

HLA 기반의 시뮬레이션 모형 개발 및 확장 연구

서혜숙*, 김태윤**
*고려대학교 컴퓨터학과
**고려대학교 컴퓨터학과
e-mail : suh@kida.re.kr

A Study of Extending for Simulation Model based on HLA

Heyi_Sook Suh*, Tae_Youn Kim**
*Dept. of Computer Science, Korea University
**Dept. of Computer Science, Korea University

요 약

최근 모델링 및 시뮬레이션(M&S) 체계 구축 관련 기술들이 미 국방부를 중심으로 활발히 연구됨은 물론 민간 사업으로 확산되는 등 그 수요가 급속히 증가하고 있다. 또한 기존의 모형 및 개발 모형들도 차세대 연동 체계인 RTI(Run_Time Infrastructure)를 모형에 적용함으로써 HLA(High Level Architecture)를 포함한 차세대 M&S 체계의 핵심 기술을 조기에 획득하여 이를 개발하고 운용하는 능력을 확보하는 것이 시급히 요구되고 있다. 본 논문에서는 첨단 정보 기술의 결집체인 HLA의 핵심 기술을 이용한 HLA 시뮬레이션 방법을 제시하였으며, HLA가 보장하는 상호 운용성과 재사용성을 모형 확장을 통해 확인하였다.

1. 서론

오늘날 M&S(Modeling& Simulation) 분야는 기반 기술에 대한 지속적인 개발 노력으로 관련 기술들은 고도로 성숙되었고 가격도 저렴해짐에 따라 관련된 컴포넌트(component)들을 통합하여 운용할 수 있는 환경을 쉽게 구축할 수 있게 되었다. 이러한 통합 환경 구축을 가능케 하는 기술이 바로 HLA(High Level Architecture)이며, HLA는 기존의 시뮬레이션 모형들을 개발하는 방법으로는 극복할 수 없는 문제점들을 해결해 준다.

기존의 개발 방법은 먼저 각 기관이나 사용자별로 제각기 개발하는 투자의 중복성 문제와 구축과 운용에 너무 많은 시간과 비용이 소요된다는 것이다. 또한 통합 환경 구축과 운용시의 문제점은 다른 유용한 M&S 컴포넌트들과 상호 연동이 불가능하고, 유지 및 확장이 어렵다는 점이다. 즉, 기존의 컴포넌트를 새로운 환경에 맞도록 확장하거나 변경해야 하는 절차가 필수적인데 어떤 경우에는 기존의 컴포넌트를 새로운 환경에 맞도록 개조하여 사용하는 것보다 완전히 시뮬레이션을 새롭게 개발하는 것이 더 쉬울 때도 있다. 이것은 바로 기존의 모형들이 재사용성과 상호 운용성이라는 두 가지 측면이 고려되지 않았기 때문인 것이다.

재사용성(reusability)이란 하나의 컴포넌트 시뮬레이션 모형이 다른 시뮬레이션 시나리오나 응용 프로그램에서 다시 사용될 수 있는 특성을 말한다. 재사용 가능한 시뮬레이션 컴포넌트는 기존의 코드를 다시 프로그램하는 재코딩(re-coding) 과정 없이도 다른 시

뮬레이션 컴포넌트들과 통합됨을 의미한다.

상호 운용성(interoperability)은 실시간 운용을 비롯하여 다른 유형의 분산 컴퓨팅 플랫폼(distributed computing platform) 상에서 시뮬레이션을 운용하는 경우에 사용자의 특별한 노력 없이도 다른 시뮬레이션 컴포넌트와 함께 잘 동작할 수 있는 능력을 말한다. 이것은 기존의 단일 프로그램 또는 단일 컴퓨터 환경에 대해 여러 시뮬레이션 컴포넌트들을 같이 연동시켜 함께 운용할 수 있는 방식이다. “네트워크가 바로 컴퓨터이다”라는 개념을 현실화시켜 주는 개념이다.

