Digital Twin Environment to Integrate Vehicle Simulation and Physical Verification

Hayato SHIKATA*, Tetsuo YAMASHITA, Kouji ARAI, Takayuki NAKANO, Kenichi HATANAKA, and Hiroyuki FUJIKAWA

In-vehicle control software has become complex, and powertrains have been diversified due to the advancement of vehicle electrification pushed by tightening environmental regulations. Nevertheless, we are required to shorten our development time and propose driver-friendly vehicle systems. We have worked on the development of mechanical and electrical co-simulation techniques as well as simulation-based quantitative evaluation methods for the trade-off between power consumption and ride comfort. To strengthen our system design and solution proposal capabilities, we have also been developing a whole vehicle simulation technique. This paper introduces our virtual car that can be connected with the real driving environment, focusing on an electric vehicle charging system.

Keywords: automotive, simulation, virtual vehicle

1. Introduction

Recently, the in-vehicle electronic control units (ECUs) and control programs implemented in ECUs have become more complicated and larger than ever before due to the automated driving and advanced driver-assistance systems (ADAS) that have been developed around the world. Meanwhile, vehicles have been increasingly electrified due to the environmental regulations that have been tightened in respective countries. Powertrains have also become more diversified, with the development of electric vehicles (EVs), hybrid electric vehicles (HEVs), and plug-in hybrid electric vehicles (PHEVs). Under these circumstances, the vehicle development period has been reduced, and suppliers have been increasingly required to not only deliver parts but also to make proposals from the viewpoint of vehicles in the process of creating specifications and drawings.

We have long been working to develop simulation technologies. We have developed vehicle simulation technology(1) for quantifying the influence of our electric power related parts on vehicle performance and riding comfort simulation technology(2) for quantifying the correlation between the electricity consumption and riding comfort (an evaluation item for electrified vehicles from the viewpoint of users). A project is underway to develop an overall vehicle simulation technology to enhance the capability to design and propose systems from the viewpoint of vehicles. We have built a virtual environment (virtual vehicle) that is linked with a real vehicle on a computer to develop a simulation technology that can accurately verify designs under various conditions even before prototyping a real system, based on the theme of an EV automatic charging system. This paper explains this simulation technology that is expected to help coordinate needs by proposing concepts at an early stage to customers, increase the speed of R&D, and improve safety and quality.

2. Digital Twin

Digital twin refers to the concept of reproducing events in the real world at factories or in products in real time in a digital virtual environment. Simulation has long been used to verify designs in a virtual environment. In the digital twin concept, the data obtained in the real world is fed back to the virtual environment to increase the accuracy of models. This minimizes the difference between the real world and the virtual environment and enables further advanced simulation. Digital twin has been drawing increasing attention as an important technology in next-generation manufacturing.

3. EV Automatic Charging System

To charge an EV, it is generally required to manually connect a quick/normal charger at home or in a public space to a vehicle via a charging cable. Given that EVs and unmanned driving achieved by automated driving will come into widespread use, manual charging is not user-friendly, and unmanned and automatic charging should be enabled. In this study, we developed an EV automatic charging system. There are two possible options to start charging automatically: parking a car so that the charging connector of the vehicle fits the charging cable of a secured charger, or wireless charging. In both cases, highly accurate parking positioning is required.

Regarding the automatic parking system, some commercially available vehicles are already equipped with a system to assist the driver in parking (parking assist function). After driving the vehicle close to the parking position, the driver sets the parking position based on the image captured by an in-vehicle camera. The system assists the driver in parking the vehicle by automatic steering. Such system is designed only for parking. As long as a vehicle can be parked within a parking bay, no further accuracy is generally required. Meanwhile, the EV automatic charging system developed in this study was designed to charge a vehicle after automatic parking. Thus, it required more accurate parking positioning.

