IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 12, 2021, accepted August 23, 2021, date of publication August 31, 2021, date of current version September 13, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3109350

Interaction-Aware Intention Estimation at Roundabouts

VINICIUS TRENTIN, ANTONIO ARTUÑEDO, JORGE GODOY, AND JORGE VILLAGRA
Centre for Automation and Robotics (CSIC-UPM), 28500 Madrid, Spain

Corresponding author: Vinicius Trentin (vinicius.trentin@csic.es)

This work was supported in part by Spanish Ministry of Science and Innovation with the National Project NEWCONTROL under Grant PCI2019-103791, in part by the Community of Madrid through SEGVAUTO 4.0-CM Programme under Grant S2018-EMT-4362, and in part by European Commission and ECSEL Joint Undertaking through the Project NEWCONTROL under Grant 826653.

ABSTRACT Roundabouts have many benefits when compared with traditional signal-controlled intersections: improve safety, reduce delay, improve traffic flow, are less expensive, and occupy less area. The behavior of traffic participants is full of uncertainties in the real world. An automated system that relies only on its perception is unable to safely enter the roundabout until a large gap occurs or the vehicle approaching has actually left the roundabout or passed the conflict area. In order to improve the driving quality, autonomous vehicles should be able to infer the correct intention at roundabouts as early as possible. In this work, a method to classify the intentions of the surrounding vehicles at unsignalized roundabouts is proposed. For each vehicle at the scene, a Dynamic Bayesian Network is instantiated and the intentions are inferred using a particle filter.

INDEX TERMS Intention-detection, interaction-aware, roundabout.

I. INTRODUCTION

The anticipation of other vehicles' intentions is crucial to safely navigate through traffic. The behavior of traffic participants is full of uncertainties in the real world. In order to improve the driving quality, autonomous vehicles should evaluate the threats, taking seriously the ones with high probability to happen and not overreacting to the ones with low probability. Probabilistic intention and motion predictions are unavoidable to accomplish safe and high-quality decision-making and motion planning for autonomous vehicles [1]. Wrong predictions can result in either too conservative motions, such as unnecessary stops/yielding, or dangerous situations, like emergency brakes and collisions.

Roundabouts have many benefits when compared with traditional signal-controlled intersections: improve safety, reduce delay, improve traffic flow, are less expensive, and occupy less area [2]. Some mobility studies [3] point out that accidents that still happen in these situations are mainly due to failure to give way when entering the roundabout, which can be interpreted as a failure to infer the intentions of the drivers inside the roundabout.

The associate editor coordinating the review of this manuscript and approving it for publication was Christopher H. T. Lee.

To infer the intention of the drivers at intersections, some works use classifiers, such as particle filters or naive Bayesian classifiers, but they do not take into account the interaction between vehicles or only consider the most probable action. These can lead to suboptimal results in complex scenarios. Others use data-driven approaches, that do take into account the interaction but depend on a large amount of data and they cannot guarantee that the results can be applied to different scenarios.

In this paper, a method to classify the intentions of the surrounding vehicles at unsignalized roundabouts is proposed. The framework here presented is evaluated in 9 roundabouts with data from the datasets INTERACTION [4], rounD [5], and openDD [6]. The main contributions of this article are:

- A generic approach to handle almost any layout and number of vehicles to:
  - find all possible navigable corridors and their relations;
  - compute the expectation at intersections;
  - estimate the intention of vehicles at roundabouts.
- The evaluation is carried out with real data from public datasets.
- The proposed strategy is compared with similar approaches with quantitative results.

The remainder of this paper is organized as follows: Section II presents a review of some related works trying to solve the identified problem. Section III describes the architecture used with the proposed intention estimation been described in Section IV. Section V presents the datasets being used and shows the results for 9 simulations. Section VI concludes.

II. RELATED WORK

Roundabouts are an increasingly researched topic in many areas of interest, such as optimization of the geometry [7], cyclists safety [8], analyze the brainwaves going through a roundabout [9] and gap acceptance [10], [11].

Compared with highways, roundabouts, as other types of intersections, offer stronger interactions between vehicles, where the collision risk is higher. Since the intention of the other drivers cannot be measured directly, many works focus on ways to estimate it.

Intention inference can be considered a classification problem of driving behaviors and can be divided into two categories: index-based and machine learning approaches [12]. Index-based approaches include Time-to-Intersection, Time-to-Collision, Time-to-React, and can be used to estimate risk [13]. A few machine learning approaches found in the literature will be described below.

A framework for assessing traffic scenes with the interaction between traffic participants is presented in [14]. They transform the possible behavior patterns of the vehicles involved into hypotheses and compute the joint probability of each hypothesis by reconstructing the individual probability of each behavior. As a result, they obtain the fully interaction-aware joint probability distribution over all the hypotheses. Their approach grows exponentially as the situation complexity and the number of vehicles involved increase.

The authors in [15] implement a Dynamic Bayesian Network to reason about the situations and the risks at intersections on a semantic level. The risk is assessed based on the comparison of the intentions with what is expected from the drivers in a given scenario. They model the expected vehicle's motions based on the road network (stop signs, give away lines), distance to the intersections, and previous pose and velocity. The intention to stop is computed based on the previous intention and current expectation. With the intention and the maneuver, the future pose and velocity can be estimated. An evolution of this approach considering also lateral expectations has been recently presented in [16] and [17].

In [18] the authors use a Dynamic Bayesian Network with a particle filter to evaluate the interaction between vehicles and estimate their route and maneuver intentions. From these intentions, an action, represented by acceleration and yaw rate values, is obtained and the motion prediction is computed. This method considers only the most probable action for the whole time horizon of the prediction, which, in complex scenarios, may negatively influence the motion planning search space.

In [19], the authors present a probabilistic method to classify the driver intention at a roundabout with a naive Bayes classifier. To generalize to unseen roundabouts, they generate optimal trajectories given a map. With the optimal leaving and remaining trajectories, a Frenet frame is determined by temporally aligning them and averaging the aligned positions. The trajectories of the vehicles are projected onto the Frenet axis and the probabilities of leaving and remaining are obtained by computing the Bayes's theorem at k points of the trajectory. Another naive Bayes classifier is used in [20] where the authors present a probabilistic method to classify the driver intention at a roundabout using Bayesian update and Dynamic Time Warping (DTW). The distances of the trajectory of the vehicles to the optimized center points are computed with DTW and the probability of being in a given corridor is updated with Bayes's theorem.

In the field of data-driven approaches, many works use LSTM [21]–[23], graph neural networks [24], reinforcement learning [25] or transformer networks [26]. These techniques can take into account features that can be hardly modeled by traditional methods. Nevertheless, large amounts of data have to be used for training and good results may be limited to scenes with similar conditions to the training set.

III. ARCHITECTURE

The framework here proposed belongs to the block *Motion prediction* from Figure 1 and can be mainly divided into 4 blocks: *Find/Reuse corridors*, *Find interactions*, *Compute intentions* and *Motion prediction*, as shown in Figure 2 where the flowchart and the data entering and leaving each block is presented. The focus here is on the estimation of the intentions and the corridors' probabilities (three first blocks), leaving the motion prediction out of the scope of this work. The entrance data can be obtained from exteroceptive sensors, V2X communication or from publicly available datasets. The output of the block goes to the maneuver planner of the ego vehicle.

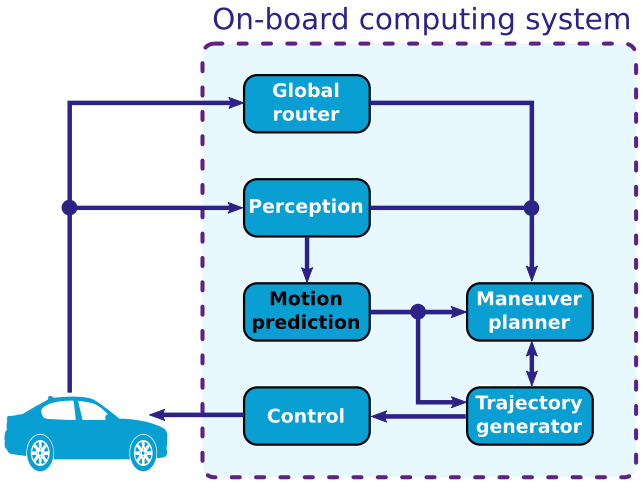


FIGURE 1. Architecture overview.

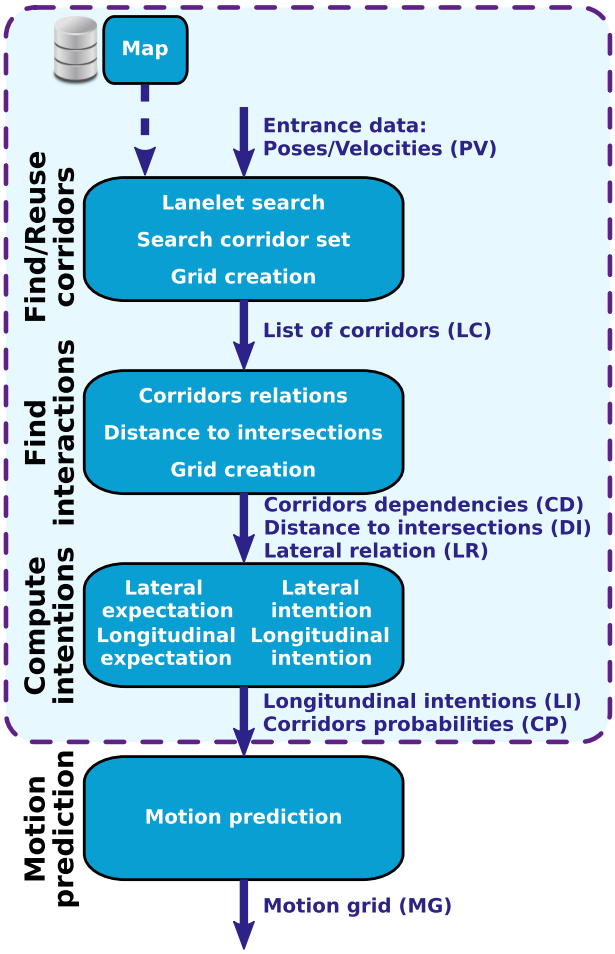


FIGURE 2. Intention inference and motion prediction flowchart.

