Towards Provably Correct Driver Assistance Systems through Stochastic Cognitive Modeling

Francisco Eiras[1] and Morteza Lahijanian[2]

***Abstract*—The aim of this study is to introduce a formal framework for analysis and synthesis of driver assistance systems. It applies formal methods to the verification of a stochastic human driver model built using the cognitive architecture ACT-R, and then bootstraps safety in semi-autonomous vehicles through the design of provably correct Advanced Driver Assistance Systems. The main contributions include the integration of probabilistic ACT-R models in the formal analysis of semi-autonomous systems and an abstraction technique that enables a finite representation of a large dimensional, continuous system in the form of a Markov model. The effectiveness of the method is illustrated in several case studies under various conditions.**

I. Introduction

When it comes to driving, the numbers do not lie; more than 90% of road accidents in the US are caused by human error [19]. In an effort to increase driver safety, some car manufacturers have introduced semi-autonomous features in the form of *Advanced Driver Assistance Systems* (ADAS). Despite this, guaranteeing safety in semi-autonomous vehicles remains a challenge, with most of the existing methods being based on testing and simulation [5, 10, 20, 22], which do not provide the guarantees required for a safety critical system [11]. Some recent works use formal verification to obtain strong guarantees about the ADAS [6, 14, 16], yet they present engineering approaches to the problem which ignore the cognitive process of the human driver, leading to solutions that might perform poorly in corner cases.

This study focuses on designing an ADAS that takes into account a stochastic model of the driver cognitive process. It employs the cognitive architecture known as *Adaptive Control of Thought-Rational* (ACT-R), a framework for specifying computational behavioral models of human cognitive performance which embodies both the abilities (e.g. memory storage or perception) and constraints (e.g. limited motor performance) of humans [1, 2, 4, 17, 18, 21]. The work builds on the human driver model in a multi-lane highway driving scenario presented in [18]. It also expands upon [6, 13] by applying verification techniques to an efficient abstraction of the model and extends it to allow the intervention of a *provably correct* (up to the level of representation of the model) ADAS based on specifications given as temporal logic statements.

The problem is defined as follows. Given the vehicle model from [15], a human driver model represented by ACT-R [18], a set of initial conditions $\mathcal{S}$, and a temporal logic formula φ [3], we are interested in (1) **verification**: computing the probability that the Human-Vehicle model satisfies φ in $\mathcal{S}$, i.e, $\mathbb{P}^{\mathcal{S}}(\varphi)$; and (2) **synthesis**: designing an ADAS that optimizes the probability of satisfying φ by the Human-Vehicle-ADAS system in $\mathcal{S}$, i.e., $\mathbb{P}^{\mathcal{S}}_{\bowtie}(\varphi)$ with $\bowtie \in \{\max, \min\}$.

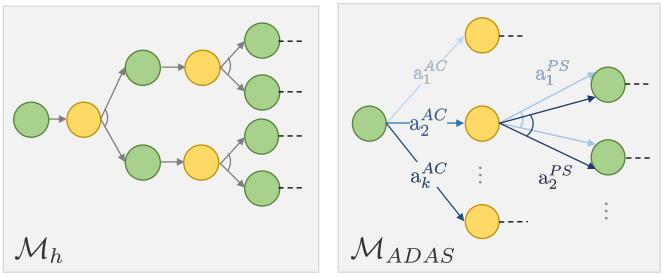

Fig. 1: State-transition representations of the Human-Vehicle ($\mathcal{M}_h$) and the Human-Vehicle-ADAS ($\mathcal{M}_{ADAS}$) systems; in green are *control* states ($\mu = 1$) and in yellow are *decision making and monitoring* ones ($\mu = 2$).

II. Methodology

To verify the human driver model under φ, we first abstract it to a Markov Chain $\mathcal{M}_h = (S, \mathbf{P}, s_0, AP, L)$, where S is a finite set of states, $\mathbf{P} : S \times S \to [0, 1]$ is a transition probability function, $s_0 \in S$ is the initial state, AP is a set of atomic propositions, and $L : S \to 2^{AP}$ is a labeling function. We achieve this by discretizing the integrated human driver ACT-R model in [18] through the use of a vehicle model [15]. We can then verify it using off-the-shelf tools, e.g. PRISM [12]. We assume a two vehicle scenario, where the ego-vehicle interacts with a lead vehicle whose motion is predictable [9]. The driver model presented in [18] is divided into three sequential modules: (i) *control*, which manages low level perception cues and the manipulation of the vehicle, (ii) *monitoring*, which maintains awareness of the position of other vehicles around the ego-vehicle; and (iii) *decision making*, which determines the tactical decision to be taken.

