

# 무선 센서 네트워크에서 에너지 효율 향상을 위한 이동 에이전트 기반 데이터 수집 알고리즘

양명준\*, 김진혁\*, 최상방<sup>o</sup>

## A Novel Data Collection Algorithm Based on Mobile Agent to Improve Energy Efficiency in Wireless Sensor Networks

Myungjoon Yang\*, Jinhyuk Kim\*, Sangbang Choi<sup>o</sup>

### 요약

무선 센서 네트워크는 정보를 수집하고자 하는 구역에 배포된 센서 노드를 이용한 네트워크이다. 센서 노드들은 배터리 교체가 어렵기 때문에 한정된 에너지로 동작하기 위해서는 에너지 효율과 전송 효율이 중요하다. 본 논문에서는 무선 센서 네트워크에서 노드 간의 거리에 따라 클러스터를 구성한 뒤 이동 에이전트를 이용하여 효율적으로 데이터를 수집하는 알고리즘을 제안한다. 제안하는 알고리즘은 가능한 노드간의 전송 충돌을 줄이고 최저 지연 시간을 보장하기 위하여 거리에 기반한 토폴로지를 구성한 후, 이를 이용하여 클러스터를 구성한다. 알고리즘의 성능 비교를 위하여 기존의 이동 에이전트를 이용한 데이터 수집 알고리즘인 GCF, LCF, TBID와 비교한 결과, 노드가 300개 일 때 기존의 알고리즘보다 네트워크 수명이 각각 194배, 124.6배, 1.46배 향상되었으며 데이터 병합 시 각각 87.5%, 85%, 45%의 에너지 효율 상승을 보여주었다.

**Key Words** : WSN, Mobile Agent, cluster, energy efficiency, multi-agent systems

### ABSTRACT

Wireless Sensor Network(WSN) is a network which consists of sensor nodes(SNs) in data collection area. It is hard for the SNs to replace battery. Thus energy and transfer efficiency is important because the energy is limited. In this paper, we propose energy-efficient data collection algorithm for wireless sensor networks by using mobile agent in distance-based cluster structure. For avoid collisions and guarantee low latency, make distance-based topology and build cluster by the topology. For performance comparison of the proposed algorithm, compare with existing mobile agent algorithm. When network constructed by 300 nodes, proposed algorithm has performance increase than existing algorithm(GCF, LCF, TBID) in network lifetime 194, 124.6, 1.46 times each and data merging energy efficiency 87.5%, 85%, 45% each.

※ 이 논문은 2010년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(2010-0020163).

♦ First Author : Inha University Department of Electronic Engineering, ymj0726@inha.edu, 학생회원

o Corresponding Author : Inha University Department of Electronic Engineering, sangbang@inha.ac.kr, 중신회원

\* Inha University Department of Electronic Engineering, graybaily@inha.edu

논문번호 : KICS2013-12-542, Received December 18, 2013; Revised July 1, 2014; Accepted August 4, 2014

## I. 서 론

무선 센서 네트워크는 정보를 수집하고자 하는 구역에 센서 노드를 배포한 후 각 노드가 정보를 수집한 뒤 구역의 정보를 종합하는 장소로 전송하는 형태의 네트워크이다. 정보를 수집하기 위해 무선 센서 네트워크에서 가장 일반적으로 사용되는 모델은 클라이언트/서버 모델로서 각 센서 노드는 데이터를 수집하여 서버의 역할을 하는 노드로 수집한 데이터를 전송한다. 클라이언트/서버 모델의 문제점은 센서 노드의 개수가 늘어나면 데이터 수집 과정에서 서버 역할을 하는 노드의 부담이 증가하게 되고 결과적으로 네트워크의 에너지 효율에 영향을 미치게 된다는 점이다. 이러한 문제점을 해결하기 위하여 클라이언트/서버 모델에서의 데이터 병합 기법이 제안되었으나<sup>1)</sup> 근본적인 해결책을 제시하지는 못하였고 이러한 문제점을 해결하기 위하여 이동 에이전트를 이용하는 방법이 제안되었다<sup>2)</sup>.

이동 에이전트는 소프트웨어 코드로 이루어져 있는 특성을 가지고 있으며 이를 이용하여 각 노드에 수정하고자 하는 코드를 이동 에이전트에 포함하여 전달할 수 있다. 이러한 이동 에이전트의 특성을 이용하여 센서 노드들이 Processing Element(PE)가 원하는 형태의 데이터 수집을 수행할 수 있도록 센서 노드의 정보를 효율적으로 재설정 하는 방법이 제안되었고<sup>3)</sup> 노드를 효율적으로 재설정하기 위하여 이동 에이전트를 무선 센서 네트워크에서의 미들웨어로 사용하기 위한 연구도 진행되었다<sup>4-6)</sup>.

또한, 이동 에이전트를 사용할 때, 이동 에이전트는 각 노드를 순차적으로 방문해야 하므로 에이전트의 경로를 결정하는 것은 전체 에너지 소모량과 병합 비용에 매우 중요한 영향을 미친다. 일정을 잘못 결정하게 된다면 비효율적인 순서로 각 노드를 방문하게 되므로 네트워크 전체의 성능이 저하되며 이러한 문제를 해결하기 위하여 이동 에이전트의 효율적인 경로를 정하기 위한 몇 가지 연구가 제안되었다<sup>7-10)</sup>. 그러나 이러한 알고리즘은 네트워크의 규모를 고려하지 않고 하나의 이동 에이전트를 이용하므로 네트워크의 규모가 커진다면 데이터 병합 및 전송을 위한 에너지 소모량이 커진다는 문제점을 가지고 있다. 이러한 문제를 해결하기 위해 네트워크 토폴로지 내에 복수의 트리를 구성하여 각각의 트리에 이동 에이전트를 할당, 데이터를 수집하는 방법이 제안되었다<sup>11)</sup>.

이동 에이전트 모델은 상기한 바와 같이 무선 센서 네트워크 환경에서 이동 에이전트의 효율적인 일정

수립 및 네트워크 규모에 따른 적절한 로드 분산이 보충된다면 데이터 수집 및 병합 시 기존의 클라이언트/서버 모델에 비해 에너지 효율 면에서 장점을 보인다.

