

# 노드 이동성을 고려한 애드 혹 네트워크의 이산 사건 시스템 기반 모델링 및 시뮬레이션 방법론

송상복<sup>1</sup> · 이규호<sup>1†</sup>

## A DEVS-based Modeling & Simulation Methodology of Enabling Node Mobility for Ad Hoc Network

Sang-Bok Song · Kyou-Ho Lee

### ABSTRACT

Modeling and Simulation, especially in mobile ad hoc network(MANET), are the most effective way to analyze performance or optimize system parameters without establishing real network environment. Focusing mainly on overall network behaviors in MANET concerns dynamics of network transport operations, which can efficiently be characterized with event based system states rather than execution details of protocols. We thus consider the network as a discrete event system to analyze dynamics of network transport performance. Zeigler's set-theoretic DEVS(Discrete Event Systems Specification) formalism can support specification of a discrete event system in hierarchical, modular manner. The DEVSsim++ simulation environment can not only provide a rigorous modeling methodology based on the DEVS formalism but also support modelers to develop discrete event models using the hierarchical composition methodology in object-orientation. This environment however hardly supports to specify connection paths of network nodes, which are continuously altered due to mobility of nodes. This paper proposes a DEVS-based modeling and simulation methodology of enabling node mobility, and develops DEVS models for the mobile ad hoc network. We also simulate developed models with the DEVSsim++ engine to verify the proposal.

**Key words** : MANET, DEVS, DES, M&S, Mobility

### 요약

MANET(Mobile Ad-hoc NETWORK)에 있어 모델링 및 시뮬레이션은 실제 시스템 환경을 구축하기 어려운 여건에서 가상의 환경을 통한 분석연구를 위한 가장 효과적이고 유용한 방법이다. MANET의 연구에 있어서 네트워크 전체의 동작특성을 관찰하기 위해서는 노드간 전달과정과 관련한 상태 및 특성변화의 추적이 중요하며 이산 사건 시스템의 특징을 가진다. Zeigler's DEVS(Discrete Event System Specification) 형식론은 계층적이고 모듈라한 기법으로 이산사건 시스템을 명세할 수 있는 수학적 형식론이며, 이에 기반한 DEVSsim++는 모델링의 무결성을 제공하며 객체지향형기법에 의한 계층적 최적화 모델링 및 시뮬레이션 환경을 제공한다. 그러나 네트워크를 구성하는 노드의 이동 특성으로 인해 네트워크의 연결 상태가 지속적으로 변하는 MANET을 모델링하기에는 어려운 부분이 있다. 본 논문에서는 DEVS방법론을 도입하여 MANET을 모델링하는 과정에서 노드의 이동 특성을 고려한, 네트워크의 변화특성에 따른 네트워크 상태의 변화를 관찰하기 위하여 네트워크 특성을 중심으로 MANET을 표현하는 방법을 제안하고 MANET DEVS 모델을 제시한다. 또한 제시한 모델을 DEVSsim++시뮬레이션 엔진에 적용하여 시뮬레이션 함으로써 모델의 동작을 실증하였다.

**주요어** : 모바일 애드혹 네트워크(MANET), DEVS, 이산사건시스템, 시뮬레이션 및 모델링, 이동성(mobility)

2009년 9월 8일 접수, 2009년 12월 17일 채택

<sup>1)</sup> 인제대학교 대학원 전자정보통신공학과

주 저 자 : 송상복

교신저자 : 이규호

E-mail; sbsong@netcus.com, kyou@inje.ac.kr

## 1. 서 론

모델링 및 시뮬레이션은 실제 시스템 환경을 구축하기 어려운 여건에서 가상의 환경을 통하여 문제점이나 성능을 분석할 수 있게 함으로써, 시스템 개발 과정에서 발생할 수 있는 오류를 미리 차단해 주거나 요구되는 설계요소의 최적값을 사전에 구할 수 있게 하는 가장 효과적이고 유용한 방법이다.

무선 통신의 한 형태인 MANET은 이동 노드들이 필요에 따라 임시로 구성하는 네트워크로서, 오래전부터 군사 네트워크, 재해 상황에서의 네트워크 등을 위하여 연구되어 왔으며, 최근에는 유비쿼터스 환경 구성을 위한 무선 통신 기술 중 하나로 각광받으면서 그 중요성이 높아지고 관련 연구가 활발하게 진행되고 있는 분야이다.

MANET의 응용서비스를 위한 네트워크나 시스템의 개발이나 연구과정에 있어 많은 비용과 시간을 필요로 하는 테스트베드 등을 직접 구축하는 방법보다는 가상 환경을 이용하는 모델링 및 시뮬레이션에 의한 방법이 일반적으로 더 효과적이다. 더구나 그러한 연구개발 과정에서 필요로 하는 최적 설계요소를 구하기 위해서는 모델링 및 시뮬레이션이 필수적이다.

지금까지 MANET의 성능 분석 연구에는 유용한 기능과 편리성을 제공하는 NS-2와 OPNET과 같은 상용 시뮬레이션 환경이 주로 활용되어 왔다. 그러나 이런 시뮬레이션 환경들은 형식론(Formalism)과 같은 이론적 모델링 방법론에 기반하지 않아 무결성 보장이 미흡하며, 또한 노드에 포함된 프로토콜의 세부 기능에 모델링 요소가 집중되어 네트워크 전체의 동작특성을 관찰하기 위하여 필요로 하는 성능지수에 최적화된 추상화 모델링이 어려운 단점을 가지고 있다.

