Comfort-based AI-driven Roundabout Control for Automated Vehicles

Sina Moradi, Ali Nadar, Jérôme Härri
Communication Systems Department
EURECOM
Sophia-Antipolis, France
{moradi,nadar,haerri}@eurecom.fr

***Abstract*—Automated vehicles are emerging with the advancement of driving algorithms and improvement in their sensing, computing, and navigating capabilities. All leading stakeholders are developing prototypes and testing them on roads. If safety is of uttermost importance, efficiency and passenger comfort are becoming critical to their success. In that context, roundabouts are particularly challenging due to the difficulties to interpret the situation and inject into traffic, leading to stop-go situations, inefficient waiting time, and uncomfortable jerk values. In this paper, we present an AI-driven comfort-oriented roundabout control mechanism for automated vehicles. We propose a throttle and steering control respecting passenger comfort metrics, an AI-driven situational analysis to optimize yielding, and a model predictive control algorithm to adjust maneuvers and minimize waiting time at roundabouts.**

***Index Terms*—Autonomous Driving, Machine Learning, Optimization, Passenger Comfort, Roundabout Control**

I. Introduction

Automated vehicles are a major upcoming innovation promising to improve traffic safety and efficiency. All major stakeholders are focusing on their development and studies and anticipate a slow ramp up to 11% by 2030 and replacing more than 60% of legacy vehicles by 2050 [1]. Several pitfalls however lie on their road to success, ranging from dependable sensing of vulnerable road users, handling of complex traffic situations, cybersecurity, or passenger acceptance.

Over the past years, various approaches for motion or trajectory planning have been investigated [2]. More recently, Model Predictive Control (MPC) strategies [3], [4] have been proposed and studied for Cooperative Adaptive Cruise Control (CACC) or Platooning [5], [6]. Yet, when facing complex scenarios with merging of trajectories typically observed at intersections, trajectory planning adopts a combinatorial nature [7], leading to optimizing control according to various potential maneuvering options.

Among various types of intersections, roundabouts have shown to provide potentially higher traffic safety and efficiency if well optimized [8]. Accordingly, a large literature is available on optimizing delay and trajectories (e.g. [8]–[11]), but the roundabout layout is over-simplified, optimizing flows with optimal knowledge of traffic conflicts, and not considering real vehicle's dynamics or passenger comfort.

In this paper, we propose an AI-driven roundabout control considering passenger comfort metrics, such as acceleration and jerk limitations. To that objective, we integrate the impact of realistic roundabout layout, vehicle dynamics, and unknown conflicts to throttle and steering control. Our contributions are threefold: we first introduce an acceleration and braking motion control respecting passenger comfort; second, we describe a steering control minimizing the deviation to a target trajectory respecting passenger comfort; finally, we rely on an AI model predicting motion conflicts and apply an MPC strategy to predict maneuvers, thus minimizing the roundabout access time. We implemented the throttle and steering control in the CARLA[1] simulator, and release the code as open-source[2].

The rest of the paper is organized as follows: Section II describes the control algorithm for comfort-based motion planning, whereas Section III introduces the AI-driven roundabout decision-making for trajectory planning. In Section IV, we provide implementation details and results. Finally, Section V concludes the paper and sheds light on future directions.

II. Comfort-oriented Roundabout Control

In this section, we describe the methodology followed to develop longitudinal and lateral control mechanisms to drive through a roundabout respecting passenger comfort, which is defined jointly as a maximal acceleration/deceleration as well as a jerk limit.

A. Throttle-Velocity Dynamics

To control the speed and acceleration of a vehicle, one needs to access the nonlinear vehicle dynamics in terms of transfer function or alternatively an adaptive state-space model. An example of a first-order control system is one whose input-output relation, usually referred to as a transfer function in the frequency domain (s-domain), is a first-order differential equation. Equation (1) shows such a frequency domain transfer function from a throttle input $U(s)$ to a velocity output $V(s)$, where $kappa$ represents the gain and τ the time delay of the first order dynamics.

$$\frac{V(s)}{U(s)} = \frac{\kappa}{\tau s + 1} \tag{1}$$

[1] CARLA Simulator: https://carla.org/

[2] https://gitlab.eurecom.fr/cats/carla/ai-driven-roundabout-control

978-1-6654-5530-5/23/$31.00 ©2023 IEEE

The state-space equations for a single-input single-output Linear Time Invariant (LTI) control system are shown in (2). The parameters a, b, c, and d are scalars; $x(t)$ is the system's state, $u(t)$ and $y(t)$ are the input and the output of the system, respectively.

