Advanced co-simulation framework for cooperative maneuvers among vehicles

Antonio Artuñedo, Jorge Godoy, Rodolfo Haber, Raúl M. del Toro
Center for Automation and Robotics
CSIC – UPM
Arganda del Rey, Spain
{antonio.artunedo, jorge.godoy, rodolfo.haber, raul.deltoro}@car.upm-csic.es

Jorge Villagrá
On-Board Villagrá
Ixion Madrid, Spain
jvillagra@ixion.es

Abstract—In this work we propose a tool for simulation of cooperative maneuvers among autonomous vehicles in which virtual and real vehicles can conjunctively interact. It is a generic simulation platform where the user can define the desired scenario using a graphical user interface. This interface facilitates insertion of controllers and models for different parts of vehicles and some elements of the infrastructure. Furthermore, vehicle to vehicle (V2V) and vehicle to infrastructure (V2I) communications are available in the framework. The simulator also enables the 3D rendering of simulation and monitoring of several variables at runtime. The main advantage of the proposed framework relies on the use of hybrid simulation by combining real and virtual vehicles for studying their dynamic behavior and interaction without the need of real expensive equipment or vehicles. Therefore, the emulation of experimental tests based on data sets from the vehicle sensors serves as a powerful tool for designing and evaluation new ADAS.

Keywords—Simulation, Intelligent transportation systems, Intelligent vehicles, Inter-vehicle communications

I. INTRODUCTION

In last decades, the progress of new communication technologies and the advancement of processing capacity of computers gave rise to new challenges for intelligent transportation systems (ITS) applications. New challenges involve developing new systems for improving the performance of different ITS aspects such as efficiency, reliability, safety and security. ITS comprises a set of technological solutions for telecommunications and information technologies, designed to improve the operation and safety of transportation. One of the main branches of the development of this set of telematics solutions is mainly oriented to road transportation. Within the road transportation branch, ITS make use of V2V and V2I communications as well as autonomous perception systems based on sensors such as lidar, radar or cameras. In general, different types of ITSs are based on radio communication technologies and the use of specialized technologies.

The reported ITS applications are growing rapidly in literature but also in available commercial solutions of vehicles manufacturers. They include urban transportation, adaptive traffic control systems (ATCS) [1], [2], [3], advanced traveler information systems (ATIS) and applications for freight and logistics [4] amongst others. Nowadays, active safety systems are mainly focused on how to refine and improve advanced driver assistance systems (ADAS), which is indeed a priority in the research and development agenda for the academic and industrial sector.

In the scope of the transportation research, simulation and rapid control prototyping methodologies take a key role in ADAS development. Nonetheless, testing algorithms in real cars is a necessary step in developing new intelligent abilities for future ITS. However, this step is sometimes hard to carry out due to hardware and vehicle physical problems, economic costs and resources availability. Moreover, it is also difficult to reproduce the same scenario several times due to variability of experimental parameters such as perception conditions. Consequently, it is hard to compare different algorithms using the same experimental conditions. In addition, some hazardous scenarios such as those that imply possible collisions between physical devices, are difficult to test in the real world.

A tool for simulating a wide range of ITS applications is proposed in this paper. This is a framework for simulating cooperative maneuvering among vehicles (real and virtual). The main goal is to address the new needs on ITS simulation such as ADAS, intelligent sensors simulation and testing of reliability and safety of controllers for autonomous vehicles, maintaining a balance between vehicle dynamics simulators and microscopic traffic simulators. Our proposal addresses the gap between traffic simulators and vehicle dynamics simulators allowing ITS researchers and developers to test on board vehicle equipment such as sensors, actuators or controllers, and cooperative transport maneuvers within urban realistic scenarios. The real-virtual interaction allows to combine real vehicles with simulated scenarios, therefore it is possible to evaluate the performance of the real vehicle in critical situations without jeopardizing the vehicles or the infrastructure.

