

# 무선 센서 네트워크에서의 에너지 분산과 QoS를 고려한 에이전트 기반의 프레임워크

신 흥 중<sup>†</sup> · 김 성 천<sup>††</sup>

## 요 약

무선 센서 네트워크는 무작위로 설치된 센서 노드가 스스로 네트워크를 형성하여 수집한 환경 정보를 전송하는 네트워크이다. 센서 노드는 매우 제한된 자원으로 동작하기 때문에 무선 센서 네트워크는 기존 네트워크에서 사용되는 기법을 적용하기 어렵다. 특히 제한된 에너지를 가지고 동작해야 하기 때문에 전송에 소모되는 에너지를 줄이기 위한 연구와 다양한 종류의 데이터를 효율적으로 전송하여 QoS를 향상시키기 위한 기법들이 연구되고 있다.

본 논문에서는 기존의 연구들이 가졌던 단점을 개선하기 위해 노드에서 소모되는 에너지의 분산과 QoS를 고려한 에이전트 기반의 프레임워크를 제안한다. 에이전트의 행동을 결정하는 정책을 유전자화 시켜 각각의 에이전트가 스스로 자신의 행동을 결정하여 동작할 수 있게 하면서 유전자 알고리즘을 통해 에이전트의 정책을 최적화 할 수 있도록 하였다. NS-2를 이용한 시뮬레이션 결과 기존의 기법에 비해 본 논문에서 제안한 기법이 노드에서 소모되는 에너지를 분산시켜 네트워크의 생존시간을 연장시키는 것을 확인할 수 있었다. 또한 긴급한 데이터의 전송 성공률을 27%, 긴급하지 않은 데이터의 전송 성공률도 14% 향상시켜 네트워크의 QoS를 향상시켰다.

키워드 : Sensor Network, MONSOON, QoS

## Agent Based Framework for Energy Distribution and QoS in Wireless Sensor Networks

Sin Hong-Joong<sup>†</sup> · Kim Sung-Chun<sup>††</sup>

## ABSTRACT

Wireless Sensor Networks are consisted of sensor nodes that communicated with each other to transmit information. Because sensor nodes have physically many limits, wireless sensor networks are hard to adopt for traditional networks. Transmissions are consumed most energy of sensor nodes. That's why energy-efficient transmission techniques and QoS support techniques for different kind of data are most important in wireless sensor networks.

The thesis proposes the agent based framework for energy distribution and QoS in wireless sensor networks. Agents have its own behavior policy by means of a gene, which is optimized by genetic operations. Agents behavior to distribute energy consumption over sensor nodes. Simulation results show that the enhanced framework extends the lifetime of sensor nodes. Successful transmission ratios of emergency data and non emergency data are increased by 27% and 14%, respectively. Also, the results demonstrate that QoS of networks are improved.

Keywords : Sensor Network, MONSOON, QoS

## 1. 서 론

무선 센서 네트워크는 일정 지역에 설치되어 있는 센서 노드를 하나의 네트워크로 연결하여 필요한 정보를 수집하는 네트워크이다. 온도, 습도, 압력 변화 등의 정보를 수집한

센서 노드는 다른 센서 노드들과의 통신을 통해 외부의 간섭 없이 스스로 네트워크를 구축하고, 구축된 네트워크를 통해 사용자에게 필요한 정보를 전송한다.[1]

무선 센서 네트워크는 각각의 센서 노드가 매우 한정된 자원을 가지고 작동하기 때문에 기존의 네트워크에서 사용되던 기법들을 적용하는 것은 부적절하다. 현재 이러한 특징을 반영하여 무선 센서 네트워크에서 에너지 효율성을 높이기 위한 많은 연구가 진행되고 있다.

본 논문은 모바일 에이전트를 적용하여 동작하는

<sup>†</sup> 준 회 원 : 서강대학교 컴퓨터공학과 공학석사  
<sup>††</sup> 정 회 원 : 서강대학교 컴퓨터학과 교수  
논문접수 : 2009년 4월 3일  
수정일 : 1차 2009년 8월 10일  
심사완료 : 2009년 8월 10일

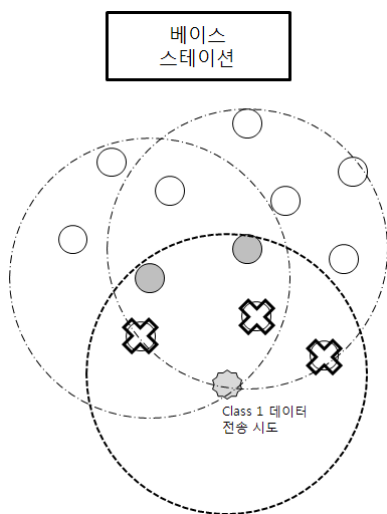
MONSOON 프레임워크를 개선하여 노드에서의 에너지 잔량을 고려하여 에너지 분산을 고려하고 데이터의 중요도에 따라 서로 다른 전송모드를 지원함으로써 QoS를 향상시킨 새로운 프레임워크를 제안한다. 제안 기법에서의 모바일 에이전트는 전송을 수행할 때 모든 노드에서 에너지를 균일하게 소비할 수 있도록 에너지 분산을 우선적으로 고려하여 특정 노드의 에너지가 일찍 고갈되는 경우를 막는다. 또한 데이터의 중요도에 따라 긴급히 데이터를 전송해야 할 경우 긴급모드를 추가로 제공하여 이를 통해 중요도가 높은 데이터를 우선적으로 전송하게 한다. 이러한 정책은 긴급도에 따라 차별화된 서비스를 제공하여 다양한 데이터에 대하여 QoS를 만족시킨다.

## 2. 무선 센서 네트워크

본 장에서는 무선 센서 네트워크에서 QoS를 보장하기 위한 전송 기법과 에너지 효율적인 전송을 하기 위한 기법 중 클러스터링 기법과 모바일 에이전트를 활용한 기법들을 간략하게 설명한다.

### 2.1 센서 네트워크에서의 QoS (Quality of Service) 기법

무선 센서 네트워크는 평소에는 주기적인 데이터를 수집하여 사용자에게 전달하는 역할을 수행하지만 긴급한 상황이 감지되었을 때 신속한 전송을 할 수 있어야 한다. 하지만 이러한 QoS에 관한 기존 연구는 무선 센서 네트워크의 다른 분야에 비해 연구가 많이 진행되지 않은 분야이다[9]. Abinash Mahapatra등이 제안한 QoS와 에너지 효율을 고려한 라우팅 기법[10]은 무선 센서 네트워크에서의 데이터를 긴급도에 따라 서로 다른 라우팅 방법을 제시했다. 이 기법에서 저자들은 긴급하게 전송되어야 하는 데이터를 Class 1으로, 긴급한 전송을 필요로 하지 않는 데이터를 Class 2로 구분하였다.



