

# SPIN을 이용한 무선 이동 센서 네트워크의 배치 알고리즘 검증

오 동 진<sup>†</sup> · 박 재 현<sup>††</sup>

## 요 약

본 논문에서는 최근 많은 연구가 진행되고 있는 무선 센서 네트워크 분야에서 이동 센서 노드의 배치 알고리즘인 DSSA(Distributed Self Spreading Algorithm)와 TBDA(Tree Based Deployment Algorithm)를 모델링 하고, 이들 알고리즘의 안정성과 정확성을 널리 사용되고 있는 모델 검증 도구인 SPIN을 이용하여 검증한다. 그리고 두 알고리즘이 무선 센서 네트워크의 중요사항인 에너지 소비면에서 효율적으로 동작하는지 SPIN 검증 도구를 이용하여 비교분석하고, DSSA에서 발생하는 진동에 대한 보완점을 제시한다.

키워드 : 모델검증, SPIN, 배치 알고리즘

## Verification of Deployment Algorithms in Wireless Mobile Sensor Networks using SPIN

Dong Jin Oh<sup>†</sup> · Jaehyun Park<sup>††</sup>

## ABSTRACT

This paper verifies deployment algorithms in wireless sensor networks using SPIN, a widely used model checking tool. In this paper, two deployment algorithms, DSSA(Distributed Self Spreading Algorithm) and TBDA(Tree Based Deployment Algorithm), are verified to check their stability against oscillation as well as energy consumption that is an important factor in wireless sensor networks.

Key Words : Verification, SPIN, Deployment Algorithm

### 1. 서 론

시스템이 다양해지고 분산화되면서 분산시스템들의 설계 단계에서 시스템의 정형적 명세(Formal specification)를 통한 분석작업의 중요성은 높아지고 있다. 정형기법[1]은 모든 가능한 해석(Interpretation)과 행위(Behavior)를 철저히 검사하기 위해 시스템의 동작 및 요구사항을 정형화된 논리식이나 수학적 방법으로 표현하여 명세하고, 검증(Verification)을 통해 증명함으로써 시스템이 정확하게 동작할 수 있다는 확신을 높일 수 있게 된다. 특히 원자력 발전소나 비행기 제어 시스템 그리고 기차 제어 시스템과 같은 안전에 민감한 시스템의 경우에는 사소한 오류가 매우 큰 손실을 가져올 수 있기 때문에 정확한 시스템의 구현이 가장 중요한 요소로 작용한다.

정형 검증(Formal Verification) 방법에는 여러 가지가 있

는데 그 중 모델 검증(Model Checking)은 수행과정이 자동화되어 있고, 검증 도구를 이용하여 검증한 결과를 분석하고 이해하는 것이 쉬운 장점을 가지고 있다. 예를 들면 결과로 그 모델의 얼마나 많은 상태가 만들어 지는지, 사용하지 않은 코드는 어떤 것이 있는지, 그리고 얼마나 많은 처리(Transaction)가 일어났는지 알 수 있다.

본 논문에서는 최근 주목 받고 있는 무선 센서 네트워크(Wireless Sensor Networks)의 이동 센서 노드(Mobile Sensor Node)를 배치하는 배치 알고리즘을 검증 하여 안정성 및 정확성을 확인한다. 배치 알고리즘은 무선 센서 네트워크의 센서 노드를 일반적으로 무작위 배치하여 통신하는 방식을 취하지 않고 이동 메커니즘을 도입하여 이동 센서 노드로 배치하는 알고리즘으로 전체 네트워크를 송수신 되지 않는 지역이 없이 최대한 커버하고 균일한 배치를 이루으로써 통신의 공정성(Fairness)을 지키고 센싱(Sensing)지역을 균일하게 유지할 수 있는 것을 목적으로 한 것이다. 또한 통신의 공정성을 지킴으로써 에너지 소비를 줄여 전체 네트워크의 수명을 연장시킬 수 있는 이득을 얻게 한다. 또

※ 본 논문은 인하대학교의 지원에 의하여 연구되었음.

† 준 회원 : 인하대학교 정보통신공학부 공학석사

†† 정 회원 : 인하대학교 정보통신공학부 교수

논문접수 : 2006년 2월 2일, 심사완료 : 2006년 4월 11일

한 무선 센서 네트워크에서는 Ad-Hoc 형태로 센서 노드들이 통신 기반 망을 스스로 구성하기 때문에 각각의 센서 노드들은 중요한 역할을 맡게 된다[2]. 만약 하나의 센서 노드라도 소실될 경우 통신 경로를 보장할 수 없으며 네트워크의 수명을 단축시킬 수 있는 요인으로 작용한다. 그래서 이동 센서 노드를 이용한 배치 알고리즘은 무선 센서 네트워크에서 중요한 역할을 하게 된다[10, 11].

본 논문에서 검증할 배치 알고리즘은 DSSA(Distributed Self Spreading Algorithm)와 TBDA(Tree-Based Deployment Algorithm)로써 이것을 모델링(Modeling)하고 이 두 모델이 각각 이상 동작 없이 기대하는 목적에 도달하는지 모델 검증 도구인 SPIN[3, 9]을 이용하여 알고리즘을 검증한다. 그리고 검증 결과를 비교 분석하여 각 알고리즘이 만족해야 하는 중요 사항에 대해서 보장할 수 있는지 검증한다. 그리고 만약 문제점이 있을 경우 개선 보완할 수 있는 대안을 제시한다.

