

IEEE 1516 HLA/RTI를 이용한 복합 시스템의 다측면적인 모델링 방법론

김병수[†] · 김탁곤

Multifaceted Modeling Methodology for System of Systems using IEEE 1516 HLA/RTI

Byeong Soo Kim[†] · Tag Gon Kim

ABSTRACT

System Entity Structure/Model Base (SES/MB) enhances organizing model families and storing and reusing model components in the multifaceted system modeling. However, the real world can be described not only an individual system but also a collection of those systems, which is called system of systems (SoS). Because SES/MB has a limitation to simulate the SoS using HLA/RTI, an extended framework is required to simulate it. Therefore, this paper proposes System of Systems Entity Structure/Federate Base (SoSES/FB) for simulation in a distributed environment (HLA/RTI). The proposed method provides the library of federates (FB) and System of System Entity Structure (SoSES) formalism, which represents structural knowledge of a collection of simulators. It also provides a methodology for the development process of distributed simulation. The paper introduces the anti-missile defense simulation using the proposed SoSES/FB.

Key words : System of systems, Multifaceted modeling simulation, Interoperation of simulator, HLA/RTI

요 약

System Entity Structure/Model Base (SES/MB) 프레임워크는 하나의 시스템이 가지는 모든 대안들을 총체적으로 표현하는 다측면적인 시스템 모델링 방법이다. 이는 모델들의 재사용성을 향상시키고, 단일 시스템의 구조를 효과적으로 관리할 수 있다. 하지만 실세계는 단일 시스템으로만 이루어진 것이 아니라, 개별적인 시스템들의 집합체인 복합 시스템으로 이루어진 경우가 많다. SES/MB는 분산 환경에서 복합 시스템의 시물레이션 하는데 한계가 있기 때문에, 복합 시스템을 위한 확장된 방법이 필요하다. 본 논문은 분산 환경에서 시물레이터들 간의 시물레이션을 위한 System of Systems Entity Structure/Federate Base (SoSES/FB)를 제안한다. 제안하는 방법은 국제 연동 표준인 HLA/RTI에 기반하며, 시물레이터 집합의 구조를 표현하는 System of Systems Entity Structure (SoSES) 형식론과 시물레이터들의 라이브러리(FB)를 포함하는 환경을 제공한다. 또한 제안된 SoSES/FB를 분산 시물레이션 개발 과정에 적용한 방법론을 제안한다. 본 논문은 제안하는 방법을 다섯 개의 독립적인 시물레이터들로 구성되어 있는 대공방어 시물레이션에 적용한 사례를 소개한다.

주요어 : 복합 시스템, 다측면적인 모델링 시물레이션, 시물레이터 연동, HLA/RTI

1. 서론

실제 세계에서 특정 시스템은 다양한 구조와 여러 가지 대안들로 구성된다. 다측면적인 시스템 모델링 (Multifaceted System Modeling)이란 시스템의 모든 구조와 대안들을 총체적으로 표현하는 모델링 방식이다 (Zeigler, 1984). 다측면적인 시스템 모델링 방법을 통해

Received: 24 February 2017, Revised: 26 May 2017,

Accepted: 26 May 2017

[†] Corresponding Author: Byeong Soo Kim

E-mail: kevinzzang@kaist.ac.kr

Dept. of Electrical Engineering, KAIST, Daejeon, Korea

하나의 시스템은 시스템의 구성 요소와 어떠한 대안들로 이루어져 있는지를 표현하게 된다. 이러한 다측면적인 시스템 모델링 시뮬레이션을 위한 대표적인 프레임워크로 System Entity Structure/Model Base(SES/MB) 프레임워크가 존재한다(Kim et al., 1990). SES/MB 프레임워크는 다양한 대안들을 가지는 시스템의 구조를 SES 형식론을 통해 구성하고, 이를 실제 시스템의 모델들이 저장되어 있는 MB를 통해 합성하여 시뮬레이터를 생성한다. SES/MB를 이용하면 모델의 구조 변경이 용이하며, 이를 통해 문제 해결에 유용하게 사용 가능하다. 또한 기존에 개발된 모델들을 재사용하여 모델 재사용성을 높여주며, 방법론적인 시스템 모델링 시뮬레이션을 가능하게 한다.

이러한 SES/MB를 통하여 단독 시스템의 모델링 및 시뮬레이션이 가능하지만, 실제 세계는 단독 시스템뿐만 아니라, 단독 시스템들의 유기적인 관계를 통해 구성된 복합 시스템(System of Systems)인 구성된 경우가 많다. 복합 시스템은 독립적인 시스템들의 집합체로, 예를 들어, 독립적인 컴퓨터 시스템을 하나의 단일 시스템으로 본다면, 그 컴퓨터 시스템들이 네트워크를 통해서 복합적으로 연결된 시스템은 복합 시스템이라고 할 수 있다. 복합 시스템은 단일 환경뿐만 아니라 많은 경우에 분산 환경에서 동작하기 때문에, 이를 시뮬레이션하기 위해서 필요한 내부 시스템 간의 연동 정보를 포함하고 있지 않은 기존의 SES/MB를 이용해서 모델링 시뮬레이션을 할 수 없다. 그렇기 때문에 분산 환경에서 복합 시스템의 다측면적인 모델링 및 시뮬레이션을 위해서는 SES/MB 개념을 확장한 프레임워크와 이를 통해 모델을 개발할 수 있는 새로운 시뮬레이터 개발 과정이 필요하다.

따라서 본 논문에서는 복합 시스템의 다측면적인 모델링 시뮬레이션을 위해서 IEEE 1516 High Level Architecture/Run Time Infrastructure(HLA/RTI) 기반의 System of Systems Entity Structure/Federate Base (SoSES/FB) 방법론을 제안한다. HLA/RTI는 서로 다른 시뮬레이터 간의 연동을 위한 표준 미들웨어로서, 본 논문에서는 복합 시스템을 구성하는 시스템 간의 연동을 위해서 HLA/RTI를 이용한다.

