

센서 네트워크에서 트래픽 적응형 동적 S-MAC 프로토콜

Traffic Adaptive Dynamic S-MAC Protocol in Sensor Networks

유대석 · 박수성 · 최승식
- 인천대학교 -

요 약

무선 센서 네트워크에서는 기본적으로 교환이나 충전이 어려운 배터리로 동작하는 센서 노드들로 구성된다따라서 센서 노드들의 에너지 소모를 줄이는 것이 무선 센서 네트워크에서는 중요한 과제이다 이러한 에너지 효율적인 MAC 프로토콜을 구현하기 위해 기존의 IEEE 802.11 Protocol을 기반으로 휴면 기간의 에너지를 효율적으로 관리하는Sensor MAC(이하 S-MAC) Protocol이 제안되었다 본 논문에서는 에너지 효율적 스케줄링 기반의 기존 S-MAC을 기반으로 네트워크 트래픽에 동적으로 동작하기 위한 동적 S-MAC을 제안하였다 동적 S-MAC 프로토콜은 네트워크 트래픽 상황에 따라 프레임 주기를 변경함으로써 S-MAC의 에너지 효율을 개선하였다 NS2 시뮬레이션을 통해 동적 S-MAC과 기존 S-MAC의 성능을 평가하였다

Key Words : Sensor, Network, MAC, Energy

I. 서 론

무선 센서 네트워크(wireless sensor networks)는 환경 모니터링 의료 시스템과 로봇을 이용한 탐사 시스템 등 여러 분야에 활용되고 있는 기술 중 하나이다 이러한 센서 네트워크는 멀티홉 센서 네트워크로 구성된 수많은 노드들로 구성되어 있다 각각의 노드들은 임베디드 프로세서와 저전력 RF칩, 그리고 배터리 등으로 구성된 하나 이상의 센서들을 가지고 있다일반적으로 무선 센서 네트워크에서, 각 노드들은 데이터 프로세싱을 수행하거나 메시지를 목적지에 전송하는 일을 수행한다이러한 무선 센서 네트워크 설계 시, 고려해야 할 사항은 다음과 같다

·**Energy** : 각 노드를 구성하는 센서들은 배터리로 작동한다 일반적으로 각 센서들의 배터리를 교체하거나 재충전하는 것이 쉽지 않기 때문에 에너지 효율성은 네트워크의 생명주기를 결정하는 중요한 요소 중의 하나가 되었다 그 중에서도 MAC 프로토콜은 무선 전송에 의한 에너지 비용을 최소화해야 한다.

·**Latency** : 지연 시간에 대한 요구 사항은 시스템에 의해 결정되는 경우가

많다. 네트워크 모니터링 시스템의 경우 특정 이벤트가 발생하였을 때 목적 노드에 바로 전송함으로써 모니터링 컴퓨터나 사용자가 즉시 반응할 수 있도록 프로토콜을 설계해야 한다

·**Throughput** : 작업 처리량 또한 시스템에 의해 결정되는 경우가 많다 온도의 변화량을 측정하는 시스템의 경우주기적으로 목적 노드에 메시지를 전송하도록 설계를 해야 하며 반대로 감시 시스템이나 화재 경보 시스템과 같은 특정 이벤트에 반응하는 시스템의 경우 이벤트 시에만 목적 노드에 메시지를 전송할 수 있도록 설계를 해야 한다

·**Scalability** : 센서의 이동성을 고려하여 네트워크 설계 시, 각 노드들이 추가되거나 삭제 시(노드가 에너지를 완전히 소모하거나 중간에 분실된 경우 등) 유연하게 대처할 수 있도록 확장성을 고려한 설계를 해야 한다

이러한 MAC 프로토콜 설계의 주요한 고려 사항 중 에너지 효율은 무선 센서 네트워크의 최우선 목표라고 할 수 있다 기존의 IEEE 802.11 프로토콜의 경우 데이터 전송 시간 외에 대기시간에도 무선 장치의 전원을 켜놓는 idle listening로 인해 에너지 효율이 좋지 않은 단점이 있었다^[1]. 이를 해결하기 위해 고정된 duty-cycle을 갖는 S-MAC^[2]이 제안되었는데 활성 구간(active interval)과 수면구간(sleep interval)을 이용하여 에너지

