quad (11)$$

where  $n_x$  is the discretized longitudinal position, see the grids in Fig. 4,  $n_y$  is the lane number, and  $n_v$  is the discretized speed.

Using the discrete state definition (11), we describe the motion of non-interactive vehicles with discrete-time Markov models, where the transition probability

$$[T_e]_{ijk} = P_e(x_e[k+1] = s_j | x_e[k] = s_i) \quad (12)$$

can be trained from traffic data, see [12] for more details.

On the other hand, the motion of the automated vehicle can be described by a maneuver automaton

$$\mathbb{M}_r = \{\mathbb{X}_r, x_r[0], \mathbb{A}_r, C_r, T_r, S_g\} \quad (13)$$

- the sample space  $\mathbb{X}_r \subset \mathbb{R}^3$ ,
- the initial state  $x_r[0] \in \mathbb{R}^3$ ,
- the input space  $\mathbb{A}_r = \{a_k, k = 1, \dots, K\}$  where the maneuver  $a_k$  is defined in (9),
- $C_r(x_r, a_r)$  is the instantaneous cost for the automated car at state  $x_r$  with action  $a_r$ ,
- the conditional transition matrix  $T_r$ ,
- the accepting states  $S_g \in \mathcal{P}(\mathbb{R}^3)$ ,

where the size of the sample space  $|\mathbb{X}_r|$  is finite, because the vehicle speed is bounded, and the vehicle position is between the position of the initial state  $x_r[0]$  and those of the accepting states  $S_g$ .

In (13) the cost  $C_r(x_r, a_r)$  may contain the cost for potential collisions, the cost-to-goal, and the action cost. Therefore, its value depends on the states and actions of neighboring human-driven vehicles. For example, we may denote  $C_r(x_r, a_r | x_e, x_h, a_h)$  at step  $i$  as  $C_r[i]$ , and define it as

$$\begin{aligned} C_r[i] &= f_{r1}(\mathcal{O}_r[i], \mathcal{O}_h[i]) + f_{r2}(\mathcal{O}_e[i], \mathcal{O}_r[i]) \\ &\quad + d(\mathcal{O}_r[i], S_g) + \gamma_4 \|a_r[i]\|^2, \end{aligned} \quad (14)$$

where  $\gamma_4 > 0$ , and by abuse of notation we denote  $\mathcal{O}_r[i]$ ,  $\mathcal{O}_h[i]$ , and  $\mathcal{O}_e[i]$  as the grids occupied during maneuver  $i$  by the automated, interactive, and non-interactive vehicles, respectively. Details on the projection from the continuous occupancy set (5) to the discrete grids can be found in [13]. The collision costs  $f_{r1}, f_{r2}$  are non-negative functions that decrease monotonically with respect to the distance between the automated vehicle and the interactive and non-interactive vehicles, e.g.,

$$f(S_1, S_2) = \max(0, k_c(\bar{N} - \|S_1, S_2\|_1^2)), \quad (15)$$

where  $\|S_1, S_2\|_1$  denotes the 2-norm for the corresponding portion of  $S_1$  and  $S_2$  that are in the same lane,  $\bar{N}$  is the minimum distance beyond which the collision cost is zero, and  $k_c > 0$  is the weight on the collision cost. The distance-to-goal cost is

$$d(\mathcal{O}_r[i], S_g) = l(\mathcal{O}_r[i], S_g) + \min_{s \in S_g} \|\mathcal{O}_r[i] - s\|_2^2, \quad (16)$$

where  $l(\mathcal{O}_r[i], S_g) = 0$  when  $\mathcal{O}_r[i]$  is in the same lane as  $S_g$  and  $l(\mathcal{O}_r[i], S_g) > 0$  otherwise. Therefore, the automated car's cost  $C(x_r, a_r)$  is a random variable with respect to the motion of human-driven vehicles.