$$\begin{cases} \dot{x}(t) = ax(t) + bu(t) \\ y(t) = cx(t) + du(t) \end{cases} \tag{2}$$

According to the system's first-order transfer function, (2) may be simplified assuming $a, c \neq 0$, $b = 1$, $d = 0$ and considering velocity $v(t)$ as the new state, leading to the formulation of our state-space model for LTI system as:

$$\begin{cases} a_x(t) = av(t) + cu(t) \\ \ y(t) = v(t) \end{cases} \tag{3}$$

B. Longitudinal Control

The LTI control described in (3) provides a match between throttle input targeting a particular velocity output. Longitudinal control mechanisms require optimizing a set of throttle inputs to reach and maintain a given velocity respecting acceleration and jerk constraints.

1) Stationary Throttle Optimization: In [12], Malaek and Moradi introduced a longitudinal control of an aircraft to perform zero-gravity flights. An optimization approach identifies the elevator control input (airplane nose up/down) under the constraints of a target acceleration. The authors defined the elevator deflection as a time series polynomial and found the best coefficients that minimize the objective function. In this paper, we use the same approach, but instead of elevator input, we consider a throttle input, with the jerk as a complementary constraint. Considering a first-order transfer function as system dynamics and jerk being the second derivative of the system's state, it suffices to define a second-degree polynomial for the throttle input as described in (4).

$$u(t) = P_1 t^2 + P_2 t \tag{4}$$

The heuristic optimization method utilized in [12] is called TCACS. It is developed based on the Tabu Search technique removing the irrelevant (hyper-)balls from the search space at each iteration, thus converging to the optimal solution (see [13] for further details). According to (4), we have a 2-dimensional search space including combinations of P_1 and P_2.

2) PID Speed Controller Optimization: The above approach considers an open-loop dynamic system and finds the throttle that makes the vehicle reach an *Operating Mode* to activate the speed controller. However, in case we want to reach and keep a target velocity, we need a closed-loop analysis. This can be reached by tuning the gains of a PID controller using popular methods such as Ziegler-Nichols. Considering passenger comfort, and without loss of generality, we tune a PID controller gains, namely, k_P, k_I, and k_D, for a scenario where a vehicle has an initial *Operational Mode* speed and aims to reach and maintain a *Target* speed.

We also need to find an integral bias so that the input at $t = 0$ matches the initial throttle u_0. Assume that the PID controller calculates the input when the vehicle's initial velocity is v_0 and its target speed is v_f. The control input can be calculated as follows:

$$\begin{aligned} &e(t) = v_f - v(t) \\ &u(t) = k_P\, e(t) + k_I \int_0^t e(\tau)\, d\tau \ + bias \\ &bias = u_0 - k_P(v_f - v_0) \end{aligned} \tag{5}$$

C. Braking Control

The main forces decelerating the vehicle while braking are aerodynamic force (f_a), rolling resistance (f_r), and the force due to the brake torque(f_b). With an air density ρ, a drag coefficient C_d, and a frontal surface of S, the aerodynamic force is computed as follows for a given speed $v(t)$.

$$f_a(t) = \frac{1}{2}\rho S C_d v(t)^2 \tag{6}$$

For vehicles with air-filled tires on dry roads, the rolling resistance force can be estimated by (7) where mg represents the weight of the vehicle and C_r is the rolling resistance coefficient.

$$f_r(t) = C_r(t)mg \tag{7}$$

To calculate f_b, assuming a proportional brake torque $\tau_{b_{max}}$ with respect to the brake pedal δ_b, a tire radius of R_T and an equal brake pressure on all tires, we will have (8).

$$f_b(t) = \frac{4\tau_{b_{max}}}{R_T}\delta_b(t) \tag{8}$$

The aerodynamic and rolling resistance forces are larger when the speed increases. Solving (9) considering the mass of the vehicle m and the maximal deceleration a_{max}, we obtain the maximum brake pedal input $\delta_{b_{max}}$.

$$f_{a_{max}} + f_{r_{max}} + f_{b_{max}} = m \cdot a_{max} \tag{9}$$

More realistic brake inputs may be obtained by solving a first-order differential equation like (10) , where η is a scalar coefficient and $\delta_{b_{max}}$ corresponds to the maximum brake input.

$$\eta\dot{\delta}_b(t) + \delta_b(t) = \delta_{b_{max}} \tag{10}$$

D. Steering Control

For steering and lateral control, we consider a simplified bicycle model as depicted in Fig. 1. The steering input is δ, the heading error with respect to the trajectory is $\Delta\psi$, and the cross-track error is e. According to the *Stanley Control Law*, the cross-track error is the distance between the trajectory and the center of the front axle. According to [14], (11) reflects the steering input, which is a path-tracking solution and works regardless of the vehicle under study. The parameters that vary from one vehicle to another are the two gains K_e and K_v.

