



**HAL**  
open science

# General and Multi-Criteria Approach to Study the Admissibility and Quality of a Driving Intention

Hugo Pousseur, Alessandro Correa Victorino

► **To cite this version:**

Hugo Pousseur, Alessandro Correa Victorino. General and Multi-Criteria Approach to Study the Admissibility and Quality of a Driving Intention. 35th IEEE Intelligent Vehicles Symposium (IV 2023), Jun 2023, Anchorage, United States. pp.1-7, 10.1109/IV55152.2023.10186528 . hal-04390002

**HAL Id: hal-04390002**

**<https://hal.science/hal-04390002>**

Submitted on 12 Jan 2024

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# General and Multi-Criteria Approach to Study the Admissibility and Quality of a Driving Intention

Hugo POUSSEUR<sup>1</sup> and Alessandro CORREA VICTORINO<sup>2</sup>

**Abstract**—Determining the admissibility and quality of driving intentions, generated by an automated intelligent vehicle and by a human driver, is a sine qua none task in a human-intelligent vehicle share navigation. Our paper proposes a generic method to quantify driving intentions. These intentions are defined by a sequence of velocities. This formulation is based on metrics already discussed in the literature. It proposes a way to use them to evaluate a state, making a judgment, and to extend this evaluation to a sequence of states. This quantification determines whether the intention is safely achievable and defines a quality taking into account several criteria (safety, comfort, context, energy consumption). It is thus possible to compare and rank the intentions according to a set of criteria. This article defines a proposed implementation that has been tested on a driving simulator. For a given scenario, we tested our solution on several intentions in order to show the interest of our solution, and the possibility to compare the intentions between them.

## I. INTRODUCTION

With the evolution of autonomous cars, quantifying and qualifying a system's commands has become an increasingly important task. This task allows either to evaluate a system or to anticipate a problem. It makes it possible to quantify and evaluate an order according to a given situation. This quantification is based on a set of metrics [1] representing a situation in a numerical way.

Our work [2] on shared navigation between the human and the autonomous system motivates us to make this quantification. In fact, in this shared task, the human and the autonomous car drive at the same time and a fusion is performed on these commands. This fusion is based in a first step on the analysis of their intention [3], which means what the entity wants to do. We define the intention [4] of an entity as follows:

$$C_{t,t+m} = \{(v_t, w_t), \dots, (v_{t+m}, w_{t+m})\} \quad (1)$$

This fusion needs to know the "best" command. The idea is to influence the command fusion towards the entity with the best proposal. This notion of "best" is not easy to define because it is a subjective notion that depends on the point of view, the criteria, and the way the intention is evaluated. In our paper, we propose a generic method for estimating the best intention, while ensuring that this intention is feasible. Our paper introduces formulations as well as concepts to achieve this quantification. We introduce two notions:

Heuristics and Diagnosis of Complex Systems, CS60319 - 60203 Compiègne Cedex, FRANCE. Cofunded by the European Commission under the H2020 Grant agreement OWheel No. 872907 <sup>1</sup> hugo.pousseur@hds.utc.fr, <sup>2</sup>alessandro.victorino@hds.utc.fr

- **Admissibility:** Defines, according to some criteria, whether a command is feasible or not;
- **Quality:** Defines the quality of a command, between 0 and 1, according to some criteria;

Each of these criteria belongs to a category (e.g. safety, comfort); Thus, the notion of best does not only depend on a notion of safety, but also includes other notions such as comfort, context, energy consumption. From these two concepts, we can then define a score for each intention. This score reflects both the feasibility of the intention and its quality, that will qualify the navigation intention.

Generic approach proposal, section II by defining concepts and formulation. Section III defines an implementation of the generic approach. Approach is implemented in a driving simulator in section IV. Section V discusses results.

## II. GENERIC APPROACH

In this article we define the sequence  $X$  as follows:  $X_{t,t+l} = \{x_t, \dots, x_{t+l}\}$ , where  $x_i$  represents a scalar.

### A. Definitions: Intention, command, intention quality

As defined by the Eq. 1, an intention is defined by a sequence of commands. The evaluation of the sequence of commands is defined by the evaluation of the states resulting from the application of these commands Eq. 2. By knowing the sequence of commands, i.e. the intention,  $C_t$ , and the current state,  $s_t$ , provided by sensors, the sequence of future states can be deduced.

$$(C_{t,t+m}, s_t) \xrightarrow{P} \{\hat{s}_{t+1}, \dots, \hat{s}_{t+m}\} \quad (2)$$

where  $s_i$  is the previewed state after the application of the control input  $(v_i, w_i)$ . Each state in this sequence can be assessed and quantified. The quantization must be more sensitive to a near state than to a far state. Let the quality of a state,  $s_i$ , noted as *quality*( $s_i$ ), the quality of a control sequence,  $C_{t,t+m}$ , on the current state,  $s_t$ :

$$Quality(C_{t,t+m}, s_t) = \frac{1}{\sum_i^m \gamma(i)} \sum_{i=1}^m \gamma(i) * quality(s_{t+i}) \quad (3)$$

With  $\gamma : \mathcal{N}^+ \rightarrow [0, 1]$ , a decreasing function called discount factor. This means that the quality of a late state has less impact on the final result than a near state. The Fig. 1 shows the relation between discount function and states. This approach is similar to the discount rate used in reinforcement learning [5].

