SUMO User Conference 2020
https://doi.org/10.52825/scp.v1i.95
© Authors. This work is licensed under a Creative Commons Attribution 3.0 DE License
Published: 28 Jun. 2022

Extending the Intelligent Driver Model in SUMO and Verifying the Drive Off Trajectories with Aerial Measurements

Dominik Salles[1], Stefan Kaufmann[2] and Hans-Christian Reuss[3]

[1] Research Institute of Automotive Engineering and Vehicle Engines Stuttgart (FKFS), Germany
[2] IT-Designers GmbH, Esslingen, Germany
[3] University of Stuttgart – Institute of Automotive Engineering (IFS), Germany
dominik.salles@fkfs.de

Abstract

Connected and automated driving functions are key components for future vehicles. Due to implementation issues and missing infrastructure, the impact of connected and automated vehicles on the traffic flow can only be evaluated in accurate simulations. Simulation of Urban Mobility (SUMO) provides necessary and appropriate models and tools. SUMO contains many car-following models that replicate automated driving, but cannot realistically imitate human driving behavior. When simulating queued vehicles driving off, existing car-following models are neither able to correctly emulate the acceleration behavior of human drivers nor the resulting vehicle gaps. Thus, we propose a time-discrete 2D Human Driver Model to replicate realistic trajectories. We start by combining previously published extensions of the Intelligent Driver Model (IDM) to one generalized model. Discontinuities due to introduced reaction times, estimation errors and lane changes are conquered with new approaches and equations. Above all, the start-up procedure receives more attention than in existing papers. We also provide a first evaluation of the advanced car-following model using 30 minutes of an aerial measurement. This dataset contains three hours of drone recordings from two signalized intersections in Stuttgart, Germany. The method designed for extracting the vehicle trajectories from the raw video data is outlined. Furthermore, we evaluate the accuracy of the trajectories obtained by the aerial measurement using a specially equipped vehicle.

1 Introduction

In the last few decades, many traffic simulation software packages have been developed to study traffic flow and movement patterns of pedestrians. Multiple reviews regarding performance, usability and portability of the programs SUMO (Lopez et al. 2018), PTV Vissim, Aimsun, Paramics, MATSim,

CORSIM and TRANSIM have been published in Kotusevski and Hawick (2009), Ejercito et al. (2017) and Dallmeyer (2014). SUMO's ability to handle large networks and its open source framework make it attractive for researchers. The source code, written in C++, can be fully examined and modified. This creates a foundation for integrating custom car-following models and devices. Another notable add-on is the "Traffic Control Interface" (TraCI) (Wegener et al. 2008), which enables the communication with SUMO via the Transmission Control Protocol (TCP).

Simulation studies and modifications utilizing SUMO are often related to connected and autonomous driving functions. VSimRTI (Queck et al. 2008) and Veins (Sommer et al. 2008) are two exemplary communication tools, which use TraCI to connect the traffic simulation SUMO with network simulators (NS2/NS3 or OMNET++), so that real-world network protocols and V2V communication can be developed and tested. A multitude of researchers have carried out investigations of how connected and autonomous driving will change the road capacity and future traffic flow, by using custom programmed devices and extensions, such as the Adaptive Cruise Control (ACC) and Cooperative Adaptive Cruise Control (CACC) models (Milanés and Shladover 2014; Xiao et al. 2017; Xiao et al. 2018). Among them, Alekszejenkó and Dobrowiecki (2019) present an intelligent traffic control algorithm coupled with platooning vehicles developed in SUMO to improve urban traffic flow. They point out that future work would need to analyze the impact of human drivers in the scenario to better quantify the improvements. Richter et al. (2019) actually study the effect of mixed traffic (autonomous vehicles and human drivers) on a highway. They use the Krauss model (Krauss et al. 1997) and define smaller time headways, reaction times and sigma values (driver imperfection) for autonomous vehicles.

Other studies in this field, e.g., Derbel et al. (2012) and Zhou et al. (2016), use the IDM to represent the automated vehicles. They show that the original IDM is particularly well equipped to replicate automated driving, while the human driving behavior is either simulated using the Two Velocity Difference Model (Derbel et al. 2012) or the Full Velocity Difference Model (Zhou et al. 2016).

SUMO is also often used for extracting realistic trajectories and their characteristic values. The trajectories are then used to retrieve typical driving cycles to calculate the energy consumption of vehicles (Macedo et al. 2013; Pfeil 2019; Donateo et al. 2010). These studies use SUMO to incorporate the effect of infrastructure and traffic dynamics on the consumption. Grumert et al. (2015) and Erdağı et al. (2019) go one step further, not merely focusing on the energy used, but also taking the emissions into account. The Krauss model was used for these investigations, which result in realistic velocities and traffic flow, but may produce unrealistic accelerations and therefore emissions. When using macroscopic values in combination with, e.g., the Handbook Emission Factors for Road Transport (HBEFA), the calculated values can be fairly accurate, but when extracting single trajectories, the results can highly depend on the acceleration. For this reason, a car-following model is needed that can produce realistic accelerations and jerks.

2 Related Work

According to the thorough review in Saifuzzaman and Zheng (2014), car-following models can be categorized as follows: safety distance models, Cellular Automata, Optimal Velocity Models, desired measures models, Gazis-Herman-Rothery models and their extensions as well as models based on the human perspective. The last category contains models with perception thresholds, models based on risk-taking of the driver or driving by visual angle. Enhancements include fuzzy logic, distraction and driver errors.

Safety distance models are designed to always provide a safe distance and prevent collisions. In the Gipps model (Gipps 1981), this is achieved without considering the speed of the respective vehicle. The virtual driver intends to always brake with parameter b, expecting the leader to decelerate with the

identical value. In heterogeneous traffic, this can result in collisions. When the braking parameter *b* of the leader is assumed to be higher, unrealistically large gaps form between the vehicles.

Optimal Velocity Models (OVM), originally developed by Bando et al. (1995), use a different approach. They employ a constant sensitivity coefficient to describe the reactivity of the driver. The model, however, is not free of collisions, and when the coefficient is selected to avoid accidents, it induces high accelerations and decelerations. Continuing developments led to the Generalized Force Model (Treiber et al. 2000) and the Full Velocity Difference Model (FVDM) (Jiang et al. 2001). In contrast to the OVM, the FVDM takes the velocity of the leader into account and, thus, stabilizes the model.

Human drivers do not react to every single change of the environment. Wiedemann incorporated this observation into his psycho-physical model (Wiedemann 1974). The model features perception thresholds, which are only exceeded when the variables change significantly. In addition, The model differentiates between four different driving modes, Free driving, approaching, following and strong braking, and it has many more functions and parameters than other car-following models.

In contrast, Cellular Automata (CA) and the Krauss model are much less detailed. Cellular Automata as car-following models were first introduced by Nagel and Schreckenberg (1992). They operate in a time- and space-discrete fashion. The road is therefore divided into equally large cells that can only be occupied by one vehicle at a time. The vehicle's speed is randomized to produce stochastic behavior. Despite its simple nature, this model can realistically reproduce traffic phenomena. The Krauss model, the default car-following model in SUMO (Krauss et al. 1997), can be considered a space-continuous version of the Nagel and Schreckenberg model. It combines the advantages of the CA model with those of the Gipps model. The Krauss model operates collision-free and contains only a few functions and parameters. Additional advantages are the asymmetrical acceleration-deceleration behavior and the emulation of the human reaction time with the simulation step time. Therefore, this model can well replicate many observed traffic phenomena (Alazzawi et al. 2018), despite producing unrealistic acceleration and jerk patterns.

This research article focuses mainly on the realistic representation of human acceleration patterns using car-following models. The selected integration scheme has a significant influence on this representation. As pointed out by Treiber and Kanagaraj (2015), the ballistic update is always more accurate than the Euler method or faster with the same accuracy. In SUMO, you can choose either one. Due to discontinuities, such as lane changes, the standard fourth-order Runga-Kutta method is not applicable for multi-lane traffic simulations. Therefore, car-following models like the OVM, FVDM, GHR and IDM have to be described in a time-discrete fashion, although they were all developed as time-continuous models. In this study, the Euler method was used, as it is the default integration scheme in SUMO.

The original IDM produces realistic accelerations and jerks almost independently of the integration method and has been continuously modified since its introduction. Shortly thereafter, Treiber and Helbing (2004) concluded that human drivers leave larger gaps when driving off than the IDM predicts, although the jerk remains realistically small in all driving situations. It should not exceed 1.5 m/s^3 (Treiber and Kesting 2013b). To achieve this goal, all discontinuities that can occur in SUMO's traffic simulation must be taken into account.

They include:

- Change of speed limits
- Reaction times, light signals and priorities at junctions before drive off
- Reaction times before braking
- Greater gaps after lane changes
- Smaller gaps after lane changes
- Driving behavior near the minimal gap

The Extended Intelligent Driver Model (EIDM) is introduced in the following section. Chapter 4 presents the case study and aerial measurement method. Similar databases to the here presented measurement method have recently been released and published by Krajewski et al. (2018) and Bock et al. (2019). Their drones observed German highways and unsignalized intersections, but urban traffic phenomena at signalized intersections are not included. The advantages of the drone measuring method are pointed out in the above mentioned articles: naturalistic driving behavior and the possibility to simultaneously capture the movement of multiple vehicles without any occlusion. The authors compare their datasets with that of the New Generation SIMulation (NGSIM) set (Kovvali et al.) and other urban intersection data. The NGSIM dataset includes trajectories of vehicles crossing intersections, but it is not possible to use the raw trajectories without first smoothing the data or re-extracting the trajectories from the video (Thiemann et al. 2008; Krajewski et al. 2018; Coifman and Li 2017). Consequently, the acceleration patterns are considered unrealistic.

3 Car-following model

This chapter starts off by reviewing the original IDM and its substantial enhancements so far. Subsequently, the EIDM is presented with all its modifications. Figure 1 defines the nomenclature, since many different definitions can be found in the literature. All listed parameters belong to the following vehicle and all variables regarding the leader will carry the subindex n.

