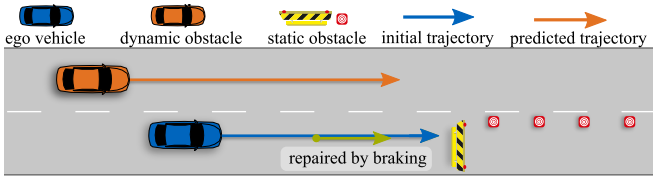
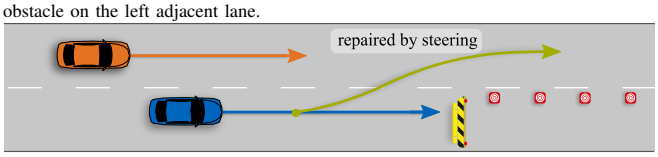
Interaction-Aware Trajectory Repair in Compliance with Formalized Traffic Rules

Youran Wang, Yuanfei Lin, and Matthias Althoff

***Abstract*— Traffic-rule compliance is crucial for motion planning of automated vehicles. If an initially-planned trajectory violates traffic rules, we suggest to repair it instead of completely replanning it to save computational time. However, there exists no trajectory repair framework that considers the interactions among traffic participants, potentially leading to conservative driving behaviors. To address this issue, we propose for the first time an interaction-aware trajectory repair algorithm based on game theory. Our novel algorithm predicts the influence of the repaired trajectory on other traffic participants and then executes the trajectory candidate with the best outcome. To demonstrate our repair mechanism, we integrate it into a receding-horizon motion planning framework. Our approach is evaluated using the CommonRoad benchmark suite, revealing that—compared to the interaction-unaware repair strategy—our approach avoids unnecessarily conservative driving behaviors and achieves a higher repair rate.**



(a) The trajectory is repaired by braking since a lane change is impeded by the dynamic obstacle on the left adjacent lane.



(b) Considering the reaction of the dynamic obstacle, the trajectory can be repaired by a lane change, avoiding the conservative braking maneuver.

Fig. 1: Sketch of trajectory repairing with and without interaction awareness. The initial trajectory of the ego vehicle needs to be repaired as it is about to collide with the static obstacle.

I. Introduction

Motion planning in compliance with traffic rules is crucial for realizing safe automated vehicles. Traffic rules formalized in temporal logic [1] aid in precisely evaluating the rule compliance of the planned trajectories. However, numerous rules need to be considered simultaneously, and there exists no scalable approach to convert arbitrary temporal logic specifications to mathematical constraints [2]. To address this issue, runtime monitors [3] are used to verify the rule compliance of the trajectory after planning. A rule-compliant trajectory repair approach using runtime monitoring is proposed in our previous work [4] to partially rectify rule-violating trajectories. Nonetheless, the existing repair framework in [4] overlooks the interactions among traffic participants. Purposefully interacting with other traffic participants makes it possible to drive less conservatively, as illustrated in Fig. 1.

A. Related Work

Below, we summarize the literature in the fields of game-theoretic and rule-compliant motion planning.

1) Game-Theoretic Motion Planning: Game theory is well-studied for modeling interactions among agents. Competitive driving scenarios such as racing can be modeled by zero-sum games [5], where players aim to maximize their own objectives while minimizing those of others. In contrast, non-competitive driving scenarios can be formulated as general-sum games [6], which can be categorized based on their solution structure: Nash equilibrium or Stackelberg equilibrium. In [7]–[9], the authors aim for Nash equilibria during motion planning, where all agents are assumed to make decisions simultaneously [10]. However, in certain scenarios (e.g., one vehicle merges in front of the other), some traffic participants exhibit their decisions first, and other traffic participants respond to them. This sequential order of decision-making is described by the Stackelberg equilibrium [11], where the leader in the game has the advantage of selecting its trajectory based on the predicted reactions of the followers. The works utilizing the Stackelberg leadership model can be categorized by the prediction models used for the followers. In [12]–[14], the followers are predicted as rational agents that generate the best response. Mathematical models such as the intelligent driver model [15] are employed to simulate the behaviors of the followers in [16], [17], while a neural network is used for this purpose in [18]. However, none of these works verify if the interactive planning outcomes are compliant with traffic rules.

2) Rule-Compliant Motion Planning: To avoid the ambiguities raised from natural language descriptions, traffic rules are formalized in temporal logic, e.g., for interstates [1], intersections [19], and waterways [20]. Using robustness evaluation [21], the rule compliance of motion planning results can be quantitatively evaluated. To be employed as constraints in motion planning problems, formalized traffic rules are integrated into reachability analysis [22], converted into mixed-integer constraints [2], or used as heuristics for sampling-based approaches [23]. Other works verify rule compliance after a trajectory is planned. In [24], rule violations are predicted by reinforcement learning, then a

The authors are with the School of Computation, Information and Technology, Technical University of Munich, 85748 Garching, Germany.
{youran.wang, yuanfei.lin, althoff}@tum.de

trajectory that is anticipated to violate traffic rules is replanned. To gain more computational efficiency, a trajectory repair framework based on satisfiability modulo theories (SMT) [25] that partially modifies the initial trajectory is proposed in [4]. However, no existing work has considered the interactive behaviors of other traffic participants during traffic rule verification.

B. Contributions

We incorporate interaction awareness in the rule-compliant trajectory repair framework proposed in [4] and integrate the repairer in a receding-horizon motion planning framework. Compared to existing works, the novelties of our approach are summarized as follows:

1) incorporating interaction awareness into trajectory repairing, which results in more efficient maneuvers and enhances the comfort of the repaired trajectory;
2) demonstrating the performance of the interaction-aware repairer within a motion planning framework that can incorporate any motion planner; and
3) considering interactive behaviors during predictive rule monitoring using online estimations of the social preferences of other vehicles.

The remainder of this paper is structured as follows: In Sec. II, required notations and definitions are introduced. A social-preference-aware trajectory prediction approach is introduced in Sec. III, followed by the interaction-aware trajectory repair algorithm in Sec. IV. We evaluate our approach in Sec. V.

II. Preliminaries and Problem Statement

A. System Description and Notations

We denote the planning horizon as $t_\mathtt{h}$ and use $t_\mathtt{p}$ to represent the replanning interval. Each time step k corresponds to the continuous time $t_k = k\Delta t$, where Δt is the length of one time step. $\square_k$ denotes the variable $\square$ at time step k. The considered vehicle dynamics are:

$$\boldsymbol{x}_{k+1} = f(\boldsymbol{x}_k, \boldsymbol{u}_k), \tag{1}$$

where $\boldsymbol{x}_k \in \mathbb{R}^n$ is the n-dimensional state vector of the vehicle, and $\boldsymbol{u}_k \in \mathbb{R}^m$ is the m-dimensional input vector to the system; $\boldsymbol{x}_k$ and $\boldsymbol{u}_k$ are subject to the admissable states $\mathcal{X}_k \subset \mathbb{R}^n$ and admissable control inputs $\mathcal{U}_k \subset \mathbb{R}^m$, respectively. By solving (1) with the initial state $\boldsymbol{x}_0$ and the input trajectory $\boldsymbol{u}([0,k])$, we obtain a state trajectory $\chi(\boldsymbol{x}_0, \boldsymbol{u}([0,k]))$. For simplicity, a complete state trajectory spanning the planning horizon $k_\mathtt{h}$ is abbreviated as χ, and $\chi(k)$ returns the k-th state of χ. We denote the i-th vehicle in the driving environment as V_i. A subscript i indicates that the variable $\square_i$ is associated with V_i. In particular, the ego vehicle in a motion planning task is denoted as $\mathrm{V}_\mathtt{ego}$, and its maximum acceleration and velocity are denoted as $a_\mathtt{max}$ and $v_\mathtt{max}$, respectively. In the context of a Stackelberg game, the leader and the follower of $\mathrm{V}_\mathtt{ego}$ are denoted as $\mathrm{V}_\mathtt{l}$ and $\mathrm{V}_\mathtt{f}$, respectively. The cost function of the i-th vehicle is denoted as J_i. We denote the states of all other traffic participants as $\boldsymbol{X}_\mathtt{obs}$ and the road network as $\mathcal{L}$, which collectively define the driving environment $\Omega \coloneqq (\mathcal{L}, \boldsymbol{X}_\mathtt{obs})$ of $\mathrm{V}_\mathtt{ego}$.