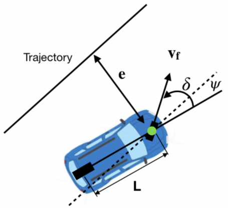

Fig. 1. The bicycle model of a vehicle following a trajectory for defining Stanley Control Law. [source: www.shuffleai.blog]

$$\delta(t) = \Delta\psi(t) + tan^{-1}(\frac{K_e e(t)}{K_v + v(t)}) \tag{11}$$

Lateral acceleration depends on the velocity and the turning radius R. To keep lateral passenger comfort, we need to find a minimum turning radius for a fixed speed and adjust the steering input according to the Stanley Control Law (11). However, since the trajectory is defined in roundabouts, we need to adjust the vehicle's velocity accordingly.

$$a_y = \frac{v^2}{R} \Rightarrow v_{max} = \sqrt{Ra_{y_{max}}} \tag{12}$$

E. Control Constraints respecting Passenger Comfort

Passenger comfort in the longitudinal direction for automated vehicles has been investigated in various literature works. In [15], the maximum deceleration is set to $-3.4\,m/s^2$. In [16], maximum acceleration and jerk (the derivative of acceleration $j(t)$) are set to $2\,m/s^2$ and $0.9\,m/s^3$, respectively. The main challenge remains to control the longitudinal acceleration and specifically the jerk at low speed.

The lateral acceleration is also important to consider. In [17], [18] authors propose that the centrifugal force should not surpass 15% of the vehicle's weight. This means,$|a_y(t)| \leq 0.15g$; where $a_y(t)$ is the lateral acceleration in the vehicle's frame of reference and g represents the gravity of Earth ($g = 9.80665\,m/s^2$).

We therefore integrate both longitudinal and lateral acceleration thresholds described in (13) to the previously described control mechanisms to respect driver comfort.

$$\begin{aligned} -3.4\ m/s^2 \leq a_x(t) \leq 2\,m/s^2 \\ |a_y(t)| \leq 0.15g\ m/s^2 \\ j(t) \leq 0.9\ m/s^3 \end{aligned} \tag{13}$$

III. AI-driven Roundabout Management

In Section II, we describe control longitudinal and lateral control for automated vehicles to drive through roundabouts. Here, we extend them to consider *if* and *where* the vehicle will conflict with other vehicles.

A. Machine Learning Exit Probability Model

We provide an answer to the *if* question by relying on an AI-driven strategy to estimate the exit probability of a vehicle in a roundabout, as proposed by Deveaux *et al.* [19]. The overall scenario is depicted in Fig. 2, where three attributes have been considered, namely, *relative heading*, *lateral position*, and *exit distance* influencing the exit probability. The lateral position influences the exit probability as vehicles located in the outer-most lane have more chances to exit rather than those on the inner-most lanes. The relative heading is considered as the relative angle of the front tire with respect to the roundabout trajectory and also provides indications of the likelihood to exit. Finally, the Exit distance is computed as the normalized distance on the roundabout arc in this paper, rather than the euclidean distance as in [19].

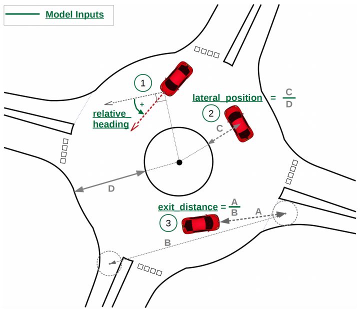
Model Inputs
1
relative heading
lateral_position = $\frac{C}{D}$
2
C
D
exit_distance = $\frac{A}{B}$
A
3
B

Fig. 2. Input attributes for the Exit Probability machine learning model [19]

Training the AI model is done by continuously spawning vehicles on all sides of the roundabout, letting them randomly choose their exists. We initially label the vehicles as 0 at all time frames. Whenever a vehicle departs, we back-propagate to change the labels to 1 until we reach the location matching the previous exit. We opted for a *logistic regression* method thanks to the low dimensional dataset. The outcome of the training indicated that the exit probability of any *staying* vehicle does not exceed a certain threshold, i.e., $P_{exit} \leq\ 0.63$. Consequently, we set the departing threshold to $P_{exit} \geq\ 0.70$.

