Efficient Computation of Invariably Safe States for Motion Planning of Self-driving Vehicles

Christian Pek[1] and Matthias Althoff[2]

Abstract— **Safe motion planning requires that a vehicle reaches a set of safe states at the end of the planning horizon. However, safe states of vehicles have not yet been systematically defined in the literature, nor does a computationally efficient way to obtain them for online motion planning exist. To tackle the aforementioned issues, we introduce invariably safe sets. These are regions that allow vehicles to remain safe for an infinite time horizon. We show how invariably safe sets can be computed and propose a tight under-approximation which can be obtained efficiently in linear time with respect to the number of traffic participants. We use invariably safe sets to lift safety verification from finite to infinite time horizons. In addition, our sets can be used to determine the existence of feasible evasive maneuvers and the criticality of scenarios by computing the time-to-react metric.**

I. Introduction

Self-driving vehicles promise many advantages over human-driven vehicles, most notably enhanced road safety. In order to achieve safety, motion plans of vehicles must be collision-free and reach a safe state at the end of the planning horizon. The latter requirement raises the question how to define and efficiently compute safe states of vehicles?

Unfortunately, this question has not yet been adequately answered. Safe motion planning of vehicles in dynamic environments involves many complex aspects, e. g., collision-avoidance, ensuring drivability of trajectories, and considering uncertain future motions of obstacles. Most works therefore focus on solving smaller subsets of the aforementioned challenges [1], e. g., being collision-free for a finite time horizon. However, these simplifications may lead to unsafe situations that endanger passengers of self-driving vehicles and other traffic participants.

Various governmental institutions have also identified the issue of unsatisfying safety definitions [2]. Legislative powers already try to impose requirements for developing and testing self-driving vehicles. Nonetheless, they clarify that defining safe states, especially in terms of motion planning, is an open problem which needs to be solved urgently [3, p. 13]. In particular, in emergency situations in which the vehicle must react in a timely manner, obtaining a safe state efficiently is crucial for protecting the lives of humans.

A. Literature Overview

Common motion planning algorithms, such as [4]–[6], check if each state of the planned trajectory is collision-free within a finite planning horizon. However, collisions might be unavoidable directly after the horizon. For this reason, one must ensure that the vehicle reaches a set of safe states at the end of the planning horizon.

Controlled invariant sets (CIS) [7]–[9] are used to guarantee persistent feasibility. By definition, for every state within a CIS, there exists at least one feasible trajectory which keeps the system within the set indefinitely long; thus, the vehicle remains safe. Unfortunately, obtaining CIS to guarantee safety in dynamic environments is challenging due to the unknown future motion of obstacles.

Instead of computing sets, one can also check if the final state is an inevitable collision state (ICS) [10] to reason over infinite time horizons. ICSs are states in which the vehicle, no matter what trajectory it executes, eventually collides with an obstacle [11], [12]. Determining ICSs is computationally intense, and most works can only handle a single trajectory prediction of traffic participants for online computation. Consequently, ICSs also suffer from the uncertain future motion of obstacles.

Reachability analysis can check if states are collision-free while accounting for any feasible future motion of dynamic obstacles [13]–[15]. Reachable sets are the set of states reachable for a system from a set of initial states. Future collisions of a vehicle can be identified by checking for intersections of its reachable set with the ones of obstacles. Reachable sets are also used to determine ICSs [16], [17]. However, reachability analysis comes with the disadvantage that unsafe regions may grow rapidly over time since any feasible future motion of obstacles is considered.

For criticality assessment of motion plans, the time-to-collision (TTC) metric has been proposed. The TTC is the time until a collision occurs based on motion predictions of obstacles and the intended trajectory of the vehicle [18], [19]. The time-to-react (TTR) [20], [21] is considered a more informative metric since it is the remaining time along the intended trajectory until one can avoid a collision. However, most approaches for criticality assessment limit the search for a feasible maneuver to discrete sets of pre-planned trajectories [22].

B. Contribution

This paper formally defines and computes invariably safe sets, which are regions in which vehicles are able to remain safe for an infinite time horizon. This concretizes our previous work [23] for a lane-based setting, which considers arbitrary traffic scenarios. In contrast to computationally expensive approaches (cf. Sec. I-A), we show

[1] Christian Pek is with the Department of Computer Science, Technical University of Munich, D-85748 Garching, Germany and BMW Group, D-85716 Unterschleissheim, Germany. Christian.Pek@bmw.de

[2] Matthias Althoff is with the Department of Computer Science, Technical University of Munich, D-85748 Garching, Germany. althoff@in.tum.de

that an under-approximation of invariably safe sets can be computed efficiently in linear time with respect to the number of traffic participants while maintaining formal safety guarantees. We demonstrate the tightness of our proposed under-approximation by comparing it to over-approximative reachable sets. The obtained invariably safe sets can be used in online motion planning to

1) lift verification from finite time horizons to infinite time horizons (cf. Def. 7),
2) determine the existence of feasible evasive trajectories (cf. Rem. 1), and
3) evaluate the criticality of scenarios (cf. Def. 8).

The remainder of this paper is organized as follows: Sec. II introduces necessary models and assumptions. Sec. III formally derives invariably safe sets and Sec. IV demonstrates the usage of invariably safe sets for motion planning. In Sec. V, we present an algorithm to efficiently compute an under-approximation of invariably safe sets. Benefits of our proposed approach are demonstrated by examples in Sec. VI. We finish with conclusions in Sec. VII.

II. Models and Assumptions

Let us introduce $\mathcal{X} \subset \mathbb{R}^n$ as the possible set of states x and $\mathcal{U} \subset \mathbb{R}^m$ as the set of admissible control inputs u of a self-driving vehicle, whose motions are governed by the differential equation

$$\dot{x}(t) = f\big(x(t), u(t)\big). \tag{1}$$

Without loss of generality, we assume that the initial time is $t_0 = 0$, and we adhere to the notation $u\big([0, t_h]\big)$ to describe an input trajectory for the time interval $[0, t_h]$. In addition, $\chi\big(t, x(0), u([0, t_h])\big)$ denotes the solution of (1) at time $t \in [0, t_h]$ subject to the initial state $x(0) = x_0$ and the input trajectory $u\big([0, t_h]\big)$. By an abuse of notation, we use $u\big([t_1, t_2]\big) = \Phi\big(x([t_1, t_2]), r_{\mathrm{ref}}\big), t_1 \leq t_2$, to emphasize that an input trajectory is generated by a feedback control law Φ for a given reference r_{ref}.

The lane-based environment $\mathcal{W} \subset \mathbb{R}^k$ of (1) is modeled as a subset of the Euclidean space. The set $\mathcal{B} \subset \mathbb{N}$ contains indices referring to all safety-relevant dynamic and static obstacles, typically obtained using on-board sensors [24]. We use $v \geq 0$ to denote velocities and a to denote accelerations.