Our abstraction combines decision making and monitoring into one module for the sake of efficiency. We define $\mathcal{M}_h$ which unifies both modules through the use of $\mu \in \{1, 2\}$, where $\mu = 1$ corresponds to the control step and $\mu = 2$ to the decision making and monitoring stage. A state $s \in S$ is a tuple $s = (\mu, x, \lambda, a, v, t)$, where x is bounded to a finite length of the road according to the situation, v is the speed of the vehicle, a is the acceleration and λ represents the index of the lane of the ego, abstracting away the y variable which reduces

[1]Francisco Eiras was a student at the Department of Computer Science, University of Oxford when developing this work and is now with FiveAI Inc francisco.eiras@five.ai

[2]Morteza Lahijanian is with the department of the Ann and H.J. Smead Aerospace Engineering Sciences, University of Colorado Boulder morteza.lahijanian@colorado.edu

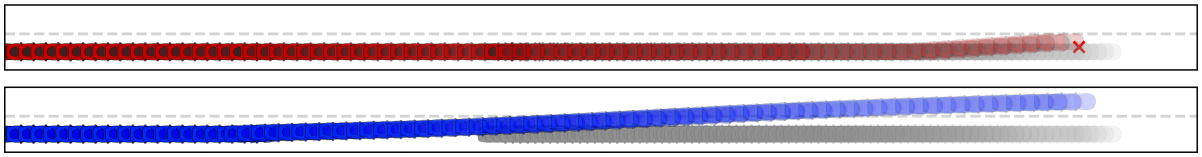

Fig. 2: Example of a trajectory under φ_1 for a given set of initial conditions $\mathcal{S}$. Top in red: human-vehicle system (no ADAS - $\mathbb{P}^S(\varphi_1) = 0.489$). Bottom in blue: human-vehicle system with ADAS ($\mathbb{P}^S_{\min}(\varphi_1) = 0.242$). Gray: the other vehicle. For readability purposes, the opacity of the cars decreases with time. The red 'x' marks a collision between the vehicles.

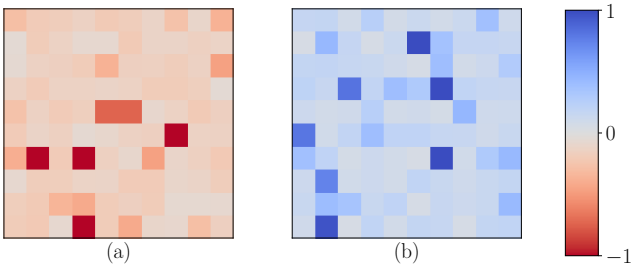

Fig. 3: $\mathbb{P}^{\mathcal{S}}_{\bowtie}(\varphi) - \mathbb{P}^{\mathcal{S}}(\varphi)$ for a randomly sampled population of 100 different initial conditions $\mathcal{S}$ for (a) $\varphi = \varphi_1$, $\bowtie = \min$ and (b) $\varphi = \varphi_2$, $\bowtie = \max$.

the size of the model. A time discretization is induced for all the continuous variables. For a given set of initial conditions $\mathcal{S}$, the state space S is automatically generated. The transition probabilities for all $s, s' \in S$ are given by:

$$\mathbf{P}(s,s') = \begin{cases} 1 & \text{if } \mu_s = 1 \wedge s' = \textsc{Control}(s), \\ \text{DMM}(s,s') & \text{if } \mu_s = 2, \\ 0 & \text{otherwise,} \end{cases}$$

where CONTROL is a deterministic transition table resulting from the simulation of the control laws from [18] and DMM is a table of transition probabilities based on the introduction of **(1)** Gaussian noise to the decision making and monitoring processes presented in [18] as a way to model the uncertainty of perception; and **(2)** stochastic uncertainty in terms of the lane changing decision based on driver variability.