본 논문의 구성은 다음과 같다. 2장에서는 SES/MB 프레임워크와 HLA/RTI에 대해 알아보고, 3장에서는 복합 시스템을 위한 SoSES 프레임워크를 제안한다. 4장에서는 제안하는 프레임워크에 기반한 SoSES/FB 개발 방법론을 제안하며, 5장에서는 제안된 방법을 대공방어 시뮬레이션에 적용한 실험 결과를 보여주며, 마지막으로 6장에서는 결론 및 향후 연구 방향에 대해 정리한다.

2. 관련 연구

2.1 SES/MB

앞 장에서도 언급했듯이, 단일 시스템의 다측면적인 시스템 모델링 방법으로 SES/MB가 사용된다. SES/MB는 시스템의 구조적인 정보를 담고 있는 SES 트리를 저장하는 SES Base와 시스템의 동작을 위한 실제 모델들을 저장하고 있는 Model Base로 구성된다.

먼저 SES 형식론은 트리 구조를 이용하여 시스템의 구성 관계, 연결 관계, 분류 관계의 세 가지 정보를 명시하는 형식론이다. 예를 들어 단일 서버 큐잉 시스템은 Figure 1과 같이 SES 형식론으로 나타낼 수 있다. 단일 서버 큐잉 시스템이 BP라는 결합 모델과 Gen이라는 원자 모델로 구성되어 있고, 모델 BP는 모델 Buff와 Proc으로 구성된다. Buff 모델은 FIFO와 LIFO의 두 가지 대안을 가질 수 있으며, Proc 모델은 다양한 proc 모델로 구성 된다. 또한 GBP 모델과 BP, Gen 모델, 그리고 BP 모델과 Buff, Proc 모델에 대한 연결 관계를 알 수 있다. 이렇듯 SES 형식론을 통해서 시스템, 즉 하나의 시뮬레이션 모델의 정보를 한 눈에 알 수 있다.

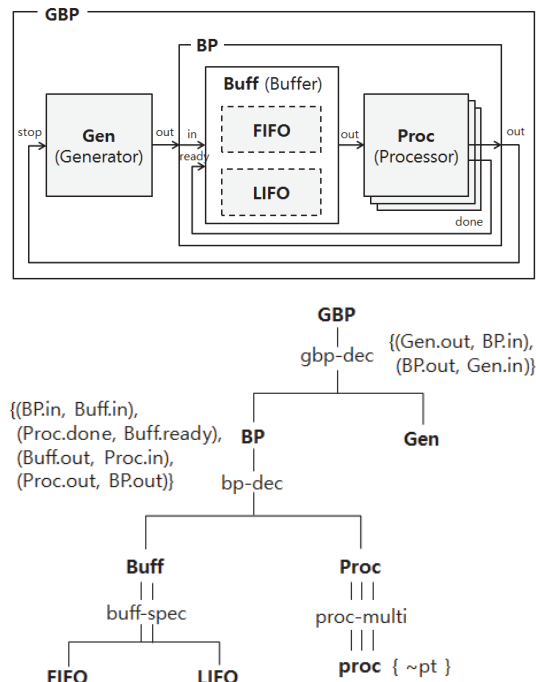


Fig. 1. SES tree of single server queuing system

시스템을 SES를 통해 트리 형태로 표현한 후에, 사용

자는 여러 가지 대안들 중 원하는 모델을 선택하는 과정을 거치고, Figure 2와 같이 Model Base에서 선택된 모델들을 불러들여 하나의 시뮬레이터가 완성된다. SES/MB는 형식론적인 모델링이 가능하며, 모델의 재사용성을 높이고 시뮬레이터를 합성하는 과정을 자동화해줌으로써 편리한 개발이 가능하다(Lee, 1994).

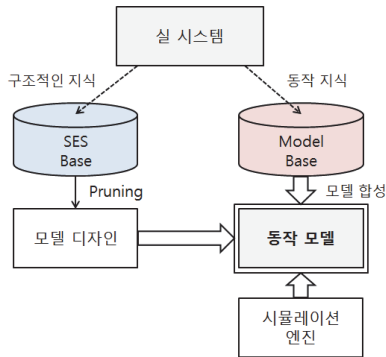


Fig. 2. Concept of SES/MB

SES/MB에 대한 연구는 다년간에 걸쳐 지속적으로 수행되어 왔으며, 이러한 연구에는 Scheme으로 SES를 표현하고 시뮬레이션 환경으로 DEVS-Scheme을 사용한 지식기반 모델 자동 합성 방법론(Kim and Zeigler, 1989), DB SQL로 SES를 표현하고 시뮬레이션 환경으로 DEVSim++를 사용한 데이터베이스 성능 분석 (Park et al., 1997), DB SQL로 SES를 표현하고 시뮬레이션 환경으로 SystemC를 사용한 디지털 시스템 설계(Kim and Kim, 2006) 등이 있다. 이렇듯 다양한 언어 및 환경을 이용하여 SES/MB 프레임워크에 대한 연구가 다각도로 진행되어 왔다. 그러나 이러한 연구는 모두 단일 시스템을 대상으로 하여 진행되어 왔고, 실제로 이러한 다측면적인 시스템 모델링 방식을 시스템 간의 연동이 고려된 복합 시스템에 적용하려는 연구가 진행 되었지만(Kim et al., 2013), 형식론, 개발 환경 및 개발 과정을 포함한 전체적인 방법론에 대한 제안이 요구된다,

2.2 IEEE 1516 HLA/RTI

복합 시스템은 여러 개의 독립적인 시스템들로 구성되어 있고, 이러한 시스템 사이에서는 정보 교환과 그 정보를 각 시스템에서 사용할 수 있는 능력인 연동이 필요하다. 즉, 연동을 통해 복합 시스템을 구성하는 시뮬레이터들 간의 정보 교환과 시간 동기화가 가능하다. 이러한 시뮬레이터 간의 연동을 위해서는 정해진 규약이 필요하다.

