PedSUMO: Simulacra of Automated Vehicle-Pedestrian Interaction Using SUMO To Study Large-Scale Effects

Mark Colley  
mark.colley@uni-ulm.de  
Institute of Media Informatics, Ulm University  
Ulm, Germany

Julian Czymmeck  
Julian.czymmeck@uni-ulm.de  
Institute of Media Informatics, Ulm University  
Ulm, Germany

Mustafa Kücükkocak  
mustafa.kuecuekkocak@uni-ulm.de  
Institute of Media Informatics, Ulm University  
Ulm, Germany

Pascal Jansen  
pascal.jansen@uni-ulm.de  
Institute of Media Informatics, Ulm University  
Ulm, Germany

Enrico Rukzio  
enrico.rukzio@uni-ulm.de  
Institute of Media Informatics, Ulm University  
Ulm, Germany

ABSTRACT

As automated vehicles become more widespread but lack a driver to communicate in uncertain situations, external communication, for example, via LEDs or displays, is evaluated. However, the concepts are mostly evaluated in simple scenarios, such as one person trying to cross in front of one automated vehicle. The traditional empirical approach fails to study the large-scale effects of these in this not-yet-real scenario. Therefore, we built PedSUMO, an enhancement to SUMO for the simulacra of automated vehicles’ effects on public traffic, specifically how pedestrian attributes affect their respect for automated vehicle priority at unprioritized crossings. We explain the algorithms used and the derived parameters relevant to the crossing. We open-source our code under https://github.com/M-Colley/pedsumo and demonstrate an initial data collection and analysis of Ingolstadt, Germany.

CCS CONCEPTS

• Software and its engineering → Software design engineering.

1 BACKGROUND AND SUMMARY

Automated driving is a growing field of research [26], with fully Automated Vehicles (AVs) being part of current discussions and research [46]. AVs could provide numerous advantages, such as improving traffic flow [35]. However, these advantages are currently only theoretical. The consequences of introducing AVs in greater numbers into public traffic can only be estimated as conducting large-scale studies in public is impossible when the safety of AVs is not clear yet [63, 65]. Also, fear of AVs is still significant in the population [28, 50]. Additionally, measuring the impact of many AVs on public traffic in many different locations might be unrealistic or expensive. Thus, creating virtual scenarios to simulate how AVs impact public traffic is more feasible.

This project examines the macroscopic effects of AVs in traffic and how the respect of pedestrians towards AVs’ priority at crossings leads to different or fluctuating traffic flows. Currently, numerous research studies are concerned about whether AVs will have to be able to communicate with vulnerable road users such as pedestrians or cyclists [32]. When AVs are regularly stopped due to pedestrian behavior, this can ripple through traffic, slowing down the overall flow. The effect is stronger with an increasing number of AVs with an external Human-Machine Interface (eHMI) as an eHMI serves as a communication between the human and the vehicles,
contributes to a higher feeling of safety around AVs [4, 57]. The following provides background information about human behavior modeling, factors on crossing decisions, and eHMIs.

Attributes Influencing Street Crossing
Several attributes contribute to pedestrian street-crossing decisions, including other pedestrians’ behavior, group size, social status, and experience with AVs [15, 57]. Yagil [73] found that pedestrians are more likely to follow traffic laws when observing similar behavior from others. However, Lefkowitz et al. [45] demonstrated that this imitation is influenced by the appearance of the other pedestrian. Contrarily, Dolphin et al. [24] argued that social status and gender do not significantly impact imitation, emphasizing the role of group size instead. In line with the importance of group size, Heimstra et al. [31] showed that children often cross streets in groups, which influences their risk-taking behavior [30, 61, 64, 71]. Studying all these factors in an empirical study is nearly impossible, therefore, simulations are necessary.

External Communication of Automated Vehicles
Current human-driven vehicles often rely on gestures and eye contact for communication [56]. Although such explicit communication is infrequent [44], eHMIs have been proposed as a solution for AVs [32]. These eHMIs can be classified based on modality, message type, and communication location [11, 12].

Several studies have explored the effectiveness of eHMIs across different populations, including children [19], visually [13, 14] or cognitively [29] impaired individuals, general pedestrians [1, 5, 8–10, 17, 21, 47], manual drivers [7], and bicyclists [34]. Various modalities, such as displays [27], LED strips [27, 48], and auditory cues [49], have been tested. Overall, eHMIs have positively affected pedestrian behavior and comprehension [13, 20]. However, current research suggests the need to address unresolved questions such as overtrust [33], scalability [16], and the social aspects of eHMIs [5, 40, 58, 59]. A major limitation of these studies is the focus on simple scenarios, often resembling 1:1 (AV:pedestrian) communication. While Colley et al. [6] approached this with an online simulation studying the effect of multiple lanes and additional simulated pedestrians, large-scale analyses are missing.

