

무선 센서 네트워크에서 에너지 효율적인 데이터 전송을 위한 스케줄링/ MAC 통합 프로토콜

정희원 조재규*, 권태경*, 최양희*

A Joint Wakeup Scheduling and MAC Protocol for Energy Efficient Data Forwarding in Wireless Sensor Networks

Jaekyu Cho*, Taekyoung Kwon*, Yanghee Choi* *Regular Members*

요약

미래 인터넷 환경에서 무선 센서 네트워크(Wireless Sensor Networks)는 광범위한 응용에 적용될 중요한 기술이다. 그러나 센서 네트워크는 기존의 네트워크와는 달리 소형이고 제한된 배터리 에너지로 운용되어야 하기 때문에 센서 네트워크가 설치된 지역에서 오랜 시간동안 많은 정보를 얻기 위해서는 에너지 효율적인 프로토콜 개발이 필수적이다. 또 센서 네트워크는 무선 멀티 홉 (multi hop) 통신을 통해 정보가 전달되기 때문에 전송 지연이 발생한다. 이런 전송 지연은 센싱된 정보가 적시에 전달되어 적절한 조치가 이루어지는 것을 어렵게 하기 때문에 센서 네트워크의 프로토콜을 설계할 때 전송 지연이 발생하지 않도록 해야 한다. 본 논문은 무선 센서 네트워크에서 일정 시간의 전송지연을 유지하면서 각 노드의 에너지의 소비를 최소화하는 데이터 전달 프로토콜을 제안한다. 제안된 프로토콜은 wakeup 스케줄링과 MAC 프로토콜을 통합하여 동작하며, 데이터를 전달하는 동안 네트워크 전체의 에너지 소모를 균등하게 해 준다. 제안된 프로토콜은 모의실험을 통해서 기존의 프로토콜과 비교하여 성능을 검증하였다.

Key Words : Wireless sensor network, Wakeup/sleep scheduling, Data forwarding protocol

ABSTRACT

Under future internet environment, wireless sensor networks will be used in a wide range of applications. A major problem for designing sensor protocol is developing the most energy efficient technique to monitor an area of interest for a long time since sensors have some constraints such as small and a limited energy level. In addition, data latency is often a critical issue since sensory data is transmitted via multi hop fashion and need to be delivered timely for taking an appropriate action. Our motivation for designing a data forwarding protocol is to minimize energy consumption while keeping data latency bound in wireless sensor networks. In this paper, we propose a data forwarding protocol that consists of wakeup scheduling and MAC protocols, the latter of which is designed to achieve load balancing. Simulation results show that the proposed framework provides more energy-efficient delivery than other protocol.

I. 서론

무선 센서 네트워크 (Wireless Sensor Networks)

는 센싱 (sensing) 및 통신능력을 가진 다수의 센서 노드를 관심지역에 배치하여 그 지역의 주변 환경이나 발생한 이벤트 (events) 등의 정보를 수집하여

※ 본 연구는 지식경제부 및 정보통신 연구진흥원의 IT 신성장 동력핵심기술 개발사업 [2007-F-038-02, 미래인터넷]과 21세기 프론티어 연구개발사업의 일환으로 추진되고 있는 지식경제부의 유비쿼터스컴퓨팅 및 네트워크 원천기반기술 개발사업[08B3-B3-10M]의 과제 로 지원되어 수행되었습니다.

* 서울대학교 전기컴퓨터공학부 멀티미디어/이동통신 연구실 (jkcho@mmlab.snu.ac.kr, tkkwon@snu.ac.kr, yhchoi@snu.ac.kr)
논문번호 : KICS2007-12-565, 접수일자 : 2007년 12월 13일, 최종논문접수일자 : 2008년 3월 12일

사용자에게 전달해 주는 네트워크이다. 무선 센서 네트워크는 기존의 애드 혹(ad hoc) 네트워크와 달리 센서 노드는 소형이고 넓은 지역에 높은 밀도로 배치된다. 또 배터리로 작동되어서 노드의 수명이 길지 않고, 낮은 주파수 대역을 사용하므로 네트워크 토폴로지의 변화가 자주 발생하는 특성을 가지고 있다. 그러나 이러한 특성에도 불구하고 무선 센서 네트워크는 최근 유비쿼터스(ubiquitous) 컴퓨팅 환경에서의 중요한 요소로 연구가 활발히 진행되고 있으며, 전쟁지역, 화재 현장, 공장 자동화, 자연환경 감시 등의 다양한 응용에 적용되고 있다. 이러한 다양한 응용에 적용되기 위해서는 센서 네트워크의 프로토콜을 설계할 때 에너지 효율성, 데이터 전송 지연(latency), 처리율(throughput), 네트워크 공정성(fairness) 등의 사항을 중요하게 고려하여야 한다.