A curvilinear coordinate system [26] is used to describe the location of a vehicle along a reference path Γ. A position $(x,y)^T$ is described by a coordinate $(s,d)^T$ in the curvilinear frame, where s is the arc length along Γ and d measures the orthogonal deviation to Γ at s. $\dot{\square}$ and $\ddot{\square}$ ($\square \in \{s,d\}$) denote the velocity and acceleration in the corresponding directions. If Γ is the centerline of a lane l, the functions $\mathtt{adj}_\mathtt{r}(\Gamma)$ and $\mathtt{adj}_\mathtt{l}(\Gamma)$ return the centerlines of the right and left adjacent lanes of l, respectively. In addition, $|\square|$ denotes the cardinality if $\square$ is a set.

B. Signal Temporal Logic

Signal temporal logic (STL) specifies temporal properties of signals in dynamical systems [27]. An STL formula ϕ is defined as [27, Sec. 2.1]:

$$\phi := \sigma \mid \neg\phi \mid \phi_1 \vee \phi_2 \mid \phi_1 \mathbf{S}_I \phi_2 \mid \phi_1 \mathbf{U}_I \phi_2 \tag{2}$$

where σ represents an atomic proposition, ϕ_i represents other STL formulas, and I is a time interval. $\mathbf{S}_I$ and $\mathbf{U}_I$ are the *since* and *until* temporal operators, respectively. The conjunction operator can be constructed from (2) as $\phi_1 \wedge \phi_2 := \neg(\neg\phi_1 \vee \neg\phi_2)$. Additionally, $\mathbf{F}_I$, $\mathbf{G}_I$, $\mathbf{P}$, and $\mathbf{O}_I$ are the *future*, *globally*, *previously*, and *once* operators, which are defined in [27, Sec. 2.1] as $\mathbf{F}_I\phi := \top\mathbf{U}_I\phi$, $\mathbf{G}_I\phi := \neg\mathbf{F}_I\neg\phi$, $\mathbf{P}\phi := \bot\mathbf{S}_{[0,\infty)}\phi$, and $\mathbf{O}_I\phi := \top\mathbf{S}_I\phi$, respectively. We denote all considered STL traffic rules in a planning problem as $\mathcal{R}$.

C. Definitions

As in [4, Sec. II.B], we denote the time remaining until the traffic rule is violated as time-to-violation (TV), and the latest time for which a rule-compliant trajectory exists as time-to-comply (TC). The state and time step from which the repaired trajectory starts are denoted as $\boldsymbol{x}_\mathtt{cut}$ and $t_\mathtt{cut}$, respectively. Additionally, we next define the set of violation-free trajectories.

Definition 1 (Violation-Free Trajectories $\boldsymbol{\chi}_\mathcal{R}^\mathtt{VF}$):
The set $\boldsymbol{\chi}_\mathcal{R}^\mathtt{VF}$ contains all rule-compliant trajectories of the ego vehicle that spans the planning horizon, defined as:

$$\boldsymbol{\chi}_\mathcal{R}^\mathtt{VF} \coloneqq \{\chi \mid \chi \models \mathcal{R}\},$$

where $\chi \models \mathcal{R}$ denotes that χ complies with $\mathcal{R}$.

We finally introduce a measure for the social preferences of the traffic participants.

Definition 2 (Courtesy Factor φ):
The courtesy factor, denoted as $\varphi \in [0,1]$, measures the social preference of an agent. In the objective function of the agent, its own objective is weighted by $1-\varphi$, and that of the others is weighted by φ.

Other quantitative measures of social preferences for traffic participants in, e.g., [28] and [29], can also be adapted in our approach.

D. Problem Statement

For a rule-violating initial trajectory $\chi_{\mathtt{ego}}^{\mathtt{int}}$, we first need to compute TC to let our trajectory repairer intervene at the latest possible time. Afterward, the repairer aims to solve the following problem:

$$\begin{aligned} \chi_{\mathtt{ego}}^{*} &= \arg\min_{\chi_{\mathtt{ego}}} \; J_{\mathtt{ego}}(\cdot) \\ \text{s.t. } \chi_{\mathtt{ego}} &\in \boldsymbol{\chi}_{\mathcal{R}}^{\mathtt{VF}}, \\ \forall k \in [0,\ \mathrm{TC}],\ \chi_{\mathtt{ego}}(k) &= \chi_{\mathtt{ego}}^{\mathtt{int}}(k), \end{aligned} \tag{3}$$

where two constraints are imposed on the repaired trajectory: a) The repaired trajectory belongs to the violation-free set $\boldsymbol{\chi}_{\mathcal{R}}^{\mathtt{VF}}$; b) The deviation from $\chi_{\mathtt{ego}}^{\mathtt{int}}$ starts from the state at TC.

III. Social-Preference-Aware Trajectory Prediction

To anticipate the influence of an intended trajectory of $\mathrm{V}_{\mathtt{ego}}$ on other traffic participants, a social-preference-aware prediction approach is introduced.

A. Leader-Follower Relationship

We predict the behaviors of the dynamic obstacles based on their pairwise leader-follower relationships with $\mathrm{V}_{\mathtt{ego}}$. In this work, for vehicles heading in the same direction of $\mathrm{V}_{\mathtt{ego}}$, we define a vehicle ahead of $\mathrm{V}_{\mathtt{ego}}$ as $\mathrm{V}_{\mathtt{l}}$ and a vehicle with a smaller longitudinal position as $\mathrm{V}_{\mathtt{f}}$. For instance, in Fig. 1, $\mathrm{V}_{\mathtt{ego}}$ recognizes the dynamic obstacle on the adjacent lane as $\mathrm{V}_{\mathtt{f}}$. At intersections, if two vehicles are on different incoming lanes, we define the vehicle with a smaller distance to the intersection as $\mathrm{V}_{\mathtt{l}}$ [30].

B. Trajectory Prediction Based on Social Preferences

In the sequential decision-making process of the Stackelberg game, $\mathrm{V}_{\mathtt{l}}$ makes decisions ahead of $\mathrm{V}_{\mathtt{ego}}$ and is not affected by $\mathrm{V}_{\mathtt{ego}}$. We generate predicted trajectories for $\mathrm{V}_{\mathtt{l}}$ using the intelligent driver model [15]. In contrast, $\mathrm{V}_{\mathtt{f}}$ makes decisions after $\mathrm{V}_{\mathtt{ego}}$ and reacts to its actions. It is noteworthy that drivers in the real world exhibit diverse social preferences, implying that $\mathrm{V}_{\mathtt{f}}$ may not necessarily yield to the behavior of $\mathrm{V}_{\mathtt{ego}}$ (e.g., in Fig. 1, the dynamic obstacle may refuse to brake, and block the lane change behavior of $\mathrm{V}_{\mathtt{ego}}$). Therefore, given the courtesy factor $\varphi_{\mathtt{f}}$, we predict the reaction of $\mathrm{V}_{\mathtt{f}}$ in a social-preference-aware manner by solving the following optimization problem:

