

A Study on the Methodology to Develop Virtual Drive Environment for Autonomous Driving Evaluation

Wonyul Kang¹ Jongho Park¹ Daeoh Kang¹

¹iVH, South Korea

Abstract

It takes many hours and restrictions once AD(Autonomous Driving) evaluation based on real tests. This paper presents a methodology for development of virtual driving environment that can replace the real vehicle test. When developing a virtual driving environment, it is important to develop the same virtual element model (Road, Vehicle model, etc.) as the real world. So the high-occupancy BRT (Bus Rapid Transit) bus route in Cheongna zone was modelled using the MMS(Mobile Mapping System) as the openDRIVE format which is the ASAM(Association for Standardization of Automation and Measuring Systems) road standard. In addition, we develop a vehicle model that simulates the dynamic performance of BRT based on Modelica language. Finally, we develop an interface module that integrates the virtual environment, the vehicle model, and the driver model. In conclusion, this paper present virtual test drive platform for AD Evaluation.

Keywords: *Autonomous, BRT, Vehicle Model, AWS*

1 Introduction

The recent leader companies in autonomous driving are not automobile manufacturers, but IT companies or venture companies. In the case of Google Waymo, it uses Carcraft software to verify edge cases that cannot be tested in real cars and performs tests by driving more than 8 million miles a day virtually. This is the distance that in virtual reality, the real vehicle travels about three times the distance that can be driven in one year. Therefore, it is essential to accelerate the autonomous driving logic based on the virtual driving environment.¹⁾

The virtual driving environment consists of basic elements such as road network/logic, road surface, traffic flow, vehicle dynamics model, driver model, weather model, and sensor model. Existing studies have evaluated representative scenarios after creating artificial road models, and in the case of vehicle dynamics models, there is a problem that the low-DOF vehicle model cannot be applied to reflect the vehicle's behavior in response.²⁾ The vehicle's responsiveness here means the time and behavior until the vehicle responds to the vehicle's steering and acceleration/deceleration pedal inputs. The factors affecting the vehicle's response are determined by the

structure and characteristics of the tire, powertrain, and suspension and in the case of mathematical vehicle models and low-DOF vehicle models, it is difficult to reproduce the characteristics of the vehicle's response to the input.

Therefore, as an alternative to the existing representative scenario-based autonomous driving evaluation method, this study proposes a virtual driving environment that can evaluate control logic in driving distance-based random traffic situations and a vehicle model construction and linkage technique that can reflect the vehicle response and dynamic characteristics of the real vehicle.

In addition, the virtual driving environment developed in this study was developed as a simulation standard defined by the Association for Standardization of Automation and Measuring Systems (ASAM) for usability. The simulation standards defined by ASAM are openDRIVE for roads, openCRG for road surfaces, and openSCENARIO for scenarios. The detailed research sequence is as follows.

This study was conducted to virtually evaluate the autonomous driving logic of the large-capacity refractive type BRT (Bus Rapid Transit) operating in the Cheongna district of Incheon.

As for the order of the study, first, a high-definition virtual driving environment was constructed. Based on MMS, the BRT route of Cheongna district was scanned. By extracting road information from the scanning data, the openDRIVE file was modeled and the actual road was virtualized. Next, a scenario model for driving on a BRT driving route was constructed. In this study, a traffic model was created around the control vehicle (Ego car) by using Pulk Traffic (Random Traffic) supported by the VTD (Virtual Test Drive). The driver model is an autonomous driving logic to be developed and evaluated, and in this study, the autonomous driving logic provided by VTD was used. The sensor model used for autonomous driving performed autonomous driving according to object list information using the Perfect sensor provided by VTD

The second is the development of a modelica-based high fidelity vehicle dynamics model. In order to reflect the dynamic performance of the actual BRT vehicle, the

vehicle dynamics model was modeled using Dymola, a 1D analysis simulation software. In addition, a BRT handling test was performed to validate model responsiveness.

