

적응 순항 제어기 성능 평가를 위한 실시간 차량 시뮬레이터 개발

한 동훈¹⁾ · 이 경수²⁾

한양대학교 자동차공학과 대학원¹⁾ · 서울대학교 기계항공공학부²⁾

Development of a Real-time Driving Simulator for ACC(Adaptive-Cruise-Control) Performance Evaluation

Donghoon Han¹⁾ · Kyongsu Yi²⁾

¹⁾Department of Automotive Engineering, Hanyang University, Seoul 133-791, Korea

²⁾School of Mechanical & Aerospace Engineering, Seoul National University, Seoul 151-742, Korea

(Received 28 June 2005 / Accepted 22 December 2005)

Abstract : An ACC driving simulator is a virtual reality device which designed to test or evaluate vehicle control algorithm. It is designed and built based on the rapid control prototyping(RCP) concept. Therefore this simulator adopt RCP tools to solve the equation of a vehicle dynamics model and control algorithm in real time, rendering engine to provide real-time visual representation of vehicle behavior and CAN communication to reduce networking load. It can provide also many different driving test environment and driving scenarios.

Key words : ACC(Adaptive Cruise Control), ITS(Intelligent Transportation System) Driving Simulator, RCP(Rapid Control Prototyping), Optimal control design(최적제어 설계), CAN(Controller Area Network)

Nomenclature

- a : acceleration, m/sec
- c : clearance or range, m
- v : vehicle speed, m/sec
- τ : time-gap, sec
- J : performance index

Subscripts

- $*$: real value
- d : desired
- o : minimum or initial value

1. 서론

지능형 교통정보 시스템(ITS)중 가상의 시험공간에서 컴퓨터를 이용한 시뮬레이션 시험을 할 수 있는 기반을 구축할 필요성이 증대되어지고 있다. 이는 완성차 수준에서의 시험 및 평가가 가능하고, 첨단안전 차량(ASV)을 대상으로 실차시험의 단점을 보완하며 새로운 개념의 차량 시스템 및 제어 알고리즘의 타당성 평가시 활용되어 지고 있다.

차량 시뮬레이터는 그 활용목적에 따라 여러 가지 형태로 분류될 수 있다. GM사의 경우 운동시스템이 없는 고정베이스와, 작은 운동범위를 갖는 운동 베이스가 장착된 두 대의 시뮬레이터를 개발 하였으며 Nippon-Denso의 경우 Dynamo Test Bench를 활용한 고정베이스의 시뮬레이터를 개발하였다.¹⁻³⁾ 운전시뮬레이터는 차량시스템 개발측면에서 운전

*Corresponding author. E-mail: kyi@snu.ac.kr

자와 차량을 포함하는 페루프 시스템을 구축하여, 새로운 개념의 차량시스템을 개발하고, 차량의 각종 제어장치의 알고리즘 개발 및 평가, 운전자교육 또는 실제실험으로는 위험한 위험상황 재연 등에 활용되고 있다.^{3,4)}

본 연구에서 고안된 ACC 차량주행 시뮬레이터의 주된 목적은 차량의 동적 테스트 및 사시, 파워트레인 제어는 물론 차량의 전복방지 알고리즘 개발 등에 활용되어 질수 있다. 이와 유사한 목적을 달성하기 위한 시뮬레이션 엔진으로, 기존에 개발된 워크스테이션 또는 개인용 컴퓨터 기반의 시뮬레이터와는 달리 RCP(Rapid Control Prototyping)개념에 기반을 두어 dSPACE사의 AutoBox를 사용하여 실시간 하드웨어로 구성하였으며, 각종 센서 및 하위 시스템간의 유기적인 결합 및 하드웨어의 사용으로 차량의 운동을 보다 현실성 있게 모사할 수 있도록 설계하였다.

RCP는 일반적으로 새로운 제어시스템을 개발 또는 시험할 때 필요한 실시간 하드웨어 장비를 말한다. RCP시스템은 그래픽환경 프로그램, 부동소수점 연산기, 고정밀도의 데이터 취득 장치로 구성되며 본 논문에서는 실시간 제어를 위해 dSPACE제품

을 사용하였다.

