A Supervisor Agent-based on the Markovian Decision Process Framework to Optimize the Behavior of a Highly Automated System

A. Castellano[1], M. Karimshoushtari[2], C. Novara[2], F. Tango[3]

[1] RE:Lab, 42122 Reggio Emilia, Italy
[2] Politecnico di Torino, 10129 Torino, Italy
[3] Centro Ricerche Fiat, 10043 Orbassano, Italy
andrea.castellano@re-lab.it, milad.karimshoushtari@polito.it
carlo.novara@polito.it, fabio.tango@crf.it

Abstract. In this paper, we explore how MDP can be used as the framework to design and develop an Intelligent Decision Support System / Recommender System, in order to extend human perception and overcome human senses limitations (because covered by the ADS), by augmenting human cognition, emphasizing human judgement and intuition, as well as supporting him/her to take the proper decision in the right terms and time.

Moreover, we develop Human-Machine Interaction (HMI) strategies able to make "transparent" the decision-making / recommendation process. This is strongly needed, since the adoption of partial automated systems is not only connected to the effectiveness of the decision and control processes, but also relies on how these processes are communicated and "explained" to the human driver, in order to achieve his/her trust.

Keywords: Intelligent Decision Support System, Recommender System, Autonomous Driving, Markovian Decision Process.

1 Introduction

Autonomous Vehicles (AVs) arise as a technological solution to mitigate the shortcomings of manual driving: reduction of human-caused accidents and the realization of a more efficient driving task in terms of energy consumption, traffic flow and driver's workload. Under this perspective, AVs are expected to fundamentally change road transport and improve life quality. In fact, the automation of vehicles has been identified as one major enabler to master the Grand Societal Challenges "Individual Mobility" and "Energy Efficiency" and highly automated driving functions (ADF) are one major step to be taken.

However, this technology is not mature enough yet for massive implementation and, in addition, it can bring to specific side-effects. In particular, the automation of the dynamic driving task removes humans from the control loop, leaving to the driver the monitoring loop. If we consider the Skills, Rules, Knowledge framework of Rasmussen in manual operation of a vehicle [1], we can say that moving from the skill-based behavior to the rule-based behavior up to the knowledge-based behavior makes the workload and the probability of errors more likely to increase.

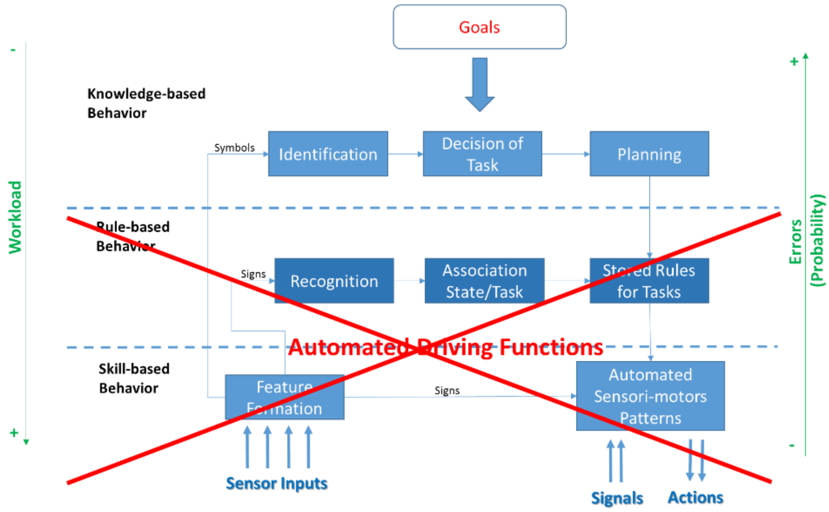

Fig. 1. SRK Framework, including the use of ADFs in the driving task.

This is exactly the risk of Automated Driving Systems (ADSs), where the first two lower levels are performed by the system, leaving the upper level (namely, knowledge-based behavior) to humans, indeed characterized by high workload and high probability of error. Under this perspective, there is also the risk that humans lose some skills, thus fundamental changes can occur to what humans are expected to learn.

Especially as machines acquire capabilities to learn deeply and actively from data [2], adaptation and personalization to human needs shall be considered. In this context, intelligent agents should be able to think and behave in ways that support humans, by providing personalized, adaptive, responsive and proactive services in a variety of settings and scenarios.

The European ECSEL research project PRYSTINE realizes Fail-operational Urban Surround perceptION (FUSION)1 based on robust Radar and LiDAR sensor fusion and control functions, in order to enable safe automated driving in urban and rural environments. With reference to the latest innovations of the PRYSTINE project, in this paper, we explore how an Intelligent Decision Support System (IDSS) can be designed and developed, in order to extend human perception and overcome human senses limitations (because covered by the ADS), by augmenting human cognition, emphasizing human judgement and intuition, as well as supporting him/her to take the proper decision in the right terms and time. A critical aspect needed to design adaptive systems is the decision-making task, which has to weight several possibly conflicting data sources in order to decide a safe driving plan. The theory of Markov Decision Processes (MDPs)

[1] For more information, see the website: https://prystine.eu/.

[3] provides the standard semantic foundation for a wide range of problems involving decision-making tasks. Indeed, MDP formalism allows a modeler to specify a stochastic decision process by means of a set of states S, in which a decision maker has to choose an action from a set of available actions Act. Then, the process randomly evolves according to a specified transition probability associated with the selected action, and it returns to the decision maker a reward depending on the chosen action and by the source and destination states.

Under this perspective, the supervisor agent based on MDP is a kind of recommendation system (RS), which aims at predicting if an item would be useful to a user based on given information (following the definition of [7] and [8]). In this sense, it can solve the information overload problem, by suggesting the proper action and personalizing the user experience, delivering accurate, personalized recommendations to users (i.e., drivers in our case), according to some criteria, such as safety and preferences. In fact, it can be challenging for a user to filter through all the available information and take away essential aspects information overload or, for a system, to decide about the optimal action to take, satisfying different and, sometime, contradictory criteria [9].

