Criticality Measures to Evaluate the Triggering Decision of Collision Avoidance Functions at Intersections

Peter Riegl and Andreas Gaull
University of Applied Sciences Ingolstadt, CARISSMA, Ingolstadt, Germany
Email: {peter.riegl, andreas.gaull}@thi.de

Michael Beitelschmidt
Technical University Dresden, Faculty of Mechanical Engineering, Dresden, Germany
Email: michael.beitelschmidt@tu-dresden.de

***Abstract*—In order to significantly reduce the amount of testing required to validate the behavior of vehicle safety functions with real vehicles, the scenarios in the simulation must reproduce the traffic as realistically as possible. For this reason, a microscopic traffic flow simulation is used. Since the road users show a perfect behavior by default, driving errors have to be induced. The necessary adjustments in the existing driver models are presented schematically in this paper. These changes in the driving behavior ensure that some road users cause situations that are critical for others. Such scenarios are required as input data for testing vehicle safety functions in a vehicle dynamics simulation. A decisive factor for the quality with which a critical situation can be successfully managed by a vehicle safety function is the trigger time. For this purpose, approaches are presented for estimating the period in which a collision can still be avoided by braking or evasive maneuver. Three intersection scenarios illustrate the functioning of these criteria.**

***Index Terms*—collision avoidance, validation, TTC, TTB, TTS, traffic flow simulation, driving error**

I. Introduction

A decisive factor for a successful market launch of vehicle safety functions is their effective validation. Because even with today's driver assistance functions, where the driver is still available as the last fallback level, testing is already associated with a huge amount of test kilometers. However, this is by no means comparable with what will be necessary in the future. Winner [1] estimates that more than 240 million kilometers will be required to achieve statistical validation in areas such as highway traffic alone. In comparison, inner-city intersections are much more complex, which means that the necessary test scope will increase significantly. One method that plays a key role in the future validation of driving functions is simulation. The reason for this is that a multitude of critical intersection scenarios can be generated in which the safety function must be triggered. For this purpose, realistic test scenarios must first be generated. This is the only way to ensure that the results obtained in the simulations concerning the correct triggering behavior can make a significant contribution to reducing the test effort required with the real vehicle.

A combinatorial test case generation like in [2]-[4] creates a large number of scenarios by varying the input parameters of the vehicle safety function to be validated. The effort required increases exponentially with the growing number of influencing factors to be considered. The immense amount of resulting scenarios contains a multitude of configurations that can never occur in real vehicle traffic. Furthermore, it cannot be stated certainly that the limits within which the variables to be considered are varied are actually the right ones. Moreover, this approach can lead to situations in which important influencing factors are unknowingly ignored and potentially dangerous malfunctions are completely overlooked.

An alternative approach that enables the generation of more realistic test scenarios and which is described in this paper assumes that these scenarios can be selected directly from traffic events. For this reason, real vehicle traffic is modeled in a traffic flow simulation in SUMO (Simulation of Urban Mobility) [5]. As in real traffic, the ego-vehicle equipped with the safety function to be tested moves on the public roads of a city. Due to the driving errors of the other road users, critical situations arise there, which have to be mastered by the safety function. By default, the road users commit no errors, and their anticipatory driving style avoids critical situations in advance. Such scenarios are unsuitable for testing a safety function, which is a key point of this paper, as there is no need for the function to intervene. For this reason, it is necessary to include errors into the driving behavior. The procedure for implementing driving errors is described in section II.

The criticality of the current traffic situation in SUMO is identified in [6] using the TTC criterion, which indicates the time remaining until the vehicles collide if they behave during this period as in the current time step.

Manuscript received October 30, 2020; revised December 16, 2020.

doi: 10.18178/jtle.8.2.63-72

However, this is not sufficient for an evaluation of the triggering time of the driving function to be tested. For this reason, further criteria are presented in section III.B. First of all, a prediction is performed which checks whether a collision of the ego-vehicle with another road user may occur. In contrast to [7], where only one co-vehicle is taken into account, all vehicles that are in the immediate vicinity of the ego-vehicle are considered. Furthermore, the time remaining to prevent a crash by a steering or braking maneuver is estimated.

In the traffic flow simulation in SUMO, all vehicles are only described by their kinematics. However, this is not sufficient for a meaningful validation of the behavior of the safety function and the ego-vehicle. For this reason, a scenario identified as critical by the TTC criterion is transferred to a vehicle dynamics simulation in CarMaker where the motion of the ego-vehicle is described by a spatial vehicle dynamics model. In addition, the ego-vehicle has a steering and braking system as well as a powertrain by default. Further vehicle components can easily be added. To ensure that the scenarios in the two software tools match, the kinematic data of all road users in the relevant area and the associated road infrastructure are transferred from SUMO to CarMaker. Since CarMaker uses a different modeling approach than SUMO, the road network must first be adapted accordingly. In [6] the tool chain of this simulation environment shown in Fig. 1 is described more in detail.

Figure 1. Toolchain of the simulation environment

The other road users who are in the immediate vicinity of the ego-vehicle are detected by sensors that have a limited range and aperture angle. Based on these data the criteria TTS and TTB are calculated in MATLAB. Furthermore, the triggering behavior and the performance of the safety function are evaluated there. In the period before the driving function is triggered, the ego-vehicle in CarMaker is controlled in the same way as by the human driver in terms of specifying the accelerator and brake pedal position as well as the steering wheel angle. An MPC controller generates these input data based on the vehicle trajectory and the speed curve.