(그림 1) QoS 기반의 class 1 데이터 전송 방식

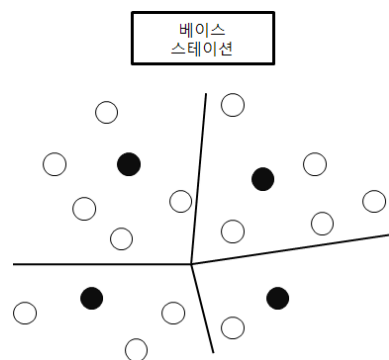
(그림 1)은 Class 1 데이터를 전송하기 위해 Abinash가 제안한 기법을 보여준다. 이 기법은 긴급한 패킷의 전송 지연시간을 줄일 수 있고, 전송에 참여할 필요가 없는 노드들을 배제할 수 있는 장점을 가진다. 하지만 네트워크의 크기가 커질 경우 Class 1 데이터를 전송하기 위해 통신에 참가하는 노드의 수가 늘어나 에너지 소비가 극심해져 네트워크 전체 수명을 짧게 하는 단점이 있다.

### 2.2 클러스터링 기법

클러스터링 기법은 클러스터 헤드 노드를 선출하고 이를 기준으로 클러스터를 구성하여 클러스터 헤드 노드가 클러스터를 대표하여 전송을 수행한다. 클러스터 헤드 노드는 자신의 클러스터에 속해있는 센서 노드들이 수집한 정보를 모아서 한꺼번에 전송을 수행하여 다른 노드들의 불필요한 전송을 줄인다.

LEACH 기법은 클러스터링 기반의 기법으로 클러스터 헤드 노드 역할을 클러스터 내부에서 주기적으로 바꾸어 클러스터 헤드 노드의 에너지 소비를 분산시키고자 한다. 이 기법은 클러스터 헤드 노드를 무작위로 선출하여 클러스터를 구성한 뒤 일정 주기마다 클러스터 내의 센서 노드들이 번갈아가면서 클러스터 헤드 노드의 역할을 수행하도록 한다. (그림 2)는 LEACH 기법을 사용하여 클러스터를 구성한 모습을 보여준다. 하지만 무작위로 선출되기 때문에 최적의 클러스터를 구성하기 어렵고 한번 구성된 클러스터를 재구성하지 않기 때문에 한 클러스터 내부에서 통신이 불가능해지는 문제가 발생한다.

이러한 방식은 클러스터 헤드 노드가 데이터를 수집하여 전송하는 방식을 이용하여 에너지 효율을 높일 수 있지만 클러스터를 유지하기 위한 비용이 많이 필요하다. 또한 클러스터 재구성이나 클러스터 헤드 노드 재선정 등의 동작을 수행하기 위한 비용이 많이 필요하게 되어 확장성, 유연성이 떨어지는 단점도 가진다.



(그림 2) LEACH 클러스터링 방식

### 2.3 에이전트 기반 기법

에이전트 기반 기법은 모바일 에이전트를 이용하여 정보 수집 및 라우팅을 수행한다. 모바일 에이전트는 이동성을 가지는 작은 크기의 소프트웨어 컴포넌트로서 지역적인 정

보를 수집하여 지능적으로 동작하기 위해 사용된다. 모바일 에이전트가 각자 자신이 가진 행동양식에 따라 최적의 결정을 선택하여 수행하기 때문에 지역적인 정보만을 가지고도 기존 기법들이 가지지 못했던 확장성, 유연성을 제공해 줄 수 있다. 모바일 에이전트의 행동양식은 자신이 수집한 정보에 따라 다음에 어떤 행동을 할지 결정하는데 사용되는 요소이기 때문에 에이전트 기반 기법에서 가장 중요한 것은 모바일 에이전트의 행동양식을 최적화 하는 것이다.

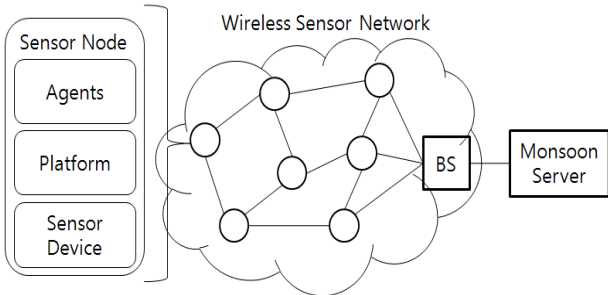
생물학적인 접근방식은 무선 센서 네트워크에서 활동하는 모바일 에이전트를 곤충의 행동양식을 적용하여 모바일 에이전트의 행동양식을 결정하고자 한다. MONSOON[6, 8]은 생물학적인 접근방식을 사용하여 모바일 에이전트가 정보를 수집하고 전송할 수 있는 환경을 제공하고 베이스 스테이션에 연결된 MONSOON 서버를 통해 일정 시간마다 좋은 성능을 가진 모바일 에이전트를 선출하는 방식을 제시했다. (그림 3)은 MONSOON에서 제안한 네트워크 구조를 보여준다.

MONSOON 에서 모바일 에이전트는 자신에게 부여된 행동양식을 이용해 이동하기 위한 다음 노드를 선택한다. 이때, 모바일 에이전트는 다음 노드를 선택하기 위해 이전 모바일 에이전트가 이동하면서 페로몬 형태로 남긴 정보를 사용한다. 페로몬 정보는 에러의 발생, 베이스 스테이션까지의 거리, 해당 노드를 통해 이동한 모바일 에이전트의 수를 포함한다. MONSOON에서 모바일 에이전트는 노드들이 가지는 가중치의 합(Weighted Sums)을 다음 수식으로 계산하여 자신이 이동해야할 노드를 결정한다[8].

$$WS_j = \sum_{t=1}^3 W_t \frac{P_{t,j} - P_{t_{min}}}{P_{t_{min}} - P_{t_{min}}} \quad (1)$$

위 (수식 1) 에서 페로몬  $P_{1,j}$ 는 노드 j를 거쳐간 에이전트의 수,  $P_{2,j}$ 는 베이스 스테이션까지의 거리,  $P_{3,j}$ 는 에러가 발생한 회수를 나타내며  $W_t$ 를 통해 각 수치들의 가중치 값으로 곱해서 더함으로써 각 노드의 합을 구해 에이전트가 이 값이 가장 높은 노드를 선택하여 이동하도록 한다.