Third, an interface module was developed to interface a virtual driving environment with a 3rd part vehicle dynamic model. The interface module interworks data between the VTD and the dymola vehicle model based on UDP (User Datagram Protocol) communication.

Finally, the driver's seat Mock-up of an actual BRT vehicle was manufactured. And a standardized high-definition virtual environment was developed.

2 Building virtual driving environments based on real roads

In Chapter 2, a virtual driving environment was developed to evaluate large-capacity BRT. For the static element, a high-definition road model was developed using MMS equipment.

2.1 Road Scanning Based on MMS

In Section 2.1, an MMS-based road network model was developed. In this study, the Pegasus 2 Ultimate equipment of Leica Geosystems was used and scanned at a speed of 50 kph. The driving route is from Cheongna International City Station to Gajeong Station. The BRT in Cheongna District has two routes, and the route bus is bus 701 and 702. The scanning data consists of image data generated from a 360-degree camera and four cameras, and is used to generate LAS (LAS) data. Table 1 shows the MMS equipment specifications, and Figure 1 is the vehicle mounting and calibration setup diagram of the MMS equipment.

Table 1 Mobile mapping system specification (Pegasus 2 Ultimate)

	Specification	Remark
Camera	4 ea	12M Pixels
360° camera	1 ea	24MP panorama
Scanner	Z+F 9012	
accuracy	0.015m(Vertical), 0.02m(Horizontal)	

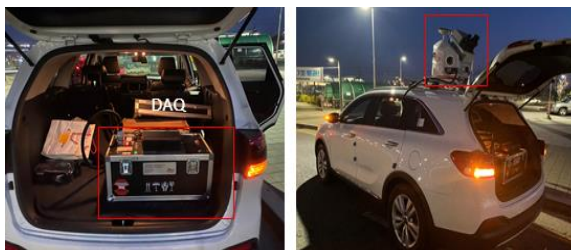


Fig. 1 Mobile mapping system calibration setup

2.2 openDRIVE Road modeling

The LAS data is generated by mapping the image information of the camera corresponding to the cloud points x, y, and z points generated from the LiDAR sensor.

The LAS data generation process is as follows. First, the x, y, and z point information, which are cloud point location information, is acquired. A colored cloud point data base is generated by inputting red green blue (RGB) values of pixel information corresponding to x, y, and z of the point by using the camera image information captured when generating the cloud point. The LAS data generated for the BRT driving route in this study is shown in Figure 2.



Fig. 2 Scanned las data of BRT route

Next is the classification of cloud points. Also called point grouping of points, LAS data was performed using the AI logic of the Road Factory. Grouping proceeds by object, and point grouping is performed by distinguishing objects such as trees, roads, buildings, and wires. Through point grouping, the user can extract only the LAS data of the desired object. In this study, a road point group was used to utilize LAS data for the road model, and later elements such as buildings and trees were separately modeled using the VTD Road Designer.

Next, openDRIVE-based BRT driving route was created. openDRIVE is an ASAM standard that defines road network information. In openDRIVE, road lanes, widths, altitudes, and bank angles were fitted with a third-order polynomial. In order to virtualize random lanes, vehicle widths, altitude information, etc. extracted based on LAS data generated from MMS equipment, it is impossible to virtualize real roads with a combination of general straight or curved shapes. Therefore, in this study, fitting based on a third-order polynomial was performed, and all road information is defined by the values of a, b, c, and d in Equation (1).

$$y = a + bx + cx^2 + dx^3 \quad (1)$$

Here, x is information on the length direction of the road, and y is the width, altitude, and bank angle of the road. In openDRIVE, road attribute information on the road length direction is expressed by the definition of a , b , c , and d values.

In openDRIVE, you can define surrounding environmental objects in style items. openDRIVE includes only location information for user data, and actual graphic data is managed in other databases. In this study, graphic data such as basic buildings, trees, guardrails, streetlights, and traffic lights provided by VTD were used, and openDRIVE includes location information of graphic data. In this way, VTD's virtual environment consists of two types of road network files called openDRIVE and osgb graphic files in the form of openscene graph. In this study, graphic elements such as road marks, stop lines, traffic lights, guardrails, trees, buildings, and underground roads were modeled. Figure 3 shows a landscape graphic model.