We also note that the conditional transition probability for  $s_i \rightarrow s_j$  given action  $a_k$  is

$$[T_r]_{ijk} = P_r(x_r[l+1] = s_j | x_r[l] = s_i, a_r[l] = a_k), \quad (17)$$

for  $i, j = 1, \dots, |\mathbb{X}_r|$ ,  $k = 1, \dots, K$ , which can be obtained using the maneuver definition (2,9). However, (13) does not specify how  $a_r[k]$  is chosen given the history  $X[0], \dots, X[k-1]$  and the accepting states  $S_g$ , this problem will be formulated and solved in the next sections.

Similarly, the dynamics of an interactive vehicle can be defined as

$$\mathbb{M}_h = \{\mathbb{X}_h, x_h[0], \mathbb{A}_h, C_h, T_h\}, \quad (18)$$

where its sample space  $\mathbb{X}_h$ , initial state  $x_h[0]$ , and action set  $\mathbb{A}_h$  are defined similarly as in (13). The transition probability

$$[T_h]_{ijk} = P_h(x_h[l+1] = s_i | x_h[l] = s_j, a_h[l] = a_k) \quad (19)$$

can also be obtained using the maneuver definition (2,9). Similarly, we may define the human's cost function  $C_h(x_h[k], a_h[k] | x_e[k], x_r[k], a_r[k])$  in terms of

$$C_h[k] = f_{h1}(\mathcal{O}_r[k], \mathcal{O}_h[k]) + f_{h2}(\mathcal{O}_e[k], \mathcal{O}_h[k]) + \gamma_a \|a_h[k]\|^2, \quad (20)$$

where  $\gamma_a$  is the weight for the input cost, while the collision cost  $f_{h1}$  and  $f_{h2}$  against the motion of the automated and the non-interactive vehicles are defined similarly as in (14).

While some previous research consider deterministic policies  $a_h[k]$  in (18), a probabilistic distribution given the state history might describe a human driver's action better [14], [15]. In the next section, we will provide details on a probabilistic model for the interactive vehicles using control improvisation.

#### IV. MODELING THE BEHAVIOR OF INTERACTIVE VEHICLES

Here, we describe the discrete-time decision-making of interactive vehicles based on the maneuver automate model (18). To break the symmetry where the decision of automated and interactive vehicles depend on each other, we assume the automated car broadcasts its intentions through advanced signaling and vehicle-to-vehicle communication, so that the human drivers driving behind or next to it may react to its intention [5]. However, due to limited rationale, human drivers are not always able to obtain the optimal policy and generate stochastic behaviors instead.

Therefore, we consider human actions  $a_h[k]$  generated by a distribution conditioned on the state of other vehicles and the automated car's action, that is,

$$a_h[k] \leftarrow P(a_h[k] | x_h[k], x_e[k], x_r[k], a_r[k]), \quad (21)$$

where  $\leftarrow$  denotes sampling from the probability on the right-hand side.

To ensure this distribution represents human decisions well in traffic, we use control improvisation to ensure that the generated behaviors obey a set of hard constraints (conditions that are satisfied deterministically), a set of soft constraints (conditions that are satisfied with a probability), and a randomness requirement on the behavior richness [7].

That is, given a finite alphabet  $\Sigma$ , we would like to generate a randomized behavior in the language  $I \subset \Sigma^*$ . Given a hard-constraint set  $S \subset \Sigma^*$ , finitely many soft-constraint sets  $A_i \subset I$  with probability bounds  $\xi_i \in [0, 1]$  where  $i \in \{1, \dots, n\}$ , and a probability bound  $\rho \in (0, 1]$ , the distribution  $D: \Sigma^* \rightarrow [0, 1]$  with support set  $S$  such that

- 1)  $S \subset I$ ,
- 2)  $\forall i = 1, \dots, n, \quad P[\omega \in A_i | \omega \leftarrow D] \geq 1 - \xi_i$ ,
- 3)  $\forall \omega \in S, \quad D(\omega) \leq \rho$ ,

is called an  $(\xi, \rho)$ -improvising distribution. The hard constraint 1) is used to specify all behaviors with non-zero probabilities (e.g., limited steering and acceleration capabilities); the soft constraint 2) is used to regulate the probability for certain sets of behaviors (e.g., the probability of the human driver performing certain motion primitives); while the randomness requirement 3) ensures a low probability for a human driver to repeat any particular maneuver sequence.