이러한 규약의 하나로 IEEE 1516 표준으로 채택된 국제 표준 연동 구조인 HLA가 존재한다. HLA는 서로 다른 시뮬레이터 간의 연동을 위해서 연동 참여 시스템들 사이에 지켜야할 규칙이나 인터페이스, 사용객체 모델 등에 대한 전체적인 아키텍처를 의미한다. RTI는 HLA를 만족하도록 구현된 분산 시뮬레이션용 표준 미들웨어이다.

HLA에서는 HLA 규격을 만족하는 단독 시뮬레이터를 페더레이트(Federate)라고 하며, 페더레이트들의 집합체를 페더레이션(Federation)이라고 명명한다. HLA에서 페더레이트들은 하나의 시뮬레이션을 수행하기 위해서 페더레이션을 구성하거나 다른 페더레이션에 참여함으로써 연동을 하게 된다(IEEE Std., 2000).

HLA에서는 페더레이션 내에서 공유되는 객체와 상호작용을 기술하는 페더레이션 객체 모델(Federation Object Model, FOM)과 페더레이트가 외부에 제공할 객체와 상호작용을 기술하는 시뮬레이션 객체 모델(Simulation Object Model, SOM)이 정의되어야 한다(IEEE Std., 2001). SOM은 각 페더레이트별로 정의가 되어 있으며 자신이 주고받을 데이터들을 정의하고 있으며, FOM은 SOM들을 통합한 페더레이션 데이터이다(Kim, 2011).

2.3 IEEE 1516.3 FEDEP

FEDEP(Federation Development and Execution Process)은 IEEE 1516.3에 정의된 페더레이션 개발 및 실행 표준 프로세스다. HLA/RTI 기반의 분산 시스템은 FEDEP의 절차에 따라 개발이 되며(Ahn and Kim, 2010), Figure 3과 같이 크게 설계 단계, 구현 단계, 실행 단계로 총 3 단계의 과정을 통해서 진행된다.

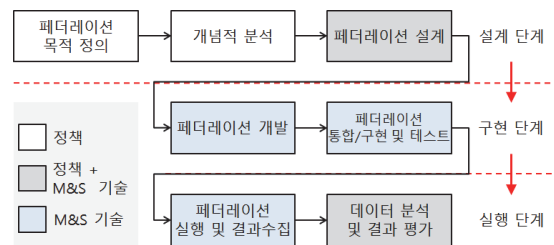


Fig. 3. Federation development and execution process

기존에 다양한 분산 시뮬레이터들이 FEDEP을 이용하여 개발되었으며, 기개발된 시뮬레이터들을 활용하여 다측면적인 시스템 모델링 및 시뮬레이션을 수행하기 위해서 본 논문에서는 SoSES/FB 개발 방법론을 제안한다.

3. SoSES/FB 프레임워크

이번 장에서는 독립적인 시스템들로 구성된 복합 시스템(System of Systems)의 다측면적인 모델링을 위해 SoSES/FB 프레임워크를 제안한다. SoSES/FB 프레임워크는 기존의 SES/MB의 개념을 확장하여, 분산 환경인 HLA/RTI에 적용 가능하도록 제안하였다. SoSES 프레임워크는 SoSES 형식론과 이를 구현한 SoSES/FB 환경으로 구성된다. SoSES 형식론은 시스템을 정확히 표현하기 위해 구성 관계, 분류 관계, 연결 관계에 대한 정보를 가지고 있는 형식론이다. SoSES는 크게 표현법과 연산으로 구성되며, 표현법은 노드와 데이터로 구성된다. SoSES/FB 환경은 SoSES 형식론을 토대로 만들어진 SoSES 트리를 관리하는 SoSES Base와 기계발된 페더레이트들을 관리하는 Federate Base로 구성된다.

3.1 SoSES 형식론

3.1.1 구성 요소

Table 1은 SoSES 형식론을 구성하고 있는 노드와 데이터의 종류를 보여준다.

Table 1. Components of SoSES formalism

	표현	정의
system 노드		A, X, Y, Z 각각 하나의 시스템 (페더레이트, 페더레이션)을 의미
inter- aspect 노드		노드 A는 X와 Y와 Z로 구성(and)됨을 의미
inter- specialization 노드		노드 B는 B1, B2 또는 B3 중 하나의 대안(or)을 가짐을 의미
data (object, interaction)	[P][S] Object X {x1, x2} [P] Interaction Z {z1, z2}	[P]는 데이터 출력 [S]는 데이터 입력

system 노드는 실제 시스템에 대응되는 노드로, 페더레이션 또는 페더레이트를 의미한다. SoSES 트리의 최상위 노드는 system 노드로써 복합 시스템, 즉 페더레이

트들로 구성된 전체 페더레이션을 의미한다. inter-aspect 노드는 하나의 system 노드가 가질 수 있는 구성 관계를 나타내는 노드이며, inter-specialization 노드는 복합 시스템의 분류 관계를 나타내는 노드이다.

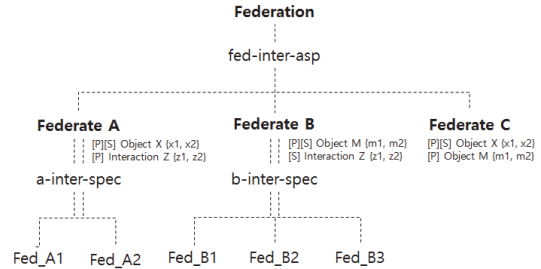


Fig. 4. Representation of SoSES formalism

SoSES 트리는 세 가지 종류의 노드 이외에도 복합 시스템에서 구성 시스템들의 연결 관계, 즉 페더레이트들이 주고받는 정보를 나타내는 데이터(data)라는 구성요소가 필요하다. 데이터에는 HLA의 SOM에 정의된 개념인 ‘object’와 ‘interaction’ 이 두 가지 정보가 포함 된다. object는 시간이 지나면 변경 될 수 있지만 변경된 정보를 유지하면서 페더레이트 사이에 공유가 필요한 경우 사용되며, ‘attribute’들을 세부 정보로 가진다. interaction은 시간이 지나면 유지할 필요가 없는 정보로, ‘parameter’들을 세부 정보로 가진다.