본 논문에서는 기존의 이동 에이전트를 이용한 데이터를 수집하는 알고리즘에서 네트워크의 규모가 커질 때 발생하는 이동 에이전트의 데이터 병합 및 전송 비용 증가를 막기 위한 방법으로 노드 간의 거리에 따라 클러스터를 구성하고 이를 이용해 이동 에이전트를 배포함으로써 센서 노드의 데이터 병합 및 전송 에너지 효율을 향상시키는 알고리즘을 제안한다.

## II. 관련 연구

이동 에이전트는 1950년대에 제안되었으며, 사용자가 원하는 작업을 자동으로 수행하는 소프트웨어로서 인공지능 분야에서 연구되어왔으나 1980년대 들어서는 분산 컴퓨팅 시스템(Distributed Computing System)에서 에이전트 간 통신(Inter-agent Communication) 개념이 도입되면서 인공지능과 분리되어 독자적으로 연구되기 시작하였다. 이동 에이전트의 원리는 이동 에이전트라 불리는 소프트웨어 코드를 시스템에 전송한 뒤 각 시스템에서 수집 된 데이터를 중복을 고려하여 최적화한 후 다음 시스템으로 전송하는 과정을 거쳐 최종적으로 처음 시스템으로 돌아가서 수집한 데이터를 전송하는 것이며 이를 기반으로 한 분산 컴퓨팅 시스템의 가장 중요한 목적은 분산 시스템 간에 데이터를 주고받을 때에 가능한 한 낮은 작업 수행 시간을 유지하면서 네트워크 트래픽의 크기를 최대한 작게 하는 것이다<sup>12)</sup>.

이를 기반으로 무선 센서 네트워크 환경에서 제안된 이동 에이전트 기반의 프로토콜은 기존의 분산 센서 네트워크에서 각각의 센서 노드들이 싱크 노드로 각자 데이터를 전송하는 것과는 다르게 이동 에이전트를 센서 노드로 정해진 일정 또는 규칙에 따라 순차적으로 전송하고 각 노드에서 수집 된 데이터를 이동 에이전트를 이용하여 최적화 함으로써 효율적인 데이터 수집을 할 수 있게 한다. 그림 1은 기존의 분산 센서 네트워크와 이동 에이전트 기반의 무선 센서 네트워크의 구조적 차이점을 보여주고 있다.

이동 에이전트는 방문 하여야 하는 노드들을 순서대로 방문 하고, 각 노드에서 수집한 데이터를 최적화한 뒤 최적화된 데이터를 가지고 PE로 돌아가 수집한 데이터를 전달한다. 클라이언트/서버 모델과 이동 에이전트의 성능비교는 이미 기존 연구<sup>12,13)</sup>에서 분석 및 시뮬레이션으로 확인 되었다. 하지만 분산 시스템

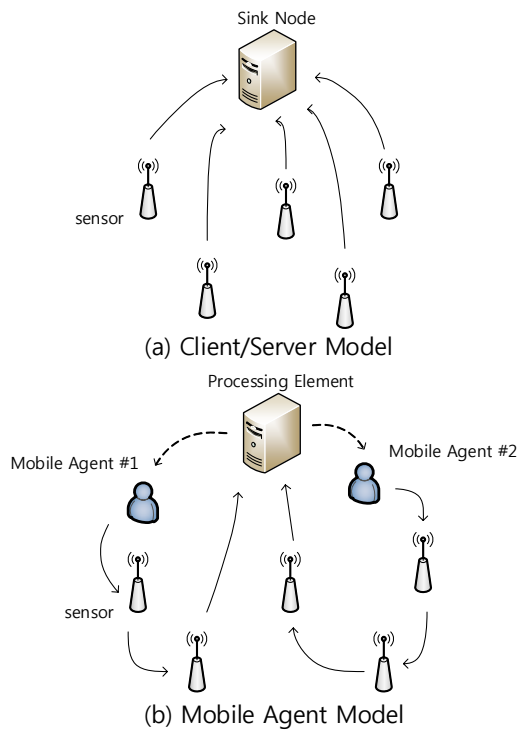


그림 1. 클라이언트/서버 모델과 이동 에이전트 기반의 무선 센서 네트워크의 구조 비교  
 Fig. 1. Architecture comparison between Client/Server model and Mobile Agent-based Wireless Sensor Network

에서의 이동 에이전트의 잠재성에도 불구하고, 이동 에이전트가 부적절하게 사용된다면 비효율적인 설계로 이어질 수 있다. 예를 들면, 데이터 수집 과정에서, 하나의 이동 에이전트를 이용하여 모든 노드를 순차적으로 순회하며 데이터를 수집한다면, 경우에 따라 기존의 클라이언트/서버 모델보다 더 큰 네트워크 과부하와 데이터 수집 시간을 요구할 수 있으며 이는 결과적으로 각 노드가 에너지를 비효율적으로 소비하게 하는 원인이 된다.

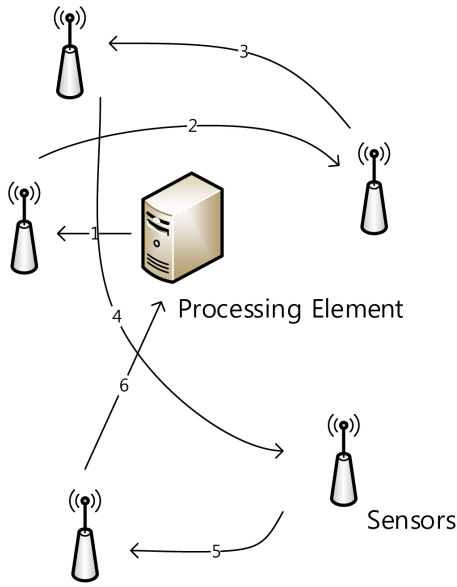
이러한 네트워크 확장에 따른 문제를 해결하기 위하여 기존의 연구<sup>13,14)</sup>들은 논리적/물리적인 구역에 따른 클러스터를 생성하고 각각의 클러스터에 이동 에이전트를 할당하여 관리하는 것으로 해결하였으나 클러스터링뿐만 아니라 이동 에이전트의 이동 경로의 최적화 또한 시스템의 효율성 향상을 위해 반드시 필요하다. 무선 센서 네트워크에서 이동 에이전트의 이동 경로는 경로 결정이 이루어지는 장소에 따라서 동적 혹은 정적으로 분류할 수 있으며, 정적 방식은 PE에서 이동 에이전트의 이동 경로를 미리 지정한 뒤 지정한 경로에 따라서 각 노드를 이동하는 것이고<sup>9,10)</sup>