Fig. 3 Landscape & tunnel graphic model

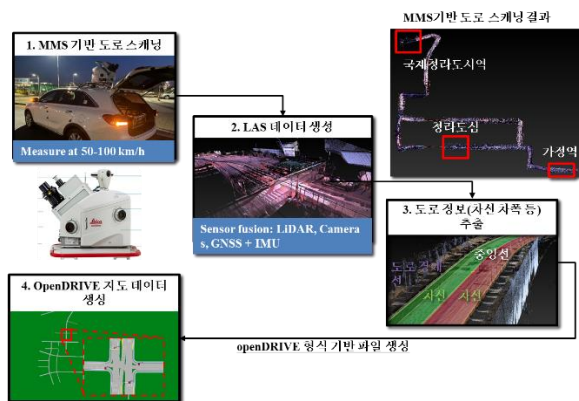


Fig. 4 openDRIVE modeling process based on MMS

Figure 4 shows the process of creating an MMS-based open DRIVE model

2.3 BRT Autonomous-Driving Testing Scenario Modeling

In this study, the scenario was modeled using the openSCENARIO standard. openSCENARIO is the scenario standard of ASAM and expresses the dynamic behavior of an object during simulation. This study evaluated the performance of the controller by modeling

the general driving situation for BRT driving routes as a scenario using Random traffic. In the next year, the controller performance evaluation will be improved by developing a test automation function that considers the edge case (cut-in, cut-out, fedarian scenario etc.) for control logic performance evaluation.

As for the scenario for BRT evaluation, a scenario model was constructed in which the location of the bus stop was modeled on the bus line BRT 701 and then stopped for 30 seconds at the bus stop and then departed. The traffic model utilized VTD's Pulk traffic. Pulk Traffic creates a traffic model around a control vehicle (ego) as random traffic provided by VTD. The traffic model generation conditions are as follows. 60 traffic volumes were generated centering on ego vehicles and the traffic flow in the urban area was reflected. In addition, the vehicle distribution was defined as front:40%, rear:30%, left:15%, and right:15% centering on the ego vehicle. The distribution of vehicle types was defined as passenger cars: 75%, van:10%, bus:5%, truck:5%, and two-wheeled vehicles:5%. Details are shown in Figure 5.



Fig. 5 Scenario model based on openSCENARIO

3 BRT Vehicle Model Modelling and Validation

In Section 3, a vehicle dynamics model based on the Modelica language was developed to implement a real vehicle response to steering input, acceleration, and braking pedal. The chassis system modeled a three-axis vehicle model that can simulate bimodal behavior and modeled the steering system on each axis. The powertrain model was developed by modeling batteries, motors, diesel engines, and controllers based on Dimola's Battery Library and Vehicle Dynamics Library. The tire model expressed nonlinear tire behavior by applying the Pacejka Tire model.

3.1 Chassis modeling

In order to implement AWS in the chassis system, the steering function independent of each axis was modeled by applying the steering system to each axis of the vehicle.

The second and third axes were modeled as a Hook (Ball joint) joint to simulate the bimodal behavior of the vehicle. The suspension types on the 1, 2, and 3 axes are all double wishbone types, and the anti-roll bar is modeled on each axis. Figure 6 shows the block diagram of the suspension type.

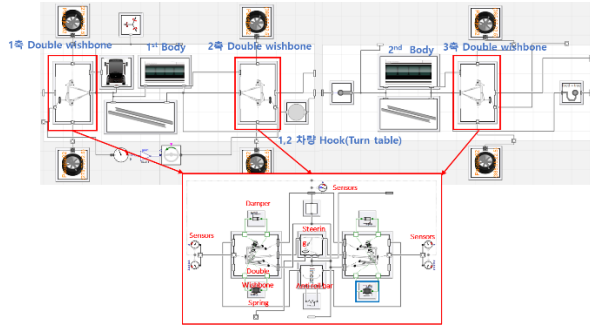


Fig. 6 Modelica based chassis model

The spring of each shaft was modeled as an air spring model, and the air spring formula and performance graph are shown in Equations (2) and Fig.7 below.

