

하이브리드 시스템 모델링 및 시뮬레이션 - 제1부: 모델링 및 시뮬레이션 방법론*

Hybrid Systems Modeling and Simulation - Part I: Modeling and Simulation Methodology

임성용**, 김탁곤***

Seong Yong Lim and Tag Gon Kim

Abstract

A hybrid system is defined as a mixture of continuous systems and discrete event systems. This paper first proposes a framework for hybrid systems modeling, called Hybrid Discrete Event System Specification (HDEVS) formalism. It then presents a method for simulators interoperation in which a continuous system simulator and a discrete event simulator are executed together in a cooperative manner. The formalism can specify a hybrid system in a way that a continuous system and a discrete event system are separately modeled by their own specification formalisms with a support of well-defined interface. We call such interface an A/E converter for analog-to-event conversion and an E/A converter for event-to-analog conversion. Simulators interoperation is based on the concept of pre-simulation in which simulation time for a continuous simulator is advanced in accordance with a discrete event simulator.

Key Words: Hybrid System, Simulators Interoperation, HDEVS Formalism, A/E and E/A Converter

* 본 논문은 2000년 삼성 휴먼테크 논문 경시대회에서 발표한(금상 수상) 내용을 보완한 것 중 후반부(1부)임.

** 한국전자통신연구원 네트워크기술연구소

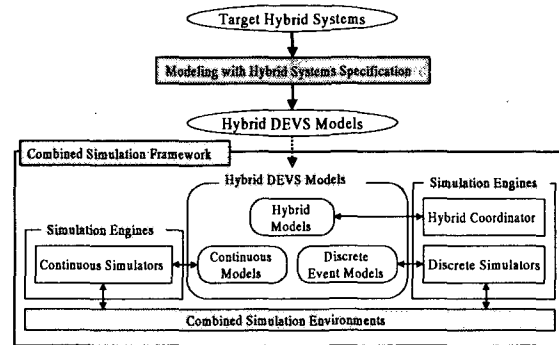
*** 한국과학기술원 전자전산학과

1. 서론

시스템 모델링 및 시뮬레이션 방법은 대상 시스템의 종류 및 분석 목적에 따라 달라져야 한다. 예를 들면, 연속적인 시간변수를 가지는 시스템에 대해서는 미분방정식 등을 사용한 자세한 표현력을 바탕으로 한 정량적인 분석을 요하게 되며, 이산적인 시간 변수를 가지는 시스템에 대해서는 유한상태 오토마타 등을 사용한 추상적인 표현력에 기반한 시스템의 정성적인 행동에 관심이 있다. 반면, 두 가지 이상의 시스템이 혼재된 - 예를 들면, 연속시스템과 이산사건 시스템 - 하이브리드 시스템에 대해서는 각각의 시스템이 가지는 요소를 유지하면서 전체 시스템을 모델링하고 시뮬레이션 할 수 있는 방법이 필요하다.

제어 시스템 분야에서는 하이브리드 시스템의 동특성을 해석적 모델로 표현하고, 이것으로부터 시스템의 특성을 분석하는데 중점을 두었다. [1]에서는 인터페이스에 기반한 하이브리드 시스템을 기술하는 형식을 제안하였고, 가제어성, 가관측성과 유연한 디자인에 관한 분석이 이루어졌다. 그러나, 제어 시스템 영역에서의 하이브리드 시스템 분석은 해석적 방법에 의존하였고, 시뮬레이션 모델이나 환경에 대해서는 관심이 없었다. 반면, 시뮬레이션 분야에서는 이산 사건 시스템과 연속 시스템을 하나의 시스템으로 혼합하여 새로운 모델로 바라보는 혼합형 형식론을 바탕으로 시뮬레이션 환경에 이르는 방법론을 제안하였다[2]. 혼합형 형식론에 바탕을 둔 시뮬레이션 방법론은 이미 개발된 모델과 기존의 시뮬레이션 도구를 사용할 수 없고, 새로운 모델과 새로운 도구들을 사용하여야 하는 제한이 있었다. 동일한 개념으로 [3]에서는 Timed Automata 형식론의 확장형태인 Hybrid Automata 형식론을 바탕으로 하이브리드 시스템을 모델링하고 분석하였지만, 혼합형 형식론이 가지는 이러한 제한점은 극복하지 못했다.

본 연구에서는 이러한 문제점을 해결하기



<그림 1> Hybrid Systems Modeling and Simulation Framework

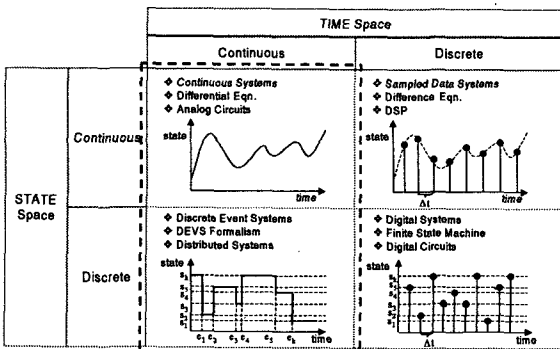
위하여 <그림 1>에 나타난 하이브리드 시스템 모델링 시뮬레이션 방법론을 제안한다. 본 논문에서 다루는 하이브리드 시스템은 연속시스템과 이산사건 시스템이 혼재된 것으로 국한한다. 제안된 방법의 핵심은 기존의 연속시스템 모델과 이산사건 시스템 모델을 재사용하고 이들 모델을 시뮬레이션 하는 시뮬레이터를 연동시켜 전체 하이브리드 시스템을 시뮬레이션 한다는데 있다. 기존 모델을 재사용하는 모델링 틀로 Hybrid Discrete Event Systems Specification(HDEVS)가 제안되었고, 이들 모델들을 연동하기 위하여 새로운 인터페이스 개념인 A/E(Analog-to-Event) 변환기와 E/A(Event-to-Analog) 변환기가 고안되었다. 시뮬레이터를 연동하기 위한 방법으로 pre-simulation 개념이 도입되었다.

본 논문의 구성은 아래와 같다. 2장에서는 시스템의 일반적인 정의와 하이브리드 시스템을 정의하고, 적용 범위를 제시한다. 3장에서는 정의된 하이브리드 시스템을 모델링 하는 형식론을 제안하고 4장에서는 시간 간격이 다른 이종의 시스템을 연동 시뮬레이션 하기 위한 개념과, 제안한 형식론에 기반한 시뮬레이터의 알고리즘을 설명한다. 5장에서는 목표추적 예제를 통해서 제안하는 방법론의 적용 예를 확인하고, 마지막으로 6장에서는 결론을 맺고 추후 과제를 제시한다.