The paper is organized as follows: In section II a brief review of related work is presented. Section III introduces the proposed framework where detailed descriptions of the structure and components of the tool are provided. In section IV the evaluation of the tool is presented. Some experiments were carried out for comparing real tests with simulation outputs. Finally, in section V the main conclusions of this work are presented as well as future work planned for further improvements and new features of the framework.
II. RELATED WORK

Currently, there are tools that allow the simulation of road traffic while others are focused on the dynamics of a single vehicle. Most of these simulation tools, both commercial and free software distributions, are focused on different aspects of the traffic. Usually these tools use macroscopic or microscopic traffic models. On the one hand macroscopic models are centered on capturing global relations of traffic flow, such as vehicle speed, traffic flow and traffic density. Some examples of macroscopic simulators are AIMSUN [5] or Visum [6]. On the other hand microscopic models consider the dynamic aspects of each vehicle individually for traffic analysis. Some of the most known microscopic simulators are SUMO [7] and PTV Vissim [8]. Nevertheless, other tools are focused on simulation of vehicle dynamics regardless traffic in order to predict the performance of a vehicle in response to driver controls in a given environment. Two of the most known of this kind of simulators are CarSim [9] and veDyna [10]. However, it is hard to find an intermediate simulation platform which considers microscopic traffic behavior, without forgetting the vehicle dynamics.

From existing applications, the closest to the concept proposed in this work are PreScan, ProSivic and USARSim. PreScan [11] is a commercial tool for model-based controller design and real-time tests with software-in-the-loop (SIL) and hardware-in-the-loop (HIL) systems. However, the simulation framework is based on simplified sensor models and it does not admit co-simulation between real and virtual vehicles unlike our approach. Likewise, ProSivic [12] is another commercial software architecture dedicated to vehicles, infrastructures, sensors and ADAS simulation. However, it also does not allow including real and virtual vehicles. USARSim [13] is an open source robot simulator focused on the behavior of a robot or vehicle rather than to permit the study and analysis of cooperative strategies for robots. This approach differs from ours since it is mainly focus on robot simulations for education, research and robot competitions.

The interaction between real and virtual vehicles is not allowed in the analyzed applications. For this reason, the development of a modular and user-friendly tool that includes this feature, takes a great interest.

III. FRAMEWORK DESCRIPTION

The simulation framework proposed in this work is focused on cooperative maneuvers among vehicles from a model-based approach. It is a generic simulation platform where the user can define the desired scenario using a graphical user interface, generate automatically a simulation model and then simulate it. The interface facilitates the insertion of controllers and models for different parts of vehicles and some elements of the infrastructure, through a models library. Furthermore, the framework permits to implement V2V and V2I communication links among the entities that comprise the scenario (real or simulated). The simulator also enables the 3D rendering of simulation and monitoring of several variables at run-time. The proposed framework uses hybrid simulation by combining real and virtual vehicles for studying their cooperative behavior without the need of real expensive equipment and vehicles. This tool also allows the recreation of experimental tests based on data sets stored from vehicle sensors.

The simulation framework was implemented in Matlab/Simulink. Although it is proprietary software, there is a wide research and development community for Matlab with an adequate level of introduction in the academic and industrial sector. From technical and computational viewpoint, this software offers many advantages for implementation with respect others. Currently it is widely used in a larger number of fields including rapid control prototyping for automotive applications. Due to the complexity of simulation models, other key advantage provided by Simulink is the possibility of building simulation models based on a graphical programming environment, which is easily interpreted by the user. The framework is structured in different components as shown in Fig. 1. This figure shows a functional diagram of the framework. Each of these parts is described below.

The main user interface, shown in Fig. 2, simplifies the definition of the scenario by enabling the user to customize all simulation parameters in a friendly environment. The most interesting features this GUI implements are the definition of vehicles configuration, communications links, infrastructures and 3D environment. Regarding to vehicles, controllers (longitudinal and lateral) and models for different vehicle
components as well as some elements of the infrastructure are selected through a blocks library. Moreover, the user can add new blocks and modify the existing ones individually. The library was developed in order to provide an easy integration of new models for any library section (vehicles dynamics, sensors, actuators, etc.). This library is structured as shown in Fig. 3.

The framework was designed so that each vehicle has a target in the simulation. In order to cover the maximum variety of simulation scenarios, the next four possible vehicle targets were defined: (i) complete a mission, (ii) follow another vehicle, (iii) follow a defined path or (iv) manual driving. The selected target sets the requirements for defining the vehicle in the simulation scenario. In the first case, complete a mission means that the vehicle is able to go from the starting point to the end point automatically, so the user must include autonomous driving controllers and specify the pose (position and orientation of the vehicle) at starting point as well as at the end point. The second option is suitable for simulating ACC (Adaptive Cruise Control) systems or CACC (Cooperative Adaptive Cruise Control) systems. In this case the target of the vehicle is to follow another vehicle so controllers for autonomous driving, the starting pose and the leader vehicle ID are required. In traffic simulation vehicles usually follow a defined path. For these cases, the third option is appropriate. The user is required to enter the path through the GUI. Finally, manual driving allows the user to control a simulated vehicle using a joystick. The vehicle definition based on these options provides a general way to design a simulation scenario containing vehicles with different targets and human interaction through manual driving of vehicles. As described in subsection C, the real-virtual vehicles interaction provides new features and extends the capabilities of the framework.