효율을 높이는 데 기여하였다. 그렇지만 수율(throughput)과 지연(delay)에 대한 trade-off 발생으로 인해 좀 더 유동적이고 적응적인 프레임 크기를 제어할 문제가 제기되었다. T-MAC은 활성구간(active interval)에 적응성을 두어 활성구간을 유동성있게 제어하는 방법을 제시하였다.^[3] 본 논문에서는 2장에서 여러 MAC 프로토콜을 비교 분석하여 에너지 효율적인 MAC 프로토콜이 가져야 할 요소들에 대해 분석하고 3장에서는 T-MAC과 같이 네트워크 트래픽과 상황에 좀 더 효율적이고 적응성있는 동적 S-MAC을 제안하고 마지막으로 4장에서는 S-MAC과의 비교 분석을 통해 성능을 확인하고자 한다.

II. 관련 연구

Idle listening 상태에서의 에너지 소비를 줄이기 위한 방법으로 여러 해결책이 발표되었다. 일반적으로 duty cycle을 기반으로 하여 수면 상태(sleep state)를 주기적으로 제어하는 방법이 제시되고 있는데 예를 들면 TDMA 기반의 프로토콜은 예약과 스케줄링을 사용하기 때문에 충돌이 없어 자연스럽게 전력 소모를 줄이는 프로토콜이다^[4]. 그러나 ad-hoc 네트워크에서 스케줄을 획득하는 것이 쉽지 않을 뿐만 아니라 이웃 노드들의 스케줄을 유지하는 것은 많은 메모리를 소모하게 된다 또한 TDMA의 각 슬롯을 할당할 때, 각 노드들에게 동등한 슬롯 할당을 위해 시간을 매우 작은 슬롯으로 나누어 할당 하게 된다. 그렇지만 정확한 타이밍을 계산하기 어렵기 때문에 clock-drift로 인하여 시스템 자원의 낭비를 초래할 수 있다.

또 다른 에너지 소비를 줄이기 위한 방법은 wake-up 라디오를 이용한 방법이다. wake-up 라디오는 통신 채널과는 다른 주파수를 사용하여 수신되는 신호를 인지한다^[5]. 이 채널은 다른 노드들을 깨우기 위한 채널로 데이터 프로세싱을 하지 않으며, 매우 낮은 전력을 사용하여 신호를 감지한다. 그러나 wake-up 채널이 비록 저전력이라 할지라도 여분의 채널을 감지하고 있는 것은 에너지 소모를 줄이기 위한 측면에서 긍정적인 효과를 얻을 수 없다. 그래서 대부분의 무선 센서 노드는 단일 주파수를 이용하는 단일 채널 방식을 사용하고 있다.

다음으로 소개할 경쟁 기반 방식의 프로토콜은 단일 채널을 이용하는 방식으로 CSMA/CA(Carrier Sense Multiple Access/Collision Avoidance) 프로토콜이 그 대표적인 예이다. CSMA/CA는 충돌을 가능한 피하는 방식으로 이웃 노드들이 자신의 제어 패킷을 도청하게 함으로써 송수신시 매체에 접근하는 것을 막아줌으로써 불필요한 에너지를 줄일 수 있게 된다. 그러나 언제 수신될 지 모르는 신호를 위해 계속 채널을 감지하고 있어야 하므로 에너지 효율이 크게 증가되지 않는다. 이러한 단점을 보완하여 소모되는 전력을 줄이기 위해 Hill과 Culler^[6], El-Hoiydi^[7]는 각각 독자적으로 낮은 레벨 반송파(carrier)의 프리앰블 감지(preamble sensing)을 통해 주기적으로 무선 통신을 위한 부분의 전원을 켜고 끄는 방법을 제안했다. 특히 El-Hoiydi는 ALOHA 방식에 이 방법을 적용하여 Preamble Sampling^[7]을 제안하였으며, 이것을 더 발전시켜 송신자가 수신자의 샘플링 스케줄을 알고 있을 때 긴 프리앰블이 필요 없음을 이용하여 수신 노드가 채널 샘플링을 시작할 때부터 긴 프리앰블이 아닌 짧은 기본 길이의 프리앰블이 포함된 데이터를 보내는 WiseMAC을 제안하였다^[8]. WiseMAC은 그림 1과 같이 프리앰블 길이를 네트워크의 트래픽에 따라 유동적으로 변화시켜 프리앰블 길이의 최적화를 통해 불필요한 에너지 소모를 줄이게 된다.

