Bio-inspired PIDP Parameters Learning from Expert Drivers for Autonomous Vehicle Navigation in Highly Dynamic Roundabouts*

Guillaume Poidatz[1] and Lounis Adouane[2]

Abstract— **A robust motion planning architecture is crucial for safe autonomous driving cars that navigate in complex environments. Learning from human drivers is a promising way to improve the decision-making process for an Intelligent Vehicle (IV), particularly in highly dynamic and interactive driving situations, such as what we can encounter in roundabouts. In this paper, a bio-inspired approach is proposed to enhance the model-based decision-making architecture proposed in [1] and [2]. An appropriate approach for capturing the decision-making process of a driver based on a bird's-eye view representation is developed. It is mainly based on the use of PIDP (Predictive Inter-Distance Profile) safety metrics as a way to catch the suitable behavior of expert drivers according to the encountered situation, while exploiting its relative dynamic according to the other road-users. In addition, a dedicated simulation tool was developed to validate the proposed bio-inspired architecture, based on several data from tight interactive situations in a roundabout.**

ACRONYMS

FIS	Fuzzy Inference System
ELC	Elliptic Limit-Cycle
IV	Intelligent Vehicle
MRAM	Multi-Risk Assessment and Management
PD	Proportional-Derivative
PIDP	Predictive Inter-Distance Profile
SNR	Safety Not Respected

I. INTRODUCTION

Roundabouts are a type of intersection designed to improve traffic flow and reduce accidents by up to 70% [3]. However, decision-making for IV at such intersections remains complex due to the absence, in general, of explicit indicators like traffic lights. Although this characteristic improves traffic flow by offering greater flexibility, it also poses significant challenges for autonomous navigation to cross it (so, go inside, navigate inside, and get out).

In this highly dynamic context, the decision-making process for IV needs to be reliable for effective navigation. In the literature, decisional architectures range from modular to end-to-end pipelines [4] [5]. In [4], the authors conclude that the modular architecture (pipeline with different modules for the perception, localization, planning and control) is explainable but hard to maintain and suffers from an information bottleneck between the different modules. For the end-to-end architecture, the paper concludes that the approach is easier to maintain, has better performance but suffers issues of explainability, security (such as adversarial attacks). In [5], the authors highlight that rule-based methods are explainable and well-known but lack generalization (adaptation to new contexts), in contrast to end-to-end machine learning approaches, which have an important ability to generalization but mainly suffer from explainability issues. Finally, for hybrid architectures, the explainability is reachable but only inside of the limits stated by an Operational Design Model (ODD) [5]. Thus, it is important to define where the behavior of the system is really critical to constraint it with rule-based explainable method. Moreover, thanks to the use of learning-based methods, decision-making architectures could better manage the adaptation to different contexts, and have thus a longer decision-making time-horizon thanks to a better comprehension of the tight interactions between the vehicles. It is important to highlight that safe and reliable hybrid architectures need both: an accurate task modeling/formalization and good data quality [5].

In the proposed paper, a hybrid control architecture is proposed for autonomous navigation within a highly dynamic roundabout. This architecture combines the performance of machine learning approaches with the intelligibility of model-based methods. Several research works have addressed the hybrid control architecture for roundabouts, such as in [6] or [7]. The study given in [6] proposes a comprehensive architecture for roundabout navigation, although it does not account for dynamic environments. This architecture stands out in that it integrates perception directly into the decision-making pipeline, whereas most IV decision-making frameworks typically treat perception as a separate, preprocessed input. The work given in [7] tackle the problem of navigation inside roundabout by computing the collision risk before insertion using 4 inputs (speeds and distances to the potential collision point of both vehicles) to train a multilayer perceptron, but does not modulate the IV's speed to the situation. The authors in [8] combine game theory and neural networks, designing a three-layer network based on vehicle payoffs computed from Time-To-Collision (TTC) and time of safe crossing. Thanks to this customized neural network, the architecture remains explainable while maintaining the performance of standard neural networks. The work given in [9] uses k-means clustering to learn speed profiles and adapt the speed of an IV when entering an intersection.

The architecture called MRAM-CS (Multi-Risk Assessment and Management Control Strategy) [1] [10] [2] ad-

*This work was supported by the French Government, partly through the ANNAPOLIS ANR project and partly through the CPER RITMEA program, with the support of the Hauts-de-France Region.

[1,2] Guillaume Poidatz and Lounis Adouane are with Université de Technologie de Compiègne (UTC), Heudiasyc Laboratary (UMR CNRS/UTC), 60203 Compiègne, France (e-mails: guillaume.poidatz105@icloud.com; lounis.adouane@hds.utc.fr).

