Development of Full Speed Range ACC with SiVIC, a virtual platform for ADAS Prototyping, Test and Evaluation

Dominique Gruyer, Steve Pechberti, Sébastien Glaser

2013 IEEE Intelligent Vehicles Symposium (IV)
June 23-26, 2013, Gold Coast, Australia

Development of Full Speed Range ACC with SiVIC, a virtual platform for ADAS Prototyping, Test and Evaluation

D. Gruyer *IEEE member*, S. Pechberti, S. Glaser

***Abstract*— LIVIC-IFSTTAR develops driving assistance services in order to improve the driving safety. These systems are tested on several real prototypes equipped with sensors and perception, decision and control modules. But tests on real prototypes are not always available, effectively some hardware architectures could be too expensive to implement, scenario may lead to hazardous situations. Moreover, lots of reasons could lead to the inability to obtain both sensors and ground truth data for ADAS evaluation. However, safety applications must be tested in order to guaranty their reliability. For this task, simulation appears as a good alternative to the real prototyping and testing stages. In this context, the simulation must provide the same opportunities as reality, by providing all the necessary data to develop and to prototype different types of ADAS based on local or extended environment perception. The sensor data provided by simulation must be as noised and imperfect as those obtained with real sensors. To address this issue, the SiVIC platform has been developed; it provides a virtual road environment including realistic dynamic models of mobile entities (vehicles), realistic sensors, and sensors for ground truth. To test real embedded applications, an interconnection has been developed between SiVIC and third party applications (ie. RTMaps). In this way, the prototyped application can be directly embedded in real prototypes in order to test it in real conditions. A Full Speed Range ACC application is presented in this paper to illustrate the capabilities and the functionalities of this virtual platform.**

I. Introduction

SiVIC platform has initially been developed to provide simulated sensors data as noised and imperfect as those provided by real embedded sensors in real conditions, mainly optical sensors outputs. Over the time, and thanks to the modular architecture of the SiVIC platform, new features have been added to provide a set of other functionalities (management of events, resources exchanges, handle of meta materials, …). Currently, SiVIC is able to simulate vehicle dynamics and a large variety of sensors (cameras, GPS, odometer, lidar, radar, ...). Its goal is to reproduce a situation as realistic as possible. The generated simulation data can be recorded by third party application like RTMaps (vehicle's data acquisition system). RTMaps platform manages, time-stamps and synchronizes all the numerical data coming from SiVIC sensors (cameras, GPS, laser scanners, inertial navigation equipments, odometers, and so on). Then, it is possible to faithfully replay the scenario previously saved. It implies that real prototypes (vehicles and embedded architectures) are not necessary in the early stages of the research & development program (initial costs reduction).

As SiVIC also integrates physical car models, it is possible to prototype a full control/command software loop (SIL), using the third party application like RTMaps platform. SiVIC providing complex dynamic vehicle and realistic sensor models, the need of an equipped vehicle for the first stages of the prototyping cycle is not necessary anymore.

SiVIC platform is not the only existing simulator to address the transportation safety issues, other simulators exist but have different perspectives.

It is the case of Racer (www.racer.nl) which is a driving simulator with a very realistic vehicle model. But Racer remains a monolithic software dedicated to the vehicle dynamic modelling. Unlike SiVIC, Racer does not provide complex and realistic sensor modelling, it is not able to integrate during the simulation new features (dynamic loading of classes), or acting on the scenario during the simulation by means of a script language manager. It cannot be used to evaluate and validate embedded algorithms. Many other simulators are dedicated to model the drivers' behaviour or to simulate the road traffic (SIM², ARCHISIM, SCANeR). In this second type of simulators, there is no easy possibilities of ADAS integration and evaluation such as emergency braking, lateral or longitudinal control systems (Stop&Go).

Moreover, a last type of virtual platform is dedicated to sensor simulation. Unfortunately, these ones often simulate only one type of sensor (RADAR, GPS, Telecommunication, Camera, IR Camera). The only one comparative sensors simulation platform is the PreScan platform from TASS. Nevertheless, PreScan software mainly provides basic and perfect sensors.

In order to anticipate the developments of future driving assistance systems, the SiVIC platform already integrates new types of sensors such as multi-plane laser scanner or cooperative Vehicle to Vehicle and Vehicle to Infrastructure (V2V and V2I) sensors.

