

VERIFCAR: A Framework for Modeling and Model checking Communicating Autonomous Vehicles

JAAMAS Track

Johan Arcile
IBISC, Univ Evry, Université
Paris-Saclay
Evry, France
johan.arcile@univ-evry.fr

Raymond Devillers
ULB
Bruxelles, Belgium
rdevil@ulb.ac.be

Hanna Kludel
IBISC, Univ Evry, Université
Paris-Saclay
Evry, France
hanna.kludel@univ-evry.fr

ABSTRACT

This paper presents a framework, called VERIFCAR, devoted to the validation of decision policies of communicating autonomous vehicles (CAVs). The approach focuses on the formal modeling of CAVs by means of timed automata, allowing a formal and exhaustive analysis of the behaviors of vehicles. VERIFCAR supports a parametric modeling of CAV systems as a network of timed automata tailored for verification and limiting the well-known state space explosion. As an illustration, VERIFCAR is applied to check robustness and efficiency, as well as to assess the impact of communication delays on the decision algorithms of CAVs, on well chosen case studies representing real-life critical situations.

KEYWORDS

Timed Automata, Formal Verification, Model Checking, Communicating Autonomous Vehicles

ACM Reference Format:

Johan Arcile, Raymond Devillers, and Hanna Kludel. 2020. VERIFCAR: A Framework for Modeling and Model checking Communicating Autonomous Vehicles. In *Proc. of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020), Auckland, New Zealand, May 9–13, 2020*, IFAAMAS, 2 pages.

Autonomous vehicles are rational sophisticated entities (agents) that, as the term already suggests, act autonomously across open and distributed environments. They may have different perceptions of the environment because of the information they possess and their differing interests in terms of the mission to be accomplished. Inter-vehicle communications affect perceptions and, in turn, individual decisions and behaviors. A system of *communicating* autonomous vehicles (CAV system) is then both a multi-agent system [9] and a real-time system [2]. More precisely, a CAV system is a network of vehicles that communicate with their neighbors to fulfill a mission as fast as possible while complying to the code rules and avoiding crashes. Moreover, its correctness also requires to respect a set of time constraints.

Through some level of abstraction, computer simulations enable to model vehicles' behaviors in a chosen environment so that various kinds of scenarios may be studied [6, 7, 10]. However, when vehicles present non-deterministic behaviors, simulation tools are

generally not exhaustive since each simulation corresponds to a single path in the graph of all the possible behaviors. This is especially true in the context of communicating agents, where agents interact on non-deterministic timed intervals, adding to the non-determinism. It is therefore appealing to formally verify the core CAV behaviors in order to be confident in the integration of autonomous vehicles into the road traffic.

Formal modeling and verification of CAV systems require not only the definition of both the vehicle states and the road (the environment) but also a specification of interactions between vehicles and an expressive query language to check properties. Optimally, the resulting model of a CAV system should be accurate enough to capture spatial and time aspects of the original system and should also provide a formal basis for the verification of properties like robustness to faults, effectiveness of maneuvers or the impossibility of collisions (safety), and for the calibration and assessment of decision-making policies. The language for stating properties should be expressive enough for our needs and appropriate for applying automatic verification techniques and tools.

A widely used automatic technique for system verification is model checking [5]. It provides algorithmic means for determining whether an abstract model – representing a hardware or software project, or a mixture of them (in our case a CAV system) – satisfies a formal specification (property) expressed as a temporal logic formula. Moreover, if the property does not hold, the method usually identifies a counterexample run that shows the/a source of the problem.

The main objective of our paper is to present a way to perform formal modeling and model checking of CAV systems, focusing on the impact of various types of communications on vehicles' safety and traffic fluidity. More specifically, we present a framework, called VERIFCAR, composed of a scalable model of a CAV system optimized for formal verification together with a method of calculating indicators allowing to evaluate the quality of a given autonomous vehicle decision-making policy when the Boolean constraints are fulfilled. The framework is designed in particular to be exhaustive on the non-determinism induced by the latency, communication delays and concurrency features. To show the usefulness of our approach, we present various examples of impacts the communications may have on safety, efficiency or traffic fluidity.