Moreover, we develop Human-Machine Interaction (HMI) strategies able to make "transparent" the aforementioned decision-making process. This is strongly needed, since the adoption of partial automated systems is not only connected to the effectiveness of the decision and control processes, but also relies on how these processes are communicated, and "explained" to the human driver, in order to achieve his/her trust. This is a crucial topic, since it is common opinion that the HMI has a crucial role in the adoption of partially and highly automated vehicles [10]. The main challenge related to this topic relies on the responsibility of the HMI as "enabler of the cooperation", i.e. on being the tool that allows the vehicle to explain its intentions and, at the same time, allows the driver to provide inputs and act as decision-maker in the driving process. Recent studies have shown the relevance of providing the correct type and amount of information and the impact of these design choices on the improvements of the decision-making capabilities [11]. At the same time, different experimental studies have demonstrated the relevance of the approach focus on increasing the transparency of the automation [12].

The HMI proposed in this paper will be a multimodal state-adaptive system, able to tailor the interaction modality according to the outcome of the intelligent decision maker and the cognitive (as well as behavioral) state of the driver. The proposed system will focus on the design and implementation of the perception-decision-action (plus interaction) cycle in common traffic situations that, even if representing most of the driving task, are currently less explored in industry and research.

2 The Supervisor Agent as an Intelligent Decision Support System

Consider now an ADS, which applies appropriate controls to the vehicle (both lateral and longitudinal) so that collisions may be avoided. When a collision is imminent, there is no doubt about what to do: the system has to react to an immediate danger (following

the “sensori-motor” level of Piaget [4] or the “Control level” in the hierarchical structure of Janssen and Michon [5]) by braking as hard as possible to bring the host-vehicle in a safety zone. This is exactly what nowadays ADAS applications do.
When there is a normal (i.e., safe) driving situation, it is less clear what the optimal actions are. For example, when approaching a slower car, should the host-vehicle follow the one ahead, or change lane for an overtaking maneuver? From one side, a car-following decision can be the “safest” solution, but on the other side it can make the trip longer and can waste time. Therefore, a decision system that supports the human driver and takes the optimal actions can really help. In particular, deciding the optimal action to perform, or what corrective controls to exercise in order to avoid a possible collision, is essentially a problem of “credit assignment”: supposing an outcome is a consequence of a sequence of decisions. In other words, the credit assignment problem calls for a system to associate decisions to their long-term outcomes. One of the most important theories for formulating and solving credit assignment in sequential decision-making problems is the aforementioned MDP theory [6]. In modelling a problem as an MDP, we contemplate a decision-maker who is required to take decisions over a sequence of discrete time periods.

2.1 Use-cases and scenarios of interest

In the PRYSTINE project, all the SW and HW components are implemented and integrated in some demonstrators. In particular, the IDSS described in this paper is included in one project application for the passenger car, employing PRYSTINE’s fail-operational autonomous driving functions (ADFs) and the related sensor data fusion (SDF) from a wide range of sensors (Radar, LiDAR, camera, V2X communication and feedback devices).

Fig. 2. Maserati prototype vehicle used by the PRYSTINE Project. The car is equipped with a range of cameras, radars, communication sensors and feedback devices, serving as a testbed for both level 2 and level 3 ADFs.

Current ADSs rely on SDF to identify the driving scenario in the vehicle proximity (i.e., in the field of view/range of sensors), possibly extended by information through V2X communication. Data from heterogeneous sources/sensors are fused to provide an overview of traffic in the surroundings of the vehicle. This information can be used to let the ADS anticipating the evolution of traffic, providing a more comfortable and efficient driving performance, especially in urban scenarios.

In the project, three use-cases (UCs) are developed: the "*Traffic Light Time-To-Green*", "*Trajectory Recognition and VRU*" and the "*Emergency Lateral Lane Stop*". For the application of our IDSS, based on MDP framework, we started from the third one, which is sketched in the following figure:

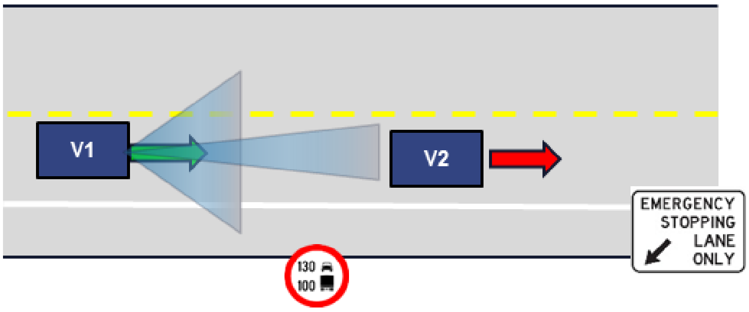

Fig. 3. Sketch of the UC3, named Emergency Lateral Lane Stop, specific for different types of scenarios.

In this scenario, the AV is travelling at a given speed, when it approaches a slower vehicle. In this case, the decision-maker has to define the next optimal action: is it better to follow the car ahead, or to overtake it? It is worth to noting here that the developed IDSS can either inform the driver about the best maneuver to do or intervene on the vehicle actuators to perform the same maneuver. Of course, the final decision depends on some factors, such as the safety of the action (e.g., if another vehicle is already overtaking the AV from the left adjacent lane, the overtaking is not considered or at least delayed), the optimization of the travelling time (i.e., maybe the car-following decision can minimize fuel consumption but make the travel too much longer) and even from the cognitive status of the driver (e.g., s/he is distracted or attentive).

