Abstract—Autonomous vehicles navigation in complex scenarios is still an open issue. One of the major challenges is the safe navigation of autonomous vehicles on roads open to public traffic. Indeed, behaviors and intentions of human-driven vehicles are hard to predict and understand. In this paper, we propose a strategy to make an autonomous vehicle able to cross safely a roundabout. Our approach relies on High-Definition (HD) maps with lane level description which allows to predict the future situation thanks to the concept of virtual vehicles. This method handles safely collision avoidance and guarantees that no priority constraint is violated during the insertion maneuver without being overly cautious. The performance is evaluated with the SUMO simulation framework. An highly interactive vehicles flow has been generated using real data from the INTERACTION dataset. We also propose strategies to extend our algorithm to multi-lane roundabouts and report how these extensions behave in terms of safety and traffic flow.

I. INTRODUCTION

Nowadays, Autonomous Driving (AD) vehicle technology is a very active field of research. In a not-too-distant future, one can imagine that AD vehicles will co-exist with manually-driven (MD) vehicles. Safe navigation has to be guaranteed in complex dynamic environments like in intersections, lane merging or roundabouts, where the risk of accident is one of the highest in the road environment.

Many works have been done for intersection crossing with only AD vehicles (see [21], [16] and [25]) and with priority constraints [4]. Some approaches solve the intersection crossing problem with optimization [21] or with model predictive control [17]. Computing and avoiding the situations where a collision is unavoidable even if an emergency braking is executed is another way to address the problem [2]. Finally, the autonomous intersection management protocol (AIM) [3] [23] and virtual platoon methods [20] [5] have shown interesting properties.

In a mixed traffic environment with AD and MD vehicles, avoidance sets can be computed with the reachability theory [15] [6]. In [22] the authors implement a collaborative motion planning algorithm that cooperates with MD vehicles. Other works as in [11] estimate the MD vehicle behavior in the worst-case (e.g. the acceleration of a MD car approaching an intersection is considered to be the highest). In [9], the behavior of a MD vehicle is modeled with a mathematical representation that captures the human way of driving.

In [19], we have have shown that using virtual vehicles along the lanes of an HD map is very efficient to predict the dynamic situation in a roundabout. This paper extends this approach with the following contributions:

- A safe and priority-preserving curvilinear decision method for one lane roundabout with several MD vehicles is fully described,
- The generation of a realistic vehicles flow for simulation studies using real data from the INTERACTION dataset [27] to be used in a SUMO-based simulation environment [14],
- Three extensions are presented to handle two-lane roundabouts with a navigation strategy in the outermost lane of the roundabout,
- An evaluation of these extensions in terms of safety and flowability of the insertion.

The paper is organized as follows. In the next section, we first introduce curvilinear representation along with high definition (HD) maps. Then, Sec. III formalizes our case study and states the different hypothesis. In Sec. IV, we present our roundabout crossing algorithm for both the single lane case and its extension to the two-lane one. The simulation of a vehicle flow using real traffic data is explained in Sec. V followed by experimental results in Sec. VI.

II. CURVILINEAR COORDINATES ALONG HD MAPS

In complex urban environments, multiple vehicles share the driving road and it is important to properly represent their spatial positions with respect to each other. Euclidean coordinates are often not well suited to represent spatial relation between road user. Curvilinear coordinates along HD maps provide an efficient way to manage the interactions between road users. Curvilinear frameworks are often preferred as they capture the lane level interaction between the vehicles,
for example in terms of conflicting trajectories w.r.t. the AD vehicle navigation corridor [11, 24].

The principle of curvilinear coordinates is to map an Euclidean pose, i.e., position and heading angle, onto a curvilinear pose with respect to a geometrical curve. This curve typically represents the nominal path of the vehicles. In many driving contexts, the center line within a lane is a sufficient approximation of the path that the vehicles follow. Throughout this paper, we assume that the center lines are stored in a HD map.

