

유비쿼터스 센서 네트워크의 매체 접근 제어 기법에 대한 개선 방안

장 호[†] · 이 명 섭^{**} · 전 우 상^{***}

요 약

본 논문에서는 실시간 유비쿼터스 센서 네트워크에 적합한 MAC(media access control) 프로토콜을 제안한다. 기존의 DCF 프로토콜은 데이터 전송을 위한 슬롯(slot)을 선택할 때 패킷 충돌로 인한 재전송이 반복될 때 마다 크기가 변하는 경쟁 윈도우(contention window) 내에서 균등 확률 분포(uniform probability distribution)를 이용한 랜덤(random) 선택 기법을 사용하지만 제안한 프로토콜에서는 센서 데이터의 전송 지연을 최대한 감소시키기 위하여 경쟁 윈도우의 크기를 고정시키고, 전송 슬롯을 보다 효율적으로 선택하도록 비 균등(non-uniform) 확률 분포를 사용하여 전송 슬롯을 랜덤하게 선정한다. 제안한 방법의 성능을 입증하기 위하여 256개의 센서가 배치된 센서 네트워크 모델을 기반으로 ns-2를 이용하여 시뮬레이션을 시행하고, 전체 센서에 발생하는 전송 평균 지연 시간이 기존의 802.11 MAC 표준에 비하여 유비쿼터스 센서 네트워크의 최적 지연 한계점(best latency bound)에 가깝게 나타남을 확인한다.

키워드 : 센서 네트워크, 매체 접근 제어, 무선 네트워크, 경쟁 기반 매체 접근, 백오프 알고리즘

An improvement of Medium Access Control Protocol in Ubiquitous Sensor Networks

Ho-Jang[†] · Myung-Sub Lee^{**} · Woo-Sang Jeon^{***}

ABSTRACT

we present more efficient method of a medium access for real-time ubiquitous sensor networks. Proposed MAC protocol is like the randomized CSMA/CA protocol, but unlike previous legacy protocols, it does not use a time-varying contention window from which a node randomly picks a transmission slot. To reduce the latency for the delivery of event reports from sensor nodes, we carefully decide to select a fixed-size contention window with non-uniform probability distribution of transmitting in each slot. We show that the proposed method can offer up to several times latency reduction compared to legacy of IEEE 802.11 as the size of the sensor network scales up to 256 nodes using a widely-used network simulation package, NS-2. We finally show that proposed MAC scheme comes close to meet bounds on the best latency being achieved by a decentralized CSMA-based MAC protocol for real-time ubiquitous sensor networks which is sensitive to latency.

Keywords : Wireless Network, Sensor Network, Mac Protocol, Contention Window, Backoff Procedure

1. 서 론

일반적으로 무선 네트워크에서는 여러 노드들이 채널을 공유하여 데이터를 전송하기 때문에 이를 중재하기 위한 매체 접근 제어(MAC) 프로토콜이 필수적이다. 과거 수십 년간 다양한 MAC 프로토콜들이 개발되었고 그 중 몇몇들은 오늘날 실제 무선 네트워크에 적용되어 사용되고 있다. 하

지만 이들 프로토콜들은 전통적인 방식의 네트워크 동작에 맞추어져 있으므로 최근에 출현한 유비쿼터스 센서 네트워크에는 적합하지 못한 것이 현실이다. 이에 본 논문에서는 새로운 관점에서 유비쿼터스 센서 네트워크의 동작 메커니즘과 특성에 대하여 면밀히 분석하여 보다 적합한 MAC 프로토콜을 제안한다.

센서 네트워크 분야에서 배터리 절약 기술은 무엇보다도 중요하므로 이에 대한 연구가 예전부터 다양하게 이루어져 왔다[1-4]. 고전적인 MAC 프로토콜에 비해 유비쿼터스 센서 네트워크 MAC 프로토콜은 다음과 같은 특징이 있다.

[†] 정 회 원 : 구미1대학 컴퓨터전공 전임강사

^{**} 정 회 원 : 영남이공대학 컴퓨터정보계열 교수 (교신저자)

^{***} 정 회 원 : 중소기업기술정보진흥원 정보화경영체계 평가위원

논문접수: 2008년 7월 24일

수정일: 1차 2008년 11월 19일, 2차 2008년 11월 19일

심사완료: 2009년 1월 29일

- 대부분의 센서 노드들은 이벤트 구동 방식이며 하나의 전송 채널을 공유하기 위하여 경쟁(contention) 시스템을 기반으로 한다.
- 모든 센서 노드들이 동시에 이벤트에 반응하는 것은 아니므로 시간에 따라 센서 네트워크 내 반응 노드들의 밀도가 변화한다.
- 빈번한 전송 지연 발생은 에너지 소모량을 증가시키고 동시에 전체 센서 네트워크의 성능도 저하시킨다.

이에 본 연구에서는 위의 일반적인 특성들에 착안하여 MAC 프로토콜의 재설계를 수행함으로써 센서 네트워크 내의 데이터 전송 지연을 최대한 감소시키는 효과적인 방법을 고안하는 데 중점을 두며 정확한 센서 네트워크의 모델링과 시뮬레이션을 통하여 고안된 프로토콜이 센서들의 에너지 소모량을 감소시키고 실시간 전송 성능의 향상에 효과적임을 보여준다.

본 연구에서 제안한 무선 센서용 MAC 프로토콜의 설계상의 주안점은 베이스 스테이션으로부터 감지 데이터를 전송하도록 요청받은 다수의 노드들(N)이 전송 채널을 선점하려고 동시에 경쟁하는 경우, 이들 노드들이 최대한 충돌 없이 요청된 리포트 데이터(R)를 목적 노드에 전송하도록 하는 것이다. 전통적인 MAC 프로토콜에서는 모든 노드들이 리포트 데이터를 전송할 경우($R=N$)의 처리율(throughput)을 최적화하는 문제를 다루는 반면, 본 논문에서는 일부의 노드들에서만 리포트 데이터가 발생하는 경우($R < N$)에서 유비쿼터스 센서 네트워크의 최적화된 성능을 도출하는 효과적인 방법을 다루고 있다.

고전적인 CSMA 프로토콜에서의 각 노드들은 현재의 경쟁 윈도우(CW : contention window) 내에서 균등한 분포의 확률에 의하여 랜덤하게 하나의 슬롯을 선택하여 선택된 슬롯의 값을 일정 시간 마다 감소시키는 백오프(backoff) 과정을 거친 후 이 값이 0이 되는 시점에 데이터를 전송하게 된다. 이 방식은 충돌문제를 아주 쉽게 해결할 수 있으나, 노드 수가 급격히 증가할 경우 노드들 간의 충돌이 폭주하게 되어 이로 인해 반응 시간의 지연을 피할 수 없게 되므로 네트워크 동작이 원활하지 못하게 된다.

