Action-points in human driving and in SUMO

Peter Wagner[1], Jakob Erdmann[1], and Ronald Nippold[1]

Institute of Transportation Systems, DLR, Berlin, Germany
peter.wagner@dlr.de,jakob.erdmann@dlr.de,ronald.nippold@dlr.de

Abstract

When following a vehicle, drivers change their acceleration at so called action-points (AP), and keep it constant in between them. By investigating a large data-set of car-following data, the state- and time-distributions of the APs is analyzed. In the state-space spanned by speed-difference and distance to the lead vehicle, this distribution of APs is mostly proportional to the distribution of all data-points, with small deviations from this. Therefore, the APs are not concentrated around certain thresholds as is claimed by psycho-physical car-following models. Instead, small distances indicate a slightly higher probability of finding an AP than is the case for large distances. A SUMO simulation with SUMO's implementation of the Wiedemann model confirms this view: the AP's of the Wiedemann model follow a completely different distribution than the empirical ones.

1 Introduction

Car-following (CF) models are around since the early 1950ies [18], and they had been developed ever since. They had their first hype in the early 1960ies [9], [8] where especially models that use a description based on differential equations (ODE – ordinary differential equations, DDE – delayed differential equations, or even stochastic SDE – stochastic differential equations) have been used. Also during this time, the first action-point (AP) based models had been introduced [21], [20]. Already this early work assumed that the APs are related to perception thresholds of the human driver, with the idea that by crossing such a threshold, an appropriate action is triggered. E.g., very small speed differences Δv to the vehicle in front are impossible to recognize by a human driver, once this speed difference crosses a critical value $\Delta v_c(g)$ (which depends on distance g, with larger distances making it more difficult to recognize a certain speed-difference). The AP-models have then been introduced in a much more refined form in the psycho-physical models [25], [7] that are being used in microscopic software packages (e.g. VISSIM). The reaction is then measurable by a fast change in the acceleration $a(t)$, where it is assumed that acceleration is constant when no AP is active.

When looking at car-following trajectory data such as those collected by Naturalistic Driving Studies (NDS) [5], [6], [2], [3], [1], [14], where data are typically resolved with a time-step size of $\Delta t = 0.1\,s$, and from the fact that vehicles are heavy objects it is clear that acceleration cannot truly jump. For driver assistant systems, as well as in recommendations about comfortable rides in public transit vehicles, typically there is a limit in the jerk j of the vehicle. The jerk is the time-derivative of the acceleration, i.e. $j = da(t)/dt$, and this is typically limited to values smaller than $|j| \leq 2.5\,m/s^3$. This is also true for the data-set used in this work. However, when analyzing these data it is not too difficult to find times where acceleration changes fast, and times where it is constant. So, it seems that human drivers do have the habit of "do nothing" as long as possible, and then change to what is needed to avoid a collision, or to avoid falling back to far behind.

The second important point is that the perception thresholds are most likely not precise curves $\Delta v(g)$ in $(\Delta v, g)$-space. They have to be understood in a probabilistic sense, where the probability to issue such an AP changes with the distance to this line, and the actual line e.g. marking the point where with certainty such an AP is issued.

Surprisingly, there is little work found by the author related to the details of these thresholds, and they are often ignored by most of the scientific literature on car-following. So, one may wonder, whether they are important at all. Not to be mis-understood: there is not the shadow of a doubt that human perception is limited, despite the fact, that the visual system of humans is capable of truly astonishingly feats. And therefore, perception threshold do exist. But it is quite another question whether they are needed to describe car-following behavior, and to what extend the behavior of human drivers is constrained by them. The driving style chosen by them might avoid them completely, and therefore be ignorant of these thresholds.

Let us note in passing, that the AP's also occur in the other output of a human driver, that is the steering. To the knowledge of these authors, this is not a very well researched area (see [13] for an example).