이러한 요소 외에도 데이터의 입출력을 구분하여 페더레이트 간의 연결 관계를 명시해줄 필요가 있다. SoSES 형식론에서는 HLA에 따라 데이터 출력을 ‘publish’라고 하여 이름 앞에 [P]로 표기하고, 데이터 입력을 ‘subscribe’라고 하여 이름 앞에 [S]로 표기한다. 입출력을 모두 필요로 하는 데이터의 경우에는 [P][S]를 모두 표기한다. 예를 들어, [P] Object X {x1, x2, x3} 라고 하면 object X는 x1, x2, x3의 attribute를 가지며 이 데이터를 publish 하겠다는 의미이다. SES에서는 메시지를 전달할 수 있도록 모델의 입출력 포트 간의 연결 관계를 명시하지만, SoSES에서는 메시지를 의미하는 interaction 외에도 object에 대한 정보가 추가적으로 필요하기 때문에 SES 형식론과는 차이점을 가진다. Figure 4는 SoSES 형식론을 이용하여 구성된 트리의 예이다.

3.1.2 SoSES 연산

SoSES 트리를 구성한 이후에, 첫 번째로 사용자에 의해 여러 대안들 중 하나의 페더레이트를 선택하는

Pruning 연산이 필요하다. Pruning은 분류 관계를 나타내는 inter-specialization 노드의 자식 노드들 중에서 하나의 system 노드를 선택하는 과정이다. 사용자에게 의해서 모든 inter-specialization 노드의 자식 노드를 선택하게 되면, 선택된 노드들은 inter-specialization 노드의 부모 노드와 치환이 되고, 선택되지 않은 노드와 inter-specialization는 삭제된다. Figure 5와 같이 a-inter-spec 노드는 Fed_A1과 Fed_A2의 자식 노드를 갖고, b-inter-spec 노드는 Fed_B1, Fed_B2, Fed_B3의 자식 노드를 갖는다. 사용자가 노드 Fed_A2와 Fed_B3를 선택하면 노드 Federate A는 Fed_A2로 치환되고, 노드 Federate B는 Fed_B3로 치환된다. 선택되지 않은 나머지 system 노드와 inter-specialization 노드들은 제거되고, 최종적으로 트리는 fed-inter-asp 노드와 Fed_A2, Fed_B3, Federate C 자식 노드들, 즉 세 가지의 페더레이트들로 구성된다. 이때 이 Pruning 과정을 거쳐 나온 트리를 Pruned Entity Structure(PES)라고 한다. Pruning 과정을 통해 복합 시스템의 구성 관계와 분류 관계에 대한 명확히 할 수 있다.

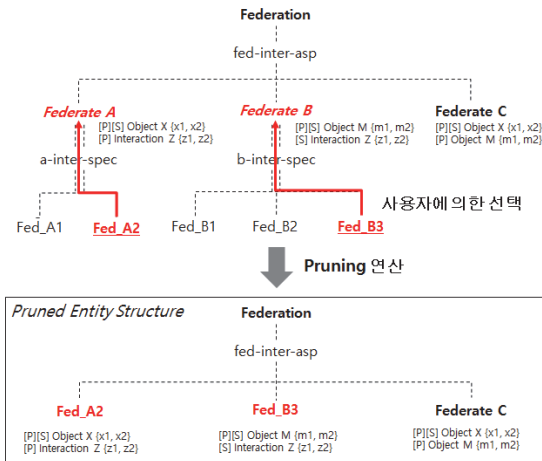


Fig. 5. Pruning operation of SoSES tree

Pruning 연산 후에, 데이터 합성 과정을 통해 복합 시스템의 연결 관계에 대한 연산이 이루어진다. 데이터 합성 과정은 Pruning 과정에서 선택된 페더레이트들의 연결 정보인 데이터(SOM)를 통합하고 합성하는 과정이다. 데이터 합성은 Figure 6과 같이 system 노드인 A1, B2, C가 가지는 데이터(object와 attribute, interaction과 parameter)들을 통합하며, 이 때 중복되는 데이터들은 하

나로 취급한다. 중복되는 데이터는 데이터의 종류(object와 interaction)와 데이터의 이름이 동일해야 한다. 각 object, interaction에 대해서 attribute, parameter의 수와 이름 역시 동일해야 한다.

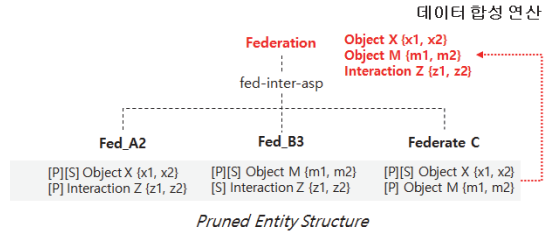


Fig. 6. Data synthesis operation of PES tree

또한 데이터 합성 시에 publish, subscribe 관계도 확인할 필요가 있다. 서로 다른 시뮬레이터들끼리 연동하기 위해서는 페더레이트 간의 데이터 교환이 필요한데, 페더레이트 사이에 교환할 데이터가 없는 경우에는 연동의 필요성이 없어진다. 즉, 어떠한 데이터를 publish하면 다른 페더레이트에서 동일한 데이터를 subscribe하는 데이터는 합성 시에 포함되지 않는다. 즉, 데이터 합성 시에 publish-subscribe 쌍을 만족하는 데이터가 존재해야 합성이 가능하다. 이 두 가지 조건을 모두 만족 시키면, 페더레이트들은 각 데이터를 통합하여 FOM을 합성한다. 만약 Pruning시에 새로운 페더레이트가 선택되면, 새로운 페더레이트가 가지고 있는 SOM을 이용하여 새로운 FOM을 구성하게 된다.

3.2 SoSES/FB 환경

3.1.절에서 제안된 SoSES 형식론을 이용하여 실제 복합 시스템의 시뮬레이션하기 위해서는 형식론을 토대로 한 SoSES/FB 환경이 필요하다. 이는 Figure 7과 같이 SoSES 트리가 저장되어 있는 SoSES Base와 실제 페더레이트들이 저장되어 있는 Federate Base로 구성된다. 또한 이들 각각을 관리하는 SoSES 관리기와 페더레이트 관리기와 페더레이션을 합성하는 합성기로 구성되어 있다. SoSES Base는 기본적으로 사용자 노드에 존재하며, Federate Base는 네트워크로 연결되어 있는 각 노드에 존재한다. 페더레이트 실행을 위한 각 노드들은 연동 미들웨어인 HLA/RTI에 의해서 연동 시뮬레이션이 가능하다. 각 요소에 대한 역할 및 세부 구성은 다음과 같다.