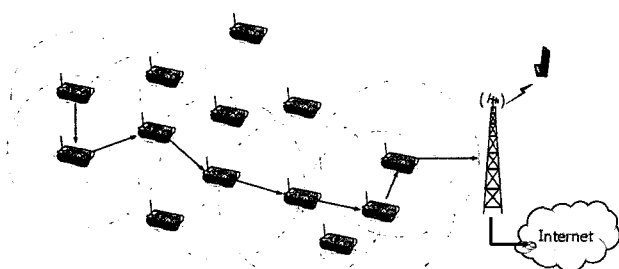
본 논문은 다음과 같이 구성되어 있다. 2장에서는 모델 검증 도구인 SPIN에 대해서 소개하고, 3장에서는 검증 대상인 무선 센서 네트워크의 배치 알고리즘에 대해서 소개한다. 그리고 4장에서는 검증 대상을 모델링하고, 특성을 명세한다. 5장에서는 모델링 한 것을 SPIN을 통하여 검증하고, 결과로 두 배치 알고리즘을 비교 분석한다. 마지막으로 본 논문의 결론을 맺는다.

## 2. 무선 센서 네트워크(Wireless Sensor Networks)의 배치 알고리즘

### 2.1 무선 센서 네트워크 소개

무선 센서 네트워크는 특정지역에 배치되어 그 지역의 환경 감시 및 조사를 목적으로 생긴 네트워크의 한 형태으로써 기존 통신 인프라가 없는 곳에서 각각의 모바일 센서 노드들을 이용하여 Ad-Hoc 형태인 스스로 기반 통신망을 구성하고 수집된 데이터나, 원하는 데이터를 사용자에게 전달하는 특징을 가지고 있다[5]. (그림 1)은 무선 센서 네트워크의 전체적인 구조를 보여준다.

또한 무선 센서 네트워크는 통신 프로토콜에 부합하여 연구가 진행되고 있는데, 그 연구들은 대부분 무선 센서 네트워크 필드에 임의로 무선 센서 노드들이 배치되어 통신하는 것으로 가정되어 진행되고 있다. 이런 경우 실제적으로 사



(그림 1) 무선 센서 네트워크의 구조

용하지 못하는 몇몇 문제들을 안고 시작하게 되는데 그 중 라우팅을 보장할 수 없는 문제와 무선 센서 노드들이 균일하게 배치되지 못해 -센싱 지역과 통신 범위의 중복과 한 쪽으로 치우친 통신이 발생- 균형잡히지(Fairness) 못한 통신이 이루어지게 되는 단점을 가지게 된다. 그래서 네트워크의 수명을 단축시키는 결과를 갖게 된다. 그런 이유로 임의의 배치가 아닌 이동 센서 노드로 배치를 가능하게 하는 배치 알고리즘이 연구되고 있다.

본 논문에서는 DSSA(Distributed Self Spreading Algorithm)와 TBDA(Tree-Based Deployment Algorithm)라는 두 가지의 대표적인 노드 배치 알고리즘을 검증한다. DSSA알고리즘은 임의로 뿌려지는 무선 센서 노드를 이동 로봇을 활용함으로써 자동으로 분산시켜 전체 네트워크 지역을 연결의 부족 없이 각각의 노드들이 송수신이 직결되게 유지하며 최대한 전체 네트워크를 커버하는 것이다[6]. DSSA는 분자의 평형을 이루는 방법에서 아이디어를 얻어 원칙으로 움직이는 중앙 제어기의 어떠한 간섭도 받지 않고 관심있는 지역에 이동 노드를 이용하여 무선 센서 네트워크 필드에 배치하고, 그 지역을 커버할 수 있는 알고리즘이다. 반면에 TBDA 알고리즘은 각 이동 센서 노드를 송수신 범위 내에서 균일하게 배치함으로써 연결의 부족 없이 전체 네트워크의 통신범위를 적절히 유지하여 최대한 전체 네트워크를 커버하는 것이다. 이 알고리즘은 DSSA와 틀리게 임의로 배치된 후가 아닌 초기에 한곳에 모여있다가 스스로 어느 한 지역에 배치되는 것이다. 그리고 이동 센서 노드는 RSSI(Received Signal Strength Indicator)를 이용하여 상대 위치를 알 수 있다고 가정한다. 그래서 각 이동 센서 노드에서 TBDA를 이용하여 배치가 진행된다[7]. 이 알고리즘의 특징은 간단하여 계산 속도를 빠르게 할 수 있는 이점을 가지고 있다. 이것은 무선 센서 네트워크의 센서 노드가 제한된 용량을 가지고 있고, 계산 능력이 부족하기 때문에 무선 센서 네트워크의 센서 노드에 적합한 알고리즘이라고 할 수 있다.

## 3. 검증(Verification)

### 3.1 모델 분석 도구

SPIN은 소프트웨어의 디자인과 비동기 프로세서 시스템의 검증을 제공하는 포괄적인 검증 도구로서 시스템의 표현을 명확히 하기 위해 PROMELA(a PROcess MEta Language)라는 상위 레벨 언어를 제공한다[3]. SPIN 검증 모델은 프로세서 상호작용(Process interactions)의 정확성(Correctness)을 증명하는데 초점을 맞추며, 내부 순차 계산으로 가능한 많은 추상을 시도한다. 프로세서 상호작용은 공유하는 변수를 이용하여 접근하는 버퍼 채널(Buffered Channels)로 비동기 메시지 전송과 rendezvous primitives와 관계한 것을 SPIN으로 명시할 수 있다. 이를 통하여 시스템의 교착상태(Deadlock), 명시되지 않은 반응(Unspecified receptions), 불완전한 플래그(flags incompleteness), 레이스

조건(Race conditions), 그리고 프로세서의 상대적인 속도에 대한 부당한 추정(Unwarranted assumptions)을 알 수 있다.