$$\begin{aligned} \arg\min_{\boldsymbol{u}_{\mathtt{f}}([0,k_{\mathtt{h}}])} & J_{\mathtt{f}}(\cdot) = (1-\varphi_{\mathtt{f}})\ g_{\mathtt{f}}\left(\chi_{\mathtt{f}}, \boldsymbol{u}_{\mathtt{f}}([0,k_{\mathtt{h}}])\right) + \\ & \varphi_{\mathtt{f}}\ g_{\mathtt{ego}}\left(\chi_{\mathtt{ego}}, \boldsymbol{u}_{\mathtt{ego}}([0,k_{\mathtt{h}}])\right) \\ \text{with } g(\chi, \boldsymbol{u}([0,k_{\mathtt{h}}])) &:= j_{\mathtt{vel}} + j_{\mathtt{acc}} + j_{\mathtt{obs}}^{\mathtt{d}} + j_{\mathtt{dis}}, \\ \text{s.t.} \quad \boldsymbol{x}_{\mathtt{f},k+1} &= f(\boldsymbol{x}_{\mathtt{f},k}, \boldsymbol{u}_{\mathtt{f},k}),\ \boldsymbol{x}_{\mathtt{f},0} = \boldsymbol{x}_{\mathtt{f}}^{\mathtt{int}}, \\ \boldsymbol{u}_{\mathtt{f},k} &\in \mathcal{U}_{\mathtt{f},k},\ \boldsymbol{x}_{\mathtt{f},k} \in \mathcal{X}_{\mathtt{f},k}, \end{aligned} \tag{4}$$

where $\boldsymbol{x}_{\mathtt{f}}^{\mathtt{int}}$ is the initial state of $\mathrm{V}_{\mathtt{f}}$ and $g(\cdot)$ represents the egoistic objective. As introduced in Sec. II-C, we use $\varphi_{\mathtt{f}}$ as a weighting factor in $J_{\mathtt{f}}(\cdot)$. Within $g(\cdot)$, $j_{\mathtt{vel}}$ penalizes the deviation from the desired velocity [31, Sec. III.B], $j_{\mathtt{acc}}$ measures comfort by penalizing the absolute acceleration as in [31, Sec. III.B], and $j_{\mathtt{obs}}^{\mathtt{d}}$ rewards the distance kept to dynamic obstacles as in [32, Tab. I]. Additionally, we define $j_{\mathtt{dis}} = w_{\mathtt{dis}}\Delta s$ to account for driving efficiency, where $w_{\mathtt{dis}}$ is a weighting factor and Δs is the distance from the terminal state of χ to the nearest point in the goal region of the vehicle.

IV. Interaction-Aware Trajectory Repair

In this section, we introduce the interaction-aware trajectory repair framework, which is based on the rule-compliant repair algorithm in [4] and generalized to consider interactions. When rule violations are detected in the current intended trajectory, we first generate high-level repair strategies using the SAT-solving algorithm from [4, Sec. IV.B]. The algorithm abstracts the violated traffic rules into propositional formulas (denoted as $\mathcal{R}^{\mathtt{p}}$) and uses the Davis-Putnam-Logemann-Loveland (DPLL) algorithm [33] to check its Boolean satisfiability. If $\mathcal{R}^{\mathtt{p}}$ is satisfiable, a rule-compliant assignment of the atomic propositions in $\mathcal{R}^{\mathtt{p}}$ is generated, denoted as Φ. Otherwise, the trajectory is considered unrepairable.

The innovation of this work lies in an interaction-aware $\mathcal{T}$-solver, which checks the satisfiability of the repair strategies from the SAT solver and generates repaired trajectories considering interactions among the traffic participants.

A. Overall Algorithm of the Interaction-Aware $\mathcal{T}$-Solver

The workflow of the novel $\mathcal{T}$-solver is presented in Alg. 1. The major enhancement compared to the $\mathcal{T}$-solver from [4] lies in line 4 and line 9, where we incorporate interaction awareness. The assignment Φ is first compared with its Boolean value at TV of the initial trajectory to determine the atomic proposition $\phi_r \in \Phi$ to be repaired (see line 1). In line 2, we acquire a set of repair maneuvers based on ϕ_r according to Tab. I, where the predicates in STL traffic rules are associated with repair maneuvers according to their categories [22]. For example, the predicate

Algorithm 1 $\mathcal{T}$-SOLVER

Input: environment model Ω, solution from the SAT solver Φ, initial assignment at TV $\Phi^{\mathtt{int}}$, TV, rule monitor $\mathfrak{M}$, estimated courtesy factors of other participants $\hat{\boldsymbol{\varphi}}_{\mathtt{obs}}$
Output: repairability $\mathfrak{r}$, repaired trajectory $\chi^{\mathtt{rep}}$

1: $\phi_r \leftarrow$ OBTAINREPAIREDPROPOSITIONS(Φ, $\Phi^{\mathtt{int}}$)
2: $\mathcal{M} \leftarrow$ SETCOMPLIANTMANEUVERS(ϕ_r)
3: **for all** m in $\mathcal{M}$ **do**
4: $\quad \mathrm{TC}_m \leftarrow$ SEARCHTC(Ω, TV, m, $\mathfrak{M}$, $\hat{\boldsymbol{\varphi}}_{\mathtt{obs}}$) ▷ Sec. IV-B
5: **end for**
6: SORT($\mathcal{M}$) ▷ Based on the descending order of TC_m
7: **for all** m in $\mathcal{M}$ **do**
8: $\quad$ **if** $\mathrm{TC}_m >= 0$ **then**
9: $\quad\quad \chi^{\mathtt{rep}} \leftarrow$ GAMETHEORETICREPAIR(Ω, m, $\mathfrak{M}$, $\hat{\boldsymbol{\varphi}}_{\mathtt{obs}}$, TC_m) ▷ Alg. 2
10: $\quad\quad$ **if** $\chi^{\mathtt{rep}}$ **is not** None **then**
11: $\quad\quad\quad$ **return** $\top$, $\chi^{\mathtt{rep}}$
12: $\quad\quad$ **end if**
13: $\quad$ **end if**
14: **end for**
15: **return** $\bot$, None

TABLE I: Association of predicates, maneuvers, reference paths, and desired velocities.

Predicate [1], [19]	**Category** [22]	**Maneuver**	$\Gamma_{\mathtt{ref}}$	$\dot{s}_{\mathtt{desired}}$
in_front_of, keeps_safe_distance_prec, not_endanger_intersection, ...	**Longitudinal position**, **Velocity**	*brake*	Γ'^{*}	$\max(\dot{s}_{\mathrm{TC}} - a_{\mathtt{max}} T_{\mathtt{rep}}{}^{**}, 0)$
		kick-down	Γ'	$\min(\dot{s}_{\mathrm{TC}} + a_{\mathtt{max}} T_{\mathtt{rep}}, v_{\mathtt{max}})$
in_same_lane, cut_in, in_lanelet_with_type, ...	**Lateral position**	*lane-change-to-the-left*	$\mathtt{adj}_{\mathtt{l}}(\Gamma')$	$\dot{s}_{\mathrm{TC}}$
		lane-change-to-the-right	$\mathtt{adj}_{\mathtt{r}}(\Gamma')$	$\dot{s}_{\mathrm{TC}}$
		steer-left-and-brake	$\mathtt{adj}_{\mathtt{l}}(\Gamma')$	$\max(\dot{s}_{\mathrm{TC}} - \lambda^{\dagger} a_{\mathtt{max}} T_{\mathtt{rep}}, 0)$
		steer-right-and-brake	$\mathtt{adj}_{\mathtt{r}}(\Gamma')$	$\max(\dot{s}_{\mathrm{TC}} - \lambda a_{\mathtt{max}} T_{\mathtt{rep}}, 0)$