2. ACC 주행 시뮬레이터

본 연구에서 개발된 ACC 주행 시뮬레이터의 하드웨어 구성을 Fig. 1에 나타내었다. ACC 주행 시뮬레이터는 실시간 운동 시스템, 시각 및 음향 시스템, 운전자의 입력을 받아들이는 각종 센서들 그리고 센서의 입력을 실시간으로 처리하는 RCP로 구성된다.

2.1 실시간 시뮬레이션

차량의 동적 특성 분석 또는 새로운 제어기 개발 및 검증 등의 목적으로 고안되어진 차량용 시뮬레이터는 차량의 운동을 실시간(Real-Time)으로 계산해주는 실시간 제어기, 운전자와 운동재현기 사이의 인터페이스를 담당하는 컴퓨터, 실시간으로 계산한 운동을 시각 및 청각으로 표현해주는 실시간 렌더링 툴 등으로 분류할 수 있으며, 이러한 하위 시스템들의 상호 유기적으로 적절히 조합되어야만 실시간 시뮬레이터로서의 의미가 있다. 이러한 요건을 어느 정도 갖춘 차량 운전시뮬레이터는 국내에서도 이미 개발된 사례가 있지만 운동을 실시간으

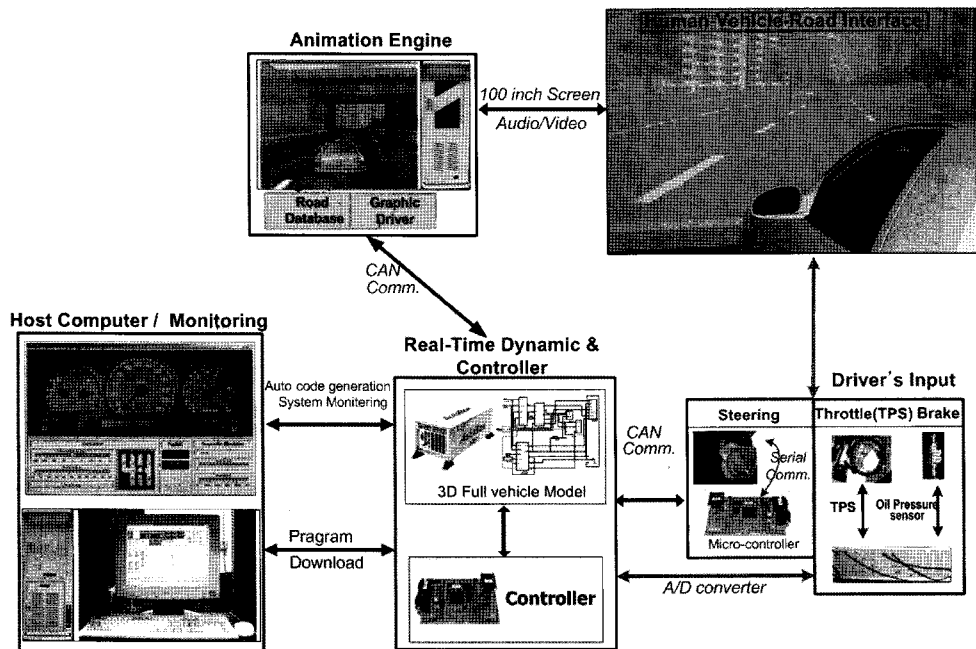


Fig. 1 Configuration of ACC vehicle simulator hardware

로 계산하지 못하거나, 렌더링이 실시간을 보장하지 못하여 그래픽 처리시간 등의 제약을 받는 경우, 통신 속도의 제약 등의 문제점으로 인해 운전시물레이터로서의 활용범위를 제한하거나 현실감을 저해하는 요인이 되고 있다.⁷⁾

본 연구에서는 기존에 개발된 운전 시물레이터에서 발견된 문제점들을 보완하여 차량의 제어기 성능 평가에 활용할 수 있도록 하고자한다.

2.1.1 호스트 컴퓨터

ACC 주행 알고리즘은 Fig. 2에서 나타나는 바와 같이 Mathworks사의 Matlab/Simulink로 구성되어져 있고 이는, 호스트 컴퓨터에 의해 수정 및 변경이 가능하며, 프로그램 구동을 위해 C코드로의 생성, 컴파일 및 실행되어져 이더넷통신에 의해 시물레이션 엔진인 AutoBox로 다운로드 된다. 또한, 호스트 컴퓨터에 의해 시물레이션의 시작, 종료 및 설계인자 변경 그리고 시물레이션 결과를 실시간으로 모니터링이 가능하다.

