

DM&S 합성환경과 고수준 아키텍처 적용

이용호*

최상영**

DM&S Synthetic Environment and High Level Architecture with Some Applications

Lee, Young-Ho*

Choi, Sang-Yeong**

요약문

국방 모델 및 시뮬레이션 고수준 아키텍처(High Level Architecture ; HLA)는 여러 가지 군사 위게임 모델, 무기체계 시뮬레이터, 야지 훈련 등을 하나로 묶어 마치 전쟁 실험실처럼 가상 전쟁환경을 조성할 수 있는 전장 합성환경(Synthetic Environment)의 구축을 위한 아키텍처이다. 합성 환경은 위게임 모델과 시뮬레이터, 실 야지 기동훈련을 통합하는 정보기술기반 모의체계로서 기존의 위게임으로부터 진일보인 동시에 전쟁을 사실에 가까운 정도로 실험해 볼 수 있는 새로운 기회를 제공하고 있다. HLA는 합성환경 조성을 용이하도록 해주는 고수준 아키텍처로 시뮬레이션 개발을 위한 표준 구조, 설계규칙, 그리고 상호 인터페이스에 대하여 규정하고 있다. 본 연구는 DM&S 합성환경과 초보적인 형태로 HLA 개념을 적용한 예를 보여주는데 그 목적이 있다.

1. 서론

국방 모델링 및 시뮬레이션(Defense Modeling and Simulation ; DM&S)은 미래 군구조의 재설계, 첨단 무기체계 획득, 교육훈련 및 교리발전 등에 획기적인 발전을 가져올 수 있는 국방 핵심기술 중의 하나이다. 최근의 국방 모델링 및 시뮬레이션의 발전 특징은 1) 각종 시뮬레이션간의 상호 운용성을 보장하고, 2) 객체지향, 컴포넌트 기반으로 개발하여 시뮬레이션 소프트웨어의 재사용성을 촉진시키고, 3) 시뮬레이션 개발비용 절감과 다양한 요구에 부응할 수 있는 조합식(Composable) 솔루션을 제공하는 것이다. 특히 미국방성 획득 및 기술 차관실 산하에 DMSO(Defense Modeling and Simulation Office)를 설치한 후 1995년 10월부터 "국방 모델링 및 시뮬레이션 장기 발전계획"을 수립하여 추진중에 있으며, 이 중에서 HLA가 가장 역점을 두는 사업중의 하나이다. HLA는 국방 분야뿐만 아니라 민수 분야의 시뮬레이션에 대한 고수준 아키텍처로도 발전될 전망이다. 본 연구의 목적은 DM&S 합성환경과 HLA의 초보적 적용 예를 보여주는데 있다.

합성환경에 대한 최초의 시도는 1983년부터 1989년까지 미국방성 산하 DARPA (Defense Advanced Research Project Agency)에 의해 개발된 SIMNET (SIMulator NETworking)에서 찾을 수 있다. SIMNET은 최초 4개의 사이트에서 각각 100여대의 전투 장갑차 시뮬레이터를 네트워크로 연결하여 가상 전장(Virtual Battle)환경을 조성한 것이었다.

SIMNET이후 보다 다양한 가상전투 공간을 창출하기 위하여 1989년 미 육군의 STRICOM (Simulation, Training, Instrumentation Command)후원하에 분산 대화식 시뮬레이션(Distributed Interactive Simulation ; DIS) 개념으로 확장시켰다. 이는 지리적으로 멀리 떨어져 있는 서로 다른 종류의 시뮬레이션을 연결하고 기상 등을 포함한 전투영향 효과를 가상전투환경에 반영하도록 하였다. DIS와 더불어 고수준의 위게임들 사이의 통합환경에 대한 연구도 이루어졌다. 1989년 DARPA는 국방 시뮬레이션 인터넷(Defense Simulation Internet ; DSI)을 이용하여 ACE-89 시험을 시도하였다. 여기서 지상전 시뮬레이션 위게임의 분산처리를 실험하였는데 신뢰성과 분산 데이터 베이스의 비밀관성 등의 문제점이 도출되어 이를 보완하여 ALSP(Aggregated

2. DM&S 합성환경 발전

* 국방대학원 무기체계전공 석사과정(e_mail : youha2@chollian.net)

** 국방대학원 교수(e_mail : SYChoi@kndu.ac.kr)

Level Simulation Protocol)를 개발하였다. ALSP는 SIMNET의 특징을 수용하면서 시뮬레이션 시간 관리, 자료 관리, 그리고 시뮬레이션 아키텍처를 포함했고, 재래식 시뮬레이션들간의 상호 운용성을 지원하기 위한 기반을 제공하였다. 현재 한미연합사에 있는 위게임 연합체계는 ALSP를 적용한 좋은 예가 된다.

