A Simulation Tool for Automated Platooning in Mixed Highway Scenarios

Michele Segata[*†], Falko Dressler[*], Renato Lo Cigno[†], Mario Gerla[‡]

[*]University of Innsbruck, Austria, [†]University of Trento, Italy, [‡]University of California, Los Angeles

{segata,dressler}@ccs-labs.org, locigno@disi.unitn.it, gerla@cs.ucla.edu

ABSTRACT

Automated platooning is one of the most challenging fields in the domain of Intelligent Transportation Systems (ITS). Conceptually, platooning means creating clusters of vehicles which closely follow each other autonomously without action of the driver, neither for accelerating, nor for braking. This leads to several important benefits from substantially improved road throughput to increased safety. The control of such platoons depends on two components: First, radar is typically to be used to control the distance between the vehicles, and secondly, Inter-Vehicle Communication (IVC) helps managing the entire platoon allowing cars to join or to leave the group whenever necessary. Platooning systems have been mostly investigated in controlled environments such as dedicated highways with centralized management. However, platooning-enabled cars will be deployed gradually and might have to travel on highways together with other non-automated vehicles. We developed a combined traffic and network simulator for studying strategies and protocols needed for managing platoons in such mixed scenarios. We show the models needed and present first results using a simple IVC-based platoon management as a proof of concept.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design - Wireless Communication; I.6.5 [**Simulation and Modeling**]: Model Development

General Terms

Design, Verification

Keywords

Platooning, vehicular networks, cooperative adaptive cruise control, simulation

1. INTRODUCTION

In Intelligent Transportation Systems (ITS) platooning, as a mean of building an Intelligent Vehicle Highway Systems (IVHS) has always been a major challenge, because it encompasses several research fields from traffic management to control theory, and from vehicle dynamics to information technology [1, 7]. The main objective is to reduce road congestion and to increase traffic safety, as well as reducing CO_2 emissions thanks to the tight distance between cars. Radar-based techniques are used for Adaptive Cruise Control (ACC), i.e., for maintaining a safety distance from the vehicle in front. On the other hand, Inter-Vehicle Communication (IVC) is the basis for creating, managing, and organizing the platoons.

MobiCom'12, August 22–26, 2012, Istanbul, Turkey.
ACM 978-1-4503-1159-5/12/08.

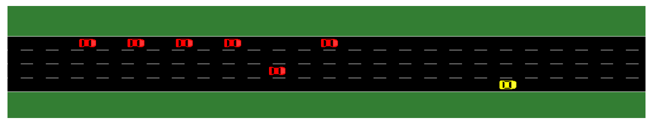


Figure 1: Screenshot of the simulator.

It is especially the IVC technique that need further investigation. So far, the ITS community mainly concentrated on fully automated and dedicated highways [1, 4, 7]. Only recently, the interest has moved to autonomous platooning, i.e., infrastructure-less operation on common roads together with human-driven vehicles [6]. We address one of the key issues in the development process of such systems, namely the integrated evaluation and performance assessment. Implementing and testing platoon management protocols in real environments without proper guarantees is not only expensive but possibly dangerous. We extended the IVC simulation toolkit Veins [8] for assessing the effectiveness of platooning management algorithms and protocols. In this paper, we introduce the resulting simulator and discuss the models we implemented using a simple IVC-based platoon management as a proof of concept. Figure 1 shows a snapshot of the simulator indicating an existing platoon and one car just leaving to take an exit.