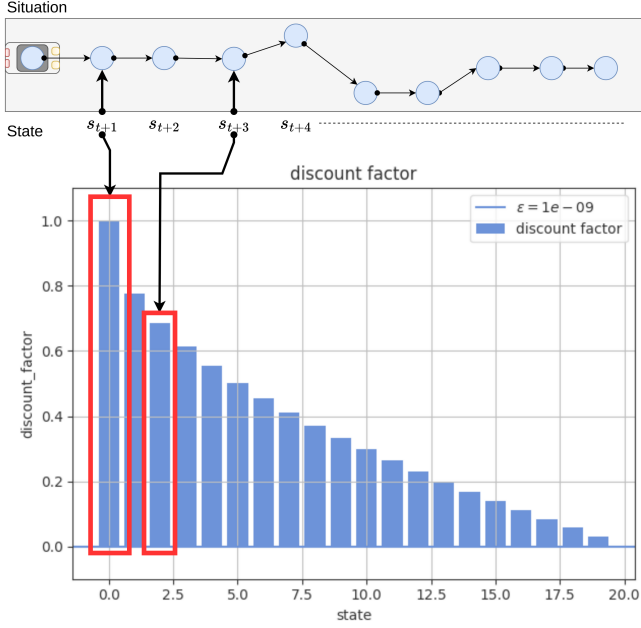


Fig. 1: Discount factor graphic.

## B. Evaluator

In order to qualify the commands sequence, we need to define how to qualify each state. As mentioned in the section I, this quantification is done according to several criteria. Our solution is to define an evaluator,  $eval_k$ , for each criterion  $k$ . An evaluator is defined as a composition of two functions: a metric function,  $metric_k(s_t)$ , whose goal is to quantify a situation, which we can compare to a feature extraction function. The result of this function is analyzed by an analyzer function whose goal is to judge this result from a knowledge, defined by  $\Theta$ ,  $analyzer_{k|\Theta}(metric_k(s_t), s_t)$ .

$$eval_k(s_t) = analyzer_{k|\Theta}(metric_k(s_t), s_t) \quad (4)$$

For each of these criteria, we can vary its influence on the final result by assigning a weight  $w_k$ . Thus, the quality of a state is defined by

$$quality(s_t) = \sum_k w_k * eval_k(s_t) \quad (5)$$

We need to standardize each quality between  $[0, 1]$ :

$$quality(s_t) = \frac{1}{\sum_k w_k} \sum_k w_k * eval_k(s_t) \quad (6)$$

## C. Quality Generic Formulation

Taking Eq. 3 in Eq. 6:

$$Quality(C_t, s_t) = \frac{1}{\sum_i \gamma(i)} \sum_{i=1}^n \gamma(i) * \left( \frac{1}{\sum_k w_k} \sum_k w_k * \underbrace{analyzer_{k|\Theta}(metric_k(s_{t+i}), s_{t+i})}_{eval_k} \right)_{quality_{t+i}} \quad (7)$$

## D. Admissibility Generic Formulation

Similarly to quality, admissibility is defined by a set of  $guard_l$  that guarantee for each criterion a check that the command is indeed admissible. Each guard is composed as follows:

$$guard(s_i) = indicator_{\Theta}(metric(s_i), s_i) \quad (8)$$

Where the function  $indicator$  indicates whether the  $metric$  of a state is admissible ( $= 1$ ) or not ( $= 0$ ). Thus, the admissibility of a sequence of commands  $C_t$ , from a state  $s_t$ , contains  $n$  criteria on a sequence of  $\tau$ , unlike the quality, the elements are multiplied together so that if one element is inadmissible, the whole sequence is inadmissible.

$$Admissibility(C_t, s_t) = \prod_{i=1}^{\tau} \left( \prod_{l=1}^n guard_l(s_{t+i}) \right)_{admissibility_{t+i}} \quad (9)$$

$$= \prod_{i=1}^{\tau} \left( \prod_{l=1}^n indicator_{l|\Theta}(metric_l(s_{t+i}), s_{t+i}) \right) \quad (10)$$

## E. Intention Score Generic Formulation

From the generic formulation of quality and admissibility, the generic formulation of score, is defined as follows. In this proposal, we add a hyperparameter  $\epsilon$ , which allows to modify the domain of definition of the quality  $[0, 1] \rightarrow [\epsilon, 1]$ . Thus, we can define the value of an intention as follows:

$$Score(C_t, s_{qst}) = Admissibility(C_t, s_t) \times (\epsilon + Quality(C_t, s_t) * (1 - \epsilon)) \quad (11)$$

So we distinguish between a good quality order that is not feasible and a low quality order that is feasible, the Fig. 2 shows the interpretation of the score. The  $\epsilon$  is a very small positive quantity used to ensure that a quality of zero does not result in a score of zero. In the rest of our work, this difference is important, because it allows us to distinguish a valid intention, but with a poor quality, from an invalid intention.