MONSOON 서버는 베이스 스테이션에 도착한 모바일 에이전트들 중에서 우수한 에이전트를 선출하여 전송한다. 이것을 받은 모바일 에이전트는 자신이 가진 행동 양식과



(그림 3) MONSOON의 네트워크 구조

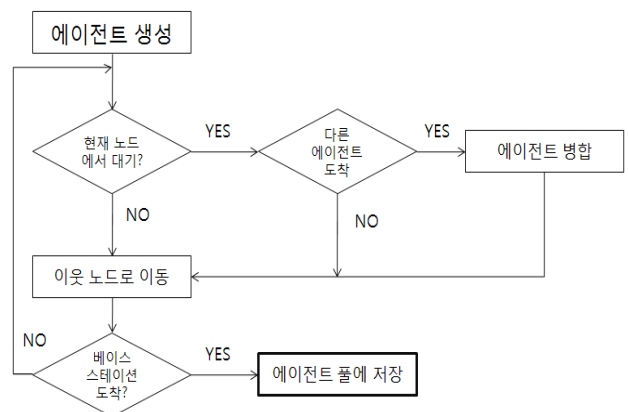
MONSOON 서버로부터 받은 정보로 유전자 알고리즘을 수행하여 개선해 나가는 방식으로 시간이 지남에 따라 최적화를 수행한다. 하지만 MONSOON 기법은 모바일 에이전트의 자율적인 동작을 위해 논리적인 에너지와 모바일 에이전트들이 이동하면서 남긴 페로몬 정보만을 이용하여 모바일 에이전트의 행동을 결정하도록 하기 때문에 실제 노드에서의 에너지 잔량과 같은 물리적인 정보를 반영하지 않는 단점을 가진다. 또한 모바일 에이전트는 많은 모바일 에이전트들로부터 선택받은 노드를 우선적으로 선택하여 경로를 설정하는 특징을 가지기 때문에 특정 노드의 에너지 소모가 빠르게 되는 단점을 가진다.

### 3. 제안 기법

본 논문은 무선 센서 네트워크에서 에너지 분산과 QoS를 고려하여 에이전트가 동작하는 프레임워크를 제안한다. MONSOON에서 사용한 것과 같이 센서 노드에서 에이전트가 동작할 수 있도록 환경을 만들어 주고 베이스 스테이션에 서버를 연결하여 유전자 알고리즘을 수행할 수 있도록 한다.

센서 노드는 자신이 충분한 데이터를 수집하였다고 판단되면 에이전트를 생성하여 베이스 스테이션으로 전송한다. 초기의 에이전트는 센서 노드가 무작위로 부여받은 유전자 값을 기준으로 행동양식이 정해지지만 차후에 유전자 알고리즘을 통해 최적값에 근접해 가도록 한다. 이동을 하기로 결정한 에이전트는 자신이 가진 유전자 값과 이웃 노드의 정보를 사용하여 이웃 노드들 중에서 최적이라고 선택된 노드로 이동한다. 긴급성이 없는 일반 데이터는 Class 2, 긴급히 전송해야할 필요가 있는 데이터는 Class 1로 분류하여 빠른 전송을 우선적으로 수행할 수 있도록 한다. (그림 4)는 Class 2 데이터를 전송하기 위한 에이전트의 동작 순서도이다.

베이스 스테이션에 연결된 서버는 베이스 스테이션에 도착한 에이전트들 중에서 최적의 성능을 보이는 에이전트를 엘리트 에이전트로 선출하여 일정 주기마다 모든 센서 노드



(그림 4) 에이전트 이동 및 병합 순서도

에게 엘리트 에이전트의 유전자 정보를 전송한다. 센서 노드는 전송 받은 엘리트 에이전트의 유전자 정보를 상속받아 자신이 생성하는 에이전트의 성능이 향상될 수 있도록 한다.

### 3.1 에이전트 병합 정책

센서 노드는 자신이 전송해야 할 데이터가 생기면 에이전트를 생성하여 데이터를 베이스 스테이션으로 전송하도록 한다. 이때 모든 에이전트가 각각 이동하게 될 경우 너무 많은 전송을 수행해야 하기 때문에 이동경로 상에서 만나는 에이전트들을 병합 하는 것이 효율적이다.

에이전트는 베이스 스테이션으로 이동하는 경로상에 있는 노드들을 지나갈 때 병합을 위한 대기 확률값( $P_w$ )를 이용하여 이동하지 않고 해당 노드에서 다른 에이전트의 도착을 기다린다. 에이전트는 다른 에이전트가 도착하게 되면 도착한 에이전트의 정보와 특성을 자신의 데이터 영역에 복사하고 도착한 에이전트를 삭제하는 방식으로 병합을 실시한다. 여러 에이전트가 한 센서 노드에서 만난 경우 에이전트의 유전자 특성이 해당 노드가 가지고 있는 특성과 가장 유사한 에이전트를 대표 에이전트로 선출하여 대표 에이전트가 나머지 에이전트들을 병합하는 과정을 수행한다. 병합 과정을 수행해서 네트워크상의 에이전트 수를 줄여서 통신을 하는데 필요한 에너지 소모를 줄일 수 있다.

모든 에이전트가 정해진 확률값을 통해 무조건 대기를 하도록 결정을 내릴 경우 최악의 경우 대기 시간이 무한대로 길어져서 에이전트가 가지고 있는 데이터가 무의미해질 수 있다. 에이전트는 전송을 수행하는 과정에서 최대 대기 회수를 제한하여 일정 시간 이상 병합을 위해 대기한 후에는 무조건 베이스 스테이션을 향해 전송을 수행한다. 이때 최대 대기 회수는 각 노드에 부여된 최대 대기 비율( $WR_{Max}$ ) 값과 노드에서 베이스 스테이션까지의 거리를 곱해서 에이전트가 생성될 때 적용한다. 최대 대기 회수가 크게 설정된 에이전트의 경우 베이스 스테이션까지 도착하는 시간이 길어지게 되지만 중간에 많은 에이전트를 병합할 수 있는 기회를 가지게 되어 에너지 소모를 줄일 수 있고 반대로 적용된 경우 병합의 기회를 적게 가지지만 베이스 스테이션에 빨리 가질 수 있는 특징을 가진다.