Once the simulation scenario has been defined in the main GUI, the tool automatically generates the Simulink model for the simulation and the 3D environment, based on the generation rules established in the code. In order to monitor the evolution of simulation variables and manual control of vehicles at run-time, a second GUI is also generated.

A general representation of the workflow with this tool is represented in Fig. 4.

A. Model structure

With the goal of facilitate the user understanding and interaction with the simulation environment, the model is generated following the structure shown in Fig. 5. This structure has four nested levels:

- The first level comprises only two interconnected blocks: One called Simulation, which includes the whole simulation model; and a second one that deals with the 3D visualization.

- At the second level, the infrastructure subsystem (if any) and vehicle subsystems are located. This level includes as many subsystems as vehicles were included during the configuration stage. Moreover, the connections between subsystems (vehicles or infrastructure) are done at this level. Current approaches for autonomous driving are frequently based on V2V or V2I communications. In both cases, information exchange between vehicles has a key role for the decision-making process. Variables such as location, velocity and longitudinal acceleration of the nearest vehicles are necessary to implement controllers for ACC or CACC.

- The third level includes the models of the different simulated systems. The diverse parts of the vehicle are modeled separately in order to obtain a detailed and parameterizable model. These are mainly: vehicle dynamics, sensors, actuators, controllers and environment. Besides these vehicle parts, some other blocks for signal adaptation are added. This is the case of signal processing for 3D visualization block. The organization of the vehicle model is shown in Fig. 6.

- At the fourth level the models of the different vehicle components such as sensors or actuators are located.
B. Graphic Environment

A main component of the framework is the 3D visualization of the simulation. The graphic environment comprises a 3D map and vehicles. The 3D modeling language employed for designing all 3D models used by the tool is Virtual Reality Modeling Language (VRML). VRML is a standardized language that aims at the visualization of three-dimensional interactive objects or scenes. VRML enables the visualization of a 3D scene composed of objects from prototypes based on basic geometric shapes or structures where specific properties are specified: the vertices, edges of each three-dimensional polygon and its surface color [14].

For the definition of the 3D world, the framework allows to obtain the 3D map from a geographical information system (GIS) and then use it for defining the simulation scenario. Our testbed scenario was created using OpenStreetMap (OSM) [15] as the source of the geographical environment and OSM2World [16] to create a 3D model from an OSM map. The latter employs 3D-related information, such as building height, in order to build a 3D model that can be exported to an obj-file. This is a commonly used geometry definition file format that is easily convertible to VRML. Regarding the vehicles, several car 3D models have been included for user selection. New vehicles 3D models can also be included by the user by following some guidelines in order to adapt their components to a common structure interpretable by the tool.

The 3D visualization features allow users to simulate scenarios influenced by frequent driving conditions, e.g. low visibility caused by adverse weather conditions such as fog, or different lighting situations such as day or night. This enables the simulation of sensors which are commonly used in the autonomous vehicles scope, e.g. image processing sensors for detecting lanes or obstacles, as well as infrastructure sensors for traffic management and surveillance.

C. Real-time and communications in co-simulation

Distributed applications in control system environments require tight synchronization in order to guarantee the delivery of control messages within defined message cycle times. Temporal aspects play a key role in communications and usually are the main constrains for this kind of applications. In the scope of driving control applications, timing requirements are in the order of milliseconds [17].

With the goal of achieve real-time simulation avoiding the possible delays related to 3D visualization, we decided to implement a scheme as the one presented in Fig. 8. As can be seen in the image, the system is composed by a Host PC, a Target PC, a V2I communications system and instrumented real vehicles. The role of the Host PC is to provide an interface between the Target PC and the user of the application. On the other hand, the Target PC is in charge of real-time simulation execution.

Target PC and V2I communications platform are connected via Ethernet using the UDP. In order to achieve real-time UDP communication, Target PC uses a dedicated Ethernet card for connect with V2I communications system.

In Host PC, the 3D visualization of the co-simulation is shown as well as the evolution of user-selected variables of vehicles. Also the user can interact with the simulation from the Host PC through the simulation control GUI in simulation run-time, by modifying parameters of the simulation model running on Target PC.