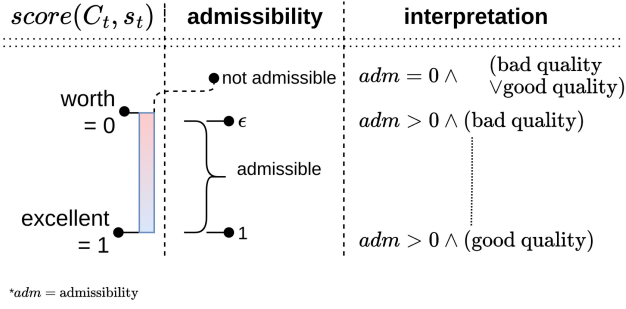


Fig. 2: Score interpretation.

### III. IMPLEMENTATION

#### A. State Definitions

We divide the current state  $s_t$ , into 2 substates,  $s_t^{env}$  describing the current state of the environment and  $s_t^{vehicle}$  describing the current state of the vehicle.

$$s_t = \{s_t^{env}, s_t^{vehicle}\} \quad (12)$$

1) *State Vehicle*: The state of a vehicle is defined by the dynamic state of the vehicle. So we define the state of a vehicle as follows:

$$s_t^{vehicle} = \begin{bmatrix} v \\ w \\ \dot{v} \\ \dot{w} \\ j \end{bmatrix}_t \quad \begin{array}{l} \text{(linear velocity)} \\ \text{(angular velocity)} \\ \text{(linear acc)} \\ \text{(angular acc)} \\ \text{(jerk)} \end{array} \quad (13)$$

From the state definition, the next state defined by  $\hat{s}_{t+1}^{vehicle}$ , can be estimated by the following relation, including  $\delta t$  time the interval time:

$$\hat{s}_{t+1}^{vehicle} = \frac{1}{\delta t} (A s_t^{vehicle} + B c_t) \quad (14)$$

Where  $A \in \mathcal{M}_{5 \times 5}$  and  $B \in \mathcal{M}_{2 \times 5}$ , and  $c_t$  the command applied on the state  $s_t^{vehicle}$ .

2) *State Environment*: The state of the environment is defined by an occupancy grid. As explained in the previous sections, we need to predict the future state, i.e. the future occupancy grid in this case. At first, we assume the environment of the car as static, so making a prediction of the future grid is like making a projection of the speed in trajectory and repositioning the position of the car on the occupancy grid.

$$s_t^{env} = [grid_t^{occ}, (x, y, \theta)_t^g] \quad (15)$$

The vector  $(x, y, \theta)_t^g$  defines the position and the orientation of the vehicle on the occupation grid. We assume this occupancy grid is static, from the commands sequence combined to interval time  $\delta t$ , we can estimate the new position of the vehicle in the initial occupancy grid ( $grid_t^{occ}$ ). The vehicle position can be estimated.

$$\begin{bmatrix} x \\ y \\ \theta \end{bmatrix}_t \xrightarrow[\delta t]{(v_t, w_t)} \begin{bmatrix} x \\ y \\ \theta \end{bmatrix}_{t+1} \quad (16)$$

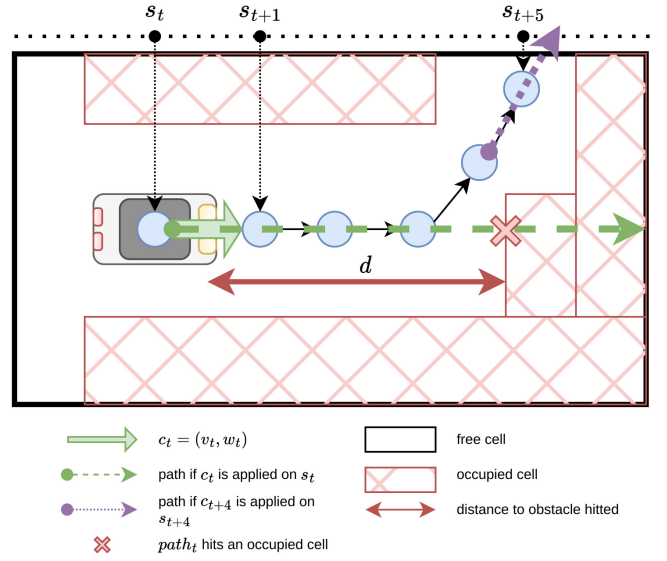


Fig. 3: Determine the distance to the obstacle by applying the command,  $c_t$ , as a constant to the current state,  $s_t$ .