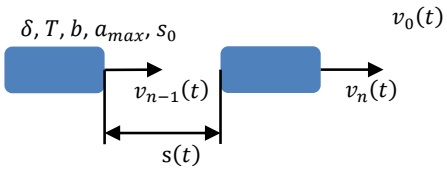

Figure 1: Notation of the car-following model

3.1 Improved Intelligent Driver Model

The IDM, first introduced as a time-continuous model (Treiber et al. 2000), consists of two main equations and five parameters, the desired time headway T, the maximum acceleration a_{max}, the desired deceleration b, the minimum gap s_0 and the acceleration exponent δ. The desired gap

$$s^*_{n-1}(t) = s_0 + max\left(0, v_{n-1}(t) * T - \frac{v_{n-1}(t) * \left(v_n(t) - v_{n-1}(t)\right)}{2 * \sqrt{a_{max} * b}}\right) \quad (1)$$

depends on three of those parameters and the velocities $v_n(t)$ and $v_{n-1}(t)$. The acceleration

$$a_{IDM}(t + \Delta t) = a_{max}\left[1 - \left(\frac{v_{n-1}(t)}{v_0(t)}\right)^{\delta} - \left(\frac{s^*_{n-1}(t)}{s(t)}\right)^2\right] \quad (2)$$

is determined by calculating the ratio between the current velocity $v_{n-1}(t)$ and the desired velocity $v_0(t)$ and the ratio between the desired gap $s^*_{n-1}(t)$ and the actual gap $s(t)$. The latter ratio represents the intelligent braking strategy and assures a collision-free execution of this model. However, this term does not allow following vehicles to reach the desired velocity in homogeneous traffic conditions and induces ever larger gaps. The Improved Intelligent Driver Model (IIDM) accounts for this negative trait by changing the model characteristics close to the desired velocity (Treiber and Kesting 2013b). The new term of free acceleration leads to more realistic gaps between vehicles. The authors calculate $a_{free}(t)$ by differentiating between two cases: $v_{n-1}(t) \leq v_0$ and $v_{n-1}(t) > v_0$. The EIDM calculates the free acceleration

$$a_{free}(t) = a_{max}\left[1 - \left(\frac{v_{n-1}(t)}{v_0(t)}\right)^{\delta}\right] \tag{3}$$

without using the case distinction of the IIDM, but by linearizing the changes in the desired velocity $v_0(t)$ (see Section 3.4). The following equation of the resulting acceleration $a(t + \Delta t)$ further differs from that of the IIDM. Instead of calculating the exponent with $a_{free}(t)$, its absolute value is used. If the exponent were negative, the acceleration would be unsteady at $v_0(t)$.

The resulting acceleration

$$a(t + \Delta t) = \begin{cases} a_{max}\left[1 - \left(\frac{s^*_{n-1}(t)}{s(t)}\right)^2\right] & s^*_{n-1}(t) \geq s(t) \\ a_{free}(t)\left[1 - \left(\frac{s^*_{n-1}(t)}{s(t)}\right)^{\frac{2*a_{max}}{|a_{free}(t)|}}\right] & otherwise \end{cases} \tag{4}$$

differentiates between two cases: driving at distances lower than the desired gap and higher than the desired gap.

3.2 Human Driver Model

The Human Driver Model (Treiber et al. 2006) was developed by the authors of the IDM. To generate human driver behavior in the model, they introduced a reaction time, imperfect estimation capabilities and temporal and spatial anticipation. Spatial anticipation has not yet been integrated into the EIDM in SUMO.

Due to the time-continuous form of the IDM, the reaction time was introduced with a Delay Differential Equation (DDE). As SUMO runs in a time-discrete fashion, the variables of the last several time steps would need to be stored in vectors to employ this method, thereby requiring significant amounts of memory. In addition, the driver always reacts to an earlier state $t - t_{reac}$ ago and the model needs to be carefully calibrated to stay stable. To solve these issues, Action Points (APs) are introduced. Simulating with APs implies that the driver can instantaneously process any information at the action time t_{AP}. Between two APs, the model uses the variables from the last AP update. Furthermore, t_{AP} can be varied throughout the simulation to overcome stability issues when the driver needs to react quickly.

The estimation errors are modeled using a Wiener process, which is defined by the variable w_i, determined at step *i*, using the correlation time $\tilde{\tau}$, a randomized number η_i of variance 1 and the time step Δt of SUMO:

$$w_i = e^{-\frac{\Delta t}{\tilde{\tau}}} * w_{i-1} + \sqrt{\frac{2\Delta t}{\tilde{\tau}}} * \eta_i \tag{5}$$

The variable w_i is then used to calculate the estimated distance $s^{est}(t)$, the estimated velocity of the leader $v_n^{est}(t)$ and a driving error $\sigma_a w_a(t)$, which is added to the acceleration term. For the systematic derivation, see Treiber et al. (2006).

In equations (6), (7) and (8), the variables $w_s(t)$, $w_n(t)$, $w_a(t)$ are the corresponding Wiener processes, represented in (5). The parameters V_s, σ_r and σ_a describe the respective magnitude of the errors.

$$s^{est}(t) = s(t) * e^{V_s w_s(t)} \tag{6}$$

$$v_n^{est}(t) = -s(t)\sigma_r w_n(t) + v_n(t) \tag{7}$$

$$\tilde{a}(t) = a(t) + \sigma_a w_a(t) \tag{8}$$

By introducing a reaction time, the model can become unstable and simulate accidents. To prevent those the model uses anticipation terms. The driver anticipates the velocity of the leader and his own acceleration to remain constant until the next AP, resulting in the predicted velocities and distances in (9), (10) and (11).

$$v_{n-1}^{pred}(t) = v_{n-1}^{est}(t_{AP}) + (t - t_{AP}) * a(t_{AP}) \tag{9}$$

$$v_n^{pred}(t) = v_n^{est}(t_{AP}) \tag{10}$$

$$s^{pred}(t) = s^{est}(t) - (t - t_{AP}) * \Delta v_{n-1}^{est}(t_{AP}) \tag{11}$$

3.3 Enhanced Intelligent Driver Model

The Enhanced IDM improves the lane changing behavior of the original IDM, since it was first developed as a single-lane model (Kesting et al. 2010). This model reduces the deceleration when gaps are instantaneously reduced after lane changes and, nevertheless, remains collision-free. A new equation calculates the Constant Acceleration Heuristic (CAH) $a_{CAH}(t)$ as follows, taking the acceleration $a_n(t)$ of the leader into account.

$$a_{CAH}(t) = \begin{cases} \dfrac{v_{n-1}^2 \tilde{a}_n}{v_n^2 - 2\mathrm{s(t)}\tilde{a}_n} & v_n(v_{n-1} - v_n) \leq -2\mathrm{s(t)}\tilde{a}_n \\ \tilde{a}_n - \dfrac{(v_{n-1} - v_n)^2 \theta}{2\mathrm{s(t)}} & otherwise \end{cases} \tag{12}$$

$$\theta = \begin{cases} 0 & v_{n-1} - v_n < 0 \\ 1 & v_{n-1} - v_n \geq 0 \end{cases} \tag{13}$$

$$\tilde{a}_n = \min(a_n(t), a_{max}) \tag{14}$$

θ is the Heaviside step function. The CAH-model cannot operate as a stand-alone model, it is used as an extension of the IDM. The acceleration a_{ACC} is calculated using the new coolness parameter c_{ACC}, with values between 0 and 1. It describes how "cool" a driver reacts when gaps are reduced.

$$a_{ACC} = \begin{cases} a_{IDM} & a_{IDM} \geq a_{CAH} \\ (1 - c_{ACC})a_{IDM} + c_{ACC}\left[a_{CAH} + b * \tanh\left(\frac{a_{IDM} - a_{CAH}}{b}\right)\right] & otherwise \end{cases} \quad (15)$$

This results in a temporary acceptance of lower gaps. Without this modification, the vehicles in SUMO often do not change lanes or brake hard after lane changes when using the IDM.

3.4 Further EIDM enhancements

In order to create a realistic human driver model in SUMO, a few more modifications have to be carried out. First of all, the introduced estimation errors und reaction times cause problems when decelerating to 0 m/s. Acceleration jumps occur at small gaps because the predicted and estimated values are wrought with intentional errors. Consequently, the virtual driver cannot smoothly approach the minimal gap s_0. That problem is solved by introducing equation (16). When reaching the gap $s_0 + \gamma$ (minimal gap plus threshold), the vehicle is forced to decelerate further, although the desired gap $s^*_{n-1}(t)$ might be smaller than the actual gap $s(t)$. This leads to vehicles stopping prior to s_0, but overcomes the effect of oscillating accelerations at low gaps. γ values between 0.3 and 0.5 were empirically determined to be suitable.

$$s^*_{n-1}(t) = \begin{cases} s(t) + 0.05 & s^*_{n-1}(t) < s(t) < s_0 + \gamma \\ s^*_{n-1}(t) & otherwise \end{cases} \quad (16)$$

Analog to the definition of the speed factor in SUMO, every vehicle is assigned an individual minimal gap s_0 from a normal distribution.

Another adjustment is made regarding changing speed limits. The presented equations in Section 3.1 do not take changing speed limits into account. We therefore use the simple linear function in (17) to continuously change the desired velocity when the speed limit changes. The model receives a new parameter T_{prev} to look $T_{prev} * v_{n-1}$ meters ahead. This results in a model-internal desired velocity v_0^{int} when driving near two edges with different speed limits v_0^i and v_0^{i+1}. The distance to the upcoming edge is represented by $s^i(t)$.

$$v_0^{int}(t) = \begin{cases} v_0^{int} - \left(v_0^i - v_0^{i+1}\right) * \Delta t / T_{prev} & s^i(t) < T_{prev} * v_{n-1} \\ v_0 & otherwise \end{cases} \quad (17)$$

Furthermore, all turns at junctions receive a speed limit according to Table 1, which is used to limit the velocity when turning. The parameters in Table 1 refer to the turn categories in SUMO, which are defined by the turn's radius. Alternatively, the limits could be calculated using the specific radius of the turn or street curvature.

