Provably-safe Cooperative Driving via Invariably Safe Sets

Edmond Irani Liu, Christian Pek, and Matthias Althoff

Abstract— **We address the problem of provably-safe cooperative driving for a group of vehicles that operate in mixed traffic scenarios, where both autonomous and human-driven vehicles are present. Our method is based on *Invariably Safe Sets* (ISSs), which are sets of states that let each of the cooperative vehicles remain safe for an infinite time horizon. The potential conflicts between the ISSs of a group of cooperative vehicles are resolved by examining and negotiating their *Safe Maneuver Corridors*. As a result, each vehicle obtains its negotiated ISS, which is used as target sets for motion planning. We demonstrate the applicability and benefits of our method on various traffic scenarios from the CommonRoad benchmark suite.**

I. Introduction

Cooperative motion planning of autonomous vehicles offers a great potential to maximize road safety and passenger comfort. As cooperative planners generate trajectories for multiple vehicles, they can more easily find feasible and better solutions for the group, while considering the individual goals of each vehicle. Cooperative planning is particularly challenging in mixed traffic since all motions need to be collision-free with respect to the unknown future motion of other non-cooperative traffic participants.

Invariably Safe Sets (ISSs) [1] are sets of states which ensure that autonomous vehicles can remain safe for an infinite time horizon. If the ISS of a vehicle is used as its target set for motion planning, one can always find motions that do not cause collisions for an infinite time horizon. The application of ISSs to cooperative planning is an auspicious method to provide strict safety guarantees; however, there is no existing method for negotiating ISSs between cooperative vehicles.

A. Related Work

We review existing work on cooperative motion planning and safety verification techniques.

a) Cooperative planning: Cooperative planning methods that are most relevant to our work fall into the reservation-based category [2]. Reservation-based methods ensure that each cooperative vehicle reserves free-space in the environment to plan trajectories. For instance, an intersection manager is presented in [3], [4], which represents lanes as a set of disjunct tiles. These tiles are allocated to vehicles by a first-come-first-serve protocol, and no tile is allocated to more than one vehicle at a time. The reservation-based method was extended in [5] for better performance in mixed traffic. Since first-come-first-serve allocation protocols may be inefficient in traffic scenarios with multiple vehicles, the allocation process has also been carried out using auctions [6]–[9], in which vehicles bid for certain tiles. This bidding process lets one to distribute tiles for maximized usage of the road. However, modeling the environment through tiles often emanates to a large overhead, since usually, only a small percentage of the tiles raise conflicts. Some methods identify possible conflicting points or regions, and allocate these to vehicles if any conflict is detected [10]–[13].

Instead of tiles, some works allocate moving space-time corridors with a predefined behavior, e.g., lane following or lane changing; vehicles that receive these corridors must adopt their behaviors accordingly [14], [15]. As a result, vehicles plan different maneuvers depending on possible future conflicts. In [16], [17], the authors devised an efficient and explicit space-time reservation protocol for cooperative maneuver planning, with which the vehicles broadcast requests on demand.

Despite the mentioned promising results, they cannot ensure safety in arbitrary traffic situations. Yet, this requirement is vital to enable cooperative driving with a high degree of autonomy.

b) Formal verification: Logical reasoning is a means to verify whether a given trajectory fulfills certain safety properties, which are often specified using higher-order logic. For instance, the safety of lane change maneuvers of autonomous vehicles is verified in [18], [19]. Moreover, safely following a preceding vehicle is verified in [20]. Although logical reasoning methods can ensure properties such as safety, the specified logical formulas are often complicated and user-defined. Also, they usually have to be adjusted as new traffic situations arise.

Reachability analysis [21] can be used to verify the safety of trajectories in arbitrary traffic situations. A reachable set is a set of states that a traffic participant can reach over time by starting from an initial set of states and considering all possible control inputs. Planned trajectories of the autonomous vehicle are safe if they do not intersect with predicted unsafe states [22]–[24]. Even though the reachability analysis lets one to consider the set of all possible trajectories of other traffic participants [25] for collision detection, no alternative safe trajectory is returned if a trajectory under examination is regarded as unsafe.

Sets of safe states can be used to ensure that trajectories planned within them remain safe beyond the planning horizon. For instance, trajectories are not allowed to contain inevitable collision states [26], [27], which are states in which the vehicles will eventually collide. In contrast to inevitable collision states, a collision-free trajectory exists

All authors are with the Department of Informatics, Technical University of Munich, 85748 Garching, Germany.
edmond.irani@tum.de, christian.pek@tum.de, althoff@tum.de

for states in control invariant sets [28], [29].

B. Contributions

In this work, we propose the first cooperative planning method for autonomous vehicles that incorporates ISSs. In particular, our method

1) provides a mechanism to distribute ISSs to a group of cooperative vehicles;
2) can integrate arbitrary planning algorithms to generate individual trajectories;
3) ensures that all trajectories planned within the ISSs are safe for an infinite time horizon; and
4) is computationally efficient and works in mixed traffic situations.

The rest of this work is organized as follows: In Sec. II, we introduce the necessary preliminaries. Subsequently, we present our solution for cooperative motion planning of autonomous vehicles using ISSs in Sec. III. Following that, we demonstrate the benefits of our method in Sec. IV with exemplary scenarios from the CommonRoad[1] benchmark suite [30]. Finally, we draw conclusions in Sec. V.

II. Preliminaries

In this section, we introduce our system, reachable sets, and invariably safe sets.

A. System Description

We represent the set of considered road lanes as $\mathcal{L} := \{L_1, L_2, \dots, L_M\}$, where each lane L_m possesses its curvilinear coordinate system that takes the centerline as the reference path. Note that we perform all computations in this work in these coordinate systems. The adoption of such a coordinate system expedites the formulation of maneuvers from the ego view of the vehicles, e.g., lane-following, stopping at an intersection, and preventing driving backward. Let V_n be the n-th vehicle in a cooperative group, $\mathcal{V} := \{V_1, V_2, \dots, V_N\}$, where $n \in \mathcal{N} := \{1, 2, \dots, N\}$. We use $\square_n$ to denote a variable of V_n, and $\square_m$ a computation performed in L_m. Also, we introduce the notations $\underline{\square}$ and $\overline{\square}$ to specify the minimal and the maximal possible value of a variable. Furthermore, the notation $[\square_n]_1^N = [\square_1, \dots, \square_N]$ compactly represents a list of variables $\square_n$ of vehicles in $\mathcal{V}$. The system dynamics of V_n is

$$\dot{x}_n(t) = f_n(x_n(t), u_n(t)), \tag{1}$$

where $x_n(t) \in \mathcal{X}_n$ is a state in configuration space, $u_n(t) \in \mathcal{U}_n$ is an input in input space, and t is the time. Given an initial state $x_n(t_0)$ at the initial time t_0, and an input trajectory $u_n([t_0, t])$, the solution to (1) at time t is denoted as $\chi_n(t; x_n(t_0), u_n([t_0, t]))$. The state of vehicle V_n, in lane L_m, is modeled as $x_n(t) = (s_n, d_n, v_n) \in \mathbb{R}^3$, where s_n, d_n, and v_n are the longitudinal position, lateral position, and the velocity of V_n, respectively. Finally, the occupancies of a set of static and dynamic obstacles $\mathcal{E}(t)$ at time t are represented by $\mathcal{O}_{\mathcal{E}}(t) \subset \mathbb{R}^2$.