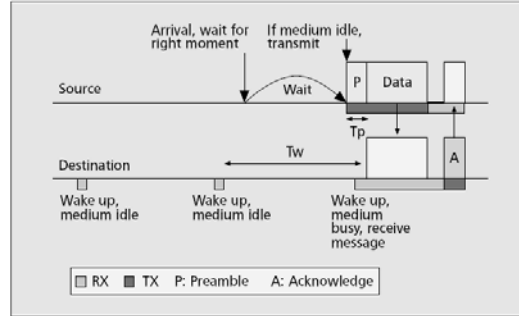


그림 1. WiseMAC 동작 방식

S-MAC^[2] 프로토콜 또한 경쟁 기반의 단일 채널을 사용하는 방식으로 그림 2와 같이 시간을 프레임 단위로 나누고 이 프레임을 활성 구간(active part)과 수면 구간(sleeping part)으로 나눈다. 수면 구간 동안은 노드의 무선 통신을 위한 부분의 전원을 끄고 활성 구간 시에는 전원을 켜서 이웃 노드와의 통신을 함으로써 에너지 소모를 줄이는 방식이다. S-MAC 또한 다른 노드들과의 동기화가 필요하지만 TDMA 방식의 프로토콜에 비해 세밀함을 요구하지 않는다. 그것은 각 프레임의 크기가 TDMA 방식의 슬롯보다 매우 크기 때문이다. S-MAC에서는 활성 구간과 수면 구간의 크기에 따라 수율과 지연에 영향을 미치게 되는 데 활성구간이 커지면 메시지 처리량이 증가하여 지연이 줄어들게 되지만 활성구간이 커진 만큼 무선 통신을 위한 부분의 전원을 켜놓고 있어야 하기 때문에 에너지 효율이 감소하게 된다. 이렇게 S-MAC에는 수율(throughput)과 지연(delay)에 대한 trade-off를 해야 하는 문제가 있다.



그림 2. S-MAC 동작 방식

T-MAC은 S-MAC의 고정된 duty-cycle에 적응성을 추가한 프로토콜로 네트워크 트래픽의 변동에 유연하게 적응할 수 있는 MAC 프로토콜이다^[3].

T-MAC은 발생하는 트래픽을 한꺼번에 모아서 활성 구간에 전송하고 수면 기간이 되기 전 정해진 time-out 시간 동안 데이터가 발생하지 않으면 수면 상태가 되는 방식으로 S-MAC 프로토콜에 비해 에너지 관리 효율이 조금 더 좋은 방식이다. 동적 S-MAC 또한 T-MAC에서 네트워크 트래픽의 변동에 유연하게 적응하는 방식에 착안하여 개발한 방식으로 T-MAC의 활성 구간에 유연성을 두는 것에 비해 동적 S-MAC은 프레임 전체에 유연성을 두는 것에 초점을 맞추었다.

III. 동적 S-MAC

본 장에서는 네트워크 트래픽 상황에 따라 프레임의 길이가 동적으로 변화하는 동적 S-MAC에 대한 기본 설계와 S-MAC과의 수학적 비교를 통해 성능을 평가한다.

3.1 동적 S-MAC 개요

동적 S-MAC 설계 시 가장 중요한 목표는 에너지 효율이다 앞서 소개한 것 처럼 에너지 효율에서 가장 문제가 되는 것은 idle listening이다. 이러한 idle listening을 줄이기 위하여 고정된 duty-cycle을 갖는 S-MAC이 고안되었으며 어느 정도 에너지 효율적인 MAC 프로토콜을 구현하게 되었다 그렇지만 고정된 duty-cycle을 가지고 있기 때문에 네트워크 트래픽에 따라 비효율적인 부분도 나타나게 되었다

첫 번째의 경우, 그림 3과 같이 메시지가 수면 상태에서 발생한 경우 활성 상태로 되기 까지 메시지가 버퍼에 저장되어 대기하고 있어야 하기 때문에 그 만큼의 데이터 전송 지연을 초래하게 된다

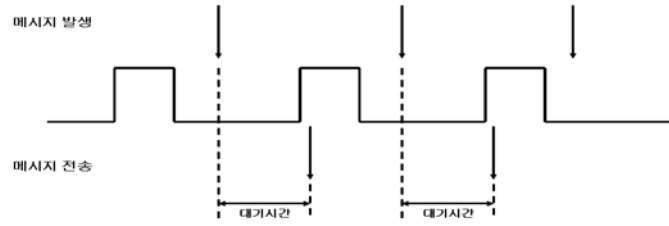


그림 3. 데이터 발생에 따른 대기 시간의 증가

T-MAC의 경우, time-out이 되기 전에 트래픽이 발생하지 않으면 수면 상태(sleep state)가 되어 위의 문제를 어느정도 해결하였다