II. Driving Errors

The traffic flow simulation in its original application aims at testing the performance of the road infrastructure during traffic disruptions such as road works or traffic jams [8]. This is only possible if the road users obey the traffic rules, do not perform risky maneuvers and intuitively adapt their behavior to that of others. To prevent a scenario from becoming critical for the ego-vehicle, others slow down in time or change lanes accordingly. However, this behavior is unsuitable when it comes to cause exactly these critical situations, which lead to collisions if the safety function does not intervene in time. For this reason, the existing driver model, which is implemented by default in SUMO, must be adapted to these requirements. The driving behavior in SUMO is composed of a car-following model, a lane change model and an intersection model. The modifications described below affect only a few road users who are randomly selected by a normal distribution. All other vehicles behave as before. A validation of the resulting driving behavior with naturalistic driving data is not performed here, as this would go beyond the scope of this paper.

A. *Car-following Model*

The Krauss model [9], which SUMO uses as default to describe the traffic in a lane, is based on the idea that a vehicle must always keep such a large distance from another vehicle in front so that a collision can be avoided at any time even if the latter performs an emergency stop. This approach cannot be used here, since exactly these scenarios should occur. In the Intelligent Driver Model by Treiber [10], a vehicle reacts in time to an approaching obstacle and reduces its speed accordingly. This also contradicts the objective of causing critical situations.

In this paper, the Wiedemann model [11] is used, which is similar to human behavior and validated in a various number of studies [12], [13]. If two vehicles moving on the same lane are far apart, they do not influence each other. If the distance decreases, the relative speed becomes relevant. The range of the transition from free driving to consciously influenced driving depends on two normally distributed parameters. These are the driver's need for safety, which determines the safety distance to be maintained, and the ability to estimate the traffic situation correctly. If the two parameters have an unfavorable combination, the rear vehicle adapts its speed to that of the vehicle in front only after a delay. This reduces the distance between the vehicles and thus the braking distance available to the rear vehicle if the front vehicle brakes hard. This makes rear-end collisions much more likely. In addition, road users who have correctly anticipated the situation but do not brake in time due to an excessively long reaction time are also included in this approach.

B. *Lane Change Model*

The main idea of the lane change model in SUMO is that a vehicle performs an intended lane change only if there is a sufficiently large gap on the target lane. It must be larger than the minimum distance that a vehicle following another one must keep according to the car-following model. If a vehicle following on an adjacent lane has a lower speed, there are no restrictions when changing lanes. If, however, it is faster than the vehicle in front, either the process is aborted or the vehicle behind cooperates by adapting its speed and distance accordingly, in order not to hinder the lane change. To ensure that critical situations occur, some vehicles are supposed to make mistakes. These road users are then no longer able to estimate the distance to the others correctly or do not pay attention to the traffic on the adjacent lanes. This means that a vehicle that intends to change lanes will not abort the process even if it is risky. Furthermore, a vehicle will cooperate too late or not at all, if it cannot realize that this is necessary. Finally, a too long reaction time after recognizing a situation can become dangerous for other road users, because the necessary actions are started too late. Besides cognitive errors, a conscious decision can also cause a critical situation. Although the approaching vehicle is recognized correctly, the decision is made to change lanes. Furthermore, not all vehicles will reduce their speed to assist others in performing their planned driving maneuvers.

C. Intersection Model

The first two models are well suited for describing the driver's behavior on country roads and highways. In cities, however, a large number of critical scenarios occur in intersection areas. For this reason, additional adjustments have to be made in the third submodel. The right before left rule, traffic signs and traffic lights define which road user has priority at an intersection. If traffic lights do not control the traffic, SUMO gives the right of way to the vehicle that arrives first. Each traffic participant informs the entry link of the following intersection about the planned time of arrival and the estimated speed from the current perspective. Based on this information, the time occupancy of each single connection between an incoming and an outgoing lane can be determined. In addition, each entry link knows not only its own occupancy but also that of the lanes, which have a higher priority due to the right before left rule, the turning sequence or traffic signs. When a vehicle reaches an entry link, it is calculated whether there is a conflict with a vehicle on a higher priority lane for the time it would take to cross the intersection. If access is denied, the speed is to be reduced, priority is to be given to that road user and a new request is to be made in the next time step. If, on the other hand, it is possible to enter, the speed of the vehicle must be adapted to that of the road users who are already in the intersection. Furthermore, the car-following model is now decisive for the driving behavior. If an intersection is equipped with a traffic light system, this system will prioritize the lanes.

In order to generate critical situations in the described model, corresponding errors must be integrated into the driving behavior. Some vehicles then show a disturbed perception of the distance to the next intersection and thus may not be able to stop at a red traffic light. As a result, they enter the intersection without permission. In addition, some vehicles notice too late that the traffic light has already switched from green to red. Therefore, the vehicles cannot brake in time. At intersections without traffic lights, either it is not recognized that someone else has the right of way, or the vehicle continues its own maneuver despite the knowledge, because it assumes that the time interval until another vehicle arrives is long enough. If the permission to enter the intersection has been granted, there are vehicles that do not adapt their speed to that of those who are already there and which must therefore be given priority.

III. EVALUATION OF THE CRITCALITY

Criteria such as TTC (Time-to-Collision), TTB (Time-to-Brake) and TTS (Time-to-Steer) have proven to be useful in evaluating criticality on country roads and highways on which road users move in one direction [14]. However, they are not easily applicable to intersection scenarios, since the collision point can lie anywhere on the contour of the co-vehicle, in contrast to rear-end and frontal collisions. The reason for this is that without knowledge of the exact crash configuration, it is not possible to calculate how far the vehicles can still move until a collision occurs. The same applies to the lateral motion, which is required for an evasive maneuver in the context of the calculation of the TTS. For this reason, the motion of all vehicles in the road section to be analyzed is predicted in this paper within a time horizon t_{pred}. In [7], the prediction uses an unscented Kalman filter to analyze intersection scenarios where only the behavior of the ego-vehicle and one co-vehicle are considered. In complex situations, however, there are other road users in the intersection, which must also be taken into account.