B. Roundabout Collision Prediction

The *where* question is answered by calculating the merge point between the approaching and the inbound vehicles. According to Fig. 3, a merging point M needs to be located, where vehicle A at a distance d_A to the entry, will impact vehicle B with an acceleration a_B, a velocity v_B, and a distance d_B to M.

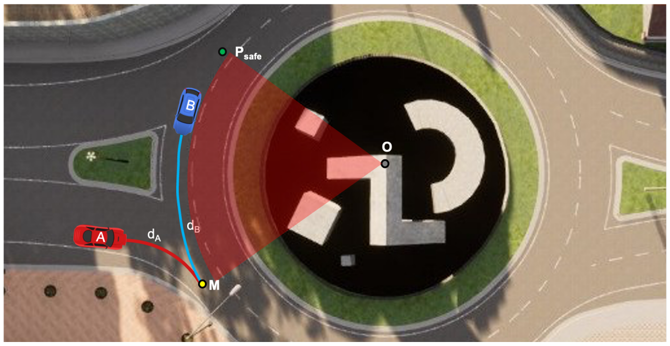

Fig. 3. Decision-making schematic for the roundabout entrance to find P_{safe}.

The time t_A it takes for vehicle A to reach the point M is of the essence, as vehicle A may adjust its own control inputs to set a speed avoiding the impact. We use a Model Predictive Control (MPC) approach to compute the vehicle's velocity and distance at each time step and find t_A accordingly. Let's assume that d_B represents the distance for vehicle B to reach point M in t_A seconds. Hypothetically, this means that both vehicles reach point M at the same time. Thus, we have (14) for d_B based on the kinematics of the vehicle B.

$$d_B = \frac{1}{2} a_B t_A^2 + v_B t_A \tag{14}$$

We also need to add the required braking distance $d_{B_{brake}}$. We considered the worst-case scenario where the vehicle B brakes to stop. This results in the larger value for $d_{B_{brake}}$. Multiplying the calculated distance by a safety factor, which we considered as $\eta_{SF} = 1.1$, we can locate the point P_{safe} on the roundabout arc $\widehat{MOP_{safe}}$ in Fig. 3 as shown in (15).

$$\widehat{MOP_{safe}} = \eta_{SF} (\frac{1}{2} a_B t_A^2 + v_B t_A + d_{B_{brake}}) \tag{15}$$

IV. Results and Discussion

Without loss of generalities, we selected the default Town03 roundabout scenario in CARLA[3]. Parameter optimizations for the PID and steering controllers have been conducted via a co-simulation framework between MATLAB and CARLA, whereas we implemented a Python version of our throttle and steering controller directly in CARLA to negate communication delays. Finally, we opted for a vehicle model supporting automated gear changing, namely a Tesla Model 3, in both MATLAB and CARLA simulation environments.

A. Longitudinal Control

In Table I, you can find the gains (κ) and the time delays (τ) of the first-order vehicle dynamics for (1). We can interpolate between these values according to the velocity of the vehicle to have an adaptive model. Note that to have a system capable of taking its initial condition as an input, we should obtain the state-space representation of that system. This can be done by applying the Inverse Laplace Transform to (1) or using the $tf2ss$ MATLAB command.

We applied the throttle input as a step function from 30% up to 75% with a 5% increment. Then, we saved the velocity output response until the vehicle reached the steady-state condition. MATLAB System Identification Toolbox proposed a first-order control system for the input-output data that we fed to it.

We then implemented the state-space dynamic model described in (3) in Simulink and optimized it separately for an initial stationary regime and for a subsequent PID controller.

Without loss of generalities, we selected to activate the PID speed controller at $12{-}15\,km/h$. We ran the optimization for 2 seconds while stabilizing the throttle at 50% until we reach the target speed. Consequently, (16) represents our optimization argument. By applying the TCACS optimization to the vehicle dynamic model, we obtain (17).