PROMELA는 SPIN에서 모델을 표현하는 언어이다. PROMELA 프로그램은 프로세서(Processes), 메시지 채널(Message channels) 그리고 변수(Variables)로 구성된다. 프로세서는 전역 객체(Global objects)이다. 메시지 채널과 변수는 프로세서에서 전역적이거나 지역적으로 선언될 수 있다. 프로세서는 행동을 명시하고, 채널(Channels)과 전역 변수(Global variables)는 프로세서가 동작하기 위한 환경으로 정의한다[4]. SPIN은 선형 시제 논리로 표현된 정확성 명세를 받아들인다. 선형 시제 논리 형식은 안정성(Safety)과 필연성(Liveness) 특성을 표현하는데 이용될 수 있다. SPIN에서는 코드내에 상태로써 assertions를 이용하거나, 코드와 분리된 선형 시제 논리를 이용하여 명세를 정의한다. 그래서 위반상태를 확인하고 안정성을 판명하게 된다.

### 3.2 검증

먼저 검증하기 위해 두 배치 알고리즘을 명세화하고 모델링 한다. 그 다음 SPIN 검증도구로 검증하기 위해 PROMELA 언어로 표현한다. 또한 알고리즘이 만족해야할 특성들을 선형 시제 논리로 명세화한 후에 PROMELA 언어로 표현한 것과 선형 시제 논리로 특성 명세한 것을 SPIN 검증도구를 통해 검증한다. 이 때 검증을 통해 알고리즘으로 표현된 명세의 정확성(Correctness), 안정성(Safety), 일관성(Consistency) 및 필연성(Liveness) 등을 알아본다. 안정성은 잘못된 상태전이, 교착상태(Deadlock), 차폐성(Livelock)이 없음을 의미하는 것이다. 교착상태는 한 상태에서 다음 상태로 나가는 천이가 존재하지 않는 경우를 의미하고, 차폐성은 어느 상태들만을 무한히 반복적으로 천이하여 다른 상태로의 천이가 존재하지 않는 것을 의미한다. 필연성은 알고리즘이 만족할 만한 상태로 들어감을 의미한다. 또한 정확성은 부분적으로 정확하며, 만족할 만한 상태로 들어감을 의미한다.

무선 센서 네트워크는 모든 노드들이 협력하여 운영되는 특징을 가지고 있다. 이 배치 알고리즘 또한 모든 노드들이 협력하여 통신하고, 협력하여 배치한다. 이러한 특징들을 가지고 배치 알고리즘이 안정성 및 요구사항의 위반 없이 협력하여 수행할 수 있는지 앞에서 언급한 명세의 특징들을 검증한다.

본 논문은 두 배치 알고리즘에서 이동성에 대한 부분에 초점을 맞추어, DSSA는 알고리즘 전체를 검증하고 TBDA의 경우는 배치되는 부분의 알고리즘만을 검증한다. 그래서 이동 센서 노드의 이동하는 알고리즘이 안정성을 가지고 있는지, 정확성을 가지고 있는지 SPIN 검증 도구로 검증해 보인다. 또한 앞에서 이야기한 요구사항에 적합한지도 검증해 보인다.

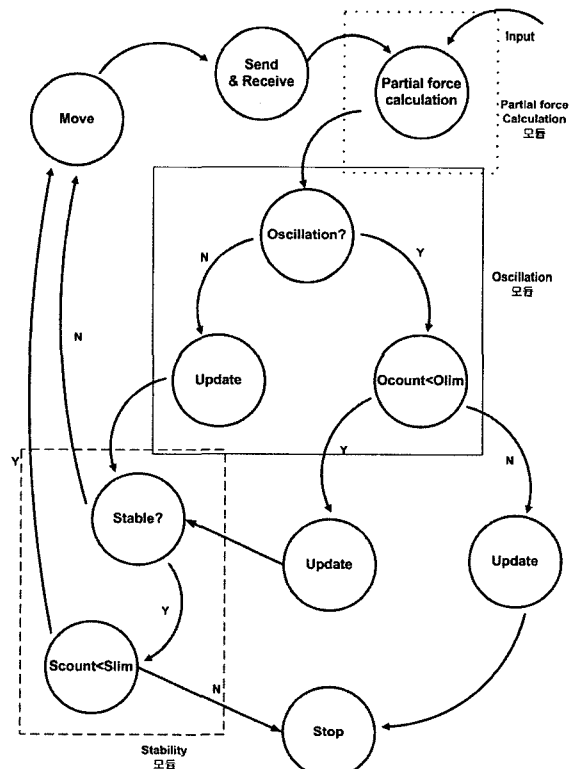
### 3.3 모델링(Modeling)

#### 3.3.1 DSSA

이동 센서 노드는 통신 범위 내에 있는 인근 이동 센서

노드와 통신하여 위치정보를 얻을 수 있다. 그래서 이 위치정보를 이용하여 DSSA를 수행한다. DSSA는 이웃 노드와의 힘을 계산하는 부분 힘의 함수 계산(Partial force calculation), 이동 센서 노드의 진동 검사(Oscillation Check), 그리고 안정된 위치 검사(Stability Check)의 세 모듈로 구성 되어 있다.