[*] Γ' denotes the reference path of $\mathrm{V}_{\mathtt{ego}}$ before m is employed. [**] $T_{\mathtt{rep}} = t_{\mathtt{h}} - \mathrm{TC}$.
[†] Considering the constraint of the fraction circle, the longitudinal acceleration is scaled by $\lambda \in [0, 1]$ when there are lateral movements.

keeps_safe_distance_prec belongs to the category **Longitudinal position**. According to Tab. I, the set of maneuvers $\mathcal{M} = \{$*brake, kick-down*$\}$ can be employed to repair this predicate. Compared to [4, Tab. I], Tab. I additionally contains the repair maneuvers that involve both lateral and longitudinal operations, e.g., *steer-left-and-brake* and *steer-right-and-brake*. These additional maneuvers are associated with the predicates in the category of lateral position. Then in lines 3-5, for each maneuver $m \in \mathcal{M}$, we compute the latest time to employ m to realize rule compliance with interaction awareness, which is defined as:

$$\mathrm{TC}_m \coloneqq \max\{k \in [0, \mathrm{TV}] \mid \chi_{m,k} \in \boldsymbol{\chi}_{\mathcal{R}}^{\mathtt{VF}}(\hat{\boldsymbol{\chi}}_{\mathtt{obs}}(\chi_{m,k}))\},$$

where $\chi_{m,k} \coloneqq \chi(x_0, [\boldsymbol{u}^{\mathtt{int}}([0,k)), \boldsymbol{u}_m([k, k_{\mathtt{h}}])])$ is the trajectory that follows the initial input $\boldsymbol{u}^{\mathtt{int}}$ until k and then takes $\boldsymbol{u}_m$ associated with the maneuver m as input. The algorithm to compute TC_m is explained in Sec. IV-B. To let the repair mechanism intervene at the latest possible time, we sort the maneuvers in $\mathfrak{M}$ based on the descending order of TC_m (see line 6). Then in lines 7-14, we iterate through all maneuvers sequentially, attempting to repair the trajectory using the game-theoretic approach introduced in Sec. IV-C. If no repaired trajectory is generated by any maneuvers in $\mathcal{M}$, Φ is evaluated to be unsatisfiable, and the SAT solver generates the next repair strategy with $\mathcal{R}^{\mathrm{p}}$ updated to $\mathcal{R}^{\mathrm{p}} \wedge \neg\Phi$, as in [4, Alg. 1].

B. Interaction-Aware TC Search

We compute TC_m using a binary search algorithm, which incorporates interaction awareness into [34, Alg. 2]. Initially, the search interval is set to $[0, \mathrm{TV}]$. Starting from each visited time step, a trajectory of $\mathrm{V}_{\mathtt{ego}}$ following the maneuver m is simulated by a point mass vehicle model [35, Sec. III.A]. We then predict the corresponding reactions of other traffic participants with the approach outlined in Sec. III. Subsequently, rule compliance (employed as the search condition) of the simulated trajectory is assessed by the rule monitor against the predicted trajectories of other vehicles, and the boundaries of the binary search are adjusted accordingly. If the initial time step is visited and fails to meet the search condition, the algorithm returns $\mathrm{TC}_m = -1$, which indicates that the initial trajectory is unrepairable using the maneuver m.

C. Game-Theoretic Repair

Alg. 2 summarizes the interaction-aware trajectory repair algorithm that solves the problem in (3). The leader-follower relationships in the Stackelberg game are established in line 1 as introduced in Sec. III-A. In the remaining part of Alg. 2, we first generate a set of repaired trajectories $\boldsymbol{\mathcal{S}}_{\mathtt{ego}}$ for $\mathrm{V}_{\mathtt{ego}}$ and evaluate each trajectory sample individually.

1) Trajectory Set Generation: We make adaptations to the trajectory sampling approach proposed in [36] to align $\boldsymbol{\mathcal{S}}_{\mathtt{ego}}$ with the selected repair maneuver m. To achieve this, we determine the reference path $\Gamma_{\mathtt{ref}}$ and the desired longitudinal velocity $\dot{s}_{\mathtt{desired}}$ according to m as summarized in Tab. I. For example, if $m =$ *lane-change-to-the-left*, we choose $\Gamma_{\mathtt{ref}} = \mathtt{adj}_{\mathtt{l}}(\Gamma')$ and $\dot{s}_{\mathtt{desired}} = \dot{s}_{\mathrm{TC}}$. Then we sample the positions and velocities of the terminal states of the trajectory samples in the neighborhood of $\Gamma_{\mathtt{ref}}$ and $\dot{s}_{\mathtt{desired}}$. Subsequently, the approach in [36] is employed to connect the sampled terminal states and $\boldsymbol{x}_{\mathtt{cut}}$ by trajectory samples that adhere to the kinematic single-track model [35, Sec. III.B], which leads to a discretized trajectory set $\boldsymbol{\mathcal{S}}_{\mathtt{ego}}$ branching from TC (see Alg. 2 line 3). Fig. 2 showcases a trajectory set $\boldsymbol{\mathcal{S}}_{\mathtt{ego}}$ following a lane change maneuver.

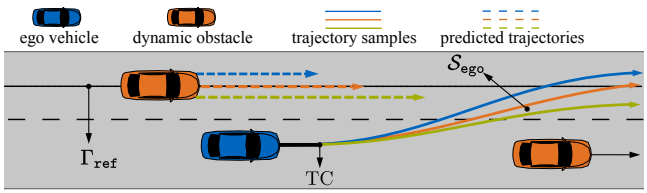


Fig. 2: Trajectory set generation and reaction prediction. $\Gamma_{\mathtt{ref}}$ and $\boldsymbol{\mathcal{S}}_{\mathtt{ego}}$ are associated with the repair maneuver *lane-change-to-the-left*. A dashed line represents the predicted reaction of the dynamic obstacle when the trajectory sample of the same color is executed by $\mathrm{V}_{\mathtt{ego}}$.

2) Trajectory Sample Selection: In Alg. 2 line 6, for each sample $\chi'_{\mathtt{ego}}$ in $\boldsymbol{\mathcal{S}}_{\mathtt{ego}}$, the trajectory predictions of other vehicles are updated by the approach introduced in Sec. III, as illustrated in Fig. 2. Rule compliance of $\chi'_{\mathtt{ego}}$ is evaluated in Alg. 2 line 7 against the updated predictions by the rule monitor. If no violations are detected, $J_{\mathtt{ego}}$ is computed in Alg. 2 line 8 and compared with the current optimal objective $J^{*}_{\mathtt{ego}}$. After the enumeration, the algorithm terminates by returning the rule-compliant trajectory sample with the lowest cost. If no trajectory sample is evaluated as rule-compliant, then no trajectory is returned. The complexity of Alg. 2 is $\mathcal{O}(|\boldsymbol{\mathcal{S}}_{\mathtt{ego}}|)$, with respect to the size of the trajectory set.