Below, each block will be detailed, starting with a description of the maps.

A. MAP

The maps are loaded at the beginning of the simulation. They are formed by lanelets [27], that are interconnected drivable road segments geometrically represented by a right and left bound. The relation between each pair of lanelets is used to create an adjacency graph.

Linked to the lanelets are the regulatory elements, that, in the case of unsignalized intersections, can describe the lanelets with the right of way, the ones that have to yield, and the physical position of the stop line. These features are necessary to locate the intersections in the map and to compute the longitudinal expectation, as described in Section IV-6.

The intersections are obtained by some geometrical operations over the lanelets of the map and are composed of a set of lanelets, entrance and exit points, connections corridors linking entrances to exits, and priorities. The priorities are defined based on a pairwise comparison of the connections corridors and result on $n \times n$ table, where n is the number of connection corridors. The pairwise comparison follows these set of rules:

- if the corridors geometrically intersect and the corridor i has the right of way and the corridor j has to yield, the position (i, j) receives 2 and the position (j, i) receives 1.
- if the corridors geometrically intersect, but none of them carries information regarding their relation, both positions, (i, j) and (j, i) receive 1.
- if the corridors do not intersect both positions, (i, j) and (j, i) are set to 0.

B. FIND/REUSE CORRIDORS

Given a map formed by lanelets, their relational and physical layers are used to obtain all the navigable corridors for the vehicles in the scene. The length of these corridors has, at minimum, the distance that the car can reach in a time interval with its current speed, assuming constant maximum acceleration.

For all vehicles, first, the current lanelet(s) where the vehicle is located is obtained by comparing its position and orientation in the physical layer. Next, a graph search is performed for surrounding lanelets starting from the vehicle lanelet(s) to create a lanelet-sequence for each corridor.

In the next iterations, the corridors found can be either expanded or removed, if necessary. The expansion occurs if the remaining distance is lower than what it can reach in the future horizon, set as 4 seconds. In this case, a search is made to add at least 10% of its current distance to the corridor end. In the expansion is also possible to add new corridors, as it happens when the vehicle is reaching an intersection. The removal occurs if the current measured orientation of the vehicle, when compared with the centerline of the corridor, has a difference bigger than a threshold (see Figure 3) or if none of the current lanelet(s) the vehicle is located belongs to the corridor.

In the cases where there is more than one lane, at each iteration, the lanelet in which the center of the vehicle is located is selected. Based on this information, each corridor is defined as being left, center, right, or not reachable with respect to the position of the vehicle. To reach the corridors at right/left, a Bézier curve is created that concatenates the two road segments (the one in the current lane with the one in the adjacent lane) with a length of $max(4\ v,\ 10)m$, being v the current vehicle's velocity and 4 is the considered duration of a lane change (in seconds). These values were defined after analyzing the patterns of a lane change.

Using the optimized center points offline computed with the method described below, the path of each corridor is found by matching its lanelet's identifiers and the lanelet's identifiers of the optimization.

A grid is created based on the shape of the road for each corridor of the surrounding vehicles. For the ego vehicle, a route is assumed and the grid is built using this information. This grid is used to project the motion prediction of the vehicle in the corridor, but the details of this process (appearing in the last block of Figure 2) is outside the scope of this paper.

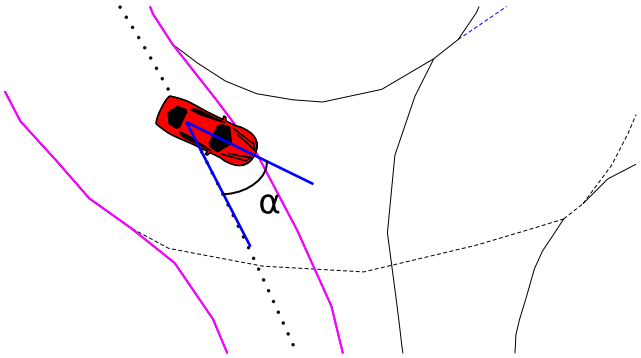


FIGURE 3. Example of corridor being removed due to its orientation difference.

At each time step, the lateral position of the vehicle with respect to the grid is updated and the centerline closest to the vehicle is select to be used in the following procedures.

An example of the corridors of a vehicle is shown in Figure 4, where for one of the corridors the grid is drawn. As mentioned before, this process is done for all vehicles, and the grid is created for all their corresponding corridors.

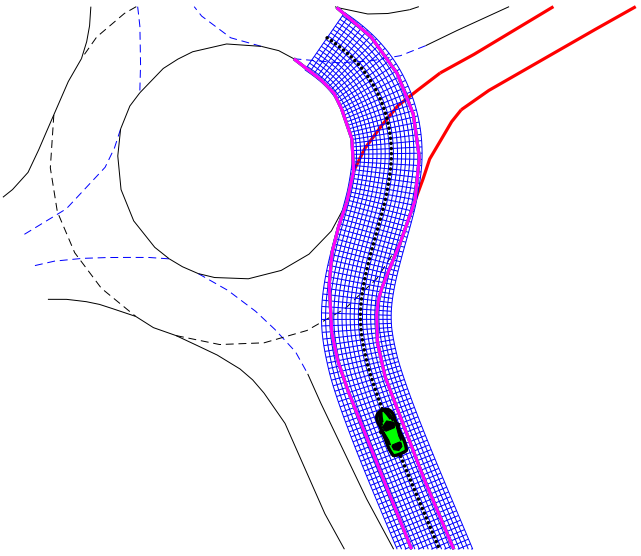


FIGURE 4. Example of corridors and grid.

Applying this search to all vehicles present at the scene results in a list of corridors (LC).

1) CORRIDORS OPTIMIZATION

The corridors generated in with the process described above, use the center points obtained from the optimization proposed in [28].

These center points are the result of the optimization of corridors linking all entrances to all exits in a process done offline only once and are loaded with the map. The cost function used here is:

$$J = w_1 l + w_2 \dot{k}^2 + w_3 \ddot{k}^2 \tag{1}$$

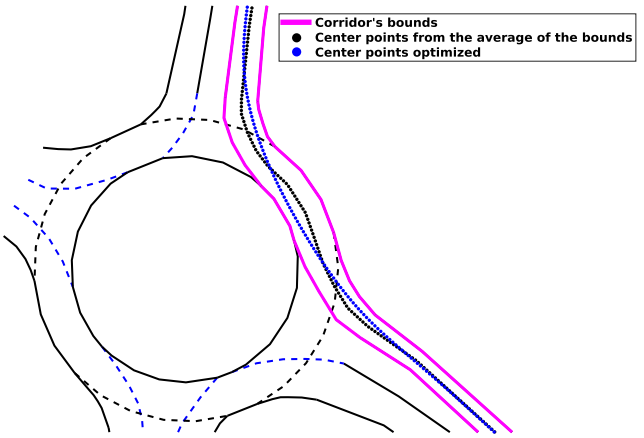


FIGURE 5. Example of the optimization. In black, the center points resulting from the average of the bounds. In blue, the center points optimized.

where l and k are the distance and curvature of the optimized path, respectively and w_1, w_2 and w_3 are weights defined as 1, 10000 and 5000, respectively.

Each centerline is divided to match the corridor's lanelets and the corresponding points are stored with the identifiers of all the lanelets.

C. FIND INTERACTIONS

In this stage of the process, the relations between vehicles and between vehicles and the elements of the map are found.

Three interactions are obtained from the current list of corridors: the corridors dependencies, the distances to the intersections, and the lateral relation.

1) CORRIDORS RELATIONS

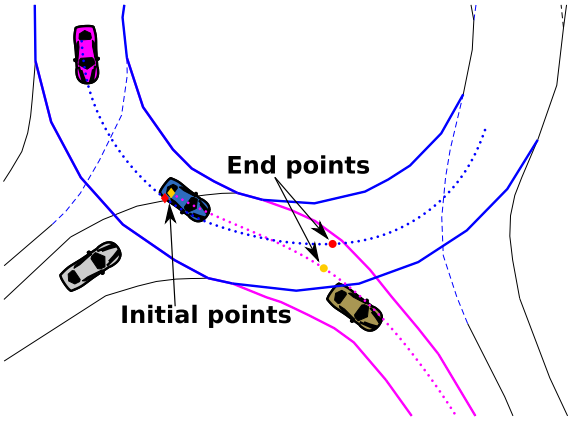
To find the dependencies between the vehicles present at the scene, the centerline of the corridors are pairwise intersected resulting in a collision list for each corridor. Figure 6 shows the collision points for a corridor (that remains inside the roundabout) from the magenta vehicle when compared to the two corridors from the blue vehicle, one leaving (Figure 6a) and one remaining inside the roundabout (Figure 6b).

In each pair, it is verified which vehicle arrives first at the initial collision points. The one that arrives first is causing the dependency on the one that arrives later. The possible collisions are then grouped and sorted considering the distance to the initial collision point and length (distance between the initial and end points). The closest (within a range) and largest one, in length, is defined as the dependency to the corridor. The reason why only one dependency is selected is due to the fact that in the motion prediction step only one corridor is considered to influence the predictions of the other at a given instant. From the example, the dependency from Figure 6b is selected, since it influences the corridor from the magenta vehicle for a longer distance.