동적 방식은 각 노드에서 이동할 다음 노드를 지정하여 이동하는 방식이다<sup>7-8)</sup>. 정적 방식은 무선 센서 네트워크 내의 노드의 구성을 사전에 파악할 수 있고 데이터를 주기적으로 수집하여 PE로 보내져야 하는 경우에 효율적이며, 동적 방식은 감시 환경이 변동하는 등 사전에 환경을 파악할 수 없는 경우에 효율적이다. 동적 방식은 이동 에이전트가 이동 중 만날 수 있는 손상된 링크 등의 네트워크 내부에서 발생할 수 있는 장애가 발견될 시 즉석에서 경로를 변경할 수 있으며, 무선 센서 네트워크에서 에너지가 고갈된 센서 노드가 생길 때에도 별도의 네트워크 재구성 과정 없이 새로운 노드로 교체하는 것만으로 네트워크를 유지할 수 있다는 장점이 있다. 그러나 동적 방식은 정적 방식보다 다음 노드를 찾는 과정에서 정적 방식과 비교하면 추가적인 시간을 소모하는 단점이 있으며 다음 노드를 찾는 연산 과정에서 센서 노드의 에너지를 더 많이 소모하고 이러한 연산을 위해 이동 에이전트의 크기가 더 증가하는 문제가 있다. 또한, 방문하는 다음 센서 노드를 현재 위치에서의 정보 기반으로 정확하게 되는 동적 방식의 이동 에이전트는 추가적인 라우팅 정보 없이 PE로 돌아가는 경로를 찾기 힘들다. 이에 비하여 PE에서 정한 경로를 움직이는 정적 방식에서는 추가적인 라우팅 정보는 필요하지 않다.

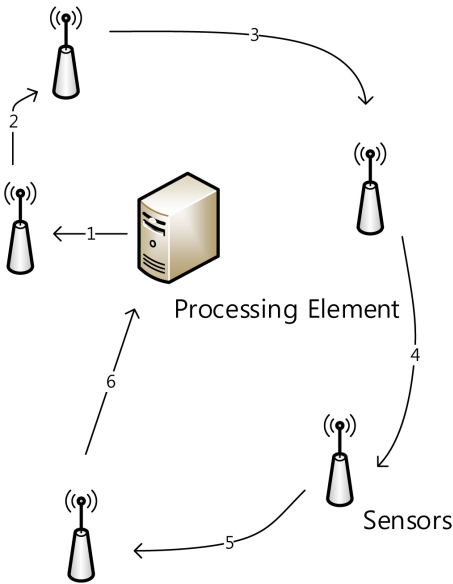
무선 센서 네트워크에서 이동 에이전트를 사용하는 연구들의 최근 경향은 이동 에이전트의 일정을 정적으로 결정하는 방법에 집중되어 있으며<sup>9)</sup> Global Closest First(GCF)와 Local Closest First(LCF)라 불리는 두 개의 알고리즘이 대표적이다. GCF는 무선 센서 네트워크에서 PE로부터 가까운 노드를 순차적으로 방문하는 방식이고 LCF는 현재 이동 에이전트가 존재하는 노드에서 가까운 노드로 순차적으로 이동하는 방식이다. 그림 2는 무선 센서 네트워크 하에서 이동 에이전트의 경로를 계획할 때 GCF 알고리즘과 LCF 알고리즘의 차이를 보여준다.

LCF는 출발한 위치에서 가장 가까운 노드를 방문하고 해당 노드에서 가장 가까운 노드를 방문하기 때문에, 이동 에이전트의 시작 위치가 결과물에 큰 영향을 미친다. 최종적으로 도착한 노드가 PE와 멀리 떨어져 있다면 수집된 데이터를 PE로 보내기 위해 많은 비용이 소비될 것이기 때문이다. GCF는 노드의 배치가 이동 에이전트의 시작점을 중심으로 진동하는 형태로 배치되어 있다면 이동 에이전트의 경로가 LCF보다 비효율적이고 긴 형태로 만들어지게 되어 결과적으로 LCF보다 낮은 성능을 보여주게 된다.

또한 GCF, LCF는 지역 내의 노드들의 물리적 위



(a) Global Closest First



(b) Local Closest First

그림 2. 무선 센서 네트워크 내에서 GCF 알고리즘과 LCF 알고리즘의 이동 에이전트 일정 계획 비교  
 Fig. 2. Comparison between GCF algorithm and LCF algorithm for Mobile Agent itinerary planning

치를 고려하지 않고 PE에서 하나의 이동 에이전트가 출발하여 모든 노드를 방문하고 돌아오는 것을 전제하고 있기 때문에 네트워크의 규모가 커지면 이동 에이전트가 각 노드를 방문하고 돌아오는 데 걸리는 시

간이 증가하며, 이동 에이전트가 수집해 오는 데이터의 크기가 커지기 때문에 전송에 드는 비용이 커지게 되므로 네트워크의 에너지 고갈이 빨라지기 때문에 알고리즘의 성능은 무선 센서 네트워크의 크기에 반비례 하여 저하된다.