Thus we define the prediction of the next state,  $\hat{s}_{t+1}^{env}$ , as follows:

$$\hat{s}_{t+1}^{env} = [grid_t^{occ}, (x, y, \theta)_{t+1}^g] \quad (17)$$

#### B. Quality

As previously defined, we have to define the quality and admissibility functions by criteria. In this article, we defined 4 quality criteria divided into 3 categories:

- safety: collision on path evaluator;
- context: speed limit evaluator;
- comfort: lateral acceleration evaluator, jerk evaluator;

As previously defined, the quality is defined on a sequence of commands, to realize our tests we applied the calculation of the quality on the first 20 elements,  $m = 20$ , indeed our data are recorded at 10hz, so our sequence is analyzed on 2 seconds. In previous work, the prediction of commands were good up to 2 seconds [3], explaining the choice of this value.

1) *Collision On Path (COP)*: This evaluator evaluates if the desired drive does not collide with an obstacle, which is in its way.

a) *Metric*: This metric defines for a state  $s_t$  the distance  $d_t$  between the vehicle and the nearby obstacle, considering that the vehicle runs at the constant speed defined by  $c_t$ . This distance is computed by projecting the trajectory of the vehicle, running at speed  $c_t$ , on the occupation grid  $grid_t^{occ}$ . Figure 3 shows how to define the obstacle distance by projecting the path estimated from command.

$$metric_{COP}(s_t) = distance_{obstacle} \quad (18)$$

b) *Analyzer*: In order to evaluate the distance to obstacle, a reference distance need to be defined. Based on the reaction time,  $s_{react} = 1.0s$  [6], and on the maximum deceleration,  $dec_{max} = 3.3m/s^2$  [6] and the current velocity,

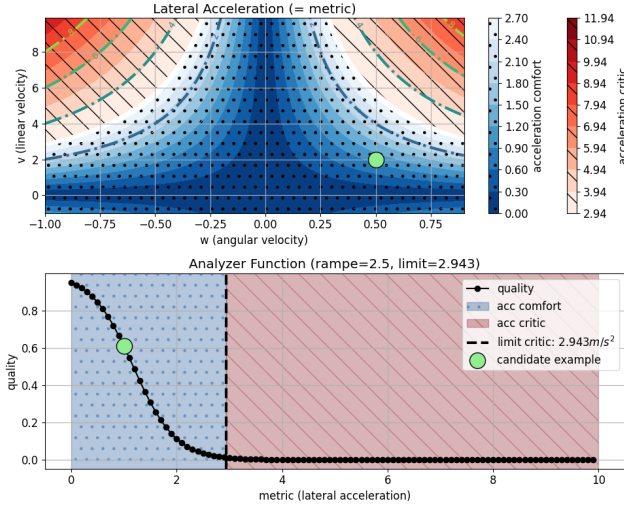


Fig. 4: Left figure represents a metric from vehicle state, based on this metric the right function analyzes the result and appropriate a judgement between 0 and 1. This analyzer is defined by 2 parameters: *rampe*, *shift*.

$v_t \in s_t$ , the reference distance can be defined as:

$$\begin{aligned} s_{safe} &= \frac{v_t}{dec_{max}} \\ d_{safe}(v_t) &= v_t * (s_{safe} + s_{react}) \\ &\quad - dec_{max} * \frac{s_{safe} * (s_{safe} + 1)}{2} \\ d(s_t) &= d_{safe}(v_t) \end{aligned}$$

Based on this estimation, the analyzer function is build in taking account this reference:

$$\begin{aligned} analyzer_{COP|\Theta}(metric_{COP}(s_t), s_t) \\ = \frac{1}{1 + e^{(-\theta_{rampe} * metric_{COP}(s_t) - d(s_t))}} \end{aligned} \quad (19)$$

Where  $\theta_{rampe}$  is a hyperparameter impacting the slope degrees.

2) *Lateral Acceleration (LA)*: This evaluator takes into account the comfort of driving. Figure 4, illustrates the metric and the analyzer used by this function.

a) *Metric*: From the linear velocity and the angular velocity, the lateral acceleration of the vehicle can be deduced by:

$$metric_{LA}(s_t) = v_t * w_t \quad (20)$$

In order to understand the relation between metric and analyzer, we defined a candidate example (refer to Fig. 4) defined by  $(w_t = 0.5, v_t = 2)$  then lateral acceleration  $a_t = (v_t * w_t) = 1.0m/s^2$ , based on this metric the result of the analyzer is 0.61.

b) *Analyzer*: In order to compare the lateral acceleration defined by the previous metric, a reference value need to be set, let the most acceptable acceleration,  $a_{ref} = 0.3g = 2.943m/s^2$  [7].

$$\begin{aligned} analyzer_{LA|\Theta}(metric_{LA}(s_t), s_t) \\ = \frac{1}{1 + e^{(metric_{LA}(s_t) * \theta_{rampe} - a_{ref})}} \end{aligned} \quad (21)$$

3) *Speed Limit (SL)*: Depending on the context in which the vehicle is operating, this evaluator assesses compliance with the speed limit.

a) *Metric*: This metric exploits the current speed of the vehicle.