TABLE I
System Identification Parameters for Vehicle Dynamics

Step Throttle	Dynamic System Parameters				
	S.S. Velocity (m/s)	κ	τ	a	c
30%	3.32	11.07	2.20	−0.451	5.023
35%	4.24	12.11	2.33	−0.429	5.200
40%	5.36	13.40	2.82	−0.355	4.752
45%	6.74	14.98	3.12	−0.321	4.801
50%	8.12	16.23	2.98	−0.336	5.446
55%	9.52	17.28	2.94	−0.340	5.878
60%	11.12	18.43	2.97	−0.337	6.205
65%	13.01	20.02	3.48	−0.288	5.755
70%	15.18	21.69	3.91	−0.256	5.551
75%	17.73	23.64	4.23	−0.237	5.594

*Specific for Tesla Model 3 blueprint in the CARLA simulator.

$$arg \min_{P_1, P_2} \{\|j(t) - j_{max}\| + \|u(t=2) - 0.5\|\} \tag{16}$$

$$u(t) = \min\{0.0401t^2 + 0.1698t\,,\, 0.5\} \;,\; 0 \le t \le 4 \tag{17}$$

Figure 4 shows the throttle, acceleration, jerk, and speed values from our longitudinal control. As can be seen, they are close to ideal. First, as they correspond to the maximum allowed jerk before reaching 50% throttle at 2 seconds, also as they strictly respect the comfort requirements as defined in Section II-E. The dynamic model values are taken from Table I with a step throttle of 30%.

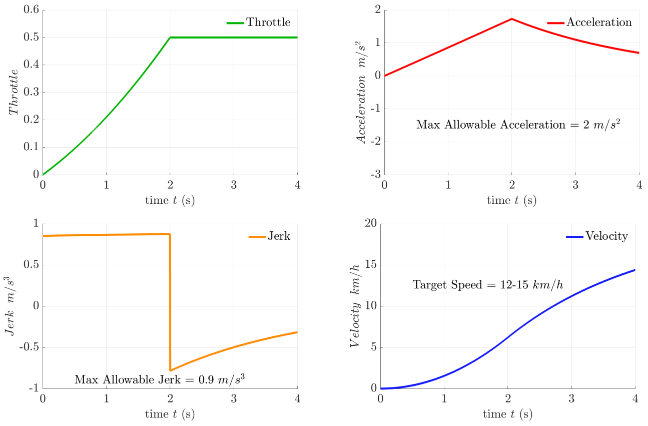

Fig. 4. The vehicle states when it starts moving for four seconds

We then relied on MATLAB PID Tuner to obtain the gains of our PID controller, namely, k_P, k_I, and k_D. Considering an initial speed of $12\,km/h$ and a target speed of $40\,km/h$, we obtained the following PID gains:

$$k_P = 0.08801 \;,\; k_I = 0.04023 \;,\; k_D \approx 0 \tag{18}$$

In Fig. 5 you can find the vehicle's response to the PID controller input. As may also be observed, the PID leads to driving parameters strictly respecting the target comfort-based values.

[3] https://carla.org/

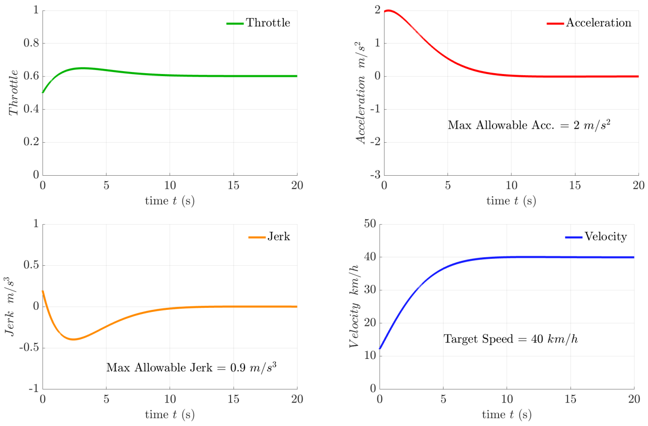

Fig. 5. The vehicle states showing the performance of the PI controller

Finally, the braking functions described in Section II-C have been implemented considering an air density of $\rho = 1.225kg/m^3$, drag coefficient of $C_d = 0.30$, and $S = 2.22m^2$ for the frontal area, tire pressure of $p_t = 3\ bars$, a tire radius of $R_T = 0.37m$, a maximum torque of $\tau_{b_{max}} = 1500N \cdot m$, and a vehicle mass $m = 1847kg$. Thus, (6), (7) and (8) maybe be simplified and $\delta_{b_{max}}$ solved to respect the comfort-based maximum deceleration value.