Pedestrian Behavior Modeling
There exist several pedestrian simulation approaches. These can be distinguished into macroscopic or microscopic [55]. Microscopic refers to simulations where each actor is simulated instead of, for example, flows. SUMO [22] represents a possibility to simulate mobility on the microscopic level. While “there are good models for optimal walking behavior, high-level psychological and social modeling of pedestrian behavior still remains an open research question that requires many conceptual issues to be clarified” [3, p. 1]. Camara et al. [3] showed that algorithms used age, gender, distraction, social group membership, cultural membership, and road safety adaptation to model pedestrian behavior. While most works use a deterministic approach, Völz et al. [70] showed a model that predicts the crossing decision at a crosswalk using support vector machines. Due to the unavailability of actual AVs on the streets equipped with eHMIs, such approaches are infeasible.

In partially related HCI domains, Savino et al. [60] evaluated bicyclist strategies to reach a given destination. It evaluates the efficacy of As-the-crow-flies (ATCF) navigation for cyclists, focusing on how different street network attributes impact the user experience. Using feature importance analysis across 1,633 cities, the paper identifies that an ideal environment for ATCF navigation has long streets, multiple turning options, few dead ends, and a grid-like structure. East Asian and North American cities are most suited for this navigation method, while Western Europe’s street networks are least suited. For this, Savino et al. [60] simulated an agent using a modified depth-first search. Ikkala et al. [36] adopt a different method, biomechanically simulating a user’s entire body. While this is a more accurate representation of a user in physical terms, the applicability to large-scale analyses is not yet possible.

2 PURPOSE
Using the microscopic traffic simulation tool SUMO [22], we vary pedestrian attributes that affect decision-making, making them more or less likely to respect AV priority at crossings. Microscopic traffic flow models focus on individual road user units, thus representing dynamic variables such as the position and velocity of each vehicle and pedestrian. ProSUMO seeks to measure macroscopic changes in traffic flow using different variables for pedestrian decision-making (e.g., gender of pedestrians, street width, vehicle size) with different percentages of AVs (with eHMI) in traffic.

3 CHARACTERISTICS
After repository cloning, install the requirements detailed in the requirements.txt. If Large Language Models (LLMs) are to be used, the requirements_llm.txt must be installed. The requirements are minimal in addition to SUMO but require new versions for increased performance. If other cities than those provided are to be used, these must be downloaded and saved in the appropriate directory. We strongly encourage community input, either as comments, issues, or additional code in the GitHub repository.
4 CODE/SOFTWARE

4.1 Algorithms
The main idea of PenSUMO is to identify unprioritized crossings with pedestrians wanting to cross in each step of the simulation (see Figure 1). Additionally, the algorithm filters those for situations in which these pedestrians would not usually be able to cross due to an oncoming vehicle. If that oncoming vehicle is an AV, a chance for the waiting pedestrian to cross the road anyway and ignore the vehicle’s right of way is calculated.

To increase performance during simulation time, a dictionary of all incoming lanes into each unprioritized crossing in the simulation (see algorithm 1) is created when the scenario is selected. To achieve this, the successor of each lane in the network is evaluated. If the successor is an internal foe of an unprioritized crossing, the original lane is added to the set of lanes of the associated crossing.

After the incoming lanes dictionary (see algorithm 1) is created, the main simulation loop (see algorithm 4) starts. This simulation loop runs until the pre-configured last simulation step (default = 3600 or 1h) is reached. At the start of each step, the terminated entities of the previous step are cleaned up, and newly added entities are adjusted. That includes assigning attributes such as age and gender to pedestrians and declaring vehicles as automated or manual. Afterward, every pedestrian’s intent is evaluated. If a pedestrian intends to walk onto an unprioritized crossing as their next lane, this pedestrian is added to a list of waiting pedestrians for that crossing.

For each of these crossings, it is then determined whether the current situation is an av_crossing_scenario (see algorithm 2). That is the case whenever a pedestrian would not usually be able to cross the road due to an oncoming vehicle, but that vehicle is marked as an AV. On the side, the closest vehicle and its time to collision and distance to the crossing are calculated for future use.

If the situation is an av_crossing_scenario, the crossing probability is calculated. To avoid redundancy, all defiance factors specific to the crossing, such as street_width_defiance_factor C.4 or the vehicle_size_defiance_factor C.6, are calculated. Then, for each pedestrian wanting to cross the evaluated crossing, their individual defiance factors, such as the waiting_time_defiance_factor C.1, are calculated. Section C lists the full list of factors and their calculation.