Here, the human driver's action in response to the automated car's intention can be formulated as satisfying the following specifications:

##### 1) **Hard constraint**

The human-driven cars must follow the acceleration and steering limits while remaining on the road, that is,

$$a_h[k] \in U, \quad x_h[k] \in \mathcal{X}, \quad \forall k \geq 0. \quad (22)$$

This hard constraint does not require a human driver to guarantee his or her action will not lead to imminent collisions. However, such faulty human behaviors has low probabilities since most human drivers are able to select safe maneuvers in normal driving conditions, as will be specified in the soft constraints below.

##### 2) **Soft constraints**

Based on the limited rationale assumption [6], the cost optimization (20) is too resource-exhausting for human drivers. Therefore, we assume human drivers associate maneuvers generating high costs with low probability, i.e.,

$$P(C_h[k] < C^{\text{th}}) \geq p^{\text{th}}, \quad \forall k \geq 0, \quad (23)$$

where  $C_h[k]$  is defined in (20),  $C^{\text{th}}$  is the cost threshold below which human drivers are not sensitive to, and  $p^{\text{th}}$  is the corresponding probability threshold. Note that (23) only considers the time horizon of one step ahead, which corresponds to the findings in [16] that the look-ahead horizon of human drivers are relatively short in uncertain decision-making settings.

##### 3) **Randomness**

The randomness condition ensures richness in the generated actions of human drivers. It requires that a particular

sequence of motion primitives has low probability. This condition is often satisfied by ensuring a sufficient number of actions inside the hard and soft constraints, see [7] for more details.

The principle of maximum entropy [15] can be used to obtain an estimation of the functions  $f_{h1}(\cdot)$ ,  $f_{h2}(\cdot)$  and parameter  $\gamma_a$  in (18) based on driving data. However, it is necessary to check that the trained distribution (21) satisfies the soft constraint (23) under given traffic conditions, see [12] for more details.

## V. INTERACTIVE MOTION-PLANNING FRAMEWORK

In this section, we formulate the motion-planning framework that exploits the interaction between the automated and human-driven vehicles while providing robustness against inaccuracies in the probabilistic human models. We also sketch the value iteration method used to obtain an optimal sequence of maneuvers in the interactive motion-planning framework.

### A. Probabilistic reach-avoid task

We model the motion of the human-robot system during one reach-avoid task using a Markov decision tree

$$\mathbb{M} = (\mathcal{X}, \mathcal{A}, C_r, \mathbb{P}, X[0], D, S_g) \quad (24)$$

with the following components:

- the sample set  $\mathcal{X} \subset \mathbb{X}_r \times \mathbb{X}_h \times \mathbb{X}_e$ ,
- the action set  $\mathcal{A} \subset \mathbb{A}_r \times \mathbb{A}_h$ ,
- $C_r(X, a)$  is the instantaneous cost for the automated car at state  $X$  with action  $a = (a_r, a_h)$ ,
- the transition probability  $\mathbb{P}(\cdot | X, a)$  for state  $X$  with action  $a$ ,
- the initial state  $X[0] = [x_e[0], x_h[0], x_r[0]]$ ,
- the maximum tree depth  $D$ ,
- the accepting states  $S_g \in \mathcal{P}(\mathbb{R}^3)$ .