SIMNET, DIS, ALSP 이후에 미국방성은 시뮬레이션의 적용범위를 교리 및 전술개발, 합동훈련, 무기체계 획득 및 분석 등 광범위한 분야로 확장하고 서로 다른 시뮬레이션 모델의 연합구성은 물론이고 무기체계, C4I체계까지 연합할 수 있도록 하며, 시뮬레이션의 재사용성을 향상시키고자 하고 있다. 따라서 앞서 언급된 "국방 모델링 및 시뮬레이션 상거 발전계획"에서는 SIMNET, DIS, ALSP의 차원을 넘는 공통기술구도(Common Technical Framework ; CTF)를 제공하여 모든 시뮬레이션은 이 구도에 부합되도록 표준화하고 1999년 이후부터는 이 구도에 부합되지 않는 시뮬레이션에는 재정적 지원을 중단하고 2001년 이후에는 연합실행에 관여할 수 없도록 한다는 것이다.

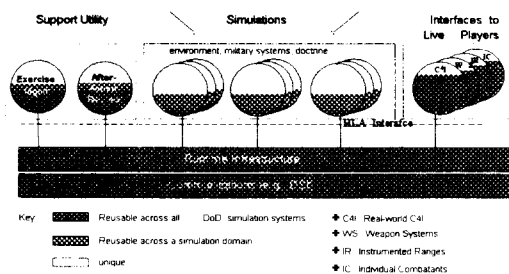
3. 고수준 아키텍처(HLA)

가. 개요

HLA는 CTF에서 미국방성이 가장 우선적인 노력을 집중하고 있는 것 중의 하나로서 HLA는 모델링 및 시뮬레이션 공동체에서 표준으로 사용되며, 이와 같은 HLA를 포함한 차기 시뮬레이션 모습은 <그림 1>과 같다.

나. 특징

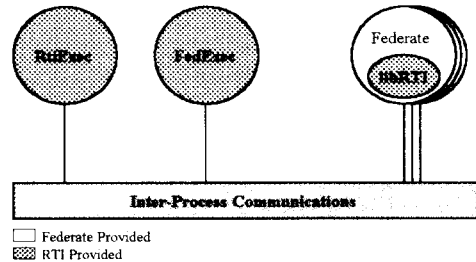
HLA는 DIS와는 다른 아키텍처를 사용한다. DIS 아키텍처에서는 임의의 시뮬레이션은 다수의 서로 다른 시뮬레이션이 실행되는 네트워크에 시뮬레이션 노드 접속을 통하여 분산 실행에 참여할 수 있고,



<그림 1> HLA를 포함한 차기 시뮬레이션 모습

분산 실행에 참여하는 각 시뮬레이션은 PDU(Protocol Data Unit) 형태의 데이터를 송수신 함으로써 자율적으로 시뮬레이션이 수행된다. 이와는 대조적으로 HLA에서는 각 시뮬레이션이 HLA RTI (Run-Time Infrastructure)라는 일종의 공통 소프트웨어

모듈을 통하여 가상세계 혹은 공간 내에서 수행되는 정보를 주고받게 함으로써 모든 참여 시뮬레이션이 하부 기반구조 하에서 연합될 수 있도록 하고 있다. HLA에서 연합 실행 멤버를 페드레이트(Federate)라고 하고 이들과 기타 지원요소를 포함한 모임을 페드레이션(Federation)이라고 한다. 각 페드레이트는 RTI에 개별적으로 접속되는데 RTI는 페드레이션 내에 분산되어 있거나 중앙에 위치할 수 있다. HLA에서는 DIS에서 브로드캐스팅(Broadcasting)을 통해 정보를 주고받는 것과는 달리 각 페드레이트가 RTI에게 자신이 받고자 하는 정보를 선언하여 필요한 정보만을 선택적으로 수신한다. 그리고 RTI는 HLA를 위한 많은 서비스를 제공해주는데, 이들 서비스 기능에는 페드레이션 관리, 페드레이트 관리, 선언 관리, 객체 관리, 소유권 관리, 시간 관리, 그리고 데이터 분배 관리 등이 있다. RTI는 이를 통하여 페드레이션 실행 서비스를 제공하고 페드레이션 참여, 그리고 페드레이트간의 상호작용 서비스를 지원한다. RTI는 HLA 인터페이스 규격의 소프트웨어적인 구현이며, 1996년 12월에 RTI F.0가 발표된 이후에 1998년 10월 현재 RTI 1.3 릴리즈 4 까지 다수의 플랫폼(예, Windows NT, SUN Solaris, SGI IRIX 등)과 언어(C++, Java, Ada 등)를 지원하는 라이브러리 형태로 공개되고 있다. RTI는 아래 <그림 2>와 같이 두 가지 프로세서 즉, RtiExec(RTI Executive)와 FedExec (Federation Executive), 그리고 페드레이트간을 연결하는 라이브러리(libRTI)로 이루어진 분산 체계라고 할 수 있다.



