Platoon Formation: Optimized Car to Platoon Assignment Strategies and Protocols

Julian Heinovski and Falko Dressler
Heinz Nixdorf Institute and Dept. of Computer Science, Paderborn University, Germany
{heinovski,dressler}@ccs-labs.org

***Abstract*—We study the problem of platoon formation, trying to optimize traveling time and fuel consumption based on car-to-platoon assignments. The general concept of platooning, i.e., cars traveling in form of a road train with minimized safety gaps, has been studied in depth and we see first field trials on the road. Currently, most research focuses on improved reliability of the necessary communication protocols to achieve perfect string stability with guaranteed safety measures. One aspect, however, remained quite unexplored: the problem of assigning cars to platoons. Based on the capabilities of individual cars (e.g., max. acceleration or speed) and preferences of the driver (e.g., min/max. traveling speed, preference on travel time vs. fuel consumption), the assignment decision will be different. We formulate an optimization problem and develop a set of protocols (centralized and distributed) to support platoon formation. In an extensive series of simulation experiments, we show that our protocols not just help forming platoons, but also take care of the individual requirements of cars and drivers.**

I. Introduction

Road traffic has been growing constantly during the last years. For example, passenger transport increased by 8 % from 2005 to 2015 in Europe and the individual car is still the major transportation system with a share of more than 71 % (2015) [1] in comparison to public transport. Having this many vehicles on the road leads to issues like environmental pollution (due to increased emissions) and congestion on the roads.

In order to cope with the continuously growing traffic needs, the concept of *platooning* has been developed [2, 3]. In platooning, multiple vehicles form a *road-train* and drive with a very small safety gap between each other to increase the road utilization. This small gap can be maintained by driving autonomously using Cooperative Adaptive Cruise Control (CACC), which combines data from local sensors (e.g., the distance to the previous vehicle measured by radar) and information from other vehicles via Inter-Vehicle Communication (IVC) [4]. Besides a better utilization of the road, platooning also brings other benefits such as a reduced air drag, thus, reducing the fuel consumption [5]. Furthermore, smoother speed changes by the autonomous driving system lead to improved traffic flows and increase the driving comfort [6].

The concept of platooning has been investigated in depth in the literature and in field trials on the road. Well known projects in the field are PATH [2] and SARTRE [7], both of which demonstrated the technical feasibility of stable platooning on the road – certainly limited to a few cars. Ongoing research mainly focuses on maintaining existing platoons, improving reliability of the necessary IVC protocols in order to achieve perfect string stability with guaranteed safety measures [8–10]. Many studies either consider pre-configured platoons or just do ad-hoc formation, i.e., a vehicle is joining the closest one in front using Adaptive Cruise Control (ACC). Therefore, platoon formation has either been artificial or only very basic, certainly not being very realistic.

However, the typical situation on a freeway will be different. Individual cars are entering a freeway and drive on their own until they find an appropriate platoon (or another individual car) to team up with in a platoon. Therefore, solving the challenge of *platoon formation* or, more specifically, selecting candidate vehicles, is the next important step towards platooning. Once candidate vehicles are selected by some formation strategy, the cars should perform maneuvers to join an existing or form a new platoon [11, 12].

For platoon formation, a car has to start searching for candidates either immediately when entering the freeway, or at a later point in time. If there is a platoon, the car may become part of it and the search is over; otherwise, a new platoon has to be formed. In any case, a dynamic formation process is necessary, consisting of, first, finding candidate platoons/vehicles and, secondly, forming the platoon via a join maneuver.

In this paper, we first study this problem analytically before presenting both a centralized and a distributed approach for platoon formation. The centralized approach uses global knowledge about all cars in the scenario to make assignment decisions, while running on a central entity in the network. In the distributed approach, the algorithm is running on every car in the scenario, therefore, having only limited local knowledge about other cars. We compare both approaches for the same strategy to study their respective advantages and weaknesses in an extensive set of simulation experiments.

Our main contributions can be summarized as follows:

- We provide an in-depth study of platoon formation challenges and analytically explore the problem,
- we develop both a centralized and a distributed strategy and the respective communication protocols, and
- we perform an extensive performance evaluation of both strategies and discuss the results.

II. Related Work

Early platoon formation solutions can be grouped into several classes. In one of the early papers, Hall and Chin [13] propose offline formation strategies, which sort vehicles into platoons

978-1-5386-9428-2/18/$31.00 ©2018 IEEE

at the entrance ramp of a highway. Vehicles are grouped according to their destination and enter the highway in a platoon formation when the group consists of enough vehicles. Their main optimization goal is to maximize the platoon size and the time a platoon stays intact. As a second criterion, the destination is used to make sure that platoons can last as long as possible. Following a similar line of thought, other approaches also sort vehicles on the entrance ramp to minimize the total trip time by optimal speed limits and entrance ramp release times [14]. Other concepts looked into optimizing the total fuel consumption for all transport assignments in the scenario, while taking into account fuel savings due to platooning as well as speed changes [15]. The complexity of such centralized optimization has been shown to be NP hard [16].

In contrast to centralized platoon formation, there also have been studies considering distributed approaches. In a very early study, Khan and Bölöni [17] developed a system for ad-hoc convoy formation on freeways. The system continuously evaluates the cost and the possible benefit of forming a platoon with other vehicles in proximity; if successful, it indicates the decision to the driver using an LED to adjust the ACC accordingly. More recently, Liang et al. [18] studied fuel-efficient distributed ad-hoc platooning for Heavy Duty Vehicles (HDVs) by analyzing the optimization problem of pairwise coordination of vehicles. The proposed algorithm for coordination lets the leading vehicle slow down and the trailing vehicle speed up, in order to make the formation process fuel-efficient and keep delivery constraints.

