

# 무선 센서 네트워크를 위한 임계치 기반 효율적인 혼잡 탐지 및 제어 알고리즘

## Efficient Congestion Detection and Control Algorithm based on Threshold for Wireless Sensor Network

이대운\*, 이태우\*, 최승권\*, 이준석\*, 김광훈\*, 이재엽\*\*  
충북대학교 컴퓨터공학과\*, (주)팬시아21\*\*

Dae-Woon Lee(dylee777@yahoo.co.kr)\*, Tae-Woo Lee(dickgap@gmail.com)\*,  
Seung-Kwon Choi(hamsterc@hanmail.net)\*, Joon-Suk Lee(joonsuk2@gmail.com)\*,  
Jin Guangxun(jinqian32@cbnu.ac.kr)\*, Jae-Youp Lee(john@pangaea.co.kr)\*\*

### 요약

본 연구에서는 불필요한 에너지 소모와 지연을 야기하는 혼잡을 탐지할 수 있는 알고리즘을 제시하고 이에 따라 혼잡 제어를 수행할 수 있는 기법을 제시한다. 기존의 혼잡 탐지는 큐 점유율만을 사용하거나 MAC 계층에서 패킷 도착율만을 가지고 혼잡을 결정하였으나 본 연구에서는 이들 두 가지를 모두 고려하여 더 정밀한 혼잡 탐지를 가능하게 하였다. 또한 혼잡도에 따라 미리 혼잡 회피를 수행하고 혼잡이 발생한 경우 패킷의 우선순위에 따라 지연에 민감한 패킷들을 먼저 스케줄링할 수 있는 방법을 제시하였다. 시뮬레이션 결과 본 논문에서 제안한 혼잡 탐지와 제어 기법은 기존의 혼잡 제어 방법들에 비해 우수한 성능을 보임을 알 수 있었다.

■ 중심어 : | 무선 센서 네트워크 | 혼잡 제어 | 혼잡 탐지 | 임계치 |

### Abstract

This paper reports a new mechanism for congestion controls. The proposed congestion detection algorithm can be provided with delay and unnecessary energy consumption. Conventional congestion control methods decide congestion by queue occupancy or mean packet arrival rate of MAC layer only, however, our method can perform precise detection by considering queue occupancy and mean packet arrival rate. In addition, the congestion avoiding method according to congestion degree and scheduling method using priority for real time packets are proposed. Finally, simulation results show that proposed congestion detection and control methods outperforms conventional scheduling schemes for wireless sensor network.

■ keyword : | Wireless Sensor Network | Congestion Control | Congestion Avoidance | Threshold |

## 1. 서 론

무선 센서 네트워크는 인간과 인간 인간과 사물 인간과 기계와 같이 인간과 타 매체를 통한 정보 수집을 위하여 효율적인 구조를 가져야 하며 인간의 오감을 대체

하고 또한 인식을 할 수 있는 센서의 역할도 중요하다. 이러한 유비쿼터스 컴퓨팅 환경을 구축하기 위하여 지능화된 초소형의 센서가 필요하며 이를 통하여 주변 공간의 상황을 자동적으로 인지하여 또한 사용자들이 물리적으로 떨어져 있더라도 원하는 곳의 사물과 주변 환

경의 변화를 인식하거나 추적하여 그에 따른 적절한 정보와 서비스를 제공할 필요가 있다.

무선 센서 네트워크의 특징으로는 첫째 센서네트워크는 대규모 숫자의 통신노드로 구성된다. 감시라는 업무 자체의 특성상 보다 많은 관찰점에서 다각도의 감시 정보를 수집한다면 보다 풍부한 상황정보를 파악할 수 있기 때문이다. 이는 기존의 단일 센서에서 수행하던 임무를 보다 많은 노드에서 분배함으로써 정보의 정확도를 높인다[1].

둘째, 무선 센서 네트워크의 노드들은 사람에 의해 배포되기 보다는 비행기 혹은 다른 운반 수단을 통해 무작위로 배치된다. 이는 사람이 쉽게 접근하기 어려운 지역에 배치하여 노드가 사람의 관리나 간섭 없이 네트워크를 자기 구성할 수 있어야 한다는 기술적 요구사항을 지닌다[2][3].

셋째, 센서네트워크의 노드는 배터리로 동작하는 저전력, 저가의 시스템이다. 그러나 현재의 배터리 기술의 한계로 인하여 노드들을 에너지 사용에 대단히 제약을 받고 배터리가 소모되어 노드의 동작이 멈추면 이는 다시 재충전하거나 교체되지 않는다[4].

에너지 제약이 심한 센서 네트워크에서의 혼잡제어는 특히 중요하다. 혼잡의 발생은 패킷의 손실로 이어져 전체 전송시간이 증가되며, 손실된 패킷의 복구를 위해 재전송이 계속되므로 에너지 소모도 급격히 증가하게 된다. 예를 들어 하나의 이벤트는 몇 바이트 길이이지만, 잦은 보고율은 네트워크의 자원(버퍼, 배터리 수명)을 빠르게 소모하여 혼잡을 발생시키고 결국 패킷 또는 이벤트의 손실로 이어지게 된다[5]. 따라서 무선 링크를 통해 전달되는 패킷이 도중에 손실되거나 중복이 발생하지 않고 전송되도록 보장하는 신뢰성 있는 전송기법이 필요하다.

혼잡을 발생시키는 원인은 버퍼 오버플로우, 동시 전송, 패킷 충돌 등 여러 가지가 있다[6]. 본 논문에서는 이러한 원인에 초점을 맞추어 무선 환경에서 신뢰성 있는 데이터 전송을 보장하고 불필요한 재전송을 최소화하기 위한 혼잡 탐지 및 회피 기법을 제안한다. 이는 혼잡 회피와 우선순위 제어를 통해 수행되며 신뢰성 있는 전송을 기반으로 한 혼잡 제어가 이루어질 수 있도록

하였다.

