RACP

Risk-Aware Contingency Planning with Multi-Modal Predictions

Mustafa, Khaled A.; Ornia, Daniel Jarne; Kober, Jens; Alonso-Mora, Javier

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

RACP: Risk-Aware Contingency Planning With Multi-Modal Predictions

Khaled A. Mustafa , Daniel Jarne Ornia , Jens Kober , and Javier Alonso-Mora

Abstract—For an autonomous vehicle to operate reliably within real-world traffic scenarios, it is imperative to assess the repercussions of its prospective actions by anticipating the uncertain intentions exhibited by other participants in the traffic environment. Driven by the pronounced multi-modal nature of human driving behavior, this paper presents an approach that leverages Bayesian beliefs over the distribution of potential policies of other road users to construct a novel risk-aware probabilistic motion planning framework. In particular, we propose a novel contingency planner that outputs long-term contingent plans conditioned on multiple possible intents for other actors in the traffic scene. The Bayesian belief is incorporated into the optimization cost function to influence the behavior of the short-term plan based on the likelihood of other agents' policies. Furthermore, a probabilistic risk metric is employed to fine-tune the balance between efficiency and robustness. Through a series of closed-loop safety-critical simulated traffic scenarios shared with human-driven vehicles, we demonstrate the practical efficacy of our proposed approach that can handle multi-vehicle scenarios.

Index Terms—Planning under uncertainty, risk-awareness, autonomous vehicles, contingency planning, multi-modality.

I. Introduction

SAFE motion planning is a prominent feature in the self-driving stack. In urban scenarios, the ego-agent needs to understand and infer the intended motion of other road users in the scene in order to move safely and efficiently. However, predicting the behavior of road users poses great challenges since they exhibit non-deterministic and multi-modal behaviors. Moreover, their intentions cannot be explicitly communicated to the ego-agent. For instance, in a non-signalized intersection, it is hard to anticipate whether a human driver will drive straight or turn right, and it is crucial to encode this uncertainty into the planning formulation. This gives rise to stochastic prediction models that provide probabilistic information over all possible intentions the human driver can exhibit [1], [2], [3], [4]. By leveraging this probabilistic information, the ego-agent's motion planner needs to generate safe trajectories yet not overly conservative in the presence of other road users.

An important aspect of planning under uncertainty is to guarantee the existence of collision-free trajectories despite the stochastic motion of the surrounding obstacles. One class of methods that deals with the uncertain behavior of dynamic agents is *robust optimization* [5] which provides safety guarantees by rigorously accounting for bounded sets of uncertainties. That is, the probability density function of the uncertainty is non-zero over a bounded domain of the ego agent's workspace and is zero elsewhere. However, since robust optimization accounts for all possible realizations of the uncertainty, its behavior is excessively conservative and may result in infeasible solutions in crowded environments, a well-studied issue in robot navigation literature known as the "frozen robot" problem [6]. In contrast, *stochastic optimization* [7] uses *chance constraints* [8], [9], [10] to loosen hard constraints and bound the probability of violating safety constraints to be within a desired confidence level δ. This, in turn, results in less-conservative behavior compared to robust optimization approaches. However, in multi-modal traffic scenarios, this can still result in conservative behavior since a single trajectory is sought to minimize the optimization's cost function along the entire planning horizon. This gives rise to contingency planning frameworks [14], [17], [30], that explicitly plan a set of conditional actions that depend on the stochastic outcome of a prediction model.

As an illustrative example, to highlight the difference between single policy planning and contingency planning, consider the three-way intersection scenario depicted in Fig. 1. In this scenario, we consider only two possible intents a human driver can express. The one depicted in blue shows that the human driver yields to the autonomous vehicle, whereas the red one indicates that the autonomous vehicle brakes since the human driver takes an aggressive left turn. A traditional planner, in that situation, seeks a single plan that is safe with respect to both intents resulting in a braking trajectory. In contrast, since only one of the predicted intentions will happen in the future, the contingency planner plans a short-term trajectory that ensures safety for both potential outcomes. Subsequently, it diverges into two specific plans, each tailored to address a distinct future intention resulting in less-conservative plans.

II. Related Work and Contribution

A. Related Work

A primary objective of the motion planner is to generate non-conservative, yet safe trajectories, for the ego vehicle to execute. In some of the proposed methods in the literature, the
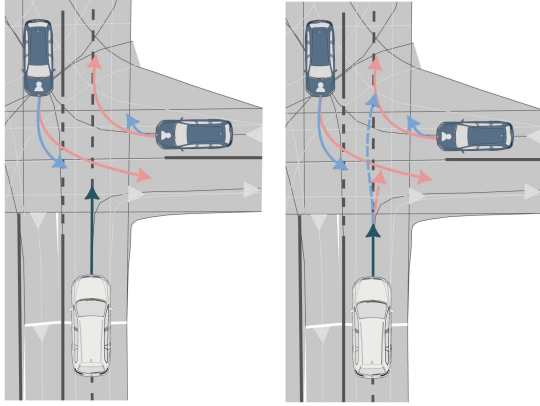
Fig. 1. Two of the possible future intents of the human-driven vehicle are shown in red and blue. On the left, the ego-vehicle seeks a single plan which is safe with respect to both intents. On the right, a short-term trajectory is planned that branches into two contingent plans for each human's intents. The illustrative example is inspired by [16].

planner optimizes for the worst-case scenario, of how the motion of other road users will propagate into the future, regardless of its likelihood [50]. Despite being safe, this causes the ego vehicle to behave defensively and overreact to low-probability dangers far into the future, e.g., the ego vehicle brakes prematurely to react to an unlikely future which would be uncomfortable and socially confusing for other road users. Human behavior, in these situations, would not be either overly conservative or completely ignorant of such rare scenarios. An alternative way is to generate a single plan, that safeguards against all possible intents along the entire planning horizon which results in inefficient and conservative plans with compromised performance [20], [21]. As a consequence, uncertain scenarios require autonomous vehicles to find a reasonable trade-off between safety and efficiency. This gives rise to the problem of *planning with multi-modal predictions*.

1) Planning With Multi-Modal Predictions: Fail-safe motion planning was introduced in [11], [12], [13] where a single trajectory is generated by considering the most probable trajectories of the other agents. Then, the safety of the proposed method is guaranteed by ensuring the existence of an emergency maneuver at each time step that accounts for every possible trajectory of the other agents. Although the fail-safe trajectory may ensure safety on a finite horizon plan, recursive feasibility is not guaranteed. Moreover, it is difficult to estimate whether the ego-vehicle is close to a collision.

Legibility-based models are widely used in the literature to alleviate the conservatism of planning under uncertainty [22], [23], [24]. A motion is considered legible if it allows an observer to confidently infer the correct agent's intent after observing a snippet of its trajectory [25], and the legibility of the motion depends on the required time until an observer can infer an agent's intent. In these approaches, a probabilistic model is used to infer the probability of a certain goal $G \in \mathcal{G}$ from an incomplete initial trajectory $\zeta_{S \to Q}$, $P(G|\zeta_{S \to Q})$, and the agent's inferred goal is modeled as the most likely one $\arg \max_{G \in \mathcal{G}} P(G|\zeta_{S \to Q})$. This may, however, result in an over-confident plan leading to

a collision since the probability that the agent moves towards a different goal is entirely ignored.

Branch Model Predictive Control (B-MPC), on the other hand, is utilized in [26], [27], [30], [31] to tackle the multi-modality arising from human-driving decision-making, where the behavior of the surrounding agents is simplified with a finite set of policies derived from a prediction model. A probabilistic scenario tree is then constructed from this finite set where each branch in the tree has an associated policy. On top of the scenario tree, a trajectory tree is built that shares the same topology as the scenario tree where the objective is to minimize the expected cost over all the branches. Yet, these approaches suffer from the *curse of dimensionality* since the tree structure grows exponentially with the prediction horizon and the number of agents, making it only feasible for short planning horizons [32].

Another line of work for planning under uncertainty is by formulating the planning problem as a partially observable Markov decision process (POMDP) by constructing a belief tree based on a discrete set of obstacle vehicle's intentions [28], [29]. Solving such problems, however, becomes computationally intractable when the problem size scales. To address this problem, multi-policy decision-making [33], [34], [35] decomposes the belief tree into a limited number of closed-loop policies by leveraging semantic information. However, these approaches solve for the best ego-policy over all possible future realizations resulting in overly-conservative plans that do not exploit the multi-modality in the obstacle-vehicle behavior. To tackle this problem, inspired by branch-MPC, TPP [36] proposes an approach that converts the continuous space motion planning problem into a tractable problem by converting both the trajectory tree and scenario tree into a finite-horizon MDP. The optimal policy is then determined via dynamic programming over the constructed MDP. Despite its scalability to multiple vehicles, TPP outputs a single optimal policy over all scenarios causing unavoidable loss of multi-modality information.

This issue can be tackled by separating the planning problem into short-term and long-term responses [14], [15], [16], [17]. This is realized by generating a short-term trajectory, for $t < t_b$, where t_b indicates a branching time, which guarantees safety with respect to all possible future intents that the other agents can exhibit. After t_b duration, it is assumed that the ego vehicle will be able to determine the intent of the human driver, and thus it is sufficient to safeguard against the most likely intent. Thus, the contingent plan can ensure safety only when the ego vehicle acquires a clear understanding of the intentions of other agents by the time of branching. However, in these works, the probability associated with the uncertain intents of the dynamic agents is assumed to be fixed, and the planning is executed in an *open-loop* fashion. Thus, an exact estimation of the branching time t_b is required, otherwise a collision may occur [17]. Learning-based approaches have been recently introduced in the contingency planning context [37], [38], however, they are not interpretable and hard to tune. [66] proposes a contingency planning approach that uses a dynamic branching point determined by a predefined heuristic. This heuristic chooses the branching time as the maximum time such that any two future scenarios starting at the current time only diverge by a maximum distance.

Despite being effective, this heuristic entails at least double the computational time since the divergence measure is invoked on the ego-vehicle trajectories.

In this paper, we adopt the idea of splitting the planning problem into short-term and long-term planning.

2) Risk-Aware Motion Planning: Another aspect to consider when planning under uncertainty is to assess the risk associated with the executed plan [55], [56], [57], [58], [59]. In [55], Gaussian process regression is utilized to establish a probabilistic model of the environment. This model is subsequently employed to formulate a risk-aware cost function using the Conditional Value at Risk (CVaR) measure, which is then incorporated into an optimal motion planning algorithm to generate trajectories that avoid high-risk areas. [57] constructs a probabilistic risk map by assessing the anticipated harm considering predicted spatio-temporal trajectories for both the ego-vehicle and other participants in the traffic scene. These maps serve as indicators of the risk associated with a planned trajectory, calculated through a rapidly-exploring random tree algorithm. Despite being effective, a drawback of this approach is that the driven trajectories of the other traffic participants need to be pre-defined and known by the ego-vehicle a priori. Along the same line as our proposed approach, [58], [59] define risk as a product of two components, the probability of collision with other traffic subjects and the severity of that potential collision. In that sense, an analytic approach is proposed to calculate the probability of spatial overlap of the ego-vehicle with dynamic obstacles at discrete times.

B. Statement of Contributions

In this work, we aim to address the state-of-the-art aforementioned limitations. In particular, the contributions of this paper can be summarized in the following points:

  i) In contrast to recent contingency planning schemes that assume an open-loop information structure, we propose a Bayesian update scheme that incorporates the observations of the human states into the motion planner cost function, influencing the short-term plan based on the belief the ego-vehicle maintains over human intentions.
 ii) We incorporate a probabilistic risk metric into the contingency planner to balance safety and efficiency.
iii) We analyze the effect of branching time and the belief over the obstacles' intents on the short-term plan, and how they relate to the maximum risk the ego-agent endures.
 iv) We show how the proposed approach can be extended to multi-agent scenarios by leveraging the permutations over all possible intentions the traffic agents can have.

III. PRELIMINARIES

In this section, we first introduce the principal tools that constitute the proposed contingency planning framework.

A. Notation

Throughout this paper, vectors, and matrices are expressed in bold, $\boldsymbol{x}$, and capital bold, $\boldsymbol{A}$, letters respectively. $||\boldsymbol{x}||$ is the Euclidean norm of $\boldsymbol{x}$, and the subscript $(\cdot)_k$ indicates the value
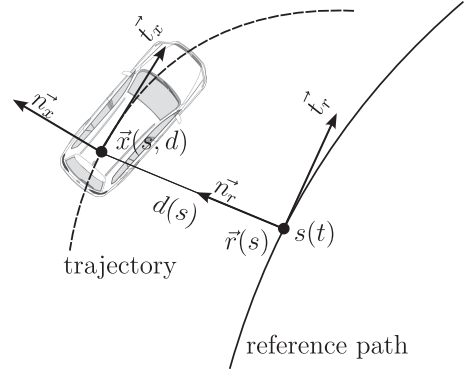

Fig. 2.   Illustration of the Frenet coordinate system.

at stage k, $f(\cdot)$ is the probability density function. The planning problem is formulated in a receding horizon fashion where only the first control input is executed, and then the whole process is reiterated with the new initial conditions and observations.