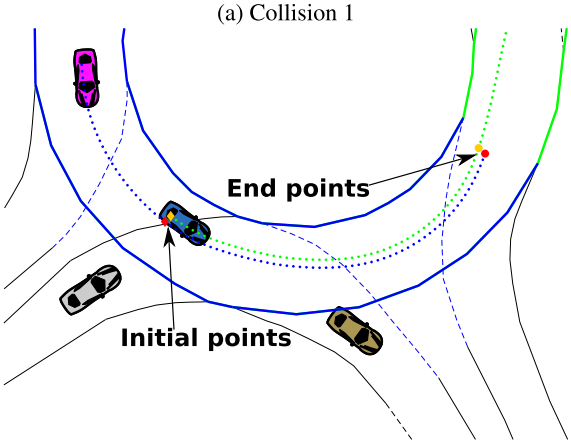
If the selected dependency falls in any of the following criteria, it is discarded and the next one is verified:

TABLE 1. Example dependencies: The color is the vehicle and the number is the exit.

Corridor with the dependency	Blue 2	Red 1	Red 2	Red 3	Green 1	Green 2	Yellow 2	Yellow 3	Yellow 4
Corridor causing the dependency	Magenta 2	Cyan 1	Blue 2	Blue 2	Red 1	Red 2	Blue 2	Blue 2	Blue 2



(a) Collision 1



(b) Collision 2

FIGURE 6. Example of collision points for one corridor of the magenta vehicle compared to the two corridors from the blue vehicle. In both cases, the magenta vehicle arrives later to the initial collision point, hence it is the one having a dependency.

- the corridor causing the dependency has another dependency before;
- the corridor causing the dependency has an intersection before and has to stop;

If a corridor has a dependency before reaching an intersection, this corridor will not be considered at this intersection. This process reduces the number of vehicles that have to be handled at the intersections.

In this approach, at least one corridor has to be free (has no dependency) to avoid a locked chain. In roundabouts, the corridor with the farthest dependency is considered free and leads the chain.

Table 1 exemplify the result of the process described above considering the frame shown in Figure 7. The corridors are described as (vehicle color and exit number (in blue)).

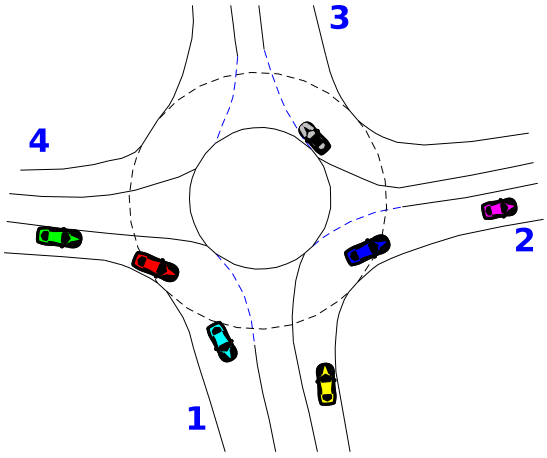


FIGURE 7. Example of the dependencies.

2) DISTANCE TO INTERSECTIONS

For each vehicle's corridors, the lanelet's identifiers of the corridor are intersected with those of the intersections to determine, if any, through which intersection the corridor passes.

At each intersection, several key variables are determined: the distance to the intersection, the entrance, and through which connection corridor the corridors go through. If the distance is bigger than what the vehicle can travel in the time horizon, the corridor will not be considered in the following processes to avoid unnecessary computation. Only the corridors that go through at the intersection are considered.

The distance to the intersection is found by the projection of the entrance point to the center points of the corridor and computing the difference between the projected position of the vehicle and the projected position of the entrance, as shown in Figure 8.

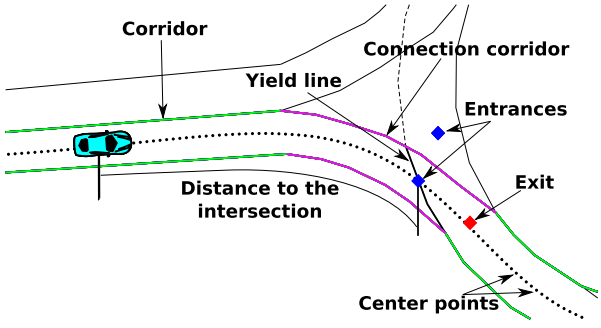


FIGURE 8. Parameter description in intersections.

The result from the corridors relations is the corridors dependencies (CD).

The result of this process is the distances to intersections (DI).

3) LATERAL INTERACTION

In this step, the surrounding vehicles to every target vehicle are found and the identifiers, distances, and velocities are set to a table, as described in [17].

The result from the lateral interaction is the lateral relation (LR).

IV. INTENTION ESTIMATION

In order to compute the intention of the traffic participants, the Dynamic Bayesian Network (DBN) proposed in [15] and used in [17] is applied. For each of the vehicles present in the scene, with the exception of the ego vehicle, the network represented in Figure 9 is instantiated, where bold arrows represent the influences of the other vehicles on vehicle n through some key variables $(E_t^n, I_t^n, \Phi_t^n, Z_t^n)$ described below.

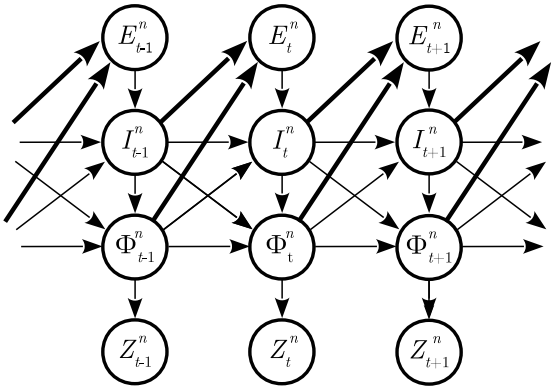


FIGURE 9. Bayesian network.

The relations among variables appearing in Figure 9 allows to model the driving scene as the following generalized distribution [15]:

$$\begin{aligned} &P(\boldsymbol{E_{0:T}}, \boldsymbol{I_{0:T}}, \boldsymbol{\Phi_{0:T}}, \boldsymbol{Z_{0:T}}) \\ &\quad = P(\boldsymbol{E_0}, \boldsymbol{I_0}, \boldsymbol{\Phi_0}, \boldsymbol{Z_0}) \\ &\qquad \times \prod_{t=1}^{T} \times \prod_{n=1}^{N} [P(E_t^n | \boldsymbol{I_{t-1}} \boldsymbol{\Phi_{t-1}}) \times P(I_t^n | \Phi_{t-1}^n I_{t-1}^n E_t^n) \\ &\qquad \times P(\Phi_t^n | \Phi_{t-1}^n I_{t-1}^n I_t^n) \times P(Z_t^n | \Phi_t^n)] \end{aligned} \tag{2}$$

where the variables are described below:

- Expected maneuver E_t^n: represents the expected behavior of the vehicle n at instant t according to traffic rules. It is divided into two parts: longitudinal and lateral. The longitudinal expectation is used to model the probability that the vehicle should stop at an intersection and can assume two values: *go* and *stop*. The lateral expectation, on the other hand, models the probability that the vehicle can make a lane change without hindering traffic. It can assume two values: *stay* and *change*.
- Intended maneuver I_t^n: represents the intention of the vehicle and is also divided into longitudinal and lateral. The lateral intention also includes the corridor the vehicle intends to follow.
- Physical vehicle state Φ_t^n: represents the pose and speed of the vehicle. They are calculated at each instant based on the intentions.
- Measurements Z_t^n: represents the real measurements of the physical state of the vehicle, extracted directly from exteroceptive sensors of the ego-vehicle or via V2X communications [29].

Algorithm 1 : Particle Filter Overview

Input: CD, DI, LR
Output: LI, CP
initialize particles
while *true* **do**
  compute lateral expectation
  compute lateral intention
  compute longitudinal expectation
  compute longitudinal intention
  update pose and velocity
  } Prediction
  compute weight
  } Update
  resample
  } Resampling
end

An exact inference of (2) is not tractable, hence a particle filter is used to estimate the hidden states $\mathbf{E_t}$, $\mathbf{I_t}$ and $\Phi_\mathbf{t}$, given the observed variables $\mathbf{Z_t}$.

The particle filter is a type of Bayesian filter and the predict/update cycle can be used to estimate the state [30]. The version here implemented can be divided into three main steps: prediction, update, and resampling. In the prediction step, the prior is computed by propagating the particles based on a system model. In the update step, a measurement Z_t is used to refine the expected state estimate. In the resampling step, new particles are randomly selected, with replacement, from the set of weighted particles [31]. Each particle represents a possible state of the situation and contains a specific corridor to each vehicle.

An overview of the particle filter implemented is presented in Algorithm 1 followed by a description of each step. The input to the algorithm is the interactions previously defined, and the outputs are the longitudinal intentions (LI) of each corridor at the next intersection (stop or go) and the probabilities (CP) of the vehicle being in each of its possible corridors.

4) LATERAL EXPECTATION

The decision to change lanes should be based on the desire to quit the current lane, the selection of the target lane, and the feasibility of the change.

For every vehicle in every particle, the vehicles' followers and leaders in all possible lanes are determined and the distances bumper-to-bumper and the velocity differences are found. This information is used to compute the expected lateral motion of the vehicles present at the scene with the with

the model of [32], following the outcomes of the comparison conducted in [17].

5) LATERAL INTENTION

The lateral intention is computed based on the previous intention (Ic_{t-1}) and the current expectation (Ec_t). The intention is 1 (change) if a random value is bigger than the probability generated by Table 2.