B. Reachable Sets

Definition 1 (Reachable Set [31]): *The reachable set $\mathcal{R}$ of a system is defined as the set of all states that can be reached at a given time t, starting from an initial set of states $\mathcal{X}_n^0 \subseteq \mathcal{X}_n$.*

During the propagation of the reachable set of vehicle V_n, we remove the set of states whose occupancy is overlapping with $\mathcal{O}_{\mathcal{E}}(t)$. Let $\mathcal{X}_n^{\mathtt{CF}}(t) = \mathcal{X}_n \setminus \mathcal{O}_{\mathcal{E}}(t)$ be the maximal set of states which are collision-free at time t, the collision-free reachable set at time t starting from $\mathcal{X}_n^0$ is

$$\begin{aligned}\mathcal{R}_n(\mathcal{X}_n^0, t) := \Big\{ \chi_n(t; x_n(t_0), u_n([t_0, t])) \Big| x_n(t_0) \in \mathcal{X}_n^0,\\ \forall \tau \in [t_0, t] : u_n(\tau) \in \mathcal{U}_n,\\ \chi_n(\tau; x_n(t_0), u_n([t_0, \tau])) \in \mathcal{X}_n^{\mathtt{CF}}(\tau) \Big\}.\end{aligned}$$

Definition 2 (Drivable Area): *Given that the operator $\operatorname{proj}(\cdot)$ returns the positions (s, d) of a given state $x_n(t)$, the drivable area $\mathcal{D}_n(t)$ of vehicle V_n at time t is defined as the projection of $\mathcal{R}_n(\mathcal{X}_n^0, t)$, thus:*

$$\begin{aligned}\mathcal{D}_n(t) :=& \operatorname{proj}(\mathcal{R}_n(\mathcal{X}_n^0, t))\\ =& \Big\{ \operatorname{proj}(x_n(t)) \Big| x_n(t) \in \mathcal{R}_n(\mathcal{X}_n^0, t) \Big\}.\end{aligned}$$

The occupancies $\mathcal{O}_E(t)$ of obstacles $E \in \mathcal{E}(t)$ required for the computation of reachable sets and ISSs are predicted with the help of SPOT [25], which is a set-based occupancy prediction tool that captures all possible future movements of other traffic participants.

C. Invariably Safe Sets

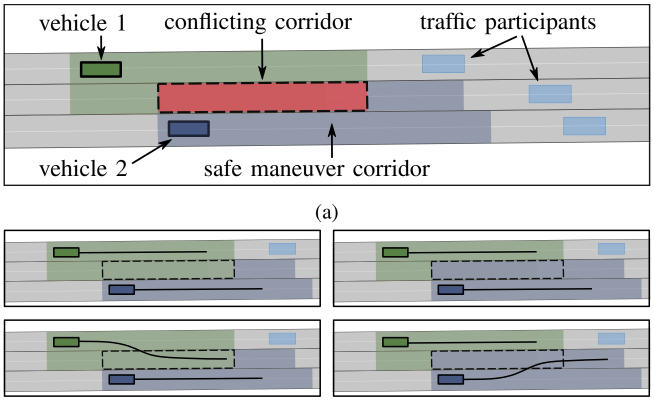
Just being in the collision-free states $\mathcal{X}_n^{\mathtt{CF}}(t)$ does not ensure the safety of the vehicle. Consider the case that a vehicle is driving with a high velocity towards a static obstacle in a near distance. Although it is collision-free at the current time t, it may eventually collide with the obstacle at time $t + t_\epsilon$, $t_\epsilon > 0$. To formally ensure its safety, we, therefore, require the vehicle to be in an invariably safe state [1]. An invariably safe state is defined recursively: a state is deemed invariably safe if a collision-free trajectory ending at another invariably safe state exists.

Definition 3 (Invariably Safe Sets): *The invariably safe set $\mathcal{S}_n(t)$ of V_n at time t is a set of invariably safe states that allows V_n to be safe for an infinite time horizon, and is defined as:*

$$\begin{aligned}\mathcal{S}_n(t) := \Big\{ x_n(t) \in \mathcal{X}_n^{\mathtt{CF}}(t) \Big| \forall \tau > t, \exists u_n([t, \tau]) :\\ \chi_n(\tau; x_n(t), u_n([t, \tau])) \in \mathcal{X}_n^{\mathtt{CF}}(\tau) \Big\}.\end{aligned}$$

In most cases, determining the maximal ISS of V_n is a computationally intensive task. Nonetheless, its tight under-approximation can be efficiently derived using safe braking distances [32] and safe evasive distances [33]. These distances ensure the safety of the ego vehicle even if the preceding obstacles and the obstacles in the adjacent lanes brake or accelerate with their maximal capability.

[1] https://commonroad.in.tum.de/



(a)

(b)

Fig. 1: Overview of our method. (a) Safe maneuver corridors are derived based on individual ISS, and the conflicts between them are examined. (b) Possible allocation plans to resolve the conflict, and exemplary trajectories of vehicles from the set of their permissible trajectories.

Proposition 1 (Under-Approximation of $\mathcal{S}_n(t)$): The union of the set $\mathcal{S}_n^{\mathtt{b}}(t)$ of states respecting safe braking distances [32] and the set $\mathcal{S}_n^{\mathtt{e}}(t)$ of states respecting safe evasive distances [33] at time t is a tight under-approximation of $\mathcal{S}_n(t)$, i.e., $\mathcal{S}_n^{\mathtt{b}}(t) \cup \mathcal{S}_n^{\mathtt{e}}(t) \subset \mathcal{S}_n(t)$. □

Proof: see [1, Sec. III-C]. ■

III. Methodology

We strive to allocate ISSs to a group of cooperative vehicles when conflicts between their individual ISS are detected. It is ensured that safe motions exist when a vehicle is within its ISS [1]. Specifically, the ISSs can be used as target sets for motion planning in two different ways:

1) Solely used in the last time step of the planning cycle to lift safety of the group to an infinite time horizon;
2) Used in every time step to guide motion planners into states for which a vehicle can continue safe motions.

Fig. 1 shows an overview of our method, which can be divided into three major steps:

1) Computation of ISSs for all cooperative vehicles in a group: the computation of the ISS of a vehicle is in turn based on its reachable set and the set-based occupancy prediction of non-cooperative traffic participants [25].
2) Derivation of safe maneuver corridors: to detect the potential conflicts between the ISSs of the vehicles, we derive their safe braking corridors and safe evasive maneuver corridors (hereafter collectively abbreviated as *corridors*) based on their individual ISS. Here, conflicting corridors indicate conflicts of ISSs.
3) Negotiation of conflicting corridors: a negotiation mechanism is established to resolve the detected conflicts of corridors, such that in the end, every vehicle receives a negotiated ISS.

We demand that for each replanning cycle with a replanning duration of t_r, all computations are carried out up to a future time horizon t_h, where $t_r < t_h$. While reaching t_r, the environmental information is updated, and a new round of negotiation within the group is carried out. Violation of such time constraints does not result in unsafe behavior of the vehicles, as their individual fail-safe trajectories [34] are also computed. We perform all computations at discrete time steps $k \geq 0$, which correspond to points in time $t_k = k\Delta t$, where $\Delta t \geq 0$ is a predefined time duration. Subsequently, we assume that all computations are performed for the last time step k of a planning cycle. Alg. 1 describes the procedure of the generation of negotiated ISSs for vehicles. We give a detailed explanation of the algorithm below.