두 번째의 경우, 그림 4와 같이 메시지 발생이 주기적으로 일어나지 않고 특정 시간에 집중적으로 발생한 뒤 오랜기간 발생하지 않을 경우, A, B 부분과 같이 메시지 전송이 없는 무의미한 활성 상태(active state)가 지속되어 에너지 효율을 떨어뜨릴 문제가 있다

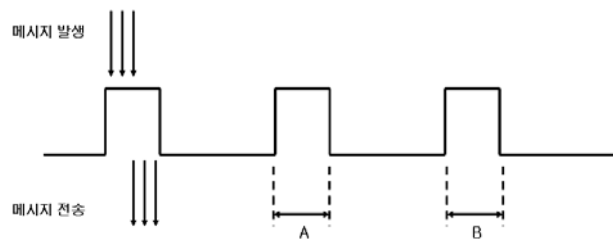


그림 4. 네트워크 트래픽 밀집도에 따른 에너지 소비

동적 S-MAC에서는 위와 같은 상황에 초점을 맞춰 설계한 방식으로 네트워크 트래픽에 따라 전체 프레임을 좀 더 유동적으로 변화시키는 데 중점을 둔 MAC 프로토콜이다.

기본적인 이론은 활성 구간에서 데이터의 발생 및 송수신 유무를 기록하는 하나의 flag를 사용하여 다음 활성 구간의 시작점에서 이 flag에 따라 활성 상태가 되는 지 수면 상태로 되는 지를 판별하는 방법이다 다음 그림 3과 같이 Flag의 값에 따라 활성 구간이 변화하게 되는 데 이전 활성 상태에서 Flag 값이 0이면 다음 활성 구간은 수면 상태가 되고 반대로 Flag 값이 1이면 활성 구간의 상태가 그대로 유지되는 방식이다

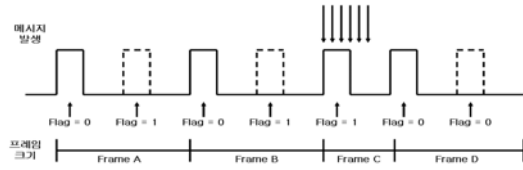


그림 5. 동적 S-MAC 동작과 프레임 변화

또한 메시지의 발생 및 송수신을 감지하도록 수면 상태가 된 활성 구간(sleeping active interval)에서는 Flag 값을 1로 하여 다음 활성 구간에서 메시지를 감지 할 수 있도록 하였다 즉, 메시지의 발생 및 송수신이 없는 경우 활성 구간이 토글(toggle)되어 수면화되는 방식이다

그림 5와 같이 네트워크 트래픽이 특정 기간에 집중적으로 발생하게 되면 그 구간의 프레임 길이를 줄여 메시지 송수신 지연 시간을 줄이기 위해 그림 5에서 Frame C 구간처럼 메시지 송수신 시에 Flag 값을 1로 하여 프레임 길이를 줄이게 된다 즉, Frame A, B, D 와 같이 메시지 송수신이 없는 경우 프레임 길이를 늘이게 되고 Frame C와 같이 메시지가 발생하면 프레임 길이를 늘이는 유동적인 방식이 동적S-MAC의 특징이라고 할 수 있다. 자세한 동작 방식은 Algorithm 1~3에 나타나 있다.

Algorithm 1. handleRTS

```
// 활성 상태에서 RTS 패킷을 받게 된 후,
CTS 패킷을 보내면서 Flag 값을 1로 조정
Send CTS packet to source node;
Flag = 1;
```

Algorithm 2. sendRTS

```
// 활성 상태에서 RTS 패킷을 보낸 후, Flag
값을 1로 조정
Send RTS packet to target or intermediate
node;
Flag = 1;
```

Algorithm 3. handleCounterTimer

```
// 현재 클락이 활성구간에 진입하면 Flag 값을
비교하여 수면상태 여부를 결정함
if ( Now time is listening interval &&
Flag == 0 ) then
    Node_state = SLEEP_INTERVAL;
    Flag = 1;
    Go to Sleep_interval_schedule;
else if ( Now time is listening interval &&
Flag == 1 ) then
    Node_state = LISTEN_INTERVAL;
    Flag = 0;
    Go to Listen_interval_schedule;
end if
```