In this paper, we present the capabilities of this platform for fast and easy prototyping, testing and evaluation of Advanced Driving Assistance Systems (detection and tracking of pedestrian and car, ...) and Partially Autonomous Driving Assistance Systems (ACC, Stop&Go, ...).

In the next section, the advantages of such a platform for application prototyping are presented. Then sub parts of this section address the functionalities in term of sensors and dynamic vehicle modelling.

Part 3 demonstrates how to obtain a complete virtual prototyping architecture with interconnection mechanism

Dominique Gruyer, Sébastien Glaser, Steve Pechberti, are with IFSTTAR-COSYS - LIVIC, 14 route de la minière, bat. 824, 78000 Versailles-Satory, France (phone: (+33) 1 40 43 29 07;e-mail: dominique.gruyer@ifsttar.fr, sebastien.glaser@ifsttar.fr, steve.pechberti@ifsttar.fr

978-1-4673-2754-1/13/$31.00 ©2013 IEEE

between SiVIC and third party application (^{RT}Maps or Matlab platforms).

Part 4 is devoted to the development of ADAS application. The presented application has been carried out for software in the loop devices. In this application, a perception module with laser scanner and dense stereovision processing will be used in order to provide targets localization for a Full Speed Range ACC application. Then, a conclusion will be given and some future works will be proposed.

II. SIVIC, A VIRTUAL ALTERNATIVE FOR APPLICATIONS PROTOTYPING

A. The SiVIC platform

Many developments in research aim to improve the safety of road environments through driving assistance systems. These studies generally take into account an ego vehicle perception and the corresponding reaction of the vehicle (e.g. braking and accelerating), as for instance with ABS (anti-block system) and ESP (electronic stability program). However, in many situations an ego perception is no longer sufficient. Additional information is needed to minimize risk and maximize the security of driving, as with Stop&Go, AEB (Autonomous Emergency Braking), ... This additional information requires additional resources which is both time-consuming and expensive.

It therefore becomes essential to have a simulation environment allowing prototyping and evaluating extended, enriched and cooperative driving assistance systems in the early stage of the system design. A virtual simulation platform has to integrate models of road environments, virtual embedded sensors (proprioceptive, exteroceptive), virtual sensors on the infrastructure and communicating devices, according to the laws of physics. Also the vehicle has a physics-based model of its dynamic behaviour coupling with actuators (steering wheel angle, torques on each wheel). SiVIC meets these requirements and is therefore a very efficient tool to develop and prototype a high level autonomous driving system with cooperative and extended environment perception.

Actually, the SiVIC software platform represents a real benefit for fast design and offers a valuable support to the development of cooperative systems such as V2V and V2I and the assessment of the performance and reliability of such systems (figure 1).

Simulation platforms such as SiVIC give several considerable advantages in the classical V development cycle and provide in fact a new development cycle called W (figure 2). Firstly it opens up to an unlimited choice of scenarios in the "requirements and feasibility analysis phase". The development is not bounded by the existing equipment or infrastructure; complex urban scenarios can for example be proposed. Collision avoidance and lane keeping systems can easily be tested to their limits, without the need for additional safety measurements. The scenarios can easily be updated and changed. The "coding phase" also benefits from the simulation tool, because realistic inputs are available for every algorithm from the very beginning of the development. The "unit tests", "integration tests" and "validation process" gain considerably in objectivity, as reference scenarios can easily and precisely be reproduced. The general advantage is that the costly and time-consuming field tests are reduced and pushed to the end of the development cycle.

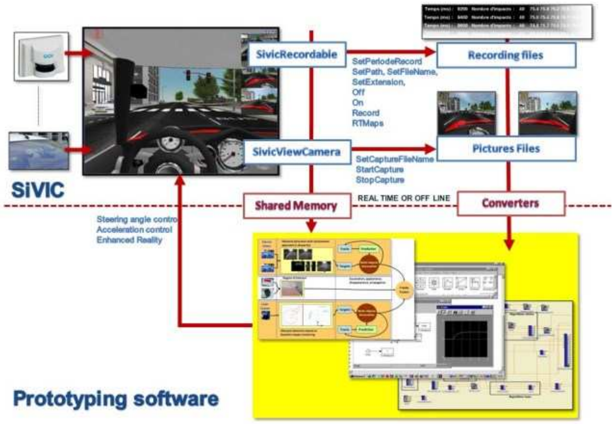

Figure 1: SiVIC, general principle.