여러 종류의 많은 컴퓨터들이 분산되어 있으면서, 다양한 시뮬레이션 컴포넌트들을 이용하여 완전한 하나의 시뮬레이션을 수행하기 위해서는 컴포넌트들은 하나의 표준을 준수해야 하며, 또한 상호 운용성을 보장할 수 있어야 한다. 이러한 시뮬레이션 컴포넌트들의 재사용성과 상호 운용성을 보장하기 위한 표준화(standardization)를 목적으로 개발된 것이 HLA이다.

본 논문의 제 2 장에서는 관련 연구로서 HLA와 RTI에 대해 개괄적으로 살펴보고, 제 3 장에서는 HLA 시뮬레이션 방법을 제시하였으며, HLA가 보장하는 주요 개념인 상호 운용성과 재사용성을 확인하고자 모형 확장을 시도하였다.

2. 관련 연구

2.1 HLA(High Level Architecture)

차세대 시뮬레이션 기술 구조인 HLA(High Level Ar-

chitecture, 상위 체계 구조)는 DIS(Distributed Interactive Simulation)을 이은 차세대 네트워크 시뮬레이션의 표준 아키텍처로서 여러 종류의 다양한 시뮬레이션 컴포넌트(component)들로 이루어진 복합 시뮬레이션을 지원하기 위해 제정된 표준 프레임워크(architecture)이다. 즉, HLA는 모든 유형의 M&S에 대해 상호 운용성(interoperability)과 재사용성(reusability)을 보장하기 위한 공통 시뮬레이션 구조를 제공한다.

HLA 규약을 준수하는 시뮬레이션 모형을 HLA 컴플라이언트(HLA compliant)라고 하며, HLA 컴플라이언트 시뮬레이션으로 구현된 것을 페더레이트(federate)라고 한다. HLA 시뮬레이션은 이러한 여러 개의 페더레이트들이 모여 구성된 페더레이션(federation)으로 이루어진다. 이러한 시뮬레이션 모형의 수는 시뮬레이션이 진행되는 동안 다양하게 변할 수 있다.

다시 말해서, HLA를 사용하는 페더레이트들은 시뮬레이션이 실행되는 도중에 페더레이션에 자유롭게 참여(join)하거나 탈퇴(resign)함으로써 전체 시뮬레이션을 모듈 단위로 운영할 수 있다. 또한 페더레이션은 운용자 및 플레이어(player), 실제 하드웨어, 범용 소프트웨어에 대한 인터페이스를 포함하고 있다.

HLA는 정의적 측면에서 HLA 규칙(rule)과 인터페이스 명세(interface specification), 객체 모델 템플릿(OMT: object model template)이라는 세가지 요소로 구성된다.

1) HLA 규칙

HLA 규칙들은 HLA 컴플라이언트인 페더레이션 및 페더레이트의 역할과 상호 관계에 관한 10가지 기본 규칙을 담고 있다. 시뮬레이션에서의 상호 작용은 이 규칙을 따라야 한다. 페더레이션 규칙들은 문서화 요구사항을 포함한 페더레이션 생성, 객체 표현, 데이터 교환, 인터페이스 요구사항, 속성 소유권에 대한 규칙들로 이루어지며, 페더레이트에 대한 사항으로 문서화, 관련 객체 속성의 제어 및 전송, 그리고 시간 관리에 대한 규칙들로 이루어진다.

2) 인터페이스 명세(interface specification)

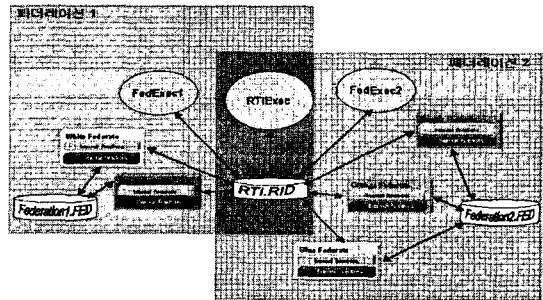
인터페이스 명세는 RTI(RunTime Infrastructure, 실시간 연동 체계)와 페더레이션 간의 기능적 인터페이스에 관한 표준을 정의하고 있으며, 이 명세는 현재 IEEE 1516.1 표준으로 제정되었다. 인터페이스 명세의 내용은 페더레이트와 페더레이션 간의 상호 작용에 관한 것으로서 RTI는 페더레이션(Federation), 선언(Declaration), 객체(Object), 소유권(Ownership), 데이터 분산(Data Distribution), 시간(Time) 관리(Management) 등의 6개 관리 서비스로 이루어진다.