<그림 2> RTI 1.3 시스템 아키텍처

RtiExec는 페드레이션 실행에서 생성과 소멸을 관리하는 프로세서이고, FedExec는 각 페드레이션 실행별 하나가 존재하며 이는 실행간 페드레이트들의 가입과 탈퇴를 관리한다. HLA와 호환되지 않는 기존 시뮬레이션으로 페드레이션에 참여하고자 한다면 기존 시뮬레이션과 RTI와 인터페이스를 제공해주는 마들웨어 혹은 게이트웨이 등과 같은 디바이스들이 필요하게 된다.

다. HLA 정의

HLA는 3가지 요소 1) 페드레이션 규칙(Federation Rules), 2) 인터페이스 규격(Interface Specification), 3) 객체 모델 템플레이트(Object Model

Template ; OMT)에 의해 정의된다.

· **페드레이션 규칙** : 페드레이션 실행간에 페드레이트의 적절한 상호작용을 달성하기 위해서 준수해야 할 규칙으로 페드레이트의 책임과 RTI의 책임에 대하여 기술한다.

· **인터페이스 규격** : RTI와 페드레이트 사이의 6가지 인터페이스 서비스 기능을 정의한다.

· **객체 모델 템플레이트** : 각 페드레이션과 페드레이트에 대하여 HLA 객체 모델에 포함되어야 할 정보 기록에 대한 공통 방법을 규정한다.

페드레이션 규칙은 전체 10가지로 여기에는 페드레이션이 준수 해야할 규칙 5가지와 페드레이트가 준수 해야할 규칙 5가지가 있다.

페드레이션 규칙

- 페드레이션은 HLA OMT 에 부합되도록 문서화된 FOM(Federation Object Model)을 가져야 한다.
- 페드레이션에서, FOM에 있는 객체의 모든 표현은 페드레이트에 있어야 하며, RTI 에 있어서는 않된다.
- 페드레이션 실행 동안에, 페드레이트간의 모든 FOM 데이터 교환은 RTI를 통해서 이루어진다.
- 페드레이션 집행 동안에, RTI와 상호작용 할 때는 HLA 인터페이스 규격을 준수해야 한다.
- 페드레이션 집행 동안에, 객체 인스턴스의 속성은 임의 시간에 오직 하나의 페드레이트에 의해 소유되어야 한다.

페드레이트 규칙

- 페드레이트는 OMT 서식에 부합되는 문서화된 SOM을 가져야 한다.
- 페드레이트는 자신의 SOM에 있는 객체 속성을 갱신시키고 반영할 수 있어야 한다. 그리고 SOM에서 정의된 대로, SOM 객체 상호작용을 외부적으로 송신하거나 수신할 수 있어야 한다.
- 페드레이트는 그들의 SOM에 명시된 바와 같이, 페드레이션 실행 동안에 속성 소유권을 동적으로 인도하거나 접수할 수 있어야 한다.
- 페드레이트는 SOM에서 정의된 대로, 객체속성 갱신을 제공하는 조건을 변경시킬 수 있어야 한다.
- 페드레이트는 다른 페드레이션과 자료를 교환 하면서, 지역시간을 관리할 수 있어야 한다.

인터페이스 규격은 각 페드레이트간의 상호 운용성을 보장하기 위해서 사용하는 공통 컴포넌트 기반인 RTI와 페드레이트 사이의 인터페이스, 그리고 RTI가 제공하는 서비스 기능을 정의하고 있다. 이 서비스 기능은 6가지로 구분되며, 페드레이트는 제공되는 서비스 기능을 이용하여 다른 페드레이트와 상호작용을 하게 된다. 서비스 기능의 종류는 다음과 같다.