KAIST의 DEVSim++은 DEVS 방법론에 의한 이산사건 시스템 모델링 및 시뮬레이션 환경으로서, DEVS형식론에 기반한 무결성을 제공하며 특히 주어진 성능지수에 최적화된 추상화 모델링에 매우 유리하다.

MANET의 연구에 있어서 네트워크 전체의 동작특성을 관찰하기 위해서는 노드간 전달과정과 관련한 상태 및 특성변화의 추적이 중요하며 이는 이산 사건 시스템의 특징을 가진다<sup>[1]</sup>.

MANET을 구성하는 노드는 각각 이동성을 가지며, 따라서 노드간의 연결상태에 의해 구성되는 네트워크 토폴로지가 지속적으로 바뀌는 특성이 있다. 이를 모델(노드)간의 연결 상태를 결합모델로 규정하는 DEVS 방법론으

로 모델링 하는 데는 어려움이 따르며 기존의 시뮬레이터들이 가진 네트워크의 변화특성을 관찰하기 어려운 단점을 그대로 가질 수 있다.

본 논문에서는 이러한 DEVS의 한계를 해결하고 네트워크의 전달 변화특성 관찰에 적절한 모델링 방법을 제안한다. 제안된 방법은 MANET의 노드 이동성에 따른 토폴로지의 계속된 변화를 기준으로 네트워크의 변화특성을 한 factor로 정의하여 모델링에 도입하였다. 또한 제안된 방법에 의하여 모델링된 MANET DEVS 모델을 제시하고, 이를 DEVSim++로 구동하여 모델의 동작을 실증하였다<sup>[2]</sup>.

본 논문은 다음과 같이 구성된다. 2장에서는 기존의 MANET 모델링 관련 연구와 다른 시뮬레이션 환경에서의 MANET 모델링 방법을 분석하고, 3장에서는 MANET을 DEVS로 표현하는 방법에 대해 제안한다. 4장에서는 해당 방법으로 모델링 된 DEVS MANET 모델을 제시하고 5장에서는 해당 모델을 통해 시뮬레이션 된 결과를 통해 모델이 MANET의 특징을 잘 반영했는지를 실증한다. 마지막으로 6장에서 결론을 맺는다.

## 2. MANET의 모델링 및 시뮬레이션

MANET의 모델링 및 시뮬레이션에 관련된 연구는 크게 세가지 관점으로 진행되어 왔다. 기존의 시뮬레이터 및 모델을 활용하여 추가 및 발전시키는 연구와 모바일 노드의 이동 특성을 분석하여 이동성을 모델링 하는 연구, 그리고 MANET이나 모바일 노드의 모델링에 관련된 연구이다. 그 중 본 논문의 논점과 가장 유사한 연구는 MANET이나 모바일 노드에 대한 모델링 연구이며, 가장 널리 사용되고 있는 시뮬레이션 환경인 NS-2 및 OPNET과, 기존의 DEVS와 본 논문에서 제안하는 모델 등을 이러한 관점에서 상호비교하면 표 1과 같다.

### 2.1 NS-2의 MANET 모델

NS-2는 네트워크 전문 시뮬레이터로 객체지향 방식의

표 1. 기존 연구간의 연구 요소 비교

요소	NS-2 모델	OP NET	기존의 DEVS	제안 모델
모델링 관점	노드동작	노드동작	노드동작	전체상태
모델링 중심	노드	노드	노드	네트워크
이동성	O	O	X	O
이론적 기반	X	X	DEVS	DEVS

모델링 환경으로, 공통적인 많은 레이어들이 이미 구현된 부모클래스로 존재하며 사용자는 부모 클래스를 상속받아 새로운 모델을 구현할 수 있다. NS-2에서의 MANET은 모바일 노드의 동작에 기반 하여 모바일 노드간의 통신과 노드의 위치 변화에 따른 노드 이동 기능을 제공한다<sup>[3]</sup>.

NS-2는 계층별로 기능 블록을 구성하고 객체로 구현된 정교한 모바일 노드 모델을 통해 MANET을 구성한다. MANET의 동작은 Routing Agent에 해당하는 부분에서 AODV(Ad-hoc On-demand Distance Vector), DSDV(Destination Sequenced Distance Vector), DSR(Dynamic Source Routing) 등 MANET의 라우팅 프로토콜들이 담당하고 송수신에 관련된 기반 모델들은 모두 NS-2에서 기본적으로 제공하는 무선 채널을 사용한다. 노드의 이동성은 파일을 통해 노드의 좌표 변화를 표시하는 이동성 파일을 불러오는 방식으로 시뮬레이션 엔진이 제공하고 있다.

NS-2에서의 MANET 모델은 노드의 세부 기능을 중심으로 모델링 되어 전체 네트워크 변화특성 관찰에 적합하지 못하며 형식론 등에 근거하고 있지 않아 무결성이 결여되어 있다.