Larson et al. [19] deploy a distributed network of virtual controllers at junctions in the road network. The controllers monitor HDVs approaching these junctions, in order to form platoons with other vehicles in proximity. Using information such as speed, position, and the destination of a vehicle, the controller calculates the cost of adjusting the speed to form a platoon with another vehicle for all approaching HDVs and the corresponding possible fuel savings by doing so. Simulation on the German *Autobahn* road network showed that only minor speed adjustments are necessary for a HDV to form a useful a platoon with other vehicles.

The concept presented by Caballeros Morales et al. [20] is closest to our solution. A distributed clustering algorithm using IVC groups cars according to their destination, speed, and position. The algorithm is executed by every car and forms groups with other vehicles by minimizing their respective deviation, in order to increase lifetime of clusters among the mobility pattern of vehicles. Simulations show that their algorithm performs well in terms of cluster lifetime, cluster-head changes, and the number of cluster re-affiliations.

The aforementioned strategies for platoon formation show that optimal groupings substantially improve the performance gain. Unfortunately, the optimization objectives are quite different, making a comparison becomes infeasible, so that a detailed comparison of centralized and distributed solutions is still missing in the literature. Furthermore, only limited optimization parameters were chosen together with a restrictive set of performance metrics. In this paper, we go one step further and, besides formally describing the platoon formation problem, we introduce both a centralized and a distributed heuristic. We compare both solutions in detail using a wide range of performance metrics.

III. Platoon Formation

A. Problem Formulation

We are using the desired driving speed as a primary similarity metric. However, since is it not useful to join a platoon far away, we also consider the position of the cars as a secondary optimization metric. In order to come up with a formation strategy, we formalize the problem as follows: Let a car be represented by the set

$$\{id, des, pos\} \ , \tag{1}$$

where id is the identifier of the car, des is the desired speed of the car, and pos is the current position of the car.

We can now consider platoon formation as the following optimization problem:

$$\forall i : \text{minimize } f_i\left(x\right), \quad \forall x \in \Omega_i \ , \tag{2}$$

where Ω_i is the neighborhood of car i (i.e., all cars x, which are in (close) proximity of car i) and

$$f_i\left(x\right) = \alpha \cdot d_s\left(x, i\right) + \beta \cdot d_p\left(x, i\right) \ , \tag{3}$$

determines the cost for car i to join car x, in order to form a platoon; with

$$d_s\left(x, i\right) = \left\| des_i - des_x \right\| \ , \tag{4}$$

$$d_p\left(x, i\right) = \begin{cases} \left\| pos_i - pos_x \right\| & \text{if } pos_x > pos_i \\ \infty & \text{if } pos_x \leq pos_i \end{cases} \ , \tag{5}$$

$$\alpha, \beta \in [0, 1], \quad \alpha + \beta = 1 \ , \tag{6}$$

and subject to the following constraints:

$$d_s\left(x, i\right) \leq p \cdot des_i, \quad p \in [0, 1] \ , \tag{7}$$

$$d_p\left(x, i\right) \leq r \ . \tag{8}$$

In summary, we try to find the best fitting platoon candidate x for each car i, maximizing their similarity. It is important to mention that the definition from Equation (5) only allows joining at the end of a vehicle or platoon.

As an example, consider the scenario depicted in Figure 1, where four cars are driving on an arbitrary road with two lanes (e.g., a freeway) and now try to find a platoon. The cars in the example are defined by their set

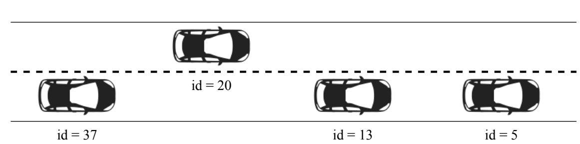

Figure 1. Example scenario: Four cars are driving on a road and try to find a platoon

of properties, $\{5, 121\,\mathrm{km/h}, 430\,\mathrm{m}\}$, $\{13, 89\,\mathrm{km/h}, 270\,\mathrm{m}\}$, $\{20, 107\,\mathrm{km/h}, 250\,\mathrm{m}\}$, $\{37, 93\,\mathrm{km/h}, 70\,\mathrm{m}\}$, and parameters

$$\alpha = 0.6, p = 0.4, r = 400\,\mathrm{m}.$$

By using these properties and parameters, the list of possible platoon candidates and their corresponding cost $f_i(x)$ can be calculated as

$$f_{13}(5) = 0.6 \cdot 32 + 0.4 \cdot 160 = 83.2$$

$$f_{20}(5) = 0.6 \cdot 14 + 0.4 \cdot 180 = 80.4$$
$$f_{20}(13) = 0.6 \cdot 18 + 0.4 \cdot 20 = 18.8$$

$$f_{37}(5) = 0.6 \cdot 28 + 0.4 \cdot 360 = 160.8$$
$$f_{37}(13) = 0.6 \cdot 4 + 0.4 \cdot 100 = 42.4$$
$$f_{37}(20) = 0.6 \cdot 14 + 0.4 \cdot 180 = 80.4\ .$$

From the list of possible candidates and their corresponding costs, the optimal solution minimizing the overall cost is

$$f_{37}(13) = 42.4$$
$$f_{20}(5) = 80.4\ ,$$

as selecting a candidate pair blocks both involved cars, making them unavailable for further selection.