The termination conditions of the Markov decision tree are

- 1) if the maximum depth  $D$  is reached ( $k = D$ ),
- 2) if the goal region is reached

$$x_r[k | x_r[0], a_r[i]] \in S_g, \quad (25)$$

where  $i = 0, \dots, k-1$  for  $k < D$ , and

- 3) if a collision happens

$$\mathcal{O}(x_r[k], a_r[k]) \cap \left( \mathcal{O}(x_h[k], a_h[k]) \cup \mathcal{O}(x_e[k], a_e[k]) \right) \neq \emptyset, \quad (26)$$

where  $\mathcal{O}(x[k], a[k])$  with subscripts r, h, and e are the occupancy set of the automated, interactive, and non-interactive vehicles during the  $k^{\text{th}}$  maneuver, respectively. Therefore, the size of the sample set  $\mathcal{X}$  can be significantly smaller than the product set  $\mathbb{X}_r \times \mathbb{X}_h \times \mathbb{X}_e$ .

Based on the Markov model (24), we define the automated car's objective as to find an optimal maneuver sequence  $a_r^*$  that minimizes its conditional risk-at-value cost during the reach-avoid task, that is,

$$a_r^* = \arg \min_{a_r \in (\mathbb{A}_r)^k} \text{CVaR}_\alpha \left( \sum_{i=0}^k C_r[i] \mid X[0], a_r, a_h \right), \quad (27)$$

where  $\text{CVaR}_\alpha$  denotes the conditional value-at-risk at caution level  $\alpha$  against the action distribution of interactive vehicles (21). When the caution level is low ( $\alpha \rightarrow 0^+$ ), the conditional value-at-risk  $\text{CVaR}_\alpha(Z)$  approaches the mean value  $\mathbb{E}(Z)$ . When the risk level is high ( $\alpha \rightarrow 1^-$ ), the conditional value-at-risk  $\text{CVaR}_\alpha(Z)$  approaches the worst-case scenario  $\max(Z)$ . In particular,  $\text{CVaR}_\alpha(Z)$  is convex in  $Z$  and continuous with respect to the caution level  $\alpha$  [10].

$\text{CVaR}_\alpha(Z)$  can be interpreted as the worst-case expectation of  $Z$  when the distribution  $\mathbb{P}(\cdot | X, a)$  is perturbed

$$\text{CVaR}_\alpha(Z) = \max_{\xi \in \mathcal{E}(\alpha, \mathbb{P})} \mathbb{E}_\xi[Z], \quad (28)$$

where  $\mathbb{E}_\xi$  denotes the  $\xi$ -weighed expectation of  $Z$  with the risk envelop

$$\mathcal{E}(\alpha, \mathbb{P}) = \left\{ \xi \mid \xi(\omega) \in \left[ 0, \frac{1}{1-\alpha} \right], \int \xi(\omega) \mathbb{P}(\omega) d\omega = 1 \right\}. \quad (29)$$

Therefore, (28) shows that the conditional value-at-risk can be used to provide specific levels of robustness in the automated car's optimal policy against the probabilistic model of interactive vehicles.

Similar to most motion-planning methods in dynamic and uncertain environments [17], the interactive motion-planning framework (21,24,27) cannot guarantee absolute collision avoidance. Instead, we may restrict the automated car's action set  $\{a_r[i]\}$  at step  $i$  to the actions that do not lead to imminent collisions against the most probable human action.

### B. Value iteration

The optimal policy problem (21,27) can be solved using the CVaR decomposition theorem in [11]. Denote the value function at the state  $s \in \mathcal{X}$  and caution level  $y \in (0, 1)$  as

$$V(s, y) = \min_{a \in \mathcal{A}} \text{CVaR}_y \left( \sum_{i=0}^k C_r[i] \mid X[0] = s, a \right) \quad (30)$$

and define the Bellman operator

$$T[V](s, y) = \min_{a \in \mathcal{A}} \left( C_r(s, a) + \max_{\xi \in \mathcal{E}(y, \mathbb{P}(\cdot | s, a))} \sum_{s' \in \mathcal{X}} \xi(s') V(s', y, \xi(s')) \mathbb{P}(s' | s, a) \right). \quad (31)$$

