CoOP: V2V-based Cooperative Overtaking for Platoons on Freeways

Martin Strunz, Julian Heinovski, and Falko Dressler

***Abstract*— Platooning can improve road traffic safety, optimize traffic flows, and improve the driver's comfort. In order to enable platooning, V2V-concepts for platoon control, management, and safety have been developed. Cooperative maneuvers, such as overtaking for entire platoons, remain an open research problem. Such overtaking becomes necessary when we consider mixed traffic or platoons driving at different target speeds. In this paper, we propose *CoOP*, a V2V-based cooperative overtaking algorithm for platoons on freeways. Our CoOP concept enables safe overtaking maneuvers for entire platoons, i.e., without the need to disassemble and reassemble the platoon for the overtaking task. CoOP makes use of V2V communication – but does not limit itself to a specific technology. We validated safety and robustness of CoOP in a wide variety of simulated traffic scenarios. Our performance evaluation shows that CoOP performs well, even compared to the theoretic optimum of artificial platooning, although CoOP only relies on local information collected via direct V2V communication.**

I. Introduction

Researchers and car manufacturers are striving to make driving more enjoyable, cost-efficient, and ecological. Based on advances in the field of autonomous driving, various new concepts for managing road traffic have been proposed. One of such is platooning, which refers to convoys of multiple vehicles driving at a very short inter-vehicle distance [1], [2].

While platooning has a lot of advantages, it requires coordination via cooperative maneuvers [3], [4] for safe and efficient operation. One of the most difficult maneuvers is automated overtaking [5] or *cooperative overtaking*. Overtaking typically consists of an initial decision process, a lane change (to the overtaking lane), passing the slow vehicle(s), and a second lane change (back to the original lane). It is important to distinguish cooperative overtaking from Cooperative Overtaking Assistance (COA), where the overtaking intention of a vehicle is transmitted to other road users [6], and Collaborative Overtaking Assistance (ClOA), where vehicles actively negotiate the overtaking maneuver with other road users [7], [8].

During the entire overtaking maneuver, the current traffic environment has to be monitored in order to avoid collisions with other road users. Thus, overtaking requires a variety of decisions to be made, which is even more complex for platoons than for an individual vehicle. Since the platoon needs to perform two lane changes within this maneuver, possibly interfering with other road users, it is also a comparably safety-critical task. Still, cooperative overtaking is a very important task, since the platoon can lose some of its major advantages when being stuck behind slower vehicles. When not driving at the optimal or desired speed, the total travel time of the platoon may be significantly increased. Thus, a platoon should assess the necessity and possibility of overtaking as soon as possible, especially on freeways where differences in speed matter over longer distances.

One possible approach for a platoon to overtake is to disassemble the platoon before the maneuver, forcing each former platoon member to overtake individually. After finishing the individual maneuvers, the platoon can be reassembled. Algorithms for overtaking by individual vehicles have already been proposed [9], [10]. With this approach, however, during the whole overtaking time, the advantage of platooning can not be achieved. Furthermore, reassembling the platoon is a complicated maneuver itself that requires additional resources. Therefore, the goal is to perform the entire overtaking maneuver without disassembling the platoon in first place. While the problems of lane-keeping and lane-changing are covered widely in the literature, the problem of automated overtaking has attracted less attention [5].

In this paper, we propose *CoOP*, a cooperative overtaking algorithm for platoons on freeways that is based on Vehicle-to-Vehicle (V2V) communication. It decides whether it is necessary and safe to overtake and, if so, executes the entire overtaking maneuver in a safe manner by continuously assessing the current traffic environment. It does so by using only local sensor data from the platoon members that is distributed via potential unreliable V2V communication. We make no further assumptions about other road traffic participants. In particular, we do not assume that other road users are equipped with the same V2V communication technology. To the best of our knowledge, CoOP is the first cooperative overtaking algorithm for platoons that is only based on local sensor data and V2V communication.

Our main contributions can be summarized as follows:

- We propose CoOP, a novel V2V-based cooperative overtaking algorithm for platoons on freeways,
- we validate safety and robustness of CoOP in a variety of simulated traffic scenarios, and
- we assess the performance of CoOP in an extensive simulation study, showing that the performance is close to the theoretic optimum (i.e., artificial platooning) and omniscient knowledge about other vehicles.

II. Related Work

A variety of overtaking algorithms have already been proposed for individual, non-cooperative autonomous vehicles. Since these solutions are the closest match to our novel CoOP solution to cooperative overtaking for platoons, we discuss examples of such overtaking solutions in the following.

School of Electrical Engineering and Computer Science, TU Berlin, Germany `strunz@campus.tu-berlin.de`, `{heinovski,dressler}@ccs-labs.org`

Düring et al. [9] proposed a modularized algorithm for trajectory planning including overtaking. They distinguish influenceable and non-influenceable road users and assign costs to possible maneuvers. Their algorithm avoids conflicts between road users by checking their possible trajectories for collisions. It is, however, limited in input data and parameterization to maintain real-time capabilities. Similarly, Petrov and Nashashibi [5] developed a kinematic model for assessing overtaking as a tracking problem of virtual trajectories. Their system is independent of road markings and V2V communication. Additionally to trajectory data, Chen et al. [11], develop an automatic lane marking detection algorithm using onboard cameras. They obtain the data via V2V communication with other road users.

Following a different approach, Gong et al. [12] developed a two-part decision making model for overtaking. After determining the current traffic situation by sensor data, their system uses a hierarchical state machine for decision making that imitates human driver behavior. This system can lead to unpredictable behavior in some situations. Following a similar approach, Naranjo et al. [13] split the overtaking maneuver into different phases, including two dedicated phases for lane changing. The system uses V2V communication to share Global Positioning System (GPS) data that is fed into fuzzy-logic controllers. Thus, it does not need a reference trajectory.

Finally, approaches using model predictive control and machine learning have been proposed. Nilsson and Sjöberg [14] developed an algorithm to determine whether an overtaking maneuver is useful and possible. It also selects the driving lane for each step in the overtaking maneuver, however, the actual lane change execution is not part of the algorithm. Going one step further, Hoel et al. [15] trained a deep reinforcement network and successfully simulated an overtaking maneuver on a road with oncoming traffic. Their approach is limited as the agent will only be able to solve the type of situations that it was exposed to in the simulations [15].