부분 힘의 계산 함수 모듈은 배치를 진행하는 동안 노드들의 이동을 정의하는 모듈이다. 이 모듈에서 노드의 다음 위치를 계산하게 되는데, 그것은 인근 노드의 수만큼 인근 노드와의 힘을 계산하고 그 힘의 값을 모두 합해 다음 위치 값을 구하게 된다. 그리고 다음 모듈인 진동 모듈로 값이 전이가 되는데, 그러면 진동 모듈에서는 이 값을 가지고 진동 여부를 판단하게 된다. 진동 여부는 주어진 진동 임계값에 의해 판단하고 true가 나올 경우 진동 제한값과 부분 힘의 계산 함수 모듈에서 사용할 값을 갱신하게 된다. 또한 진동이 발생할 경우 진동을 카운트하여 진동의 제한값을 넘는지 넘지 않는지 확인 후 넘을 경우 이동 센서 노드는 멈추게 된다. 진동의 제한값을 넘지 않을 경우 다음 모듈인 안정된 위치 검사 모듈로 전이가 된다. 이 모듈에서는 이동 센서 노드가 안정화된 위치를 -통신과 센싱을 균일하게 하기 위해 원하는 위치- 판단하는 모듈로 안정된 위치의 임계값 보다 작게 움직이게 되면 안정화 상태에 도달하였음을 판단하여 노드는 배치가 되었음을 나타낸다. 그리고 이동을 멈추게 된다. DSSA를 모델링한 것을 (그림 2)에서 보여준다.



(그림 2) DSSA

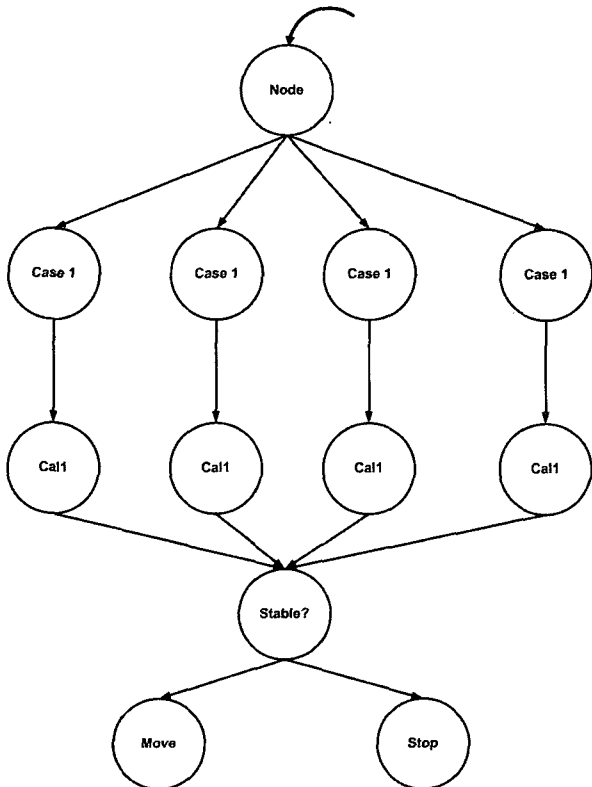
### 3.3.2 TBDA

TBDA는 크게 두 부분으로 나눌 수 있는데 하나는 부모 자식간 연결 형성 프로세서와 다른 하나는 노드간 연결을 유지하며 배치하는 프로세서로 나눌 수 있다. 본 논문에서는 부모 자식간 연결을 형성한 후에 이동 센서 노드가 이동 연결을 유지하며 이동하는 프로세서를 모델링한다.

노드간 이동 연결을 유지하며 이동하는 프로세서 모델은 세 부분으로 나눌 수 있다. 하나는 거리를 유지해야 할 노드를 구분하는 모듈과, 유지해야 할 노드와의 거리와 위치를 계산하고 힘의 함수를 계산하여 다음 위치를 찾아내는 모듈 그리고 노드의 위치가 안정된 위치에 있는가를 판단하는 모듈로 나눌 수 있다.

거리를 유지해야 할 노드를 구분하는 모듈은 부모 자식간 연결을 형성 후 알게된 노드를 판단하여 부모 노드, 형제 노드 그리고 한 홉의 이웃 노드를 구분하여 거리와 위치를 계산할 때 사용하게 된다. 노드 구분 후 다음 모듈로 전이 가 되는데 다음 모듈에서는 판단할 대상 노드의 거리와 위치를 계산하고, 이것을 가지고 힘의 세기와 방향을 구해 이동 노드는 다음 위치를 구하게 된다. 다음 위치값은 다음 모듈로 전이 되고, 이 모듈에서는 안정된 위치에 노드가 위치하였는지를 판단하여 배치가 완료되었거나 움직이지 않아도 되는지를 판단하게 된다.

(그림 3)은 TBDA의 이동 연결을 유지하며 이동하는 프로세서를 모델링한 것이다.



(그림 3) 부모노드에 대한 자식노드의 프로세서

### 3.4 특성 명세(Property specifications)

다음 <표 1>로 두 알고리즘의 특성 명세를 나타내었다. DSSA는 인근 노드의 위치에 의존한다. 그래서 진동 여부를 확인한다. 또한 진동 한계값을 정의해 놓았고 한계값을 넘어서면 안된다. 진동의 한계값을 넘어 계속 동작을 한다면 인근 다른 노드들의 이동 위치에 영향을 미치기 때문에 배치가 진행되는 로컬 지역 네트워크에 문제가 생겨 목적을 달성하지 못하게 된다. 또한 안정된 위치에 도달하면 이동 노드는 이동을 멈추어야 한다. 이동 노드들은 균일한 거리에 위치하여 전체 네트워크를 커버할 수 있어야 한다.

TBDA는 거리 측정을 위해 자신의 노드가 누구와 거리 측정을 하여 위치값을 계산하는지가 중요하다. 만약 TBDA 논문[7]에서 정의한 조건 외의 조건이 발생하여 다음 위치를 계산할 수 없는 상황이 일어나면 다음 수행이 불가능해진다. 또한 이 알고리즘이 안정된 위치에 도달하는지 특성 명세하여 나타내었다.