The first step of the curvilinear representation is to get the lane followed by the vehicles. For an AD vehicle, it is simply given by the path planning module. In contrast, the driving lanes of the MD vehicles need to be estimated by a map-matching procedure. It is well known that map-matching ambiguities may arise when a road splits or when the vehicle changes lanes. We do the map-matching process using the method described in [19] where multiple candidate lanes can be simultaneously occupied by creating virtual copies of the MD vehicle, one for each lane possibly occupied.

Once the path has been defined, the curvilinear abscissa $s$ of the vehicle is computed from an origin point of the path. This representation is particularly useful to do virtual platooning with a vehicle driving on another road [20]. By taking the curvilinear origin at the intersection of two lanes, the curvilinear abscissas can be used to compute the relative virtual gap between two vehicles lying on two intersecting lanes as illustrated in Fig. 2. Practical computation of the curvilinear coordinates can be found in [10].

In this work, we use an HD map that explicitly represents all the lanes by polylines, i.e., sequences of line segments. Another commonly used map representation is the lanelet one [1] which uses the left and right bounds of the lane. In this case, a conversion into a center line representation is needed beforehand as done in [12].

### III. Problem Statement

The state of a vehicle is represented as $V_i = [s_i, \bar{s}_i, v_i, P_i]$, where $[\bar{s}_i, s_i]$ are the lower and upper bounds over the curvilinear occupancy of $V_i$ encompassing both the size and the uncertainty (bounded) over its position estimate, $v_i$ is its estimated speed and $P_i$ its predicted path. These parameters are typically provided by a perception system able to detect, track and map-match the vehicles nearby.

In this work, we constrain the AD vehicle to navigate only on the outermost lane of the roundabout. In other words, we do not allow the AD vehicle to overtake and change lane during the roundabout crossing. With this simplification, the navigation algorithm has only to control the longitudinal motion of the AD vehicle to perform the task, the lateral control being done by lane keeping. The traffic regulations that we consider are as follows: the priority lanes are situated inside the roundabout ring while the non-priority ones are in the entering branches. Three rules are important in our strategy. An AD vehicle

1. Has to keep a safe inter-distance w.r.t. the vehicle ahead.
2. Can enter into the roundabout if it does not make another vehicle inside the roundabout decelerate (vehicles inside the roundabout have the right of way).
3. Avoids stopping on a road lane except if rules (1) or (2) require it to do so.

This means that a vehicle on a non-priority lane is allowed to enter into the roundabout only if the insertion maneuver does not force a priority vehicle to decrease its speed. A priority vehicle must follow its reference speed profile and should perform an inter-distance regulation only with respect to vehicles that have the same priority level. It is also desired that the AD vehicle makes an insertion as smooth as possible without making a stop at the entrance of the roundabout which requires it to anticipate the behavior of the other vehicles.

### IV. Roundabout Insertion Algorithm

#### A. Roundabout Lanes Classification

In accordance to the problem statement and the three rules listed in Sec. III, we propose to decompose a roundabout into three zones as illustrated in Fig. 1. Each zone describes sub-steps of the insertion maneuver, as follows:

a) The Decision Zone (in green in Fig. 1) is before the merging into the roundabout ring. In this zone, the AD vehicle does not have priority w.r.t. vehicles in the roundabout. It has to evaluate the possibility of a safe insertion without violating priority constraints.

b) The Transition Zone (in yellow in Fig. 1) is the last part of the entering lane where it merges with the roundabout ring. In this part, the AD vehicle performs a transition to enter into the roundabout. When the AD vehicle is here, a safety inter-distance w.r.t. an eventual incoming MD vehicle on the roundabout ring must be kept in order to allow a safe insertion.

c) The Ring Zone (in red in Fig. 1) corresponds to the roundabout ring. In this zone, the insertion is completed and the AD vehicle follows the nearest MD vehicle in the roundabout or drives at its nominal speed if it is alone.