B. Ego-Motion Sampler

The general form of the planning problem can be formulated as follows:

$$\tau^* = \arg\min_{\tau \in \mathcal{T}_n} J(\tau; \Gamma), \tag{1a}$$

$$\text{s.t.} \quad g_j(\tau) \leq b_j, \quad j = 1, \ldots, n \tag{1b}$$

where the optimal trajectory τ^* is defined as the one giving the minimum total cost $J(\tau)$ from a set of sampled trajectories $\mathcal{T}_n$ given the ego-state. $\tau \in \mathcal{T}_n$ is a continuous path through the state space and is characterized by a sequence of points $\tau = \{\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ defined over a horizon of length N with regular intervals Δt, where $\boldsymbol{x}_k = (x_k, y_k)$. The optimal trajectory must adhere to a set of time-dependent constraints $g_j(\tau)$ imposed by the surrounding dynamic obstacles, vehicle kinematics, and other user-defined constraints.

Instead of formulating the problem directly in the Cartesian coordinate system, we switch to the *Frenet Frame* to exploit the lane-geometric information. In that sense, the trajectory is parameterized by the total arc-length $s(t)$ traveled along the reference path Γ parameterized by time t, and the orthogonal lateral deviation $d(s)$ parameterized by arc-length s as shown in Fig. 2. In this paper, we adopt a sampling-based motion planning approach that is widely used in the intelligent vehicles community [39], [40], [41], [42], [43], [44].

To generate a set of possible candidate trajectories, we first need to sample a set of terminal states for both longitudinal and lateral trajectories. To ensure the diversability of the candidate trajectories, it is crucial to emphasize that the sampled terminal states should cover various maneuvers which include maintaining the current velocity, accelerating to a certain speed, yielding velocity profiles for the longitudinal trajectories, and lane-keeping, lane-change, and nudging for lateral trajectories. Given the current ego-state, with respect to the reference path, and the sampled terminal states, piecewise quartic and quintic polynomials can be used to generate the longitudinal and lateral trajectories respectively [39]. The Frenet state defined as

$[s, \dot{s}, \ddot{s}, d, d', d'']$ can then be converted to the global coordinates $[x, y, \theta, \kappa, v, a]$, where $(\dot{\cdot}) := \frac{\mathrm{d}(\cdot)}{\mathrm{d}t}$, and $(\cdot)' := \frac{\mathrm{d}(\cdot)}{\mathrm{d}s}$ indicate the parameters derivatives with respect to time and arc-length respectively.

After generating a set of trajectories, some of them are then pruned based on some imposed constraints $g(\tau)$. These constraints are affected by the kinodynamic feasibility of the ego-vehicle, in addition to collision avoidance constraints concerning the surrounding obstacles. Since the motion of the dynamic obstacles is not known a priori, this necessitates the need for a probabilistic prediction model that models their behavior forward in time which is described in Subsection III-C. After pruning the invalid trajectories, a user-defined cost function $J(\tau)$ is assigned to each valid trajectory, and the trajectory τ with the minimum cost is selected. The details of $g(\tau)$ and $J(\tau)$ formulations are given in Section V.

C. Probabilistic Prediction Model

The planner is required to stochastically forecast other agents' behavior to make informed decisions. This necessitates the need for a prediction model that can effectively capture the diverse and nuanced intentions exhibited by different drivers. For instance, a neighboring vehicle may opt to either remain in its lane or merge in front of us. To address the inherent uncertainty and complexity associated with the multi-modal behavior of road users, the application of Mixture-of-Gaussians (MoG) distributions is widely used in literature, see e.g., [46], [47]. It is important to highlight that other multi-modal prediction methods could also be used in conjunction with our proposed approach. In our settings, there exists a set of dynamic obstacles $o \in \mathcal{I}_o := \{1, \dots, n_o\}$, at position $\boldsymbol{\delta}^o \in \mathbb{R}^2$. The probability measure associated with the uncertainty of the perception of the dynamic obstacles is denoted by $\mathbb{P}$ and defined over the probability space Δ. An MoG model serves as a comprehensive method for articulating uncertainty characterized by multiple modes, achieved through the integration of multiple continuous probability distributions,

$$f_k^o(x, y) = \sum_{i=1}^{n} \phi_i f_{k,i}^o(x, y), \tag{2}$$

where n is the number of modes of the MoG, ϕ_i represents the weight of each mode such that $\sum_{i=1}^{n} \phi_i = 1$, and $f_{k,i}^o(\cdot)$ is the probability density function of each mode with mean $\boldsymbol{\mu}_i \in \mathbb{R}^n$ and covariance $\boldsymbol{\Sigma}_i \in \mathbb{R}^{n \times n}$,

$$f_{k,i}^o(\cdot) = \mathcal{N}(\boldsymbol{\mu}_i(t), \boldsymbol{\Sigma}_i(t)), \tag{3}$$

The output of the prediction model is subsequently a sequence of predicted state distributions along the prediction horizon for a given mode. This will be later leveraged to calculate the risk associated with the planned trajectories as detailed in the following subsection.

D. Collision Chance Constraints

The ego-vehicle v and the obstacle o are mutually collision-free if $||\boldsymbol{x}_k^v - \boldsymbol{\delta}_k^o|| \leq r$, where $\boldsymbol{x}^v, \boldsymbol{\delta}^o \in \mathbb{R}^2$ denote the positions of the ego-vehicle and obstacle respectively, and r is the safety distance. However, since the positions of the obstacles are defined as random variables, the collision avoidance constraints can only be satisfied in a probabilistic manner, and thus defined as chance constraints at each timestep k,

$$\mathbb{P}(||\boldsymbol{x}_k^v - \boldsymbol{\delta}_k^o|| \leq r) \leq 1 - \delta \tag{4}$$

where $\mathbb{P}$ indicates the probability measure, and $\delta \in (0, 1]$ is the collision probability threshold.

IV. PROBLEM FORMULATION

In traditional motion planning frameworks, a single trajectory is sought to minimize the expected cost over all plausible predicted futures along the entire planning horizon [50], which may result in sub-optimal and overly-conservative trajectories. The contingency planning paradigm, on the other hand, generates a distinct set of trajectories conditioned on the different outcomes from the prediction model.

A. Contingency Planning

Due to the stochastic nature of the surrounding agents' intentions, contingency planning outputs multiple policies Π where each policy $\pi \in \Pi$ is specified for a single agent's intent $\tilde{\lambda}$. Given the ego-vehicle's incapacity to concurrently traverse multiple contingency plans, the initial segment of each plan $\tau_{0:t_b}$ is restricted to remain consistent. The contingency planning problem can be formulated as,

$$\arg \min_{\tau} J_{\text{shared}}(\tau_{0:t_b}) + \sum_{\lambda \in \Lambda} p(\lambda) J_{\text{conting}}(\tau_{t_b:T}, \lambda) \tag{5a}$$

$$\text{s.t.} \quad g_j(\tau) \leq b_j, \quad j = 1, \dots, n \tag{5b}$$

where J_{shared} is the cost of the shared part of the plan that takes into consideration all possible modes of the prediction model, t_b is the branching time representing the time at which the shared plan bridges into different contingent plans, J_{conting} is the cost associated with each contingent plan, and $p(\lambda)$ indicates the probability of each possible intention given by the prediction model introduced in Section III-C.

However, with this problem formulation, the following challenges arise:

i) It is assumed that by t_b, the uncertainty about the other agents' intentions is resolved, and the ego-vehicle branches to the predicted true hypothesis. Thus, t_b has to be calculated accurately, otherwise, the ego-vehicle may choose the wrong branch which can result in a collision.

ii) The hypothesis probabilities estimated by the prediction model, $p(\lambda)$, are usually of a bad quality and entirely relying on them could result in collisions [48].

In [17], [18], [19], an offline reachability analysis approach is proposed to alleviate the first issue by estimating the branching time t_b. However, this approach requires a discretization of the state space and takes into consideration the worst behavior of the other agents which leads to the worst-case estimate of t_b, that is the latest time at which the ego-vehicle becomes certain about the other obstacle's intention. Moreover, due to the curse of dimensionality, this approach can barely extend to multi-agents.

Therefore, in our proposed approach, introduced in the upcoming section, we tackle the first issue by introducing a belief updater that updates the ego-vehicle's prior belief about obstacles' intentions based on the online measurement it perceives. Moreover, to address the inherent trade-off between safety and efficiency, we introduce risk-aware contingency planning by augmenting contingency planning with a probabilistic risk measurement.

V. PROPOSED APPROACH

A. Planning Under Uncertain Intentions

To alleviate the conservatism of the planned trajectory, we propose to have a probabilistic inference over the possible intents that an obstacle can have by introducing a Bayesian belief updater instead of solely relying on the prediction model to make an informed estimate of the unknown intent.

1) Belief Updater: At each time step, we assume that the ego-vehicle can observe the true state $\hat{\boldsymbol{\delta}}_t^o$ of the dynamic obstacle, but not its internal state. This enables the ego-vehicle to retrospectively assess the likelihood of the dynamic obstacle's observed states under the prediction model. Thus, the ego-vehicle always maintains a belief, $b(\lambda)$, over the set of possible intents Λ of other agents, i.e., $\lambda \in \Lambda$. A *Bayesian filter* is used to update the ego vehicle's belief over the obstacle's intents based on the new observations the ego-vehicle perceives. The update rule for the obstacle's intent is given by,

$$b(\lambda)_+^t = \frac{f(\hat{\boldsymbol{\delta}}_t^o; \boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)b(\lambda)_-^t}{\sum_{\tilde{\lambda}} f(\hat{\boldsymbol{\delta}}_t^o; \boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)b(\tilde{\lambda})_-^t} \quad (6)$$

where $b(\tilde{\lambda})_-^t$ represents the prior belief of the ego-vehicle on the intent $\tilde{\lambda}$ at time t. $b(\tilde{\lambda})_+^t$ indicates the same probability a posteriori. $f(\hat{\boldsymbol{\delta}}_t^o; \boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)$ denotes the probability density function of a single mode of the MoG, given by the prediction model defined in Section III-C, with mean $\boldsymbol{\mu}_t$ and covariance Σ_t, and evaluated at the observed state $\hat{\boldsymbol{\delta}}_t^o$.

Remark 1: Based on the provided prediction model and what the ego-agent can observe, the probability density function $f(\cdot)$ can either be defined on the obstacle's state or the control input.

2) Multi-Agent Scenario: In a multi-agent setting, it is not sufficient to only consider the belief that the ego-agent maintains over a single agent's intentions, but rather to consider how the traffic scene would evolve as a whole into the future. To address this issue, it is required to consider all the permutations $\Theta = \{\theta_1, \ldots, \theta_{n_s}\}$ where n_s is the total number of realizations the traffic scene can evolve to, and Θ is determined by the Cartesian product of all obstacles policies, $\Theta = \Lambda_1 \times \ldots \times \Lambda_{n_s}$. The total number of realizations, n_s, is defined by the cardinality of Θ, $n_s = |\Theta|$. The probability of each realization can, subsequently, be calculated by,

$$p(\theta_j)_+ = \frac{p(\theta_j)_- \prod_{i=1}^{n_o} b_i(\lambda)_+^t}{\sum_{j=1}^{n_s} p(\theta_j)_- \prod_{i=1}^{n_o} b_i(\lambda)_+^t} \quad (7)$$

where $b_i(\lambda)_+^t$ is determined by (6), and λ is the corresponding intention for obstacle i that belongs to θ_j. $p(\theta_j)_+$ is used as a weight for the contingent plans in the cost function defined

in (13a). This mimics a scene-centric prediction model by outputting modes of joint trajectories with respect to all agents in the scene.

3) Probabilistic Risk Assessment: In our proposed approach, to achieve probabilistic collision avoidance, we rely on an existing *risk metric*, motivated by our previous work [54], that maps the distribution of a random variable, defined in (4), to a real number.

Definition 1 (Risk Metric): Let $\mathcal{Z}$ denote the set of random variables representing the uncertainty of the obstacles' motion in the x and y directions. The risk metric maps the distribution of the random variables to a real number indicating the induced risk, $\mathcal{R}: \mathcal{Z} \mapsto \mathbb{R}$.

