

Middle Ware 를 이용한 네트워크 시뮬레이터와 교통 시뮬레이터의 연동에 관한 연구

*강성재, *김태원, **백송남, *정재일
 *한양대학교 전자컴퓨터통신공학과
 **현대모비스 융합시스템연구팀
 e-mail : ksj10401@naver.com

A Study of Interworking between Network Simulator and Traffic Simulator through Middle Ware

*Seong-Jae Kang, *Tae-Won Kim, **Song-Nan Bae, *Jae-Il Jung
 *Dept. of Electronic and Computer Engineering, Hanyang University
 **Dept. of Research of Convergence System, Hyundai Mobis

요 약

최근 차량 네트워크를 통한 각종 서비스가 주목 받고 있으며 그 중 안전 서비스나 안전 서비스 어플리케이션 등의 테스트와 테스트를 위한 테스트베드에 대한 필요성이 생기고 있다. 본 논문은 네트워크 시뮬레이터와 교통 시뮬레이터 간의 연동을 통하여 실제 차량 통신 환경과 유사한 시나리오를 시뮬레이션 할 수 있도록 구현 하였다. 네트워크 시뮬레이터에서는 실제 차량의 이동성을 충분히 반영하기가 어렵고 대단위의 시나리오를 생성하기 어렵기 때문에 네트워크 시뮬레이터에 차량 이동성 정보 생성을 위하여 교통 시뮬레이터에서 생성 된 정보를 사용 할 수 있도록 하였다. 이러한 서로 다른 시뮬레이터간의 연동은 동기화의 문제가 존재하게 되는데 본 논문에서는 이를 해결하기 위한 각기 다른 세 가지의 동기화 방식을 연구 하였다.

1. 서론

최근들어 차량 통신의 필요성이 증가하고 있으며 V2X 통신을 통하여 안전 서비스를 제공하여 운전자의 안전성을 확보하기 위한 노력들이 증가하고 있다. 현재 여러 국가, 회사에서 차량 안전 서비스 또는 차량 안전 어플리케이션을 평가하기 위한 테스트를 하고 있는데, 실제 차량을 통해 하는 경우가 대부분이다. 이렇게 실차를 통한 테스트는 테스트 환경 구축에 따른 많은 비용 또는 자원을 필요로 하게 된다. 이러한 문제점을 해결하고자 본 논문에서는 차량 안전 서비스를 테스트 할 수 있도록 각기 다른 시뮬레이터간 연동을 통하여 차량 통신 시나리오를 생성하고 이를 통하여 V2X 통신을 이용한 안전 서비스나 안전 서비스 어플리케이션을 테스트 할 수 있는 환경을 구성 하였고 이러한 시뮬레이터간 연동에서 따라오게 되는 문제점들을 해결하기 위한 방안들을 제시 하였다. 또한 서로 다른 시뮬레이터간 동기화 문제를 해결하기 위한 방안을 제시하였고, 각각의 시나리오나 테스트 상황에 맞는 각각 다른 세 가지 동기화 방식에 대한 연구를 수행 하였다.

2. Middle Ware 동작 방식

차량 통신 환경을 시뮬레이션 하기 위한 기능을 두

가지 측면으로 나누어 보면 크게 네트워크 측면과 교통 상황 측면으로 구분 할 수 있다. 여기서 네트워크 측면은 통신을 하기 위한 통신 프로토콜과 지형 지물에 따른 통신의 변화 등을 반영하고 이에 따른 네트워크 상황을 시뮬레이션 할 수 있어야 하는 부분을 말한다. 교통 상황 측면은 차량의 이동 속도나 위치에 따라 통신 결과가 달라 질 수 있고, 현실적인 상황에서의 안전 서비스나 어플리케이션을 테스트 하기 위해서는 현실적인 차량의 움직임이 필요하게 되는 부분을 말하게 된다.

본 논문에서는 이러한 두 가지 측면을 모두 반영한 시뮬레이션 환경을 구성하기 위하여 교통 시뮬레이터인 SUMO 와 네트워크 시뮬레이터인 Qualnet 을 연동하기 위하여 Middle Ware 를 배치시켰으며 구조는 그림 1 과 같다.

