

단일 및 분산 컴퓨팅 환경에서 모델 연동을 위한 플러그인 방식의 인터페이스 설계 방안 연구

심준용*, 이용현*, 김세환*

*LIG넥스원(주) Maritime연구소

e-mail: jyshim79@lignex1.com

A Study on the Design Idea for Plug-in based Simulation Network on Single and Distributed Computing Environment

Jun-Yong Shim*, Yong-Heon Lee*, Sae-Hwan Kim*

*Maritime R&D Center, LIG Nex1 Co., Ltd.

요 약

국방 분야의 모델링 및 시뮬레이션(Modeling & Simulation) 분야에서 재사용성 및 상호운용성 확보는 기 개발된 모델을 연동하여 시뮬레이션을 확장할 수 있다는 측면에서 핵심이 된다. 이러한 핵심기술 확보를 위해 시뮬레이션 소프트웨어의 공통 컴포넌트를 설계하고, 표준 연동 미들웨어인 Run-Time Infrastructure(RTI)를 적용한 프레임워크가 개발되었다. 한편, 시뮬레이션 환경은 목적에 따라 시뮬레이션 모델의 기능이 달라지거나, 동일한 모델이 분산 컴퓨팅뿐만 아니라 단일 컴퓨팅 환경에서도 수행되도록 요구된다. 본 논문은 이러한 요구사항을 만족시킬 수 있도록 모델 연동을 위한 플러그인 방식의 인터페이스 설계 방안을 제시하고, 분산 시뮬레이션의 동일한 모델을 단일 시스템에서 시뮬레이션 할 경우 모델 컴포넌트 간 데이터 교환 및 메시지 순서화를 제공할 수 있는 메커니즘을 기술한다.

1. 서론

국방 소프트웨어 산업에서는 무기체계 또는 비 무기체계 개발 시 구성 요소의 재사용을 늘리고 체계 간 상호연동을 제공하는 기술이 핵심이 되고 있다. 특히, 국방 분야의 모델링 및 시뮬레이션(Modeling & Simulation, 이하 M&S) 시스템 구축 시 개발 모델의 이식성을 높이고, 시뮬레이션의 상호 연동성을 높이기 위한 기술 표준으로 High Level Architecture(HLA)[1,2]의 적용을 늘리고 있다. 또한, HLA 기반의 시뮬레이션 소프트웨어 개발에 요구되는 기능을 제공하기 위해 M&S 프레임워크[3,4]가 개발되었으며, 개발 프레임워크는 멀티 쓰레딩, 시간 스케줄링 및 네트워킹과 관련된 공통의 서비스 구현과 구성 요소 간 메시지 교환 프로토콜을 제공함으로써 신뢰성 및 재사용성의 향상을 가져왔다.

한편, 시뮬레이션 환경은 목적에 따라 동일한 모델이 분산 컴퓨팅 환경뿐만 아니라 단일 컴퓨팅 환경에서도 수행될 수 있으며, 시뮬레이션 모델에서 제공하는 기능이 달라질 수 있다. 예를 들어, 시뮬레이션 모델 각각의 수행 알고리즘과 시나리오가 요구되는 경우 실행 환경의 부하를 줄이면서 모델 간 연동을 수행할 수 있도록 분산 컴퓨팅 환경이 제공되어야 하는 반면, 각 모델들의 인터페이스 수준을 시뮬레이션 할 경우에는 단일 컴퓨팅 환경에서의 통합 시뮬레이션 수행이 분산 컴퓨팅 환경보다 효율적이

다. 이때 분산 시뮬레이션 환경인 M&S 프레임워크의 연동 모델을 단일 컴퓨팅 환경에서 운용하기 위해서는 기존 프레임워크 엔진과 모델 간 인터페이스 수정이 불가피하며, 기능 변경에 따른 코드 수정으로 인해 재 컴파일 및 링크 작업이 수행되어야 한다. 플러그인(Plug-in) 방식을 사용하면 기 개발된 모의 소프트웨어의 코드 수정 없이 모델의 교환만으로 시뮬레이션 수행이 가능하며[5,6], 동일한 모델을 기반으로 단일 및 분산 컴퓨팅 환경에서 시뮬레이션 구축이 용이해진다. 이러한 방식을 적용하기 위해서 모델과 시뮬레이션 프레임워크 사이의 인터페이스 규약이 필요하며, 교환 메시지의 구조가 정의되어야 한다.