## II. 관련 연구

### 1. 비율 기반 혼잡제어

비율 기반(rate-based) 기법의 기본 아이디어는 포워딩 노드가 각 상향 이웃들로부터 들어오는 흐름의 수를 추정하고 혼잡이 감지되면 공평성에 기반하여 전송율을 할당하는 것이다. [7]에서는 혼잡 제어를 위한 이벤트-싱크간의 신뢰성 있는 전송 프로토콜이 제안되었다. 각 센서 노드는 자신의 지역 버퍼를 모니터링하며 버퍼 오버플로우가 발생할 경우 싱크로 전송되는 패킷의 혼잡 알림 비트를 셋팅하여 이를 알린다. 싱크가 혼잡 알림으로 표시된 패킷을 수신하면 혼잡이 발생한 것을 알 수 있으며 모든 소스노드들에게 이들의 보고 주기를 증가시키도록 제어 신호를 브로드캐스트 한다. [8]에서는 분산 혼잡제어와 회피 프로토콜이 제안되었으며, [9]에서는 홑대홑 흐름 제어, 전송율 제한과 우선순위 MAC 계층을 결합한 혼잡제어 프로토콜이 제안되었다. 이 기법은 올바르게 동작하기 위해서는 트리형태의 라우팅 구조를 필요로 한다. [10]에서는 군집 공평성 프로토콜을 위한 지역화된 알고리즘이 제안되었다. 센서가 자신이 포워드 할 수 있는 것보다 많은 패킷을 수신하면 센서는 가중치를 부여한 공평성 함수를 통해 상향 이웃들의 데이터율을 계산하고 재할당 한다. 그러나 여기서 공평성 함수는 남은 버퍼 크기와 전송율을 동시에 고려하지 않은 문제점이 있다.

### 2. Buffer-Based 혼잡제어

버퍼 기반 기법들에서는 센서  $i$ 는 자신의 하향노드  $j$ 의 버퍼크기가 충분할 경우에만  $j$ 로 패킷을 전송한다. [11]에서는 센서 네트워크에서의 경량 버퍼 관리에 기반한 혼잡 회피 프로토콜이 제안되었다. 이 기법은 숨겨진 터미널 문제를 해결하기 위해 1/6 버퍼 알고리즘이 사용되었다. 모든 센서는 자신의 남은 버퍼중 단지 1/6만을 광고한다. 이는 포워딩의 관점에서 패킷 폐기를 방지할 수 있지만 버퍼 활용율이 낮은 단점을 가진다.

### III. 혼잡 탐지 및 제어 알고리즘

#### 1. 시스템 모델

[그림 1]에는 본 논문에서 사용한 네트워크 모델을 보이고 있다. 혼잡제어는 상황으로만 동작하며 센서 네트워크는 단일경로 라우팅을 지원한다고 가정한다. 또한 여기서는 CSMA와 유사한 MAC 프로토콜을 사용하며 각 센서 노드는 소스 트래픽과 경유 트래픽의 두 종류를 다룰 수 있다. 소스 트래픽은 각 센서에서 지역적으로 생성된 것이며 경유 트래픽은 소스 노드에서 생성된 트래픽을 전달하는 역할을 한다. 따라서 각 센서 노드는 소스 노드이거나 경유 노드가 될 것이며 자식 노드가 없다면 이 노드는 소스 노드로만 동작하고 소스 트래픽만을 가진다.

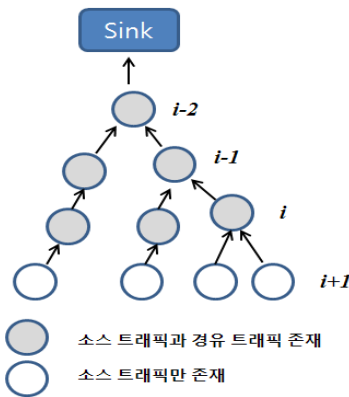


그림 1. 센서 네트워크의 구성

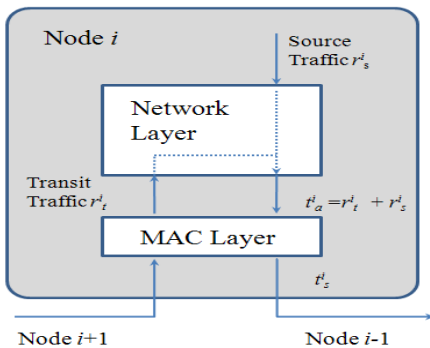


그림 2. 센서노드의 큐잉 모델

[그림 2]는 특정한 센서 노드  $i$ 에서의 큐잉 모델을 보이고 있다. 노드의 경유 트래픽  $r_t^i$ 는 MAC 계층을 통해 자신의 자식노드인  $i+1$  노드로부터 수신된다. 소스 트래픽은  $r_s^i$ 의 생성율로 지역적으로 발생되며 소스 트래픽과 경유 트래픽은  $i$ 의 부모 노드인  $i-1$  노드로 전송되기 전에 병합된다. 패킷은 전체 입력 트래픽을  $t_a^i = r_s^i + r_t^i$ 가 MAC 계층 전송율인  $t_s^i$ 를 넘어서면 큐에 쌓이게 된다. 패킷 전송율  $t_s^i$ 는 MAC 프로토콜과 연관이 있다. 여기서는 CSMA와 유사한 프로토콜을 사용한다고 가정하였으므로 활성 센서 노드의 수는 트래픽 밀도와 함께 노드의 패킷 전송 속도에 영향을 주게 된다.

패킷 입력율인  $t_a^i$ 가 패킷 전송율  $t_s^i$ 를 넘어서면 패킷은 노드  $i$ 에 쌓이게 되며 노드 수준의 혼잡이 시작된다. 따라서 이 시점에서는  $t_a^i$ 를 감소시키거나  $t_s^i$ 를 증대시켜야 할 필요가 있다.  $t_s^i$ 는 MAC 프로토콜을 조정하여 증가시킬 수 있지만 일반적으로  $r_s^i$ ,  $r_t^i$ 를 병목시켜  $t_a^i$ 를 감소시키는 것이 훨씬 쉽다. 소스 트래픽 생성을  $r_s^i$ 는 샘플링(보고) 주기를 변경함으로써 지역적으로 감소시킬 수 있다. 경유 트래픽  $r_t^i$ 는  $i+1$  노드의 전송율을 조절함으로써 간접적으로 감소시킬 수 있다. 다른 한편으로 노드  $i$  주변의 링크에서 충돌이 발생한다면 노드  $i$ 와 그 이웃 노드들은 링크 수준 혼잡을 예방하기 위해 채널 접근을 감소시킨다. 이러한 작업은 MAC을 통해 수행될 수도 있지만  $t_a^i$ 를 감소시키는 것이 더 쉬운 해결 방법이다.

#### 2. 혼잡 탐지 및 제어 프로토콜 제안

본 논문에서는 혼잡 탐지에서 혼잡 회피, 비율제어 이르는 총괄적인 혼잡 제어 프로토콜을 제안하고자 한다. 이는 다음과 같은 이유로 필요하다.