특히 센서 노드는 배터리를 교체하거나 충전하는 것이 제한되는 지역에 배치되기 때문에 네트워크의 전체 수명을 오랫동안 유지하기 위한 에너지 효율성과 전장 감시 및 화재 감시등의 응용에서처럼 센싱된 정보가 사용자에게 신속하게 전달되어 적절한 행동을 취할 수 있도록 하는 데이터 지연은 아주 중요한 고려사항이라 할 수 있다. 그러나 에너지 효율성과 전송 지연 사이에는 서로 상반되는 측면(tradeoff)이 존재한다. 센서 네트워크 프로토콜에서 각 노드의 에너지 소비를 절약하기 위해 사용되는 가장 일반적인 방법은 노드가 통신할 때에는 wakeup 상태를 유지하여 데이터를 전송하고, 통신하지 않을 때에는 sleep 상태를 유지하는 wakeup/sleep 스케줄링 방법이 있다. 그러나 이런 wakeup/sleep 스케줄링 방법은 데이터 전송지연을 증가시킨다. 왜냐하면, 데이터 전송을 해야 하는 송신 노드는 데이터를 수신해야 하는 수신 노드가 깨어날 때까지 기다려야 하기 때문이다. 반대의 경우, 전송지연을 줄이기 위해 노드들이 항상 깨어있으면 그 동안의 에너지 사용량이 많아져 노드의 수명을 짧게 한다.

본 논문에서는 이러한 문제점을 줄이기 위해 일정 시간의 전송지연을 유지하면서 에너지 소모를 최소화할 수 있도록 wakeup/sleep 스케줄링과 MAC 프로토콜을 통합한 데이터 전송 프로토콜을 제안한다. 제안된 프로토콜을 다음 몇 가지의 특징을 가진다. 첫째, 미리 정해진 wakeup/sleep 스케줄링에 의해서 데이터가 소스(source) 노드로부터 싱크(sink)까지 순차적으로 전달되기 때문에 전송지연이 감소된다. 이 때 송신 노드와 수신 노드는 언제 깨어나

서 통신을 해야 하는지에 대한 정보를 스케줄링 기법에 의해 알 수 있다. 둘째, 특정 노드로 트래픽(traffic)이 집중되는 것을 방지하기 위해서 데이터를 전달할 때마다 전달 노드(relaying node)를 교체한다. 이것은 특정 노드가 많은 데이터를 전송함으로써 다른 노드에 비해 빨리 에너지를 소모하는 것을 방지하여 네트워크 전체의 에너지 소모를 균등하게 해 준다. 마지막으로, 제안된 프로토콜은 센서 노드의 하드웨어적 고장 등으로 인해 데이터를 전송받을 다음 홉(hop)의 이웃 노드가 없을 경우에도 동적으로 자신의 wakeup/sleep 스케줄을 바꾼 뒤에 데이터를 전달하기 때문에 네트워크 토폴로지가 자주 변하는 상황에서도 적용할 수 있다.

본 논문의 구성은 다음과 같다. II장에서는 관련 연구로 기존에 제시된 wakeup/sleep 스케줄링 기법과 MAC 프로토콜에 대해서 설명하고 III장에서는 제안하는 데이터 전달 프로토콜에 대해서 제안한다. IV장에서는 모의실험을 통한 성능평가 결과를 보여주고 V장에서는 본 논문의 결론을 맺는다.

II. 관련 연구

최근 센서 네트워크의 에너지 소비 또는 데이터 전송지연을 줄이기 위해 wakeup/sleep 스케줄링 또는 MAC 프로토콜을 바탕으로 하는 많은 연구들이 진행되어 왔다¹⁻⁷. 특히 에너지 소비를 절약하기 위해서 가장 일반적으로 사용되는 접근방법은 센서 노드의 대기상태(idle) 시간을 줄이는 것이었다⁸⁻¹¹.

[8]에서는 SMAC이라는 센서 MAC 프로토콜이 제안되었다. SMAC에서는 모든 노드가 대기상태 시간동안의 에너지 소비를 줄이기 위해 주기적으로 wakeup/sleep 한다. 이웃 노드와 통신을 하기 위해서 각 노드는 서로의 스케줄을 교환하여 동기화한다. 그러나 SMAC에서는 각 홉마다 데이터를 전송하기 위해서는 수신 노드가 깨어날 때 까지 기다려야 하기 때문에 데이터 전송지연이 크게 일어난다. 이런 단점을 보완하기 위해 이웃 노드들의 통신을 overhear 하고 있다가 자신이 전송받을 시간이 되면 깨어나는 adaptive listening 메커니즘을 제안하였지만, 데이터 전송지연을 줄이는 데는 한계가 있다.

