2018 IEEE Intelligent Vehicles Symposium (IV)
Changshu, Suzhou, China, June 26-30, 2018

X-in-the-loop advanced driving simulation platform for the design, development, testing and validation of ADAS

Sikandar Moten
Simulation and Test Solutions
Siemens PLM Software
Leuven, Belgium
sikandar.moten@siemens.com

Francesco Celiberti
Simulation and Test Solutions
Siemens PLM Software
Leuven, Belgium
francesco.celiberti@siemens.com

Marco Grottoli
Simulation and Test Solutions
Siemens PLM Software
Leuven, Belgium
marco.grottoli@siemens.com

Anne van der Heide
Mechanical Engineering
Delft University of Technology
Delft, The Netherlands
a.vanderheide@student.tudelft.nl

Yves Lemmens
Simulation and Test Solutions
Siemens PLM Software
Leuven, Belgium
yves.lemmens@siemens.com

***Abstract*— This paper presents a X-in-the-loop (where X: Model, Software, Hardware, Driver/Human, etc.) driving simulation platform, developed at Siemens PLM Software, that facilitates the design, development, testing and validation of Advanced Driver Assistance System (ADAS). The paper outlines the essential components of the simulator and demonstrates the usefulness by two autonomous driving functionalities i.e. adaptive cruise control and autonomous intersection crossing. In addition, the paper highlights the key features of the platform.**

Keywords*— *X-in-the-loop simulation, Advanced Driver Assistance System, Autonomous Driving Functionality, Driving simulator.

I. Introduction

A. General Background

In order to enhance the performance of the future vehicles in terms of safety, comfort and environmental friendliness, automotive manufacturers are continuously striving for increased automation. The Society of Automotive Engineers (SAE) have categorized the levels of driving automation for on-road vehicles into 6 different levels starting from no automation at Level 0 to fully autonomous vehicle at Level 5 [1]. However, the design, development, testing and validation of Advanced Driver Assistance System (ADAS) are not trivial in nature and require enormous time and effort resources. For this reason, in view of different level of automation, there is a dire need of a modular and scalable platform that allows X-in-the-loop simulation. By using the virtual verification and validation environment, the scientific and engineering community can significantly reduce the overall test execution time which ultimately effects time-to-market for a newly designed prototype. Reduction in the actual tests also reduces CO_2 emissions. Moreover, virtual simulation platform can help to increase the reproducibility of the different scenarios, which is very difficult to achieve otherwise in real life.

To this end, at Siemens Product Lifecycle Management (PLM) Software, a simulation platform has been developed that is not only capable of model, software, hardware and driver/human-in-the-loop simulation but also provides a highly realistic driving experience to the human/driver. The current paper is an extension of the previous work presented on the driving simulator [2].

B. Overview of the paper

In what follows, a X-in-the-loop driving simulation platform is described that can be used for the design, development, testing and validation of ADAS. To this end, Section II describes the system architecture and essential components of the simulation platform. Next, in order to demonstrate the usefulness of the developed platform, Section III demonstrate two autonomous driving functionalities namely, adaptive cruise control and autonomous intersection crossing. Overall conclusion and recommendations are presented in Section IV of the paper.

II. Components of the XiL Driving Simulation Platform

The complete simulation platform comprises of several key components. These include but not limited to the real-time models of the vehicle and its critical sub-systems capable of simulating vehicle behavior, virtual environment that can interact with vehicle model, control algorithms, sensor models, and communication models, and a real-time co-simulation platform for deterministic computation. Moreover, an appropriate tactile, motion, acoustic and visual feedback is crucial for immersive driver-in-the-loop capabilities. Last but not least, appropriate hardware is necessary both for manual driving mode and autonomous driving mode (e.g. human machine interfacing or HMI). The system architecture for the simulation platform is shown in Figure 1. In what follows, various components highlighted above are discussed in detail.