$$\begin{aligned} f_a(t) &= 0.408 \cdot v(t)^2 \\ f_r(t) &= (0.005 + (\frac{1}{3})[0.01 + 0.0095(\frac{3.6 \times v(t)}{100})^2]) \cdot mg \\ f_b(t) &= 16216 \cdot \delta_b(t) \\ \delta_{b_{max}} &= 0.375 \end{aligned}$$

Setting $\delta_b(t=1) = 0.95\,\delta_{b_{max}}$, $\eta = \frac{1}{\ln 20}$ results. Thus, the optimal brake input may be obtained as

$$\delta_b(t) = 0.375(1 - e^{3t})$$

Fig. 6 shows the vehicle's response at $40\,km/h$ to this brake input. As it can be seen, the deceleration remains smooth and respect the maximum comfort-based deceleration as required in Section II-E.

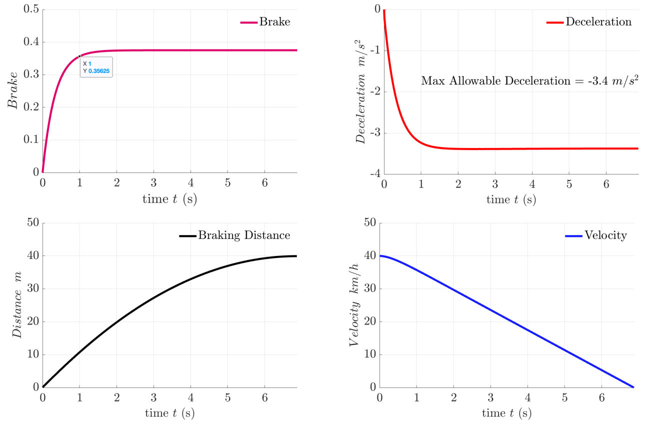

Fig. 6. The vehicle response to the brake input at $40\,km/h$

B. Steering Control

To tune the Stanley control gains K_e and K_v from (11), we considered a large enough search space for them to apply the TCACS optimization algorithm. The objective function to minimize is defined as the norm of the cross-track error for the trajectory where the vehicle enters the roundabout, traverses a full circle, and departs at the end. Upon convergence, the optimal gain values $K_e = 18.9$ and $K_v = 2.2$ have been selected, which provides the steering input as

$$\delta(t) = \Delta\psi(t) + tan^{-1}(\frac{18.9e(t)}{2.2 + v(t)}) \quad (19)$$

Considering the maximum comfort-related acceleration values in Section II-E, (12) is reduced to $v_{max} = \sqrt{0.15Rg}$, where g is the gravity and R the turning radius. Accordingly, we jointly adjust the steering and PID controller to respect the target trajectory and the maximum speed corresponding to the curvature of the trajectory.

Figure 7 compares our optimized comfort-based steering controller and the CARLA *Autopilot*. There are four locations on the roundabout creating abrupt steering changes. Our optimized steering control manages to correct and stabilize it, whereas the *Autopilot* creates oscillatory steering, impacting not only the trajectory but also the comfort of the passengers. A video is available here.

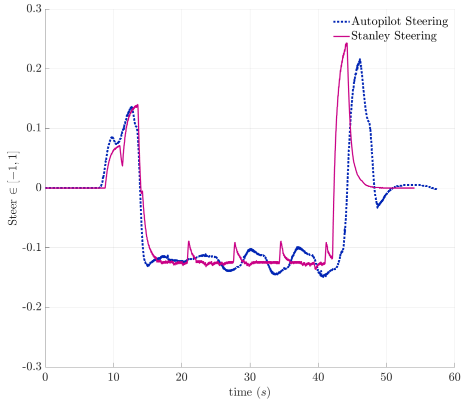

Fig. 7. Steering control comparison against CARLA Autopilot

C. AI-driven Roundabout Access Management

We implemented the roundabout access decision-making as a flow chart described in Fig. 8. The *Turtle approach* refers to the case when the automated vehicle must defer its access by slowing down rather than stopping. The vehicle approaching the roundabout first creates a list of incoming vehicles provided by the AI model and computes their respective P_{safe}. If the nearest vehicle is far enough, the vehicle may access the roundabout at the current speed. Otherwise, it decelerates and either performs a Turtle approach and recomputes P_{safe} in the next iteration or stops. Meanwhile, if the AI model-driven exit probability of the closest coming vehicle shows a probability higher than 70%, we remove the vehicle from the list. Figure 9 depicts a screenshot of the CARLA roundabout with our proposed control strategy. A video is available here.