<표 1> DSSA와 TBDA 특성 명세

	선형 시제 논리	PROMELA 정의
DSSA	[(OSC)	#define OSC (osc_count < osc_lim)
	[(STBT)	#define STBT (stb_count < stb_lim)
TBDA	[(CAST)	#define CAST (myState == leftnode    myState == oneleft    myState == rightnode    myState == othleftnode)
	[(STB)	#define STBT (stb_count < stb_lim)

## 4. 검증 결과 분석

### 4.1 시나리오

우선 각 알고리즘의 기본적인 사항을 SPIN을 이용하여 검증해 본다. 기본적인 사항은 선형 시제 논리로 특성 명세 하지 않고 이 알고리즘이 오류 없이 안정적으로 동작하는지를 검증한다. 그 다음 특성 명세한 것과 알고리즘을 표현한 것을 함께 SPIN 검증도구로 안정성 및 필연성을 검증한다. 그리고 각각의 알고리즘을 다섯 개의 노드에 적용하여 검증해 본다. 이것은 다섯 개의 프로세서를 생성하여 각각의 프로세서의 동작을 확인한다. 입력 값으로 위치 정보가 들어가고 그 정보를 가지고 알고리즘이 추구하는 것을 수행하게 된다. 그리고 두 알고리즘을 검증 도구의 결과로 나타나는 시스템 수행 단계 정보와, 상태의 수, 전환의 값을 비교 분석한다. 이것을 통해 두 알고리즘의 처리량을 비교해 볼 수 있다.

또한 SPIN의 시뮬레이션 도구인 XSPIN을 이용하여 무선 센서 네트워크의 중요사항인 오랫동안 센싱 지역에서 사용되기 위한 에너지 소비측면에서 두 알고리즘을 비교 분석한다[12, 13]. 마지막으로 위치변화에 대한 시뮬레이션을 함으로써 진동여부를 확인한다.

### 4.2 DSSA

검증은 Full statespace 검색의 옵션을 이용하여 부분 검증이 아닌 전체 검증 방식을 이용하였다. 검증 결과는 단독 DSSA의 검증 결과와 이동 노드 프로세서를 다섯 개 생성

<b>(Spin Version 4.2.5 -- 2 April 2005)</b>	
<b>Full statespace search for:</b>	
never claim	+
assertion violations	+ (if within scope of claim)
cycle checks	-(disabled by -DSAFETY)
invalid end states	-(disabled by never claim)
<b>State-vector 24 byte, depth reached 612, errors: 0</b>	
6277 states, stored	
1912 states, matched	
8189 transitions (= stored+matched)	
1893 atomic steps	
hash conflicts: 4 (resolved)	
<b>(Spin Version 4.2.5 -- 2 April 2005)</b>	
<b>Bit statespace search for:</b>	
never claim	+
assertion violations	+ (if within scope of claim)
cycle checks	-(disabled by -DSAFETY)
invalid end states	-(disabled by never claim)
<b>State-vector 44 byte, depth reached 99999, errors: 0</b>	
3.63405e+07 states, stored	
1.49282e+08 states, matched	
1.85622e+08 transitions (= stored+matched)	
3.38007e+07 atomic steps	
hash factor: 1.84667 (best if > 100.)	

(그림 4) DSSA 안정성 검증 결과(상:단독, 하:다섯노드)

하여 나온 결과이다. 다음 (그림 4)의 결과물에 에러가 발생하지 않았기 때문에 이 알고리즘은 안정성이 있음을 확인할 수 있다. 그리고 결과물로 상태 하나의 크기, 상태의 총 수 등을 확인할 수 있다. 검증 결과물로 never claim은 선형 시제 논리로 특성 명세하여 검증하였음을 나타내는 것이며, assertion violations은 명세된 것에 위반이 있는지 여부를 확인하는 것으로 '+'로 명시 되어 명세하였음을 나타내는 것이다.

4.3 TBDA

<b>(Spin Version 4.2.5 -- 2 April 2005)</b>	
<b>Full statespace search for:</b>	
never claim	+
assertion violations	+ (if within scope of claim)
cycle checks	-(disabled by -DSAFETY)
invalid end states	-(disabled by never claim)
<b>State-vector 28 byte, depth reached 137, errors: 0</b>	
69 states, stored	
1 states, matched	
70 transitions (= stored+matched)	
0 atomic steps	
hash conflicts: 0 (resolved)	
<b>(Spin Version 4.2.5 -- 2 April 2005)</b>	
<b>Full statespace search for:</b>	
never claim	+
assertion violations	+ (if within scope of claim)
cycle checks	-(disabled by -DSAFETY)
invalid end states	-(disabled by never claim)
<b>State-vector 44 byte, depth reached 225, errors: 0</b>	
750667 states, stored	
2.63847e+06 states, matched	
3.38914e+06 transitions (= stored+matched)	
0 atomic steps	
hash conflicts: 1.30603e+06 (resolved)	

(그림 5) TBDA 안정성 검증 결과(상: 단독, 하: 다섯노드)

DSSA와 마찬가지로 안정성 여부를 검증한 결과이다. 이것 또한 DSSA와 마찬가지로 검증하였으며, 검증 결과로 단독 TBDA 검증 결과와 다섯개의 이동 노드 프로세서를 생성하여 나온 결과이다. 다음 (그림 5)의 결과로 앞에서 정의한 특성 명세에 만족함을 알 수 있고 알고리즘에 위배되는 사항도 없는 것으로 나타났다.