The total crossing probability is then calculated by multiplying each factor with the base_automated_vehicle_defiance. The decision to cross is simulated by comparing this probability with a random number. If the pedestrian “decides” to cross, they are set to ignore all vehicles until they completely cross the crossing. Additionally, the danger of the situation is evaluated (see algorithm 3). This is done by calculating and then comparing the minimal stopping distance of the closest incoming vehicle in terms of time to collision with its distance to the crossing. If the stopping distance is larger than the vehicle’s distance to the crossing, the situation is deemed dangerous.

Our implementation also allows the use of different LLMs provided by the HuggingFace transformers library [72] to identify potentially realistic behavior (see Park et al. [51]). Therefore, a prompt given the scenario values could start with:

*You are a pedestrian. You are standing at a street with some automated vehicles trying to decide whether you will cross it. You are distracted by your smartphone. There are no children in your vicinity. The approaching automated vehicle has an interface attached that communicates with you. You are not walking. The street is five meters wide. The vehicle has a front area of three square meter. [...]*

After each crossing is evaluated, pedestrians who were altered in previous steps to ignore vehicles and successfully crossed their crossing get their alterations reset, and the next simulation step can begin. The usage of LLMs depends on the size of the Video Random Access Memory (VRAM) available and the chosen model. We suggest using 12GB VRAM or more.

4.2 Simulated Pedestrian Crossing Factors
Adjustable factors are diverse and have a different impact by default. Table 1 shows a description of each factor with the corresponding source for reference: The relevant formulae determining the distribution of probabilities are described in Section C.

4.3 Measurements/Logging
In addition to SUMO’s standard output (see [23]), we log the parameters shown in Table 2 in a CSV file. Each crossing event has all factors listed that are explained in section 4.2, including defiance values and their impact during the crossing event. Additionally, the static percentage of AVs (with eHMI) in all vehicles in traffic and the following data are logged in this file for every crossing event. These can, as such, easily be used as independent variables.

5 USAGE NOTES
While SUMO generally allows the use of an OpenStreetMap (OSM) integration to simulate road networks, these often have to be fine-tuned due to errors. Therefore, we provide already curated scenarios in Ingolstadt, Wildau, Monaco, and Bologna. Additionally available for simulation are Ulm and Manhattan, which were generated and adapted using SUMO’s OSMWebWizard.

While the current implementation is based on the scientific literature, we highlight that the simulation cannot necessarily be seen as a true representation of the interaction between an AV and pedestrians. However, in line with Park et al. [51], the simulacra of human behavior with PenSUMO can generate insights that plausibly define future behavior. This is currently the most appropriate avenue to study large-scale effects of eHMI and AVs on traffic flow.

AVs represent a specific manifestation of robots and are, therefore, directly relevant to the HRI community (e.g., see [2, 42, 43, 53]). However, the current implementation can also serve as a basis for including simulated robots in communication with pedestrians. This is currently researched in the CHI and HRI community [52].

6 EVALUATION
As we were interested in the large-scale effects of AVs and eHMIs on traffic, we simulated Ulm, Ingolstadt, Monaco, and Bologna (e.g., see Figure 2). Due to time constraints, we chose a step size of 0.2 for the prevalence of AVs, eHMIs, and the base defiance, resulting in 5 × 5 × 5 = 125 logs per city. A descriptive data report per city was generated via DataExplorer [18] and is attached in the GitHub repository under data. Due to the data size (between 275 MB and 4.2 GB), we will make the data available upon request. All relevant...
tables for the analyses are also available in the repository. We provide an initial overview of results for Ingolstadt, Germany, due to its realistically modeled traffic (taken from [69]). Because of the large number of data entries, using R or Python was too time-consuming. Therefore, we provide a Julia script which can be expanded. This reduced the runtime from hours to a few minutes. Due to our focus on providing the code, the analysis is not exhaustive.

6.1 Heatmap of Interactions
First, we provide a heatmap of all interactions over all parameter combinations in Figure 3. This heatmap shows that interactions occurred over the entire city. Attention: due to limits in Julia’s visualization, the city had to be inverted vertically.

![2D Histogram with Overlay](image)

Figure 3: Heatmap of interactions between pedestrians and AVs in Ingolstadt, Germany over all parameter combinations.

6.2 Interaction Effects on Crossing Probability
We fitted a linear mixed model to predict crossing probability with regard to AV density, eHMI density, and base AV defiance (see Figure 4). For a detailed description, see the repository.

![Crossing Probability](image)

Figure 4: Crossing Probability. Linear mixed model results.