We assume the existence of a set-based prediction, e. g., [25], which considers any feasible future motion (including initial uncertainties) of dynamic obstacles to account for their uncertain future motion. The set of possibly occupied points of dynamic obstacles at a certain point in time t is represented by an occupancy set:

Definition 1 (Occupancy Set $\mathcal{O}$)
The occupancy set $\mathcal{O}(t)$ describes the set of (possibly) occupied points by an obstacle at a point in time t. For a time interval $[t_1, t_2], t_1 \leq t_2$, we define $\mathcal{O}\big([t_1, t_2]\big) = \bigcup_{t_1 \leq t \leq t_2} \mathcal{O}(t)$.

In order to realize efficient collision checking, we introduce a relation from the configuration space of (1) to the environment in world coordinates:

Definition 2 (Relation to Environment occ)
The operator $\mathrm{occ}(x) : \mathcal{X} \to \mathcal{P}\big(\mathcal{W}\big)$ relates the state vector x to the set of points occupied in $\mathcal{W}$, where $\mathcal{P}(\mathcal{W})$ is the power set of $\mathcal{W}$. Given a set $\mathcal{X}$, we define the operator $\mathrm{occ}(\mathcal{X}) := \{\mathrm{occ}(x) \,|\, x \in \mathcal{X}\}$.

We are now able to define the maximal set of collision-free states at a point in time t (cf. Fig. 1):

Definition 3 (Collision-free States $\mathcal{F}$)
The set $\mathcal{F}^t \subseteq \mathcal{X}$ is the maximal set of states of (1) which are collision-free at time t: $\mathrm{occ}\big(\mathcal{F}^t\big) \cap \mathcal{O}_{\mathcal{B}}(t) = \emptyset, \mathcal{O}_{\mathcal{B}}(t) = \bigcup_{i \in \mathcal{B}} \mathcal{O}_i(t)$.

In Sec. III, we use backward reachability to compute the set of states from which (1) is able to reach a goal set collision-free [26].

Definition 4 (Collision-free Backward Reachable Set $\mathcal{R}$)
The collision-free backward reachable set $\mathcal{R} \subseteq \mathcal{X}$ is the set of states from which (1) is able to reach a goal set $\mathcal{X}_f \subset \mathcal{X}$ collision-free within a certain time $t \geq 0$ considering the set of inputs $\mathcal{U}$:

$$\begin{aligned}\mathcal{R}\big(t, \mathcal{O}([t_f - t, t_f]), \mathcal{X}_f\big) := \big\{x \,\big|\, \exists r \in [0, t] : \forall \xi \in [t_f - r, t_f] :\\ \mathrm{occ}\big(\chi(\xi, x, u([t_f - r, t_f]))\big) \cap \mathcal{O}(\xi) = \emptyset, u(\xi) \in \mathcal{U},\\ \chi\big(t_f, x, u([t_f - r, t_f])\big) \in \mathcal{X}_f\big\}.\end{aligned}$$

Note that in this work we do not consider the correctness of software components of self-driving vehicles with respect to their specification, as discussed in [27], [28], for example. Furthermore, we assume redundant hardware, allowing us to ignore hardware faults, e. g., as described in [29] or [30]. Lastly, we do not incorporate the influence of psychological and social aspects on safety [31].

III. Invariably Safe Sets

A. Definition

We define safe states by making use of recursion: we call a state safe if a collision-free trajectory to another safe state exists. This recursive definition allows us to derive subsets of $\mathcal{F}^t$ (cf. Fig. 1), which only contain states that guarantee a safe transition to another safe state for an infinite time horizon. By definition, these subsets do not include ICSs and thus, are invariably safe.

Definition 5 (Invariably Safe Set $\mathcal{S}$)
The Invariably Safe Set $\mathcal{S}^t$ for a point in time t allows (1) to be safe for an infinite time horizon and is defined as

$$\mathcal{S}^t := \big\{x \in \mathcal{F}^t \,\big|\, \forall \tau > t : \chi\big(\tau, x, \Phi(x([t, \tau], r_{\mathrm{ref}})\big) \in \mathcal{F}^\tau\big\}.$$

Determining the maximal invariably safe set is again a computationally demanding task in most scenarios. However, we show that an under-approximation of the maximal invariably safe set can be computed efficiently from a known safe set.

B. Backwards Computation of Invariably Safe Sets

Let us focus on finding an invariably safe set which allows us to inductively derive other invariably safe sets. Based on traffic rules [32, Art. 13 and Art. 31], we can state that if a preceding obstacle comes to a stop, being in standstill behind it within a certain area is a safe state.

We introduce $\Gamma(b,\beta) \subset \mathcal{W}$ as the allowed area in a lane for standstill behind a stopped obstacle $b \in \mathcal{B}$ within a distance β which is at least as long as the length of the self-driving vehicle. We show that the set of collision-free states behind a preceding obstacle within Γ is an invariably safe set.

Lemma 1 (Invariably Safe Set $\mathcal{S}^\tau$ at Standstill)
Assuming that the preceding obstacle b stops at any future time $\tau > t$, the set $\mathcal{S}^\tau := \{x \,|\, v_{[x]} = 0 \wedge \mathrm{occ}(x) \subseteq \Gamma(b,\beta) \wedge \mathrm{occ}(x) \cap \mathcal{O}_{\mathcal{B}}(\tau) = \emptyset\}$ is an invariably safe set according to Def. 5, where $v_{[x]}$ describes the velocity in state x.

Proof: By definition, states $x \in \mathcal{S}^\tau$ are collision-free. Thus, $\mathcal{S}^\tau \subseteq \mathcal{F}^\tau$. All $x \in \mathcal{S}^\tau$ remain collision-free $\forall \tau' > \tau$ by choosing $u(\tau')$ such that $v_{[x(\tau')]} = 0$. ■

Let us now use backward reachable sets (cf. Def. 4) to derive invariably safe sets for times prior to τ. In order to make use of induction, we determine the sets for time intervals prior to τ, rather than single points in time. Therefore, $\mathcal{S}(k) := \mathcal{S}^{\tau(k)}, k \in \mathbb{N}_+$, denotes the invariably safe set for the time interval $\tau(k) := [\tau - k\epsilon, \tau - (k-1)\epsilon]$, prior to τ, where $\epsilon \in \mathbb{R}_+$ is an arbitrarily small step size.

Theorem 1 (Determining Invariably Safe Sets)
The invariably safe set for the time interval $\tau(k)$ is $\mathcal{S}(k) := \mathcal{S}^{\tau(k)} = \mathcal{R}\big(\epsilon, \mathcal{O}_{\mathcal{B}}(\tau(k)), \mathcal{S}(k-1)\big)$, where $\mathcal{S}(0) = \mathcal{S}^\tau$.