- **페드레이션 관리(Federation Management)** : 페드레이션, 페드레이트의 생성, 동적 통제, 실행 및 제거를 위한 서비스 제공.
- **선언 관리(Declaration Management)** : 객체 속성과 상호작용 선언.
- **객체 관리(Object Management)**, : 객체 인스턴스의 생성 및 제거, 속성 및 상호작용의 통신.
- **소유권 관리(Ownership Management)** : 객체 속성의 소유권 이양 관리.
- **시간 관리(Time Management)**, : 시물레이션 시간 진행의 조정 협조.
- **데이터 분배 관리(Data Distribution Management)** : 효율적인 데이터 라우팅 지원.

객체 모델 템플레이트는 HLA 객체 모델 정보를 표현하기 위한 공통의 방법이고 페드레이트에서 외부와 가시성이 있는 모든 객체와의 상호작용에 대한 표현은 이 방법을 준수해야 한다. 객체 모델에는 FOM(Federation Object Model), SOM(Simulation Object Model), 그리고 MOM(Management Object Model)이 있다. FOM은 각 페드레이션에 1개씩 있고 모든 공유 정보(예, 객체, 상호작용 등)를 기술하고 페드레이션 상호간의 약정(예, 데이터 인코딩 계획, 송수신 방법 등)을 규정한다. SOM은 각 페드레이트에 1개씩 존재하고 페드레이트 내부 운용을 기술하고 또한 외부와 관계가 있는 객체와 상호작용을 나타낸다. MOM은 페드레이션 관리에 사용되는 객체와 상호작용을 식별한다.

4. 적용

HLA개념을 도입하여 분산 방공 시물레이션 모델(Distributed Air Defense Simulation Model ; DADSim) 에서 비행 시나리오 전시를 구현한 예를 보이고자 한다.

이를 위해 다음의 몇 가지 개념을 정의한다.

· **공간 및 좌표계**

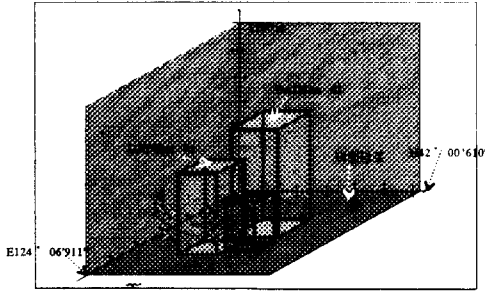
비행 시물레이션에서 공간은 중요한 개념이다. 여기서는 아래 <그림 3>과 같이 3차원 공간을 의미하고, “광역공간”과 “지역공간”으로 구분한다. “광역공간”의 범위는 한반도 지역 전체인 반면에 “지역공간”의 범위는 한반도의 임의 전장지역에 한정된다. 광역공간에서 임의 위치를 나타낼 때 사용하는 좌표계를 “광역 좌표계”라고 하고, 지역공간에서 임의 위치를 나타낼 때 사용하는 좌표계를 “지역 좌표계”라고 한다. 광역 좌표계의 기준은 WGS-84체계를 따른다. 그러나 “지역 좌표계”는 대상 지역마다 서로 다르며 사용자 정의 기준점을 사용할 수 있다.

· **공표공간 및 서명공간**

공표(Publication)와 서명(Subscription)은 HLA 개념에서 비롯되었다. 이는 분산실행에 있어서 임의 시물레이션 애플리케이션이 자신의 변수 값을 다른 애플리케이션에 송신하고자 할 때, 이 변수에 대해

서 공표를 하고, 반면에 수신하고자 하는 측에서는 서명한다. DADSim에서 광역공간이 곧 공표공간이 되고, 지역공간은 각 DADSim 멤버의 서명공간이 된다.

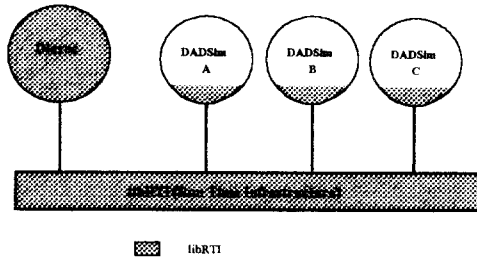
따라서 <그림 3>에서 보는 바와 같이 한반도 광역 공간에서 항공기가 이동할 때, 그 위치가 어떤 DADSim의 서명공간으로 들어가면 항공기 정보가 그 DADSim으로 송신되고, 해당 DADSim에서 교전이 모의된다. 만약 교전간에 항공기가 제압되었다면, 항공기 정보는 더 이상 유효하지 않게 된다.