- 1) 무선 센서 네트워크에서 센서 노드는 이들의 기능과 위치에 따라 서로 다른 우선순위를 가질 수 있다. 따라서 혼잡 제어 프로토콜은 센서의 우선순위에 따른 공평성을 보장할 수 있어야 한다.
- 2) 혼잡제어 프로토콜은 에너지 효율성을 증대시켜

야 하고 패킷 전달 지연, 처리율, 패킷 손실율의 관점에서 전통적인 QoS를 보장해야 한다.

이러한 문제를 해결하기 위해 본 연구에서는 약간의 오버헤드로 공평성을 보장하고 패킷 손실을 감소시킬 수 있는 알고리즘을 제안한다. 제안 알고리즘은 크게 혼잡 탐지, 혼잡 회피와 우선순위 기반 비율제어의 세 부분으로 구성된다.

먼저 혼잡 탐지는 패킷의 도착율과 큐의 점유율을 기반으로 혼잡도 CD(congestion degree)를 결정한다. 패킷의 도착율과 큐 점유율은 현 시점에서 혼잡의 정도를 반영하며 이를 통해 혼잡 제어를 위한 기본 정보를 마련한다.

다음으로 혼잡 탐지를 통해 결정된  $i$ 노드의 혼잡도  $CD(i)$ 의 값에 따라 혼잡 회피를 수행해야 할 것인지 비율 제어를 수행해야 할 것인지를 결정한다.

일단 가까운 시간 내에 혼잡이 발생할 것으로 예상되면 혼잡 회피 절차를 수행한다. 이는 혼잡이 발생하기 전에 경로 변경을 통하여 과부하 노드로 유입되는 트래픽을 감소시킨다.

혼잡도  $CD(i)$ 의 값이 현재 과도한 혼잡이 발생한 상태이면 [그림 3]의 스케줄러를 이용하여 즉시 혼잡 제어를 수행한다. 스케줄러는 네트워크 계층과 MAC 계층 중간에 위치하며 소스 트래픽과 경유 트래픽을 위한 두 개의 큐를 가지고 있다. 스케줄링 알고리즘은 WFQ(Weighted Fair Queueing)나 WRR(Weighted Round Robin)등의 알고리즘이 소스와 경유 트래픽 간의 공평성을 보장하면서 사용될 수 있다. 소스와 경유 트래픽의 우선순위는 스케줄링 알고리즘의 가중치로 사용될 수 있으며 혼잡 탐지시 가중치를 변경하여 두 트래픽의 비율을 조절할 수 있다.

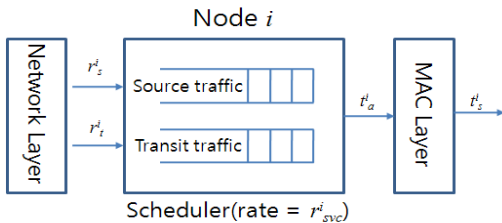


그림 3. 제안 기법의 노드 모델

비율 제어는 주어진 우선순위에 기반하여 각 센서 노드들이 유연한 공평성을 가지는 동시에 일정 수준 이상의 처리율을 보장하기 위해 사용된다. 비율 제어는 먼저 높은 우선순위를 가진 트래픽에게 더 많은 대역폭을 제공하고 가중치에 따른 유연한 공평성을 제공한다. 예를 들어 싱크가 각 센서 노드들로부터 같은 수의 패킷을 수신하길 원한다면 모든 노드들에 대하여 똑같은 우선순위를 제공하면 된다. 반면에 싱크가 특정 센서 노드들로부터 센싱된 데이터를 좀 더 상세히 받고자 한다면 이 노드들에게 높은 우선순위를 부여하여 더 높은 대역폭을 제공하면 된다.

### 2.1 혼잡 탐지

TCP와 같은 전통적인 프로토콜에서 혼잡은 IP 패킷의 ECN 비트나 전송자의 타이머, 또는 ACK 메시지를 통해 알 수 있다. 센서 네트워크에서는 중간 노드들이 큐 점유율이나 무선 채널 상태 등을 검사하여 혼잡을 탐지한다. 그러나 이러한 방법으로는 혼잡 수준을 명확히 반영할 수 없다.

이에 따라 본 논문에서는 노드의 혼잡 수준을 명확히 측정하기 위해 큐 점유율에 더하여 패킷 평균 도착율과 패킷 전송율을 함께 혼잡 탐지에 사용하였다. 패킷의 평균 도착율을  $t_a^i$ , 패킷 전송율을  $t_s^i$ 라 하고 큐 점유율을  $q_i^i$ 라 한다. 여기서 패킷 도착율은 소스나 경유 트래픽으로부터 순차적으로 도착하는 패킷 사이의 시간간격을 의미하며, 패킷 전송율은 노드에서 처리를 마치고 다음 노드로 출발하는 간격이다. 큐 점유율을 특정 시간  $t$ 에서의 큐 사용량을 의미한다.

패킷 도착율과 패킷 전송율은 한 노드의 혼잡을 결정하는데 중요한 역할을 하며 노드  $i$ 에서 이들 둘 사이의 비율  $c(i)$ 를 실제 혼잡탐지에서 사용하게 된다.

$$c(i) = t_s^i / t_a^i \quad (1)$$

여기서  $c(i) < 1$ 인 경우 패킷 도착율에 여유가 있다는 의미이며  $c(i) > 1$ 인 경우 노드의 처리 능력을 넘어서므로 패킷 도착율을 낮추어야 한다.

평균 패킷 도착율  $t_a^i$ 는 다음과 같이 구할 수 있다.

먼저 노드의 평균 경유 트래픽  $r_t^i$ 은

$$r_t^i = (1 - w_a)t_a^i + w_a(t - t') \quad (2)$$

이며, 여기서  $w_a$ 는  $0 < w_a < 1$ 의 값을 가지는 상수,  $t$ 는 패킷의 도착 시간,  $t'$ 는 이전 패킷의 도착 시간이다.

따라서 노드  $i$ 가 전송해야 할 전체 데이터는 소스 트래픽과 경유 트래픽의 합으로 다음과 같이 표시할 수 있다.

$$t_a^i = r_s^i + r_t^i \quad (3)$$

또한 노드의 패킷 전송율  $t_s^i$ 는 다음과 같다.

$$t_s^i = (1 - w_s)t_s^i + w_s t_s' \quad (4)$$

여기서  $w_s$ 는  $0 < w_s < 1$ 의 값을 가지는 상수이며,  $t_s'$ 는 방금 전송된 패킷의 서비스 시간이다.