일반적으로 유비쿼터스 센서 네트워크에서는 초기에 발생하는 리포트 데이터의 수가 상대적으로 적으며 이때 선택되는 초기의 슬롯들은 보다 느슨한 경쟁 체제에 있다. 이로 인해 슬롯 간의 충돌이 거의 발생하지 않아 전체적으로 낮은 전송 지연 시간(latency)을 요구한다. 본 연구는 이러한 개념에 기초하여 다음과 같은 내용의 수정된 프로토콜을 제시한다. 즉, 전송 지연 시간을 최소화하기 위해 경쟁 윈도우의 크기를 일정하게 유지하고, 전송 슬롯을 선택하는 과정에서 선택 확률을 차등화하기 위해 전통적으로 사용하던 균등(uniform) 확률 분포 대신 기하 확률 분포(geometric probability distribution)를 특별히 설계하여 사용한다.

본 논문의 구성은 2장에서 관련 연구에 대하여 간단히 살

펴보고, 3장에서는 제안한 프로토콜의 동작 메커니즘을 구조적으로 설명하며, 4장에서 기존의 센서 네트워크용 MAC 프로토콜과 제안한 MAC 프로토콜의 성능을 시뮬레이션을 통하여 비교하고, 마지막으로 결론과 향후 과제를 5장에서 서술한다.

2. 관련 연구

본 절에서는 유비쿼터스 센서 네트워크에 사용되는 기존의 CSMA/CA 방식 기반의 매체 접근 제어 프로토콜의 메커니즘에 대하여 알아보고자 한다.

IEEE 802.11 standard working group에 의하면 노드들의 매체에 대한 접근 제어 방법은 MAC 부계층에 DCF(distributed coordination function) 방식과 PCF(point coordination function) 방식, 그리고 이들의 혼합 방식이 정의되어 있다[5]. DCF 방식은 IEEE 802.11 MAC에서 사용되는 기본적인 접근 방법이며 이 방식은 경쟁기반의 서비스를 제공하며 접근 방법으로 랜덤 백오프 알고리즘(random backoff algorithm)을 사용한다. 반면, PCF 방식은 선택적으로 사용되는 방법으로서 AP(access point)와 같은 중재기에 의한 폴링(polling) 기법을 이용하여 노드들의 채널 점유 방법을 관리한다.

DCF 방식에서의 백오프 알고리즘은 일정한 시간 간격마다 전송을 위하여 선택한 슬롯의 번호를 감소시켜 그 값이 0이 될 때 전송을 시작하는 방법이다. 그러므로 시간이 충분히 흐른 시점에서는 경쟁 윈도우의 전반부에 경쟁 노드들이 집중되어 있을 가능성이 크므로 경쟁 윈도우 내에서 전반(前半)부에 위치한 작은 번호의 슬롯들이 후반(後半)부에 위치한 큰 번호의 슬롯들에 비해서 충돌 확률이 상대적으로 증가하게 되는 것이다. 즉 이전 단계에서 이미 슬롯을 할당받아 백오프 절차를 진행해오던 노드들과 이전 단계에서 충돌로 인하여 백오프 절차가 취소됨으로써 현 단계에서 다시 슬롯을 할당받아야 하는 노드들, 그리고 현 단계에서 새로 서비스를 요청받아 백오프 절차를 새로 시작하는 노드들이 경쟁 윈도우 내의 전반부에 상대적으로 집중되게 되어 시간에 따른 경쟁 과열 현상이 발생하기 때문이다. 이러한 단점 때문에 기존의 DCF방식에서의 백오프 방법은 전체 통과율을 저하시키고 패킷의 전송 지연이 늘어나게 된다.

본 논문에서는 효율적인 슬롯 선택 확률을 제안함으로써 이러한 문제를 극복함과 동시에 제안한 메커니즘이 실시간 유비쿼터스 센서 네트워크에 보다 효과적으로 적응함을 보이고자 한다.

2.1 DCF에서의 랜덤 백오프 타임

경쟁 기반 서비스인 DCF 방식에서는 매체 최소 비사용(idle) 시간인 DIFS(distributed interframe space)를 이용하여 전송 매체의 가용 여부를 판단한다. 프레임 전송이 완료되고 DIFS가 경과된 후에는 경쟁 윈도우가 나타나게 되는

데 이는 다수의 슬롯들로 나뉘며 슬롯 하나의 길이 (SlotTime)는 매체 의존적이며 고속 물리 계층일수록 더 짧다. 매체 접근을 위하여 경쟁하는 노드들은 랜덤으로 슬롯 하나를 선택하여 일정 시간을 기다린 후 매체를 점유하게 되는데 이때 모든 슬롯의 선택 확률은 균등하다. 슬롯이 랜덤하게 선택됨으로써 백오프 타이머(backoff timer)는 $[random(i) \times SlotTime]$ 크기의 지연 시간을 발생시키며 이를 랜덤 백오프 타임이라고 한다.

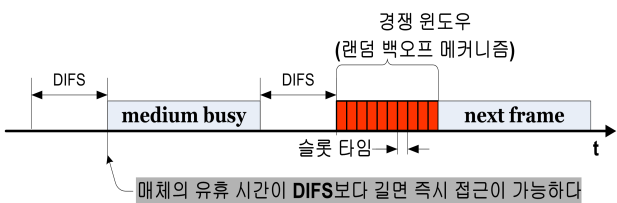
이러한 백오프 알고리즘은 다중 노드 간의 매체 점유를 위한 경쟁 상황에서 노드 간의 충돌을 미연에 회피하여 이를 최소화하기 위한 방법으로 사용된다.

2.2 DCF의 백오프 절차를 이용한 경쟁 기반 접근 방법

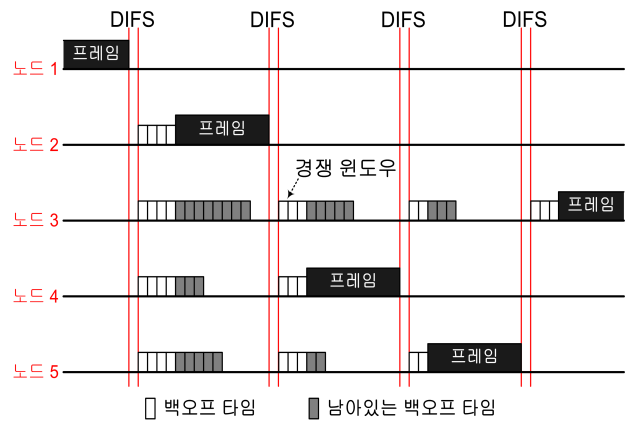
일반적으로 (그림 1)에 나타난 바와 같이 DCF 접근 환경 하에서는 DIFS 기간 이상으로 매체의 유휴(idle) 상태가 지속되면 전송이 임박한 노드의 MPDU(MAC protocol data unit) 부터 즉시 전송할 수 있다.

만약 이러한 조건에서 반송과 감지 메커니즘에 의하여 매체가 사용 중이라고 감지되었을 경우에는 랜덤 백오프 알고리즘에 의해서 경쟁 윈도우의 크기가 바뀌게 되며(보통의 경우에는 이전 단계의 2배가 된다), MPDU를 전송하거나 혹은 재전송하게 되더라도 MPDU와 ACK 신호의 전송을 허락하기에 시간이 충분하지 않다면 노드는 선택된 랜덤 백오프 타임에 의하여 전송을 연기하게 된다[6].