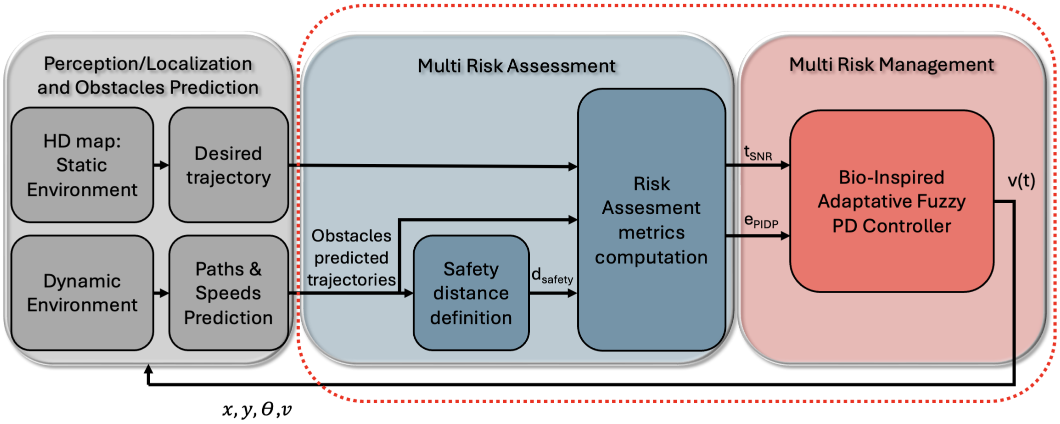


Fig. 1: The MRAM-CS Architecture and its update to include the bio-inspired part. It is divided into 3 main parts: The first part (in grey) is the path, speed prediction and behavior characterization of the vehicle. The second part (in blue) is the assessment of the risk based on the PIDP metrics, and the third part (in red) is the management of this risk and control of the vehicle which use in original architecture an adaptive fuzzy inference system (FIS) PD controller.

dresses the challenge of safe and reliable dense roundabout crossing by leveraging the PIDP (Predictive Inter-Distance Profile) metrics [11] for decision-making. This architecture is explainable, a crucial feature for critical systems like IVs, and is also highly generalizable. Nevertheless, the MRAM-CS architecture can be further improved by tuning its parameters using machine learning techniques, enhancing its performance (while becoming not too conservative), without compromising neither the explainability of the model-based approach nor its safety.

Indeed, the MRAM-CS architecture approach computes all its parameters based on automatic based control principles (cf. section II). This approach could be too conservative in certain configurations and dynamic of the road-users, and can be quite different from a human behavior resulting in a lack of comfort for the IV's passengers. These limitations could be solved as highlighted for instance in [12], where a learning-based approach achieved a 62.8% reduction in driver intervention time and a 62.2% reduction in the number of interventions compared with a standard Adaptive Cruise Control (ACC) system. Moreover, the use of learning-based approach is usually more generalizable than the already proposed MRAM-CS approach. It is thus proposed in this paper an extension of MRAM-CS approach while enabling online learning of the architecture parameters for a customized driving according to the preferences of the passengers, increasing comfort and the architecture performances. This online learning system also allows us to learn from outlier situations to continuously improve the system's safety. An interesting data-set to address this challenge of a bio-inspired MRAM-CS architecture is the rounD dataset [13]. This dataset has been utilized in [14] for motion forecasting. In their study, they benchmarked motion forecasting models using three approaches: a Graph Neural Network, a Long Short-Term Memory network, and their own architecture, which combines a Gated Recurrent Unit with graph-based modules. In this paper, an improvement of the MRAM-CS architecture [2] is proposed by incorporating bio-inspired parameters derived from expert drivers.

This paper is structured as follows. Section II will present the MRAM-CS global architecture as initial background. Section III will give the details of the proposed process of learning the parameters from expert drivers. Section IV will show the experimental setup and the simulation results. Among other elements, this last section presents the simulation tool developed to validate the algorithm, along with the criteria guiding the dataset selection. Section V will give some conclusions and prospects.

II. AN OVERVIEW OF MRAM-CS ARCHITECTURE

To contextualize the main contribution of this paper, this section briefly introduces the MRAM-CS architecture along with the PIDP metric at its core.

A. Path Planning

The MRAM-CS (cf. Fig. 1) architecture first pre-computes a curvilinear path based on the ELC (elliptic limit-cycled) [15] [16]. Such planning approach for IV has been developed and used to deal with different tasks, such as for overtaking in highway [11] and/or roundabout crossing [2]. The dynamical and reactive path generated by the IV to navigate in the roundabout is given by equations given in (1).

$$\begin{cases} \dot{x}_s = m y_s + \mu x_s \left(1 - \frac{x_s}{a_l^2} - \frac{y_s^2}{b_l^2} + c x_s y_s\right) \\ \dot{y}_s = -m x_s + \mu y_s \left(1 - \frac{x_s}{a_l^2} - \frac{y_s^2}{b_l^2} + c x_s y_s\right) \end{cases} \quad (1)$$

Where: x_s and y_s are the coordinates of the center of the roundabout. m being the direction of the trajectory (equal to "1" for clockwise direction and "-1" for anti-clockwise direction). a_l and b_l being the axes of the ellipse. Here, as it is a roundabout (circle), $a_l = b_l$. μ is the parameter to set the speed of the IV trajectory convergence toward the ellipse of influence. This parameter is directly related to the