Since a car can only be in one maneuver at the same time, at most two maneuvers can be ongoing in parallel. After these maneuvers are finished, the cars in the scenario will be grouped into two platoons: $\{13, 37\}$ and $\{5, 20\}$.

In order to solve this optimization problem, a mathematical solver is necessary. However, due to computational and time constraints, we use a heuristic to select feasible candidates which follows a greedy approach: We calculate the cost $f_i(x)$ for all cars in the neighborhood which do not violate the constraints given by Equations (7) and (8) and add an entry for them to a list of possible matches. Then, we select the candidate x with the smallest cost (i.e., deviation in speed and position) from this list and let the searching car i join this selected candidate x.

If the join maneuver is successful, car i afterwards is part of a platoon with car x (which was just formed or x was already part of). Once cars become platoon members, they do not leave the platoon until they reach their destination. In this study, every car has the same destination. Therefore, the whole platoon sticks together until this destination is reached.

B. Centralized Approach

In our centralized approach, the optimization problem is solved for every car in the scenario at the same time. Since the central server has global knowledge about all cars and their corresponding information, it can use this information to make decisions about platoon assignments. We assume that this global knowledge is collected by means of an infrastructure based network such as LTE.

We use the heuristic given in Algorithm 1 to create the list of possible matches. An entry $\{id_i, id_x, f_i(x)\}$ in this list contains cars i and x and the cost for letting car i join car x.

Algorithm 1 Centralized heuristic for finding candidate pairs

Input: meta info of all cars in the scenario
  for all cars i in the scenario **do**
    if i **not** in platoon **and** i **not** in maneuver **then**
      for all cars x in the scenario with $x \neq i$ **do**
        if x in platoon **and** x **not** leader **or** x in maneuver **then**
          next;
        if $d_s(x, i) > p \cdot des_i$ **or** $d_p(x, i) > r$ **then**
          next;
        add $\{i, x, f_i(x)\}$ to list of possible matches
Output: list of possible matches $\mathrm{list}(\{i, x, f_i(x)\})$

Algorithm 2 Centralized heuristic for best candidate selection

Input: list of possible matches $\mathrm{list}(\{i, x, f_i(x)\})$
  for all unique cars i in the list of possible matches **do**
    $m \leftarrow x \in \mathrm{list}(\{i, x, f_i(x)\})$;
    if $\|m\| > 0$ **then**
      $b \leftarrow \{x \mid \min f_i(x), x \in m\}$ ▷ *Select best candidate* x
      remove all entries containing cars i and x
      let i join b
Output: pairs of cars to perform join maneuver

Note that this is not symmetric as the cost from car i to x might not be the same as from car x to i.

Once all possible matches and their costs are computed, we use Algorithm 2 to select the best match for every searching car i to let it join a candidate car x. In particular, we select the match with the smallest deviation $f_i(x)$ and remove all entries which contain cars i and x. This heuristic is greedy from the perspective of a searching car i as it denies other searching cars to join the same car x later in the process. Due to its nested for-loop, the computational complexity of this approach is $O(n^2)$.

Re-considering the example from Figure 1, the centralized heuristic selects the following matches out of all possible ones:

$$f_{13}(5) = 83.2$$
$$f_{37}(20) = 80.4$$

After selecting car 13 to join car 5, both cars 13 and 5 are blocked, thus, leaving no match for car 20. Car 37 also cannot join car 13, hence the heuristic selects car 37 to join car 20. Although this approach also produces two platoons after successful finishing of the join maneuver, it does not compute the aforementioned optimal solution. However, as we will show in the evaluation, the heuristic performs quite well for the global scenario.

C. Distributed Approach

In our distributed approach, every car i has to execute the aforementioned greedy heuristic individually. In order to run any kind of selection algorithm, cars first of all have to become aware of other cars in their neighborhood. Therefore, all cars

Algorithm 3 Distributed heuristic for finding candidate pairs

Input: neighbor table storing the information of neighboring cars x for a fixed car i
  for all cars x in the neighbor table **do**
    if $d_s(x,i) > p \cdot des_i$ **or** $d_p(x,i) > r$ **then**
      next;
    add $\{i, x, f_i(x)\}$ to list of possible matches;
Output: list of possible matches list $(\{i, x, f_i(x)\})$

are transmitting their meta information via periodic beacons using IVC protocols such as IEEE 802.11p and maintain this data in a local neighbor table.

Using the entries in the neighbor table, the heuristic given in Algorithm 3 is executed to prepare the list of possible matches. Then, a heuristic very similar to Algorithm 2 is used to select a candidate car x with the smallest cost to join. The computational complexity of this approach is $O(n)$.

Conceptually, the same matches as in the centralized approach are selected. However, the selection of possible matches is limited to the restricted nature of the neighbor table and, therefore, depends on the time the heuristic is evaluated. Also, the quality of the heuristic now depends on the quality of the neighbor information, which depends on the used beacon protocol [21].

D. Model Implementation

We implemented all algorithms in the simulation tool Plexe [22]. Plexe can simulate platoons, utilizing SUMO [23] for simulation of road traffic and Veins [24] for simulation of realistic wireless communication and thus provides all relevant functionality for maintaining platoons.

We implemented the centralized approach in form of a global module in the scenario, that directly accesses the cars' information (e.g., speed and position). Based on this information, it runs the heuristics described by Algorithms 1 and 2 and computes join tasks, which are assigned to the involved cars. The cars then start a join maneuver with their corresponding platoon candidates. For the distributed approach, the heuristics described by Algorithms 2 and 3 are implemented in the application layer module of every car.