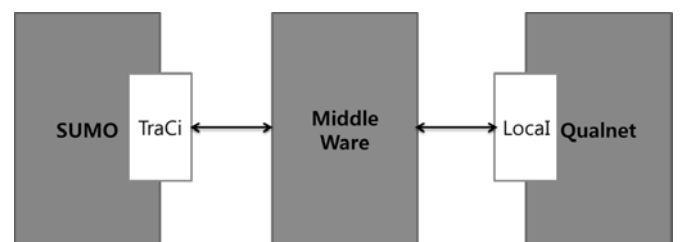


그림 1. Middle Ware 동작 방식

Middle Ware 는 두 시뮬레이터 간 원활한 상호작용과 네트워크 시뮬레이터의 부하를 분산하고자 설계했다. SUMO 와 Middle Ware 는 TCP/IP 를 사용하는 TraCI 라는 External Interface 로 연결하여 동작하게 되고, Location Server 와 QualNet 은 Connected UDP/IP 를 사용하는 Local 라는 External Interface 로 연결하여 동작한다. 또한 Location Server 와 차량 안전 서비스 어플리케이션은 Connected UDP/IP 를 사용하여 연결하여 동작한다. TraCI 를 제외한 나머지 인터페이스들에 Connected UDP 를 사용하여 시스템의 Processing Delay 를 줄이는 목적으로 설계했다.

3. 시뮬레이터간 동기화 방식

이처럼 서로 다른 시뮬레이터들을 연동하게 되면 동기화에 대한 문제가 필연적으로 발생하게 된다. 본 논문에서는 동기화 방식을 서로 다른 세 가지 방식으로 접근하였고 시나리오나 시뮬레이션 상황에 맞는 동기화를 사용 할 수 있도록 하였다.

제안하는 동기화 방식 중 첫 번째는 파일 기반 처리이다. 이는 두 시뮬레이터가 동시에 동작하도록 하는 실시간 동기화가 아니라 교통 시뮬레이터에서 각각의 노드들에 대한 이동성 정보를 Middle Ware 의 요청을 통해 수신 받아 네트워크 시뮬레이터에서 사용할 수 있는 형식의 파일로 출력하여 네트워크 시뮬레이터에서 사용하는 방식이다. 이러한 파일 기반 처리의 장점은 매번 교통 시뮬레이터를 동작 시킬 필요 없이 한번의 시뮬레이션으로 만들어진 파일을 통하여 필요한 시점에 네트워크 시뮬레이터에서 동작 시킬 수 있다는 것이다. 또한 교통 시뮬레이터의 동작 없이 반복적인 시뮬레이션이 가능 하기 때문에 교통 시뮬레이터를 설치하지 못한 장소에서 언제든 사용이 가능하고 연산량이 줄어들게 된다.

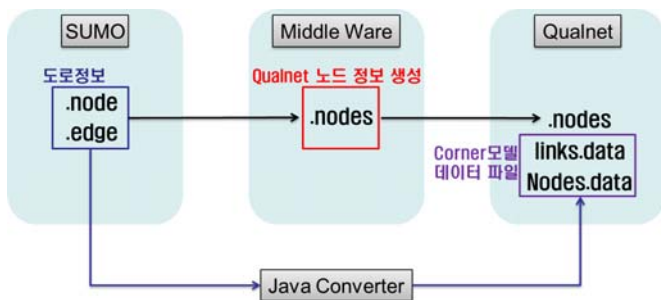


그림 2. 파일 기반 처리

두 번째는 Time Step 기반 처리이다. 이는 그림 3 과 같이 각각의 시뮬레이터에서 사용하는 시뮬레이션 타임을 동기화 하여 동작하는 방식이다. 파일 기반 처리와 유사하지만 실시간으로 교통 시뮬레이터와 네트워크 시뮬레이터가 연동 되는 방식이다.

Time Step 기반 동작 방식은 Middle Ware 에서는 원하는 시뮬레이션 시간의 데이터를 교통 시뮬레이터에 요청을 하게 되고 해당 시간의 각각의 노드에 대한 위치정보를 교통 시뮬레이터로부터 수신하게 된다. 이를 Middle Ware 가 네트워크 시뮬레이터에 맞는 형

식으로 변환하여 전달하면 네트워크 시뮬레이터는 각 노드의 위치를 적용하게 되고 동작이 완료되면 Middle Ware 에 수신 확인 응답을 보내 다음 시뮬레이션 시간의 데이터를 교통 시뮬레이터에 요청하게 된다. 이와 같이 동작 하게 되면 교통 시뮬레이터와 네트워크 시뮬레이터 간의 시뮬레이션 타임이 동기화 될 수 있으며 프로세서의 처리 능력에 따라 현실 시간보다 빠르게 시뮬레이션을 할 수 있다. 또한 프로세서의 능력이 떨어질 경우에는 현실 시간보다 느리게 동작하여 어떠한 환경에서도 잘 동작 할 수 있는 동기화 방식이다.