2. 하이브리드 시스템

2.1 시스템의 분류

일반적으로 수학적 시스템은 상태변화의 원인이 되는 입력과 상태변화의 결과로 야기되는 출력 및 시스템의 상태를 기록하는 상태 변수로 구성된다. 입력과 출력의 연관성에 의해서 동적 시스템(Dynamic Systems)은 현재의 입력과 더불어 과거의 입력에 의하여 결정되어지는 특성을 보인다. 시스템 내부에 상태 변수의 도입은 과거의 입력 순서와 동등한 정보를 기록할 수 있으며, 이로 인하여 시스템의 출력은 시스템 내부의 상태 변수와 현재의 입력으로 결정된다. 시스템은 시스템을 이루는 입력, 출력, 그리고 상태변수의 종류에 따라서 <그림 2>[8]와 같이 분류할 수 있다.



<그림 2> System Taxonomy

연속 시스템 : 시간 변수가 연속이고, 상태 변수가 연속인 영역을 지칭하며 미분방정식으로 모델링 할 수 있다. 그 예로는 아날로그 회로가 있다(<그림 2> 좌측상단).

샘플 데이터 시스템 : 시간 변수가 이산적이고, 상태 변수가 연속인 영역을 지칭하며, 차분방정식으로 모델링 할 수 있으며, 이산적인 누적기법으로 시뮬레이션 할 수 있다. 그 예로는 디지털 신호 처리기가 있다(<그림 2> 우측상단).

디지털 시스템 : 시간 변수가 이산적이고, 상태 변수도 이산적인 영역을 지칭하며 FSM(Finite

State Machine)으로 모델링 할 수 있다. 그 예로는 디지털 회로가 있다(<그림 2> 우측하단).
이산 사건 시스템 : 시간 변수가 연속이고, 상태 변수는 이산적인 영역을 지칭하며, DEVS (Discrete Event Systems Specification)형식론으로 모델링 할 수 있다. 그 예로는 분산 시스템이 있다(<그림 2> 좌측하단).

일반적으로 하이브리드 시스템은 두 가지 이상의 개념이 혼재되어 있는 시스템을 가리킨다. 본 논문에서는 앞서 정의한 대로 연속 시스템과 이산 사건 시스템이 혼재되어 있는 경우의 하이브리드 시스템을 대상으로 한다.

2.2 하이브리드 시스템의 적용 범위

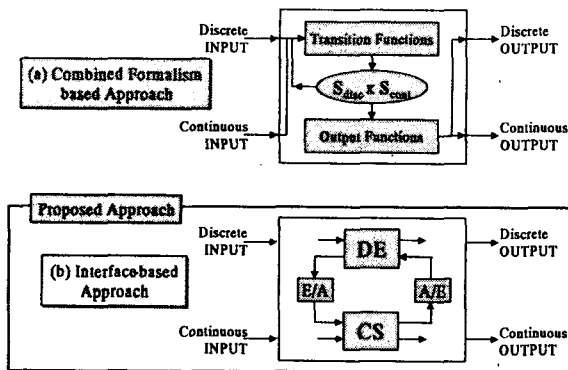
하이브리드 시스템 연구의 필요성은 여러 분야에서 발견할 수 있는데 가장 많은 연구가 있었던 분야는 제어 분야이다. [4]에서는 'Free Floating Robotic System'에서 로봇의 행동은 연속 시스템의 요소로 보고, 상호 배타적인 로봇 팔의 움직임은 이산 사건 시스템의 요소로 간주하였다. [5]에서는 'Intelligent Vehicle Highway Systems'에서 자동차의 행동은 연속 시스템으로 보고, 관제를 담당하는 제어기는 이산 사건 시스템으로 보았다. 제어 분야에서 축적된 하이브리드 시스템의 기술 표현력은 시뮬레이션 분야에 적용되어 [6]에서는 'Manufacturing Process Plan'에서 미시적인 Process 는 연속 시스템의 요소로 보고, 거시적인 상태 전환은 이산 사건의 요소로 보았다.

3. 하이브리드 시스템 기술 형식론: HDEVS

3.1 HDEVS 형식론의 개요

하이브리드 시스템을 모델링하는 방법은 시스템 내부에 존재하는 연속 시스템 컴포넌트와 이산 사건 시스템 컴포넌트를 모델링하는 방법

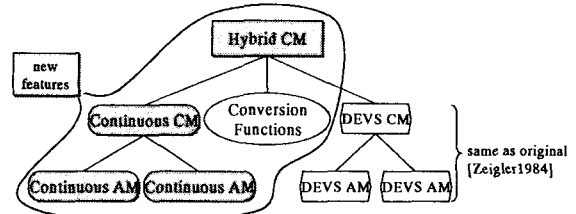
에 따라 <그림 3>과 같이 분류할 수 있다. <그림 3>(a)의 혼합 형식론 접근 방법은 이산 사건 요소와 연속 요소가 혼재되어 있는 새로운 시스템 기술 형식론을 기반으로 한다. 시스템의 상태 변수는 이산 사건 상태 변수와 연속 상태 변수의 곱집합으로 구성되며 이산 사건 입력과 연속 입력에 의해서 상태 천이가 발생할 수 있다. 시스템의 출력은 상태변수와 입력을 바탕으로 연속 출력과 이산 사건 출력을 함께 낼 수 있다. <그림 3>(b)의 인터페이스 접근 방법은 기존의 형식론을 유지하고자, 이산 사건 시스템과 연속 시스템을 각각 고유의 형식론으로 기술하고 두 시스템 사이의 입출력 교환을 위하여 외부적인 인터페이스를 기술하는 방법을 사용하고 있다. 인터페이스 접근 방법은 기존에 기술된 모델을 재사용 할 수 있는 장점과 더불어 각각의 모델을 시뮬레이션하기 위한 도구 역시 재사용 할 수 있다는 장점이 있다. 따라서, 본 논문에서는 인터페이스 접근 방법에 의거한 하이브리드 시스템 모델링 형식론을 제안하고자 한다.



<그림 3> Approaches for Hybrid Systems Specification

집합론에 근거한 DEVS형식론은 이산 사건 시스템을 모듈별로 나누고 이를 계층적인 연결로 모델링 할 수 있는 수학적 기반을 제공한다[2]. 본 논문에서 제안하는 Hybrid Discrete Event Systems Specification (HDEVS) 형식론

은 <그림 4>에서와 같이 이산 사건 시스템 컴포넌트들은 기존의 DEVS 형식론에 기반하여 기술한다. 반면, 연속 시스템 컴포넌트들은 미분방정식에 바탕을 둔 모델들을 집합론으로 표현하여 Continuous Atomic Model(CAM)과 Continuous Coupled Model(CCM)로 정의한다. 또한, 연속 시스템과 이산 사건 시스템이 연결되는 인터페이스를 위하여 Hybrid Coupled Model(HCM)과 Conversion Functions(CF)를 추가로 정의하였다.