<그림 3> DADSim 공간 및 좌표계

아래의 <그림 4>에서는 분산 항공 시물레이션 모델의 구조를 보여주고 있다.

모델 구조는 DADSim, Diexec, libRTI로 이루어져 있다. DADSim은 독립적으로 혹은 서로 다른 DADSim들과 분산 실행을 할 수 있다.

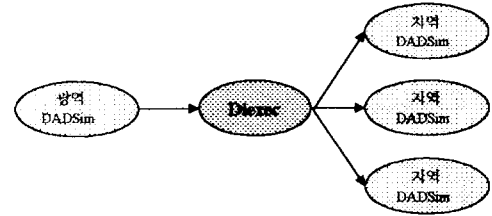


<그림 4> 분산 항공 시물레이션 모델 구조

DADSim은 Active 형과 Passive 형으로 구분한다. Active DADSim은 광역공간에서 광역 좌표계로 항공기 비행을 모의한다. Passive DADSim은 지역 공간에서 지역 좌표계로 항공기를 나타낸다. 그리고 Active DADSim은 광역공간을 공표할 수 있고, Passive DADSim은 광역공간의 임의 공간에 대하여 서명할 수 있다.

Diexec는 HLA에서 FedExec와 유사한 역할을 하며, DADSim의 분산실행 참여를 관리한다. 그리고 항공 시물레이션에서는 선언 관리, 객체 관리, 데이터 분배 관리를 한다. 선언 관리는 광역공간에서 공표 혹은 서명 관리를 한다. 객체 관리를 통하여 선언 관리에서 선언된 항공기 정보들을 갱신 또는 삭제하고 상호작용을 송수신하는 일을 수행한다. 데이

터 분배 관리에서는 DADSim의 공표와 서명 지역에 따라 항공기 정보를 선택적으로 멀티캐스팅(Multi-casting) 한다. 이를 그림으로 나타내면 <그림 5>와 같다.



<그림 5> DADSim 멀티캐스트 그림

그리고 libRTI는 Diexec 서비스를 각 DADSim으로 확장할 수 있는 라이브러리를 제공한다.

5. 결론

본 연구에서는 국방 M&S 합성환경과 HLA에 대해 소개하였고 HLA 개념을 초보적인 수준에서 적용한 예를 나타내었는데 이 분산 항공 시물레이션은 차후 HLA하부 기반인 RTI를 적용하여 더욱 발전시킬 계획이다. 비록 HLA가 미국방성에서 제시된 것이지만, 그 잠재력과 주변여건으로 보아 국방 분야 뿐만 아니라 민수분야에 걸쳐 분산 시물레이션의 고수준 아키텍처로 발전될 전망이다. 따라서 이에 대한 많은 연구가 요구된다.

참고문헌

- 1) DMSO, High Level Architecture Run-Time Infrastructure Programmers Guide Version 1.0, 미 DoD, 1997. 5. 15.
- 2) DMSO, HLA Interface Specification, Version 1.3, 1998. 4. <http://www.dmsomil>
- 3) DMSO, HLA Rules Version 1.2, 1997.8. <http://www.dmsomil>
- 4) DMSO, HLA Object Model Development Process and Supporting Tools, 1998. 4. <http://www.dmsomil>
- 5) DMSO, Federate Testing Process, <http://www.dmsomil>
- 6) DMSO, Standard for Distributed Interactive Simulation Application Protocols Version 2.0, University of Central Florida Division of Sponsored Research, 1994. 3. 16.
- 7) Terry Quatrani, Visual Modeling with Rational Rose and UML, 1998. 3.
- 8) 이상무, HLA 기반 분산 비행 시물레이션 모델 설계 및 구현, 국방대학원, 석사논문, 1998.
- 9) 최상영, 군사 시물레이션 모델링 기술, 국방대학원, 참고교재, 1996. 8.
- 10) 최상영 외, 대공방어체계 분석모델 개발연구, 무기체계 개념 연구센터 1차년도 보고서, 1998.
- 11) 한·미 국방 모델링/모의실험 워크샵 자료, 국방연구원/주한미군합동군사업무단, 1997.10
- 12) 고희동 외 3인, 분산 시물레이션을 위한 기반기술 소개, 한국과학기술원, 1997.10.