For the actual lane change, Ulbrich and Maurer [16] propose to use the output of two signal processing networks as an input for a Markov-based decision algorithm. The networks define whether a lane change is possible and beneficial, and are intended to simplify the Markov model. The authors define three different regions of interest around the ego vehicle to assess the lane change possibility. Following up on this, Ulbrich and Maurer [17] study the lane change possibility and its benefits in more detail. In particular, they define a technique for determining how hard a vehicle approaching from behind must brake if another vehicle changes to its lane. Deceleration thresholds for the decision and lane change phases restrict lane change maneuvers. This system can be generalized to develop cooperative overtaking algorithms for platoons.

Samiee et al. [18] propose a lane change collision avoidance system that consists of three multi-layer controllers: decision making, path planning, and vehicle control. This work also considers dynamic traffic situations that change after the initial decision to execute the maneuver and explores possible threats during lane changing. A downside is the assumption that the lane changing vehicle has zero longitudinal acceleration and lane changes take place at a constant longitudinal velocity.

While these models and algorithms assess overtaking for individual autonomous vehicles, they do not cover overtaking for platoons. A very early proposal for a cooperative lane change maneuver for platoons has been made by Hsu and Liu [19]. While the authors focus on the operational level including lateral control, they do not describe when the platoon should change lanes, and safety aspects of their approach do not include other traffic. Another, still experimental, overtaking algorithm for platoons exists in Plexe 2.1 [20]. While it does perform individual lane changes for all platoon members, it uses oracle (i.e., omniscient) knowledge of the simulation environment to assess the presence of other road users. This, of course, makes the algorithm not realistic.

Overall, there is still a lack of proper solutions to cooperative overtaking for platoons. Such solutions need to include decision making, lane changing, and continuous assessments of the maneuver's safety, while not splitting the platoon into individual vehicles. To the best of our knowledge, our CoOP approach is the first V2V-based cooperative overtaking algorithm for platoons on freeways, supporting overtaking of the entire platoon without disassembling it.

III. CoOP – Cooperative Overtaking for Platoons

Driving tasks in general can be grouped into the following three categories: (1) navigation tasks on the *strategic level*, (2) guidance tasks on the *tactical level*, and (3) stabilization tasks on the *operational level* [21]. Our cooperative overtaking algorithm for platoons operates on the tactical level. Strategic navigation as well as detailed vehicle control are out of scope and can be handled by application-specific route planning and established platoon controllers, respectively.

A. Basic Assumptions

We make the following assumptions regarding the platoon members, the surrounding infrastructure, and the V2V system: Each platooning vehicle is equipped with Cooperative Adaptive Cruise Control (CACC) as well as V2V communication for the exchange of relevant maneuver-specific information. Following the CACC requirements, each vehicle is equipped with front and rear mid-range radar sensors that can measure the distance, speed, and position of objects with a distance of up to 160 m (front) and 80 m (rear). Such systems are already available in the industry, e.g., by BOSCH. Side cameras are responsible for detecting objects next to the vehicle on neighboring lanes. The resulting measurement areas are depicted in Figure 1. We define six areas of interest (based on [16]), with the *front left (FL)*, *rear left (RL)*, *front right (FR)*, and *rear right (RR)* being the most important ones. In addition, the *L* and *R* areas are defined as the combination of the *FL* & *RL* and *FR* & *RR* areas, respectively. Based on

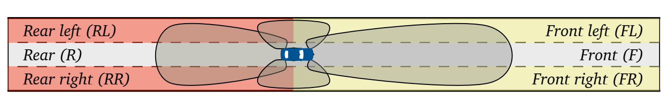


Fig. 1. A vehicle's sensor areas and the different areas of interest around.

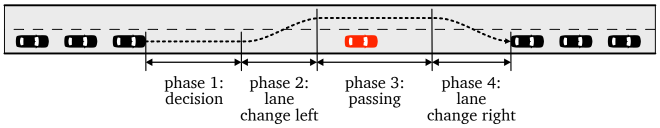


Fig. 2. The four phases of an overtaking maneuver in an exemplary scenario.

this, as an example, the closest vehicle in the *FL* area is called *FL* vehicle, its speed is v_{FL}. We assume that a vehicle can detect lanes automatically and is able to perform an automatic lane change to an adjacent lane when requested [5], [22]. An overtaking maneuver always starts on the original lane, passing the slower vehicle will take place on the overtaking lane. Depending on the direction of the lane change, the target lane is either the original lane or the overtaking lane. In this work, we assume right-hand-side driving.

We assume that the platoons performing the overtaking maneuvers have already been formed in an optimal way [23], e.g., regarding the order of the platoon members and the platoon's desired driving speed. Detailed modeling of V2V communication is also out of scope. V2V communication can, for example, be realized by means of cellular technologies like 4G/5G or ad-hoc technologies like DSRC [24]. We assume that platoon members can exchange messages among each other without errors. Messages can, however, be delayed due to retransmissions.

B. Overtaking Model

We model our overtaking algorithm in form of Finite State Machines (FSMs), allowing for a modular design. We divide the overtaking maneuver into the following four phases (see Figure 2) that are based on [5], [13], [25]: (1) decision for overtaking, (2) lane change to left (to the overtaking lane), (3) passing the slower vehicle, and (4) lane change right (back to the original lane). In phases (1) and (3), the decision of changing lanes is made, while in phases (2) and (4), the lane changes are actually carried out. Since the platoon leader is in charge of the entire overtaking maneuver, it is the only platoon member that technically performs an overtaking maneuver. Other platoon members only respond to requests by the leader and carry out its decisions (i.e., the lane change operation).

The corresponding FSM for the overtaking for the platoon leader is shown in Figure 3 and contains the aforementioned four phases: (1) the *idle* and *vehicle ahead* states, (2) the *lane change left* super-state, (3) the *passing* state, and (4) the *lane change right* super-state. The super-states represent a sequence of states for a lane change and are described in Section III-B and Section III-D for the platoon leader and all platoon followers, respectively.

