



Extending the vehicular network simulator Artery in order to generate synthetic data for collective perception

Christoph Allig¹ and Gerd Wanielik²

¹Corporate Research Department, DENSO AUTOMOTIVE Deutschland GmbH, 85386 Eching, Germany

²Department of Communication Engineering, Chemnitz University of Technology, 09126 Chemnitz, Germany

Correspondence: Christoph Allig (c.allig@denso-auto.de)

Received: 30 January 2019 – Accepted: 13 August 2019 – Published: 19 September 2019

Abstract. A fundamental for an automated driving car is the awareness of all its surrounding road participants. Current approach to gather this awareness is to sense the environment by on-board sensors. In the future, Vehicle-to-X (V2X) might be able to improve the awareness due to V2X's communication range superiority compared to the on-board sensors' range. Due to a limited amount of communication partners sharing their own ego states, current research focuses particularly on cooperative perception. This means sharing objects perceived by local on-board sensors of different partners via V2X. Data collections using vehicles, driving on real roads, is challenging, since there is no market introduction of cooperative perception yet. Using test cars, equipped with the required sensors are rather expensive and do not necessarily provide results representing the true potential of cooperative perception. Particularly, its potential is highly dependent on the market penetration rate and the amount of vehicles within certain vicinity. Therefore, we consider to create synthetic data for cooperative perception by a simulation tool. After reviewing suitable simulation tools, we present an extension of Artery and its counterpart SUMO by modelling realistic vehicle dynamics and probabilistic sensor models. The generated data can be used as input for cooperative perception.

tem is not able to provide a complete environmental model. On-board sensors like camera or radar are limited in their perception range and due to possible occlusions. Additionally, in unfavourable weather conditions the perception system might fail. However, for automated driving of level 3 and higher, a best possible perception of the environment is required in order to understand the scene adequately and to make the right decision. In order to resolve these issues, Vehicle-to-X (V2X) communication might be a key technology. The communication range is significantly higher than the sensors' perception range and V2X communication is robust to occlusions and weather conditions. In the first few years after market launch of V2X communication, the ratio of V2X enabled road participants will be limited. Additionally, there might always be non-connected road participants, e.g. pedestrians or cyclists not carrying any mobile phone. Therefore, only a certain ratio of road participants are able to broadcast ego state information, why current research focuses particularly on cooperative perception. This means sharing objects perceived by local on-board sensors of different partners via V2X. For the development of cooperative perception, simulation tools generating synthetic data are a reasonable complement to data collections. In this paper, we extend an open-source simulation tool such that it is able to generate synthetic data that can be used to develop cooperative perception applications. Firstly, we review in Sect. 2 simulation tools for cooperative automated driving (CAD). We have selected Artery as simulation tool, which extends the Vehicles in Network Simulation (Veins) by an implementation of the European Telecommunications Standards Institute Intelligent Transport Systems (ETSI ITS)-G5 protocol (ETSI, 2019) and a local perception toolbox. Veins couples the traffic simulator Simulation of Urban Mobility (SUMO)

1 Introduction

Today's Advanced Driver Assistance Systems (ADASs) rely on on-board sensors, which perceive other road participants and obstacles within their Field-of-View (FOV). The perception system fuses measurements from all of the on-board sensors, such as camera and radar, and generates an environmental model. Due to limited FOVs, the perception sys-

with the network-simulator OMNeT++. In Sect. 3 our extensions to SUMO are described in order to model realistic vehicle dynamics. Modifications to Artery are described in Sect. 4. This includes probabilistic sensor models, sensor fusion algorithms and a service for disseminating and receiving Cooperative Perception Messages (CPMs). In Sect. 5 an evaluation of the vehicle dynamics described in Sect. 3 is presented.

2 Review of relevant simulation tools

A simulation tool for CAD must provide sufficient realistic network communication models and sufficient realistic vehicle movements and sufficient realistic sensors, while achieving simultaneously sufficient scalability. The specifications sufficient realism and sufficient scalability are contradictory and result in a trade-off. The choice in favour of the realism or the scalability depends on the desired application. For example, if the application focuses on investigating characteristics of the network of Inter-Vehicle Communication (IVC), scalability and network communication models are in favour and realistic vehicle models might be ignored. Whereas, if the application aims at cooperative manoeuvring or evaluating tracking performance of cooperative perception, realism is in favour and requires to model e.g. the powertrain, lateral vehicle dynamics and distinct probabilistic sensors. A standalone simulation tool for CAD is not available to date. However, for simulating network models and vehicles there are various network, respectively traffic simulators available. A well-known option for CAD simulations is the bidirectional coupling of simulators, like coupling a traffic simulator and a network simulator. The simulators run in parallel and exchange data among them. Subsequently, we list non-exhaustively relevant simulators and coupling approaches.