TABLE 2. Lateral intention.

Ic_{t-1}	Ec_t	Probability
1	1	0.9
1	0	0.5
0	1	0.5
0	0	0.1

If the intention is to change lanes, a corridor containing the target lanelet is selected. In those cases where there is only one possible lane, the particle can also change corridors following the probabilities from Table 2, given that the expectation is 0 (stay).

6) LONGITUDINAL EXPECTATION

At each intersection, a pairwise evaluation of the corridors is performed to determine the expected maneuver of the vehicles resulting in a conflict matrix. The pairwise evaluation was defined to be able to handle most types of unsignalized intersections, not only roundabouts.

For each pair of vehicles the evaluation takes the following rules into consideration:

- if the priorities are 0, both vehicles are expected to *go*;
- if the priority of the vehicle i is bigger than the priority of vehicle j, vehicle i is expected to *go* and vehicle j will have its expectation computed (adapted from [33]) as:

$$E = \frac{1.05}{1 + (\frac{\eta}{3})^{-4}} < rand \tag{3}$$

  where the left part represents the probability to accept the gap and η is the absolute value of the difference between the arriving time at the intersection.
- if the priority is one for both vehicles, the arrival time is compared:
  - if any vehicle arrives more than one second before the other, this vehicle is expected to *go* and the expectation of the other is computed with (3).
  - if both vehicles arrive within one second and one entrance is on the right of the other, the vehicle coming from the right is expected to *go* and the other to *stop*.
  - if none of the above applies, the expectation for both vehicles is computed with (3).

This pairwise evaluation is done to all vehicles going through the intersection and will result in a conflict matrix containing all the expectations. To define which vehicles can *go* and which vehicles should *stop*, the columns of the conflict matrix are summed and the vehicles corresponding to the lowest values are expected to *go* and the others to *stop*.

As an example, the evolution of the expectation for a vehicle waiting to enter the roundabout is presented in Figure 10. In this scenario, mainly three vehicles are involved: blue, yellow, and magenta. The three main events of the situation are shown in Figure 11 and are marked with discontinuous black lines in Figure 10. Table 3 is the conflict matrix for a single particle and contains the probability to accept the gap and the expectations for the frame from Figure 11a. In this particle, the magenta vehicle intends to remain inside the roundabout. If the corridor assigned to this vehicle was leaving the roundabout, the vehicle would not be considered in the computation of the expectation at this intersection.

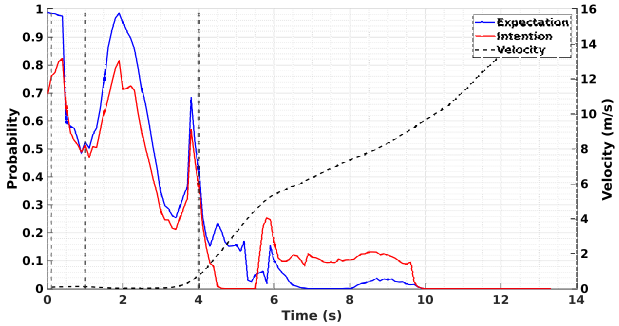


FIGURE 10. Longitudinal expectation, intention and velocity of the blue vehicle.

TABLE 3. Conflict matrix: probability to accept the gap and the longitudinal expectation in parenthesis.

vehicle \ vehicle	yellow	magenta	blue
yellow	X	1 (0)	0.023 (1)
magenta	1 (0)	X	0.294 (0)
blue	1 (0)	1 (0)	X

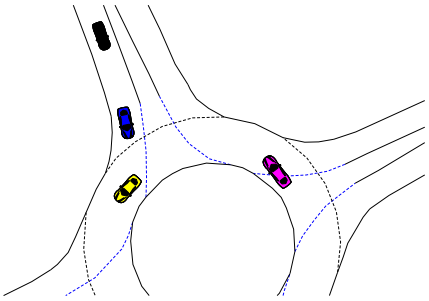
The way to read Table 3 is: the magenta vehicle has the priority when compared with the blue vehicle. Therefore, the cell (blue, magenta) has 1 as the probability to accept the gap and 0 (*go*) as the expected maneuver. For the cell (magenta, blue), the probability to accept the gap, computed with (3), is 0.294, resulting in an expected maneuver to *go* when compared with a random number. The sum of the columns results in [0 0 1], which is interpreted as an expected maneuver to *go* for the yellow and magenta vehicles and to *stop* for the blue vehicle.

7) LONGITUDINAL INTENTION

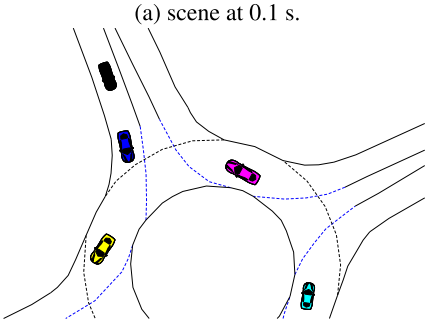
As for the lateral intention, the longitudinal intention is computed based on the previous intentions (I_{t-1}) and the current expectations(E_t). The intention is 1 (stop) if a random value is bigger than the probability generated by Table 4 [34].

The longitudinal intention is applied to compute the motion prediction for vehicles at intersections.

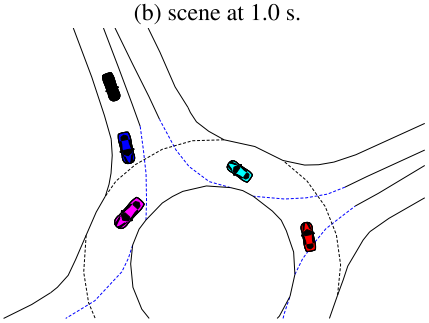
For the example presented in Figure 11a, the intention of the blue vehicle is shown in Figure 10 with a red line.



(a) scene at 0.1 s.



(b) scene at 1.0 s.



(c) scene at 4.0 s.

FIGURE 11. Evolution of a situation.

TABLE 4. Longitudinal intention.

Is_{t-1}	Es_t	Probability
0	0	0.9
0	1	0.5
1	0	0.5
1	1	0.1

8) PHYSICAL VEHICLE STATE UPDATE

After the computation of the intentions, it is necessary to update the pose and velocity for each vehicle in each particle.

a: POSE

A visualization of the pose update can be seen in Figure 12. For all vehicles from all particles, the new position is computed in two steps. Firstly, the current position is projected into the centerline of the corridor and the distance traveled in the last time interval is added as follows (the indices for the particle and the vehicle are omitted):

$$P^t_{proj} = P^{t-1}_{proj} + \lfloor \frac{v^{t-1}\Delta t}{sl} \rfloor$$
$$P^t_1 = cp(:, P^t_{proj}) + (v^{t-1}\Delta t \bmod sl) * \frac{pv}{\|pv\|} \quad (4)$$

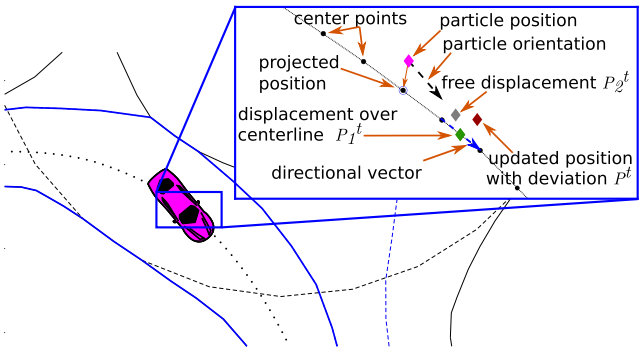


FIGURE 12. Pose update for a vehicle from a particle.

where cp is a $2 \times n$ matrix of the center points, P^t_{proj} and v^t are the projected position of the particle into the center line and the velocity of the particle for the vehicle at the instant t, respectively, sl is the segment length, and pv is the directional vector between the two center points near P^t_1.

Secondly, it is considered that the particle can move freely from its previous position P to a new one P_2 using this expression:

$$P^t_2 = P^{t-1} + v^{t-1}\Delta t \begin{bmatrix} cos(\theta^{t-1}) \\ sin(\theta^{t-1}) \end{bmatrix} \quad (5)$$

The final position update P^t is the average of (4) and (5), including an additive random deviation:

$$P^t = \frac{P^t_1 + P^t_2}{2} + rand \quad (6)$$

The orientation is updated by adding a deviation to the direction of the vector in the centerline.

b: VELOCITY

The update of the velocity for a given vehicle in a particle is performed as follows: if the intention for the particle is to *go*, the update is done by sampling one acceleration from the acceleration distribution used in the motion prediction; if the intention is to *stop*, the update is done using a speed profile.

The acceleration distribution is a probability distribution that represents the acceleration the vehicle must probably have in the current instant. It is computed with Algorithm 2, that takes into account the current acceleration a from the vehicle and a defined set of accelerations intervals ranging from $-3\ m/s^2$ to $2\ m/s^2$. At each instant, it is verified in which interval a falls, and the result is averaged with the result from the previous time to avoid sudden changes. The velocity is updated as follows (the indices for the particle and the vehicle are omitted):

$$v^t = v^{t-1} + a_s\Delta t \quad (7)$$

where a_s is a sample drawn from the initial acceleration distribution.