본 논문은 플러그인 방식의 모델을 기반으로 단일 및 분산 컴퓨팅 환경에서 시뮬레이션 연동이 가능한 인터페이스 설계 방안을 제시한다. 특히, Dynamic Linked Library(DLL) 기반의 플러그인 컴포넌트 개발을 위해 컴포넌트 간 인터페이스 규약 및 메시지 구조를 정의하고, 단일 컴퓨팅 환경에서 통합된 모델들의 시뮬레이션 수행 시 메시지 순서화(Total Ordering)를 위한 스케줄링 방법을 기술한다. 논문의 구성은 다음과 같다. 2장은 단일 및 분산 컴퓨팅 환경에서 시뮬레이션 모델의 연동 방법을 살펴보고, 3장에서 기 개발된 프레임워크의 구조와 단일 컴퓨팅 환경에서의 시뮬레이션 개발 시 문제점을 알아본다. 4장은 단일 및 분산 컴퓨팅 환경에서 시뮬레이션 모델을

연동할 수 있도록 플러그인 방식의 인터페이스 구조를 제시하고, 연동 모델의 구현 방법을 보여준다. 마지막으로 5장에서 결론을 다룬다.

2. 시뮬레이션 모델 연동

2.1 연동 개념

시뮬레이션 연동은 시뮬레이터 간 데이터 교환과 시뮬레이션 데이터 순서화를 위한 시각 동기화를 제공하는 것이다. 즉, 시뮬레이터들은 정보를 교환할 수 있도록 공유 객체 및 이벤트 전송을 사용하면서 발생 사건의 인과성(Causality)을 보장하기 위해 논리 시간을 사용한다. 각 시뮬레이터는 다른 시뮬레이터들의 시간 진행을 알 수 없기 때문에 일반적인 데이터 분산 시스템과 다르게 동기화 방법이 반드시 제공되어야 한다.

2.2 단일 및 분산 컴퓨팅 환경의 모델연동

모델을 시뮬레이션하기 위한 연동 환경은 시뮬레이션 목적 또는 개발 방식에 따라 단일 컴퓨팅(단일 프로세스)과 분산 컴퓨팅(다중 프로세스) 환경으로 구분된다. 단일 컴퓨팅 환경에서 모델 연동은 모델을 구성하는 모듈 또는 컴포넌트 간 함수를 호출하는 방식과 공유 메모리를 사용하여 데이터를 교환하는 방식을 사용할 수 있다. 그림 1은 단일 프로세스에서 모듈 간 데이터 교환 방식을 보여준다.