Algorithm 2 GAMETHEORETICREPAIR

Input: Ω, m, $\mathfrak{M}$, $\hat{\boldsymbol{\varphi}}_{\mathtt{obs}}$, TC
Output: repaired trajectory χ^*

1: INITIALIZEGAME(Ω)
2: $\Gamma_{\mathtt{ref}}$, $\dot{s}_{\mathtt{desired}} \leftarrow$ GENERATEREFERENCE(m) ▷ Tab. I
3: $\boldsymbol{\mathcal{S}}_{\mathtt{ego}} \leftarrow$ GENERATETRAJECTORYSET($\Gamma_{\mathtt{ref}}$, $\dot{s}_{\mathtt{desired}}$, TC)
4: $J^*_{\mathtt{ego}} \leftarrow \infty$, $\chi^* \leftarrow$ None
5: **for all** $\chi'_{\mathtt{ego}}$ in $\boldsymbol{\mathcal{S}}_{\mathtt{ego}}$ **do**
6: $\quad\hat{\boldsymbol{\chi}}_{\mathtt{obs}} \leftarrow$ PREDICT($\chi'_{\mathtt{ego}}$, $\Omega.\boldsymbol{X}_{\mathtt{obs}}$, $\hat{\boldsymbol{\varphi}}_{\mathtt{obs}}$) ▷ Sec. III
7: $\quad$**if** $\mathfrak{M}$.EVALUATE($\chi'_{\mathtt{ego}}$, $\hat{\boldsymbol{\chi}}_{\mathtt{obs}}$, Ω) $== \top$ **then**
8: $\quad\quad J_{\mathtt{ego}} \leftarrow$ COMPUTECOST($\chi'_{\mathtt{ego}}$, $\hat{\boldsymbol{\chi}}_{\mathtt{obs}}$)
9: $\quad\quad$**if** $J_{\mathtt{ego}} < J^*_{\mathtt{ego}}$ **then**
10: $\quad\quad\quad\chi^* \leftarrow \chi'_{\mathtt{ego}}$, $J^*_{\mathtt{ego}} \leftarrow J_{\mathtt{ego}}$
11: $\quad\quad$**end if**
12: $\quad$**end if**
13: **end for**
14: **return** χ^*

Algorithm 3 INTERACTIONAWAREMOTIONPLANNING

Input: Ω, $\mathfrak{M}$, trajectory planner $\mathcal{P}$

1: $\chi_{\mathtt{ego}} \leftarrow \mathcal{P}$.PLAN($\cdot$), $\chi_{\mathtt{safe}} \leftarrow$ FAILSAFEPLAN($\cdot$), $k \leftarrow 0$
2: **while** $k < k_{\mathrm{p}}$ **do**
3: $\quad\Omega \leftarrow$ UPDATEENVIRONMENT(Ω)
4: $\quad\hat{\boldsymbol{\varphi}}_{\mathtt{obs}} \leftarrow$ UPDATECOURTESYFACTOR($\Omega.\boldsymbol{X}_{\mathtt{obs},k}$) ▷ [28, Alg. 2 in the Appendix]
5: $\quad\hat{\boldsymbol{\chi}}_{\mathtt{obs}} \leftarrow$ PREDICT($\chi_{\mathtt{ego}}$, $\Omega.\boldsymbol{X}_{\mathtt{obs},k}$, $\hat{\boldsymbol{\varphi}}_{\mathtt{obs}}$) ▷ Sec. III
6: $\quad$**if** $\mathfrak{M}$.EVALUATE($\chi_{\mathtt{ego}}$, $\hat{\boldsymbol{\chi}}_{\mathtt{obs}}$, Ω) $== \bot$ **then**
7: $\quad\quad\mathfrak{r}$, $\chi^*_{\mathtt{ego}} \leftarrow$ INTERACTIONAWAREREPAIR(Ω, $\chi_{\mathtt{ego}}$, $\hat{\boldsymbol{\varphi}}_{\mathtt{obs}}$, $\mathfrak{M}$) ▷ Sec. IV
8: $\quad\quad$**if** $\mathfrak{r} == \top$ **then**
9: $\quad\quad\quad\chi_{\mathtt{ego}} \leftarrow \chi^*_{\mathtt{ego}}$
10: $\quad\quad$**else**
11: $\quad\quad\quad\chi_{\mathtt{ego}} \leftarrow \chi_{\mathtt{safe}}$
12: $\quad\quad$**end if**
13: $\quad\quad$PUBLISH($\chi_{\mathtt{ego}}$) ▷ Publish $\chi_{\mathtt{ego}}$ to the controller.
14: $\quad$**end if**
15: $\quad k \leftarrow k + 1$
16: **end while**

D. Trajectory Repair in a Receding-Horizon Motion Planning Framework

Our interaction-aware trajectory repairer is demonstrated by integrating it in a motion planning framework. Alg. 3 shows the procedure of the overall algorithm in one planning cycle. The input includes an arbitrary motion planner $\mathcal{P}$. Without knowing the courtesy factors of other traffic participants, their behaviors are initially predicted based on guessed values. Afterward, during the online planning process, the particle filter adapted from [28, Alg. 2 in the Appendix] is used to estimate their courtesy factors based on observations of their trajectories.

We initially use the trajectory planner $\mathcal{P}$ to generate an intended trajectory $\chi_{\mathtt{ego}}$ for $\mathrm{V}_{\mathtt{ego}}$ (see Alg. 3 line 1). Meanwhile, to guarantee safety in an infinite time horizon, we generate a fail-safe trajectory using the approach proposed in [37]. Afterward, at every consecutive time step, the estimation of courtesy factors of other vehicles is refined based on the latest observations of their trajectories (see line 4). The updated courtesy factor $\hat{\boldsymbol{\varphi}}_{\mathtt{obs}}$ is fed into the prediction algorithm in Sec. III to update trajectory predictions $\hat{\boldsymbol{\chi}}_{\mathtt{obs}}$ (see line 5). The rule monitor is subsequently utilized to verify the rule compliance of $\chi_{\mathtt{ego}}$ against $\hat{\boldsymbol{\chi}}_{\mathtt{obs}}$ (see line 6). When rule violations are detected, we execute the interaction-aware repair algorithm proposed in Sec. IV (see line 7). If the repairability $\mathfrak{r} \in \{\bot, \top\}$ evaluates to true, $\mathrm{V}_{\mathtt{ego}}$ regards $\chi^*_{\mathtt{ego}}$ as its new intended trajectory, which will be verified again for rule compliance in the subsequent time step. If $\chi_{\mathtt{ego}}$ is not repairable by this algorithm, the fail-safe trajectory is executed (see line 11).

V. CASE STUDIES

The interaction-aware trajectory repair framework is evaluated in two scenarios from the CommonRoad benchmark suite [35]: an obstacle avoidance scenario[1] (scenario I) and an intersection scenario[2] (scenario II). The initial configurations of the scenarios are shown in Fig. 3a and Fig. 4a. The time step size of both scenarios is $\Delta t = 0.1\ s$. The social preferences of other vehicles are unknown to $\mathrm{V}_{\mathtt{ego}}$. When calculating $j_{\mathtt{vel}}$, we define the initial velocity of a vehicle as its desired velocity. When generating trajectory samples, we use $\lambda = \frac{\sqrt{2}}{2}$ (cf. footnote of Tab. I).

The related STL traffic rules are listed in Tab. II. To verify the rule compliance of lane change maneuvers according to [38], we additionally formalize the rule to keep a safe distance to the following vehicle on the target lane during the lane change as R_LC in Tab. II.

TABLE II: Traffic rules considered in our experiments.