DCF 방식에서 이루어지는 백오프 절차는 전송 노드가 전송에 실패하였다고 추측될 때 실시된다. 모든 백오프 슬롯들은 매체가 DIFS 기간 동안에 대하여 이상적이라고 판단될 때 다음 기간에 발생한다. 만약 매체가 유휴 상태에 있음이 감지된다면 백오프 동작은 SlotTime에 의해 백오프 타임을 줄이는 절차를 수행한다. 반면에 매체가 백오프 과정 동안 사용 중이라고 판단되면 백오프 타임은 줄어들지 않고 멈추어 있다. 최종적으로 데이터 전송은 백오프 타이머가 0이 될 때마다 시작된다[6]. (그림 2)는 DCF 백오프 절차에 의해 5개의 노드들이 경쟁하는 상황을 나타낸다. 이러한 백오프 과정을 통하여 다중 노드들이 전송을 연기하거나 랜덤 백오프 동작으로 진입할 때 마다 랜덤 함수에 의해 결정된 최소의 백오프 타임을 선택하게 된다. 이전의 사실로부터 노드들이 매체 접근 경쟁에서 이겨 데이터 전송에 성공하기 위한 기본 조건은 최소의 백오프 타임을 획득하는 것임을 알 수 있다.



(그림 1) IEEE 802.11에서 DCF를 이용한 접근 방식



(그림 2) DCF에서의 백오프 절차

결과적으로 경쟁에 참여한 다수의 노드들 중에서 충돌을 최대한 피할 수 있는 유일한 방법은 중복되지 않은 백오프 타임을 할당 받는 것이다. 하지만 경쟁에 참여한 노드들의 수가 급격히 증가할수록 기존의 균등 확률 분포에 기반을 둔 802.11 계열의 MAC 프로토콜에서는 중복되지 않은 백오프 타임을 선택할 확률이 대체적으로 낮아지게 된다.

2.3 경쟁 기반 접근 방식의 센서 네트워크용 MAC 프로토콜의 예

현재까지 센서 네트워크를 위해 제안된 MAC 프로토콜들은 CSMA/CA를 활용한 경쟁기반 방식, TDMA를 활용한 스케줄기반 방식, 여러 개의 채널을 활용하는 방식, 이들 방식을 혼합한 하이브리드 방식 등으로 분류될 수 있다.

여기에서는 경쟁기반 방식의 센서 네트워크 MAC 프로토콜들 중 대표적인 BMAC과 SMAC에 대하여 장단점을 중심으로 간단히 살펴보고자 한다.

BMAC

BMAC 프로토콜에서는 RTS/CTS와 ACK 등과 같은 제어 기능이 상위 계층에서 수행되도록 하여 프로토콜의 구조를 가급적 간단하게 하였으며, 유휴 청취에 의한 전력소모를 감소시키기 위하여 휴면 구간 보다 긴 프리앰블 구간을 샘플링하는 저전력청취(LPL: low power listening) 방법을 사용한다.

하지만, BMAC에는 자체의 RTS/CTS 기능과 프레임 분할 기능, 그리고 주기적인 사이클 동기화 과정이 없기 때문에 청취 시작 시점이 고정되어 있지 않다. 또한 숨겨진 단말기 문제의 해결이나 다중 패킷 전송기능, 프레임 분할 기능, 그리고 동기화 과정을 지원하지 않으므로 필요한 경우 SMAC이나 TMAC 등이 상위 계층에 추가되어 함께 수행되어야 한다[7].

SMAC

SMAC은 무선 센서 네트워크를 위해 고안된 방법으로 슬롯 개념을 사용하고 단일 채널을 사용하는 경쟁기반의 MAC 프로토콜이다. 또한, 주기적으로 청취 구간과 휴면 구간을 반복하는 센서 노드들의 사이클에 기반을 두고 있으며, 노드들 간의 사이클을 동기화하기 위한 가상 클러스터링 작업이 추가로 필요한 프로토콜이다[8, 9].

SMAC은 기본적으로 시간을 프레임 단위로 나누고, 이 프레임을 청취 구간과 휴면 구간으로 나눈다. SMAC의 휴면 구간에서는 센서 노드가 무선 통신을 위한 전원을 끄고 전력을 거의 소모하지 않는 상태를 유지하고, 전력소모를 줄이기 위해 청취 구간의 듀티사이클을 가능한 줄이는 방법을 사용한다.

그러나 전송율과 지연, 그리고 전력소모율 간의 절충점을 찾기 위한 세밀한 조정이 어렵다는 문제가 있다. SMAC 프로토콜의 장점은 무엇보다도 유휴 청취에 의한 전력소모를 휴면 모드를 이용하여 감소시키고 구현이 쉽다는 것이다. 반면 RTS/CTS를 사용하지 않고 브로드캐스트되는 데이터 패킷들은 충돌확률이 높아지며, 사전에 고정된 휴면 구간과 청취 구간의 크기로 인하여 가변적인 트래픽 부하 환경에서는 효율성이 떨어지는 단점이 있다[2].

3. 수정된 MAC 프로토콜

802.11[5, 6], BMAC[7], SMAC[8, 9], MACAW[10]와 같은 CSMA 기반의 프로토콜에서는 노드 간 충돌이 발생할 때 마다, 충돌 노드들의 경쟁 윈도우 크기를 2배씩 증가시키는 BEB(binary exponential backoff) 방법을 사용한다. 다시 말하면 노드 간의 충돌을 최소화할 만큼 충돌 노드의 경쟁 윈도우 크기를 현재의 활성(active)화된 노드 수에 적용된 크기로 증가시키는 방법이다. 이러한 방법은 다음과 같은 문제점들을 안고 있다. 첫째, 유휴 기간 이후 많은 수의 센서들이 이벤트를 동시에 기다리는 상황처럼 감지된 데이터를 전송할 준비가 된 활성 노드의 수(N)가 많아지면 경쟁 윈도우가 이를 수용할 수 있는 적절한 크기로 증가되는데 많은 시간이 요구된다. 둘째, 활성 노드의 수가 적음에도 불구하고 이전의 트래픽 충돌로 인해 경쟁 윈도우의 크기가 필요 이상으로 커져 있다면 이후의 백오프 과정에 사용되는 대역폭이 불필요하게 허비될 수밖에 없다. 즉, 대부분의 CSMA 프로토콜들은 활성화된 모든 노드가 충돌을 피하여 데이터를 전달하는 방법에만 초점을 맞추고 있으므로 비효율적이라고 할 수 있다.