### 3.2 에너지 분산을 고려한 에이전트 이동 정책

에이전트의 이동은 센서 노드가 수집한 정보의 중요도에 따라 긴급히 전송해야 할 데이터는 Class 1 데이터로, 일반적인 전송방법을 선택해도 되는 경우에는 Class 2 데이터로 나누어서 서로 다른 전송 방법을 사용한다. 본 논문에서 제안하는 기법은 데이터의 종류에 따라 긴급성을 보장할 수 있도록 다른 전송방법을 택하여 무선 센서 네트워크의 QoS를 증가시키고자 하였다.

에이전트가 현재 위치한 노드에서 이동을 하기로 결정을 하면, 에이전트는 자신의 이웃 노드들의 정보와 자신이 가진 유전자 정보를 이용하여 전송을 위한 최적의 노드를 선택하여 이동한다. 다음 노드에서도 같은 방식으로 에이전트

가 이동하는 멀티홉 방식을 통해 베이스 스테이션에 도착할 때까지 반복한다. 베이스 스테이션에 도착한 에이전트는 에이전트 풀에 저장되어 일정 주기마다 엘리트 에이전트를 선출한다. 베이스 스테이션은 모든 센서 노드에게 선출된 엘리트 에이전트의 정보가 담긴 메시지를 전송하고, 센서 노드는 이 정보를 이용해 베이스 스테이션까지의 거리를 산출한다. 만약 에이전트가 이웃 노드로의 전송에 실패하게 되면 에러가 발생했음을 기록하고 다른 노드로의 전송을 시도한다.

#### 3.2.1 일반적인 데이터의 전송

Class 2 데이터는 긴급성을 요구하지 않는 데이터이기 때문에 에너지 효율적인 경로를 선택하여 전송된다. 에이전트는 이동할 노드를 결정하기 위해 자신의 이웃 노드들의 거리정보와 에러발생횟수, 다른 에이전트의 유무 정보 그리고 남아있는 에너지 량을 사용한다. 에이전트는 이와 같은 정보와 자신이 가지고 있는 유전자 정보를 활용하여 (수식 2)을 통해 계산된 값을 통해 가장 큰 값을 가진 노드를 선택하여 이동한다. (수식 2)는 다음과 같다.

$$\frac{E_j}{E_{Max}} (W_1 \frac{H_{cur} - H_j}{H_{cur} - H_{Min}} + W_2 \times ERR_j + W_3 \times A_j) \quad (2)$$

(수식 2)를 통해 에이전트는 1-hop 안에 있는 이웃 노드의 적합도를 계산한다.  $j$ 는 이웃 노드의 ID이고  $E_j$ 는 노드  $j$ 의 에너지 잔량,  $E_{Max}$ 는 초기 에너지량을 의미한다. 괄호 안의  $W_1$ ,  $W_2$ ,  $W_3$ 는 각 에이전트가 유전자 정보로써 가지는 값으로 거리, 에러율, 에이전트 유무에 가중치를 주어 합산하는데 사용한다.  $H_{cur}$ 은 현재 에이전트가 위치 하고있는 노드와 베이스 스테이션의 거리,  $H_{Min}$ 은 1-hop 이웃중에서 베이스 스테이션과 가장 가까운 노드의 거리,  $H_j$ 는 이동하고자 하는 목적 노드와 베이스 스테이션의 거리를 나타내는 값이다.

$$ERR_j = \frac{ER_j - ER_{Min}}{ER_{Max} - ER_{Min}} \quad (3)$$

$ERR_j$ 는 (수식 3)을 이용하여 이웃 노드들 중에서 에러가 발생한 비율을 반영한다.  $ER_{Max}$ 는 이웃 노드중 가장 많은 에러가 발생했던 노드의 에러횟수,  $ER_{Min}$ 은 에러가 가장 적게 발생했던 노드의 에러 횟수,  $ER_j$ 는 목적지 노드의 에러 횟수이다.

$$A_j = \begin{cases} 1, & \text{if Node } j \text{ has Agents,} \\ 0, & \text{otherwise.} \end{cases} \quad (4)$$

$A_j$  수식은 (수식 4)와 같이 에이전트가 있을 경우에 1, 없을 경우에는 0값을 반환하여서 (수식 2)에서 에이전트가 있는 노드가 우선적으로 선택될 수 있도록 한다. 에이전트가 이동을 하기 위한 노드를 선택할 때 에이전트가 있는 노드로의 이동을 선택하게 되면 에이전트가 병합될 수 있는 기

회가 늘어나 에너지 효율을 증가시킬 수 있다.

(그림 5)(a)는 에이전트가 Class 2 데이터를 베이스 스테이션까지 전달하는 과정으로 그 순서는 다음과 같다.

- Step 1. 노드 F에서 에이전트 A1이 생성되어 다음 노드 C로 이동한다.
- Step 2. 노드 E에서 에이전트 A2가 생성되어 주변 노드인 노드 B와 노드 D중에서 (그림 5)(b)의 에너지 잔량과 (수식 2)를 통해 계산된 값이 가장 큰 노드 D로 이동을 수행한다.
- Step 3. 에이전트 A1은 노드 C에서 다른 에이전트와의 병합을 위해 대기한다.
- Step 4. 에이전트 A2는 노드 B와 노드 C 중에서 (그림 5)(b)의 값을 고려하여 에너지 잔량이 높고 다른 에이전트가 존재하는 노드 C로 이동하여 에이전트 A1과 병합과정을 수행한다.
- Step 5. 에이전트 A1은 베이스 스테이션과 가장 가까운

노드 A로 이동한다.

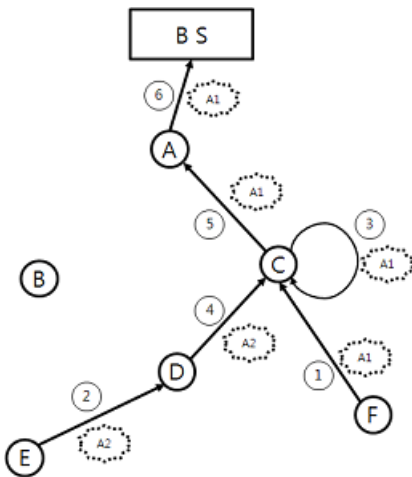
Step 6. 에이전트 A1은 베이스 스테이션으로 이동하여 전송을 마무리한다.