Algorithm 1 Generation of Negotiated ISSs

```
Inputs: $[\mathcal{R}_n(k-1)]_1^N$, $\mathcal{E}_n(k-1)$, $\mathcal{L}$
Output: $[\mathcal{S}_n^{\mathtt{N}}]_1^N$
1: $\mathcal{O}_{\mathcal{E}} \leftarrow$ OccupancyPrediction($\mathcal{E}_n(k-1)$)
2: $[\mathcal{R}_n]_1^N \leftarrow$ ReachableSet($[\mathcal{R}_n(k-1)]_1^N$, $\mathcal{O}_{\mathcal{E}}$)
3: $[\mathcal{S}_n]_1^N \leftarrow$ ISS($[\mathcal{R}_n]_1^N$, $\mathcal{L}$, $\mathcal{O}_{\mathcal{E}}$)
4: $[\mathcal{C}_n^{\mathtt{U}}]_1^N \leftarrow$ SafeManeuverCorridor($[\mathcal{S}_n]_1^N$)
5: $\mathcal{C}^{\mathtt{C}} \leftarrow$ ConflictingCorridor($[\mathcal{C}_n^{\mathtt{U}}]_1^N$)
6: $P^* \leftarrow$ OptimalAllocation($[\mathcal{S}_n]_1^N$, $\mathcal{C}^{\mathtt{C}}$)
7: $[\mathcal{S}_n^{\mathtt{N}}]_1^N \leftarrow$ NegotiatedISS($[\mathcal{S}_n]_1^N$, $P^*$)
8: return $[\mathcal{S}_n^{\mathtt{N}}]_1^N$
```

Algorithm 2 Computation of ISS

```
function ISS($[\mathcal{R}_n]_1^N$, $\mathcal{L}$, $\mathcal{O}_{\mathcal{E}}$)
    for $n \leftarrow 1$ to $N$ do
        $\mathcal{D}_n \leftarrow$ Projection($\mathcal{R}_n$)
        $\mathcal{L}_n^{\mathtt{R}} \leftarrow$ ReachableLanes($\mathcal{D}_n$, $\mathcal{L}$)
        $\mathcal{S}_n^{\mathtt{b}} \leftarrow$ SafeBrakingSet($\mathcal{D}_n$, $\mathcal{L}_n^{\mathtt{R}}$, $\mathcal{O}_{\mathcal{E}}$)
        $\mathcal{S}_n^{\mathtt{e}} \leftarrow$ SafeEvasiveSet($\mathcal{D}_n$, $\mathcal{L}_n^{\mathtt{R}}$, $\mathcal{O}_{\mathcal{E}}$)
        $\mathcal{S}_n \leftarrow \mathcal{S}_n^{\mathtt{b}} \cup \mathcal{S}_n^{\mathtt{e}}$
    end for
    return $[\mathcal{S}_n]_1^N$
end function
```

A. Computation of Reachable Sets (Alg. 1, line 2)

The dynamics of vehicle V_n are modeled as two double integrators with limited velocities and accelerations in the s_n and d_n directions:

$$\ddot{s}_n(t) = a_{n,s}(t), \quad \ddot{d}_n(t) = a_{n,d}(t), \tag{2a}$$

$$\underline{v}_{n,s} \leq v_{n,s}(t) \leq \overline{v}_{n,s}, \; \underline{v}_{n,d} \leq v_{n,d}(t) \leq \overline{v}_{n,d}, \tag{2b}$$

$$|a_{n,s}(t)| \leq \overline{a}_{n,s}, \quad |a_{n,d}(t)| \leq \overline{a}_{n,d}. \tag{2c}$$

Since model (2) captures the real vehicle behavior, we can use this simple model to efficiently prove safety. The reachable set of V_n, denoted by $\mathcal{R}_n$, is computed as in [31].

B. Computation of ISSs (Alg. 1, line 3)

The computation of individual ISSs of cooperative vehicles are based on their own reachable sets, which is explained as follows (see Alg. 2).

1) Projection: The drivable area of vehicle V_n is $\mathcal{D}_n := \mathrm{proj}(\mathcal{R}_n)$ (cf. Def. 2).

2) Reachable Lanes: $\mathcal{D}_n$ only overlaps with a subset of $\mathcal{L}$, which we refer to as the reachable lanes of V_n, given by:

$$\mathcal{L}_n^{\mathtt{R}} := \big\{L \in \mathcal{L} \big| \mathcal{D}_n \cap L \neq \emptyset\big\}.$$

3) ISSs: The ISS $\mathcal{S}_n$ is computed for every vehicle V_n considering $\mathcal{L}^{\mathsf{R}}_n$. Each of the lanes $L^{\mathsf{R}}_{n,m} \in \mathcal{L}^{\mathsf{R}}_n$ can be split into smaller segments $\mathcal{G}^{i,j}_{n,m}$ which are delimited by different pairs of consecutive obstacles $E_i, E_j \in \mathcal{E}$ located in the lane. We shrink these segments in the driving direction by the half-length of V_n, denoted by $l_n/2$, to accommodate for the occupancy of V_n (see Fig. 2). For simplicity, we omit the superscripts i and j in the following computations, and assume that the occupancies of the obstacles are all within $L^{\mathsf{R}}_{n,m}$. We obtain the drivable area $\mathcal{D}_{n,m}$ that overlaps with the segment $\mathcal{G}_{n,m}$, and its minimal and maximal values along the longitudinal and lateral directions as shown in Fig. 2:

$$\begin{aligned}\mathcal{D}_{n,m} :=& \mathcal{D}_n \cap \mathcal{G}_{n,m},\\ [\underline{s}_{n,m}, \overline{s}_{n,m}] =& \operatorname{lon}(\mathcal{D}_{n,m}),\\ [\underline{d}_{n,m}, \overline{d}_{n,m}] =& \operatorname{lat}(\mathcal{D}_{n,m}).\end{aligned}$$

where the operators $\operatorname{lon}(\cdot)$ and $\operatorname{lat}(\cdot)$ return the longitudinal and lateral positions $[\underline{s}, \overline{s}]$ and $[\underline{d}, \overline{d}]$ of the given element, respectively.

In the following, we use $q \in \{\mathsf{b}, \mathsf{e}\}$ as the superscript for the sets related to braking and evasive maneuvers. Let $\Delta^{\mathsf{b}}_s(v, E_j)$ be the safe braking distance [32], and $\Delta^{\mathsf{e}}_s(v, d, E_j)$ the safe evasive distance [33], both for a preceding obstacle E_j. We also require the safe braking and evasive distances for a following obstacle E_i, which we denote by $\Delta^{\mathsf{b}}_s(v, E_i)$ and $\Delta^{\mathsf{e}}_s(v, d, E_i)$, respectively. The sets $\mathcal{S}^{\mathsf{b}}_{n,m}$ and $\mathcal{S}^{\mathsf{e}}_{n,m}$ (recall Prop. 1) of vehicle V_n in segment $\mathcal{G}_{n,m}$ are obtained as presented in [1, Alg. 1]:

$$\begin{aligned}\mathcal{S}^{\mathsf{b}}_{n,m} := \Big\{ (s,d,v) \Big| & s \in [\underline{s}_{n,m}, \overline{s}_{n,m}], \qquad (3)\\ & d \in [\underline{d}_{n,m}, \overline{d}_{n,m}], v \in [\underline{v}_{n,s}, \overline{v}_{n,s}],\\ & \forall (s_i, d_i) \in \mathcal{O}_{E_i} : s \geq s_i + \Delta^{\mathsf{b}}_s(v, E_i),\\ & \forall (s_j, d_j) \in \mathcal{O}_{E_j} : s \leq s_j - \Delta^{\mathsf{b}}_s(v, E_j) \Big\},\end{aligned}$$

$$\begin{aligned}\mathcal{S}^{\mathsf{e}}_{n,m} := \Big\{ (s,d,v) \Big| & s \in [\underline{s}_{n,m}, \overline{s}_{n,m}], \qquad (4)\\ & d \in [\underline{d}_{n,m}, \overline{d}_{n,m}], v \in [\underline{v}_{n,s}, \overline{v}_{n,s}],\\ & \forall (s_i, d_i) \in \mathcal{O}_{E_i} : s \geq s_i + \Delta^{\mathsf{e}}_s(v, d, E_i),\\ & \forall (s_j, d_j) \in \mathcal{O}_{E_j} : s \leq s_j - \Delta^{\mathsf{e}}_s(v, d, E_j),\\ & \forall \tau \in [0, \Delta^{\mathsf{e}}_t], \exists d' : (s + v\tau, d', v) \in \mathcal{S}^{\mathsf{b}}_{n,m'}(\lceil \tfrac{\tau}{\Delta t} \rceil) \Big\},\end{aligned}$$

where m' denotes the index of an adjacent lane of $L^{\mathsf{R}}_{n,m}$ into which the vehicle evades while performing the evasive maneuver, and Δ^{e}_t the evasive time required to fully enter the adjacent lane $L_{m'}$. The velocity constraints $[\underline{v}_{n,s}, \overline{v}_{n,s}]$ are extracted from the reachable set $\mathcal{R}_n$. Refer to [1, Alg. 1] for the derivation of these sets. The ISS of V_n on lanes $\mathcal{L}^{\mathsf{R}}_n$ is defined as:

$$\mathcal{S}_n := \bigcup_{m,q} \mathcal{S}^q_{n,m}. \qquad (5)$$

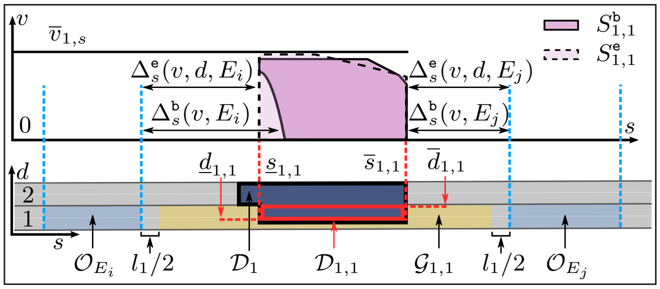


Fig. 2: Relation of various components for ISS computation. The result is shown for a vehicle with $n = 1$, in lane $m = 1$. Lane IDs are labeled at the beginning of the lanes.

C. Computation of Safe Maneuver Corridors (Alg. 1, line 4)

We incorporate the set-based occupancy prediction with respect to non-cooperative traffic participants. The potential future collisions between the cooperative vehicles can be detected by examining the future space and time required by the cooperative vehicles to perform safe braking or safe evasive maneuvers, starting from their current ISSs, which we refer to as their *Safe Maneuver Corridors*.

Definition 4 (Safe Maneuver Corridors): *The safe maneuver corridors of vehicle V_n are segments of lanes that V_n requires to perform either a safe braking or a safe evasive maneuver, starting from a state in its ISS.*

A corridor derived from an invariably safe state $(s, d, v) \in \mathcal{S}^q_{n,m}$ in lane L_m is represented by a three-element tuple $(\underline{s}, \overline{s}, k_h)$, which indicates the required position interval $[\underline{s}, \overline{s}]$ in a lane and the time horizon k_h.

For the braking maneuvers, $\Delta^{\mathsf{b}}_t(v)$ and $\Delta^{\mathsf{b}}_s(v)$ are the time required for the vehicle to fully brake, and the distance traveled while braking, respectively; for the evasive maneuvers, we analogously use $\Delta^{\mathsf{e}}_t(d)$ and $\Delta^{\mathsf{e}}_s(v)$. These are given by:

$$\Delta^{\mathsf{b}}_t(v) = \delta^{\mathsf{b}} + \tfrac{v}{\overline{a}_{n,s}}, \quad \Delta^{\mathsf{b}}_s(v) = \delta^{\mathsf{b}} v + \tfrac{v^2}{2\overline{a}_{n,s}}, \qquad (6a)$$

$$\Delta^{\mathsf{e}}_t(d) = \delta^{\mathsf{e}} + \sqrt{\tfrac{2d^{\mathsf{e}}_{m'}(d)}{\overline{a}_{n,d}}}, \quad \Delta^{\mathsf{e}}_s(v) = \Delta^{\mathsf{e}}_t v, \qquad (6b)$$

where δ^{b} and δ^{e} denote the reaction times for braking and evading, respectively, and $d^{\mathsf{e}}_{m'}(d)$ the distance to fully enter an adjacent lane $L_{m'}$ of the lane L_m. For brevity, we omit the road curvature; however, one can easily integrate it as in [1, Alg. 1]. We denote the braking and the evasive corridors of V_n for its ISS $\mathcal{S}^q_{n,m}$ in lane L_m by $\mathcal{C}^{\mathsf{b}}_{n,m}$ and $\mathcal{C}^{\mathsf{e}}_{n,m}$, respectively, given by:

$$\begin{aligned}\mathcal{C}^{\mathsf{b}}_{n,m} :=& \Big\{ (\underline{s}, \overline{s}, k_h) \Big| (s,d,v) \in \mathcal{S}^{\mathsf{b}}_{n,m}, \qquad (7)\\ \underline{s} =& s - l_n/2, \overline{s} = s + \Delta^{\mathsf{b}}_s(v) + l_n/2, k_h = \Big\lceil \tfrac{\Delta^{\mathsf{b}}_t(v)}{\Delta t} \Big\rceil \Big\},\\ \mathcal{C}^{\mathsf{e}}_{n,m} :=& \Big\{ (\underline{s}, \overline{s}, k_h) \Big| (s,d,v) \in \mathcal{S}^{\mathsf{e}}_{n,m} \cup \mathcal{S}^{\mathsf{e}}_{n,m'}, \qquad (8)\\ \underline{s} =& s - l_n/2, \overline{s} = s + \Delta^{\mathsf{e}}_s(v) + l_n/2, k_h = \Big\lceil \tfrac{\Delta^{\mathsf{e}}_t(d)}{\Delta t} \Big\rceil \Big\}.\end{aligned}$$

To simplify the identification of the conflicting corridors in the next step, we formulate tuples $\mathcal{C}^{q,\mathsf{B}}_{n,m}$, which has the extremums over each component of $(\underline{s}, \overline{s}, k_h)$ in $\mathcal{C}^q_{n,m}$ as its