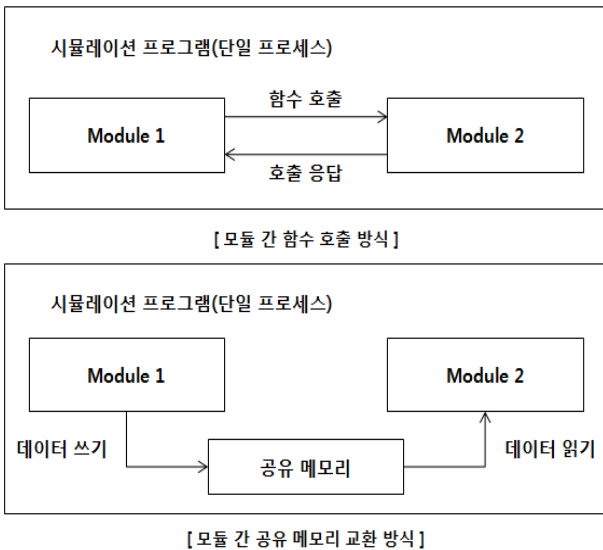


그림 1 단일 컴퓨팅 환경의 메시지 교환 방식

함수 호출 방식은 구현이 간단하지만 수행 규칙에 따른 모듈 간 동기화 방식으로 데이터를 교환하기 때문에 데이터의 양이 많을 경우 데이터 처리에 의한 지연이 발생할 수 있다. 반면, 공유 메모리 방식은 모듈 간 수행 규칙 없이 공유 메모리를 통해 데이터를 교환할 수 있으며, 데이터 쓰기과 읽기가 비동기 방식으로 처리되므로 많은 양의 데이터 처리 시 메시지 교환에 의한 부하를 줄일 수 있다. 한편, 분산 컴퓨팅 환경에서 모델 연동은 소켓 프로그램

램을 사용하거나 분산 미들웨어를 사용하여 구현할 수 있다. 그림 2는 분산 컴퓨팅 환경의 교환 방식을 보여준다.

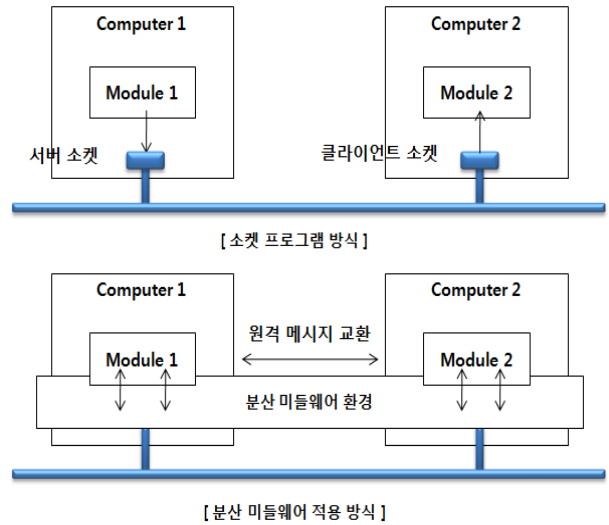


그림 2 분산 컴퓨팅 환경의 메시지 교환 방식

일반적으로 분산 시스템의 규모가 커짐에 따라 소프트웨어 연결 복잡도가 높아지므로 규모 확장성(Scalability)을 고려하면 소켓 프로그램보다 분산 미들웨어를 사용한 시스템이 시험 및 유지보수에 용이하다.

3. M&S 프레임워크 구조 및 재사용 문제

3.1 M&S 프레임워크 구조

HLA 기반의 무기 체계 시뮬레이션 개발을 위한 M&S 프레임워크는 사용자 GUI와 상호작용을 위한 UI관리자, 구현 모델의 정보를 관리하는 모의 관리자, 원격 개체를 제어 및 감시하기 위한 통제 관리자, 프레임워크 내부 관리자를 통제하는 설정통제 관리자, 실 장비와 연결된 링크 모의 개체 정보를 관리하는 파라미터 관리자, 시뮬레이션 시나리오를 관리하는 시나리오 관리자 그리고 RTI(Run Time Infrastructure)를 통해 분산 시뮬레이션 환경을 제공하는 시뮬레이션 관리자로 구성되어 있다.