3) 객체 모델 템플릿(OMT: Object Model Template)

재사용성과 상호 운용성을 위해 페더레이트는 모든 객체와 인터랙션(interaction)을 관리할 수 있어야 하고, 페더레이트의 외부에서도 이러한 객체와 인터랙션을 볼 수 있어야 하며, 이들을 객체 모델에 따라 세부적으로 기술할 수 있어야 한다. OMT는 바로 HLA에서의 객체 모델들을 문서화하기 위한 공통 표현 형식이다.

2.2 RTI(Run-Time Infrastructure)

실시간 연동 체계인 RTI는 HLA 인터페이스 명세에 따라 구현된 소프트웨어로서 RTI는 분산 운영 체제가 어플리케이션에 서비스를 제공하는 것과 유사한 방법으로 페더레이션을 구성하고 있는 페더레이트들에게 상호 연동에 필요한 서비스를 제공한다. <그림 1>은 페더레이션 1과 페더레이션 2라는 두개의 페더레이션이 연동되는 형태로서 하나의 RTI에 여러 개의 페더레이트로 구성된 다수의 페더레이션이 연동되는 상황을 보여주는 HLA 시스템의 예이다.



<그림 1> HLA를 사용하는 시스템의 예

3. HLA 시뮬레이션 모형

3.1 모형 개요

화학전 피해 상황 모형을 참조 모형으로 하여 HLA/RTI 기술을 적용한 화학 오염 확산 모형은 특정 지역에 화학 가스가 투하되었을 때, 그 지역의 피해 면적이 확산되어 가는 과정을 모의하게 된다. 여기에는 고유한 이름을 가지는 피해 지역명과 피해 면적이라는 두 가지 속성만을 갖는다. 시뮬레이션이 시작되면 화학 가스가 투하된 것으로 간주하고 사용자가 지정한 시간과 시간 간격에 따른 화학 오염의 확산 정도를 피해 면적의 변화로서 나타내게 되는데, 피해 면적의 증가는 각 시간 간격당 1%씩 증가하게 된다.

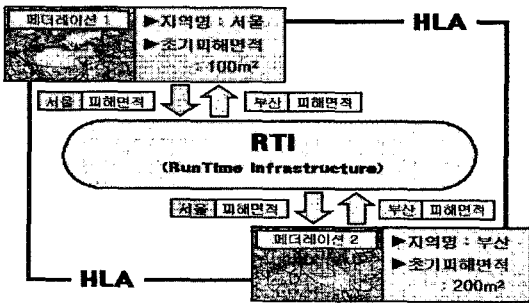
피해 지역을 나타내는 지역 페더레이트를 사용하고, 각 페더레이트는 한 피해 지역의 초기 피해 면적을 속성으로 가지며, 이 피해 면적은 시뮬레이션이 진행됨에 따라 지수적(exponentially)으로 증가한다. 페더레이트는 각 시간 tick마다 현재의 피해 면적을 공표(publish)하고 다른 페더레이트가 공표한 피해 면적을 수신한다.