4.4 비교 분석 및 제안

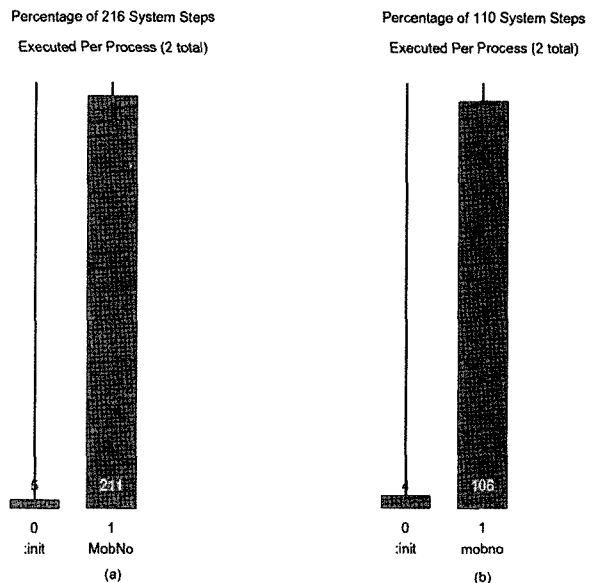
두 알고리즘은 사용 용도가 같으나 쓰이는 상황이 약간 차이가 있다. 하지만 실제적으로 동작하려고 하는 목적은 같다. 다음 (그림 6)은 SPIN의 XSPIN[8]을 이용한 시뮬레이션 결과로 각 알고리즘에서 하나의 노드만을 생성하여 시스템의 스텝을 나타낸 것이다.

(그림 6)에서 나타난 것은 한 노드가 수행한 스텝을 보여주는 것으로 이동 노드 하나가 수치만큼의 작업을 수행하게 되는 것이다. DSSA가 TBDA보다 약 두 배 가량 많음을 알 수 있다.

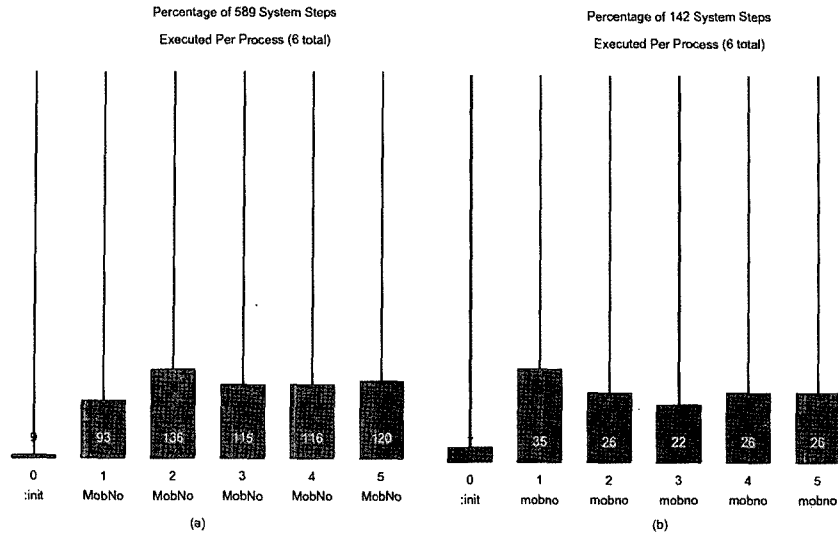
또한 다음 (그림 7)은 다섯 노드를 시뮬레이션한 결과이다.

다섯 노드를 이용할 경우, 전체 시스템 스텝이 DSSA가 월등히 높음을 알 수 있다. 이것은 DSSA가 TBDA보다 작업량이 많다는 것을 보여준다. 이것을 무선 센서 네트워크 필드에 적용할 경우 DSSA는 수행하는 작업의 양이 많아 에너지 소비가 TBDA 보다 높을 것이고 그 만큼 배치되는 시간 또한 오래 걸릴 것이다. 또한 노드 수가 더 늘어날 경우 더 많은 작업을 수행하게 된다.

다음으로 두 알고리즘을 검증 결과물을 이용하여 비교해 본다. 검증은 부분 명령(Partial Order Reduction)을 사용하지 않고 자세하게 검증할 수 있도록 '-DNOREDUCE' 옵션을 설정하고, 기본값으로 설정되어 있는 full statespace search를 사용하지 않고 Bitstate search 옵션을 설정하여 검증하였다. 그리고 특성 명세는 앞에서 안정성을 확인하였



(그림 6) 한 노드의 시스템 스텝 : (a) DSSA (b) TBDA



(그림 7) 다섯 노드의 시스템 스텝 : (a) DSSA (b) TBDA

기 때문에 여기에서는 고려하지 않고 검증하였다.

<표 2>와 <표 3>은 두 알고리즘을 이동 노드의 수를 증가시켜 각각에 대한 검증 결과를 나타내어 비교해 보았다. State-vector는 시스템을 표현하기 위해 필요한 메모리 용량을 나타내는 것으로 한 상태당 1 byte이다. 그리고 Depth reached는 트리의 루트로부터 가장 긴 깊이 우선 검색 경로를 나타내는 것이다. SPIN은 기본적으로 깊이 우선 검색 방법을 사용하여 상태를 검색한다.

<표 2>에서 두 알고리즘은 노드 수가 적을 경우 상태 벡터 값은 큰 차이를 보이지 않지만 4번째의 경우 큰 차이를 보임을 알 수 있다. depth reached의 경우는 DSSA가 큼을 알 수 있다. 이것은 TBDA에 비해 더 많은 경우의 수가 나타날 수 있음을 보여주는 것이다.