To update the desired velocity with the maximum assigned speed of the next turn, the model uses Equation (17) with a look ahead of $T_{prev} * v_{n-1}$ in order for the vehicle to reach that speed before turning.

Parameter LINKDIR	TURN	TURN _LEFTHAND	LEFT	RIGHT	PARTLEFT	PARTRIGHT
v_0 [m/s²]	5.0	5.0	9.0	8.0	12	12

Table 1: Speed limits for turns at junctions in SUMO

According to Wagner and Lubashevsky (2003), the time period between subsequent human driver decisions can amount to several seconds. For the model, such large reaction times result in hard braking behavior. This is solved by introducing variable APs (Treiber and Kesting 2017). Equation (18) is based on a similar formula, but uses a predefined action time t_{AP} and a constant threshold ε instead of a random number. This stabilizes the model during critical events, still allowing for potentially long reaction times. The modeled driver reacts instantaneously when the car-following model calculates negative acceleration changes smaller than $-\varepsilon$.

$$a(t) - a(t - t_{AP}) < -\varepsilon \tag{18}$$

APs have an additional effect on the model: At standstill, just before drive off, the acceleration can jump to a value as high as a_{max}. In reality, the jerk is limited by the inertia of the vehicle and the powertrain. This characteristic is incorporated in the EIDM by applying a simple hyperbolic tangent function, thereby introducing a new correction factor $a_{corr}(t)$ (see Equation (19)) that limits the jerk during drive off. This requires the detection of the time at drive off t_{off} and the definition of a new parameter: the time duration t_{amax} between drive off and reaching the maximal acceleration.

$$a_{corr}(t) = \begin{cases} \left(\tanh\left(\left((t - t_{off}) * 2/t_{amax} - M_{bg}\right) * M_{fl}\right) + 1\right)/2 & t - t_{off} \le t_{amax} \\ 1 & otherwise \end{cases} \tag{19}$$

The new $a_{corr}(t)$-function is multiplied with the maximal acceleration a_{max}, thus better replicating drive off procedures of measured trajectories. Examples showing the correction factor over time for specific parameter sets are plotted in Figure 2. Parameter M_{fl} defines the flatness of the acceleration curve and should optimally take on values between 1.5 and 3. A change in parameter M_{bg} shifts the curve in the direction of the x-axis. Lowering the value results in reaching higher a_{corr} values earlier in time.

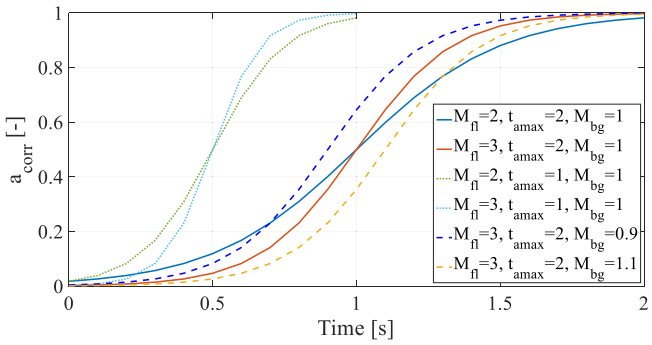

Figure 2: Drive off correction factor

Finally, the decrease of the ratio $s^*_{n-1}(t)/s(t)$ is limited. This modification produces more realistic jerks, especially in a simulation environment with junctions, traffic lights and lane changes, where the actual gap $s(t)$ and the desired gap $s^*_{n-1}(t)$ can instantaneously change. These discontinuities can be countered by the following Equation (20), which limits the change of the ratio to a specific magnitude. For $s^*_{n-1}(t) \geq s(t)$ (see Equation (4)), this adjustment guarantees that the jerk of the vehicle never exceeds j_{max}, which is a freely selectable, positive parameter.

$$\frac{s^*_{n-1}(t)}{s(t)} = \begin{cases} \sqrt{\left(\frac{s^*_{n-1}(t-\Delta t)}{s(t-\Delta t)}\right)^2 - \frac{\Delta t * j_{max}}{a_{max}}} & \left(\frac{s^*_{n-1}(t-\Delta t)}{s(t-\Delta t)}\right)^2 - \left(\frac{s^*_{n-1}(t)}{s(t)}\right)^2 > \frac{\Delta t * j_{max}}{a_{max}} \\ \frac{s^*_{n-1}(t)}{s(t)} & otherwise \end{cases} \tag{20}$$

To simulate driving on multi-lane roads, car-following models need to be coupled with a lane change model. This study uses the default lane change model integrated in SUMO (Erdmann 2014). The model is modified for an improved performance with the EIDM. Whether the modifications also improve the operation of the other car-following models, will require further investigation and, therefore, are not proposed in this paper.

4 Case study

In this chapter, real-world trajectories are compared to traffic simulation results generated with SUMO and the EIDM. First, the environment and the specific configuration of the scenario are introduced. The details of the aerial measurements are described, including an accuracy evaluation. The results are then compared to those of the SUMO simulation by means of time headways, speeds and other characteristic values.

4.1 Environment

Figure 3 shows one frame of drone videos recorded on Monday, July 2, 2018, between 07:00 and 07:35. It depicts a junction in Stuttgart, Germany, often referred to as “Neckartor”. The geographical coordinates are approximately 48°47'18.7"N 9°11'28.9"E. The video frame extends 230 m from left to right. The arrows represent the street and direction of traffic considered in this study. The vehicles drove freely after passing the intersection with a speed limit of 50 km/h. The light signal at this junction operated in a fixed-time fashion with a cycle time of 120 s, divided into the following phases in the direction of the arrows: 33 s of green light, 3 s yellow, 24 s red, 34 s green, 3 s yellow and 23 s red. Between red and green phases, the light signals switched to red-yellow for 1 s.

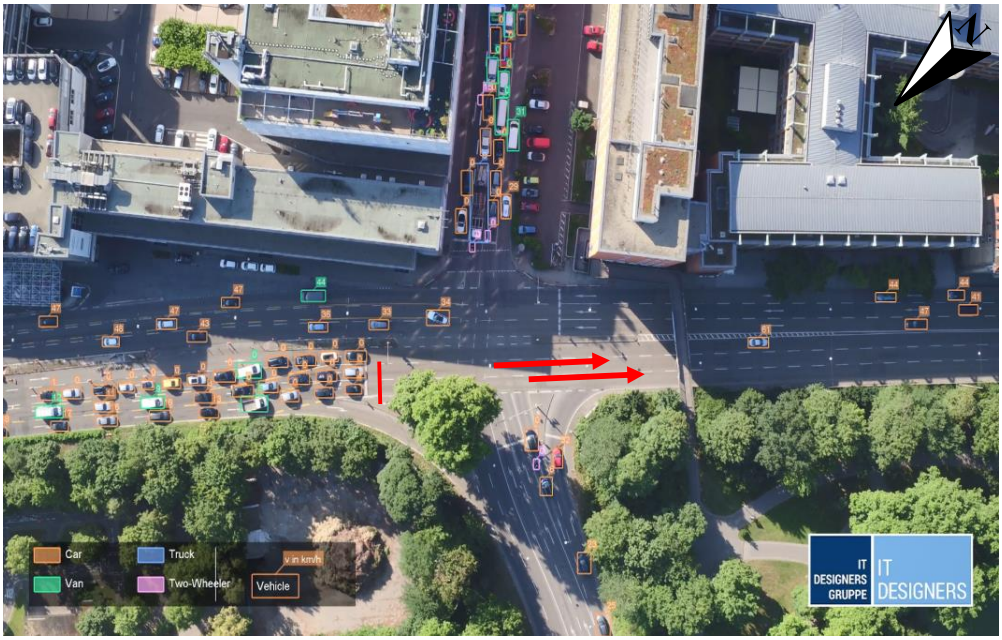

Figure 3: Drone recording of the signalized junction with bounding boxes and vehicle types

Specific configurations and anomalies need to be considered:

- The lanes before the junction are narrower than behind the junction.
- Only phases, when exclusively passenger cars and vans crossed, are analyzed, this results in disregarding three green phases when heavy duty vehicles crossed.

- The red line shown in Figure 3, used as observation reference line, is located approximately 3.5 m to the right of the actual stop line, because some vehicles stopped slightly past it.
- Only the two left lanes are analyzed, because many vehicles in the right lane are temporarily covered by a tree and some also turn right.

In sum, 1050 vehicle trajectories and 540 accelerations starting at 0 km/h were observed (30 green phases consisting of 9 vehicles in 2 lanes each).

4.2 Drone data

The full dataset contains over three hours of video recordings in segments of 7-15 minutes. The dataset used for the evaluation of the car-following model consists of 5 videos, each about 7 minutes long. The videos were recorded with a Zenmuse X5R camera mounted onto a DJI Inspire 1 quadcopter. The camera generated RAW-files with a resolution of 3850p25.

Figure 4 shows the applied traffic measurement process. After the video is recorded, geolocations are referenced at about 10 different reference positions within a single frame. The references are used to define the camera location, which is then tracked for all frames of the recording. The camera tracking algorithm uses automatically detected, well trackable, stationary markers, a method previously used by Kaufmann et al. (2018).