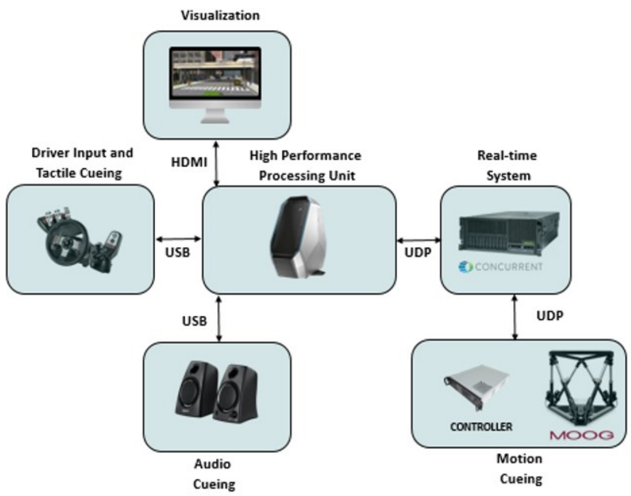

Figure 1: System architecture for the driving simulation platform

A. Real-time Multi-fidelity Models

Mathematical models are widely used in almost every discipline of science and engineering, and therefore, the field of cyber-physical systems in general and autonomous vehicles in particular is no exception. Usually, the fidelity of the model dictates the quality of the simulation [3]. In order to virtually test and validate the developed algorithms and technologies for automated driver functionalities, for a wide number of scenarios, a representative/realistic model of the vehicle or a digital twin is required which is capable to capture both the longitudinal and lateral dynamics. Therefore, it is decided to model the vehicle by using a multi-body modelling approach. To this end, Simcenter Motion is used to analyse the dynamic behavior of the vehicle. In addition, a lumped parameter approach is used to model the other critical sub-systems of the vehicle, which do not require a formal description of the geometry. The individual models are coupled by means of energy relations that are independent of their physical domain i.e. bond graph method. This means that interfacing is realised by using effort and flow variables that uniquely define the power for the coupling, see [4] and [5]. Simcenter AMESim provides a platform for modelling and analysis of physical multi-domain systems. This platform is used for the modelling of the critical sub-system governed by ordinary differential equation ODE or differential algebraic equations DAE. i.e. Powertrain with internal combustion engine (PT with ICE), Electric Assisted Power Steering (EPS), and Anti-lock braking system (ABS) with Electronic Stability Program (ESP). In what follows, a brief description of these individual models are outlined.

a. Multibody model of a vehicle

In order to model the realistic dynamic behaviour of the vehicle, such as ride and handling characteristics, a multibody modelling approach is proposed. A extensive survey for the (flexible-) multi-body modelling approach is presented in [6] and [7]. This implies that the complete vehicle behaviour can be modelled by interconnected rigid or flexible bodies, each of which may undergo large translational and rotational displacements. However, deriving the equation of motion manually for three-dimensional motion involving multiple bodies is tedious and cumbersome. For this reason, the model of the vehicle is developed, as shown in Figure 2, by using a template based Siemens Driving Dynamics tool and comprises the body of the vehicle, engine body, driveline, suspension, stabilizer, steering wheels, and other subsystem interconnected with each other to simulate the realistic dynamic behaviour of the vehicle.

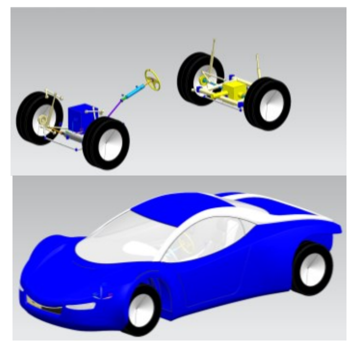

Figure 2: Multibody model of the vehicle

b. Powertrain with Internal Combustion Engine

Next, a powertrain mechanism with internal combustion engine, shown in Figure 3, is developed. The complete model of powertrain comprises an internal combustion engine with electronic control unit for regulation, a manual gearbox including clutch, a differential for rear-axle with engine feedback. The model of ICE is capable to compute the engine torque, fuel consumption, thermal loses, exhaust gas flow rate and emissions such as CO, NOx, HC and soot. The model of ICE can be appended with any drivetrain. In this case, it will concatenated with a manual gear box which will ultimately transmits the power to the differential for rear axle.