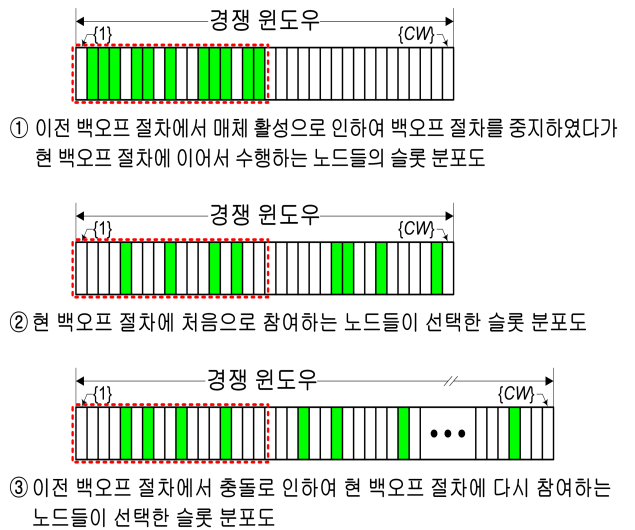
그러므로 본 논문에서는 노드간의 충돌로 발생하는 기존 프로토콜의 문제점을 해결하기 위한 보다 효율적인 동적 충돌 회피 기법으로서 경쟁 윈도우의 크기를 최소(현재 32개의 슬롯)로 고정하고, 활성화된 노드의 수에 능동적으로 대처할 수 있는 비 균등 확률 분포에 기반을 둔 경쟁 슬롯 선택

시스템을 제안한다. 이전의 CSMA 기반 무선 MAC 프로토콜들과 비교하여 제안한 프로토콜의 전송 슬롯 선택 방법은 원천적으로 차등화된 확률에 의하여 슬롯을 선택함으로써 슬롯의 중복 선택률을 최소화하는 것이 가장 큰 차이점이다.

3.1 경쟁 슬롯을 선택하는 기존 방법의 문제점

일반적인 802.11의 MAC에서 경쟁 윈도우 내의 모든 슬롯들은 모두 동일한 확률로 선택된다. 그러므로 많은 수의 센서 노드들이 매체 접근을 시도하는 경쟁 상태에서는 현재의 경쟁 사이클에 새롭게 참여하는 노드들이나 이전 사이클에서의 충돌로 인하여 다시 참여하는 노드들 모두 크기가 변화된 경쟁 윈도우 내에서 같은 확률로 하나의 슬롯을 선택하게 된다. 이러한 슬롯 선택 메커니즘은 경쟁 윈도우 크기의 변화에도 불구하고 시간이 경과함에 따라 전체적으로 비어있는 슬롯(empty slot)의 선택 확률을 점차 떨어뜨리게 된다.

(그림 3)에서와 같은 백오프 절차의 경우를 가정해 보면, 먼저 가장 작은 번호의 슬롯($\{WinSlot_{i-1}\}$)이 채널을 할당받아 데이터 전송을 성공적으로 수행하게 될 것이고, 전송에 성공한 슬롯 이후의 나머지 슬롯들은 백오프 절차를 종료하지 못하여 다음 백오프 절차에 다시 참여할 수밖에 없다. 이때는 이들 잔여 슬롯들이 자연적으로 경쟁 윈도우의 전반부로 이동하게 되는데, 여기에 백오프 절차에 새롭게 참여하는 노드들과 이전 경쟁에서 충돌로 인해 다시 백오프 절차를 수행해야 하는 노드들이 추가로 진입함으로써 경쟁 상황이 더욱 가중되는 불가피한 결과가 발생하게 된다. 이와 같은 상황에서는 많은 경쟁 슬롯들 중 하나의 전송 슬롯을 선택하기 위하여 균등 확률 분포에 의한 기존의 랜덤 방식을 이용한다면 노드 간의 충돌 가능성은 심각하게 높아질 수밖에 없다.



(그림 3) 일반적인 BEB 방법에서의 슬롯 선택 확률 변화

결과적으로 유비쿼터스 센서 네트워크가 고(高)부하상태라고 가정한다면 경쟁 윈도우 크기의 한계성과 슬롯 선택 확률의 균등성으로 인해 경쟁 슬롯들의 중복 선택률이 전체적으로 높아지게 된다. 특히 백오프 절차가 진행됨에 따라 경쟁 슬롯들이 경쟁 윈도우의 전반부로 몰리는 현상이 초래되고 이로써 노드 간의 충돌이 자주 발생하게 되는데, 이는 경쟁 윈도우 크기를 반복적으로 증가시킬 수밖에 없는 가장 큰 원인이다. 이러한 구조적인 문제점으로 인해서 매체 접속을 위해 경쟁하는 센서 노드들의 백오프 타임은 대체적으로 길어지고 각 센서들의 전송 지연 시간과 전력 소비율이 갑자기 높아지는데, 이로써 센서 네트워크의 전체 성능은 급격하게 저하되게 된다.

3.2 경쟁 슬롯 선택 방법의 수정

앞에서 언급한 문제점들에 대한 해결책으로 본 논문에서는 슬롯 선택 확률 분포를 차등화하는 방법을 제안한다. 문제 해결을 위한 기본적인 개념은 충돌을 최대한 줄이는 동시에 백오프 타임을 최소화할 수 있는 슬롯을 선택하는 확률을 경쟁 윈도우 내의 위치에 따라 상대적으로 높여주는 것이다.

이를 위하여 우선 경쟁 윈도우의 크기를 고정시킨 상태에서 다른 센서 노드들에 의해 선택되지 않고 비어있는 슬롯을 찾아내는 방법을 생각해보자. (그림 4)에서와 같이 백오프에 이용되는 경쟁 윈도우 내에서 임의의 슬롯 하나가 전송에 성공한다면 이 슬롯의 위치를 기준으로 이전 위치의 슬롯들은 절대 선택되지 않아야 한다. 다시 말하면 전송에 성공한 슬롯을 기준으로 이후에 위치한 슬롯들만이 의도적으로 선택될 수 있는 확률 분포 함수를 설계하는 것이 필요하다. 이러한 성질의 확률 분포 함수는 (그림 4)에 나타난 바와 같이 전체 슬롯들에 대하여 임의의 슬롯을 기준으로 이전 슬롯들의 선택되지 않을 확률과 이후 슬롯들의 선택될 확률을 곱함으로써 유도할 수 있다.

설계된 확률 분포 함수의 세부적인 유도과정을 살펴보면, 먼저 현재의 경쟁 윈도우 내에 경쟁이 없는 경우라면 {1}번

째 슬롯을 선택하면 가장 좋은 성능을 얻을 수 있다(그림의 첫 번째 스트림, $i_{win} = 1$ 인 경우). 만일 {1}번째 슬롯이 이미 다른 노드에 의해 선택되었다면 {2}번째 슬롯을 선택하는 것이 가장 좋다. 이와 같은 슬롯 선택 방식에서 { i }번째 슬롯이 선택되어 있다는 의미는 그 이전의 모든 슬롯들은 다른 노드들에 의해 이미 선택되었음을 의미한다. 그러므로 최소한의 지연 시간으로 충돌 없이 데이터를 전송하기 위해서는 { $i+1$ }번째 슬롯을 선택하면 된다.