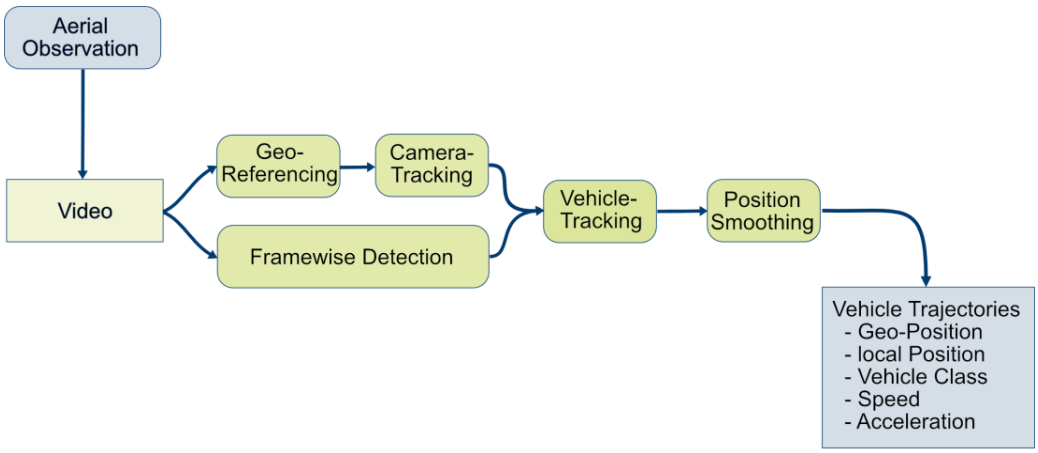

Figure 4: UAV based traffic measurement process

The frame-by-frame vehicle detection uses an artificial neural network for object detection. For this recording, a Faster R-CNN (Ren et al. 2015) with Resnet101 (He et al. 2016) is chosen. Afterwards, the screen positions of the detections are converted into geographic coordinates, using the camera model information and the tracked camera position for each frame.

The tracking algorithm eventually joins the single frame detections to form vehicle trajectories. First, an Intersection over Union (IOU) tracker (Bochinski et al. 2017) combines the detections that closely overlap in successive frames into short vehicle trajectories. Then we use a particle filter to predict the path of these short trajectories and connect them with others. This leads to trajectories that usually cover the entire recording area and correspond to a single vehicle. Some mismatches are adjusted manually.

In order to reduce measurement errors, the vehicle positions are smoothed, after which the velocity is immediately derived. This will be discussed in detail in the next section.

4.3 Measurement accuracy evaluation

The applied measuring method contains two main sources of error: First, the bounding boxes of the same vehicle differ slightly in each frame. This results in a position jitter. Secondly, even the tracking of the stationary reference points causes a small jitter in the camera position, which affects the transformation of the vehicle positions into world coordinates.

For a measurement with 25 frames/s and with a position error Δp, the error propagates to the speed with $\Delta v = 1/t\,\Delta p = 25\,1/s\,\Delta p$. The acceleration error increases quadratically $\Delta a = 1/t^2\,\Delta p = 625\ 1/s^2\,\Delta p$. As solution for this problem, we apply an averaging procedure using a moving linear regression (MLR) as shown in Figure 5.

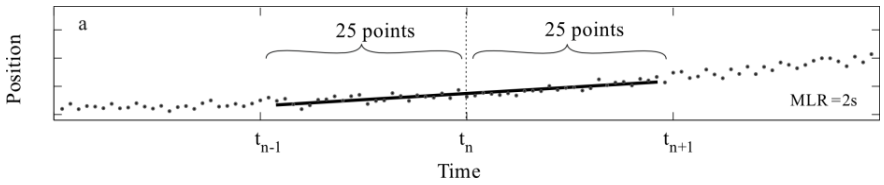

Figure 5: Explanation of the linear regression procedure. With a framerate of 25 fps, the MLR interval is 2 s (Kaufmann et al. 2018).

We take a measurement point related to the time instant t_n. Then, we perform the linear regression for all measurements within a time interval $t_n - 0.5T_S \leq t \leq t_n + 0.5T_S$, where T_s is the time interval duration of the linear regression. In the following, we refer to T_s as the MLR interval.

In order to obtain the vehicle speed $v^2 = v_x^2 + v_y^2$, we first calulate v_x and v_y separatly via the MLR, using the same interval. Afterwards, we use the MLR to obtain the acceleration from the speed using a different interval.

In order to find suitable MLR intervals, a measurement was carried out with a reference vehicle in the MEC-View research project (Gabb et al. 2019). The vehicle was equiped with an Automotive Dynamic Motion Analyzer (ADMA), a highly precise Inertial Measurement Unit (IMU) that uses a Differential Global Positioning System (DGPS). We use the speed measurement and the derived acceleration as the ground truth for a comparison of the smoothing parameters.

Figure 6 shows the first measured scenario: the vehicle stops at an intersection. In the second scenario, in Figure 7, the vehicle drives across an intersection. The second column shows the measurement results using an MLR with an interval of 1 s, and the third column shows the results using an MLR with an interval of 2 s. The first row shows the measured speed, the second row only the measured ADMA acceleration, the third row shows the acceleration as derivative of the speed, and the last row shows the resulting acceleration using an MLR with an interval of 1 s.

The results in Figure 6 show that a speed MLR interval of 1 s leads to a strong acceleration error propagation. Only an MLR interval of 2 s reduces the error sufficiently, but may sometimes cut off short-term spikes. Nevertheless, we consider the data quality to be quite usable.

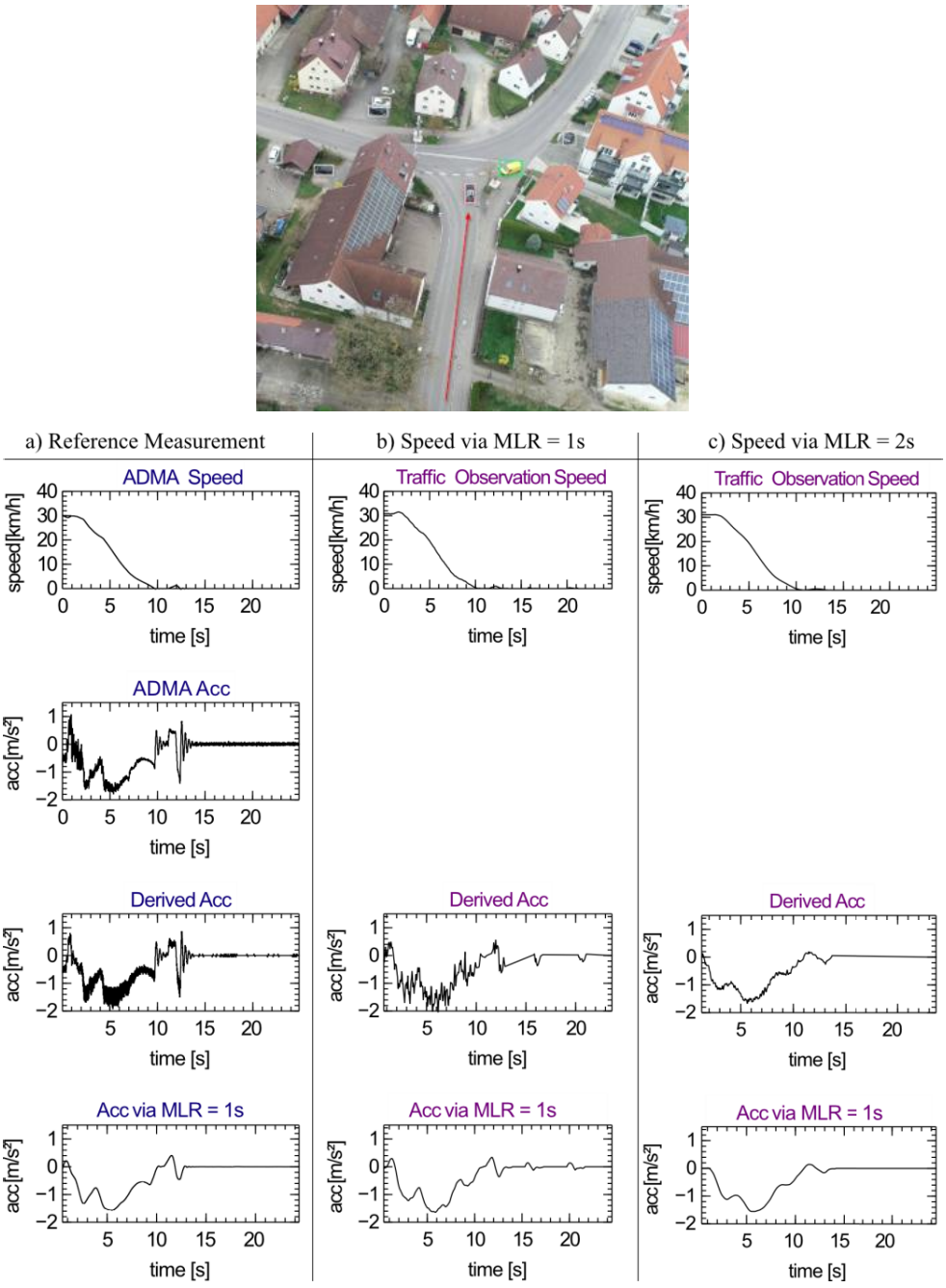

Figure 6: Reference Measurement Scenario 1: Vehicle stops at an intersection. **a)** the ADMA ground truth, **b)** the UAV based measurement with an MLR of 1 s and **c)** the UAV based measurement with an MLR of 2 s