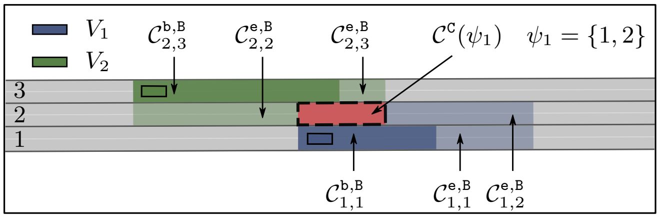

V_1
V_2
$\mathcal{C}^{\mathtt{b},\mathtt{B}}_{2,3}$
$\mathcal{C}^{\mathtt{e},\mathtt{B}}_{2,2}$
$\mathcal{C}^{\mathtt{e},\mathtt{B}}_{2,3}$
$\mathcal{C}^{\mathtt{C}}(\psi_1)$
$\psi_1 = \{1, 2\}$
3
2
1
$\mathcal{C}^{\mathtt{b},\mathtt{B}}_{1,1}$
$\mathcal{C}^{\mathtt{e},\mathtt{B}}_{1,1}$
$\mathcal{C}^{\mathtt{e},\mathtt{B}}_{1,2}$

Fig. 3: Illustration of the (conflicting) safe maneuver corridors. The two vehicles have non-conflicting ISSs in lanes 1 and 3, respectively; however, their corridors intersect in lane 2.

elements (see Fig. 3):

$$\begin{aligned}\mathcal{C}^{q,\mathtt{B}}_{n,m} &:= (\underline{s}', \overline{s}', k'_h), \qquad (9)\\ \underline{s}' &= \min\{\underline{s} | (\underline{s}, \overline{s}, k_h) \in \mathcal{C}^q_{n,m}\},\\ \underline{s}' &= \max\{\overline{s} | (\underline{s}, \overline{s}, k_h) \in \mathcal{C}^q_{n,m}\},\\ k'_h &= \max\{k_h | (\underline{s}, \overline{s}, k_h) \in \mathcal{C}^q_{n,m}\}.\end{aligned}$$

The union of the maneuver corridors of V_n in its reachable lanes $\mathcal{L}^{\mathtt{R}}_n$ over the braking and the evasive maneuvers are:

$$\mathcal{C}^{\mathtt{U}}_n := \bigcup_{m,q} \{\mathcal{C}^{q,\mathtt{B}}_{n,m}\}. \qquad (10)$$

D. Identification of Conflicting Corridors (Alg. 1, line 5)

Since conflicting corridors occur between at least two cooperative vehicles, we denote the set of all conflicting subsets of vehicles $\mathcal{V}$ by $\mathcal{I}_{\geq 2}(\mathcal{N})$, which is the power set of $\mathcal{N}$ with a minimum cardinality of two. Each element of $\mathcal{I}_{\geq 2}(\mathcal{N})$ is referred to as a *coalition*, and we denote the i-th coalition in $\mathcal{I}_{\geq 2}(\mathcal{N})$ as ψ_i. The conflicting corridors $\mathcal{C}^{\mathtt{C}}(\psi_i)$ of vehicles within a coalition ψ_i (see Fig. 3), and the tuple of all conflicting corridors $\mathcal{C}^{\mathtt{C}}$ within all coalitions are obtained from

$$\begin{aligned}\mathcal{C}^{\mathtt{C}}(\psi_i) :=& \big\{(\underline{s}', \overline{s}', k'_h) | l, l' \in \psi_i, l \neq l', \qquad (11)\\ & (\underline{s}_1, \overline{s}_1, k_{h,1}) \in \mathcal{C}^{\mathtt{U}}_l, (\underline{s}_2, \overline{s}_2, k_{h,2}) \in \mathcal{C}^{\mathtt{U}}_{l'},\\ & [\underline{s}', \overline{s}'] = [\underline{s}_1, \overline{s}_1] \cap [\underline{s}_2, \overline{s}_2], k'_h = \min(k_{h,1}, k_{h,2})\big\},\\ \mathcal{C}^{\mathtt{C}} :=& (\mathbf{c}_1, \ldots, \mathbf{c}_j),\ \mathbf{c}_1, \ldots, \mathbf{c}_j \in \mathcal{C}^{\mathtt{C}}(\psi_i), \psi_i \in \mathcal{I}_{\geq 2}(\mathcal{N}). \qquad (12)\end{aligned}$$

E. Negotiation of Conflicting Corridors (Alg. 1, lines 6–7)

The process of finding the optimal allocation plan P^* for the conflicting maneuver corridors is presented in Alg. 3. We explain the main functions as follows:

1) Generation of All Possible Allocation Plans (Alg. 3, line 4): We indicate the j-th element in $\mathcal{C}^{\mathtt{C}}$, which has a conflict within coalition ψ_i, as $\mathbf{c}_j(\psi_i^{(j)})$. A possible allocation of the conflicting corridors is referred to as an allocation plan P, which is a tuple $(P^{(1)}, \ldots, P^{(j)})$, with $P^{(j)}$ representing the index of the winning vehicle of corridor $\mathbf{c}_j(\psi_i^{(j)})$. The set $\mathcal{P}$ of all possible allocation plans P of the conflicting corridors is composed of the Cartesian product of their coalitions $\psi_i^{(j)}$. As an example, for $\mathcal{C}^{\mathtt{C}} = \big\{\mathbf{c}_1(\psi_1^{(1)}), \mathbf{c}_2(\psi_3^{(2)})\big\}$, where $\psi_1 = \{1, 2\}$ and $\psi_3 = \{1, 3\}$, $\mathcal{P} = \psi_1 \times \psi_3 = \{(1, 1), (1, 3), (2, 1), (2, 3)\}$ (see Fig. 4).

Algorithm 3 Finding Optimal Allocation Plan P^*

```
1:  function OptimalAllocation($[\mathcal{S}_n]_1^N$, $\mathcal{C}^{\mathtt{C}}$)
2:      $P^* \leftarrow \emptyset$,  $J^* \leftarrow \infty$ ▷ initialization of optimal plan and cost
3:      if $\mathcal{C}^{\mathtt{C}} \neq \emptyset$ then
4:          $\mathcal{P} \leftarrow$ AllPossiblePlans($\mathcal{C}^{\mathtt{C}}$)
5:          for $P \in \mathcal{P}$ do
6:              $\mathcal{Z}(P) \leftarrow \emptyset$
7:              for $n \in \mathcal{N}$ do
8:                  $S^{\mathtt{P}}_n(P) \leftarrow$ ISSwithPlan($\mathcal{S}_n$, $P$)
9:                  $\mathcal{Z}(P) \leftarrow \mathcal{Z}(P) \bigcup \mathrm{vol}(S^{\mathtt{P}}_n(P))$
10:             end for
11:             $J(P) \leftarrow$ Cost($\mathcal{Z}(P)$, $[S^{\mathtt{P}}_n(P)]_1^N$)            ▷ see (15)
12:             if $J(P) < J^*$ then
13:                 $P^* \leftarrow P$,  $J^* \leftarrow J(P)$
14:             end if
15:         end for
16:     end if
17:     return $P^*$
18: end function
```

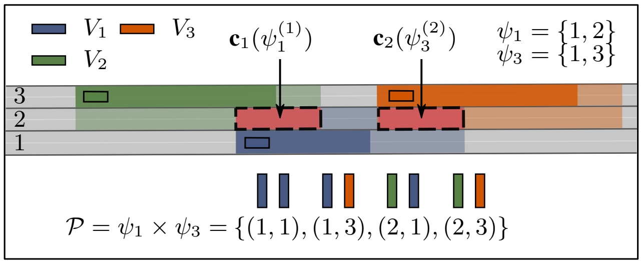

V_1
V_3
V_2
$\mathbf{c}_1(\psi_1^{(1)})$
$\mathbf{c}_2(\psi_3^{(2)})$
$\psi_1 = \{1, 2\}$
$\psi_3 = \{1, 3\}$
3
2
1
$\mathcal{P} = \psi_1 \times \psi_3 = \{(1, 1), (1, 3), (2, 1), (2, 3)\}$

Fig. 4: Illustration of all possible allocation plans.