Phase 1 – Decision: Initially, the platoon is driving on a freeway lane at its desired speed and in optimal formation. When the platoon leader detects a vehicle in the *F* area, it has to decide whether it is useful and possible to overtake this vehicle. This is done by using information about the platoon itself (e.g., the desired driving speed) and the speed of the vehicle in front. As this is just a decision phase, the algorithm does not intervene with the current traffic situation.

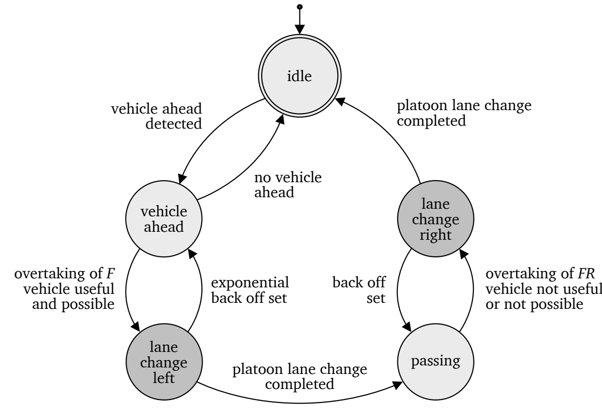


Fig. 3. FSM describing the complete cooperative overtaking maneuver for a platoon from the platoon leader's perspective. The four main states correspond to the four phases. The states *lane change left* and *lane change right* are super-states that are explained in more detail later in the paper.

For assessing the potential velocity gain in the neighboring lane on the left [26], the leader monitors the *FL* area for vehicles. A neighboring lane is considered faster and therefore beneficial, if

$$\min(v_{\text{desired}}, v_{\text{limit}}) - v_{\text{F}} \geq v_{\Delta}, \tag{1}$$

where v_{desired} is the platoon's desired speed, v_{limit} is the speed limit on the neighboring lane, v_{F} is the speed of the vehicle in front, and v_{Δ} is the minimum required speed difference of the platoon and the car to be overtaken. v_{Δ} depends on legal restrictions; we set $v_{\Delta} = 0.1\,\text{m/s}$.

Besides the mentioned minimum speed difference from Equation (1), there might be other restrictions to consider when determining the possibility of overtaking (e.g., other legal regulations). CoOP can easily be extended to include such regulations. However, for generalization of our algorithm, we do not consider such restrictions in this work.

Phase 2 – Lane Change Left: In this phase, the platoon executes the lane change to the left (i.e., to the overtaking lane), while the leader continuously assesses the safety of the maneuver (see Section III-C). Based on the sensor data from all of the followers, the leader decides whether to initiate the lane change. If the situation is considered unsafe, the platoon will stay on its original lane and, after a back-off time, the leader restarts the algorithm by going back to phase (1). Otherwise, the leader will change to phase (3) after completion of the lane change.

Phase 3 – Passing: The platoon is now driving in the overtaking lane to pass the slower vehicle. It can attempt to change back to the original lane, if no vehicle in the *FR* area is detected by the leader (anymore). As the algorithm will identify the slower vehicle as a vehicle that should (still) be overtaken, this assessment can start right after changing lanes and before passing the slower vehicle. The corresponding set of rules does not differ from phase (1), thus, the leader will make this decision only considering his own sensor data. Following the decision, the leader will change to phase (4) to perform a lane change to the original lane (i.e., to the right).

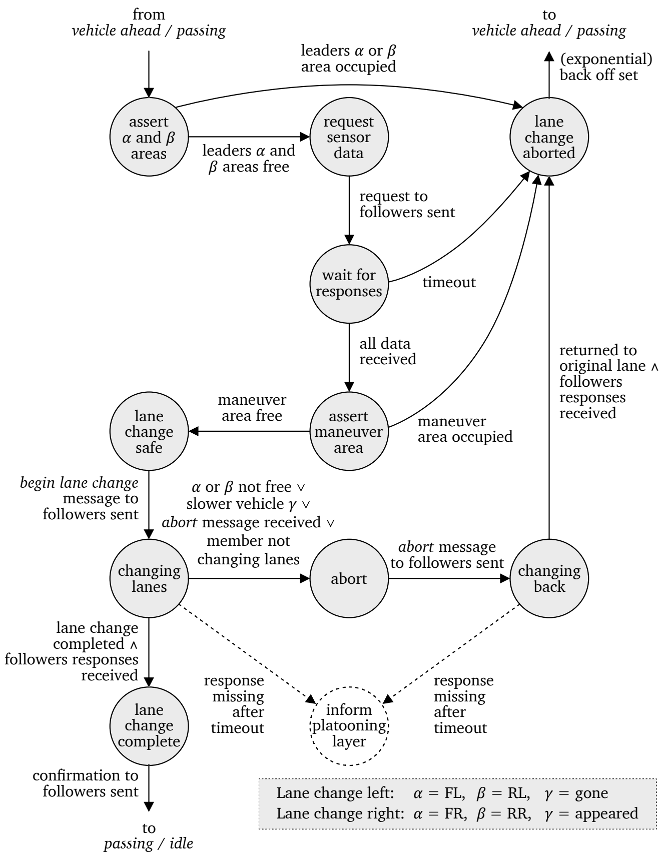


Fig. 4. FSM of the lane change sub-states for the platoon leader.

Phase 4 – Lane Change Right: In this phase, the platoon executes the lane change to the right (i.e., back to the original lane), while the leader continuously assesses the safety of the maneuver (see Section III-C). This is done in exactly the same manner as in phase (2) except that now the areas on the right are monitored. When the platoon successfully changed back to the original lane, the overtaking maneuver is completed. The leader changes back to phase (1). This concludes the overtaking maneuver and the algorithm can start from the beginning.

C. Lane Changing Model – Platoon Leader

The aforementioned *lane change left* and *lane change right* super-states of the overtaking FSM (see Figure 3) are, in fact, a sequence of states and represent the lane change FSM of the leader. We focus on the *lane change left* maneuver.