ASAP (Adaptive Sleeping and Awakening Protocol)^[9]에서는 네트워크의 동기화를 위해 이웃 노드들의 주소 또는 위치와 같은 식별자(identifier) 정보를 유지한다. 각 노드는 획득한 식별자 정보를 해쉬(hash)함수를 이용하여 이웃 노드들의

wakeup/sleep 스케줄을 계산한다. 이렇게 계산된 스케줄을 이용하여 이웃 노드가 깨어날 때 송신 노드는 데이터를 전송한다. ASAP에서는 주변의 노드들이 불필요한 데이터 전송을 받지 않기 때문에 이에 따른 에너지 소모를 줄일 수도 있고, 데이터 전송을 받을 준비가 되어 있는 가장 적절한 수신노드를 선택할 수 있기 때문에 전송지연을 최소화 할 수 있는 경로를 선택할 수 있다. 하지만, 이웃 노드들의 정보를 유지하기 위해서는 주기적으로 정보를 교환해야 하기 때문에 통신 오버헤드(overhead)가 크고, 이웃 노드가 하드웨어적 결함이나 이동으로 인해 통신에 참여할 수 없을 경우는 큰 데이터 전송지연이 발생하는 단점이 있다.

[10]에서는 각 센서 노드는 싱크로부터의 홉 카운트를 기준으로 데이터 경로를 설정한다. 설정된 경로에 따라 각 노드의 스케줄이 정해지고, 데이터는 순차적으로 전달된다. 각 노드는 데이터를 전송할 때만 전원을 on하고 그 외에는 off 하여 에너지 소비를 줄인다. 그러나 센서가 하드웨어 고장을 일으키거나 에너지를 다 소모하여 데이터를 전달할 수 없을 경우에 데이터 경로 설정 및 wakeup/sleep 스케줄을 다시 설정해야 하는 단점이 있다. 또 데이터 전달할 때, 미리 선정된 노드에 의해서만 전달되기 때문에 에너지 소비의 불균형을 초래하여 전체적인 네트워크의 수명을 단축한다.

[11]에서는 데이터 수집 트리(data gathering tree)를 이용하여 높은 에너지 효율성과 낮은 전송지연을 지원하는 DMAC이 제안되었다. 네트워크 셋업 초기에 경로설정을 위해 데이터 수집 트리를 구성하고 구성된 트리에 의해 노드는 자신의 activity 스케줄을 결정한다. 스케줄의 active 슬롯(slot)은 송신과 수신으로 나누어진다. 수신 슬롯에서는 자신의 자식 노드로부터 데이터를 전송받아서 송신 슬롯에 부모 노드로 데이터를 전송한다. DMAC에서는 전송지연을 줄이기 위해 More-to-Send 라는 메커니즘을 제안하고 있는데, 다른 노드의 전송에 방해가 되지 않게 하기위해서 세 번의 타임슬롯 후에 다시 active 상태를 유지하여 자식 노드로부터 추가 데이터를 전송받는다. 하지만, DMAC에서는 소스 노드로부터 싱크까지 데이터를 전송하는 것만 고려하고 있고, 싱크에서 소스 노드로 쿼리(query)를 전송하는 방법은 고려하고 있지 않다. 또 트리에 의해 데이터가 전송되기 때문에 특정 노드에게 트래픽이 집중 될 경우 빠른 에너지 소모로 인해 트리가 단절되는 단점이 있다.

III. 스케줄링/ MAC 통합 데이터 전송 프로토콜

본 장에서는 제안하는 에너지 효율적인 데이터 전송 프로토콜을 위한 시스템 모델을 먼저 설명하고, 프로토콜의 기본적인 동작방법에 대해 기술한다.

3.1 시스템 모델

무선 센서 네트워크는 다수의 센서 노드와 하나의 싱크로 구성되며, 다수의 센서 노드가 수집된 정보를 멀티 홉 통신을 통해 싱크에게 전달한다. 이 때 배치된 다수의 노드는 싱크로부터의 홉 거리를 기준으로 클래스(class)라는 집합에 포함되며 class_id라는 식별자를 가지게 된다. 그림 1은 센서 노드와 클래스의 관계를 나타낸 것이다.

제안된 프로토콜의 wakeup/sleep 스케줄링을 위해 다수의 타임슬롯(timeslot)으로 구성된 타임프레임(timeframe)을 정의한다. 타임프레임은 센서로부터 싱크까지 데이터가 전달되는 업스트림(upstream) 타임프레임과 싱크로부터 센서까지 쿼리가 전달되는 다운스트림(downstream) 타임프레임으로 구분된다. 업스트림 타임프레임은 m_u 개의 타임슬롯을 가지고, 다운스트림 타임프레임은 m_d 개의 타임슬롯을 가진다. 다수의 업스트림 타임프레임과 한 번의 다운스트림 타임프레임은 일정한 주기를 가지고 반복되는데 이 주기는 응용의 요구사항에 따라 달라질 수 있다. 노드는 정해진 타임슬롯에 wakeup하여 데이터를 송/수신하거나 에너지를 절약하기 위해 sleep한다. 타임슬롯의 길이는 하나의 메시지를 전송하고 이에 대한 ACK (Acknowledgement) 을 받을 수 있는 충분한 시간으로 정한다. 그림 2는 센서 노드의 class_id가 3이고 m_u 가 8일 때 업스트림 타임프레임의 구조를 나타낸 것이다.