Traffic simulators are distinguishable by the applied model, respectively the simulation's level of detail. There are macro-, meso-, micro- and submicroscopic modelling approaches. Macro- and mesoscopic models are listed only for the sake of completeness, but are not of relevance for CAD simulations. Macroscopic simulators model roads as smallest entity characterised by e.g. density or mean speed. Mesoscopic simulators model partitioned roads or a convoy of vehicles and represent an intermediate level in between macro- and microscopic simulators. Microscopic simulators model the movement of each vehicle by vehicle-following. VISSIM (PTV Planung Transport Verkehr AG, 2018) and its open-source counterpart SUMO (Krajzewicz et al., 2002) are popular representatives. Submicroscopic simulators extend microscopic models by additionally modelling specific parts of the vehicle, e.g. power train, vehicle dynamics or sensors. Submicroscopic traffic simulators are also often called, nanoscopic traffic simulator, or driving simulator and often included in robotics simulators. A non-exhaustive list of representatives for submicroscopic simulators: SiVIC (Gruyer

et al., 2014), PreScan (International TASS, 2018), TORCS (Wymann et al., 2015), RACER (Gonzalez and Kalyakin, 2009), VDrift (Kehrle et al., 2011), Webots (Michel, 2004) and PHABMACS (Massow and Radusch, 2018).

The most popular network simulators for CAD are OMNeT++ (OMNeT++, 2018) and ns-3 (ns-3, 2018).

Subsequently, different frameworks that enable the bidirectional coupling of traffic and network simulators are listed. Rondinone et al. (2013) presents the open-source simulation platform iTetris that integrates the traffic simulation SUMO and the network simulation ns-3. Llatser et al. (2017) couples the commercial submicroscopic vehicle simulator Webots and the network simulator ns-3, which achieves a high realism in vehicle dynamics. In Gomez et al. (2014) the submicroscopic traffic simulator MORSE and the network simulator ns-3 are coupled and applied for cooperative adaptive cruise control (CACC). Schünemann (2011) proposes VimSIMRTI, which allows flexible coupling of different network and traffic simulators, e.g. ns-3, OMNet++ and VSimRTI cellular, respectively SUMO and PHABMACS. The open-source tool Veins (Sommer et al., 2011) provides simulation modules for vehicular networking. It executes two simulators in parallel: OMNeT++ for network simulations and SUMO for traffic simulations. Both simulators are connected by the Traffic Control Interface (TraCI).

In recent research, there is also the trend for a coupled micro- and submicroscopic traffic simulation. Pereira and Rossetti (2012) couples the microscopic traffic simulator SUMO and the submicroscopic traffic simulator USARSim. Also in Barthauer and Hafner (2018), a micro- and submicroscopic traffic simulator are bidirectionally coupled, namely SUMO and SILAB. Aramrattana et al. (2017) presents a simulation framework consisting of the VTI's driving simulator, the traffic Simulator Plexe-SUMO and the network simulator Plexe-VEINS. This framework is applied for testing and evaluation of CACC. Plexe (Segata et al., 2015) is an open source extension to VEINS and permits the simulation of CACC or platooning systems. Communication protocols for platooning are realized in VEINS, while distinct cruise controllers and engine dynamics are implemented in SUMO. Schiller et al. (2015) proposes an approach for coupling traffic simulators (SUMO and VIRES Virtual Test Drive) of different resolutions to satisfy the constraints of performing simulations in real time.

In this work, we use the simulation framework Artery (Riebl et al., 2015), which extends Veins by an implementation of the ETSI ITS-G5 network and transport protocols, called Vanetza. Additionally, each vehicle is individually configurable regarding its capabilities of ITS-G5 services. For example, Artery includes a ITS-G5 service for disseminating Cooperative Awareness Messages (CAMs). In Günther et al. (2015), Artery is extended by a local perception toolbox, which allows for the introduction of individual local perception sensors for each vehicle. The framework has been used so far for investigating the network performance and to

Table 1. Longitudinal and lateral vehicle model parameters

	parameter	value
vehicle parameters	c_{air}	0.3
	A	2.1 [m ²]
	ρ	1.2 [kg m ⁻³]
	m	1300 [kg]
	g	9.81 [m s ⁻²]
	c_{r1}	0.0136
	c_{r2}	5.18×10^{-7}
	N_c	4
	a_0	-40.222
	a_1	0.065
	a_2	-2.159×10^{-5}
	a_3	4.360×10^{-9}
	a_4	-3.130×10^{-13}
	η	0.9
	l_f	1.6 [m]
	l_r	1.9 [m]
	I_z	10 000 [kg m ²]

compare the awareness ratio between on-board sensors and CAMs (Günther et al., 2016, 2018).