A. Collision Detection

The prediction with a time increment Δt considers only those vehicles in the current time step that are in the immediate vicinity of the ego-vehicle. The distance must not exceed a distance of $v_{Es} \cdot t_{pred}$ depending on the speed of the ego-vehicle v_{Es}. The kinematic data of all road users in this region are retrieved in SUMO via the TraCI interface. In CarMaker, however, these data are measured by the sensors of the ego-vehicle. Since these sensors have a limited aperture angle and range it may happen that not all vehicles can be detected. In addition, the road users are supposed to move on tracks that correspond to the course of the road section. This means that the orientation $\psi_k(i)$ of the k-th vehicle is already given. Thus, it can only move tangential s_k and normal n_k to the respective trajectory as seen in Fig. 2. The motion of the entire vehicle is therefore described in a simplified way by a point mass. Based on these assumptions, the motion of the kinematic reference points of all K road users at the prediction time step i is to be determined. Therefore results for the speed and the position of the k-th vehicle:

$$v_{ks}(i) = v_{ks}(i-1) + a_{ks} \cdot \Delta t \tag{1}$$

$$v_{kn}(i) = v_{kn}(i-1) + a_{kn} \cdot \Delta t \tag{2}$$

$$s_k(i) = s_k(i-1) + v_{ks}(i-1) \cdot \Delta t \tag{3}$$

$$n_k(i) = n_k(i-1) + v_{kn}(i-1) \cdot \Delta t \tag{4}$$

The accelerations of the vehicle a_{ks} and a_{kn} are supposed to be constant. The shape of a road is described in SUMO by a polygon course. A linear progression is assumed between the reference points. To determine the global position of the point $\boldsymbol{P}_k$, which is described by the vector $\boldsymbol{r}_k$, first the two points along the road are identified between which the vehicle must be located according to s_k and the length of the single road segments. Then, by linear interpolation, the point $\boldsymbol{P}_{k0}$ is calculated at which the vehicle would be located if it had no lateral displacement with respect to the centerline of the lane. To finally obtain the position vector $\boldsymbol{r}_k$, the point $\boldsymbol{P}_{k0}$ is shifted by $n_k \cdot \boldsymbol{e}_n$ in normal direction.

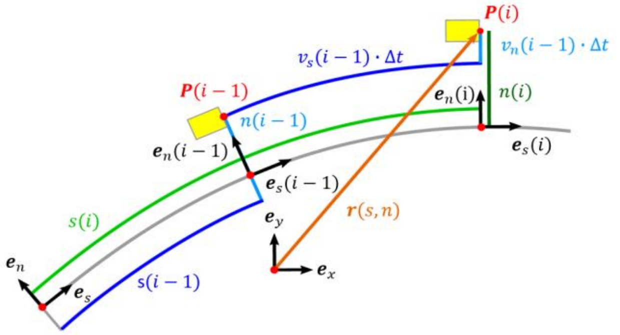

Figure 2. Prediction of the motion along the road course

The vehicle contour is assumed rectangular, as shown in Fig. 3. Furthermore, the kinematic reference point is located in the middle of the front of the vehicle. The four corner points can be specified using the vehicle's length l_k and width b_k:

$$\boldsymbol{Q}_k = \boldsymbol{P}_k + \boldsymbol{A}_k \cdot \begin{bmatrix} 0 \\ \frac{b_k}{2} \end{bmatrix} \qquad \boldsymbol{R}_k = \boldsymbol{P}_k + \boldsymbol{A}_k \cdot \begin{bmatrix} 0 \\ -\frac{b_k}{2} \end{bmatrix}$$
$$\boldsymbol{S}_k = \boldsymbol{P}_k + \boldsymbol{A}_k \cdot \begin{bmatrix} -l_k \\ -\frac{b_k}{2} \end{bmatrix} \qquad \boldsymbol{T}_k = \boldsymbol{P}_k + \boldsymbol{A}_k \cdot \begin{bmatrix} -l_k \\ \frac{b_k}{2} \end{bmatrix} \tag{5}$$

Here the rotation matrix reads

$$\boldsymbol{A}_k = \begin{bmatrix} \cos(\psi_k) & -\sin(\psi_k) \\ \sin(\psi_k) & \cos(\psi_k) \end{bmatrix}. \tag{6}$$

Using a GJK algorithm [15] it is checked whether the vehicle contours of two or more vehicles intersect within the prediction horizon. If this is the case, the TTC is calculated from the prediction time step i_C where a collision occurs first

$$TTC = i_C \cdot \Delta t. \tag{7}$$

B. Criticality Measure

After a collision has been detected within the prediction horizon, criteria must now be defined which indicate how long it is possible to wait before triggering the safety function until a collision can no longer be avoided. On the one hand, the speed of the ego-vehicle can be reduced so that the vehicles with which a collision has been detected can continue driving unhindered. On the other hand, the critical area can be bypassed by an evasive maneuver. For this purpose, the necessary lateral offset must first be calculated. Therefore, a second prediction with a new horizon t_{predS} is performed for the motion of the co-vehicles starting at the time of collision. In the following, it is assumed that the evasion trajectory of the ego-vehicle in the collision area is parallel to the track on which a crash with another vehicle has been predicted. For that reason, the minimum lateral displacement necessary to avoid a collision with all road users within the whole prediction horizon is to be determined. Algorithm 1 calculates in each time step the distance that the ego-vehicle has to swerve to the left and right to avoid a collision with the critical co-vehicle. The coordinates describing its contour represent the columns of the matrix $\boldsymbol{H}$. First, the difference vector between each point of the contour of the co-vehicle and a point of the ego-vehicle's front is calculated. For an evasive maneuver to the left, the point $\boldsymbol{R}_E(k)$ must be used, because it is located rightmost. Similarly, for a motion to the right, the point $\boldsymbol{Q}_E(k)$ is decisive. The scalar product of this vector with the lateral direction vector $\boldsymbol{e}_{lat}$ yields the required displacement at the respective point of the vehicle contour. To generate an evasion trajectory, only the maximum distances to the left and right are necessary, since only these avoid a collision with the entire contour. Fig. 3 shows such a configuration in which the necessary distance s_L for an evasive trajectory to the left has been identified.

