

Safety Verification of Autonomous Vehicles for Coordinated Evasive Maneuvers

Matthias Althoff, Daniel Althoff, Dirk Wollherr and Martin Buss

Abstract—The verification of evasive maneuvers for autonomous vehicles driving with constant velocity is considered. Modeling uncertainties, uncertain measurements, and disturbances can cause substantial deviations from an initially planned evasive maneuver. From this follows that the maneuver, which is safe under perfect conditions, might become unsafe. In this work, the possible set of deviations is computed with methods from reachability analysis, which allows to verify evasive maneuvers under consideration of the mentioned uncertainties. Since the presented approach has a short response time, it can be applied for real time safety decisions. The methods are presented for a numerical example where two autonomous cars plan a coordinated evasive maneuver in order to prevent a collision with a wrong-way driver.

I. INTRODUCTION

Recently, numerous autonomous vehicle projects have been realized, among them the project *Cognitive Automobiles* [1] in which this work has been partly carried out. It is out of question that safety is of paramount importance for autonomous vehicles, since their development should reduce the number of road accidents. For slow moving autonomous vehicles, it is sufficient to check if the set of occupied positions does not intersect any obstacle when the planned trajectory is almost perfectly followed [2]. However, one cannot assume that a planned trajectory is perfectly followed for evasive maneuvers due to e.g. tire slip, uncertain parameters, uncertain initial states, disturbances and so on. In order to compute the reachable positions under the mentioned uncertainties, the reachable set of other states such as e.g. velocity, orientation, and side slip angle has to be computed as well. The online safety verification of planned trajectories using reachability analysis is the subject of this paper.

The literature on reachability analysis applied to autonomous vehicles and car-like robots is rather limited. Reachable sets of obstacles with a velocity bound can be described by circles around the initial position and are often used for path planning in dynamic environments, see e.g. [3]. Circles can also represent the reachable sets of the slightly more complicated maximum acceleration model [4]. An approximate solution of reachable positions of robots has been used in [5] for a multi robot system. For the lateral control of vehicles, reachable sets of the deviation along a planned path have been investigated in [6]. However, in that work, the reachable set is obtained from worst case simulations, but no proof is given which guarantees the enclosure of all reachable states by the worst case simulations. Provable

results using simulations can only be obtained with special methods developed in e.g. [7]–[9].

There is more literature on safety verification of transportation systems which is not necessarily performed by reachability analysis. There is work on verification algorithms for platooning of road vehicles [10], [11], air traffic safety [12]–[14] and rail traffic safety [15], [16]. However, a verification algorithm for evasive maneuvers of road vehicles is novel to the best knowledge of the authors. An overview of reachable set algorithms for different problem classes can be found in [17].

II. CONTRIBUTION AND MOTIVATION

The contribution of this work is to prove the safety of evasive maneuvers with constant velocity for autonomous vehicles before the maneuver is executed. This is done by considering the set of tracking errors for a planned trajectory into the collision detection. It is assumed that the trajectory of evasive maneuvers is provided from a planner which is not subject of this work.

The tracking errors are mainly caused by measurement uncertainties, uncertain system parameters, and disturbances. In order to prove the safety, the set of positions possibly occupied by each vehicle under the mentioned uncertainties has to be obtained. This set is also referred to as the *occupancy set* from now on. If the occupancy sets of all traffic participants do not intersect for a predefined time horizon $t \in [0, t_f]$, the maneuver of each traffic participant is safe for this time horizon. In order to extend the safety verification for an infinite time horizon ($\forall t > 0$), possible follow-up maneuvers in which all traffic participants come to a standstill have to be planned. Since the vehicles are not moving anymore after a finite time horizon, this trick allows to verify infinite time horizons if the original and the follow-up maneuver (which is not executed) are collision-free.

If the planned trajectory is unsafe, the trajectory planner has to replan the trajectory or execute the trajectory with the least intersection of occupancy sets.

A. Known Versus Unknown Plans of Other Vehicles

Next, it is motivated why the concept of occupancy sets is only reasonable when the plan of other traffic participants is known, e.g. broadcast via wireless communication. Thereto, a scenario is considered in which two vehicles pass each other on a straight road, see Fig. 1. Assuming that the maximum force applicable between tires and road is isotropic (independent of the direction) according to *Kamm's circle*, the reachable set of a vehicle at time t can be described

All authors are with Institute of Automatic Control Engineering (LSR), Technische Universität München, 80290 München, Germany {althoff, da, dw, mb}@tum.de

by a circle [4]. Since the circles overlap already after a prediction horizon of only one second ($v_0 = 20$ m/s for both vehicles) an everyday situation has to be classified as unsafe, see Fig. 1. However, when the vehicles broadcast their plans of driving straight, the situation is verified as safe, unless the tracking errors are abnormally large.