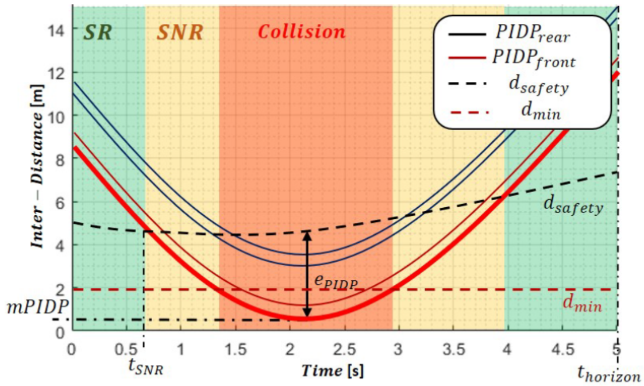


Fig. 2: e_{PIDP} and t_{SNR} extraction. The e_{PIDP} is the difference between: the predicted most dangerous value of the PIDP at the moment the PIDP is predicted as the smallest one, and the inter-distance which is considered to be safe at this moment (represented by d_{safe} in the figure). The t_{SNR} is the first time based on the predicted most dangerous PIDP when the PIDP is below d_{safe} [10].

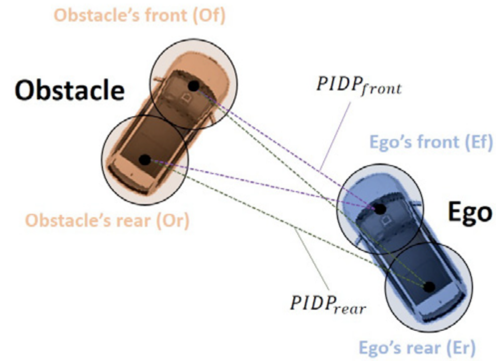


Fig. 3: Computation of the PIDPs: front-front, front-rear, rear-front, rear-rear (IV - obstacle vehicle). Based on the most dangerous PIDP, the taken action will be different: acceleration for rear-front and rear-rear and deceleration for front-rear and front-front [1].

longitudinal speed of the vehicle. Please refer to [15] [16] for more details.

B. Risk Assessment Metric

The used risk assessment is based on two metrics derived from the PIDP [11]: the PIDP error (e_{PIDP}) and the time of "Safety Not Respected" (t_{SNR}) (cf. Fig. 2). The PIDP represents the progress of the Euclidean distance between two vehicles based on their predicted motions, providing continuous spatio-temporal risk assessment and thus offering a more detailed assessment of the risk throughout the maneuver. As illustrated in Fig. 3, four PIDPs are constructed for each potential collision scenario (front-front, front-rear, rear-front, rear-rear), and the minimum value across the time horizon is selected as the most dangerous case, as at this moment the obstacle vehicle and the IV are predicted to be the closest throughout the used horizon of time.

From this PIDP, two values are extracted (cf. Fig. 2): the mPIDP, representing the closest predicted moment between the IV and the obstacle vehicle, and the t_{SNR}, the moment where the PIDP crosses d_{safety}. This d_{safety} (equation 2) combines a static term — the sum of both vehicle radius and a margin depending on the obstacle behavior [1] (cf. equation 3), and a dynamic term, equal to the relative speed times t_{safety}, set to 2 seconds [17]. The e_{PIDP} (cf. equation 4) is then computed as the difference between mPIDP and d_{safety}, and used alongside t_{SNR} to determine K_p and K_d in the PD controller (cf. equation 5).

$$d_{safety} = d_{min} + t_{safety}|\Delta v_{lon}| \tag{2}$$

$$d_{min} = R_{VA} + R_{VO} + Marge \tag{3}$$

$$e_{PIDP} = d_{safety} - m_{PIDP} \tag{4}$$

$$v(t) = v(t-1) + K_p e_{PIDP_t} + K_d \frac{e_{PIDP_t} - e_{PIDP_{t-1}}}{\Delta t} \tag{5}$$

C. The Adaptive Fuzzy PD Controller

After computing the e_{PIDP} and t_{SNR} metrics, the architecture then calculates the parameters of an adaptive fuzzy inference system (FIS) PD controller. This FIS controller adapts the speed of the IV, throughout the equation 5 to the current interactive situation. Only the longitudinal speed is adapted, the path remains the same, based on an ELC surrounding the roundabout. In [10], the velocity profile is based on an FIS PD controller, determined by a multi-criteria optimization [10]. In the proposed work, an improvement of this FIS controller is enabled by learning directly from real drivers.