### 2.2 OPNET의 MANET 모델

OPNET은 현재 상용으로 판매되고 있는 네트워크 시뮬레이터 시장의 80% 이상을 차지하고 있다고 알려진 장

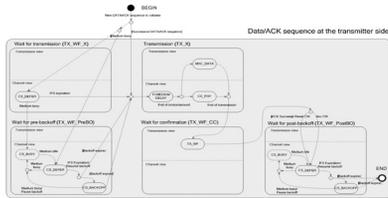


그림 1. OPNET 모바일 노드 송신부 모델

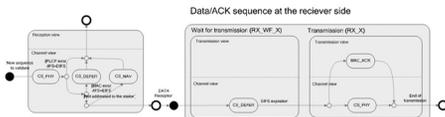


그림 2. OPNET 모바일 노드 수신부 모델

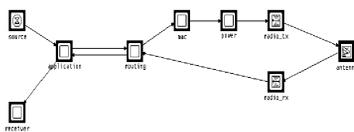


그림 3. OPNET 모바일 노드 모델

력한 네트워크 시뮬레이터이다. OPNET은 NS-2에 비해 직관적이고 계층적인 모델링을 지원하는데 Network, Node, Process 별로 에디터를 두고 계층적으로 모델링이 가능하다. OPNET의 네트워크 에디터를 통해 노드의 위치와 이동성 모델 등을 편집하여 MANET을 구성하며, Process 흐름도를 그대로 사용할 수 있는 강력함을 제공한다<sup>[4]</sup>.

그림 1과 그림 2는 OPNET으로 모델링 된 모바일 노드의 송신부와 수신부를 각각 나타낸다. OPNET의 노드 모델은 이와 같은 송수신부를 그림 3과 같이 노드 에디터를 통해 여러 요소들과 조합하여 모바일 노드를 만들어낸다<sup>[4][5]</sup>.

이러한 OPNET의 상용화된 그래픽 사용자환경은 다른 시뮬레이션 환경에 비해 사용자의 편의성을 제공하는 장점이 있지만 NS-2와 마찬가지로 모델링의 무결성이 결여되는 단점과, 네트워크의 상태, 즉 노드와 노드간의 관계 등의 표현이 어려워 네트워크의 변화특성을 관찰하기 위한 모델링에는 한계가 있다.

### 2.3 노드 중심의 MANET DEVS 모델

DEVS의 모델링 프로세스에 맞춰 Top-Down 방식으로 MANET을 이동 노드의 집합으로 보면, MANET은 그림 4와 같이 노드들의 집합으로 생각할 수 있다.

그림 5와 같은 모바일 노드들의 집합으로 그림 4와 같은 하나의 MANET이 구성되며, 노드들은 그림 6의 AODV 라우팅 프로토콜과 같은 MANET 라우팅 프로토콜을 통해 네트워크를 구성하게 된다. 그러나 그림 4에서 보듯이, 결합 모델의 연결은 고정적인데 비해 MANET의 모델(노드)간 연결은 계속해서 바뀌어야 한다. 노드간의 연결 상태인 토폴로지가 계속 바뀌는 것이 MANET의 가장 큰 특징이기 때문이다. 이러한 방법으로는 결합 모델의 연결 상태를 수시로 바꾸는 것이 어렵기 때문에 DEVS에 의하여 노드의 이동성을 고려한 MANET을 모델링하기가 어렵다. 따라서 DEVS의 해당 문제점을 보완할 새로운 모델링 기법이 필요하다. 또한 네트워크의 상태 변화

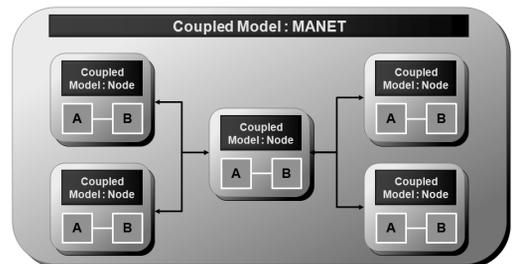


그림 4. 노드 중심의 MANET Coupled Model

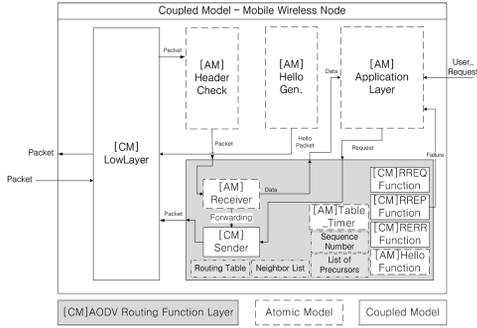


그림 5. 모바일 노드의 Coupled Model

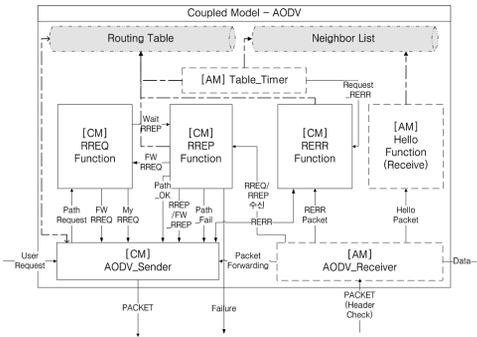


그림 6. MANET 라우팅 프로토콜 AODV 모델

를 통해 네트워크의 변화특성을 관찰하기에 기존의 노드 중심의 설계를 그대로 따라가는 것은 적절하지 않다.

본 논문에서는 네트워크의 상태인 토폴로지의 변화에 관점을 둔 MANET 표현 방법을 제안하고, 제안된 방법을 적용하여 DEVS로 MANET을 모델링 하는 방법을 제안하고 모델링 된 MANET DEVS 모델을 제시한다.