$$metric_{SL}(s_t) = |v_t| \quad (22)$$

b) *Analyzer*: The speed of the vehicle should be closed to the  $\theta_{target}$  limit speed. The analyzer exploits hyperparameters  $\theta_{target}$  the limit speed and  $\theta_{\sigma}$  the tolerance. This analyzer is a normalized Gaussian, centered on the speed limit, defined by the context.

$$analyzer_{SL|\Theta}(metric_{SL}(s_t), s_t) = e^{(\frac{1}{2} * (\frac{metric_{SL}(s_t) - \theta_{target}}{\theta_{\sigma}})^2)} \quad (23)$$

4) *Jerk (J)*: The jerk represents the variation of the acceleration, this data illustrates abrupt changes of gait. We use this information to determine if the ride is comfortable or not.

a) *Metric*: Our analysis requires using only the current jerk information of the vehicle state.

$$metric_J(s_t) = |j_t| \quad (24)$$

b) *Analyzer*: The analyzer compares the actual jerk to a reference,  $\theta_{ref} = 9.9m/s^3$  [8]. This reference value makes it possible to define from which value the driving is uncomfortable.

$$\begin{aligned} analyzer_{J|\Theta}(metric_J(s_t), s_t) \\ = \frac{1}{1 + e^{(metric_J(s_t) * \theta_{rampe} - \theta_{ref})}} \end{aligned} \quad (25)$$

### C. Admissibility

Similar to the evaluators, for the admissibility part we have to define the guards composed of metric and indicator. Like for the quality, we need to define the sequence time analyzed by the guard, we choose to use 10 first elements, the sequence is recorded at 10hz, then the sequence analyzed is 1.0 second refer to the time reaction of the human [6].

#### 1) TTC On Path (TOP):

a) *Metric*: This indicator exploits the metric  $metric_{COP}$ , defined above (III-B.1.a), and deduces from the current speed the time to collision.

$$metric_{TOP}(s_t) = \frac{metric_{COP}(s_t)}{|v_t| + \epsilon} \quad (26)$$

Where  $\epsilon$  is a small quantity avoiding dividing by 0, when the linear velocity is 0.

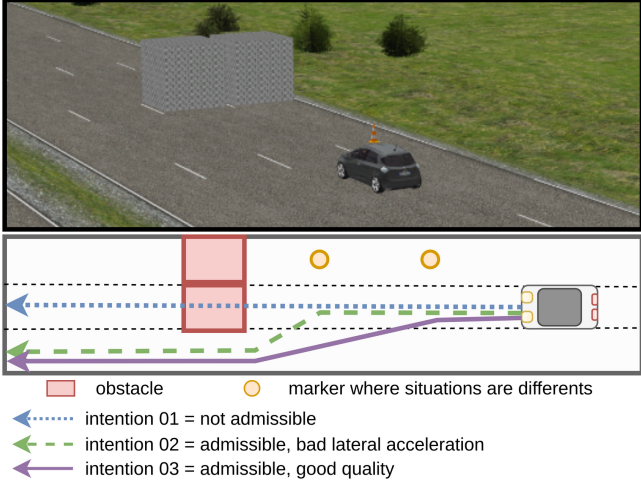


Fig. 5: Scenario support and intentions tested.

*b) Indicator:* From a reference time,  $\theta_{critic.time}$ , we can determine the admissible or not admissible times and thus define according to this criterion the admissible intentions.

$$indicator_{TOP|\Theta}(s_t) = metric_{TOP}(s_t) > \theta_{critic.time} \quad (27)$$

The  $\theta_{critic.time}$  can be adapted the context [7]. In our implementation, the  $\theta_{critic.time}$  is fixed at 1.0 seconds.

#### 2) Collision Around (CA):

*a) Metric:* For a given state  $s_t$  this metric defined the closest obstacle to the vehicle position, around the vehicle. This metric exploits the occupancy grid and find the closest obstacle to the vehicle.

$$metric_{CA}(s_t) = distance_{obstacle} \quad (28)$$

*b) Indicator:* Similar to the previous indicator, from a reference distance,  $\theta_{critic.distance}$ , we can filter admissible intentions.

$$indicator_{CA|\Theta}(s_t) = metric_{CA}(s_t) > \theta_{critic.distance} \quad (29)$$

## IV. RESULTS

The implementation defined in the previous section was written in Python. Our solution has been tested in the SCANeR studio simulator, which allows us to realize not admissible intention data. We tested our solution on the same scenario but with 3 different intentions to show the efficiency of our solution to compare 3 intentions, as presented in Fig. 5. In this scenario, the vehicle is driving on 3 lanes, obstacles are placed on 3 lanes. The first intention drives and hits the obstacle, the second avoids the obstacles with a high lateral acceleration and the last avoids the obstacle but with a low lateral acceleration.

evaluator	weight
collision on path	0.4
speed limit	0.2
lateral acceleration	0.2
jerk	0.2

TABLE I: Evaluator weights table.