$$V_0 = A_v * 0.2$$

$$F_p = dP * A_f$$

$$F_s = c_k * s_{rel}$$

$$V = V_0 + A_v * s_{rel}$$

$$dP = \frac{(P_0 + P_{atm}) * V_0^n}{V^n} - P_{atm} - P_0$$

(2)

$$-F + F_0 + F_p = F_s$$

$$s_{rel} = \text{displacement of Flange.b} \\ - \text{displacement of Flange.b}$$

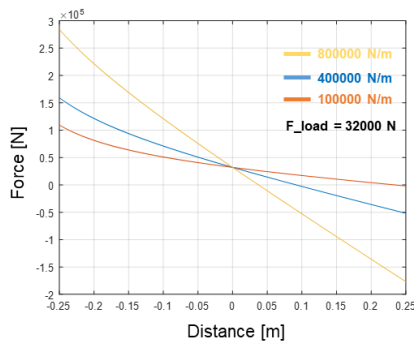


Fig. 7 Air spring stiffness

3.2 Powertrain modeling

The large-capacity BRT is a diesel engine-based serial hybrid vehicle, and Euro 6 209 kW diesel engine and 140 kW generator are applied. The battery was modeled based on Modelon's Battery Library, and the motor was constructed as a model that controls the torque as an angular acceleration target using a controller. The controller converts the displacement of the accelerator pedal into voltage to power the vehicle.

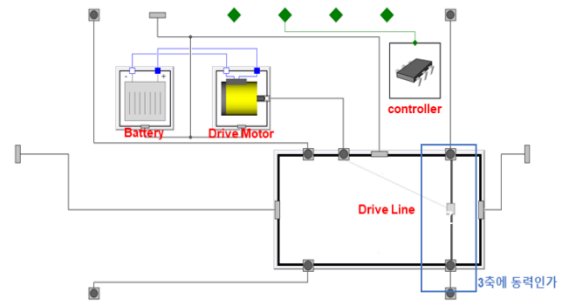


Fig. 8 Modelica based powertrain model

The figure below shows the powertrain integration template. The brake module and powertrain module are interlocked with the chassis module for each shaft.

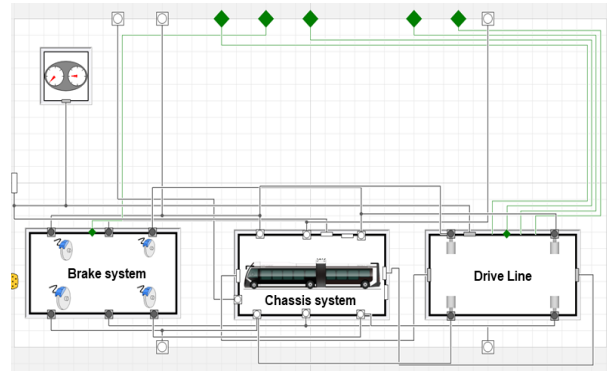


Fig. 9 Modelica based BRT template model

The vehicle dynamics model becomes a sub-module of the test mode, enabling virtual simulation of vehicle dynamics tests.

3.3 Vehicle model Validation

In order to validate the vehicle dynamics model developed in Section 3.2, a real vehicle-based dynamic characteristic test was performed. In this study, tests were conducted in the airways under construction due to the test vehicles and vehicle movement conditions, and acceleration, braking, and double lane change test modes were selected to validate the vehicle dynamics model. The maximum speed of a large-capacity BRT is 80 kph. In this study, the acceleration test measured data from a stopped state to reaching 70 kph, and in the case of a braking test, the test data from 70 kph to a stop was measured. Due to the safety of the BRT vehicle of the double lane change, steering

was limited so that a lateral acceleration of 0.2 g occurred at 60 kph for safety reasons.