As a consequence, the crossing of the whole transition zone is taken into account in the decision making procedure. In facts, once a vehicle enters into that zone, it can no longer change its decision. If it needs to stop, it means that the decision of entering into the roundabout has been wrongly taken which makes the approaching vehicle to decelerate in order to avoid a collision. Finally, once the AD vehicle has
crossed the transition zone, it gains the same priority as all the vehicles in the roundabout ring.

**B. Single-lane roundabout**

The principle of virtual platooning is to establish a crossing order for complex scenarios such as intersection [20] or roundabout [19]. By using intervals \([s, \bar{s}]\) to represent the curvilinear occupancy of vehicles, cases where no total order between vehicles may arise. Let us consider the case illustrated in Fig. 3a, where the trajectory of the AD vehicle (in blue) crosses the one of an MD vehicle (in green) at a point \(n\). Let us define \([\underline{s}, \bar{s}]\) the curvilinear occupancy of the AD vehicle w.r.t. the origin \(n\), and similarly \([\underline{s}_i, \bar{s}_i]\) for a MD vehicle \(V_i\). Fig. 3b illustrates the six possible relative positions between the AD vehicle and the MD ones with their corresponding virtual projections (Fig. 3c).

Note that if there is no priority constraint between the vehicles, using intervals does not lead to a unique order among the vehicles. Therefore, one needs to choose an insertion policy [19]. This issue is out of the scope of this paper, as we consider that vehicles inside the roundabout have the highest priority. From Fig. 3c, one can see that in all the cases (2), (3),..., (6), the AD vehicle cannot be guaranteed to be in front of the incoming MD vehicle. The only case where the AD vehicle may expect to be in front of the MD vehicle is the case (1). As soon as the AD vehicle enters the transition zone, it starts to have physical interaction with other vehicles in the ring zone. Therefore, it needs to be in the case (1) and guarantee that the (virtual) inter-distance between the two vehicles \(d_{ref} = s - \bar{s}_1\) is greater than a minimal safety distance \(d_{safe}\) during the whole time it takes for the AD vehicle to insert into the roundabout (i.e., for the whole transition from the decision zone to the Ring zone).

Let us consider the scenario depicted in Fig. 2, where the AD vehicle \(V\) is in the decision zone of the roundabout and an incoming priority MD vehicle \(V_1\) in the ring zone. Let us define \(t_0\) as the time when the front part of the AD vehicle enters into the transition zone. At a given time \(t < t_0\) (i.e., when the AD vehicle is still in the decision zone), one can see that having

\[
d_{ref}(t_0) = s(t_0) - \bar{s}_1(t_0) > d_{safe},
\]

is not sufficient for the AD vehicle to ensure a safe and priority preserving insertion maneuver. Indeed, if the speed \(v_1\) of vehicle \(V_1\) is greater than the speed \(v\) of \(V\), the inter-distance between the two vehicles will shrink over time. This shows that vehicle dynamics must be taken into account at the decision making level.

The AD vehicle needs to guarantee that Eq. (1) will be satisfied during the whole insertion maneuver, i.e., from the moment that the upper bound \(\bar{s}\) enters the transition zone until the lower bound \(s\) leaves it. Let \(\Delta t\) be the time needed by the AD vehicle to go completely through the transition zone, i.e., the back of the AD vehicle has exited it. The decision to enter the roundabout is taken if:

\[
\forall t \in [t_0, t_0 + \Delta t], \quad s(t) - \bar{s}_1(t) > d_{safe}.
\]  