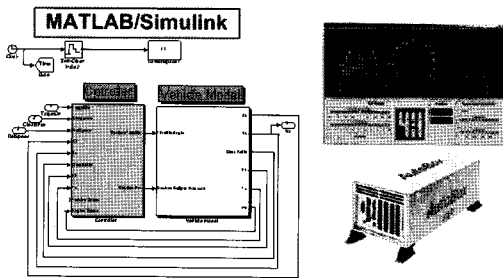


Fig. 2 ACC simulation program and engine

2.1.2 시물레이션 엔진

ACC주행 시물레이터는 가장 큰 특징은 차량의 운동규 생성에 있어서 실시간을 보장한다는 데 있다. 기존에 개발된 시물레이터에서 채택한 워크스테이션 급 또는 PC기반의 시물레이터에 비해 뛰어난 성능을 가지며, 제어변수 등을 쉽게 변경할 수 있다는 장점을 가지고 있다.

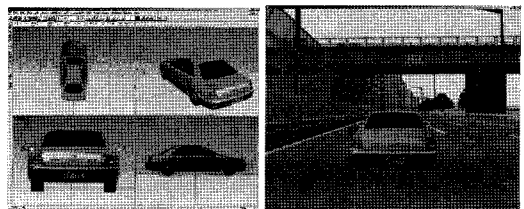
시물레이션 엔진으로 채택하고 있는 dSPACE사의 AutoBox(DS1005)는 800MHz의 CPU를 장착하고 있는 Power PC 750으로, 여러 종류의 센서 신호들을 입·출력할 수 있는 보드 등을 지원하고 있다.

2.1.3 그래픽 엔진

운전자는 주로 시각을 통해 주행 상황을 인지하므로 운전시물레이터에 있어서 현실감을 확보하기 위한 가장 중요한 부분이 시각 시스템이다. 화면에 재현되는 이미지의 연속성 및 현실감을 보장하기 위하여 시각 이미지 생성기 및 생성된 이미지의 실시간 처리 성능은 대단히 중요한 요소이므로 시스템 설계초기 단계에서부터 반드시 고려하여야 한다. 그러나 형상 모델링에 의존한 주행 환경 객체는 과도한 폴리곤과 객체의 시각적 특성 정보를 필요로 하며 모델링 시 많은 시간과 노력을 필요로 한다. 이러한 노력에도 불구하고 과도한 폴리곤 등의 형상 정보는 실시간 시물레이션을 불가능하게 만들게 된다.⁸⁾

본 연구에서는 이미지 생성기로 OpenFlight Format을 가지며 실시간 응용프로그램의 목적에 적합한 계층구조를 가진 MPI사의 Multigen Creator를 사용하였다. 이는 시점으로부터의 거리에 따라 다른 맵핑 이미지를 사용하고, 폴리곤의 수도 다층화하여 사용하는 LOD(Level of Detail)기법을 제공한다. 이렇게 모델링된 대상물을 실시간으로 묘사하는 데는 고성능의 컴퓨터 사양을 요구할 뿐만 아니라, 최적화된 렌더링 툴 또한 요구한다. 실시간 렌더링 툴로는 VEGA를 사용하였다. 이러한 이미지 생성기 및 렌더링 소프트웨어를 구동하기 위하여 AMD사의 두 개의 Opteron CPU를 사용하였고, 그래픽 카드로 Quadro Fx 1100을 사용하였다.

가상환경에서 사용될 객체들을 Fig. 3(a)와 같이 모델링 하고 주행환경을 생성하여 Fig. 3(b)와같이 실시간 렌더링을 수행하였다.