3-1 Details of the system

Figure 1 shows the overview of the EV automatic charging system that we developed in this study. Regarding the process of automatic charging, the markings of a target parking bay are captured by the in-vehicle backup camera and detected on the image by using an image analysis technique. The relative distance and angle to the detected markings are calculated, and the driving path from the current stop position to the target parking position is calculated. Subsequently, the vehicle is moved to the target stop position along the calculated driving path by controlling the steering angle and driving speed. After stopping, charging of the EV battery is commenced if the relative position error between the parking target position and the vehicle is within a certain range. The system is also equipped with obstacle sensors (ultrasonic sensors) to ensure safety. If any persons or objects are detected around the vehicle, the vehicle stops before collision.

To realize these functions, this system is equipped with the following three ECUs.

(1) Positioning ECU

The Positioning ECU is connected to the backup camera, and detects the position of the power transmission unit (target parking position) by image analysis processing.

(2) Automatic parking ECU

The automatic parking ECU calculates the driving path from the target parking position, and controls the steering angle and driving speed. It is connected to ultrasonic sensors, and stops the vehicle if an obstacle is detected.

(3) Charge control ECU

The charge control ECU controls the start and stop of charging after stopping at the target parking position.

We chose an electric golf cart as the base vehicle to install these ECUs for three main reasons.

(1) The electric golf cart is equipped with a lithium ion battery that can be charged from outside.
(2) The electric golf cart is an EV whose motor is driven by a battery.
(3) The structure and controls are simpler than those of passenger vehicles. This makes it easy to improve the plant model to attain a higher level of perfection.

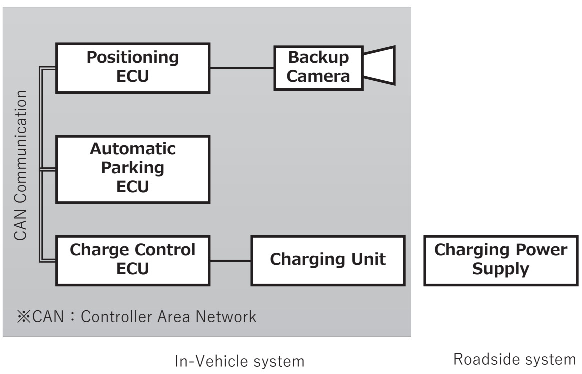


Fig. 1. Overview of the automatic parking system

4. Overview of the Simulation

The overview of the simulation is shown in Fig. 2. The virtual environment (virtual vehicle) for this study consists of a vehicle model for rendering the physical behavior of the vehicle and a virtual ECU (virtual microcomputer) model for controlling the virtual environment. The results were rendered using a 3D rendering engine.

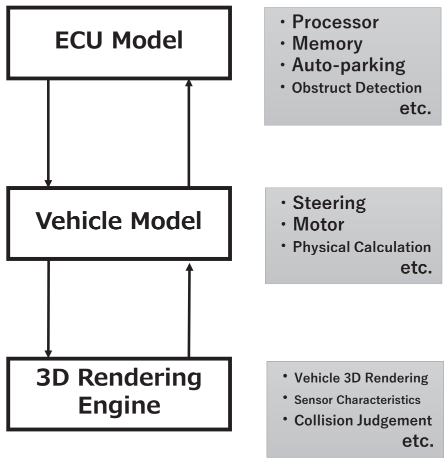


Fig. 2. Overview of the simulation

4-1 Construction of a virtual vehicle (plant model)

To represent the physical behavior of the vehicle (golf cart), the vehicle was broken down into elements such as steering, steering potentiometer, and motor drive system, and each element was modeled. Modeling and fitting were performed using the main specifications obtained from the golf cart manufacturer and the characteristics data of the transient behavior of the vehicle such as acceleration and deceleration. The automatic parking system mainly requires steering while reversing at low speeds. Thus, the Kinematic Bicycle Model*1 was used as a model to render the transient behavior and attitude of the vehicle.

In addition to the vehicle motion, we also modeled the characteristics of functional parts such as ultrasonic sensors and buzzers, and operation was enabled in simulation.