III. LEARNING PROCESS BASED ON BIO-INSPIRED PIDP PARAMETERS

Ensuring both the safety and intelligibility of autonomous systems, while maintaining its high performance, is ensued in this paper through a hybrid architecture, combining model-based and data-driven approaches. While the core principles of the MRAM-CS architecture remain unchanged, this section introduces the proposed contribution aiming at learning from expert drivers to enhance system performance and human-likeness. First, the hypothesis of the dynamic/path of the road users will be presented in sub-section III-A. Then the expert drivers' selection process will be detailed (cf. sub-section III-B) because it is only with this appropriate data that the used machine learning approach will be relevant. Then the data extraction of the PD controller inputs and outputs from raw data will be explained (cf. sub-section III-C). Finally, from the refined data, the training of the adaptive FIS PD will be explained (cf. sub-section III-D).

A. Hypothesis of the dynamic/path of the road users

In order to build a structured dataset with MRAM-CS-specific metrics for both training and validation, some hypothesis have been taken. Indeed, to compute the inputs of

Fig. 4: Hypothesis on the road-users' predicted trajectory. The vehicles follow a constant velocity model based on a data history of 400ms. The horizon of time of the prediction is 5 seconds. The trajectory of the IV (in red) is based on ELC trajectory and the obstacle vehicles are following a circular trajectory. The interactive area. Is considered as an obstacle vehicle (in interaction with the IV), a vehicle inside the ring of the roundabout and in the area of interaction of the IV. This area of interaction is considered in this study as 20 m radius circle. This radius is function of the size of the roundabout (25 in the studied example).

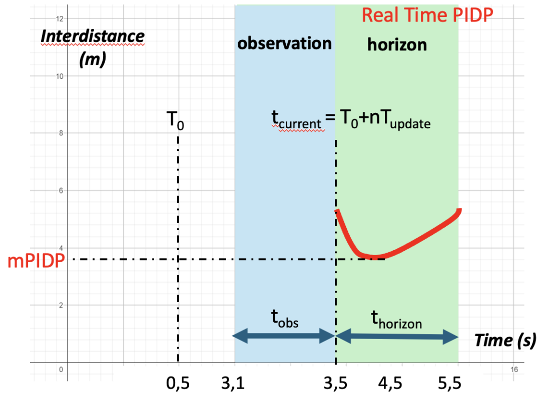
the adaptive FIS PD controller (the e_{PIDP} and the t_{SNR}), some assumptions on the motion prediction have to be done. The first assumption is that the obstacle vehicles trajectories predictions are considered as a constant radius circular trajectory around the roundabout (cf. Fig. 4). The second assumption is that the vehicle driven by the expert driver (the ego-vehicle in our study) will follow the precomputed trajectory such as defined in section II-A. With this model, the future motion of the driver is assumed to be centered within the ring of the roundabout. The final assumption regarding the motion prediction of both, the obstacle vehicles and the expert-driven vehicle is that they follow a constant velocity model. Although some experiments were conducted using a constant acceleration model, the constant velocity approach proved to be more accurate and was thus ultimately selected. This choice is also relevant considering the state of the art that often uses a constant velocity model as baseline particularly when the prediction horizon does not need to be high [18].

The used constant velocity model considers a history of $400ms$ in order to project the future maintained constant velocity. With this choice, the model is enough reactive but in the same time it is long enough to be relevant and smooth, avoiding capturing and propagating noise in the model. The horizon of the prediction is set at $5s$. Although the prediction horizon may appear too long for a simple constant velocity model, the process is executed at a very high frequency, enabling the trajectory of the IV to be gradually corrected. Also, a $5s$ horizon captures longer term interactions which is particularly useful in a dense interaction area such as a roundabout. Making these assumptions, rather than relying on ground-truth future trajectories, is essential. Indeed, when the expert driver decides to accelerate or decelerate, they do so without knowledge of the future; instead, they rely solely on estimations of other drivers' behavior. Similarly, the expert does not know precisely their own future speed, as it continuously adapts to the evolving traffic context. The only information available to the driver concerns the intended exit from the roundabout and whether they will circulate on the left or right side of the ring, depending on the chosen exit.