Then for all states  $s \in \mathcal{X}$  and caution level  $y \in (0, 1)$ , the stationary point  $V^*(s, y)$  of the Bellman operator

$$T[V^*](s, y) = V^*(s, y) \quad (32)$$

is the optimal conditional value-at-risk

$$V^*(s, y) = \min_{a_r \in (\mathbb{A}_r)^k} \text{CVaR}_y \left( \sum_{i=0}^k C_r[i] \mid X[0] = s, a_r, a_h \right), \quad (33)$$

with the optimal policy

$$a_r^*[k] = a^*(s[k], y[k]), \quad (34)$$

for the iterative caution level

$$y[k] = y[k-1] \xi_{s[k-1]}^*(s[k]). \quad (35)$$

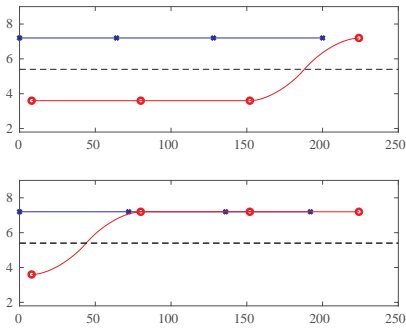


Fig. 5. Nominal trajectories of the automated car (red circles) and the human-driven car (blue cross) in high-speed driving under different caution levels. Both longitudinal and lateral distances are in the unit of meters. Upper panel: high caution level  $\alpha = 0.9$ ; Lower panel: low caution level  $\alpha = 0.1$ ;

The Bellman equation (31,32) can be solved using a value iteration scheme as described in [11].

## VI. CASE STUDIES

In this section, we consider a high-speed scenario and a low-speed scenario for the interactive motion planning of an automated vehicle changing into the left lane. For simplicity, here we only consider one automated vehicle and one interactive vehicle, while no non-interactive vehicles are considered, cf.  $x_r$  and  $x_h$  in Fig. 4.

For the high-speed scenario, we consider the steady-state speed for both the automated and interactive vehicle as 16 [m/s], while the time step for each maneuver (9) is 4 [s]. The automated car has 6 maneuvers that correspond to {accelerate in the lane, constant-speed in the lane, decelerate in the lane, change the lane with acceleration, change the lane with the same speed, change the lane with deceleration}. The interactive vehicle has 3 maneuvers corresponding to {accelerate in the lane, constant-speed in the lane, decelerate in the lane}. While the motion-planning framework allows larger maneuver sets and more human-driven vehicles, to demonstrate its benefits, we present this most simplified case.

In Fig. 5, we consider the case where the automated car's initial longitudinal position (red circle) is 8 [m] in front of the human-driven vehicle (blue star). In the upper panel of Fig. 5, the automated car has the high caution level  $\alpha = 0.9$ . Therefore it plans against the most aggressive human behavior, where the human driver will not slow down to let him cut in front. Thus, the automated car uses two lane-keeping maneuvers to increase its longitudinal distance from the human-driven car, before eventually executing the lane-change maneuver.

On the other hand, in the lower panel of Fig. 5, with the low caution level  $\alpha = 0.1$ , the automated car fully exploits the understanding that the human driver is highly likely to respond to the automated car's behaviors so that no imminent collision will happen (23). Therefore, the automated car decides to make bolder actions, i.e., cutting in front of the human-driven car at the first step.

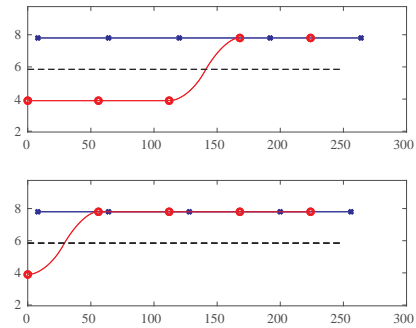


Fig. 6. Nominal trajectories of the automated car (red circles) and the human-driven car (blue cross) in high-speed driving under different caution levels. Both longitudinal and lateral distances are in the unit of meters. Upper panel: high caution level  $\alpha = 0.9$ ; Lower panel: low caution level  $\alpha = 0.1$ ;

Fig. 6 shows a symmetric case, where the automated car starts slightly behind the human-driven car. In the upper panel, the higher caution level prompts the automated car to employ several lane-keeping maneuvers over which its distance to the human-driven car increases. However, for the lower caution level, the automated car merges behind the human-driven car immediately, fully exploit the belief that the human driver is unlikely to brake heavily.