The inputs of the system are related to the perception of the external environment, constituted by the Radar (front/blind spot), front camera, ultrasonic sensors and LiDAR. In addition, the system considers the use of a Driver Monitoring System (DMS), which detects the driver status to understand if s/he is still capable to control the vehicle, or alternatively, if s/he is able to get back into the control-loop in case of a "take over request" (TOR) from the system. If a critical case is detected, a safe-stop maneuver is necessary (e.g., the driver is impaired for drowsiness). In details, the DMS includes biometric devices and dynamic vehicle algorithm, to detect drowsiness, cognitive load and visual distraction.

The output is represented by the longitudinal/lateral controls of the vehicle, to avoid potential collision, to act an overtaking and, if necessary, to perform a safe stop maneuver (emergency lights activation and stop in the emergency lane, if possible).

As it is now, in the current ADAS / ADS applications (even at prototypical level), the choice between an overtaking and a car-following action can be conflictual and not smooth (if the "adaptive cruise control" and the "support to overtaking" functions do not communicate each other); moreover, if the driver is not responding (as aforementioned, due to drowsiness, for example) to a TOR after some time (because the system reached the limits of its Operational Design Domain, or ODD in short), the AV "simply" stops in the current driving lane. Thanks to the super-visor agent, which "knows" the situation, the most appropriated action is taken: with reference to the previous example, the system can decide to minimize the travelling time and, given that such a decision is safe and the driver is attentive, an overtaking maneuver is initiated. On the other way, if the driver is impaired, after checking that the emergency lane is present, a safe lane change is performed.

2.2 The system architecture and its main components

Following the "*Perception Cognition Action*" (PCA) framework, the following figure shows the overall system architecture and related components:

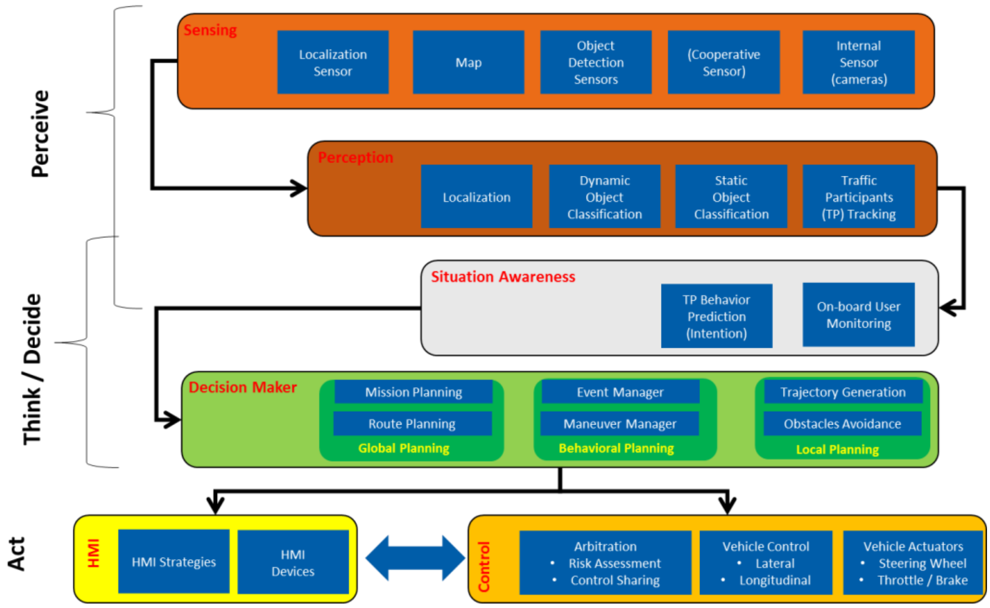

Fig. 4. Sketch of the logical architectural scheme in the PRYSTINE system.

The first layer, **Perceive**, includes both the sensing and the perception parts, where the raw information coming from the sensors are elaborated and processed, in order to derive a detailed picture of the external scene. Of course, in case a driving simulator is used, this is done automatically in the simulation.

Then, such an information is assessed in the second layer, **Think/Decide**, which includes a module (named *situation awareness*) for the prediction of the dynamic evolution of the external scene and the monitoring of the internal scene (namely, the status of the driver, what s/he is doing, and so on). The Decision-maker module that we have developed, is part of this layer; considering the three parts in which it is divided, we focused on the **behavioral planning**: our system can manage the events and the related maneuver, identifying the optimal actions to do.

This output, third layer (**Act**), can be provided directly to the user (in case of a Recommender system in ADAS applications) or to the vehicle actuators (in case of AD applications). In both situation a dedicated HMI is necessary: to support the driver and help to select the best solution in the first option; to inform the driver about what the systems is doing and what is expected from him/her in the second option.

3 Implementation of the IDSS

In this section, we first recall the general definition of MDP. Then, we propose an MDP for the considered use case (overtaking maneuver). The subsequent parts of the section are dedicated to present the proposed HMI concept and implementation.

3.1 MDP Definition

A MDP is a control or decision-making process, finalized at obtaining a desired behavior of a system of interest. The system of interest is often called the process to control or also the plant.

Mathematically, a MDP is defined as a 4-tuple (S,A,P,R), where:

S is the set of all plant states of interest. S is called the state space.

A is the set of actions that can be performed at a given time instant.

P is the state probability transition. In particular, given two states $S1,S2 \in S$ and an action $a \in A$, P(a,S1,S2) is the probability that the action a yields a transition from state S1 to state S2.

R is the immediate reward. It is possible to assign a reward to promote desire state transitions (positive reward) and penalize other transitions (negative reward).