3.2 HLA 시뮬레이션

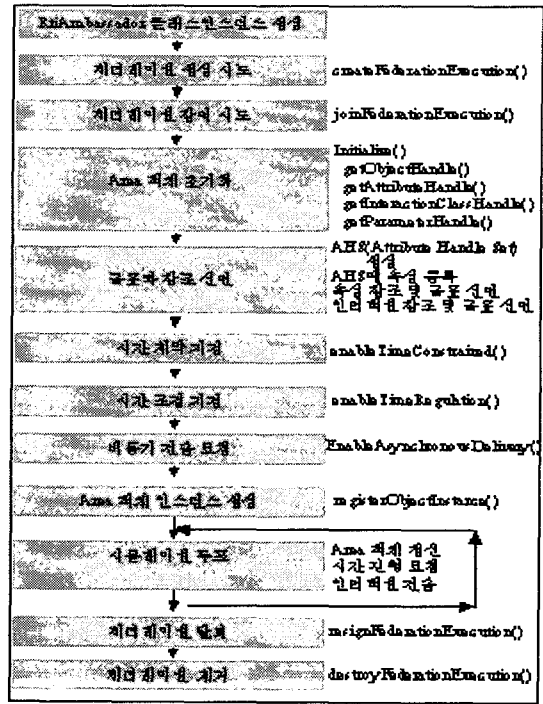
화학 오염 확산 모형을 HLA를 적용시키기 위한 시나리오는 다음과 같다. 1) 고유의 이름 값을 가지는 지역명으로서 페더레이트를 구성한다. 2) 모의 대상은 지역적으로 떨어져있는 두개의 지역으로 설정한다. 3) 모의 대상 지역에 대한 화학 가스 투하 시간 및 화학 지속 능력도 각기 다르게 한다. 4) 모의 대상 지역의 피해 면적을 시간 단계별로 모의한다. 5) 본 시뮬레이션에서는 모의 대상 지역에 대해 기타 다른

영향 요인은 없는 것으로 가정한다.

위의 시나리오에 따라 모의 대상지역에 대한 화학 오염 확산 모의를 하나의 시뮬레이션으로 규정, 이러한 상황을 페더레이션으로 정의한다. 또한 각 모의 지역을 하나의 독립적인 페더레이트로 구성한다. 페더레이션은 한 지역에 화학 가스가 투하되었을 때 개시되며, 각 지역의 화학 지속 시간이 다르므로 각 페더레이트의 시간 간격은 사용자에게 의해 각기 다르게 지정된다. 화학 가스 투하 시간도 다르게 모의되므로 각 페더레이트가 페더레이션에 참여하는 시간은 제각기 다르게 된다. 각 페더레이트는 피해 지역의 이름과 현재의 피해 면적을 서로 공표함으로써 서로 연동하게 된다.



<그림 2> 화학오염확산모형의 HLA시뮬레이션 구성도



<그림 4> 화학 오염 모형 페더레이트의 흐름도

3.3 페더레이트 구성

HLA를 적용시켜 화학 오염 확산 모형을 하나의 페더레이트로 구성하기 위해 3개의 클래스로 구성하였으며, 각 클래스는 JAVA 파일로 이루어졌다.

1) ChemicalJava 클래스(ChemicalJava.java)

페더레이트를 JAVA 버전으로 구현한 것으로 주 프로그램과 시뮬레이션의 이벤트 루프(event loop)에 해당된다.

2) Area 클래스 (Area.java):

실제로 시뮬레이션을 담당하는 클래스로 한 Area 클래스의 인스턴스는 하나의 피해 지역을 나타낸다.

3) ChemicalJavaFedamb 클래스(ChemicalFedamb.java)

FederateAmbassador 클래스에 해당되며 RTI:FederateAmbassador 클래스로부터 파생되어 가상 함수를 구현한다.

3.4 페더레이션 실행 데이터(FED) 파일 구성

페더레이션 실행 데이터(FED) 파일은 FOM에 의해 생성되거나 페더레이션 실행(Federation Execution) 동안에 RTI가 사용하는 정보를 포함한다.

3.5 페더레이션 실행

화학 오염 확산 모의에 참여하는 페더레이트의 동작 순서에 따라 흐름도를 나타내면 <그림 4>와 같다.