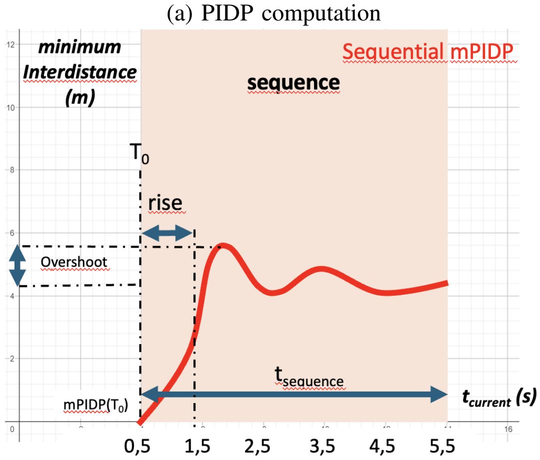
B. Selection of Expert Drivers

In this work, the dataset used is the rounD dataset [13], which provides bird's-eye-view representations of roundabouts. After a review of the state of the art of bird-view datasets including the Argoverse2 [19] and Waymo Open Dataset [20], the RounD dataset has been selected as it is one of the few specifically focused on roundabouts, alongside the INTERACTION dataset [21]. The other datasets offer generally unique scenarios, making them more difficult to observe vehicle behavior in roundabout situations. The rounD dataset was chosen over the INTERACTION dataset due to its higher sampling frequency, and the absence of clear information on the accuracy of INTERACTION dataset. To ensure high-quality learning, the dataset has been filtered to keep only data corresponding to safe drivers in interactive situations. First, a manual annotation of all the vehicles driven by a human has been made in order to classify each insertion situation as interactive or not interactive reducing the number of relevant situations by half. Then, an interaction area has been created (cf. Fig. 4) and only the vehicles that were inside this area at the beginning of the insertion have been selected as obstacle vehicles.

The last filtering before using the data is to observe that there is indeed a control that has been done by the driver to regulate its e_{PIDP} with the obstacle vehicle (cf. section II-B and Fig. 2). In order to do this, for each relevant data that has been previously selected, a sequential mPIDP is built (cf Fig. 5). Each sequence is starting at a time T0 which is the beginning of insertion and at each timestep T_{update}, a new PIDP is computed. The time horizon of the prediction is decreasing until 0. Thus, every T_{update}, the risk assessment metrics are updated and converging to the real inter-distance between the vehicle driven by a human and the obstacle vehicle at the end of the horizon of time. After getting these sequential mPIDP, relevant sequential data are filtered if a "plateau" appears at the end of sequence (cf. Fig. 5). With the T_{update} of 0.12 seconds that has been chosen, the sequential mPIDPs have 41 real-time mPIDP samples. To detect a "plateau", the criteria used here is the gap between the last value and the 31^{st} value that should be below 1%. After all these filtering, the training is done on 47 human drivers. The initial number of drivers was over 450 drivers, thus only 10.5 percent of them has been kept for the training, since they are the most interacting scenarios (between the considered ego-vehicle which starts its crossing and the other obstacles-vehicles already in the roundabout).



(a) PIDP computation



(b) Progress of m_{PIDP}

Fig. 5: Sequential mPIDP. T0: beginning of the sequence. The PIDP is computed at each T_{update} until $t_{horizon}$ is null. Each real-time PIDP is performed over a progressively shorter time window, ensuring that the prediction horizon remains aligned with the prediction horizon at T0.

C. Data Extraction based on an optimization process

After the extraction of relevant insertion sequences only, the data has been conditioned in order to build a database that can be used to train a model. This database needs as inputs the e_{PIDP} and t_{SNR} of each sequence and as outputs the most suitable K_p and K_d (cf. section II-B).

The e_{PIDP} and t_{SNR} of each sequence are based on the first real-time PIDP of each sequence. In order to train the adaptive FIS PD controller, the outputs of each sequence have been built. Indeed, the K_p and K_d cannot be directly extracted from the raw data of the database and thus, have to be estimated. In order to do so, J (cf. equation 6) is minimized with the Limited-memory Broyden–Fletcher–Goldfarb–Shanno (L-BFGS) [22] algorithm. This is a Quasi-Newton method with which it is possible to put some constraints on the values of K_p and K_d (upper and lower limits) and thus accelerating the convergence to a solution.

The equation (6) of J is a combination between two cost functions: the first linked to the PD controller parameters based on a deterministic response evaluation (cf. equation 7) as given in [10], and the biomimetic cost function part based on mPIDP (cf. equation 8). In this equation $\omega_{PD} = 0.5$ and $\omega_{mPIDP} = 0.5$.

The PD cost function is composed of 4 sub-cost functions. The first one is $J_{RiseTime}$ which is the time necessary for the IV to reach 95% of d_{safety}. The second PD cost function is $J_{Overshoot}$ which is the amplitude of the first overshoot above d_{safety}. The third PD cost function is $max(J_{AccOrDec})$ which aims to minimize excessively high acceleration of the IV. The fourth PD cost function is J_{area} which is the cumulative mPIDP error over the time horizon. The used weights are respectively $\omega_1 = 0.1$, $\omega_2 = 0.4$, $\omega_3 = 0.3$ and $\omega_4 = 0.4$. The mPIDP cost function tries to minimize the difference between the profile of the expected sequential mPIDP (the vehicle driven by a human) and the sequential mPIDP with the MRAM-CS architecture. The hybridization of these two cost functions prevents overfitting and ensures a reliable behavior of the IV while closely matching the behavior of human drivers.

$$J = \omega_{PD} J_{PD} + \omega_{mPIDP} J_{mPIDP} \tag{6}$$

$$\begin{aligned} J_{PD} = \omega_{RiseTime} \times J_{RiseTime} + \omega_{Overshoot} \times J_{Overshoot} \\ + \omega_{AccOrDec} \times J_{AccOrDec} + \omega_{area} \times J_{area} \end{aligned} \tag{7}$$

$$J_{mPIDP} = \frac{1}{n} \| mPIDP_{biomimetic} - mPIDP_{real} \|_2^2 \tag{8}$$

D. Training of the Bio-inspired Adaptive Fuzzy PD Controller

After pre-processing the database, the Bio-inspired PD controller can be trained. The selected algorithm to generate the adaptive FIS PD controller is the Fuzzy C-Means clustering algorithm. The generated system is a Type-1 Mamdani fuzzy inference system with Gaussian partitions [23]. The Euclidean distance is used as the distance metric to assess the proximity between the data points and cluster centers, and then generate the fuzzy partitions. The maximum number of iterations is set to 100, with a minimum improvement threshold of 10^{-5}.