The presented scenario demonstrates that if no maneuver plans are exchanged, one has to work with probabilistic methods. Those methods would compute a low crash probability for the presented scenario, which would be a sufficient condition for continuing the plan of following the straight road. Probabilistic approaches for the safety analysis of traffic scenes have been presented in e.g. [18], [19].

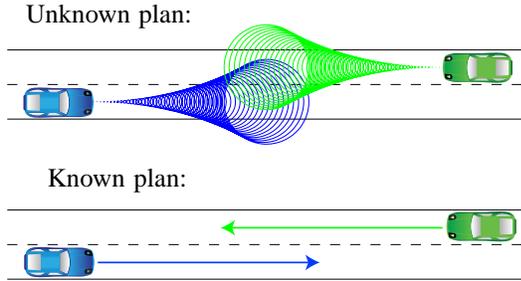


Fig. 1. Known versus unknown plans in a straight road scenario.

B. Braking Versus Evasive Maneuvers

Finally, it is motivated why this work focuses on evasive maneuvers since the autonomous vehicle might also plan a braking maneuver. This is justified since the computation of reachable sets for braking maneuvers is trivial, as shown next.

The reachable set of a braking maneuver is influenced by the deceleration which is uncertain within $a^{\text{dec}} \in [\underline{a}^{\text{dec}}, \bar{a}^{\text{dec}}]$ due to varying tire friction, where $\underline{a}^{\text{dec}}$ is the lower and \bar{a}^{dec} is the upper limit. The initial position $s_0 \in [\underline{s}_0, \bar{s}_0]$ and velocity $v_0 \in [\underline{v}_0, \bar{v}_0]$ are also uncertain due to measurement uncertainties. Clearly, after the integration of the acceleration and the velocity, the limits of reachable positions $s(t)$ are:

$$\underline{s}(t) = \underline{s}_0 + \underline{v}_0 t + \frac{1}{2} \underline{a}^{\text{dec}} t^2, \quad \bar{s}(t) = \bar{s}_0 + \bar{v}_0 t + \frac{1}{2} \bar{a}^{\text{dec}} t^2.$$

After additionally considering the size of the vehicle body, the occupancy set is obtained.

Next, the model for the lateral vehicle dynamics is introduced which allows to compute the reachable set of more complex evasive maneuvers.

III. SYSTEM MODEL

One of the most widely used models for road vehicles is the bicycle model whose name originates from the fact that the two wheels of each axle are lumped into one wheel located at the middle of the vehicle. This is also depicted in Fig. 2, where x_{CG} is the center of gravity, l_r , l_f , d_T , d_S are the distances from the center of gravity to the axles, the front, and the tail sensor. The sensors measure the distances Δy_T and Δy_S to the reference trajectory and are chosen

according to the control concept introduced subsequently. The reference trajectory is modeled as piecewise circular arcs with curvature ρ_{ref} . The steering angle is denoted by δ_f and the velocity by v . The state space model of the bicycle model is according to [20]:

$$\begin{bmatrix} \Delta \dot{y}_S \\ \Delta \ddot{y}_S \\ \Delta \dot{y}_T \\ \Delta \ddot{y}_T \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 1 & 0 & 0 \\ a_{21} & a_{22} & -a_{21} & a_{24} \\ 0 & 0 & 0 & 1 \\ a_{41} & a_{42} & -a_{41} & a_{44} \end{bmatrix}}_A \begin{bmatrix} \Delta y_S \\ \Delta \dot{y}_S \\ \Delta y_T \\ \Delta \dot{y}_T \end{bmatrix} + \underbrace{\begin{bmatrix} 0 & 0 \\ b_2 & -v^2 \\ 0 & h_4 v \\ b_4 & -v^2 \end{bmatrix}}_B \begin{bmatrix} \delta_f \\ \rho_{\text{ref}} \end{bmatrix} \quad (1)$$

where

$$\begin{aligned} a_{21} &= \frac{h_2}{M h_4} - \frac{d_S h_1}{I_\Psi h_4}, & a_{22} &= \frac{h_1 - d_T h_2}{M v h_4} + \frac{d_S (d_T h_1 - h_3)}{I_\Psi v h_4}, \\ a_{41} &= \frac{h_2}{M h_4} + \frac{d_T h_1}{I_\Psi h_4}, & a_{42} &= \frac{h_1 - d_T h_2}{M v h_4} - \frac{d_T (d_T h_1 - h_3)}{I_\Psi v h_4}, \\ b_2 &= \mu c_f \left(\frac{1}{M} + \frac{d_S l_f}{I_\Psi} \right), & a_{24} &= -\frac{h_1 + d_S h_2}{M v h_4} + \frac{d_S (d_S h_1 + h_3)}{I_\Psi v h_4}, \\ b_4 &= \mu c_f \left(\frac{1}{M} - \frac{d_T l_f}{I_\Psi} \right), & a_{44} &= -\frac{h_1 + d_S h_2}{M v h_4} + \frac{d_T (d_S h_1 + h_3)}{I_\Psi v h_4}, \end{aligned}$$

for which the auxiliary variables

$$\begin{aligned} h_1 &= \mu(c_r l_r - c_f l_f), & h_2 &= \mu(c_f + c_r), \\ h_3 &= \mu(c_r l_r^2 + c_f l_f^2), & h_4 &= d_S + d_T, \end{aligned}$$

have been used.