### 3.2.2 Class 1 데이터의 전송

Class 1 데이터는 Class 2 데이터와 달리 베이스 스테이션으로 빨리 전송되어야 하는 데이터이기 때문에 Class 2 데이터와 다른 방식으로 전송을 수행한다. 에이전트가 긴급한 상황을 감지하면, 자신을 긴급모드로 설정하고 Class 1 데이터를 전송하기 위해 (수식 2) 대신 (수식 5)를 이용하여 목적지 노드를 선택한다. (수식 5)는 다음과 같다.

$$W_1 \frac{H_{cur} - H_j}{H_{cur} - H_{Min}} + W_2 \times ERR_j \quad (5)$$

(수식 5)에서 사용되는 파라미터들은 (수식 2)에서 사용



(a) 데이터 전송 diagram

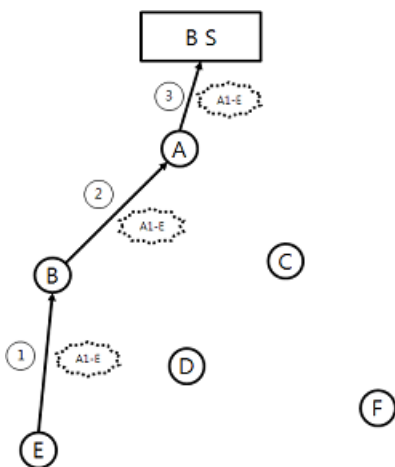
Node	Dist	Energy	Error
A	1	0.8	0.13
B	2	0.2	0.1
C	2	0.75	0.08
D	3	0.9	0.05
E	3	0.7	0.21
F	3	0.8	0.14

(b) Node 정보

Agent	$P_w$	$WR_{max}$	$W_1$	$W_2$	$W_3$
A1	0.13	0.8	0.7	-0.3	0.4
A2	0.25	0.7	0.5	-0.2	0.3

(c) Agent 정보

(그림 5) Class 2 데이터 전송 예제



(a) 데이터 전송 diagram

Node	Dist	Energy	Error
A	1	0.8	0.13
B	2	0.2	0.1
C	2	0.75	0.08
D	3	0.9	0.05
E	3	0.7	0.21
F	3	0.8	0.14

(b) Node 정보

Agent	$P_w$	$WR_{max}$	$W_1$	$W_2$	$W_3$
A1-E	0.25	0.7	0.5	-0.2	0.3

(c) Agent 정보

(그림 6) Class 1 데이터 전송 예제

되는 파라미터와 같다. (수식 5)는 Class 2 데이터 전송에서 고려되었던 파라미터 중 에너지와 에이전트의 유무에 관한 파라미터를 제거하고 거리와 에러율 만을 반영한다. 에이전트는 (수식 5)를 이용하여 이웃 노드들 중에서 베이스 스테이션으로 가장 빨리 보낼 수 있는 노드를 선택하여 전송을 수행한다.

(그림 6)(a)는 Class 1 데이터를 전송하기 위해 긴급 모드로 에이전트가 이동하는 과정을 보여준다.

- Step 1. 에이전트 A1-E이 긴급한 데이터를 전송하기 위해 긴급모드로 전송을 하기로 결정하고 주변 노드인 노드 B와 노드 D중 (그림 5)(b)의 값과 (수식 5)를 통해 유리한 노드를 선택하여 전송을 수행한다. 일반 전송모드일 경우 에너지 잔량이 높은 노드 D를 선택하지만 긴급 모드에서는 거리와 에러율만을 반영하기 때문에 노드 B를 선택하여 전송한다.
- Step 2. 긴급모드로 이동중인 에이전트 A1-E은 노드 A와 노드 C중 베이스 스테이션과의 거리가 가까운 노드 A를 선택하여 이동한다.
- Step 3. 에이전트 A1-E은 베이스 스테이션으로 이동하여 전송을 마무리한다.

3.3 엘리트 에이전트 선출

베이스 스테이션은 일정 주기마다 에이전트 풀에 저장된 에이전트들을 대상으로 엘리트 에이전트를 선출한다. 에이전트들이 초기에 자신의 유전자 값을 무작위로 할당 받은 뒤 유전자 알고리즘을 통해 최적의 유전자를 찾기 위해서 수행해야 하는 과정이다. 에이전트 풀에 저장된 에이전트들을 대상으로 지연시간, 통신비용, 병합효율성을 고려하여 각각 우수한 성능을 보인 에이전트를 엘리트 에이전트로 선출한다. 엘리트 에이전트를 선출하기 해 다음과 같은 값들을 사용한다.

$$Communication\ Cost = \frac{The\ vumber\ of\ migration}{Distance\ of\ the\ source\ node}$$

$$Delivery\ Latency = \frac{Agent\ traveling\ \nabla y}{Actual\ distance}$$

$$Aggregation\ Efficiency = \frac{The\ vumber\ of\ aggregation}{Total\ vumber\ of\ aggregation}$$

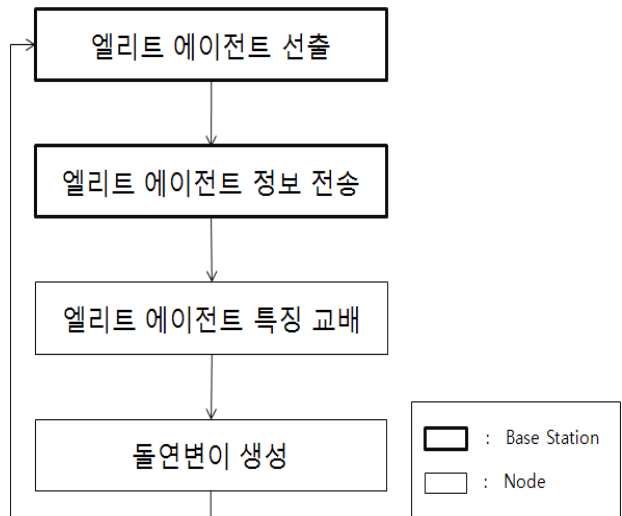
통신비용은 에이전트가 생성된 노드와 베이스 스테이션간의 거리와 에이전트가 이동해온 회수를 비교하여 최단거리를 이동해 온 에이전트를 선출하기 위해 사용한다. 지연시간은 에이전트가 병합을 위해 대기하는 시간을 모두 포함하여 실제 거리에 비례해 전송하는데 걸린 시간을 계산해 이동 시간이 가장 짧은 에이전트를 선출하기 위해 사용된다.

병합효율성은 엘리트 에이전트를 뽑기 전까지 이루어진 모든 병합의 회수와 해당 에이전트의 병합 회수를 비교하여 병합의 회수가 높은 에이전트를 선택할 수 있도록 도와준다. 하나의 특성만을 고려하여 엘리트 에이전트를 선출하는 것이 아니라 세 가지의 서로 다른 특성에서 우수한 성능을 보인 에이전트를 엘리트 에이전트로 선출하여 노드들이 유전자 값을 상속 받도록 한다.