In the second case, two speed profiles are defined given two lateral accelerations, one maximal and one average

Algorithm 2 Acceleration Distribution for a Vehicle

```
/* current acceleration, previous
   distribution                          */
Input  : a, accDist^{k-1}
/* current acceleration distribution
   */
Output accDist^k
:
/* acceleration's intervals              */
accInt = [-3, -2, -1, 0, 1, 2]
Δa = 0.3        // acceleration threshold
for i ← 1 to 5 do
    accDist^k(i) ← (accInt(i) − Δa) ≤ a <
    (accInt(i + 1) + Δa)
end
accDist^k = normalize(accDist^k)
accDist^k = (accDist^k + accDist^{k-1})/2)
```

(set by design), and the curvature κ of the path with Equation 8.

$$v_d = \sqrt{\frac{a_{lat_d}}{|\kappa|}}, \quad d = avg, max \tag{8}$$

Both velocity vectors, v_{max} and v_{avg}, are limited by the maximum allowed velocity in the corridor for both curves. The maximum velocity allowed can be retrieved from the map in the form of a regulatory element, or, in the absence of such element, be defined as the maximum velocity used in the motion prediction.

An example of the two curves is shown in Figure 13b for the corridor from Figure 13a. Note that the velocity in the intersection point is set to 0 m/s and the profiles are updated to fulfill the acceleration and deceleration bounds set by design.

With the points obtained from these profiles, the velocity is updated as follows [33]:

$$v_t = vatB - \frac{vatB - vmtB}{vatA - vmtA}(vatA - v_{t-1}) + rand \tag{9}$$

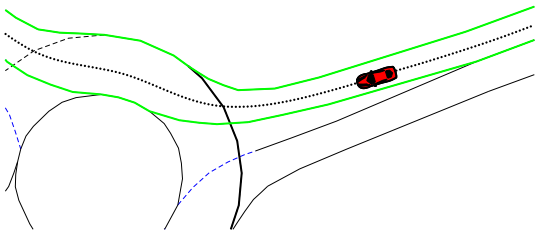
where *vmtA* and *vatA* are the maximum and average velocities in the previous projected position (P^{t-1}_{proj}), and *vmtB* and *vatB* are the maximum and average velocities in the current projected position (P^{t}_{proj}).

9) WEIGHT UPDATE

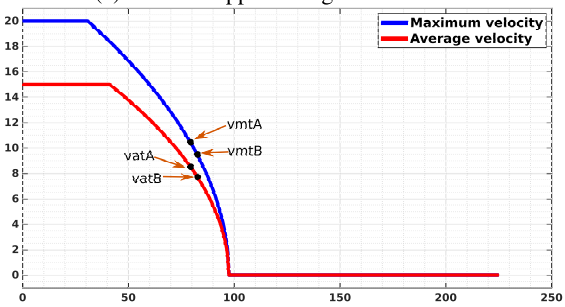
The weight of each particle is updated considering, for each vehicle, a four-variate normal distribution centered on the true state Z^n_t with no correlation between x, y, Θ and v.

$$P(Z^n_t|\Phi^n_t) = \mathcal{N}(\hat{x}^n_t, x^n_t, \sigma_x) \times \mathcal{N}(\hat{y}^n_t, y^n_t, \sigma_y) \times \mathcal{N}(\hat{\Theta}^n_t, \Theta^n_t, \sigma_\Theta) \times \mathcal{N}(\hat{v}^n_t, v^n_t, \sigma_v) \tag{10}$$

where $\mathcal{N}(a, b, c)$ evaluates the estimated value a, considering a mean b and a deviation c.



(a) Corridor approaching an intersection



(b) Speed profiles for the corridor when the intention is to *stop* at the intersection

FIGURE 13. Example of speed profile for an intersection.

The total weight w of a particle k is the product of (10) over all N vehicles present:

$$w^t_k = \prod_{n=1}^{N} P(Z^n_t|\Phi^n_t) \tag{11}$$

10) RESAMPLING

In the resampling step, the particles are selected according to their weights. The selection is performed using the Low Variance Resampling method [35] to prevent the loss of diversity.

The criterion for implementing resampling is the effective sample size (ESS) N_{eff} which reflects the degree of weight degeneracy [36].

$$N_{eff} = \frac{1}{\sum_{k=1}^{K} (w^t_k)^2} \tag{12}$$

Resampling occurs when the ESS falls below a selected threshold, set as half of the number of particles.

V. DATASET PROCESSING AND EVALUATION RESULTS

To evaluate the framework proposed in previous Sections, publicly available datasets were used. This data needs to be prepared beforehand, with the process described below. To evaluate the results, one situation is selected and explained in Section V-C1. The results from the 9 selected situations are presented in Section V-C2.

A. DATASET PROCESSING

The data used in the simulations are obtained from publicly available datasets recorded from a bird-eye-view perspective.

TABLE 5. Roundabouts.

Roundabout	Origin dataset	Radius	Number of entrances/exits	Duration	Number of vehicles	Number of bifurcations
A	INTERACTION	11.7 m	3	28.4 s	15	13
B	RounD	11.4 m	4	24.7 s	15	21
C	OpenDD	17.0 m	4	21.4 s	19	24
D	OpenDD	17.4 m	4	40.9 s	37	42
E	OpenDD	14.8 m	4	25.2 s	11	17
F	OpenDD	19.5 m	4	33.1 s	20	29
G	OpenDD	14.4 m	4	42.6 s	26	52
H	OpenDD	16.8 m	4	32.9 s	31	44
I	OpenDD	13.9 m	3	33.9 s	33	29

These data contain public traffic data from the traffic participants present at the scenes. For each participant, they include their pose, velocity, acceleration, size, and also the frames where it appears. Within the considered public available datasets (INTERACTION [4], rounD [5], and openDD [6]), 9 different roundabouts were identified, from which the situation with a higher interactivity, was selected.

In order to use these data for the purpose of this work, they have been downsampled and filtered to remove undesired participants, such as static vehicles, pedestrians, motorcycles, and large trucks. After binary tagging each frame as containing or not these participants, the intervals are grouped and the vehicle that stays the longest in each interval is defined as the ego vehicle. The frames in which this vehicle is not present are discarded.

Table 5 contains relevant information regarding each case, where the number of bifurcations represents the number of times a vehicle can either leave or remain inside the roundabout.

B. LONGITUDINAL EXPECTATION/INTENTION AT THE INTERSECTION

The longitudinal intention (LI) is usually evaluated by a risk metric computed as the difference between what is expected and what is the inferred intention of the vehicle. Since the data is real and obtained from public datasets, situations of real risk are rare and, in the selected cases, none.

In this framework, the LI is applied and evaluated in the motion prediction step: applied as the probability of the vehicle to stop at the next intersection and evaluated by the accuracy of the predictions in future time steps. Since motion prediction is out of the scope of this paper, the evaluation is performed by assessing how well the intention reflects the expectation.

Similar to the risk, the evaluation is done by comparing the expected value to the inferred intention at each instant for every vehicle arriving at the intersection. If the absolute difference between the expected and inferred values is bigger than a threshold, this instant is flagged as an error. The threshold here used is set to 0.3, the same value used to detect risk in [33].

The metrics used to evaluate the longitudinal intention are the *overestimation* and *underestimation*, defined as:

- *overestimation*: the difference is bigger than the threshold. The expectation is more conservative than the inferred intention;
- *underestimation*: the difference is smaller than the negative threshold. The intention is more conservative than the computed expectation.

To illustrate the meaning of these categories, an example with a vehicle from roundabout *C* is selected. Its expectation, intention and difference are shown in Figure 14 and the trajectory followed by this vehicle is shown in Figure 15. The instants where the vehicle is behind the yield line are marked in magenta in the map in Figure 15 and are represented by continuous lines in Figure 14 where the bars classify the instants as *good* (blue), *overestimation* (green) and *underestimation* (red). In this example, from the 57 instants the vehicle is present at the scene, 4 are classified as overestimation (7.02%), and 1 as underestimation (1.75%).

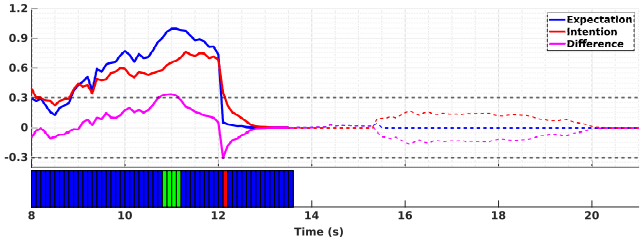


FIGURE 14. Expectation, intention and difference. The instants are classified as *good* (blue), *overestimation* (green) or *underestimation* (red).

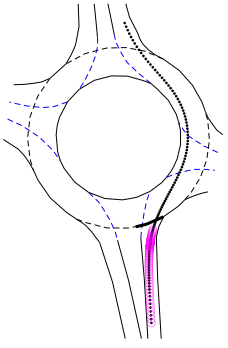


FIGURE 15. Trajectory followed by the vehicle from the example from Figure 14. The instants the vehicle is behind the yield line are highlighted in magenta.

The results for the 9 cases are presented in Table 6. *Instants* shows the overall number of situations where the vehicles are arriving at the roundabout, and *Overestimation* and *Underestimation* represent the percentage of mismatch between

TABLE 6. Percentage of *overestimation* and *underestimation* for the 9 cases from Table 5.

Situation	Instants	Overestimation	Underestimation
A	378	17 (4.50%)	0 (0%)
B	742	6 (0.81%)	1 (0.13%)
C	542	28 (5.17%)	2 (0.37%)
D	1363	40 (2.93%)	11 (0.80%)
E	347	0 (0%)	0 (0%)
F	778	3 (0.39%)	0 (0%)
G	861	19 (2.20%)	1 (0.12%)
H	1344	67 (4.98%)	14 (1.04%)
I	1148	26 (2.26%)	9 (0.78%)
Total	7503	206 (2.75%)	38 (0.51%)

expectation and intention. From the 7503 instants a vehicle is facing the entrance to a roundabout, 206 (2.75%) are classified as overestimation and only 38 (0.51%) as underestimation, which can also be interpreted as 96.74% of similarity between expectation and intention.