<표 2> 검증 결과 비교 - I

	노드수	DSSA		TBDA	
		State-vector	Depth reached	State-vector	Depth reached
1	5	40	1203	40	88
2	7	48	1021	48	108
3	15	44	5230	68	188
4	30	104	5609	56	338

<표 3> 검증 결과 비교 - II

	노드수	DSSA			TBDA		
		States stored	States matched	transitions	States stored	States matched	transitions
1	5	4.6e+06	1.7e+07	2.1e+07	981721	3.9e+04	4.9e+06
2	7	4.7e+06	1.8e+07	2.2e+07	4.6e+06	2.2e+07	2.7e+07
3	15	4.6e+06	2.1e+07	2.5e+07	4.9e+06	2.4e+07	2.8e+07
4	30	4.8e+06	2.2e+07	2.7e+07	4.8e+06	2.3e+07	2.7e+07

<표 3> 또한 두 알고리즘의 검증결과를 나타내는 것이다. 이 표에서는 알고리즘의 상태가 얼마나 많이 생기는지 나타내었고, 변이(Transition)가 얼마나 많이 일어나는지 나타내었다. 변이는 검증을 완전하게 수행하였을 때 작업한 통계로 검색에서 조사된 변이의 수이다. 상태의 경우 노드의 수가 적을 경우에는 TBDA가 DSSA보다 적게 생성이 되는데 노드 수가 증가함에 따라 TBDA와 DSSA는 같아지고 있음을 알 수 있다.

다시 알고리즘에서 논의하면 두 알고리즘은 인근 이동 노드와 힘의 영향을 이용하여 멀어지거나 가까워지는 역할을 수행하게 된다. 이것은 노드 수가 많아지면 상태가 증가하게 됨을 알 수 있다. DSSA의 경우 인근 노드 모두와 힘의 합수를 계산하고, TBDA의 경우 링크를 형성한 노드와 한 홉 이웃 노드와 힘의 합수를 계산한다. 그렇기 때문에 둘은 차이를 보일 수 있고, 상태는 DSSA가 증가함을 알 수 있다. 그리고 <표 2>에서 나타났듯이 DSSA는 depth reached가 월등히 높음을 알 수 있다. 그 만큼 많은 조건들이 생성될 수 있게 된다.

그러나 두 알고리즘 모두 노드 수가 증가함에 따라 상태가 커지고 같아지고 있음을 알 수 있다. 이것은 무선 센서 네트워크에서 요구하는 사항 중의 하나인 수 많은 노드를 효율적으로 사용해야 하는 것에 문제가 될 수 있다. 두 알고리즘 모두 풀어야 할 연구로 많은 이동 노드를 가지고 수행할 경우를 위해 알고리즘의 최적화 연구가 계속 진행되어야 할 것이다.

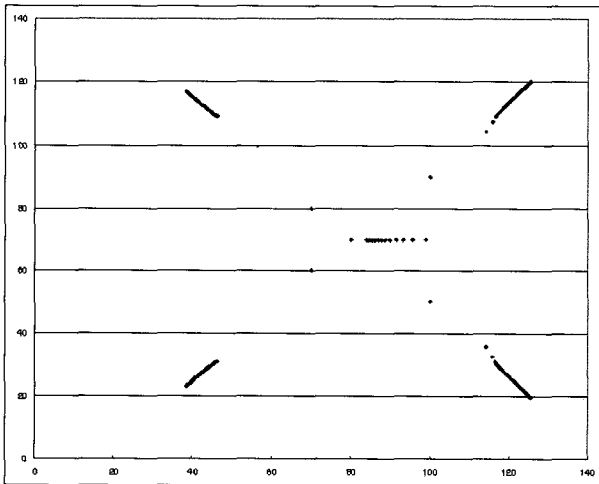
여기서 우리는 SPIN 검증도구로부터 나온 각 알고리즘의 pan.c를 이용하여 각 알고리즘에서 계산되는 위치값을 확인할 수 있게 수정하여 다음과 같이 각각 (그림 8)과 (그림 9)에 데이터를 산출하였다. (그림 8)은 DSSA를 이용한 것으로 진동 여부 및 안정된 위치에 노드들이 배치될 수 있는 지 값을 확인한다.

DSSA의 경우 가운데 있는 노드가 주변 여러 노드의 영

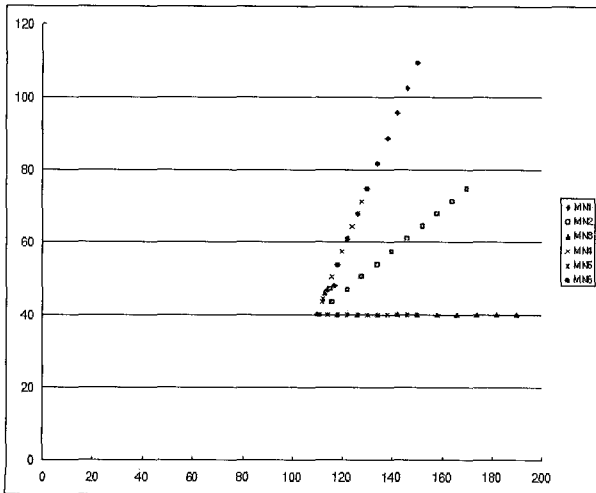
향으로 약간의 진동이 발생하는 것을 알 수 있고, 멀리 갔다가 다시 원래 있던 자리로 돌아오는 현상이 발생함을 알았다. 이것을 방지하기 위해서는 둘러싸여 있는 노드임을 확인하기 위해 힘의 함수를 계산하는데 필요한 인근 밀도인자를 계산한 후 인근 밀도의 임계값과 인근 밀도를 비교한다. 그리고 다음 단계인 힘의 함수를 계산할 수 있도록 하는 상태를 추가한다. 그래서 자기 자신이 둘러싸여 있음을 판단할 수 있게 하여 대기상태로 머물러 있을 것인지 아니면 움직일 것인지 판단하도록 한다. 만약 대기상태로 있을 경우 인근 노드가 일정 거리로 떨어지게 되는 것을 확인한 후 대기상태에서 풀려 다시 배치 알고리즘을 수행할 수 있도록 한다.

다음으로 TBDA로 얻은 데이터는 (그림 9)에 나타내었다.