Platooning has been studied since the 80's as a mean of increasing the throughput of the streets. PATH in California was one of the first pioneering projects [7]. Later on, Auto21 CDS [4] focused on the technologies needed for smooth merging and splitting. However, they considered dedicated highways for platooning-enabled cars only. Moreover, platoons were to be managed by a centralized center controlling form and composition of the platoons, a model that is still used in more recent studies [3]. One of the few projects considering fully autonomous platoons traveling on the road together with "common" drivers is SARTRE [6]. Here, the leader of the platoon is assumed to be a specially skilled driver, while all other vehicles are free to join and leave the platoon. Unfortunately, no information is available on how these maneuvers are to be performed or what IVC protocols are needed [1]. A Cooperative Adaptive Cruise

```
onInit:;
  setController(DRIVER);
onLeaveInRamp:;
  moveToLeftmostLane();
  setCCSpeed(150km/h);
  setController(ACC);
  sniffBeacons();
onStopSniffing:;
  if no beacons then
      startPlatoon();
      startBeaconing();
  else
      platoonId ← closestInFront();
      sendJoinRequest(platoonId);
  end
onJoinReply:;
  setCCSpeed(180km/h);
  setController(CACC);
onReplyTimeout:;
  startPlatoon();
  startBeaconing();
```

Algorithm 1: Vehicle start-up protocol

```
onDistanceToEnd < 1500:;
  sendLeaveNotification();
  leaveLeftMostLane();
  setController(DRIVER);
onLeaderLeaveNotification:;
  sendCurrentPosAndSpeed();
onChangeLeader(newLeader):;
  if me == newLeader then
      becomeLeader();
      startBeaconing();
      sendChangeAck();
  else
      updateLeader(newLeader);
  end
```

Algorithm 2: Follower life-cycle protocol

```
onDistanceToEnd < 1000:;
  if no followers then
      stopBeaconing();
      leaveLeftMostLane();
      setController(DRIVER);
  else
      Every 1 second:;
          sendLeaderLeaveNotification();
  end
onPosAndSpeedReceived:;
  saveFollowerData();
  if all data received then
      newLeader ← closestFollower();
      sendChangeLeader(newLeader);
      stopSendingLeaveNotifications();
  end
onChangeAck:;
  stopBeaconing();
  leaveLeftMostLane();
  setController(DRIVER);
```

Algorithm 3: Leader life-cycle protocol

Control (CACC) controller has been implemented in [2]. Yet, no ACC nor actuation lags have been considered and, again, only fully automated highways are supported.

With the emerging capabilities of IVC, there is now again a growing interest in autonomous platooning using CACC. Obviously, tool chains for testing systems and protocols and assessing their performance are strongly needed. We present a novel simulation toolkit allowing to study such platoon management systems, capturing both the mobility and the inter-vehicle communication issues. To the best of our knowledge, this is the first toolkit for investigating automated platoons in combination with other cars on a single highway.

2. SIMULATION ENVIRONMENT

We implemented the simulation tool based on the Veins simulator [8], which in turn uses OMNeT++[1] for network simulation and SUMO[2] for road traffic simulation. The main modifications have been made to SUMO, implementing a new car following model, which is able to either behave as a human driver or as an automated vehicle using ACC or CACC.

As a reference, we used the controllers detailed in [5]. Due to the lack of space, we only briefly described their properties. The simple Cruise Control (CC) follows the control equation $\ddot{x}_{des} = -k_p(\dot{x} - \dot{x}_{des})$, where $\dot{x}$ and $\dot{x}_{des}$ are the current and the desired speed, respectively, k_p is a design constant, and $\ddot{x}_{des}$ is the acceleration that should be applied. Whereas CC only controls the speed, ACC instead uses a radar in order to continuously measure the distance to the car in front. ACC uses $\ddot{x}_{i\_des} = -1/T\left(\dot{\varepsilon}_i + \lambda\delta_i\right)$ where $\dot{\varepsilon}_i$ is the speed relative to the vehicle in front, δ_i is the spacing error, so the difference to the desired gap distance, λ is a design constant, and T is the time headway, i.e., the desired distance in seconds to the vehicle in front (usually $T > 1\,\mathrm{s}$).