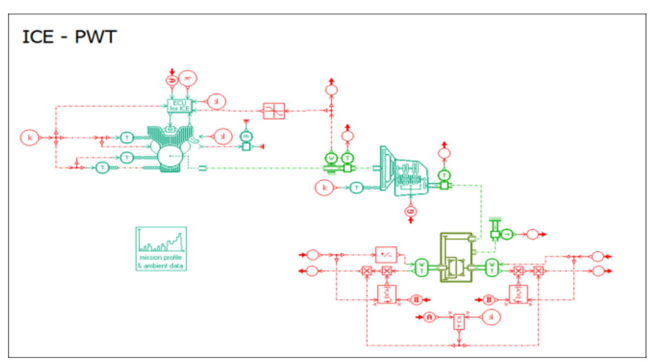

Figure 3: PT with ICE model

c. Anti-lock Braking System / Electronic Stability Program

Safety of the vehicle (consequently, the safety of drivers, passengers, other road actors and infrastructure) is always the priority. To this end, modern vehicle are equipped with state-of-the-art automobile safety systems such as antilock braking system, antiskid regulation (ASR) and electronic stability program. In this work, a lumped parameter approach is used to develop these safety systems. The ESP improves vehicle handling and control in critical situations. This is achieved by applying appropriate braking forces to each wheel of the vehicle. Whereas, as the names suggest, ABS and ASR allow the wheels to preserve tractive contact with the road while braking by preventing the wheels from locking and slipping. In conjunction with the ABS and ASR, ESP (as shown in Figure 4) improves vehicle stability and control and consequently make the vehicle safer to drive.

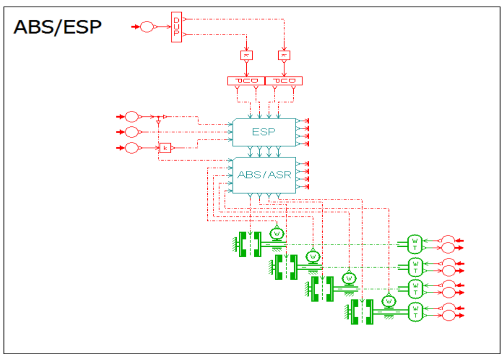

Figure 4: ABS/ASR and ESR model

d. Electric Power Steering (EPS)

In order to reduce driver's fatigue, modern vehicles are equipped with power steering systems. These systems help the driver by providing assistance in steering. Such assistance can be provided by a hydraulic, an electro-hydraulic or just an electric system. As electric systems require less number of auxiliary components (belt driven mechanism, hydraulic pump etc.), are safer and cleaner as compared to hydraulic counterpart, an electric power assisted steering is modeled, shown in Figure 5.

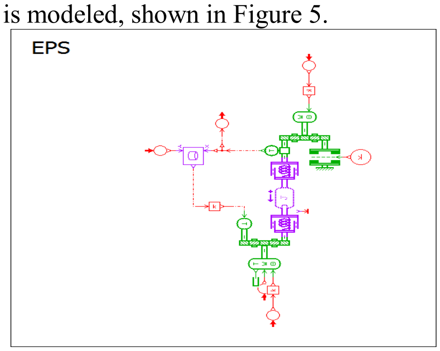

Figure 5: EPS model

B. Virtual environment, sensor and communication models

In order to simulate or test an ADAS functionality with driver-in-the-loop capabilities, the modeling of the virtual world is indispensable, see Figure 6. To this end, a physics based simulation solution Prescan® can help to design scenarios (e.g. ground plan, road conditions and infrastructure etc.), model the environment (e.g. pedestrians, other vehicles/obstacles/objects and weather modules), sensors (such as radar, laser/LIDAR, camera and GPS) and communication technologies (for instance vehicle-to-vehicle and vehicle-to-infrastructure) for evaluating Advanced Driver Assistance Systems (ADAS). In this paper, different scenario, sensor models and communication methods are used for the demonstration purposes and will be outlined in the sequel.

C. Automated driving functionality

In order to perform the closed-loop simulation, besides the vehicle model and other simulation components, an appropriate driving functionality is also required. In this paper, two autonomous driving functionalities (ADF) are investigated and implemented for the demonstration purposes. Although the existing state of these controllers are at an infancy level, in what follows, a brief description is outlined.