데이터 전송 시 초기 데이터 패킷이 전송되어 목적 노드까지 전송하기 위한 중간 노드들의 Flag 값을 1로 설정하는 역할을 한다 그러므로 초기 데이터 전송 시 약간의 지연이 발생하게 된다 그러나 이 후의 패킷부터는 줄어든 프레임 기반으로 데이터를 전송하므로 계속적으로 지연시간이 줄어들게 되어 평균적으로 지연시간 감소를 얻을 수 있다

3.2 동기화 문제

동적 S-MAC은 Flag값에 따라 활성 및 수면 상태를 결정하는 방법으로 Flag값이 0이 되면 기존 프레임 크기의 2배가 되는 방식으로 동작한다 그러므로 주변노드와의 동기화 시 기본적으로는 S-MAC의 동기화 방식과 동일하며 SYNC 패킷에 보내지는 자신의 스케줄을 Flag가 1일 경우의 스케줄, 즉 그림 5에서의 Frame C의 경우에 해당하는 스케줄을 주변노드에 전송하여 동기화를 하도록 한다

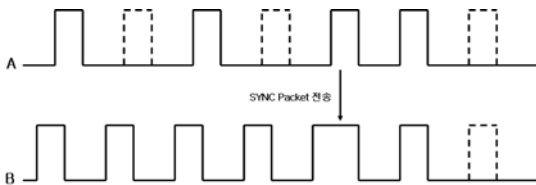


그림 6. 동적 S-MAC 동기화 과정

그림 6는 Node A에서 보낸 SYNC 패킷을 Node B에서 받게 되어 동기화 되는 과정을 보여주는 그래프이다 SYNC 패킷에 포함된 Flag값이 1인 경우의 스케줄을 참고하여 주변노드와 동기화를 하게 된다

3.3 주변 노드와의 연계

그림 7은 주변 노드와의 동기화 후 데이터를 전송하는 과정에서 주변 노드의 스케줄링을 나타낸 그림이다

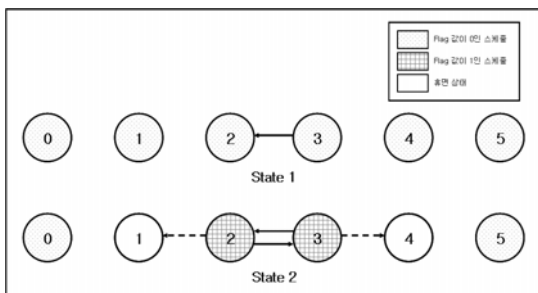


그림 7. 주변 노드와의 연계

State 1에서는 3번 노드가 2번 노드에 데이터 전송을 시도하기 위해 RTS 패킷을 보내게 된다 2번 노드는 RTS 패킷을 받은 뒤 데이터를 받을 준비가 완료되면 CTS 패킷을 보내 데이터를 요청한다 State 2와 같이 노드 2, 3이 서로 데이터와 ACK를 주고 받으며 이 때 Flag 값이 계속적으로 1이 되므로 원래 프레임의 1/2로 동작하게 된다 이 때, 1, 4 노드의 경우 2, 3 노드의 데이터 및 ACK 패킷의 헤더에 있는 duration에 의해 계속적으로 휴면 상태에 들어가게 되고 결국 0번과 5번 노드는 데이터의 주고 받음이 없

으므로 Flag값이 0이 되어 기존의 프레임 크기를 유지할 수 있게 된다 즉, 노드의 데이터 전송 시 주변 노드의 스케줄링에 기여를 하지 않게 동작하게 된다.

3.4 수학적 분석

S-MAC과 동적 S-MAC의 수학적 분석에 앞서 분석에 필요한 항목들과 기호를 정의하도록 한다

- Carrier sense delay : 한 노드가 채널을 점유하기 위해서는 Carrier Sensing 절차 과정을 거치게 되는 데 이 때 소요되는 평균 대기 시간을 t_{cs} 로 정의한다
- Transmission delay : 패킷 전송 시 채널의 bandwidth, 패킷 크기, 코딩 스키마에 따라 전송 시간이 차이가 나게 된다 그러므로 분석 과정에서는 Manchester encoding 방식에 의한 코딩 스키마를 사용하며 패킷의 길이가 고정되어 있다고 가정한다 그리고 실제 패킷 전송 시 소요되는 시간을 t_{tx} 로 정의한다
- Sleep delay : 각각의 노드들은 periodic sleeping에 의해 대기시간이 증가하게 된다 이러한 periodic sleeping algorithm에 의해 발생하는 대기 시간을 t_s 라고 정의한다
- Frame length : 활성 상태와 수면 상태에 소요되는 시간 전체를 T_f 로 정의한다