The first step of a lane change is to assess whether the leader's own maneuver areas α and β are free (see Figure 4). These areas are, depending on the direction of the lane change, *FL* & *RL* or *FR* & *RR* for a lane change to the left or to the right, respectively. If they are free, it sends a request to its followers to assess their own maneuver areas α and β. They send their information back to the leader, who can now evaluate the safety of the whole platoon's maneuver area. If it is free, the leader orders its followers to change lanes. A lane change is considered unsafe if at least one of the assessed areas is occupied; the maneuver will be aborted.

During the lane change, the safety of the maneuver is continuously assessed by monitoring the maneuver areas. If another vehicle is suddenly detected in the platoon's maneuver area (e.g., by merging into the same target lane), the lane change will be aborted to avoid a collision. If the slower vehicle that is to be overtaken accelerates or even disappears, the maneuver will be aborted as well.

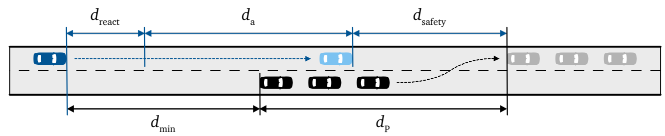


Fig. 5. Derivation of d_{min}. The approaching *RL* vehicle drives d_{react} during its driver's reaction time, d_{a} while decelerating to the speed of the platoon, and obeys its safety distance d_{safety}. At the same time, the platoon drives d_{P}, which gives the *RL* vehicle more space for deceleration. d_{min} denotes the minimum distance to perform a lane change.

When the leader has successfully reached the center of the target lane, he waits for acknowledgements from each follower indicating a successfully completed lane change. Due to the unreliable V2V communication system, messages can be received with a substantial delay. Our assumption that they will eventually be received successfully does not hurt the proper safety of the lane change. If message delays, e.g., for a *begin lane change* or *lane change complete* message, were infinite (i.e., the message is lost), a general communication problem between the platoon leader and its followers has occurred. Thus, not only the overtaking maneuver but the platooning operation itself is endangered and the leader changes to the *inform platooning layer* state.

When the platoon's lane change is completed, the leader informs its followers about this. All vehicles now change to the *passing* or *idle* state of their corresponding FSM, depending on the direction of the completed lane change.

1) Assertion of α and β Areas: The leader assesses whether his *FL* and *RL* (we focus on a *lane change left* maneuver here) areas are free. The *FL* area is being evaluated as free if the closest vehicle in this area is at least the safety distance (headway time) away. Even more important is the *RL* area [16]. A vehicle approaching from the rear should not be forced to brake, at least not very severely. We propose the following approach to compute the safety distance d_{min} (see Figure 5), which allows for the *RL* vehicle to decelerate from v_{RL} to v_{P} (based on [17]). The following two cases need to be considered:

(1) The *RL* vehicle is faster than the platoon ($v_{\text{RL}} > v_{\text{P}}$) and may be forced to brake with $a < 0$

$$d_{\text{min}} = -\frac{1}{2a}\left(v_{\text{P}} - v_{\text{RL}}\right)^2 + v_{\text{RL}}T_{\text{r}} + v_{\text{P}}T_{\text{g}}, \tag{2}$$

(2) The *RL* vehicle is at most as fast as the platoon ($v_{\text{RL}} \leq v_{\text{P}}$) and does not accelerate ($a \leq 0$)

$$d_{\text{min}} = v_{\text{RL}}\left(T_{\text{r}} + T_{\text{g}}\right), \tag{3}$$

with T_{r} being the reaction time and T_{g} being the desired time gap of the *RL* vehicle's driver. The *RL* area is considered free if the closest vehicle in the *RL* area is at least d_{min} away or if no *RL* vehicle is detected. In all other cases the *RL* area is considered occupied.

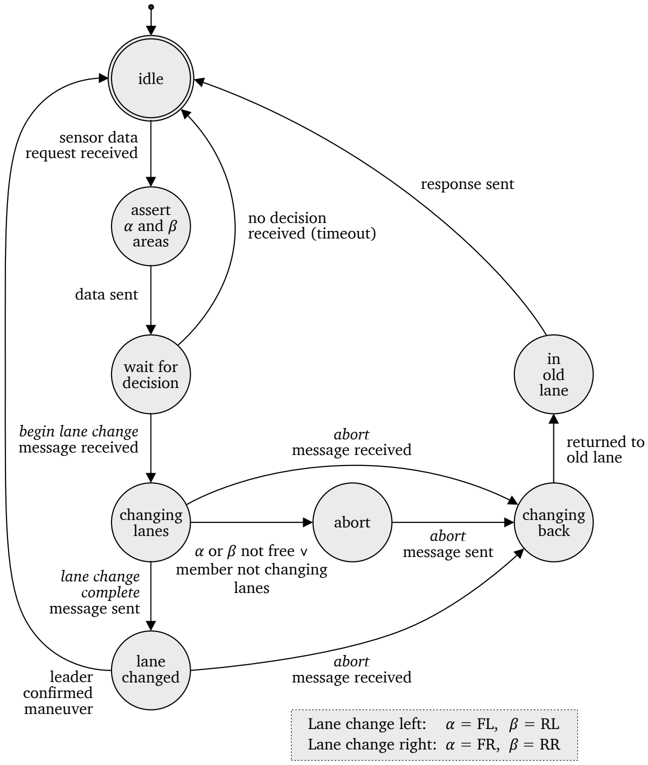


Fig. 6. FSM of the lane change sub-states for the platoon followers.

To compute d_{min}, we set the acceptable deceleration of the rear vehicle to $a = -1\,\mathrm{m/s^2}$ [17] for a lane change to the left. As a lane change to the right should not hinder any vehicle, we use $a = 0\,\mathrm{m/s^2}$ in this case. We use a reaction time of $T_{\text{r}} = 1.0\,\mathrm{s}$ and a time of $T_{\text{g}} = 0.8\,\mathrm{s}$ for the desired time gap [17], leading to a headway time of 1.8 s.