상기한 알고리즘이 네트워크 내의 모든 노드를 하나의 이동 에이전트를 이용하여 순차적으로 방문하기 때문에 발생하는 성능저하를 개선하기 위해 Tree-Based Itinerary Design(TBID)은 트리 구조의 토폴로지를 구성한 뒤 각 트리에 배정된 복수의 이동 에이전트를 사용하여 데이터를 수집하는 알고리즘을 제안한다<sup>[11]</sup>. TBID는 각 노드의 위치를 노드들의 최대 전송 전력  $r_{max}$  를 사용해 구분한 구역으로 판단한다. 트리를 결정하는 단계에서 각 노드를 연결하는 에지는 Performance Cost(PC)로 정의되는 값을 가지며 PC는 이동 에이전트가 해당 에지를 경유하는 시점까지의 일정으로 수집된 누적된 데이터의 크기를 가지고 트리를 구성할 때 이를 이용하여 각 트리를 가능한 비용 균등하게 구성한다. 트리를 구성한 후, PE는 각 트리에 대하여 이동 에이전트의 일정을 결정하며 이 과정에서 계산의 복잡도를 줄이기 위하여 보조 이진 트리를 이용한다. PE는 일정을 계산한 후 구성된 각 트리에 배정된 이동 에이전트를 전송하며 이동 에이전트는 각 트리의 노드를 후위 순회 방식으로 순회하며 데이터를 수집한 뒤 PE로 돌아간다. 이 방법은 복수의 이동 에이전트를 이용하기 때문에 네트워크 확장에 따른 데이터 수집 및 병합 시간에 있어서 기존의 알고리즘 보다 향상된 성능을 보인다. 또한 일정이 정해지면 토폴로지가 변경될 때 까지는 일정이 변하지 않기 때문에 첫 데이터 수집 과정에서 각 노드에 다음 일정에 관한 정보를 저장할 수 있으므로 다음 수집 때부터는 일정을 포함하지 않고 이동 에이전트가 이동하는 것이 가능하다는 장점이 있다. 그러나 이 방법은 네트워크의 토폴로지가 변경되면 네트워크 내의 모든 노드가 PC 계산을 다시 하여야하기 때문에 토폴로지의 변경에 취약하며 후위 순회 방식으로 데이터를 수집할 때 다른 부트리보다 규모가 큰 부트리가 존재한다면 전체 네트워크의 데이터 전송 효율이 떨어진다라는 문제점이 있다. 그림 3은 TBID의 트리 구성 및 이동 에이전트의 일정에 대한 예제이다. 각 실선은 PC에 의해 만들어진 트리 구조이며, 화살표는 트리에서 이동 에이전트가 방문하는 순서이다. PE는 이동 에이전트가 후위 순회 방식으로 트리 내의 노드를 방문하도록 일정을 결정한다.

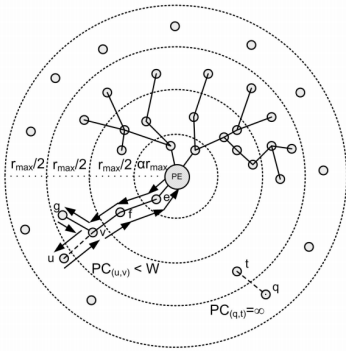


그림 3. TBID의 트리 구성 및 이동 에이전트의 일정 예  
 Fig. 3. TBID's tree build and mobile agent itineraries planning example

III. 알고리즘 제안

기존의 이동 에이전트 기반의 알고리즘은 모두 이동 에이전트가 이동할 때, 하나의 노드에서 다음 노드로 순차적으로 이동하는 연결 리스트 방식으로 이동하였다. 이 방법은 무선 센서 네트워크에서 데이터 전송에 따르는 충돌을 줄일 수 있지만, 이동 에이전트가 각 노드를 일일이 방문하여야 하기 때문에 이동에 걸리는 시간이 커지므로 데이터의 정확도가 떨어지고 나중에 도달하는 노드일수록 누락된 데이터양이 커지기 때문에 네트워크의 규모가 커질수록 일정상 마지막 노드의 에너지 효율이 저하되는 단점이 있다.

본 논문에서는 충돌을 회피하고 최소 전송 시간을 보장하면서 이동 에이전트를 전송하기 위해 홉 카운트와 노드 간의 거리를 전송 기준으로 삼고 이를 이용하여 토폴로지를 구성한 후, 구성한 토폴로지를 한 홉 크기의 클러스터로 구분하여 생성하는 정책에 대하여 제안한다.

토폴로지 초기화 단계는 효율적인 토폴로지 구성을 위하여 부모 노드 선정 및 브로드캐스트 지연 시간 계산 단계와 클러스터 구성 단계, 두 단계로 이루어진다. 모든 노드는 첫 단계에서 자신의 수신하는 메시지 및 부모 노드 선정 규칙에 따라 부모 노드를 선정하며 부모 노드를 선택한 노드는 토폴로지를 확장하기 위하여 자식 노드를 받기 위한 메시지를 브로드캐스트한다. 이러한 과정을 거쳐 모든 노드가 부모와 자식 노드를 가지는 구조를 생성한 뒤 두 번째 단계에서는 이를 이용하여 클러스터를 구성한다. 자식 노드가 존재하지 않는 모든 노드는 부모 노드를 클러스터 헤드로 가지며 만일 자식 노드를 가지고 있다면 자신이 클러스터 헤드가 되어서 자식 노드로부터 전송되는 데이

터를 병합하여 부모 노드로 전송하여 최종적으로 병합된 데이터가 PE로 전송될 수 있도록 한다.

3.1 부모 노드 및 브로드캐스트 지연시간 계산

이 단계에서 각 노드들은 수신한 메시지를 이용하여 자신의 부모 노드를 결정하고 부모 노드로부터 받은 메시지를 언제 다시 브로드캐스트 해서 토폴로지 구성을 시작할지 계산한다.

그림 4는 토폴로지 구성 과정에서 각 노드에서 사용하는 부모 노드 선정 및 전송 시작 대기 시간 계산 알고리즘의 순서도이다. 모든 노드는 토폴로지 구성 메시지를 받은 후 받은 메시지의 홉 카운트를 다른 메시지와 비교하여 홉 카운트가 가장 적은 노드를 부모 노드로 선택한다. 이는 가장 효율적인 토폴로지를 구성하기 위해서는 PE와 각 노드가 가능한 한 최저의 홉 카운트를 가지도록 연결되는 것이 데이터를 전송할 때 효율적이기 때문이며 만일 복수 노드의 홉 카운

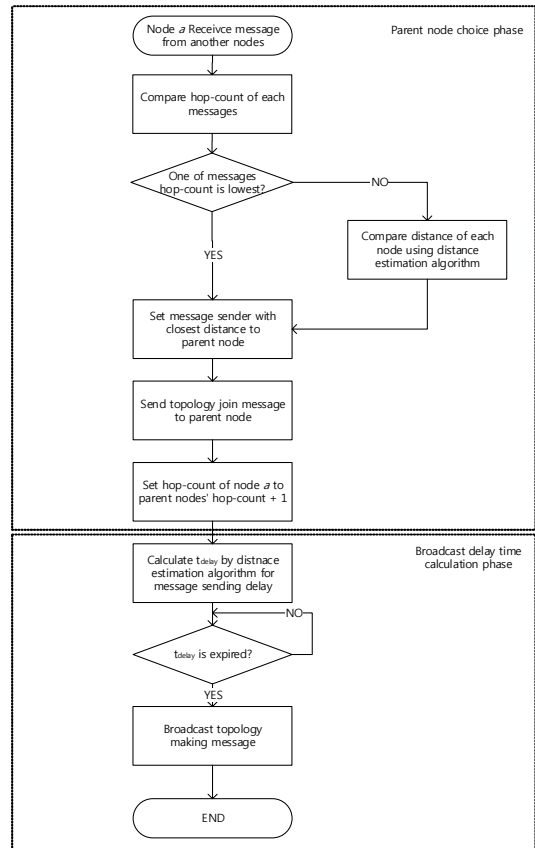


그림 4. 부모 노드 선정 및 브로드캐스트 지연시간 계산 알고리즘의 순서도  
 Fig. 4. Flow chart of parent node choice and broadcast delay time calculation algorithm