ACC is not suitable for tight car following in the order of 5 m to 10 m). Thus, a CACC controller is needed for platooning. CACC, takes into account the distance to the car in front using radar as well as information received from the platoon leader via IVC. The latter periodically broadcasts its current speed and acceleration. This way, all the followers can travel really close to each other without the risk of collisions. CACC control uses

$$\ddot{x}_{i\_des} = \alpha_1\ddot{x}_{i-1} + \alpha_2\ddot{x}_l + \alpha_3\dot{\varepsilon}_i + \alpha_4\left(\dot{x}_i - \dot{x}_l\right) + \alpha_5\varepsilon_i$$

where $\ddot{x}_{i-1}$ and $\ddot{x}_l$ are the accelerations of the vehicle in front and the leader, respectively, $\dot{\varepsilon}_i$ is the relative speed to the vehicle in front, $\dot{x}_i$ and $\dot{x}_l$ are the speeds of the car and of the leader, and ε_i is the spacing error. The α_i are design constants.

The desired acceleration $\ddot{x}_{i\_des}$ computed by the controllers cannot be applied immediately, because, due to the dynamics of the vehicle, there will be a certain actuation lag. As described in [5], this lag can be modeled as a first order lag, e.g., using a first order low pass filter with a time constant in the order of 0.5 s. The actual applied acceleration will be $\ddot{x}_t = \alpha \cdot \ddot{x}_{des} + (1-\alpha)\cdot\ddot{x}_{t-1}$ where $\ddot{x}_{t-1}$ is the acceleration at the previous time instant and $\alpha = 0.1666$, assuming a time constant of 0.5 s and a sampling time of 0.1 s.

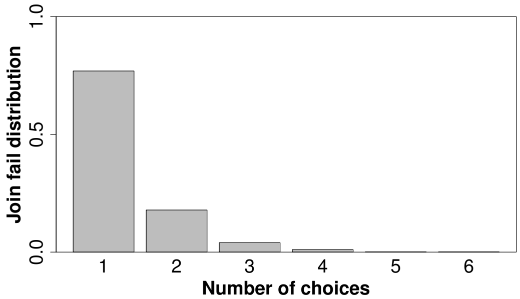
We implemented all the controllers within SUMO and extended the TraCI interface used by Veins to perform operations like *switchOnACC*, *setCACCLeaderData*, *changeToLane(x)*, etc. These commands simplify the implementation of an application-level protocol in the Veins environment to manage the platoon and to evaluate its performance using metrics like traffic improvement or robustness to communications failures.

3. A SIMPLE PLATOONING PROTOCOL

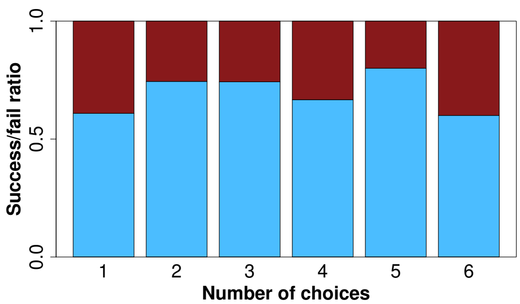
As a proof of concept, we developed a simple platoon management protocol. We considered a highway scenario with five in/exit ramps in which the leftmost lane is reserved for platoons, similar to carpool lanes in the U.S. We configured SUMO to generate cars with different routes, different desired speeds and different capabilities. Human driven vehicles just follow their route following the human-driver model. Platooning-enabled cars (30 % of the all cars) instead follow the rules according to Algorithms 1, 2, and 3.

[1] http://www.omnetpp.org/

[2] http://sumo.sourceforge.net



(a) Distribution of choices upon failure



(b) Success/failure ratio

Figure 2: Distribution of join failure as a function of the number of available platoons (a) and comparison between successfully (lightblue) and failed (darkred) joins (b).