Rule[1]	STL formula
R_G1 [1]	$\mathbf{G}(\text{in\_same\_lane}(\boldsymbol{x}_{\mathtt{ego}}, \boldsymbol{x}_i) \wedge \text{in\_front\_of}(\boldsymbol{x}_{\mathtt{ego}}, \boldsymbol{x}_i)$ $\wedge\neg\mathbf{O}_{[0,t_c]}(\text{cut\_in}(\boldsymbol{x}_i, \boldsymbol{x}_{\mathtt{ego}}) \wedge \mathbf{P}(\neg\text{cut\_in}(\boldsymbol{x}_i, \boldsymbol{x}_{\mathtt{ego}})))$ $\Longrightarrow \text{keeps\_safe\_distance\_prec}(\boldsymbol{x}_{\mathtt{ego}}, \boldsymbol{x}_i))$
R_IN3 [19]	$\mathbf{G}(\text{on\_incoming\_left\_of}\,(\boldsymbol{x}_{\mathtt{ego}}, \boldsymbol{x}_i)$ $\wedge\neg\ \text{relevant\_traffic\_light}\ \ (\boldsymbol{x}_{\mathtt{ego}}) \wedge \text{same\_priority}\,(\boldsymbol{x}_{\mathtt{ego}}, \boldsymbol{x}_i)$ $\Longrightarrow (\mathbf{G}(\text{not\_endanger\_intersection}\,(\boldsymbol{x}_{\mathtt{ego}}, \boldsymbol{x}_i)^2)$ $\vee\neg\text{on\_lanelet\_with\_type}\,(\boldsymbol{x}_{\mathtt{ego}}, \textit{intersection})))$
R_LC	$\mathbf{G}(\text{in\_front\_of}(\boldsymbol{x}_{\mathtt{ego}}, \boldsymbol{x}_i) \wedge \mathbf{O}_{[0,t_c]}(\text{cut\_in}(\boldsymbol{x}_{\mathtt{ego}}, \boldsymbol{x}_i))$ $\Longrightarrow \text{keeps\_safe\_distance\_rear}(\boldsymbol{x}_{\mathtt{ego}}, \boldsymbol{x}_i))$

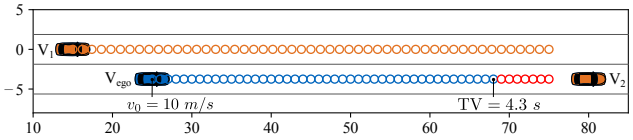
[1] In scenario I, $\mathcal{R}$:= R_G1 $\wedge$ R_LC. In scenario II, $\mathcal{R}$:= R_IN3.
[2] See META-1 in [19, Tab. V] for the definition of this proposition.

A. Interaction-Aware Trajectory Repair

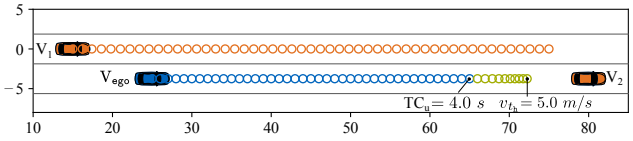
We first demonstrate the repair results at the initial time step. Without knowing the courtesy factors of other vehicles, our interaction-aware repairer (referred to as the IA repairer from now on) initially assumes their courtesy factors to be 0.5, which means they are operating socially. The outcomes are compared with that of the interaction-unaware repairer (referred to as the IU repairer from now on) from [4] to showcase the benefits of incorporating interaction awareness.

1) Scenario I: In scenario I, $\mathrm{V}_{\mathtt{ego}}$ violates the rule R_G1 because it does not keep a safe distance to the

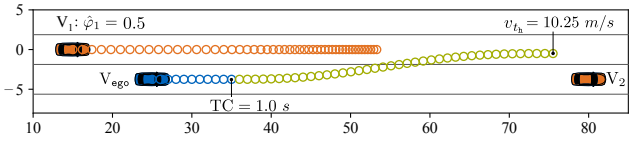
[1]CommonRoad ID: C-ZAM_Highway-2_1_T-1
[2]CommonRoad ID: C-ZAM_Intersection-2_1_T-1



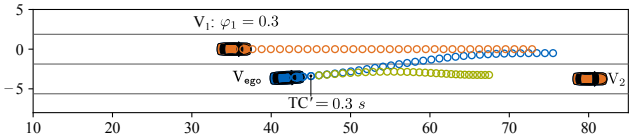
(a) Initial scenario.



(b) Repaired without interaction awareness.



(c) Repaired with interaction awareness.



(d) Repaired with interaction awareness based on φ estimation at $k = 17$.

Fig. 3: Trajectory repair results of scenario I.

slowly proceeding vehicle V_2. The first repair strategy generated from the SAT solver is to negate the proposition in_same_lane($\boldsymbol{x}_{ego}, \boldsymbol{x}_2$), which leads to a repair maneuver that requires a lane change to the left. Without interaction awareness, the $\mathcal{T}$-solver rejects this maneuver since it is impossible to adhere to the rule R_LC while conducting the lane change. Therefore, the IU repairer employs the next satisfying solution: keeps_safe_distance_prec($\boldsymbol{x}_{ego}, \boldsymbol{x}_2$), which requires V_{ego} to brake, as shown in Fig. 3b. However, when considering the reactions of V_1, the lane change is feasible, as shown in Fig. 3c. The IA repairer takes V_2 as the leader of V_{ego} and recognizes V_1 as the follower of V_{ego}. Based on the prediction that V_1 brakes in reaction to the cut-in maneuver of V_{ego}, a rule-compliant lane change trajectory for V_{ego} can be generated. The cost comparisons shown in Tab. III indicate that when using the IA repairer, we achieve less deviation to the desired velocity (lower $j_{\mathtt{vel}}$), shorter distance to the goal region (lower $j_{\mathtt{dis}}$), and more comfortable driving experience (lower $j_{\mathtt{acc}}$). J_1 is increased by 45.5% when V_1 yields to V_{ego}.

2) Scenario II: As shown in Fig. 4a, the initial trajectory of V_{ego} violates the rule R_IN3, because it endangers V_1 which is heading to the intersection on a right incoming lane. The SAT solver generates the repair strategy of satisfying not_endanger_intersection($\boldsymbol{x}_{ego}, \boldsymbol{x}_1$), which corresponds to the maneuvers *kick-down* and *brake*. The IA repairer recognizes V_1 as a follower of V_{ego}. Therefore, it predicts that if V_{ego} employs the maneuver *kick-down*, V_1 brakes to let V_{ego} pass first. As shown in Fig. 4c, a repaired trajectory where V_{ego} rushes into the intersection is evaluated as rule-compliant. On the contrary, without the capability to anticipate the possible deceleration behavior of V_1, the IU repairer results in a conservative braking maneuver (see Fig. 4b). The cost values in Tab. III relating to scenario II demonstrate that the IA repairer improves the comfort and speed maintenance of the repaired trajectory and enables V_{ego} to proceed closer to the goal region. J_1 is increased by 17.7% when V_1 yields to V_{ego}.

TABLE III: Comparisons of the cost of the repaired trajectories from the IA and IU repairers. Superior values are indicated in bold.

Scenario	Repairer	$j_{\mathtt{vel}}$	$j_{\mathtt{dis}}$	$j_{\mathtt{acc}}$	Cost Improvement [1]
Scenario I	IA	**0.4**	**25.2**	**6.2**	54.6%
	IU	11.2	27.6	30.7	
Scenario II	IA	**7.7**	**7.0**	**7.6**	93.6%
	IU	223.4	31.1	92.0	

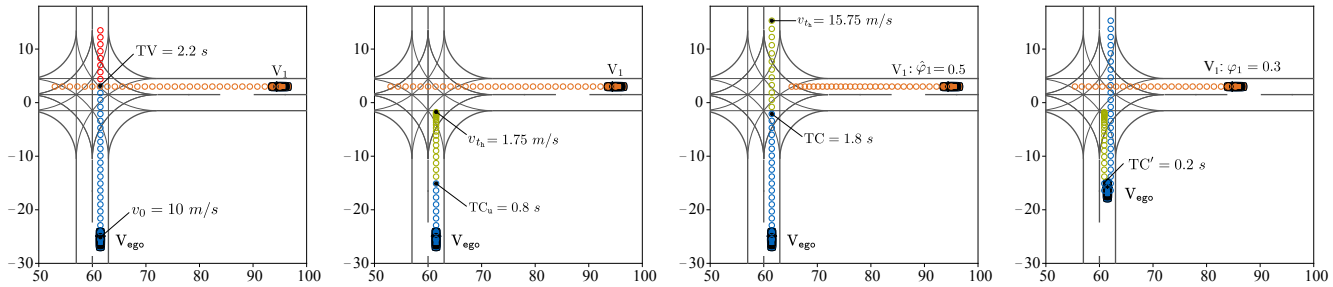
[1] Improvement of the overall cost J_{ego}.