C. INTENTION TO LEAVE OR REMAIN INSIDE THE ROUNDABOUT

The intention of the vehicles should be classified as early as possible. For this reason, the lead time was selected to evaluate the estimations. For each vehicle, the intention of leaving or remaining inside the roundabout is estimated by the sum of the probability of the corridors leaving and remaining inside the roundabout. With a threshold of 0.5, and allowing a maximum of three spikes up to 0.7, the lead time is defined as the interval between the time the sum of the correct corridors has a bigger probability than the sum of the wrong corridors until the time these wrong corridors are eliminated from the list of possible corridors (as explained in Section III-B). Here, correct and wrong corridors can be either the corridors leaving or remaining in the roundabout depending on the trajectory of the vehicle. Lead times larger than 4 s are reduced to 4 s.

1) QUALITATIVE RESULT

Due to the small number of vehicles, a selected roundabout (*E*) is used to describe and illustrate the metrics used for the intention estimation evaluation. Figure 16 shows the trajectories followed by each vehicle in this driving scenario, where each color represents a different car.

To take into account the stochastic nature of the particle filter, each situation is simulated three times and the final result is the average of the three.

Figure 17 shows the evolution of the probabilities for the given situation. Each line of the figure represents a vehicle identified by the red number on the left. The black dashed vertical lines are the instants where a corridor leaving the roundabout is removed and the vehicle remains inside the roundabout, whereas the black continuous vertical lines are the ones where the vehicle leaves the roundabout and the corridors remaining inside are removed. The cyan dashed line is the threshold used to define the lead time and the blue line is the estimation of the wrong intention, where 0 represents

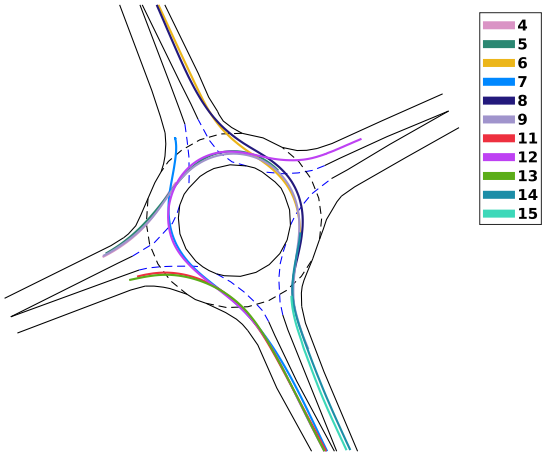


FIGURE 16. Trajectories follow by the vehicles throughout the whole simulation.

a totally accurate intention estimation and 1, the opposite. This value is computed by the sum of the probabilities of the corridors leaving and remaining inside the roundabout. After the last vertical line, the probability is one since the vehicle has left the roundabout. For each bifurcation the lead time is presented in Table 7.

TABLE 7. Lead time (larger is better) for the evolution from Figure 17.

Case	Lead time	Case	Lead time
A	1.3 s	J	2.2 s
B	2.1 s	K	4.0 s
C	0.8 s	L	1.0 s
D	1.7 s	M	1.3 s
E	0.8 s	N	2.8 s
F	2.7 s	O	2.7 s
G	0.7 s	P	4.0 s
H	2.0 s	Q	0.8 s
I	1.6 s		

Figure 18 details the evolution for the vehicle 8. Each vertical line is complemented with the position of the vehicle in the roundabout:

- continuous green (Figure 18b): instant the first lead time starts to count. As already mentioned, a maximum of three spikes up to 0.7 are allowed.
- discontinuous black (Figure 18c): instant the corridor leaving the roundabout on the east exit is removed. At this point, the first lead time is completed.
- continuous magenta (Figure 18d): instant the second lead time starts to count.
- continuous black (Figure 18e): instant the corridors remaining in the roundabout on the east exit are removed. At this point, the second lead time is completed.

2) COMPARISON

The framework proposed (Method *A*) was applied in the 9 roundabouts defined in Table 5 and compared with other intention estimation strategies:

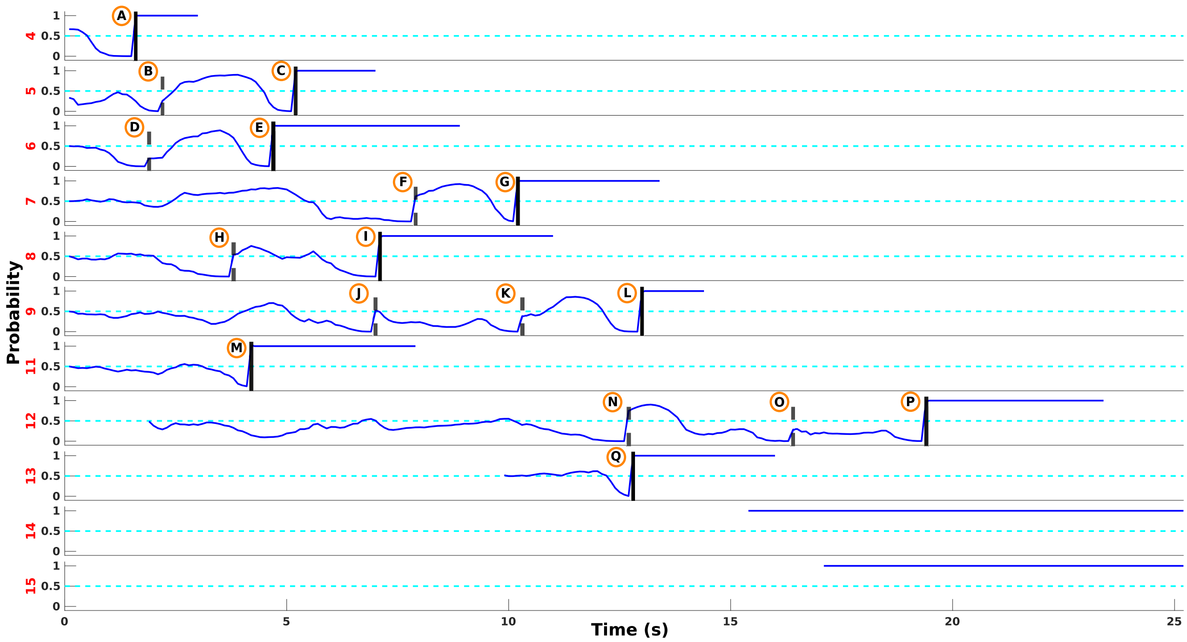


FIGURE 17. Evolution of the probabilities of leaving or remaining inside the roundabout for each vehicle.

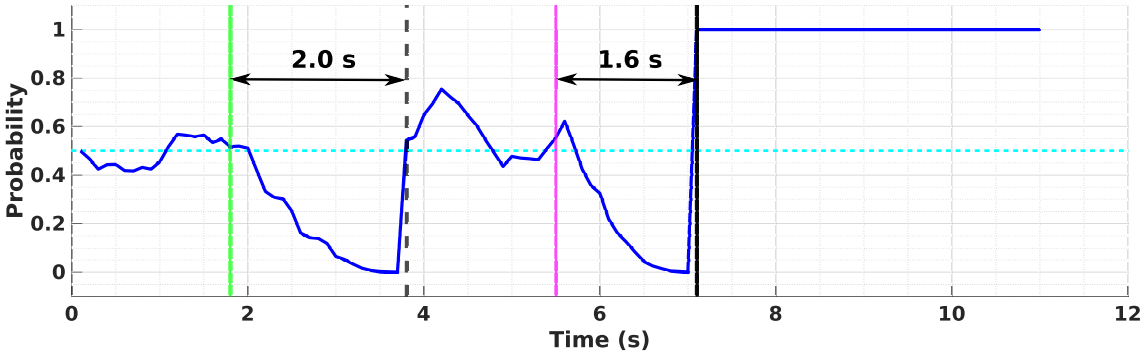
TABLE 8. Comparison of the proposed method (A) with two baselines (B and C) and two implementations of works found in the literature (D and E). For each method, columns 1-3 contain the number of cases the lead time interval occurs in the situation, and columns 4-5 contain the minimum and mean lead time (in seconds), respectively. For the minimum and mean lead time, the larger the value, the better. The values in bold are the best results in each situation.