Figure 6: Modeling of the virtual world, Top: A 3rd person view, Bottom: Driver's view

a. Adaptive Cruise Control (ACC)

The first ADF investigated and used on the simulator for demonstration is Adaptive Cruise Control (ACC). It is an automatic control that helps the driver to keep a safe distance with the respect to a leading car. The aim of the control is to avoid collision by automatically releasing the throttle and/or actuating the brake pedal of the EGO vehicle (i.e. car with autonomous functionality). The actions taken by the ACC control algorithm are based upon the information given by two scanning sensors modeled in PreScan: long range radar (LRR) 150 m range with narrow beam of 9 degrees and short range radar (SRR) with 30 m detecting range and a wide beam of 80 degrees. The sensors continuously evaluates the presence of a lead car. In the case a slower car gets inside the sensor measuring range, the control procedure computes the relative distance and speed between the two vehicles. The brake system is triggered once the relative distance and relative speed between the EGO and lead car goes over a pre-defined value. In case the relative speed is around zero, the control will release the throttle pedal. In the case when safe distance is violated and the cars relative speed is above a predefined threshold, the control procedure acts on the EGO brake by modulating its intensity in order to maintain the safe distance. The extensive details of the complete ACC algorithm used for the current work is beyond the scope of the paper and hence omitted. However, different ACC algorithms can be found in the literature, such as [8] and [9].

b. Autonomous Intersection Crossing (AIC)

The second ADF implemented on the simulator for demonstration is AIC. In this case, a cross-road scenario is chosen and it is assumed that the vehicles approaching the crossing from perpendicular direction. The complete autonomous intersection crossing functionality comprises classical pure pursuit controller and collision avoidance. The pure pursuit algorithm is a classical path tracking algorithm and enforces the EGO vehicle to move to a look-ahead point in front of it from its current position on a pre-defined trajectory. The algorithm then changes the look-ahead point on the trajectory depending on the existing position of the vehicle, see [10]. For the collision avoidance at the intersection, an algorithm presented in [11] and [12] is adopted and implemented. The algorithm enforces the EGO vehicle to either brake or accelerate depending on the states of both the vehicles in order to avoid a collision. This is achieved by defining a certain bad set where the collisions happen. The process involves the generation of a feedback map that prevents the flow of vehicles from entering the bad set and consequently avoiding a collision. In this way one vehicle will try to exit the intersection as fast as possible and the other will try to slow down as much as possible to avoid a collision between both vehicles. The extensive details of this algorithm can be found in [11] and [12].

D. Real-time Co-simulation

It is necessary that the complete vehicle model interacts with other components of the validation platform such as control model, sensors model and environment model via suitable interfaces and has to by performant enough to be executed in a real-time environment. The capability of interfacing Prescan® with Simulink® dictates to use it as a co-simulation platform. Therefore, it is intended that the complete vehicle model (mechanical assembly and its sub-systems), appropriate ADF, sensors models, traffic and environment models will be simulated in a co-simulation fashion using Simulink®. The proposed solution is a flexible and modular validation platform and can be used to validate various automated driving tasks. It is anticipated that the developed platform will not only facilitate for model-in-the-loop and software-in-the-loop simulations but will also enable hardware-in-the-loop and driver-in-the-loop simulations. Therefore, a real-time computing of the models is desirable. To this end, a Concurrent real-time platform with Redhawks Linux (a real-time operating system) and SIMulation Workbench® (a multi-level integration platform) is proposed to run the developed models in real-time. The full vehicle model, the sub-systems models and ADFs are combined in a co-simulation environment. The code is then automatically exported to solve the complete model on a real-time machine, where the solvers are running (see Figure 1). Needless to mention that not all the components of the simulation require hard real-time implementation and therefore this condition can be relaxed to soft real-time for visual, audio and tactile feedback. To this end, a high performance processing unit is used. Once the simulation starts, the input/output communication between different machines is done via appropriate interfaces e.g. UDP, see Figure 1. This allows to send signals to the machine responsible for input acquisition and visualization, as well as to the real-time controller for the motion system used.

E. Cueing

For an immersive human/driver-in-the-loop capabilities, appropriate cueing is indispensable. Out of all, visual feedback is the key ingredient for the realistic driving experience. To this end, high resolution visual cues are projected on 3 screens covering approximately 120 degrees field of view in front of the driver.