4-2 3D rendering engine

We used a 3D rendering engine as the environment to calculate the vehicle plant model and display the calculation results. The engine was originally developed for developing games. The 3D rendering engine has attracted attention for its rendering capability. At present, it is also used for the visualization of movies, tutorials, and simulation. The 3D rendering of simulation results is more intuitive and persuasive compared to the conventional text-based output of results. To create the vehicle model, we constructed an environment that can import the 3D-CAD data of the vehicle (Fig. 3). This enabled simulation by linking with the 3D-CAD design data. For example, it was possible to verify whether changes in the installation position of the ultrasonic sensors affected the safety (braking distance of the vehicle).

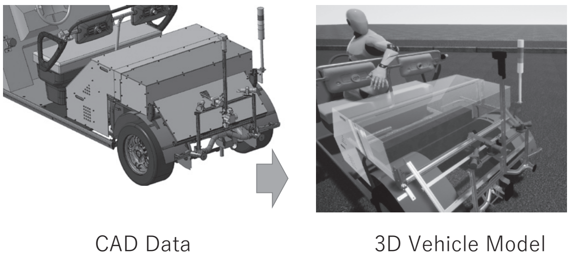


Fig. 3. CAD data and 3D vehicle model

4-3 Construction of an ECU model

Regarding the ECU model for controlling the plant model (vehicle model), we broke down the microcomputer functions into elements including the central processing unit, memory, and general-purpose input/output and modeled them using a hardware description language.*2 To ensure operation of the ECUs, it was necessary to develop a microcomputer model and a program that operated on the microcomputer. Recently, it has been required to reduce the development period. To seamlessly implement the V-model development process, the development assets such as the source code verified in simulation should be readily usable in the downstream process. Thus, we built an environment where executable files (including abs files) that were compiled and built from the source code developed in C language for the real system ECUs (real system microcomputers) and operated on real system microcomputers were readily usable (Fig. 4).

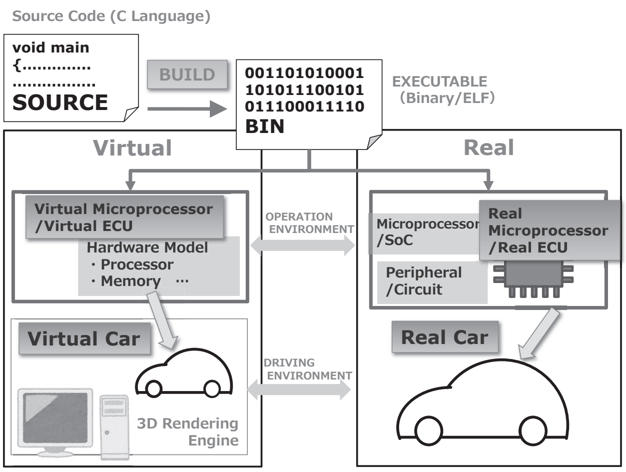


Fig. 4. Virtual environment and real vehicle environment

4-4 Integration of the vehicle model and the ECU model

The vehicle model simulator and ECU model simulator operated as separate processes on the computer. To link the operation of these simulators, we used interprocess communication to exchange information on the time, vehicle control, and vehicle positions. The exchange of the time information for synchronization produced stable simulation results even when there was a difference in the simulation execution speed due to the increase in the calculation load of one simulator. The separate process configuration also made it possible to arrange the simulators on different computers. The computers can be increased and upgraded depending on the complexity of the vehicle model and ECU model as well as the number of models to ensure scalability (extensibility).

5. Verification of Algorithm by Simulation

We verified the stop position error after automatic parking by using the simulation environment that we created (Fig. 5). At the beginning of the development, the error was dozens of centimeters from the target position. We decreased the error to several centimeters under certain conditions by improving the algorithm and taking into account the physical characteristics of the vehicle. In the process of improving the algorithm, we repeated the cycle of formulating a hypothesis based on the results obtained, developing an idea for improvement, and implementing and verifying the hypothesis. We automated the parameter setting for simulation execution, execution, and collection and analysis of results by using a CI*3 tool. The series of processes from algorithm improvement, source code implementation, simulation execution, and analysis of results was conducted continuously and quickly.