Cars send platoon advertisements via wireless beacons, including information about themselves as well as the platoon they are part of. This information is stored and maintained in a 1-hop neighbor table, which is used by the heuristic. Additionally, cars periodically broadcast their speed and position in cooperative awareness messages, later used for platoon maintenance. Due to the transmission range, cars conceptually only have local knowledge about the scenario, i.e., about cars in wireless transmission range.

After the heuristic selects a candidate, the car tries to join this candidate by executing a join maneuver, using control messages via wireless communication as well. This join maneuver is performed by Plexe, which we extended to support dynamic joining to arbitrary vehicles. We will release all implemented modules and updates to Plexe as Open Source.

Table I
SIMULATION PARAMETERS FOR MOBILITY AND ROAD NETWORK

Parameter	Value
Freeway length	30 km
Number of lanes	4
Spawn position of vehicles	First entry ramp
Destination	End of the freeway
Max acceleration	2.5 m/s^2
Max deceleration	9.0 m/s^2
Vehicle length	4 m
Car Following (CF) model	ACC and CACC
Desired speed v_d	$U(80, 130)$ km/h
Min speed v_{min}	0 km/h
Max speed v_{max}	140 km/h
Driver imperfection σ	0.5
Driver's desired minimum headway τ	0.5 s
ACC headway T	1.2 s
CACC desired gap d_d	5 m
CACC bandwidth ω_n	0.2 Hz
CACC damping ratio ξ	1
CACC weighting factor C_1	0.5
Arrival traffic	$B(1, 0.5) \rightarrow$ 2000 veh/h
SUMO update interval	0.1 s
ACC headway for approaching T_{join}	$\frac{1}{2} \cdot T = 0.6$ s
Response timeout	5 s
Lane Change (LC) timeout	20 s
Approach timeout	60 s
Maneuver timeout (leader)	$20\,\text{s} + 60\,\text{s} + 5\,\text{s} = 85\,\text{s}$
CACC switch threshold	$1.5 \cdot T_{\text{join}} \cdot v$

IV. EVALUATION

We evaluated and compared both the centralized and the distributed algorithm in an extensive set of simulations using Plexe. Additionally, we added a baseline scenario without platoon formation. We picked several metrics, some of which (e.g., travel time [13] and fuel consumption [19]) are also used in other studies, to understand the impact of platoon formation as such and to show the differences between the centralized and the distributed algorithm. In general, we assume platoon control as stable [25] and do not further investigate CACC properties such as string stability.

A. Simulation Setup

We use a freeway scenario for our simulation as shown in Figure 2. The freeway has a length of 30 km and contains four lanes. It has additional entry and exit lanes connected to a road with one lane, which is used as spawn point for vehicles. In the simulation, cars only spawn at the first entry and drive to

Figure 2. Screenshot of the simulation scenario. The red car is approaching the entrance ramp of the freeway while two other cars are performing a join maneuver (the green car is joining the purple one).

Table II
SIMULATION CONTROL PARAMETERS AND VALUES FOR PARAMETER STUDY

Parameter	Value
Simulation time	2700 s
Warm-up period	1350 s
Repetitions	10
Max. concurrent vehicles	500
deviation: Deviation from desired speed	0.1 to 0.3, step 0.1
range: Deviation in position	200 m to 1000 m, step 400 m
alpha: Weight of speed deviation	0.0 to 1.0, step 0.2

the end of the freeway (i.e., a trip of 30 km). The most relevant mobility parameters are summarized in Table I.

As soon as a car reaches the entrance ramp and merges onto the freeway, it starts advertising itself as a possible platoon candidate and begins searching for existing platoons and other cars to form a platoon with. Additionally, cars periodically broadcast their speed and position in cooperative awareness messages, later used for platoon maintenance. We use IEEE 802.11p for both the join maneuver and the neighborhood management. After the heuristics described in Section III selected a candidate, the car tries to join by executing the join maneuver.

Table II lists simulation control parameters, we used for the simulation. We ran our simulation for 2700 s, which is twice the minimum time a car driving the slowest desired speed (i.e., 80 km/h) needs to reach the end of the freeway. We use the first half of this simulation time as a warm-up period and ignore all results in this interval. Table II also lists the different values we used for studying the tuning parameters of the heuristics.

B. Number of Platoon Candidates

The found candidates metric counts the number of possible candidates for platoon formation as identified by the Algorithms 1 and 3 for a single car. The higher the value, the more similar cars are known to the respective algorithm and the more cars can be used to identify the one with highest similarity (i.e., lowest cost).

The average number of candidates found by the approaches for every car is shown in Figure 3. As can be seen, in contrast to our initial expectation, the centralized approach finds less possible candidates per car than the distributed approach (the median is only 0.8 in comparison to 1.8). Moreover, when considering the 95th percentile where 2.7 candidates in comparison to 4.7 were found.

To explain the contrary effect of a higher number of candidates for the distributed approach, we have to look at the cars which are filtered and therefore not considered as candidates. Cars technically not violating the constraints of the optimization problem defined by Equations (7) and (8) are filtered, if they are already in a maneuver, thus, not being applicable for another one. The distributed approach does not have the corresponding knowledge and, thus, cannot filter candidates in the aforementioned sense.

This shows that the centralized approach indeed finds more platooning opportunities in general because from the perspective of a single car, it is aware of more (i.e., all) other cars. Nevertheless, many of those candidates are filtered because they are already involved in maneuver.