For the control of vehicles along planned trajectories, the controller presented in [20] is used which has been verified experimentally within the California PATH program. The controller is a state feedback controller $\delta_f = -k^T x$ (x : state, δ_f : steering input, k : controller gain), where

$$k^T = [0.510 \quad 0.087 \quad -0.280 \quad -0.024]. \quad (2)$$

The parameters of the considered car are listed in Tab. I. It is remarked that the same control approach has been applied in another study on steering controllers [21]. Other works that have developed steering controllers and evasive maneuvers can be found in [22]–[24].

For the evading scenario considered later, the initial velocity of the vehicle is uncertain within an interval such that the elements a_{22} , a_{42} , a_{24} and a_{44} of the system matrix A are uncertain within an interval. Matrices whose elements can take values within intervals are also referred to as *interval matrices*. From now on, the set of system and input matrices is denoted by $A = [\underline{A}, \bar{A}]$ and $B = [\underline{B}, \bar{B}]$, respectively. The elements of A and B which are not uncertain have the same left and right limit. The interval matrices allow to describe the bicycle model in (1) together with the controller in (2) by the differential inclusion

$$\dot{x} \in \mathcal{A}x + \mathcal{B}u, \quad (3)$$

where x is the state vector and u the input vector. The form of writing the system equations as a linear differential

inclusion allows to apply the proposed reachability algorithm as presented in the next section. If the velocity was modeled as another state instead of an uncertain parameter, the above model would become nonlinear and thus more complicated to analyze. There are methods for the reachability analysis of nonlinear systems, but they would not yet fulfill the tight restrictions on the allowed computational time [25]. This problem is subject of future work.

TABLE I
VEHICLE PARAMETERS.

M	I_{Ψ}	l_f	l_r
1573 kg	2873 kg m ²	1.1 m	1.58 m
d_S	d_T	$c_f = c_r$	μ
1.96 m	2.49 m	8000 N/rad	1

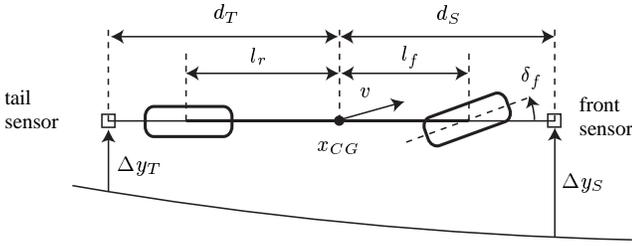


Fig. 2. Bicycle model.

IV. REACHABILITY ANALYSIS

For the dynamic model (3) described in the previous section, the set of reachable states is computed. The computation considers uncertain initial states $x(0) \in X_0$, uncertain inputs $u(t) \in U$, and uncertain parameters, where the latter is considered by the interval matrices \mathcal{A} and \mathcal{B} . The exact set of reachable states $R^e(r)$ for a time $t = r$ can be formally written as:

$$R^e(r) = \left\{ x(r) \left| x(t) = \int_0^t [Ax(\tau) + Bu(\tau)] d\tau, \right. \right. \\ \left. \left. x(0) \in X_0, A \in \mathcal{A}, B \in \mathcal{B}, \forall t : u(t) \in U \right\}.$$

However, the reachable set of linear systems can be exactly computed only for time invariant linear systems whose eigenvalues are rational or purely imaginary [26]. Since this is not the case for the bicycle model, the reachable set $R(r)$ has to be over-approximated, so that $R(r) \supset R^e(r)$. The over-approximated set for a time interval $t \in [0, r]$ is defined as the union of all $R(t)$ for $t \in [0, r]$: $R([0, r]) = \bigcup_{t \in [0, r]} R(t)$.

A. Overview of Reachable Set Computations

This section deals with the computation of reachable sets for linear continuous systems with uncertain system and input matrices as in (3). In order to focus on the application of reachability analysis to evasive maneuvers, only the most important aspects of reachability analysis are treated. Many approaches, e.g. [27]–[30] developed for the

over-approximative computation of reachable sets of linear systems use the following three basic steps:

- 1) computation of the reachable set \hat{R} without input at the point in time $t_k := k \cdot r$, where r is the time increment,
- 2) generation of the convex hull of the time point solutions at t_{k-1} and t_k ,
- 3) enlargement of the convex hull to ensure enclosure of all trajectories for the current time interval $t \in [t_{k-1}, t_k]$ under all possible inputs.