There exist various definitions of risk in the literature. Among them, safety standards, [51], [52] define risk as a combination of the probability of occurrence of harm and the severity of that harm. Inspired by this definition, different approaches in robotics and autonomous driving community [53], [54], [58], [60] define risk as the product of two quantities, namely the probability of collision that the planned trajectory has with any of the surrounding obstacles and the level of severity linked to that potential collision at every time-step, k, along the planning horizon,

$$\mathcal{R}_k^o(\boldsymbol{x}_k^d) = \mathcal{C}_k^o(\boldsymbol{x}_k^d)\mathcal{S}_k^o(\boldsymbol{x}_k^d), \quad \forall k, o, d \quad (8)$$

Here, we approximate the ego-vehicle by two discs, where each disc is referred to by d, and calculate the probability of collision for each disc. Note that we model the ego-vehicle as two discs such that we can consider all its shape in calculating the induced risk. Given the probability density function of the prediction model as indicated in (2), the probability of collision, for each ego-vehicle's disc ate each planning stage k per obstacle o, can be calculated by estimating the spatio-temporal overlap of the predicted modes with the ego-vehicle's plan τ,

$$\mathcal{C}_k^o(\boldsymbol{x}_k^d) = \iint_{x_k^d, y_k^d \in D} f_k^o(x, y)dxdy, \quad \forall k, o, d, \quad (9)$$

which is an integral of the Mixture of Gaussians (MoG) probability density function over a specified domain where the integration domain D is defined as a circle whose center is located at the predicted vehicle pose $\boldsymbol{x}_k^d$ at stage k along the prediction horizon, and its radius r is the sum of the vehicle and obstacle radii since the PDF does not account for the obstacle-vehicle's shape.

Another aspect of the risk assessment is the determination of the severity of a potential collision for each planned trajectory. The expected collision severity is of high importance in case a collision is inevitable and the best behavior has to be selected to reduce upcoming damage. The collision severity definition is motivated by the work proposed in [53], [58] which can be determined, $\forall k, o, d$, by

$$S_k^o(\boldsymbol{x}_k^d) = \frac{\text{m}^v}{\text{m}^v + \text{m}^o}((v_k^v)^2 + (v_k^o)^2 - 2v_k^v v_k^o \cos\alpha)^{\frac{1}{2}} \quad (10)$$

where m^v and m^o represent the masses of the ego-vehicle and obstacle vehicle respectively whereas v^v and v^o are their corresponding velocities, and α is the collision angle. Here, it is

important to emphasize that establishing an appropriate severity measure is a challenging problem and constitutes a dedicated area of research that is beyond the scope of this paper. Notably, the severity metric outlined here does not account for ethical considerations associated with the resultant damage, including the vulnerability of road users. This can, nevertheless, be embedded in (10), in case semantic information of the road users is provided by a perception model. In this context, leveraging semantic categorization allows for the scaling of the severity metric within a range of 0 to 1. A severity value of 0 corresponds to a collision causing no harm, while a severity value of 1 signifies the highest level of damage, particularly involving vulnerable road users. This scaled metric can then be incorporated into the sampling-based planner to discard trajectories that lead to collisions with vulnerable road users.

Furthermore, we adopt a discounted chance constraint formulation, as proposed by [61], where violation probabilities close to the early stages of the plan have higher penalization compared to the ones in the far future. The discounted risk at every stage k concerning an obstacle o is defined as

$$\mathcal{R}_k^o(\boldsymbol{x}_k^d) = (\gamma)^k \mathcal{C}_k^o(\boldsymbol{x}_k^d) \mathcal{S}_k^o(\boldsymbol{x}_k^d), \quad \forall k, o, d \qquad (11)$$

where $\gamma \in (0, 1]$ is the discounting factor.

After calculating the risk for each ego-vehicle's disc at every timestep k along the planning horizon per each obstacle o, the predicted risk η at the current time step is defined by maximizing risk for all obstacles, $o \in \mathcal{I}_o$, at all planning horizon stages k and all discs d,

$$\eta = \max_{o \in \mathcal{I}_o, k, d} \mathcal{R}_k^o(\boldsymbol{x}_k^d), \qquad (12)$$

The max operator in (12) ensures that the worst-case risk over the planned trajectory is below the threshold δ.

Remark 2: According to [62], the quantified risk metric given in (11) is non-coherent since it does not fulfill all coherence axioms. Nevertheless, it is efficient in capturing the underlying uncertainty associated with the motion of the dynamic obstacles and scales monotonically as the level of risk increases. Here it should be emphasized that, since our planner is sampling-based, it is agnostic to the risk metric employed and a comparative study on different risk metrics will be part of future research.

Thus, the optimization problem can be formulated as follows,

$$\min_{\tau \in \mathcal{T}} \quad J_{\text{shared}}(\tau_{0:t_b}) + \sum_{j=1}^{n_s} p(\theta_j) + J_{\text{conting}}(\tau_{t_b:T}(\theta_j)) \qquad (13a)$$

$$\text{s.t.} \quad g_j(\tau) \le b_j, j = 1, \ldots, n \qquad (13b)$$

$$p(\theta_j)_+ = \frac{p(\theta_j)_- \prod_{i=1}^{n_o} b_i(\lambda)_+^t}{\sum_{j=1}^{n_s} p(\theta_j)_- \prod_{i=1}^{n_o} b_i(\lambda)_+^t}, \qquad (13c)$$

$$\max_{o \in \mathcal{I}_o, k, d} \mathcal{R}_k^o(\boldsymbol{x}_k^d) \le \delta \qquad (13d)$$

It is worth mentioning that the number of planned contingency trajectories is determined by the number of modes of the prediction model, $|\Lambda|$.

4) Cost Function: The cost function $J(\tau)$ consists of subcosts that focus on different aspects of the plan's performance

such as safety, passenger comfort, progress, and tracking. It is, thus, defined as,

$$J(\tau) = \boldsymbol{w}^T c(\tau; \Gamma) \qquad (14)$$

where the weight vector $\boldsymbol{w} \in \mathbb{R}_+$ captures the weights associated with each cost term. These costs include, for instance, how much the final point of the planned trajectory deviates from the reference path $c_d = |d - d_{\text{ref}}|$, $c_v = |\dot{s} - \dot{s}_{\text{ref}}|$ is the deviation from the reference velocity, $c_p = \sum_{k=0}^{N-1} ||\boldsymbol{x}_{k+1}^v - \boldsymbol{x}_k^v||$ is the cost representing progress, i.e., the total traveled distance along the reference path Γ, and $c_j = \sum_{k=0}^{N} \frac{\dot{s}_k^{:2} + d_k'''^2}{N}$ penalizes the mean square sum of longitudinal and lateral jerks used as a comfort indicator.

5) Kinematic Constraints: To guarantee a smooth and comfortable transition between the short-term shared plan and the contingency plans, when concatenated, we need to impose some constraints on the curvature at the branching point. From [43],

$$d'' = -[\kappa_r d]' \tan\theta + \frac{1 - \kappa_r d}{\cos^2 \theta}\left[\kappa_p \frac{1 - \kappa_r d}{\cos\theta} - \kappa_r\right], \quad (15)$$

where κ_r and κ_r' denote the curvature of the reference path Γ and its derivative respectively. By rearranging terms, an explicit formulation of the trajectory's curvature can be obtained,

$$\kappa_p = [d'' - (\kappa_r d)' \tan\theta] \frac{\cos^3 \theta}{(1 - \kappa_r d)^2} + \frac{\kappa_r \cos\theta}{1 - \kappa_r d} \qquad (16)$$

After calculating the curvature and every point along the trajectory, a box constraint is defined as

$$|\kappa_p| \le \kappa_{\max} \qquad (17)$$

where $\kappa_{\max}$ at every timestep is parameterized by the planned velocity of the ego-vehicle and the maximum allowable lateral acceleration. Finally, to ensure that the planned trajectory is kinematically feasible, the following box constraints must be satisfied at every timestep,

$$v_{\min} \le v(t) \le v_{\max} \qquad (18a)$$

$$a_{\min} \le a(t) \le a_{\max} \qquad (18b)$$

$$j_{\min} \le j(t) \le j_{\max} \qquad (18c)$$

where $v_{\min}/v_{\max}$ represent the minimum and maximum velocity, $a_{\min}/a_{\max}$ represent the minimum and maximum acceleration, and $j_{\min}/j_{\max}$ represent the minimum and maximum jerk.

B. Planner Design

To better explain the proposed approach, we recall the illustrative example in which the ego-vehicle is not fully aware of the future trajectories of another agent or even its intended goal, as illustrated in Fig. 4. In this example, we take into consideration only two likely futures. The one depicted in blue is where the other agent yields to the ego-agent, and the one in red is where the ego-agent must brake before entering the intersection since the oncoming vehicle is taking an aggressive left turn.

The proposed approach is summarized in Algorithm 1:
  i) In lines (2–8), we sample a large set of shared plans $\mathcal{T}_{0:t_b}$ using the ego-motion sampler explained in section III-B.

(a) Sampling short-term trajectories till t_b.

(b) Sampling long-term trajectories till T.
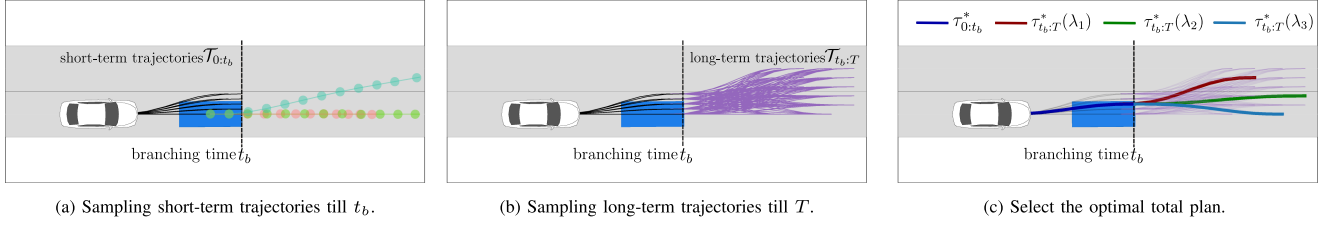
(c) Select the optimal total plan.

Fig. 3. Snapshots from the CommonRoad simulation framework for an overtaking scenario where the ego-vehicle is depicted by the car icon and the obstacle-vehicle is illustrated as a blue rectangle. The obstacle-vehicle has three policies that it can execute, $\Lambda = \{\texttt{maintainspeed}, \texttt{slowdown}, \texttt{lanechange}\}$, where the obstacle-vehicle's predicted trajectories are illustrated in distinct colors in 3(a). We first sample a set of short-term plans, $\mathcal{T}_{0:t_b}$ visualized in black, till the branching time t_b shown in 3(a). We then sample a set of long-term plans, $\mathcal{T}_{t_b:T}$ depicted in purple, conditioned on the terminal states of the short-term plans, shown in 3(b). For the long-term plans, a cost is assigned to each plan per obstacle-vehicle policy λ_i. In Fig. 3(c), the total cost of the entire plan is computed by (13a), and the total optimal plan is the one given by $\tau^* = \tau^*_{0:t_b} \cup \{\tau^*_{t_b:T}(\lambda_1), \tau^*_{t_b:T}(\lambda_2), \tau^*_{t_b:T}(\lambda_3)\}$. In this example, the belief over the first policy is dominant, $b(\lambda_1) = 0.63$, compared to the rest causing the optimal short-term plan $\tau^*_{0:t_b}$ to be more biased towards $\tau^*_{t_b:T}(\lambda_1)$.
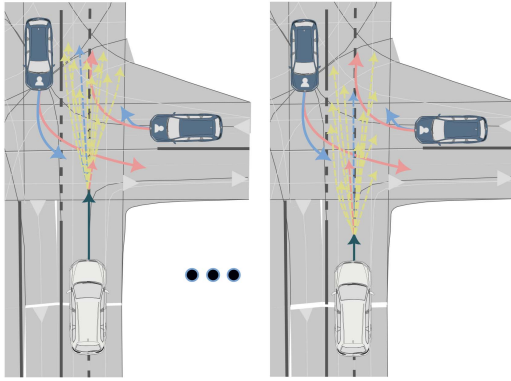

Fig. 4. Contingency planning paradigm. A large set of short-term plans are sampled together with multiple long-term plans. The total plan with the minimum cost for each possible future realization is selected (indicated in blue and red).
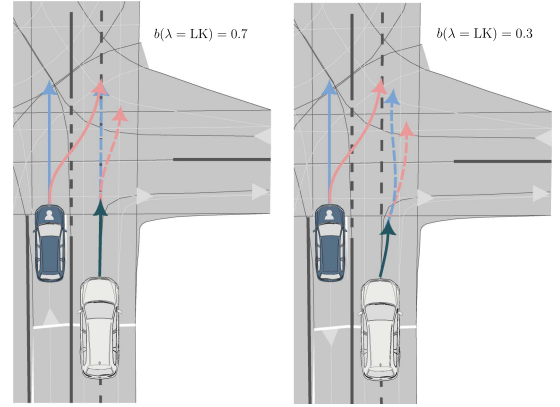

Fig. 5. The human-driven vehicle can have two different policies, lane-keeping (LK) or lane-change (LC). On the left, the ego-vehicle has a higher belief that the human-driven vehicle executes the LK policy. That's why the short-term trajectory tends to accelerate. On the right, the ego-vehicle has a higher belief that the human-driven vehicle executes the LC policy causing the short-term plan to decelerate and steer to the right.