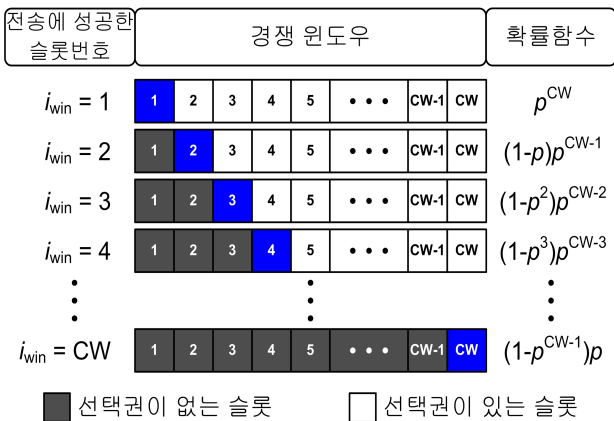
이러한 최적의 선택 방법을 유지하기 위하여 임의의 슬롯을 기준으로 이전 슬롯들의 선택되지 않을 확률과 이후 슬롯들의 선택될 확률을 이용하여 확률 분포 함수를 유도하여 보면, 각 센서 노드가 경쟁 윈도우 범위 내($i \in [1, CW]$)에서 { i }번째 슬롯을 선택할 확률 $f(i)$ 는 모수 p 인 기하분포 (geometric distribution)를 따른다고 할 수 있으므로 그 확률 질량 함수(probability mass function)를 다음 식(1)과 같이 도출할 수 있다.

$$f(i) = \begin{cases} (1-p^{i-1})p^{CW-i+1}, & i = 1, \dots, CW \\ 0, & \text{기타} \end{cases} \quad (1)$$

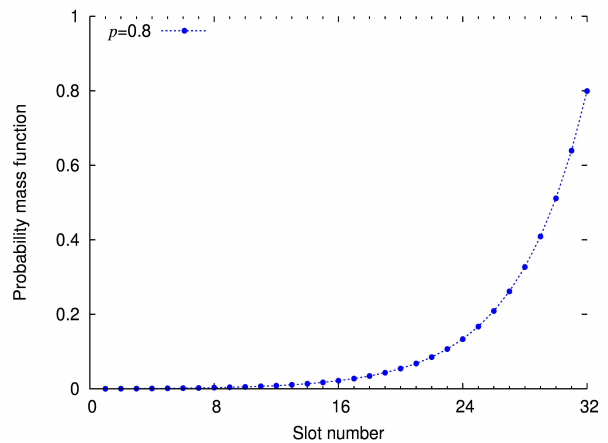
결과적으로 제안한 확률 질량 함수 $f(i)$ 는 (그림 5)에서와 같이 $0 < p < 1$ 의 범위에서 i 가 증가할수록 기하급수적으로 증가하는 형태를 띠며, 이는 노드들에 의해 특정 슬롯이 선택될 확률은 경쟁 윈도우 내의 후반부로 갈수록 전반부에 비해 상대적으로 높아짐을 의미한다.

식 (1)에서 p 는 매개 변수(distribution parameter)로써 슬롯이 비어있을 확률을 간접적으로 나타내며 활성 노드 수 (N)에 의해 결정된다.

여기에서 { i }번째 슬롯을 선택할 확률(S_i)을 가상의 결정 과정을 통해 계산해보자. 첫 번째 단계의 활성 노드의 수를 N_1 이라고 하고 이들 중 하나의 노드가 {1}번째 슬롯을 선택한다고 가정하자. 만일 {1}번째 슬롯을 선택한 노드가 하



(그림 4) 노드 간 경쟁을 최소화하기 위한 슬롯 선택 확률



(그림 5) 경쟁 윈도우 내의 슬롯 번호에 따른 확률 분포

나도 없다면 두 번째 단계에서 N_1 을 N_2 로 감소시키고 {2} 번째 슬롯을 선택한다. 계속적으로 {2} 번째 슬롯도 선택되지 못했다면 N_2 를 N_3 로 감소시키고 동일한 과정을 단계적으로 반복한다. 전체 과정에서 $[1, \dots, i-1]$ 의 슬롯들이 모두 비어있지 않다면 최종적으로 N 은 N_i 로 감소됨을 짐작할 수 있다.

정리하면 $\{i\}$ 번째에서의 선택 확률은 $N_i = N_1$ 일 때 가장 높으며, 단계가 진행됨에 따라 N_i 는 N_1 에서 1로 일정하게 감소되는 성질이 있다. 그러므로 N_i 가 상대적으로 크고 S_i 가 작을 경우 $\{i\}$ 번째 이전까지의 슬롯들이 모두 비어있지 않은 상태에서 오직 하나의 센서에 의해 $\{i\}$ 번째 슬롯이 선택될 확률은 다음과 같다고 할 수 있다.

$$N_i S_i (1 - S_i)^{M-1} \cong N_i S_i e^{-N_i S_i} \quad (2)$$

위의 식에서 $N_i S_i$ 를 상수처럼 취급한다면 $\{i\}$ 번째 슬롯 선택 확률($N_i S_i e^{-N_i S_i}$)은 마지막 슬롯으로 가면 갈수록 점진적으로 감소됨을 나타낸다. 다시 말해서 작고 고정된 크기의 경쟁 윈도우로 많은 수의 매체 접근 노드(N)에 효율적으로 대응하기 위하여 매체 접근 노드의 수를 일정한 비율(Δ)로 줄여나가는 방식을 채택한 것이다.

$$\frac{N_{i+1}}{N_i} = \Delta \quad (0 < \Delta < 1) \quad (3)$$

식 (1)과 (3)에서 $p = \Delta$ 이면 다음의 조건을 만족한다.

$$N_i S_i \cong N_{i+1} S_{i+1} \quad (4)$$

여기에서 각 센서는 독립적으로 슬롯을 선택하며 선택된 슬롯들은 가장 작은 번호의 슬롯을 선택한 센서가 백오프 과정을 통해 전송에 성공할 때까지 계속 경쟁한다는 점을 고려한다면 각 센서들에 대한 슬롯 선택 확률은 식 (1)에서 부터 다음과 같이 유도될 수 있다.

$$\frac{S_i}{S_{i+1}} = \frac{(1-p^{i-1})p^{CW-i}}{(1-p^i)p^{CW-i}} \cdot p \approx p \quad (5)$$

이는 $\{CW\}$ 번 이전의 슬롯들($[i = 1, \dots, CW-1]$)에 대하여 다음 식(6)과 같이 전개될 수 있다.

$$\frac{S_1}{S_2} \cdot \frac{S_2}{S_3} \cdot \dots \cdot \frac{S_{CW-1}}{S_{CW}} = p^{CW-1} \quad (6)$$

이를 식 (4)에 적용하고 다음의 과정을 거쳐 최종적으로 식 (7)과 같은 결과를 얻을 수 있다.

$$\begin{aligned} \frac{N_2}{N_1} \cdot \frac{N_3}{N_2} \cdot \dots \cdot \frac{N_{CW}}{N_{CW-1}} &= p^{CW-1} \\ \therefore \frac{N_{CW}}{N_1} &= p^{CW-1} \end{aligned} \quad (7)$$

앞에서 언급한 바와 같이 $\{CW\}$ 번 슬롯을 단 하나의 활성화된 센서에 의해 선택되도록 $N_{CW} = 1$ 이라고 설정하면 $\frac{1}{N_1} = p^{CW-1}$ 이 된다.