In order to guarantee the inequality of equation 2, we need to know how both \(s\) and \(\bar{s}_1\) evolves over time. In this work, we assume that both vehicles drive at a constant speed. This assumption may seem simplistic, but it is representative of the driver behavior in the INTERACTION dataset. Within the roundabout ring of the INTERACTION dataset, the speed profiles of the MD vehicles have a standard deviation less than 1 m/s in average. Under this assumption, we have \(\Delta t = l/v\), where \(l\) is the length of the transition zone, and the kinematics of each interval can be expressed as follows (bearing in mind the aforementioned considerations):

\[
s(t) = s(t_0) + v \cdot (t - t_0),
\]

\[
\bar{s}_1(t) = \bar{s}_1(t_0) + v_1 \cdot (t - t_0).
\]

Substituting equations (3) and (4) in (2), we obtain

\[
[s(t_0) - \bar{s}_1(t_0)] + (v - v_1) \cdot (t - t_0) \geq d_{safe}.
\]  

The inequality (5) needs to hold \(\forall t \in [t_0, t_0 + \Delta t]\).

If \(v > v_1\), it leads to

\[
s(t_0) - \bar{s}_1(t_0) - d_{safe} \geq 0.
\]

It means that if the AD vehicle drives faster than \(V_1\), it can insert if it is sufficiently ahead of \(V_1\) at \(t_0\).

Otherwise, if \(v < v_1\), we have

\[
s(t_0) - \bar{s}_1(t_0) - d_{safe} + \left(1 - \frac{v}{v_1}\right) l \geq 0.
\]

We can see that the relative speed \(v_1/v\) needs to be taken into account in the decision.

Inequality (7) allows us to decide if an AD vehicle has enough space to keep a safety inter-distance w.r.t. an eventual
incoming vehicle, knowing its velocity and its occupancy at

time \( t_0 \). In the case where equation (7) is not satisfied, the
ego vehicle slows down to perform a safe stop at the end of
the decision zone (that coincides with the give way marking).

Nevertheless, once the speed of the AD vehicle is close to
zero, it is difficult for the AD vehicle to find a sufficiently
large gap to effectuate the insertion. This is due to the
singularity present in equation 7 when \( v = 0 \). In facts,
considering the function \( \hat{h}(v) = (1 - \frac{v}{v_1}) \), one can see
that has a peak towards \(-\infty\) for \( v \to 0 \). This drastically
degrades the performances of our algorithm as once the AD
has stopped, it will only insert once no incoming vehicle is
present. To overcome this, we propose to replace \( \hat{h}(v) \) with
another function for the case \( v < v_1 \). In particular, we look
for a function that meets the following criteria:

1) As \( v \to 0 \), the value of the function becomes less
dependent on \( v \).
2) For \( v = 0 \) the value of the function depends at least
on \( v_1 \).

The main idea is to have a function that allows to set
safety gap that depends at least only on the other vehicle
speed \( v_1 \). This solution is convenient when the dynamics
of the system is not well known and we need to perform
a prediction without being too pessimist. In this work, we
choose a function with the following form:

\[
\hat{h}(v) = A \left[ \frac{1}{1 + e^{-\alpha(v-v_1)}} - \frac{1}{2} \right] \tag{8}
\]

Where the two parameters \( A \) and \( \alpha \) have been tuned during
the real experiments phase to choose a convenient insertion
maneuver VI. Equation 7 now becomes

\[
\begin{cases}
\delta(t_0) - \delta_1(t_0) - d_{safe} \geq 0 & \text{if } v > v_1, \\
\delta(t_0) - \delta_1(t_0) - d_{safe} + \hat{h}(v)l \geq 0 & \text{else}.
\end{cases} \tag{9}
\]

C. Two-lane Roundabout

We propose an extension of the aforementioned method to
navigate into two-lane roundabouts. We consider again the
roundabout shown in Fig. 1. One of the most difficult issues
in this scenario is the handling of vehicles that perform lane
change maneuvers from the inner ring of the roundabout to
outer one. In practice, it is very challenging to predict a lane
change maneuver, especially when vehicles attempt to do a
lane change with nudging [15]. We propose three different
strategies to handle the navigation of multiple vehicles inside
the a two-lane roundabout.