2 The data-analysis

The data used here are from the German project simTD [2]. This project was not a NDS in the strict sense, its goal was to look into car-to-car communication. For this purpose, about 100 cars had been instrumented with communication devices, and these devices collected anything that could be collected from the CAN-bus of the vehicles, together with the proper geo-location and GPS-time. So, some of the vehicles recorded the distance and speed-difference to the lead vehicle, some of them recorded the gas-pedal (throttle $\theta(t)$) and brake-pedal usage, and all of them recorded the speeds $v(t)$, accelerations (lateral, as well as longitudinal $a(t)$), and GPS positions as well as the speeds from the GPS, together with a measure of the GPS error. The recording was asynchronous and in different time granularity, it ranged from $\Delta t = 1\,s$ for the GPS readings, most variables had $\Delta t = 0.1\,s$, to even shorter intervals for the acceleration data ($\Delta t = 0.01\,s$). With the exception of the GPS-data, these data have been enforced for the following analysis to a common time-step size of $\Delta t = 0.1\,s$. If more than one value appeared in such a time-interval, only the average had been recorded together with a number telling how many values had been averaged.

The data had been collected in four months from September 2012 to December 2012 by about 1,000 volunteers who drove these vehicles around according to a certain protocol that was invented to maximize what can be learned from car-to-car communication. So, although the drivers where aware that they had been recorded, it was not done to look at their driving behavior.

To identify APs in the data set, several approaches had been tested (using the acceleration $a(t)$, the throttle position $\theta(t)$, or the speeds $v(t)$). It turns out, that the speed data $v(t)$ of each vehicle yield the clearest signal to find the APs, a result that has also been reported in [11]. Identification of APs has been done, then, by applying the Ramer-Douglas-Peucker (RDP) [17], [4] and the Visvalingam-Whyatt (VW) [22] algorithm to the time-series $\{t, v(t)\}_{t \geq 0}$. These algorithms have each one parameter that determines the degree of simplication to the time-series, and these have been chosen by try-and-error – there are no objective criteria that can be used here. (These algorithms have been invented originally to simplify geographical objects; there the goal is to eliminate detail but still keep the visual appearance e.g. of a coast line intact and recognizable.) The analysis has been done with the Python library simplification [12]. The rest of this analysis has been done with R [16], and the front-end RStudio [19].

The Figure 1 shows an example of the effect of the two simplification methods on these data. It could also been seen, that the VW algorithm seems to do a slightly better job, but this of course depends on the parameters chosen.

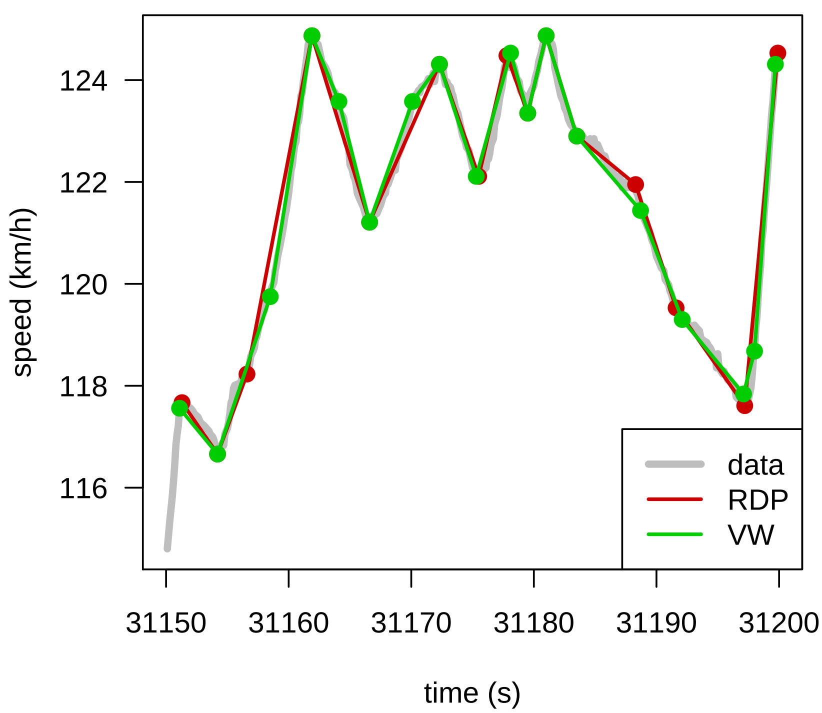

Figure 1: A short piece of the a speed time-series $v(t)$ (in gray), together with the results of the two simplification algorithms described in the text. RDP is in red, VW in green.