이는 마지막으로 식 (8)과 같이 정리된다.

$$\therefore p = N_1^{\frac{-1}{CW-1}} \quad (0 < p < 1) \quad (7)$$

이제 식 (8)을 이용하여 $N = N_1$ 이라고 가정하면 최적의 슬롯 선택 확률 p 의 값을 계산할 수 있다. 즉, 센서 노드의 수가 256개의 네트워크 규모에서 32개의 슬롯을 가진 경쟁 윈도우를 사용하여 매체에 접근한다면 p 값은 대략 0.8 ($p = 256^{\frac{-1}{32-1}} \approx 0.8362$) 정도에서 결정되는 것이다.

3.3 수정된 MAC 프로토콜의 특성 비교

TinyOS의 MAC 계층이나 B-MAC[7] 등은 고정 윈도우 기반의 경쟁 슬롯들을 균등하게 선택하는 CSMA 프로토콜의 한 종류들이다. 센서 네트워크의 MAC 설계에서 고정된 크기의 경쟁 윈도우를 기반으로 하는 이유는 실제 환경에서 비교적 좋은 성능을 유지하면서 설계상의 단순함(simplicity) 때문이라고 할 수 있다. 반면에 다량의 센서 노드를 수용해야 하는 고부하 상태 센서 네트워크로의 확장성(scalability)은 취약하다는 단점이 있다.

Bharghavan 등이 제안한 무선 랜 MAC 프로토콜인 MACAW[10]는 BEB 방법을 사용하지만 채널 상태 정보는 공유하지 않는다. 이로 인해 노드들 중 하나가 전송에 성공하면 경쟁 윈도우의 크기가 최소값으로 초기화됨으로 다음 전송을 위해 처음부터 다시 경쟁을 시작해야만 한다. 이는 매체에 접근하는 센서 노드가 특정 구간에 집중되며 경쟁 윈도우 크기의 편차가 심해져 매체 접근 시 오버헤드를 발생시킨다. MACAW에서는 이러한 문제점을 학습을 통한 방식으로 해결했는데, 패킷 전송이 성공적으로 이루어지면 다음 경쟁에서는 경쟁 윈도우 크기를 새롭게 초기화(reset)하지 않고 이전 경쟁에서 사용했던 경쟁 윈도우 크기를 일정한 크기만큼 감소시켜 사용한다(MILD: Multiplicative Increase, Linear Decrease).

802.11에서는 서비스의 공정성(fairness) 문제를 메모리 기법을 이용하여 해결한다. 센서 노드가 경쟁에 참여하면서 경쟁 윈도우 내의 슬롯 중 하나를 균등한 확률로 랜덤하게 선택하게 되는데 선택된 슬롯의 값은 카운터다운 타이머에 설정된다. 매체가 busy 상태일 때는 카운터다운을 멈추고,

〈표 1〉 경쟁 윈도우 기반 CSMA 프로토콜들의 비교

프로토콜	학습 기법	메모리 기법	경쟁 윈도우	확률 분포
802.11	○	×	가변	균등분포
MACAW	×	○	가변	균등분포
BMAC,SMAC	×	×	고정	균등분포
제안한 DPSMAC	×	×	고정	기하분포

매체가 유휴 상태이면 카운터다운을 지속한다. 카운터다운 타이머의 값이 0(expire)이 되면 해당 센서 노드는 전송을 시작한다. 전송이 완료되면 경쟁 윈도우의 크기를 정해진 최소의 값으로 초기화한다. 결과적으로 센서 노드는 적절한 경쟁 윈도우의 크기를 찾아야 하기 때문에 허비되는 대역폭이 발생할 수밖에 없다.

이러한 문제점을 해결하기 위하여 본 논문에서는 고정된 크기의 경쟁 윈도우와 비 균등 확률에 의한 랜덤 슬롯 선택 기법을 채택한 새로운 MAC 프로토콜인 DPSMAC(differential probability of selection MAC)을 제안하였다. 제안한 DPSMAC 프로토콜은 기존 802.11 계열의 MAC 프로토콜들에 비하여 프로토콜의 구조가 단순함에도 불구하고 노드 간의 충돌이나 지연 시간을 최소화하면서 서비스의 공정성도 비교적 일정하게 유지해주는 장점을 가지고 있다.

〈표 1〉에서는 유비쿼터스 센서 네트워크에 일반적으로 적용되는 경쟁 기반 MAC 프로토콜들의 특성을 비교하여 보여주고 있다.

4. 실험 결과 분석

본 장에서는 네트워크 시뮬레이터인 ns-2 버전 2.30[11]을 사용하여 제안한 DPSMAC 프로토콜의 성능을 실험하였다. 시뮬레이션 동작 모델은 공통 베이스 스테이션이 미치는 평탄한 영역 내에 모든 센서 노드들을 배치시키고 이들 중 특정 이벤트에 반응한 센서 노드가 베이스 스테이션을 통해 작은 크기의 리포트(report) 패킷을 전송하는 형태이다.

실험은 경쟁 슬롯의 수가 32일 때 $p=0.8$ 인 조건에서 재설계한 DPSMAC 프로토콜과 대표적인 센서 네트워크용 CSMA 프로토콜[7-9, 12]인 SMAC 및 MACAW의 전체 지연 시간을 비교해 보았다. 지연 시간과 처리량을 측정하기 위한 성능 평가에서는 기본적으로 RTS/CTS를 비활성화하고 센서 노드들의 데이터 패킷을 40 바이트 크기로 설정한 상태에서 랜덤 초기 값을 달리하여 평균 20회 정도 실시하였다.

4.1 제안한 DPSMAC 프로토콜의 패킷 처리량

먼저 패킷 처리량을 측정하기 위한 실험에서는 네트워크 부하를 가변시켜 가면서 무선 매체의 가용 한도가 포화 상

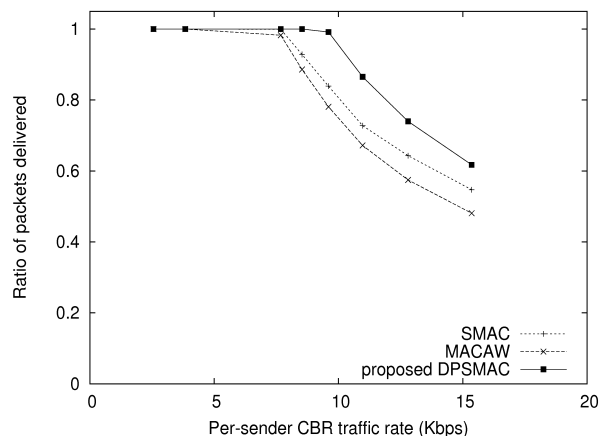
태에 이르게 한 다음, 제안한 방식의 패킷 처리량과 SMAC 및 MACAW의 패킷 처리량을 비교하였다. 또한, 제안한 프로토콜이 이벤트 기반 네트워크 환경에서도 원활하게 동작함을 보여주기 위하여 실험에서는 정상(定常) 상태(steady state)의 애드혹 네트워크(ad-hoc network) 환경에서 32개의 송신 노드에 의해 동시에 발생하는 CBR(constant-bit-rate) 플로우들을 모델링하고 이때의 패킷 처리량을 각각 측정하였다.