<그림 4> Structure of HDEVS

3.2 Atomic 모델

Atomic 모델은 가장 기본적인 모듈로서 시스템의 동적특성을 기술하는 모델이다. 하이브리드 시스템의 동특성을 기술하기 위해서 이산 사건 Atomic 모델과 연속 Atomic 모델을 정의한다. 먼저, 이산 사건 Atomic 모델은 DEVS 형식론에 기반하여 이산 사건 입력 집합, 이산 사건 출력 집합, 이산 사건 상태 집합, 모델의 다음상태를 기술하는 전이 함수들과 출력을 내기 위한 출력함수 및 시간 진행 함수로 구성되며, 수학적 표현은 다음과 같다.

Def. 1 이산 사건 Atomic 모델 (Discrete Event Atomic Model: DEVS-AM)

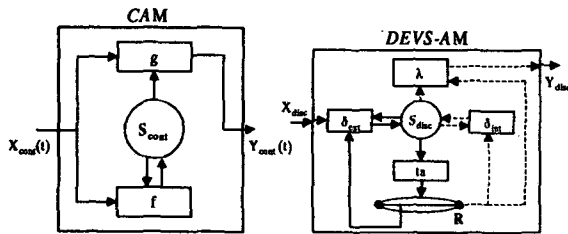
$$DEVS-AM = \langle X_{disc}, S_{disc}, Y_{disc}, \delta_{int}, \delta_{ext}, \lambda, ta \rangle$$

- X_{disc} : 이산 사건 입력 집합
- S_{disc} : 일련의 이산 사건 상태의 집합

- Y_{disc} : 이산 사건 출력 집합
- $\delta_{int}: Q \rightarrow S_{disc}$: 내부 상태 전이 함수
- $\delta_{ext}: Q \times X_{disc} \rightarrow S_{disc}$: 외부 상태 전이 함수
 $Q = \{(s, e) \mid s \in S_{disc}, 0 \leq e \leq ta(s)\}$: 전체상태
- $\lambda: Q \rightarrow Y_{disc}$: 이산 사건 출력 함수
- $ta: S \rightarrow R_{0, \infty}^+$: 시간 진행 함수

- S_{cont} : 연속 상태의 집합
- Y_{cont} : 연속 출력 집합
- $f: \frac{d}{dt} S_{cont}(t) = f(S_{cont}(t), X_{cont}(t), t)$: 연속 상태 전이 함수
- $g: Y_{cont}(t) = g(S_{cont}(t), X_{cont}(t), t)$: 연속 출력 함수

<그림 5>의 우측에서와 같이 이산 사건 Atomic 모델은 두 개의 상태 전이 함수를 가지며, 외부 상태 전이 함수 δ_{ext} 는 어떤 상태에서 외부로부터 이산 사건 입력을 받아서 상태가 바뀌는 것을 표현하는 함수이다. 출력함수는 어떤 상태에서 일정한 시간이 경과한 후 생기는 내부 상태 전이가 일어날 때 발생하는 출력 사건을 정의한다. 이 때, 발생한 출력은 다른 모델들의 입력이 되므로 한 Atomic 모델의 내부 상태 전이는 다른 Atomic 모델들의 외부 상태 전이를 야기하게 된다.



<그림 5> CAM and DEVS-AM

미분방정식을 기본적인 개념으로 하는 연속 시스템을 기술하는 형식론은 <그림 5>의 좌측에서와 같이 연속 입력, 연속 출력과 함께 연속 상태 변수의 전이함수와 출력함수를 포함한다. 위의 요소를 가지는 연속 Atomic 모델의 수학적 표현은 다음과 같다.

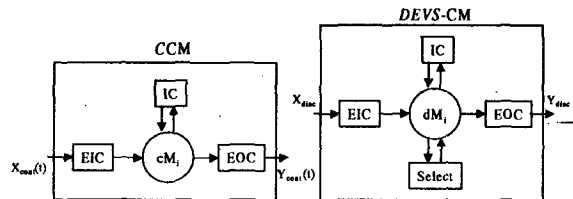
Def. 2 연속 Atomic 모델(Continuous Atomic Model: CAM)

$$CAM = \langle X_{cont}, S_{cont}, Y_{cont}, f, g \rangle$$

- X_{cont} : 연속 입력 집합

3.3 Coupled 모델

Coupled 모델은 여러 개의 컴포넌트 모델로 구성되며 이들 컴포넌트 모델들은 Atomic 혹은 Coupled 모델로 구성된다. 따라서, Coupled 모델은 이러한 컴포넌트 모델들과 외부 입력, 외부 출력 및 내부 컴포넌트 모델들 사이의 연결 정보로 표현된다. 하이브리드 시스템의 Coupled 모델에는 컴포넌트 모델의 성격에 따라서 이산 사건 모델만을 포함하는 이산 사건 Coupled 모델과 연속모델만을 포함하는 연속 Coupled 모델로 구성된다. <그림 6>에서와 같이 일반적으로 Coupled 모델은 외부 입력과 내부 모델의 입력을 연결하는 외부 입력 연결 관계(EIC)와 내부 모델의 출력과 외부 출력을 연결하는 외부 출력 연결 관계(EOC), 그리고 내부 모델의 출력과 내부 모델의 입력을 연결하는 내부 연결 관계(IC)를 공통으로 포함하게 된다. 이산 사건 Coupled 모델의 경우에는 동일한 시각에 발생하는 사건을 가지는 모델들에 대한 선택함수를 포함하지만, 연속 Coupled 모델은 시스템의 특성에 따라서 동시에 일어나는 변화에 대해서 동시에 처리해 주어야 하기 때문에 선택함수는 제외된다. 이산 사건 Coupled 모델과 연속 Coupled 모델의 수학적 표현은 다음과 같다.



<그림 6> CCM and DEVS-CM

Def. 3 이산 사건 Coupled 모델 (Discrete Event Coupled Model: DEVS-CM)

$$DEVS-CM = \langle X_{disc}, Y_{disc}, \{DM_d\}, EIC, EOC, IC, Select \rangle$$

- X_{disc} : 이산 사건 입력 집합
- Y_{disc} : 이산 사건 출력 집합
- $\{DM_d\}$: 이산사건 컴포넌트 모델들의 집합
- $EIC \subseteq X_{disc} \times \bigcup X_i$: 외부 입력 coupling 관계
- $EOC \subseteq \bigcup Y_i \times Y_{disc}$: 외부 출력 coupling 관계
- $IC \subseteq \bigcup Y_i \times \bigcup X_j$: 내부 coupling 관계
- $Select : 2^{\{DM_d\}} - \emptyset \rightarrow DM_d$: 동시에 출력을 발생하는 다수의 모델들에서 한 개의 모델을 선택하는 함수