These steps are also illustrated in Fig. 3. Note that \hat{R} denotes the reachable set of the homogeneous solution (no input), while R denotes the overall reachable set consisting of the homogeneous and the inhomogeneous solution. The separate computation of the homogeneous and the inhomogeneous solution is possible due to the superposition principle of linear systems. Finally, the reachable set for the complete time interval is obtained by the union of the intermediate time intervals:

$$R([0, t_f]) = \bigcup_{k=1}^{t_f/r} R([t_{k-1}, t_k]).$$

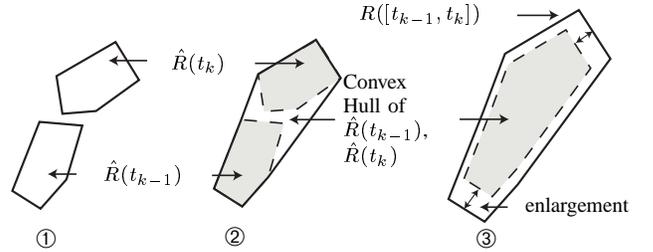


Fig. 3. Main steps for the computation of reachable sets.

B. Algorithmic Formulation

The first step for the computation of the reachable set in Fig. 3 is the computation of the reachable set at the next time step t_k . It is well known that the homogeneous solution of a linear system is $x(t_k) = e^{A r} x(t_{k-1})$, where A is the system matrix and r the time increment. Thus, the reachable set of the homogeneous solution is

$$\hat{R}(t_k) = e^{A r} \hat{R}(t_{k-1}). \quad (4)$$

Since $e^{A r}$ shifts the time by r , the reachable set for a time interval can be updated in the same manner: $\hat{R}([t_k, t_{k+1}]) = e^{A r} \hat{R}([t_{k-1}, t_k])$. However, the system matrix $A \in \mathcal{A}$ is uncertain in this work which requires to compute the set of exponential matrices $e^{A r} = \{e^{A r} | A \in \mathcal{A}\}$. This procedure and the computation of reachable sets for a time interval $\hat{R}([0, r])$ is explained in detail in [30] for the interested reader.

As the superposition principle is applicable to linear systems, the reachable set of the inhomogeneous solution denoted by $\tilde{R}(t)$ is computed separately as shown in [30]. Combining the homogeneous solution with the inhomogeneous solution and applying the time shift from (4) yields the main algorithm for the computation of the reachable set:

$$\begin{aligned}\hat{R}([t_k, t_{k+1}]) &= e^{A r} \hat{R}([t_{k-1}, t_k]), \text{ (hom. sol.)} \\ \bar{R}([t_k, t_{k+1}]) &= e^{A r} \bar{R}([t_{k-1}, t_k]) + \bar{R}([0, r]), \text{ (inhom. sol.)} \\ R([t_k, t_{k+1}]) &= \hat{R}([t_k, t_{k+1}]) + \bar{R}([t_k, t_{k+1}]) \text{ (overall sol.)}\end{aligned}$$

Note that the addition of two sets $\hat{R}([t_k, t_{k+1}])$ and $\bar{R}([t_k, t_{k+1}])$ is also called Minkowski addition where each element of one set is added to each other value of the other set: $A + B = \{a + b | a \in A, b \in B\}$. The algorithm can be simplified to

$$R([t_k, t_{k+1}]) = e^{A r} R([t_{k-1}, t_k]) + \bar{R}([0, r]). \quad (5)$$

The set of reachable states is used in the next section to obtain the set of occupied positions of the considered vehicles.

V. SET OF OCCUPIED POSITIONS

This section deals with the computation of the occupancy set, i.e. the set covering all areas occupied by the vehicle body for time intervals $[t_{k-1}, t_k]$. The set of vehicle centers is simply obtained from the velocity uncertainty $v \in [\underline{v}, \bar{v}]$ and the initial position $s_0 \in [\underline{s}_0, \bar{s}_0]$:

$$\underline{s}^*([t_{k-1}, t_k]) = \underline{s}_0 + \underline{v} t_{k-1}, \quad \bar{s}^*([t_{k-1}, t_k]) = \bar{s}_0 + \bar{v} t_k.$$

Since $s^*(t)$ refers only to the position of the vehicle center, the length l of the vehicle has to be taken into account: $\underline{s} = \underline{s}^* - 0.5 l$ and $\bar{s} = \bar{s}^* + 0.5 l$.