위의 정의된 항목을 사용하여 S-MAC과 동적 S-MAC의 성능을 비교분석하도록 한다 두 개의 노드를 사용하여 패킷을 전송하는 single-hop의 토폴로지를 구성하여 비교하며 이 때 채널 용량의 제한으로 인해 propagation delay와 processing delay는 무시하도록 한다

1) S-MAC의 대기 시간 : S-MAC에서 다른 노드의 패킷을 전송할 시에는 Carrier sensing에 의한 절차와 활성 구간이 되기 전 까지의 대기 시간이 소요된다 즉, S-MAC에서 패킷 전송에 필요한 대기 시간 D는 아래와 같다.

$$D = t_{cs} + t_{tx} + t_s \quad (1)$$

S-MAC에서 Sleep delay t_s 의 경우, 그 길이가 $(0, T_f)$ 사이에서 가변적으로 변하게 된다 그러므로 평균 대기 시간 E[D]는 아래와 같다

$$E[D] = E[t_{cs} + t_{tx} + t_s] = t_{cs} + t_{tx} + T_f/2 \quad (2)$$

2) 동적 S-MAC의 대기 시간 : 동적 S-MAC의 경우에는 두 개의 duty cycle이 존재하게 된다 그러므로 프레임 길이가 T_f 일 경우의 비율을 p_1 , $T_f/2$ 일 경우의 비율을 p_2 라고 정의하면 평균 대기 시간 E[D]는 다음과 같이 나타낼 수 있다.

$$E[D] = E[t_{cs} + t_{tx} + t_s] \quad (3)$$

$$= t_{cs} + t_{tx} + T_f(p_1/2 + p_2/4) \quad (4)$$

$$= t_{cs} + t_{tx} + \frac{T_f(2p_1 + p_2)}{4} \quad (5)$$

이 때, $p_1 + p_2$ 의 값은 1이 되고,

$$2p_1 + p_2 < 2 \quad (6)$$

$$\frac{2p_1 + p_2}{4} < \frac{1}{2} \quad (7)$$

$$E[D]_{\text{동적 S-MAC}} < E[D]_{\text{S-MAC}} \quad (8)$$

위의 분석에서 S-MAC의 Sleep delay t_s 의 길이가 동적 S-MAC의 t_s 의 길이보다 크다는 것을 쉽게 증명할 수 있다 즉, 평균 대기 시간이 기존의 S-MAC에 비해 동적 S-MAC이 감소하는 것을 볼 수 있다

IV. 실험 및 평가

4.1 실험환경

LBNL(Lawrence Berkeley National Laboratory) 의 NS2(Network Simulator)^[9] 2.29 버전을 사용하였다 NS2 시뮬레이터는 매우 신뢰성 있는 시뮬레이터로 무선 센서 네트워크에서 효율성을 검증하기에 적합한 시뮬레이터이다 노드 구성은 그림 8와 같이 구성하였다



그림 8. 노드 구성도

4.2 에너지 효율 성능 실험

그림 9은 패킷 전송이 없을 경우의 S-MAC과 동적 S-MAC의 에너지 소모율을 나타낸 그래프이다

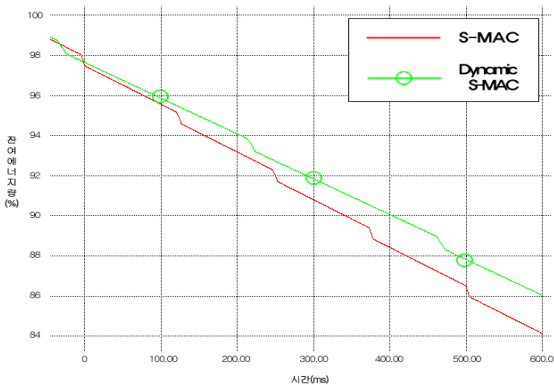


그림 9. 데이터 전송이 없을 시, SMAC과 동적 S-MAC 에너지 소모율

동적 S-MAC의 경우, 데이터의 발생이 없기 때문에 Flag 값이 0이 되어, S-MAC에 비해 프레임의 크기가 커지게 되어 그만큼의 휴면 기간이 늘어났기 때문에 에너지 효율이 높아진 것을 볼 수 있다