Def. 4 연속 Coupled 모델 (Continuous Coupled Model: CCM)

$$CCM = \langle X_{cont}, Y_{cont}, \{CM_d\}, EIC, EOC, IC \rangle$$

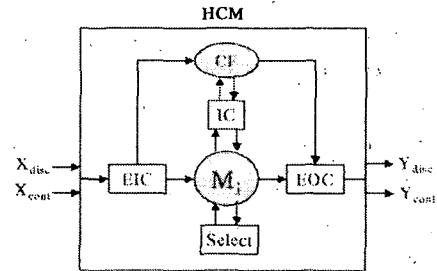
- X_{cont} : 연속 입력 집합
- Y_{cont} : 연속 출력 집합
- $\{CM_d\}$: 연속 컴포넌트 모델들의 집합
- $EIC \subseteq X_{cont} \times \bigcup X_i$: 외부 입력 coupling 관계
- $EOC \subseteq \bigcup Y_i \times Y_{cont}$: 외부 출력 coupling 관계
- $IC \subseteq \bigcup Y_i \times \bigcup X_j$: 내부 coupling 관계

3.4 하이브리드 시스템 모델

HDEVs는 이산 사건 시스템과 연속 시스템을 각각 기술하는 모델 요소와 함께 이종의 시스템 사이의 상호 운용을 위해서 두 가지의 하이브리드 모델 요소를 포함하고 있다. 먼저, 내부에 이산 사건 시스템 영역과 연속 시스템 컴포넌트들이 혼재되어 있는 하이브리드 시스템을 기술하는 Hybrid Coupled Model(HCM)과 서로 다른 종류의 두 시스템 사이의 입출력 교환을 위한 Conversion Functions(CF)로 구성된다.

HCM은 Coupled Model의 한 종류로 컴포넌

트 모델로는 CCM, CAM, DEVS-CM, DEVS-AM, 그리고 또 다른 HCM 등이 있다. 이러한 내부 모델들을 연결하기 위한 연결 정보를 EIC, EOC, IC등에 담고 있으며, 연속적인 변화를 가지는 연속 시스템의 입출력 정보와, 이산적인 변화를 가지는 이산 사건 시스템의 입출력 정보 사이의 상호 교환을 위한 함수를 CF에 정의되어 있다. <그림 7>과 같은 일반적인 형태를 가지는 HCM의 수학적 표현은 다음과 같다.



<그림 7> Hybrid Coupled Model

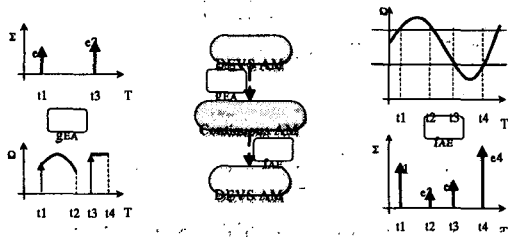
Def. 5 Hybrid Coupled 모델 (Hybrid Coupled Model: HCM)

$$HCM = \langle X, Y, \{M_d\}, EIC, EOC, IC, Select, CF \rangle$$

- $X = X_{cont} \cup X_{disc}$: 이산 사건, 연속 입력 집합
- $Y = Y_{cont} \cup Y_{disc}$: 이산 사건, 연속 출력 집합
- $\{M_d\} \subseteq \{CM_d\} \cup \{DM_d\} \cup \{HCM_d\}$: 내부 모델들의 집합
- $EIC \subseteq X \times \bigcup X_i$: 외부 입력 coupling 관계
- $EOC \subseteq \bigcup Y_i \times Y$: 외부 출력 coupling 관계
- $IC \subseteq \bigcup Y_i \times \bigcup X_j$: 내부 coupling 관계
- $Select : 2^{\{DM_d\}} - \emptyset \rightarrow DM_d$: 동시에 출력을 발생하는 다수의 모델들에서 한 개의 모델을 선택하는 함수
- $CF = \langle f_{AB}, g_{BA} \rangle$: Conversion Functions

하이브리드 Coupled 모델의 정의에 속해 있는 Conversion Functions 은 이산 사건 모델들과 연속 모델들 사이의 입출력 정보의 교환을

위한 함수이다. 이산 사건 모델의 출력은 사건의 형태로 발생되는데 연속 모델의 입력으로는 구간 연속 입력이 들어가게 되므로 사건을 구간 연속 입력으로 변환시켜 주는 함수가 필요하다. 같은 방법으로 연속 모델의 출력은 구간 연속 출력의 형태로 발생되며 이는 이산 모델의 입력인 사건의 형태로 변환되는 함수가 필요하게 된다. 전자의 경우, <그림 8>의 좌측에서 사건에 대한 구간 연속 출력의 연결은 쉽게 정의될 수 있지만, 후자의 경우 일반적인 구간 연속 출력에 대한 사건의 대응은 비슷한 구간 연속 출력간의 구분과 많은 사건의 대응을 필요로 한다는 복잡성을 내재하고 있다. 이를 해결하기 위해서 조건을 만족하는 구간 연속 출력에 각각 대응하는 사건을 정의하는 방법을 사용한다. <그림 8>의 좌측에서와 같이 어떤 구간 연속 출력이 출력 영역의 변화를 야기하게 되면 출력 영역 변화에 대응되는 사건이 정의되고 따라서 어떤 구간 연속 출력에 다대일 대응 하는 사건을 정의할 수 있다. 이렇게 정의되는 Conversion Functions의 수학적 표현은 다음과 같다[1].



<그림 8> A/E and E/A Conversions

Def. 6 Conversion Functions

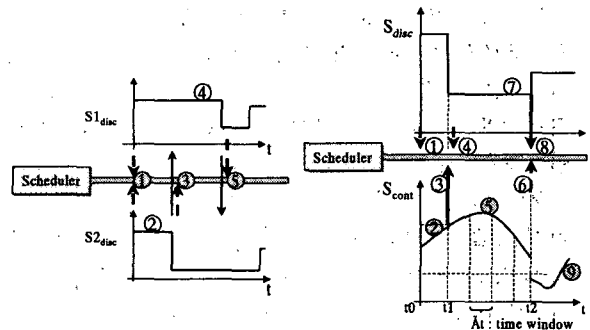
$$CF = \langle f_{AE}, g_{EA} \rangle$$

- $g_{EA} : \Sigma \rightarrow \Omega$ (Event-to-Analog Converter)
 - Σ : 사건 집합
 - Ω : 허용되는 구간 연속 출력 집합
- $f_{AE} : \Omega \rightarrow 2^{\Sigma}$ (Analog-to-Event Converter)
 - Ω : 허용되는 구간 연속 출력 집합
 - 2^{Σ} : 모든 가능한 사건 조합의 집합

4. 연동 시뮬레이션 개념 및 알고리즘

4.1 Pre-simulation: 연속 모델의 다음 사건 시간 결정

두 개 이상의 시뮬레이터가 연동되어 동작하기 위해서는 다음 두 가지 조건을 만족해야 한다. 첫째 조건은 이들 시뮬레이터 사이의 시뮬레이션 시각 진행이 동기화 되어야 하고, 둘째 조건은 이들 시뮬레이터 사이에 데이터 교환이 가능해야 한다. 첫째 조건을 만족하기 위해서는 연속 시뮬레이터의 시뮬레이션 진행 구간의 시작 시각과 끝마치는 시각이 이산사건 시뮬레이터의 사건 발생 시각과 동기 되어야 한다. 둘째 조건을 만족하기 위하여 이산사건 데이터와 연속 데이터의 데이터 변환이 필요하며 이를 위하여 앞 절에서 A/E 변환기와 E/A 변환기를 제안하였다.