3.2 The super-visor agent implementation

As discussed in Section 2.1, three UCs are developed in the PRYSTINE project. For the application of our IDSS-MDP framework, we consider the third one, sketched in Figure 2. In this scenario, the AV is traveling at a given speed, when it approaches a slower vehicle in the same lane, and the decision-maker has to define the next action. In the following, we formally introduce all the quantities that define the proposed MDP, that is the core of our IDSS.

Road scenario

Road, 3 lanes, from right to left:

- Lane 0: emergency lane.
- Lane 1: normal traveling lane.
- Lane 2: overtake lane.

Vehicles:

- AV: autonomous vehicle, initially in lane 1.

- PV: vehicle preceding AV, always in lane 1.
- OVs: other vehicles but AV and PV, possibly traveling in lane 1 and/or lane 2.

Main variables:

- v_x: longitudinal speed of AV.
- v_m: maximum longitudinal speed allowed on lanes 1 and 2.
- $v_r \leq v_m$: AV desired speed.
- $v_p < v_r \leq v_m$: longitudinal speed of PV.

MDP states

We consider a road scenario in a suitable neighborhood of AV.

AV states:

- $L0$: AV stopped in lane 0, $v_x = 0$.
- $L1$: AV lane 1 keeping, $v_x = v_p$ (before overtaking), $v_x = v_r$ (after overtaking).
- $L2$: AV lane 2 keeping, $v_x = v_m$.

Lane 1 states:

- $F1$: no PV, no OVs in lane 1.
- $P1$: PV in lane 1, no OVs in lane 1.
- $O1$: PV and OVs ahead of PV in lane 1.

Lane 2 states:

- $F2$: no OVs in lane 2.
- $O2$: OVs in lane 2.

Driver states:

- DA: healthy driver.
- DD: impaired driver.

The full scenario state is $S = (S_E, S_{L1}, S_{L2}, S_D) \in \mathcal{S}$, where $S_E \in \{L0, L1, L2\}$, $S_{L1} \in \{F1, P1, O1\}$, $S_{L2} \in \{F2, O2\}$, $S_D \in \{DA, DD\}$. The state space $\mathcal{S}$ is defined as $\mathcal{S} \doteq \{L0, L1, L2\} \times \{F1, P1, O1\} \times \{F2, O2\} \times \{DA, DD\}$. The total number of possible states is $\mathrm{card}(\mathcal{S}) = 3 \times 3 \times 2 \times 2 = 36$. For the sake of simplicity, we define a smaller number of aggregate states, allowing us to capture the relevant situations that may occur. In the following, the logic symbols $\forall$ (for all, for any), $\vee$ (or), $\wedge$ (and) will be used.

Aggregate states:

- $S0 \doteq (L0, \forall, \forall, \forall)$: AV stopped in lane 0, end state (12 non-aggregate states).
- $S11 \doteq \{(L1, F1, \forall, \forall), (L1, P1, F2, DD), (L1, P1, O2, \forall), (L1, O1, \forall, \forall)\}$: AV traveling in lane 1, overtaking not possible/not useful (11 non-aggregate states).

- $S12 \doteq (L1, P1, F2, DA)$: AV traveling in lane 1, overtaking possible (1 non-aggregate state).
- $S21 \doteq \{(L2, P1, \forall, \forall), (L2, O1, \forall, \forall)\}$: AV traveling in lane 2, re-entry in lane 1 not possible (8 non-aggregate states).
- $S22 \doteq (L2, F1, \forall, \forall)$: AV traveling in lane 2, re-entry in lane 1 possible (4 non-aggregate states).

The corresponding state space is $\mathcal{S}_A \doteq \{S0, S11, S12, S21, S22\}$.

MDP actions

We distinguish between two kinds of actions: inputs, i.e., actions decided by the MDP system (or by the driver), and events, i.e., actions coming from the external world, independent of the MDP.

Inputs:

- llc: left lane change.
- rlc: right lane change.

Events:

- vea: one or more OVs arrive in lane 1 ahead of PV and/or in lane 2, impeding to change lane.
- vem: all OVs in lane 1 and/or in lane 2 which impede to change lane get sufficiently far from AV.
- dba: driver becomes healthy.
- dbd: driver becomes impaired.

Aggregate events:

- $ovn = vea \vee dbd$: overtaking becomes impossible.
- $ovy = (vem \wedge (dba \vee DA) \vee (P1 \wedge F2 \wedge dba)$: overtaking becomes possible.
- $rey = vem$: re-entry becomes possible.

The set of input is $U \doteq \{llc, rlc\}$, the set of aggregate events is $E \doteq \{ovn, ovy, rey\}$ and the overall set of actions is $A \doteq U \cup E$.

MDP probability functions

The state transition probability functions for the two input actions llc and rlc are as follows:

$$P(llc, S_a, S_b) \quad = \begin{cases} p_o, & S_a = S12,\ S_b = S21 \\ 1 - p_o, & S_a = S12,\ S_b = S12 \\ 0, & \text{otherwise} \end{cases}$$

$$P(rlc, S_a, S_b) = \begin{cases} 0.5, & S_a = S22, S_b = S11 \\ 0.5, & S_a = S22, S_b = S12 \\ 0, & \text{otherwise.} \end{cases}$$

MDP graph

The MDP graph is shown in the figure below. It can be noted that, for each state, at most one action is defined. The MDP thus corresponds to a Markov Chain and no rewards need to be defined. In more complicated scenarios, rewards can be used to provide the MDP with more flexibility and capability to deal with different situations.