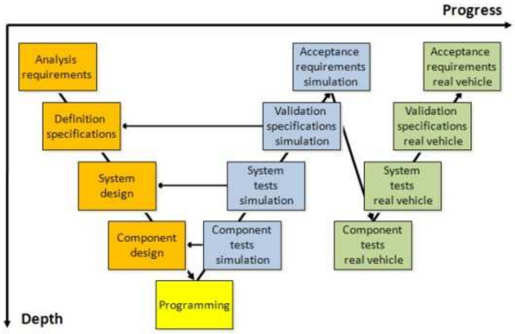
Progress
Analysis requirements
Definition specifications
System design
Component design
Programming
Component tests simulation
System tests simulation
Validation specifications simulation
Acceptance requirements simulation
Component tests real vehicle
System tests real vehicle
Validation specifications real vehicle
Acceptance requirements real vehicle
Depth

Figure 2: The W development cycle

B. The simulation engine dedicated to sensors modeling.

The focus of the simulator being to simulate multiple sensors in the same environment, techniques have been developed to allow to define different rendering methods. This so-called "multi-rendering" function allows to scale computation accuracy as well as to simulate different sensor technologies. As an example, it is possible to define such a renderer for basic optical rendering (for screen display or scene overview), Radar propagation modelling, GPS propagation, or more advanced optical modelling (which includes features such as High Dynamic Range simulation, texturing, shadowing, and processing filter addition). Another advantage of this approach is the ability of quickly developing renderers adapted to specific metrics rather than limiting to sensor view of the scene, enabling to finely monitor the simulation.

At the present time, two rendering methods are available to perform the optical signal rendering. The first method ("mgFastRenderer") provides a classical 3D rendering, using various approximations to improve computation efficiency. The second method ("mgAccurateRenderer") is focused on much more accurate light simulation, meaning multiple dynamic lights can be introduced with dynamic shadowing of objects.

Among other features available, we may underline optical reflections on surfaces (planar reflection and cube-map reflection), rain and fog simulation. Additionally, several post-processing filters are available to model sensor effects and defects (glow, blur, exposure, focus and exposure control).

More generally, an object animation system (based on interpolation) and an event management engine allow to simulate complex scenarios. A generic visibility layer system may be used to select visibility of objects that is specific to certain sensors.

C. Dynamic Plug-ins for transportation application prototyping

SiVIC relies on the graphical engine described above for the graphical and physical rendering stages. It also has a dynamic loader for the many external modules which simulate all the actors of a road situation. In order to have access to all parameters of the sensors and the vehicles, the communication protocol uses the same rules that the ones used in the graphical engine. Thus the mechanism used for the communications protocol is modular and is distributed on all the SiVIC modules.

In order to reproduce a coherent situation with the reality and to be able to generate all the data coming from the sensors embedded on LIVIC's prototype vehicles, a set of proprioceptive sensors as well as exteroceptive sensors are modelled and developed inside SiVIC. Currently, these sensors plug-ins are:

- A camera which simulates a set of different cameras. This one is configurable either by using the traditional parameters of a camera (size of matrix, focal distance...) or by using the parameters related to an OpenGL field of view (fovy, Znear, Zfar and ratio).
- An Inertial Navigation System which simulates a sensor of inertial type including the accelerometers and the 3-axis gyro meters. Two types of INS sensor are available. The first one is embedded in a vehicle and takes into account the dynamic parameters of the vehicle. The second model is applied to objects without dynamic modelling.
- An odometer which simulates the optical coder type providing the distance covered (curvilinear X-coordinate) by a vehicle.
- A telemetric scanner which simulates the working of a laser scanner. In order to carry out this simulation, two methods were implemented. The first one uses the ray tracing method. Unfortunately, this one takes a lot of computer time resources. The second one is a simpler method using the depth matrix (Z-Buffer) in order to emulate the distances of the impacts for each beam of the laser scanner. This second method is much faster.
- A beacon which models transponders. This sensor is composed of two types of module. Firstly, a transmitter takes into account the data frame to be sent and the communication range. Secondly, a receiver, attached and embedded into a vehicle, receives the frame from the transmitter. It is indeed possible to manage simultaneously several transmitters with different messages and several receivers embedded in the vehicle.
- A Radar which simulates electromagnetic sensor. Several levels of modelling are available. A first level is based on a specific processing of the Z-buffer with some modifications in order to obtain similar targets than radar. Level 2 is based on the use of RCS, a ray tracing mechanism, and a antenna modelling in order to build a radar analogical signal. The last level is an innovative modelling based on GPU use in order to allow a fast computation and highly noised returned signal.