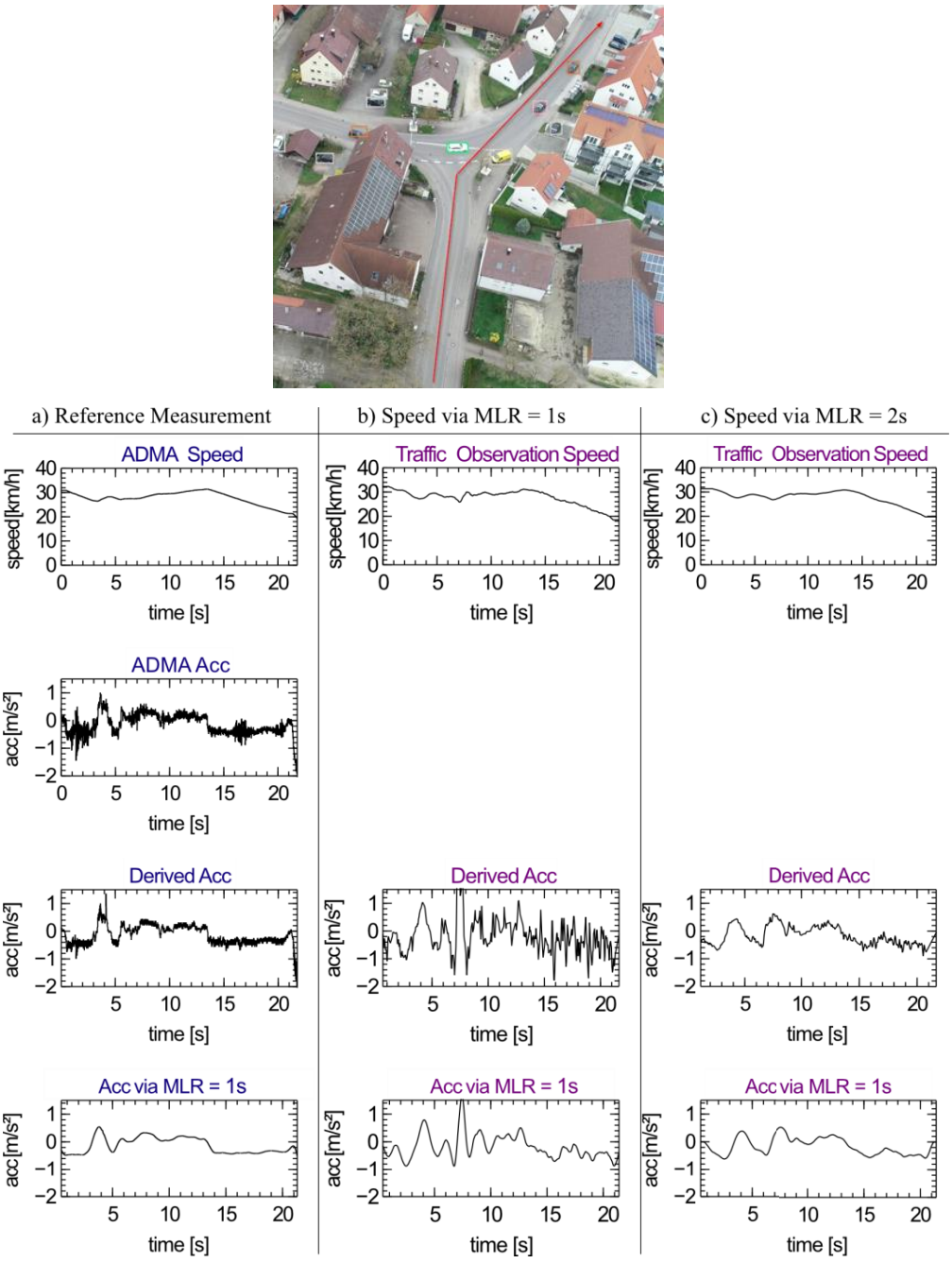

Figure 7: Reference Measurement Scenario 2: Vehicle merges at an Intersection. **a)** the ADMA ground truth, **b)** the UAV based measurement with an MLR of 1 s and **c)** the UAV based measurement with an MLR of 2 s

4.4 Analysis of the drive off trajectories

The aerial measurement method cannot detect the initial movement of the vehicles until they reach an average speed of about 1 m/s. An offset has to be applied to the trajectories in order to remedy this undesirable feature. For this reason, data from multiple vehicle measurements are extracted and used for comparison. The data is provided by the FKFS, where over the past several years many vehicle studies have been carried out, each collecting a vast amount of data at high measurement resolutions. Vehicle, route and driver specifications of the data samples can be found in the literature (Fried 2004; Rumbolz et al. 2010; Wagner et al. 2010).

The data was entirely collected in the broader area of Stuttgart. To identify a comparable offset all start-up procedures that fulfill the following specifications are extracted:

- The vehicle reaches 10 km/h in less than 3 s.
- The vehicle reaches 30 km/h in less than 8 s.
- The vehicle is slower than 60 km/h for the full 10 s of drive off.
- Right before the start-up the vehicle comes to a standstill for at least 2 s.
- The acceleration is between -5 m/s^2 and 5 m/s^2 for the full 10 s of drive off.
- The vehicle has an internal combustion engine.

This specific method resulted in over 2000 comparable drive off trajectories, showing that the vehicles reach 1 m/s after an average time of 0.8 s and their maximal acceleration after an average time of 2.3 s. The average time the vehicles need to reach 1 m/s is then added to every drive off detected by the aerial measurements.

Figure 8a) shows the mean accelerations of the drive off procedures of the first nine vehicles (with offset) after the light signal turns green. A drop and rise of the acceleration between approximately 3 s and 8 s can be recognized. This phenomenon originates from powertrains with manual transmissions. The difference between vehicles with automatic and those with manual transmissions becomes evident when comparing their mean acceleration curves. Figure 9 shows curves from above mentioned measured drive offs of the various vehicle studies. The curves in Figure 8b) and dashed lines in Figure 9 represent the absolute standard deviation of the corresponding acceleration curves, which are approx. ± 0.5 m/s^2 over the whole time period except before the acceleration peak is reached and during gear shifting, where the absolute standard deviation is higher because of different gear shifting times and durations of the drivers.

Interestingly Figure 8 reveals that the mean maximal acceleration drops until it reaches a plateau with the fourth and following vehicles. This corresponds to the time headways, shown in Figure 10a), which also drop significantly until the fourth vehicle passes the intersection.

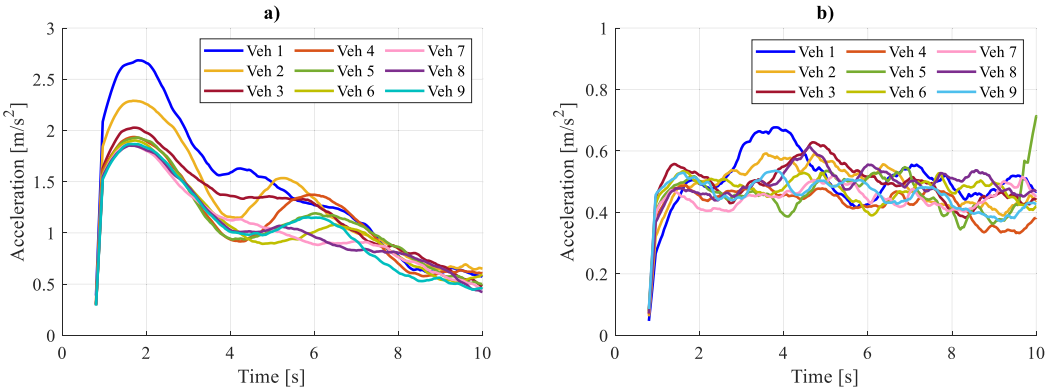

Figure 8: a) Mean acceleration curves of the first 9 vehicles in the queue from the aerial measurement data and **b)** their corresponding absolute standard deviation

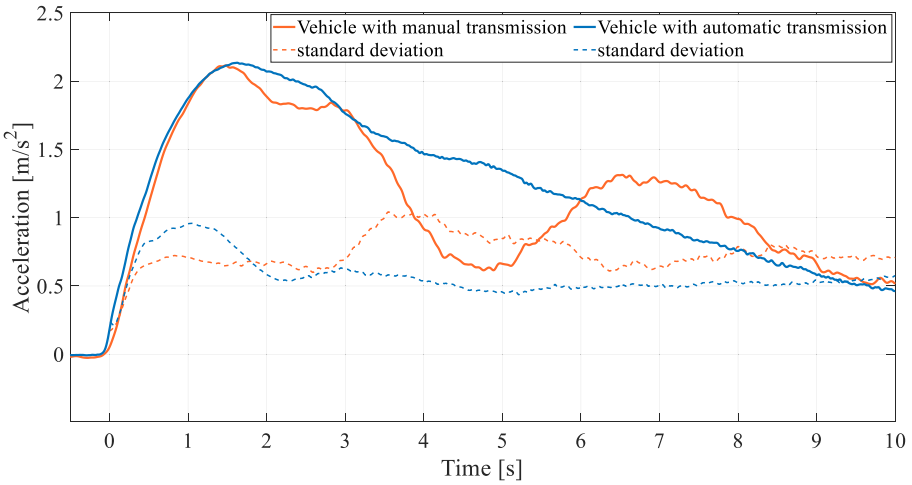

Figure 9: Mean acceleration (solid lines) and absolute standard deviation (dashed lines) curves of vehicles with a manual and an automatic transmission when driving off

4.5 Simulation

To set up the simulation environment delineated in Figure 3, a respective map is extracted from OpenStreetMap and converted to a SUMO network. After manual changes to the map, such as changing the traffic light program to the one used that day and insuring the correct edge and lane configurations, the traffic flow is inserted by defining one flow passing the intersection in the direction of the above mentioned arrows. The flow is defined in a manner to ensure that all vehicles passing the traffic light during a green phase have before come to a complete stop. The simulation duration is identical to that of the real-world measurement (30 green phases in 35 minutes).

Apart from the car-following model, the selection of the specific parameter set for each vehicle has a major effect on the simulation. As an extensive parameter identification is not part of this publication, we run simulations with each of the three car-following models. Every simulation contains four different sets of vehicle parameters, distributed evenly throughout the simulation. Table 2 shows these sets, where parameters existing in each model are varied. Specific parameters for the EIDM, listed in Table 3, are identical for all vehicles and are taken from literature or, in the case of newly introduced parameters, derived from first empirical observations. Lastly, SUMO version 1.0.1 is used with a time step Δt of 0.1 s for the Intelligent Driver Models and 0.5 s for the Krauss Model.