네트워크 내의 시간 동기화는 이루어진 것으로 가정한다. 채널 페이딩(channel fading)에 의한 어려

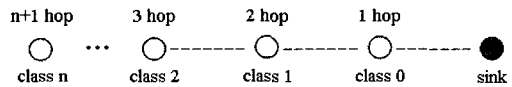


그림 1. 센서 노드의 클래스 설정

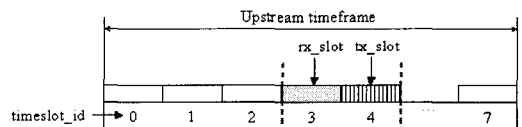


그림 2. 업스트림 타임프레임의 구조

는 없다고 가정하며, 단지 충돌에 의한 전송에러만을 고려한다. 모든 무선 링크는 양방향으로 통신이 가능하며, 모든 센서 노드는 동일한 통신 장비를 소유하고 있고, 동일한 전송거리를 가지는 것으로 가정한다.

3.2 프로토콜의 동작

본 절에서는 제안한 데이터 전송 프로토콜의 기본 동작에 대해서 설명한다. 제안하는 전송 프로토콜은 네트워크 설정단계, 센서 노드로부터 싱크로 데이터를 전송하는 업스트림 전송단계, 싱크로부터 센서 노드로 쿼리를 전송하는 다운스트림 전송단계로 구분된다. 네트워크 설정 후 응용의 요구사항에 맞게 여러 번의 업스트림 전송단계를 진행하고 이후 한 번의 다운스트림 전송단계를 진행한다. 추가적으로 네트워크 복구절차를 설명한다.

3.2.1 네트워크 설정단계

초기 네트워크 설정단계에서 싱크는 전체 네트워크로 셋업 (setup) 메시지를 전송한다. 이 셋업 메시지는 싱크의 ID, 시퀀스 (sequence) 번호, 타임스케줄 정보가 들어있다. 시퀀스 번호는 싱크가 새로운 셋업 메시지를 전송할 때마다 증가하는 값으로 이전에 보냈던 셋업 메시지와 구분을 할 수 있게 해준다. 타임 스케줄 정보에는 업스트림/다운스트림의 타임프레임 정보, 타임프레임별 타임슬롯의 개수 (m_u, m_d), 업스트림 단계에서 다운스트림 단계로 전환 시기 등이 포함되어 있다. 그림 3은 업스트림 단계에서 다운스트림 단계로 전환을 보여준다.

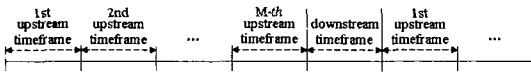


그림 3. 업스트림/ 다운스트림 전환

3.2.2 업스트림/다운스트림 전송단계

업스트림 단계에서 센서 노드는 싱크로부터의 홉 카운트를 바탕으로 자신의 업스트림 class_id를 다음과 같이 계산한다.

$$class_id_u = (hop\ count - 1) \bmod m_u \tag{1}$$

자신의 class_id를 계산한 후, 노드는 업스트림 타임프레임의 타임슬롯 중 자신이 깨어나야 할 액티브

(active) 타임슬롯을 정한다. 액티브 타임슬롯은 데이터를 송신하는 tx_slot과 데이터를 수신하는 rx_slot으로 구분된다. class_id가 n 인 센서 노드의 업스트림 tx_slot과 rx_slot은 다음과 같이 계산한다.

$$tx_slot_u = m_u - n - 1 \tag{2}$$

$$rx_slot_u = (tx_slot_u - 1) \bmod m_u$$

그림 2는 class_id가 3이고 m_u 가 8인 센서 노드의 업스트림 tx_slot과 rx_slot을 보여준다. class_id_u, tx_slot_u, rx_slot_u을 계산한 후, 센서 노드는 쿼리에 해당하는 사건을 감지하면 자신의 tx_slot에 데이터를 전송한다. 전송할 데이터가 있는 센서노드는 자신의 tx_slot에서 매체경쟁에 승리한 후 다음 클래스에 속한 노드들 중 릴레이(relay) 노드를 찾기 위해 요청(request) 메시지를 전송한다. 이 때, 요청 메시지에는 노드 ID, 송신노드의 class_id, NAV(Network Allocation Vector) 등이 포함되어 있다. 자신의 rx_slot에 있는 다음 클래스 노드들은 릴레이 노드가 되기 위해서 매체경쟁을 한다. 매체 경쟁에서 승리한 노드는 릴레이 노드가 되어서 응답(response) 메시지를 송신 노드에게 전달한다. 성공적으로 요청, 응답 메시지가 교환되면, 송신 노드는 릴레이 노드에게 즉시 데이터를 전송하고 ACK을 받는다. 그림 4는 업스트림 단계에서 데이터가 전달되는 과정을 나타낸 것이다. 그림에서 보는 바와 같이 class_id=4인 노드 i는 timeslot_id=2에서 데이터를 전송받아서 timeslot_id=3에서 class_id=3인 노드 j에게 데이터를 전송한다. 노드 j도 같은 방법으로 노드 k에게 데이터를 전달하며, 마지막 싱크까지 순차적으로 데이터가 전달된다. 이런 방법으로 데이터가 전달되면 별도의 전송지연이 발생하지 않고서도 데이터가 소스로부터 싱크까지 전달된다.