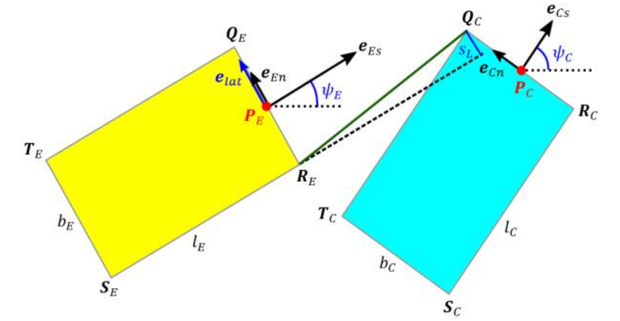

Figure 3. Lateral displacement for an evasion to the left

Algorithm 1. Minimum lateral shift

Input: corner points of ego-vehicle's front $\boldsymbol{Q}_E$ and $\boldsymbol{R}_E$, orientation of the ego-vehicle ψ_{Ecr}, matrix of the contour of the co-vehicle $\boldsymbol{H}$, position of the ego-vehicle $\boldsymbol{r}_{Ecr}$ at collision time, number of prediction times steps n_{pred}, lateral safety distance d_S

Output: lateral motion to the left s_L and to the right s_R, target points of the evasion trajectory $\boldsymbol{P}_L$ and $\boldsymbol{P}_R$

```
e_lat = [−sin(ψ_Ecr)   cos(ψ_Ecr)]^T
s_L = −∞
s_R = ∞
for k ← 1, n_pred do
    for n ← 1,4 do
        d_L = e_lat^T · (H(:, n) − R_E(k))
        if d_L > s_L then
            s_L = d_L
            P_L = r_Ecr + (d_S + s_L) · e_lat
        end if
        d_R = e_lat^T · (H(:, n) − Q_E(k))
        if d_R < s_R then
            s_R = d_R
            P_R = r_Ecr + (−d_S + s_R) · e_lat
        end if
    end for
end for
return s_L, s_R, P_L, P_R
```

The algorithm for calculating the path of the evasion trajectory, which is presented in section III.C, requires the target points $\boldsymbol{P}_L$ and $\boldsymbol{P}_R$ in addition to the orientation ψ_{Ecr} of the ego-vehicle at the predicted collision time. These points result from the lateral displacement of the predicted crash position of the kinematic reference point $\boldsymbol{r}_{Ecr}$ to the left and right. The distance that the ego-vehicle has to move laterally is composed of s_L or s_R and a safety distance d_S, which the ego-vehicle should keep to the obstacle during the evasion maneuver.

When determining the correct reaction to handle a critical scenario, the last possible time at which a collision can still be prevented by a braking or evasive maneuver is of decisive importance. For this reason, triggering criteria need to be defined. In this paper this is done exemplarily using the criticality measures TTB and TTS. The time required for swerving depends on the lateral distance to be covered, the current speed v_{Es} and the maximum yaw rate $\dot{\psi}_m$ that can be reached due to the dynamic limits of the vehicle

$$t_{SL} = \sqrt{\frac{2 \cdot s_L}{v_{Es} \cdot \dot{\psi}_m(v_{Es})}} \qquad t_{SR} = \sqrt{\frac{2 \cdot s_R}{v_{Es} \cdot \dot{\psi}_m(v_{Es})}}\,. \tag{8}$$

The time that can still be waited until a collision can no longer be avoided by an evasive maneuver to the left or right results in

$$TTS_L = TTC - t_{SL} \qquad TTS_R = TTC - t_{SR}\,. \tag{9}$$

By reducing the speed of the ego-vehicle, a collision between the road users can be avoided. Depending on the relative orientation of the vehicles at the current time $\Delta\psi = \psi_C - \psi_E$, one of the approaches presented below is used to calculate the duration of the necessary braking maneuver. If the difference angle is less than 90°, the situation can be managed by the ego-vehicle adapting its current speed v_{Es} to that of the co-vehicle v_{Cs}. Assuming that braking is performed with maximum deceleration a_B this yields to

$$TTB = \frac{v_{Es} - v_{Cs} + TTC \cdot (a_B - a_{Cs})}{a_B - a_{Es}}\,. \tag{10}$$

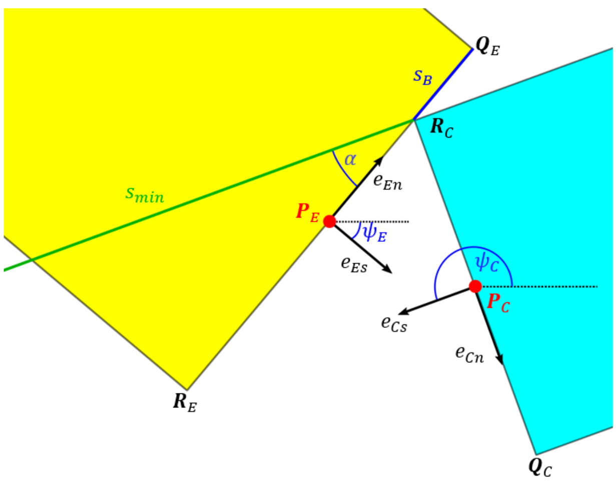

Figure 4. Predicted configuration of a head-on collision

If, on the other hand, a front or side crash occurs, as shown in Fig. 4, the braking maneuver should ensure that the co-vehicle can pass before the ego-vehicle reaches the critical area. The minimum distance s_{min} that the co-vehicle must travel is composed of its length l_C and a component s_B resulting from the intersection of the ego-vehicle's contour at the time of the collision with the predicted motion of the co-vehicle afterwards

$$s_{min} = l_C + \frac{s_B + \frac{b_E}{2}}{\cos(\alpha)}\,. \tag{11}$$

The time t_P, which the co-vehicle needs to move the distance s_{min}, is

$$t_P = \frac{-v_{Cs} + \sqrt{v_{Cs}^2 + 2 \cdot a_{Cs} \cdot s_{min}}}{a_{Cs}}\,. \tag{12}$$