Proof: We prove the theorem inductively.
Base case ($k = 1$): $\mathcal{S}(1) = \mathcal{S}^{[\tau-\epsilon,\tau]} = \mathcal{R}\big(\epsilon, \mathcal{O}_{\mathcal{B}}([\tau - \epsilon, \tau], \mathcal{S}^\tau)\big)$. For every state $x \in \mathcal{S}(1)$, there exists a collision-free trajectory to the invariably safe set $\mathcal{S}^\tau$ (cf. Lem. 1), i. e., $\forall x \in \mathcal{S}(1) : \exists r \leq \epsilon : \exists u\big([\tau - r, \tau]\big) : \chi\big(\tau, x, u([\tau - r, \tau])\big) \in \mathcal{S}^\tau$ to remain safe for times $\tau' > \tau$.
Inductive step: assuming $\mathcal{S}(k), k = c$ is an invariably safe set for any random integer $c \in \mathbb{N}$, we show that $\mathcal{S}(k+1)$ is an invariably safe set. $\mathcal{S}(k+1) = \mathcal{R}\big(\epsilon, \mathcal{O}_{\mathcal{B}}(\tau(k+1)), \mathcal{S}(k)\big)$ allows us to determine a collision-free trajectory to $\mathcal{S}(k)$ for every state $x \in \mathcal{S}(k+1)$ (analogous to base case). Since $\mathcal{S}(k)$ is an invariably safe set, every invariably safe set $\mathcal{S}(j), j \leq k$, is reachable from $\mathcal{S}(k+1)$ collision-free.

■

C. Under-approximation of Invariably Safe Sets

We efficiently derive a tight under-approximation of $\mathcal{S}^t$ (cf. Fig. 1) using 1) formal safe distances for vehicle following according to [33] and 2) evasive distances according to [34]. Evasive distances allow changing to an adjacent lane while respecting formal safe distances to obstacles in the target lane and are usually shorter than safe distances for higher velocities [35].

Proposition 1 (Under-Approximation of $\mathcal{S}^t$)
The union of the set $\mathcal{S}_1^t$ of states respecting safe distances [33] and of the set $\mathcal{S}_2^t$ respecting evasive distances [34] to a preceding obstacle at time t is an under-approximation of $\mathcal{S}^t$, i. e., $\mathcal{S}_1^t \cup \mathcal{S}_2^t \subset \mathcal{S}^t$.

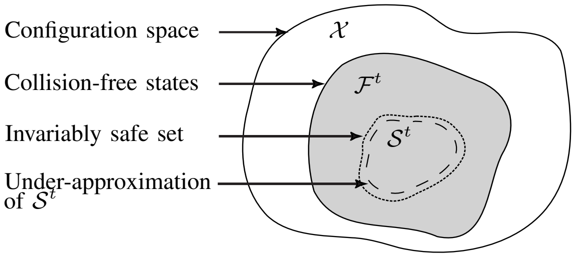


Fig. 1: Relation of the configuration space $\mathcal{X}$, collision-free states $\mathcal{F}^t$, and invariably safe sets $\mathcal{S}^t$.

Proof: The soundness of safe and evasive distances has been shown for all cases in [33], [34], [36]. To show that the resulting set is an under-approximation, we provide a counterexample: based on [35], the last possible evasive maneuver must be a combination of braking and steering. ■

IV. Impacts on Motion Planning

Invariably safe sets offer many advantages for motion planning of self-driving vehicles. Usually, planned trajectories $u\big([0, t_h]\big)$ are verified as collision-free within the optimization horizon prior to their execution:

Definition 6 (Collision-free Input Trajectory)
An input trajectory $u\big([0, t_h]\big), 0 < t_h$, is called a collision-free input trajectory for the time horizon t_h if $\forall t \in [0, t_h]$: $\chi\big(t, x(0), u([0, t_h])\big) \in \mathcal{F}^t$.

However, the vehicle may not remain safe for times $t' > t_h$. Furthermore, the feasibility of trajectories starting at t_h may not be ensured, which is important for cyclic replanning approaches such as model predictive control. Our invariably safe sets guarantee both properties, since a feasible and collision-free trajectory exists at any time:

Definition 7 (Invariably Safe Input Trajectory)
An input trajectory $u\big([0, t_h]\big), 0 < t_h$, is called an invariably safe input trajectory if $u\big([0, t_h]\big)$ is a collision-free input trajectory (cf. Def 6) and $\chi\big(t_h, x(0), u([0, t_h])\big) \in \mathcal{S}^{t_h}$.

Note that if collisions occur due to misbehaviors of other obstacles, e. g., crashing into a tailback or performing a cut-in, the vehicle would not be accountable, since the obstacle violated traffic rules [32].

The existence of a feasible and collision-free trajectory[1] to another safe state can also be advantageously exploited in time-critical emergency situations.

Remark 1 (Existence of Evasive Trajectories)
Even in emergency situations, invariably safe sets guarantee the existence of a feasible trajectory to avoid a collision.

Another use of our invariably safe sets is to obtain the *time-to-react* (TTR) [20, Sec. II].

Definition 8 (Time-To-React)
Assuming that $x(0) \in \mathcal{S}^0$, the time-to-react (TTR) is the maximum time the vehicle can continue the current trajectory $u\big([0, t_h]\big)$ for which the existence of an evasive

[1] Note that computing such a trajectory is not the focus of this work.

trajectory is guaranteed, i. e., $t_{\mathrm{TTR}} := \sup\big\{t \,|\, t \in [0, t_h] \wedge \chi\big(t, x(0), u([0, t_h])\big) \in \mathcal{S}^t\big\}$.

V. Algorithmic Realization for Online Motion Planning

In this section, we demonstrate how the under-approximation of $\mathcal{S}^t$ can be computed efficiently.

A. Environment

We make use of a curvilinear coordinate system [4] aligned with the driving direction of the lane (cf. coordinate system in Fig. 2) to describe the environment $\mathcal{W}$ and the occupancies of obstacles. The state of the self-driving vehicle is modeled as $x = (s, d, v)^T \in \mathbb{R}^3$, where s is the longitudinal position, d the lateral position, and v the velocity along the lane. Positions $(s, d)^T$ describe the geometric center of the vehicle and ℓ denotes its length.

To account for the limited field of view of the self-driving vehicle, we place static obstacles at its border to guarantee that the vehicle is able to stop within its sensor view. Road boundaries and varying lane widths are integrated by limiting the allowed lateral positions d.

We divide the drivable area of a lane in sections $\mathcal{C}_{b_i,b_j} \subset \mathcal{W}, b_i, b_j \in \mathcal{B}$, delimited by the occupancies of a pair of obstacles (cf. Fig. 4a). For instance, for obstacle b_1 and obstacle b_2 and occupancies $\mathcal{O}_1(t)$ and $\mathcal{O}_2(t)$, respectively, $\mathcal{C}_{b_1,b_2} = \{s \in \mathbb{R} \,|\, \forall s_1 \in \mathcal{O}_1(t), \forall s_2 \in \mathcal{O}_2(t) : s_1 + {}^{\ell_{\mathrm{ego}}}\!/_2 < s < s_2 - {}^{\ell_{\mathrm{ego}}}\!/_2 + \Delta s_{2,\mathrm{stop}}\}$, where $\Delta s_{2,\mathrm{stop}}$ is the stopping distance of obstacle b_2.