3 SUMO

As already mentioned, SUMO is a microscopic traffic simulator and thus does not include any vehicle model. The consequences are instantaneous accelerations or instantaneous and unbounded yaw rates. This is impractical as a simulation tool used for development or evaluation of cooperative perception. In Segata et al. (2015), longitudinal vehicle dynamics have already been added to SUMO. We further extend this approach by modelling additionally lateral vehicle dynamics.

Subsequently, the modifications to SUMO are described. Section 3.1 describes the longitudinal vehicle dynamics, while Sect. 3.2 describes the lateral vehicle dynamics. Further theoretical background on vehicle dynamics can e.g. be found in Rajamani (2006). The vehicle parameters applied in the longitudinal and lateral models are summarized in Table 1. In Sect. 3.3, the Pure Pursuit algorithm is presented, which is used to keep the vehicle centred in its current lane.

3.1 Longitudinal vehicle dynamics

The longitudinal acceleration \ddot{x} of a vehicle is dependent on the mass of the vehicle m and the sum of the forces acting in longitudinal direction on the vehicle:

$$m\ddot{x} = F_x - F_A - F_R - F_G, \quad (1)$$

where F_A , F_R and F_G are the air resistance, the rolling resistance, respectively the gravitational force:

$$F_A = \frac{1}{2}c_{\text{air}}A\rho\dot{x}^2, \quad (2)$$

$$F_R = mg(c_{r1} + c_{r2}\dot{x}^2), \quad (3)$$

$$F_G = mg \sin(\theta_{\text{road}}), \quad (4)$$

where c_{air} is the aerodynamic drag coefficient, ρ is the density of air and A is the cross-sectional area of the vehicle. The parameters c_{r1} and c_{r2} are dependent on the tires and their pressure. The gravitational acceleration and the inclination of the road are denoted as g , respectively θ_{road} . The acceleration force F_x is dependent on the desired acceleration $\ddot{x}_{\text{des}}[t]$, which is determined in a longitudinal control system. However, the desired acceleration is not immediately the true acceleration of the vehicle. Indeed, there is an actuation lag due to communication delays between distinct controllers and the physical process until the force is transmitted to the tires and accelerates the vehicle. This actuation lag is modelled using a first order lag:

$$\ddot{x}[t] = \alpha\ddot{x}_{\text{des}}[t] + (1 - \alpha)\ddot{x}[t - 1]. \quad (5)$$

The parameter α depends on the simulation sampling time Δt and the actuation lag τ :

$$\alpha = \frac{\Delta t}{\tau + \Delta t}, \quad \tau = \tau_{\text{inj}}(n) + \tau_{\text{burn}}(n) + \tau_{\text{exh}}. \quad (6)$$

Opening the fuel valves and injecting the fuel is modelled by the parameter $\tau_{\text{inj}}(n)$, while burning the fuel and accelerating the crankshaft is considered by the parameter $\tau_{\text{burn}}(n)$:

$$\tau_{\text{inj}}(n) = \frac{2(N_c - 1)}{nN_c}, \quad \tau_{\text{burn}}(n) = \frac{3}{2n}. \quad (7)$$

Additionally, a constant transport delay τ_{exh} of 100 ms is considered. N_c is the number of cylinders and n is the engine speed expressed in [rps]. The maximum engine force depends on the the engine power P_{eng} , which again is dependent on the current engine speed N_{eng} in [rpm]:

$$F_{\text{eng}} = \frac{\eta P_{\text{eng}}(N_{\text{eng}})}{\dot{x}}, \quad (8)$$

where η denotes the engine efficiency and \dot{x} is the speed in [m s⁻¹]. Engine power curves can be found e.g. at Automobile-Catalog (2019). We model the engine power using a polynomial function:

$$P_{\text{eng}}(N_{\text{eng}}) = \sum_{i=0}^N a_i N_{\text{eng}}^i. \quad (9)$$

So far, we only considered forces in case of positive acceleration, where the acceleration force is $F_x = F_{\text{eng}}$. On the other hand, if the vehicle is braking, the acceleration force F_x is negative. The maximum braking force is $F_x = -\mu mg$. The braking actuation lag is fixed to $\tau = 200$ ms.

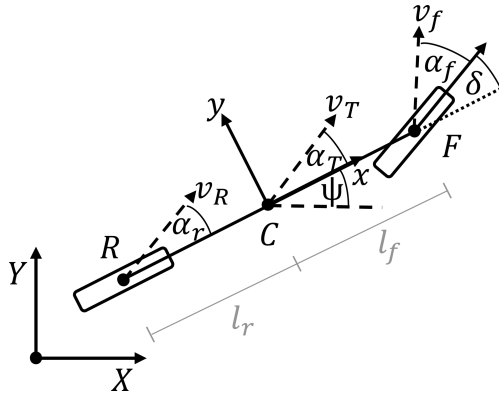


Figure 1. Dynamic bicycle model.