3.4 유전자 알고리즘 적용

센서 노드는 네트워크가 초기화 될 때 유전자 정보인  $P_w, WR_{Max}, W_1, W_2, W_3$  값을 무작위로 할당 받고 이 값을 이용하여 에이전트를 생성한다. 에이전트는 할당받은 유전자 값을 기준으로 자신의 행동을 결정해서 베이스 스테이션으로 이동하여 에이전트 풀에 저장된다. 베이스 스테이션에서 선출된 엘리트 에이전트의 유전자 정보는 모든 노드에게 다시 전송되어 유전자 알고리즘을 수행한다. 에이전트가 생성될 때 부여되는 유전자 정보는 교배(crossover)와 돌연변이(mutation) 과정을 수행하여 시간이 흐름에 따라 최적의 값을 찾을 수 있도록 한다. (그림 7)은 유전자 알고리즘의 적용 방식을 간략히 설명한다.

(그림 7)에서 굵은 실선으로 표시된 부분은 베이스 스테이션에서 이루어지는 동작이며 가는 실선으로 표시된 부분은 센서 노드에서 이루어지는 동작이다. 엘리트 에이전트의 선출은 3.3장에서 설명한 방법을 통해 수행되며 선출된 엘리트 에이전트의 정보는 베이스 스테이션에서 모든 노드들에게 전송된다. 이 정보를 받은 각 센서 노드는 자신의 유전자 정보와 가장 유사한 유전자를 가지고 있는 엘리트 에이전트를 선택해서 자신의 유전자와 교배를 수행하여 엘리트 에이전트의 특성을 상속받는다. 교배가 끝난 노드는 일정 확률로 돌연변이를 일으켜서 유전자 정보가 지역적인 최적값에 머무르지 않고 최적의 값을 찾아갈 수 있도록 한다.



(그림 7) 유전자 알고리즘 동작 방식



### 4. 성능 평가

4장에서는 NS-2 (Network Simulator 2) 시뮬레이터를 이용하여 본 논문에서 제안하는 기법의 성능을 평가해 본다.

#### 4.1 성능 평가 모델

성능 평가를 수행하기 위해 NS-2 상에서 LEACH[2], MONSOON[8]을 구현하여 제안 기법과 비교하였다. 시뮬레이션 환경은 <표 1>과 같다.

각 센서 노드는 1J의 초기 에너지를 가지고 동작하며 배터리가 0.2J 이하로 떨어졌을 경우 Class 1 패킷을 우선적으로 전송하도록 설정하였다. 센서 노드에서의 이벤트는 1초 단위로 발생하며 60초를 한 라운드로 설정하여 한 라운드마다 베이스 스테이션에서 엘리트 에이전트를 선출하여 유전자 알고리즘을 수행한다. 네트워크가 초기화과정에서 센서 노드에 부여된 유전자 정보 중  $W_2$ 를 제외한 나머지 값은 0과 1 사이의 난수로 채워지고  $W_2$ 는 에러율을 반영하기 때문에 -1과 0 사이의 난수로 채워진다. 교배를 실시한 이후 돌연변이를 발생시킬 확률은 0.025로 설정하여 해당 노드에서 돌연변이가 발생할 경우 유전자 정보 중 한 가지를 무작위로 선택해 새로운 난수값을 할당하는 방식을 사용해 돌연변이를 발생시키도록 하였다.

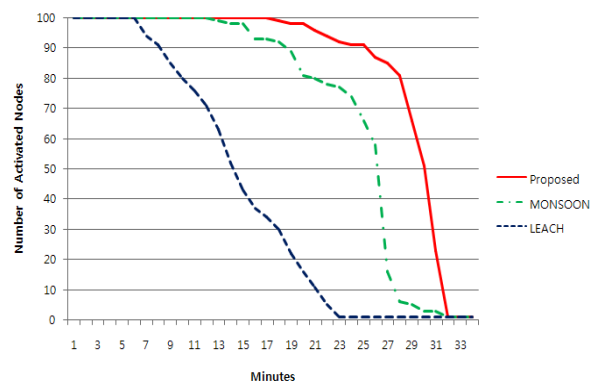
<표 1> 시뮬레이션 환경

Simulation Environment	
Network Size	100m X 100m
Maximum Transmission range	15m
Total number of Sensor nodes	100
Topology Configuration mode	Randomized
MAC Protocol	IEEE 802.15.4
Initial Energy	1 J
Sensing data Specification	
Size of Sensing data	Default : 1 Kbytes
Duty-Cycle Duration	1 second
Round Duration	60 seconds
Genetic Operation Specification	
Mutation Probability	0.025
Range of $P_w$ , $W_{Rmax}$ , $W_1$ , $W_3$	$0 \leq P_w, W_{Rmax}, W_1, W_3 \leq 1$
Range of $W_2$	$-1 \leq W_2 \leq 0$

#### 4.2 네트워크 유지시간

네트워크의 유지시간을 측정하기 위해 일정시간동안 네트워크를 동작시킨 뒤 시간별로 동작하고 있는 센서 노드의 수를 추출하였다. (그림 8)는 기법별로 시간의 흐름에 따라 생존해 있는 노드의 수를 그래프로 보여준다. (그림 8)에서 X축은 시간의 흐름을 분단위로 나타내고 Y축은 동작하고 있는 노드의 수를 나타낸다.

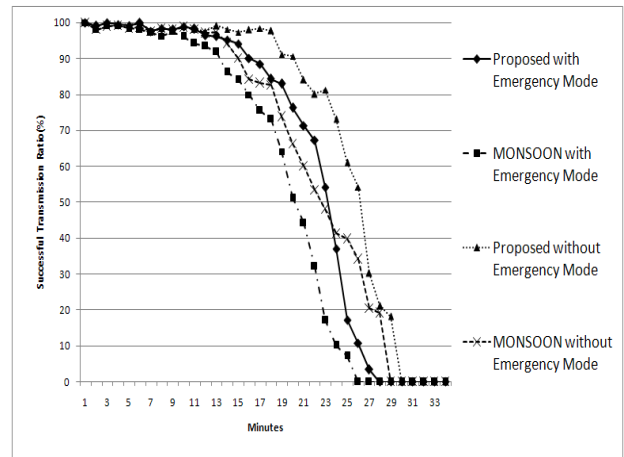
(그림 8)에서 LEACH는 다른 두 기법에 비해 에너지 소모량이 많고 클러스터 헤드로의 에너지 소모가 집중되어 6분경부터 동작하지 못하는 노드가 발생하여 급속히 생존해 있는 노드의 수가 줄어드는 것을 확인할 수 있었다. MONSOON과 제안기법의 경우 모든 에이전트가 개별적인 행동 기준을 가지고 최적의 경로를 찾아 전송을 수행하기 때문에 LEACH에 비해서 노드들의 생존시간이 늘어나 성능의 향상을 보이고 있음을 알 수 있다. 노드가 생존해 있는 시간을 기준으로 봤을 때 제안 기법은 LEACH에 비해 107%, MONSOON에 비해 18%의 성능향상을 보이고 있음을 알 수 있다.