2) Changing Lanes: As the algorithm intervenes with the dynamic traffic situation, unforeseen disruptions can occur and require further hazard management. The necessity to abort the maneuver is evaluated like in the *assert α and β areas* state. However, Ulbrich and Maurer [17] propose a higher allowed deceleration for the *RL* vehicle, when the lane change is already in progress. In this case, we allow for $a = -3.5\,\mathrm{m/s^2}$.

D. Lane Changing Model – Platoon Followers

The FSM of the lane changing for platoon followers is shown in Figure 6. As the platoon leader is mainly in charge of the overtaking maneuver, the followers basically only react on requests of the leader and, eventually, perform the corresponding lane change. If something unexpected happens during the lane change, the follower will abort it and return to the old lane. A corresponding abort message is sent to the leader, who distributes it to all other platoon members. This procedure, which leaves the leader in charge, is not a threat to the other followers. The forwarding via the leader introduces an additional delay for the reception of the *abort* message. However, a follower will always immediately abort the overtaking maneuver if a dangerous situation occurs. A vehicle in danger never waits for the leader's decision.

IV. Evaluation

For validation and performance evaluation, we implemented our CoOP approach in a simulation environment. We assess the performance by comparing CoOP to the following cases:

- No-Overtaking – The platoon cannot overtake. This configuration serves as a worst case scenario.
- Artificial Long Vehicle (ALV) – The platoon is modeled by a single vehicle with the same length as the actual platoon. This represents the (unrealistic) optimal case for overtaking, as it avoids synchronization among the platoon members via V2V communication, individual lane changes for all platoon members, and their coordination.
- Oracle – The platoon utilizes the non-communication-based *coordinated lane change algorithm* of Plexe 2.1 [20]. This represents the (unrealistic) sub-optimal case for overtaking, since the algorithm avoids synchronization of platoon members via V2V communication, but uses individual lane changes for all platoon members.

A. Simulation Setup

Our simulations are based on Plexe 2.1 [20] and SUMO 1.7.0 [27]. Plexe allows to simulate platooning in a realistic manner by providing implementations of the underlying CACCs and a framework for handling platoons. Its Python API provides easy access to this functionality. This version of Plexe does not include simulation of V2V communication,

TABLE I
MOST RELEVANT SIMULATION PARAMETERS.

Parameter	Value
Freeway Scenario	
Number of lanes (one direction)	3
Lane width	3.2 m
Vehicle Type *Car*	
SUMO Type	passenger
Speed limit	33.3 m/s
Vehicle length	4.7 m
Driver's desired minimum headway time τ	1.8 s
Vehicle Type *Truck*	
SUMO vType	trailer
Speed limit	22.2 m/s
Vehicle Type *Platoon Car*	
SUMO vType	passenger
SUMO CF model	CC (Plexe)
Plexe CF model leader	ACC
Plexe CF model follower	CACC
Desired velocity	30.6 m/s
ACC headway time	1.0 s
CACC constant gap d_d	5.0 m
CoOP Algorithm	
Min. overtaking speed delta v_Δ	0.1 m/s
Max. deceleration for *RL* vehicle (before overtaking) a	−1.0 m/s^2
Max. deceleration for *RL* vehicle (during overtaking) a	−3.5 m/s^2
Max. deceleration for *RR* vehicle a	0.0 m/s^2
Max. distance to *F* vehicle when changing to the left	160 m
Time to stay in original lane after changing back t_{stay}	10.0 s

TABLE II

AVERAGE HOURLY TRAFFIC (DAYTIME, BETWEEN 6AM AND 10PM) IN ONE DIRECTION ON TYPICAL GERMAN FREEWAYS.[1]

Traffic Density	Cars/h	Trucks/h
Low	852	124
Medium	1790	244
High	2807	312

TABLE III

NORMAL DISTRIBUTION OF THE SPEED FACTOR AND THE RESULTING SPEED INTERVALS FOR TRUCKS AND CARS.

Spawn lane	1 (*Truck*)	2 (*Car*)	3 (*Car*)
Mean	1.0	1.0	1.0
Deviation	0.2	0.2	0.2
Min value	0.875	0.75	1.0
Max value	1.25	1.0	1.25
Speed Range	19.4 - 27.8 m/s	25.0 - 33.3 m/s	33.3 - 41.7 m/s

since it abstracts the exchange of control messages for the platoon operation. Therefore, we developed a simple "send and receive" V2V communication system that is only used for our algorithm. Messages can be sent during every simulation step and are received in one of the next simulation steps. Thus, we model a very simple message inter-arrival time that mimics channel access as well as retransmissions with an exponentially distributed delay [28].

Our simulation scenario is a typical freeway with three lanes in one direction. The most relevant simulation parameters are summarized in Table I (we do not show default values of the used tools). For the traffic, we use three different vehicle classes that are based on SUMO's vehicle classes: passenger cars, trucks, and platooning cars. Since it is necessary to simulate the correct lane change behavior, we use SUMO's sub-lane model SL2015 [29]. This allows to simulate two vehicles simultaneously merging from different lanes to the same target lane, which can lead to an unsafe situation that has to be handled by our algorithm. The truck type shares most of the parameters of the passenger car type except for its physical abilities like maximum acceleration. A platoon is built of passenger cars and consists of four or eight cars. The platoon enters the simulation at the beginning of the freeway on the first lane after 380 s and drives for 20 km.

To model realistic road traffic for our simulations, we computed three different traffic densities.[1] The resulting parameters are shown in Table II. The speed factor of each vehicle is sampled from a normal distribution (see Table III).

We performed verification and validation tests to ensure that the FSMs are implemented correctly and that our algorithm meets the intended application. Our validation tests include various different traffic situations to test the decision making as well as the lane changing. We show an example of our test cases for CoOP in Figure 7. To address safety of the

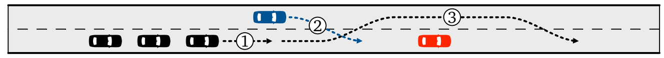


Fig. 7. Example of our validation test cases for CoOP: (1) A vehicle (red) in front of the platoon (black) is only slightly slower than the platoon, hence it is not overtaken. (2) Another vehicle (blue) merges into the gap between the platoon and the front vehicle and becomes so slow that overtaking (it) is possible. (3) The platoon now needs to overtake both vehicles (blue and red) as the gap between both vehicles is too small to change back to the original lane immediately after overtaking only the merged vehicle.

platoon and robustness of the protocol, in all test cases, we ensure the correct transitioning through the FSMs and that various pre-defined constraints (e.g., no collisions) are met.