(a) Graphic object (b) Real-time rendering

Fig. 3 Graphic object modeling and rendering

2.1.4 입출력 시스템

운전자의 입력으로는 브레이크압력, 쓰로틀 각,

조향각 등이 있으며 Fig. 4와 같이 각종 센서들로부터 측정되어진다. 운전자 조향각은 0.5° 의 분해능을 가지며 Gray코드를 내보내는 SSI(Synchronous Serial Interface)타입의 절대 엔코더를 사용하였으며, 스프로틀은 TPS(Throttle Positioning Sensor), 브레이크는 Kyowa사의 압력센서를 사용하였다. 또한 운전자에게 유사한 브레이크 페달 힘을 전달하기 위해 전기모터를 사용하였다.

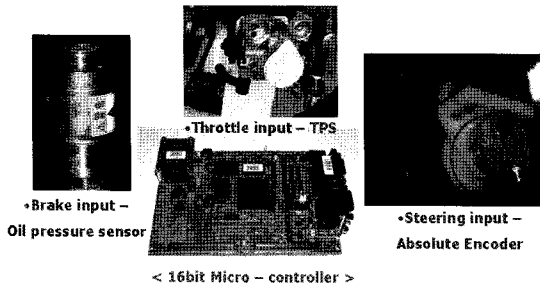


Fig. 4 Driver's input/output system

운전자의 입력을 시뮬레이션 엔진으로 전달하기 위하여 RCP 장비로써 DS2201과 DS4302를 사용하였다. DS2201은 20개의 12bit A/D채널과 8개의 12bit DA채널을 가지고 있어서 스프로틀 및 브레이크 입력을 시뮬레이션 엔진으로 전달하고, DS4302는 CAN(Control Area Network)신호를 사용하기 위한 목적으로 절대엔코더로부터 Gray코드를 받아 16bit 마이크로 컨트롤러를 이용하여 생성된 CAN신호를 시뮬레이션 엔진으로 전달한다. 또한 시뮬레이션 엔진에서 생성된 차량의 위치, 자세, 뷰 포인트 그리고 주변 환경의 움직임 정도 등을 그래픽 엔진으로 전달한다.

이와 같이 BOSCH사에서 제안한 직렬 분산 네트워크인 CAN통신은 각각의 시스템에서 필요한 자료들에 고유 ID를 부여하여 서로 공유할 수 있는 네트워크 방식으로 전압차를 이용한 신호전달방식을 사용한다. 또한 오류검출능력이 뛰어나며, 노이즈에 대한 강인성을 가져 통신 에러율을 최소화하여 높은 신뢰성을 가지므로 기존의 차량시뮬레이터에서 사용하고 있는 이더넷 또는 RS-232C방식의 통신에서 발생할 수 있는 실시간 시뮬레이션 저해 요소를 제거하였다.

3. 차량모델 검증

시뮬레이션 모델은 현대자동차 그랜저 XG 2.5에 대해서 Mathwork 사의 Matlab/Simulink 를 이용하여 구성하였으며 제어기를 제외한 종방향 차량 모델의 거동에 대해서 실제 운전자 실험 데이터와 비교하였다.⁷⁾

3.1 가속성능검증(No brake input)

실제 운전자의 실험을 통해서 획득한 스프로틀, 브레이크 토크를 시뮬링크로 구성된 차량 모델에 입력한 후에 시뮬레이션 실행 결과를 실제 실험을 통해 얻은 속도, 가속도 그리고 기어변속 데이터와 비교하였다. Fig. 5는 가속성능 시험으로써 브레이크 입력이 작용하지 않았을 경우에 대해서 각각의 결과를 비교한 것이다.

3.2 제동성능 검증

Fig. 6은 고속의 주행상황에서 강하게 브레이크를 입력을 가하였을 때 실제 차량의 거동을 동일한 스프로틀 각, 브레이크 입력에 대한 시뮬레이션 결과와 비교한 결과를 나타내고 있다.

Fig. 5, Fig. 6의 결과에서 알 수 있듯이 시뮬레이션에서 사용된 차량모델의 거동이 실제 차량과 흡사함을 알 수 있다.