트가 같을 경우 노드 간의 연결 신뢰성을 고려하여 거리가 가까운 노드를 부모 노드로 선택한다. 노드 간의 거리는 거리 추정 알고리즘을 이용하여 구한다. 부모 노드를 선택한 노드는 자신의 부모 노드에게 참여 메시지를 전송하여서 자신이 자식 노드로 속했음을 부모 노드에게 알린다. 또한 참여 메시지를 받은 노드는 자식 노드 목록에 메시지를 보낸 노드를 추가하여 이후 클러스터 구성 단계에서 사용한다. 부모 노드를 선정하는 과정을 마친 노드는 토폴로지 구성을 위한 메시지를 보내기 위해 전송 시작 대기 시간인  $t_{delay}$  만큼 대기한 뒤 메시지 전송을 시도하며 자신이 부모 노드와 멀리 떨어져 있을수록 더욱 짧은 대기 시간을 가지도록 한다. 이는 PE로부터 가능한 한 멀리 떨어지고 적은 홉 카운트를 가지는 노드부터 순차적으로 메시지를 브로드캐스트 하는 것이 짧은 시간 내에 적은 홉 카운트를 가지는 토폴로지를 구성하는데 효율적이기 때문이며 또한 부모 노드로부터 메시지를 받은 복수의 노드가 동시에 메시지를 전송할 경우 발생할 수 있는 충돌을 줄이기 위한 목적을 가지고 있다.  $t_{delay}$  는 RSSI를 이용하여 계산한 거리 추정 값을 이용하여 산출한다. RSSI는 거리의 log값에 반비례하여 감소하는 특성을 가지고 있으며 이는 다음과 같은 식으로 표현할 수 있다.

$$RSSI = -\alpha (\log_{10}(distance) + \beta) (dBm) \quad (1)$$

$\alpha$ 와  $\beta$ 는 현재 노드가 사용하고 있는 주파수와 환경에 따른 실험적 결과물로 결정된다. 주변에 장애물이 존재하지 않는 일반적인 대기 환경에서  $\alpha$ 는 10의 값을 가지고  $\beta$ 는  $\log(\lambda^2 / (4\pi)^2 (distance)^3)$ 의 값을 가진다.  $\lambda$ 는 측정된 신호의 파장이며  $\beta$ 는 측정 환경의 장애물 등의 유무에 따라 바뀔 수 있다.

토폴로지 구성 과정을 마치면 모든 노드는 최저한의 홉 카운트를 가지면서 부모와 자식 노드 간에 가능한 연결 신뢰성이 높도록 구성된다.

그림 5는 이러한 알고리즘을 이용하여 PE에서 어떻게 토폴로지를 구성하는지 보여주는 예제이다. 처음 PE가 토폴로지 구성 메시지를 브로드캐스트 하고 이를 인접한 노드 1과 2가 수신한다. 메시지를 수신한 노드는 부모 노드로 PE를 선택한 뒤, 각자 토폴로지 구성 메시지를 브로드캐스트 할 준비를 하는데, 이 때 PE로부터 멀리 떨어진 2가 먼저 메시지를 브로드캐스트 하고 이를 수신한 노드 5는 노드 2를 부모 노드로 선정하게 된다. 이 후, PE로부터 가까운 노드인 노드

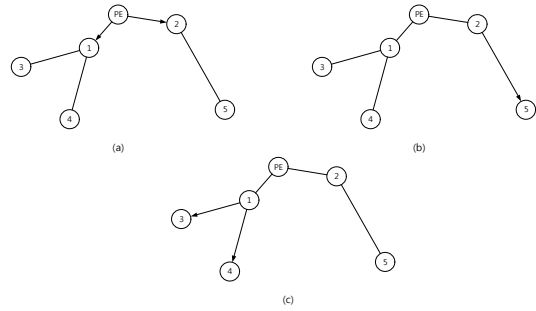


그림 5. 토폴로지 구성 과정 순서 예제 (a) PE에서 노드 1,2로 메시지 전송 (b) 노드 2에서 노드 5로 메시지 전송 (c) 노드 1에서 노드 3,4로 메시지 전송  
Fig. 5. Topology build sequence example (a) PE to Node 1,2 (b) Node 2 to Node 5 (c) Node 1 to Node 3,4

1이 메시지를 브로드캐스트 하고 이를 노드 3, 4가 수신하고 노드 1을 부모 노드로 선정한다.

토폴로지 구성 과정을 마치면 모든 노드는 최저한의 홉 카운트를 가지면서 부모 와 자식 노드 간에 가능한 연결 신뢰성이 높도록 구성된다. 이 과정을 모두 마친 후에 PE는 클러스터를 구성하는 단계를 진행한다.

### 3.2 클러스터 구성 단계

각 노드가 최저 홉 카운트를 가지는 가장 가까운 노드를 부모 노드로 선정하여 토폴로지를 구성한 뒤, 데이터 수집 및 병합의 효율성을 위하여 클러스터를 만드는 과정을 수행한다. 모든 클러스터는 한 홉 거리에 있는 노드로 이루어지며 자식 노드가 없는 노드는 부모 노드를 클러스터 헤드로 선정하고 자식 노드가 존재하는 노드는 자신이 클러스터 헤드가 된다.