상기의 식들을 통해 혼잡은 3가지 형태로 구분될 수 있다.

- 1) 첫 번째 경우 :  $q_t^i$ 는 임계치  $q_{th}^i$ 를 넘지 않았으나  $c(i)$ 가 1을 넘어선 경우 즉,

$$q_t^i < q_{th}^i \text{ and } c(i) > 1$$

이러한 경우 큐 점유율은 점차 높아질 것이며 가까운 시간 내에 혼잡이 발생할 수 있으므로 혼잡 회피를 수행한다.

- 2) 두 번째 경우 :  $q_t^i$ 는 임계치  $q_{th}^i$ 를 넘어 서고  $c(i)$ 가 1보다 작은 경우 즉,

$$q_t^i > q_{th}^i \text{ and } c(i) < 1$$

이러한 경우 높은 큐 점유율로 인해 가까운 시간내에 혼잡이 발생할 가능성이 있으므로 혼잡 회피를 수행한다.

- 3) 세 번째 경우 :  $q_t^i$ 가 임계치  $q_{th}^i$ 를 넘고  $c(i)$ 가 1을 넘어선 경우,

$$q_t^i > q_{th}^i \text{ and } c(i) > 1$$

이는 현재 과도한 혼잡 상태이며 즉시 혼잡 제어가 필요하다.

따라서 첫 번째, 두 번째 경우는 혼잡 회피를 위해, 세 번째 경우는 이미 혼잡이 발생한 상태로서 혼잡 제어가 필요하게 된다.

제안 기법에서는 큐 점유율과 패킷 도착율/패킷 전송율의 비를 모두 고려하여 혼잡도  $CD(i)$ 를 계산한다. 혼잡도의 개념은 [6]에서도 사용되고 있다. 여기서 기존의 혼잡도는 패킷 도착율/패킷 전송율만을 활용하나 제안 기법은 큐 점유율과 패킷 도착율/패킷 전송율을 함께 사용한다는 점에서 차이가 있다.

노드  $i$ 에서의 혼잡도  $CD(i)$ 는 다음과 같다.

$$\frac{(\alpha \cdot q_t^i / q_{th}^i) + (\beta \cdot c^i)}{\alpha + \beta} \quad (5)$$

여기서  $\alpha$ 와  $\beta$ 는 큐 점유율과 패킷 도착율 중 어느 것에 가중치를 둘 것인가를 결정하는 상수로 값의 범위는  $0 \leq \alpha, \beta \leq 1$ 이다.

## 2.2 혼잡 공지

혼잡 공지는 ECN(Explicit Congestion Notification)이나 ICN(Implicit Congestion Notification)을 사용할 수 있다. ECN은 부가적인 오버헤드를 유발하는 특별한 제어 메시지를 통해 혼잡을 공지한다. 이에 반해 ICN은 데이터 패킷의 헤더에 혼잡 정보를 피기배킹하여 전달한다. 무선 채널의 브로트캐스트 특성을 활용하여 자식 노드는 그들의 부모 노드로부터 혼잡 정보를 받을 수 있다. ICN은 이러한 방법으로 부가적인 제어 메시지의 전송 없이 혼잡 공지가 가능하다.

제안 기법에서는 각 센서 노드  $i$ 에서 ICN을 사용하며 포워드 될 데이터 패킷의 헤더에 혼잡 정보를 추가한다. 혼잡 공지는 2.1절의 혼잡 형태 중 어느 하나의 상황이라도 발생하면 즉시 전송된다.

## 2.3 혼잡 회피

혼잡 회피는 혼잡 형태중 첫 번째, 두 번째 경우가 발생할 때 수행된다. 일반적으로 혼잡도의 값이  $0.5 < CD(i) < 1$ 인 경우 현재 패킷 도착율이 높거나 큐 점유율이 임계치에 근접한 경우이므로 가까운 시간내에 혼잡이 발생할 확률이 높다. 이는 패킷 도착율과 큐 점

유율에 따라 과부하 노드를 판별하며 과부하 노드는 싱크로의 경로를 변경하여 다른 노드로 부하를 분산한다. [그림 4]에서는 혼잡 회피 알고리즘의 개념을 보이고 있다. 여기서 과부하가 발생한 노드를  $i-1$ 라 할때 혼잡 회피 알고리즘은 노드  $i-1$ 의 자식들에 대한 새로운 부모를 탐색하여 싱크로의 경로를 변경한다. 그림에서 잠재적으로 부모가 될 수 있는 노드는 자식노드들과 통신이 가능한 무선 범위내에 존재하는  $q$ 이다.

그러나 새로운 부모 노드를 찾는 데에는 두 가지 잠재적인 문제점이 존재한다.

- 1) 연쇄반응 : 연쇄반응은 자식 노드  $i$ 가 새로운 부모  $q$ 를 등록하였으나  $i$ 의 출력이 계속하여 과부하 데이터를 출력하는 경우이다. 과부하는  $q$ 에서부터 싱크 노드까지 계속하여 발생할 것이다.
- 2) 수렴 : 수렴은 몇몇 자식 노드들이 새로운 부모로 모두  $q$ 를 선택하는 경우이다. 따라서  $q$ 는 새롭게 연합한 부하들에 따라 과부하 상태가 된다.

연쇄반응은 경로를 변경하기 전에 새로운 부모의 평균 부하를 미리 검사하면 피할 수 있다. 또한 수렴은 과부하 상태인 개별 자식 노드에 의해 경로를 변경하지 않고 중앙집중적인 방식으로 부모를 선택하도록 하면 피할 수 있다.