2) ISSs with respect to plan P (Alg. 3, line 8): We construct a set of corridors that are not allocated to vehicle V_n according to plan P by:

$$\begin{aligned}\mathcal{C}^{\mathtt{L}}_n(P) := \big\{(\underline{s}, \overline{s}, k_h) | \psi_i^{(j)} \cap n \neq \emptyset, \qquad (13)\\ n \neq P^{(j)}, (\underline{s}, \overline{s}, k_h) \in \mathcal{C}^{\mathtt{C}}(\psi_i^{(j)})\big\}.\end{aligned}$$

Some of the states in the previously-computed ISS of V_n should be removed as their maneuver corridors intersect with $\mathcal{C}^{\mathtt{L}}_n(P)$. The maximal subset of the ISS $\mathcal{S}^{\mathtt{P}}_n(P) \subseteq \mathcal{S}_n$ that is still safe with respect to plan P is obtained through:

$$\mathrm{cor}(\mathcal{S}^{\mathtt{P}}_n(P)) = \mathrm{cor}(\mathcal{S}_n) \setminus \mathcal{C}^{\mathtt{L}}_n(P),$$

with the operator $\mathrm{cor}(\cdot)$ returning the set of corridors $\mathcal{C}^{\mathtt{U}}_n$ (cf. (7)–(10)) of a given input ISS.

3) Cost Function (Alg. 3, lines 6–14): We aim for a fair allocation of the conflicting corridors. To this end, we compute the variance of the volumes of $\mathcal{S}^{\mathtt{P}}_n(P)$, which are returned by the operator $\mathrm{vol}(\cdot)$, of vehicles $\mathcal{V}$ for each plan. Let $\mathcal{Z}(P) := \{Z_1, Z_2, \ldots, Z_N\}$ be the set of volumes Z_n of $\mathcal{S}^{\mathtt{P}}_n(P)$, the variance of $\mathcal{Z}(P)$, which is returned by $\mathrm{var}(\cdot)$, is computed by:

$$\tilde{Z} = \frac{\sum_1^N Z_n}{N}, \ \mathrm{var}(\mathcal{Z}(P)) := \frac{\sum_1^N (Z_n - \tilde{Z})^2}{N}. \qquad (14)$$

In the best case $\mathrm{var}(\mathcal{Z}(P)) = 0$, implying that all of the cooperative vehicles receive equally large ISS.

In addition, we reward allocation plans that promote higher overall traffic efficiency, which is reflected by summing the length of the longest maneuver corridors in the longitudinal direction of all cooperative vehicles. The proposed cost function for a plan P has the form of

$$J(P) = \mathrm{var}(\mathcal{Z}(P)) - \\ w \sum_{1}^{N} \max\left\{ \mathrm{len}(\mathbf{c}) \middle| \mathbf{c} \in \mathrm{cor}(\mathcal{S}^{\mathtt{P}}_{n}(P)) \right\}, \qquad (15)$$

where w is an user-tunable parameter, and the $\mathrm{len}(\cdot)$ operator returns the length $|\overline{s} - \underline{s}|$ of the given corridor. Alternatively, one can also include other terms into the cost function. The plan with the minimal cost is regarded as the optimal allocation plan P^{*}, and the ISSs of vehicles with respect to P^{*} is referred to as their *Negotiated ISSs* $\mathcal{S}^{\mathtt{N}}_{n} := \mathcal{S}^{\mathtt{P}}_{n}(P^{*})$.

F. Incorporating Negotiated ISSs into Every Time Step

To incorporate the negotiated ISSs into every time step of the current planning cycle (cf. Sec. III), for a vehicle V_n that has a non-empty $\mathcal{C}^{\mathtt{L}}_{n}(P^{*}, k)$ at time step k, we stipulate that it respects the negotiation result by not entering the corridors $\mathbf{c} \in \mathcal{C}^{\mathtt{L}}_{n}(P^{*}, k)$, up to its persisting time step $k + k_h$. This is achieved by removing the occupancies of $\mathbf{c}$ in the computation of the reachable set of V_n in subsequent time steps (cf. Def. 1). Additionally, we require that V_n respects safe distances with regards to $\mathbf{c}$, with their velocity profiles extracted from the winning vehicles at longitudinal positions $\mathrm{lon}(\mathbf{c})$, to ensure safety within the planning cycle. To this end, we formulate a set of phantom obstacles $\mathcal{E}^{\mathtt{F}}_{n}(k)$, which has $\mathcal{C}^{\mathtt{L}}_{n}(P^{*}, k)$ and the persisting corridors $\mathbf{c}$ from previous time steps as its elements:

$$\mathcal{E}^{\mathtt{F}}_{n}(k) := \mathcal{C}^{\mathtt{L}}_{n}(P^{*}, k) \cup \left\{ \mathbf{c} \in \mathcal{E}^{\mathtt{F}}_{n}(k-1) \middle| k \leq \mathrm{per}(\mathbf{c}) \right\},$$

with the operator $\mathrm{per}(\cdot)$ returning the persisting time step of the given corridor.

G. Computational Complexity

Assuming that the set-based occupancy prediction of obstacles and the reachable sets of cooperative vehicles are readily available (see [25], [31] for their respective complexity analysis), the computational complexity of the proposed method for each time step k can be split into three parts:

1) Computation of ISSs: for a vehicle V_n, the complexity of computing its ISS $\mathcal{S}_n$ is $O(N)$, where N denotes the number of considered obstacles in lanes $\mathcal{L}^{\mathtt{R}}_{n}$.
2) Derivation of Safe Maneuver Corridors: the complexity of computing the safe maneuver corridors of V_n is again $O(N)$.
3) Negotiation of Conflicting Corridors: for a group of N vehicles that collectively access M reachable lanes, in the worst case, they could have conflicts of corridors on all reachable lanes, thus raising a complexity of $O(N^{M})$. Nevertheless, the computational tractability can still be ensured, since
    a) the upper bounds of both N and M can be controlled by forming smaller cooperative groups, e.g. using methods as in [35]; and
    b) the proposed method can be applied in *anytime* fashion, i.e., the negotiated ISSs with respect to the best plan evaluated up to the time of request is returned.

Alternatively, one can adopt auction-based methods, as in [6], to obtain a sub-optimal allocation of conflicting corridors with polynomial complexity with regard to M.

TABLE I: PARAMETERS FOR NUMERICAL EXPERIMENTS.

Parameters		**Scenario Identifiers**		
symbol	**unit**	I	II	III
Δt	s	0.1	0.1	0.1
t_h	s	2.5	2.0	2.0
$\overline{v}_{n,s}$	m/s	30.0	30.0	30.0
$\underline{v}_{n,s}$	m/s	0.0	0.0	0.0
$\overline{v}_{n,d}$	m/s	3.0	3.0	4.0
$\underline{v}_{n,d}$	m/s	-3.0	-3.0	-4.0
$\overline{a}_{n,s}$	m/s^2	8.0	8.0	8.0
$\overline{a}_{n,d}$	m/s^2	3.0	3.0	4.0
δ^{b}	s	0.3	0.3	0.3
δ^{e}	s	0.3	0.3	0.3

IV. Evaluation

We demonstrate the applicability of our method on three distinct scenarios. A list of selected parameters for all the cooperative vehicles can be found in Tab. I. We use an optimization-based motion planner [34] to generate trajectories within $\mathcal{S}^{\mathtt{N}}_{n}(k)$.