### 3. 노드의 이동성과 Edge Cost

#### 3.1 네트워크 관점에서의 노드 이동성

그림 7을 보면 시간 'T'에 5개의 노드가 모두 서로 연결되어 있는 상태로 머물러 있다. 시간이 흘러 시간이 'T+'가 되었을 때 노드 'C'가 이동하여 T+에 해당하는 오른쪽 토폴로지로 변했다. 이 토폴로지 네트워크의 상태가 천이하면서 바뀐 것은 A-C 링크와 D-C 링크가 단절되었다는 것이다. 여기서 중요한 점은 단순히 노드의 위치가 바뀌는 것은 네트워크 관점에서 의미가 없다는 점이다. 노드의 위치가 아무리 많이 변해도 모든 노드가 같은 방향으로, 같은 속도로 이동하면 원칙적으로 네트워크는 계속해서 같은 상태를 가진다. 즉, 네트워크 관점에서 노드의

Adjacency_Matrix <sub>1</sub>					Adjacency_Matrix <sub>end</sub>				
	1	2	...	n		1	2	...	n
1	O	O	O	O	1	O	X	X	X
2	O	O	O	O	2	X	O	X	X
...	O	O	O	O	...	X	X	O	X
n	O	O	O	O	n	X	X	X	O

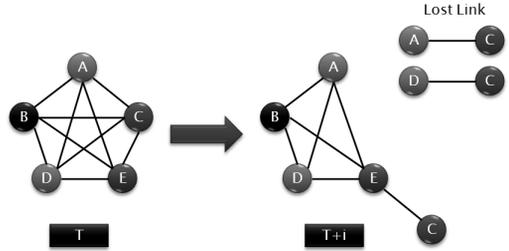


그림 7. 네트워크 관점에서의 노드 이동성

이동성이란, 노드가 위치적으로 변화하는 것이 아니라 네트워크상에서의 노드와 노드간 연결 상태 변화를 의미한다.

$$AMatrix_1 = ALL\ Edge\ Connect$$

$$AMatrix_{end} = ALL\ Edge\ Disconnect$$

$$State_{MANET} = \{AMatrix_1, \dots, AMatrix_{end}\}$$

#### 3.2 Edge와 MANET의 상태

노드와 노드간 연결 상태를 나타내는 링크를 흔히 Edge라 부르는데, 이 Edge를 도식적으로 표시하여 네트워크를 표현하는 방법을 흔히 네트워크 토폴로지라고 부른다. 즉, 네트워크의 상태란 네트워크에 속한 모든 노드들에 대한 Edge 연결 상태를 나타내면 표현할 수 있는데 흔히 이런 상태를 표시할 때 ‘인접 행렬(Adjacency Matrix)’을 사용한다. 인접행렬은 노드의 개수 n에 대한 n × n 행렬로 Edge의 상태를 나타낸다.

위와 같이 MANET의 상태집합은 모든 Edge가 연결된 상태에서 모든 Edge가 단절된 상태까지 한 시점에서의 인접행렬 집합으로 나타내어진다.

특히 MANET은 모든 Edge가 연결되어 모든 노드가 서로간에 연결될 가능성을 가진다. 이렇게 모든 Edge가 연결되어 모든 노드가 다른 노드로 직접 통신할 수 있는 상태에 머물러 있는 네트워크를 흔히 메시 네트워크(Mesh Network)라고 특징지어 표현한다.

#### 3.3 Edge Cost와 Edge Limit

노드의 이동에 의해 Edge의 연결 상태가 변화하는 것이 네트워크 관점에서 노드의 이동성이라고 하면 Edge가

끊어지고 이어지는 것만 고려하면 정확한 모델링이 불가능하다. 왜냐하면 이어진 Edge 중에서도 연결 상태가 모두 같지 않기 때문이다. 이것은 노드와 노드간의 거리나 전파의 송수신 감도, 송수신 가능 대역폭 등으로 표현된다. 이럴 때 Bellman-Ford 알고리즘 등 잘 알려진 라우팅 알고리즘에서는 Edge에 비용을 두어 Edge의 상태를 고려하여 최단거리를 찾아내곤 한다. 본 논문에서도 그 방식을 도입하여 사용하며 Edge에 부여되는 비용을 Edge Cost라고 부르고 앞으로 영문약자인 EC로 표기한다.

EC는 노드와 노드간의 거리, 노드 사이의 장애물과 노드의 높이, 안테나의 종류 및 이득, 기상상태 등 수많은 요소에 의해 결정될 수 있다. EC를 가장 정확히 모델링하는 방법은 Hata-Okumura model<sup>[6][7]</sup>, Longley-Rice Model<sup>[8]</sup>, Ray-Tracing model<sup>[9][10]</sup> 등의 전파환경 예측모델을 활용하는 것으로 본 논문의 논점에서 벗어나므로 본 논문에서는 EC를 노드와 노드간의 거리로 가정한다.

EC는 노드간 거리의 변화에 따라 계속해서 변화하며 어느 시점에서 노드와 노드간의 더 이상 통신이 불가능해지면 단절된다. Edge가 단절되고 연결되는 기준 값을 정하는 것은 여러 가지 요소가 감안될 수 있으며 대체로 노드의 최대수신 전력과 송수신 노드간의 최대 수신 전력 차에 따라 결정될 수 있다. 본 논문에서는 이와 같은 기준 값을 Edge Limit로 표현하고 EL로 표기한다. EL은 하드웨어의 성능과 관련된 부분으로 노드의 종류에 따라 달라질 수 있으며 모델링 과정에서는 특정한 값으로 가정하여 사용할 수 있다.