The ego-vehicle covers the distance to the collision point in the time TTC. If, however, an emergency braking is initiated at time TTB, the period until the predicted collision point is reached is extended to $t_G = TTC + t_P$. Thus, the TTB is defined as follows

$$TTB = t_G + \frac{\sqrt{(a_B - a_{Es})^2 \cdot t_G^2 - q \cdot (a_B - a_{Es})}}{a_B + a_{Es}} \tag{13}$$

$$\text{where } q \coloneqq 2 \cdot v_{Es} \cdot t_P + a_B \cdot t_G^2 - a_{Es} \cdot TTC^2 \tag{14}$$

C. Collision Avoidance

After time criteria have been defined in the previous section, that indicate how much time remains until a collision can no longer be avoided by a braking or evasive maneuver, an avoidance strategy is presented in the following. The associated pseudo code is listed in Algorithm 2. This enables a decision to be made as to which of the two maneuvers is best suited to solve the critical situation with regard to the space required and the other road users. The safety function should be triggered as late as possible. If the TTB is longer than a threshold t_m when having decided to brake, no execution takes place in the current time step. The same applies to the value of the TTS_L or TTS_R in case of swerving.

The necessary lateral motion determined in section III.B must not exceed the distances of the ego-vehicle to the road boundaries d_{BL} and d_{BR} during an evasive maneuver. If there is not enough space left, this maneuver cannot be executed and an emergency braking must be initiated. Otherwise, the occupancy of the lanes in the relevant area by other road users is checked in the second step. To avoid new critical situations by this maneuver, there must be a sufficiently large speed-dependent gap with the limits d_{Cr} and d_{Cf}. If the required gap cannot be guaranteed at the current time, the relative velocity of the vehicles must also be considered more closely. If a co-vehicle is on an adjacent lane behind the ego-vehicle, its speed v_{Cr} must be lower. Similarly, a co-vehicle in front with the speed v_{Cf} must not be slower. If both the gap and the speed criterion cannot be met, there is nothing else left to do except to perform an emergency braking.

Algorithm 2. Decision maker

Input: lateral motion to the left s_L and to the right s_R, distance to road boundaries d_{BL} and d_{BR}, position of ego-vehicle s_E, speed of ego-vehicle v_{Es}, time to brake TTB, time to steer TTS_L and TTS_R, gap on the adjacent lane with limits d_{Cr} and d_{Cf}, speeds of vehicles on adjacent lane v_{Cr} and v_{Cf}

Output: reaction to manage the critical situation $react$

$react = none$
if $s_L < d_{BL}$ **then**
  if $d_{Cr} < s_E < d_{Cf}$ **and** $TTS_L < t_m$ **then**
    $react = swerve\ left$
  else if $v_{Cr} < v_{Es} < v_{Cf}$ **and** $TTS_L < t_m$ **then**
    $react = swerve\ left$
  else if $TTB < t_m$ **then**
    $react = brake$
  end if
else if $s_R < d_{BR}$
  if $d_{Cr} < s_E < d_{Cf}$ **and** $TTS_R < t_m$ **then**
    $react = swerve\ right$
  else if $v_{Cr} < v_{Es} < v_{Cf}$ **and** $TTS_R < t_m$ **then**
    $react = swerve\ right$
  else if $TTB < t_m$
    $react = brake$
  end if
else if $TTB < t_m$
  $react = brake$
end if
return $react$

The trajectory of an evasion or braking maneuver is generated based on trajectory chains [16]. During a braking maneuver, the ego-vehicle remains on its current path. Only the velocity along this curve is reduced until either standstill is reached or the co-vehicle has passed the area previously identified as critical. After that, the safety function is no longer activated and the vehicle can accelerate again to its initial speed.

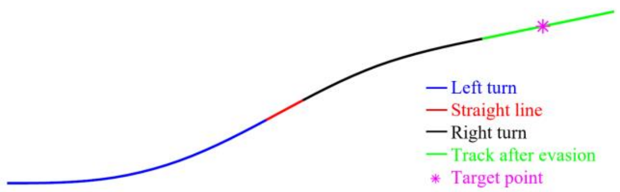

Figure 5. Evasion trajectory

If, on the other hand, an evasive maneuver is chosen, a new trajectory with the orientation ψ_{Ecr} must be created running through either $\boldsymbol{P}_L$ or $\boldsymbol{P}_R$. These target points are calculated in Algorithm 1. After the ego-vehicle has bypassed the obstacle, a second chain is created to ensure that it returns to the originally planned path. The procedure for creating these trajectories is described in [15] and [16]. An evasion trajectory, as shown in Fig. 5, has the shape of an s-bend and consists of a left and a right turn as well as a straight line in between. In the concept of a trajectory chain, the trajectory to be created is divided into segments with a speed-dependent length of $v_{Es} \cdot \Delta t$. Within a chain link, the orientation is constant. Changing the orientation relative to the previous link is only possible in the chain joints. The maximum possible deflection between two links $\Delta\psi_{max}$ corresponds to the maximum yaw rate at the respective speed.

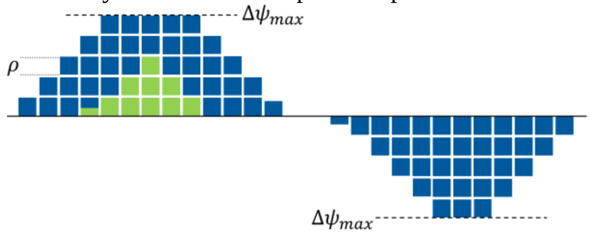

Figure 6. Pyramid algorithm

A so-called pyramid algorithm calculates how far a chain link is actually deflected within the left and right turn. This is because the yaw rate between the links can only change by a fixed value, which must be smaller than the maximum yaw acceleration of the ego-vehicle. To create an s-bend, the difference between the orientation of the ego-vehicle at the current time ψ_E and the orientation of the target trajectory ψ_{Ecr} is needed first. This difference angle forms the core of the first pyramid, which is shown in green in Fig. 6. Successively, elements are added to this pyramid and to a second pyramid representing the counter-steering, until the necessary lateral motion to the target point $\boldsymbol{P}_L$ or $\boldsymbol{P}_R$ is reached.