그림 10, 11은 패킷 발생 Interval이 30ms일 경우와 5ms일 경우의 S-MAC과 동적 S-MAC의 에너지 효율을 비교한 그래프이다

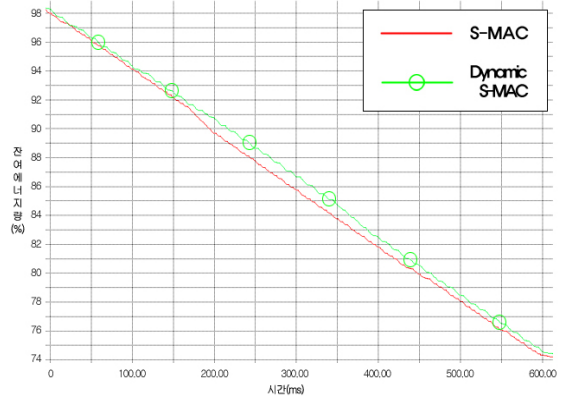


그림 10. Packet Interval이 30ms일 경우, S-MAC과 동적 S-MAC 에너지 소모율

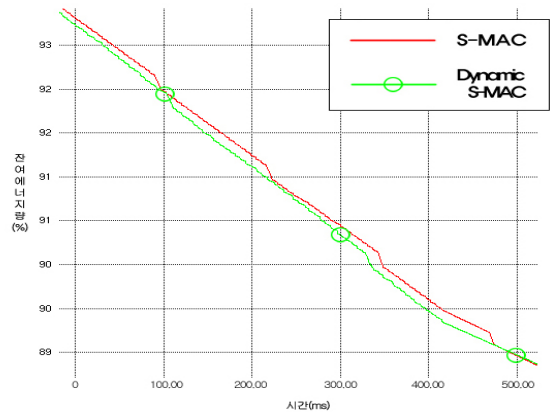


그림 11. Packet Interval이 5ms일 경우, S-MAC과 동적 S-MAC 에너지 소모율

그림 10의 경우, Packet Interval이 비교적 크기 때문에 Flag값이 0이 되는 구간이 많아짐으로 S-MAC에 비해 에너지 효율이 높게 되는 것을 볼 수 있다. 반면에 그림 11의 경우, Packet Interval이 비교적 작기 때문에 Flag값이 1이 되는 구간이 많아짐으로 인해 S-MAC과 거의 같은 프레임 길이로 동작하기 때문에 에너지 효율이 비슷한 것을 볼 수 있다

4.3 지연시간(Latency) 성능 실험

지연 시간 성능 실험 시 동적 S-MAC의 Flag값이 0일 경우의 프레임 크기와 S-MAC과의 프레임 크기를 동일하게 했을 경우와 2배로 늘려서 동작했을 경우의 두 가지를 사용하여 지연시간 성능을 실험하였다

그림 12은 S-MAC과 동적 S-MAC의 두 가지 종류에 대한 지연시간 성능 실험 그래프이다

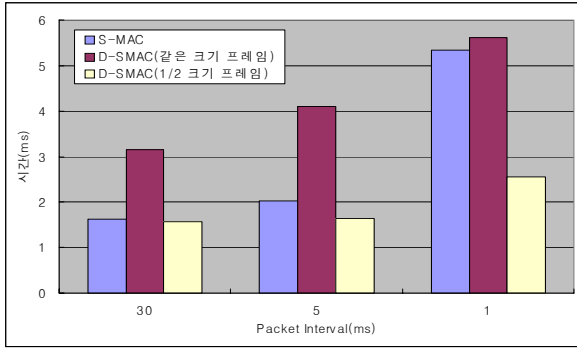


그림 12. S-MAC과 동적 S-MAC 의 지연시간 성능 평가

S-MAC의 프레임 크기와 동일한 크기로 설정한 동적 S-MAC의 경우, Packet Interval이 클 수록 S-MAC의 지연 시간에 비해 2배에 가까운 시간이 소요된다 이것은 패킷 발생이 빈번하지 않을 수록 Flag 값이 0일 경우가 많아지게 되며, 그 만큼 프레임 크기가 S-MAC의 2배가 되는 구간이 많아지게 된다. 그러므로 패킷 발생이 적을 수록 지연 시간이 커지게 됨을 알 수 있다. 하지만, Packet Interval이 작을 수록 S-MAC의 지연시간에 가까워지는 것을 볼 수 있다. 이것은 반대로 Flag 값이 1이 되는 경우가 많아지기 때문에 S-MAC과 거의 동일한 프레임 크기로 동작하여 비슷한 성능을 내고 있음을 알 수 있다.