EL 값을 하나의 값으로만 정해서 Edge의 연결 상태를 결정하는 것은 현실에 맞지 않다. 그것은 이력현상(履歷

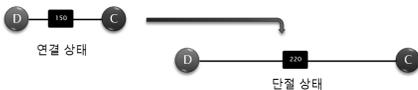


그림 8. EC에 따른 연결과 단절

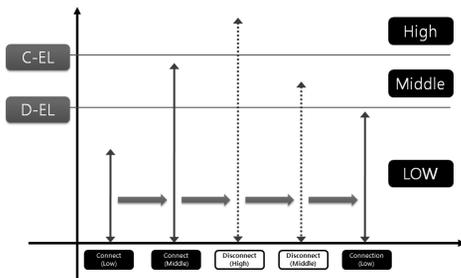


그림 9. EC의 변화에 따른 Edge의 상태변화

現象, Hysteresis)에 의한 이력감쇠효과(Hysteresis Loss) 이론에 맞지 않기 때문이다. 하나의 EL 값만을 사용하면 지나치게 단절과 연결이 순식간에 반복될 수 있다. 그래서 Edge가 연결된 상태에서 단절을 결정하는 EL값과 단절된 상태에서 다시 연결되는 EL값을 분리하며 각각 Connect-EL(C-EL)과 Disconnect-EL(D-EL)로 표기한다. 이에 따라 EC는 그림 9와 같이 총 3개의 구간(Low, Middle, High)으로 나뉘며 Edge의 상태는 4가지가 존재하게 된다.

$$Status_{Edge} = \left\{ \begin{matrix} Connect_{Low}, Connect_{Middle}, \\ Disconnect_{Middle}, Disconnect_{High} \end{matrix} \right\}$$

### 3.4 MANET의 상태와 MANET의 표현

앞 절에서 정의한 개념과 더불어 ECE(Edge Cost based manet Expression method) 즉, 에지 비용 기반 MANET 표현법으로 MANET(네트워크)의 상태를 표현하면 다음과 같은 4단계로 표현할 수 있다.

$$1. Status_{Edge} = \left\{ \begin{matrix} Connect_{Low}, Connect_{Middle}, \\ Disconnect_{Middle}, Disconnect_{High} \end{matrix} \right\}$$

$$2. MANET = \{ Edge_1, Edge_2, \dots, Edge_{end} \}$$

$$3. AdjMatrix_x = \begin{bmatrix} 1 & 2 & \dots & n \\ 1 & C_L & C/D_{LMH} & C/D_{LMH} & C/D_{LMH} \\ 2 & C/D_{LMH} & C_L & C/D_{LMH} & C/D_{LMH} \\ \dots & C/D_{LMH} & C/D_{LMH} & C_L & C/D_{LMH} \\ n & C/D_{LMH} & C/D_{LMH} & C/D_{LMH} & C_L \end{bmatrix}$$

$$4. Status_{MANET} = \{ AdjMatrix_1, AdjMatrix_2, \dots, AdjMatrix_{end} \}$$

- 1) Edge의 상태는 4가지로 표현된다.
- 2) MANET 혹은 네트워크는 이런 4가지의 상태를 가진 Edge들의 집합으로 구성된다
- 3) 어느 한 시점의 네트워크 토폴로지 정보는 Edge 상태에 대한 인접행렬을 통해 표현될 수 있다. 예를 들어 아래와 같은 인접 행렬을 가진 토폴로지는 그림 10과 같다.
- 4) MANET의 상태집합은 Edge 상태를 나타내는 인접행렬의 집합으로 구성된다. MANET의 가장 큰 특징을

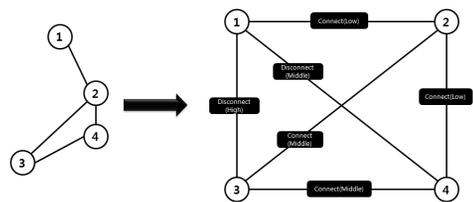


그림 10. Edge 상태를 통한 토폴로지 표현

토폴로지가 수시로 변하는 네트워크라는 점을 고려하면, MANET은 상태(Edge 상태를 나타내는 인접행렬의 종류)가 수시로 변하는 시스템이다.

$$AdjacencyMatrix_{\text{그림10}} = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 1 & C_L & C_L & D_H & D_M \\ 2 & C_L & C_L & C_M & C_L \\ 3 & D_H & C_M & C_L & C_M \\ 4 & D_M & C_L & C_M & C_L \end{bmatrix}$$

따라서 MANET은 모든 Edge가 연결될 가능성이 있는 가장 하에 수시로 변하는 EC 값과 C-EL, D-EL 값을 기준으로 한 Edge의 상태 변화를 통해 표현될 수 있다. 이는 노드의 동작이 아닌 네트워크의 상태 변화를 통해 MANET을 모델링하는 방법이다.

그림 11은 ECE의 한 예를 보이기 위함이다.