B. Occupancy Prediction

We make use of the set-based prediction tool *SPOT* [25] and the motion assumptions listed in Tab. I to predict the occupancies $\mathcal{O}_{\mathcal{B}}(t)$ of obstacles $\mathcal{B}$ at time t. In addition, we enlarge $\mathcal{O}_{\mathcal{B}}(t)$ for collision checking by adding the dimensions of the self-driving vehicle [37]. Note that the set of motion assumptions is not binding in our approach: if obstacles violate certain assumptions, the occupancies become larger and our obtained safe sets smaller.

TABLE I: List of motion assumptions based on [32].

Assumption	**Description**
A_{amax}	Maximum absolute accelerations $\lvert a_{\max,b}\rvert \geq \lvert a_{\max,ego}\rvert$ of traffic participants $b \in \mathcal{B}$ are known
A_{safe}	Safe distances to other vehicles have to be respected to comply with traffic rules
A_{vmax}	Positive longitudinal acceleration is stopped when a parameterized speed $v_{\max}$ is reached
A_{back}	Driving backwards in a lane is not allowed, i. e., $v \geq 0$
A_{lane}	Changing the lane is only allowed if the new lane has the same driving direction
A_{over}	While being overtaken, a vehicle is not allowed to accelerate

C. Algorithm

Without loss of generality, we assume that the intended routes of the self-driving vehicle are given. Alg. 1 computes the under-approximation of $\mathcal{S}_1^t \cup \mathcal{S}_2^t \subset \mathcal{S}^t$ for a time t and a section $\mathcal{C}_{b_i,b_j}$ along arbitrary road networks (cf. Fig. 4). The algorithm must be applied to every section, which is parallelizable.

a) Velocity and acceleration constraints: We determine the maximum feasible velocity $v_{\mathrm{crit}}(s), s \in \mathcal{C}_{b_i,b_j}$ considering the lane's curvature (cf. dashed line in Fig. 3) in line 2 of Alg. 1. Our approach further incorporates any given legal speed limits $v_{\mathrm{limit}}(s)$ (cf. solid line in Fig. 3). The resulting maximum velocity constraints are given by $v_{\max}(s) = \min\big(v_{\mathrm{crit}}(s), v_{\mathrm{limit}}(s)\big)$. In lines 4-5, we compute the feasible lateral and longitudinal accelerations, $a_d(v)$ and $a_s(v)$, for all possible velocities and the lane's curvature $\kappa(s), s \in \mathcal{C}_{b_i,b_j}$ based on [38, Eq. 2-4].

b) Safe distance: Line 7 of Alg. 1 computes the safe distance to a preceding obstacle b_j with velocity v_j for a provided vehicle velocity v and reaction time δ_{brake} by $\Delta^t_{\mathrm{safe}}(v, b_j)$ according to [33, Eq. 17]. Variables $a_{d,\max}$ and $a_{s,\max}$ denote the maximum feasible accelerations in lateral and longitudinal direction, respectively.

c) Evasive distance: The distance $d_{\mathrm{eva}}(d)$ describes the lateral distance necessary to fully enter an adjacent lane from a given lateral position d. For the sake of clarity, we omit the dependence on d in Alg. 1. Line 8 computes the time $t_{\mathrm{eva}}(v)$, required for the swerving maneuver over d_{eva} considering the available lateral acceleration $a_d(v)$ and reaction time δ_{steer} [34, Eq. 11]. In line 9, we translate $t_{\mathrm{eva}}(v)$ into a formal evasive distance by $\Delta^t_{\mathrm{eva}}(v, b_j)$ [34, Eq. 12-13].

Algorithm 1 invariablySafeSets()

```
Input: $t$, $\mathcal{C}_{b_i,b_j}$, $\kappa$, $v_{\mathrm{limit}}$, $\mathcal{O}_j(t)$, $v_j(t)$, $\delta_{\mathrm{brake}}$, $\delta_{\mathrm{steer}}$
Output: Under-approximation of $\mathcal{S}^t$
    a) Acceleration constraint subroutines [38, Eq. 2-4]:
 1: $r_{\min} \leftarrow \min\big({}^{1}\!/_{|\kappa(\mathcal{C}_{b_i,b_j})|}\big)$
 2: $v_{\mathrm{crit}} \leftarrow \sqrt{r_{\min} a_{d,\max,\mathrm{ego}}}$
 3: $v_{\max} \leftarrow \min\big(v_{\mathrm{crit}}, v_{\mathrm{limit}}\big)$
 4: Let $a_d(v) := a_{d,\max,\mathrm{ego}}\big({}^{v}\!/_{v_{\mathrm{crit}}}\big)^2$
 5: Let $a_s(v) := a_{s,\max,\mathrm{ego}}\sqrt{1 - \big({}^{v^2}\!/_{v^2_{\mathrm{crit}}}\big)^2}$
    b) Safe distance subroutines [33, Eq. 17]:
 6: Let $\zeta(v, b_j) := \big({}^{v_j^2}\!/_{-2|a_{s,\max,j}|}\big) - \big({}^{v^2}\!/_{-2|a_s(v)|}\big) + \delta_{\mathrm{brake}} v$
 7: Let $\Delta^t_{\mathrm{safe}}(v, b_j) := \max\big(\zeta(v, b_j), 0\big)$
    c) Evasive distance subroutines [34, Eq. 11-13]:
 8: Let $t_{\mathrm{eva}}(v) := \sqrt{\big({}^{2d_{\mathrm{eva}}}\!/_{(a_{d,\max,\mathrm{ego}} - a_d(v))}\big)} + \delta_{\mathrm{steer}}$
 9: Let $\Delta^t_{\mathrm{eva}}(v, b_j) := v t_{\mathrm{eva}}(v) - (v_j(t) t_b - \frac{1}{2} a_{s,\max,j} t_b^2)$,
          $t_b = \min\big({}^{v_j(t)}\!/_{a_{s,\max,j}}, t_{\mathrm{eva}}(v)\big)$
    d) Invariably safe sets $\mathcal{S}_1^t$ and $\mathcal{S}_2^t$:
10: $\mathcal{S}_1^t \leftarrow \{(s, d, v)^T \in \mathcal{X} \,|\, \forall s_j \in \mathcal{O}_j(t) : s \leq s_j - \Delta^t_{\mathrm{safe}}(v, b_j)$
          $\wedge v \leq v_{\max} \wedge s \in \mathcal{C}_{b_i,b_j}\}$
11: $\mathcal{S}_2^t \leftarrow \{(s, d, v)^T \in \mathcal{X} \,|\, \forall s_j \in \mathcal{O}_j(t) : s \leq s_j - \Delta^t_{\mathrm{eva}}(v, b_j) \wedge$
          $v \leq v_{\max} \wedge s \in \mathcal{C}_{b_i,b_j} \wedge (\forall r \in [0, t_{\mathrm{eva}}(v)] :$
          $(s + vr, d', v)^T \in \mathcal{S}_1^{t+r})\}$
12: return $\mathcal{S}_1^t \cup \mathcal{S}_2^t$
```