6.3 Automated Vehicle Density → Collisions
We fitted a linear model to find the correlation between AV density and collisions (see Figure 5). The linear model shows a downward trend of collisions with higher AV density.

![Collisions with regard to AV density.](image)

Figure 5: Collisions with regard to AV density.

7 DISCUSSION AND FUTURE WORK
In this work, we presented an implementation and preliminary data to study the effect of AVs and attached eHMIs in their interaction with pedestrians on a large scale. Our simulacra implementation relies on empirical data. However, scientific data can be scarce regarding certain factors, showing a potential flaw in how scientific results are reported by solely reporting differences but not quantifying them. Therefore, some numbers may be educated guesses rather than extracted from studies and statistics. Nonetheless, we argue it is the most appropriate way to study the large-scale effects. Additionally, we enable the usage of LLMs for deriving crossing decisions. Our first evaluations reported in Section 6 show that we can simulate crossings in various areas of the cities and that, for example, the impact of AV density on collisions seems negatively correlated (i.e., more AVs lead to reduced collisions).

Very recently, Tian et al. [67] provided a novel model for the interaction of pedestrians and AVs. However, they do not provide an implementation, severely reducing applicability. In the future, we aim to re-implement this model to compare it against ours. Furthermore, we envision including additional mobility concepts, such as micromobility, in the interaction simulation and implementing interaction between manual drivers and other vulnerable road users. Besides, our approach can be extended to investigate the macroscopic effects of novel in-vehicle user interfaces (see [38, 39]) on traffic. Also, the extensive resulting datasets suggest that spatio-temporal automotive user interface analysis [37] could facilitate future simulation analysis.

ACKNOWLEDGMENTS
We thank the SUMO developers for their support.
A ALGORITHMS

Algorithm 1: Creation of Incoming Lanes Dictionary

Result: A dictionary containing a set of all incoming lanes into each unprioritized crossing in the simulation

for lane in traci.lane.getIDList()do
    if ("c" in lane) and ("cluster" not in lane) then
        internal_foes_dict[lane] = traci.lane.getInternalFoes(lane);
end

for lane in traci.lane.getIDList()do
    for successor_tuple in traci.lane.getLinks(lane) do
        if successor_tuple[5] == 'M' then
            internal_successor = successor_tuple[4];
            if internal_successor != "" then
                for crossing in internal_foes_dict do
                    if internal_successor in internal_foes_dict[crossing] then
                        cross_dict.setdefault(crossing, set()).add(lane);
                    end
                end
            end
        end
    end
end
return cross_dict;

Algorithm 2: check_for_av_crossing_scenario

Result: True if pedestrians would be unable to cross but the closest vehicle is an AV, False otherwise

est_time_needed_to_cross = traci.lane.getLength(crossing + "_0")/est_walking_speed;

for incoming_lane in crossing_dict[crossing + "_0"] do
    for vehicle in traci.lane.getLastStepVehicleIDs(incoming_lane) do
        distance_from_start_of_lane = traci.vehicle.getLanePosition(vehicle);
        if distance_from_start_of_lane > furthest_distance_from_start_of_lane then
            furthest_distance_from_start_of_lane = distance_from_start_of_lane;
            closest_vehicle = vehicle;
        end
    end
end
if closest_vehicle = "" then
    distance = traci.lane.getLength(incoming_lane) - furthest_distance_from_start_of_lane;
    if traci.vehicle.getSpeed(closest_vehicle) != 0 then
        if ttc < lowest_ttc_total then
            lowest_ttc_total = ttc;
            closest_vehicle_total = closest_vehicle;
        else
            lowest_ttc_total = 10.0;
            closest_vehicle_total = closest_vehicle;
        end
        closest_vehicles_dict[incoming_lane] = {"vehicle": closest_vehicle,
                                                "distance": distance, "ttc": ttc};
    else
        lowest_ttc_total = 10.0;
        closest_vehicles_dict[incoming_lane] = {"vehicle": closest_vehicle,
                                                "distance": distance, "ttc": 10.0};
    end
end
if closest_vehicles_dict[incoming_lane]||["ttc"] < est_time_needed_to_cross then
    if closest_vehicle in avs then
        av_crossing_scenario = True;
    else
        av_crossing_scenario = False;
    break;
end
end
return av_crossing_scenario;
\textbf{B CONFIGURABLE FACTORS}

Input: vehicle string ID of the closest vehicle to the crossing  
Result: True if the current situation is evaluated as dangerous, False otherwise  