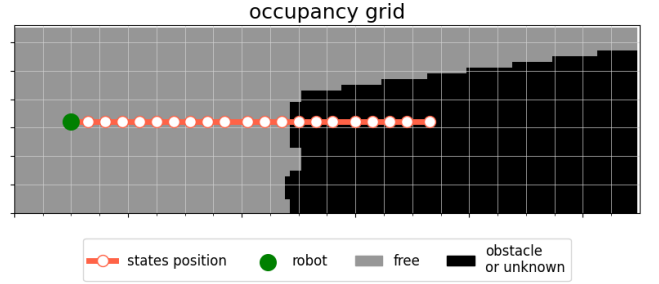


Fig. 6: Occupancy grid with intention trajectory (intention 1).

#### A. Intention 1: Hit Obstacle

In the first intention, the vehicle is moving and hits the obstacle. Figure 6 shows the projection predicted states of the intention on the current occupation grid. In the case where an order is admissible, we can directly conclude that the score of this intention is 0. So we must first make sure that it is valid. In this situation a guard is not validate, that shows the Fig. 7, because the vehicle is too close of the obstacle and doesn't have the time to decelerate before hitting the obstacle. So in this situation the final quality and score are 0.

#### B. Intention 2: Avoid Obstacle With High Lateral Acceleration

In the first intention the driver avoids the obstacle but makes an abrupt lane change. Like the previous intention, the Fig. 8 shows the occupancy grid combined with the trajectory of the vehicle estimated from the intention. The following

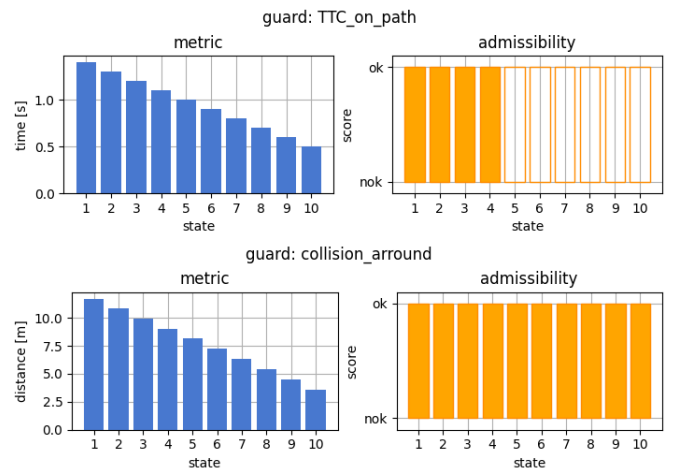


Fig. 7: Guard results (intention 1).

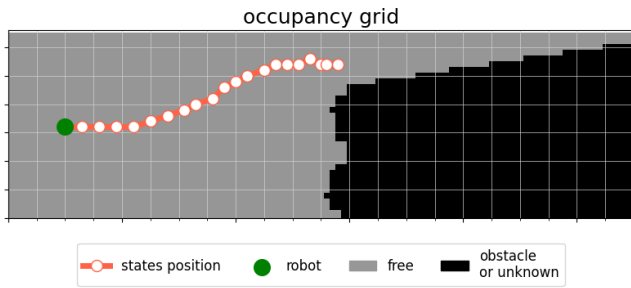


Fig. 8: Occupancy grid with intention trajectory (intention 2).

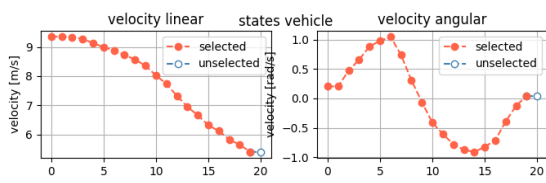


Fig. 9: Vehicle velocities (intention 2).

Fig. 9 shows the speeds of the vehicle in future states. In this case, compared to the previous one, all the guards are good, which means the intention is admissible.

The Fig. 12 shows the result per evaluator. Unlike the lateral acceleration, the other evaluations are goods. We can see that poor acceleration affects the final quality. The final score is equal to 0.746.

### C. Intention 3: Avoid Obstacle With Good Quality

In this case, the intention must be good, instead of intention 02, the lane change is made earlier than the previous one. Then, the steering wheel angle is small, which does not result in a high angular velocity and therefore the resulting lateral acceleration is not very important. Figures 13, 14 show the states of the intention 03. In this case the admissibility is validated, like Fig. 15. The following Fig. 16 shows the quality per criterion and the global quality. The global quality

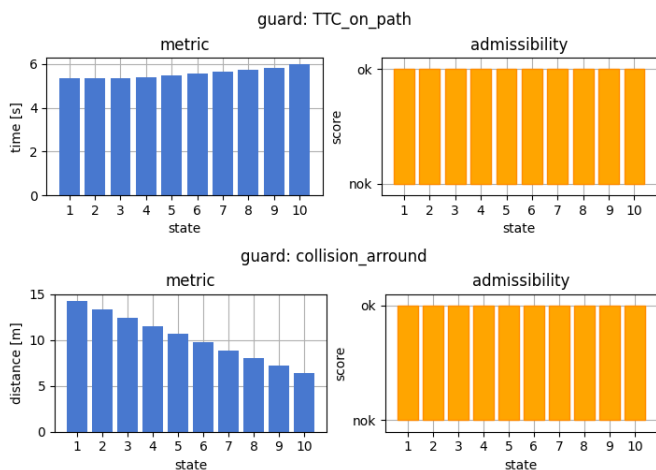


Fig. 10: Guard results (intention 2).