The first method consists in occupying systematically both
lanes of the roundabout ring if at least one lane is occupied.
This means that, if a vehicle is occupying the innermost lane,
the outermost one will result occupied too. Fig. 4a illustrates
this method. The second method consists in occupying the
lanes only when there is a significant physical occupancy of
a vehicle. This implies that during a lane change maneuver
there is always one occupied lane at most, i.e., the outer lane
becomes occupied only when the first half of the vehicle
has already crossed the bound between the two lanes of the
roundabout (Fig. 4b). The third method occupies the outer
lane only when the intention of a vehicle to change lane
is detected. This approach is shown in Fig. 4c. In this case, we
assume to have a system that is able to predict when a driver
decides to change lane, e.g., by detecting blinkers or lateral
distance from the lane center.

V. Vehicles Flow Generation

To simulate a realistic vehicles flow, we propose to use
data from real traffic. Several datasets can be found in
the literature such as Common Road [18], ACFR [28] or
INTERACTION [27]. We have chosen the latter one because
it provides data for highly interactive road users. For each
roundabout of the dataset, traffic flow data recordings at
microscopic level in a dense traffic flow situation is provided.

To quantify the degree of interaction between traffic
agents in the vehicle flow, the authors propose to use the
\( \Delta TTIC_{min} \) metric (minimum time-to-conflict-point differ-
ence for an interactive pair of vehicles [26]). The resulting
\( \Delta TTIC_{min} \) computations for every scenario are provided
too. For a given interactive pair of vehicles, the corresponding
\( \Delta TTIC_{min} \) can be computed as

\[
\Delta TTIC_{min} = \min_{t \in [T_{start}, T_{end}]} (TTIC_1^t - TTIC_2^t),
\]

where \( TTIC_i^t = \Delta d_i^t / \upsilon_i^t, i = 1, 2 \), is the traveling time to
the conflict point of each vehicle in the interactive pair. If
\( \Delta TTIC_{min} \leq 3 \) s, it means that there is interaction be-
tween the vehicles. Moreover, an interaction between a pair
of vehicle is defined intense if \( \Delta TTIC_{min} \leq 1 \) s. In this
study, we focused on the data from the USA_Roundabout_FT
scenario because it is the most interactive two-lane round-
about with the highest number of vehicles and recorded
sequences.

The main idea of this part is to generate with the SUMO
simulator a flow of vehicles based on these traffic data that
has the same degree of interaction in terms of \( \Delta TTIC_{min} \).
We decided to use SUMO because it is a widely used simulator for microscopic traffic simulation in the field of collision avoidance [13] [7]. This has several advantages. First, it allows us to quantify the performance of our algorithm w.r.t. the interaction degree of the traffic scenario. Then, we can obtain a vehicle flow with the same degree of interaction on a different roundabout testbed (for example the one in Fig. 1). Finally, a simulated vehicles flow in SUMO can react dynamically to the ego vehicle decisions (e.g. brake when the insertion maneuver is too aggressive), while it is not the case for a recorded flow dataset.

To do this, we have implemented an algorithm to compute common intersection points between vehicles trajectories (similar to the one presented in section IV-B) and the corresponding $\Delta TTIC_{min}$ for every interactive pair. Note that vehicles trajectories are not present in the INTERACTION dataset and they need to be computed offline. Furthermore, the distances $\Delta d^i$ have been computed in a curvilinear framework considering an HD map representation obtained from the Lanelet2 HD maps provided with the dataset visualization interface.

In order to feed SUMO simulator, we compute for every track of the USA_Roundabout_FT the arrival times, arrival speeds, inter-distances w.r.t. the vehicle ahead and behind. Fig. 5a shows the resulting inter-distances distributions, while Fig. 5b gives the inverse cumulative distribution of vehicle speeds. Regarding the arrival times, a uniform distribution has been estimated.