(그림 8) 시간에 따른 생존 노드 개수

#### 4.3 전송 성공률

센서 노드에서 발생한 이벤트의 전송 성공률을 시간의 흐름에 따라 측정하여 제안기법과 MONSOON의 성능을 비교한다. 발생한 이벤트의 긴급도에 따라 긴급히 전송해야 하는 데이터는 Class 1, 그 외의 데이터는 Class 2로 분류하여 약 2:8의 비율로 이벤트를 발생시켜 실험을 진행하였다. 제안기법과 MONSOON에서 Class 1 데이터를 전송하기 위한 긴급 모드를 적용한 실험 결과와 적용하지



(그림 9) 시간에 따른 Class 2 데이터 전송 성공률

않은 실험결과를 비교하여 성능 향상이 있었음을 확인하였다.

4.3.1 Class 2 데이터 전송 성공률

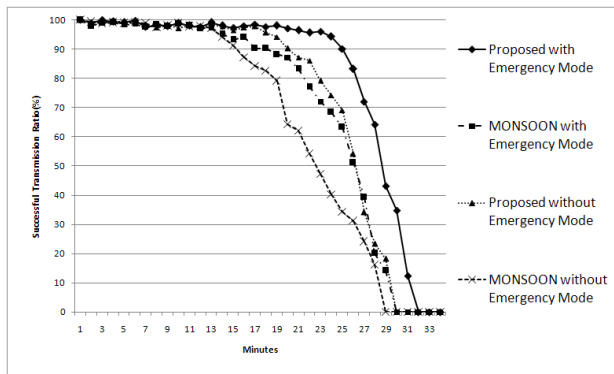
긴급한 전송을 필요로 하지 않는 Class 2 데이터는 제안 기법과 MONSOON에서 일반적인 전송방법을 사용하여 베이스 스테이션으로 전송된다. 제안 기법이 MONSOON보다 에너지 소비가 균형적으로 이루어 질 수 있도록 배려하여 많은 노드가 오랜 시간동안 동작할 수 있도록 함으로써 Class 2 데이터 전송 성공률을 높이는 것을 (그림 9)에서 확인할 수 있다. (그림 9)에서 X축을 시간의 흐름을 분 단위로 나타내고 Y축을 1분 동안의 Class 2 데이터 전송 성공률을 나타내어 긴급모드를 탑재한 제안기법, MONSOON, 긴급모드를 탑재하지 않은 제안기법과 MONSOON의 전송 성공률을 비교한다.

Class 1 데이터를 빠르게 전송하기 위한 긴급 모드를 탑재한 경우 제안기법과 MONSOON은 Class 1 데이터를 전송하기 위해 일정량 이하의 에너지가 남은 노드들을 무시하고 전송을 시도하기 때문에 긴급모드를 탑재하지 않은 경우에 비해서 전송 성공률이 떨어지는 것을 보여준다. 이는 Class 1 데이터를 우선적으로 전송하도록 하여 QoS를 확보하기 위한 제안 기법의 주요 단점으로 작용한다.

4.3.2 Class 1 데이터 전송 성공률

Class 1 데이터에 대한 전송 성공률 비교를 통해 제안기법과 MONSOON의 성능을 비교하였다. 제안 기법은 (수식 3)을 통해 에이전트가 이동할 때 에너지 분산을 고려하여 되도록 많은 노드가 Class 1 데이터를 전송할 수 있는 상태를 유지하도록 고려하여서 시간에 따른 전송 성공률이 MONSOON에 비해 상승하였다. (그림 10)은 이러한 결과를 보여준다.

긴급 모드를 탑재한 경우 제안기법과 MONSOON 모두 탑재하지 않은 경우에 비해서 데이터 전송 성공률이 증가한다. Class 1 데이터 전송 성공률이 80% 이하로 떨어지는 시점을 기준으로 보았을 때 긴급모드를 탑재한 제안 기법은 긴급모드를 탑재한 MONSOON에 비해서 25%의 성능향상



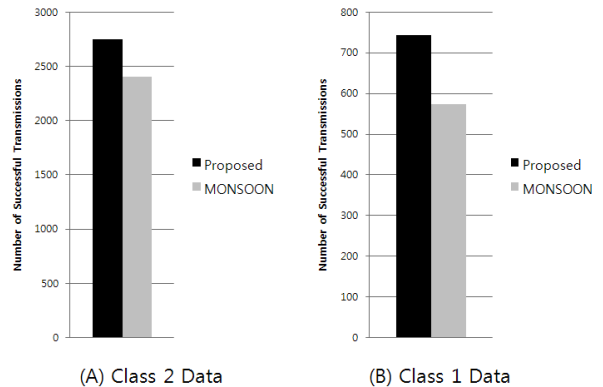
(그림 10) 시간에 따른 Class 1 데이터 전송 성공률

을 보인다.

4.3.3 데이터 종류에 따른 전송 성공 횟수

Class 1 데이터를 전송하기 위한 긴급 모드를 탑재한 제안기법과 MONSOON을 대상으로 센서노드가 동작하는 시간동안의 전송 성공 횟수를 비교하였다. (그림 11)는 Class 1,2 데이터에 대한 전송 성공률을 나타내는 그래프로서 Y축은 모든 노드가 동작하지 못한 시간까지의 전송 성공 횟수를 나타낸다.

제안 기법은 (수식 2)를 통해 Class 2 데이터를 에너지 분산을 고려하여 전송하는 방식을 채택하여 MONSOON에 비해서 전체 전송 성공 횟수를 약 14% 증가 시켰다. Class 1 데이터의 경우 제안기법에서 적용된 긴급 모드를 통해 Class 1 데이터의 긴급성을 반영하여 에너지 분산과 관계없는 빠른 경로를 통해 전송할 수 있도록 하였으며 Class 2 데이터 전송에서 에너지 분산을 고려함으로써 더 많은 노드가 오랜 시간동안 Class 1 데이터를 전송하기 위해 생존할 수 있도록 하였다. 그 결과 제안기법이 MONSOON보다 Class 1 데이터 전송 성공 횟수에서 약 27%의 성능향상을 보였다. 제안기법은 Class 1 데이터와 Class 2 데이터에 각기 다른 경로 선택 기법을 통해 다양한 종류의 데이터를 처리할 수 있도록 QoS를 증가시켰다.