3.6 페더레이션 수행 결과

화학 오염 확산 모형은 두개의 페더레이트를 운영하

하므로 개별적으로 실행시킨다. HLA 개념에 따른 시뮬레이션을 수행한 결과, 실제 시뮬레이션이 수행되는 논리 부분을 제외한 나머지 부분은 표준화된 FOM이나 SOM, MOM에 따라 정의된 모든 자료나 인터랙션들이 RTI 서비스를 통해 송수신됨을 알 수 있었다. 바로 상호 운용성은 HLA 인터페이스 명세에 따라 구현된 RTI를 통해 보장받을 수 있음을 확인하였다.

3.7 모형의 확장: 재사용성이 보장되는가?

HLA가 보장하는 것 중의 하나는 기존 코드를 다시 사용할 수 있는 재사용성(reusability)이다. 재사용성은 기존 페더레이션에서 이를 새로운 목적에 부합되게 페더레이트를 확장해 봄으로써 확인할 수 있다.

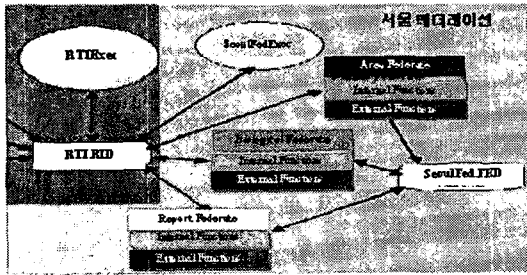
이를 위해 화학 오염 확산 모형에 새롭게 추가할 것은 두 가지로서 하나는 화학 가스가 투하된 지역에 일정시간이 지나면 박테리아가 성장한다는 가정에서 출발하여 그 과정을 모의하는 것이다. 또 하나는 기존 및 추가된 페더레이트로부터 통합된 결과물을 산출하는 결과 보고서를 만드는 것이다. 기존 페더레이트가 코드 변경없이 어떻게 확장되어 새로운 능력이 추가되는 지를 확인할 수 있는 경우이다.

1) 새로운 페더레이트의 추가

추가되는 기능에 대한 시나리오는 다음과 같다.

- 기존의 피해 지역 모의 페더레이트에 박테리아 성장을 모형화한다.
- 기존 및 추가된 모형으로부터 통합된 결과 보고서를 산출한다.

첫번째 기능을 위해서는 기존 페더레이트에서 가용한 자료로부터 생물학적 모의 페더레이트(Biological Federate)를 추가한다. 두번째 기능을 위해서는 결과 보고 페더레이트(Report Federate)를 추가하여 기존 페더레이트와 새로운 생물학적 페더레이트에서 산출한 결과를 통합한다. 이와 같이 기능이 추가되어 확장된 페더레이션의 구성도는 <그림 5>와 같다.



<그림 5> 확장된 페더레이션의 구성도

- 변경사항이 없는 기존의 피해 면적 모의 페더레이트(Area Federate)
- 추가된 생물학적 모의 페더레이트(Biological Federate)와 결과 보고 페더레이트(Report Federate)
- 이전의 FED 파일이 확장됨에 따라 새롭게 생성된 FED 파일(SeoulFed.fed)
- 이전에 사용되었던 것과 동일한 RTI(RTI.RID)

2) 기존의 페더레이트에 영향없이 FOM 확장

기존의 페더레이트인 피해 면적 모의 페더레이트는 객체, 속성, 상호작용, 매개변수 등의 자료들을 계속 사용할 수 있다. 생물학적 모의 페더레이트는 피해 면적에 대한 자료들을 사용해서 새로운 자료를 생산한다. 결과 보고 페더레이트는 피해 면적 페더레이트와 생물학적 모의 페더레이트에게 화학 피해 상황 모형에 대한 자료를 요구하게 된다.

이와 같이 최초 FED에서 정의된 과거 자료를 기존 및 새롭게 추가된 페더레이트에 전달하고 또한 이전 자료 및 새로운 자료를 추가된 페더레이트에 전달할 필요가 있다. 이를 처리하기 위해 HLA는 FOM의 확장을 허용한다. 생물학적 모의 페더레이트 자료인 박테리아 수(bacteriumCount)를 기존의 클래스에 새로운 클래스의 속성으로서 추가한다. 결과보고서 페더레이트는 각각의 피해 면적 모의 페더레이트와 생물학적 모의 페더레이트로부터 결과를 받아 쓰므로 FED에 추가할 내용은 없다. 기존의 클래스의 속성들과 객체 삭제 권한, 지역명, 피해 면적 등은 변함없이 그대로 남아있다. 결과적으로 박테리아 수를 FOM에 추가하여 확장된 페더레이션의 새로운 FED 파일이 되는 것이다.