On the basis of this data, we randomly sampled one value for each distribution in order to create vehicle objects with the same properties as the true ones. Considering the inter-distances distribution, only the one w.r.t. vehicles ahead has been used as the minGap distance for vehicle objects, with an outlier rejection for too large gaps (right tail of Fig. 5a).

Finally the speed of each vehicle has been bounded by the speed limit provided by the road shape and the path of each vehicle object has been chosen randomly giving more weight to routes intersecting the AD vehicle one.

The output of the system is a traffic flow with the $\Delta TTIC_{min}$ distribution similar to the one computed with the real traffic data. Fig. 5 illustrates the output of the simulation for a recorded traffic sequence of INTERACTION. One can see that a highly dynamic behavior ($\Delta TTIC_{min} \leq 1$ s) is present in both scenarios.

The high degree of interaction between interactive pairs is mostly due to the vehicles navigation into the double lane of the roundabout ring. In other words, if two or more vehicles navigate very close to each other on parallel lanes of the roundabout ring, a high interaction between them exists.

VI. EXPERIMENTAL SETUP AND RESULTS

To validate our approach, we have used two simulators. The SUMO simulator [14] was used for microscopic traffic flow generation while a ROS-based simulation framework was used to implement the navigation algorithm and the AD vehicle dynamics. The coupling and synchronization of both simulators has been achieved with the TraCi SUMO library and its Python API. An explanation about time synchronization and coupling of the two simulators can be found in [8].

Furthermore, we have imported in the SUMO simulation environment the HD map representation of the testbed roundabout. This has been achieved with the Netedit and Netconvert tools included in the SUMO suite [12].

For each simulation, a random high density vehicle flow that meets the $\Delta TTIC_{min}$ criterion has been generated over a fixed time horizon $T = 200$ s. Each simulation contained between 200 and 400 seconds for a total amount of more than 1 hour of simulation. The number of vehicles for each flow randomly varied between 50 and 175, for a total amount of more than 5000 vehicles.

These limits have been chosen to capture a wide range of scenarios, starting from a sparse traffic flow until a more dense vehicle stream. Moreover, to simulate both localization (AD vehicle) and perception (MD vehicles) uncertainty, we consider a $\pm 1$ m bound to add at the curvilinear interval $[s_i, \bar{s}_i]$ and $[\bar{s}_i, \bar{s}]$, which represent the projection of the vehicle footprint on the centerlane.

A. One lane Roundabout

To experimentally validate our approach, we consider several scenarios with the aforementioned technique to generate high density traffic flows. The simulations have been carried out with the HD map representation of the “Guy Dénielou” roundabout, in the city of Compiègne, France (Fig. 1). In
the first part of the tests, only the outermost lane has been considered.

To better quantify the performance of our algorithm, we have computed the distributions of the inter-distances w.r.t. the vehicle ahead and behind during the crossing of the transition zone. Fig. 6 shows the inter-distances distributions for both the ahead and behind gaps. As one can see, the behind safety gap meets always the safety criterion. Conversely, considering the gap w.r.t. vehicles ahead, there is a slight violation of the safety bound. This violation is due to some controller imperfection and can be neglected. Moreover, we are interested in quantifying the insertion rate w.r.t. the vehicles flow. Table I illustrates the average insertion rate and waiting time as function of the number of vehicles in the flow. The insertion time is computed considering that the total length of the decision and transition zones is 33.4 m and the nominal speed in this zone is 30 km/h, which gives us a nominal waiting time of 4.2 s.

As one can see, the number of insertions decreases w.r.t. the number of vehicles in the flow. Conversely, the waiting time increases.