This is followed by pruning trajectories that violate the risk upper limit or kinematic constraints. Here, the risk analysis is done with respect to all prediction modes.

ii) Line 9 iterates over the number of modes coming from the prediction model.

iii) In lines (10–21), for each shared plan, conditioned on its end state, we sample a set of long-horizon trajectories $\mathcal{T}_{t_b:T}$. A risk analysis is performed for each long-term trajectory with respect to a single mode from the prediction model, and trajectories that violate constraints are pruned. A cost is assigned to each trajectory according to (14).

iv) Lines (22–23) sort the long-term trajectories based on their associated costs and pick the trajectory with the minimum cost. The belief over the prediction mode, we iterate over, is updated using (6).

v) In lines (25–27), the shared-trajectory is concatenated with all contingent plans. The total cost of the entire plan is computed by the expected cost of the contingent plans in addition to the cost of the shared plan itself as indicated in (13a).

vi) We find the optimal response, for each shared plan, associated with each scenario from the corresponding long-horizon trajectories in line 30.

This process is then repeated at every time step in a receding horizon manner.

Remark 3: In case no valid trajectory is obtained from the set of candidate trajectories $\mathcal{T}$, we apply the trajectory with the least risk as long as it is dynamically feasible.

Here it is crucial to emphasize that an advantage of using a sampling-based planner in the Frenet frame as a basis for our contingency planning framework is that the sampled trajectories cover various maneuvers that the ego-vehicle can have. Thus, in contrast to the approaches proposed in [30], [31], [32], our approach eliminates the necessity for a pre-established trajectory tree or branching topology to formulate contingent plans. Subsequently, the ego-vehicle is not confined to a specific set of predefined policies. This concept is further explained in Fig. 3 where we show how the optimal plan is selected from the sampled short-term and long-term trajectories. Additionally, it illustrates how the short-term plan is shaped based on the belief the ego-vehicle maintains over the obstacle-vehicle policies. Fig. 5 shows how the belief the ego-vehicle maintains over the long-term plans affects the behavior of the short-term plan. It is observed that when the ego-vehicle has a higher belief in one of the human's intents, the shared plan tends to be biased towards

---

Algorithm 1: Risk-Aware Contingency Planning.

Input: Prediction model per obstacle $f_{k,i}^o(\cdot)$, horizon T, branching time t_b, risk tolerance level δ, reference path Γ, prior belief per intention λ, $b(\lambda)_- = \frac{1}{|\Lambda|}$, two priority queues for sorting candidate trajectories Q, Q_{final}.

Output: An optimal shared plan $\tau_{0:t_b}$, concatenated with $|\Lambda|$ contingent plans $\mathcal{T}_{t_b:T}$.

1: **for all** $t = 1, 2, \ldots$ **do**
2: $\mathcal{T}_{0:t_b} \leftarrow$ SampleSharedTrajectories($\boldsymbol{x}_{\text{init}}$).
3: **for each** $\tau_{0:t_b} \in \mathcal{T}_{0:t_b}$ **do**
4:   $\eta \leftarrow$ EvaluateRisk($\boldsymbol{x}_k^d, \boldsymbol{\delta}_k^v$), $\forall k, v, d$ (12).
5:   **if** $\eta \geq \delta$ **then**
6:    **continue**
7:   **end if**
8:   $J_{\text{shared}}(\tau_{0:t_b}) \leftarrow$ ComputeCost($\boldsymbol{w}, \tau_{0:t_b}, \Gamma$) (14).
9:   **for each** $\lambda \in \Lambda$ **do**
10:    $\mathcal{T}_{t_b:T} \leftarrow$ SampleContingentPlans($\tau_{0:t_b}$(end)).
11:    $Q = \{\}$.
12:    **for each** $\tau_{t_b:T} \in \mathcal{T}_{t_b:T}$
13:     $\eta \leftarrow$ EvaluateRisk($\boldsymbol{x}_k^d, \boldsymbol{\delta}_k^v, \lambda$), $\forall k, v, d$.
14:     **if** $\eta \geq \delta$ **then**
15:      **continue**
16:     **end if**
17:     $J_{\text{cont}}(\tau_{t_b:T}) \leftarrow$ ComputeCost($\boldsymbol{w}, \tau_{t_b:T}, \Gamma$).
18:     **if** $\tau_{t_b:T}$ passed constraint check **then**
19:      add $\tau_{t_b:T}$ to Q.
20:     **end if**
21:    **end for**
22:    $\tau_{t_b:T}(\lambda)_{\text{best}} \leftarrow$ pop the first candidate from Q.
23:    $b(\lambda)_+ \leftarrow$ UpdateBelief($f_{k,i}^o(.), b(\lambda)_-$) (6).
24:   **end for**
25:   $\tau = \tau_{0:t_b} \cup \{\tau_{t_b:T}(\lambda_1)_{\text{best}}, \ldots, \tau_{t_b:T}(\lambda_{|\Lambda|})_{\text{best}}\}$.
26:   $p(\theta_j)_+ \leftarrow$ UpdatePermutations($b_i(\lambda)_+$) (7).
27:   Evaluate total cost using (13a).
28:   add τ to Q_{final}.
29: **end for**
30: $\tau_{\text{best}} \leftarrow$ pop the first candidate from Q_{final}.
31: **end for**

---

the corresponding contingent plan. In particular, as illustrated in Fig. 5-left, when the ego-agent has a higher belief that the human-driven vehicle aims at executing the lane-keeping policy, the shared-plan biases its motion to accelerate along its lane. On the other hand, when the higher belief is assigned to the lane-change maneuver, the shared-plan tends towards steering a bit to the right while decelerating. This is achieved by balancing the cost of the shared-plan according to the likelihood of the beliefs.

Remark 4: It should be pointed out that, in the human-driven vehicle's lane-keeping policy, we still generate a contingent plan for the other possible intent after the branching point, allowing the ego-vehicle to smoothly steer to the right while decelerating in case the human-driven decides to execute the lane-change policy.

VI. RESULTS

A. Experimental Setup

We evaluate our approach in two safety-critical simulated scenarios inspired by autonomous driving interactions. The illustrated scenarios are included in the CommonRoad benchmark suite [63] for reproducibility. The first scenario highlights the reactive behavior that the ego-vehicle's plan induces on an obstacle-vehicle in an overtaking scenario. Four different baselines are introduced to compare our proposed contingency planning with.

Baseline 1: Multi-policy planning [30], this baseline uses a branch-MPC whose objective is to minimize the expected cost across all branches within a trajectory tree.

Baseline 2: Robust baseline [45] that optimizes a single trajectory along the planning horizon that is robust with respect to all predicted modes regardless of their probabilities.

Baseline 3: Maximum-likelihood estimate [50], that only considers the most probable mode given by the prediction model while ignoring the rest.

Baseline 4: Similar to [15], we use the mode probabilities, provided by the prediction model, directly in the contingency planning cost function instead of the beliefs obtained from the belief updater. This baseline is used to analyze the effect of the Bayesian belief updater on the contingency planner's performance.

In contrast to the branch-MPC approach [30] which encounters scalability challenges when addressing multiple obstacle vehicles, primarily due to its exponential complexity requiring a pruning protocol, our proposed approach exhibits seamless adaptability to multi-vehicle scenarios. This is illustrated in the T-junction, and intersection scenarios, where the ego-vehicle interacts with multiple vehicles whose intents are not known to the ego-vehicle a priori.[1] The computer running the simulations is equipped with an Intel Core™ i7 CPU@2.6 GHz.

B. Scenario 1. Overtaking in Highway Driving:

In the first scenario, an autonomous vehicle seeks to initiate a lane change maneuver, by overtaking the obstacle vehicle in the designated lane, while grappling with its level of uncertainty.

1) Obstacle-Vehicle Model: As a benchmark, we compare our approach to the branch-MPC introduced in [30]. For this purpose, the same prediction model is leveraged in which it is assumed that the obstacle vehicle has three different policies that it can execute $\Lambda = \{\texttt{maintainspeed}, \texttt{slowdown}, \texttt{lanechange}\}$ where the direction of the lane change is towards the left lane. The output of the prediction model is represented as a scenario tree, as depicted in Fig. 6, in which it is assumed that the obstacle-vehicle can change its policy the next time step or after 8 steps along the horizon. In the meantime, the obstacle vehicle maintains its policy. The obstacle vehicle trajectories are constructed by forward propagating its dynamics with respect to the selected policy where the vehicle dynamics are modeled using the kinematic bicycle model [64]. The probability of

---

[1]A video of the simulated experiments accompanies this paper.

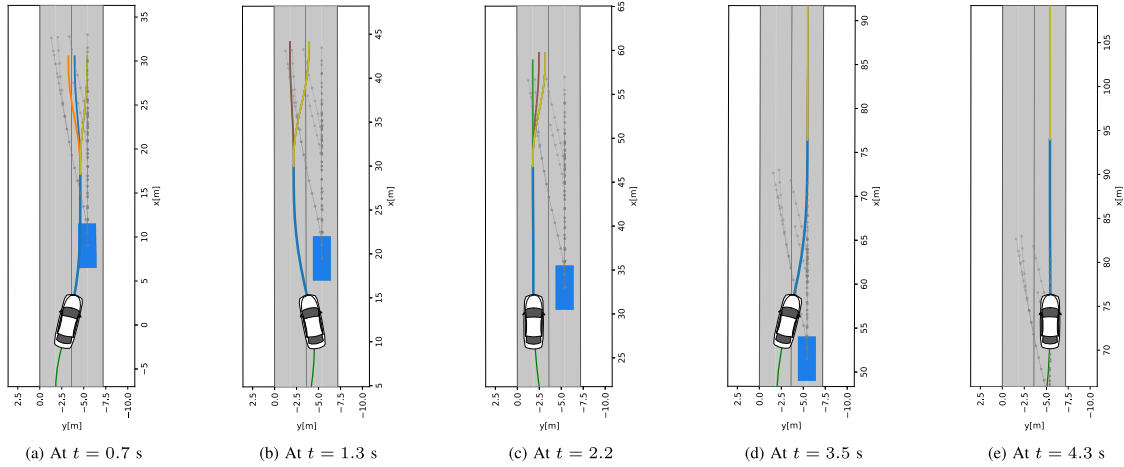(a) At $t = 0.7$ s   (b) At $t = 1.3$ s   (c) At $t = 2.2$ (d) At $t = 3.5$ s   (e) At $t = 4.3$ s

Fig. 6. Snapshots from the overtake simulated environment in CommonRoad at different time instants. The obstacle vehicle is represented by a blue rectangle. The driven trajectory by the ego-vehicle is depicted in green. The short-term planned trajectory is illustrated in blue, whereas contingent plans are depicted in distinct colors. The predicted scenario tree of the obstacle vehicle is represented in gray.
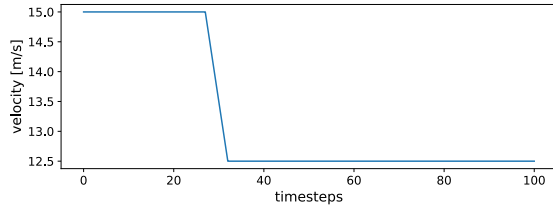

Fig. 7. The evolution of the obstacle-vehicle's velocity with time.

Fig. 8. Belief evolution over obstacle vehicle's policies along the timesteps. b_1, b_2, b_3 represent maintain fixed speed, slow down, and lane-change policies, respectively. The gray vertical lines denote the time instants at which the snapshots in Fig. 6 are taken.

executing each policy is calculated by introducing a collision avoidance measure, $\xi(\tau, \lambda_i \in \Lambda)$, that determines the collision probability that the obstacle vehicle has, under each policy, with respect to the ego-vehicle's planned trajectory. The policy λ_i with the least collision probability will have a higher probability of being executed by the obstacle vehicle where each probability is defined by a softmax function as described in [30]. It is important to emphasize that although the policy of the obstacle vehicle is influenced by the planned trajectory of the ego-vehicle, the responsibility of preventing collisions rests solely upon the ego-vehicle.