It remains to consider the set of deviations from the reference trajectory. In order to obtain a simple geometry of the occupancy set, it is represented by a trapezoid, see Fig. 4. The directions of the parallel sides are determined by the straight line from the position $\underline{s}([t_{k-1}, t_k])$ to $\bar{s}([t_{k-1}, t_k])$. The directions of the non-parallel lines are given by the normal vectors in $\underline{s}([t_{k-1}, t_k])$ and $\bar{s}([t_{k-1}, t_k])$. The parallel sides are pushed outwards due to:

- The *curvature of the arc segment*. An arc with length $\Delta s([t_{k-1}, t_k]) = \bar{s}([t_{k-1}, t_k]) - \underline{s}([t_{k-1}, t_k])$ covers an angle range $\Delta\phi = \Delta s \rho_{\text{ref}}$, where ρ_{ref} is the curvature. From elementary geometry it follows that the corresponding parallel side of the trapezoid have to be pushed outside by $h = \frac{1}{\rho_{\text{ref}}}(1 - \cos(0.5 \Delta s \rho_{\text{ref}}))$, where $\lim_{\rho_{\text{ref}} \rightarrow 0} h = 0$.
- The *intervals of deviation* $\Delta y_S([t_{k-1}, t_k])$ and $\Delta y_T([t_{k-1}, t_k])$ from the reference trajectory which is obtained from the reachable set of the controlled bicycle model (3).

In order to retain the trapezoidal representation of the occupancy set, the parallel sets have to be pushed outwards by the extreme values of Δy_S and Δy_T . The maximum deviation interval $\Delta y \in [\Delta \underline{y}, \Delta \bar{y}]$ is: $\Delta \underline{y} = \min(\Delta \underline{y}_S, \Delta \underline{y}_T)$ and $\Delta \bar{y} = \max(\Delta \bar{y}_S, \Delta \bar{y}_T)$.

- The *width* w of the vehicle.

The values contributing to the enlargement of the occupancy set are also marked in Fig. 4.

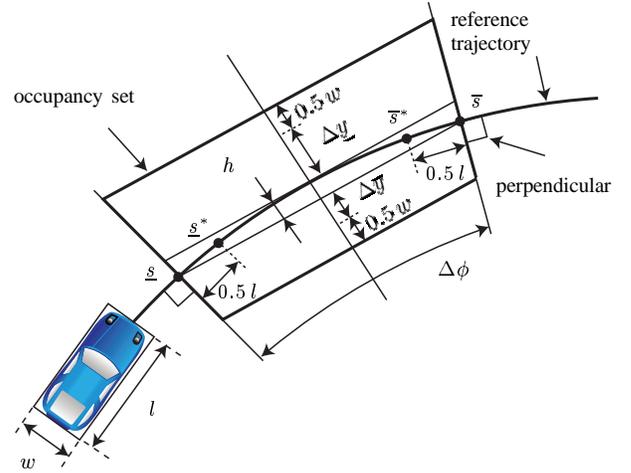


Fig. 4. Occupancy set of the vehicle.

VI. NUMERICAL EXAMPLE

The numerical example considers a situation in which a wrong-way driver threatens two autonomously driving vehicles on a road with three lanes. In order to minimize the risk of a crash with the wrong-way driver, both vehicles A and B plan a coordinated lane change maneuver as shown in Fig. 7. It is assumed that the wrong-way driver does not change lanes so that the task is to clear the leftmost lane as fast as possible. Since the verification is limited to communicating vehicles, the wrong-way driver is not considered in the verification.

In this scenario, both evading vehicles have parameters as listed in Tab. I and the controller parameters from (2). The reference trajectory of both vehicles consists of two arcs. The curvature values ρ_{ref} are chosen such that the nominal lateral accelerations along the reference trajectory are $a_{lat}^A = 0.4 g$, $a_{lat}^B = 0.3 g$ for vehicle A and B, where g is the gravitational acceleration. The velocity of vehicle A is $v^A = 20 \pm 1$ m/s and $v^B = 24 \pm 1$ m/s for vehicle B. After combining the controller with the bicycle model of the vehicle and after insertion of the vehicle and controller parameters, the dynamic model for vehicle A is:

$$\begin{aligned}\begin{bmatrix} \Delta \dot{y}_S \\ \Delta \dot{y}_T \\ \Delta \dot{y}_T \end{bmatrix} &\in \begin{bmatrix} 0 & 1 & 0 & 0 \\ -3.96 & [-1.40, -1.36] & 1.41 & [0.27, 0.33] \\ 0 & 0 & 0 & 1 \\ 4.33 & [0.14, 0.18] & -3.75 & [0.02, 0.08] \end{bmatrix} \begin{bmatrix} \Delta y_S \\ \Delta \dot{y}_S \\ \Delta y_T \\ \Delta \dot{y}_T \end{bmatrix} \\ &+ \begin{bmatrix} 0 \\ [-441, -361] \\ [84.6, 93.5] \\ [-441, -361] \end{bmatrix} \rho_{\text{ref}}\end{aligned}$$

The model of vehicle B is similar but differs due to the different velocity interval. The initial states are within $[-0.2, 0.2]$ m for Δy_S , Δy_T and within $[-0.2, 0.2]$ m/s for $\Delta \dot{y}_S$ and $\Delta \dot{y}_T$ for both vehicles. The reachable set is computed for a time step size of 0.04 sec.