In order to provide an accurate reference and a ground truth, a set of accurate observers are implemented. Four types of reference sensors have been developed. The car observer, the pedestrian observer, the object observer and finally the road observer (data provided at the vehicle location). In addition to these observers, the use of layer mechanism allows to generate a labelling of the obstacle for the validation of detection algorithms based on optical sensors.

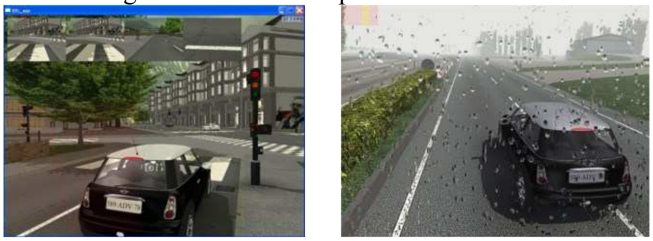

Figure 3: SiVIC platform: Multi-sensors management in a complex environment, and degraded weather conditions.

The camera model uses the rendering functionalities provided by the graphical engine. Among the most useful functionalities, the mechanism of filter addition is probably the most used. Effectively, it is possible to add several existing filters during the simulation process in order to tune the picture rendering and then produce a realistic image. These filters include noise, optical distortion, depth of field, glow, fog, rain drops, rain fall, auto exposure and auto focus (figure 3 and 4). It has been demonstrated in parallel work that combination of such filters can be sufficient to match real camera behaviour with good accuracy (see paper [5]).

A. Complex vehicle modeling

The output of virtual sensor in a given situation is not only defined by sensor and environment characteristics, but also by vehicle behavior (this is even more obvious when we consider closed loop simulation). So, to provide realistic enough sensor data, different constraints on the vehicle model appears.

It was particularly necessary that the vehicle’s movement could be reproduced through rotation around all three axis, and includes the effect of shock absorbers. The vehicle model is based on works of S. Glaser [6] (see also e.g. [7]).

The vehicle body has 6 degrees of freedom (3 in translation and 3 in rotation). Each wheel also has 3 degrees of freedom (vertical movement, wheel rotation and steering). The link between the wheel and the vehicle body includes shock absorber. The tire road forces are described using non linear coupled forces [8], [9].

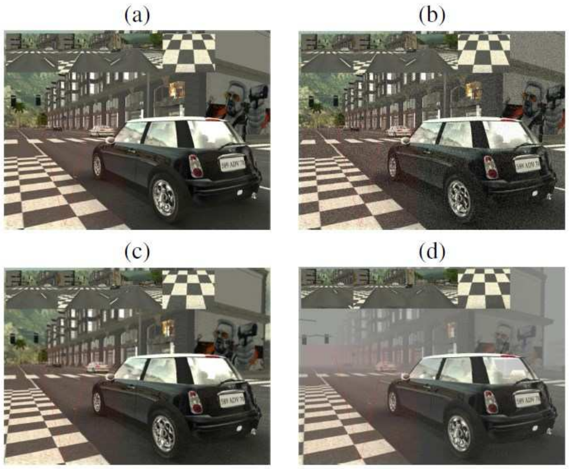

Figure 4: multi-rendering and filters mechanism (a: initial image, b: noise, c: noise and depth of field, d: noise, DoF and fog).

Figure 5 shows the level of complexity of the model and the links between all the different physical modules involved in this model. Each module is completed with a list of available parameters. These parameters can be modified during the simulation stage. Vehicles can be controlled longitudinally by torques on the wheels and laterally by the steering wheel angle; controls are decoupled.