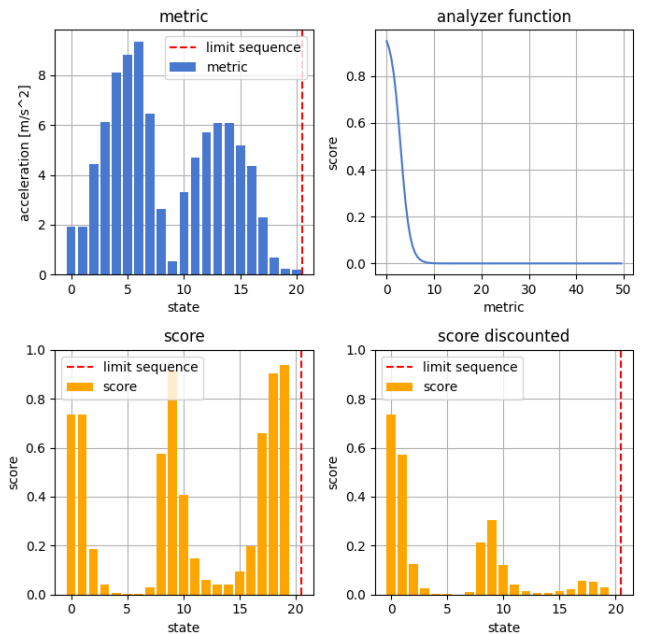


Fig. 11: Vehicle velocities (intention 2).

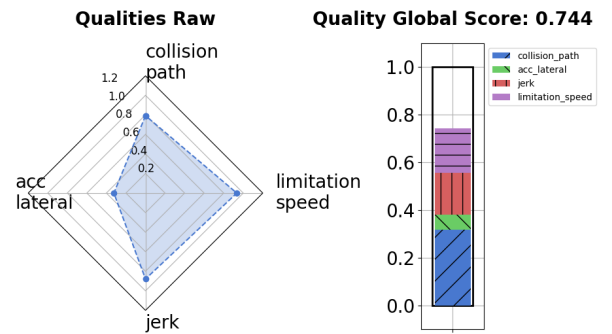


Fig. 12: Quality results (intention 2).

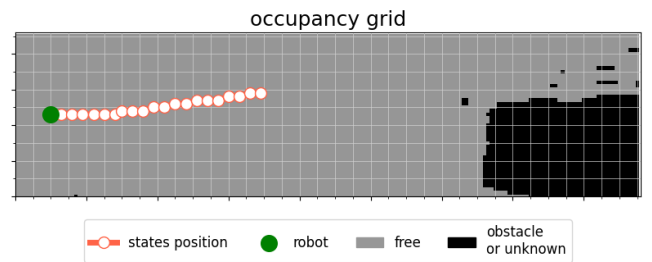


Fig. 13: Occupancy grid with intention trajectory (intention 3).

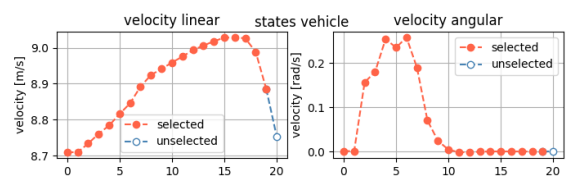


Fig. 14: Vehicle velocities (intention 3).

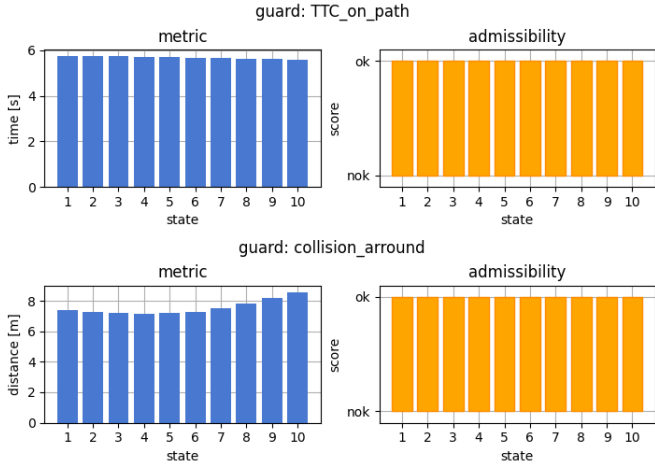


Fig. 15: Guard results (intention 3).