The underlying modeling formalism that we use in VERIFCAR to specify the behavior of CAVs is a model of Timed Automata [2], which is a standard supported by several verification tools, e.g., the model checker UPPAAL [8]. The timed automata formalism is the

most well-established model for the specification and verification of distributed real-time system designs. Among many advantages, it allows:

- to create a clear and concise abstract model of the considered CAV systems;
- to assess the robustness of a vehicle decision policy through a fault injection, and
- to apply model checking algorithms and tools, in particular the algorithms designed for timed properties expressed in the temporal logic TCTL [1].

To the best of our knowledge, this kind of formal approach does not seem to have been exploited up to now. VERIFCAR is particularly well adapted for the exhaustive analysis of critical situations involving a few number of vehicles, such as overtaking, insertion lanes, crossroads or traffic roundabouts. It builds on general concepts borrowed from [3] to model such systems of CAVs but brings several novel ideas and improvements. First, it implements a totally different timed automata model, using broadcast synchronizations instead of handshake, which greatly simplifies the automata and contributes to accelerate the verification processes. Thanks to suitable discretization and approximations, the exhaustive techniques of model checking can be used while limiting the state explosion phenomena. Next, it supports two additional forms of cooperation: negotiations between CAVs and communications with an intelligent road infrastructure. Finally, it provides pertinent indicators for the analysis of behaviors, together with a computation methodology using model checking. As shown in a companion paper [4], this method can be used jointly with simulations, contributing to give more insight on the studied systems.

As an illustration, we applied VERIFCAR to assess and compare decision-making policies for CAV systems on well chosen scenarios. To do so we implemented decision algorithms (including negotiations and infrastructure), computed the associated indicators and

discussed the obtained results. We also checked the robustness through faulty environments, pointing out vulnerabilities and thus illustrating how VERIFCAR could be used to detect and thwart those vulnerabilities.

Nevertheless, there is still room for improvement, especially concerning the computation of numerical indicators, restrictions on the solvable specification formulas, sensibility to scalability, and we are presently working on it.

REFERENCES

- [1] R. Alur, C. Courcoubetis, and D. Dill. 1993. Model Checking in Dense Real-Time. *Information and Computation* 104, 1 (1993), 2–34.
- [2] Rajeev Alur and David Dill. 1990. Automata for modeling real-time systems. In *Automata, Languages and Programming*, Michael S. Paterson (Ed.). Springer Berlin Heidelberg, Berlin, Heidelberg, 322–335.
- [3] Johan Arcile, Raymond Devillers, Hanna Kludel, Witold Kludel, and Bożena Woźna-Szcześniak. 2018. Modeling and checking robustness of communicating autonomous vehicles. In *Distributed Computing and Artificial Intelligence, 14th International Conference*, Sigeru Omatu, Sara Rodríguez, Gabriel Villarrubia, Pedro Faria, Paweł Sitek, and Javier Prieto (Eds.). Springer International Publishing, Cham, 173–180.
- [4] Johan Arcile, Jérémy Sobieraj, Hanna Kludel, and Guillaume Hutzler. 2018. Combination of Simulation and Model-Checking for the Analysis of Autonomous Vehicles' Behaviors: A Case Study. In *Multi-Agent Systems and Agreement Technologies*, Francesco Belardinelli and Estefania Argenste (Eds.). Springer International Publishing, Cham, 292–304.
- [5] E. M. Clarke, O. Grumberg, and D. Peled. 1999. *Model Checking*. MIT Press.
- [6] Fan Bai and H. Krishnan. 2006. Reliability Analysis of DSRC Wireless Communication for Vehicle Safety Applications. In *2006 IEEE Intelligent Transportation Systems Conference*. 355–362. <https://doi.org/10.1109/ITSC.2006.1706767>
- [7] Martin Treiber and Arne Kesting. 2013. *Trajectory and Floating-Car Data*. Springer Berlin Heidelberg, Berlin, Heidelberg, 7–12. https://doi.org/10.1007/978-3-642-32460-4_2
- [8] UPPAAL [n. d.]. UPPAAL. <http://www.uppaal.org/>.
- [9] M. Wooldridge. 2009. *An introduction to multi-agent systems - Second Edition*. John Wiley & Sons.
- [10] S. Zhang, W. Deng, Q. Zhao, H. Sun, and B. Litkouhi. 2013. Dynamic Trajectory Planning for Vehicle Autonomous Driving. In *2013 IEEE International Conference on Systems, Man, and Cybernetics*. 4161–4166. <https://doi.org/10.1109/SMC.2013.709>