기존의 클러스터링 알고리즘은 자신의 통신 범위 내에 있는 노드들 끼리 클러스터 헤드 경쟁을 한 후 클러스터 헤드를 선정하기 때문에 일반적으로 한 홉 거리에서 통신이 불가능하게 클러스터가 구성되며 이 경우에는 각 클러스터 외곽에 있는 노드를 인접한 클러스터와의 통신을 위한 게이트웨이 노드를 설정해야 한다. 이러한 다중 홉 통신은 특정 노드의 과도한 에너지 소모를 초래하여 네트워크의 수명을 단축시킨다. 이에 비해 본 알고리즘의 클러스터 구성 방식은 클러스터 헤드 간의 직접적인 통신을 가능하게 하여 더욱 효율적으로 에너지를 소비할 수 있게 한다. 그림 6은 각 노드의 클러스터 헤드 선정 알고리즘에 대한 알고리즘을 나타내며, 그림 7은 알고리즘을 이용하여 구성된 토폴로지의 예를 보여준다. 같은 색의 노드들은 같은 클러스터에 속했음을 나타낸다.

클러스터 구성이 끝나면 PE는 데이터를 수집하기



**Cluster head selection algorithm**

```

When node a receive broadcast message
if node a have child node then
    Set Cluster ID of node a to ID of node a
else if node a haven't child node then
    Set Cluster ID of node a to ID of parent node
end if
    
```

그림 6. 클러스터 헤드 선정 알고리즘에 대한 의사코드  
Fig. 6. Pseudo code of cluster head selection algorithm

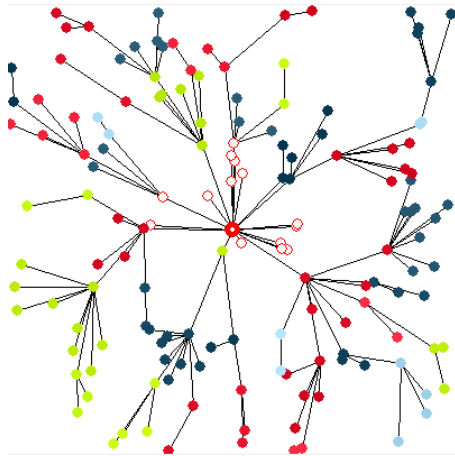


그림 7. 클러스터 구성 단계 결과 예제  
Fig. 7. Example of cluster build result

위한 규칙을 전달하기 위하여 이동 에이전트를 전송한다. 일반적인 데이터 수집 알고리즘은 한 번 수집하기 위한 데이터의 종류를 정하게 되면 이를 변경할 수 없지만, 이동 에이전트는 소프트웨어 코드를 포함하고 있으며 이를 이용하여 각 노드에서 데이터를 수집하기 위한 규칙을 변경할 수 있다는 이점이 있다.

PE는 데이터를 수집해야 할 때 이동 에이전트를 보내며 만일 병합 혹은 수집 규칙이 변경된다면 이를 이동 에이전트에 포함 시켜서 하위 노드로 브로드캐스트한다. 기존의 이동 에이전트 알고리즘은 유선 네트워크에서 동작하는 것과 유사하게 연결 리스트 형태로 동작하였으나 본 알고리즘에서는 각 클러스터 헤드가 이동 에이전트를 수신한 후 클러스터에 속한 노드들에 브로드캐스트 하는 방식으로 이동 에이전트를 배포한다. 클러스터 헤드로부터 이동 에이전트를 받은 노드는 만일 자신이 클러스터 헤드라면 자신에게 속한 노드에 다시 이동 에이전트를 브로드캐스트 하여 최종적으로 모든 노드가 데이터 수집 규칙을 갱신할 수 있도록 한다.

이동 에이전트를 배포할 때, 이동 에이전트를 수신한 노드는 브로드캐스트 지연 계산 알고리즘과 동일

하게  $t_{delay}$  를 이용하여 브로드캐스트 시작 시간을 결정하여 인접한 다른 클러스터 헤드와의 전송 충돌을 방지하도록 한다. 이동 에이전트를 받은 각 노드들은 소프트웨어 구역에 받은 데이터 수집 규칙을 갱신하고 갱신한 규칙에 따라서 데이터를 수집한다.

**IV. 시뮬레이션 결과**

본 논문에서 제안한 이동 에이전트를 이용한 데이터 수집 알고리즘과 기존에 알려진 이동 에이전트 기반의 데이터 수집 알고리즘 3종(LCF, GCF, TBID)과 비교하여 동일한 환경을 기준으로 데이터 병합에 드는 비용, 네트워크 수명을 분석하였다.

**4.1 실험 환경**

네트워크의 구성은 각 알고리즘이 제안한 토폴로지의 네트워크에서 수집된 데이터가 PE로 전달되는 상황으로 가정하였다. 토폴로지에 존재하는 노드의 개수를 100, 200, 300, 400개로 변화시키며 채널 전송률은 센서 네트워크를 구성하는데 일반적으로 사용되는 규격 Zigbee의 전송률 250Kbps로 정하였다. 본 논문에서 제안한 알고리즘에 사용된 하나의 전송슬롯의 크기는 1ms로 설정하였고 전체 노드의 데이터 생성량과 이동 에이전트의 기본 크기는 모두 동일하다고 가정한다. 실험은 각기 다르게 센서 노드가 배치된 환경에서 총 100회를 시행하여 평균을 구했다. 표 1은 시뮬레이션에 사용된 환경 정보이다.

표 1. 시뮬레이션 환경  
Table 1. Simulation environment

Item	Value
Simulation plane	500x500 ( $m^2$ )
Number of nodes	100, 200, 300, 400
Transfer rate	250 Kbps
Initial energy of sensor node	1 J
Data transfer energy	50 nJ (per byte)
Data aggregation energy	5 nJ (per byte)
Data collection energy	2 nJ (per byte)
Maximum transfer range	100 m

**4.2 데이터 병합에 드는 비용**

데이터 병합 비용 비교는 24시간 동안 데이터를 수집하여 병합 할 때 이동 에이전트가 데이터를 병합 하는데 총 소모하는 에너지에 대한 실험이다.