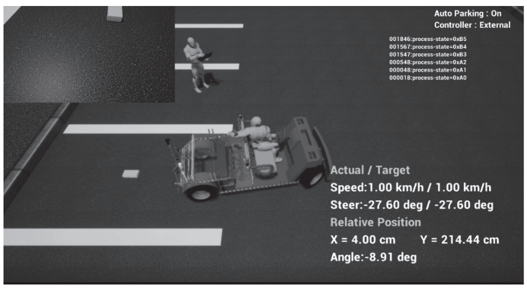


Fig. 5. Screenshot of automatic parking simulation

6. Linkage with the Real Vehicle Environment

We constructed a virtual vehicle environment, and were set to attain a certain level of performance by improving the algorithm by simulation. Thus, we started verification using a real vehicle. As discussed above, the executable files operated on the real system microcomputer were verified by simulation. Thus, verification of a real vehicle could be conducted with a high level of perfection. When improvement points and defects were found in the real vehicle verification, we could determine whether the performance could be improved by creating programs that implemented the solutions and conducting verification by simulation in a virtual environment again. It was also possible to verify in advance whether changes in the test conditions would have unexpected adverse effects.

7. Conclusion

We constructed a virtual vehicle on a computer to develop an EV automatic charging system. Simulation-based verification using a virtual vehicle made it possible to verify and improve the algorithm even before a real system is completed. Incorporation of a 3D rendering engine was found to be effective in linking with the 3D-CAD data and verifying safety.

In Europe, there have been developments to apply a virtual test (which utilizes simulation without testing a real vehicle) to type approval of vehicles. It has been increasingly important to cope with digital engineering including simulation. In this study, we chose an EV automatic charging system as the simulation target. We will expand the scope of application of a virtual vehicle beyond charging-related products. We hope to utilize it to propose V2X-related products, electric power related products, and harness products from the viewpoint of vehicles.

8. Acknowledgements

In this study, we received tremendous support from Yamaha Motor Powered Products Co., Ltd. in terms of various data related to the golf cart and advice about control. We express our deepest appreciation.

Technical Terms

∗1 The Kinematic Bicycle Model: One of the vehicle models that show the vehicle motion based on the dynamic relationship. The model focuses on the center of gravity of a vehicle. The speed vector of the front wheels is handled separately from that of the rear wheels. Thus, the speed vector at the center of a vehicle during steering does not match the direction of the vehicle. The slip angle of wheels increases in the high speed range, resulting in low accuracy. Thus, this model is suitable for the low speed range.

∗2 A hardware description language: A language for describing the operation specifications of hardware. It is used to design large-scale integrated circuits, and can describe the element configuration, operation conditions, and connections. It is different from software programming languages in that it can express the concurrency of processing of parts and passage of time. Typical languages include VHDL, VerilogHDL, and SystemC.

∗3 Continuous integration (CI): A practice in software development to repeat the series of automated processes (including building, testing, and deployment) when a developer changes the source code, to reduce or quickly identify problems in the respective processes, and to minimize the influence on quality, delivery time, and cost. Tools such as Jenkins and CircleCI are available to support CI.

References

(1) K.Arai et al., “Mechanical and electrical co-simulation technology for in-vehicle power systems,” SEI Technical Review, No.84, April 2017, P120 (2017)

(2) K.Arai et al., “Vehicle Simulation-Based Quantitative Evaluation for Power Consumption and Ride Comfort,” SEI Technical Review, No.87, October 2018, P10 (2018)

Contributors

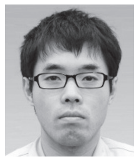
The lead author is indicated by an asterisk (∗).

H. SHIKATA*
• Assistant Manager, Advanced Automotive Systems R&D Center

T. YAMASHITA
• Assistant Manager, Advanced Automotive Systems R&D Center

K. ARAI
• Advanced Automotive Systems R&D Center

T. NAKANO
• Advanced Automotive Systems R&D Center



K. HATANAKA
• Manager, Advanced Automotive Systems R&D Center

H. FUJIKAWA
• General Manager, Advanced Automotive Systems R&D Center