CF-Model	*Length [m]*	*MinGap [m]*			*speedFactor [-]*				*Accel [m/s²]*	*Decel [m/s²]*	*Tau [s]*	*ActionStep [s]*	δ [-]
		Mean	*Min*	*Max*	*Mean*	*Dev*	*Min*	*Max*					
Krauss Model	3	2.5	2.5	2.5	1.1	0.2	0.9	1.4	2.25	1.75	1.1	0.5	-
	4								2.50	2.00	1		
	4.5								2.75	2.25	0.9		
	5								3.00	2.50	0.8		
Intelligent Driver Model	3	2.5	2.5	2.5	1.1	0.2	0.9	1.4	2.25	2.5	1.1	0.1	4
	4								2.50	2.75	1		
	4.5								2.75	3.00	0.9		
	5								3.00	3.25	0.8		
Extended Intelligent Driver Model	3	2.5	2.0	3.0	1.1	0.2	0.9	1.4	2.40	2.40	1.2	0.5	2
	4								2.70	2.70	1.1	0.5	
	4.5								3.00	3.00	1	0.4	
	5								3.50	3.50	0.9	0.4	

Table 2: Parameter sets for the SUMO simulations

CF-model	T_{prev} *[s]*	$\tilde{\tau}_{est}$ *[s]*	$\tilde{\tau}_{err}$ *[s]*	V_s *[-]*	σ_r *[-]*	σ_a *[-]*	c_{ACC} *[-]*	t_{amax} *[s]*	M_{fl} *[-]*	M_{bg} *[-]*	j_{max} *[m/s³]*
EIDM	4	10	3	0.1	0.02	0.1	0.99	1.2	2	0.7	3

Table 3: Additional parameters for the EIDM

4.6 Comparison between observations and simulations

This chapter provides a first brief comparison between the real-world data and the results of the simulations. As the model parameters are selected empirically, this section does not focus on the absolute differences between the models, but rather on model characteristics and shows the capability of the EIDM to replicate the observed acceleration curves.

Figure 10a) depicts the mean time headway of the simulated and real-world vehicles. The model results generally show good agreement with the observations. With more suitable parameter sets, both the IDM and the Krauss model could replicate the measured headways even more accurately. The advantage of the Extended IDM becomes evident, when we compare the mean speed of the vehicles in Figure 10b). While, in this study, the parameters of the IDM and the Krauss model could not be tuned to reproduce the observed velocity curve, the EIDM reflects the real-world behavior with the chosen parameter set. The velocity curve of the IDM simulation levels off due to its characteristic of never reaching the desired velocity: when increased in an effort to match the curve of the real-world data, the first vehicles drive unrealistically fast. The IIDM resolves this issue, but vehicles still start off fast, reach the saturation speed early and experience sudden acceleration changes when combined with reaction times. The Krauss model can only match the observed speed curve with low acceleration parameters, which stands in contrast to the measured acceleration curves and maximum values.

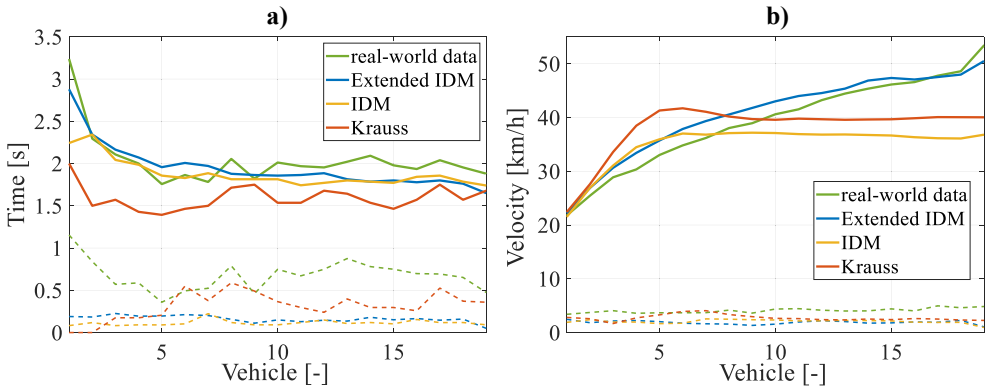

Figure 10: a) Mean time headways between vehicles (solid lines) and the absolute standard deviation of each sample (dashed lines) when passing the reference stop line, **b)** mean speed of vehicles (solid lines) and the absolute standard deviation of each sample (dashed lines) when passing the reference stop line

The combination of IIDM with reaction times and acceleration correction (Equation (19)) produces realistic values with respect to time headways, velocities and accelerations. The acceleration curves of the first 9 vehicles in the different simulations are shown in Figure 11, where each row features the results of the different models. Column a) shows the mean acceleration curves, b) the absolute standard deviation of each sample and c) the Mean Bias Error (MBE) between the model and the real-world measurement. The Mean Absolute Error (MAE) between the mean acceleration curves is calculated seperatly for each vehicle position.

This first analysis of the EIDM shows more accurate acceleration trajectories compared with the IDM and the Krauss model. However, the results of the Krauss model improve significantly when we cut off the first acceleration peak, illustrated in Figure 11a) third row, and delay the drive off to a later point in time.

The EIDM possesses a unique characteristic in that the mean maximal acceleration drops until it reaches a plateau with approximately the fourth vehicle. Still, the observed vehicles reach their maximal acceleration later than those modeled by the EIDM. Consequently, the parameters of the correction term need to be slighty adjusted. Increasing the amount and distribution of the parameter sets can further improve the model.

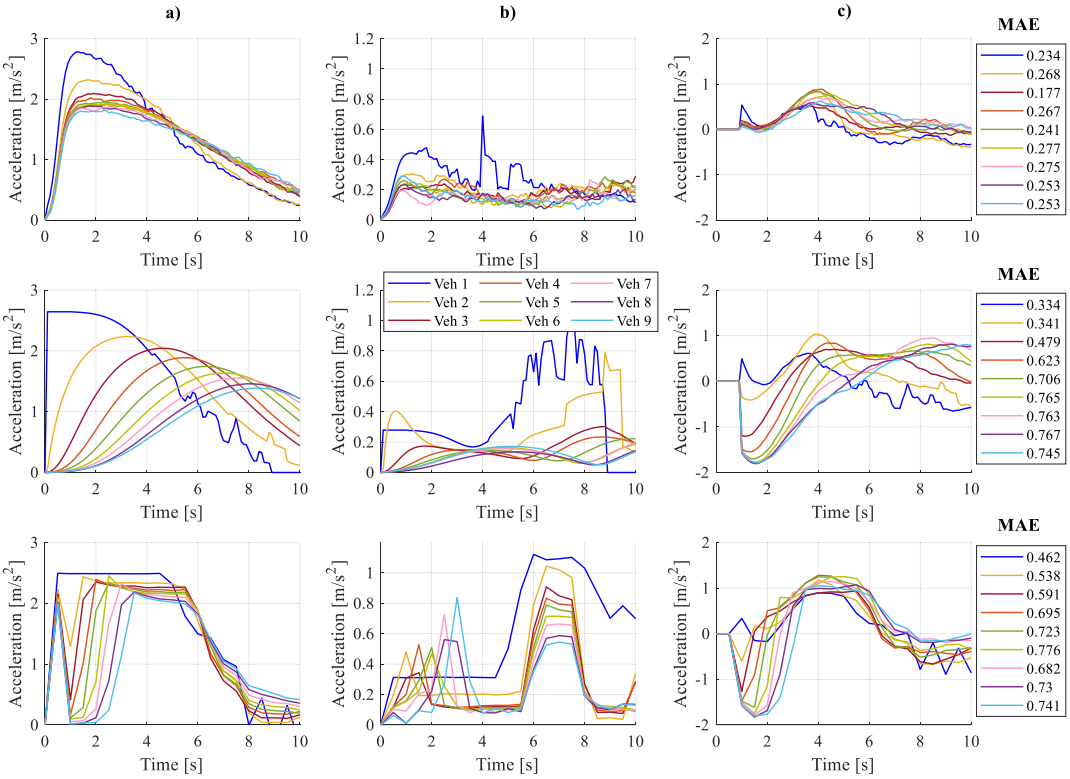

Figure 11: a) Mean acceleration curves of the first 9 vehicles in the queue from the simulation with the EIDM (first row), the IDM (second row) and the Krauss Model (third row), **b)** the corresponding absolute standard deviation of each sample, **c)** the Mean Bias Error between the real-world mean accelerations of the queued vehicles and the simulation results, depicted with their respected Mean Absolute Error values over the first 9 s of drive off (disregarding the first second, as the movement can't be correctly detected there)

In the past, different parameter identification techniques have been introduced to calibrate the IDM using floating car data, but that usually results in low a_{max} values, which do not agree with the data presented in this study.

Table 4 lists such calibrated parameter sets.

Kovács et al. (2016) calibrate the IDM to obtain accurate flow capacities and time headways at signalized intersections. They use a reliable method to calculate the IDM parameters based on the time headway saturation. On the negative side, the resulting a_{max} value and the desired time headway are rather low for urban traffic flow (Schulz 2013).

In summary, there are many models that can accurately describe time headways and speeds at intersections (Le Vine et al. 2016; Dey et al. 2013; James A. Bonneson; Li and Chen 2017; Jumsan KIM et al. 2005). But they can generally only calculate the time headway, velocity and acceleration at specific points and do not generate trajectories as car-following models can.

Parameter	EIDM	Kovács et al. (2016)	Kesting and Treiber (2008)	Treiber and Kesting (2013b)	Dallmeyer (2014)	Treiber and Kesting (2013a)
a_{max} [m/s^2]	2.5-3.5	1.6	1.5-1.6	1.0	1.5	1.4
$\bar{v}_0$ [m/s]	13.89	15.28	16.1	15	13.89	16.1
T [s]	1.1-1.3	0.86	1.3-1.4	1.0	1.5	1.2
s_0 [m]	1.5-2.5	2	1-1.6	2.0	2.0	1.5
b [m/s^2]	2.5-3.5	5	0.6-0.75	1.5	2.0	0.65
δ [-]	2	4	-	4	4	-

Table 4: IDM parameters used in literature compared to those in this study

5 Discussion and conclusions

In this paper we extended the IDM in SUMO to replicate human driver behavior. First, previously published IDM enhancements were introduced and discussed, as they have not yet been integrated in SUMO. Additional enhancements of the IDM to account for discontinuities in SUMO's time-discrete simulation were presented, with a special emphasis on the drive off process.