<그림 9> Advance of Next Event Time
Left: Discrete Event Simulator
Right: Hybrid Simulator

이산사건 모델을 시뮬레이션하는 DEVS 시뮬레이션 알고리즘에서는 먼저, 모든 모델은 다음 사건 시각(next event time)을 예약(scheduling)한다. 다음으로, 전체 모델에서 다음 사건 시각이 최소인 모델을 선택하여 시뮬레이션 한 후 이 모델의 다음 사건 시각을 다시 예약한다. 처음과 동일한 방법으로 모든 모델 중 다음 사건 시각이 최소인 모델을 골라 시뮬레이션하고 이 과정을 반복한다[2]. <그림 9>의 좌측에서는 이러한 스

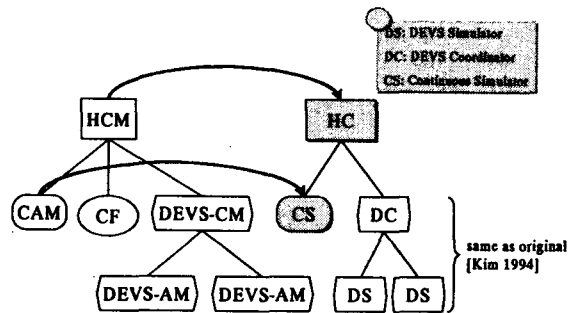
케줄링 방법을 두 개의 모델인 경우 예를 들어 설명하고 있다. 스케줄러는 두 모델이 수행되어야 하는 사건의 다음 사건 시각을 읽은 후에 ① 다음 사건 시간이 적은(빠른) 모델을 골라 시뮬레이션 수행을 명령하고 ② 수행 결과에 의해 발생한 다음 사건 시각을 읽어 들인다 ③. 위 순서를 반복하면 순차적인 시간을 갖는 시뮬레이션을 진행 할 수 있다.

한편, <그림 9>의 우측에서와 같이 연속 모델이 포함된 혼합 시뮬레이션의 시간 진행을 수행하기 위해서는 앞에서 설명한 이산 사건 모델의 시간 진행에 기반 하면서도, 연속 모델의 시간 진행을 먼저 수행하여 어떤 전방 시간 이전에는 연속 모델에서 사건의 발생이 존재 않음을 보증하는 처리가 필요하다. 스케줄러는 이산 사건 모델들로부터 다음 사건 시각을 읽고 최소의 다음 사건 시각을 계산한다 ①. 최소의 다음 사건 시간 이내에 연속 모델들로부터 어떤 사건이 발생하는지 미리 알기 위해서 연속 모델들을 최소의 다음 사건 시간까지 정해진 시간 간격 단위로 먼저 진행시킨다 ②, 그 결과에 의한 입출력 영향을 처리하고 최종 결과인 다음 사건 시각을 스케줄러는 돌려 받는다 ③④. 위의 과정을 반복함으로써 연속 모델과 이산모델의 시뮬레이션 시각이 동기화 되어 순차적으로 진행될 수 있다. 연속 모델에서 이러한 시뮬레이션 진행을 Pre-Simulation 이라 정의하고 이를 바탕으로 HDEVS 형식론에 근거한 모델들을 연동 시뮬레이션 할 수 있는 알고리즘을 제안한다.

4.2 연동 시뮬레이터 알고리즘

연동 시뮬레이션에서 필수적인 것은 시뮬레이터간의 시뮬레이션 진행 시각이 동기화 되어야 한다. 이를 위해 본 연구에서는 기존의 DEVS 모델을 시뮬레이션 하는 추상 시뮬레이터 알고리즘[7]을 연속 시스템 시뮬레이션이 가능하도록 확장하였다. 추상 시뮬레이터 개념은 시스템을 기술한 모델과 시뮬레이션을 수행하는 추상 시뮬레이터를 모듈화 하여 독립적으로

기술한다. <그림 10>에서 이산 사건 모델들의 수행 알고리즘을 내재하는 시뮬레이터는 Atomic DEVS 모델을 위한 DEVS Simulator (DS)와 Coupled DEVS 모델을 위한 DEVS Coordinator(DC)로 구성되어 있다. 한편, 연속 Coupled 모델은 [2]에서 증명되어 있듯이 또 다른 하나의 연속 Atomic 모델로 대체될 수 있다. 대체된 하나의 연속 Atomic 모델을 시뮬레이션 하기 위하여 Continuous Simulator(CS)가 사용되며, 이들은 상위 시뮬레이터로부터 메시지를 받아서 시뮬레이션을 수행 한 후 그 결과를 상위 시뮬레이터에게 다시 보낸다. 위와 같이, 이산 사건 모델과 연속 모델들의 개별적인 시뮬레이터들과 더불어, 이종의 모델들 간의 연결 정보를 포함하는 하이브리드 Coupled 모델에 필요한 시뮬레이션 알고리즘을 구현하는 Hybrid Coordinator(HC)가 필요하다. <그림 11>은 통합형 혼합 시뮬레이션 환경을 구성하는 시뮬레이터들의 기능을 설명하고 있다.



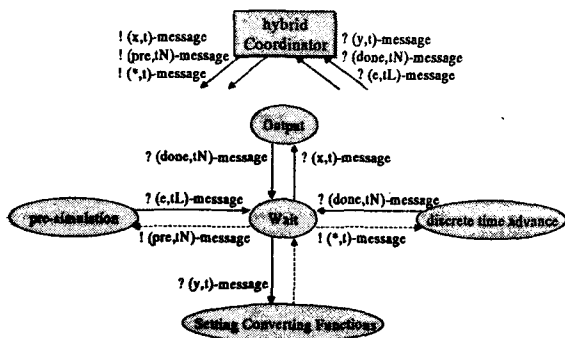
<그림 10> interoperable Simulation Framework

Name	Type	Meaning
Coordinators	Hybrid	Scheduling of continuous models and discrete models by pre-simulation and obtaining next event time
	DEVS	Input/Output messages and time advance messages routing to its child or its parent
Simulators	Continuous	Pre-simulation of continuous models until timeN using Differential Equation Solver Engines
	DEVS	Time advance until timeN and output messages routing to its parent

<그림 11> interoperable Simulation Engines

4.3 시뮬레이터 알고리즘

본 절에서는 <그림 10>에서 보인 시뮬레이터들 중연동 시뮬레이션을 위해 추가된 새로운 시뮬레이터들-Hybrid Coordinator, Continuous Simulator-의 시뮬레이션 알고리즘에 대해서 설명한다.