To obtain the Human-Vehicle-ADAS system, we augment $\mathcal{M}_h$ with possible realistic interventions by the ADAS, as presented in Fig. 1. These interventions can be of two types: *passive suggestions* (PS) and *active control* (AC). In passive suggestions, we assume that the assistance system cannot change the decision making directly, as it is a human cognitive process, but it can influence it to a certain degree through suggestions [8], i.e. an action at this level, a_i^{PS}, induces the probability distribution $\text{DMM}_i(s,s')$, which is biased towards the desired outcome. In active control, the actions available to the ADAS, a_i^{AC}, can have corrective control-based interventions at the level of acceleration and steering (with ADAS variables constrained to ensure incremental interventions), deterministically leading to different states according to $\textsc{Control}_i(s)$. The optimal ADAS design is reduced to finding an optimal policy over $\mathcal{M}_{\text{ADAS}}$ for a certain specification given as a temporal logic formula φ defined over AP. Off-the-shelf tools, such as PRISM [12], can be employed for this computation.

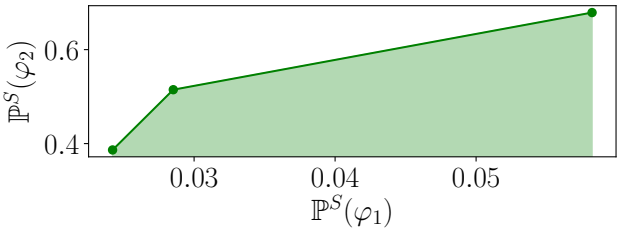

Fig. 4: Pareto frontier resulting of the multi-objective optimization of minimizing φ_1 and maximizing φ_2 in $\mathcal{M}_{ADAS}$.

III. Experimental Results

The framework was implemented in Python using PRISM and the code is available on Github[1]. To study its applications, we considered a simplified two lane scenario of length $x_{\max}$ where the lead vehicle is assumed to be moving at a constant speed. We also considered two interesting properties:

$$\varphi_1 = \Diamond\, \textsc{crash}, \quad \text{and} \quad \varphi_2 = \Diamond\, \big((x = x_{\max}) \wedge (t \leq T)\big),$$

which we want to minimize and maximize, respectively. Intuitively, φ_1 refers to how unsafe the system is, while φ_2 corresponds to the time efficiency of it.

Fig. 2 shows an example of a trajectory under φ_1 for a given highly unsafe initial situation $\mathcal{S}$, in which the ADAS effectively leads the system to a safer situation. Fig. 3 showcases the difference in probabilities of satisfying (a) φ_1 and (b) φ_2, assuming each specification to be optimized individually. In both cases, all randomly generated scenarios tested lead to a decrease in the case of φ_1 and an increase in the case of φ_2, i.e. improved satisfaction of the desired properties.

These results refer to optimizing each of the properties individually and do not offer any insight into how optimizing one influences the satisfaction of the other. Through our framework, we are also able to study the relationships between properties using multi-objective optimization techniques. Fig. 4 presents the Pareto frontier of optimizing φ_1 and φ_2 for a given $\mathcal{S}$ in a multi-objective setting, showing that, as expected, there is a trade-off between the two properties.

A more in-depth analysis can be found in [7], including situations with more vehicles and complex specifications.

IV. Final Remarks

The approach proposed in this paper enables the study of safety of semi-autonomous vehicles in various conditions and the design of ADAS that are robust with formal guarantees. In the future, the specifications passed to the ADAS could be learnt so as to match the behavior of expert drivers.

[1]https://github.com/fgirbal/cbc_adas

References

[1] John R Anderson. *The Architecture of Cognition*. Psychology Press, 2013.

[2] John R Anderson, Michael Matessa, and Christian Lebiere. ACT-R: A theory of higher level cognition and its relation to visual attention. *Human-Computer Interaction*, 12(4):439–462, 1997.

[3] Christel Baier and Joost-Pieter Katoen. *Principles of model checking*. MIT press, 2008.

[4] Tina Balke and Nigel Gilbert. How do agents make decisions? A survey. *Journal of Artificial Societies and Social Simulation*, 17(4):13, 2014.