IV. Test Scenarios

The criteria for triggering collision avoidance functions presented in the previous sections will now be illustrated by means of three intersection scenarios. The safety function is triggered as soon as at least one of the values of TTS_L, TTS_R or TTB falls below the critical time threshold t_m, which is 0.05 s in this paper. According to Algorithm 1, the type of maneuver $none$, $brake$, $swerve\ left$ or $swerve\ right$ depends both on the available space and on which criterion has undershot the threshold.

TABLE I shows the speeds of the ego-vehicle and of the critical co-vehicle at the time the safety function is triggered. Furthermore, the distance between the two vehicles at the time of triggering and their minimum distance within the entire scenario are also listed there.

TABLE I. Kinematic Data of the Scenarios

Number of scenario	1	2	3
Velocity ego-vehicle [km/h]	48.509	52.841	52.961
Velocity co-vehicle [km/h]	22.585	30.816	31.445
Distance when triggering [m]	7.950	22.068	27.897
Minimum distance during maneuver [m]	0.685	2.570	1.511

In the first two traffic situations, shown in Fig. 7 and Fig. 11 a black vehicle changes from a secondary road to a main road ignoring the right of way of other road users. The white ego-vehicle is initially on the right lane of a main road. A few meters in front of it is a red vehicle on the middle lane. Additionally, a green vehicle drives in the opposite direction.

In each time step, the ego-vehicle predicts within the time horizon $t_{pred} = 2\ s$ its own motion and that of the other road users, having a maximum distance of $v_{Es} \cdot t_{pred}$ from the ego-vehicle. In the traffic flow simulation in SUMO, all vehicles in the road network have access to the complete kinematic data of all road users and the entire infrastructure at each time step. However, this does not represent the real traffic. For this reason, only the information about objects located within a radius of 100 m from the ego-vehicle is used. In the vehicle dynamics software CarMaker, on the other hand, only kinematic data are used to calculate the criticality, which can be measured with the sensors in the ego-vehicle. The range and aperture angle of these sensors are limited. In both simulation environments, the vehicles can only move along and perpendicular to fixed tracks during the prediction, as already mentioned in section III.A.

A. Black Vehicle Turning Right

The prediction in a time step ends either when the prediction horizon is reached or immediately if a collision with another vehicle is detected. In the scenario shown in Fig. 7, a black vehicle turns right from a side road into the main road. It disregards the right of way of the white ego-vehicle. After a collision has been detected within the prediction horizon (cf. Fig. 8), a second prediction of the motion of the co-vehicles, which starts at the predicted time of collision, determines the necessary lateral motion to avoid a collision. Fig. 9 shows that both the middle and the right lane can be used for an evasive maneuver, because the red vehicle is far ahead of the ego-vehicle. Since the ego-vehicle collides with the rear of the black vehicle, equation (10) needs to be used to calculate the time required for braking. The time course of the criteria TTS_L and TTB in Fig. 10 illustrates that an evasive maneuver can be started much later than an emergency stop, because TTS_L falls below the threshold t_m only after 1.2 s. For this reason, an evasive maneuver is performed in this scenario. The corresponding trajectory is also shown in Fig. 9. The criterion TTS_R is not used in this scenario, since the ego-vehicle is already on the right lane. Over time, there is an increase from 0.8 s to a maximum at 1.1 s. The reason for this is that the ego-vehicle already reacts to the co-vehicle. The deceleration of about 3m/s^2 is not sufficient to prevent a collision which can only be avoided by the intervention of the safety function.

Figure 7. Right turn of the black vehicle in CarMaker

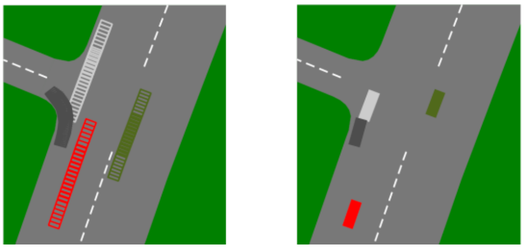

Figure 8. Predicted collision with a black vehicle turning right (Left: whole prediction, Right: collision time step)

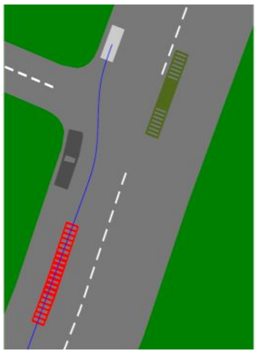

Figure 9. Black vehicle turning right: Prediction to find to lateral distance for swerving and the evasion trajectory

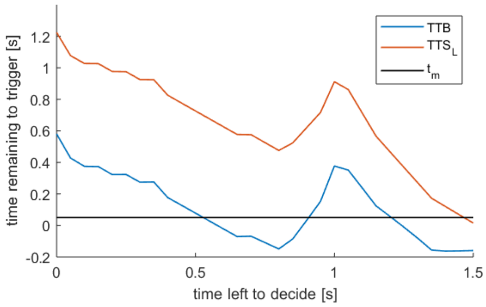

Figure 10. Black vehicle turning right: Remaining time to trigger

B. Black Vehicle Turning Left

In contrast to the first scenario, the black vehicle in Fig. 11 now turns left into the main road. The prediction in Fig. 12 shows that the white ego-vehicle collides almost vertically with the driver's side of the black vehicle. For this reason, equations (11)-(14) must be used to determine the time required for the necessary braking maneuver. To determine TTS_L and TTS_R, a prediction of the motion of the co-vehicles, which starts at the time of collision, is performed in the same way as in the first scenario. Swerving is not possible here because, as seen in Fig. 13, the turning black vehicle blocks all lanes. It is also not possible to switch to the opposite lane, since the green vehicle is already in the relevant area. For this reason, the situation can only be solved by a braking maneuver. The same result is obtained when considering the temporal courses of TTB and TTS_L (cf. Fig. 14). Due to the motion of the black vehicle perpendicular to that of the white ego-vehicle, the time for the necessary lateral motion is significantly longer than the duration of a braking maneuver.