<그림 12> State Diagram of HC Algorithm

하이브리드 Coupled 모델은 내부 Component 모델로 이산 사건 모델과 연속 모델, 그리고 하이브리드 Coupled 모델을 가질 수 있다. 포함되는 모델들을 시스템 분류에 따라 나누어 보면 연속 모델 성분과 이산 사건 모델로 구분할 수 있다. 하이브리드 Coupled 모델의 시뮬레이션 알고리즘은 Hybrid Coordinator가 수행한다. HC는 가장 상위에 존재하는 시뮬레이터로서 하위 시뮬레이터로 연속 모델 시뮬레이터인 CS와 이산 사건 모델 시뮬레이터인 DC를 가질 수 있다. HC는 연동 업무 전체를 담당하고 있기 때문에 시뮬레이션 진행 시각 동기화와 데이터 교환을 책임져야 한다. <그림 12>에서와 같이 Wait상태의 Hybrid Coordinator는 먼저 연속 모델에게 tN 까지 시뮬레이션을 일단 진행하라는 pre-simulation 명령 메시지 (pre, tN) 를 보낸다. 만약 연속 모델의 시뮬레이션 출력을 A/E 변환한 결과, 사건 e 가 존재하면 (e,tL) 을 이산사건 모델로 전달하고, 그렇지 않은 경우에는 이산 사건 모델에 스케줄 시간이 종료되었음을 알리는 $(*,t)$ 메시지를 보내게 된다. <그림

13>은 위의 알고리즘을 Pseudo Code 형태로 서술하였다.

```

Hybrid Coordinator
; tc -> current simulation time
; tN -> next event time

Main loop
if tc = tN
    send (*, tN) to its discrete child
else if tc < tN
    send (pre, tN) to its continuous child
else "error"
End loop

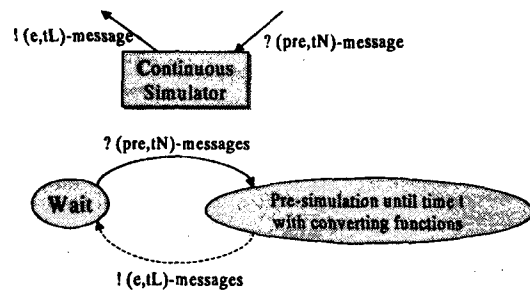
When receive (done, tN') from its discrete child
    tN = tN'

When receive (e, tL) from its continuous child through A/E converter
    tc = tL
    send (e, tL) to its discrete child

When receive (y, t)
    converting (y, t) using E/A converter and
    send it to its discrete child

End Hybrid Coordinator
    
```

<그림 13> Hybrid Coordinator Algorithm



<그림 14> State Diagram of CS

HC의 하위에 존재하는 CS는 연속 모델의 시뮬레이션 알고리즘을 수행한다. <그림 14>에서와 같이 CS 시뮬레이터는 Wait 상태에서 상위 HC로부터 받은 (pre,tN) 메시지를 정해진 시간 tN 까지 수행한다. 이러한 연속 시뮬레이션 수행 결과를 A/E 변환기를 거쳐 상위 HC에 전달하게 된다. 구체적으로 말하면 연속 출력이 입력된 A/E 변환기가 이산 출력 e 를 발생하면 e 가 tL 에 발생했다는 메시지 (e,tL) 을 상위 Coordinator로 전달하고 그렇지 않은 경우에는 이산 출력이 없다는 NULL 정보를 담은 $(NULL,tL)$ 을 상위 Coordinator에게 전달한다. <그림 15>의 Pseudo Code 형태로 서술한 것이다.

```

Continuous Simulator
; tL -> last event time
; tN -> next event time

When receive (pre, tN)
  simulate between tL and tN
  if an event e is found at tL'
    tL = tL'
    send (e, tL') to its parent
    return
  end if
end simulate
tL = tN
send (NULL, tL) to its parent

End Continuous Simulator
    
```

<그림 15> Continuous Simulator Algorithm

5. 하이브리드 시스템 모델링 예제

본 절에는 제안된 하이브리드 시스템 모델링 형식론 및 연동 시뮬레이션 개념을 의미 있는 실제 시스템에 적용하려고 한다. 모델링 대상이 되는 실제 시스템은 미사일이 목표를 향해 쫓아가는 목표 추적 시스템(Target Tracking System)이며, 이 시스템을 간략화 하여 하이브리드 시스템으로 모델링 하였다.

5.1 목표 추적 시스템 개요

<그림 16>에서와 같이 목표 추적 모델은 목표를 추적하는 '미사일'과 미사일의 추력을 조절하는 '컨트롤러' 그리고 미사일의 추적 결과와 환경적인 영향을 대표하는 '환경' 등은 연속 시스템으로 생각할 수 있다. 이러한 미사일의 추적을 원활하게 하기 위하여 컨트롤러의 모드를 세 가지로 조절하는 '이산 사건 결정 모델'을 추가할 수 있다. 목표 추적 예제 모델들의 기본적인 수행 요소는 다음과 같다.

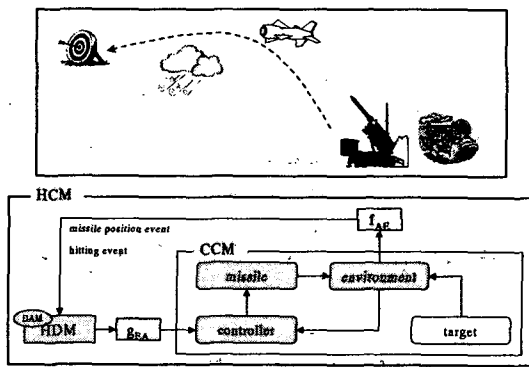
미사일 미사일의 운동방정식은 미사일의 자체 추력과 속도 성분에 반대 방향으로 비례하는 저항력, 그리고 중력을 원소로 한다.

환경 미사일의 위치정보를 받아 들여 목표와의 충돌을 일으킨다. 또한 위치와 속도 정보를 레이더에 표시해 준다.

컨트롤러 미사일의 추력을 결정하는 모델로 mode1에서는 미사일의 상승을 유도하고, mode2에서는 미사일의 가로 방향전진을

유도한다. 마지막으로 목표에 가까워지면 mode3으로 전환되어 목표에 빠르게 접근한다.