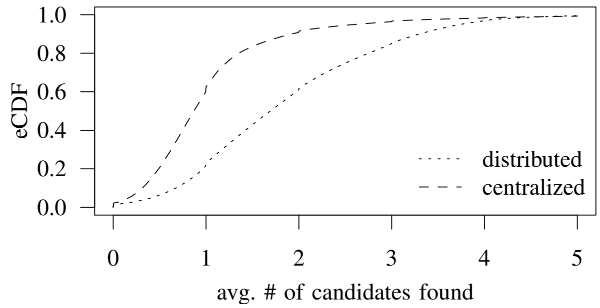

Figure 3. eCDF showing the average number of candidates found by the platoon formation strategy in each iteration per car

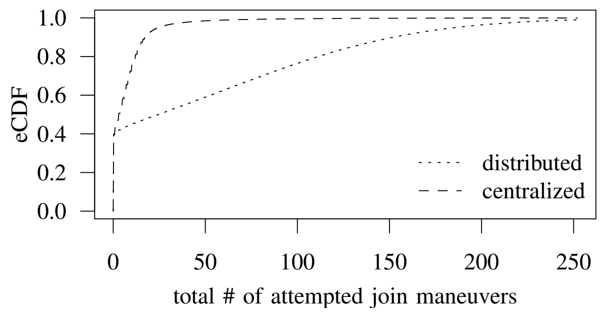

Figure 4. eCDF showing the total number of attempted joins per car

C. Join Maneuver

The number of attempted joins helps understanding the success of the join maneuvers. Whenever a candidate is selected by one of the heuristics, the searching car sends a message to the candidate to request the start of the join maneuver. Independent of the outcome of this message, that is whether it is positive, negative, or no response at all is received, it is counted as an attempted join maneuver.

Once a platoon assignment for a car is created (i.e., a candidate to join has been selected), the car attempts a join maneuver with the selected candidate by sending a join request. Figure 4 shows the total number of such attempted join maneuvers per car. It is evident that the effect of more found candidates has a direct impact on the number of attempted join maneuvers. Trying to join candidates which are not applicable anymore because they are already in a maneuver, leads to a much higher number of total attempted join maneuvers per car in the distributed approach. Interestingly, in both approaches almost 40 % of the cars never get a single platooning opportunity and, thus, do not attempt a join maneuver at all.

When looking at the high numbers of up to 500 (and more) attempted join maneuvers per vehicle in the simulation, and the fact that vehicles cannot change or leave a platoon, once they become a member, it is clear that many maneuvers do not succeed and are aborted. Most importantly, the join request could be aborted, particularly because the car is already in

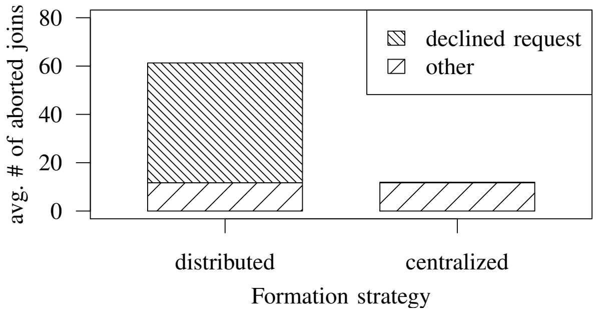

Figure 5. Bar plot showing the average number of aborted joins of per car

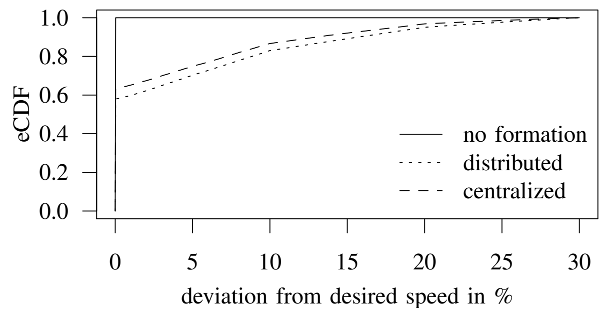

Figure 7. eCDF showing the relative deviation from a car's desired speed when reaching the destination

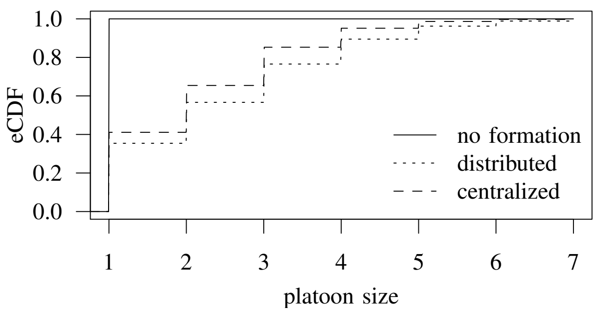

Figure 6. eCDF showing a car's platoon size when reaching its destination

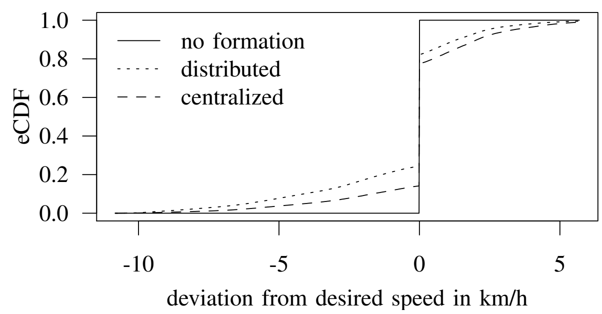

Figure 8. eCDF showing the absolute deviation from a car's desired speed when reaching its destination

a join maneuver with another car. Figure 5, therefore, shows the average number of aborted join maneuvers per car and different abort causes; causes other than being declined are grouped in *other*. Aborted maneuvers occur in both approaches, being caused by various reasons, such as missed timeouts and communication failures. In the distributed approach, we see that the majority of aborts is caused by declines of the join request by the leading vehicle. This is because the distributed approach always selects the candidate with the smallest cost, independent from many times the join already failed, and retries to join that candidate until the maneuver is completed successfully.