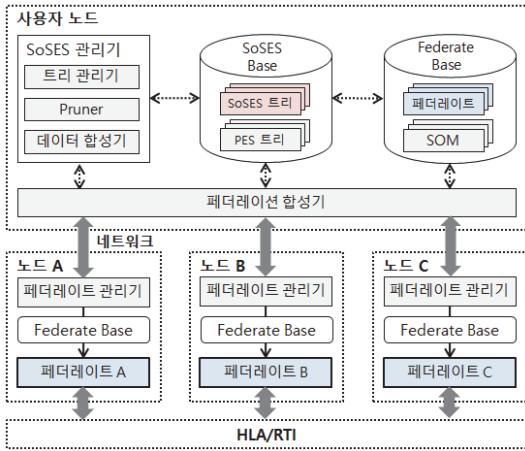


Fig. 7. SoSES/FB Environment

3.2.1 SoSES Base

SoSES Base는 SoSES 형식론에 따라 구성된 SoSES 트리를 저장하고 있는 저장소이다. 실제 사용자에게 의해서 구성된 SoSES 트리를 저장하고 불러와서 사용할 수 있으며, 이러한 SoSES 트리 이외에도 Pruning 과정 이후에 만들어진 PES 트리 또한 저장되어 있다.

3.2.2 Federate Base

Federate Base는 실제 개발된 페더레이트들이 저장되어 있는 저장소이다. SoSES 트리가 페더레이트의 정보를 가지고 있다면, Federate Base는 실행 가능한 페더레이트와 SOM 정보로 구성되어 있다. SoSES 트리에서 Pruning을 거쳐 합성이 될 때 SoSES 트리를 구성하고 있는 system 노드들은 그 이름을 참조하여 Federate Base에 있는 실제 페더레이트들을 불러와 페더레이션을 합성한다. Federate Base에는 추후에 페더레이트들을 추가하거나 수정, 삭제 할 수 있다. Federate Base는 기본적으로 각 노드에 존재하며, 사용자 노드에도 존재할 수 있어 필요시에 사용자 노드의 Federate Base로부터 페더레이트가 요청되어 각 노드로 전송될 수 있다.

3.2.3 SoSES 관리기

SoSES 관리기는 SoSES 트리를 구성하고 관리하는 역할을 한다. SoSES 관리기는 크게 트리 구조를 관리하는 트리 관리기와 Pruning 연산을 하는 Pruner, 그리고 데이터를 합성하는 데이터 합성기로 구성된다. SoSES 트리는 Federate Base에서 페더레이트 목록을 받아 구성

가능하며, 또한 기존의 도메인 전문가가 구성해 놓은 SoSES 트리를 SoSES Base에서 수정 및 사용할 수 있다. 트리가 구성된 이후엔 Pruning 알고리즘이 적용된 Pruner에서 연산을 수행하고, 데이터 합성기를 통해 FOM이 합성된다(Figure 8). 합성된 FOM은 페더레이트 관리기로 페더레이트 정보와 함께 보내지고, 페더레이트가 실행되어진다.

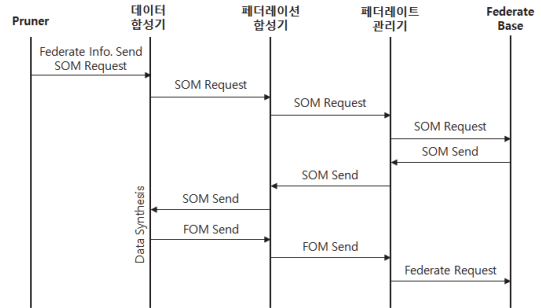


Fig. 8. Sequence of Data Synthesis

3.2.4 페더레이트 관리기

페더레이트 관리기는 Federate Base의 페더레이트들을 관리하는 역할을 한다. 페더레이트 관리기는 페더레이션 합성기로부터 Pruning과 데이터 합성 결과로 나온 페더레이트 정보와 합성된 데이터(FOM)를 받아서 Federate Base로 보내주고 각각의 페더레이트를 Federate Base에서 불러와 실행시켜주는 역할을 한다. 실행된 페더레이트들은 HLA/RTI에 의해 연동 시뮬레이션에 참여하게 된다(Figure 9).

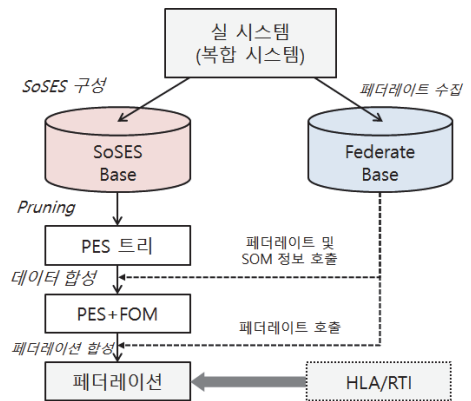


Fig. 9. Concept of federation synthesis using SoSES/FB

4. SoSES/FB를 통한 페더레이션 개발 과정

본 장에서는 IEEE 1516.3 FEDEP (IEEE Std., 2003)의 절차에 따라 SoSES/FB를 활용한 페더레이션 개발 과정을 소개한다. 이는 FEDEP과 마찬가지로 설계 단계, 구현 단계, 실행 단계의 총 3단계로 구성되며, 기개발된 기존 페더레이트들을 효율적으로 재사용하여 페더레이션을 구성할 수 있도록 도와줄 수 있다(Kim and Kim, 2012). 전체적인 개발 과정은 Figure 10과 같다.