Finally, to validate our strategy in a real scenario, we implemented the whole architecture on an autonomous Renault Zoé and we tested the insertion first in an hybrid environment (i.e. with simulated vehicles moving on the experimental circuit) then with real road agents detected with a Lidar-based perception system

B. Two-lane Roundabout

For every strategy listed hereafter (Section IV-C), several simulations on SUMO with a randomly generated vehicle flow on the full roundabout (two lanes) have been carried out. To properly model a lane change maneuver inside SUMO, we consider a nonzero duration for lane changes and the other hypotheses done in [12]. In order to quantify the performance of our approach, Fig. 7 and 8 show respectively the inter-distances w.r.t. the vehicles ahead and behind for the three strategies. The first method always ensures safety. For the other two, it is not the case. In fact, the second method violates the safety constraints in both forward and backward cases. This is due to the late detection of lane changes. As a consequence of that, it can happen that the AD vehicle performs an insertion maneuver when another vehicle has already decided to change lane. In this situation, two scenarios can happen: The AD vehicles force the other car to change its intentions or the other car complete the lane change despite of the AD vehicle presence. In the first case, this driving behavior is called nudging. In the second case, the other vehicle cut off the road of the AD vehicle, resulting on an hazardous maneuver or on a collision. The second case behaves as a compromise between the two.

To analyze this aspect, Table II shows the collision probabilities, average insertion times and average waiting times at give way for the three strategies. Logically, if one wants to have safety ensured, the average waiting time increases. This is due to the fact that with a double lane roundabout the vehicles capacity on the ring is more w.r.t. a single lane one. As a consequence, the first method (Fig. 4a) tends to be too overly conservative. Conversely, the second method (Fig. 4b) is much more aggressive because of the lack of lane change prediction. However, this method has a large unsafe set of configurations. Finally, the third method (Fig. 4c) behaves as a compromise in-between.

VII. Conclusion

In this work, an method to perform a safe roundabout crossing based on virtual vehicles methods has been presented. It has been tested under a simulated traffic flow similar to real data. The degree of interaction of the generated flow has been used to generate a scenario close to real world driving. We have shown that the proposed insertion maneuver on a single-lane roundabout is always safe. We have also proposed some performance indexes to evaluate its efficiency.
TABLE I: The average insertion times, the percentage of average waiting time relative to the nominal case and the average number of insertion as function of vehicles flow for the single-lane roundabout case.

<table>
<thead>
<tr>
<th>Crossing Time (s)</th>
<th>Number of Insertions</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>24</td>
</tr>
<tr>
<td>75</td>
<td>21</td>
</tr>
<tr>
<td>100</td>
<td>18</td>
</tr>
<tr>
<td>125</td>
<td>16</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Probability of a safety bound violation (ahead and behind), average crossing times and effective average waiting times for the three methods.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
</tr>
<tr>
<td>Two lanes occupancy</td>
</tr>
<tr>
<td>Only one lane occupancy</td>
</tr>
<tr>
<td>Intention detection</td>
</tr>
</tbody>
</table>

in terms of traffic fluidization. To handle the problem of a double-lane roundabout, a lane change intention detector is required to obtain safe and not overly cautious performance. If this technology is unavailable, we suggest to use instead a worst-case occupation method that provides always a safe insertion.

It is our opinion that, in order to safely cross a roundabout ensuring safety and without being too overly conservative, an accurate lane change intention detection algorithm is required. Moreover, if the lane change detector is able to detect also nudging, one can discriminate between a real intention of a driver to effectuate a lane change and a false alarm. This approach should add efficiency in the insertion maneuver, decreasing waiting times without compromising safety.

As a future perspective, the we aim to extend the tests with a real vehicle to more complex scenarios (e.g. two-lanes roundabouts) and to include in their architecture a lane change intention detector to validate experimentally the obtained results.

Acknowledgment: This work was carried out in the framework of Equipex ROBOTEX (ANR-10-EQPX-44-01) and Labex MS2T (ANR-11-IDEX-0004-02). It was also carried out within SIVALab, a shared laboratory between Renault and Heudiasyc CNRS/UTC, through the TORNADO project.

REFERENCES