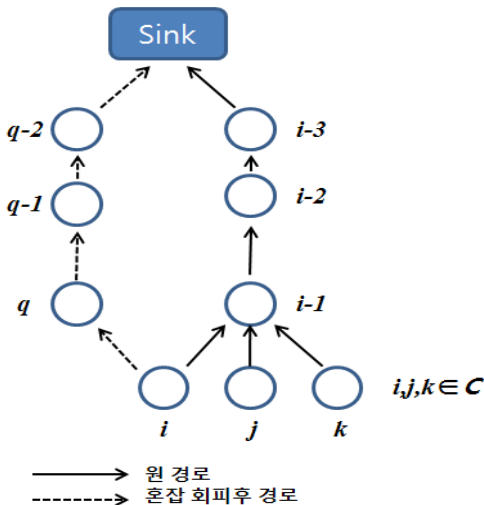


그림 4. 혼잡회피를 위한 자식 노드의 경로 변경

제안 알고리즘에서 과부하가 발생한 노드  $i-1$ 은 자식 그룹  $C$ 에게 다른 부모를 사용할 것을 요청한다. 자식 노드  $i \in C$ 는 잠재적 부모들의 집합  $Q$ 에 정보 수집을 위한 메시지를 브로드캐스트하며 루프 방지를 위해  $q \in Q$ 는  $i-1$  노드를 자신의 조상으로 가지지 않는다는 것을 확인해야 한다.  $i$ 의 혼잡도  $CD(i)$ 가 0.5 이하로  $q$ 가 수용할만한 수준이라면  $Q$ 는 자신의 평균 부하와 레벨 데이터를 전송한다. 노드  $i$ 는 혼잡도가 가장 낮은 노드  $q$ 를 자신의 새로운 부모로 등록하고 이를 통해 데이터를 전송한다.

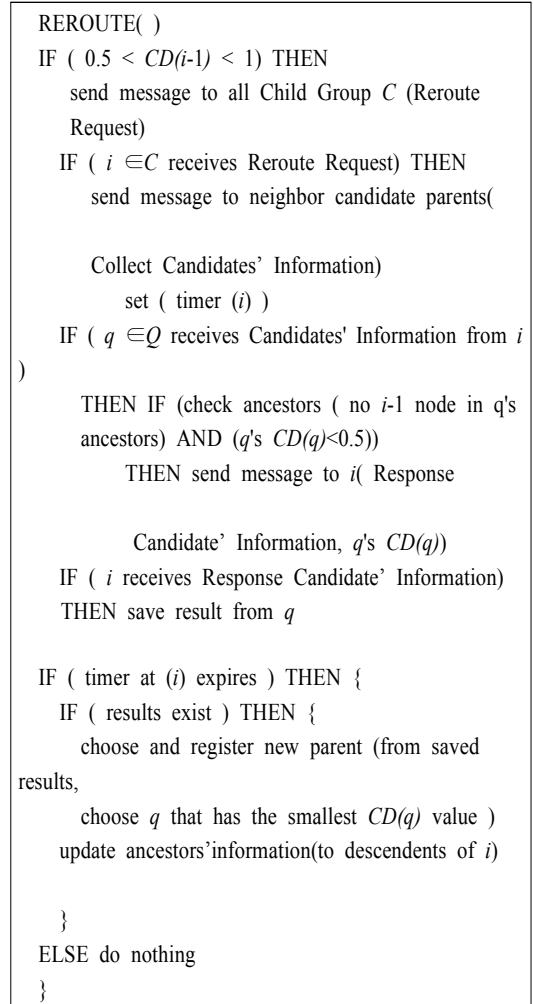


그림 5. 혼잡회피를 위한 경로 변경 알고리즘

### 2.4 비율 제어

비율 제어는 혼잡 형태 중 세 번째 경우에 수행된다. 일반적으로  $CD(i) > 1$ 인 경우 현재 큐 점유율 또는 패킷 도착율이 임계치를 넘어선 경우로 이미 혼잡이 발생한 상태이다.

비율 제어에서는 기본적으로 혼잡 회피가 실패하여 더 이상 가용한 부모 노드가 존재하지 않는다. 이러한 경우 전송되는 데이터를 감소시킬 수 밖에 없으며 스케줄러의 가중치를 변경하여 우선순위를 제어함으로써 우선순위가 낮은 패킷의 포워딩을 먼저 감소시키도록 한다. 그림 3에서 본 바와 같이 스케줄링율  $r_{svc}^i$ 는 노드의 전송율  $t_a^i$ 보다 낮을 경우 혼잡 없이 패킷을 전송할 수 있다. 따라서 혼잡이 발생할 경우 스케줄링율  $r_{svc}^i$ 를 조절하면 혼잡을 제어하거나 회피할 수 있다.

혼잡을 처리하는 데에는 TCP에서 사용되는 것과 유사한 혼잡 공지(CN)비트를 사용할 수 있다. 그러나 CN은 혼잡을 알려줄 수는 있으나 혼잡의 종류와 정도에 대한 정보는 없으므로 제한적인 정보만을 제공한다. 따라서 노드가 혼잡에 대한 정보를 받고 얼마나 비율을 조절해야 하는지를 알려주어야 하는 다른 방법이 필요하다.

제안 기법에서는 좀 더 효율적인 비율 제어를 위해 혼잡 형태와 우선순위에 대한 정보를 노드들에게 제공한다. 먼저 혼잡도  $CD(i)$ 와 우선순위  $P$ 를 통해 추가적인 정보를 제공하도록 한다. 제안 기법에서 노드는 혼잡에 대한 공지를 자신의 부모 노드로부터 수신하며 링크 수준 혼잡과 노드 수준 혼잡을 제어하기 위해 스케줄링율  $r_{svc}^i$ 와 소스 생성율  $r_s^i$ , 경유 트래픽 전송율  $r_t^i$ 를 조절하도록 한다.

스케줄링 비율 조절은 다음과 같은 절차로 수행된다. 먼저 각 노드  $i$ 는 단지 하나의 부모만을 가진다고 가정하며 노드  $i$ 는 부모 노드  $i-1$ 로부터 전달되는 패킷을 통하여 노드  $i-1$ 의 혼잡 정보를 알 수 있다. 노드  $i$ 는 이 정보를 이용해 자신의 스케줄링율과 소스 생성율을 조절한다. 초기 스케줄링율은  $r_{svc}^i$ 의 초기값으로 이는  $t_s^i$ 와 같다. 혼잡에 따른 스케줄링율의 조절은 다음의 2가지 경우를 생각해볼 수 있다.

1) 노드  $i$ 가 경유 노드로만 활용되다가 자신의 소스 트래픽을 생성하기 시작하면 패킷 도착율과 전송율의 비  $c(i-1)$ 은 증가하며 혼잡도  $CD(i-1)$ 은 증가할 것이다. 이 경우 노드  $i$ 는 자신의 스케줄링율  $r_{svc}^i$ 를 낮추어야 한다.

2) 자식노드의 집합  $C$ 는 그대로 이나 작은 트래픽을 보내던 노드들이 대량의 트래픽을 만들어 내기 시작하는 경우, 또는 새로운 소스 노드의 경유 노드로  $i$ 가 추가되는 경우를 생각해볼 수 있다. 이 경우 혼잡도는 증가하나 노드  $i$ 는 높은 링크 활용율을 유지하기 위해 스케줄링율  $r_{svc}^i$ 를 상위 노드  $i-1$ 에서 혼잡 공지가 수신될 때까지 그대로 유지한다.