A platooning-enabled car entering the highway simply moves to the reserved lane and switches ACC on. The duty of steering the car in and out of the reserved lane is left to the driver. Moreover, we consider all CACC enabled cars to drive at the same speed of 150km/h while in the platooning lane. Deciding what is the best thing to do when a faster platoon reaches a slower one is an open problem which has to be addressed in future. If it receives a beacon message from platoons in front within 5 s, it asks the closest one for permission to join. If that fails, it simply starts its own platoon, otherwise it switches to CACC and accelerates to join the platoon. For leaving the platoon after getting close to its destination, a car notifies the platoon leader, exits the reserved lane, and gives the control back to the driver. If the leader exits, it elects a the new leader (the one following closest), and sends a *changeLeaderNotification*.

In order to illustrate the developed simulator, a screenshot of the simulator indicating a platoon of six cars (red) together with a second car leaving the platoon and a human driven car (yellow) is shown in Figure 1.

For the analysis of the protocol we plotted three graphs. The first two, depicted in Figure 2, show, as function of the number of platoons in front found during the discovery phase, how the join failures are distributed, and the ratio between successful and failed joins. The first one tells us that in the majority of the cases, when a failure happens, the car has no other choices (at least in front of it). In some rare cases, the cars were aware of two or three platoons, meaning that the protocol could be improved by trying to join another platoon. The second plot emphasizes the latter statement, since the ratio of success/failures is independent from the number of choices, i.e., probably due to communication failures.

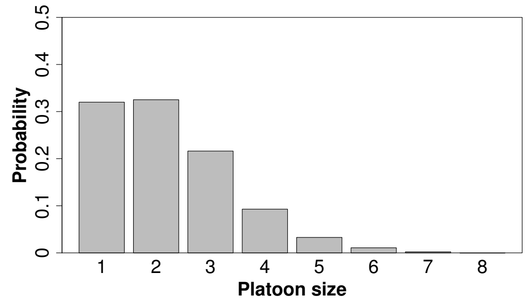


Figure 3: Distribution of platoon sizes computed from duration.

The fact that we are able to determine the existence of up to six platoons in the vicinity of a car, means that there may be several small platoons, thus the efficiency of the protocol might not yet be optimal. This is clearly depicted in Figure 3, where the distribution of platoon sizes is shown. We can see that the majority of the platoons have a size of one or two cars for the 60 % of the time. Only in some rare cases we have platoons of five to eight cars.

These early results and simple protocol implemented, show the power of the developed toolkit. They also suggest that reliable communications are fundamental for platooning efficiency. Moreover, it is clear that efficient and safe platooning needs means for merging, splitting and managing platoons, for instance by complementing the IVC communications with Road Side Units that enable inter-platoon communications even if these are out of direct communication range.

4. REFERENCES

[1] L. Baskar, B. De Schutter, J. Hellendoorn, and Z. Papp. Traffic control and intelligent vehicle highway systems: a survey. *IET Intelligent Transport Systems*, 5(1):38–52, 2011.

[2] P. Fernandes and U. Nunes. Platooning of autonomous vehicles with intervehicle communications in SUMO traffic simulator. In *IEEE ITSC 2010*, pages 1313–1318, Madeira Island, Portugal, September 2010.

[3] P. Fernandes and U. Nunes. Algorithms for management of a multi-platooning system of IVC-enabled autonomous vehicles, with high traffic capacity. In *IEEE ITSC 2011*, pages 1935–1941, Washington, D.C., October 2011.

[4] S. Hallé, B. Chaib-draa, and J. Laumonier. Car Platoons Simulated As A Multiagent System. In *4th Workshop on Agent-Based Simulation*, pages 57–63, Montpellier, France, April 2003.

[5] R. Rajamani. *Vehicle dynamics and control*. Springer, 2011.

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

[7] S. Shladover. PATH at 20 – History and Major Milestones. In *IEEE ITSC 2006*, pages 22–29, September 2006.

[8] C. Sommer, R. German, and F. Dressler. Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis. *IEEE Transactions on Mobile Computing*, 10(1):3–15, January 2011.