B. Simulation Results

For the evaluation of our algorithm, we analyze various (i.e., 40) randomized simulation runs per configuration, carefully monitoring the confidence intervals. For better interpretation of the results, we plot average results plus standard deviation. Because one main goal of overtaking is the reduction of total travel time, we report the platoon's average speed for each algorithm. Additionally, we measure the platoon's average lateral position among all available lanes on the freeway during its 20 km drive. To analyze how long it takes the platoon to change to the overtaking lane, we measure the average time between the detected usefulness of overtaking and driving in the overtaking lane (later referred to as *lane change time*).

We first look at the average speed in Figure 8a. As expected, the ALV case achieves with 29.3 m/s the highest average speed of the three algorithms. CoOP is the second fastest (29.1 m/s) and only 1 % slower than ALV. The Oracle case comes in third with 28.6 m/s, being 2 % slower than ALV. The No-Overtaking case is significantly slower (21.5 m/s), making our algorithm 35 % faster than not overtaking at all.

There are two main reasons why ALV is the fastest one: (1) its capability of changing faster to the overtaking lane than CoOP (30.3 s vs. 42.7 s, see Figure 8b) and (2) it is driving longer on the faster overtaking lane (average lateral position of 2.6 m, see Figure 8c).[2]

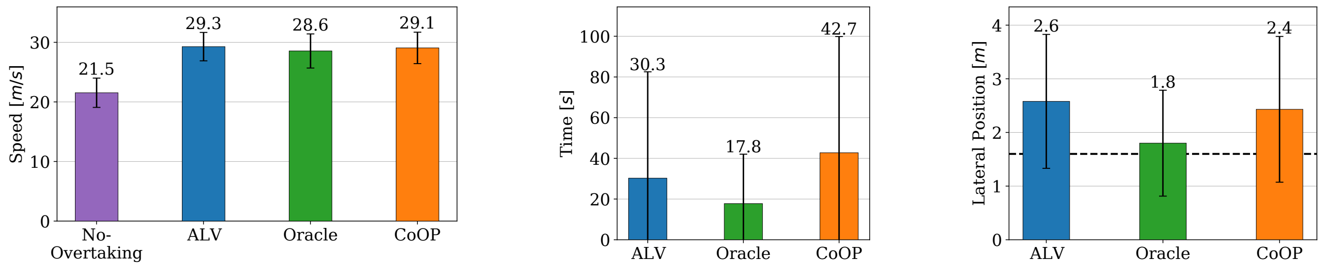
CoOP has the longest lane change time (42.7 s) due to the delay-impacted V2V messaging system. Nevertheless, its average speed is slightly higher than the Oracle case due to its higher average lateral position (2.4 m), resulting in driving more in the faster overtaking lane. The Oracle case has the fastest lane change time (17.8 s), which could be a candidate for the highest average speed. But it is the one with the lowest average lateral position (1.8 m).[3]

For a better understanding of when the different cases attempt to overtake, we removed the passenger car traffic from the simulation, leaving only trucks and the platoon in the simulation. When the platoon catches up with a truck and

[1] Data based on Federal Highway Research Institute (Bundesanstalt für Straßenwesen, BASt) – Automatische Zählstellen 2018, `https://www.bast.de/BASt_2017/D/Verkehrstechnik/Fachthemen/v2-verkehrszaehlung/Aktuell/zaehl_aktuell_node.html` (last accessed: 07/10/2020)

[2] The center of lane 1 and lane 2 is at 0 m and at 3.2 m, respectively. Therefore, a value of >1.6 m indicates that the platoon is driving mainly in the overtaking lane.

[3] Actually, the value for the Oracle platoon should be a little higher as it directs the platoon not towards the center of the middle lane (as CoOP and ALV do), but only to its edge. As a result, the platoon's lateral position for lane 2 is lower.

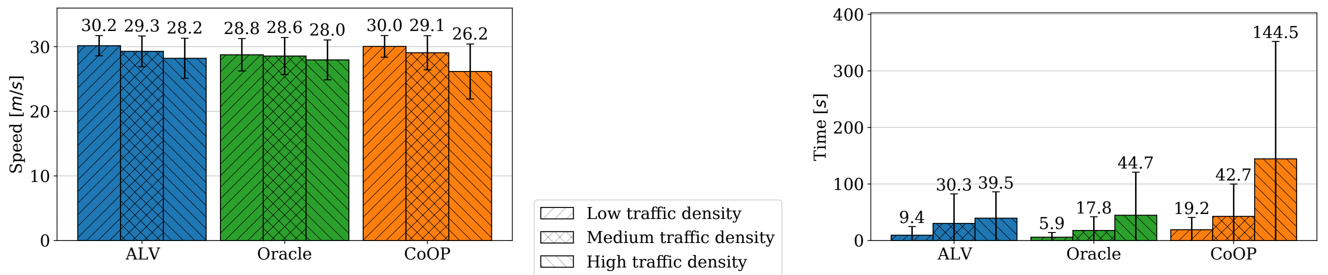


(a) Average speed with standard deviation.

(b) Average lane change time with standard deviation.

(c) Average lateral position with standard deviation. The dashed black line indicates the lane marking between lane 1 and lane 2.

Fig. 8. Platoon driving performance for Medium traffic density (cf. Table II) and a platoon length of four vehicles (*benchmarking scenario*).



(a) Average speed with standard deviation.

(b) Average lane change time with standard deviation.

Fig. 9. Platoon driving performance for varying traffic densities and a platoon length of four vehicles.

overtaking is useful, it will always be possible and safe to overtake (we have no trucks on the second lane). CoOP begins its lane change the earliest: when the slower truck is detected by the leader's radar. Therefore, the platoon manages to lose no speed at all. ALV changes lanes later than the proposed algorithm, but it does not lose any speed either. The Oracle case changes lanes even later than ALV, resulting in a speed drop from 30.6 m/s down to 24.8 m/s before overtaking. This explains why the Oracle case does not represent an optimal case for overtaking.