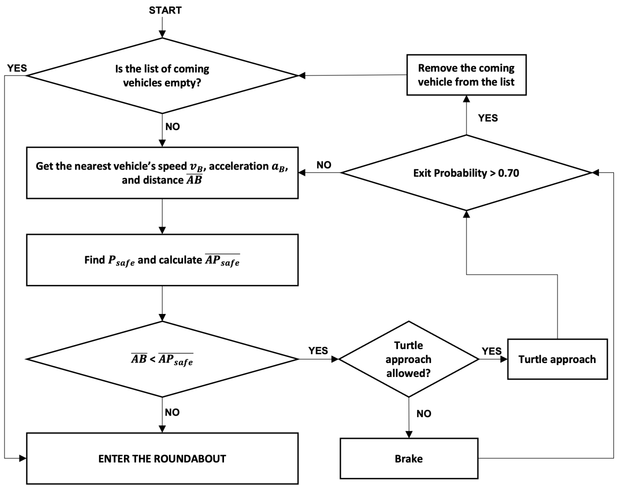
START
Is the list of coming vehicles empty?
YES
NO
Get the nearest vehicle's speed v_B, acceleration a_B, and distance $\overline{AB}$
Find P_{safe} and calculate $\overline{AP_{safe}}$
$\overline{AB} < \overline{AP_{safe}}$
YES
NO
Turtle approach allowed?
YES
NO
Turtle approach
Brake
ENTER THE ROUNDABOUT
Exit Probability > 0.70
YES
NO
Remove the coming vehicle from the list

Fig. 8. The flowchart for decision-making at the roundabout entrance.

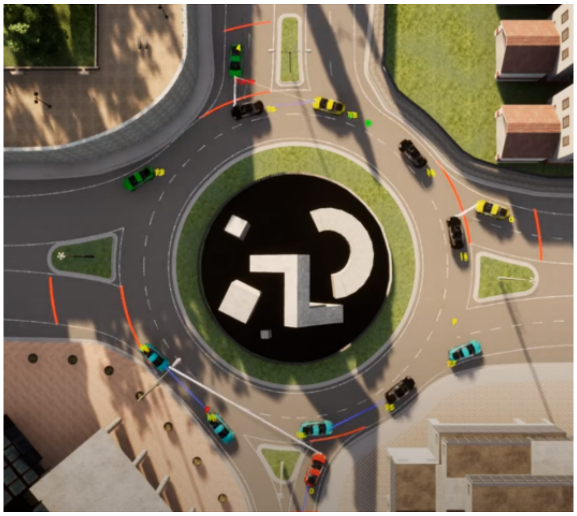

Fig. 9. Illustration of the comfort-based AI-driven control mechanism for Roundabout

V. Conclusion

In this paper, we proposed a control algorithm and conflict avoidance strategy for automated vehicles to drive through roundabouts safely, efficiently, and comfortably. In terms of safety, our control strategy integrates a predictive AI-driven maneuver to optimize yielding at roundabouts, and in terms of efficiency, a turtle mechanism is implemented to mitigate stopping. Regarding comfort, the jerk is particularly challenging as it requires specific control inputs at a low throttle regime and stable steering. Our proposed mechanism respects it for both throttle and steering control. We used MATLAB to calculate the various parameters of our control mechanisms, then implemented it in Python for CARLA, and released it as open-source[4]. In future work, we will extend our control mechanisms to support maneuver coordination and multi-lane roundabouts. We will also integrate an AI-as-a-Service (AIaaS) framework to obtain optimal AI models.

[4] https://gitlab.eurecom.fr/cats/carla/ai-driven-roundabout-control

References

[1] D. Milakis, M. Snelder, B. v. Arem, B. v. Wee, and G. Homem de Almeida Correia, "Development and transport implications of automated vehicles in the netherlands: scenarios for 2030 and 2050," *European Journal of Transport and Infrastructure Research*, vol. 17, no. 1, Jan. 2017. [Online]. Available: https://journals.open.tudelft.nl/ejtir/article/view/3180

[2] B. Paden, M. Čáp, S. Z. Yong, D. Yershov, and E. Frazzoli, "A survey of motion planning and control techniques for self-driving urban vehicles," *IEEE Transactions on Intelligent Vehicles*, vol. 1, no. 1, 2016.