상기의 두 가지 상황에 따라 노드  $i$ 의 소스 생성율과 경유 패킷의 전송율을 조절하여 스케줄링율을 구한다. 패킷의 스케줄링율은 트래픽 우선순위  $P$ 에 의해 갱신된다. 한 노드에는 두 가지의 트래픽이 있으므로 우선 순위는 소스 트래픽 우선순위  $P_s^i$ 와 경유 트래픽 우선 순위  $P_t^i$ 를 가지며 이들은 실제 스케줄러에서 가중치를 결정하기 위해 다음과 같은 로컬 우선순위  $LP_s^i$ 와  $LP_t^i$ 로 변경되어 사용된다.

$$LP_s^i = \frac{P_s^i}{P_t^i + P_s^i} \tag{6}$$

$$LP_t^i = \frac{P_t^i}{P_t^i + P_s^i} \text{ or } 1 - LP_s^i \tag{7}$$

우선순위는 패킷에 하나의 비트로 표시할 수 있으며 패킷의 종류에 따라 또는 사용자에 의해 값이 결정될 수 있다.

비율 제어에 대한 알고리즘은 다음과 같다. 먼저 현재 서비스 할 수 있는 평균 전송율을 초기 스케줄링율로 설정하고 노드의 전체 서비스율을 설정한다. 만일 혼잡이 발생하여  $CD(i-1)$ 이 증가하면 우선순위에 따른 비율제어를 수행하고 스케줄러의 서비스율  $r_{svc}^i$ 를 스케줄러 제어 상수  $\gamma (0 < \gamma < 1)$ 만큼 낮추어 상위 노드  $i$ 로 보내는 트래픽을 감소시킨다.  $CD(i-1)$ 값이 다시 혼잡 수준 아래로 감소하면 스케줄러를 리셋한다.

```

RATE_ADJUSTMENT()
total_rate = t_s^i
r_svc^i = total_rate

IF ( i ∈ C creates new traffic) THEN
    r_svc^i = r_svc^i • (1-(r_s^i/total_rate))
IF ( CD(i-1) ≥ 1) THEN{
    r_s^i = LP_s^i • total_rate
    r_t^i = LP_t^i • total_rate
    WHILE ( CD(i-1) ≥ 1)
    {
        r_svc^i = r_svc^i • γ
        check ancestor node i-1's CD(i-1) value
    }
    r_svc^i = t_s^i
}
    
```

그림 6. 비율제어 알고리즘

#### IV. 실험 및 결과 분석

##### 1. 실험 환경

시뮬레이션은 NS simulator에서 구현되었으며 사용되는 파라미터 설정은 다음의 [표 1]과 같다. 무선 대역 폭은 100kbps로 설정하였다.

표 1. 시뮬레이션 파라미터

시뮬레이션 파라미터	값
센싱 영역	60m, 60m~210m, 210m
인터페이스 대기행렬 크기	50
센서 노드의 수	50, 100
노드의 전송 거리	30.5m
패킷의 크기	64 byte
센싱 이벤트의 발생 분포	포아송 프로세스
전체 센싱 이벤트의 발생률	10~40
스케줄러의 큐 크기	40 패킷
사용한 스케줄링 기법	WRR
상수	$\alpha = 0.5, \beta = 0.5, \gamma = 0.9$

그리고 전체 네트워크의 크기는 실험의 종류에 따라 가변적으로 가정하였으며, 실험 결과는 10번 실험의 평균값이다.

##### 2. 결과 분석

본 절에서는 제안 혼잡 탐지 및 제어 알고리즘의 시뮬레이션을 통해 성능의 우수성을 증명하도록 한다. 혼잡도 CD를 구하기 위한 상수로  $\alpha=0.5, \beta=0.5$ 를 사용하였다. 이는 큐 길이와 패킷 서비스율을 동등하게 취급하여 혼잡을 탐지하도록 함을 의미한다. 비율제어를 위한 상수  $\gamma=0.9$ 로 이는 노드에서 패킷 폐기가 발생할 정도의 혼잡인 경우 CD가 1 이하로 감소할 때까지 지속적으로 10%씩 노드의 서비스율을 낮추도록 하였다.

첫 번째 실험은 네트워크 효율성에 대하여 실험해 보았으며, 네트워크 효율성은 다음과 같이 정의된다.

$$\eta = \frac{\sum_{u \in U} hops(u)}{\sum_{p \in P} \sum_{i \in hops(p)} xmits(p, i)} \quad (8)$$

여기서 U는 유용한 패킷의 집합, P는 모든 패킷의 집합, hops(p)는 패킷 p가 차지하는 각 홉 간의 범위, xmits(p, i)는 hop i에서 패킷 p의 전송 수이다.

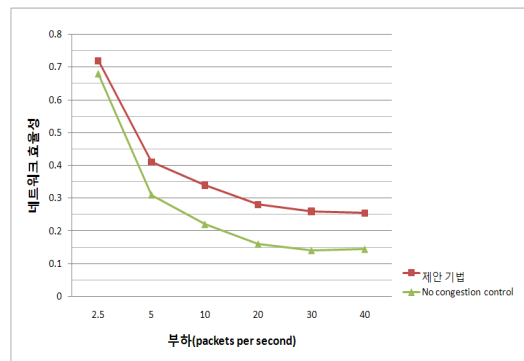


그림 7. 네트워크 효율성 η

[그림 7]은 네트워크 효율성 η를 제안 기법과 혼잡 제어가 없는 경우에 대하여 비교한 것이다. 그림에서 두 가지 기법 모두 센서 부하의 증가에 따라 효율성은



감소한다. 이는 전송 패킷 수가 많아 질수록 패킷의 오류나 재전송이 증가하며 MAC 계층의 경쟁이 증가하기 때문으로 볼 수 있다. 따라서 네트워크 효율성은 부하의 증가에 따라 감소하는 경향을 보이게 된다.