C. Parameter Study

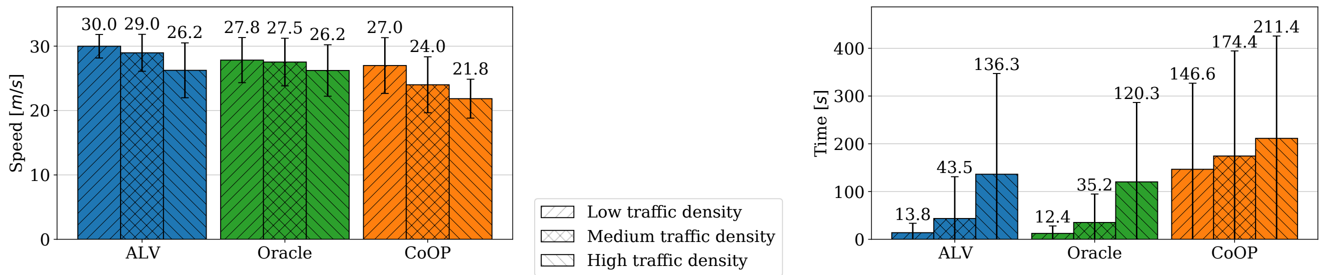
In a next step, we analyze the influence of certain parameters on the different platoon overtaking algorithms.

1) Traffic Density: We first study the impact of the traffic density (see Table II). In principle, low traffic density increases the platoon's average speed for all algorithms, while high traffic density decreases it (see Figure 9a). This was to be expected because it becomes more easy or more difficult to find a gap in the overtaking lane. In low traffic density, CoOP is only 1 % slower on average than ALV, because they drive in almost the same lateral position (1.7 m vs. 1.8 m). Oracle, as the slowest algorithm in low traffic, now drives with an average lateral position of only 1.3 m mostly in the slower original lane, resulting in the slowest speed. ALV still achieves the highest speed in all traffic densities, while Oracle is the most robust algorithm with a speed delta of only 0.8 m/s between low and high traffic density. CoOP handles high traffic density not as good as the other cases and Oracle became faster (+1.8 m/s). It becomes more difficult to find a gap in the overtaking lane for all algorithms, but, in addition, the delay-impaired communication makes it even more difficult: If there exists a gap, but an intra-platoon message is delayed until the gap has closed again, overtaking is further delayed. This results in a high lane change time of 144.5 s (see Figure 9b) and the highest speed delta between low and high traffic density (3.8 m/s).

2) Platoon Length: We now analyze the case's performance for different platoon lengths. Switching from four to (exemplary) eight car platoons, the platoon decreases the average speed in all scenarios for all cases because of the greater difficulty to find a gap in the overtaking lane. But the effects vary in strength (see Figure 10a). While in low and medium traffic density the speed reduction compared to four vehicles is rather small for ALV (about 0.3 m/s) and Oracle (about 1.1 m/s), it is bigger for CoOP: The speed decreases 3.0 m/s in low traffic density and 5.1 m/s in medium traffic density. This is because of the decreased probability of receiving all necessary messages in time for an overtaking attempt due to the doubled size of the platoon. The speed is impacted more significantly for all cases in high traffic density. Oracle repeats its comparatively good performance and loses only 1.8 m/s compared to four vehicles, and ALV loses 2.0 m/s. CoOP looses the most speed (4.4 m/s), especially due to the delay-impaired V2V messaging system. Again, Oracle is the most robust regarding changes in traffic density.

V. Conclusion

In this paper, we proposed *CoOP*, a Vehicle-to-Vehicle (V2V)-based cooperative overtaking algorithm for platoons on freeways. Our CoOP concept enables safe overtaking maneuvers for entire platoons, i.e., without the need to disassemble and reassemble the platoon for the overtaking task. CoOP makes use of V2V communication – but does



(a) Average speed with standard deviation.

(b) Average lane change time with standard deviation.

Fig. 10. Platoon driving performance for varying traffic densities and a platoon length of eight vehicles.

not limit itself to a specific technology. It has only few technical requirements besides standard Cooperative Adaptive Cruise Control (CACC) capabilities and does not require communication with road users beyond the platoon members. We validated safety and robustness of CoOP in a wide variety of simulated traffic scenarios, ensuring appropriate behavior in unsafe situations. Our performance evaluation shows that CoOP competes well with the theoretic optimum (i.e., the Artificial Long Vehicle (ALV) case) and the Oracle case. Cooperative driving using CoOP is only slightly slower compared to the ALV case and outperforms the Oracle case in some scenarios, even though it only relies on local information collected by means of direct, delay-impaired V2V communication.

Despite the very good results, there is still room for improvement in future work. One of the main goals of the proposed algorithm is to overtake cooperatively without splitting up the platoon into individual vehicles. In some cases, however, splitting up a long platoon could be useful behavior. Further research could analyze under which conditions this would be advisable.

References

[1] D. Jia, K. Lu, J. Wang, X. Zhang, and X. Shen, "A Survey on Platoon-Based Vehicular Cyber-Physical Systems," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 263–284, 2016.

[2] T. Sturm, C. Krupitzer, M. Segata, and C. Becker, "A Taxonomy of Optimization Factors for Platooning," *TITS*, Apr. 2020.

[3] M. Segata, B. Bloessl, S. Joerer, F. Dressler, and R. Lo Cigno, "Supporting Platooning Maneuvers through IVC: An Initial Protocol Analysis for the Join Maneuver," in *IEEE/IFIP WONS 2014*, Obergurgl, Austria: IEEE, Apr. 2014, pp. 130–137.

[4] S. Santini, A. Salvi, A. S. Valente, A. Pescapè, M. Segata, and R. Lo Cigno, "Platooning Maneuvers in Vehicular Networks: a Distributed and Consensus-Based Approach," *T-IV*, vol. 4, no. 1, pp. 59–72, Mar. 2019.

[5] P. Petrov and F. Nashashibi, "Modeling and Nonlinear Adaptive Control for Autonomous Vehicle Overtaking," *TITS*, vol. 15, no. 4, pp. 1643–1656, Aug. 2014.