Note, that not all driving maneuvers allow for the proper assignment of APs. Especially periods of strong acceleration and deceleration often are more volatile, with acceleration changing faster than is visible in these data. However, car-following periods typically have small accelerations, and in these cases the method seems to work very well.

2.1 Distribution of the time intervals between APs

From such a division, the distribution of time-differences δt between subsequent APs can be computed as well, see Figure 2 for the result. Note, that due to holes in the data, some APs might have been missed. The distribution is compatible with a log-normal distribution. The two maxima are around a value of 1.5 s, while the mean values are about 3 s.

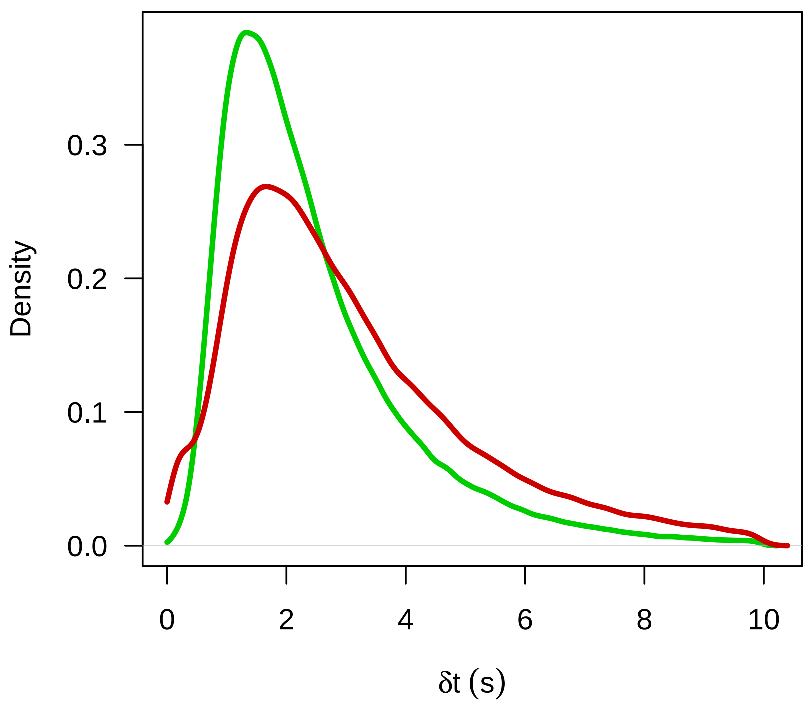

Figure 2: Probability density $p(\delta t)$ of the time-difference δt between subsequent APs. Red line is for the RDP, green line for the VW algorithm. Data are cut-off at ten seconds.

2.2 Car-following

So far, all data have been used for the analysis to state some basic facts about the APs. Now, a closer look at car-following episodes are performed. Car-following is identified as follows. First of all, there should be a data-point that has both distance and speed-difference to the vehicle in front. To give an estimate, this is the case for about 25% of the data. Additional tests have been applied to clean the data. The most important of them is to check for a dynamic consistency. Note $g(t)$ as the gap at time t, and $g(t+\Delta t)$ the gap at a (small) time-step Δt later. The speed-difference is named $\Delta v(t) = V(t) - v(t)$, where $V(t)$ is the speed of the leading vehicle. Then:

$$g(t + \Delta t) = g(t) + \Delta t \Delta v(t) + \mathcal{O}((\Delta t)^2) \tag{1}$$

The size of the $\mathcal{O}$-term is the difference between the acceleration of the lead and the following car, multiplied by the square of the time-step size. By assuming "normal" accelerations of $2.5\, m/s^2$ and a time-step size $\Delta t = 0.1$, the expected error here is of the order of $0.05m$. So, the gap error ε_g:

$$\varepsilon_g = g(t + \Delta t) - (g(t) + \Delta t \Delta v(t)) \tag{2}$$

can be used to filter out bad data-points. The same can be done with the speed and acceleration by defining a speed error ε_v:

$$\varepsilon_v = v(t + \Delta t) - (v(t) + \Delta t a(t)) \tag{3}$$

By setting a limit of 1 m for the gap-error and 0.25 m/s for the speed-error (the interquartile distance for the gap-error is of the order 0.2 m, and 0.05 m/s for the speed-error), about 25% of the data gets eliminated.