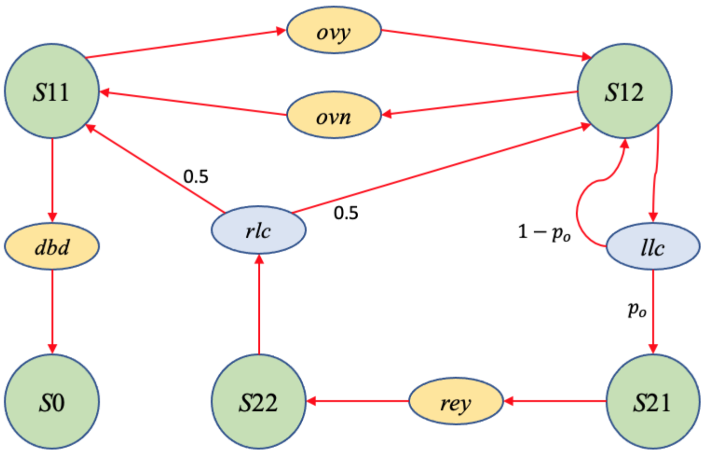

Fig. 5. MDP graph. Where not indicated, the transition probability associated with an edge is 1.

3.3 The Human-Machine Interface (HMI) Implementation

In the context mentioned in Chapter 1, the main goal of a Human-Machine Interaction system is to maximize the effectiveness of the cooperation between the human and the automated agent. In order to do that, this system shall be able to be easily understood by the driver, to increase his/her awareness about the situation, and (most important) to be trusted.

The HMI described in this paper is designed to exploit the potential of the Supervisor Agent, i.e., to easily represent its mental model in order to:

- Provide effective information / explanation when decisions are taken by the system.
- Encourage the cooperation when decisions and actions are shared between the human and the automated agents.

- Avoid unnecessary information when decisions and actions are delegated to the human driver, in order to avoid an overload in terms of cognitive and physical resources.

The HMI is deployed in a multimodal full-digital instrument cluster. It includes all relevant information related to the driving task (e.g., current speed, gear, automation mode etc.) as well as evidence about the driver's state (e.g. if he/she is distracted) and the action required to achieve an optimal driving (i.e. the suggested behavior). The HMI has been designed following the theories related to the negotiation-based interaction approach [13]. This means that the main goal of the HMI is to "explain rather than warn", in order to cooperate with the driver in achieving a pleasant, comfortable and safe drive.

According to the decision made by the supervisor agent, the HMI will inform the driver about what the vehicle expects from him/her, and provides messages related to the reasons that lead to the request of interaction. This is provided through:

- **Graphical explanations** provided through interactive 3D representation at the center of the HMI, where the road environment as well as the surrounding road actors are reconstructed (e.g., from digital maps plus vehicle's sensors) and displayed through a stylized representation
- **Messages** provided through audio signals and text.

The following figure, for example, shows a situation where the driver and the automation are sharing the vehicle control, and the car informs the driver that they are approaching a vehicle, that will be followed.

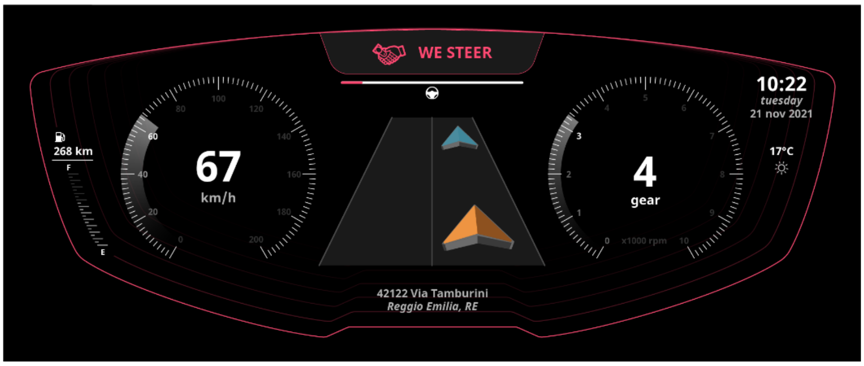

Fig. 6. Sketch of HMI for a control sharing between human driver and automated system.

The following figure shows a situation where the automation is engaged and, due to a combination of sensory limitation – i.e., lack of visibility - and the implementation of a cautious behavior, it actually informs that driver that the "car following" (CF) will results, unless the driver would intentionally override the system to perform a manual overtake. In this case, the 3D representation is highly focused on explaining the reasons behind the behavior (i.e., the visibility constraint) rather than the actual action requested to the driver, that is relegated to a small message on the upper right part of the screen.

This design choice relies on the implicit interactions [14], since it is aimed at fostering a behavior rather than explicitly force a reaction.

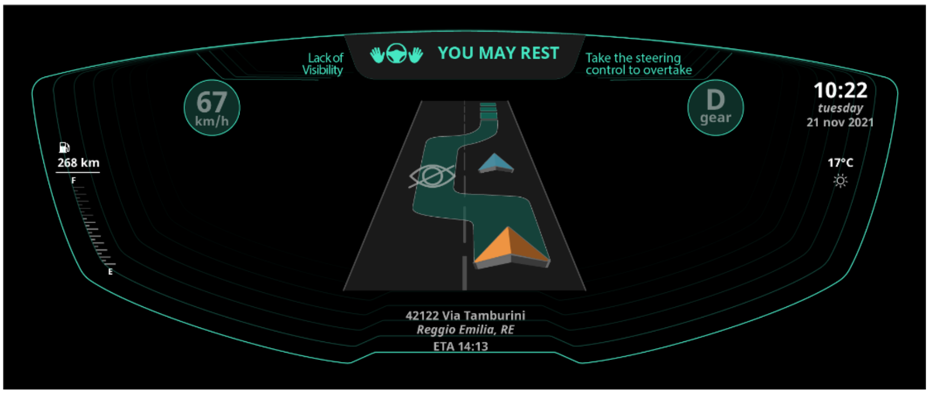

Fig. 7. Sketch of HMI informing driver about the action selected by the system (CF maneuver in this case).