A. Scenario I: Highway

Our first scenario illustrates a highway with two communicating vehicles V_1 and V_2 in the presence of two leading other traffic participants (see Fig. 5). V_1 and V_2 face a conflict of maneuver corridors at the very first time step $k = 1$. Our algorithm allocates the conflicting corridor to V_1. As the reachable sets of V_1 progresses to lane 2 at time step $k = 10$, another conflicting corridor arises, which is then allocated to V_2. The negotiated ISSs demand that V_2 keeps driving straight while allowing V_1 to choose freely from keeping its lane or swerving into the center lane. Fig. 6 visualizes the negotiated ISSs and the states of the planned trajectories; as can be seen, the states are all within the negotiated ISSs, thus safety is ensured. Alternatively, one can only use the negotiated ISSs in the last time step, if the safety within the planning cycle is already ensured by the motion planners.

B. Scenario II: Merging

In our second scenario, V_1 and V_2 approach a merging point at a roundabout. Note that for complex road geometries, the maneuver corridors of vehicles are projected onto the intersecting lanes to compute the conflicts (see Fig. 7). V_1 gets the conflicting corridor allocated at time step $k = 6$, and hence, is entitled to keep driving along its lane in the

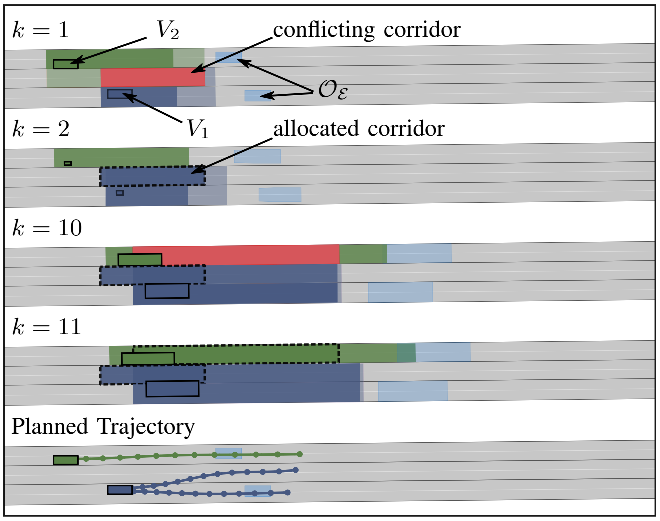
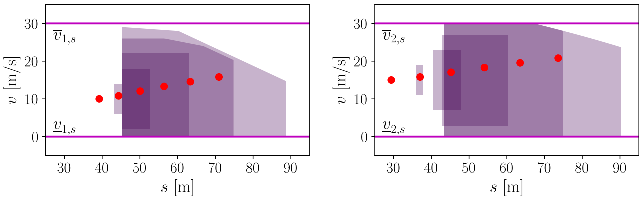


Fig. 5: Scenario I: Highway. The driving direction is from left to right. In the subplots for time step $k = 1$, and the planned trajectory, we show the initial positions of vehicles; in other cases the drivable area. Based on the negotiation result, vehicle V_1 is let to keep or change its lane.



(a) $\mathcal{S}_n^{\mathrm{N}}(k)$ of V_1 on lane 1, keep lane. (b) $\mathcal{S}_n^{\mathrm{N}}(k)$ of V_2 on lane 3, keep lane.

Fig. 6: The negotiated ISSs $\mathcal{S}_n^{\mathrm{N}}(k)$ (with $d \approx 0$ shown) of vehicles for time steps $k = 0$ to 25 with a step size of 5. The red dots indicate the state of the vehicles at each time step.

subsequent time steps. On the other hand, the ISS of V_2 is constrained to remain right in front of the merging point before V_1 passes.

C. Scenario III: Safety-Critical

Our last scenario demonstrates a safety-critical situation (see Fig. 8): a truck driving in the rightmost lane had an accident, causing it to block the two rightmost lanes. V_1 is already too close to the truck to perform a safe braking maneuver. The only way that V_1 can remain safe is to perform an evasive maneuver with the help of vehicle V_2. Even in such a critical situation, our method can determine the correct coordination of V_1 and V_2 to ensure safety. Similar situations can arise due to the presence of unexpected pedestrians, construction sites on roads, etc.

V. Conclusions

In this work, by incorporating ISSs, we propose a novel cooperative planning method that ensures the safety of a group of cooperative vehicles for an infinite time horizon. Our method computes ISS for each cooperative vehicle and derives the negotiated ISSs by resolving conflicts between their safe maneuver corridors. These conflicting corridors are allocated to the vehicles, and they serve as constraints for the propagation of subsequent reachable sets. In various scenarios with mixed-traffic, we demonstrate that our method ensures the overall safety of the cooperative group while not overly restricting the set of possible trajectories for the vehicles.

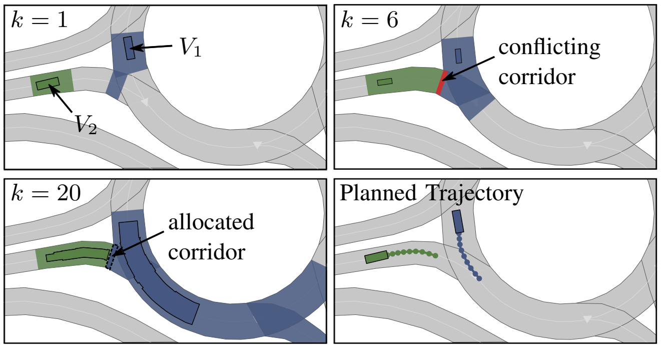


Fig. 7: Scenario II: Merging. The driving direction is indicated by the white arrows on the lanes. In the subplots for time step $k = 1$, and the planned trajectory, we show the initial positions of vehicles; in other cases the drivable area. Based on the negotiation result, vehicle V_1 is prioritized.

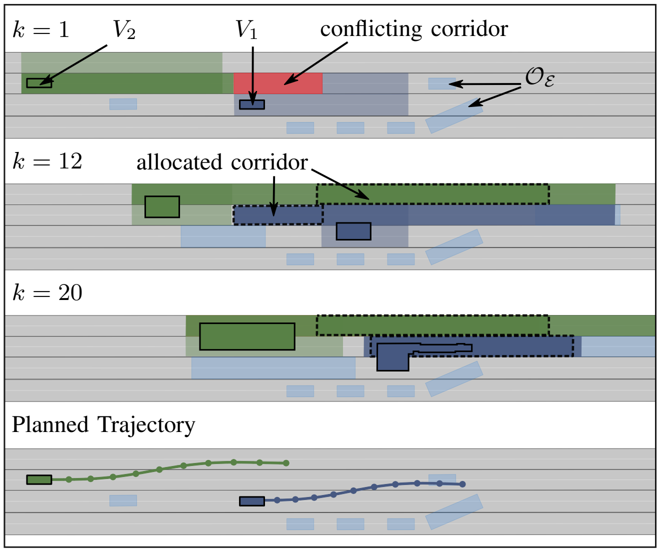


Fig. 8: Scenario III: Safety-Critical. The driving direction is from left to right. In the subplots for time step $k = 1$, and the planned trajectory, we show the initial positions of vehicles; in other cases the drivable area. Based on the negotiation result, both vehicles change lane to ensure the safety.