The remaining data are now being used to search for a difference between the distribution of the APs in the CF-plane $(\Delta v, g)$ and the distribution of all the data. This is done by sampling the data into two histograms $h(\Delta v, g)$ and $H(\Delta v, g)$ for the APs and all the remaining data. There are several means how two distributions can be compared. Here, a method is chosen that bears a strong resemblance with the well-known χ^2-test. It works as follows: let h_{ij} be the histogram sampled from the APs by a particular tesselation of the CF-plane, and H_{ij} the histogram for all the data-points, using the same tesselation of the CF-plane. Then, each bin in the AP-histogram is related to the same bin in the full histogram. The simplest assumption one may have is that the AP-histogram is just a down-scaled version of the full histogram, i.e. it is expected, that the AP histogram can be computed by the multiplication of the full histogram with an empirically defined factor, which for the data here turned out to be around $f = 0.05$:

$$\hat{h}_{ij} = f\, H_{ij} \tag{4}$$

Then, the Pearson residuum χ_{ij} can be defined:

$$\chi_{ij} = \frac{h_{ij} - \hat{h}_{ij}}{\sqrt{\hat{h}_{ij}}}. \tag{5}$$

Clearly, the sum over χ_{ij}^2 is just the χ^2 value. However, in the context here, more can be learned than the simple fact that these two distributions are different: by plotting $\chi(\Delta v, g)$ as function of $(\Delta v, g)$, the Figure 3 is found.

The result in Figure 3 displays no one-dimensional lines where the AP-distribution would have been larger. In general, the difference between the two distributions is weak, but it displays a clear pattern. For large distances, the AP density is smaller than what can be expected on the basis of Equation (5), while for smaller distances, drivers issue more APs. There is a slight asymmetry between positive and negative Δv, which is as expected: negative values of Δv belong to the dangerous area where the vehicle is approaching. However, the difference itself is weak, values of $|\chi| \sim 2$ correspond to a 5% error probability.

Note the unequal tiling of the plane: this has been chosen so that roughly the same number of data-points fall into each box in Δv as well as in g-direction. It improves the statistics, at