[3] J. Ziegler, P. Bender, T. Dang, and C. Stiller, "Trajectory planning for bertha — a local, continuous method," in *2014 IEEE Intelligent Vehicles Symposium Proceedings*, 2014, pp. 450–457.

[4] J. Nilsson, P. Falcone, M. Ali, and J. Sjöberg, "Receding horizon maneuver generation for automated highway driving," *Control Engineering Practice*, vol. 41, pp. 124–133, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0967066115000726

[5] V. Milanés and S. E. Shladover, "Modeling cooperative and autonomous adaptive cruise control dynamic responses using experimental data," *Transportation Research Part C: Emerging Technologies*, vol. 48, pp. 285–300, 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0968090X14002447

[6] R. H. Patel, H. Wymeersch, J. Härri, and C. Bonnet, "Buffer-aided model predictive controller to mitigate model mismatches and localization errors," *IEEE Transactions on Intelligent Vehicles, October 2018*, 2018.

[7] P. Bender, O. S. Tas, J. Ziegler, and C. Stiller, "The combinatorial aspect of motion planning: Maneuver variants in structured environments," in *2015 IEEE Intelligent Vehicles Symposium (IV)*, 2015, pp. 1386–1392.

[8] L. Zhao, A. Malikopoulos, and J. Rios-Torres, "Optimal control of connected and automated vehicles at roundabouts: An investigation in a mixed-traffic environment," *IFAC-PapersOnLine*, vol. 51, no. 9, pp. 73–78, 2018, 15th IFAC Symposium on Control in Transportation Systems CTS 2018.

[9] R. Mohebifard and A. Hajbabaie, "Connected automated vehicle control in single lane roundabouts," *Transportation Research Part C: Emerging Technologies*, vol. 131, p. 103308, 2021.

[10] A. Danesh, W. Ma, C. Yu, R. Hao, and X. Ma, "Optimal roundabout control under fully connected and automated vehicle environment," *IET Intelligent Transport Systems*, vol. 15, no. 11, pp. 1440–1453, 2021.

[11] A. Koumpis, D. Zorzenon, and F. Molinari, "Automation of roundabouts via consensus-based distributed auctions and stochastic model predictive control," in *2022 European Control Conference (ECC)*, 2022, pp. 14–20.

[12] S. Malaek and S. Moradi, "Trajectory optimization of microgravity atmospheric flights based on a hybrid heuristic algorithm," *5th CEAS Conference on Guidance, Navigation and Control*, April 3-5, 2019, Politecnico di Milano, Italy, pp. 1-20.

[13] A. Karimi, H. Nobahari, and P. Siarry, "Continuous ant colony system and tabu search algorithms hybridized for global minimization of continuous multi-minima functions," *Computational Optimization and Applications*, vol. 45, no. 3, pp. 639-661, 2010.

[14] G. M. Hoffmann, C. J. Tomlin, M. Montemerlo, and S. Thrun, "Autonomous automobile trajectory tracking for off-road driving: Controller design, experimental validation and racing," in *2007 American Control Conference*, 2007, pp. 2296–2301.

[15] J. Wang, K. Dixon, H. Li, and J. Ogle, "Normal deceleration behavior of passenger vehicles at stop sign-controlled intersections evaluated with in-vehicle global positioning system data," *Transportation Research Record*, vol. 1937, pp. 120–127, 01 2005.

[16] I. Bae, J. Moon, and J. Seo, "Toward a comfortable driving experience for a self-driving shuttle bus," *Electronics*, vol. 8, p. 943, 08 2019.

[17] D. Martin and D. Litwhiler, "An investigation of acceleration and jerk profiles of public transportation vehicles," 06 2008, pp. 13.194.1–13.194.13.

[18] I. Bae, J. Moon, and J. Seo, "Toward a comfortable driving experience for a self-driving shuttle bus," *Electronics*, vol. 8, no. 9, 2019. [Online]. Available: https://www.mdpi.com/2079-9292/8/9/943

[19] D. Deveaux, T. Higuchi, S. Uçar, J. Härri, and O. Altintas, "A knowledge networking approach for ai-driven roundabout risk assessment," in *2022 17th Wireless On-Demand Network Systems and Services Conference (WONS)*, 2022, pp. 1–8.