Next to the visual, according to previous studies, tactile feedback or haptic cueing is most crucial sense for driving a vehicle [13]. For modern vehicles with steering assist systems, this implies that a reaction torque related to tire and road interaction is necessary to feed back to the driver. Various methods are available to provide a simulated haptic feedback to the driver [14]. For the driving simulator, a steering wheel is used for providing the simulated tactile input, proportional to the steering wheel angle, to the driver.

Inertial or motion cue plays a significant role for the achievement of an immersive feeling in motion driving simulators. Ideally, the inertial motion provided by the motion system should be the same as the one experienced by a real driver. But, due the physical limitations of the adopted motion system, the resulting inertial motion differs from the ideal case. The Motion Cueing Algorithm (MCA) is a control logic which generally consider as input the linear accelerations and the angular velocities of the simulated vehicle and computes the reference signal (in positions, velocities and accelerations) for the motion system. Different MCAs have been implemented for driving simulators in the past decades [15]-[18]. One of the most generally adopted algorithms is known as Classical MCA and it is based on linear filters. In the Classical MCA, the input signals are first scaled, then a high-pass filter is used to remove the sustained components of the signals. Finally, the signals are integrated to be converted to positions or orientation signals and sent to the motion system. In order to reproduce the sustained part of the longitudinal and lateral accelerations of the simulated vehicle, the corresponding signals are low-pass filtered and converted to rotational commands. This strategy, also known as tilt coordination, uses a component of the gravitational acceleration to reproduce the sustained acceleration. Finally the resulting signals are high-pass filtered in order to ensure that the motion remains within the system limitations.

Sound or acoustic feedback is vital for a realistic driving or riding (in case of completely autonomous vehicle) experience. For the developed driving simulator, LMS Test.Lab Virtual Car Sound is used to generate the acoustic feedback for the driver. The process involves the gathering of real-world data, analysing it and creating a Sound Quality Equivalent (SQE) model which can create a synthetic sound which is similar enough to the real-world measurements in order to generate a realistic feel of the simulator to the subject. The SQE model will receive required information from the dynamical model of the vehicle to determine the nature of the sound during simulation. In our case, the input to the sound model are vehicle speed, gear number and engine's revolution per minute.

F. Driver, operator and personnel safety

In order to ensure safety not only for the driver of the developed simulator shown in Figure 7, but also the operator and other personnel at the premises, while the simulation platform is in motion, several safety measures are taken into account. These include but not limited to a metallic cage around the moving base, a dedicated 5-point seat belt harness for the driver, and the emergency stop buttons both for the driver and the operator and emergency exit facility.

Figure 7: Driving simulator for ADAS activities at SIEMENS PLM

III. EXAMPLES

Although the focus of the paper is to discuss the general aspects of the simulator, the following examples are presented for the sake of completeness.

A. ACC on a highway scenario

For this particular example, a section of E40 highway scenario near Bertem in Belgium is made virtually using the real world map data. Both virtual scenario, sensor models and traffic simulation are developed using Prescan®. A dedicated button on the steering wheel is used for enabling/disabling the ACC functionality. For instance, in Figure 8, the driver enables the ACC module at 17.46 sec. It can be seen that the ACC control procedure automatically brakes and releases throttle, irrespective of EGO driver input, in order to restore the safe condition with respect to an arbitrary lead car moving with a velocity of approximately 40 km/hr.