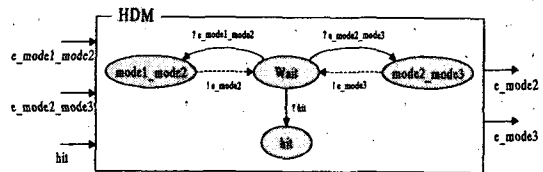
결정모델 환경으로부터 미사일의 정보를 받아서 미사일의 위치에 따르는 mode 전환명령을 보내게 된다.



<그림 16> Example Target Tracking System

5.2 이산 사건 시스템 모델링

연속 시스템으로부터 미사일의 위치에 따르는 입력 사건을 받아들이면 미사일의 mode를 변경하기 위해서 상태변수를 전이하고 출력으로 모드상수 값을 증가시킨다. <그림 17>에서 보이는 상위 결정 모델의 상태 변화를 제안된 하이브리드 모델링 형식론으로 나타내면 아래와 같다.



<그림 17> High-Level Decision Model(HDM)

$$HDM = \langle X_{disc}, S_{disc}, Y_{disc}, \delta_{int}, \delta_{ext}, \lambda, ta \rangle$$

$$X_{disc} = \{ hit, e_mode1_mode2, e_mode2_mode3 \}$$

- $S_{disc} = \{ wait, mode1_mode2, mode2_mode3 \}$
- $Y_{disc} = \{ e_mode2, e_mode3 \}$
- δ_{ext} :
 - $\delta_{ext}(wait, e, e_mode1_mode2) = mode1_mode2$
 - $\delta_{ext}(wait, e, e_mode2_mode3) = mode2_mode3$
 - $\delta_{ext}(wait, e, hit) = hit$
- δ_{int} :
 - $\delta_{int}(mode1_mode2) = wait$
 - $\delta_{int}(mode2_mode3) = wait$
- λ :
 - $\lambda(mode1_mode2) = e_mode2$
 - $\lambda(mode2_mode3) = e_mode3$
- ta :
 - $ta(wait) = INFINITY$
 - $ta(mode1_mode2) = SWITCHINDELAY$
 - $ta(mode2_mode3) = SWITCHINDELAY$
 - $ta(hit) = INFINITY$

$$A = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\frac{k}{m} & 1 \\ 0 & 0 & 0 & -\frac{k}{m} \end{pmatrix}, \quad B = \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ \frac{1}{m} & 0 \\ 0 & \frac{1}{m} \end{pmatrix}, \quad C = \begin{pmatrix} 0 \\ 0 \\ 0 \\ -g \end{pmatrix}$$

- g :
 $Y_{cont}(t) = S_{cont}(t)$

컨트롤러는 입력으로 미사일의 위치와 속도를 받아 들여서 출력으로 미사일을 직접 제어하는 추력을 발생한다. 특별한 상태변수는 존재하지 않으며, 입력으로 들어오는 모드상수에 의해서 컨트롤러가 가지고 있는 제어 함수가 결정된다. 초기 모드에는 y축 방향으로 최대힘을 가하여 y축 방향으로 상승을 그리고, 활강 모드에서는 중력만을 상쇄하고 나머지 힘은 x축 방향으로 내게 하여, x축 방향으로 전진하는 제어 방법을 사용한다. 마지막으로 추적모드에서는 목표 위치와 미사일의 위치, 속도를 변수로 하여 비례 미분 제어기(PD controller)에 의해서 추적하게 된다. 다음은 컨트롤러를 CAM으로 기술한 것이다.

5.3 연속 시스템 모델링

미사일은 입력으로 x축 방향의 추력과 y축 방향의 추력을 받아들인다. 미사일은 위치 변수와 속도 변수를 상태변수로 가지며, 동시에 출력하게 된다. 일반적인 물리 법칙에 의거하여 속도 성분에 비례하는 점성 저항력과 중력, 그리고 입력으로 들어오는 추력을 바탕으로 운동 방정식이 결정되며, 다음은 미사일 시스템을 제안된 하이브리드 시스템 모델링 형식론의 Atomic 연속모델로 기술한 것이다.

$$MISSLE = \langle X_{cont}, S_{cont}, Y_{cont}, f, g \rangle$$

- $X_{cont} = \{ f_x(t), f_y(t) \}$
- $S_{cont} = \{ s_x(t), s_y(t), v_x(t), v_y(t) \}$
- $Y_{cont} = \{ y_{s_x}(t), y_{s_y}(t), y_{v_x}(t), y_{v_y}(t) \}$
- f :
 $\frac{d}{dt} S_{cont}(t) = A \times S_{cont}(t) + B \times X_{cont}(t) + C$

$$CONTROLLER = \langle X_{cont}, S_{cont}, Y_{cont}, f, g \rangle$$

- $X_{cont} = \{ x_{s_x}(t), x_{s_y}(t), x_{v_x}(t), x_{v_y}(t), m(t) \}$
- $S_{cont} = \{ \}$
- $Y_{cont} = \{ y_{f_x}(t), y_{f_y}(t) \}$
- g :

$$\begin{pmatrix} y_{f_x}(t) \\ y_{f_y}(t) \end{pmatrix} = \begin{pmatrix} -k_x & 0 & -1 & 0 \\ 0 & -k_y & 0 & -1 \end{pmatrix} \times \begin{pmatrix} x_{s_x}(t) \\ x_{s_y}(t) \\ x_{v_x}(t) \\ x_{v_y}(t) \end{pmatrix} + \begin{pmatrix} k_x z_x \\ k_y z_y + mg \end{pmatrix}$$

$$\times (m(t) - 2) \times (m(t) - 1)$$

$$+ \left(\sqrt{\frac{f_{max}^2 - (mg)^2}{mg}} \right) \times (m(t) - 1) \times (m(t) - 3)$$

$$+ \begin{pmatrix} 0 \\ f_{max} \end{pmatrix} \times (m(t) - 2) \times (m(t) - 3)$$

5.4 Conversion Functions 모델링

연속시스템에서 출력되는 연속적인 미사일

의 위치와 속도 정보를 이산 사건 시스템의 상위 결정 모델로 전달하기 위해서 A/E 변환이 필요하며, 상위결정모델에서 출력되는 모드 변화 사건을 모드상수의 증가로 변환하기 위해서 E/A 변환이 필요하다. A/E 변환기는 x축 좌표 100 혹은 y축 100을 지나는 입력을 받으면 모드 1에서 모드 2로 전이하는 사건을 발생하고 x축 좌표 200 혹은 y축 좌표 1000을 지나갈 때 모드 2에서 모드 3으로 전이하는 사건을 발생한다. E/A 변환기는 모드 2 사건을 받으면 2의 값을 가지는 상수 함수를 사건이 들어오는 시간부터 입력으로 발생하고, 모드 3 사건을 받으면 3의 값을 가지는 상수 함수를 사건이 들어오는 시간부터 발생한다. 위의 내용을 형식론으로 기술하면 다음과 같다.