\texttt{reaction\_distance = speed \times cf\_driver\_reaction\_time;}  
\texttt{breaking\_distance = pow(speed, 2) / (2 \times traci.vehicle.getEmergencyDecel(closest\_vehicle));}  
\texttt{stopping\_distance = reaction\_distance + breaking\_distance;}  
\texttt{distance\_to\_crossing = traci.lane.getLength(lane) - traci.vehicle.getLaneLength(lane);}  
\texttt{if stopping\_distance > distance\_to\_crossing then}  
\hspace{1em} \texttt{return True;}  
\hspace{1em} \texttt{else}  
\hspace{1.5em} \texttt{return False;}  
\hspace{1em} \texttt{end}

\begin{algorithm}
\caption{check\_for\_dangerous\_situation}
\end{algorithm}

\begin{algorithm}
\caption{Simulation Loop}
\end{algorithm}
Table 1: Table with configurable factors. The Value refers to the pre-configured value based on literature and own assumptions.

<table>
<thead>
<tr>
<th>Name</th>
<th>Abb</th>
<th>Value Range</th>
<th>Source</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>av_density</td>
<td></td>
<td>0.0 &lt; x &lt;= 1.0</td>
<td></td>
<td>density of AVs compared to the total number of vehicles.</td>
</tr>
<tr>
<td>ehmi_density</td>
<td></td>
<td>0.0 &lt; x &lt;= 1.0</td>
<td></td>
<td>density of AVs with eHMI compared to density of AVs.</td>
</tr>
<tr>
<td>est_walking_speed</td>
<td>1.0</td>
<td>x &gt;= 0.0</td>
<td></td>
<td>assumed average walking speed of pedestrians in meters per second.</td>
</tr>
<tr>
<td>driver_reaction_time</td>
<td>0.5</td>
<td>x &gt;= 0.0</td>
<td></td>
<td>assumed driver reaction time in seconds.</td>
</tr>
<tr>
<td>base_automated_vehicle_defiance</td>
<td></td>
<td>x &gt;= 0.0</td>
<td></td>
<td>base probability for a pedestrian to defy the right of way of incoming AVs.</td>
</tr>
<tr>
<td>ehmi_dfv</td>
<td>1.3</td>
<td>x &gt;= 0.0</td>
<td>[4]</td>
<td>defiance factor value (DFV) to defy priority of AVs with eHMI.</td>
</tr>
<tr>
<td>walking_pedestrian_dfv</td>
<td>1.2</td>
<td>x &gt;= 0.0</td>
<td>[4]</td>
<td>pedestrian-already-walking DFV.</td>
</tr>
<tr>
<td>group_size_dfv_two_to_three</td>
<td>1.2</td>
<td>x &gt;= 0.0</td>
<td>[4]</td>
<td>DFV for groups of two or three.</td>
</tr>
<tr>
<td>ttc_lower_extreme_time</td>
<td>1.4</td>
<td>x &gt;= 0.0</td>
<td>[1]</td>
<td>time to collision (TTC) in seconds under which extreme DFV is used.</td>
</tr>
<tr>
<td>ttc_lower_bound_time</td>
<td>ttc_lbt</td>
<td>3.0</td>
<td>x &gt;= ttc_lbt</td>
<td>TTC in seconds under which the lower bound DFV is used.</td>
</tr>
<tr>
<td>ttc_upper_bound_time</td>
<td>ttc_upt</td>
<td>6.0</td>
<td>x &gt;= ttc_lbt</td>
<td>TTC in seconds over which the upper bound DFV is used.</td>
</tr>
<tr>
<td>ttc_dfv_under_lower_extreme</td>
<td>ttc_ulo</td>
<td>0.01</td>
<td>x &lt;= ttc_ulo</td>
<td>DFV for extremely low TTC.</td>
</tr>
<tr>
<td>ttc_dfv_under_upper_bound</td>
<td>ttc_uob</td>
<td>3.0</td>
<td>x &gt;= ttc_uob</td>
<td>DFV if TTC is under lower bound.</td>
</tr>
<tr>
<td>waiting_time_dfv_over_accepted_value</td>
<td>wt_av</td>
<td>28</td>
<td>x &gt;= 0.0</td>
<td>accepted waiting time for pedestrians in seconds.</td>
</tr>
<tr>
<td>time_dfv_over_accepted</td>
<td>wt_ips</td>
<td>0.0494</td>
<td>x &gt;= 0.0</td>
<td>DFV increase per second if waiting time is above-accepted value (linear increase).</td>
</tr>
<tr>
<td>neutral_street_width</td>
<td>7.0</td>
<td>x &gt;= 0.0</td>
<td>[57]</td>
<td>street width in meters that is considered neutral.</td>
</tr>
<tr>
<td>start_age</td>
<td>14</td>
<td>x &gt;= 0.0</td>