4. ACC 차량 주행 제어기법

4.1 제어기법

ACC 시스템은 안정된 승차감을 요구 하는데, 운전자와 승객이 편안하게 느껴지기 위해서는 ACC 시스템이 실제 운전자의 주행 상태와 유사하게 부드럽고 자연스런 거동을 보이도록 설계되어야 한다. 기존 연구의 결과로부터 운전자 주행 실험 데이터 분석을 통해 운전자 추종 형태, 가속도 특성, 시간지연 등 운전자 주행 특성을 분석하였다. 이러한 분석 결과를 기반으로 ACC시스템을 설계하였다.⁸⁾

ACC 차량의 속도 및 거리 제어를 위해 목표 가속도는 레이더 센서를 사용하여 측정된 차량의 속도 및 거리 정보를 이용하여 최적 제어 이론을 통해 구하였으며, 스프로틀/브레이크 제어기를 설계하여 제어 차량의 목표 가속도를 추종할 수 있도록 하였다.

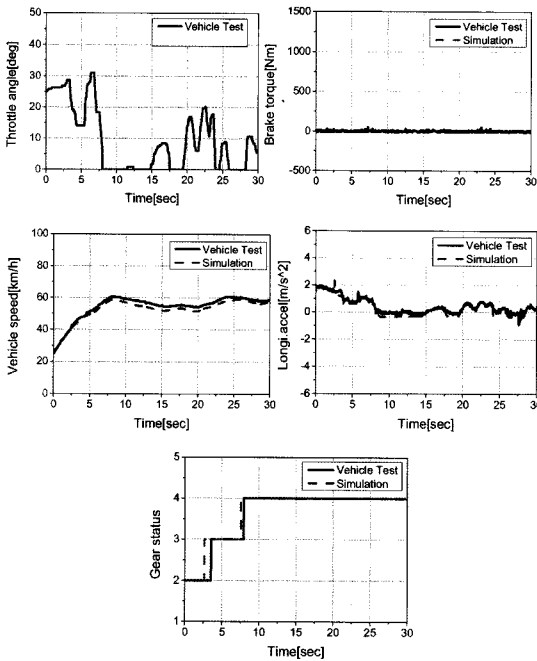


Fig. 5 Acceleration performance test

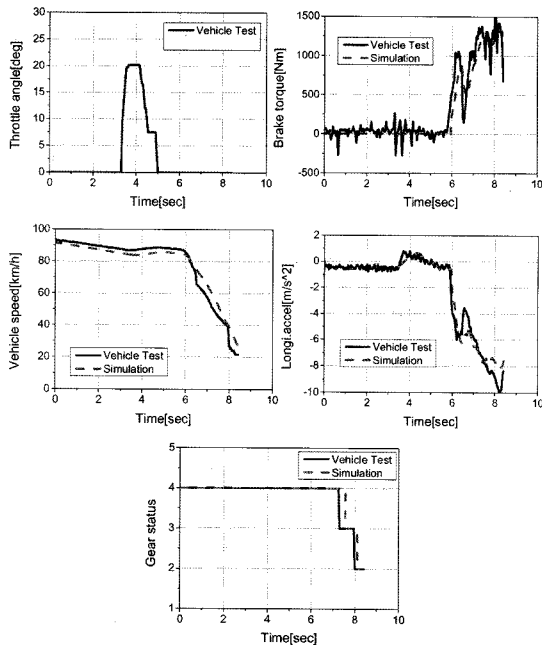


Fig. 6 Brake performance test

4.1.1 목표 가속도 설계

최적제어 이론을 적용하여 선행 차량 추종을 위한 목표 가속도를 설계하였다. 먼저 목표 상대거리

는 식 (1)과 같이 정의 하였다.

$$c_d = c_0 + \tau \cdot v_p \quad (1)$$

운전자 주행 실험 데이터 분석으로부터 구한 τ, c_0 를 사용하여 목표 상대거리를 계산하였다. 식 (2)는 선행차량과 제어차량 사이의 상태 방정식이다.

$$\begin{aligned} \dot{x} &= Ax + Bu + \Gamma w \\ &= \begin{bmatrix} 0 & -1 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ -1 \end{bmatrix} u + \begin{bmatrix} \tau \\ 1 \end{bmatrix} w \end{aligned} \quad (2)$$

식 (2)에서 u 는 제어 차량의 가속도 값으로 입력 요소이며, w 는 선행 차량의 가속도로 외란 요소이다. 시스템의 상태 변수는 식 (3)과 같다.

$$x^T = [x_1 \ x_2] = [(c_d - c) \ (v_p - v_s)] \quad (3)$$

식 (4)와 같은 성능 지수(Performance Index)를 정의하고 입력 $u = -kx$ 에서 게인 k 값은 성능 지수를 최소화 하도록 선정하였다.

$$J = \int_0^{\infty} (x^T Q x + u^T R u) dt \quad (4)$$

식 (4)의 Q, R 은 가중치 요소로서 추종 성능, 승차감, 주행 특성 등을 고려하여 선정 하였다.