$$CF = \langle f_{AE}, g_{EA} \rangle$$

- f_{AE} :

$$f_{AE}(w_1(t)) = e_mode1_mode2$$

where

$$h_1(w_1(t)) = 0 \text{ and } \frac{dh_1(w_1(t))}{dt} < 0$$

$$h_1(x, y) = (x - 100) \times (y - 100)$$

$$f_{AE}(w_2(t)) = e_mode2_mode3$$

where

$$h_2(w_2(t)) = 0 \text{ and } \frac{dh_2(w_2(t))}{dt} < 0$$

$$h_2(x, y) = (x - 200) \times (y - 1000)$$

- g_{EA} :

$$g_{EA}(e_mode2) = 2 \times u(t - e_mode2.time)$$

$$g_{EA}(e_mode3) = 3 \times u(t - e_mode3.time)$$

6. 결론

본 논문에서는 연속 시스템과 이산사건 시스템이 혼재되어 있는 하이브리드 시스템을 기술하는 모델링 형식론인 HDEVS를 제안하였다. HDEVS는 하이브리드 시스템의 연속 시스템과 이산사건 시스템을 현재 사용하는 모델링 방법

을 그대로 사용하여 모델링 할 수 있는 틀을 제공한다. 본 연구에서 제안된 모델링 방법은 연속시스템인 경우 미분방정식 모델, 이산사건 시스템인 경우 DEVS 모델을 사용할 수 있도록 하였다. 또한 이들 모델간의 메시지 변환을 위하여 A/E 변환기와 E/A 변환기를 제안하였다. 이러한 모델링 방법은 모델을 실제 시물레이션 할 경우 예상되는 시물레이터의 연동을 통하여 기존 모델 라이브러리를 재 사용할 수 있는 잇점을 제공한다.

하이브리드 시스템의 이산사건 및 연속 시스템 모델들 사이의 시물레이션 시각 연동을 위한 시각 동기화 방법이 제안되었다. 제안된 방법은 동기화 과정에서 이산사건 시스템의 시간 진행에 따라 연속 시스템의 시간이 진행되도록 하는 Pre-Simulation 개념을 도입하였고, 이를 토대로 추상 시물레이터 알고리즘을 고안하였다.

제안된 모델링 및 시물레이션 방법론의 적용 예로 목표 추적 시스템을 하이브리드 시스템으로 모델링 하였다. 모델링된 하이브리드 시스템은 미사일과 컨트롤러의 물리적 운동방정식은 연속 시스템으로, 미사일의 위치에 따르는 적응형 컨트롤러의 전환을 담당하는 결정 시스템은 이산 사건 시스템으로 보고 각각을 모델링 하였다.

본 논문의 2부에서는 하이브리드 시스템을 시물레이션 하기 위한 연동 시물레이션 환경의 개념 및 구현 그리고 구현된 연동 시물레이터의 성능에 대하여 다룬다.

참고문헌

- [1] Panos J. Antsaklis, James A. Stiver and Michael D. Lemmon, "Interface and Controller Design for Hybrid Control Systems", *Hybrid Systems II*, Springer, pp. 462-492, 1994.
- [2] Bernard P. Zeigler, Herbert Praehofer and Tag Gon Kim, *Theory of Modeling and*

-
- Simulation 2nd edition*, Academic Press, 2000.
- [3] Thomas A. Henzinger and Pei-Hsin Ho, "HYTech: The Cornell Hybrid TECHNOlogy Tool", *Hybrid Systems II*, Springer, pp. 265-293, 1994.
- [4] Michael D. Lemmon, Kevin X. He and Ivan Markovsky, "Supervisory Hybrid Systems", *IEEE Control Systems*, pp. 42-55, 1999.
- [5] John Lygeros, Datta Godbole and Shankar Sastry, "Simulation as a Tool for Hybrid System Design", *1994 AIS*, pp. 16-22, 1994.
- [6] John T. Olson, John Britanik and Michael M. Marefat, "Combined Discrete/Continuous Simulation of Process Plans I: Simulation Methodology", *Transactions of SCS*, Vol. 16, No. 4, pp. 133-148, 1999.
- [7] Zeigler B. P., *Multifaceted Modeling and Discrete Event Simulation*. Academic Press, 1984.
- [8] Tag G. Kim, Lecture Note: EE612(Discrete Event Systems Modeling and Simulation), EECS Dept., KAIST, <http://smlab.kaist.ac.kr/Course/EE612>

● 저자소개 ●



임성용

한국과학기술원 전기/전자공학과 학사 (1999)

한국과학기술원 전기/전자공학과 석사 (2001)

졸업 후 현재 전자통신연구원, 네트워크기술연구소에 재직하고 있으며 주 연구분야는 하이브리드 시스템, 분산시물레이션, 시뮬레이터 연동, 네트워크 시물레이션 등임.

2000년 삼성 휴먼텍 논문 경시대회에서 본 논문으로 금상 수상.



김탁곤

1988: 아리조나대학교 전자/컴퓨터공학과 공학박사

1980 - 1983: 국립수산대학(현: 부경대학교) 통신공학과 전임강사

1987 - 1989: 아리조나 환경연구소 연구 엔지니어

1989 - 1991: 캔사스 대학교, 전자전산학과, 조교수

1991 - 1998: 한국과학기술원, 전기전자공학과 조/부교수

1998 - 현재: 한국과학기술원, 전자전산학과 교수

모델링/시물레이션 방법론 및 환경 개발, 시물레이션에 기반한 시스템 분석등에 관하여 국제적으로 활발한 연구 활동을 하고 있으며 국제학술지/국제학술발표대회 논문지에 100 편 이상의 논문을 게재하였으며, 시물레이션 모델링에 관련된 다수의 국제학술지 편집을 담당하고 있으며, 현재 국제시물레이션학회(SCS)에서 발간하는 학술자인 *Transactions of SCS* 의 Editor-in-Chief 을 맡고 있으며, 그외 다수의 국제 학술지 (*SIMULATION; Int. Journal of Intelligent Systems and Control; IEEE Simulation Digest*) 의 Associate Editor를 맡고 있음. 저서(공저자: B.P. Zeigler, H. Praehofer)로는 2000년 미국 Academic Press에서 발간한 모델링 시물레이션 전문 교재인 *Theory of Modeling and Simulation(2nd edition)* 이 있음. 한국시물레이션 학회 초대 편집위원장을 역임하였고, 현재 학술부회장임. 국제 학술 단체인 IEEE(국제 전기전자 학회) 및 SCS(국제 시물레이션 학회)의 Senior Member이며, ACM(미국전산학회) 및 Eta Kappa Nu 의 Regular Member 임.