In this study, the data required for validation were measured through the IMU (Inertial Measurement Unit) sensor installed in the target vehicle. In the IMU sensor, vehicle state information and three-axis data were measured to measure the result of each test mode. In the test, commercial electric buses are refractive buses, and IMU sensors were attached to Front and Rear to validate bimodal behavior to obtain measurement results, respectively.

The steering signal and pedal information were measured through the vehicle's controller area network (CAN) communication. The time vs steering angle information obtained in the actual vehicle test was applied to the vehicle dynamics model validation environment, and the dynamic simulation result value and the measured value in the actual vehicle test were compared. The figure below shows that a sensor for measurement is installed on a refractive bus.



Fig. 10 BRT test equipment setup

The vehicle model validation built in this study was performed based on the results of the actual vehicle test conducted by the experimental method defined. the simulation validation environment was modeled as shown in the figure below to validate the results and model for the acceleration test, braking test, and double lane change test.

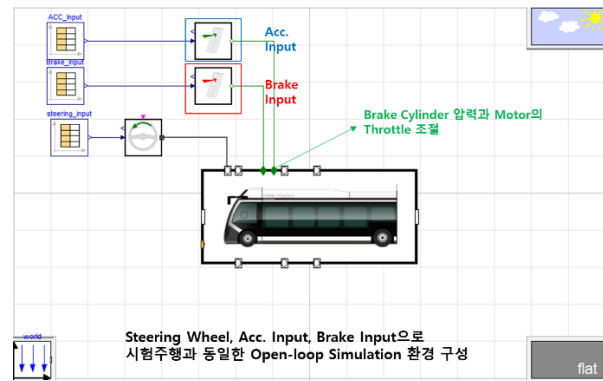


Fig. 11 Vehicle model validation environment

For the validation environment, the steering angle and acceleration and braking pedal values measured in the vehicle dynamics test were applied to the vehicle model as input values.

The actual vehicle test was conducted three times each of acceleration, braking, and double lane change tests, and the analysis of the performance index was performed to validate the vehicle dynamics model as the average of the three performance index values in each test.7)