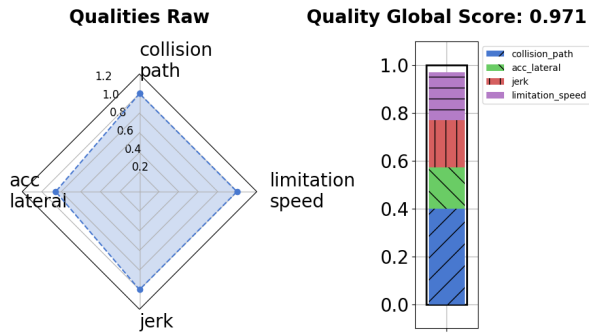


Fig. 16: Quality results (intention 3).

is close to 1 (0.971), which means perfect quality. The intention score is 0.971.

#### D. Comparison

From previous studies, we can summarise and compare the quantification of each intention with Tab. II.

### V. DISCUSS

These evaluations of these different intentions highlight the interest of our formulation and its effectiveness.

The first test (Sec. IV-A) highlights the need to define admissibility, as it allows an intention that involves a risk for the users of the vehicle to cancel the quality. This case highlights the importance of distinguishing between admissibility and quality.

The second test (Sec. IV-B) illustrates an admissible case where one of the criteria is not met. Contrary to the previous case, this command does not represent a danger but an inconvenience, this command is possible. The quality according to

intention	admissibility	quality	score
intention 1	0	-	0
intention 2	1	0,744	0,746
intention 3	1	0,971	0,971

TABLE II: Summarize tests.

the lateral acceleration, illustrates the inconvenience of this intention.

The last test (Sec. IV-C) shows that our formulation can be used to highlight a good intention. In this case, the intention is risk-free and respects the other criteria, and this respect of the criteria is reflected in the quality of this intention, which is good (0.971).

With this quantification, we can rank these different intentions and choose the best one that is considered feasible and of better quality according to the criteria we have defined. Based on the Tab. II, its 3 intentions can be ranked as follows: the best intention is 'intention 3', followed by 'intention 2' and 'intention 1' is considered the worst, as it is not feasible.

It is important to note that our approach is to define analyzers that rely on knowledge. The definition of analyzers is not trivial and potentially involves a set of hyperparameters, which adds constraints to the implementation of this method.

### VI. CONCLUSION

In our approach, we set up a generic formulation that allows to assign a score to an intention that reflects both the feasibility (defined by the admissibility) and the quality of the intention. Our implementation and case study show that our formulation allows us to perform this quantification task and to rank multiple intentions for the same situation. Furthermore, our implementation highlights the importance of distinguishing between admissibility and quality.

In our next work, we intend to integrate this quantification of intentions in more complex situations, integrate new criteria such as the energy consumption of the car. We then plan to integrate this quantification into our fusion of the shared navigation between the human and the autonomous system. This quantification will make it possible to influence the resulting command towards the intention with a better score, thus guaranteeing better driving.

For now, we have applied our solution in a case where the environment is static to facilitate the prediction of the next state. A next approach, we will be able to perform state estimation in a dynamic environment using prediction models [9] of the surrounding cars.

### REFERENCES

- [1] S. S. Mahmud, L. Ferreira, M. S. Hoque, and A. Tavassoli, "Application of proximal surrogate indicators for safety evaluation: A review of recent developments and research needs," *IATSS research*, vol. 41, no. 4, pp. 153–163, 2017.
- [2] S. C. Jugade, A. C. Victorino, and V. Berge-Cherfaoui, "Shared driving control between human and autonomous driving system via conflict resolution using non-cooperative game theory," *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pp. 2141–2147, 2019.
- [3] H. Pousseur and A. C. Victorino, "Prediction of human driving behavior using deep learning: a recurrent learning structure," in *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*, pp. 647–653, IEEE, 2022.
- [4] S. Mozaffari, O. Y. Al-Jarrah, M. Dianati, P. Jennings, and A. Mouzakis, "Deep learning-based vehicle behavior prediction for autonomous driving applications: A review," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 1, pp. 33–47, 2020.
- [5] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement learning: A survey," *Journal of artificial intelligence research*, vol. 4, pp. 237–285, 1996.



- [6] N. Uno, Y. Iida, S. Itsubo, and S. Yasuhara, "A microscopic analysis of traffic conflict caused by lane-changing vehicle at weaving section," in *Proceedings of the 13th Mini-EURO Conference-Handling Uncertainty in the Analysis of Traffic and Transportation Systems, Bari, Italy*, pp. 10–13, 2002.
- [7] S. Azadi, R. Kazemi, and H. R. Nedamani, "Chapter 6 - dynamic behavior and stability of an articulated vehicle carrying fluid," in *Vehicle Dynamics and Control* (S. Azadi, R. Kazemi, and H. R. Nedamani, eds.), pp. 213–260, Elsevier, 2021.
- [8] O. Bagdadi and A. Várhelyi, "Jerky driving—an indicator of accident proneness?," *Accident Analysis & Prevention*, vol. 43, no. 4, pp. 1359–1363, 2011.
- [9] S. H. Park, B. Kim, C. M. Kang, C. C. Chung, and J. W. Choi, "Sequence-to-sequence prediction of vehicle trajectory via lstm encoder-decoder architecture," in *2018 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1672–1678, IEEE, 2018.