(그림 11) 데이터 종류에 따른 전송 성공 횟수

5. 결 론

본 논문에서는 무선 센서 네트워크에서 에너지 분산과 QoS를 고려하여 에이전트가 동작하는 프레임워크를 제안하였다. 에이전트는 이웃 노드의 정보만을 활용하여 특정 노드의 에너지 소비가 집중되지 않도록 다음 노드로의 이동을 결정하고 경로 상에서 다른 에이전트를 만날 경우 병합을 수행하여 통신의 효율성을 높인다. 에이전트의 행동 양식을 결정하기 위한 유전자를 도입하고 유전자 알고리즘을 통해 에이전트의 동작이 시간에 흐름에 따라 최적화 될 수 있도록 한다. 또한 다양한 QoS를 만족시키기 위해서 데이터의 긴급도에 따라 서로 다른 전송모드를 지원한다.



NS-2를 이용하여 제안 기법과 MONSOON, LEACH의 성능 비교를 수행한 결과 네트워크의 유지시간과 데이터의 전송 성공률에 대해 보다 좋은 성능을 나타내는 것을 확인하였다. 네트워크 유지시간에 대해서 시간의 흐름에 따라 동작 가능한 노드의 수를 비교하여 50%의 노드가 동작 가능한 시점을 기준으로 제안 기법이 LEACH, MONSOON 보다 78~107%의 성능향상을 보여준다. 긴급도에 따라 Class 1, 2 데이터로 나누어 전송 모드를 지원한 실험 결과 제안 기법은 MONSOON에 비해 Class 2 데이터의 경우 14%, Class 1 데이터의 경우 27%의 전송 성공 횟수가 향상되어 요구했던 QoS를 만족시켰다.

본 논문에서 제안한 기법은 에이전트를 활용하여 에너지 분산과 QoS를 동시에 만족시킬 수 있었다. 하지만 긴급도에 따른 전송 방식을 다르게 택하면서 긴급도가 떨어지는 Class 2 데이터의 전송 성공률이 떨어지는 단점을 보였다. 향후에는 다양한 긴급도를 가진 전송방식을 제공 하면서도 각 전송방법에서 발생하는 손실을 최소화 할 수 있는 연구를 진행하고자 한다.

## 참 고 문 헌

- [1] Chee-Yee Chong and Srikanta P. Kumar, "Sensor Networks: Evolution, Opportunities, and Challenges," *In Proceedings of the IEEE*, Vol.91, pp.1247-1256, 2003.
- [2] Wendi R. Heinzelman, Anantha Chandrakasan, and Hari Balakrishnan, "Energy-Efficient Communication Protocol for Wireless Microsensor Networks," *In Proceedings of the 33rd Hawaii International Conference on System Sciences*, Vol. 2, pp.10-19, 2000.
- [3] Min Chen, Taekyoung Kwon, Yong Yuan, and Victor C.M. Leung, "Mobile Agent Based Wireless Sensor Networks," *In Journal of Computers*, Vol.1, No.1, pp.14-21, 2006.
- [4] Tracy Mullen, Viswanath Avasarala, and David L. Hall, "Custom-Driven Sensor Management," *In Magazine of INTELLIGENT SYSTEMS*, Vol.21, No.2, pp.41-49, 2006.
- [5] Geoffrey Mainland, David C. Parkes, and Matt Welsh, "Decentralized, Adaptive Resource Allocation for Sensor Networks," *In Proceedings of the 2nd USENIX/ACM Symposium on Networked Systems Design and Implementation (NSDI 2005)*, Vol.2, pp.315-328, 2005.
- [6] Pruet Boonma and Junichi Suzuki, "BiSNET: A Biologically-Inspired Middleware Architecture for Self-Managing Wireless Sensor Networks," *In Journal of Computer Networks*, Vol.51, No.16, pp.4599-4616, 2007.
- [7] Pruet Boonma and Junichi Suzuki, "MONSOON: A Coevolutionary Multiobjective Adaptation Framework for Dynamic Wireless Sensor Networks," *In Proceedings of the 41st Hawaii International Conference on System Sciences*, pp.497, 2008.
- [8] Pruet Boonma and Junichi Suzuki, "Exploring Self-star Properties in Cognitive Sensor Networking," *In Proceedings of IEEE/SCS International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS)*, 2008.
- [9] Dazhi Chen and Parmond K. Varshney, "QoS Support in Wireless Sensor Networks: A Survey," *In Proceedings of the 2004 International Conference on Wireless Networks (ICWN 2004)*, pp.227-233, 2004.
- [10] Abinash Mahapatra, Kumar Anand, and Dharma P. Agrawal, "QoS and energy aware routing for real-time traffic in wireless sensor networks," *In Journal of Computer Communications*, Vol.29, pp.437-445, 2005.
- [11] Wendi R. Heinzelman, Anantha Chandrakasan, and Hari Balakrishnan, "An Application-Specific Protocol Architecture for Wireless Microsensor Networks," *IEEE Transactions on Wireless Communications*, Vol.1, No.4, pp.660-670, 2002.
- [12] Alan D. Amis, Ravi Prakash, Thai H.P. Vuong, and Dung T. Huynh, "Max-Min D-Cluster Formation in Wireless Ad Hoc Networks," *In Proceedings of IEEE INFOCOM*, pp.32-41, 2000.



## 신 흥 중

e-mail : siceman00@gmail.com

2007년 서강대학교 컴퓨터학과(학사)

2009년 서강대학교 컴퓨터공학과(공학석사)

관심분야: 무선통신망, 센서네트워크 등



**김 성 천**

e-mail : ksc@sogang.ac.kr

1975년 서울대학교 공업교육학전기전공  
(학사)

1979년 Wayne State Univ. 컴퓨터공학  
(공학석사)

1982년 Wayne State Univ. 컴퓨터공학  
(공학박사)

1982년~1984년 캘리포니아주립대 조교수

1984년~1985년 금성반도체(주) 책임연구원

1985년~현재 서강대학교 컴퓨터학과 교수

관심분야: 병렬처리시스템(Parallel Computer Architecture, Inter-connection Network), WDM technology를 이용한 cluster system, 유비쿼터스 컴퓨팅, Pervasive Computing