그림 11의 MANET은 시간 T에서 T+i로 변하면서 노드 1이 이동하여 토폴로지가 바뀌었다. 이는 아래와 같은 인접행렬, 즉 MANET의 상태 변화로 표현할 수 있다. 이 네트워크의 상태 변화는 EC의 변화에 따라 DEVS 모델 내에서 C/D-EL값을 기준으로 상태 천이가 발생된다. 이를 맡는 모델이 다음절에서 설명할 ECG(Edge Cost Generator)이다.

$$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 1 & C_L & C_L & D_H & D_M \\ 2 & C_L & C_L & C_M & C_L \\ 3 & D_H & C_M & C_L & C_M \\ 4 & D_M & C_L & C_M & C_L \end{bmatrix}_T \rightarrow \begin{bmatrix} 1 & 2 & 3 & 4 \\ 1 & C_L & D_H & C_L & C_L \\ 2 & D_H & C_L & C_M & C_L \\ 3 & C_L & C_M & C_L & C_M \\ 4 & C_L & C_L & C_M & C_L \end{bmatrix}_{T+i}$$

### 4. MANET DEVS 모델링

본 장에서는 ECE를 통해 표현된 MANET에 대한 DEVS 모델링 방법을 제안하고 MANET DEVS 모델을 제시한다.

#### 4.1 대상 시스템 분석

앞에서 정의한 ECE 개념으로부터 MANET이라는 시

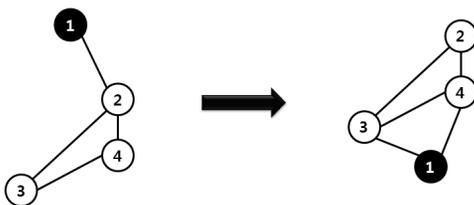


그림 11. 노드의 이동에 따른 토폴로지의 변화

스템의 구성요소를 찾아내면 다음과 같다.

- ① 모든 Edge가 연결될 가능성
- ② 수시로 바뀌는 EC 값과 C-EL, D-EL

그 중에서 모든 Edge가 연결될 가능성에서 모델링 대상은 Edge를 구성하는 ‘노드’와 노드와 노드간의 연결인 Edge를 구성해주는 실체인 ‘무선 채널’, 그리고 ‘모든 노드를 연결하는 Edge’로 나누어 진다.

이를 통하여 구성 요소를 다시 정리하면 다음과 같다.

- 노드
- 무선 채널
- 모든 노드를 연결하는 Edge
- Edge에 따른 EC 값과 C-EL, D-EL
- EC의 변화

#### 4.2 필수 구성 요소 특정 및 결합모델 모델링

여기서 노드의 구체적 동작은 큰 의미가 없다. 노드는 패킷을 ‘무선 채널’로 연결되는 Edge로 보내고 받는 역할만을 한다. ECE를 통한 MANET 모델에서 중요한 것은 EC와 EC를 변화시킬 모델, 그리고 변화된 EC에 맞춰 Edge의 상태를 결정 지어줄 모델이다.

이에 따라 세 가지의 필수 구성요소가 특정된다.

- ① Edge Cost Table(ECT) : EC를 비롯한 Edge의 상태에 대한 정보를 저장하는 테이블이다.
- ② Edge Cost Generator(ECG) : EC를 정해진 시간과 공식에 맞춰 변화 시키는 모델이다.
- ③ Wireless Channel(WC) : Edge의 상태에 따라 송수신을 결정하는 결정자이다. 실제 시스템에서 노드가 패킷을 무선 채널로 송신하면 무선 채널의 상태인 Edge의 상태에 따라 수신감도가 달라지듯, 전송성공을 결정하는 모델이다.

위의 3가지 필수 구성요소와 함께 무선 노드 모델, 노드의 정보를 포함하는 노드 정보, 경로의 정보를 포함하는 인접행렬, 그리고 트래픽의 정보가 필요하다. 이에 따

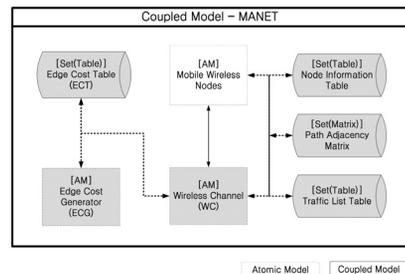


그림 12. MANET의 Coupled Model

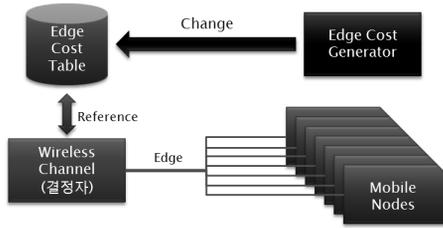


그림 13. MANET 모델들의 동작

라 MANET의 결합 모델이 그림 12와 같이 모델링 된다 (원기동: 테이블, [AM]-Atomic Model).

### 4.3 주요 원자모델 동작

그림 12의 MANET DEVS 모델의 동작을 간략히 표현하면 그림 13과 같다.

그림 13과 같이 MANET 모델은 ECG가 노드의 이동에 맞춰 ECT의 EC값을 바꾸게 되고 WC모델이 무선 노드의 Edge 상태를 결정해 전송 가능 여부를 결정해주게 된다.

그림 14의 네트워크의 예를 들면 A가 비용 80만큼 이동했다면 ECG는 A와 연결된 Edge들의 비용을 변화시켜 네트워크 토폴로지, 즉 MANET의 상태를 바꿔 MANET 시스템의 동작을 표현한다.

주요 구성 요소의 모델링 및 기능은 다음과 같다.