기존 페더레이트의 클래스 속성에 대한 공표 및 참조는 바뀌지 않은 채 남아있게 된다. 바로 기존 페더레이트의 코드에 변경사항이 가해지지 않음을 말해주는 것이다. 중요한 것은 기존의 페더레이트는 기존 FED 파일을 사용하고 클래스에 새로 추가된 속성은 기존의 페더레이트에게는 영향을 주지 않는다는 점이다.

3) 소유권 공유

HLA에서 모델링의 책임은 인스턴스의 속성에 대한 소유를 의미한다. 인스턴스 속성에 대한 소유권의 이전은 각 페더레이트간의 협력을 의미하는 것이다. 새로운 인스턴스인 박테리아 수(bacterium Count)는 등록시에는 소유권이 없었으나, 생물학적 모의 페더레이트가 등록을 인지한 즉시 그 소유권을 갖게 되며, 그 값의 갱신 등의 책임이 있다.

4) 재사용성 확인

새로운 목적을 가진 페더레이트를 추가하여 페더레이션을 다시 구성해 봄으로써 HLA의 주요 개념 중의 하나인 재사용성을 확인하였다. 기존에 사용하였던 페더레이션 실행 데이터인 FED 파일에 새롭게 추가되는 객체의 속성을 삽입함으로써 기존의 페더레이트에는 아무런 변경사항을 가함이 없이 새롭게 페더레이션을 구성할 수 있었다.

4. 결론

본 논문은 최신 M&S 관련 기술의 조기 획득과 개발 능력 확보 및 적용 가능성 확인이라는 현실적 요구에 부응하기 위한 연구로서, 첫째, HLA 개념을 비롯한 각종 규칙 및 RTI의 구성 및 동작 등에 관해 설명하였다. 둘째, HLA 기술을 실제 모형에 적용시키는 HLA 시뮬레이션 방법을 제시하였으며, 모형 확장을 통해 HLA가 상호 운용성과 재사용성을 보장함을 확인하였다.

본 논문의 HLA/RTI 기술 연구는 최신 기술 확보 차원에서 기초자료로써 활용될 수 있을 것이다. 차후 연구로서는 M&S 체계는 장기적인 계획에 의거 개발되는 개념이므로 이와 관련된 신기술의 지속적인 획득 및 개념의 적용 가능성 확인이 반드시 필요하며, 응용 분야 탐구를 통해 고부가가치 창출을 도모해야 할 것이다.

참고문헌

1. 한국국방연구원, 국방모의분석체계 구축을 위한 상위체계구조(HLA) 기술 연구, 연구보고서, 1999.
2. 한국국방연구원, 차세대 연동체계를 적용한 시뮬레이션 모형 연동, 연구보고서, 2000.
3. DMSO, DoD HLA Run-Time Infrastructure Programmer's guide RTI 1.3 Ver.6, 1999, URL=http://www.dmsomil.
4. DMSO, DoD HLA Interface Specification Version 1.3, 1999, URL=http://www.dmsomil.
5. DMSO, HLA/RTI Verification, 2000, URL=http://www.dmsomil.
6. Frederick Kuhl, Richard Weatherly, Judith Dahmann, Creating Computer Simulation Systems, Prentice Hall PTR, 2000.
7. Rassul Ayani, Gary Tan, Farshad Moradi, Liang Xu, Yung-song Zhang, Distributed Real-time Simulation and High Level Architecture, 1999, URL=http://www.comp.nus.edu.sg/~rpsim
8. Roy Crosbie, John Zenor, High Level Architecture Training Module 1~6, California State University, 2000.