D. Platoon Size

We use the number of cars in a platoon to describe the ratio of successful platoon formations. If a car does not find a feasible platoon candidate and, therefore, is not able to become a platoon member, it will not be in a platoon when reaching the destination. Since a platoon stays intact once it has formed, and it only can get more members over time, we consider this value at the end of the scenario. The results are shown in Figure 6.

As expected, the baseline shows that all cars arrive as individuals. About 41 % in the centralized and 35 % in the distributed approach, respectively, have not joined a platoon at the end of the scenario. This is either due to not getting an opportunity or due to not finishing the join process. In both approaches it may take some time until cars are in a platoon. The centralized approach leads to smaller platoons (on average, 2.14 cars per platoon), whereas the distributed solution tends to form larger platoons (on average 2.47 cars per platoon). This is due to the fact that the distributed algorithm does not pick a different candidate if the best one is blocked. Therefore, multiple cars eventually join the same platoon, thus, leading to longer platoons.

E. Deviation from Desired Speed

The main optimization goal in our strategy is the desired speed, thus, the deviation from it is of particular interest. The smaller the deviation from the desired speed, the better. The results are shown in Figure 7.

The deviation of desired speed is a constraint for considering cars as possible candidates, being simulated at discrete values of 10, 20 and 30 %. Thus, the maximum is at 30 %. As more cars are in a platoon in the distributed approach, more cars deviate from their desired speed to form the corresponding platoon (42 % in comparison to 37 % in the centralized case).

When considering the absolute deviation as well, as shown in Figure 8, an additional effect can be seen. Not only do more cars deviate and to a bigger extent in the distributed approach, they also tend to deviate more negatively, hence, decreasing their initial speed. On the 1st percentile, cars have to decelerate by −9.2 km/h in the distributed and by −7.8 km/h in the centralized approach, respectively. This is in contrast to the 99th percentile, where cars have to accelerate only by 4.75 km/h and by 5.69 km/h, respectively. On average, cars have to slow down by −0.6 km/h in the distributed case compared to speeding up by 0.01 km/h in the centralized approach.

F. Travel Time

Looking at the travel time, the effects of the deviation from the desired speed can be seen. When merging onto the freeway, every car estimates the time it is going to travel to its destination, assuming a constant speed at the desired value. Upon arrival,

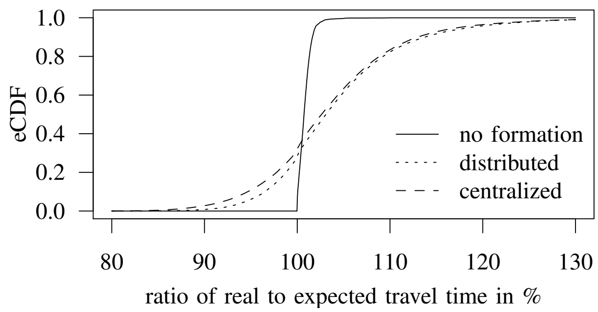

Figure 9. eCDF showing the ratio of expected to real travel time per a car. A value smaller than 100 % means that the car reaches its destination faster than expected, a value greater than 100 % means that the car is slower.

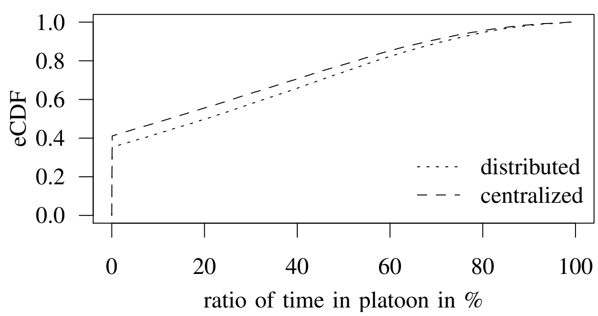

Figure 10. eCDF showing the platoon time ratio per car

cars also record their real travel time, calculating the travel time ratio. We use this metric to show the impact of platooning on the travel time. Cars have to make compromises when they want to do platooning with other vehicles. One of these is to drive at a speed different to their desired one, thus, influencing their travel time. In this study, the speed of a (new) platoon is not calculated as a consensus between all members, but is the speed of the platoon leader. Also, join maneuvers can have an impact on the travel time.

As shown in Figure 9, the baseline is almost always at 100 % and only deviates to slower speeds due to traffic. When platooning is enabled, speed deviations in both directions can be observed. During the join process, the car can be faster than its desired speed to close the gap to the platoon. During the trip, the speed can divert from the desired speed both positively as well as negatively. There is only a slight difference between the centralized and the distributed platoon formation approach visible. Due to the aforementioned calculation of a vehicle's new speed, vehicles in a platoon tend to be a bit slower than the desired speed. On average, the distributed case shows a deviation to 104.36 %, whereas the centralized approach leads to a deviation to 102.61 %.

In order to understand these effects in more detail, we also look at the platoon time ratio, i.e., the time a car spends in a platoon over the total travel time. The results of this ratio are shown in Figure 10. It is slightly larger for the distributed approach. Here, platoons are longer and cars stay in the platoon for a longer time, as they travel with slower speeds. On average, 28 % of the time is spent in a platoon in the distributed case compared to 24 % in the centralized case.