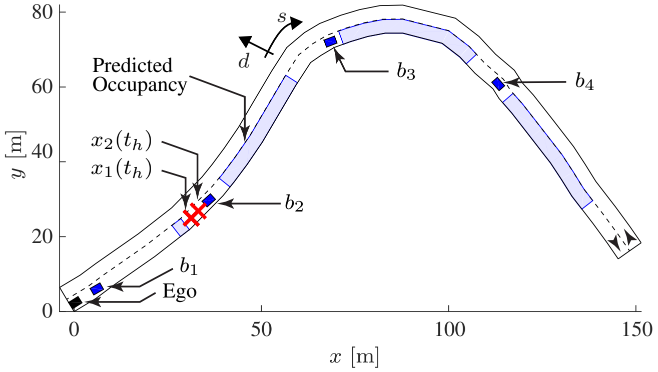


Fig. 2: Urban scenario with dynamic obstacles $b_i, i \leq 4$ and their predicted occupancies at $t_h = 3.5\,\mathrm{s}$ (light blue). Final positions, $x_1(t_h)$ and $x_2(t_h)$, of the two overtaking trajectories of the ego vehicle are shown in red.

d) Invariably safe sets $\mathcal{S}_1^t$ and $\mathcal{S}_2^t$: We compute the set $\mathcal{S}_1^t$ of states which respect a safe distance to preceding obstacles at time t using the predicted occupancies. The set $\mathcal{S}_2^t$ contains states which respect the evasive distance to preceding obstacles and a safe distance to obstacles on the adjacent lane; we check this by finding states $(s+vr, d', v)^T \in \mathcal{S}_1^{t+r}$. Note that we can also consider safe distances to following obstacles to prohibit the vehicle from directly merging in front of another obstacle during lane changes; this is omitted in Alg. 1 for the sake of clarity but can be obtained analogously to preceding obstacles.

D. Computational Complexity

Assuming that the prediction is given, the computational complexity of computing the under-approximation of $\mathcal{S}^t$ for all sections is $O(n)$ with $n = |\mathcal{B}|$, as one has to perform a constant number of calculations per section.

VI. Evaluation

In this section, we compute the under-approximation for different scenarios and demonstrate its usage for motion planning. We implemented Alg. 1 in MATLAB R2015b on a machine with an Intel i5-4260U 1.4GHz processor and 8GB of DDR3 1600MHz memory and use the MPT toolbox V3.0 [39] to visualize $\mathcal{S}^t$ by approximating it with half-spaces. We denote the self-driving vehicle as the ego vehicle and the under-approximation as $\mathcal{S}$ in the following.

A. Trajectory Verification

We investigate an urban scenario[2] (cf. Fig. 2) to illustrate the verification of trajectories for infinite time horizons (cf. Def. 7). The scenario consists of two lanes (direction of travel indicated by arrows). Four other traffic participants $b_i, i \leq 4$ occupy the lane of the ego vehicle (parameters given in Tab. II). The feasible velocity profile and the speed limit are shown in Fig. 3. The task for the ego vehicle is to overtake the preceding vehicle b_1.

We plan two overtaking trajectories $u_1\big([0, t_h]\big)$ and $u_2\big([0, t_h]\big)$ with equal time horizons $t_h = 3.5\,\mathrm{s}$, but differing goal velocities, $10.3\,\mathrm{m/s}$ and $11.1\,\mathrm{m/s}$, respectively (final positions are shown as red crosses in Fig. 2).

TABLE II: Parameters of the urban scenario.

Parameter	**Value**
Ego vehicle	$(s, d, v)^T_{\mathrm{ego}} = (1.5\,\mathrm{m}, 0\,\mathrm{m}, 8.3\,\mathrm{m/s})^T$
Vehicle b_1	$(s, d, v)^T_{\mathrm{b1}} = (8.5\,\mathrm{m}, 0\,\mathrm{m}, 6.9\,\mathrm{m/s})^T$
Vehicle b_2	$(s, d, v)^T_{\mathrm{b2}} = (43.8\,\mathrm{m}, 0\,\mathrm{m}, 11.1\,\mathrm{m/s})^T$
Vehicle b_3	$(s, d, v)^T_{\mathrm{b3}} = (101.7\,\mathrm{m}, 0\,\mathrm{m}, 8.3\,\mathrm{m/s})^T$
Vehicle b_4	$(s, d, v)^T_{\mathrm{b4}} = (150.9\,\mathrm{m}, 0\,\mathrm{m}, 11.1\,\mathrm{m/s})^T$
Vehicle lengths	$\ell = 3.0\,\mathrm{m}$
Speed limit v_{limit}	$v_1 = 11.1\,\mathrm{m/s}$, $v_2 = 8.3\,\mathrm{m/s}$, $v_3 = 13.9\,\mathrm{m/s}$
Maximum acceleration	$\lvert a_{s,\max}\rvert = 8.0\,\mathrm{m/s^2}$, $\lvert a_{d,\max}\rvert = 3.0\,\mathrm{m/s^2}$
Reaction times	$\delta_{\mathrm{brake}} = 0.3\,\mathrm{s}$, $\delta_{\mathrm{steer}} = 0.1\,\mathrm{s}$

Using SPOT and Alg. 1, we compute $\mathcal{S}^t$ for the initial scenario at $t = 0\,\mathrm{s}$ and for the end of the planning horizon at $t_h = 3.5\,\mathrm{s}$ (cf. Fig. 4) in order to apply Def. 7. The computation of the under-approximation in this scenario and to check whether $x \in \mathcal{S}^{t_h}$ requires less than $0.3\,\mathrm{ms}$. Note that the predicted occupancy of vehicle b_1 is shorter due to assumption A_{over} (cf. Tab. I). Our proposed approach is able to consider safe distances to following vehicles (e. g., for overtaking). This is illustrated for vehicle b_2 in Fig. 4 by regarding states x with low velocities and small relative distances to vehicle b_2 as unsafe, i. e., $x \notin \mathcal{S}$.

The final states $x_1(t_h) = (37.2\,\mathrm{m}, 0\,\mathrm{m}, 10.3\,\mathrm{m/s})^T$ and $x_2(t_h) = (39.9\,\mathrm{m}, 0\,\mathrm{m}, 11.1\,\mathrm{m/s})^T$ of $u_1\big([0, t_h]\big)$ and $u_2\big([0, t_h]\big)$, respectively, are indicated with red crosses. Both trajectories are collision-free within the time interval $[0, t_h]$ (cf. Def. 6). However, we note that $x_1(t_h) \in \mathcal{S}^{t_h}$, but $x_2(t_h) \notin \mathcal{S}^{t_h}$. Only if the ego vehicle executes the *invariably safe* input trajectory $u_1\big([0, t_h]\big)$, reaching $\mathcal{S}^{t_h}$, it can come to a stop without colliding with vehicle b_2. We validated our findings by simulating the scenario for times $t > t_h$. The simulations can be found in the video attachment of this paper and at https://mediatum.ub.tum.de/1451838.