Figure 11. Left turn of the black vehicle in CarMaker

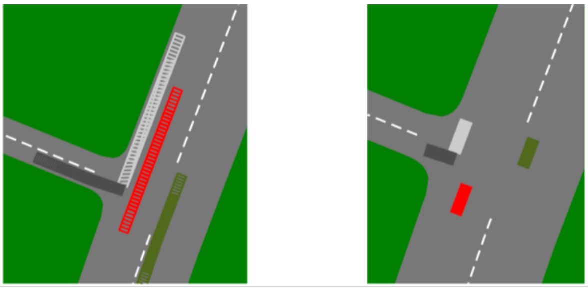

Figure 12. Predicted collision with a black vehicle turning left (Left: whole prediction, Right: collision time step)

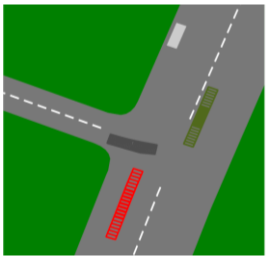

Figure 13. Black vehicle turning left: Prediction to find to lateral distance for swerving

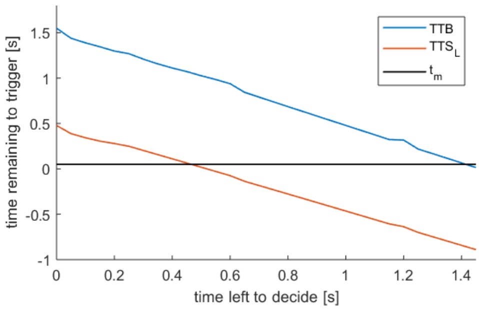

Figure 14. Black vehicle turning left: Remaining time to trigger

C. *Green Vehicle Turning Left*

As in the two previous scenarios, the white ego-vehicle and the red co-vehicle drive straight ahead on the main road in Fig. 15. The black vehicle turns right into the main road. The green vehicle turns left into the secondary road. The latter two vehicles ignore the right of way of the other two ones. The prediction of the motion of the road users shown in Fig. 16 detects a collision between the white ego-vehicle and the green co-vehicle. From the prediction of the motion of the co-vehicles after the predicted collision time in Fig. 17, it can be concluded that the critical situation can only be solved a braking maneuver, as the green vehicle blocks all lanes.

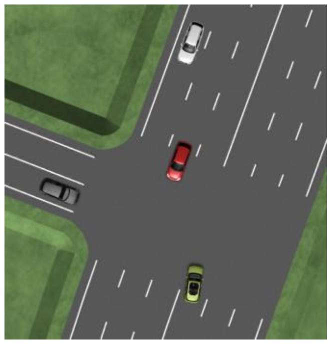

Figure 15. Left turn of the green vehicle in CarMaker

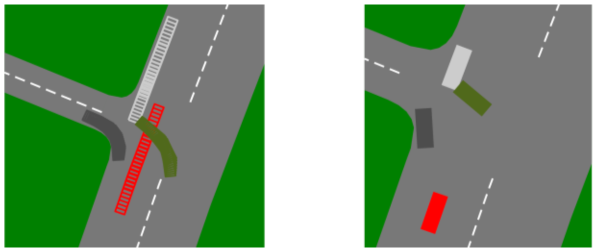

Figure 16. Predicted collision with a green vehicle turning left (Left: whole prediction, Right: collision time step)

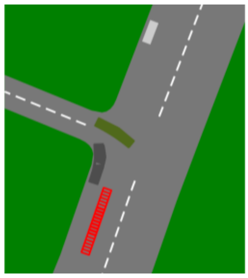

Figure 17. Green vehicle turning left: Prediction to find to lateral distance for swerving

In the temporal course of the criticality measures in Fig. 18 it is noticeable that at the beginning the TTS_L is considerably longer than the TTB. The reason for this is that the black vehicle is initially identified in the prediction as the one with which a collision could occur. Since it approaches the ego-vehicle from the right, the critical situation can be solved by an evasive maneuver. From 0.1 s on, however, the time course of the two criticality measures changes in such a way that a collision with the green vehicle is now considered more likely. As this vehicle is moving perpendicular to the trajectory of the ego-vehicle, the evasive maneuver must be started much earlier than braking.

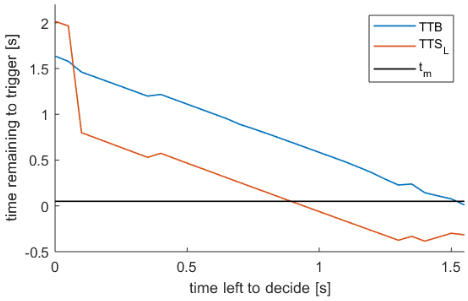

Figure 18. Green vehicle turning left: Remaining time to trigger

V. Test Scenarios

This paper presents criteria to identify critical situations in a traffic flow simulation and to evaluate the triggering decision of a vehicle safety function. To ensure that collisions between the traffic participants can occur, changes must first be made in the driver model in SUMO. In each time step, the ego-vehicle in SUMO and CarMaker predicts its own and the other road users' motion. If there is an overlapping of the vehicle hulls, this scenario is considered relevant for the test of the safety function in CarMaker. The TTB, TTS_L and TTS_R criteria determine when the safety function must be triggered at the latest and which maneuver should then be performed. To further illustrate the presented approach, the procedure is applied to three intersection scenarios. The tool chain presented in [6] together with the criteria from the present paper can be used in future work to perform a large-scale validation of vehicle safety functions with realistic scenarios. For this purpose, the TTC criterion is used to select those scenarios from the urban traffic that would end in a collision if the safety function did not intervene. The triggering behavior can be assessed by comparing the TTS and TTB with the actual triggering times. In addition, the performance of the safety function in handling the critical situation can be evaluated based on the criteria presented

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Peter Riegl conducted the research. All three authors wrote the paper and had approved the final version.