3.2 Lateral vehicle dynamics

For modelling the lateral vehicle dynamics, the dynamic bicycle model as shown in Fig. 1 is applied. The distances of the front axle F and the rear axle R to the center of the vehicle C are denoted as l_f , respectively l_r . The yaw angle ψ defines the orientation of the vehicle's body frame with regard to the global frame. The side slip angle α_t and the yaw angle define together the driving direction of the vehicle with velocity v_t . The direction of the velocity v_f at the front axle is dependent on the front slip angle α_f and the steering wheel angle δ . The rear slip angle α_r determines the direction of the velocity v_r at the rear axle.

Using again Newton's second law, the lateral acceleration is dependent on the mass of the vehicle and the sum of the forces acting in lateral direction on the vehicle:

$$ma_y = m(\ddot{y} + v_x \dot{\psi}) = F_{yf} + F_{yr}, \quad (10)$$

where F_{yf} and F_{yr} are the lateral tire forces of the front and rear wheels, respectively. The inertial lateral acceleration a_y comprises two terms, the acceleration along the y -axis \ddot{y} and the centripetal acceleration $v_x \dot{\psi}$. The angular acceleration $\dot{\psi}$ is dependent on the moment of inertia I_z , the lateral tire forces and the distances of the front l_f and rear l_r tire to the center of the vehicle:

$$I_z \dot{\psi} = l_f F_{yf} - l_r F_{yr}. \quad (11)$$

The tire forces are due to deformations which occur during manoeuvring. These deformations result in a slip angle α and vice versa the tire forces can be expressed as a non-linear function depending on the slip angle. The slip angle of the front wheel α_f and the rear wheel α_r can be determined by:

$$\alpha_f = \tan^{-1} \left(\frac{v_y + l_f \dot{\psi}}{v_x} \right) - \delta, \quad \alpha_r = \tan^{-1} \left(\frac{v_y - l_r \dot{\psi}}{v_x} \right). \quad (12)$$

To model the front and rear tire forces, the well-known Pacejka tire model is applied. The generated force can be ex-

pressed as a function of the slip angle α :

$$F_y = F_{y,n}(\alpha_{eq}) + S_v, \quad \text{with } \alpha_{eq} = \alpha - S_h, \quad (13)$$

where

$$F_{y,n} = D \cdot \sin \left(C \tan^{-1} \left(B \alpha_{eq} - E \left(B \alpha_{eq} - \tan^{-1} (B \alpha_{eq}) \right) \right) \right). \quad (14)$$

F_y might be the front lateral tire force F_{yf} or the rear lateral tire force F_{yr} and α the front slip angle α_f or the rear slip angle α_r . For further information on the model parameters B , C , D , E , S_h and S_v , see e.g. Rajamani (2006).

Rewriting Eqs. (10) and (11) results in the lateral acceleration and the angular acceleration:

$$\ddot{y} = \frac{F_{yf} + F_{yr}}{m} - v_x \dot{\psi}, \quad \ddot{\psi} = \frac{l_f F_{yf} - l_r F_{yr}}{I_z}. \quad (15)$$

In order to be able to model the previously described lateral vehicle dynamics in SUMO, it is required to detach the vehicle from its lane and that the vehicle is able to move freely through the cartesian space. As shown in Fig. 2, in SUMO the vehicle's movement is usually described by a position on lane value. The vehicle is always driving on the center of lane, wherefore the lateral offset equals zero. A global cartesian position might be calculated by a function that uses additionally the start and end position of the present lane. We determine the vehicle's movement in the global frame and update always the redundant values that describe the longitudinal and lateral position on the current lane. The steering wheel angle δ is the input parameter of the lateral vehicle dynamics. The steering wheel angle is controlled by the path tracking algorithm that is described in the next subsection.

3.3 Path tracking

For path tracking we apply the Pure Pursuit controller, which is a popular geometric controller that determines the desired steering wheel angle δ according to:

$$\delta = \tan^{-1} \left(\frac{2L \sin(\theta)}{l_d} \right). \quad (16)$$

It's basic principle is to follow a carrot-point, that is defined based on the look-ahead distance l_d and the planned trajectory. In our case corresponds the planned trajectory to the center of lane. The angle θ is given through the velocity vector of the vehicle and the look-ahead vector. The parameter L corresponds to the wheelbase of the vehicle.