2) Environment Setup: To guarantee a fair comparison, for all approaches, we set the maximum speed, and maximum allowed acceleration to the same value which are $20\,\text{m}\,/\,\text{s}, 4\,\text{m/s}^2$ respectively. The upper bound of the induced risk is assigned to $\delta = 5\%$, for the baselines and the proposed approach, which was found to provide a good balance between safety and efficiency. A horizon of $N = 16$ steps is defined, with a discretization step of 0.2 s, resulting in a time horizon of 3.2 s, and a branching time $t_b = 1.2$ s is defined.

3) Qualitative Results: Fig. 8 shows the evolution of the ego-vehicle's belief over the obstacle-vehicle policies over time. At the beginning, the ego-vehicle reveals its lane-change intention by swerving into the obstacle's vehicle lane. Due to the reactive behavior of the obstacle-vehicle, the probability of it executing a lane-change in the ego-vehicle's lane drops.

However, the ego-vehicle could not complete the lane change since no valid trajectory is obtained that does not violate the
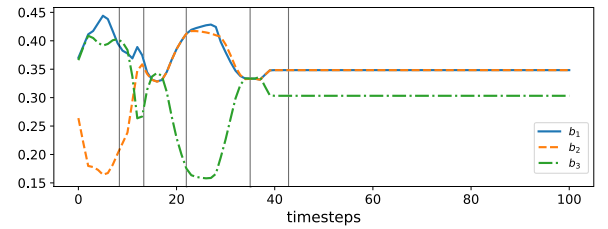
safety constraints, and thus returns to its original lane. Since the lane-change policy of the obstacle-vehicle, b_3, becomes relatively low, the ego-vehicle then initiates another attempt to overtake the obstacle vehicle which probes it to decelerate allowing the ego-vehicle to complete the lane change. To visualize how the obstacle-vehicle's policy changes with the timesteps, Fig. 7 shows the evolution of the obstacle-vehicle's velocity with the timesteps. As depicted, the ego-vehicle exhibits `maintain speed` policy till 2.7 s. It then switches to `slow down` policy from 2.7 s to 3.2 s. This corresponds to the moment at which the ego-vehicle initiates its lane-change maneuver. It then switches back to the `maintain speed` policy.

On the other hand, it was observed that the branch-MPC [30] brakes strangely while executing the lane-change maneuver relying on the obstacle-vehicle to yield to the ego-vehicle.[2] However, such a maneuver is risky since it may result in collisions if the obstacle-vehicle does not react on time.

4) Quantitative Results: In each scenario, the initial state of the ego-vehicle remains fixed, while the obstacle vehicle's initial state is systematically altered across 30 distinct positions. These positions are selected from a uniform grid surrounding the nominal starting conditions. As efficiency metrics, average

[2]The reader can refer to the video supplement, at 02:01, to observe such behavior.

TABLE I
STATISTICAL RESULTS OVER 30 EXPERIMENTS

Planning Approach	Task Duration [s]	Velocity [m/s]	Min. Distance [m]	CR [%]	Max. Risk	Jerk [m/s^3]
Our method	3.84 (0.516)	17.14 (0.42)	0.61 (0.14)	0.00 %	0.0491	2.686 (4.28)
FISS+ [45]	-	14.87 (0.51)	1.20 (0.00)	0.00 %	0.0393	2.417 (5.08)
Branch-MPC [30]	4.68 (0.098)	15.25 (0.45)	0.82 (0.04)	0.00 %	0.0475	4.494 (5.58)
Non-cont. MLE [50]	3.71 (0.324)	17.62 (0.51)	0.48 (0.12)	2.74 %	0.1081	5.278 (6.13)
Contingent w/o belief updater [15]	4.46 (0.308)	16.68 (0.78)	0.84 (0.08)	0.00 %	0.0462	2.693 (4.63)

The results are reported as "average (standard deviation)".

speed, and duration to complete the overtaking maneuver are calculated. Moreover, the average minimum distance between the ego-vehicle and the obstacle vehicle is recorded as a measure of conservatism. The quantitative results are summarized in Table I where the *non-contingent robust baseline* [44] refers to the case in which a single plan is optimized along the entire horizon that accounts for all obstacle-vehicle policies. As shown in Table I, our approach can complete the overtake maneuver in less duration and with a higher average speed, while providing the same safety guarantees as the branch-MPC. This could be attributed to the fact that although the branch-MPC plans a distinct trajectory for each branch in the scenario, the optimization problem minimizes the expectation over all branches. This causes the ego-vehicle to overreact to branches with low probabilities resulting in a more conservative plan. The non-contingent robust baseline, on the other hand, fails to complete the maneuver. Since a single trajectory is optimized that avoids all obstacle predictions, it could not find a safe trajectory to execute the lane change. As expected, the MLE baseline is the least-conservative approach, among the ones in comparison, since it only considers the most probable mode. This, however, results in collisions in some of the scenarios due to its *over-confidence* in relying solely on the highest probable mode of the prediction model. Our approach mitigates the limitation of the MLE method by inferring a posterior distribution over the obstacle-vehicle's intent allowing the ego-vehicle to account for uncertainty and generate safer yet efficient plans. Finally, the baseline that uses the same contingency planner as ours but lacks a belief updater, can complete the lane change maneuver safely in all experiments. Nonetheless, the absence of a belief updater prolongs the time it takes for the predictive model to assign a diminished probability to the obstacle vehicle's lane change maneuver. Consequently, this leads to a more conservative planning approach compared to the contingency planner equipped with a belief updater. It can also be seen that all approaches respect the maximum risk threshold, $\delta = 0.05$, except the non-contingent MLE approach.

Remark 5: It is worth mentioning that, for the robust baseline, the belief updater is also utilized to weigh the evaluated collision probability of the planned trajectory with each possible mode, which, in turn, affects the calculated risk. Thus, the baseline re-plans every cycle with the newly observed obstacles' states as well.

C. Scenario 2. Urban T-Junction:

In this scenario, the ego vehicle is approaching a T-junction, *with no traffic rules*, in which its mission is to follow its designated lane while being uncertain about the intentions of

the other vehicles as shown in Fig. 9. Here, we consider the case of a multi-vehicle traffic scenario in which two obstacle vehicles approach the T-junction where $\Lambda_1 = \{\texttt{lanekeep}, \texttt{leftturn}\}$, and $\Lambda_2 = \{\texttt{leftturn}, \texttt{yield}\}$. In this case, it is not sufficient to consider the belief of a single agent's intention as we did in the previous scenario, however, instead, we need to get a belief about how the traffic scene will evolve by considering all permutations the traffic participants can have as stated in (7).

1) Environment Setup: The states of both obstacles are randomly initialized and their corresponding policies are randomly assigned from the set of potential policies. The initial velocities of the obstacles are selected in such a way that they arrive at the intersection before the ego-vehicle, forcing the ego-vehicle to react and avoid collisions actively. In this scenario, a horizon of $N = 25$ steps is defined, with a discretization step of 0.2 s, resulting in a time horizon of 5.0 s, and a branching time $t_b = 2.4$ s is defined. As a benchmark comparison, we evaluated the same task using a robust planner that optimizes a single plan that considers all the modes that the other agents could have, and a greedy baseline that only considers the most probable predicted mode of each obstacle. All optimization parameters for both methods are set to be identical to guarantee a fair comparison. At the start of the simulation, all permutations, $\theta_i \in \Theta$, are initialized with equal likelihoods, $\theta_i = 0.25$. The evolution of the belief over both obstacle modes is illustrated in Fig. 10.

In Fig. 10, we present the dynamic evolution of the ego-vehicle's belief regarding various intentions of surrounding obstacles over time. Specifically, Fig. 10(a) illustrates this evolution for a leftward-bound vehicle encountering a T-junction, where the obstacle vehicle faces the choice between continuing straight or executing a left turn. Similarly, Fig. 10(b) portrays the belief dynamics for an upward-bound vehicle confronted with the options of turning left or yielding to the ego-vehicle. In Fig. 10(a), we observe the ego-vehicle's initial struggle with uncertainty regarding the leftward vehicle's intentions, reflected in an equal belief distribution ($b(\lambda_1) = b(\lambda_2) = 0.5$) as its state aligns with the mean of both distributions. However, as the obstacle vehicle's state gradually deviates from this equilibrium, the belief over the left-turn maneuver diminishes, leading to a corresponding increase in belief regarding the alternative mode ($b(\lambda_1)$). This nuanced adjustment allows the ego-vehicle to attenuate its emphasis on the left-turn possibility, thereby facilitating an accelerated trajectory within the T-junction. Analogous dynamics are observed for the second obstacle vehicle, as depicted in Fig. 10(b). Subsequently, the updated beliefs per mode, $\lambda \in \Lambda$, are utilized to update the probabilities over the different permutations, $\theta \in \Theta$, the traffic scene can evolve to

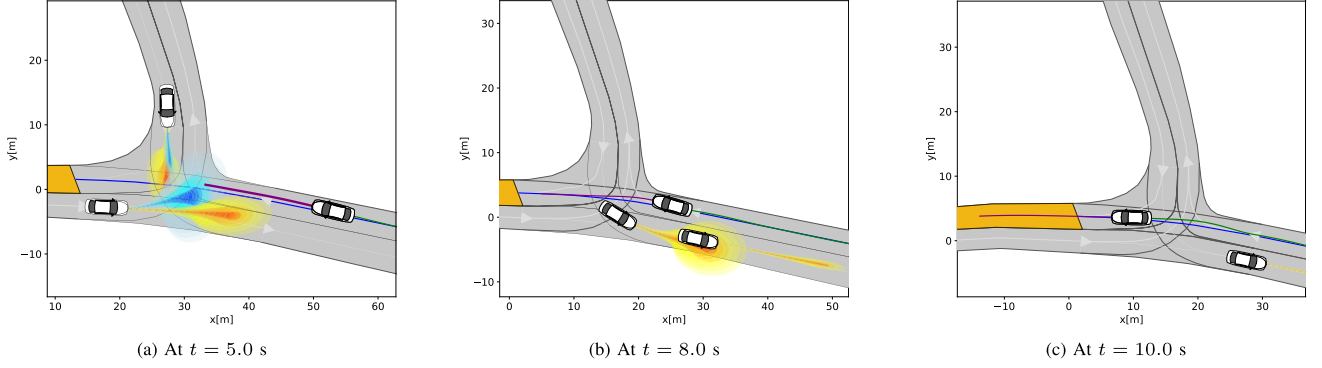(a) At $t = 5.0$ s   (b) At $t = 8.0$ s   (c) At $t = 10.0$ s

Fig. 9. Snapshots from the T-junction simulated environment in CommonRoad at different time instants. The ego-vehicle's plan is to drive from right to left while avoiding collisions with the other two obstacle vehicles. In Fig. 9(a), the prediction of both modes of both obstacles is represented by colored ellipsoids where the blue ellipsoids depict the `left turn` and `yield` policies for the leftward and upward vehicles respectively whereas the yellow ellipsoids illustrate the `lane keep` and `left turn` policies. The driven trajectory by the ego-vehicle is depicted in green. The short-term planned trajectory is illustrated in purple.
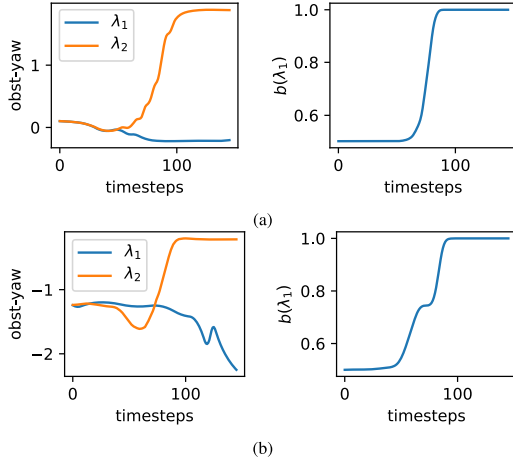

Fig. 10. The evolution of the belief of both obstacles based on their observed orientations according to (6). Top: the belief evolution for obstacle 1 where λ_1 represents the lane keep policy and λ_2 represents the left turn policy. Bottom: the belief evolution for obstacle 2 where λ_1 represents the yield policy whereas λ_2 represents the left turn policy. Note that this represents one of the four permutations that the traffic scene can evolve to.
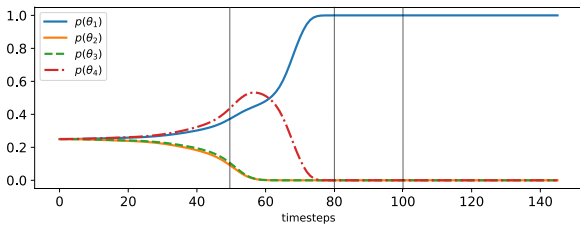

Fig. 11. The evolution of the belief over permutations in the T-junction scenario where θ_1 corresponds to `lane keep` and `left turn` for the leftward and upward vehicles respectively. The gray vertical lines denote the time instants at which the snapshots in Fig. 9 are taken.

as shown in Fig. 11. As illustrated in Fig. 11, the evolution of permutations depicts a gradual decrease in the belief regarding θ_2 and θ_3 over successive iterations, ultimately diminishing after approximately 58 time steps where their influence on the ego-vehicle's plan is disregarded. Furthermore, by the 76th time-step, the ego-vehicle attains a high level of certainty that θ_1

is the accurate hypothesis adopted by the obstacles. As a result, only this prediction mode significantly impacts the planner's decision-making process.