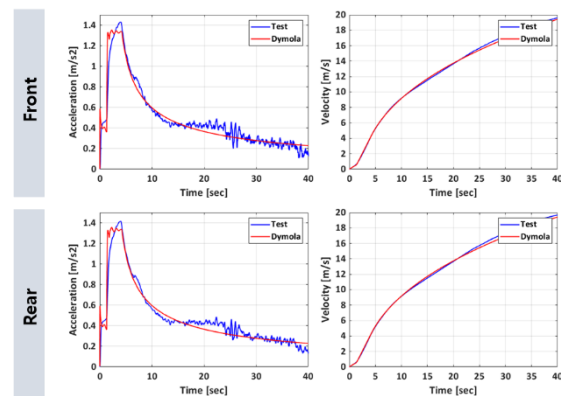


Fig. 12 Acceleration test result

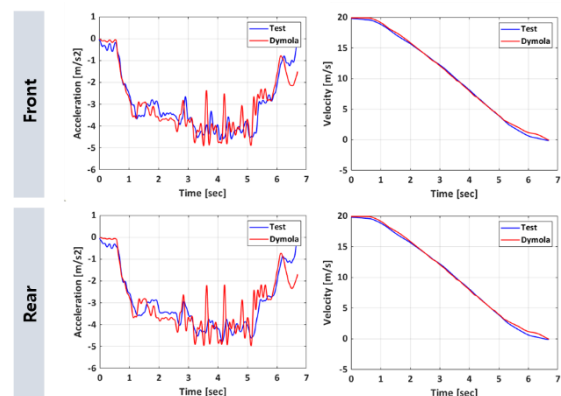


Fig. 13 Braking test result

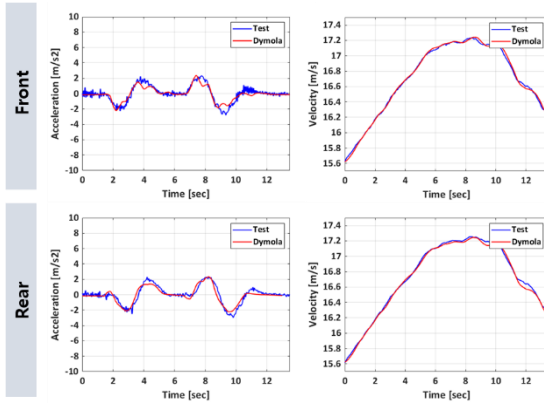


Fig. 14 Double lane change result

The validation was performed based on the speed and acceleration of the front and rear body of the large-capacity BRT. Acceleration and braking tests were verified based on the longitudinal speed and acceleration of the vehicle body, and in the case of a double lane change, they were verified based on the lateral speed and acceleration. The index is expressed as RMSPe as shown in Equation (10), and it shows that the model accuracy of the vehicle dynamic characteristics test results in Table 4 is over the target 80%. In addition, it is thought that the model developed by satisfying the model accuracy of 80% or more and the real-time performance of the model at the same time can be sufficiently used as a vehicle model in a virtual driving environment. Details of the vehicle model validation results are shown in the table below.

$$Error(\%) = \sqrt{\frac{1}{n} \sum \left(\frac{y_s - y_t}{y_t} \right)^2} \times 100 \quad (10)$$

where y_s = simulation result
 y_t = test result

Table 1 Vehicle Test results(RMSPe)

		Acceleration test	Braking test	Double lane change test
Front	Acceleration RMSPe(%)	11.9	16.81	16.91
	Velocity RMSPe(%)	1.92	2.24	1.2
Rear	Acceleration RMSPe(%)	7.34	18.56	19.38
	Velocity RMSPe(%)	2.14	2.31	1.6

4 Development of Virtual Driving Environment and Vehicle Model Interface Module

In Section 4, an interworking environment between a virtual driving environment and a vehicle model was constructed. First, after defining the interworking data items, the virtual environment can be controlled based on the C++ source code defined by the user in the virtual driving environment Plug-in environment. The C++ source code includes vehicle control and communication modules, which can be linked with vehicle models. In this study, a communication module in a virtual driving environment and a communication module of a vehicle dynamics model were constructed, respectively. Figure 15 shows the virtual driving environment and the interface environment between vehicle models. 8)

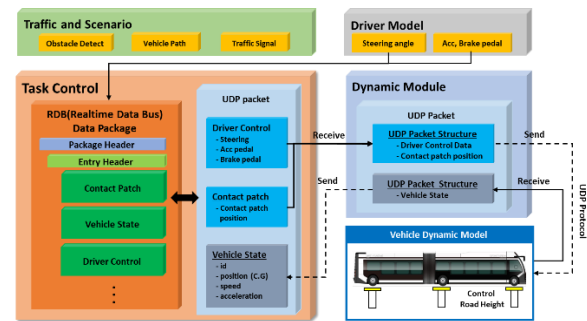


Fig. 15 Data flow based on interface modules

4.1 Modelling UDP Communication Module for Virtual Driving Environments

All data in a virtual driving environment is defined in a data structure called a Runtime Data Bus (RDB). RDB is a standard that defines all data calculated in a virtual driving environment as a structure. Data refers to all components for a virtual environment, such as speed, acceleration, driver signal, location information, pedestrian location, traffic light location, current signal, etc., for all vehicles. The RDB can be output and controlled by a user in a C++ development environment and functions the same as CAN data of a vehicle. In a C++ development environment, the RDB has HEADER and DATA items, similar to a general Packet, and is defined in a structure format.

In order to communicate the RDB data of the virtual driving environment, the C++ source code was generated and the code was executed in conjunction with the virtual driving environment in a Plug-in format. For example, in the RDB structure mentioned in the above section, a code for outputting and transmitting vehicle state information was modeled and a file generated after compile was plug-in to the virtual environment to build an interface environment.