The drive off of the EIDM was validated using a drone dataset. The vehicle trajectories of the drone video were extracted using a Faster R-CNN with Reset101. The specific measurement method was outlined and the accuracy of the method was evaluated. The errors were determined to be negligibly small.

Nevertheless, the car-following model was only evaluated with a small amount of data in a specific environment: Drive off procedures at a saturated junction. The model will still need to prove that it can realistically reproduce other driving situations. In addition, only mean and characteristic values were used to compare the observed data with the simulation. The parameter set in this study is not comprehensive enough to replicate the standard deviations and typical log-normal distributions (Jin et al. 2009) during drive off. A detailed parameter identification for each vehicle could lead to a better agreement of modeled and observed behavior. By introducing the new drive off equation, the parameters identifying the drive off process can be separated from those characterizing vehicle following situations.

Additional variables could further increase the precision and influence of the individual driver by including road slope (Schulz 2013), spatial anticipation (Chen et al. 2009) or using action points dependent on the driving situation. Another factor to be considered are aerodynamics, which limit the maximal acceleration at high speeds and could explain why the IDM-parameter $a_{\max}$ is small when calibrating the model using highway traffic data.

6 Future work

Future work includes the integration of SUMO into an Unreal Engine driving simulator environment. Such linking has already been performed using Unity3D (Biurrun-Quel et al. 2017). The ultimate objective is the integration of simulative constructed drive trains (Ebel et al. 2017) into the environment, as done in DYNA4 (Kaths et al. 2019) and by Riegl et al. (2019). Additional research is needed to investigate the positive effects of more accurate acceleration patterns on energy consumption and emission calculations.

The EIDM will benefit from additional development with respect to cooperative lane changes. We plan to integrate the model into the current SUMO version.

7 Acknowledgment

This study was partially funded by the Robert Bosch GmbH. The authors would like to thank Helix4Motion for providing the drone videos and especially the German Aerospace Center and all SUMO contributors for creating such a versatile Open Source mobility platform.

References

Alazzawi, Sabina; Hummel, Mathias; Kordt, Pascal; Sickenberger, Thorsten; Wieseotte, Christian; Wohak, Oliver (2018): Simulating the Impact of Shared, Autonomous Vehicles on Urban Mobility – a Case Study of Milan. In. SUMO 2018- Simulating Autonomous and Intermodal Transport Systems: EasyChair (EPiC Series in Engineering), 94-76.

Alekszejenkó, Levente; Dobrowiecki, Tadeusz P. (2019): SUMO Based Platform for Cooperative Intelligent Automotive Agents. In. SUMO User Conference 2019: EasyChair (EPiC Series in Computing), pp. 107–189.

Bando; Hasebe; Nakayama; Shibata; Sugiyama (1995): Dynamical model of traffic congestion and numerical simulation. In *Physical review. E, Statistical physics, plasmas, fluids, and related interdisciplinary topics* 51 (2), pp. 1035–1042. DOI: 10.1103/PhysRevE.51.1035.

Biurrun-Quel, Carlos; Serrano-Arriezu, Luis; Olaverri-Monreal, Cristina (2017): Microscopic Driver-Centric Simulator: Linking Unity3D and SUMO. In Álvaro Rocha, Ana Maria Correia, Hojjat Adeli, Luís Paulo Reis, Sandra Costanzo (Eds.): Recent Advances in Information Systems and Technologies, vol. 569. Cham: Springer International Publishing (Advances in Intelligent Systems and Computing), pp. 851–860.

Bochinski, Erik; Eiselein, Volker; Sikora, Thomas (2017): High-Speed tracking-by-detection without using image information. In. 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS). Lecce, Italy, 2017: IEEE, pp. 1–6.

Bock, Julian; Krajewski, Robert; Moers, Tobias; Runde, Steffen; Vater, Lennart; Eckstein, Lutz (2019): The inD Dataset: A Drone Dataset of Naturalistic Road User Trajectories at German Intersections. Available online at http://arxiv.org/pdf/1911.07602v1.

Chen, Xiqun; Li, Ruimin; Xie, Weijun; Shi, Qixin (2009): Stabilization of traffic flow based on multi-anticipative intelligent driver model. In. 12th International IEEE Conference on Intelligent Transportation Systems (ITSC): IEEE, pp. 1–6.

Coifman, Benjamin; Li, Lizhe (2017): A critical evaluation of the Next Generation Simulation (NGSIM) vehicle trajectory dataset. In *Transportation Research Part B: Methodological* 105, pp. 362–377. DOI: 10.1016/j.trb.2017.09.018.

Dallmeyer, Jörg (2014): Simulation des Straßenverkehrs in der Großstadt. Das Mit- und Gegeneinander verschiedener Verkehrsteilnehmertypen. Teilw. zugl.: Frankfurt, Univ., Diss., 2013 u.d.T.: Akteursorientierte multimodale Straßenverkehrssimulation. Wiesbaden: Springer Vieweg (Research).

Derbel, Oussama; Mourllion, Benjamin; Basset, Michel (2012): Extended safety descriptor measurements for relative safety assessment in mixed road traffic. In. 15th International IEEE Conference on Intelligent Transportation Systems - (ITSC 2012): IEEE, pp. 752–757.

Dey, Partha; Nandal, S.; Kalyan, R. (2013): Queue discharge characteristics at signalised intersections under mixed traffic conditions. In *European Transport - Trasporti Europei*.

Donateo, Teresa; Pacella, Damiano; Laforgia, Domenico (2010): Simulation and Optimization of the Energy Management of ITAN500 in the SUMO Traffic Model Environment. In.

Ebel, André; Baumgartner, Edwin; Orner, Markus; Reuss, Hans-Christian (2017): Bewertung simulativ ausgelegter Antriebsstränge am Stuttgarter Fahrsimulator. In *MTZ Extra* 22 (S1), pp. 40–43. DOI: 10.1007/s41490-017-0004-9.

Ejercito, Paolo M.; Nebrija, Kristine Gayle E.; Feria, Rommel P.; Lara-Figueroa, Ligaya Leah (2017): Traffic simulation software review. In. 8th International Conference on Information, Intelligence, Systems & Applications (IISA): IEEE, pp. 1–4.

Erdağı, İsmet Gökşad; Silgu, Mehmet Ali; Çelikoğlu, Hilmi Berk (2019): Emission Effects of Cooperative Adaptive Cruise Control: A Simulation Case Using SUMO. In. SUMO User Conference 2019: EasyChair (EPiC Series in Computing), 92-82.

Erdmann, Jakob (2014): Lane-changing model in SUMO. In, vol. 24.

Fried, Oliver (2004): Betriebsstrategie für einen Minimalhybrid-Antriebsstrang. Aachen: Shaker (Berichte aus der Fahrzeugtechnik).

Gabb, Michael; Digel, Holger; Muller, Tobias; Henn, Rudiger-Walter (2019): Infrastructure-supported Perception and Track-level Fusion using Edge Computing. In. IEEE Intelligent Vehicles Symposium (IV). Paris, France, 2019: IEEE, pp. 1739–1745.

Gipps, P. G. (1981): A behavioural car-following model for computer simulation. In *Transportation Research Part B: Methodological* 15 (2), pp. 105–111. DOI: 10.1016/0191-2615(81)90037-0.

Grumert, Ellen; Ma, Xiaoliang; Tapani, Andreas (2015): Analysis of a cooperative variable speed limit system using microscopic traffic simulation. In *Transportation Research Part C: Emerging Technologies* 52, pp. 173–186. DOI: 10.1016/j.trc.2014.11.004.

He, Kaiming; Zhang, Xiangyu; Ren, Shaoqing; Sun, Jian (2016): Deep Residual Learning for Image Recognition. In. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, NV, USA: IEEE, pp. 770–778.

James A. Bonneson: Modeling Queued Driver Behavior at Signalized Junctions. In : Transportation Research Record, vol. 1365, pp. 99–107.

Jiang, R.; Wu, Q.; Zhu, Z. (2001): Full velocity difference model for a car-following theory. In *Physical review. E, Statistical, nonlinear, and soft matter physics* 64 (1 Pt 2), p. 17101. DOI: 10.1103/PhysRevE.64.017101.

Jin, Xuexiang; Zhang, Yi; Wang, Fa; Li, Li; Yao, Danya; Su, Yuelong; Wei, Zheng (2009): Departure headways at signalized intersections: A log-normal distribution model approach. In *Transportation Research Part C: Emerging Technologies* 17 (3), pp. 318–327. DOI: 10.1016/j.trc.2009.01.003.

Jumsan KIM; Zunhwan HWANG; Sungmo RHEE (2005): Vehicle Passing Behavior Through The Stop Line of Signalized Intersection. In *Journal of the Eastern Asia Society for Transportation Studies* 6, pp. 1509–1517. DOI: 10.11175/easts.6.1509.

Kaths, Jakob; Schott, Benedikt; Chucholowski, Frederic (2019): Co-simulation of the virtual vehicle in virtual traffic considering tactical driver decisions. In. SUMO User Conference 2019: EasyChair (EPiC Series in Computing), 21-12.

Kaufmann, Stefan; Kerner, Boris S.; Rehborn, Hubert; Koller, Micha; Klenov, Sergey L. (2018): Aerial observations of moving synchronized flow patterns in over-saturated city traffic. In *Transportation Research Part C: Emerging Technologies* 86, pp. 393–406. DOI: 10.1016/j.trc.2017.11.024.

Kesting, Arne; Treiber, Martin (2008): Calibrating Car-Following Models by Using Trajectory Data. In *Transportation Research Record* 2088 (1), pp. 148–156. DOI: 10.3141/2088-16.

Kesting, Arne; Treiber, Martin; Helbing, Dirk (2010): Enhanced intelligent driver model to access the impact of driving strategies on traffic capacity. In *Philosophical transactions. Series A, Mathematical, physical, and engineering sciences* 368 (1928), pp. 4585–4605. DOI: 10.1098/rsta.2010.0084.

Kotusevski, G.; Hawick, Ken (2009): A Review of Traffic Simulation Software. In *Res. Lett. Inf. Math. Sci.* 13.