B. Comparison to Existing Approach

In this subsection, we evaluate the tightness of our under-approximation in Sec. VI-A by computing an over-approximation using reachability analysis [17]. The obtained over-approximation provides us with the approximated set of

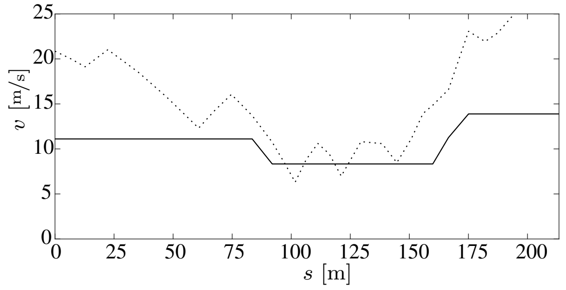
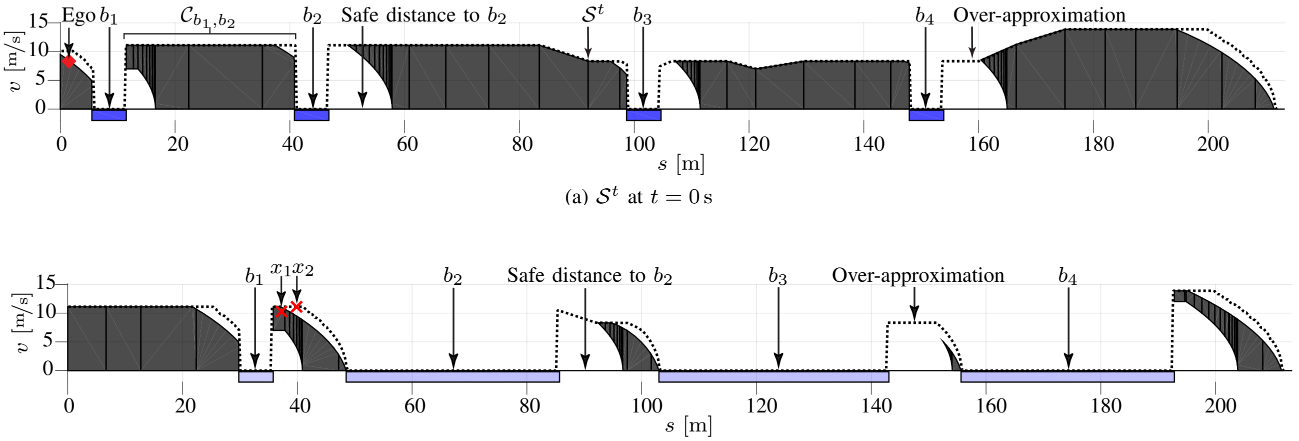


Fig. 3: Feasible velocity profile considering the curvature (dashed) and speed limit (solid) along the ego vehicle's lane.

[2] Scenario ZAM_Urban-1_1_S-1 of the CommonRoad benchmark suite Version 2018a [40].



(a) $\mathcal{S}^t$ at $t = 0\,\mathrm{s}$

(b) $\mathcal{S}^{t_h}$ at $t_h = 3.5\,\mathrm{s}$

Fig. 4: Invariably safe set $\mathcal{S}^t$ (gray) for scenario in Fig. 2 at $t = 0\,\mathrm{s}$ (a) and at $t_h = 3.5\,\mathrm{s}$ (b) as a projection on the s-v-plane. The over-approximation is shown as a dashed line, occupancies in blue, and initial and final ego positions in red.

states for which it may be possible to find a collision-free trajectory under the given velocity constraints.

The approximated outer boundary of the over-approximation is illustrated as a dashed line in Fig. 4. The computation of the boundary took about $1\,\mathrm{s}$ per sampled longitudinal position s. The boundary of the maximal invariably safe set $\mathcal{S}$ must be located between the boundary of our proposed under-approximation and the computed over-approximation. The largest deviation between the under-approximation and over-approximation is $\Delta s = 3.1\,\mathrm{m}$, which is less than a typical vehicle length and thus our under-approximation can be considered as tight.

C. Evasive Trajectories

We present a safety-critical scenario in which the ego vehicle is endangered by a cut-in vehicle (parameters given in Tab. III). The ego vehicle is driving in the right lane of a straight two-lane motorway with $v_{\mathrm{ego}} = 20\,\mathrm{m/s}$. Vehicle b_1 is driving on the adjacent lane with $v_1 = 13.5\,\mathrm{m/s}$ and relative distance $\Delta s = 15.0\,\mathrm{m}$ to the ego vehicle and suddenly changes to the lane of the ego vehicle (cf. Fig. 5a).

We compute $\mathcal{S}^t$ at $t = 0\,\mathrm{s}$ to check if the ego vehicle remains safe and see that $x_{\mathrm{ego}}(0) \in \mathcal{S}^0$ (cf. red diamond in Fig. 5a). Thus, an evasive trajectory to the left lane exists to remain safe if vehicle b_1 suddenly performs emergency braking after merging.

The intended trajectory $u([0, t_h])$ of the ego vehicle is traveling at constant speed and is illustrated in $50\,\mathrm{ms}$ time steps in Fig. 5a. We obtain $t_{\mathrm{TTR}} = 0.15\,\mathrm{s}$ (computed by applying Def. 8), which corresponds to a high criticality so that the evasive maneuver must be executed as soon as vehicle b_1 starts braking.

Fig. 5b shows the corresponding evasive maneuver, which has been obtained using a sampling-based planner. The maneuver starts at $x(t_{\mathrm{TTR}})$ along $u([0, t_h])$. The predicted positions of both vehicles at $t = t_{\mathrm{TTR}} + t_{\mathrm{eva}} = 0.99\,\mathrm{s}$, where t_{eva} is the time required for the ego vehicle to reach the adjacent lane, are shown in Fig. 5b. In a next step, we increase the complexity of the scenario: the left lane is blocked by a static obstacle, illustrated in Fig. 5c. In this situation, a safe solution to avoid a collision with b_1 exists if the ego vehicle is allowed to use the shoulder lane.

TABLE III: Parameters of the cut-in scenario.