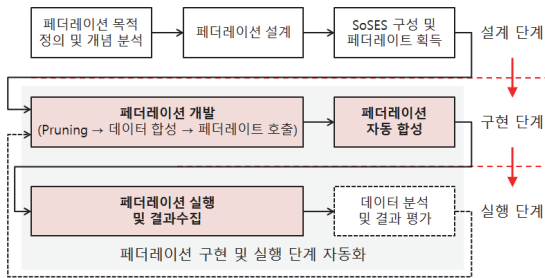


Fig. 10. Proposed federation development process

4.1 설계 단계

페더레이션 개발 과정에서는 가장 우선적으로 설계 단계가 수행이 되어야한다. 설계 단계에서는 먼저 사용자의 요구사항을 파악하고 대상 페더레이션의 목적을 정의한다. 미리 설정된 목적에 따라 페더레이션 설계 및 요구사항 명세가 이루어진다. 이 때 전체적인 페더레이션 시스템을 간단하게 설계하고 여기에 필요한 부 시스템인 페더레이트들을 살펴본다. 또한 페더레이트들끼리 주고받는 데이터 교환 인터페이스를 설계한다.

이후 설계된 페더레이션을 토대로 SoSES 트리를 구성하거나 기존 구성된 SoSES 트리를 활용 할 수 있다. 이에 따라 필요한 페더레이트 및 SOM 정보를 획득한다. SoSES 트리에는 페더레이션이 어떠한 페더레이트들로 구성되어 있으며, 어떠한 대안을 가지고, 어떠한 정보를 주고받는지에 대한 정보를 포함한다. 또한 Federate Base에 기개발된 페더레이트이 확보되어야 하며, 상황에 따라 새로운 페더레이트들을 개발하는 과정이 필요하다.

4.2 구현 단계

페더레이션에 대한 설계가 끝나면 실제 페더레이션 개발을 위한 구현 단계가 수행이 되어야 한다. 페더레이션 개발 단계에서는 구성된 SoSES에서 사용자의 목적에 의해서 페더레이트들을 선택하는 Pruning이 진행된

다. Pruning을 통해 원하는 페더레이트 정보와 그 페더레이트의 attribute인 SOM 정보를 얻을 수 있다. Pruning 과정 후에 선택된 페더레이트들의 정보를 가지고 데이터 합성 과정이 진행되며, 각 노드는 필요한 페더레이트들을 Federate Base로부터 호출하여 페더레이션이 합성된다. 사용자가 SoSES를 구성 또는 호출하고, 그로부터 원하는 페더레이트들을 선택하면, 이후 Pruning부터 페더레이션 생성까지 모든 과정은 자동으로 진행된다.

4.3 실행 단계

페더레이션이 자동 합성된 이후, 페더레이트들은 HLA/RTI에 참여하여 시뮬레이션을 하게 되며, 시뮬레이션 종료 후 결과가 수집된다. 시뮬레이션 결과가 얻어지면 데이터 분석 과정을 통해 원하는 결과의 획득 유무를 확인하고, 원하는 결과가 얻어지면 개발 과정을 마무리하고, 그렇지 않을 경우 다시 Pruning 이전으로 돌아가 과정을 다시 진행하여 새로운 페더레이션과 시뮬레이션 결과를 획득할 수 있다. 즉, 이 과정을 통해서 FEDEP의 반복적인 목적 설정 과정을 줄이고, 기존 페더레이트들의 효율적인 재사용과 정형적인 페더레이션 구조 관리를 통해서 페더레이션 개발을 효율적으로 할 수 있으며, 개발 및 실행 단계를 자동화함으로써 빠르고 편리한 개발이 가능함을 알 수 있다.

5. 사례 연구

본 장에서는 제안하는 SoSES/FB 개발 방법을 적용한 복합 시스템의 모델링 및 시뮬레이션을 수행한다. 특별히 분산 환경인 HLA/RTI에서 동작하도록 개발되어진 이산 사건 기반의(DEVS, Discrete Event System Specification) 전함의 대공방어 시뮬레이션을 이용하며(Zeigler et al., 2000), 이를 통해 제안하는 방법이 실제 페더레이션 개발 및 실험 수행에 유용함을 보여준다.



Fig. 11. Scenario of air defense simulation

5.1 시나리오

대공방어 시뮬레이션의 시나리오는 Figure 11과 같다. 먼저 아군 전함이 존재하고, 전함을 향해 미사일(타겟)이 발사된다. 이때 전함에 있는 레이더를 통해 미사일의 위치와 전함의 위치를 파악해서 전함에 전달한다. 전함은 미사일과의 거리에 따라 방어 시스템의 작동 여부를 판단하며 1차적으로 대공방어 무기를 통해 방어를 시도하며 이 때 실패했을 경우에 2차적으로 근접 무기 시스템을 통해 최종적으로 방어를 시도한다. 방어를 시도하면 방어 시스템을 통해 미사일의 격추 여부와 동시에 전함의 생존 여부가 결정되고, 이를 반복하여 수행함으로 전함의 평균 생존율을 구할 수 있다(Sung et al., 2009). 이 때 미사일을 탐지하는 레이더는 알고리즘이나 특성들에 따라서 몇 가지 모델(페더레이트)을 가질 수 있으며, 전함의 평균 생존율이 80%를 넘기는 레이더 모델을 찾는 것이 본 시뮬레이션의 목적이다. 대공방어 시뮬레이션은 Figure 12와 같이 전함, 대공방어(AAW), 근접무기(CIWS), 레이더, 타겟 페더레이트들로 구성되어 있으며 이들은 독립적인 각각의 페더레이트로 개발되어 있다.