The communication protocols used among the framework components are Ethernet/UDP between computers and V2I communication system, and IEEE 802.11p protocol for V2I communication. Since Ethernet protocol is non-deterministic, the communication delays had to be monitored in order to obtain a value range. Typically, delays values were insignificant over a 100 Mbps Ethernet. Thus, potential delays have acceptable values for this application. The V2I communication system and its possible delays were analyzed and validated previously [18].

IV. Evaluation

In order to analyze the result obtained with the simulation platform and validate the simulation model of the vehicle, several experiments were carried out. In these experiments a real car was used. The parameters of the simulated vehicle were adjusted using the technical information of the real car.

The real vehicle is a Citroën C3 which has been adapted for autonomous driving, including several sensors such as GPS, Camera and inertial measurement unit (IMU); and actuators for controlling the throttle, brake, steering wheel and gearbox. The vehicle also counts with an On-Board Unit (OBU), which is in charge of the control of the vehicle and an IEEE 802.11p communication system for V2I and V2V communications [19].
The vehicle model used in the simulator includes a basic model of the engine, automatic gearbox, and vehicle dynamics, where friction and aerodynamic forces are considered. In the case of simulated vehicles, the sensors, controllers and actuators were simulated aiming to obtain similar behaviors between simulated and real vehicles.

The test scenario consisted on the cruise control of a car by actuating on vehicle pedals: throttle and brake. The longitudinal controller deployed in the real car was based on fuzzy logic. This controller has two inputs: speed and speed error; and only one output: the position of the pedal. Positive values of the output refer to the throttle while negative values refer to the brake [20].

With the goal of validate the vehicle model, the same controller was used on real and simulation tests. Fig. 9 shows the evolution of the vehicles speed as the cruise control reference is changed. For both cases, the vehicle response is similar. The main difference, at around second 12 of the test, is due to the gearboxes. While the simulated vehicle implements an automatic gearbox, the real car has an electronically-managed manual gearbox.

A comparison of the cruise controller output is shown in Fig. 9. As for the speed graph, one can appreciate that the response of the simulated vehicle is similar to the real one; therefore the vehicle model implemented on the simulation framework is adequate.

Once verified the correct behavior of the simulated vehicle model, another experiment was carried out. In this case, the testing scenario was composed of three vehicles. Two of them were simulated and the other was the real one used in previous experiment. The model used for the simulation of the two vehicles was the same as in the previous experiment in order to obtain a similar behavior of all vehicles for later comparisons.

The goal of this vehicle fleet was to drive by following the leader trajectory while performing an ACC. Thus, the individual goals of the vehicles were: (i) the vehicle leader (a simulated one) was following a predefined path. (ii) The second one (the real one) was following the first car. Finally (iii) the third car was following the second one. To that end, the vehicles implemented the same lateral and longitudinal controllers used on the 100-km experiment performed previously with two real cars [19].

Fig. 10 shows the paths of the three vehicles throughout the test. The results of the test demonstrated that it is possible the
cooperation between simulation and real vehicles. In Fig. 11 two pictures taken from inside the vehicle during the test are presented. These images include the simulated vehicles taken from the 3D visualization of the simulation at the same time that the pictures were taken.

V. Conclusions

This communication proposes a co-simulation framework for the development of applications on the autonomous driving field. The tool has a modular structure, allowing the user to implement several models and components at different levels of the simulation environment. Moreover, the implementation of V2V and V2I communication links allows the information exchange among the different entities; which is necessary for the development of cooperative applications between the different systems.

The simulator also enables the 3D rendering of simulation and monitoring of several variables at run-time. Therefore, the emulation of experimental tests based on data sets from the vehicle sensors serves as a powerful tool for designing and evaluation new ADAS.

As proven, the tool also makes possible real-time interaction between real and simulated cars allowing them to cooperate. Also this new functionality enable to use the simulation framework as a hardware abstraction layer between hardware and control systems. Ongoing research focus on the use of this feature for developing and testing Cyber-Physical Systems in the ITS scope.

Future work will be focused on improving models of vehicle, sensors, actuators and controllers as well as the simulation of wireless communication systems.

Acknowledgment

This work was supported by Spanish Ministry of Industry, Energy and Tourism by means of Research Grants ART-010000-2013-2, by the Spanish Ministry of Science and Innovation by means of Research Grants TRA2011-27712-C02-01 and TRA2011-14112-E and by the European Commission and in the ARTEMIS JU 333020 “Adaptive Cooperative Control of Urban Subsystems (ACCUS)”.

References