B. Trajectory Repair with Courtesy Factor Estimation

To demonstrate the performance of the IA repairer in a receding-horizon motion planning framework, we run Alg. 3 online in the two scenarios. The interactive behaviors of the dynamic obstacles are generated by manually defining their courtesy factors and solving the problem in (4). To show the ability of our repairer to rectify aggressive maneuvers of V_{ego} stemming from an incorrect initialization of $\hat{\varphi}_1$, we configure $\varphi_1 = 0.3$ in the subsequent simulations, which means V_1 is more selfish than our initial assumption $\hat{\varphi}_1 = 0.5$. The sampling-based planner proposed in [36] is utilized as $\mathcal{P}$ in Alg. 3, and we configure $t_p = 3.0\ s$.

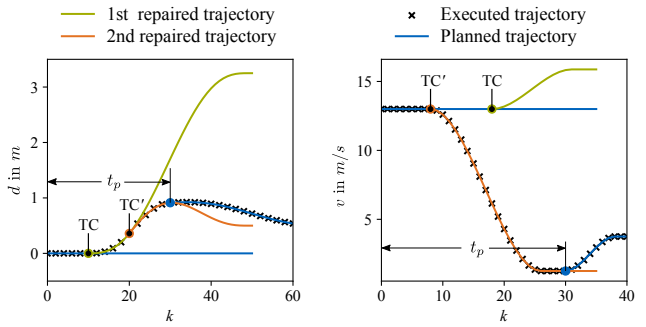
In both scenarios, V_{ego} initially follows the repaired trajectory generated at the initial time step as introduced in Sec.V-A. Fig. 3d shows the configuration of scenario I at $k = 17$. The estimation of φ_1 at this time step is 0.33, based on which the repaired trajectory following the lane change maneuver violates the rule R_LC with regard to V_1. Therefore, the IA repairer continues to repair the trajectory by steering to the right while braking, which lets V_{ego} abort the lane change while also keeping a safe distance to V_2. The lateral position profile of V_{ego} is depicted in Fig. 5a. In scenario II, although V_{ego} chooses to rush into the intersection initially, the updated estimation $\hat{\varphi}_1 = 0.31$ indicates that V_1 does not intend to yield to V_{ego}, leading to a violation of the rule R_IN3. As shown in Fig. 4d, our approach slows down the ego vehicle to give way for V_1, which successfully avoids violating the rules. We present the velocity profile of V_{ego} in Fig. 5b. In conclusion, in both scenarios, when V_1 is a selfish agent with $\varphi_1 = 0.3$, V_{ego} fails to make V_1 yield to its behavior.

The experimental results from Sec. V-A and Sec. V-B demonstrate that our IA repairer enables V_{ego} to enhance its own objective by affecting the trajectory of its follower. However, this objective may not be achieved when encountering uncooperative followers. Our motion planning framework can identify this situation through courtesy factor estimation and further repair the trajectory to be rule-compliant.



(a) Initial scenario. (b) Repaired without interaction awareness. (c) Repaired with interaction awareness. (d) Repaired with interaction awareness based on φ estimation at $k = 7$.

Fig. 4: Trajectory repair results of scenario II.

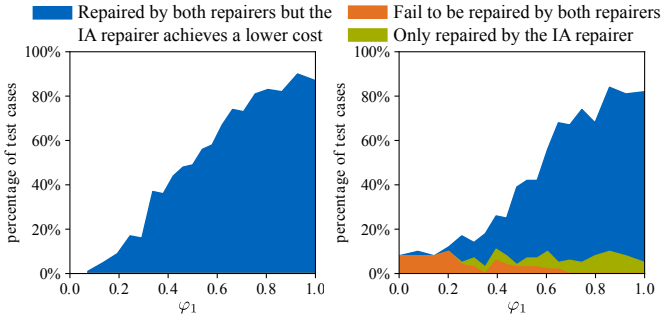


(a) Lateral position profiles of scenario I. (b) Velocity profiles of scenario II.

Fig. 5: Profiles of motion planning results. In (a), V_{ego} initially attempts to cut in front of V_1, starting from TC. However, during the lane change operation, V_{ego} steers right at TC′ to stay in its original lane since the observations indicate that V_1 is not cooperative enough for V_{ego} to complete the lane change. Similarly, in (b), V_{ego} abandons the first repair strategy (starting accelerating from TC) and instead brakes at TC′.

C. Evaluation with Randomly Sampled Initial Conditions

To further test the performance of the proposed framework and demonstrate the benefits of incorporating interaction awareness, we sample 20 φ_1 values with equal intervals within $[0, 1]$. For each φ_1 value, we generate 100 initial configurations for both scenarios by adding Gaussian noise to the initial longitudinal positions and velocities of all the vehicles in the scenarios. The means of the Gaussian noise correspond to the initial configurations used in Sec. V-A, and the standard deviations are 10 m and 2 m/s for longitudinal positions and velocities, respectively. We rule out the scenarios where the initial planned trajectories for V_{ego} are rule-compliant to highlight the performance of trajectory repair. Afterward, we use the IA repairer and the IU repairer along with the sampling-based planner to control the motion of V_{ego} for two planning cycles in all test cases. The results are shown in Fig. 6, where the green area indicates that a higher repair rate can be achieved by using the IA repairer. Overall, the IA repairer successfully repairs 97.2% of test cases, whereas the IU repairer achieves a repair rate of 91.6%. For the test cases that are successfully repaired by both strategies, we illustrate the proportion of the test cases where the IU repairer employs the conservative maneuver *brake*, while the IA repairer results in superior objectives by executing less conservative maneuvers (i.e., *lane-change-to-the-left* in scenario I and *kick-down* in scenario II). The results further demonstrate that interaction awareness leads to improved objectives, especially when interacting with traffic participants that exhibit high courtesy factors.



(a) Scenario I. (b) Scenario II.

Fig. 6: Results of the experiments based on randomly sampled initial conditions.

VI. Conclusions

We propose a novel trajectory repair algorithm that is capable of generating rule-compliant and interaction-aware repaired trajectories. Compared to a repairer that ignores interactions, the interaction-aware repairer achieves a higher repair rate and avoids unnecessary conservative repair maneuvers, especially when interacting with traffic participants that have high courtesy levels. Additionally, our repairer is capable of working within a receding-horizon motion planning framework and rectifying the rule-violating trajectory of the ego vehicle online.

Acknowledgments

This work is funded by the TUM Georg Nemetschek Institute (GIN) within the project *Smart Public Transport Infrastructure Control System - an AI-Based Approach to Ensure Safety, Accessibility & Efficiency of Public Transport* (OBACHT) and by the German Federal Ministry for Digital and Transport (BMDV) within the project *Cooperative Autonomous Driving with Safety Guarantees* (KoSi). The authors also thank Zekun Xing for implementing of the intersection traffic rules in the rule monitor.

References

[1] S. Maierhofer, A.-K. Rettinger, E. C. Mayer, and M. Althoff, "Formalization of interstate traffic rules in temporal logic," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2020, pp. 752–759.

[2] V. Raman, A. Donz, M. Maasoumy, R. M. Murray, A. Sangiovanni-Vincentelli, and S. A. Seshia, "Model predictive control with signal temporal logic specifications," in *Proc. of the IEEE Conference on Decision and Control*, 2014, pp. 81–87.