[6] R. Toledo-Moreo, J. Santa, and M. A. Zamora-Izquierdo, "A Cooperative Overtaking Assistance System," in *IEEE/RSJ IROS 2009, PPNIV Workshop*, St. Louis, MO: IEEE, Oct. 2009, pp. 50–55.

[7] O. Sawade and I. Radusch, "Survey and Classification of Cooperative Automated Driver Assistance Systems," in *IEEE VTC 2015-Fall*, Boston, MA: IEEE, Sep. 2015, pp. 1–5.

[8] J. C. Mertens, J. Hauenstein, F. Diermeyer, L. Jahn, and S. Kraus, "Cooperative Truck Overtaking on Freeways," in *EVER 2020*, Monte-Carlo, Monaco: IEEE, Sep. 2020, pp. 1–15.

[9] M. Düring, K. Franke, R. Balaghiasefi, M. Gonter, M. Belkner, and K. Lemmer, "Adaptive Cooperative Maneuver Planning Algorithm for Conflict Resolution in Diverse Traffic Situations," in *ICCVE 2014*, Vienna, Austria: IEEE, Nov. 2014, pp. 242–249.

[10] A. Marefat, R. Aboki, A. Jalooli, E. Shaghaghi, M. R. Jabbarpour, and R. M. Noor, "An adaptive overtaking maneuver assistant system using VANET," in *IEEE APWiMob 2014*, Bali, Indonesia: IEEE, Aug. 2014, pp. 316–321.

[11] J. Chen, K. Wang, H. Bao, and T. Chen, "A Design of Cooperative Overtaking Based on Complex Lane Detection and Collision Risk Estimation," *IEEE Access*, vol. 7, pp. 87 951–87 959, 2019.

[12] J. Gong, Y. Xu, C. Lu, and G. Xiong, "Decision-making model of overtaking behavior for automated driving on freeways," in *IEEE ICVES 2016*, Beijing, China: IEEE, Jul. 2016, pp. 7–12.

[13] J. E. Naranjo, C. Gonzalez, R. Garcia, and T. de Pedro, "Lane-Change Fuzzy Control in Autonomous Vehicles for the Overtaking Maneuver," *TITS*, vol. 9, no. 3, pp. 438–450, Sep. 2008.

[14] J. Nilsson and J. Sjöberg, "Strategic decision making for automated driving on two-lane, one way roads using model predictive control," in *IEEE IV 2013*, Gold Coast, Australia: IEEE, Jun. 2013, pp. 1253–1258.

[15] C.-J. Hoel, K. Wolff, and L. Laine, "Automated Speed and Lane Change Decision Making using Deep Reinforcement Learning," in *IEEE ITSC 2018*, Maui, HI: IEEE, Nov. 2018, pp. 2148–2155.

[16] S. Ulbrich and M. Maurer, "Probabilistic online POMDP decision making for lane changes in fully automated driving," in *IEEE ITSC 2013*, The Hague, Netherlands: IEEE, Oct. 2013, pp. 2063–2067.

[17] ——, "Situation Assessment in Tactical Lane Change Behavior Planning for Automated Vehicles," in *IEEE ITSC 2015*, Las Palmas de Gran Canaria, Spain: IEEE, Sep. 2015, pp. 975–981.

[18] S. Samiee, S. Azadi, R. Kazemi, and A. Eichberger, "Towards a Decision-Making Algorithm for Automatic Lane Change Manoeuvre Considering Traffic Dynamics," *Promet - Traffic & Transportation*, vol. 28, no. 2, pp. 91–103, Apr. 2016.

[19] H. C.-H. Hsu and A. Liu, "Kinematic Design for Platoon-Lane-Change Maneuvers," *TITS*, vol. 9, no. 1, pp. 185–190, Mar. 2008.

[20] M. Segata, S. Joerer, B. Bloessl, C. Sommer, F. Dressler, and R. Lo Cigno, "PLEXE: A Platooning Extension for Veins," in *IEEE VNC 2014*, Paderborn, Germany: IEEE, Dec. 2014, pp. 53–60.

[21] E. Donges, "A Conceptual Framework for Active Safety in Road Traffic," *VESD*, vol. 32, no. 2-3, pp. 113–128, 1999.

[22] R. Schubert, K. Schulze, and G. Wanielik, "Situation Assessment for Automatic Lane-Change Maneuvers," *TITS*, vol. 11, no. 3, pp. 607–616, Sep. 2010.

[23] J. Heinovski and F. Dressler, "Platoon Formation: Optimized Car to Platoon Assignment Strategies and Protocols," in *IEEE VNC 2018*, Taipei, Taiwan: IEEE, Dec. 2018.

[24] C. Sommer and F. Dressler, *Vehicular Networking*. Cambridge University Press, 2014.

[25] S. Lam and J. Katupitiya, "Cooperative Autonomous Platoon Maneuvers on Highways," in *IEEE/ASME AIM 2013*, Wollogong, Australia: IEEE, Jul. 2013, pp. 1152–1157.

[26] S. Ulbrich and M. Maurer, "Towards Tactical Lane Change Behavior Planning for Automated Vehicles," in *IEEE ITSC 2015*, Las Palmas de Gran Canaria, Spain: IEEE, Sep. 2015, pp. 989–995.

[27] P. Alvarez Lopez, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flötteröd, R. Hilbrich, L. Lücken, J. Rummel, P. Wagner, and E. Wießner, "Microscopic Traffic Simulation using SUMO," in *IEEE ITSC 2018*, Maui, HI: IEEE, Nov. 2018, pp. 2575–2582.

[28] Y. Yao, Y. Hu, G. Yang, and X. Zhou, "On MAC Access Delay Distribution for IEEE 802.11p Broadcast in Vehicular Networks," *IEEE Access*, vol. 7, pp. 149 052–149 067, Jan. 2019.

[29] M. Semrau and J. Erdmann, "Simulation framework for testing ADAS in Chinese traffic situations," in *SUMO 2016*, vol. 30, Berlin, Germany: DLR, May 2016, pp. 103–115.