<td>[54]</td>
<td>up to what age a person is viewed as a child.</td>
</tr>
<tr>
<td>female_present_dfv</td>
<td>0.85</td>
<td>x &gt;= 0.0</td>
<td>[54]</td>
<td>DFV if a girl is present.</td>
</tr>
<tr>
<td>boy_present_dfv</td>
<td>0.9</td>
<td>x &gt;= 0.0</td>
<td>[54]</td>
<td>DFV if a boy is present.</td>
</tr>
<tr>
<td>smombie_start_age</td>
<td>8</td>
<td>x &gt;= 0.0</td>
<td>[57]</td>
<td>start age in years for linear increase in chance to be a smombie.</td>
</tr>
<tr>
<td>smombie_end_age</td>
<td>50</td>
<td>x &gt;= 0.0</td>
<td>[66]</td>
<td>peak chance to be a smombie at smombie_end_age (linear decrease from smombie_start_age).</td>
</tr>
<tr>
<td>neutral_vehicle_size</td>
<td>2.52</td>
<td>x &gt;= s_nv</td>
<td></td>
<td>front area in square meters for a VW Scirocco 3.</td>
</tr>
<tr>
<td>small_vehicle_size</td>
<td>1.755</td>
<td>x &gt;= s_svs</td>
<td></td>
<td>front area in square meters for an ElectraMecanica Solo.</td>
</tr>
<tr>
<td>large_vehicle_size</td>
<td>4.0</td>
<td>x &gt;= s_nv</td>
<td></td>
<td>front area in square meters for a Hummer H2.</td>
</tr>
<tr>
<td>neutral_vehicle_size_dfv</td>
<td>1.0</td>
<td>x &gt;= 0.0</td>
<td>[41]</td>
<td>DFV for average-sized vehicles.</td>
</tr>
<tr>
<td>large_vehicle_size_dfv</td>
<td>0.7</td>
<td>x &gt;= 0.0</td>
<td>[41]</td>
<td>DFV for large vehicles (linear increase from neutral_vehicle_size_dfv).</td>
</tr>
<tr>
<td>neutral_vehicle_size_dfv</td>
<td>0.001</td>
<td>x &gt;=</td>
<td>[41]</td>
<td>smombie chance for ages outside the defined interval.</td>
</tr>
<tr>
<td>small_vehicle_size_dfv</td>
<td>0.001</td>
<td>x &lt;=</td>
<td>[41]</td>
<td>smombie chance for ages outside the defined interval.</td>
</tr>
<tr>
<td>large_vehicle_size_dfv</td>
<td>0.01</td>
<td>x &lt;= s_svs</td>
<td>[41]</td>
<td>smombie chance for ages outside the defined interval.</td>
</tr>
<tr>
<td>neutral_vehicle_size_dfv</td>
<td>1.3</td>
<td>x &gt;= 0.0</td>
<td>[41]</td>
<td>upper bound DFV for small vehicles (linear increase from neutral_vehicle_size_dfv).</td>
</tr>
<tr>
<td>lane_low_occupancy_rate</td>
<td>0.02</td>
<td>0 &lt;= x &lt;= 1.0</td>
<td>[62]</td>
<td>lower bound lane occupancy rate in (length of all vehicles) / (street length).</td>
</tr>
<tr>
<td>lane_high_occupancy_rate</td>
<td>0.1</td>
<td>1 &lt;= x &lt;= 1.0</td>
<td>[62]</td>
<td>upper bound lane occupancy rate (0.1 means 10% of street is filled with vehicles).</td>
</tr>
<tr>
<td>low_occupancy_rate_dfv</td>
<td>1.2</td>
<td>x &gt;= 0.0</td>
<td>[62]</td>
<td>upper bound DFV for a low lane occupancy rate (linear increase with decreasing occupancy rate).</td>
</tr>
<tr>
<td>high_occupancy_rate_dfv</td>
<td>0.8</td>
<td>x &gt;= 0.0</td>
<td>[62]</td>
<td>lower bound DFV for a high lane occupancy rate.</td>
</tr>
<tr>
<td>male_gender_dfv</td>
<td>1.8</td>
<td>x &gt;= 0.0</td>
<td>[66]</td>
<td>DFV for male pedestrians.</td>
</tr>
<tr>
<td>female_gender_dfv</td>
<td>1.0</td>
<td>x &gt;= 0.0</td>
<td>[66]</td>
<td>DFV for female pedestrians.</td>
</tr>
<tr>
<td>other_gender_dfv</td>
<td>1.4</td>
<td>x &gt;= 0.0</td>
<td>[66]</td>
<td>DFV for diverse pedestrians.</td>
</tr>
<tr>
<td>impaired_vision_dfv</td>
<td>1.2</td>
<td>x &gt;= 0.0</td>
<td>[25]</td>
<td>DFV for pedestrians with impaired vision.</td>
</tr>
<tr>
<td>healthy_vision_dfv</td>
<td>1.0</td>
<td>x &gt;= 0.0</td>
<td>[25]</td>
<td>DFV for pedestrians with healthy vision.</td>
</tr>
</tbody>
</table>