[3] M. Leucker and C. Schallhart, "A brief account of runtime verification," *The Journal of Logic and Algebraic Programming*, vol. 78, no. 5, pp. 293–303, 2009.

[4] Y. Lin and M. Althoff, "Rule-compliant trajectory repairing using satisfiability modulo theories," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2022, pp. 449–456.

[5] M. Wang, Z. Wang, J. Talbot, J. C. Gerdes, and M. Schwager, "Game theoretic planning for self-driving cars in competitive scenarios." in *Proc. of the Robotics: Science and Systems*, 2019, pp. 1–9.

[6] D. Fridovich-Keil, E. Ratner, L. Peters, A. D. Dragan, and C. J. Tomlin, "Efficient iterative linear-quadratic approximations for nonlinear multi-player general-sum differential games," in *Proc. of the IEEE International Conference on Robotics and Automation*, 2020, pp. 1475–1481.

[7] S. Le Cleac'h, M. Schwager, and Z. Manchester, "LUCIDgames: Online unscented inverse dynamic games for adaptive trajectory prediction and planning," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 5485–5492, 2021.

[8] N. Li, Y. Yao, I. Kolmanovsky, E. Atkins, and A. R. Girard, "Game-theoretic modeling of multi-vehicle interactions at uncontrolled intersections," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 2, pp. 1428–1442, 2022.

[9] L. Peters, A. Bajcsy, C.-Y. Chiu, D. Fridovich-Keil, F. Laine, L. Ferranti, and J. Alonso-Mora, "Contingency games for multi-agent interaction," *IEEE Robotics and Automation Letters*, vol. 9, no. 3, pp. 2208–2215, 2024.

[10] C. A. Holt and A. E. Roth, "The Nash equilibrium: A perspective," *Proceedings of the National Academy of Sciences*, vol. 101, no. 12, pp. 3999–4002, 2004.

[11] M. Breton, A. Alj, and A. Haurie, "Sequential Stackelberg equilibria in two-person games," *Journal of Optimization Theory and Applications*, vol. 59, pp. 71–97, 1988.

[12] D. Sadigh, S. Sastry, S. A. Seshia, and A. D. Dragan, "Planning for autonomous cars that leverage effects on human actions." in *Proc. of the Robotics: Science and Systems*, 2016, pp. 1–9.

[13] C. Li, T. Trinh, L. Wang, C. Liu, M. Tomizuka, and W. Zhan, "Efficient game-theoretic planning with prediction heuristic for socially-compliant autonomous driving," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 10 248–10 255, 2022.

[14] C. Burger, J. Fischer, F. Bieder, Ö. Ş. Taş, and C. Stiller, "Interaction-aware game-theoretic motion planning for automated vehicles using bi-level optimization," in *Proc. of the IEEE International Conference on Intelligent Transportation Systems*, 2022, pp. 3978–3985.

[15] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," *Physical review E*, vol. 62, no. 2, pp. 1805–1824, 2000.

[16] N. Evestedt, E. Ward, J. Folkesson, and D. Axehill, "Interaction aware trajectory planning for merge scenarios in congested traffic situations," in *Proc. of the IEEE International Conference on Intelligent Transportation Systems*, 2016, pp. 465–472.

[17] K. Ji, M. Orsag, and K. Han, "Lane-merging strategy for a self-driving car in dense traffic using the Stackelberg game approach," *Electronics*, vol. 10, no. 8, 2021, Art. no. 894.

[18] Z. Huang, H. Liu, J. Wu, W. Huang, and C. Lv, "Learning interaction-aware motion prediction model for decision-making in autonomous driving," in *Proc. of the IEEE International Conference on Intelligent Transportation Systems*, 2023, pp. 4820–4826.

[19] S. Maierhofer, P. Moosbrugger, and M. Althoff, "Formalization of intersection traffic rules in temporal logic," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2022, pp. 1135–1144.

[20] H. Krasowski and M. Althoff, "Temporal logic formalization of marine traffic rules," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2021, pp. 186–192.

[21] Y. Lin, H. Li, and M. Althoff, "Model predictive robustness of signal temporal logic predicates," *IEEE Robotics and Automation Letters*, vol. 8, no. 12, pp. 8050–8057, 2023.

[22] E. Irani Liu and M. Althoff, "Specification-compliant driving corridors for motion planning of automated vehicles," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 9, pp. 4180–4197, 2023.

[23] C.-I. Vasile, V. Raman, and S. Karaman, "Sampling-based synthesis of maximally-satisfying controllers for temporal logic specifications," in *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2017, pp. 3840–3847.

[24] J. Lin, W. Zhou, H. Wang, Z. Cao, W. Yu, C. Zhao, D. Zhao, D. Yang, and J. Li, "Road traffic law adaptive decision-making for self-driving vehicles," in *Proc. of the IEEE International Conference on Intelligent Transportation Systems*, 2022, pp. 2034–2041.

[25] R. Sebastiani, "Lazy satisfiability modulo theories," *Journal on Satisfiability, Boolean Modeling and Computation*, vol. 3, no. 3-4, pp. 141–224, 2007.

[26] G. Wrsching and M. Althoff, "Robust and efficient curvilinear coordinate transformation with guaranteed map coverage for motion planning," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2024.

[27] O. Maler and D. Nickovic, "Monitoring temporal properties of continuous signals," in *Proc. of the International Symposium on Formal Techniques in Real-Time and Fault-Tolerant Systems*, 2004, pp. 152–166.

[28] W. Schwarting, A. Pierson, J. Alonso-Mora, S. Karaman, and D. Rus, "Social behavior for autonomous vehicles," *National Academy of Sciences*, vol. 116, no. 50, pp. 24 972–24 978, 2019.

[29] X. Zhao, J. Sun, and M. Wang, "Measuring sociality in driving interaction," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–14, 2024, Early Access.

[30] Z. Qin, A. Ji, Z. Sun, G. Wu, P. Hao, and X. Liao, "Game theoretic application to intersection management: A literature review," *IEEE Transactions on Intelligent Vehicles*, pp. 1–19, 2024, Early Access.

[31] J. Ziegler, P. Bender, T. Dang, and C. Stiller, "Trajectory planning for Bertha - A local, continuous method," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2014, pp. 450–457.

[32] W. Xu, J. Wei, J. M. Dolan, H. Zhao, and H. Zha, "A real-time motion planner with trajectory optimization for autonomous vehicles," in *Proc. of the IEEE International Conference on Robotics and Automation*, 2012, pp. 2061–2067.

[33] R. Nieuwenhuis, A. Oliveras, and C. Tinelli, "Solving SAT and SAT modulo theories: From an abstract Davis–Putnam–Logemann–Loveland procedure to DPLL (T)," *Journal of the ACM*, vol. 53, no. 6, pp. 937–977, 2006.

[34] A. Tamke, T. Dang, and G. Breuel, "A flexible method for criticality assessment in driver assistance systems," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2011, pp. 697–702.

[35] M. Althoff, M. Koschi, and S. Manzinger, "CommonRoad: Composable benchmarks for motion planning on roads," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2017, pp. 719–726.

[36] M. Werling, S. Kammel, J. Ziegler, and L. Grll, "Optimal trajectories for time-critical street scenarios using discretized terminal manifolds," *The International Journal of Robotics Research*, vol. 31, no. 3, pp. 346–359, 2012.

[37] C. Pek, S. Manzinger, M. Koschi, and M. Althoff, "Using online verification to prevent autonomous vehicles from causing accidents," *Nature Machine Intelligence*, vol. 2, no. 9, pp. 518–528, 2020.

[38] C. Pek, P. Zahn, and M. Althoff, "Verifying the safety of lane change maneuvers of self-driving vehicles based on formalized traffic rules," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2017, pp. 1477–1483.