Finally, the figure below shows the case where an automatic emergency maneuver is actuated by the vehicle; in this case, the explanation is provided before the actual stop of the vehicle, to allow the human driver to take back the control before the stop of the car. The cooperation here is provided showing the upcoming decision / behavior of the automation, i.e., to stop in the emergency lane.

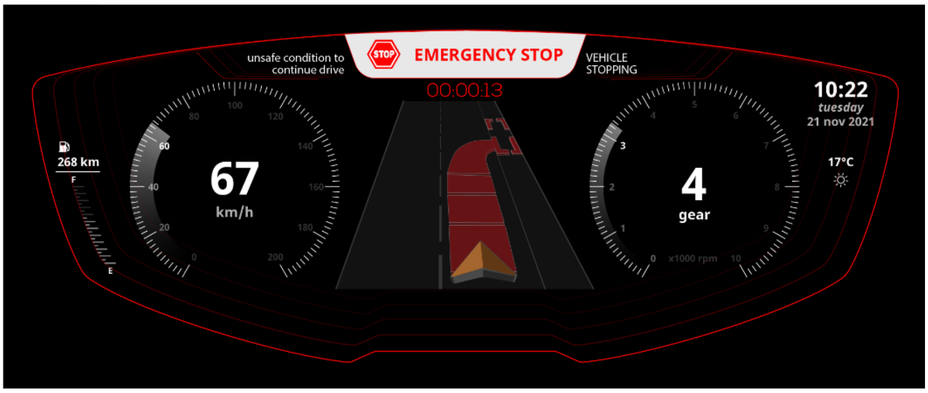

Fig. 8. Sketch of the HMI that informs the driver about the reason of the actuation for an emergency maneuver.

4 Data Analysis and Results

In the considered scenario, the autonomous vehicle (AV) is travelling at a speed of 70 km/h, when it approaches a preceding vehicle (PV), travelling at the lower speed of 45 km/h in the same lane, see Fig. 3 (AV=V1, PV=V2). The decision-maker has to choose in real-time the action to perform. Note that the IDSS can either inform the driver about the best maneuver to do or intervene on the vehicle actuators to perform the same maneuver. In the present case study, we have adopted the second approach, where the IDSS takes the action, based on what decided by the MDP. One important constraint that we impose is that the IDSS-MDP module cannot work completely alone: it needs in any case the driver to be healthy and aware. The following sub-scenarios have been considered:

Sub-scenario 1. The driver is healthy and aware. The MDP decides to either overtake or follow the preceding vehicle. Other two vehicles are traveling at a speed of 45 km/h in the opposite direction on the lane to be used for overtaking.

Sub-scenario 2. At a certain time, the driver becomes impaired. According to the imposed constraint, the IDSS-MDP module cannot work in a completely autonomous mode. Hence, after checking that the emergency lane is present, the MDP imposes a safe right lane change and a stop in the emergency lane.

For both sub-scenarios, two simulations were carried out: one corresponding to a MDP sporty strategy ($p_o = 1$), the other one corresponding to a MDP cautious strategy ($p_o = 0.1$). According to the probability functions defined, the sporty strategy performs an overtaking every time that is possible, while the cautious strategy performs an overtaking when it is possible but only with probability p_o.

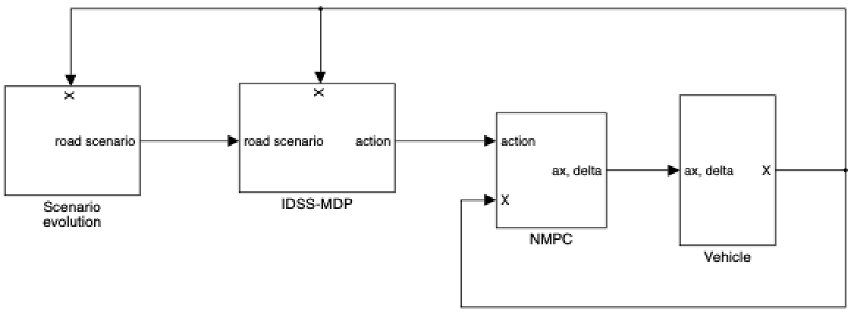

Fig. 9. Autonomous driving simulator block diagram.

The simulations were carried out by means of an autonomous driving simulator developed in MATLAB/SIMULINK. The simulator block diagram is shown in **Fig. 9** and is characterized by the following blocks:

Scenario evolution: This block generates the road scenario, (roads, vehicles, obstacles) and computes its evolution over time. Mathematically speaking, it provides the IDSS-MDP module with the coordinates of all elements that appear in the scenario.

Vehicle: Block describing the vehicle lateral and longitudinal dynamics. The block is essentially the “Vehicle Body 3DOF Dual Track” model, taken from the MATLAB Vehicle Dynamics Toolbox. The following parameter values were used: $l_f = 1.2\,\mathrm{m}$, $l_r = 1.6\,\mathrm{m}$ (lengths of front and rear longitudinal semi-axes), $m = 1575\,\mathrm{kg}$

(mass), $J = 4000\,\mathrm{kg\,m^2}$ (moment of inertia), $c_f = 27e3\,\mathrm{N/rad}, c_r = 20e3\,\mathrm{N/rad}$ (front and rear cornering stiffness coefficients). X is the vehicle state vector, containing the relevant kinematic and dynamic variables (linear and angular positions, linear and angular velocities), a_x is the requested longitudinal acceleration and δ is the commanded steering angle.

NMPC: Low-level controller, performing trajectory planning, and lateral and longitudinal AV dynamics control. The controller is based on a Nonlinear Model Predictive Control (NMPC) approach, see, e.g., [15,16]. NMPC is a general and flexible approach to nonlinear system control. It allows us to deal with input and trajectory constraints, and to manage systematically the trade-off between performance and command effort. The approach is based on two main operations (accomplished at each time step): (i) a prediction over a given time horizon is performed, using some vehicle model; (ii) the command input is chosen as the one yielding the ``best" prediction (i.e., the prediction closest to the desired behavior) by means of some on-line optimization algorithm. The NMPC controller works with a sampling time $T_s = 0.05\ s$.