A. Reachable Set

The reachable set of vehicle A for the first arc of the reference trajectory together with exemplary trajectories are

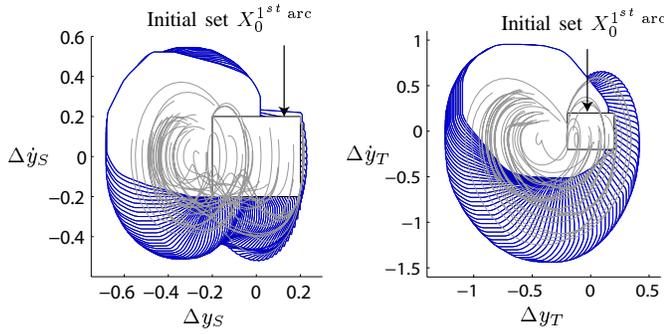


Fig. 5. Reachable set for the movement along the first arc. The curves represent exemplary trajectories starting in the set of initial states.

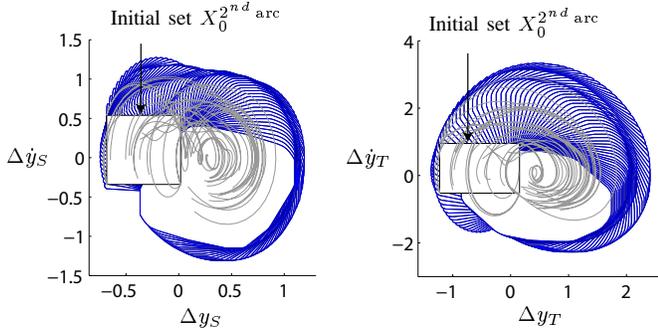


Fig. 6. Reachable set for the movement along the second arc. The curves represent exemplary trajectories starting in the set of initial states.

plotted in Fig. 5. It can be observed that no trajectory leaves the reachable set and that the reachable set is not overly conservative, i.e. the trajectories are not far away from the boundaries of the reachable set. The result for vehicle B is similar and thus not shown.

For the second arc, the input ρ_{ref} is changed. Due to the uncertain vehicle velocity v , the time point for the change of ρ_{ref} is also uncertain within $[\underline{t}_{switch}, \bar{t}_{switch}]$. The computed over-approximative set of initial states for the computation along the second arc is: $X_0^{2^{nd} arc} = R^{1^{st} arc}([\underline{t}_{switch}, \bar{t}_{switch}])$. The union of reachable sets for the time interval $[\underline{t}_{switch}, \bar{t}_{switch}]$ is over-approximated by an axis-aligned box and serves as the new initial set for the reachability computations along the second arc, see Fig. 6. For the remaining time intervals, zonotopes are used as a representation of the reachable set [30]. The reachable set together with randomly generated trajectories of vehicle A for the second arc are shown in Fig. 6.

Note that the evasive trajectory can be composed by more than two arcs so that other paths such as clothoids can be approximated. In order to obtain an over-approximation of the reachability computations for more complicated paths, one can additionally specify the curvature ρ_{ref} to be uncertain within an interval for each arc segment.

B. Occupancy Set

In order to determine if a crash can occur, the reachable set of the states Δy_S , Δy_S , Δy_T and Δy_T is used to obtain the occupancy set for each time interval as presented in Sec.

V. The resulting occupancy sets for the described scenario are found in Fig. 7(a). In order to efficiently check if the occupancy sets of vehicle A and B intersect for any time interval, candidates for possible intersection are searched by over-approximating the occupancy set with bounding boxes and checking for their intersection. For all intersection candidates, it is finally checked if the trapezoids of the occupancy sets intersect.

The evasive maneuver in Fig. 7(a) is verified as safe. However, for a lateral acceleration of $a_{lat}^A = 0.6g$, the evasive maneuver cannot be verified collision-free as shown in Fig. 7(b). It has also been checked if any of the vehicles leaves the road boundary.

The computation time for the reachable set is 0.34 sec for one vehicle on an AMD Athlon64 3700+ processor (single core) in Matlab. This computational time is deterministic and scales linearly with the time horizon. The collision check took 0.05 sec on the same CPU. Note that the reachable sets can be computed from each vehicle and then send to the vehicle that does the collision check. The occupancy set that has to be broadcast in this scenario is 25.8 kb which can be broadcast with modern car to car communication in less than 0.05 sec such that the total computation time is 0.34 sec + 0.05 sec = 0.39 sec assuming an on-the-fly collision check. Note that it is also possible to compute with several alternative reference trajectories in parallel which drastically increases the probability of finding a safe evasive maneuver. In case no safe maneuver is found, one could execute the one causing the least intersection of occupancy sets. The computational time can be further decreased by using specialized hardware such as GPUs or DSPs since the computation of reachable sets is mainly performed using matrix multiplications, see (5) and [30].