References

[1] H. Winner, "ADAS, Quo Vadis?" in *Handbook of Driver Assistance Systems: Basic Information, Components and Systems for Active Safety and Comfort*, H. Winner, S. Hakuli, C. Lotz, C. Singer, Springer Reference, Cham, 2016, pp. 1557-1584.

[2] G. Dhadyalla, N. Kumari, and T. Snell, "Combinatorial testing for an automotive hybrid electric vehicle control system: A case study," *Cleveland*, OH, 2014, pp. 51-57.

[3] C. E. Tuncali, G. Fainekos, H. Ito, and J. Kapinski, "Simulation-based adversarial test generation for autonomous vehicles with machine learning components," Changshu, 2018, pp. 1555-1562.

[4] T. Menzel, G. Bagschik, and M. Maurer, "Scenarios for development, test and validation of automated vehicles," in *Proc. IEEE Intelligent Vehicles Symposium (IV)*, Changshu, 2018, pp. 1821-1827.

[5] P. A. Lopez, *et al*., "Microscopic traffic Simulation using SUMO," in *Proc. 21st International Conference on Intelligent Transportation Systems*, Maui, HI, 2018.

[6] P. Riegl, A. Gaull, and M. Beitelschmidt, "A tool chain for generating critical traffic situations for testing vehicle safety functions," in *Proc. IEEE International Conference of Vehicular Electronics and Safety (ICVES)*, Cairo, 2019.

[7] G. R. D. Campos, *et al*., "Collision avoidance at intersections: A probabilistic threat-assessment and decision-making system for safety interventions," in *Proc. 17th International IEEE Conference on Intelligent Transportation Systems*, Qingdao, 2014, pp. 649-654.

[8] D. Smith, S. Djahel, and J. Murphy, "A SUMO based evaluation of road incidents' impact on traffic congestion level in smart cities," in *Proc. 39th Annual IEEE Conference on Local Computer Networks Workshops*, Edmonton, 2014, pp. 702-710.

[9] S. Krauss, P. Wagner, and C. Gawron, "Metastable states in a microscopic model of traffic flow," *Physical Review E*, vol. 55, no. 5, pp. 5597-5602, 1997.

[10] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," Physical Review E, vol. 62, no. 2, pp. 1805-1824, 2000.

[11] R. Wiedeman and U. Reiter, "Microscopic traffic simulation: the simulation system MISSION, background and actual state," in CEC Project ICARUS (V1052) Final Report, vol 2, Brussels, 1992.

[12] M. Fellendorf and P. Vortisch, "Validation of the microscopic traffic flow model VISSIM in different real-world situations," in *Proc. 80th Annual Meeting of the Transportation Research Board*, Washington, DC, 2001.

[13] B. Higgs, M. Abbas, and A. Medina, "Analysis of the Wiedemann car following model over different speeds using naturalistic data," in *Proc. 3rd International Conference on Road Safety and Simulation*, Indianapolis, 2011.

[14] C. Ackermann, *et al*., "Design of a decision maker for an evasive or braking maneuver for collsion avoidance," in *Proc. 14. Internationales Stuttgarter Symposium - Automobil- und Motorentechnik*, Springer Vieweg, Wiesbaden, 2014, pp. 401-415.

[15] B. Simon, P. Riegl, and A. Gaull, "Motion planning with flexible trajectory chains," in *Proc. IEEE International Conference on Vehicular Electronics and Safety (ICVES)*, Madrid, 2018.

[16] B. Simon, P. Riegl, and A. Gaull, "Path planning with discrete geometric shape patterns," in *Proc. IEEE International Conference on Vehicular Electronics and Safety (ICVES)*, Cairo, 2019.

Peter Riegl is a PhD student at the University of Applied Sciences Ingolstadt, focusing on the validation of vehicle safety functions in simulation environments that are based on real road traffic. He received his diploma in mechanical engineering from the Technical University of Dresden in 2013 and his master's degree in automotive engineering from the University of Applied Sciences Ingolstadt in 2017. His main research interests include the modeling of driving behavior, vehicle dynamics simulations as well as testing and evaluation of vehicle safety functions.

Andreas Gaull received his diploma in mechanical engineering from the Technical University of Munich in 2005. He worked as a research assistant from 2005-2010 and received his doctorate in mechanical engineering in 2010. From 2010 to 2014 he worked at LMS International (now Siemens PLM) and at Audi AG. Since 2014 he is professor for Chassis Technology and Dynamics at the University of Applied Sciences Ingolstadt where he is also a member of the automotive research center CARISSMA. His research focuses on multibody simulation and vehicle dynamics, in particular in the context of automotive safety systems and automated driving functions.

Michael Beitelschmidt received his diploma in mechanical engineering from the Technical University of Munich in 1992, where he worked as a research assistant from 1992-1998 and received his doctorate in mechanical engineering in 1998. From 1998 to 2004 he worked as a department head at Sulzer Innotec. From 2005 to 2010 he was professor for vehicle modeling and simulation and since 2010 he is head of the chair for dynamics and mechanical design. His research focuses on multi-body and structural dynamics in the application to vehicles and drives, acoustics, drones and robotics as well as the development of novel simulation methods for these application areas.