다운스트림 단계에서는 업스트림 단계에서와 비슷하게 다운스트림 class_id, tx_slot, rx_slot을 다음과 같이 계산한다.

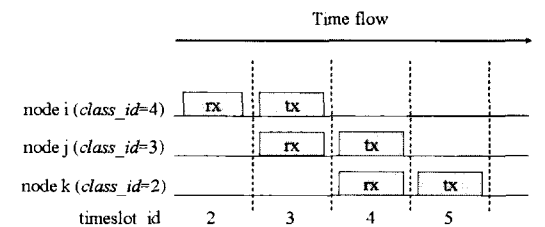


그림 4. 데이터 릴레이 과정

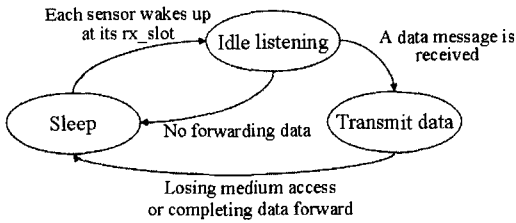


그림 5. 센서 노드의 상태전이 다이어그램

$$\begin{aligned} class_id_d &= hop\ count - 1 & (3) \\ tx_slot_d &= n + 1 \\ rx_slot_d &= n \end{aligned}$$

계산된 $class_id_d$, tx_slot_d , rx_slot_d 을 통해서 업스트림과 동일한 방법으로 싱크에서 모든 센서 노드로 쿼리를 전달한다. 다운스트림 단계에서는 업스트림 단계와 달리 쿼리를 전송할 때는 플러딩 (flooding) 방법을 사용하고, 쿼리 ID를 통해서 중복된 쿼리를 구분한다.

그림 5는 업스트림/다운스트림 타임프레임 내에서 센서 노드의 상태전이 다이어그램을 나타낸 것이다. 센서는 sleep 상태에서 자신의 rx_slot이 되면 깨어난다. 이 때 전송할 데이터가 없으면 에너지 소비를 줄이기 위해서 다시 sleep 상태로 돌아간다. 전송할 데이터가 있는 센서 노드는 자신의 tx_slot에 데이터를 전송한다. 만약 데이터 전송을 위해 매 경쟁에서 승리하지 못한 경우에는 즉시 sleep 상태로 들어가고 다음 타임프레임에서 데이터를 전송한다.

3.2.3 네트워크 복구단계

무선 센서 네트워크에서는 하드웨어 고장 또는 에너지 자원의 부족 등으로 인해 네트워크 토폴로지의 변화가 자주 발생한다. 제안하는 데이터 전송 프로토콜에서도 토폴로지의 변화로 인해 데이터를 전달할 다음 클래스의 노드가 없는 경우가 발생한다. 이런 문제점을 해결하기 위해 다음과 같은 네트워크 복구절차를 제안한다.

송신 노드가 자신의 tx_slot에서 요청 메시지를 전송한 후에 응답 메시지를 받지 못하면 재전송 제한횟수까지 요청 메시지를 재전송한다. 만약 재전송 제한횟수까지 요청 메시지를 전송한 후 응답 메시지를 받지 못하면 노드는 다음 클래스 노드가 없다고 판단하고 자신의 class_id를 하나 증가시킨다. 이후에는 증가된 class_id의 wakeup 스케줄에 따라 자신의 rx_slot, tx_slot를 재조정하고 데이터를 전송한다.

IV. 성능 평가

본 논문에서는 모의실험을 통하여 제안된 프로토콜의 성능평가를 수행하였으며, 기존의 프로토콜인 DMAC의 성능과 상호 비교분석하였다. 이를 위해 모의실험 도구인 NS-2^[12]를 사용하였다.