Indeed, planning with low caution levels requires high confidence in the human interaction model (21), especially when the automated car has higher priority (e.g., a police car or ambulance), or when the gesture of the human driver and the motion of the interactive vehicle indicate amicable driving behavior. However, if the caution level is too low compared with the accuracy of the interactive model, the planned path may have to be terminated and re-planning engaged.

We also note that this planning framework (21,27) can be simplified to widely-used motion-planning methods such as the A\* algorithm, especially when the human-driven vehicle's model (21) is replaced with non-interactive assumptions such as the constant-speed assumption. Without interactive models, a planning algorithm can have difficulties generating a path for merging or lane-changing in dense traffic.

Therefore, we consider a low-speed driving scenario where the automated car needs to change to the right lane within 10 grids (with in a longitudinal distance of 80 [m]), see Fig. 7. The left and middle panels show the cases when the automated car has low caution level. In both cases, it maintains the constant speed of 4 [m/s] for the first step while indicating the lane-change intention. Then, as the human driver reacts to such an action either by speeding up and letting the automated car to merge behind it (left) or by slowing down and letting the automated car to cut in front of it (middle), the automated car decides to decelerate or accelerate at the second step. However, when the caution level is high (right panel), the automated car plans with the maximum cost against human behaviors, and it decides to accelerate considering the worst case where the human driver would also accelerate. While the right panel shows one

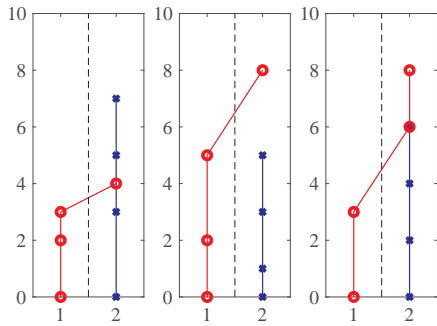


Fig. 7. State sequences of the automated car and the human-driven car in low-speed driving under different caution levels. The longitudinal distance is in the unit of 8 [m], while the lateral distance is in the unit of the lane width. Left panel: Caution level  $\alpha = 0.05$ , and the human driver does not wish to let the automated car cut in front. Middle panel: Caution level  $\alpha = 0.05$  and the human driver lets the robot to cut in front. Right panel: Caution level  $\alpha = 0.95$  where regardless of the human behavior, the automated car tries to accelerate before lane-changing.

run where the lane-change task is successfully finished, not utilizing the human response can lead to a larger probability of not fulfilling the lane-change task.

Through these three simulations, we demonstrate the flexibility the caution level  $\alpha$  brings to interactive motion planning. In particular, it explicitly balances between exploiting the friendly behavior of human drivers and retaining robustness against inaccuracy in the interaction model, which may not be achieved through many existing graph-search methods.

## VII. CONCLUSIONS

We proposed an interactive motion-planning framework for automated vehicles when they share the road with human-driven cars. We constructed maneuver automata to describe the dynamics of both automated vehicles and human-driven vehicles using motion primitives. Then we built a probabilistic model to describe the action of human drivers responding to the intention of an automated vehicle. We used a conditional value-at-risk formulation to describe the interactive motion-planning problem, exploiting the probabilistic human response model while providing desired levels of robustness against mismatches between the learned human model and human driving behaviors. Finally, we demonstrated through simulations that the interactive motion-planning strategy allows the behavior of an automated car to adapt to different caution levels regarding the human model. In future research, it is desirable to ensure probabilistic constraints on guaranteed safety and analyze the influences on traffic flow.

## REFERENCES

- [1] D. Gonzalez, J. Prez, V. Milans, and F. Nashashibi, "A review of motion planning techniques for automated vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 4, pp. 1135–1145, 2016.
- [2] E. Frazzoli, M. A. Dahleh, and E. Feron, "Maneuver-based motion planning for nonlinear systems with symmetries," *IEEE Transactions on Robotics*, vol. 21, no. 6, pp. 1077–1091, 2005.