End of table
C  FORMULAE
C.1 get_waiting_time_defiance_factor

\[ f(\text{waiting\_time}) = \begin{cases} \frac{\text{wt\_nav}}{2} & \text{if waiting\_time} \leq \text{wt\_av} \\ 1.0 + \left( \text{waiting\_time} - \text{wt\_av} \right) \cdot \text{wt\_ips} & \text{otherwise} \end{cases} \]  

(1)

C.2 get_smombie_defiance_factor

\[ \text{distraction\_chance} = \begin{cases} s_{\text{csa}} + \left( \text{ped\_age} - s_{\text{sa}} \right) \times \frac{s_{\text{pa}} - s_{\text{csa}}}{s_{\text{pa}} - s_{\text{sa}}} & \text{if } s_{\text{sa}} \leq \text{ped\_age} \leq s_{\text{pa}} \\ s_{\text{cpa}} - \left( \text{ped\_age} - s_{\text{pa}} \right) \times \frac{s_{\text{pa}} - s_{\text{cea}}}{s_{\text{ea}} - s_{\text{pa}}} & \text{if } s_{\text{pa}} \leq \text{ped\_age} \leq s_{\text{cea}} \end{cases} \]

\[ f(\text{distraction\_chance}) = \begin{cases} \text{smombie\_dfv} & \text{if random\_number} \leq \text{distraction\_chance} \\ 1.0 & \text{otherwise} \end{cases} \]

(2)

C.3 get_child_present_defiance_factor

\[ f(\text{ped}) = \begin{cases} \text{boy\_present\_dfv}, & \text{if ped\_age} \leq \text{child\_age} \text{ and ped\_gender} = \text{”male”} \\ \text{girl\_present\_dfv}, & \text{if ped\_age} \leq \text{child\_age} \text{ and ped\_gender} = \text{”female”} \\ \frac{\text{boy\_present\_dfv} + \text{girl\_present\_dfv}}{2}, & \text{if ped\_age} \leq \text{child\_age} \text{ and ped\_gender} \neq \text{”male”} \text{ and ped\_gender} \neq \text{”female”} \\ 1.0, & \text{otherwise} \end{cases} \]

(4)

C.4 get_street_width_defiance_factor

\[ f(\text{crossing\_length}) = \frac{1}{\text{crossing\_length}} \cdot \frac{\text{neutral\_street\_width}}{\text{neutral\_street\_width}} \]

(5)

C.5 get_ped_speed_defiance_factor

\[ f(\text{ped\_speed}) = \begin{cases} \text{walking\_pedestrian\_dfv} & \text{if ped\_speed} > 0.6, \\ 1.0 & \text{otherwise}. \end{cases} \]

(6)

C.6 get_vehicle_size_defiance_factor

\[ f(\text{vehicle}) = \begin{cases} \text{small\_vehicle\_size\_dfv}, & \text{if vehicle\_front\_area} \leq \text{small\_vehicle\_size} \\ \text{dmax\_dfv} \times \frac{\text{dmax\_dfv} - \text{dmax\_dfv}}{\text{dmax\_dfv} - \text{dmax\_dfv}}, & \text{if small\_vehicle\_size} < \text{vehicle\_front\_area} < \text{neutral\_vehicle\_size} \\ \text{neutral\_vehicle\_size\_dfv}, & \text{if vehicle\_front\_area} = \text{neutral\_vehicle\_size} \\ \text{dmin\_dfv} \times \frac{\text{dmin\_dfv} - \text{dmin\_dfv}}{\text{dmin\_dfv} - \text{dmin\_dfv}}, & \text{if neutral\_vehicle\_size} < \text{vehicle\_front\_area} < \text{large\_vehicle\_size} \\ \text{large\_vehicle\_size\_dfv}, & \text{if vehicle\_front\_area} \geq \text{large\_vehicle\_size} \end{cases} \]

\[ \text{where area\_diff} = |\text{neutral\_vehicle\_size} - \text{vehicle\_front\_area}| \]

(7)

C.7 get_road_occupancy_rate_defiance_factor

\[ f(\text{occupancy\_rate}) = \begin{cases} \text{low\_occupancy\_rate\_dfv}, & \text{if occupancy\_rate} \leq \text{lor} \\ \text{high\_occupancy\_rate\_dfv}, & \text{if } l < \text{occupancy\_rate} \leq \text{hor} \\ \text{high\_occupancy\_rate\_dfv}, & \text{if } \text{occupancy\_rate} \geq \text{hor} \end{cases} \]