4.1 실험 환경

표 1은 모의실험에 사용된 파라미터 값을 보여준다. 모의실험에서 설정된 센서지역의 크기는 $100m \times 100m$ 이며, 센서 노드의 수는 100개이다. 센서 노드는 지역 내에 랜덤하게 배치되고 이동성은 없으며 싱크는 지역 내 중앙에 배치되었다. 데이터를 전송하는 소스 노드는 싱크로부터 5홉 떨어진 4개의 노드를 랜덤하게 선택하였다. 각 센서 노드의 통신거리는 10m이고 물리계층의 대역폭은 802.15.4의 표준을 참조하여 250kbps이다. 송신, 수신 및 대기상태의 평균 사용전력은 0.66 W, 0.395 W, 0.35 W이다. 하나의 타임프레임은 8개의 타임슬롯으로 구성되어 있으며, 제안된 프로토콜의 타임슬롯의 크기는 35.1 milliseconds이고, DMAC에서의 슬롯크기는 33.8 milliseconds이다. 왜냐하면 DMAC에서는 제안된 프로토콜과 달리 요청 및 응답 메시지를 사용하지 않기 때문이다. DMAC의 라우팅 프로토콜은 DMAC 논문에서 사용한 simple tree 라우팅 프로토콜을 사용하였다. 데이터와 컨트롤 메시지의 크기는 각각 1,024bytes, 10bytes이며 데이터 전송률은 초당 0.5에서 1.1로 설정되었으며 버퍼의 크기는 5개의 데이터 메시

표 1. 모의실험에 사용된 파라미터

파라미터	값
센서지역의 크기	100m × 100m
노드의 수	100
통신거리	10m
물리계층 대역폭	250 kbps
송신/ 수신/ 대기 상태 전력	0.66 W / 0.395 W / 0.35 W
타임프레임의 크기	8타임슬롯
타임슬롯의 크기	35.1 milliseconds (제안된 프로토콜) 33.8 milliseconds (DMAC)
데이터/ 컨트롤 메시지 크기	1,024bytes / 10bytes
데이터 전송률	0.5 ~ 1.1 (message/second)
버퍼크기	5 데이터 메시지

지를 저장할 수 있다. 그래프에 표시된 실험 데이터는 10회 이상 실시한 후 평균값이며, 이에 대한 표준편차를 표시하였다.

4.2 성능결과

본 절에서는 데이터 전송률을 변화시키면서 기존의 프로토콜과 제안한 프로토콜의 성능을 비교 분석하였다.

그림 7은 데이터 전송률에 따른 데이터 전송지연 시간을 나타낸 것이다. 그림 7에서 보는 바와 같이 제안된 프로토콜과 DMAC은 데이터 전송률이 0.5 ~ 1.1 까지는 일정한 시간으로 데이터가 전송되는 것을 보여준다. 이러한 이유는 제안된 프로토콜과 DMAC 모두 데이터 전달시 전송경로 상에 위치한 노드들이 순차적으로 깨어나서 데이터를 전달하기 때문이다. 그리고 전송률 1.1 이하까지는 설정된 네트워크의 용량을 초과하지 않은 트래픽이므로 큐잉에 의한 전송지연이 발생하지 않았다. 제안된 프로토콜이 DMAC보다 조금 높은 전송시간을 보여주고 있는데 이는 제안된 프로토콜의 타임프레임의 크기가 DMAC보다 더 크기 때문이다. 하지만 전송지연 시간의 차이가 거의 나지 않기 때문에 전송시간 측면에서는 동일한 성능을 보인다고 할 수 있다.

그림 8은 데이터 전송률이 변화함에 따라 전체 네트워크의 에너지 소모량을 나타낸다. 그림 8에서 보는 바와 같이 제안된 프로토콜이 DMAC 보다 30~40% 정도의 에너지 소모량이 적은 것을 알 수 있다. DMAC의 경우는 자식 노드들 중에서 데이터 전송을 하지 못한 노드들을 위해서 매 4번의 타임슬롯마다 깨어나서 수신모드에 들어가기 때문에 더 많은 에너지를 소모하기 때문이다. 그러나 제안된 프로토콜은 송신 슬롯에서 전송할 데이터가 없거나

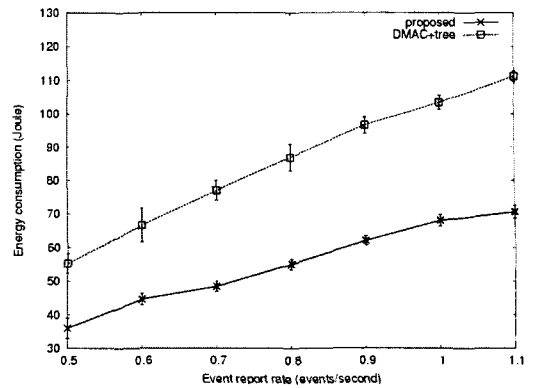


그림 8. 에너지 소모량

수신 슬롯에서 요청 메시지를 받지 못하면 즉시 sleep 상태로 들어가기 때문에 불필요한 에너지 소모를 하지 않는다.