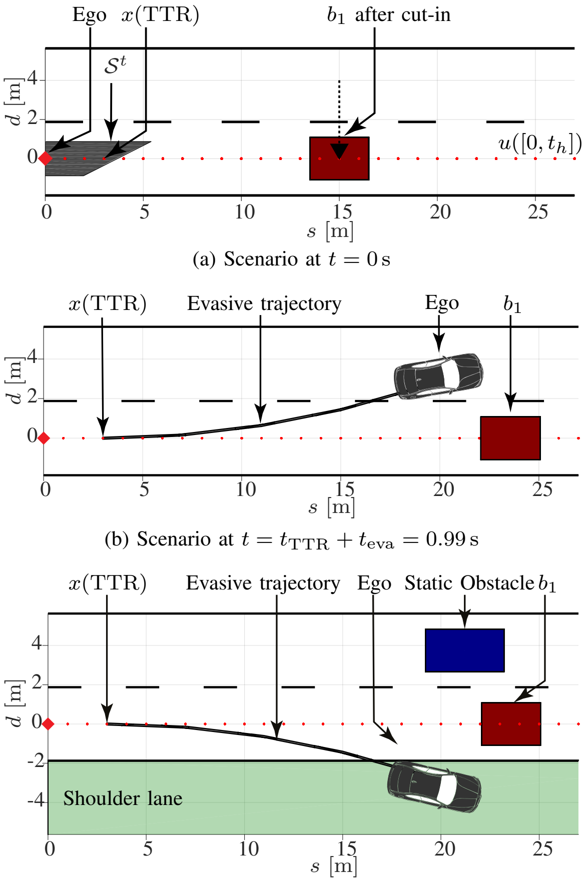
Parameter	**Value**
Ego vehicle	$(s, d, v)^T_{\mathrm{ego}} = (0\,\mathrm{m}, 0\,\mathrm{m}, 20.0\,\mathrm{m/s})^T$
Vehicle b_1	$(s, d, v)^T_{\mathrm{b1}} = (15.0\,\mathrm{m}, 3.75\,\mathrm{m}, 13.5\,\mathrm{m/s})^T$
Vehicle lengths	$\ell = 3.0\,\mathrm{m}$
Evasive distance	$d_{\mathrm{eva}} = 3.75\,\mathrm{m}$
Maximum acceleration	$\lvert a_{s,\max}\rvert = 8.0\,\mathrm{m/s^2}$, $\lvert a_{d,\max}\rvert = 8.0\,\mathrm{m/s^2}$
Reaction times	$\delta_{\mathrm{brake}} = 0.3\,\mathrm{s}$, $\delta_{\mathrm{steer}} = 0.1\,\mathrm{s}$

D. T-junction

A more complex urban scenario is shown in Fig. 6a: the ego vehicle approaches a T-junction with three other vehicles $b_i, i \leq 3$ (parameters given in Tab. IV). Even if the intended route, driving straight or turning right, of the ego vehicle is not yet known in the behavioral layer, we are able to consider both route options during the computation of our invariably safe sets. Without loss of generality, we assume that the behavioral layer decides the route at $t = 2\,\mathrm{s}$. We compute the under-approximation $\mathcal{S}^t_s$ and $\mathcal{S}^t_r$ for each route at $t = 2\,\mathrm{s}$ and obtain $\mathcal{S}^t = \mathcal{S}^t_s \cap \mathcal{S}^t_r$, visualized in Fig. 6b. The obtained under-approximation ensures safety for both possible route options.

VII. Conclusions

This paper addresses the issue of defining safe states for the domain of self-driving vehicles by introducing invariably safe sets. These are regions which allow vehicles to remain safe for infinite time horizons. In contrast to computationally expensive approaches, a tight under-approximation of the



(a) Scenario at $t = 0\,\mathrm{s}$

(b) Scenario at $t = t_{\mathrm{TTR}} + t_{\mathrm{eva}} = 0.99\,\mathrm{s}$

(c) Scenario at $t = t_{\mathrm{TTR}} + t_{\mathrm{eva}} = 0.99\,\mathrm{s}$

Fig. 5: Emergency situations: (a) vehicle b_1 changes to the ego vehicle's lane but the ego vehicle (diamond) remains within an invariably safe set (gray); (b) a collision with vehicle b_1 can be avoided; (c) if the left lane is occupied by a static obstacle, the ego vehicle can only swerve to the shoulder lane.

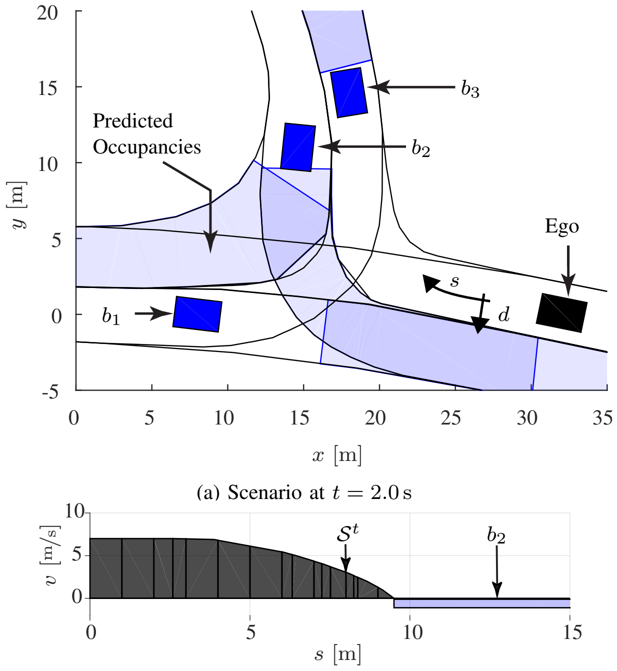
proposed sets can be obtained in real-time. In different examples, we tackle difficult motion planning problems by just using invariably safe sets. The proposed sets guarantee the existence of feasible evasive maneuvers and can be used to compute the last point in time to avoid collisions.

ACKNOWLEDGMENTS

The authors thank Markus Koschi for the development of SPOT and Carmella Schürmann for the voice-over in the video attachment. This work is partially funded by the German Federal Ministry of Economics and Technology through the research initiative Ko-HAF (`https://www.ko-haf.de/`).

TABLE IV: Parameters of T-junction scenario.

Parameter	**Value**
Begin of occupancy $\mathcal{O}_2$	$s_{\min} = 11.0\,\mathrm{m}$
Begin of occupancy $\mathcal{O}_3$	$s_{\min} = 20.0\,\mathrm{m}$
Vehicle lengths	$\ell = 3.0\,\mathrm{m}$
Maximum acceleration	$\lvert a_{s,\max}\rvert = 10.0\,\mathrm{m/s^2}$, $\lvert a_{d,\max}\rvert = 10.0\,\mathrm{m/s^2}$
Reaction times	$\delta_{\mathrm{brake}} = 0.3\,\mathrm{s}$, $\delta_{\mathrm{steer}} = 0.1\,\mathrm{s}$



(a) Scenario at $t = 2.0\,\mathrm{s}$

(b) $\mathcal{S}^t$ at $t = 2.0\,\mathrm{s}$

Fig. 6: T-junction scenario: (a) dynamic obstacles $b_i, i \leq 3$ and their predicted occupancies at $t = 2.0\,\mathrm{s}$ (light blue); (b) the corresponding invariably safe set $\mathcal{S}^t$ which ensures safety for both route options, driving straight and turning right.

REFERENCES

[1] D. Gonzalez, J. Perez, V. Milanes, and F. Nashashibi, "A review of motion planning techniques for automated vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 4, pp. 1135–1145, 2016.

[2] J. M. Anderson, N. Kalra, K. D. Stanley, P. Sorensen, C. Samaras, and O. A. Oluwatola, *Autonomous vehicle technology: a guide for policymakers*. RAND Corporation, 2014.

[3] German Federal Ministry of Transport and Digital Infrastructures, "Ethics comission - Automated and connected driving," 2017. [Online]. Available: https://www.bmvi.de/SharedDocs/DE/Anlage/Presse/084-dobrindt-bericht-der-ethik-kommission.pdf

[4] M. Werling, J. Ziegler, S. Kammel, and S. Thrun, "Optimal trajectory generation for dynamic street scenarios in a Frenet frame," in *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 2010, pp. 987–993.