In order to obtain the best sensor data comparatively to a real situation, it is necessary both to handle a dynamical model of the vehicle and to simulate realistic models of actuators. The current actuators implemented in SiVIC are the motor or braking torques applied on each wheel and the steering wheel angle. It is then possible to simulate front wheel drive, rear wheel drive and four wheels drive.

Moreover, in order to manage a basic traffic situation, several modes are available. The first one is the control of a vehicle by a human driver. The second mode is done for basic traffic modeling: the vehicle follows a given trajectory without vehicle dynamic consideration. The third mode is also a trajectory following mode, but following a desired trajectory is achieved by use of lateral and longitudinal controllers. In this third mode, the vehicle dynamic is taken into account. The last mode is dedicated to the control of vehicles with inputs coming from third applications (RTMaps, Matlab, ...).

The first approach is an optimized shared memory initially developed to carry out the interconnection of SiVIC and RTMaps. This first data exchange mechanism is dedicated to the management of data coming from sensors and orders for actuators.

III. A DISTRIBUTED SOLUTION FOR COMPLEX PROTOTYPING

Three mechanisms are available in SiVIC in order to share information between SiVIC instances, or SiVIC and third party application in mono or multiple computers solutions.

A new complementary solution (DDS: Distributed Data Storage) was proposed. It relies on a data distributed architecture in which SiVIC instances share state vector objects [10]. This mechanism manages the publication and subscription to raw data frames (octet arrays) of resources (object position and current state, events, sensor data, ground truth, weather conditions ...).

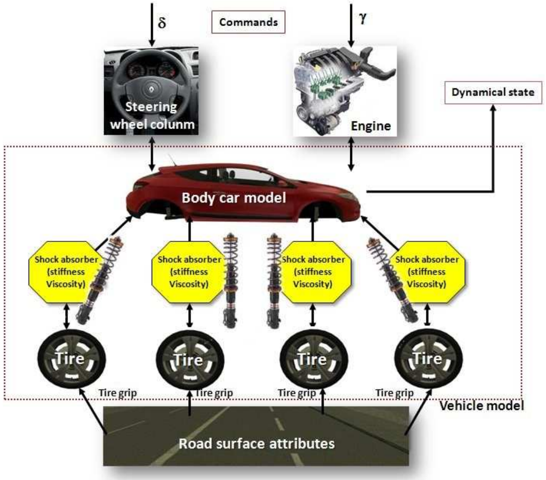

Figure 5: Vehicle model in the SiVIC with its attributes.

A DDS data is identified by an IP address or a DNS alias of the computer which publishes it and by a generic name for these data. The general architecture is presented in the following scheme (figure 6).

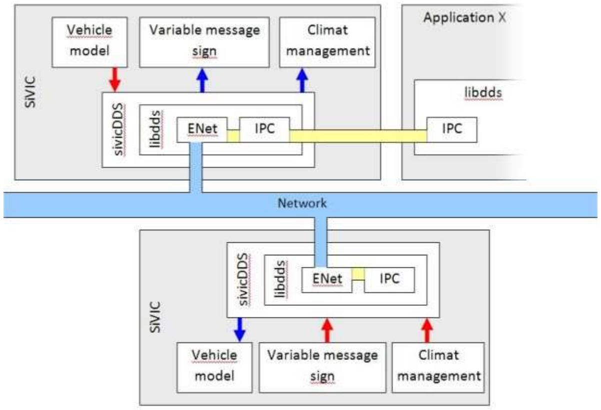

Figure 6: Distributed architecture: DDS bus

The red arrow indicates the publication of data on DDS, and the blue arrow represents the consumption of shared data.

A third mechanism (SensORB: Sensors Object Request Broker) has been developed in order to manage both raw data flow (sensor outputs) and object properties (state vector) requests between several applications or computers. This third approach can be seen, for raw data flow mechanism, as a virtual CAN bus.

A current work in progress has for main task to standardize and to merge these 3 data exchange modes in an only one generic communication bus.

IV. APPLICATIONS AND PROJECTS

The application that is presented here is the Full Speed Range ACC [12](FSRACC). The aim of this application is to control the vehicle speed accordingly with the driver's desired speed and the environment, namely a leading vehicle whose speed is lower than the driver's desired speed.