IV. EXPERIMENTAL SETUP AND SIMULATION RESULTS

The proposed methodology was tested on a custom-made simulation tool using a set of sample cases to demonstrate its effectiveness. This section presents the results, along with all the necessary elements to understand the simulation environment, including the dataset and the simulation results.

A. Simulation tool

The simulation tool shown in Fig. 6 has two main purposes: to analyze the dataset and to validate the bio-inspired MRAM-CS architecture. For dataset analysis, the interface allows the user to play or pause the visualization, navigate forward or backward, adjust the playback speed, and directly select specific frames for detailed inspection. Numerous metrics such as speed, acceleration, PIDP, mPIDP, and sequential mPIDP are available by simply clicking on a vehicle. Upon selection, the vehicle driven by the human driver is highlighted in red, the one controlled by the MRAM-CS architecture in white, and obstacle vehicles are shown in various colors (except blue). Additionally, the motion predictions for all relevant vehicles are displayed. For the validation of the bio-inspired MRAM-CS architecture, the system is directly integrated into the tool, allowing users to toggle between different control architectures. In this case, users can switch between the bio-inspired PD controller and a constant PD controller which can be tuned in real time through the interface. The metric used for the validation is the longitudinal displacement error between the vehicle controlled with the bio-inspired MRAM-CS architecture and the vehicle controlled by the driver. The simulation also offers a video export feature, enabling users to record and save demonstrations.

B. Simulation results

A first result is shown in Fig. 7. This figure presents a sample from 10 mPIDP sequences selected for training. It can be observed that all vehicles driven by a human effectively regulate their speed in response to the interactive obstacle vehicles, with the sequence tending to stabilize towards a plateau. After optimizing the sequential mPIDP to determine the K_p and K_d outputs from the dataset, the resulting sequences can be analyzed and compared. An example is shown in Fig. 8, the figure 11 shows the beginning and the end of the scenario corresponding to the graph of figure 12. The L-BFGS optimization process produces a sequence that closely matches the real one.

After training, 2 decision surfaces are generated (cf. Fig. 9), sharing the same overall shape as those from the multi-criteria optimization method (equation 7) but noticeably different (cf. Fig. 10). The most influential parameter is t_{SNR}, with both K_p and K_d increasing as $|e_{PIDP}|$ increases and t_{SNR} decreases, reflecting higher reactivity with increasing danger. Notably, the influence of e_{PIDP} on K_d is negligible compared to the multi-criteria approach.

Finally, to validate the bio-inspired MRAM-CS architecture, several scenarios have been processed on the simulation tool. The longitudinal distance between the IV and the driver-controlled vehicle is visualized in the same Frenet frame, eliminating the lateral component. Note that d_{safety} is not

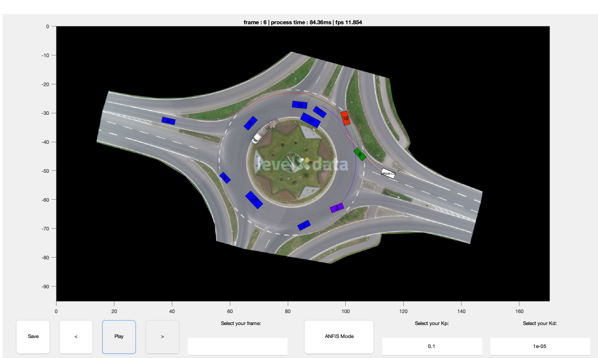


Fig. 6: The developed simulation tool.

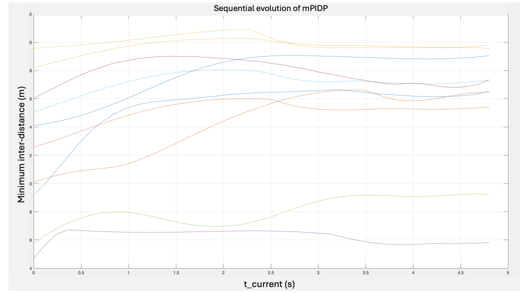


Fig. 7: A sample of 10 mPIPD sequences used for the training of the adaptive FIS PD controller.