그림 9는 배치된 노드 중 제일 먼저 자신의 에너지를 소모하여 동작을 멈춘 시간을 나타낸 것이다. 그림 9에서 보는 바와 같이 데이터 전송률이 증가함에 따라 노드들은 자신의 에너지를 빨리 소모한다. 특히 DMAC의 경우 트리에 의한 라우팅으로 인해 각 소스 노드로부터 싱크까지의 데이터 전송 경로가 고정되어 있고, 중간 노드의 경우 여러 개의 자식 노드로부터 많은 데이터를 전송받기 때문에 빠른 에너지 소모를 보이고 있다. 이러한 경우, 중간 노드는 에너지가 완전히 소모되어 데이터를 전달할 수 없고 네트워크의 기능을 중단시킨다. 그러나 제안된 프로토콜에서는 데이터를 전송할 때 데이터 전달에 참여하는 중간 노드들이 고정되어 있지 않고 데이터를 전달할 때마다 중간노드가 변하기 때문에 한 노드가 집중적으로 에너지를 소모하는 경우가 발생하지 않는다. 또 중간노드가 에너지를 완전히 소모하더라도 네트워크 복구절차에 의해

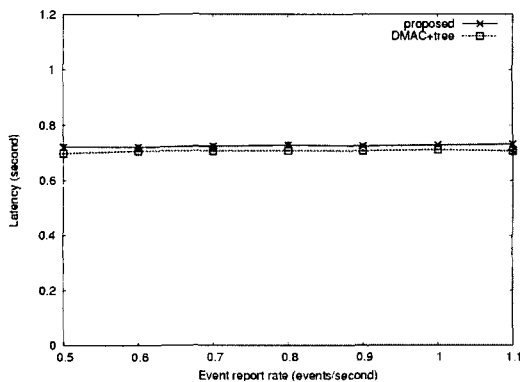


그림 7. 데이터 전송지연 시간

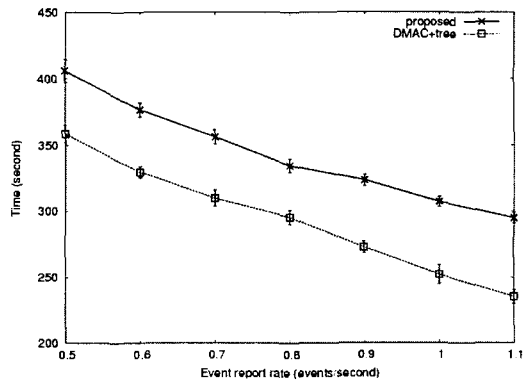


그림 9. 가장 먼저 에너지를 소모한 노드의 동작시간

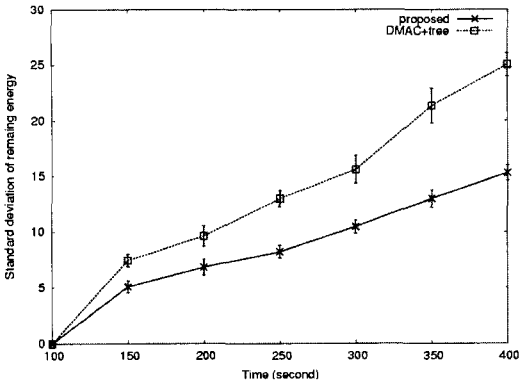


그림 10. 노드의 잔여 에너지 표준편차

같은 클래스의 다른 노드로 데이터가 전달되어 네트워크의 수명을 증가시킨다.

그림 10은 실험이 종료될 때까지 센서 노드의 남아있는 에너지의 표준편차를 나타낸 것이다. 그림 10에서 보는 바와 같이 DMAC의 경우에는 남아있는 에너지의 편차가 크게 발생하여 네트워크의 효율성이 떨어지는 것을 볼 수 있다. 이는 아직 동작할 수 있는 센서노드가 많음에도 불구하고 네트워크의 기능이 종료된 것으로 그림 9에서 설명한 바와 같이 네트워크의 부하가 분산되지 못하고 일부 노드에게 집중되기 때문이다. 그러나 제안된 프로토콜은 대부분의 노드가 데이터 전달에 참여하여 네트워크의 부하가 특정 노드에게 집중되지 않고 여러 노드에 분산함으로써 각 노드의 에너지 소모를 줄이고 결과적으로 네트워크의 수명을 증가시키는 것을 알 수 있다.