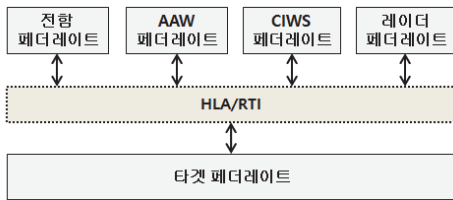


Fig. 12. Federates of air defense simulation

5.2 대공방어 페더레이션 합성 과정

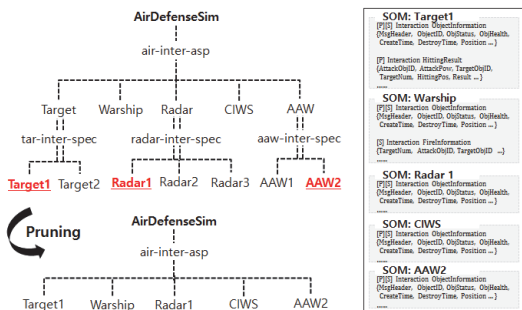


Fig. 13. SoSES tree of air defense simulation

대공방어 시뮬레이션을 구성하고 있는 각 페더레이트 들은 실제로 다양한 대안들을 가진다. 예를 들어 레이더

페더레이트는 탐지 알고리즘 및 내부 구조에 따라 다양한 종류가 존재하고, 각각의 페더레이트가 서로 대체 가능하다. 대공방어 시뮬레이션의 구성과 대안, 그리고 연결 관계는 Figure 13과 같이 SoSES 형식론으로 표현 가능하다.

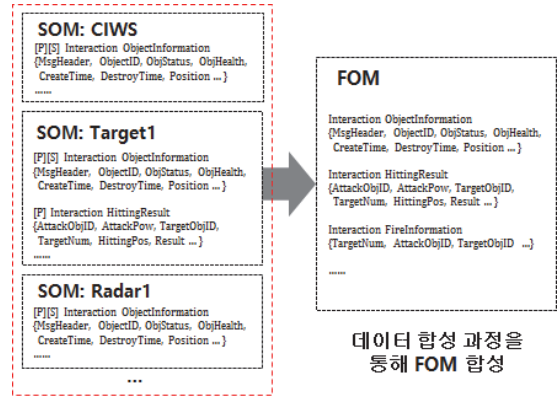


Fig. 14. FOM synthesis through data synthesis process

SoSES 형식론을 통해 복합 시스템이 구성되면, 사용자에 의해 다양한 대안들 중에 원하는 한 노드만 선택하는 Pruning 과정을 거치게 되고, 여기서는 타겟1, 센서2, 무기체계2의 페더레이트가 선택하였고, Figure 13의 아래와 같은 PES 트리를 얻을 수 있다. Pruning 이후에는 3.1.2에서 설명한 데이터 합성 과정에 따라 페더레이션에 필요한 데이터인 FOM이 합성된다(Figure 14). FOM이 구성된 이후, 합성기는 페더레이트 관리기로 페더레이트 정보와 FOM을 보내고, 동시에 페더레이트 관리기에게 각 페더레이트를 실행하라는 명령을 보내며 각 페더레이트 관리기는 Federate Base에 존재하는 페더레이트를 실행한다. 그 후 각 페더레이트가 페더레이션에 참여하고 시작함으로 결과적으로 전체 시뮬레이션이 시작된다. Figure 15는 페더레이션 합성 과정을 통한 시뮬레이션 실행 결과를 보여준다.

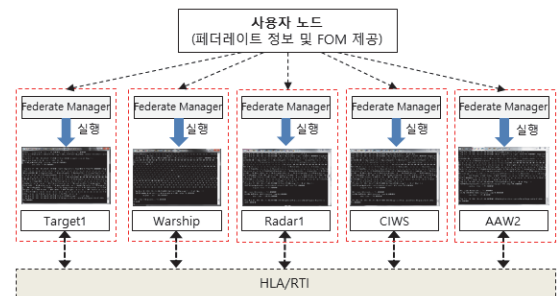


Fig. 15. Federation execution

5.2 시뮬레이션 결과

초기에 설정한 반복 횟수만큼 시뮬레이션이 수행된 후에 전함의 생존율을 확인할 수 있다. 본 실험에서는 100 회의 반복 시뮬레이션을 통해 평균 생존율을 구하였으며 Table 2는 시뮬레이션 결과를 보여준다. 본 대공방어 시뮬레이션의 목적이 평균 생존율이 80%를 넘기는 레이더 모델을 찾는 것이므로 처음 개발과정에서 Radar1 페더레이트를 사용했을 때의 평균 생존율 58%는 이를 만족하지 못한다. 그렇기 때문에 앞서 제안한 페더레이션 개발 과정에 따라 조건을 만족하지 못했기 때문에 다시 앞의 단계로 돌아가 Radar2 페더레이트를 선택하여 과정을 다시 진행한다. 이 과정을 초기 목적을 만족할 때까지 반복하여, Table 2와 같이 Radar3 페더레이트가 평균 생존율 83%를 가짐을 확인할 수 있다. 이 때 Radar1, 2, 3 페더레이트들은 탐지 알고리즘 등에 따라 여러 페더레이트들을 가질 수 있지만, 본 실험에서는 간단히 FSR(Fire Control Radar)의 개수만 다르게 하여 차이를 두었다. 이렇듯 페더레이션 개발 목적에 따라 80%를 넘는 탐지 시스템을 찾았기 때문에 전체 과정을 마치게 된다.

Table 2. Simulation results

페더레이트 이름	내부 특성	평균 생존율
Radar1	FSR 2개 사용	58%
Radar2	FSR 3개 사용	78%
Radar3	FSR 4개 사용	83%

6. 결론

본 논문에서는 HLA를 이용하여, 복합 시스템의 다측면적인 시스템 모델링 시뮬레이션을 위한 SoSES/FB 프레임워크를 제안하였다. 기존의 다측면적인 시스템 모델링 시뮬레이션 프레임워크인 SES/MB 프레임워크는 단일 시스템에 한해 모델링하고 시뮬레이션 하는 방법이였기 때문에 복합 시스템을 시뮬레이션하기에는 한계점이 존재한다. 이에 따라 복합 시스템을 위한 새로운 프레임워크가 필요하며, 본 논문은 복합 시스템의 모델링 시뮬레이션을 위한 SoSES/FB 프레임워크를 제안하였다. 이를 위해 복합 시스템의 구성과 모든 대안들을 트리 구조를 이용하여 나타낼 수 있는 SoSES 형식론을 제안하고, 실제 HLA-Compliant한 페더레이트들이 저장되어 있는 Federate Base를 제안하였다.