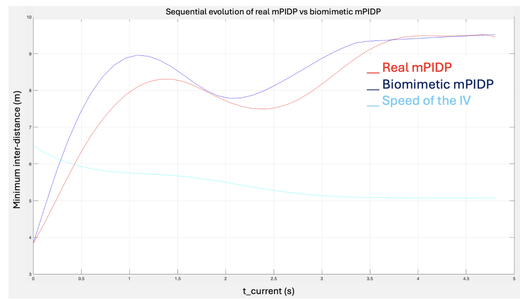
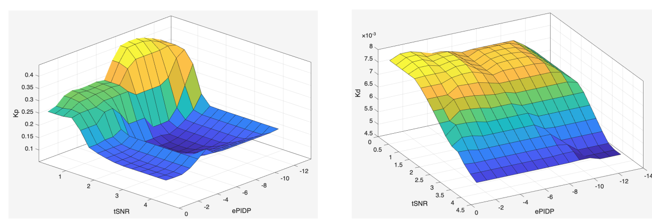
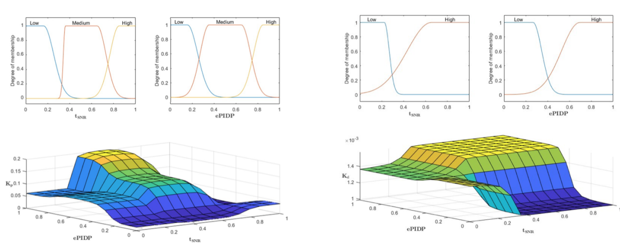


Fig. 8: Actual compared to learned mPIDP with biomimetic approach (blue: biomimetic, red: actual) for IV id 43 interacting with obstacle vehicle id 42, starting at frame 1673 of the rounD dataset (training scenario).



(a) K_p (b) K_d

Fig. 9: Decision surfaces of the adaptive FIS parameters controller generated with the bio-inspired approach. e_{PIDP} is in [m] meters and t_{SNR} is in [s].

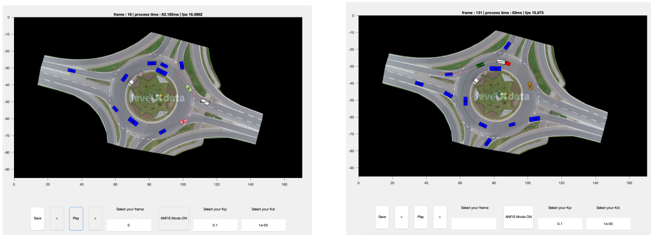


(a) K_p (b) K_d

Fig. 10: Decision surfaces of the adaptive FIS parameters controller with multi-criteria optimization approach [1].

learned from drivers, which explains the final distance offset. As shown in Fig. 12 and Fig. 11, the distance starts at 0, increases until reaching a plateau corresponding to the difference between the MRAM-CS d_{safety} and the driver

d_{safety}, demonstrating that the IV insertion is correctly managed.



(a) beginning of the scenario (b) end of the scenario

Fig. 11: IV insertion scenario: MRAM-CS controlled vehicle (white), driver-controlled vehicle (red), interacting obstacle vehicles (non-blue). Video of this scenario with the measure below: https://shorturl.at/A5FuI

.

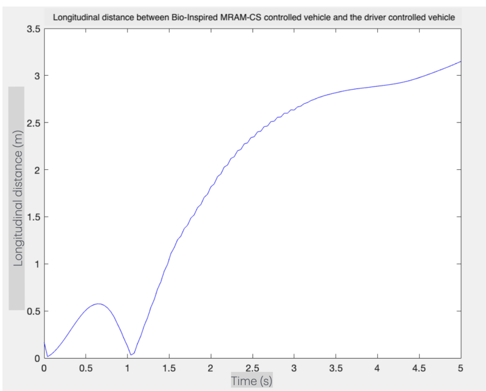


Fig. 12: Distance between the IV with the MRAM-CS architecture and the same vehicle controlled by a real driver.

V. Conclusion and Prospects

In this paper, a biomimetic approach is proposed to enhance the model-based decision-making architecture introduced in [1] and [2]. In addition, a dedicated simulation tool was developed to validate the bio-inspired architecture, based on several data of tight interactive roundabout situations. An appropriate approach to capture the decision-making process of a driver based on a bird-view representation is developed. The improvement of the MRAM-CS architecture was proposed. A machine learning approach was implemented to learn the parameters governing the IV speed profile control. The algorithm was trained and tested on the rounD dataset with promising results. Future works aim to implement the improved bio-inspired architecture on real vehicle of the Heudiasyc platform and the learning of the d_{safety} parameter.

Acknowledgment

Special thanks to Rhandy Cardenas, PhD Student at Renault Group and Heudiasyc laboratory, for his help in refining this article. Thanks also for the help of Kevin Bellingard in all his explanations on the MRAM-CS architecture.

References

[1] K. Bellingard, L. Adouane, and F. Peyrin, "Adaptive and Reliable Multi-Risk Assessment and Management Control Strategy for Autonomous Navigation in Dense Roundabouts," *IFAC*, vol. 56, no. 2, pp. 1294–1300, Jan. 2023.