Roundabout	Method *A*					Method *B*					Method *C*					Method *D*					Method *E*				
	$ld \leq 0.1\ s$	$0.1\ s < ld \leq 1.0\ s$	$1.0\ s < ld \leq 2.0\ s$	*minimum ld*	*mean ld*	$ld \leq 0.1\ s$	$0.1\ s < ld \leq 1.0\ s$	$1.0\ s < ld \leq 2.0\ s$	*minimum ld*	*mean ld*	$ld \leq 0.1\ s$	$0.1\ s < ld \leq 1.0\ s$	$1.0\ s < ld \leq 2.0\ s$	*minimum ld*	*mean ld*	$ld \leq 0.1\ s$	$0.1\ s < ld \leq 1.0\ s$	$1.0\ s < ld \leq 2.0\ s$	*minimum ld*	*mean ld*	$ld \leq 0.1\ s$	$0.1\ s < ld \leq 1.0\ s$	$1.0\ s < ld \leq 2.0\ s$	*minimum ld*	*mean ld*
A	0	2	8	**0.8 s**	1.70 s	0	3	3	0.3 s	**2.05 s**	0	6	3	0.5 s	1.62 s	2	3	0	0.3 s	2.04 s	0	8	4	0.3 s	0.88 s
B	0	5	4	**0.7 s**	**2.43 s**	0	7	5	0.3 s	2.22 s	1	5	3	0.3 s	2.34 s	2	5	4	0.3 s	2.18 s	0	17	2	0.3s	0.9 s
C	0	12	6	**0.3 s**	1.55 s	0	13	8	0.2 s	1.15 s	2	8	7	0.2 s	**1.61 s**	5	11	3	0.2 s	1.29 s	1	18	4	**0.3 s**	0.73 s
D	0	18	10	**0.4 s**	**1.78 s**	0	23	9	0.3 s	1.44 s	0	18	12	**0.4 s**	1.53 s	8	13	10	0.2 s	1.62 s	0	34	9	0.3 s	0.69 s
E	0	5	5	**0.7 s**	**1.91 s**	0	4	7	0.3 s	1.71 s	0	7	3	0.2 s	1.89 s	3	4	6	0.2 s	1.56 s	0	12	3	0.3 s	0.86 s
F	0	6	15	**0.5 s**	**1.88 s**	0	10	11	0.4 s	1.60 s	0	9	11	0.3 s	1.71 s	7	9	2	0.2 s	1.83 s	0	29	0	0.3 s	0.45 s
G	0	14	10	0.3 s	**2.43 s**	0	21	18	0.3 s	1.60 s	0	17	12	**0.4 s**	2.17 s	8	19	10	0.2 s	1.56 s	0	38	4	0.3 s	1.03 s
H	0	12	16	0.4 s	**1.96 s**	0	14	14	**0.5 s**	1.88 s	0	15	17	0.3 s	1.68 s	3	23	11	0.2 s	1.32 s	0	37	2	0.3 s	0.95 s
I	0	7	11	**0.9 s**	**2.07 s**	0	9	5	0.6 s	2.04 s	2	16	4	0.2 s	1.32 s	0	11	6	0.2 s	1.93 s	0	23	6	0.3 s	0.61 s

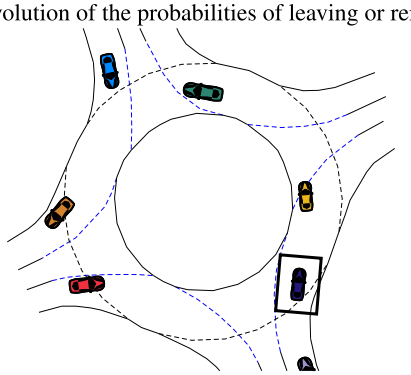
- Centerlines optimized (Method *B*): the centerlines are obtained directly from the optimization from the Section III-B1;
- Centerlines directly extracted as the geometrical center of the lanelets (Method *C*);
- The implementation of two representative works found in the literature that also have a way to classify the intention in roundabouts:
  - Dynamic Time Warping + Naive Bayes Classifier [20] (Method *D*);
  - Frenet Frame + Naive Bayes Classifier [19] (Method *E*).

Table 8 presents the results for the 5 methods in each of the considered roundabouts. As can be seen, the framework here proposed outperforms the other methods in 7 of the 9 cases, both in the minimum and in the mean lead time.

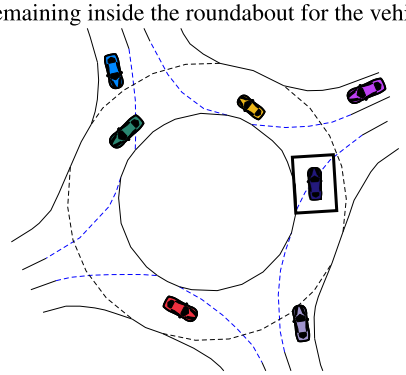
In order to better understand the results the lead time ld was classified into four intervals: $ld \leq 0.1\ s$, $0.1\ s < ld \leq 1.0\ s$, $1.0\ s < ld \leq 2.0\ s$ and $ld > 2.0\ s$. $ld \leq 0.1\ s$ represents the cases where the method do not give a higher priority to the correct corridors until these corridors are removed from the list of possible corridors. The last interval $ld > 2.0\ s$ is not present in the table since it is the total number of bifurcations (Table 5) minus the quantity of the other intervals. Note that



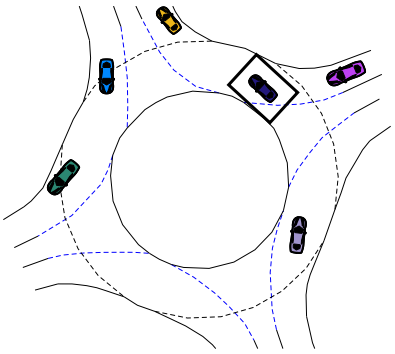
(a) Evolution of the probabilities of leaving or remaining inside the roundabout for the vehicle 8.



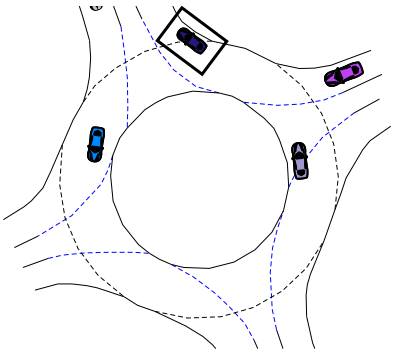
(b) Snapshot at the green line.



(c) Snapshot at the discontinuous black line.



(d) Snapshot at the magenta line.



(e) Snapshot at the continuous black line.

FIGURE 18. Evolution of the probabilities of leaving or remaining inside the roundabout for the vehicle 8 and snapshots of the scene in 4 instants. The target vehicle is enclosed by a black box.

methods *A* and *B* have no case where the correct intention is not estimated ($ld \leq 0.1\ s$), whereas the other methods have at least one case.

VI. CONCLUSION

In this work, the framework currently being used by the AUTOPIA Group for the interaction-aware intention classification of vehicles at roundabouts is presented. The foundation of the framework, based on geometrical maps containing structural data of the roads (left and right bounds) and intersections (right of way, yield, reference line) and general navigable corridors obtained from the relational layer from the map, can be applied to any context. The interaction-aware estimation is performed with a Dynamic Bayesian Network where the inference of the intentions is done with a particle filter.

The longitudinal expectation/intention at intersections was evaluated by assessing how well the intention reflects the expectation. With the threshold used, 2.75 % of the instants were classified as overestimation and only 0.51 % were classified as underestimation.

A comparison of the intention classification with four baseline techniques was presented. The results found after evaluation of 9 driving situations in roundabouts with data obtained from publicly available datasets showed that the approach here presented yields better performance both in the minimum lead time and in the mean lead time.

REFERENCES

[1] W. Zhan, A. de La Fortelle, Y.-T. Chen, C.-Y. Chan, and M. Tomizuka, "Probabilistic prediction from planning perspective: Problem formulation, representation simplification and evaluation metric," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 1150–1156, doi: 10.1109/IVS.2018.8500697.

[2] Washington State Department of Transportation. *Roundabout Benefits*. Accessed: Jun. 30, 2021. [Online]. Available: https://wsdot.wa.gov/Safety/roundabouts/benefits.htm

[3] A. Montella, "Identifying crash contributory factors at urban roundabouts and using association rules to explore their relationships to different crash types," *Accident Anal. Prevention*, vol. 43, no. 4, pp. 1451–1463, Jul. 2011, doi: 10.1016/j.aap.2011.02.023.

[4] W. Zhan, L. Sun, D. Wang, H. Shi, A. Clausse, M. Naumann, J. Kümmerle, H. Königshof, C. Stiller, A. de La Fortelle, and M. Tomizuka, "Interaction dataset: An international, adversarial and cooperative motion dataset in interactive driving scenarios with semantic maps," Sep. 2019, *arXiv:1910.03088*. [Online]. Available: https://arxiv.org/abs/1910.03088

[5] R. Krajewski, T. Moers, J. Bock, L. Vater, and L. Eckstein, "The round dataset: A drone dataset of road user trajectories at roundabouts in Germany," in *Proc. IEEE 23rd Int. Conf. Intell. Transp. Syst. (ITSC)*, Sep. 2020, pp. 1–6, doi: 10.1109/ITSC45102.2020.9294728.

[6] A. Breuer, J.-A. Termöhlen, S. Homoceanu, and T. Fingscheidt, "OpenDD: A large-scale roundabout drone dataset," in *Proc. IEEE 23rd Int. Conf. Intell. Transp. Syst. (ITSC)*, Sep. 2020, pp. 1–6, doi: 10.1109/ITSC45102.2020.9294301.

[7] H. Pilko, S. Mandžuka, and D. Barić, "Urban single-lane roundabouts: A new analytical approach using multi-criteria and simultaneous multi-objective optimization of geometry design, efficiency and safety," *Transp. Res. C, Emerg. Technol.*, vol. 80, pp. 257–271, Jul. 2017, doi: 10.1016/j.trc.2017.04.018.

[8] A. Silvano and A. Linder, "Traffic safety for cyclists in roundabouts: Geometry, traffic, and priority rules a literature review," Swedish Nat. Road Transp. Res. Inst., Sweden, Tech. Rep. 31A-2017, 2017.

[9] B. Monsalve, E. Puertas, J. Fernández, and N. Aliane, "Using wearable devices in naturalistic driving to analyze brain activity in roundabout maneuvers," in *Proc. IEEE Int. Conf. Veh. Electron. Saf. (ICVES)*, Sep. 2018, pp. 1–6, doi: 10.1109/ICVES.2018.8519500.

[10] R. Guo, L. Liu, and W. Wang, "Review of roundabout capacity based on gap acceptance," *J. Adv. Transp.*, vol. 2019, pp. 1–11, Feb. 2019, doi: 10.1155/2019/4971479.

[11] O. Giuffrè, A. Granà, and M. L. Tumminello, "Gap-accepteance parameters for roundabouts: A systematic review," *Eur. Transp. Res. Rev.*, vol. 8, no. 1, pp. 1–20, Mar. 2016, doi: 10.1007/s12544-015-0190-4.