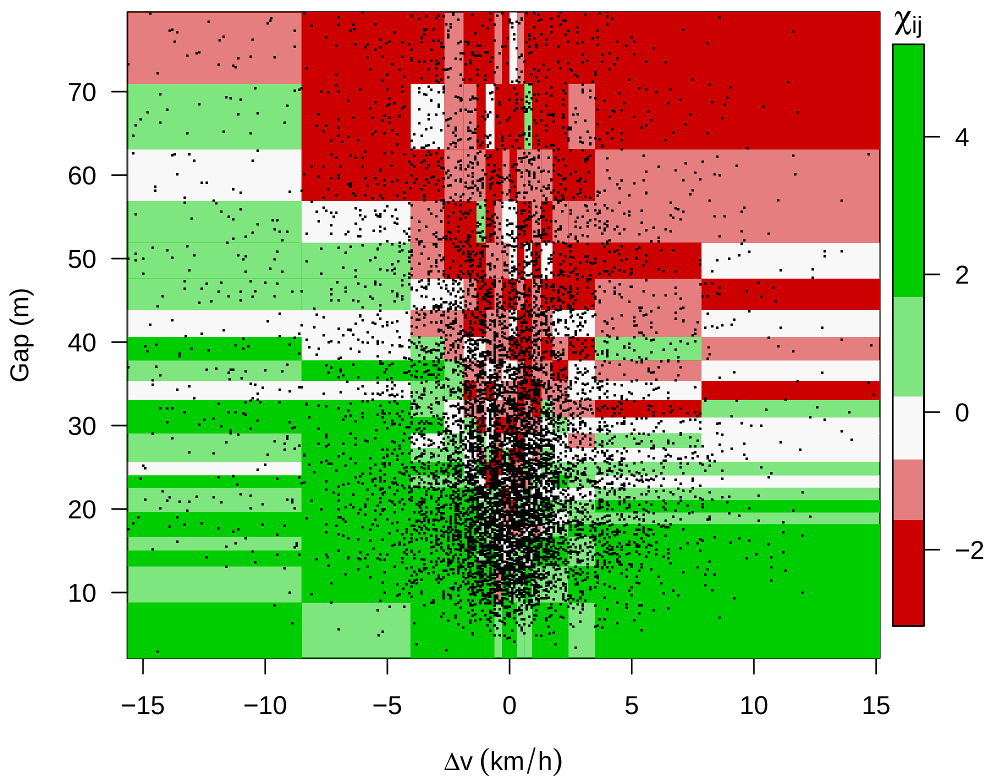

Figure 3: Distribution of the difference between the AP distribution and the distribution of the full frequency distribution $p(\Delta v, g)$. Note, that this plot is roughly divided into two areas: for large distances, less APs are issued, while for short distances, more APs are needed.

the expense of the accuracy in the location of the boxes. Small boxes corresponds to a large probability to find the system there, and in fact, that maximum of the $p()$-distribution is around $20\,m$. About 1.4M data-point had been used to compute this diagram, about 5% of those have been labelled as APs.

3 Running SUMO with Wiedemann's model

To use SUMO [15] to generate similar data, the following set-up has been used. A large rectangle of a one-lane road has been built with netedit, with changing speed limits to sample from different car-following regimes. Altogether six vehicles had been put to this network, at the start of the simulation they were at a standstill. The first (lead) vehicle was a SUMO default vehicle that drove with constant speed. Its speed-factor had been set to 0.5, so that is drives with half of the speed-limits on the four edges, i.e. at $35, 30, 25$, and $20\,m/s$. The five following cars were configured as

```
<vType id="followVIS" length="4.61" maxSpeed="70.0" minGap = "1.0">
<carFollowing-Wiedemann accel="1.8" decel="4.5" sigma="0.9"/>
</vType>
```

SUMO's own deterministic AP mechanism `default.action-step-length value="0.1"` had been set to the step-size of 0.1 s, to be close to the empirical data. The data had been sampled from the simulation by using the netstate dump mechanism `netstate-dump value="wiedemannAP.xml"` and subsequently analyzed with R [16]. AP's have been found by searching for points where the acceleration of the vehicle has changed by more than 0.1 m/s^2, the acceleration itself has been computed from the (recorded) speed by a simple difference scheme $a(t) = (v(t) - v(t - \Delta t))/\Delta t$. All the data from the five following cars had been used and analyzed together, as had been done with the simTD data. This yields the results in Figure 4.