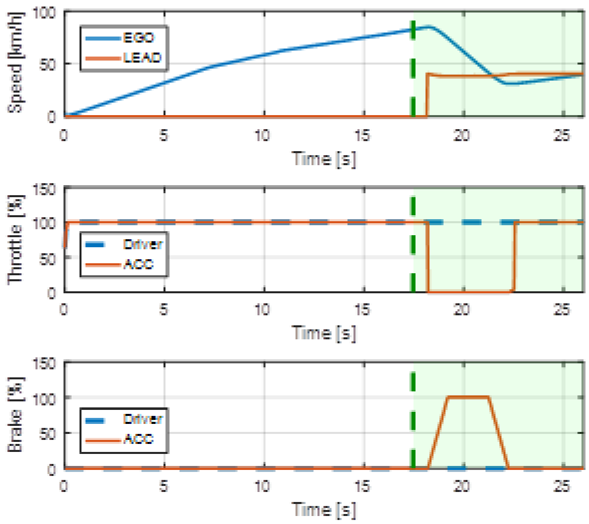

Figure 8: Demonstration of the ACC on a highway scenario

B. Autonomous Intersection Crossing

In order to validate the developed control algorithm, a scenario with two vehicles approaching an intersection crossroads at the same speed of 50 km/h is realized. Two situations are simulated: (i) the initial starting position of vehicle 1 is 1 m closer to the intersection as compared to vehicle 2 and (ii) the initial starting position of vehicle 2 is 1 m closer to the intersection as compared to vehicle 1. If no control would be applied, both of these situations would results in collision.

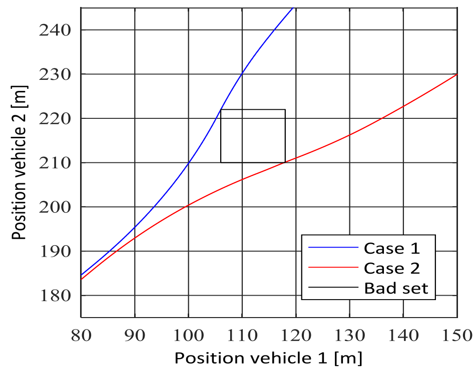

Figure 9: Resulting trajectories of the two vehicles for cases 1 and 2

Figure 9 shows the resulting trajectories of the two vehicles with the control strategy applied. In the first case the vehicle 1 takes priority by going for full acceleration and vehicle 2 goes for maximum braking, in the other case it is the other way around. It can be seen that both trajectories will not enter the bad set. Figure 10 shows a visualization of the results. From these figures, it can be seen that no collision occurs between both vehicles.

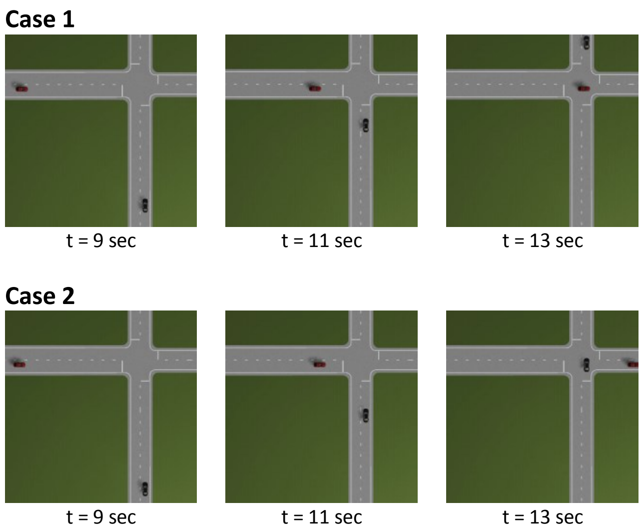

Figure 10: Visualization of the AIC for cases 1 and 2

IV. Conclusion and Recommendations

A X-in-the-loop driving simulation platform is presented for assisting the design, development, testing and validation of ADAS. The modularity of the platform offers not only model-in-the-loop (MIL), and software-in-the-loop (SIL) but also hardware-in-the-loop (HIL) and driver-in-the-loop (DIL) simulations without major changes in the basic hardware and software setup for different validation focuses. The co-simulation platform allows the virtual validation of an ADAS for different variants of vehicle, environment models, sensor models, decision making algorithms or driver behaviors. The inherent flexibility and scalability of the platform offer engineers to test a large number of scenarios, available models (vehicle, sensor or controller), and weather and traffic conditions. Virtual design, development and testing helps to increase trust/confidence in the developed algorithms and technologies, provide proof-of-concept for these and ultimately increase the safety both for the test operators, road actors and infrastructure. Moreover, it would be feasible to test scenarios virtually which are difficult and often impossible or too expensive to test in real life such as certain weather conditions, fault/failure of components (such as camera, LIDAR, RADAR).

The platform can be used in the future to perform a variety of research activities including but not limiting to development of human machine interface for future autonomous vehicles, human perception in autonomous cars, drivers training for autonomous cars, model-based testing and validation.