The Driver Model uses the actual steering device and pedal device to provide steering signals by the actual

driver and controls the vehicle with autonomous driving logic when switching control. The user-defined Driver signal is controlled by the Task Control module, a central control module of the virtual driving environment, and reflects the vehicle response by transmitting the driver control signal to the vehicle model.

4.2 UDP Communication Module Modelling for Vehicle Model

Vehicle data was defined with structure data to control the BRT model. The structure is composed of vehicle state information and altitude information at each wheel to reflect the altitude of the road surface in a virtual driving environment.

One contact point between the tire and the road in the vehicle dynamics model was designated per wheel. Figure 16 shows the contact mechanism between road and vehicle models on openDRIVE.

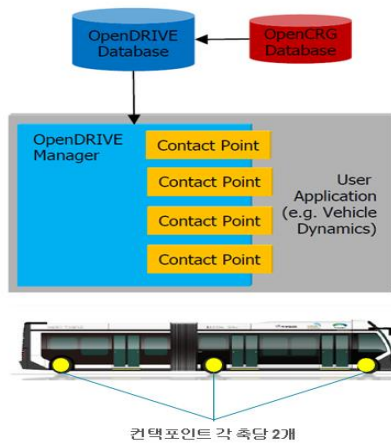


Fig. 16 Road contact mechanism

An interface module was constructed to receive RDB data transmitted in the virtual environment from the vehicle model and transmit vehicle data to the virtual environment. A module capable of receiving actual driver input signals or vehicle controller signals on the steering, acceleration, and deceleration pedals of BRT vehicles, and a module that transmits vehicle state information to the virtual environment were constructed as shown in the figure below.

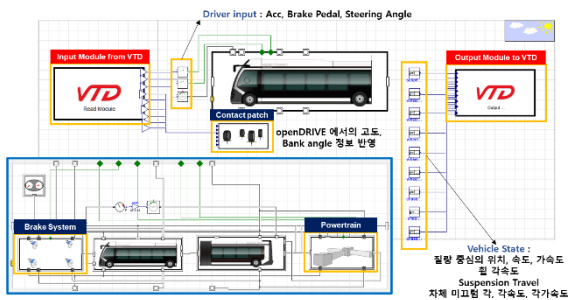


Fig. 17 Interface module for vehicle model

5 Development of Virtual Driving Environment Platform for Autonomous Driving Evaluation

In Section 5, a virtual driving environment platform was constructed using Vires' Virtual Test Drive. In this study, in order to virtually develop the autonomous driving logic of a large-capacity BRT running on the real road in Cheongna District, a virtual driving environment platform was constructed that integrated the previously built road, scenario, and vehicle model based on interface modules. Figure 18 is a summary of the virtual driving environment platform presented in this study.

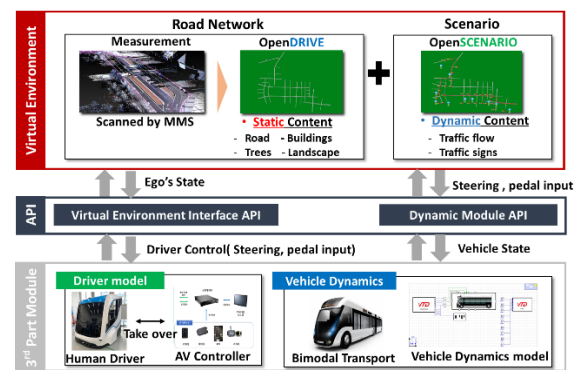


Fig. 18 Virtual drive environment platform for autonomous driving evaluation

In Section 5, the virtual environment of bus routes 701 and 702 of the Cheongna district constructed in this study. The driving simulator manufactured the driver seat of the actual BRT vehicle as a mock-up and the steering angle and pedal input of the controller are implemented through the actuator. In the driving simulator, when the control authority is transitioned, the driver can control the vehicle, and the driver's steering and pedal inputs are applied to the vehicle model, constructing an environment in which the driver can directly drive. The figure below is a diagram of driving the BRT bus route based on an actual driving simulator. The driving rate of the simulation was checked in a random traffic situation, and the driving rate was defined as the number of deviation cases and collisions per number of round trips on the driving route. Based on the VTD autonomous driving logic, 0 cases of route deviation and 0 cases of collision were shown during 236 round trips.