VII. CONCLUSIONS

An approach for the safety verification of evasive maneuvers of autonomous vehicles has been presented. The main feature of this approach is the possibility to guarantee safety under uncertainties (measurements, system parameters, disturbances) by computing the reachable set of the vehicle. The reachable set allows obtaining the set of occupied positions which then guarantees safety if not intersected with any other set of occupied positions. The computations are efficient allowing the algorithms to be used online for the decision of executing planned evasive maneuvers. Possible extensions of the presented approach are the consideration of evasive maneuvers with time varying velocity and paths that are not limited to arc segments.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge partial financial support of this work by the Deutsche Forschungsgemeinschaft (German Research Foundation) within the Transregional Collaborative Research Centre 28 *Cognitive Automobiles* and the excellence initiative research cluster *Cognition for Technical Systems – CoTeSys*.

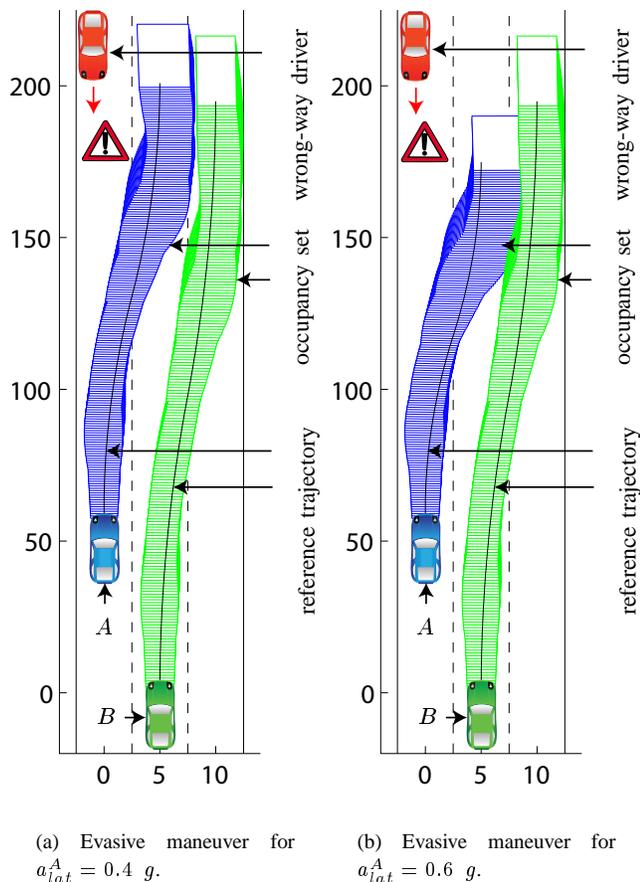


Fig. 7. Occupancy set of evasive maneuvers (the axis coordinates are distances in meters).

REFERENCES

- [1] C. Stiller, G. Färber, and S. Kammel, "Cooperative cognitive automobiles," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2007, pp. 215–220.
- [2] M. Vendittelli, J.-P. Laumond, and C. Nissoux, "Obstacle distance for car-like robots," *IEEE Transactions on Robotics and Automation*, vol. 15, pp. 678–691, 1999.
- [3] J. van den Berg, "Path planning in dynamic environments," Ph.D. dissertation, Utrecht University, 2007.
- [4] C. Schmidt, F. Oechsle, and W. Branz, "Research on trajectory planning in emergency situations with multiple objects," in *Proc. of the IEEE Intelligent Transportation Systems Conference*, 2006, pp. 988–992.
- [5] C. F. Chung, T. Furukawa, and A. H. Göktogan, "Coordinated control for capturing a highly maneuverable evader using forward reachable sets," in *Proc. of the IEEE International Conference on Robotics and Automation*, 2006, pp. 1336–1341.
- [6] J.-Y. Wang and M. Tomizuka, "Reachability analysis of hybrid lateral control problem for automated heavy-duty vehicles," in *Proc. of the American Control Conference*, 2001, pp. 1–6.
- [7] A. Girard and G. J. Pappas, "Verification using simulation," in *Hybrid Systems: Computation and Control*, ser. LNCS 3927. Springer, 2006, pp. 272–286.
- [8] A. Donzé and O. Maler, "Systematic simulations using sensitivity analysis," in *Hybrid Systems: Computation and Control*, ser. LNCS 4416. Springer, 2007, pp. 174–189.
- [9] J. Kapinski, B. H. Krogh, O. Maler, and O. Stursberg, "On systematic simulation of open continuous systems," in *Hybrid Systems: Computation and Control*, ser. LNCS 2623. Springer, 2003, pp. 283–297.
- [10] J. Lygeros, D. N. Godbole, and S. Sastry, "A verified hybrid controller