[5] J. Ziegler and M. Werling, "Navigating car-like robots in unstructured environments using an obstacle sensitive cost function," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2008, pp. 787–791.

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

[7] K. Berntorp, A. Weiss, C. Danielson, and S. Di Cairano, "Automated driving: safe motion planning using positively invariant sets," in *Proc. of the IEEE Int. Conf. on Intelligent Transportation Systems*, 2017, pp. 1–6.

[8] D. Althoff, M. Althoff, and S. Scherer, "Online safety verification of trajectories for unmanned flight with offline computed robust invariant sets," in *Proc. of the IEEE Int. Conf. on Intelligent Robots and Systems*, 2015, pp. 3470–3477.

[9] M. Jalalmaab, B. Fidan, S. Jeon, and P. Falcone, "Guaranteeing persistent feasibility of model predictive motion planning for autonomous vehicles," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2017, pp. 843–848.

[10] T. Fraichard and H. Asama, “Inevitable collision states. A step towards safer robots?” in *Proc. of the IEEE Int. Conf. on Intelligent Robots and Systems*, 2003, pp. 388–393.

[11] L. Martinez-Gomez and T. Fraichard, “An efficient and generic 2D inevitable collision state-checker,” in *Proc. of the IEEE Int. Conf. on Intelligent Robots and Systems*, 2008, pp. 234–241.

[12] D. Althoff, M. Buss, A. Lawitzky, M. Werling, and D. Wollherr, “On-line trajectory generation for safe and optimal vehicle motion planning,” in *Autonomous Mobile Systems*, 2012, pp. 99–107.

[13] M. Althoff and J. M. Dolan, “Online verification of automated road vehicles using reachability analysis,” *IEEE Transactions on Robotics*, vol. 30, no. 4, pp. 903–918, 2014.

[14] P. Falcone, M. Ali, and J. Sjöberg, “Predictive threat assessment via reachability analysis and set invariance theory,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 4, pp. 1352–1361, 2011.

[15] S. L. Herbert, M. Chen, S. Han, S. Bansal, J. F. Fisac, and C. J. Tomlin, “FaSTrack: a modular framework for fast and guaranteed safe motion planning,” in *Proc. of the IEEE Conference on Decision and Control*, 2017, pp. 1517–1522.

[16] A. Lawitzky, A. Nicklas, D. Wollherr, and M. Buss, “Determining states of inevitable collision using reachability analysis,” in *Proc. of the IEEE Int. Conf. on Intelligent Robots and Systems*, 2014, pp. 4142–4147.

[17] S. Söntges and M. Althoff, “Determining the nonexistence of evasive trajectories for collision avoidance systems,” in *Proc. of the IEEE Int. Conf. on Intelligent Robots and Systems*, 2015, pp. 956–961.

[18] W. Wachenfeld, P. Junietz, R. Wenzel, and H. Winner, “The worst-time-to-collision metric for situation identification,” in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2016, pp. 729–734.

[19] A. Berthelot, A. Tamke, T. Dang, and G. Breuel, “A novel approach for the probabilistic computation of time-to-collision,” in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2012, pp. 1173–1178.

[20] 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.

[21] J. Hillenbrand, A. M. Spieker, and K. Kroschel, “A multilevel collision mitigation approach - its situation assessment, decision making, and performance tradeoffs,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 7, no. 4, pp. 528–540, 2006.

[22] Y. Kuwata, S. Karaman, J. Teo, E. Frazzoli, J. How, and G. Fiore, “Real-time motion planning with applications to autonomous urban driving,” *IEEE Transactions on Control Systems Technology*, vol. 17, no. 5, pp. 1105–1118, 2009.

[23] C. Pek, M. Koschi, M. Werling, and M. Althoff, “Enhancing motion safety by identifying safety-critical passageways,” in *Proc. of the IEEE Conference on Control and Decision*, 2017, pp. 320–326.

[24] S. Steyer, G. Tanzmeister, and D. Wollherr, “Grid-based environment estimation using evidential mapping and particle tracking,” *IEEE Transactions on Intelligent Vehicles*, 2018.

[25] M. Koschi and M. Althoff, “SPOT: A tool for set-based prediction of traffic participants,” in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2017, pp. 1686–1693.

[26] I. M. Mitchell, “Comparing forward and backward reachability as tools for safety analysis,” in *Hybrid systems: computation and control*. Springer, 2007, pp. 428–443.

[27] E. M. Clarke, O. Grumberg, and D. Peled, *Model checking*. MIT press, 1999.

[28] L. Luthmann, S. Mennicke, and M. Lochau, “Compositionality, decompositionality and refinement in input/output conformance testing,” in *Proc. of Formal Aspects of Component Software*, 2016, pp. 54–72.

[29] O. Rooks, M. Armbruster, S. Bchli, A. Sulzmann, G. Spiegelberg, and U. Kiencke, “Redundancy management for drive-by-wire computer systems,” in *Proc. of the Int. Conf. on Computer Safety, Reliability, and Security*, 2003, pp. 249–262.

[30] R. Isermann, R. Schwarz, and S. Stolzl, “Fault-tolerant drive-by-wire systems,” *IEEE Control Systems*, vol. 22, no. 5, pp. 64–81, 2002.

[31] M. Maurer, J. C. Gerdes, B. Lenz, and H. Winner, *Autonomous Driving – Technical, legal and social aspects*. Springer, 2016.

[32] Economic Comission for Europe: Inland Transport Committee, “Vienna Convention on Road Traffic,” 1968. [Online]. Available: http://www.unece.org/fileadmin/DAM/trans/conventn/crt1968e.pdf

[33] A. Rizaldi, F. Immler, and M. Althoff, “A formally verified checker of the safe distance traffic rules for autonomous vehicles,” in *NASA Formal Methods Symposium*, 2016, pp. 175–190.

[34] 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.

[35] C. Schmidt, F. Oechsle, and W. Branz, “Research on trajectory planning in emergency situations with multiple objects,” in *Proc. of the IEEE Int. Conf. on Intelligent Transportation Systems*, 2006, pp. 988–992.

[36] S. Magdici and M. Althoff, “Adaptive cruise control with safety guarantees for autonomous vehicles,” in *Proc. of the 20th World Congress of the Int. Federation of Automatic Control*, 2017, pp. 5939–5946.

[37] J. Ziegler and C. Stiller, “Fast collision checking for intelligent vehicle motion planning,” in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2010, pp. 518–522.

[38] E. Velenis and P. Tsiotras, “Optimal velocity profile generation for given acceleration limits; the half-car model case,” in *Proc. of the IEEE Int. Symposium on Industrial Electronics*, 2005, pp. 361–366.

[39] M. Herceg, M. Kvasnica, C. Jones, and M. Morari, “Multi-Parametric Toolbox 3.0,” in *Proc. of the European Control Conference*, Zürich, Switzerland, 2013, pp. 502–510, http://control.ee.ethz.ch/~mpt.

[40] 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.