S-MAC의 프레임 크기의 1/2로 설정한 동적 S-MAC의 경우, Packet Interval이 클 수록 S-MAC의 지연 시간에 밀접해 짐을 볼 수 있다. 이것 또한 Flag 값이 0이 되는 경우가 많아지기 때문에 그만큼 S-MAC과 동일한 크기의 프레임으로 동작하여 비슷한 성능으로 동작함을 알 수 있다. 하지만 Packet Interval이 작아 질 수록 S-MAC에 비해 동적 S-MAC이 월등한 성능향상을 보여주고 있는 데 이것은 Flag 값이 1이 되는 경우가 많아져 S-MAC 프레임의 1/2크기로 동작하여 지연 시간이 그만큼 많이 줄어든 결과로 볼 수 있다.

V. 결론

S-MAC은 고정된 duty-cycle을 이용하여 idle listening으로 인한 에너지 낭비를 해결하였다 하지만 고정된 duty-cycle은 네트워크 상태에 따라 변화하는 트래픽에 효율적으로 대처할 수 없는 문제를 가지고 있었다. 동적 S-MAC은 이러한 S-MAC의 고정된 duty-cycle에 좀 더 유연함을 제공하여 네트워크 트래픽의 변화에 좀 더 능동적으로 대처할 수 있는 방법을 제시하였다. 실험결과에서 확인하였듯이 동적 S-MAC은 기존의 S-MAC에 비해 유동적인 프레임 크기를 구현하여 네트워크 트래픽에 따른 에너지 효율의 증대 및 지연시간 향상을 보여주었다.

물론 지연시간 실험의 경우 두 가지 타입의 동적 S-MAC에 따라 성능향상에 변수를 가지고 있었지만 Flag 값의 좀 더 효율적인 제어방법을 찾아낸다면 에너지와 데이터 전송 효율 모두에서 뛰어난 성능을 보일 수 있을 것

으로 예상된다

앞으로의 연구계획은 실제의 환경에서 동적 S-MAC을 적용시키기 위해 micaz와 같은 모트에 직접 탑재하여 그 성능과 여러 가지 변수사항을 실험하고 판별하는 과정을 거쳐 더 효율적인 MAC 프로토콜을 연구할 예정이다

참고 문헌

- [1] IEEE Standard 802.11, "Wireless LAN Medium Access Control(MAC) and Physical Layer(PHY) Specifications," 1999.
- [2] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks," In 21st Conference of the IEEE Computer and Communications Societies (INFOCOM), volume 3, pages 1567-1576, June 2002.
- [3] T.van Dam And K.Langendoen, "An adaptive energy -efficient MAC protocol for wireless sensor networks," in Proc.ACM Sensys 2003, pp. 171-180, Nov. 2003
- [4] P. Havinga and G. Smit. "Energy-efficient TDMA medium access control protocol scheduling," In Asian International Mobile Computing Conference (AMOC 2000), pages 1-9, November 2000.
- [5] S. Singh and C. Raghavendra, "PAMAS: Power aware multi-access protocol with signalling for ad hoc networks," ACM SIGCOMM Computer
- [6] J. Hill and D.Culler.Mica, "A wireless platform for deeply embedded networks," IEEE Micro, Vol. 22, pp. 12-24, Nov. 2002
- [7] A.El-Hoiydi, "Aloha with preamble sampling for sporadic traffic in ad hoc wireless sensor networks," in Proc. IEEE ICC 2002, Vol.5, pp. 3418-3423, Apr.2002
- [8] A.El-Hoiydi, J.-D. Decotignie, C.Enz and E.Le Roux, "Poster abstract : WiseMAC, an Ultra Low Power MAC Protocol for the WiseNET Wireless Sensor Network," in Proc.ACM Sensys 003, Nov.2003.
- [9] The CMU Monarch Project, "The CMU Monarch Project's Wireless and Mobility Extensions to NS."