4 Artery

We start with a brief review of the local perception toolbox of Artery (Günther et al., 2015), which is shown in Fig. 3. The

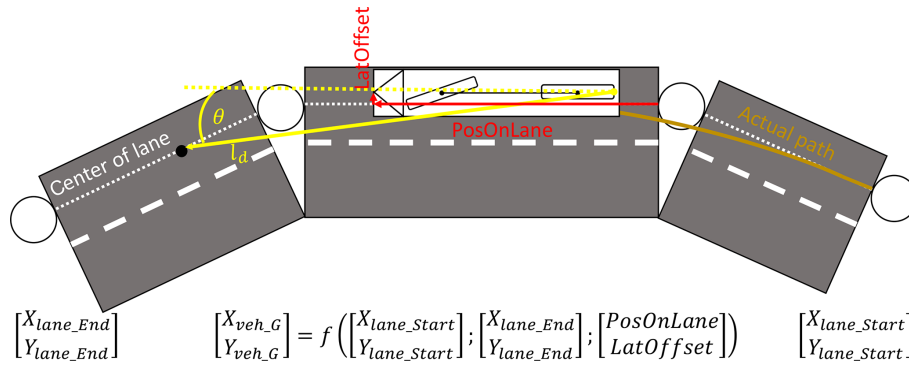


Figure 2. Representation of the position in SUMO.

local perception toolbox of Artery is based on the Global Environmental Model (GEM), the Local Environmental Model (LEM) and the local perception sensors.

The GEM acts as a global database containing all vehicles within the simulation. Each vehicle is described by a Global Environmental Object (GEMO). The GEMO retrieves continuously values of the corresponding vehicle in SUMO via TraCI. Each vehicle might be equipped with local perception sensors. Those sensors are configurable by the perception range, the opening angle as well as the mounting point on the vehicle. The GEM provides functionality to determine if other vehicles are perceived by the installed sensors.

Every vehicle equipped with a sensor creates its own instance of a LEM. The LEM acts as the database of all vehicles that are perceived by the ego vehicle. Whenever a measurement is performed by the sensor, the vehicles within its perception range, respectively their corresponding Local Environmental Model Objects (LEMOs) are updated. Whenever a vehicle is first measured by a perception sensor, i.e. the vehicle has not been sensed before by the ego vehicle, a new LEMO is created for that particular vehicle. In Günther et al. (2015), there is one LEMO per sensor per perceived object. The LEMO stores the last measurements. Whereas, in our approach, a LEMO comprises one object tracker processing the measurements of all perception sensors installed in the ego vehicle. In Sect. 4.1 a probabilistic radar and camera model are described. In Sect. 4.2 we describe the operating principle of the involved modules in cooperative perception.

4.1 Sensor modelling

We extend Artery by a probabilistic radar and camera sensor model. The radar sensor measures target range r , azimuth angle θ and range rate \dot{r} with frequency $f = 10$ Hz:

$$r = \sqrt{(x_o - x_e)^2 + (y_o - y_e)^2} + w_r,$$

$$\theta = \tan^{-1} \left(\frac{y_o - y_e}{x_o - x_e} \right) + w_\theta,$$

$$\dot{r} = v \cos \left(\phi_o - \tan^{-1} \left(\frac{y_o - y_e}{x_o - x_e} \right) \right) + w_{\dot{r}}.$$

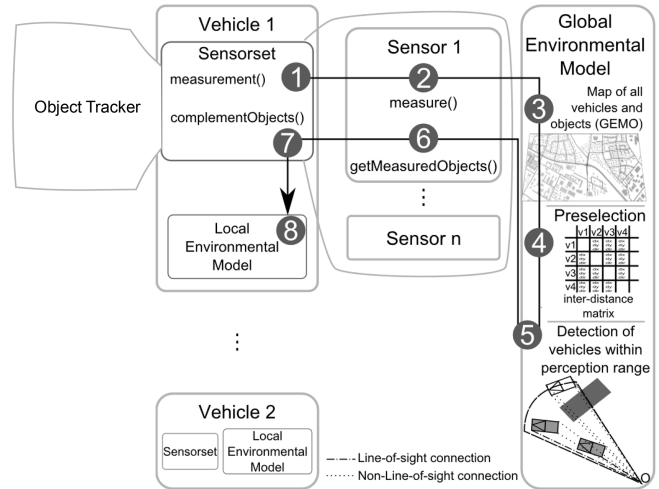


Figure 3. Artery's local perception toolbox (Günther et al., 2015).

The respective measurement noise is described by w_r , w_θ and $w_{\dot{r}}$, which are zero mean white Gaussian noises with standard deviations $\sigma_r = 1.2$ m, $\sigma_\theta = 0.01$ rad and $\sigma_{\dot{r}} = 0.45$ m s⁻¹. The camera sensor measures the position in picture coordinates u and v with the same frequency:

$$u = \frac{q_{11}(x_o - x_e) + q_{12}(y_o - y_e) + q_{14}}{q_{31}(x_o - x_e) + q_{32}(y_o - y_e) + q_{34}} + w_u, \quad (17)$$

$$v = \frac{q_{21}(x_o - x_e) + q_{22}(y_o - y_e) + q_{24}}{q_{31}(x_o - x_e) + q_{32}(y_o - y_e) + q_{34}} + w_v. \quad (18)$$

Here the respective measurement noise is given by w_u and w_v , which are zero mean white Gaussian noises with standard deviations $\sigma_u = 6$ and $\sigma_v = 6$ pixels. The variables q_{ij} correspond to the elements of the projection matrix. The global position of the detected vehicle and the ego vehicle are given by x_o and y_o , respectively by x_e and y_e . The detected vehicle drives with velocity v towards the direction ϕ_o . The sensor models and its parameters have been taken over from the data fusion software BASELABS Create (BASELABS, 2019).