그림 8은 PE가 중앙( $X=250, Y=250$ )에 있을 때 노드 수를 100개에서 400개로 증가시킬 때 네트워크 내의 데이터병합 총 비용에 대한 그래프이다. 노드

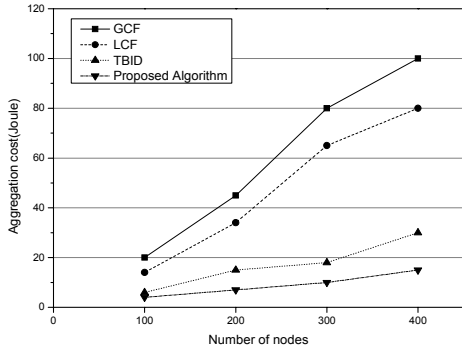


그림 8. 데이터 병합을 위한 총 비용 ( $X=250, Y=250$ )  
Fig. 8. Total aggregation cost ( $X=250, Y=250$ )

가 300개 일 때, GCF는 80J, LCF는 65J, TBID는 18J, 제안한 알고리즘은 10J의 에너지를 소모하였다. 제안한 알고리즘은 GCF, LCF, TBID에 비하여 데이터 병합 비용에서 각 87.5%, 85%, 45%의 에너지 효율 상승을 보여주었다.

그림 9는 PE가 시뮬레이션 구역 중앙 상단에 있을 때와 시뮬레이션 구역 중앙 상단( $X=250, Y=100$ )에 존재할 때 병합에 소모한 총 에너지에 대한 그래프이다. PE가 구역 중앙 상단에 존재하고 노드가 300개 일 때 GCF는 85J, LCF는 70J, TBID는 20J, 제안한 알고리즘은 12J의 에너지 소비를 보이며 제안한 알고리즘은 GCF, LCF, TBID에 비하여 데이터 병합에 각 87.5%, 85%, 45%의 에너지 효율 상승을 보여주었다.

이러한 데이터 병합 비용 성능 향상은 제안하는 알고리즘의 클러스터링 토폴로지 기반의 전송 기법과 연관이 있다. 하나의 이동 에이전트를 이용하여 모든 노드를 방문하여 데이터를 병합하는 GCF와 LCF 알

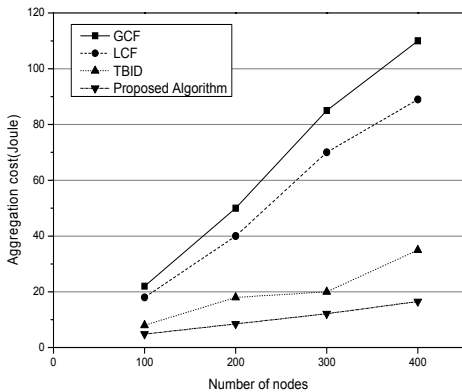


그림 9. 데이터 병합을 위한 총 비용 ( $X=250, Y=100$ )  
Fig. 9. Total aggregation cost ( $X=250, Y=100$ )

고리즘과 트리구성을 통하여 데이터를 병합해가는 TBID 알고리즘의 경우 전송 수신 횟수가 클러스터링 기반의 제안하는 알고리즘 보다 증가하며 특히 GCF, LCF의 경우 노드를 방문할수록 이동 에이전트가 수집하는 데이터의 크기가 크게 증가하게 되므로 데이터 병합 비용이 증가한다.

### 4.3 네트워크 수명

무선 센서 네트워크에서의 네트워크 수명은 첫 번째로 에너지가 고갈되는 노드가 나오는 시점을 의미한다. 1분마다 PE에서 이동 에이전트가 출발하는 상황을 가정하여 본 실험을 진행하였으며, PE의 위치는 중앙으로 고정하고 수집하는 노드의 개수를 변화시키며 실험을 진행하였다. 그림 10과 11은 PE가 중심에 있는 환경( $X=250, Y=250$ )에서 노드 수의 변화에 따른 네트워크 내에서 첫 번째로 에너지가 고갈되는 노드가 나오는 시점이다. 그림 10에서 노드가 300개

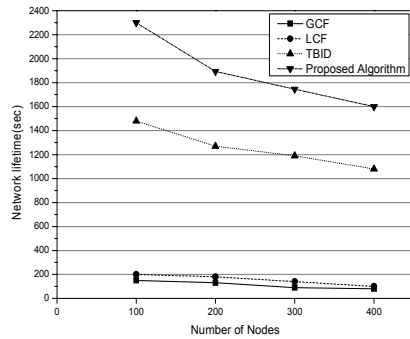


그림 10. 네트워크 수명 (데이터 크기 = 200바이트) ( $X=250, Y=250$ )  
Fig. 10. Network lifetime (data size = 200bytes) ( $X=250, Y=250$ )

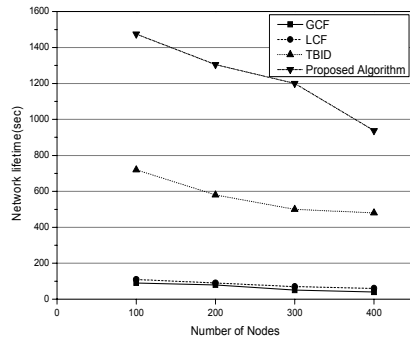


그림 11. 네트워크 수명 (데이터 크기 = 500바이트) ( $X=250, Y=250$ )  
Fig. 11. Network lifetime (data size = 500bytes) ( $X=250, Y=250$ )



일 때, GCF는 90분, LCF는 140분, TBID는 1190분, 제안한 알고리즘은 1745분의 시점에서 에너지가 고갈된 노드가 처음 발생했으며 이는 네트워크 내에서 얼마나 균등하게 에너지가 소모되는지 알 수 있는 척도이다. GCF와 LCF는 모든 데이터를 하나의 이동 에이전트가 수집하므로 일정상 나중에 도달하는 노드일수록 에너지 소모가 심하며 이는 불균등한 에너지 소모로 이어진다. 이에 비하여 TBID와 제안한 알고리즘은 하나의 이동 에이전트가 GCF와 LCF에 비하여 적은 개수의 노드를 이동하기 때문에 노드 간 에너지 불균형이 심하지 않으므로 네트워크 수명이 상대적으로 증가한다. 하지만 TBID의 경우에도 트리 구성시 부모 노드와 자식노드의 계층적 전송 경로를 생성하기 때문에 제한한 알고리즘에 비해 복수의 자식노드를 가지고 있는 부모노드의 경우 전송횟수가 증가하여 에너지 불균형이 발생하게 된다.

제안한 알고리즘은 노드가 300개 일 때 GCF, LCF, TBID에 비하여 각 194배, 124.6배, 1.46배로 네트워크 수명이 향상되었다.