- for automated vehicles," in *Proc. of the 35th Conference on Decision and Control*, 1996, pp. 2289–2294.
- [11] R. Horowitz and P. Varaiya, "Control design of an automated highway system," *Proceedings of the IEEE*, vol. 88, pp. 913–925, 2000.
- [12] C. Livadas, J. Lygeros, and N. A. Lynch, "High-level modeling and analysis of the traffic alert and collision avoidance system (tcas)," *Proceedings of the IEEE*, vol. 88, pp. 926–948, 2000.
- [13] C. Tomlin, I. Mitchell, and R. Ghosh, "Safety verification of conflict resolution maneuvers," *IEEE Transactions in Intelligent Transportation Systems*, vol. 2, pp. 110–120, 2001.
- [14] A. Platzer and E. M. Clarke, "Formal verification of curved flight collision avoidance maneuvers: A case study," in *Proc. of the 16th International Symposium on Formal Methods*, ser. LNCS 5850. Springer, 2009, pp. 547–562.
- [15] A. Platzer and J.-D. Quesel, "European train control system: A case study in formal verification," in *Formal Methods and Software Engineering: 11th International Conference on Formal Engineering Methods*, ser. LNCS, vol. 5885. Springer, 2009, pp. 246–265.
- [16] W. Damm, A. Mikschl, J. Oehlerking, E.-R. Olderog, J. Pang, A. Platzer, M. Segelken, and B. Wirtz, "Automating verification of cooperation, control, and design in traffic applications," in *Formal Methods and Hybrid Real-Time Systems*, ser. LNCS 4700. Springer, 2007, pp. 115–169.
- [17] E. Asarin, T. Dang, G. Frehse, A. Girard, C. Le Guernic, and O. Maler, "Recent progress in continuous and hybrid reachability analysis," in *Proc. of the 2006 IEEE Conference on Computer Aided Control Systems Design*, 2006, pp. 1582–1587.
- [18] A. Eidehall and L. Petersson, "Statistical threat assessment for general road scenes using Monte Carlo sampling," *IEEE Transactions on Intelligent Transportation Systems*, vol. 9, pp. 137–147, 2008.
- [19] M. Althoff, O. Stursberg, and M. Buss, "Model-based probabilistic collision detection in autonomous driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 10, pp. 299 – 310, 2009.
- [20] J. Guldner, W. Sienel, H.-S. Tan, J. Ackermann, S. Patwardhan, and T. Bünte, "Robust automatic steering control for look-down reference systems with front and rear sensors," *IEEE Transactions on Control Systems Technology*, vol. 7, pp. 2–11, 1999.
- [21] J. I. Hernandez and C. Y. Kuo, "Lateral control of higher order nonlinear vehicle model in emergency maneuvers using absolute positioning gps and magnetic markers," *IEEE Transactions on Vehicular Technology*, vol. 53, pp. 372–384, 2004.
- [22] D. Smith, R. Benton, and J. Starkey, "Nonlinear-gain-optimised controller development and evaluation for automated emergency vehicle steering," *International Journal of Vehicle Design*, vol. 24, pp. 79–99, 2000.
- [23] G. Indiveri, A. Nüchter, and K. Lingemann, "High speed differential drive mobile robot path following control with bounded wheel speed commands," in *Proc. of the IEEE International Conference on Robotics and Automation*, 2007, pp. 2202–2207.
- [24] M. Durali, G. A. Javid, and A. Kasaiezadeh, "Collision avoidance maneuver for an autonomous vehicle," in *Proc. of the 9th International Workshop on Advanced Motion Control*, 2006, pp. 249–254.
- [25] M. Althoff, O. Stursberg, and M. Buss, "Reachability analysis of nonlinear systems with uncertain parameters using conservative linearization," in *Proc. of the 47th IEEE Conference on Decision and Control*, 2008, pp. 4042–4048.
- [26] G. Lafferriere, G. J. Pappas, and S. Yovine, "Symbolic reachability computation for families of linear vector fields," *Symbolic Computation*, vol. 32, pp. 231–253, 2001.
- [27] A. Girard, "Reachability of uncertain linear systems using zonotopes," in *Hybrid Systems: Computation and Control*, ser. LNCS 3414. Springer, 2005, pp. 291–305.
- [28] A. Chutinan and B. H. Krogh, "Computational techniques for hybrid system verification," in *IEEE Transactions on Automatic Control*, vol. 48, no. 1, 2003, pp. 64–75.
- [29] O. Stursberg and B. H. Krogh, "Efficient representation and computation of reachable sets for hybrid systems," in *Hybrid Systems: Computation and Control*, ser. LNCS 2623. Springer, 2003, pp. 482–497.
- [30] M. Althoff, O. Stursberg, and M. Buss, "Reachability analysis of linear systems with uncertain parameters and inputs," in *Proc. of the 46th IEEE Conference on Decision and Control*, 2007, pp. 726–732.