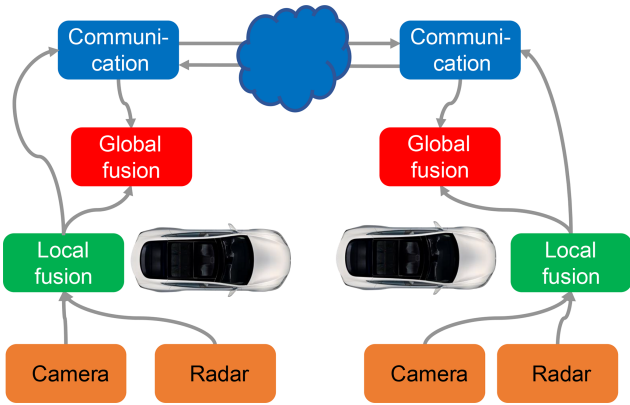


Figure 4. Architecture of the cooperative perception.

4.2 Cooperative perception

The implemented architecture of the cooperative perception is shown in Fig. 4. The in Sect. 4.1 described sensors generate noisy measurements which are fused in the respective local fusion module. The result of the local fusion module is broadcasted via V2X communication. The information received by the communication module is then fused into the global fusion module with the result of the local fusion module.

Whenever a LEMO is updated, its corresponding local fusion is triggered, i.e. there are multiple single object tracker. The object tracker firstly compensates the ego motion. Secondly, the track is predicted using the Unscented Kalman Filter (UKF) according to the system model. Depending on the sensor that has triggered the local fusion, a noisy measurement is generated. Finally, the track is updated by the UKF using the corresponding measurement model. A CPM service is implemented that creates a CPM message containing information about the ego state, the FOVs of the local sensors and the tracked objects. The ego state information is interpretable as a downsized CAM and describes the dynamic state of the broadcasting vehicle. It is provided by Artery's GEMO. The tracked object states are obtained by an access to the LEMOs' local fusion module.

Artery with its extensions described so far is used to evaluate the network load. Additionally, Artery outputs the ground truth and non-noisy measurements that might be used for Monte-Carlo simulations, i.e. the noise is added later in a separate fusion framework. Both opportunities are shown in Fig. 5.

5 Evaluation

This section evaluates the proposed extensions of SUMO, i.e. the longitudinal and lateral vehicle models as well as the path tracking. For evaluation a map of the sub-urban area Garching is used. Figure 6a shows the trajectory the vehicle

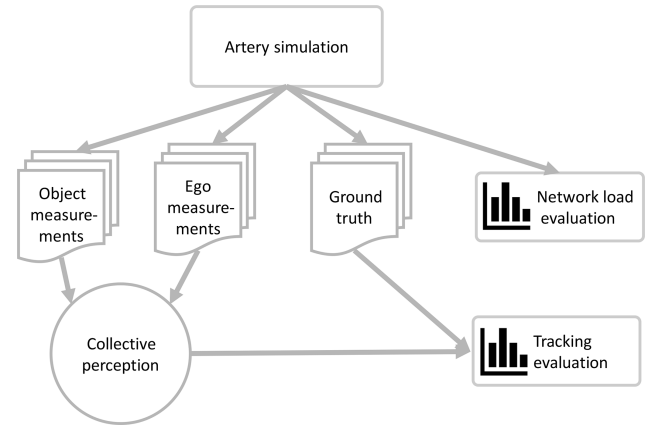


Figure 5. Artery data generation.

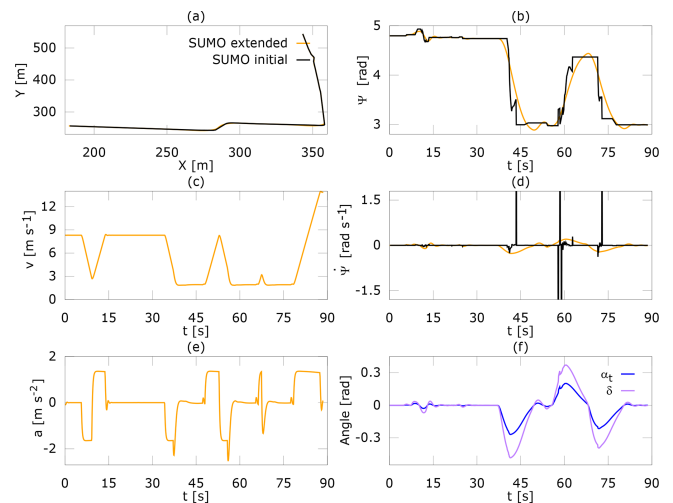


Figure 6. Vehicle dynamics.