#### 4.3.1 Edge Cost Table

```

Edge Cost Table의 구현(c++)
enum EDGESTATE {CL, CM, DM, DH};
typedef struct {
    int edgeNumber; // 에지 번호
    int nodeX; // 에지 노드 X
    int nodeY; // 에지 노드 Y
    bool RE; //에지 종류
    EDGESTATE connectionStatus; //현재 연결 여부
    int edgeCost; //에지 비용
    int C_edgeLimitCost; // C-ELC
    int D_edgeLimitCost; // D-ELC
}edgeCostTable;
    
```

ECT는 위와 같이 8개의 구성요소로 이루어지며 Edge의 상태와 번호, Edge를 구성하는 노드와 EC등을 가진다. ECT를 생성해 내는 것은 ECE 표현법에 의한 MANET 모델링에서 가장 중요한 것으로 두 가지 과정이 필요하다. 네트워크에 따른 Edge의 총 수를 구해내는 것과 Edge를

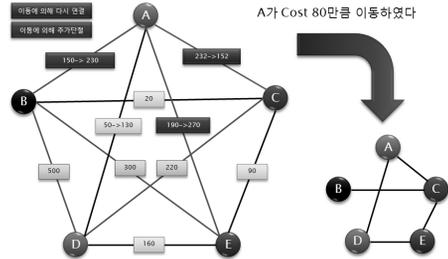


그림 14. MANET 모델의 동작 예

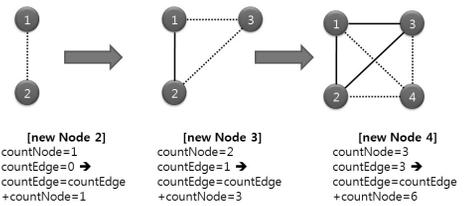


그림 15. Edge의 숫자 구하기

생성해 내는 것이다.

Edge가 형성되는 것은 네트워크에 새로운 노드가 추가될 때마다 기존에 있던 노드와 하나씩의 Edge를 연결해주는 과정이다. 그 과정이 그림 15에 나타나 있으며 Edge의 숫자는 이를 통하여 공식을 유도할 수 있으며 아래의 식으로 나타낼 수 있다.

$$countEdge = \sum_{current = 1}^{countNode - 1} current$$

or

$$\sum_{current = countNode - 1} 1$$

그림 15의 과정을 응용하여 빠짐없이 네트워크를 구성하는 Edge를 생성하여 ECT를 구성하는 방법은 Edge의 숫자를 구해내는 과정과 유사하다. Edge를 생성하는 알고리즘은 아래의 의사코드로 나타낼 수 있다.

```

Pseudo Code : ECT 생성 알고리즘
nx 는 1부터 시작
ny 는 nx+1(2)부터 시작
반복문 시작 : countEdge의 횟수 만큼 반복한다{
    ECT에 Edge(nx, ny) 구성
    ECT 나머지 부분 초기화
    [*]초기화시 Edge가 RE인지 IE인지 판단하여 초기화
    ny는 다음 노드로 증가
    if(ny가 countNode 보다 크면) 패턴 증가(nx 1증가,
    ny는 nx+1로 초기화);
    if(nx==countNode) 반복문 종료 else 다시 반복
} 반복문 종료
    
```



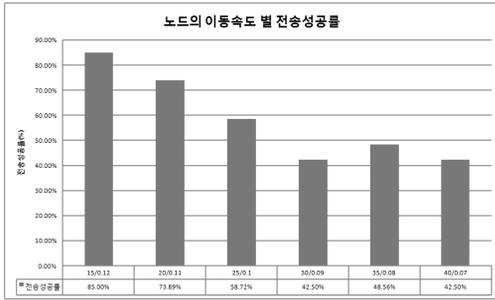


그림 19. 노드의 이동속도 별 전송성공률 변화

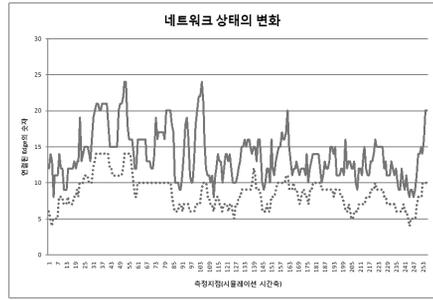


그림 20. 네트워크의 상태 변화

를 나타내고 높을수록 더 높은 확률로 전송이 성공할 수 있다. 이를 통해 그림 19의 결과가 정확함을 알 수 있다.

## 6. 결론

MANET의 연구와 응용서비스 인프라 개발에 있어 모델링 및 시뮬레이션은 매우 중요한 요소이다. 하지만 기존의 시뮬레이터들은 강력한 기능에도 불구하고 이론적 모델링 방법론에 기초하지 않아 시뮬레이션 결과의 무결성이 결여되는 단점과 함께 무선 노드의 세부 동작에 치중하여 전체 네트워크의 변화특성을 관찰하기에 적절하지 않다는 단점이 있다.

이에 본 논문에서는 MANET의 특성을 분석하기에 적합한 모델링 방법론인 DEVS를 도입하여 이 문제를 해결하고자 하였다. DEVS를 통해 MANET을 노드의 동작을 중심으로 MANET을 모델링하고자 할 때 DEVS로 표현하기 어려운 점의 극복과 함께 네트워크의 변화특성을 관찰하고 분석하기에 적절한 모델을 모델링하기 위해 MANET의 네트워크 특성을 분석하고 그를 통하여 Edge Cost와 Edge Limit를 기준으로 한 Edge의 상태를 통해 MANET을 표현하는 방법인 ECE를 제안하고 ECE를 통한 DEVS MANET 모델을 제시하였다. 또한 ECE DEVS MANET 모델을 검증된 시뮬레이션 엔진인 DEVSim++로 구동하여 모델의 동작을 실증하였다.