TBDA는 한 곳에 모여있다 배치가 이루어지게 되는데 루트 노드 외에 모든 노드들이 기대되는 위치로 배치됨을 볼 수 있다. 이 알고리즘은 트리 계층 구조를 이루고 있기 때문에 각 이동 노드는 인근 노드가 아닌 연결 형성이 이루



(그림 8) DSSA를 이용한 노드들의 위치 변화



(그림 9) TBDA를 이용한 노드들의 위치 변화

어진 노드들만을 가지고 조건에 맞추어 거리측정을 하기 때문에 진동의 여부는 찾아볼 수 없었다. 검증 결과에서 비교한 것과 시뮬레이션한 결과 이동 노드가 다섯 개일 경우 TBDA가 DSSA에 비해 이동 노드가 배치하는 과정에 효율성이 있음을 알 수 있다. 또한 무선 센서 네트워크에서 에너지 문제를 해결해 줄 수 있을 거라고 예상한다. 또한 두 알고리즘은 모두 전체 네트워크를 커버할 수 있는 위치로 수렴해 가는 것을 알 수 있다. 그러므로 실제적으로 적용하게 되면 이동 노드들은 기대되는 위치에 배치될 수 있을 것이다.

### 5. 결론 및 향후 과제

본 논문은 모델 검증 도구인 SPIN을 이용하여 무선 센서 네트워크에서 이동 노드의 배치를 위한 DSSA와 TBDA의 알고리즘을 검증하였다.

두 알고리즘 모두 만족하는 결과를 보였다. 하지만 DSSA의 검증 결과에서 나타났듯이 많은 상태들이 일어남을 알 수 있었다. 이것은 노드의 수가 증가함에 따라 많은 상태가 생기는 것으로, 더 많은 처리 작업이 생기는 것을 확인할 수 있었다. 그러므로, 이 알고리즘을 무선 센서 네트워크 필드에 많은 수의 노드로 실제 적용할 경우 검증 결과에 미루어 전체 네트워크상의 이동하는 부분에서 발생하는 작업이 많아, 부하로 작용할 수 있음을 알 수 있다. 또한 TBDA의 경우 알고리즘의 동작 여부와 노드의 수에 대한 상태 변화에서 노드 수가 증가할 때 따라 처리하는 양이 증가하여 많은 이동 센서 노드를 이용할 경우 효율성 면에서 떨어질 수 있음을 보였다.

### 참고 문헌

- [1] J. Chaves, "Formal Methods at AT&T, An Industrial Usage Reprt," Proc. Fourth FORTE Conf. Formal Description Techniques, pp.83-90, Sydney, Australia, 1991.
- [2] S. Tilak, N. Abu-Ghazaleh and W. Heinzelman, "A Taxonomy of Wireless Micro-sensor Network Models," ACM SIGMOBILE Mobile Comp. Commun. Rev., Vol.6, No.2, pp.28-36, Apr., 2002.
- [3] G. J. Holzmann, "The Model Checker SPIN," IEEE Transactions on Software Engineering, Vol.23, No.5, pp.279-295, May, 1997.
- [4] I. Barland, "PROMELA AND SPIN REFERENCE", Version 1.3, Aug., 2004.
- [5] I. Akyildiz et al., "A Survey on Sensor Networks," IEEE Commun. Mag., Vol.40, No.8, pp.102-114, Aug., 2002.
- [6] N. Heo and P.K. Varshney, "A Distributed Self Spreading Algorithm for Mobile Wireless Sensor Networks," Proc Wireless Communications and Networking, 2003. (WCNC 2003), 2003 IEEE Vol.3, pp.1597-1602, Mar., 2003.
- [7] 문종천, "이동 센서 네트워크에서의 트리 기반 배치 알고리즘에 관한 연구", 인하대학교 대학원 석사학위논문, 2006. 2.
- [8] <http://spinroot.com/spin/whatispin.html>

- [9] 방기석, 최진영, "Gerald J. Holzmann의 "The Model Checker SPIN"에 대하여", 정보과학논문지(B) 제25권 제11호, pp.1648-1656, 1998.
- [10] B. Boigelot and P. Godefroid, "Model Checking in Practice: An Analysis of the ACCESS Bus Protocol Using SPIN," Proc. Formal Methods Europe (FME96), Oxford, England, Lecture Notes in Computer Science 1, 051, pp.465-478. Springer-Verlag, Mar., 1996.
- [11] C-T. Chou and D. Peled, "Verifying a Model-Checking Algorithm," Proc. Tools and Algorithms for the Construction and Analysis of Systems (TACAS96), Passau, Germany, Lecture Notes in Computer Science 1,055, pp.241-257. Springer-Verlag, May, 1996.
- [12] W. Heinzelman, A. Chandrakasan and H. Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks," Proc. 33rd Hawaii Int'l. Conf. Sys. Sci., Jan., 2000.
- [13] F. Ye et al., "A Scalable Solution to Minimum Cost Forwarding in Large Sensor Networks," Proc. 10th Int'l. Conf. Comp. Commun. and Networks, pp.304-309, 2001.



### 오 동 진

e-mail : djoh@emcl.inha.ac.kr

2003년 인하대학교 자동화공학과(학사)

2006년 인하대학교 정보통신공학과

(공학석사)

2006년~현재 (주)아이콘트롤스 연구원

관심분야: 실시간 시스템, 임베디드 시스템,

무선 센서 네트워크, 검증 등



### 박 재 현

e-mail : jhyun@inha.ac.kr

1994년 서울대학교 제어계측공학과(박사)

1995년~현재 인하대학교 정보통신공학과

교수

관심분야: 임베디드 시스템, 실시간 시스

템, 컴퓨터네트워크, 무선 센서

네트워크