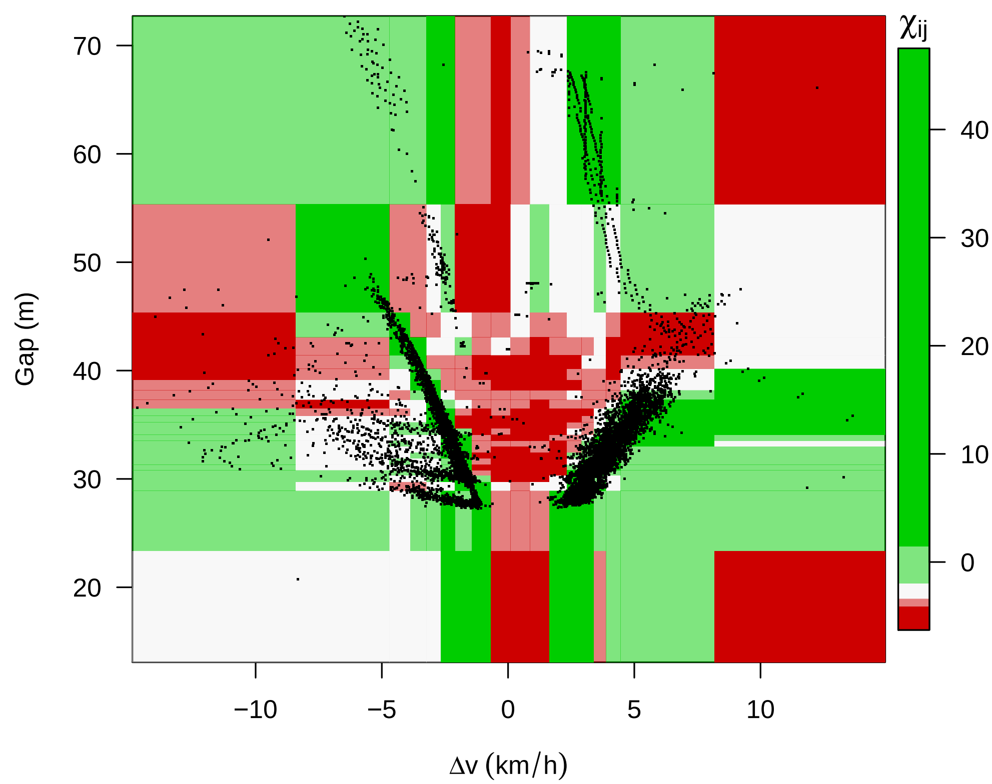

Figure 4: Distribution of the difference between the AP distribution and the distribution of the full frequency distribution $p(\Delta v, g)$ for the Wiedemann model, together with the AP's themselves.

The results in Figure 4 demonstrate that there is a strong difference between the APs generated by the Wiedemann model, and the APs identified in the empirical analysis. Nevertheless, these results are completely in line with what to expect from Wiedemann's model of the AP distribution: they are lined up in a scattered manner along the perception thresholds as defined in the Wiedemann model.

4 Conclusions

These results indicate, that human car-following is not controlled in any manner by perception thresholds. Similar results have also been found in [23] and [10]. The statistical analysis above demonstrates, that APs in fact have a non-trivial distribution: they are issued more often when the situation is dangerous, but the effect is not a very strong one.

There is still a (small) loop-hole in this analysis for the existence of perception thresholds: to gain statistical power, all the data for all the drivers have been put together, in this case these had been 96 drivers. The data from just one driver are not enough to find those patterns, since they do not happen that often, they are roughly 5% of the total amount of car-following data. So, an experimental design that would look explicitely for those thresholds would try to collect long car-following data from just one driver. However, we think it is highly unlikely for such an approach to succeed, since the distribution of APs found in the simTD data is so completely different from the one from the simulation with Wiedemann's model.

If this result holds true, it might be asked why do the thresholds do not play a prominent role. One of the answers comes from the average time between two APs of $1 \ldots 3$ seconds. Drivers correct their driving style much more often than what would be needed by the thresholds. Therefore, they do not take care of the thresholds. In addition, especially when in car-following, drivers are typically relaxed, at least this can be concluded when looking at the accelerations that are realized in this mode. And this means, that they are also not fully concentrated, and it might be assumed that their perception error is larger too. And again, this would lead to a smoothing out of any threshold.

Let us finally note that this does not mean that the Wiedemann model is a bad model. On average, the acceleration function $a(\Delta v, g, v)$ of this model does still the correct things. It indicates, however, that this model has a feature (the perception thresholds) that cannot be found in the simTD data. However, some papers have seen these thresholds (at least we have found this reference [24], but we remember to have seen others) and the associated increase in APs issued at these thresholds. It can only be speculated what has been seen in these data, and as mentioned already, it might well be that with a more careful preparation of the car-following experiments those thresholds can be seen.

References

[1] EuroFOT, 2012. URL: http://wiki.fot-net.eu/index.php?title=EuroFOT, last access 31.7.2019.

[2] Safe and intelligent mobility – test field germany, 2012. URL: http://www.simtd.de/index.dhtml/enEN/index.html, last access 7.7.2015.

[3] SHRP2 naturalistic driving study, 2019. URL: https://insight.shrp2nds.us/, last access 31.7.2019.

[4] David Douglas and Thomas Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *The Canadian Cartographer*, 10(2):112–122, 1973.

[5] P. Fancher, R. Ervin, J. Sayer, M. Hagan, S. Bogard, Z. Bareket, M. Mefford, and J. Haugen. *Intelligent Cruise Control Field Operational Test.* U.S. DOT, HS 808 849 edition, 1998. University of Michigan Transportation Research Institute.

[6] FHWA. Next Generation Simulation Program. http://ngsim.camsys.com/, 2006. accessed Sept. 2006.