[2] ——, "Safe and Adaptive Roundabout Insertion for Autonomous Vehicle based Limit-cycle and Predicted Inter-Distance Profiles," in *ECC*, Jun. 2023, pp. 1–7.

[3] A. Deluka Tibljaš *et al.*, "Introduction of Autonomous Vehicles: Roundabouts Design and Safety Performance Evaluation," *Sustainability*, vol. 10, no. 4, p. 1060, Apr. 2018, number: 4.

[4] A. Tampuu *et al.*, "A survey of end-to-end driving: Architectures and training methods," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 4, pp. 1364–1384, 2022.

[5] R. Cardenas *et al.*, "Context-aware and reliable long-term decision-making for safe intelligent vehicles: A survey," *IEEE Transactions on Intelligent Vehicles*, vol. 10, no. 12, pp. 5288–5307, 2025.

[6] M. A. H. Ali *et al.*, "Autonomous Road Roundabout Detection and Navigation System for Smart Vehicles and Cities Using Laser Simulator–Fuzzy Logic Algorithms and Sensor Fusion," *Sensors*, vol. 20, no. 13, p. 3694, Jan. 2020, number: 13.

[7] D. Tollner, H. Cao, and M. Zöldy, "Artificial Intellgence based Decision Making of Autonomous Vehicles Before Entering Roundabout," in *2019 IEEE CINTI-MACRo'19*, Nov. 2019, pp. 000 181–000 186.

[8] M. Liu *et al.*, "A Three-Level Game-Theoretic Decision-Making Framework for Autonomous Vehicles," *IEEE ITS*, vol. 23, no. 11, pp. 20 298–20 308, Nov. 2022.

[9] P. De Beaucorps *et al.*, "Decision-making for automated vehicles at intersections adapting human-like behavior," in *2017 IEEE IV*. Los Angeles, CA, USA: IEEE, Jun. 2017, pp. 212–217.

[10] K. Bellingard, L. Adouane, and F. Peyrin, "Risk Assessment and Management based on Neuro-Fuzzy System for Safe and Flexible Navigation in Unsignalized Intersection," in *2023 IEEE IV*, Jun. 2023, pp. 1–7.

[11] D. Iberraken, L. Adouane, and D. Denis, "Multi-Level Bayesian Decision-Making for Safe and Flexible Autonomous Navigation in Highway Environment," in *2018 IEEE IROS*, Oct. 2018, pp. 3984–3990.

[12] Z. Zhao *et al.*, "Real-time learning of driving gap preference for personalized adaptive cruise control," in *2023 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2023, pp. 4675–4682.

[13] R. Krajewski *et al.*, "The rounD Dataset: A Drone Dataset of Road User Trajectories at Roundabouts in Germany," in *2020 IEEE ITSC'23*, Sep. 2020, pp. 1–6.

[14] Z. Zhang, "Exploring rounD Dataset for Domain Generalization in Autonomous Vehicle Trajectory Prediction," *Sensors*, vol. 24, no. 23, p. 7538, Jan. 2024, number: 23.

[15] L. Adouane, "Reactive versus cognitive vehicle navigation based on optimal local and global PELC," *Robotics and Autonomous Systems*, vol. 88, pp. 51–70, Feb. 2017.

[16] ——, "Toward fully autonomous vehicle navigation: From behavioral to hybrid multi-controller architectures," in *2017 11th International Workshop on Robot Motion and Control (RoMoCo)*. IEEE, 2017, pp. 85–98.

[17] M. Green, "How Long Does It Take to Stop? Methodological Analysis of Driver Perception-Brake Times," *Transportation Human Factors*, vol. 2, no. 3, 2000.

[18] F. Diehl *et al.*, "Graph Neural Networks for Modelling Traffic Participant Interaction," in *2019 IEEE IV*, Jun. 2019, pp. 695–701.

[19] B. Wilson *et al.*, "Argoverse 2: Next Generation Datasets for Self-Driving Perception and Forecasting," Jan. 2023, arXiv:2301.00493 [cs].

[20] S. Ettinger *et al.*, "Large Scale Interactive Motion Forecasting for Autonomous Driving : The Waymo Open Motion Dataset," Apr. 2021, arXiv:2104.10133 [cs].

[21] W. Zhan *et al.*, "INTERACTION Dataset: An INTERnational, Adversarial and Cooperative moTION Dataset in Interactive Driving Scenarios with Semantic Maps," Sep. 2019, arXiv:1910.03088 [cs].

[22] D. C. Liu and J. Nocedal, "On the limited memory BFGS method for large scale optimization," *Mathematical programming*, 1989.

[23] T. J. Ross, "Properties of membership functions, fuzzification, and defuzzification," in *Fuzzy Logic with Engineering Applications*. John Wiley Sons, Ltd, 2010, ch. 4, pp. 89–116.