Acknowledgments

This project was funded by the Deutsche Forschungsgemeinschaft (German Research Foundation) within the Priority Programme SPP 1835 *Cooperative Interacting Automobiles* under grant No. AL 1185/4-2, and within the EU Horizon 2020 Programme *interACT* under grant No. 723395. The authors appreciate the fruitful collaboration with the project partners.

References

[1] C. Pek and M. Althoff, "Efficient computation of invariably safe states for motion planning of self-driving vehicles," in *Proc. of the IEEE Int. Conf. Intell. Robot. Syst.*, 2018, pp. 3523–3530.

[2] J. Rios-Torres and A. A. Malikopoulos, "A survey on the coordination of connected and automated vehicles at intersections and merging at highway on-ramps," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 5, pp. 1066–1077, 2017.

[3] K. Dresner and P. Stone, "Human-usable and emergency vehicle-aware control policies for autonomous intersection management," in *Workshop Agents Traffic Transp.*, 2006, pp. 17–25.

[4] ——, "Turning the corner: improved intersection control for autonomous vehicles," in *Proc. of the IEEE Intell. Veh. Symp.*, 2005, pp. 423–428.

[5] G. Sharon and P. Stone, "A protocol for mixed autonomous and human-ooperated vehicles at intersections," in *Proc. of the Int. Joint Conf. Auton. Agent. Multi-Agent Syst.*, 2017, pp. 151–167.

[6] S. Manzinger and M. Althoff, "Tactical decision making for cooperative vehicles using reachable sets," in *Proc. of the IEEE Int. Conf. Intell. Transp. Syst.*, 2018, pp. 444–451.

[7] D. Carlino, S. D. Boyles, and P. Stone, "Auction-based autonomous intersection management," in *Proc. of the IEEE Int. Conf. Intell. Transp. Syst.*, 2013, pp. 529–534.

[8] M. Vasirani and S. Ossowski, "A market-inspired approach for intersection management in urban road traffic networks," *J. Artific. Intell. Res.*, vol. 43, pp. 621–659, 2012.

[9] H. Schepperle and K. Böhm, "Auction-based traffic management: towards effective concurrent utilization of road intersections," in *Proc. of the IEEE Conf. E-Commerce Technol. and IEEE Conf. Enterprise Comput, E-Commerce, E-Services*, 2008, pp. 105–112.

[10] S. Manzinger and M. Althoff, "Negotiation of drivable areas of cooperative vehicles for conflict resolution," in *Proc. of the IEEE Int. Conf. Intell. Transp. Syst.*, 2018, pp. 1–8.

[11] M. W. Levin, H. Fritz, and S. D. Boyles, "On optimizing reservation-based intersection controls," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 3, pp. 505–515, 2017.

[12] I. A. Ntousakis, I. K. Nikolos, and M. Papageorgiou, "Optimal vehicle trajectory planning in the context of cooperative merging on highways," *Transp. Res. Part C Emerg. Technol.*, vol. 71, pp. 464–488, 2016.

[13] F. Zhu and S. V. Ukkusuri, "A linear programming formulation for autonomous intersection control within a dynamic traffic assignment and connected vehicle environment," *Transp. Res. Part C Emerg. Technol.*, vol. 55, pp. 363–378, 2015.

[14] D. Marinescu, J. Čurn, M. Bouroche, and V. Cahill, "On-ramp traffic merging using cooperative intelligent vehicles: a slot-based approach," in *Proc. of the IEEE Int. Conf. Intell. Transp. Syst.*, 2012, pp. 900–906.

[15] D. Marinescu, J. Čurn, M. Slot, M. Bouroche, and V. Cahill, "An active approach to guaranteed arrival times based on traffic shaping," in *Proc. of the IEEE Int. Conf. Intell. Transp. Syst.*, 2010, pp. 1711–1717.

[16] M. Nichting, D. Heß, J. Schindler, T. Hesse, and F. Köster, "Explicit negotiation method for cooperative automated vehicles," in *Proc. of the IEEE Int. Conf. Veh. Electron. Safety*, 2019, pp. 1–7.

[17] D. Heß, R. Lattarulo, J. Pérez, J. Schindler, T. Hesse, and F. Köster, "Fast maneuver planning for cooperative automated vehicles," in *Proc. of the IEEE Int. Conf. Intell. Transp. Syst.*, 2018, pp. 1625–1632.

[18] W. Damm, H.-J. Peter, J. Rakow, and B. Westphal, "Can we build it: formal synthesis of control strategies for cooperative driver assistance systems," *Math. Struct. in Comp. Science*, vol. 23, no. 04, pp. 676–725, 2013.

[19] M. Hilscher, S. Linker, and E.-R. Olderog, "Proving safety of traffic manoeuvres on country roads," in *Theories of Programming and Formal Methods*, 2013, pp. 196–212.

[20] S. Mitsch, S. M. Loos, and A. Platzer, "Towards formal verification of freeway traffic control," in *Proc. of the IEEE Conf. Cyber-Physical Syst.*, 2012, pp. 171–180.

[21] I. M. Mitchell, "Comparing forward and backward reachability as tools for safety analysis," in *Hybrid Systems: Computation and Control*, 2007, pp. 428–443.

[22] M. Althoff and J. M. Dolan, "Online verification of automated road vehicles using reachability analysis," *IEEE Trans. Robot.*, vol. 30, no. 4, pp. 903–918, 2014.

[23] 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 Conf. Decis. Control*, 2017, pp. 1517–1522.

[24] P. Falcone, M. Ali, and J. Sjöberg, "Predictive threat assessment via reachability analysis and set invariance theory," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1352–1361, 2011.

[25] M. Koschi and M. Althoff, "SPOT: a tool for set-based prediction of traffic participants," in *Proc. of the IEEE Intell. Veh. Symp.*, 2017, pp. 1686–1693.

[26] T. Fraichard, "A short paper about motion safety," in *Proc. of the IEEE Int. Conf. Robot. Autom.*, 2007, pp. 1140–1145.

[27] D. Althoff, J. J. Kuffner, D. Wollherr, and M. Buss, "Safety assessment of robot trajectories for navigation in uncertain and dynamic environments," *Auton. Robots*, vol. 32, no. 3, pp. 285–302, 2012.

[28] 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. Intell. Robot. Syst.*, 2015, pp. 3470–3477.

[29] 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. Intell. Transp. Syst.*, 2017, pp. 1–6.

[30] M. Althoff, M. Koschi, and S. Manzinger, "CommonRoad: composable benchmarks for motion planning on roads," in *Proc. of the IEEE Intell. Veh. Symp.*, 2017, pp. 719–726.

[31] S. Söntges and M. Althoff, "Computing the drivable area of autonomous road vehicles in dynamic road scenes," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 6, pp. 1855–1866, 2018.

[32] 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*, vol. 9690, 2016, pp. 175–190.

[33] 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 Intell. Veh. Symp.*, 2017, pp. 1477–1483.

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

[35] C. Frese, J. Beyerer, and P. Zimmer, "Cooperation of cars and formation of cooperative groups," in *Proc. of the IEEE Intell. Veh. Symp.*, 2007, pp. 227–232.