그림 3. Time Step 기반 처리

세 번째는 Real Time 기반 처리이다. 이는 각각의 시뮬레이터가 현실의 시간에 맞추어 동작 하는 것으로 시뮬레이션 시간을 현실의 시간 흐름에 맞추는 방식이다. 그림 4 와 같이 Time Step 기반과 달리 Middle Ware 와 네트워크 시뮬레이터가 Real Time 으로 동작 함을 볼 수 있다.

이 때 네트워크 시뮬레이터로부터 Middle Ware 로 수신 확인을 하지 않게 되는데 이는 각각의 시뮬레이터의 타임머가 Real Time 을 보장한다고 가정 하고 있기 때문이며 교통 시뮬레이터는 Middle Ware 의 요청에 따라 시뮬레이션이 진행 되기 때문에 Middle Ware 가 Real Time 으로 동작 하므로 Real Time 이 보장 되게 된다. Real Time 기반으로 동작을 하게 되면 시뮬레이터에 안전 서비스 어플리케이션을 연동 하는 것이 가능하며, 원하는 시나리오 상에서 안전 서비스 어플리케이션을 테스트 해 볼 수 있게 된다는 장점이 있다. 하지만 시뮬레이터가 동작하는 프로세서의 성능이 부족하여 실시간으로 시뮬레이션을 수행하지 못하게 되면 동작하기 어렵게 된다는 단점이 있다.

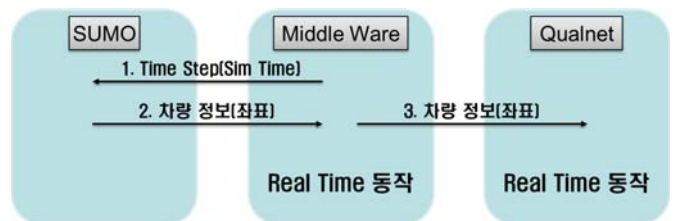


그림 4. Real Time 기반 처리

4. Interface Fault

네트워크 시뮬레이터는 시뮬레이션 시작 전에 미리 시뮬레이션에 참여 할 노드들을 초기화 해야 한다.

때문에 교통 시뮬레이터에서 현재 시뮬레이션 되고 있는 노드 수가 적더라도 앞으로 생성 될 노드들까지 미리 네트워크 시뮬레이터에 생성 되어 있기 때문에 교통 시뮬레이터에서 현재 시뮬레이션 되고 있지 않은 노드들도 네트워크 시뮬레이터에서는 동작 하게 된다. 이를 해결하고자 네트워크 시뮬레이터에서 지원하고 있는 인터페이스 on/off 기능을 하는 fault 파일을 통해 교통 시뮬레이터에서 현재 시뮬레이션 되고 있지 않은 노드들을 네트워크 시뮬레이터에서 off 하도록 구현 하였다. Middle Ware 는 파일 기반 처리를 통해 시뮬레이션을 하게 되면 매 step 마다 차량이 현재 시뮬레이션 되고 있는지를 체크하고 시뮬레이션 되고 있는 차량의 통신 인터페이스를 on 시킨다. 반대로 현재 시뮬레이션 되고 있지 않은 차량 노드의 통신 인터페이스를 off 함으로써 실제 시뮬레이션 되고 있지 않은 노드가 네트워크 시뮬레이션에 참여 하게 되는 문제를 해결 하였다.

5. 결론

본 논문에서는 이중의 시뮬레이터간 연동을 통하여 단일 시뮬레이터만으로 수행하기 힘들었던 차량 안전 서비스나 어플리케이션을 테스트 할 수 있는 환경을 구현 하였다. 또한 시뮬레이터간 연동에 있어서 필요한 동기화 방식에 대해 연구를 하였고 구현 하였다.

본 논문에서 제안하는 동기화 방식 중 Real Time 방식은 구현 한 시뮬레이터 환경이 약간의 시간 오차를 허용 하기 때문에 따로 타이밍을 맞추는 작업은 하지 않았지만 오차를 허용하지 않는 환경에서는 이에 대한 동기화에 대한 연구가 필요 할 것이다.

참고문헌

- [1] A. Wegener, M. Piorkowski, M. Raya, H. Hellbrück, S. Fischer, and J.-P. Hubaux. TraCI: an interface for coupling road traffic and network simulators. In Proc. CNS 2008, pages 155–163, April 2008.
- [2] QualNet network simulator. <http://www.scalable-networks.com/>.
- [3] D. Krajzewicz, G. Hertkorn, C. Rossel, and P. Wanger. Sumo (simulation of urban mobility); an open-source traffic simulation. In Proc. of the 4th Middle East Symposium on Simulation and Modelling (MESM2002), pages 183–187, Sep 2002.
- [4] SUMO Simulation of Urban MObility. <http://sumo.sourceforge.net/>.