다음은 큐 점유율에 대한 것으로 제안 기법과 혼잡 제어가 없는 경우를 비교하여 보았다. 여기서 큐 사이즈는 40패킷을 저장할 수 있다고 가정하였으며 제안 기법에서  $q_{th}=20$ 과 30패킷으로 실험해 보았다. 두 경우 모두 혼잡 제어가 없는 경우에 비해서 평균 큐 길이가 감소하며 이는 오버플로우로 인한 패킷의 폐기 확률을 낮추어 주고 현재 노드와 상위 노드의 부하를 감소시킬 수 있다. [그림 8]에서는  $q_{th}=20$ 인 경우 큐 점유율이 매우 낮게 나오며 이는 낮은 큐 임계치 값으로 인해 경로 변경이 자주 발생하여 부하를 분산시키기 때문이다. 잦은 경로 변경은 지연과 에너지 소모를 유도하므로 제안 알고리즘을 활용할 경우 가능한 높은 큐 임계치를 정하여 사용하는 것이 좋다는 것을 알 수 있다.

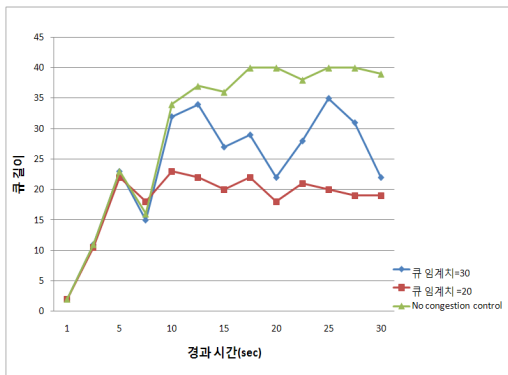


그림 8. 큐 점유율 비교

[그림 9]와 [그림 10]에서는 전체 네트워크 센서 노드의 스케줄링 큐에서 폐기된 패킷의 수와 부하에 따른 평균 지연 시간을 보이고 있다. 여기서 제안 기법은 우선순위가 높은 패킷과 우선 순위가 낮은 패킷을 구분하여 표시하였다.

먼저 [그림 9]를 보면 제공된 부하에 따라 큐의 크기를 넘어서 패킷들이 버퍼 오버플로우로 인해 폐기된 수를 보이며 부하의 증가에 따라 자연스럽게 패킷의 폐기도

증가하게 된다. 다만 전체 네트워크의 부하가 증가할수록 혼잡 제어가 없는 경우와 제안 기법을 사용한 경우의 차이가 감소하는 경향을 볼 수 있다. 이는 혼잡회피로 처리할 수 없을 정도의 과부하의 경우 노드의 스케줄링 큐의 서비스율이 우선순위에 따라 감소하게 되므로 패킷의 폐기는 피할 수 없기 때문이다.

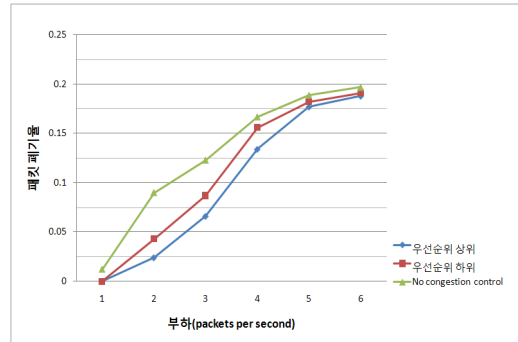


그림 9. 패킷 폐기율 비교

[그림 10]은 부하에 따른 평균 패킷 지연을 나타낸 것으로 제안 기법은 우선순위에 따라 지연시간을 나타내었다. 지연시간은 제안기법에서 우선순위가 낮은 패킷의 경우 혼잡제어가 없는 경우보다도 더 증가하게 되는데 이는 큐에 여유가 있어도 잦은 경로 변경을 수행하므로 경로변경에 따른 지연, 최적 경로의 사용 감소 등의 이유로 지연은 증가하기 때문이다. 이에 반해 우선순위가 높은 패킷은 과부하시 스케줄링 큐에서 더 높은 서비스율로 패킷을 전송하므로 지연시간이 감소하게 된다.

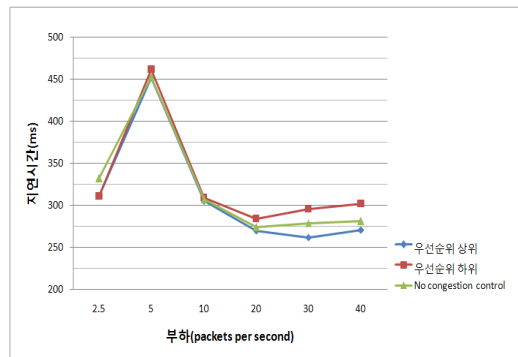


그림 10. 지연시간 비교

다음의 [그림 11]은 혼잡 제어를 수행하면서 전송율의 관점에서 트래픽 간의 공평성을 살펴본 것이다. 공평성을 평가하는 일반식은 다음과 같다.

$$\phi = \frac{\left(\sum_{i=1}^n x_i\right)^2}{n \sum_{i=1}^n x_i^2} \quad (9)$$

여기서  $x_i$ 는  $i$ 번째 센서로부터 전달된 패킷의 전송 성공율이며  $n$ 은 네트워크 내의 센서 개수이다. [그림 11]은 네트워크 부하의 증가에 따른 공평성의 변화를 혼잡제어가 없는 경우와 비교한 것으로 제안 기법이 패킷 손실을 감소시키면서 우선순위에 따른 스케줄링을 수행하여 더 우수한 성능을 보임을 알 수 있다.

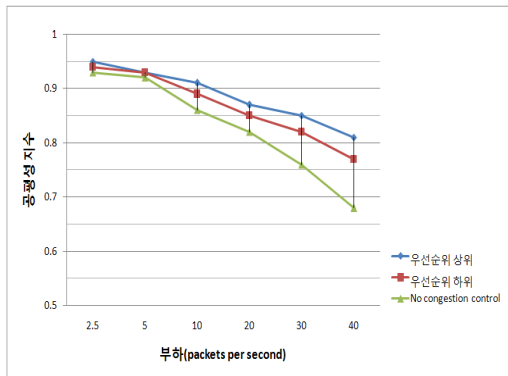


그림 11. 공평성 비교

### V. 결론

유비쿼터스 환경의 발전과 더불어 최근 많은 관심을 받고 있는 센서 네트워크에서는 시간 제약과 에너지 제약이 가장 중요한 연구 이슈로서, 본 논문에서는 센서 네트워크의 혼잡을 감소시켜 이러한 문제를 해결할 수 있는 혼잡 탐지 및 혼잡 제어 기법에 대한 연구를 수행하였다.