그림 11은 데이터 수집 조건을 500바이트로 변경하고 실험한 결과를 보여준다. 노드가 300개 일 때, GCF는 50분, LCF는 70분, TBID는 500분, 제안한 알고리즘은 1200분이 되는 시점에서 처음으로 에너지가 고갈되는 노드가 등장하였다. 제안한 알고리즘은 GCF, LCF, TBID에 비하여 데이터를 500바이트 수집하고 노드수가 300개 일 때 각 240배, 171.4배, 2.4배의 네트워크 수명 연장을 보여주었다.

## V. 결론 및 향후 연구 과제

이동 에이전트를 무선 센서 네트워크 환경에서 데이터 수집에 적용하게 되면 기존의 클라이언트/서버 모델에 비하여 효율적인 데이터 수집 시스템을 구축할 수 있다. 그러나 효율적인 일정 결정 알고리즘이 뒷받침 되지 않는다면 오히려 클라이언트/서버 모델에 비하여 저하된 성능을 보여주게 된다.

본 논문에서 제안한 알고리즘은 클러스터를 구성함으로써 이동 에이전트의 배포 및 각 노드가 이동 에이전트의 규칙에 따라 데이터를 수집한 뒤 PE로 전송하는 과정이 효율적으로 진행될 수 있도록 하였다.

구현한 알고리즘은 기존의 이동 에이전트를 활용한 데이터 수집 알고리즘보다 네트워크의 트래픽을 감소시키는 것을 목적으로 하였고 클러스터에 기반한 토폴로지 구성과 홉 단위 전송 알고리즘을 적용하여 기존 알고리즘에 비해 전송 횟수 및 전송 경로를 단축하

였으며 실험을 통하여 데이터 병합 비용이 감소하고 네트워크 수명이 증가하였음을 확인하였다.

지금까지의 연구는 일반적으로 모든 노드가 고정된 무선 센서 네트워크 환경을 가정하고 연구되었기 때문에 향후, 노드가 이동하는 환경에서 적용할 수 있도록 본 알고리즘을 확장하기 위해서는 이동 에이전트의 효율적 전송을 위한 라우팅 알고리즘의 연구가 필요할 것이다.

## References

- [1] F. Zhao and L. Guibas, *Wireless Sensor Networks*, Morgan Kaufmann Publishers, 2004.
- [2] V. Pham and A. Karmouch, "Mobile software agents: An overview," *IEEE Commun. Mag.*, vol. 36, no. 7, pp. 26-37, Jul. 1998.
- [3] M. Chen, S. Gonzalez, and V. Leung, "Application and design issues for mobile agents in wireless sensor networks," *IEEE Wireless Commun.*, vol. 14, no. 6, pp. 20-26, Dec. 2007.
- [4] A. Boulis, C. Han, and M. Srivastava, "Design and implementation of a framework for efficient and programmable sensor networks," in *Proc. ACM MobiSys '03*, pp. 187-200, May 2003.
- [5] C. Fok, G. Roman, and C. Lu, "Mobile agent middleware for sensor networks: An application case study," in *Proc. IEEE IPSN '05*, pp. 387-387, Apr. 2005.
- [6] J. Lee and Y. Lee "Elimination of the redundant sensor data using the mobile agent middleware," *J. Korean Soc. Internet Inf.*, vol. 12, no. 3, pp. 27-36, 2011.
- [7] M. Chen, T. Kwon, Y. Yuan, Y. Choi, and V. Leung, "Mobile agent-based directed diffusion in wireless sensor networks," *EURASIP J. Advances in Signal Processing*, vol. 2007, pp. 1-13, 2007.
- [8] Y. Xu and H. Qi, "Mobile agent migration modeling and design for target tracking in wireless sensor networks," *Ad Hoc Netw.*, vol. 6, no. 1, pp. 1-16, Jan. 2008.
- [9] H. Qi and F. Wang, "Optimal itinerary

analysis for mobile agents in ad hoc wireless sensor networks,” in *Proc. Wireless '01*, vol. 1, pp. 147-153, Jul. 2001.

- [10] Q. Wu, N. Rao, J. Barhen, S. Iyengar, V. Vaishnavi, H. Qi, and K. Chakrabarty, “On computing mobile agent routes for data fusion in distributed sensor networks,” *IEEE Trans. Knowledge and Data Eng.*, vol. 16, no. 6, pp. 740-753, Jun. 2004.
- [11] C. Konstantopoulos, A. Mpitzopoulos, D. Gavalas, and G. Pantziou, “Effective determination of mobile agent itineraries for data aggregation on sensor networks,” *IEEE Trans. Knowledge and Data Eng.*, vol. 22, no. 12, pp. 1679-1693, Dec. 2010.
- [12] D. Gavalas, “Mobile software agents for network monitoring and performance management,” Ph.D. dissertation, Univ. of Essex, 2001.
- [13] Y. Xu and H. Qi, “Distributed computing paradigms for collaborative signal and information processing in sensor networks,” *J. Parallel and Distributed Computing*, vol. 64, no. 8, pp. 945-959, Aug. 2004.
- [14] H. Qi, Y. Xu, and X. Wang, “Mobile-agent based collaborative signal and information processing in sensor networks,” in *Proc. IEEE*, vol. 91, no. 8, pp. 1172-1183, Aug. 2003.

**양 명 준 (Myungjoon Yang)**



2012년 2월 : 인하대학교 전자공학과 졸업  
 2012년 3월~현재 : 인하대학교 전자공학과 석사  
 <관심분야> 컴퓨터 네트워크, 병렬 및 분산 컴퓨팅

**김 진 혁 (Jinhyuk Kim)**



2009년 2월 : 인하대학교 전자공학과 졸업  
 2011년 2월 : 인하대학교 전자공학과 석사  
 2011년 3월~현재 : 인하대학교 전자공학과 박사과정  
 <관심분야> 멀티미디어 통신, 무선 통신, 컴퓨터 네트워크, 병렬 및 분산 컴퓨팅

**최 상 방 (Sangbang Choi)**



1981년 2월 : 한양대학교 전자공학과 졸업  
 1981년~1986년 : LG 정보통신(주).  
 1988년 2월 : University of washinton 석사  
 1990년 2월 : University of washinton 박사

1991년~현재 : 인하대학교 전자공학과 교수  
 <관심분야> 컴퓨터 구조, 컴퓨터 네트워크, 무선 통신, 병렬 및 분산 처리 시스템