(그림 6)은 CBR 트래픽이 증가할수록 제안한 프로토콜의 패킷 처리량이 다른 비교 대상의 MAC 프로토콜들의 처리량을 능가하는 결과를 보여주고 있으며, 이와 함께 프로토콜들의 패킷 처리량이 트래픽이 증가함에 따라 약간의 손실이 발생함을 나타내고 있다. 이러한 손실의 원인은 전송되는 패킷들 간의 충돌로 인한 것이 아니라 경쟁 윈도우 내에서 선택한 슬롯이 상대적으로 뒤편에 위치함으로써 백오프 절차에 소요되는 시간이 비교적 길어지기 때문으로 생각된다.

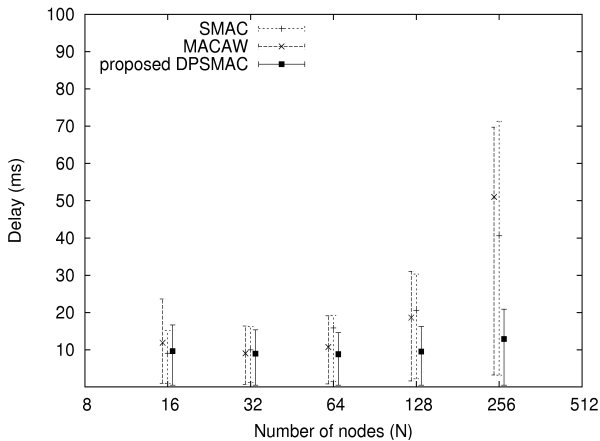
무엇보다도 제안한 프로토콜에서는 패킷 간의 충돌이 비교적 적게 발생하고, 전송에 참여하는 센서의 수가 증가함에도 불구하고 슬롯 선택 확률 분포의 최적화로 인하여 경쟁 윈도우 내에서 데이터 전송에 성공하는 슬롯의 위치가 경쟁 윈도우의 전반부로 이동하게 됨을 알 수 있다. 이러한 현상은 전송 매체에 접근할 때 발생하는 대역폭 집중 현상이 대체로 줄어드는 결과를 가져온다고 할 수 있다.

4.2 제안한 DPSMAC 프로토콜의 전송 지연 시간

전송 지연 시간 측정 실험에서는 일정한 간격으로 발생하는 이벤트 부하에 대하여 제안한 프로토콜의 지연 시간을 측정하였다. 이때 지연 시간의 증감에 대한 변화량을 보다 정확하게 측정하기 위해 센서 노드에 탑재되는 소프트웨어 시스템에 의한 지연이나 센서의 전자 특성에 의한 편차 등을 최대한 고려하였다. 이를 반영하기 위하여 실험에서는 각 센서가 자신의 이벤트 정보를 전송하는 시간에 0ms~1ms 범위의 랜덤 값을 각각 더해주었다.



(그림 6) CBR 트래픽에서 센서 노드들의 패킷 전송률



(그림 7) 센서 노드의 수에 따른 전송 지연 시간의 비교

실험에서 측정된 센서 노드 수의 변화에 따른 전송 지연 시간의 결과는 (그림 7)에 표시되어 있다. 이는 각 센서 노드로부터 전송되는 패킷들을 베이스 스테이션에 도착하는 순서에 따라 백분위로 정렬시킨 후, 첫 번째와 중간, 그리고 마지막 90%에 도착한 패킷들의 전송 지연 시간을 각각 추출하여 에러바(error bar)의 하단과 중앙, 그리고 상단 표시점으로 각각 표시하였다.

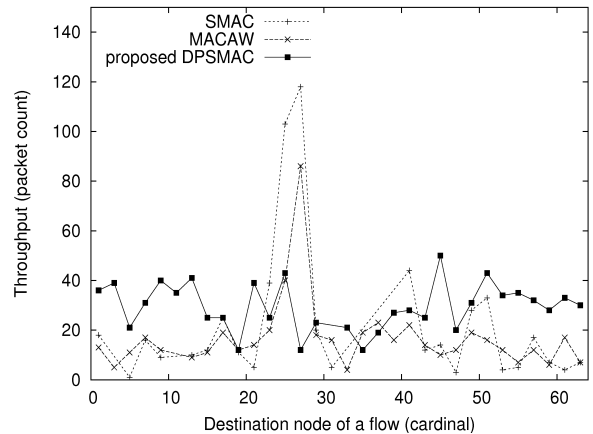
그래프의 전송 지연 시간을 표시한 에러바의 하단 부분으로부터 센서 노드의 수가 적을 때는 SMAC과 BMAC의 최소 경쟁 윈도우 크기는 노드 간의 경쟁 문제를 신속히 해결하기에 충분하게 크다고 볼 수 있지만, 최초의 데이터 전송을 성공시키기 위하여 센서 노드의 수가 증가할수록 이들의 경쟁 윈도우 크기는 사전에 지속적으로 증가되어 있어야 한다는 사실을 유추할 수 있다.

반면에 제안한 프로토콜의 경우는 센서 노드의 수와는 무관하게 경쟁 윈도우의 크기가 고정됨으로 인하여 경쟁 문제를 해결하는 시간이 일정하게 소비되고 있음을 확인할 수 있다. 이와 함께 제안한 프로토콜에서는 전체적인 센서 노드들의 데이터 전송에 소요되는 전체 지연 시간도 센서 노드의 수와 상관없이 향상된 성능을 보여주고 있다.

이로써 센서 노드가 데이터 전송을 수행하는 동안의 전체 전송 지연 시간을 측정된 결과를 놓고 보더라도 제안한 프로토콜이 다른 경쟁 기반의 센서 네트워크용 MAC 프로토콜에 비하여 뒤떨어지지 않는 성능을 유지한다는 사실이 입증되었다.

4.3 매체 접근에 대한 공정성 실험

마지막으로 본 논문에서 제안한 MAC 프로토콜이 충돌을 일으킨 일부 센서 노드들에서 발생할 수 있는 서비스 기회의 감소현상(starvation)을 효과적으로 해결하기 위하여 센서 노드들에게 스테이션에 접근하기 위한 대역폭을 공정하게 할당하는지를 고찰하였다. 해당 실험에서는 무선 전파 범위 내에 트래픽 소스의 역할과 싱크의 역할을 모두 할 수



(그림 8) 접근 매체 수에 따른 공정성 비교

있는 짝수개의 센서 노드를 적절히 배치하고, 노드 패킷의 크기는 1500 바이트, 노드 간 데이터 교환방식은 RTS/CTS가 활성화된 상태로 설정하였다. 무선 센서 네트워크의 수용 능력을 초과하는 부하 환경을 시뮬레이션하기 위해서 각 센서로부터 전송되는 패킷의 적체 현상도 감안하였다. (그림 8)은 각 센서 노드들의 처리량을 10초 동안 측정할 것이다.

그림에서 보는 바와 같이 제안한 방식의 처리량이 어느 특정 센서 노드에 집중되지 않고 전반적으로 고른 분포를 보이는 것은 기존의 센서 MAC 프로토콜들과 비교하여 비교적 공정한 매체 접근을 유지하고 있음을 나타내는 것이다.