목표 가속도는 식 (5)와 같이 정의 하였다.

$$a_d(t) = \begin{cases} a_{\max}(v) & \text{if } a_{\max}(v) < a_*(t) \\ a_*(t) & \text{elseif} \\ a_{\min}(v) & \text{if } a_*(t) < a_{\min}(v) \end{cases}, \quad (5)$$

$$a_*(t) = -k_1(v) \cdot (c_d(t) - c(t)) - k_2(v) \cdot (v_p(t) - v_s(t))$$

$$v = v_s(t)$$

여기서 게인, 최대 가속도 및 최소 가속도는 제어 차량의 속도에 따른 함수로 정의 하였다.

제어기 설계에 사용한 최소상대거리, 최적제어 게인, 최대 가속도 및 최소 가속도등의 제어 인자들을 제어기 설계 후 시뮬레이션 결과와 운전자 주행 실험 데이터를 비교, 분석하여 모든 속도 영역의 운전자 주행 거동과 유사하도록 조정 하였다.

4.1.2 스톱/브레이크 제어기 설계

스톱-브레이크 제어기는 4.1.1절에서 설계한 목표 가속도를 추종하기 위해 스톱-브레이크 액추에이터를 제어하는 것이다. 차량 동역학을 이용한 모델링을 통해 엔진 맵과 토크 컨버터 맵, 기어

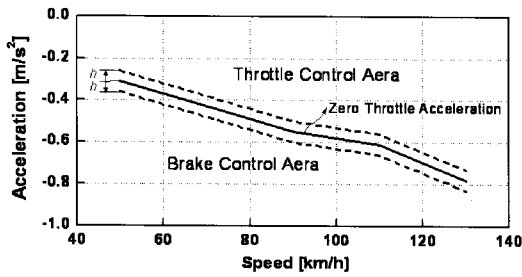


Fig. 7 Throttle/Brake switching logic

비 등을 이용하여 스로틀-브레이크 제어를 설계하였다.⁹⁻¹²⁾

Fig. 7은 차량의 Zero-Throttle 가속도 실험으로부터 구한 스로틀 브레이크 변환 곡선으로 특정 속도에서 스로틀 입력이 없을 경우 차량에서 발생한 최소 가속도를 나타낸다. 이러한 가속도 곡선에 적절한 경계층을 설정하여 잦은 스로틀-브레이크 변환을 방지하도록 설계하였다.

4.2 시뮬레이션을 통한 검증

ACC 차량 시뮬레이터가 얼마나 현실감이 있는지를 검증하기 위하여 Fig. 8과 같은 주행 환경 하에서 다양한 운전자들을 대상으로 시각적인 테스트를 수행하였다.

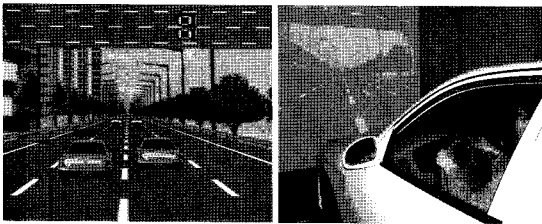


Fig. 8 ACC driving environment

본 연구에서 개발된 ACC 차량 시뮬레이터가 물리적으로 실제와 같은 차량응답을 보여주는지를 검증하기 위해 3절에서 검증된 차량 모델 및 4.1절에서 설계된 제어 기법을 이용한 결과와 실제 실험 데이터를 비교하였다. Fig. 9에서와 같이 실제 실험 데이터와 차량의 종방향 응답 즉, 상대거리, 속도 및 가속도 그리고 실제 차량의 엔진속도 등에서 시뮬레이터의 결과와 잘 일치함을 보여주고 있다.