V. 결 론

본 논문은 작은 전송지연을 유지하면서 에너지 소모를 최소화할 수 있는 wakeup/sleep 스케줄링과 MAC 프로토콜을 통합한 데이터 전송 프로토콜을 제안하고 있다. 제안된 프로토콜은 미리 정해진 wakeup/sleep 스케줄링에 의해서 데이터가 소스 노드로부터 싱크까지 순차적으로 전달되기 때문에 적은 전송지연을 유지한다. 또한 특정 노드로 트래픽이 집중되지 않게 하여 다른 노드에 비해 빨리 에너지를 소모하는 것을 방지하고 네트워크 전체의 에너지 소모를 균등하게 해주어 네트워크의 수명을 연장하게 해준다. 더불어 네트워크 복구 기법을 제안하여 센서 노드의 하드웨어적 고장 등으로 인해 네트워크 토폴로지가 동적으로 변하더라도 데이터를 전송하게 해 주는 장점이 있다. 모의실험을 통하여

기존 프로토콜과 비교를 통해 제안된 기법이 데이터 전송 지연, 에너지 소모율, 부하의 균등분배 등에서 더 효율적인 성능을 보여주고 있다.

본 논문의 연구는 센서 네트워크에서 데이터 전달 프로토콜을 설계하는데 좋은 지침이 될 것으로 판단되며 향후 연구에는 수학적 분석을 통하여 제안된 프로토콜의 효율성을 증명하고, 다양한 통신능력을 가진 센서 노드들에 의해 구성된 센서 네트워크에서 데이터를 전달하는 프로토콜에 대해서 연구할 것이다.

참 고 문 헌

- [1] M. Heissenbuttel, T. Braun, M. Walchli and T. Bernoulli, "Optimized Stateless Broadcasting in Wireless Multi-hop Networks," In proc. Infocom 2006, Apr. 2006.
- [2] S. Cui, R. Madan, A. Goldsmith and S. Lall, "Cross-layer Energy Minimization in TDMA-based Sensor Networks," In proc. 42th-Allerton Conference on Communication, Control, and Computing, Sep. 2004.
- [3] Y. Ghiassi-Farokhfal, M. R. Pakravan, "Cross-layer Flooding for Sensor Networks without Location Information," In proc. The 2nd IEEE International Conference on Mobile Ad-hoc and Sensor Systems, Nov. 2005.
- [4] B. Tavli and W. Heinzelman, "PN-TRACE: Plain Network Wide Broadcasting Through Time Reservation Using Adaptive Control For Energy Efficiency," In proc. The IEEE Military Communications Conference, Oct. 2004.
- [5] T.V. Dam and K. Langendoen, "An Adaptive Energy-Efficient MAC Protocol for Wireless Sensor Networks", In proc. The First ACM Conference on Embedded Networked Sensor Systems (Sensys'03), Nov. 2003.
- [6] Z. Chen and A. Khokhar, "Self Organization and Energy Efficient TDMA MAC Protocol by Wake Up For Wireless Sensor Networks," In proc. IEEE SECON 2004, Oct. 2004.
- [7] S. Kulkarni, A. Iyer and C. Rosenberg, "An Address-light, Integrated MAC and Routing Protocol for Wireless Sensor Networks," In IEEE/ACM Transactions on Networking, Aug.

- 2006.
- [8] W. Ye, J. Heidemann and D. Estrin, "An Energy-Efficient MAC Protocol for Wireless Sensor Networks," In proc. INFOCOM 2002, New York, Jun. 2002.
 - [9] K. Balachandran, J. H. Kang and W. Lau, "Adaptive Sleeping and Awakening Protocol (ASAP) for Energy Efficient Adhoc Sensor Networks," In proc. ICC 2005, May. 2005.
 - [10] M. L. Sichitiu, "Cross-Layer Scheduling for Power Efficiency in Wireless Sensor Networks," In proc. Infocom 2004, Mar. 2004.
 - [11] G. Lu, B. Krishnamachari and C. Raghavendra, "An Adaptive Energy-Efficient and Low-Latency MAC for Data Gathering in Wireless Sensor Networks," In proc. IPDPS 2004, Santa Fe, Apr. 2004.
 - [12] <http://www.isi.edu/nsnam/ns/>

조 재 규 (Jaekyu Cho)

정회원



1994년 2월 금오공과대학교 기계공학과 졸업
2004년 1월 국방대학교 전산정보학과 졸업
2005년 3월~현재 서울대학교 전기컴퓨터공학부 박사과정
<관심분야> 센서 네트워크, 무선 이동 통신

권 태 경 (Taekyoung Kwon)

정회원



1993년 2월 서울대학교 컴퓨터공학과 졸업
1995년 2월 서울대학교 컴퓨터공학과 석사
2000년 2월 서울대학교 컴퓨터공학과 박사
2002년 3월~현재 서울대학교 전기컴퓨터공학부 교수

<관심분야> 센서네트워크, 유비쿼터스 컴퓨팅

최 양 희 (Yanghee Choi)

정회원



1975년 2월 서울대학교 전자공학과 졸업
1977년 2월 한국과학기술원 석사
1984년 2월 프랑스 ENST 전산학 박사
1991년 3월~현재 서울대학교 전기컴퓨터공학부 교수

<관심분야> 미래 인터넷, 멀티미디어 통신