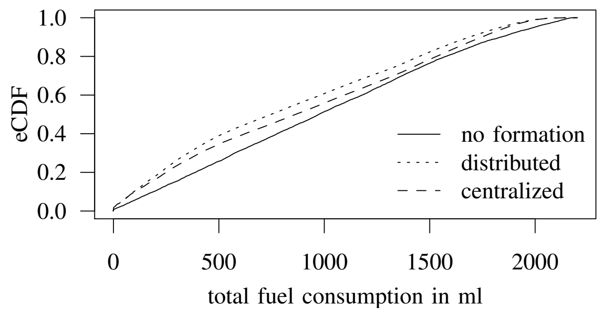

Figure 11. eCDF showing the total fuel consumption per car

G. Fuel Consumption

Since a major benefit of platooning is the reduced air drag due to the small gap between cars, and, thus, a lower fuel consumption, we consider this effect in our study as well. The consumption depends on the speed and also on the distance to a preceding car, since this has an influence on the air drag. In order to simulate this effect, we added a model to Plexe to calculate the fuel consumption dependent on the reduced air drag due to the small gap.

Sovran [26] define a correlation between the change of the fuel consumption $\tilde{g}$ and the change of the air drag C_D by using a factor η:

$$\frac{\Delta\tilde{g}}{\tilde{g}} = \eta \cdot \frac{\Delta C_D}{C_D} \text{ , where} \tag{9}$$

$$\Delta\tilde{g} = \tilde{g} - \tilde{g}_{\text{platoon}} \; . \tag{10}$$

In order to simplify Equation (9), we define $\delta = \frac{\Delta C_D}{C_D}$, so that the fuel consumption $\tilde{g}_{\text{platoon}}$ for a car in a platoon can be modeled as

$$\tilde{g}_{\text{platoon}} = (1 - \eta \cdot \delta) \cdot \tilde{g} \; . \tag{11}$$

Cappiello et al. [27] derive a model to calculate the fuel consumption of a car from measurements and model fitting. Thus, we define the normal fuel consumption for a car not in a platoon $\tilde{g}$ as

$$\tilde{g} = \tilde{g}_{\text{Cappiello}} \; . \tag{12}$$

We use the following values for δ from Bruneau et al. [28], Table 5: $\delta_{\text{Lead}} = 0.12$, $\delta_{\text{Middle}} = 0.27$, and $\delta_{\text{Last}} = 0.23$.

Using Equations (11) and (12), $\eta = 0.46$ [26], and the values for δ, the fuel consumption of a car in a platoon $\tilde{g}_{\text{platoon}}$ can be calculated as

$$\tilde{g}_{\text{platoon}} = (1 - \eta_{\text{Sovran}} \cdot \delta_{\text{Bruneau}}) \cdot \tilde{g}_{\text{Cappiello}} \; , \tag{13}$$

where δ_{Lead} is used for the leading car, δ_{Last} for the last car, and δ_{Middle} for every other car in the platoon.

The resulting fuel consumption is plotted in Figure 11. The values plotted represent the total fuel consumption of cars until reaching the destination. The absolute values might be partially misleading as the model gives negative values when the deceleration is too high; it assumes that values are capped by different thresholds [29]. The qualitative effects, however, are correct and we can thus study the relation of the two platoon formation approaches.

As expected, platooning indeed helps saving fuel compared to the baseline. However, the distributed solution outperforms the centralized approach. Even though not the optimal platoons may be formed, overall, there are more cars in platoons and for a longer time. Also, the driving speed is slower in the distributed approach. Both aspects help reducing the fuel consumption.

V. Discussion and Conclusion

In this paper, we investigated platoon formation as an optimization problem from the perspective of cars searching to join platoons. We developed both a centralized and a distributed approach using greedy heuristics to solve this optimization problem. We simulated both approaches and compared them using several metrics for platooning.

Comparing both approaches presented, we see that the centralized approach has more knowledge. It is aware of more vehicles and, thus, more candidates. However, many vehicles are filtered due to knowledge of their maneuver status. This has the advantage of fewer aborted join maneuvers. On the other hand, the shown data evidences that being greedy (i.e., trying to keep joining the same candidates) eventually pays off. The distributed solution leads to longer platoons, as several cars eventually join the same platoon.

Overall, both approaches need some time to find a platoon. Here, the distributed solution is slightly worse. It leads to more negative speed deviation and the deviation is larger in general as platoons are longer and more cars need to adjust the speed to the same leader. Fuel savings can be acknowledged for platooning in general. Certainly, more time in a platoon also leads to higher savings.

In future work, we will consider more sophisticated join maneuvers and merging of platoons. Also, we need to consider smarter approaches for the distributed solution compared to the busy wait whenever a join failed.

References

[1] "EU transport in figures - Statistical pocketbook 2017," European Commission, Luxembourg City, Luxembourg, Catalogue number MI-AA-16-002-EN-N, Oct. 2017.

[2] S. Shladover, C. Desoer, J. Hedrick, M. Tomizuka, J. Walrand, W.-B. Zhang, D. McMahon, H. Peng, S. Sheikholeslam, and N. McKeown, "Automated Vehicle Control Developments in the PATH Program," *IEEE Transactions on Vehicular Technology*, vol. 40, no. 1, pp. 114–130, Feb. 1991.

[3] C. Bergenhem, H. Pettersson, E. Coelingh, C. Englund, S. Shladover, and S. Tsugawa, "Overview of Platooning Systems," in *ITS World Congress*, Vienna, Austria, Oct. 2012.

[4] F. Dressler, F. Klingler, M. Segata, and R. Lo Cigno, "Cooperative Driving and the Tactile Internet," *Proceedings of the IEEE*, 2018.