2) Prediction Model: In this scenario, we use a synthesized prediction model that incorporates the multi-modality in the obstacle-vehicle's intentions. Specifically, our multi-modal prediction model works as follows:

i) By identifying target lanes in the T-junction, we extrapolate the intended trajectories of surrounding vehicles.

ii) By employing a motion planner for each vehicle, we generate ground truth trajectories towards these lanes, resulting in multi-modal trajectories per vehicle.

iii) Our multi-modal prediction strategy involves:
  • Utilizing a uni-modal prediction model trained on extensive CommonRoad datasets [67] to generate Gaussian trajectory distributions for each potential mode, corresponding to the trajectories from Step (ii).
  • Amalgamating these distributions into a Gaussian Mixture Model (GMM), weighted by their likelihoods, inspired by prior works such as [18], [19].

iv) Mode weights, determining the likelihood of each trajectory mode, are computed based on collision avoidance metrics. These metrics, quantifying collision probabilities between obstacle vehicle trajectories and the ego-vehicle's planned trajectory, dynamically adjust mode weights using a softmax function inspired by works such as [30].

Remark 6: We emphasize that our approach is agnostic to the prediction model employed. Any prediction model capable of providing Gaussian distributions over the predicted modes can be utilized.

3) Quantitative Results: The quantitative results are reported in Table II. The significant enhancement in the ego-vehicle's performance is attributed to its ability to delay the braking decision, thanks to multiple contingent plans, as long as it is capable of safely braking later when it gets more certainty about other obstacles' intents to react to any possible outcome. This, as expected, comes at the expense of stopping closer to the obstacle, and braking more aggressively in situations in which the ego-vehicle has to yield to the obstacles. Despite this delayed

TABLE II
STATISTICAL RESULTS OVER 100 EXPERIMENTS FOR A T-JUNCTION SCENARIO

Approach	Progress [m]	Velocity [m/s]	Min. Distance. [m]	Max. Acc. [m/s^2]	CR[%]	Jerk [m/s^3]
Our method	91.324 (0.169)	7.017 (0.133)	1.58 (0.06)	2.931 (0.436)	0.00 %	0.841 (0.71)
FISS+ [45]	83.216 (0.413)	6.126 (0.264)	2.08 (0.08)	1.243 (0.617)	0.00 %	0.716 (0.42)
Non-cont. MLE [50]	96.581 (0.648)	7.443 (0.338)	1.06 (0.18)	1.424 (0.556)	3.94 %	0.688 (0.29)
Contingent w/o belief updater [15]	85.649 (0.372)	6.581 (0.162)	1.94 (0.11)	2.983 (0.632)	0.00 %	1.438 (0.96)

The comparison is done with respect to the traveled distance by the ego-vehicle, ego-vehicle velocity, minimum distance to the obstacles, and the maximum deployed accelera-tion. The results are reported as "average (standard deviation)". The branching time is set to t_b = 2.4 s.
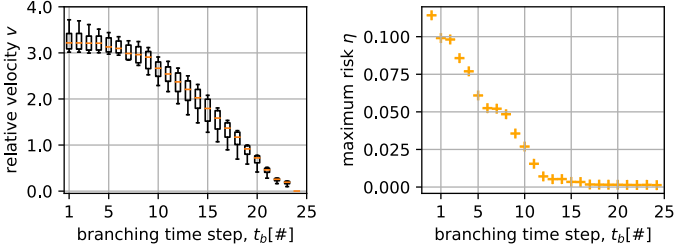

Fig. 12. Left: the performance gap regarding the relative velocity for different branching times, right: the maximum risk, η, recorded in all simulations for different branching times.

TABLE III
AVERAGE COMPUTATIONAL TIME t_c AND STANDARD DEVIATION σ_{t_c} FOR
INCREASING NUMBER OF AGENTS

Number of agents	1	2	3
t_c (ms)	134.53	186.12	204.31
σ_{t_c} (ms)	4.7	6.9	11.8

decision-making, the maximum risk encountered by the ego-vehicle, in the contingency planning case, is still significantly below the defined upper-bound in the chance constraints, as depicted in Fig. 12, showing that performance improvement is attained without compromising safety. As in the previous scenario, the MLE approach has the best performance in terms of average velocity and progress along the driving route. This is, however, achieved at the expense of resulting in collisions making it not safe to be deployed. Similar to the overtaking scenario, the contingency planning without a belief updater has a less efficient performance compared to our proposed approach showing that the belief updater improves the planner's performance without compromising safety.

Here it is worth pointing out that the proposed algorithm is implemented in Python to interface with CommonRoad. The average computational time over the experiments is 174.98 ms. Table III shows how the computational time t_c scales with the number of agents. We emphasize that since the sampling-based approach is parallelizable, the computational time can be further improved by evaluating the constraints of the sampled trajectories through parallelizable computations.

So far, we considered a certain value that we assign to the branching time t_b. In the case of *open-loop* planning, as in [14], [17], the branching time is not an independent design parameter, and it has to be estimated correctly, otherwise, the ego-vehicle will branch to an over-confident contingent plan by t_b which can result in a collision. In our proposed approach, however, thanks to planning in a *closed-loop* with a *belief updater*, the

branching time does not need to be estimated exactly. However, low branching times may lead the ego-vehicle to inevitable states from which it could not recover in case of certain obstacles' permutations, due to the limited dynamics capabilities. Thus, restrictions still apply when it comes to assigning a branching time which we discuss in the following section.

4) Effect of Branching Time on the Plan: In this section, an analysis of how the branching time affects contingency planning is conducted. For this purpose, we run experiments for all values of branching time $t_b \in [\Delta t, T]$, where Δt is the discretization step that we set to 0.2 s. For each branching time, we run 100 simulated experiments in which the obstacles' intents and their initial states are randomly initialized. We analyze the effect of the branching time on the relative average velocity the ego-vehicle exhibits with respect to the baseline, $t_b = T$. Moreover, the maximum risk among all experiments for each branching time is recorded. The results are reported in Fig. 12. As shown, for larger values of the branching time, the performance gap between both methods is small since most of the plan is constituted by the shared plan and thus the effect of the belief updater in the cost function is not pronounced. Indeed, when $t_b = T$, the disparity in performance disappears, as our proposed approach aligns with the baseline method under such conditions. For earlier branching times, however, the performance gap becomes more pronounced since the future information gain beyond the branching time is well exploited in the cost function because of the additional degrees of freedom introduced by the contingent plans. By inspecting the maximum risk plot depicted in Fig. 12, it can be observed that the maximum risk, η, increases as the branching time becomes shorter. This can be attributed to the over-confidence in the planned trajectory after the branching time causing the ego-vehicle to take more risky maneuvers. For sufficiently short branching times, $t_b \leq 2.0$ s in this example, the ego-vehicle could not find a feasible trajectory that does not violate the maximum risk, $\delta = 0.05$, in the chance constraint, and subsequently, we execute the planned with the least risk that is dynamically feasible as we indicated earlier in Remark 3. This concludes that the branching time in contingency planning is related to the maximum risk the ego-vehicle perceives. More analysis regarding the estimation of the branching time t_b based on the ego-vehicle's dynamics capabilities is left for future work.

5) Branching Time Estimation: In this section, we examine the effect of updating the branching time online, based on the updated belief the ego-vehicle maintains over different predic-tion modes, compared to fixing the branching time t_b to a certain value. To do so, three different baselines are considered.

Baseline 5. "Oracle" branching time: This estimator recon-structs the actual branching time by initially simulation the
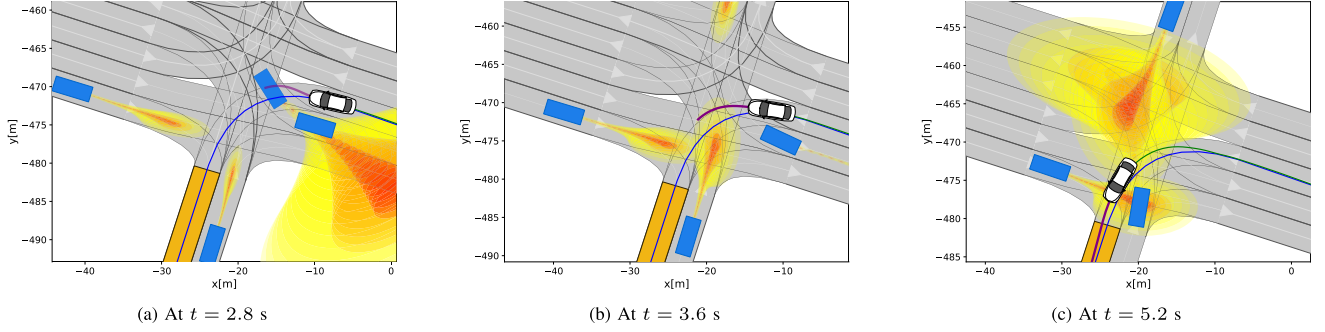
Fig. 13. Snapshots from the intersection simulated environment in CommonRoad at different time instants. The ego-vehicle's plan is to take a left turn while avoiding collisions with the other two obstacle vehicles. The ego-vehicle is depicted as a car icon whereas the obstacle vehicles are represented by blue rectangles. The prediction of the most probable mode of the obstacle vehicles is represented by colored ellipsoids. The driven trajectory by the ego-vehicle is depicted in green. The short-term planned trajectory is illustrated in purple.

planning problem using a nominal branching time. Subsequently, it extracts the moment of certainty from the retrospective evolution of beliefs. It is important to note that the oracle relies on access to the true human intent and, as such, is not implementable in real-world scenarios. Nevertheless, we incorporate this variant to illustrate the potential performance attainable with the true branching time.

Baseline 6. Branching time heuristics adopted from [65]: This heuristic considers the entropy of the belief that the ego-vehicle maintains over each hypothesis, $\theta \in \Theta$, as an indication of how the observed obstacles' states are distinct,

$$\mathcal{H}(p(\theta_i)) = -\sum_{\theta_i \in \Theta} p(\theta_i) \log_{|\Theta|}(p(\theta_i)) \qquad (19)$$

To estimate the branching time, the predicted trajectories of each obstacle along each hypothesis from the previous time-step, $\delta^o_{\theta,t-1}$ are considered as hypothetical observations that can be inferred from their prediction model to estimate the associate belief according to (6). Another operator, $B(\delta^o_{\theta,t-1}, \theta, k)$, is introduced that takes as input the first k steps from the hypothetical observation, $\delta^o_{\theta,t-1}$, and returns the updated belief. Accordingly, the branching time is estimated as

$$t_b = \max_{\theta \in \Theta} \min_{k \in \{2,...,T\}} \omega.k$$

$$\text{s.t.} \qquad \mathcal{H}[B(\delta^o_{\theta,t-1}, \theta, k)] \leq \epsilon \qquad (20)$$

where ω indicates the discretization step. This heuristic estimates the branching time as the first time at which all predicted beliefs reach a certain threshold ϵ, assuming that the obstacles behave rationally with respect to their prediction models.

Baseline 7. Branching time heuristic adopted from [66]: This heuristic chooses the branching time as the maximum time such that any two future scenarios starting at the current time only diverge by a maximum distance,

$$t_b = \max_{k \in 2,...,T} \omega.k$$

$$\text{s.t.} \quad \mathcal{M}(\theta, k) \leq \epsilon, \quad \forall \theta \in \Theta \qquad (21)$$

where $\mathcal{M}$ represents the divergence measure. This heuristic, however, entails at least double the computational time since the divergence measure is invoked on the ego-vehicle trajectories.