is driving in the initial SUMO version and after the extension. The initial trajectory corresponds to the center of lane, which is used as input for the Pure Pursuit algorithm. The vehicle drives from the top to the left bottom. During cornering, the vehicle slightly cuts the corner, which is a typical challenge for the Pure Pursuit controller. In Fig. 6c the velocity is shown. Before cornering the vehicle is braking, and afterwards accelerating. As visible in Fig. 6e the acceleration is not instantaneous available. Figure 6b compares the yaw angles. Adding the lateral vehicle dynamics as described in Sect. 3.2 results in a continuous yaw angle, i.e. there are no abrupt jumps. The yaw rates are compared in Fig. 6d. The initial yaw rate is changing very rapidly and achieves large values due to the discontinuities in the yaw angle. In contrast, the yaw rate after the extension is within a realistic range and continuous. In Fig. 6f, the steering wheel angle and the slip angle are shown.

6 Conclusions

This paper describes a number of extensions to the simulation tool Artery and its counterpart SUMO. We add longitudinal and lateral vehicle dynamics as well as a lateral controller to the vehicle simulator SUMO. Moreover, Artery's local perception toolbox is extended by probabilistic sensor models and a service for disseminating CPMs. Afterwards, we evaluate the changes regarding the vehicle dynamics. In our view, the framework is now able to be applied for investigation of cooperative perception, which we plan for future work. Adequate network communication models and scalability have already been available and the extensions provide in our view sufficient realistic vehicle movements and sensors. The future work may e.g. include to test different sensor fusion methods or different message formats.

Code availability. DENSO AUTOMOTIVE Deutschland GmbH has not granted permission for the public accessibility of the source code.

Author contributions. CA investigated appropriate simulation tools. CA conceived the idea to extend the simulation tool Artery as described and carried out the implementation and evaluation. GW supervised the work.

Competing interests. The authors declare that they have no conflict of interest.

Special issue statement. This article is part of the special issue "Kleinheubacher Berichte 2018". It is a result of the Kleinheubacher Tagung 2018, Miltenberg, Germany, 24–26 September 2018.

Financial support. This research has been supported by the European Commission, H2020 (ADAS&ME (grant no. 688900)).

Review statement. This paper was edited by Madhu Chandra and reviewed by Andreas Danklmayer and Madhu Chandra.

References

Aramrattana, M., Larsson, T., Jansson, J., and Nåbo, A.: A Simulation Framework for Cooperative Intelligent Transport Systems Testing and Evaluation, *Transport. Res. F-Traf.*, 61, 268–280, <https://doi.org/10.1016/j.trf.2017.08.004>, 2017.

Barthauer, M. and Hafner, A.: Coupling Traffic and Driving Simulation: Taking Advantage of SUMO and SILAB Together, *EPiC Series in Engineering*, 2, 56–66, 2018.

BASELABS: Sensor Fusion Library for Algorithm Implementation, available at: <https://www.baselabs.de/create/>, last access: 15 January 2019.

ETSI: Intelligent Transport Systems (ITS); Access Layer Specification for Intelligent Transport Systems Operating in the 5 GHz Frequency Band, EN 302 663 V1.3.0, ETSI, 2019.

Gomez, A. E., Dos Santos, T. C., Filho, C. M., Gomes, D., Perafan, J. C., and Wolf, D. F.: Simulation Platform for Cooperative Vehicle Systems, in: *IEEE Intelligent Transportation Systems Conference*, 1347–1352, 8–11 October 2014, Hyatt Regency Qingdao, Qingdao, China, 2014.

Gonzalez, N. and Kalyakin, I.: RACER : A Non-Commercial Driving Game which Became a Serious Tool in the Research of Driver Fatigue, in: *Design and Use of Serious Games*, Springer, 2009.

Gruyer, D., Choi, S., Boussard, C., and Andréa-novel, B.: From Virtual to Reality, How to Prototype, Test and Evaluate New ADAS: Application to Automatic Car Parking, in: *IEEE Intelligent Vehicles Symposium*, 261–267, 8–11 June 2014, Ypsilanti, Michigan, USA, 2014.

Günther, H. J., Trauer, O., and Wolf, L.: The Potential of Collective Perception in Vehicular Ad-Hoc Networks, in: *International Conference on ITS Telecommunications*, 1–5, 2–4 December 2015, Copenhagen, Denmark, 2015.

Günther, H. J., Riebl, R., Wolf, L., and Facchi, C.: Collective Perception and Decentralized Congestion Control in Vehicular Ad-Hoc Networks, in: *IEEE Vehicular Networking Conference*, 8–10 December 2016, Columbus, OH, USA, 2016.