The scenario, in which the application has been tested, takes place on the Satory's test track in Versailles - France. To obtain a high quality 3D immersion for the real drivers in the virtual environment, a realistic 3D reproduction of the real track was produced by LIVIC. Figure 7 shows the similarity of this simulation in comparison with the same perspective on the real track. In this figure, the first line provides virtual renderings and the second line gives the real pictures from the same point of view.

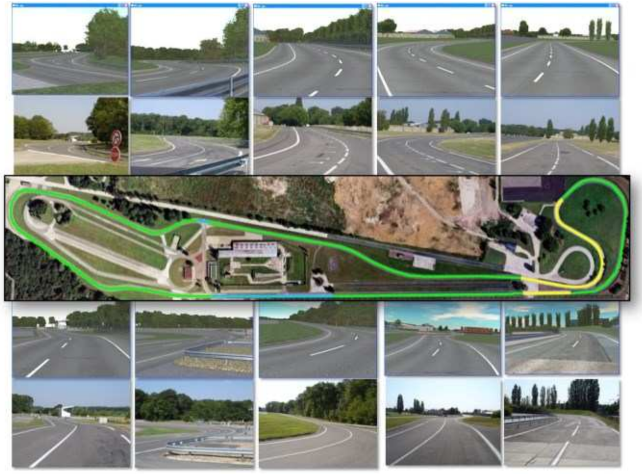

Figure 7: Simulation of the Satory's track in the SiVIC.

In this application, the vehicle needs several sensors in order to identify possible targets and to measure the relative distance. FSRACC systems commonly rely on a perception module to assess the relative distance. In order to make it more robust, a merging of lidar and stereovision is done. The general architecture of the application is shown in Figure 8.

At the lowest level, we define the environment and the scenario: the roads, the environment and the actors are managed by SiVIC. Next each vehicle could be defined separately (in SiVIC), in term of hosted sensors, actuators. They can be a passive actor, following a defined trajectory or an active actor, reacting accordingly with its environment, its sensing capacity, and the decision and control system.

In this last case, the dataflow from the vehicle sensing capacity (sensors, communication, ...) is transmitted to the algorithms layer which is here handled by RTMaps. Under RTMaps, each dataflow is seen as the corresponding sensor.

In figure 9, we can see an overview of the application under SiVIC and the sensors representation and the algorithm under RTMaps.

Figure 12 has a closer look at the obstacle detection and target tracking algorithms: on the left side we can see the view of one of the camera from SiVIC, with an overlay containing obstacle's bounding boxes and distance information and on the right side the resulting detection.

Finally, the figure 10 shows the interdistance between the ego vehicle and the leader vehicle, using the stereovision sensor. The feedback is then generated in this software environment and transmitted back to the actuators of the considered vehicle in SiVIC.

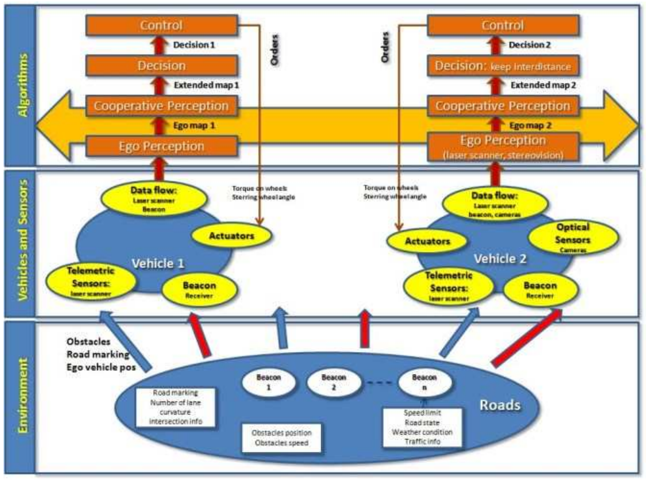

Figure 8: General diagram of the FSRACC application.