TABLE IV
STATISTICAL RESULTS OVER 100 EXPERIMENTS FOR A T-JUNCTION SCENARIO
WITH DIFFERENT BRANCHING TIME ESTIMATES

Approach	Progress [m]	Velocity [m/s]	Max. Risk
Oracle	91.688 (0.178)	7.045 (0.132)	0.0464
$t_b = 2.4$ (Ours)	91.324 (0.169)	7.017 (0.133)	0.0461
Heuristics in [65]	89.948 (0.148)	6.912 (0.192)	0.0426
Heuristics in [66]	88.836 (0.163)	6.825 (0.151)	0.0418

TABLE V
STATISTICAL RESULTS OVER 100 EXPERIMENTS FOR THE INTERSECTION
SCENARIO WHERE CR REFERS TO THE COLLISION RATE

Approach	Progress [m]	Velocity [m/s]	CR
Our method	40.696 (0.582)	8.139 (0.147)	0.00 %
FISS+ [45]	29.225 (0.473)	5.851 (0.168)	0.00 %
Non-cont. MLE [50]	42.261 (0.327)	8.452 (0.263)	3.47 %
Cont. w/o belief updater	37.693 (0.372)	7.531 (0.126)	0.00 %

For all baselines, the branching time is updated at every time step and a Monte Carlo study is conducted to analyze the performance of updating the branching time at every time step compared to fixing the branching time to a certain value, $t_b = 2.4$, that we used in the evaluations in Sections VI-B and VI-C.

As shown in Table IV, the performance gap between fixing the branching time to a certain value, and using an oracle estimate is very small. This can be attributed to the fact that since the short-term plan cost is weighted by the belief the ego vehicle maintains over the long plans, the short-term plan tends to be biased toward the long-term plan with the highest belief.

D. Scenario 3. Intersection

In this scenario, the ego-vehicle is tasked with executing a left turn within an urban intersection, all while navigating interactions with multiple obstacle vehicles simultaneously. Each of these obstacle vehicles within the intersection has the option to either yield to the ego-vehicle, thereby allowing it to complete its left turn unimpeded, or to challenge the ego-vehicle and take priority, thereby compelling the ego-vehicle to yield. Similar to the T-junction scenario, we evaluate the efficacy of our proposed approach against established baselines. Quantitative results are presented in Tab V. For consistency, we utilize a horizon of $N = 25$ steps, with a discretization step of 0.2 s, resulting

in a time horizon of 5.0 s. Additionally, a branching-time of $t_b = 2.4$ s is defined. Here it should be noted that the prediction model in this scenario is similar to the one employed in the T-junction scenario.

The outcomes of this scenario mirror those of previous ones, demonstrating that our proposed approach enables the ego-vehicle to successfully execute the left-turn maneuver, as shown in Fig. 13, with enhanced efficiency compared to the robust baseline, where the ego-vehicle is required to yield while waiting for other vehicles to execute their maneuvers inside the intersection.[3] However, it's noteworthy that the MLE approach outperforms our method in terms of performance, albeit at the cost of a higher collision rate, as it only considers the most probable mode without accounting for potential deviations.

VII. CONCLUSION

This paper introduced a novel contingency planning framework that integrates the ego-agent's beliefs regarding the potential multi-modal behaviors exhibited by surrounding agents. This belief is continuously updated based on inferred states of observed obstacles from a predictive model. The methodology involves decomposing the planning task into short-term and long-term plans, with each long-term plan being tailored to a specific obstacle policy. The resultant contingency plans contribute to the overall plan's cost by factoring in their costs along with the associated belief values derived from the belief updating process. The effectiveness of the proposed approach was evaluated in the context of two safety-critical driving scenarios. Through comprehensive closed-loop simulations, we compared our proposed planner against different baselines. We demonstrated that our approach achieves less conservative driving behavior compared to a state-of-the-art multi-policy algorithm while maintaining equivalent safety assurance. Our approach has also outperformed the traditional planner that optimizes over all possible modes provided by a prediction model. To analyze the effect of the Bayesian belief updater on contingency planning, we showed that the belief updater improves the planner's performance without compromising safety. The influence of branching time on the planner's performance was investigated, and the adaptability of the proposed approach to scenarios involving multiple vehicles was explored.

APPENDIX
BASELINES COMPARISON

To ensure a fair comparison between our proposed contingency planning approach and the branching-MPC approach proposed in [30], the following measures are considered. Except for the branch-MPC [30], all planners employ the ego-motion sampler detailed in Section III-B in the Frenet frame with the same cost function and constraints to rank the generated samples. Nevertheless, although the branch-MPC utilizes a different planner, we modified the cost function such that it is aligned closely to the one used with the Frenet planner. The utilized cost

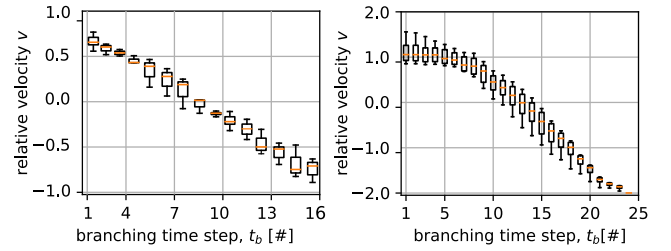| Description	Notation	Value	Unit
Maximum velocity	$v_{\max}$	20	m/s
Minimum velocity	$v_{\min}$	0	m/s
Maximum acceleration	$a_{\max}$	4	m/s^2
Lane width	lw	3	m
Receding horizon	N	16	-
Discretization step	ω	0.2	-
Branching time	t_b	1.2	s
Risk threshold	δ	5	%
Vehicle width	W	2.5	m
Vehicle length	L	4	m
Speed cost weight	w_v	1	-
Lat. deviation cost weight	w_d	1	-
Acceleration cost weight	w_a	2	-
Steering rate cost weight	$w_{\dot{\delta}}$	1	-

Fig. 14. The performance gap regarding the relative velocity for different branching times. Left: overtake scenario, right: T-junction scenario.

function for the Frenet planner is given as,

$$J(\tau) = w_v c_v + w_d c_d + w_a c_a + w_{\dot{\delta}} c_{\dot{\delta}}$$

where c_v, c_d are the costs for velocity and reference tracking, whereas $c_a, c_{\dot{\delta}}$ penalize the acceleration and steering angle rate respectively. On the other hand, for the branch-MPC, similar to the original paper, a unicycle model is adopted where the states are given by $x = [X, Y, v, \psi]^T$, and the inputs $u = [a, \dot{\delta}]$. The cost function for the branch-MPC is, accordingly, defined as

$$J(\tau) = (x - x_{\text{ref}})^T Q (x - x_{\text{ref}}) + u^T R u$$

where $Q = \texttt{diag}(0, w_d, w_v, 0)$, and $R = \texttt{diag}(w_a, w_{\dot{\delta}})$. In this way, we ensure that the cost functions used by the Frenet and branch-MPC planners are similar. Additionally, the same kinematic constraints are applied to all planners including the curvature constraints and the box constraints imposed on the velocity, acceleration, and jerk. Table VI summarizes the parameters utilized by the planners in the evaluations.

BRANCHING TIME SELECTION

As mentioned in Section VI-C4, the branching time is a critical parameter that affects the contingency planning efficiency. To justify our branching time selection for the simulated scenarios in Section VI, we conducted an ablation study that measures the performance gap between different branching times and an oracle that has access to the true human intent. Fig. 14 illustrates the performance gap regarding the relative velocity for different branching times for both the overtake and T-junction scenarios.

As shown, for both scenarios, for the branching times in the middle of the planning horizon, the performance gap compared to the oracle gets smaller. As the branching time gets smaller, a less conservative approach compared to the oracle can be achieved. This comes, however, at the expense of having higher risk as we discussed earlier in Section VI-C4. In contrast, with large branching times, the contingency planning becomes more conservative compared to the oracle. Based on the obtained empirical results, we can conclude that fixing the branching time to a certain value in the middle of the planning horizon can achieve close performance to the updating it based on an oracle in hindsight.

REFERENCES

[1] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone, "Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 683–700.

[2] N. Rhinehart, R. McAllister, K. Kitani, and S. Levine, "PRECOG: Prediction conditioned on goals in visual multi-agent settings," in *IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 2821–2830.

[3] S. Casas, C. Gulino, S. Suo, K. Luo, R. Liao, and R. Urtasun, "Implicit latent variable model for scene-consistent motion forecasting," in *Eur. Conf. Comput. Vis.*, 2020, pp. 624–641.

[4] Y. Chen, B. Ivanovic, and M. Pavone, "ScePT: Scene-consistent, policy-based trajectory predictions for planning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 17103–17112.

[5] A. Ben-Tal and A. Nemirovski, "Robust convex optimisation," *Math. Operations Res.*, vol. 23, no. 4, pp. 769–805, 1998.

[6] P. Trautman and A. Krause, "Unfreezing the robot: Navigation in dense, interacting crowd," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2010, pp. 797–803.

[7] A. Mesbah, "Stochastic model predictive control: An overview and perspectives for future research," *IEEE Control Syst. Mag.*, vol. 36, no. 6, pp. 30–44, Dec. 2016.

[8] H. Zhu and J. Alonso-Mora, "Chance-constrained collision avoidance for MAVs in dynamic environments," *IEEE Robot. Automat. Lett.*, vol. 4, no. 2, pp. 776–783, Apr. 2019.

[9] A. Wange, A. Jasour, and B. Williams, "Non-Gaussian chance- constrained trajectory planning for autonomous vehicles under agent uncertainty," *IEEE Robot. Automat. Lett.*, vol. 5, no. 4, pp. 6041–6048, Oct. 2020.

[10] O. de Groot, B. Brito, L. Ferranti, D. Gavrila, and J. Alonso-Mora, "Scenario-based trajectory optimization in uncertain dynamic environments," *IEEE Robot. Automat. Lett.*, vol. 6, no. 3, pp. 5389–5396, Jul. 2021.

[11] S. Magdici and M. Althoff, "Fail-safe motion planning of autonomous vehicles," in *Proc. IEEE Int. Conf. Intell. Transp. Syst.*, 2016, pp. 452–458.

[12] C. Pek and M. Althoff, "Computationally efficient fail-safe trajectory planning for self-driving vehicles using convex optimization," in *Proc. IEEE Int. Conf. Intell. Transp. Syst.*, 2018, pp. 1447–1454.

[13] C. Pek and M. Althoff, "Fail-safe motion planning for online verification of autonomous vehicles using convex optimization," *IEEE Trans. Robot.*, vol. 37, no. 3, pp. 798–814, Jun. 2021.

[14] J. Hardy and M. Campbell, "Contingency planning over probabilistic obstacle predictions for autonomous road vehicles," *IEEE Trans. Robot.*, vol. 29, no. 4, pp. 913–929, Aug. 2013.

[15] W. Zhan, C. Liu, C. -Y. Chan, and M. Tomizuka, "A non-conservatively defensive strategy for urban autonomous driving," in *Proc. IEEE Int. Conf. Intell. Transp. Syst.*, 2016, pp. 459–464.

[16] A. Cui, S. Casas, A. Sadat, R. Liao, and R. Urtasun, "LookOut: Diverse multi-future prediction and planning for self-driving," in *Proc. IEEE Int. Conf. Computer Vis.*, 2021, pp. 16087–16096.

[17] A. Bajcsy, A. Siththaranjan, C. J. Tomlin, and A. D. Dragan, "Analyzing human models that adapt online," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2021, pp. 2754–2760.

[18] S. Bansal, A. Bajcsy, E. Ratner, A. D. Dragan, and C. J. Tomlin, "A hamilton-Jacobi reachability-based framework for predicting and analyzing human motion for safe planning," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 7149–7155.

[19] A. Bajcsy, S. Bansal, E. Ratner, C. J. Tomlin, and A. D. Dragan, "A robust control framework for human motion prediction," *IEEE Robot. Automat. Lett.*, vol. 6, pp. 24–31, Jan. 2021.

[20] S. Kousik, S. Vaskov, M. Johnson-Roberson, and R. Vasudevan, "Safe trajectory synthesis for autonomous driving in unforeseen environments," in *Proc. ASME Dynamic Syst. Control Conf.*, 2017.

[21] Y. Chen, U. Rosolia, C. Fan, A. D. Ames, and R. Murray, "Reactive motion planning with probabilistic safety guarantees," in *Proc. Conf. robot Learn.*, 2021.

[22] A. D. Dragan, K. C. T. Lee, and S. S. Srinivasa, "Legibility and predictability of robot motion," in *Proc. IEEE/ACM Int. Conf. Hum.-Robot Interact.*, 2013, pp. 301–308.