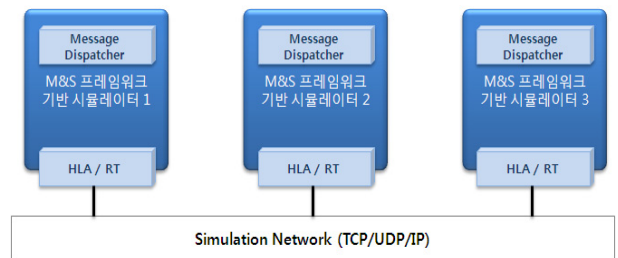


그림 3 프레임워크 상호작용 구조

그림 3과 같이 메시지 기반 프로토콜인 메시지 디스패처(Message Dispatcher)는 각 관리자 간 정보 교환을 제공하며, HLA/RT는 시뮬레이터들의 메시지 교환을 제공

한다. 메시지 디스패처는 공유 메모리를 사용하여 모듈 간 메시지 복사를 줄이고 메시지 포인터를 관리함으로써 잘못된 메모리 접근을 막아준다[7]. 따라서 메시지 디스패처를 기반으로 시뮬레이션 모델의 인터페이스를 구현할 경우 데이터 교환이 필요한 각 관리자의 식별자를 프레임워크의 라우팅 테이블에 작성한 후 교환 메시지를 생성하면 된다. 이러한 모델들은 HLA/RTI를 통해 원격지에 있는 모델과 통신을 수행한다.

3.2 단일 컴퓨팅 환경에서 모델 재사용 문제

시뮬레이션 목적이 모델의 알고리즘이나 시나리오 수행이 아닌 다른 모델들과의 상호작용만을 모의한다면 분산 컴퓨팅 환경이 아닌 단일 컴퓨팅 환경에서의 통합 시뮬레이션 방식이 시스템 구축을 용이하게 하고, 비용절감 얻을 수 있다. M&S 프레임워크의 개발 모델은 각 관리 모듈과 통합되어 있으므로 그림 4와 같이 코드 수준의 재사용이 가능하며, 통합 환경과 상호작용하기 위한 인터페이스를 구현해야 한다.

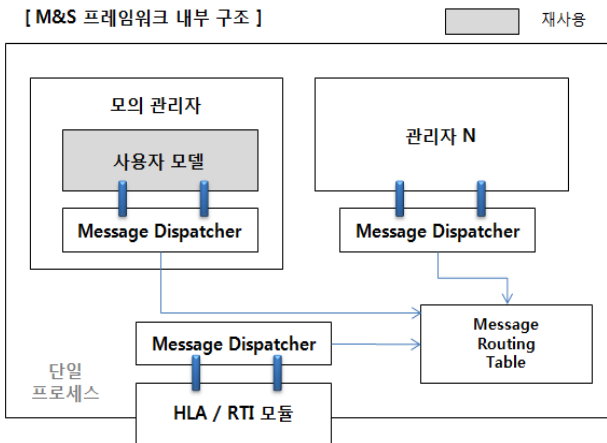


그림 4 사용자 모델의 재사용 단위

M&S 프레임워크는 HLA/RTI의 시간 관리(Time Management) 서비스를 사용하여 메시지 순서화를 위한 동기화 메커니즘을 제공하고 있다. 따라서 단일 컴퓨팅 환경에서 모델들을 통합 시뮬레이션하기 위해서는 메시지 동기화를 위한 설계가 고려되어야 한다.

4. 모델 연동을 위한 플러그인 기반 인터페이스

분산 컴퓨팅 환경의 구현 모델을 단일 컴퓨팅 환경에서의 재사용을 용이하게 하기 위해서 코드 수준이 아닌 컴포넌트 수준의 재사용 방식이 필요하다. 논문에서 제안하는 컴포넌트 플러그인 방식은 정의된 인터페이스와 메시지 구조를 사용하여 프레임워크 환경에 쉽게 조립 또는 분해하여 시뮬레이터를 개발할 수 있다. 기존 프레임워크의 각 관리자는 플러그인이 가능한 DLL로 구현되며, Publish-Subscribe 방식[8]의 데이터 교환 구조를 적용함

으로써 교환 메시지 타입을 기반으로 상호작용이 가능하다. 제안된 플러그인 기반의 모델 연동을 제공할 수 있는 프레임워크의 구조는 그림 5와 같다.