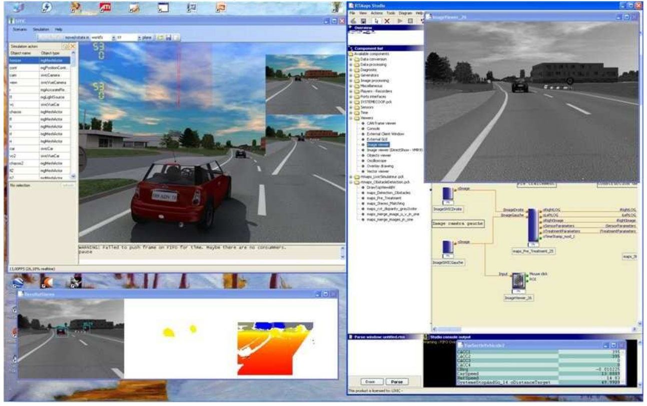

Figure 9: Overview of the implemented application with SiVIC vehicles and sensors, and RTMaps packages.

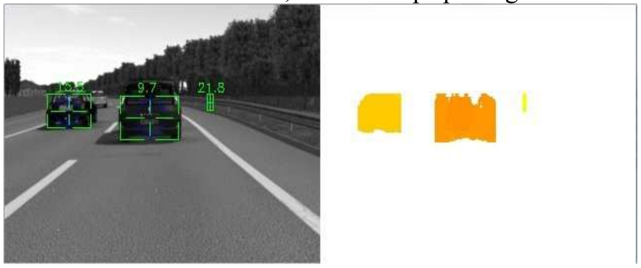

Figure 10: Obstacle detection with dense stereovision and target tracking (left: target tracking, right: disparity map).

To simulate the full system, we suppose that the front vehicle must follow a speed profile defined by the infrastructure and transmitted to the leader using a transponder system. This speed profile is defined in the figure 12 and presents large variation from 6m/s to 19m/s.

As the lead vehicle does not have any other constraints, its speed is only limited by this speed profile (figure 12). The lead vehicle speed can be seen in figure 13 (black). The ego vehicle speed is limited either by a driver's speed or by the lead vehicle speed. In our case, it is this second limit that is the lowest. Given the perception system described before, the decision and control algorithm can regulate the vehicle inter distance and the relative speed. The resulting speed profile is shown in figure 13 (green).

In this example, we have simulated the full driving assistance system, from the sensing to the control tasks.

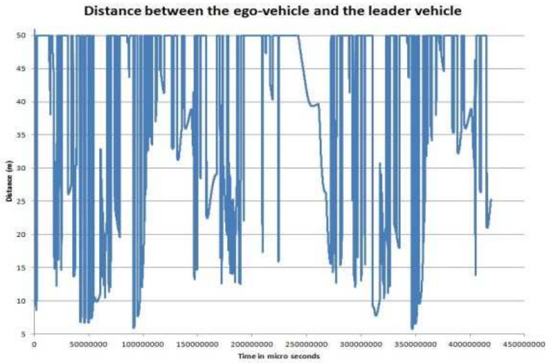

Figure 11: Inter distance between vehicles.

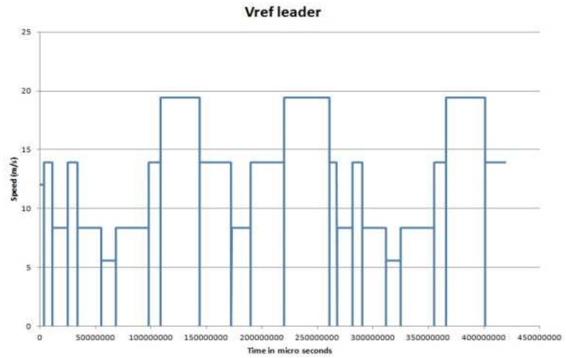

Figure 12: Speed references from road side unit (Beacons)

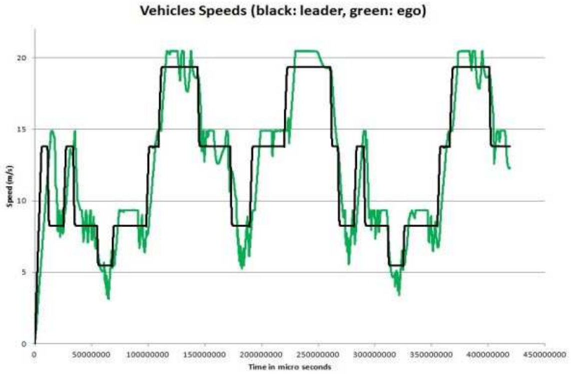

Figure 13: Speed profiles for leader and follower vehicles.