[5] A. A. Alam, A. Gattami, and K. H. Johansson, "An Experimental Study on the Fuel Reduction Potential of Heavy Duty Vehicle Platooning," in *IEEE ITSC 2010*, Madeira Island, Portugal: IEEE, Sep. 2010, pp. 306–311.

[6] B. van Arem, C. van Driel, and R. Visser, "The Impact of Cooperative Adaptive Cruise Control on Traffic-Flow Characteristics," *IEEE Transactions on Intelligent Transportation Systems*, vol. 7, no. 4, pp. 429–436, Dec. 2006.

[7] T. Robinson, E. Chan, and E. Coelingh, "Operating Platoons On Public Motorways: An Introduction To The SARTRE Platooning Programme," in *ITS 2010*, Busan, Korea, Oct. 2010.

[8] X. Liu, A Goldsmith, S. Mahal, and J. Hedrick, "Effects of Communication Delay on String Stability in Vehicle Platoons," in *IEEE ITSC 2001*, Oakland, CA: IEEE, Aug. 2001, pp. 625–630.

[9] P. Fernandes and U. Nunes, "Platooning With IVC-Enabled Autonomous Vehicles: Strategies to Mitigate Communication Delays, Improve Safety and Traffic Flow," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 1, pp. 91–106, Mar. 2012.

[10] M. Segata, F. Dressler, and R. Lo Cigno, "Jerk Beaconing: A Dynamic Approach to Platooning," in *IEEE VNC 2015*, Kyoto, Japan: IEEE, Dec. 2015, pp. 135–142.

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

[12] 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.

[13] R. Hall and C. Chin, "Vehicle sorting for platoon formation: Impacts on highway entry and throughput," *Transportation Research Part C: Emerging Technologies*, vol. 13, no. 5, pp. 405 –420, Dec. 2005.

[14] L. D. Baskar, B. D. Schutter, and H. Hellendoorn, "Dynamic Speed Limits and On-Ramp Metering for IVHS using Model Predictive Control," in *IEEE ITSC 2008*, Beijing, China: IEEE, Oct. 2008, pp. 821–826.

[15] S. van de Hoef, K. H. Johansson, and D. V. Dimarogonas, "Fuel-Optimal Centralized Coordination of Truck Platooning Based on Shortest Paths," in *ACC 2015*, Chicago, IL: IEEE, Jul. 2015, pp. 3740–3745.

[16] E. Larsson, G. Sennton, and J. Larson, "The vehicle platooning problem: Computational complexity and heuristics," *Transportation Research Part C: Emerging Technologies*, vol. 60, pp. 258 –277, Nov. 2015.

[17] M. A. Khan and L. Bölöni, "Convoy driving through ad-hoc coalition formation," in *IEEE RTAS 2005*, San Francisco, CA: IEEE, Mar. 2005, pp. 98–105.

[18] K. Y. Liang, J. Mårtensson, and K. H. Johansson, "Heavy-Duty Vehicle Platoon Formation for Fuel Efficiency," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 4, pp. 1051–1061, Apr. 2016.

[19] J. Larson, K. Y. Liang, and K. H. Johansson, "A Distributed Framework for Coordinated Heavy-Duty Vehicle Platooning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 1, pp. 419–429, Feb. 2015.

[20] M. M. Caballeros Morales, C. S. Hong, and Y.-C. Bang, "An Adaptable Mobility-Aware Clustering Algorithm in vehicular networks," in *APNOMS 2011*, Taipei, Taiwan: IEEE, Sep. 2011, pp. 1–6.

[21] F. Dressler, F. Klingler, C. Sommer, and R. Cohen, "Not All VANET Broadcasts Are the Same: Context-Aware Class Based Broadcast," *IEEE/ACM Transactions on Networking*, vol. 26, no. 1, pp. 17–30, Feb. 2018.

[22] 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.

[23] D. Krajzewicz, G. Hertkorn, C. Rössel, and P. Wagner, "SUMO (Simulation of Urban MObility); An Open-source Traffic Simulation," in *MESM 2002*, Sharjah, UAE, Sep. 2002, pp. 183–187.

[24] C. Sommer, R. German, and F. Dressler, "Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis," *IEEE Transactions on Mobile Computing*, vol. 10, no. 1, pp. 3–15, Jan. 2011.

[25] R. Rajamani, H.-S. Tan, B. K. Law, and W.-B. Zhang, "Demonstration of Integrated Longitudinal and Lateral Control for the Operation of Automated Vehicles in Platoons," *IEEE Transactions on Control Systems Technology*, vol. 8, no. 4, pp. 695–708, Jul. 2000.

[26] G. Sovran, "Tractive-Energy-Based Formulae for the Impact of Aerodynamics on Fuel Economy Over the EPA Driving Schedules," SAE International, Technical Paper 830304, Feb. 1983.

[27] A. Cappiello, I. Chabini, E. Nam, A. Lue, and M. Abou Zeid, "A statistical model of vehicle emissions and fuel consumption," in *IEEE ITSC 2002*, Sep. 2002, pp. 801–809.

[28] C.-H. Bruneau, K. Khadra, and I. Mortazavi, "Flow analysis of square-back simplified vehicles in platoon," *International Journal of Heat and Fluid Flow*, vol. 66, pp. 43–59, Aug. 2017.

[29] M. A. Zeid, I. Chabini, E. K. Nam, and A. Cappiello, "Probabilistic modeling of acceleration in traffic networks as a function of speed and road type," in *IEEE ITSC 2002*, Singapore: IEEE, Sep. 2002, pp. 472–478.