Günther, H. J., Riebl, R., Wolf, L., and Facchi, C.: The Effect of Decentralized Congestion Control on Collective Perception in Dense Traffic Scenarios, *Comput. Commun.*, 122, 76–83, 2018.

International TASS: PreScan/TASS International, available at: <https://tass.plm.automation.siemens.com/prescan> (last access: 10 January 2019), 2018.

Kehrle, F., Frasch, J. V., Kirches, C., and Sager, S.: Optimal Control of Formula 1 Race Cars in a VDrift Based Virtual Environment, *IFAC Proceedings Volumes*, 44, 11907–11912, 2011.

Krajzewicz, D., Hertkorn, G., and Wagner, P.: SUMO (Simulation of Urban MObility) – An Open-Source Traffic Simulation, in: *Proceedings of the 4th Middle East Symposium on Simulation and Modelling*, 183–187, January, Sharjah, United Arab Emirates, 2002.

Llatser, I., Jornod, G., Festag, A., Mansolino, D., Navarro, I., and Martinoli, A.: Simulation of Cooperative Automated Driving by Bidirectional Coupling of Vehicle and Network Simulators, in: *IEEE Intelligent Vehicles Symposium*, 1881–1886, 11–14 June 2017, Los Angeles, CA, USA, 2017.

Massow, K. and Radusch, I.: A Rapid Prototyping Environment for Cooperative Advanced Driver Assistance Systems, *J. Adv. Transport.*, 2018, 1–32, <https://doi.org/10.1155/2018/2586520>, 2018.

Michel, O.: Webots TM : Professional Mobile Robot Simulation, *Advanced Robotic Systems*, 1, 39–42, 2004.

ns-3: ns-3/a discrete-event network simulator for internet systems, available at: <https://www.nsnam.org/> (last access: 14 January 2019), 2018.

OMNeT++: OMNeT++, Discrete Event Simulator, available at: <https://omnetpp.org/> (last access: 13 January 2019), 2018.

Pereira, J. L. F. and Rossetti, R. J. F.: An Integrated Architecture for Autonomous Vehicles Simulation, in: *Proceedings of the 27th*

- Annual ACM Symposium on Applied Computing, 286–292, 26–30 March 2012, Trento, Italy, 2012.
- PTV Planung Transport Verkehr AG: PTV Vissim, available at: <http://vision-traffic.ptvgroup.com/en-us/products/ptv-vissim/> (last access: 13 January 2019), 2018.
- Rajamani, R.: *Vehicle Dynamics and Control*, Springer, 2006.
- Riebl, R., Günther, H. J., Facchi, C., and Wolf, L.: Artery: Extending Veins for VANET applications, in: 2015 International Conference on Models and Technologies for Intelligent Transportation Systems, 450–456, 3–5 June 2015, Budapest, Hungary, 2015.
- Rondinone, M., Maneros, J., Krajzewicz, D., Bauza, R., Cataldi, P., Hrizi, F., Gozalvez, J., Kumar, V., Röckl, M., Lin, L., Lazaro, O., Leguay, J., Härrä, J., Vaz, S., Lopez, Y., Sepulcre, M., Wetterwald, M., Blokpoel, R., and Cartolano, F.: ITETRIS: A Modular Simulation Platform for the Large Scale Evaluation of Cooperative ITS Applications, *Simul. Model. Pract. Th.*, 34, 99–125, 2013.
- Automobile-Catalog: The Cataloge of Cars, Cars Specs Database, available at: <https://www.automobile-catalog.com/>, last access: 23 August 2019.
- Schiller, M., Dupuis, M., Krajzewicz, D., Kern, A., and Knoll, A.: Multi-Resolution Traffic Simulation for Large Scale High Fidelity Evaluation of VANET Applications, SUMO User Conference 2015 – Intermodal Simulation for Intermodal Transport, 17–36, Springer, Berlin, Germany, 2015.
- Schünemann, B.: V2X Simulation Runtime Infrastructure VSimRTI: An Assessment Tool to Design Smart Traffic Management Systems, *Comput. Netw.*, 55, 3189–3198, 2011.
- Segata, M., Joerer, S., Bloessl, B., Sommer, C., Dressler, F., and Cigno, R. L.: Plexe: A Platooning Extension for Veins, in: IEEE Vehicular Networking Conference, 53–60, 3–5 December 2014, Paderborn, Germany, 2015.
- Sommer, C., German, R., and Dressler, F.: Bidirectionally coupled network and road simulation for improved IVC analysis, *IEEE T. Mobile Comput.*, 10, 3–15, 2011.
- Wymann, B., Sumner, A., Dimitrakakis, C., Guionneau, C., and Espi, E.: TORCS: The Open Racing Car Simulator, 2015.