IDSS-MDP: Intelligent Decision Support System, based on the MDP designed in Section 3.2. This block collects the information coming from the road scenario and the AV and indicates to the NMPC block the best action to perform at each time step, with a sampling time $T_d = 0.5\ s$. Note that the AV consists of the three blocks Vehicle, NMPC and IDSS-MDP.

The simulation results can be summarized as follows (see also **Fig. 10** to **Fig. 12**).

Simulation 1 (sub-scenario 1 and sporty MDP strategy). AV (traveling with speed 70 km/h) approaches PV (traveling with speed 45 km/h). Since another vehicle (OV1) is traveling in the opposite direction, AV reduces its speed, in order to follow PV. As soon as OV1 has gone, AV overtakes PV. After the overtake, a fourth vehicle (OV2) comes from the opposite direction, but this does not affect the behavior of AV.

Simulation 2 (sub-scenario 1 and cautious MDP strategy). AV (traveling with speed 70 km/h) approaches PV (traveling with speed 45 km/h). Since another vehicle (OV1) is traveling in the opposite direction, AV reduces its speed in order to follow PV. Here, the cautious strategy prefers to wait to overtake. In the meanwhile, OV2 comes from the opposite direction. AV waits also OV2 to move away and then it overtakes PV.

Simulation 3 (sub-scenario 2 and sporty MDP strategy). When the driver becomes impaired, AV performs a right lane change and then stops in the emergency lane.

Simulation 4 (sub-scenario 2 and cautious MDP strategy). Results similar to those of Simulation 3.

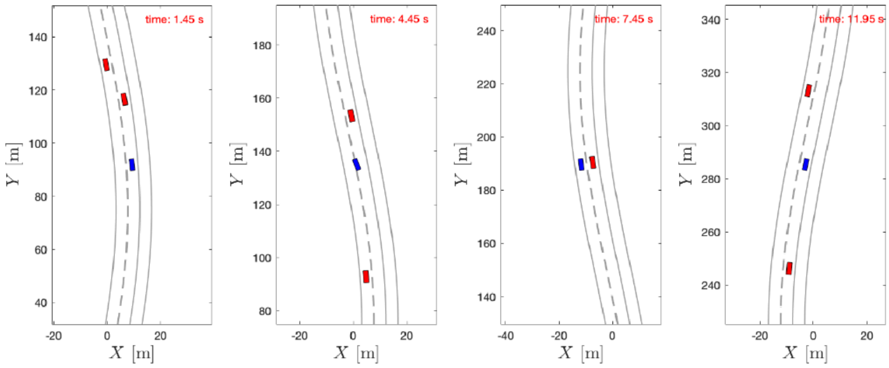

Fig. 10. Simulation 1. AV and PV travel upward. The other vehicles travel downward.

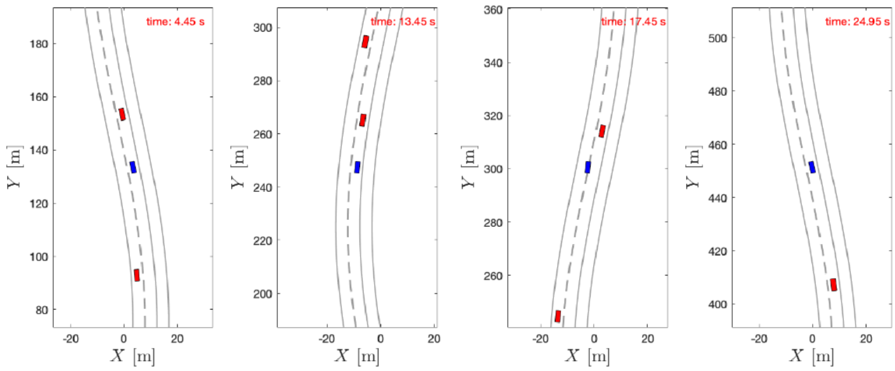

Fig. 11. Simulation 2. AV and PV travel upward. The other vehicles travel downward.

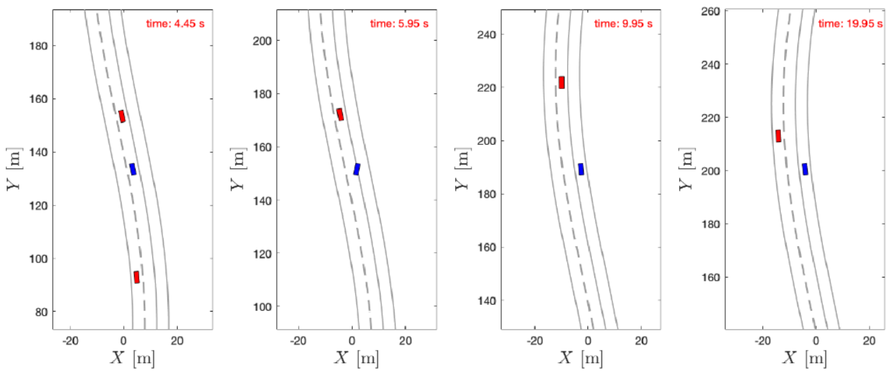

Fig. 12. Simulation 3. AV and PV travel upward. The other vehicles travel downward.

To evaluate the performance of the IDSS-DMP strategies in the two sub-scenarios, the following KPIs have been used:

- KPI1 [s]: Time taken to cover a given distance (450 m).
- KPI2 [s]: Time taken to stop in the emergency lane.
- KPI3 [m/s^2]: Root Mean Square (RMS) value of the lateral acceleration.
- KPI4 [m/s^2]: RMS value of the longitudinal acceleration. This KPI is clearly related to the fuel consumption.