전함 시뮬레이션을 SoSES/FB 프레임워크에 적용함으

로서 기개발된 페더레이션이 기존의 HLA 통해 구성된 페더레이션과 동일하게 동작함을 확인 할 수 있었다. 또한 제안한 SoSES/FB 프레임워크가 가지는 장점들에 대해서 확인할 수 있었다. SoSES/FB 프레임워크는 페더레이션을 합성하는 과정을 자동화함으로써 사용자 편의성을 높이고, 페더레이션 개발 시간을 단축시킬 수 있으며, 또한 SoSES 형식론을 사용하여 페더레이트의 구조를 효율적으로 관리할 수 있으며, 그 구조를 바꾸기가 용이하여 여러 가지 대안에 대하여 시뮬레이션하기에 편리하다.

현재 제안된 SoSES 형식론은 HLA에 제한된 표현법을 가지므로 HLA를 통해 개발된 시뮬레이터에만 사용 가능하기 때문에 일반적인 분산 환경에서도 적용이 가능하도록 SoSES 형식론을 확장할 필요가 있다. 현재는 기존에 HLA-Compliant하게 개발된 페더레이트들을 사용한다. 하지만 이러한 가정은 다른 페더레이트들을 사용하는 데에 제약을 준다. 그렇기 때문에 추후에는 페더레이트에 따라서 인터페이스를 자동 생성하는 연구를 진행하고자 한다.

References

- Ahn, J.H. and T.G. Kim (2010) "Development Process of Distributed Systems based on IEEE 1516 HLA", '10 Summer Symposium of the Institute of Electronics and Information Engineers, 33(1), 1582-1585.
- (안정현, 김탁곤 (2010) "IEEE 1516 HLA 표준에 기반한 분산 시스템 개발 프로세스", 대한전자공학회 '10 하계학술대회, 33(1), 1582-1585).
- IEEE Std. 1516.3-2003 (2003) IEEE Recommended Practice for High Level Architecture (HLA) Federation Development and Execution Process (FEDEP), IEEE Computer Society.
- IEEE Std. 1516-2000 (2000) IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) - Framework and Rules, IEEE Computer Society.
- IEEE Std. 1516-2000 (2001) IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA) - Object Model Template (OMT), IEEE Computer Society.
- Kim, B.S., C.B. Choi and T.G. Kim (2013) "Multifaceted Modeling and Simulation Framework for System of Systems Using HLA/RTI," 2013 Spring Simulation

- Multiconf., San Diego, CA, USA, April.
- Kim, B.S. and T.G. Kim (2012) "Development Process of Distributed Systems using System Entity Structure Formalism", '12 Summer Symposium of the Institute of Electronics and Information Engineers, 35(1), 1997-2000.
- (김병수, 김탁곤 (2012) "System Entity Structure 형식론을 이용한 분산 시스템 개발 프로세스", 대한전자공학회 '12 하계학술대회 35(1), 1997-2000).
- Kim, J.K. and T.G. Kim (2006) "A Plan-Generation-Evaluation Framework for Design Space Exploration of Digital Systems Design", IEICE Transactions on Fundamentals of Electronics, 89(3), 772-781.
- Kim, T.G. (2011) IE 801 Lecture Note, Industrial & Systems Engineering, KAIST, <http://sim.kaist.ac.kr/>.
- Kim, T.G. and B.P. Zeigler (1989) "A Knowledge-Based Environment for Investigating Multicomputer Architectures", Journal of Information and Software Technology, 31(10), 512-520.
- Kim, T.G., C. Lee, E.R. Christensen and B.P. Zeigler (1990) "System Entity Structuring and Model Base Management", IEEE Transactions on Systems, Man and Cybernetics, 20(5), 1013-1024.
- Lee, W.B. (1994) "Development of The Multifaceted System Modelling / Simulation Environment", Master Thesis, Korea Advanced Institute of Science and Technology, Daejeon, Republic of Korea.
- Park H.C., W.B. Lee and T.G. Kim (1997) "RASES: A Database Supported Framework for Structured Model Base Management" Simulation Practice and Theory, 5(4), 289-313.
- Sung C.H., J.H. Hong and T.G. Kim (2009) "Interoperation of DEVS Models and Differential Equation Models using HLA/RTI: Hybrid Simulation of Engineering and Engagement Level Models" 2009 Spring Simulation MultiConf., San Diego, CA, USA, March.
- Zeigler, B.P. (1984) Multifaceted Modelling and Discrete Event Simulation, ACADEMIC PRESS.
- Zeigler, B.P., H. Praehofer and T.G. Kim (2000) Theory of Modeling and Simulation, ACADEMIC PRESS.



김 병 수 (kevinzzang@kaist.ac.kr)

2010 KAIST 전기 및 전자공학과 학사
 2012 KAIST 전기 및 전자공학과 석사
 2012~ 현재 KAIST 전기 및 전자공학과 박사과정

관심분야 : 모델링 및 시뮬레이션, 시뮬레이터 연동, 빅 데이터, 데이터 모델링



김 탁 곤 (tkim@kaist.ac.kr)

1975 부산대학교 전자공학과 학사
 1980 경북대학교 전자공학과 석사
 1988 Univ. of Arizona, 전기및컴퓨터공학과 박사
 1980~1983 부경대학교 통신공학과 전임강사
 1987~1989 (미)애리조나 환경연구소 연구엔지니어
 1989~1991 Univ. of Kansas 전기및컴퓨터공학과 조교수
 1991~현재 KAIST 전기 및 전자공학과 교수

- 한국시뮬레이션 학회 회장 역임
- 국제시뮬레이션학회(SCS) 논문지(Simulation) Editor-In-Chief 역임
- SCS Fellow
- 모델링 시뮬레이션 기술사(미국)
- *Who's Who in the World* (Marguis 16thEdition, 1999) 등재
- 연합사, 국방부/합참, 기품원 자문위원 역임
- KIDA Fellow 역임
- ADD 자문위원(현)

관심분야 : 모델링/시뮬레이션 이론, 방법론 및 환경개발, 시뮬레이터 연동