시뮬레이션 결과 ECE DEVS MANET 모델은 MANET의 가장 큰 특징인 노드의 이동에 따른 토폴로지, 즉 네트워크 상태의 변화를 이상 없이 표현하고 있으며 네트워크 상태 변화에 따른 전송성공률 등 네트워크 관점의 전달성능 변화를 관찰하는데 적절했다.

하지만 본 논문에서 제시한 ECE DEVS MANET 모델은 모바일 노드의 모델을 아주 간단한 입출력 장치로 고려하여 모바일 노드의 라우팅 프로토콜에 따른 MANET

의 성능 변화 등 노드 기준의 MANET 성능분석을 하는데 어려움이 존재한다. 이를 해결하기 위해서는 2.3절에서 제시한 모바일 노드 모델을 반영하는 추가 연구가 필요하다. 또한 EC의 값을 노드와 노드간의 거리 뿐만 아니라 노드와 노드 사이의 지형지물과 장애물등을 모두 고려하여 정확하게 뿐만 아니라 변화를 나타내기 위해서는 3.3절에서 언급한 전파환경 예측모델을 적용한 EC 모델링에 대한 추가 연구가 필요하다. 또한 랜덤한 형태의 노드 이동성을 가진 MANET 뿐만 아니라 특정한 이동 특성을 가진 노드로 구성된 MANET을 모델링 하기 위해서는 MANET의 이동관련 모델을 ECE의 ECG에 반영하는 추가 연구도 필요하다.

이러한 사항뿐 아니라 DEVS를 통하여 MANET을 모델링하고자 했던 기존의 연구에 대한 추가적인 비교 분석과 DSDEVS(Dynamic Structure DEVS)<sup>[11]</sup> 등 동적 환경을 모델링 하기위한 Formalism에 대한 추가 연구도 필요하다. 또한 본 논문에서 제안한 ECE MANET DEVS 모델에서 생략되거나 추상화된 부분들은 모두 모델링을 통해 분석하고자 하는 분석 목적에 맞춰 추가 연구되어야 하는 점으로서 목적 지향적 모델링 및 시뮬레이션의 관점에서 ECE MANET DEVS Model를 보완하기 위한 향후 연구가 필요할 것으로 생각된다.

## 참고 문헌

1. Zeigler, N.P. 외 2, *Theory of Modeling and Simulation: Integrating Discrete Event and Continuous Complex Dynamic Systems*, Academic, 2000.
2. 안명수 외 2, “DEVSim++ : 의미론에 기반한 이산사건 시스템의 객체지향 모델링 및 시뮬레이션 환경.” *한국정보과학회논문지*, 제21권, 제9호, pp. 1652-1664, 1994.
3. 송상복 외 2, “MANET의 전달성능 분석을 위한 NS-2 및 DEVSim++기반 모델링 시뮬레이션 비교연구.” *시뮬레이*

- 선학회 2009 춘계학술대회 논문집, pp. 146-150, 2009.
4. J. S. Pegon and M.W. Subbarao, "Simulation Framework for a Mobile Ad Hoc Network," *Proceedings of OPNETWORK, MIL3*, Sept. 1999.
  5. Ben-El-Kezadri, et al., "Towards MANET Simulators Massive Comparison and Validation," *Indoor and Mobile Radio Communications*, 2007.
  6. Y. Okmura, E. Ohmori, T. Kawano and K. Fukuda, "Field strength and its variability in the VHF and UHF land mobile radio service," *Rev. Elec. Commun. Lab*, 1968.
  7. M. Hata, "Empirical formula for propagation loss in land mobile radio services," *IEEE Transactions on Vehicular Technology*, Vol. 29, pp. 317-325, 1980.
  8. A. G. Longley and P. L. Rice, "Prediction of Tropospheric Radio Transmission Loss Over Irregular Terrain," *ESSA Technical Report*, ERL 79-ITS 67, July 1968.
  9. H. W. Son and N. H. Myung, "A Deterministic Ray Tube Method for Microcellular Wave Propagation Prediction Model," *IEEE Transactions on Antennas and Propagation*, Vol. 47, No. 8, pp. 1344-1350, Aug. 1999.
  10. Raymond J. Luebbers, "A Heuristic UTD Slope Diffraction Coefficient for Rough Lossy Wedges," *IEEE Antennas and Propagation*, Vol. 37, No. 2, Feb. 1989.
  11. Barros F. J. , "The Dynamic Structure Discrete Event System Specification Formalism," *Transactions of the Society for Computer Simulation International*, Vol. 13, No. 1, 1996.



**송 상 복** (sbsong@netcus.com)

2008 인제대학교 정보통신공학과 공학사  
2008~현재 인제대학교 대학원 전자정보통신공학과 석사과정

관심분야 : DEVS, 모델링&시뮬레이션, Ubiquitous Network, MANET



**이 규 호** (kyou@inje.ac.kr)

1980 경북대학교 전자공학과 공학사  
1982 경북대학교 대학원 전자공학과 공학석사  
1998 The University of Gent, Belgium, 정보/컴퓨터공학 공학박사(Ph.D)  
1986~1988 미국 AIT Inc, 연구원  
1983~2004 한국전자통신연구원(ETRI) 책임연구원/팀장  
2005~현재 인제대학교 정보통신공학과 부교수

관심분야 : Variable Structure & Discrete Event System, Embedded System, Ubiquitous Network, 고속 패킷처리기술, 네트워크시스템 기술