Kovács, Tamás; Bolla, Kálmán; Gil, Rafael Alvarez; Csizmás, Edit; Fábián, Csaba; Kovács, Lóránt et al. (2016): Parameters of the intelligent driver model in signalized intersections. In *Teh. vjesn.* 23 (5). DOI: 10.17559/TV-20140702174255.

Kovvali, Vijay Gopal; Alexiadis, Vassili; Zhang P.E., Lin: Video-Based Vehicle Trajectory Data Collection. In : Transportation Research Board 86th Annual Meeting.

Krajewski, Robert; Bock, Julian; Kloeker, Laurent; Eckstein, Lutz (2018): The highD Dataset: A Drone Dataset of Naturalistic Vehicle Trajectories on German Highways for Validation of Highly Automated Driving Systems. In. 21st International Conference on Intelligent Transportation Systems (ITSC): IEEE, pp. 2118–2125.

Krauss, S.; Wagner, P.; Gawron, C. (1997): Metastable states in a microscopic model of traffic flow. In *Physical review. E, Statistical physics, plasmas, fluids, and related interdisciplinary topics* 55 (5), pp. 5597–5602. DOI: 10.1103/PhysRevE.55.5597.

Le Vine, Scott; Liu, Xiaobo; Zheng, Fangfang; Polak, John (2016): Automated cars: Queue discharge at signalized intersections with 'Assured-Clear-Distance-Ahead' driving strategies. In *Transportation Research Part C: Emerging Technologies* 62, pp. 35–54. DOI: 10.1016/j.trc.2015.11.005.

Li, Li; Chen, Xiqun (2017): Vehicle headway modeling and its inferences in macroscopic/microscopic traffic flow theory: A survey. In *Transportation Research Part C: Emerging Technologies* 76, pp. 170–188. DOI: 10.1016/j.trc.2017.01.007.

Lopez, Pablo Alvarez; Wiessner, Evamarie; Behrisch, Michael; Bieker-Walz, Laura; Erdmann, Jakob; Flotterod, Yun-Pang et al. (2018): Microscopic Traffic Simulation using SUMO. In. 21st International Conference on Intelligent Transportation Systems (ITSC): IEEE, pp. 2575–2582.

Macedo, Jose; Kokkinogenis, Zafeiris; Soares, Guilherme; Perrotta, Deborah; Rossetti, Rosaldo J. F. (2013): A HLA-based multi-resolution approach to simulating electric vehicles in simulink and SUMO. In. 16th International IEEE Conference on Intelligent Transportation Systems - (ITSC 2013): IEEE, pp. 2367–2372.

Milanés, Vicente; Shladover, Steven E. (2014): Modeling cooperative and autonomous adaptive cruise control dynamic responses using experimental data. In *Transportation Research Part C: Emerging Technologies* 48, pp. 285–300. DOI: 10.1016/j.trc.2014.09.001.

Nagel, Kai; Schreckenberg, Michael (1992): A cellular automaton model for freeway traffic. In *J. Phys. I France* 2 (12), pp. 2221–2229. DOI: 10.1051/jp1:1992277.

Pfeil, Raphael (2019): Methodischer Ansatz zur Optimierung von Energieladestrategien für elektrisch angetriebene Fahrzeuge (Wissenschaftliche Reihe Fahrzeugtechnik Universität Stuttgart). Available online at https://doi.org/10.1007/978-3-658-25863-4.

Queck, Tobias; Schüenemann, Björn; Radusch, Ilja (2008): Runtime infrastructure for simulating vehicle-2-x communication scenarios. In Varsha Sadekar, Paolo Santi, Yih-Chun Hu, Martin Mauve (Eds.): Proceedings of the fifth ACM international workshop on VehiculAr Inter-NETworking - VANET '08. the fifth ACM international workshop. San Francisco, California, USA, 15.09.2008 - 15.09.2008. New York, New York, USA: ACM Press, p. 78.

Ren, Shaoqing; He, Kaiming; Girshick, Ross; Sun, Jian (2015): Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. Available online at http://arxiv.org/pdf/1506.01497v3.

Richter, Gerald; Grohmann, Lukas; Nitsche, Philippe; Lenz, Gernot (2019): Anticipating Automated Vehicle Presence and the Effects on Interactions with Conventional Traffic and Infrastructure. In. SUMO User Conference 2019: EasyChair (EPiC Series in Computing), 230-215.

Riegl, Peter; Gaull, Andreas; Beitelschmidt, Michael (2019): A tool chain for generating critical traffic situations for testing vehicle safety functions. In. IEEE International Conference on Vehicular Electronics and Safety (ICVES): IEEE, pp. 1–6.

Rumbolz, Philip; Piegssa, Anne; Reuss, Hans-Christian (2010): Messung der Fahrzeug-internen Leistungsflüsse und der diese beeinflussenden Größen im 'real-life' Fahrbetrieb. In *VDI-Berichte 2105*, pp. 175–188.

Saifuzzaman, Mohammad; Zheng, Zuduo (2014): Incorporating human-factors in car-following models: A review of recent developments and research needs. In *Transportation Research Part C: Emerging Technologies* 48, pp. 379–403. DOI: 10.1016/j.trc.2014.09.008.

Schulz, Ralph (2013): Blickverhalten und Orientierung von Kraftfahrern auf Landstraßen. 1. Aufl. Dresden: TU, Lehrstuhl Gestaltung von Straßenverkehrsanlagen (Schriftenreihe des Lehrstuhls Gestaltung von Straßenverkehrsanlagen, H. 10).

Sommer, Christoph; Yao, Zheng; German, Reinhard; Dressler, Falko (2008): On the need for bidirectional coupling of road traffic microsimulation and network simulation. In Minkyong Kim,

Cecilia Mascolo, Mirco Musolesi (Eds.): Proceeding of the 1st ACM SIGMOBILE workshop on Mobility models - MobilityModels '08. Proceeding of the 1st ACM SIGMOBILE workshop. Hong Kong, Hong Kong, China, 26.05.2008 - 26.05.2008. New York, New York, USA: ACM Press, p. 41.

Thiemann, Christian; Treiber, Martin; Kesting, Arne (2008): Estimating Acceleration and Lane-Changing Dynamics from Next Generation Simulation Trajectory Data. In *Transportation Research Record* 2088 (1), pp. 90–101. DOI: 10.3141/2088-10.

Treiber; Hennecke; Helbing (2000): Congested traffic states in empirical observations and microscopic simulations. In *Physical review. E, Statistical physics, plasmas, fluids, and related interdisciplinary topics* 62 (2 Pt A), pp. 1805–1824. DOI: 10.1103/PhysRevE.62.1805.

Treiber, Martin; Helbing, Dirk (2004): Visualisierung der fahrzeugbezogenen und verkehrlichen Dynamik mit und ohne Beeinflussungs-Systemen. In, pp. 323–334.

Treiber, Martin; Kanagaraj, Venkatesan (2015): Comparing numerical integration schemes for time-continuous car-following models. In *Physica A: Statistical Mechanics and its Applications* 419, pp. 183–195. DOI: 10.1016/j.physa.2014.09.061.

Treiber, Martin; Kesting, Arne (2013a): Microscopic Calibration and Validation of Car-Following Models – A Systematic Approach. In *Procedia - Social and Behavioral Sciences* 80, pp. 922–939. DOI: 10.1016/j.sbspro.2013.05.050.

Treiber, Martin; Kesting, Arne (2013b): Traffic flow dynamics. Data, models and simulation. Heidelberg, New York: Springer.

Treiber, Martin; Kesting, Arne (2017): The Intelligent Driver Model with Stochasticity -New Insights Into Traffic Flow Oscillations. In *Transportation Research Procedia* 23, pp. 174–187. DOI: 10.1016/j.trpro.2017.05.011.

Treiber, Martin; Kesting, Arne; Helbing, Dirk (2006): Delays, inaccuracies and anticipation in microscopic traffic models. In *Physica A: Statistical Mechanics and its Applications* 360 (1), pp. 71–88. DOI: 10.1016/j.physa.2005.05.001.

Wagner, C.; Salfeld, M.; Knoll, S.; Reuss, H.-C. (2010): Quantifizierung des Einflusses von ACC auf die CO2-Emissionen im kundenrelevanten Fahrbetrieb. In *Proceedings Stuttgart International Symposium, Nr. 10*.

Wagner, Peter; Lubashevsky, Ihor (2003): Empirical basis for car-following theory development.

Wegener, Axel; Piórkowski, Michał; Raya, Maxim; Hellbrück, Horst; Fischer, Stefan; Hubaux, Jean-Pierre (2008): TraCI: an interface for coupling road traffic and network simulators. In Aftab Ahmad, Arnold Bragg (Eds.): Proceedings of the 11th communications and networking simulation symposium on - CNS '08. the 11th communications and networking simulation symposium. Ottawa, Canada, 14.04.2008 - 17.04.2008. New York, New York, USA: ACM Press, p. 155.

Wiedemann, Rainer (1974): Simulation des Straßenverkehrsflusses. Hochschulschrift, Karlsruhe.

Xiao, Lin; Wang, Meng; Schakel, Wouter; van Arem, Bart (2018): Unravelling effects of cooperative adaptive cruise control deactivation on traffic flow characteristics at merging bottlenecks. In *Transportation Research Part C: Emerging Technologies* 96, pp. 380–397. DOI: 10.1016/j.trc.2018.10.008.

Xiao, Lin; Wang, Meng; van Arem, Bart (2017): Realistic Car-Following Models for Microscopic Simulation of Adaptive and Cooperative Adaptive Cruise Control Vehicles. In *Transportation Research Record* 2623 (1), pp. 1–9. DOI: 10.3141/2623-01.

Zhou, Mofan; Qu, Xiaobo; Jin, Sheng (2016): On the Impact of Cooperative Autonomous Vehicles in Improving Freeway Merging: A Modified Intelligent Driver Model-Based Approach. In *IEEE Trans. Intell. Transport. Syst.*, pp. 1–7. DOI: 10.1109/TITS.2016.2606492.