Acknowledgment

The authors would like to acknowledge ENABLE-S3 project that has received funding from the ECSEL Joint Undertaking under grant agreement No. 692455. This joint undertaking receives support from the European Union's HORIZON 2020 research and innovation programme and Spain, Portugal, Poland, Ireland, Belgium, France, Netherlands, United Kingdom, Slovakia, Norway.

References

[1] Taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles, *Ground Vehicle Standard J3016*, SAE International, September 2016.

[2] T. Benoit, S. Moten and Y. Lemmens, "Real-time physics-based simulation of mechanisms and systems", in *IEEE/ACM 21st International Symposium on Distributed Simulation and Real Time Applications (DS-RT),* 2017.

[3] E. Huang, J. Xu, S. Zhang, and C.-H. Chen, "Multi-fidelity model integration for engineering design," *Procedia Computer Science*, vol. 44, pp. 336–344, 2015.

[4] J. Croes, A. Reveillere, S. Iqbal, D. Coemelck, B. Pluymers, and W. Desmet, "A combined 1D-3D simulation approach for the energy analysis of a high speed weaving machine," in *Proceeding of International Conference on Noise and Vibration Engineering - ISMA*, 2012.

[5] P. C. Breedveld, "Port-based modeling of mechatronic systems," *Mathematics and Computers in Simulation*, vol. 66(2-3), pp. 99–127, 2004.

[6] A. A. Shabana, "Flexible multibody dynamics: Review of past and recent developments," *Multibody System Dynamics*, vol. 1, pp. 189–222, 1997.

[7] A. A. Shabana, *"Dynamics of multibody systems*", Cambridge university press, 2013.

[8] T. Stanger and L. del Re, "A Model Predictive Cooperative Adaptive Cruise Control Approach", in *Proceedings of American Control Conference*, pp. 1374-1379, 2013.

[9] J. E. Naranjo, C. González, R. García, and T. de Pedro, "ACC+Stop&Go Maneuvers With Throttle and Brake Fuzzy Control", in *IEEE Transactions on Intelligent Transportation Systems*, vol. 7(2), pp. 213-225, June 2006.

[10] R. Coulter, "*Implementation of the Pure Pursuit Path Tracking Algorithm*", Carnegie Mellon University, Pittsburgh, Pennsylvania, Jan 1990.

[11] D. Del Vecchio, M. Malisoff, and R. Verma, "A Separation Principle for Hybrid Automata on a Partial Order", in *Proceedings of American Control Conference*, pp. 3638–3643, 2009.

[12] M. R. Hafner, D. Cunningham, L. Caminiti, and D. Del Vecchio, Cooperative collision avoidance at intersections: Algorithms and experiments, *IEEE Transactions on Intelligent Transportation Systems*, vol. 14(3): pp.1162–1175, 2013.

[13] A. Liu and S. Chang, "Force feedback in a stationary driving simulator", in *IEEE Transaction on System, Man and Cybernetics*, vol. 2, pp. 1711-1716, 1995.

[14] H. Mohellebi, A. Kheddar and S. Espie, "Adaptive Haptic Feedback Steering Wheel for Driving Simulators." in *IEEE Transactions On Vehicular Technology* vol. 58(4), 2009.

[15] S. F. Schmidt and B. Conrad, "*Motion drive signals for piloted flight simulators*", Technical report, NASA, Washington, 1970.

[16] L. D. Reid, and M. A. Nahon, "*Flight Simulator Motion-Base Drive Algorithms: Part 1 - Developing and Testing the Equations",* Technical report, University of Toronto Institute for Aerospace Studies, 1985.

[17] M. Dagdelen, G. Reymond, A. Kemeny, M. Bordier and N. Maïzi, "Mpc Based Motion Cueing Algorithm: Development and Application to the ULTIMATE Driving Simulator", *in Driving Simulation Conference (DSC)*, 2004, pp. 221–233.

[18] Z. Fang and A. Kemeny, "Motion cueing algorithms for a real-time automobile driving simulator", *in Driving Simulation Conference (DSC)*, Paris, France, 2012, pp. 1–12.