V. Conclusion

The platform SiVIC has been developed to a state where it appears suitable and fit for different ADAS applications such as ACC, Stop & Go, by reproducing in a very realistic manner the behaviour of the vehicle, embedded sensors and environment.

Various works have been carried out to demonstrate the efficiency of the camera models and post processing stages that can address most common camera defects (which could be optical, sensor, electronical defects). A module of precipitation simulation (rain, snow...) with effects on the light reflection is also integrated and usable.

More importantly than its graphical capabilities, SiVIC allows to test and to validate perception and control algorithms. This feature is one of the most significant advantages of the platform. Indeed, this enables to test and evaluate in real time and under extreme conditions the innovating approaches of obstacle and road marking detection and of vehicle control.

On the the SiVIC platform, methods have been developed to reset simulation status, while giving access to object property modifications, and to launch particular events (according to time, object position, …) automatically during scenarios. This ensures the control over scenario repeatability, enabling validation process by performing simulations on multiples instances of scenarios that are each time slightly different.

While allowing to build realistic driving scenarios and improving the validation process, the SiVIC platform may also be seen as an education and communication tool that may demonstrate the objectives, methods, solution and result, even when the system may not have been physically developed.

The platform SiVIC has been made available for research and industrial usage under the name Pro-SiVIC, which is developed and distributed by the company CIVITEC. Major graphical user interface developments have been carried out to facilitate scenario development and data generation, and several scenario assessment state tools have been developed to monitor simulation results. Upcoming work includes sensor model enrichment, and continuing the validation of the simulation process through industrial applications.

References

[1] B. Steux, « RTMaps, un environnement logiciel dédié à la conception d'applications embarquées temps-réel. Utilisation pour la détection automatique de véhicules par fusion radar/vision », Thèse de doctorat de l'Ecole des Mines de Paris, décembre 2001.

[2] S. Glaser « Modélisation et analyse d'un véhicule en trajectoires limites. Application au développement de systèmes d'aide à la conduite », Thèse de doctorat de l'Université d'Evry, janvier 2004.

[3] D. Gruyer, C. Royere, N. du Lac, G. Michel and J.-M. Blosseville, "SiVIC and RTMaps, interconnected platforms for the conception and the evaluation of driving assistance systems", ITSC'06, october 2006, London.

[4] D. Gruyer, S. Glaser, B. Monnier, "SiVIC, a virtual platform for ADAS and PADAS prototyping, test and evaluation", in proceeding of FISITA'10, Budapest, Hungary, 30 may-4 june 2010

[5] D. Gruyer, M. Grapinet, P. Desouza, "Modeling and validation of a new generic virtual optical sensor for ADAS prototyping", in IV 2012

[6] S. Glaser, 'Modélisation et analyse d'un véhicule en trajectoires limites.Application au développement d'un système d'aide à la conduite', Phd Thesis, Université d'Evry Val d'Essonne, 2004

[7] N. Kiencke, L. Nielsen, 'Automotive Control Systems: For Engine, Driveline and Vehicle', Springer Verlag, 2004

[8] H. Dugoff, P.S. Fancher, L. Segel, 'An analysis of tire traction properties and their influence on vehicle dynamic performance', in SAE, (700377),1970.

[9] R. Rajamani, "Vehicle Dynamics and Control", in Springer, USA, 2006

[10] D. Gruyer, S. Glaser, R. Gallen, S. Pechberti, N. Hautiere, "Distributed Simulation Architecture for the Design of Cooperative ADAS", in FAST-ZERO (Future Active Safety Technology) 2011, Tokyo, Japan, September 5-9, 2011

[11] B. Steux, « RTMaps, un environnement logiciel dédié à la conception d'applications embarquées temps-réel. Utilisation pour la détection automatique de véhicules par fusion radar/vision », PhD. thesis from Ecole des Mines de Paris, december 2001.

[12] ISO 22179:2009, Intelligent transport system, Full speed range adaptive cruise control system, performance requirements and test procedures.