[12] Y. Jeong and K. Yi, "Target vehicle motion prediction-based motion planning framework for autonomous driving in uncontrolled intersections," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 1, pp. 168–177, Jan. 2021, doi: 10.1109/TITS.2019.2955721.

[13] S. Lefèvre, D. Vasquez, and C. Laugier, "A survey on motion prediction and risk assessment for intelligent vehicles," *ROBOMECH J.*, vol. 1, no. 1, pp. 1–14, Dec. 2014, doi: 10.1186/s40648-014-0001-z.

[14] S. Klingelschmitt, F. Damerow, V. Willert, and J. Eggert, "Probabilistic situation assessment framework for multiple, interacting traffic participants in generic traffic scenes," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2016, pp. 1141–1148.

[15] S. Lefèvre, C. Laugier, and J. Ibañez-Guzmàn, "Intention-aware risk estimation for general traffic situations, and application to intersection safety," INRIA, France, Res. Rep. RR-8379, 2013.

[16] J. Villagra, A. Artuñedo, V. Trentin, and J. Godoy, "Interaction-aware risk assessment: Focus on the lateral intention," in *Proc. IEEE 3rd Connected Automated Vehicles Symp. (CAVS)*, Nov. 2020, pp. 1–6.

[17] V. Trentin, A. Artuñedo, J. Godoy, and J. Villagra, "A comparison of lateral intention models for interaction-aware motion prediction at highways," in *Proc. 7th Int. Conf. Vehicle Technol. Intell. Transp. Syst. (VEHITS)*, Prague, Czech Republic, 2021, pp. 180–191.

[18] J. Schulz, C. Hubmann, J. Löchner, and D. Burschka, "Interaction-aware probabilistic behavior prediction in urban environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2018, pp. 3999–4006, doi: 10.1109/IROS.2018.8594095.

[19] M. Sackmann, H. Bey, U. Hofmann, and J. Thielecke, "Classification of driver intentions at roundabouts," in *Proc. 6th Int. Conf. Vehicle Technol. Intell. Transp. Syst. (VEHITS)*, 2020, pp. 301–311.

[20] Y. Hu, W. Zhan, L. Sun, and M. Tomizuka, "Multi-modal probabilistic prediction of interactive behavior via an interpretable model," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2019, pp. 557–563, doi: 10.1109/IVS.2019.8813796.

[21] T. Zhang, W. Song, M. Fu, Y. Yang, and M. Wang, "Vehicle motion prediction at intersections based on the turning intention and prior trajectories model," *IEEE/CAA J. Autom. Sinica*, vol. 8, no. 10, pp. 1657–1666, Oct. 2021, doi: 10.1109/JAS.2021.1003952.

[22] Y. Jeong, S. Kim, and K. Yi, "Surround vehicle motion prediction using LSTM-RNN for motion planning of autonomous vehicles at multi-lane turn intersections," *IEEE Open J. Intell. Transp. Syst.*, vol. 1, pp. 2–14, 2020, doi: 10.1109/OJITS.2020.2965969.

[23] C. Vogl, M. Sackmann, L. Kürzinger, and U. Hofmann, "Frenet coordinate based driving maneuver prediction at roundabouts using LSTM networks," in *Proc. Comput. Sci. Cars Symp.* New York, NY, USA: Association for Computing Machinery, Dec. 2020, pp. 1–9, doi: 10.1145/3385958.3430475.

[24] S. Carrasco, D. F. Llorca, and M. Sotelo, "SCOUT: Socially-consistent and UndersTandable graph attention network for trajectory prediction of vehicles and VRUs," 2021, *arXiv:2102.06361*. [Online]. Available: https://arxiv.org/abs/2102.06361

[25] L. Sun, W. Zhan, and M. Tomizuka, "Probabilistic prediction of interactive driving behavior via hierarchical inverse reinforcement learning," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2018, pp. 2111–2117, doi: 10.1109/ITSC.2018.8569453.

[26] A. Quintanar, D. Fernández-Llorca, I. Parra, R. Izquierdo, and M. A. Sotelo, "Predicting vehicles trajectories in urban scenarios with transformer networks and augmented information," 2021, *arXiv:2106.00559*. [Online]. Available: https://arxiv.org/abs/2106.00559

[27] P. Bender, J. Ziegler, and C. Stiller, "Lanelets: Efficient map representation for autonomous driving," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2014, pp. 420–425.

[28] A. Artuñedo, J. Godoy, and J. Villagra, "A primitive comparison for traffic-free path planning," *IEEE Access*, vol. 6, pp. 28801–28817, 2018, doi: 10.1109/ACCESS.2018.2839884.

[29] J. Godoy, V. Jiménez, A. Artuñedo, and J. Villagra, "A grid-based framework for collective perception in autonomous vehicles," *Sensors*, vol. 21, no. 3, p. 744, Jan. 2021.

[30] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174–188, Feb. 2002, doi: 10.1109/78.978374.

[31] J. Elfring, E. Torta, and R. van de Molengraft, "Particle filters: A hands-on tutorial," *Sensors*, vol. 21, no. 2, p. 438, Jan. 2021.

[32] T. Toledo, C. F. Choudhury, and M. E. Ben-Akiva, "Lane-changing model with explicit target lane choice," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 1934, no. 1, pp. 157–165, Jan. 2005, doi: 10.1177/0361198105193400117.

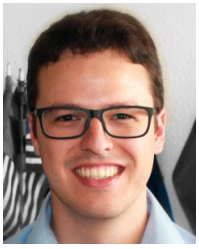
[33] S. Lefèvre, "Risk estimation at road intersections for connected vehicle-safety applications," Ph.D. dissertation, Univ. Grenoble, Grenoble, France, 2012.

[34] S. Lefèvre, D. Vasquez, C. Laugier, and J. Ibañez-Guzmán, "Intention-aware risk estimation: Field results," in *Proc. IEEE Int. Workshop Adv. Robot. Social Impacts (ARSO)*, Jun. 2015, pp. 1–8, doi: 10.1109/ARSO.2015.7428204.

[35] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. Cambridge, MA, USA: MIT Press, 2005.

[36] L. Tiancheng, B. Miodrag, and D. M. Petar, "Resampling methods for particle filtering: Classification, implementation, and strategies," *Signal Process. Mag.*, vol. 32, no. 3, pp. 70–86, May 2015, doi: 10.1109/MSP.2014.2330626.

VINICIUS TRENTIN was born in São Carlos, Brazil, in 1993. He received the bachelor's degree in control and automation engineering from the Federal University of Santa Maria, Brazil, in 2018. During his graduation, he was part of an exchange program in Germany, where he studied at TU Dresden and, later, made an internship with Fraunhofer IIS EAS, Dresden. Since 2019, he has been a Predoctoral Researcher with the Centre for Automation and Robotics (CSIC-UPM), AUTOPIA Program's Team. His research interests include autonomous driving, interaction awareness, and motion prediction.



ANTONIO ARTUÑEDO received the B.Sc. degree in electrical engineering from the Universidad de Castilla–La Mancha, Spain, in 2011, the M.Sc. degree in industrial engineering from the Universidad Carlos III de Madrid, in 2014, and the Ph.D. degree in automation and robotics from the Technical University of Madrid (UPM), Spain, in 2019, with the AUTOPIA Program. During his Predoctoral period, he made a research stay at the Integrated Vehicle Safety Group, TNO, The Netherlands, in 2017. He is currently a Postdoctoral Researcher with the Centre for Automation and Robotics (CSIC-UPM), AUTOPIA Group, Madrid, Spain. He has been working on both national and European research projects in the scope of autonomous vehicles. He has published and peer-reviewed multiple journals and conference papers focused in this field. His research interests include system modeling and simulation, intelligent control, motion planning, and decision-making systems. His thesis won the prize of the Best Ph.D. on Intelligent Transportation Systems by the Spanish Chapter of the IEEE-ITS Society, in 2020.

JORGE GODOY was born in Maracay, Venezuela, in 1986. He received the degree in electronics engineering from Universidad Simón Bolívar, in 2008, and the M.E. and Ph.D. degrees in automation and robotics from the Universidad Politécnica de Madrid, in 2011 and 2013, respectively. From 2013 to 2017, he was a Technical Coordinator of the AUTOPIA Program funded by research contracts from National and European research projects. His research interests include intelligent transportation systems, autonomous driving, path planning, and embedded AI-based control for autonomous vehicles. In 2009, he was granted with a Predoctoral JAE Fellowship from CSIC for researching on autonomous vehicles at the Centre of Automation and Robotics (CSIC-UPM). In November 2017, he was granted with a Juan de la Cierva Fellowship for Postdoctoral Research at the Universidad Politécnica de Madrid.

JORGE VILLAGRA was graduated in industrial engineering from the Universidad Politécnica of Madrid, in 2002. He received the Ph.D. degree in real-time computer science, robotics and automatic control from the École des Mines de Paris, France, in 2006. From 2007 to 2009, he was a Visiting Professor with the University Carlos III of Madrid, Spain. From August 2013 to August 2016, he led the Department of ADAS and Highly Automated Driving Systems at Ixion Industry and Aerospace SL, where he also coordinated all the activities in the EU Research and Development funding programmes. He has been leading the AUTOPIA Program at CSIC-UPM, since October 2016. He has developed his research activity in six different entities with a very intense activity in project setup and management, through over 40 international and national research and development projects, where he is or has been IP of 17 of these projects. He has published over 100 articles in international journals and conferences on autonomous driving, intelligent transportation systems, model-free control, and probabilistic approaches for embedded components in autonomous vehicles.