현실적으로 유비쿼터스 센서 네트워크에는 같은 정보를 전송하는 여분의 노드(redundant node)들이 대부분 포함되어 있으므로, 제안한 MAC 프로토콜은 프로토콜 구조의 단순함에도 불구하고 전체 네트워크의 효율성은 분산 공정 스케줄링(distributed fair scheduling) 방식과 비교하여 손색이 없다고 할 수 있다.

5. 결론 및 향후과제

본 논문에서는 센서 노드들의 원활한 정보 전송을 위해 공간적으로 밀집도가 높은 대규모의 유비쿼터스 센서 네트워크 환경에 적합한 효율적인 매체 접근 기법을 기존의 경쟁 윈도우 기법을 수정하여 제안하였다. 즉, 제안한 프로토콜은 돌발적인 환경 변화가 빈번히 발생하더라도 전송률을 최대한 일정하게 유지할 수 있는 적응력이 강한 MAC 프로토콜이라고 할 수 있다.

다시 말하면, 유비쿼터스 센서 네트워크의 베이스 스테이션으로부터 데이터 전송에 대한 요청이 발생하면 이에 반응하는 일부 센서들을 제외한 나머지 센서들의 전송 동작이 억제되어야 하는 현실적인 상황에서도 제안한 MAC 프로토콜은 이상적으로 동작함을 시뮬레이션을 통하여 확인하였다.

무엇보다도 실시간 유비쿼터스 센서 네트워크에서 데이터의 효과적인 전송을 위하여 본 연구에서 제안한 프로토콜의 핵심적인 메커니즘은 센서 노드들이 경쟁 윈도우 내의 전송

슬롯을 선택하는 경우에 가능한 앞부분에 위치한 슬롯을 선택할 수 있도록 슬롯의 위치에 따라 증가하는 기하 확률 형태의 분포 함수를 채택하여 슬롯 선택 확률을 원천적으로 차등화하였다는 것이다.

이에 더하여 대부분의 전통적인 CSMA/CA[13] 기반의 MAC 프로토콜은 가변적인 경쟁 윈도우 기법을 사용하는 반면 본 연구의 프로토콜은 고정 크기의 경쟁 윈도우를 사용하여 지연 시간에 대한 저항성을 높였다는 특징이 있다.

앞으로 제안한 MAC 프로토콜을 실제의 이벤트 기반 유비쿼터스 센서 네트워크[14, 15] 환경에 적용하여 보다 더 현실적인 성능 평가를 진행하고, 이를 시뮬레이션에서 측정 한 성능 결과와 비교 분석하는 작업이 수반되어야 할 것이다.

참 고 문 헌

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A Survey on Sensor Networks," IEEE Communications Magazine, vol.40, no.8, pp.102-116, Aug. 2002.

[2] I. Demirkol, C. Ersoy, and F. Alagöz, "MAC Protocols for Wireless Sensor Networks: a Survey," IEEE Communications Magazine, 2005.

[3] V. Rajaravivarma, Y. Yi, and Y. Teng, "An overview of wireless sensor network and applications," in Proceedings of the 35th Southeastern Symposium on System Theory, pp.432-436, Morgantown, WV, USA, Mar. 2003.

[4] A. Abu-El Humos, "Low Latency and Energy Efficient MAC Protocols for Wireless Sensor Networks," Ph.D. Dissertation, Florida Atlantic University, Boca Raton, Florida, Aug. 2005.

[5] IEEE Std. 802.11-1999, Part 11, 'Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications', Reference number ISO/IEC 8802-11:1999(E), IEEE Std. 802.11, 1999 edition, 1999.

[6] M. Gast, '802.11 Wireless Networks: The Definitive Guide', 2nd ED., O'Reilly, Apr. 2005.

[7] J. Polastre, J. Hill, and D. Culler, "Versatile Low Power Media Access for Wireless Sensor Networks," In Proceedings of the Second ACM Conference on Embedded Networked Sensor Systems(SenSys), Baltimore, MD, pp.95-107, Nov. 2004.

[8] W. Ye, J. Heidemann, and D. Estrin, "Medium Access Control With Coordinated Adaptive sleeping for Wireless Sensor Networks," IEEE/ACM Transactions on Networking, vol.12, no.3, pp.493-506, June 2004.

[9] W. Ye, J. Heidemann, and D. Estrin, "An Energy Efficient MAC protocol for Wireless Sensor Networks," In Proceedings of IEEE INFOCOM 2002, Vol.21, No.1, pp.1567-1576, 2002.

[10] V. Bharghavan et al., "MACAW: A Media Access Protocol

for Wireless LANs," In Proceedings of ACM SIGCOMM '94, Vol.24, No.4, pp.212-225, 1994.

[11] USC ISI, 'ns-2 Notes and Documentation', 2007.

[12] A. Woo, and D. Culler, "A Transmission Control Scheme for Media Access in Sensor Networks," In Proceedings of ACM MobiCom '01, Rome, Italy, pp.221-235, Jul. 2001.

[13] G. Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordination Function," IEEE Journal on Selected Areas in Communication, Vol.18, No.3, pp.535-547, Mar. 2000.

[14] 공형윤, 최정호, 정휘재, "전력 및 대역폭 효율성 있는 무선 센서 네트워크 협력 전송에 관한 연구," 한국정보처리학회논문지 C, 제13권, 2호, pp.185-194, 2006.

[15] 김지은, 김세한, 정운철, 김내수, "USN 센서노드 기술 동향," 전자통신동향분석, 제22권, 3호, pp.90-103, Jun. 2007.



장 호

e-mail : hojang@ynu.ac.kr

1994년 영남대학교 전산공학과 졸업(학사)

1996년 영남대학교 대학원 전산공학과 졸업(공학석사)

2008년 영남대학교 대학원 컴퓨터공학과 졸업(공학박사)

2001년~2005년 구미1대학 컴퓨터전공 전임강사

관심분야: 실시간 유비쿼터스 센서 네트워크, 실시간 운영체제, 무선 센서 네트워크 보안 등



이 명 섭

e-mail : skydream@ync.ac.kr

1996년 경일대학교 컴퓨터공학과 졸업(학사)

1998년 영남대학교 대학원 컴퓨터공학과 졸업(공학석사)

2003년 영남대학교 대학원 컴퓨터공학과 졸업(공학박사)

2001년~2002년 경동정보대학 인터넷정보계열 전임강사

2003년~2006년 영남대학교 전자정보공학부 연구교수

2007년~2008년 영남대학교 무선멀티미디어시스템 설계기술 RIC 책임연구원

2008년~현 재 영남이공대학 컴퓨터정보계열 교수

관심분야: 통신망관리, 애드혹네트워크, 센서네트워크, 영상처리 등



전 우 상

e-mail : wsjeon@ynu.ac.kr

1997년 영남대학교 전자공학과 졸업(학사)

1999년 영남대학교 대학원 전자공학과 졸업(공학석사)

2002년 영남대학교 대학원 전자공학과 졸업(공학박사)

2006년~2008년 영남대학교 전자정보공학부 객원교수

2004년~현재 중소기업기술정보진흥원 정보화경영체제 평가위원, 표준화연구회 위원, Gerson Lehrman Group 자문위원

관심분야: 통신망관리, 애드혹네트워크, 센서네트워크, 영상처리, 임베디드 등