[23] T. Brudigam, K. Ahmic, M. Leibold, and D. Wollherr, "Legible model predictive control for autonomous driving on highways," *IFAC-PapersOnline.*, vol. 51, no. 20, pp. 215–221, 2018.

[24] N. J. Hetherington, E. A. Croft, and H. F. M. Van der Loos, "Hey robot, which way are you going? Nonverbal motion legibility cues for human-robot spatial interaction," *IEEE Robot. Automat. Lett.*, vol. 6, pp. 5010–5015, Jul. 2021.

[25] A. D. Dragan and S.S. Srinivasa, "Generating legible motion," in *Robotics: Sci. Syst.*, 2013.

[26] J. P. Alsterda, M. Brown, and J. C. Gerdes, "Contingency model predictive control for automated vehicles," in *Proc. Amer. Control Conf.*, 2019, pp. 717–722.

[27] I. Batkovic, U. Rosolia, M. Zanon, and P. Falcone, "A robust scenario MPC approach for uncertain multi-modal obstacles," *IEEE Control Syst. Lett.*, vol. 5, no. 3, pp. 947–952, Jul. 2021.

[28] W. Liu, S. -W. Kim, S. Pendleton, and M. H. Ang, "Situation-aware decision making for autonomous driving on urban road using online POMDP," in *Proc. IEEE Intell. Veh. Symp.*, 2015, pp. 1126–1133.

[29] C. Hubmann, J. Schulz, M. Becker, D. Althoff, and C. Stiller, "Automated driving in uncertain environments: Planning with interaction and uncertain maneuver prediction," *IEEE Trans. Intell. Veh.*, vol. 3, no. 1, pp. 5–17, Mar. 2018.

[30] Y. Chen, U. Rosolia, W. Ubellacker, N. Csomay-Shanklin, and A.D. Ames, "Interactive multi-modal motion planning with branch model predictive control," *IEEE Robot. Automat. Lett.*, vol. 7, no. 2, pp. 5365–5372, Apr. 2022.

[31] R. Oliveira, S. H. Nair, and B. Wahlberg, "Interaction and decision making-aware motion planning using branch model predictive control," in *Proc. IEEE Intell. Veh. Symp.*, 2023, pp. 1–8.

[32] V. Fors, B. Olofsson, and E. Frisk, "Resilient branching MPC for multi-vehicle traffic scenarios using adversarial disturbance sequences," *IEEE Trans. Intell. Veh.*, vol. 7, no. 4, pp. 838–848, Dec. 2022.

[33] A. G. Cunningham, E. Galceran, R. M. Eustice, and E. Olson, "MPDM: Multipolicy decision-making in dynamic, uncertain environments for autonomous driving," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2015, pp. 1670–1677.

[34] E. Galceran, A. G. Cunningham, R. M. Eustice, and E. Olson, "Multipolicy decision-making for autonomous driving via changepoint-based behavior prediction: Theory and experiment," *Auton. Robots.*, vol. 41, no. 6, pp. 1367–1382, 2017.

[35] L. Zhang, W. Ding, J. Chen, and S. Shen, "Efficient uncertainty-aware decision-making for automated driving using guided branching," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 3291–3297.

[36] Y. Chen, P. Karkus, B. Ivanovic, X. Weng, and M. Pavone, "Tree-structured policy planning with learned behavior models," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2023, pp. 7902–7908.

[37] N. Rhinehart et al., "Contingencies from observations: Tractable contingency planning with learned behavior models," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2021, pp. 13663–13669.

[38] C. Packer et al., "Is anyone there? learning a planner contingent on perceptual uncertainty," in *Proc. 6th Conf. Robot Learn.*, 2022, pp. 1607–1617.

[39] M. Werling, S. Kammel, J. Ziegler, and L. Gröll, "Optimal trajectories for time-critical street scenarios using discretized terminal manifolds," *Int. J. Robot. Res.*, vol. 31, no. 3, pp. 346–359, 2012.

[40] I. Mutlu, M. Freese, K. Alaa, and F. Schroedel, "Case study on model free fetermination of optimal trajectories in highly automated driving," in *Proc. 9th Symp. Adv. Automot. Control*, 2019, pp. 205–211.

[41] R. Vosswinkel, I. Mutlu, K. Alaa, and F. Schroedel, "A modular and model-free trajectory planning strategy for automated driving," in *Proc. Eur. Control Conf.*, 2020, pp. 1186–1191.

[42] X. Jin, Z. Yan, G. Yin, S. Li, and C. Wei, "An Adaptive Motion PLanning Technique for on-road autonomous driving," *IEEE Access*, vol. 11, pp. 2655–2664, 2021.

[43] J. Cheng, Y. Chen, Q. Zhange, L. Gan, C. Liu, and M. Liu, "Real-time trajectory planning for autonomous driving with Gaussian process and incremental refinement," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2022, pp. 8999–9005.

[44] S. Sun, Z. Liu, H. Yin, and M. H. Ang, "FISS: A trajectory planning framework using fast iterative search and sampling strategy for autonomous driving," *IEEE Robot. Automat. Lett.*, vol. 7, no. 4, pp. 9985–9992, Oct. 2022.

[45] S. Sun et al., "FISS+: Efficient and focused trajectory generation and refinement using fast iterative search and sampling strategy," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2023, pp. 10527–10534.

[46] Y. Chai, B. Sapp, M. Bansal, and D. Anguelov, "MultiPath: Multiple probabilistic anchor trajectory hypotheses for behavior prediction," in *Proc. Conf. Robot Learn.*, 2019.

[47] J. Hong, B. Sapp, and J. Philbin, "Rules of the road: Predicting driving behavior with a convolutional model of semantic interactions," in *Proc. IEEE/CVF Comput. Vis. Pattern Recognit. Conf.*, 2019, pp. 8454–8462.

[48] K. Nakamura and S. Bansal, "Online update of safety assurances using confidence-based predictions," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2023, pp. 12765–12771.

[49] S. Fruhwirth-Schnatter, *Finite Mixture and Markov Switching Models.* Berlin, Germany: Springer, 2006.

[50] W. Xu, J. Pan, J. Wei, and J. M. Dolan, "Motion planning under uncertainty for on-road autonomous driving," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2014, pp. 2507–2512.

[51] ISO26262, "International organization for standardization: Road vehicle - functional safety," 2018. [Online]. Available: https://www.iso.org/standard/68383.html

[52] ISO21448, "International organization for standardization: Safety of the intended functionality," 2022. [Online]. Available: https://www.iso.org/standard/77490.html

[53] M. Geisslinger, F. Poszler, and M. Lienkamp, "An ethical trajectory planning algorithm for autonomous vehicles," *Nature Mach. Intell.*, vol. 5, pp. 137–144, 2022.

[54] K. Mustafa, O. de Groot, X. Wang, J. Kober, and J. Alonso-Mora, "Probabilistic risk assessment for chance-constrained collision avoidance in uncertain dynamic environments," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2023, pp. 3628–3634.

[55] F. S. Barbosa, B. Lacerda, P. Duckworth, J. Tumova, and N. Hawes, "Risk-aware motion planning in partially known environments," in *Proc. IEEE Conf. Decis. Control*, 2021, pp. 5220–5226.

[56] X. Huang, A. Jasour, M. Deyo, A. Hofmann, and B. C. Williams, "Hybrid risk-aware conditional planning with applications in autonomous vehicles," in *Proc. IEEE Conf. Decis. Control*, 2018, pp. 3608–3614.

[57] F. Damerow and J. Eggert, "Balancing risk against utility: Behavior planning using predictive risk maps," in *Proc. IEEE Intell. Veh. Symp.*, 2015, pp. 857–864.

[58] X. Wang, J. Alonso-Mora, and M. Wang, "Probabilistic risk metric for highway driving leveraging multi-modal trajectory predictions," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 10, pp. 19399–19412, Oct. 2022.

[59] A. Philipp and D. Goehring, "Analytic collision risk calculation for autonomous vehicle navigation," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2019, pp. 1744–1750.

[60] M. Geisslinger, F. Poszler, J. Betz, C. Luetge, and M. Lienkamp, "Autonomous driving ethics: From trolley problem to ethics of risk," *Philosophy Technol.*, vol. 34, pp. 1033–1055, 2021.

[61] S. Yan, P. Goulart, and M. Connan, "Stochastic model predictive control with discounted probabilistic constraints," in *Proc. Eur. Control Conf.*, 2018, pp. 1003–1008.

[62] A. Majumdar and M. Pavone, "How should a robot assess risk? Towards an axiomatic theory of risk in robotics in Robotics Research," in *Proc. Robot. Res.*, 2020, pp. 75–84.

[63] M. Althoff, M. Koschi, and S. Manzinger, "CommonRoad: Composable benchmarks for motion planning on roads," in *Proc. IEEE Int. Veh. Symp.*, 2017, pp. 719–726.

[64] J. Kong, M. Pfeiffer, G. Schildbach, and F. Borrelli, "Kinematic and dynamic vehicle models for autonomous driving control design," in *Proc. IEEE Intell. Veh. Symp.*, 2015, pp. 1094–1099.

[65] L. Peters et al., "Contingency games for multi-agent interaction," *IEEE Robot. Automat. Lett.*, vol. 9, no. 3, pp. 2208–2215, Mar. 2024.

[66] T. Li, L. Zhang, S. Liu, and S. Shen, "MARC: Multipolicy and risk-aware contingency planning for autonomous driving," *IEEE Robot. Automat. Lett.*, vol. 8, no. 10, pp. 6587–6594, Oct. 2023.

[67] M. Geisslinger, and P. Karle, and J. Betz, and M. Lienkamp, "Watch-and-learn-net: Self-supervised online learning for probabilistic vehicle trajectory prediction," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, 2021, pp. 869–875.

Khaled A. Mustafa received the B.Sc. degree (*cum laude*) from the Faculty of Mechatronics Engineering, German University in Cairo, New Cairo, Egypt, in 2017, and the M.Sc. degree (*cum laude*) in systems and control from the University of Twente, Enschede, The Netherlands, in 2019. He has been a Visiting Researcher with RWTH Aachen University, Aachen, Germany, in 2016. From 2019 to 2022, he was a Research Engineer with Volkswagen AG, Germany. In June 2022, he joined the Cognitive Robotics (CoR) Department, Delft University of Technology, Delft, The Netherlands, where he is working toward the Ph.D. degree. His research interests include probabilistic safe motion planning, model predictive control, and autonomous vehicles.

Daniel Jarne Ornia received the Ph.D. degree in control and learning for multi-agent systems from the Delft University of Technology, Delft, The Netherland. He is a Postdoctoral Researcher with the Cognitive Robotics Department, Delft University of Technology. He has been a Visiting Researcher with the University of Oxford, Oxford, U.K. His research interests include formal methods in multi-agent systems, cooperative robotic systems, reinforcement learning, and uncertainty quantification for autonomous agents in human environments.

Jens Kober received the Ph.D. Degree in engineering from Technische Universit Darmstadt, Darmstadt, Germany, in 2012, and the Max-Planck-Institut für Intelligente Systeme, Stuttgart, Germany, in 2012. He is an Associate Professor with the Delft University of Technology, Delft, The Netherlands. He was a Postdoctoral scholar jointly with the CoR-Lab, Bielefeld University, Bielefeld, Germany, and the Honda Research Institute Europe, Offenbach, Germany. His research interests include motor skill learning, (deep) reinforcement learning, imitation learning, interactive learning, and machine learning for control. For his research he was the recipient of the Georges Giralt Ph.D. Award for the best Ph.D. thesis in robotics in Europe, 2018 IEEE RAS Early Academic Career Award, 2022 RSS Early Career Award, and has received an ERC Starting grant.

Javier Alonso-Mora received the Ph.D. degree in robotics from ETH Zürich, Zürich, Switzerland. He is an Associate Professor with the Cognitive Robotics department, Delft University of Technology, Delft, The Netherland, and the Principal Investigator with the Amsterdam Institute for Advanced Metropolitan Solutions, Amsterdam, The Netherlands. Before joining the Delft University of Technology, Delft, The Netherlands, he was a Postdoctoral Associate with the Massachusetts Institute of Technology, Cambridge, MA, USA. His main research interests include navigation, motion planning, and control of autonomous mobile robots, with a special emphasis on multi-robot systems, on-demand transportation, and robots that interact with other robots and humans in dynamic and uncertain environments. He was the recipient of an ERC Starting Grant (2021), the ICRA Best Paper Award on Multi-Robot Systems (2019), an Amazon Research Award (2019), and a talent scheme VENI Grant from the Netherlands Organisation for Scientific Research (2017).