Fig. 19 Driving simulator based on virtual drive environment platform

6 Conclusion

In this study, a high-definition virtual driving environment was developed to evaluate the autonomous driving logic in the preceding stage before converting a large-capacity BRT operating in Cheongna into an autonomous driving system. The VTD (Virtual Test Drive) control logic and the virtual driving environment were linked to the same communication environment as the autonomous driving control logic to be applied to the real vehicles, and the autonomous driving performance for bus routes was evaluated. The VTD autonomous driving control logic will be evaluated by being replaced by the autonomous driving control logic that will be developed for real vehicles in the future. In addition, the vehicle model developed in this study implemented pedal input and powertrain characteristics that could not be implemented when using a low-DOF vehicle model, and a high-fidelity vehicle model was constructed that reflected the response and behavioral characteristics of the vehicle. Through this, a methodology was proposed to develop a controller in consideration of the vehicle response delay phenomenon when developing an autonomous driving controller in virtual. Finally, large-capacity BRT is a public transportation that is actually in operation, and there are many restrictions when evaluating autonomous driving logic based on actual vehicles, and there is a high risk of accidents. Therefore, using the virtual driving environment developed in this study, the control logic was evaluated virtually for a driving distance of 7080 km one-way in the preceding stage. The results derived from this paper are as follows.

(1) A real road virtualization technique based on MMS was established, and a road model that can be evaluated in a simulation environment was constructed by creating a road model of openDRIVE, the ASAM standard.

(2) Through the development of a model-based vehicle dynamics model, a high-precision vehicle model that simulates the dynamic characteristics of the actual vehicle was constructed.

(3) Through the development of interface modules, an environment that can interface 3rd part modules (vehicle models, driver models, and logic) to a virtual driving environment in a plug-in format was established.

(4) Based on the ASAM standard, a highly useful platform was established by constructing a virtual driving environment for autonomous driving evaluation. In the future, when changing the road model, scenario, and vehicle model of the ASAM standard, the platform developed in this study can be used as it is.

In this study, control logic evaluation was conducted in random traffic situations around control vehicles, and actual traffic volume will be applied using traffic simulation tools in the future. In addition, we plan to evaluate the control logic based on the mileage by applying edge cases, accidents, and unexpected situations to the traffic model. Control logic is planning to build and upgrade an evaluation automation environment based on this platform by applying the BRT autonomous driving logic for real vehicles.

7 References

- 1) State of California Department of Motor Vehicles, Disengagement Report 2019, <https://www.dmv.ca.gov/portal/vehicle-industry-services/autonomous-vehicles/disengagement-reports/>, 2020.
- 2) Schiller M, Dupius M, Krajewicz D, Kern A, & Knoll A, "Multi-resolution traffic simulation for large-scale high-fidelity evaluation of VANET applications.", Springer Cham, In Simulating Urban Traffic Scenarios, pp.17-36, 2019.
- 3) J Jo, W Kang, D Kang, "Vehicle Dynamics Model and Tire Filer for Ride Comfort Analysis", Transactions of KSAE, Vol. 28, No. 12, pp.859-864, 2020.
- 4) Vehicle Dynamics Library, Modelon, 2019.
- 5) Vehicle Dynamics – theory and application, Reza N. Jazar, Springer, 2015.
- 6) Hans B. Pacejka, "Tyre and Vehicle Dynamics", pp.483-512, 2006.
- 7) M Hyun, J Yoon, G Lee, "Study on Suspension Bush Model for Predicting Frequency and Amplitude Dependent Nonlinear Dynamic Characteristics", Transactions of KSAE, Vol. 28, No.9, pp.621-628. 2020.
- 8) M Lee, "A study on the construction of commercial electric vehicle fuel efficiency analysis platform based on real road virtual test driving", Master's thesis, Kookmin university, Seoul, 2020