[5] Sonia Baltodano, Srinath Sibi, Nikolas Martelaro, Nikhil Gowda, and Wendy Ju. The rrads platform: a real road autonomous driving simulator. In *Proceedings of the 7th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, pages 281–288. ACM, 2015.

[6] Taolue Chen, Marta Kwiatkowska, Aistis Simaitis, and Clemens Wiltsche. Synthesis for multi-objective stochastic games: An application to autonomous urban driving. In *International Conference on Quantitative Evaluation of Systems*, pages 322–337. Springer, 2013.

[7] Francisco Eiras. To err is human: Designing correct-by-construction driver assistance systems using cognitive modelling. Master's thesis, University of Oxford, 2018.

[8] Ray Fuller. Towards a general theory of driver behaviour. *Accident analysis & prevention*, 37(3):461–472, 2005.

[9] David Sierra González, Jilles Steeve Dibangoye, and Christian Laugier. High-speed highway scene prediction based on driver models learned from demonstrations. In *2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)*, pages 149–155. IEEE, 2016.

[10] Andreas Gruber, Michael Gadringer, Helmut Schreiber, Dominik Amschl, Wolfgang Bösch, Steffen Metzner, and Horst Pflügl. Highly scalable radar target simulator for autonomous driving test beds. In *Radar Conference (EURAD), 2017 European*, pages 147–150. IEEE, 2017.

[11] Philip Koopman and Michael Wagner. Challenges in autonomous vehicle testing and validation. *SAE International Journal of Transportation Safety*, 4(1):15–24, 2016.

[12] Marta Kwiatkowska, Gethin Norman, and David Parker. PRISM 4.0: Verification of probabilistic real-time systems. In *International conference on computer aided verification*, pages 585–591. Springer, 2011.

[13] Morteza Lahijanian, Sean B. Andersson, and Calin Belta. Temporal logic motion planning and control with probabilistic satisfaction guarantees. *IEEE Transactions on Robotics*, 28(2):396–409, Apr. 2012.

[14] Petter Nilsson, Omar Hussien, Yuxiao Chen, Ayca Balkan, Matthias Rungger, Aaron Ames, Jessy Grizzle, Necmiye Ozay, Huei Peng, and Paulo Tabuada. Preliminary results on correct-by-construction control software synthesis for adaptive cruise control. In *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*, pages 816–823. IEEE, 2014.

[15] Rajesh Rajamani. *Vehicle dynamics and control*. Springer Science & Business Media, 2011.

[16] Dorsa Sadigh, Katherine Driggs-Campbell, Alberto Puggelli, Wenchao Li, Victor Shia, Ruzena Bajcsy, Alberto Sangiovanni-Vincentelli, S Shankar Sastry, and Sanjit Seshia. Data-driven probabilistic modeling and verification of human driver behavior. In *2014 AAAI Spring Symposium Series*, 2014.

[17] Dario Salvucci, Erwin Boer, and Andrew Liu. Toward an integrated model of driver behavior in cognitive architecture. *Transportation Research Record: Journal of the Transportation Research Board*, (1779):9–16, 2001.

[18] Dario D Salvucci. Modeling driver behavior in a cognitive architecture. *Human factors*, 48(2):362–380, 2006.

[19] Santokh Singh. Critical reasons for crashes investigated in the national motor vehicle crash causation survey. Technical report, 2015.

[20] Daniele Sportillo, Alexis Paljic, Mehdi Boukhris, Philippe Fuchs, Luciano Ojeda, and Vincent Roussarie. An immersive virtual reality system for semi-autonomous driving simulation: a comparison between realistic and 6-dof controller-based interaction. In *Proceedings of the 9th International Conference on Computer and Automation Engineering*, pages 6–10. ACM, 2017.

[21] Niels A Taatgen, Christian Lebiere, and John R Anderson. Modeling paradigms in ACT-R. *Cognition and multi-agent interaction: From cognitive modeling to social simulation*, pages 29–52, 2006.

[22] Mofan Zhou, Xiaobo Qu, and Sheng Jin. On the impact of cooperative autonomous vehicles in improving freeway merging: a modified intelligent driver model-based approach. *IEEE Transactions on Intelligent Transportation Systems*, 18(6):1422–1428, 2017.