[7] H.-T. Fritzsche. A model for traffic simulation. *Transportation Engineering And Control*, (5):317–321, 1994.

[8] Denos C. Gazis, Robert Herman, and Richard W. Rothery. Nonlinear follow-the-leader models of traffic flow. *Operations Research*, 9(4):545–567, 1961.

[9] R. Herman, E.W. Montroll, R.B. Potts, and R.W. Rothery. Traffic dynamics: Analysis of stability in car following. *Operations Research*, 7(1):86–106, 1959.

[10] Raymond G. Hoogendoorn, Serge P. Hoogendoorn, Karel A. Brookhuis, and Winnie Daamen. Longitudinal driving behavior under adverse conditions: A close look at psycho-spacing models. *Procedia - Social and Behavioral Sciences*, 20:536 – 546, 2011. The State of the Art in the European Quantitative Oriented Transportation and Logistics Research – 14th Euro Working Group on Transportation & 26th Mini Euro Conference & 1st European Scientific Conference on Air Transport.

[11] Serge Hoogendoorn, Raymond G. Hoogendoorn, and Winnie Daamen. Wiedemann revisited: New trajectory filtering technique and its implications for car-following modeling. *Transportation Research Record*, 2260(1):152–162, 2011.

[12] Stephan Hügel. simplification: Fast linestring simplification using RDP or visvalingam-whyatt and a Rust binary, 2016–. [Online; accessed 31 Aug 2018].

[13] Victor L. Knoop and Serge P. Hoogendoorn. Relation between longitudinal and lateral action points. In Mohcine Chraibi, Maik Boltes, Andreas Schadschneider, and Armin Seyfried, editors, *Traffic and Granular Flow '13*, pages 571–576, Cham, 2015. Springer International Publishing.

[14] Robert Krajewski, Julian Bock, Laurent Kloeker, and Lutz Eckstein. The highd dataset: A drone dataset of naturalistic vehicle trajectories on german highways for validation of highly automated driving systems. In *2018 IEEE 21st International Conference on Intelligent Transportation Systems (ITSC)*, 2018.

[15] Pablo Alvarez Lopez, Michael Behrisch, Laura Bieker-Walz, Jakob Erdmann, Yun-Pang Flötteröd, Robert Hilbrich, Leonhard Lücken, Johannes Rummel, Peter Wagner, and Evamarie Wießner. Microscopic traffic simulation using sumo. In *The 21st IEEE International Conference on Intelligent Transportation Systems*. IEEE, 2018.

[16] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2018.

[17] Urs Ramer. An iterative procedure for the polygonal approximation of plane curves. *Computer Graphics and Image Processing*, 1(3):244–256, 1972.

[18] A. Reuschel. Fahrzeugbewegung in der Kolonne bei gleichförmig beschleunigtem oder verzögertem Leitfahrzeug. *Zeitschrift des österreichischen Ingenieur und Architektenvereins*, page 95, 7/8 1950. In German.

[19] RStudio Team. *RStudio: Integrated Development Environment for R*. RStudio, Inc., Boston, MA, 2015.

[20] E. P. Todosiev. *The actionpoint model of driver vehicle system*. PhD thesis, The Ohio State University, Columbus, Ohio, USA, 1963.

[21] E. P. Todosiev and L. C. Barbosa. A proposed model for the driver-vehicle-system. *Traffic Engineering*, 34:17–20, 1963/64.

[22] Maheswari Visvalingam and James Duncan Whyatt. Line generalisation by repeated elimination of the smallest area. Technical report, Cartographic Information Systems Research Group, University of Hull, 1992.

[23] Peter Wagner. Empirical description of car-following. In Serge P. Hoogendoorn, Stefan Luding, and Piet H. L. Bovy, editors, *Traffic and Granular Flow '03*, pages 15–28. Springer, 2005.

[24] R. Wiedemann and U. Reiter. Microscopic traffic simulation: the simulation system mission, background and actual state. Technical report, Project ICARUS (V1052) Final Report. Brussels, CEC, 2.

[25] Rainer Wiedemann. Simulation des Straßenverkehrsflußes. Technical report, Institut für Verkehrswesen, Universität Karlsruhe, 1974. Heft 8 der Schriftenreihe des IfV, in German.