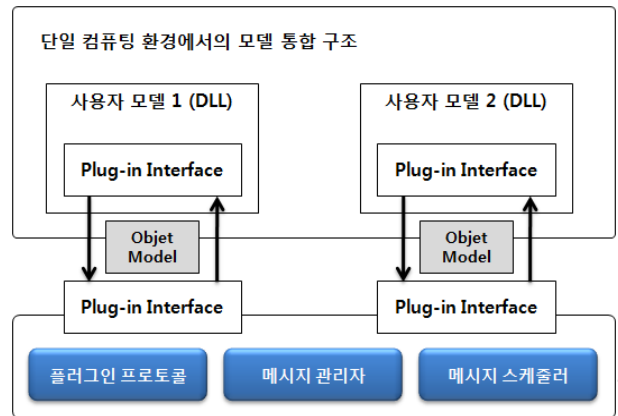


그림 5 모델 연동을 위한 플러그인 인터페이스 구조

제안 구조는 플러그인 프로토콜, 메시지 관리자 및 메시지 스케줄러로 구성된다. 플러그인 프로토콜은 DLL의 플러그인 인터페이스를 정의하고, Publish-Subscribe 메커니즘을 구현하며, 메시지 관리자는 DLL간 교환 정보를 기술(Description)할 수 있도록 XML 형태의 객체 모델(Object Model)을 정의하고 해석한다. 특히, 기존 메시지 디스패처의 메시지 라우팅 테이블을 동적으로 생성하게 함으로써 코드 수정 없이 모델의 추가나 삭제가 가능하다. 메시지 스케줄러는 시뮬레이션 모델 간 교환 정보의 동기화 기능을 제공한다. 따라서 분산 컴퓨팅 환경에서 단일 컴퓨팅 환경으로 연동 체계를 변경할 경우 프레임워크 및 연동 모델의 인터페이스 수정 없이 객체 모델을 재작성하면 된다.

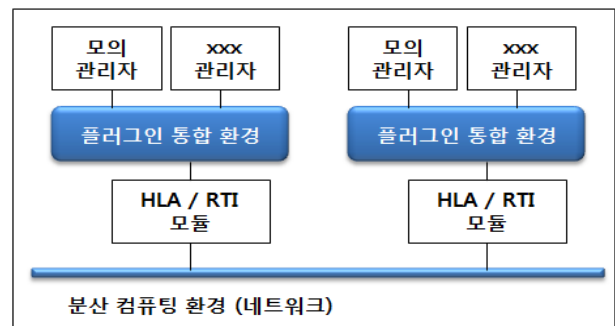
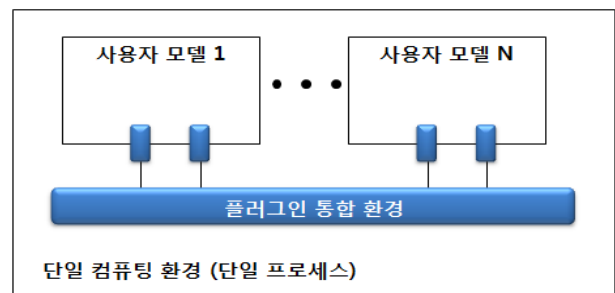


그림 6 플러그인 기반 모델 연동 환경

그림 6은 플러그인 통합 환경이 단일 및 분산 컴퓨팅 환경에서 적용되는 방법을 보여준다. 기존 M&S 프레임워크는 모델과 HLA/RTI 기능이 단일 컴포넌트로 구현되어 있는 반면, 제안 구조를 적용한 프레임워크는 모든 기능을 플러그인 기반의 독립 컴포넌트로 구현하여 조립 형태로 시뮬레이터를 구성한다. 단일 컴퓨팅 환경에서 통합 시뮬레이션 되는 경우 동기화를 위한 메시지 진행 방법은 그림 7의 절차를 수행한다.