(9)

C.8 get_group_size_defiance_factor

\[ f(\text{group\_size}) = \begin{cases} 1.0, & \text{if group\_size} = 1 \\ \text{group\_size\_dfv\_two\_to\_three}, & \text{if } 2 \leq \text{group\_size} \leq 3 \\ \text{group\_size\_dfv\_over\_three}, & \text{if } \text{group\_size} > 3 \end{cases} \]

(10)

C.9 get_time_to_collision_defiance_factor

\[ f(\text{ttc}) = \begin{cases} \text{ttc\_dfv\_under\_lower\_extreme}, & \text{if } \text{ttc} \leq \text{ttc\_lower\_extreme\_time} \\ \text{ttc\_dfv\_under\_lower\_bound}, & \text{if } \text{ttc\_lower\_extreme\_time} < \text{ttc} \leq \text{ttc\_lower\_bound\_time} \\ \text{ttc\_llb\_dfv} + (\text{ttc} - \text{ttc\_ltb}) \times \frac{\text{ttc\_ullb} - \text{ttc\_llb}}{\text{ttc\_ubt} - \text{ttc\_ltb}}, & \text{if } \text{ttc\_lower\_bound\_time} < \text{ttc} < \text{ttc\_upper\_bound\_time} \\ \text{ttc\_dfv\_over\_upper\_bound}, & \text{if } \text{ttc} \geq \text{ttc\_upper\_bound\_time} \end{cases} \]

(11)
## DATA LOGGING

Table 2: Custom data generated and accessible after a simulation.

<table>
<thead>
<tr>
<th>Information</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>timestamp</td>
<td>Date &amp; Time</td>
<td>The real-world time at which the event occurred.</td>
</tr>
<tr>
<td>step</td>
<td>Integer [1; ∞]</td>
<td>The simulation step at which the event occurred.</td>
</tr>
<tr>
<td>scenario</td>
<td>String</td>
<td>The scenario in which the event occurred.</td>
</tr>
<tr>
<td>pedestrianID</td>
<td>String</td>
<td>ID of the pedestrian that had to choose whether to ignore AV priority.</td>
</tr>
<tr>
<td>crossingID</td>
<td>String</td>
<td>ID of the crossing where the event occurred.</td>
</tr>
<tr>
<td>final crossing probability</td>
<td>Float [0; ∞]</td>
<td>Final crossing probability calculated using all factors.</td>
</tr>
<tr>
<td>effective final crossing probability</td>
<td>Float [0; 1]</td>
<td>Adjusted probability to be between 0 and 1.</td>
</tr>
<tr>
<td>crossing decision</td>
<td>['cross', 'not cross']</td>
<td>Decision of pedestrian to respect AV priority at crossing.</td>
</tr>
<tr>
<td>dangerous situation</td>
<td>Boolean</td>
<td>Calculated estimation if the situation was dangerous.</td>
</tr>
<tr>
<td>waiting time</td>
<td>Integer [0; ∞]</td>
<td>Time in seconds that the pedestrian waited at the crossing.</td>
</tr>
<tr>
<td>pedestrian location x</td>
<td>Integer [-∞; ∞]</td>
<td>Pedestrian location x at time of decision taking.</td>
</tr>
<tr>
<td>pedestrian location y</td>
<td>Integer [-∞; ∞]</td>
<td>Pedestrian location y at time of decision taking.</td>
</tr>
<tr>
<td>closest vehicle location x</td>
<td>Integer [-∞; ∞]</td>
<td>Closest vehicle location x to pedestrian at time of decision making.</td>
</tr>
<tr>
<td>closest vehicle location y</td>
<td>Integer [-∞; ∞]</td>
<td>Closest vehicle location y to pedestrian at time of decision making.</td>
</tr>
<tr>
<td>gender</td>
<td>['male', 'female', 'other']</td>
<td>Gender of pedestrian</td>
</tr>
<tr>
<td>vision</td>
<td>['healthy', 'impaired']</td>
<td>Vision health of pedestrian.</td>
</tr>
<tr>
<td>age</td>
<td>Integer [6; 99]</td>
<td>Age of pedestrian.</td>
</tr>
<tr>
<td>probability estimation method</td>
<td>['normal', 'lim']</td>
<td>Whether the LLM was used for this crossing decision.</td>
</tr>
<tr>
<td>defiance values</td>
<td>Integer/Float [0; ∞]</td>
<td>All defiance values described in 4.2.</td>
</tr>
</tbody>
</table>