- [3] B. Schürmann and M. Althoff, "Convex interpolation control with formal guarantees for disturbed and constrained nonlinear systems," in *Proceedings of the 20th International Conference on Hybrid Systems: Computation and Control*, 2017, pp. 121–130.
- [4] B. Schürmann, D. Heß, J. Eilbrecht, O. Stursberg, F. Kster, and M. Althoff, "Ensuring drivability of planned motions using formal methods," in *Proceedings of the 20th International Conference on Intelligent Transportation Systems (ITSC)*, 2017, pp. 1–8.
- [5] D. Sadigh, S. Sastry, S. A. Seshia, and A. D. Dragan, "Planning for autonomous cars that leverages effects on human actions," in *Proceedings of the Robotics: Science and Systems Conference (RSS)*, 2016, pp. 66–73.
- [6] T. L. Griffiths, F. Lieder, and N. D. Goodman, "Rational use of cognitive resources: Levels of analysis between the computational and the algorithmic," *Topics in Cognitive Science*, vol. 7, no. 2, pp. 217–229, 2015.
- [7] D. J. Fremont, A. Donzé, S. A. Seshia, and D. Wessel, "Control improvisation," in *35th IARCS Annual Conference on Foundation of Software Technology and Theoretical Computer Science (FSTTCS)*, vol. 45, 2015, pp. 463–474.
- [8] A. Donzé, R. Valle, I. Akkaya, S. Libkind, S. A. Seshia, and D. Wessel, "Machine improvisation with formal specifications," in *Proceedings of the 40th International Computer Music Conference (ICMC)*, September 2014.
- [9] I. Akkaya, D. Fremont, R. Valle, A. Donzé, E. A. Lee, and S. A. Seshia, "Control improvisation for probabilistic temporal specifications," in *Proceedings of the 1st IEEE International Conference on Internet-of-Things Design and Implementation (IoTDI)*, 2016, pp. 55–70.
- [10] R. T. Rockafellar and S. Uryasev, "Optimization of conditional value-at-risk," *Journal of Risk*, vol. 2, pp. 21–41, 2000.
- [11] Y. Chow, A. Tamar, S. Mannor, and M. Pavone, "Risk-sensitive and robust decision-making: a cvar optimization approach," in *Advances in Neural Information Processing Systems 28*, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, Inc., 2015, pp. 1522–1530.
- [12] J. I. Ge and R. M. Murray, "Voluntary lane-change policy synthesis with control improvisation," in *Proceedings of the IEEE 57th International Conference on Decision and Control*, 2018, pp. 1–8.
- [13] M. Althoff, O. Stursberg, and M. Buss, "Model-based probabilistic collision detection in autonomous driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 10, no. 2, pp. 299–310, 2009.
- [14] D. Sadigh, K. Driggs Campbell, A. A. A. Puggelli, W. Li, V. Shia, R. Bajcsy, A. L. Sangiovanni-Vincentelli, S. S. Sastry, and S. A. Seshia, "Data-driven probabilistic modeling and verification of human driver behavior," EECS Department, University of California, Berkeley, Tech. Rep. UCB/EECS-2013-197, Dec 2013.
- [15] B. D. Ziebart, A. Maas, J. A. Bagnell, and A. K. Dey, "Maximum entropy inverse reinforcement learning," in *Proc. AAAI*, 2008, pp. 1433–1438.
- [16] D. Carton, V. Nitsch, D. Meinzer, and D. Wollherr, "Towards assessing the human trajectory planning horizon," *PLOS ONE*, vol. 11, no. 12, pp. 1–39, 2016.
- [17] N. E. D. Toit and J. W. Burdick, "Robot motion planning in dynamic, uncertain environments," *IEEE Transactions on Robotics*, vol. 28, no. 1, pp. 101–115, 2012.