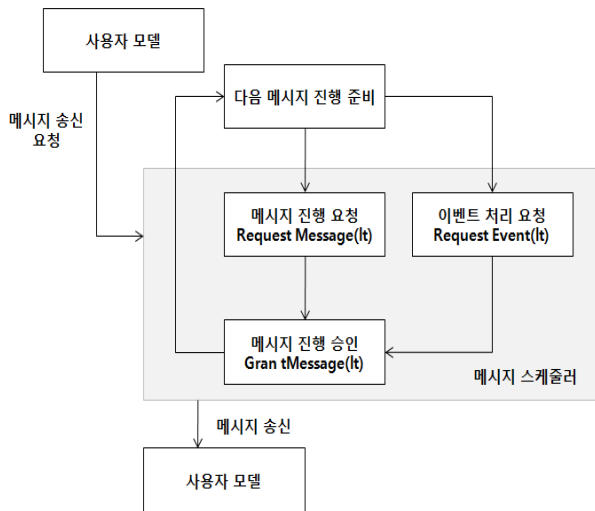


그림 7 메시지 스케줄러 동작 방식

시뮬레이터 개발자는 객체 모델 작성 시 전송 메시지에 대한 스케줄링 사용여부(Enable/Disable)를 설정한다. 즉, HLA/RTI의 시간 관리 서비스를 사용한다면 메시지 스케줄링을 Disable 모드로 변경하면 된다.

5. 결론

HLA 기반 시뮬레이션 소프트웨어 개발을 위한 M&S 프레임워크는 메시지 디스패처와 HLA/RTI를 통해 원격지의 시뮬레이션 모델과 상호작용 할 수 있도록 기능을 지원한다. 한편, 시뮬레이션 목적에 따라 분산 컴퓨팅 환경의 모델들은 단일 컴퓨팅 환경에서 통합 운용될 수 있으며, 모델의 기능을 변경하여 동일한 환경에서 시뮬레이션 될 수 있다. 이때 기존 프레임워크의 구조는 모델과 시뮬레이션 엔진이 메시지 디스패처를 기반으로 코드 결합을 갖기 때문에 재사용 시 인터페이스 수정이 불가피하다.

본 논문은 재사용 단위를 DLL 컴포넌트로 제공함으로써 프레임워크 환경과 쉽게 조립 및 분해가 가능한 플러그인 인터페이스 구조를 제시했다. 플러그인 인터페이스는 교환 정보에 대한 XML 형태의 객체 모델을 작성하고 해석함으로써 메시지 라우팅 테이블을 자동으로 생성시킨다. 또한, 분산 시뮬레이션과 동일한 환경을 제공하기 위해 교환 메시지의 동기화 메커니즘을 구현한 메시지 스케줄링 방법을 기술하고, 제안 인터페이스를 통해서 개발된 모델이 단일 및 분산 컴퓨팅 환경에서 동일한 구조로 사용될

수 있음을 보였다.

참고문헌

- [1] IEEE, "IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) - Framework and Rules." IEEE Standard Mo: 1516-2000.
- [2] IEEE, "IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) - Federate Interface Specification." IEEE Standard Mo: 1516.1-2000.
- [3] 심준용, 진정훈, 김세환, "M&S Framework를 적용한 효율적인 분산객체 통신모듈 설계", 한국소프트웨어공학회 학술대회 논문집 제10권 1호, 2008
- [4] 김성용, 윤근호, 김세환, 정태일, "데이터 링크 시뮬레이터 설계를 위한 M&S 프레임워크", 한국군사과학기술학회 종합학술대회 논문집, 2008
- [5] Argent, R.M., "An overview of model integration for environmental applications-components, frameworks and semantics", Environmental Modeling & Software 19, 219-234, 2004
- [6] Jan Himmelspach and Adelinde M. Uhrmacher, "Plug'n simulate." Proceedings of the 40th Annual Simulation Symposium (ANSS '07), pp.137-143, Mar. 2007.
- [7] http://www.boost.org/doc/libs/1_44_0/index.html
- [8] Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal, "Pattern-Oriented Software Architecture Volume 1", Wiley, 2002.