The KPI values obtained in the various simulations are reported in **Table 1**. As expected, the MDP sporty strategy allows quicker maneuvers but implies larger lateral and longitudinal accelerations (and thus a higher fuel consumption) with respect to the cautious strategy. In any case, according to the MDP designed in Section 3.2, both strategies are allowed to overtake only if this maneuver is safe and both of them are able to command an emergency stop in short times.

Table 1. KPI values obtained in the simulations.

Simulation	KPI1	KPI2	KPI3	KPI4
1	20.15	-	1.37	1.69
2	24.6	-	1.23	1.54
3	-	18.8	1.52	2.35
4	-	19.3	1.52	2.35

5 Conclusions

The system we illustrated in this paper, based on MDP framework, can be regarded as a decision-maker (when applied to ADFs) and even as a recommender tool (when applied to ADAS). These types of systems are widely used in many fields, to provide recommendations and suggestions based on some criteria, such as user's preferences and styles (see the "sporty strategies" and the "caution strategies" in our simulations), safety (e.g., no other vehicles are already overtaking the AV in the adjacent lane) and comfort (avoiding too strong lateral/longitudinal accelerations). With the ever-growing volume of information online, these systems can be a useful tool to overcome information overload, or to suggest a proper action to automation, with the related explanation to the user about what is happening and why. In literature, there are many types of recommendation/decision-making systems with different methodologies and concepts.

Various applications include e-commerce, healthcare, transportation, agriculture, and media. This paper provided our proposed solutions for an intelligent system supporting the decision (IDSS) in the context of AD. We defined it as "intelligent", because it is able to adapt to the different states of the user (e.g., aggressive / cautious, distracted

/ attentive, and so on) and to the external conditions (e.g., the lane for overtaking is free), as well as because it provides the best actions, in the sense that it satisfies optimal criteria in terms of travelled time, safety and comfort.

This work is preparatory for the final phase of the PRYSTINE project, in which the MDP-based IDSS will be integrated and implemented in the project demonstrator, the prototype Maserati vehicle (presented in figure 2). In particular, we will apply our solution to the use-case 3, for the emergency lane-change maneuver (described in figure 3).

Declarations

Funding: This work was supported by the Electronic Components and Systems for European Leadership Joint Undertaking (ECSEL), which funded the PRYSTINE project under Grant 783190.

Conflict of interest: Authors declares that there are no conflicts of interests.

References

1. Rasmussen, J. (1982). "Human errors. A taxonomy for describing human malfunction in industrial installations." Journal of Occupational Accidents **4**(2-4): 311-333
2. Siau, K., & Wang, W. (2018). Building trust in Artificial Intelligence, Machine Learning, and Robotics. Cutter Business Technology Journal, 31(2), 47–53.
3. M.L. Puterman. "Markov Decision Processes. Discrete Stochastic Dynamic Programming". Wiley, Chichester, 2005.
4. Jean Piaget's Theory and Stages of Cognitive Development, by Saul McLeod, Simply Psychology, updated 2018.
5. Michon, John A. (1985). A critical view of driver behavior models: what do we know, what should we do?, in Evans, L. & Schwing, R.C. (eds), Human.
6. Tango F., Aras R. and Pietquin O. (2010, June-July, 30-02). Learning optimal control strategies from interactions for a Partially Autonomous Driver Assistance System. In the Proceeding of I Workshop on Human Modelling in Assisted Transportation (HMAT). Belgirate, Italy. Published by Springer.
7. Ricci, F.; Rokach, L.; Shapira, B. Introduction to recommender systems handbook. In Recommender Systems Handbook; Springer: Boston, MA, USA, 2011; pp. 1–35
8. Fayyaz Z., Ebrahimian M., Nawara D., Ibrahim A. and Kashef R. Recommendation Systems: Algorithms, Challenges, Metrics, and Business Opportunities. Appl. Sci. 2020, 10(21), 7748; https://doi.org/10.3390/app10217748.
9. Beel, J.; Langer, S.; Genzmehr, M.; Gipp, B.; Breitinger, C.; Nürnberger, A. Research paper recommender system evaluation: A quantitative literature survey. In Proceedings of the International Workshop on Reproducibility and Replication in Recommender Systems Evaluation, Hong Kong, China, 12 October 2013; pp. 15–22.
10. Carsten, Oliver, and Marieke H. Martens. "How can humans understand their automated cars? HMI principles, problems and solutions." Cognition, Technology & Work 21.1 (2019): 3-20.
11. Sharma, Anshuman, et al. "Is an informed driver a better decision maker? A grouped random parameter with heterogeneity-in-means approach to investigate the impact of the connected

environment on driving behavior in safety-critical situations." Analytic methods in accident research 27 (2020): 100127.

12. Castellano, Andrea, et al. "Is your request just this? New automation paradigm to reduce the requests of transition without increasing the effort of the driver." 25th ITS World Congress, Copenhagen, Denmark. Vol. 17. 2018
13. Gowda, Nikhil, Wendy Ju, and Kirstin Kohler. "Dashboard design for an autonomous car." Adjunct proceedings of the 6th international conference on automotive user interfaces and interactive vehicular applications. 2014.
14. Ju, Wendy. "The design of implicit interactions." Synthesis Lectures on Human-Centered Informatics 8.2 (2015): 1-93
15. Findeisen, R., Allgower, F., & Biegel, L. (2007). Assessment and future directions of non-linear model predictive control. In Lecture notes in control and information sciences. Springer.
16. Grune, L., & Pannek, J. (2011). Nonlinear model predictive control - theory and algorithms. In Communications and control engineering. Springer.