본 논문에서는 큐 점유율과 평균 패킷 도착율/패킷 전송율을 기반으로 하여 혼잡 탐지를 수행하게 하였으며 이는 현재 혼잡의 정도와 앞으로 혼잡이 발생할 가

능성을 예측하기 위함이다. 이를 통하여 혼잡도  $CD(i)$ 를 구하고 혼잡도를 활용하여 현재의 혼잡 정도와 혼잡 회피가 필요한가에 대한 결정을 내리게 된다.

혼잡 회피는 혼잡이 발생한 노드의 자식 노드에서 싱크 노드로의 경로를 재라우팅하여 혼잡을 감소시킨다. 혼잡도가 높은 경우에는 재라우팅으로 혼잡을 해결하기 어려운 경우로서 노드 스케줄러의 서비스율을 우선순위에 따라 조정하여 혼잡을 처리하였다. 우선 순위는 경우 트래픽과 소스 트래픽의 두 가지 트래픽 패킷에 지정될 수 있으며 우선 순위에 따라 서비스율을 제어하여 우선 순위가 높은 패킷이 더 작은 지연과 패킷 손실을 가지도록 하였다.

시뮬레이션을 결과 본 논문에서 제안한 혼잡 탐지 및 제어 기법은 큐의 상태와 패킷 도착율 등의 간단한 계산만 필요로 하므로 구현이 쉽고 실시간 특성을 우수한 성능으로 지원할 수 있음을 보였다. 특히 부하를 분산시켜 에너지 소모가 특정 경로, 또는 특정 노드에 집중되는 것을 방지할 수 있으며 우선 순위 제어에 효율적임을 알 수 있었다.

센서네트워크는 다양한 어플리케이션들을 가지며 각 어플리케이션마다 특성화된 혼잡 제어 또는 스케줄링 알고리즘이 필요할 수도 있다. 본 논문에서 두 가지 트래픽에 대해서만 우선순위를 제공하고 스케줄러는 WRR을 사용하였으나 일반적인 센서 네트워크 응용에서는 제안 기법의 특성만으로도 충분히 요구사항을 만족시킬 수 있을 것이다.

향후 연구 방향으로는 다양한 형태로 구현되는 혼잡 탐지와 제어 기법들의 특성 분석이 필요하며, 목적에 따라 적합한 알고리즘을 유연하게 선택할 수 있도록 도와주는 연구가 필요할 것이다.

### 참고 문헌

[1] C. Chi and M. Hatler, "Wireless Sensor Networks Mass Market Opportunities," ON World, 2004.  
 [2] D. Estrin, D. Culler, K. Pister, and G. Sukhatme,

"Connecting the Physical World with Pervasive Networks," Pervasive Computing IEEE, Vol.1, No.1, pp.59-66, 2002.

[3] National Research Council, "Embedded Everywhere: A Research Agenda for Networked Systems of Embedded Computers," National Academy Press, 2001.

[4] Frost and Sullina, "Wireless Sensors and integrated wireless Sensor Networks," Technical insights, 2002.

[5] Moufida Maimour, Congduc Pham, and Julien Amelot, "Load Repartition for Congestion Control in Multimedia Wireless Sensor Networks with Multipath Routing," Proc. of ISWPC 2008, Santorini, Greece, 2008.

[6] Rekha Chakravarthi, C. Gomathy, Suraj K. Sebastian, and Pushparaj. K, "A Survey on Congestion Control in Wireless Sensor Networks," International Journal of Computer Science & Communication, Vol.1, No.1, pp.161-164, 2010(1).

[7] Y. Sankarasubramaniam, Özgür B. Akan, and Ian F. Akyildiz, "ESRT: event-to-sink reliable transport in wireless sensor networks," Proc. of MobiHoc, pp.1003-1016, 2003(10).

[8] C. Y. Wan, S. B. Eisenman, and A. T. Campbell, "Congestion detection and avoidance in sensor networks," Proc. of SenSys, pp.266-279, 2003(11).

[9] B. Hull, K. Jamieson, and H. Balakrishna, "Mitigating congestion in wireless sensor networks," Proc. of SenSys, pp.134-147, 2004(11).

[10] S. Chen and Z. Zhang, "Localized algorithm for aggregate fairness in wireless sensor networks," Proc. of MobiCOM, pp.274-285, 2006(9).

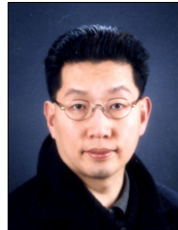
[11] S. Chen and N. Yan, "Congestion avoidance

based on lightweight buffer management in sensor networks," Proc. of ICPADS, pp.934-946, 2006(9).

저 자 소 개

이 대 윤(Dae-Woon Lee)

정회원



- 1985년 : 미국, NYIT Graduate School, Computer Graphics Dept. 졸업
- 2009년 : 충북대학교 컴퓨터공학과 박사수료

<관심분야> : 센서네트워크, 디지털방송

이 태 우(Tae-Woo Lee)

정회원



- 2004년 2월 : 충북대학교 컴퓨터공학과(공학석사)
- 2003년 12월 ~ 2008년 4월 : (주)LG전자 단말연구소 선임연구원
- 2010년 2월 : 충북대학교 컴퓨터공학과 (박사수료)

- 2008년 9월 ~ 현재 : 충북대학교 컴퓨터공학과 강사

<관심분야> : 센서네트워크, 유비쿼터스 네트워킹

최 승 권(Seung-Kown Choi)

정회원



- 2001년 2월 : 충북대학교 컴퓨터공학과 공학박사
- 2009년 3월 ~ 2010년 2월 : 충북대학교 전자정보대학 겸임교수
- 2009년 5월 ~ 현재 : (주)네트리스치엔컨설팅 연구소장

<관심분야> : 센서네트워크, 유비쿼터스 네트워킹

이 준 석(Joon-Suk Lee)

정회원



- 2007년 8월 : 충북대학교 컴퓨터 공학과 공학석사
- 2010년 2월 : 충북대학교 컴퓨터 공학과(박사수료)

<관심분야> : HCI, 유비쿼터스 네트워킹

김 광 훈(Jin Guangxun)

정회원



- 2009년 8월 : 충북대학교 컴퓨터 공학과 공학석사
- 2009년 9월 ~ 현재 : 충북대학교 컴퓨터공학과 박사과정

<관심분야> : USN, 3D Animation

이 재 엽(Jae-Youp Lee)

정회원



- 2010년 2월 : 충북대학교 컴퓨터 공학과(박사수료)
- 2004년 3월 ~ 현재 : (주)팬지아 21 E-컨버전스사업부 이사

<관심분야> : USN, E-컨버전스