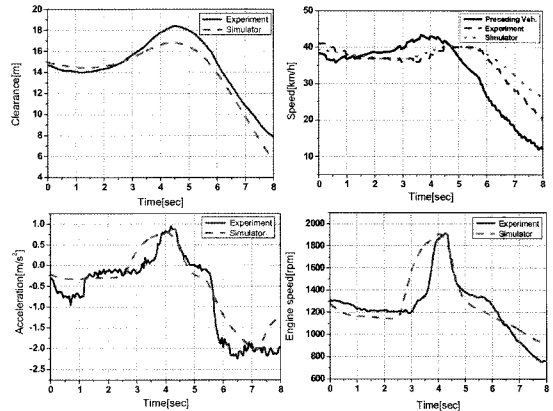


Fig. 9 Test data of ACC performance

5. 결론

본 연구에서는 ACC 주행 시험과 제어를 검증하기 위해 운전 시뮬레이터를 개발하였다. 현실감 있는 시뮬레이터를 위해, 기본적으로 실시간 제어가 가능한 RCP에 기본 개념을 두었다. 시뮬레이터는 여러 개의 하부 시스템의 결합으로 구성되어 있으므로 하부 시스템 간에 유기적으로 실시간 해석이 가능하여야만 한다. 따라서, 모델링 및 랜더링 또한 실시간이 가능한 그래픽 엔진을 채택하였으며 하부 시스템간의 통신시간 지연 및 오차를 줄이기 위해 차량 통신에서 많이 활용하는 CAN통신을 사용하였다. 실제 차량으로 실험한 환경을 시뮬레이터 상에서 그대로 재현하여 본 바 시뮬레이터상의 차량 응답이 실제 차량과 유사한 결과를 얻을 수 있었다.

후 기

이 연구는 과학 기술부 국가지정 연구실(NRL)사업의 지원에 의해 수행 되었습니다.

References

- 1) J. A. Greenberg and T. J. Park, "The Ford Driving Simulator," SAE 940176, 1994.
- 2) R. W. Allen, T. J. Rosenthal and Z. Parseghian, "Low Cost Driving Simulation for Research, Training, and Screening Application," SAE 950171, 1995.
- 3) G. P. Bertollini, C. M. Johnston, J. W. Kuiper,

- J. C. Kukula, M. A. Kulczycka and W. E. Thomas, "The General Motors Driving Simulator," SAE 940176, 1994.
- 4) L. D. Chen, Y. Papelis, G. Watson and D. Solis, "NADS at the University of Iowa : A tool for Driving Safety Research," Proc. 1st Human-Centered Transportation Simulation Conf., IA, 2001.
 - 5) C. Cho, "Development of a Real-Time Driving Simulator," Kookmin University, 1997.
 - 6) C. Chang, "Construction of Virtual Environment for a Vehicle Simulator," Transactions of KSAE, Vol.8, No.4, pp.158-168, 2000.
 - 7) I. Moon, "Intelligent Vehicle Cruise Control with Multi-target Tracking Algorithm," Han-yang University, 2004.
 - 8) K. Yi and D. Han, "A Vehicle Stop-and Go Control Strategy based on Human Drivers Driving Characteristics," Journal of Mechanical Science and Technology, Vol.19, No.4, pp. 993-1000, 2005.
 - 9) K. Yi, J. Hong and Y. D. Kwon, "A Vehicle Control Algorithm for Stop-and-Go Cruise Control," Proceedings of the Institution of Mechanical Engineers, Part D, Vol.215, No.10, pp.1099-1115, 2001.
 - 10) K. Yi, H. J. Yoon, K. Huh, D. Cho and I. Moon, "Implementation and Vehicle Tests of a Vehicle Stop-and-Go Cruise Control System," Proceedings of the Institution of Mechanical Engineers, Journal of Automobile Engineering, Part D, Vol.216, pp.537-544, 2002.
 - 11) K. Yi and Y. Kwon, "Vehicle-to-Vehicle Distance and Speed Control Using an Electronic Vacuum Booster," JSAE Review 22, pp.403-412, 2001.
 - 12) C. Lee and K. Yi, "An Investigation of Vehicle-to-Vehicle Distance Control Laws Using Hardware-in-the Loop Simulation," Journal of KSME, Part A, Vol.26, No.7, pp.1401-1407, 2002.