

랑데부 윈도우와 스니프 모드를 이용한 트래픽 적응 동적 통합 스케줄링

정희원 박세롬*, 이태진*

Traffic-Adaptive Dynamic Integrated Scheduling Using Rendezvous Window and Sniff Mode

Sae-Rom Park*, Tae-Jin Lee* *Regular Members*

요 약

블루투스는 무선으로 근거리 디바이스들을 연결하는 통신 기술로, 하나의 마스터와 하나 이상의 슬레이브 기기가 피코넷을 구성하고, 피코넷들이 연결되어 스캐터넷을 이루게 된다. 스캐터넷에서 여러 피코넷에 속하면서, 피코넷간 연결을 해주는 디바이스를 브릿지 또는 게이트웨이 노드라고 하는데 스캐터넷이 효율적으로 동작하도록 하기 위해서는 피코넷 내부의 마스터와 슬레이브 간의 효과적인 피코넷 스케줄링과 함께 브릿지 노드를 효과적으로 스케줄링 해주는 스캐터넷 스케줄링이 필요하다. 본 논문에서는 랑데부 포인트와 랑데부 윈도우를 이용한 스캐터넷 스케줄링 알고리즘과 함께 스니프 모드를 이용한 피코넷 스케줄링 알고리즘을 제안하고, 시뮬레이션을 통해 기존 방식과의 성능을 비교 분석하였다. 그 결과 트래픽의 상태에 따라 링크에 할당되는 대역폭을 가변함으로써 무선 자원을 효과적으로 분배할 수 있음을 보였다. 또한 제안된 알고리즘은 전력 소비 절약 보드인 스니프 모드의 사용으로 전력 소비를 절감할 수 있다는 장점을 갖는다.

Key Words : Bluetooth; WPAN; scatternet scheduling; piconet scheduling.

ABSTRACT

Bluetooth is a communication technology enabling short-range devices to be wirelessly connected. A master and one or more slave devices are connected to form a piconet, and piconets are joined to form a scatternet. The units participating in two or more piconets in a scatternet, is called bridge or gateway nodes. In order to operate the scatternet efficiently, both piconet scheduling for the master and slaves of a piconet, and scatternet scheduling for the bridge nodes are playing important roles. In this paper, we propose a traffic-adaptive dynamic scatternet scheduling algorithm based on rendezvous points and rendezvous windows. The performance of the proposed algorithm is compared and analyzed with that of a static scheduling algorithm via simulations. Simulation results show that our algorithm can distribute wireless resources efficiently to bridge nodes depending on the traffic characteristics.

I. 서 론

블루투스 [1][2][3] 기술은 저전력, 저비용, 단거리 무선 통신 기술로 Frequency Hopping Spread Spectrum (FHSS)을 사용하고, 2.4GHz 비인가

Industrial Scientific Medical (ISM) 밴드에서 동작한다. 호핑 주파수는 초당 1600홉에 달하고, 주파수 스펙트럼은 각각 1MHz 대역의 79채널로 나뉜다. 그리고 무선 링크의 신뢰성을 위해, 재전송 및 오류 검출/정정 방법인 Automatic Repeat Request

* 성균관대학교 정보통신공학부 네트워크시스템 연구실(saerom@ece.skku.ac.kr)

논문번호 : 030081-0225, 접수일자 : 2003년 2월 25일

※본 논문은 정보통신부 정보통신연구진흥원의 정보통신기초기술연구지원사업 지원으로 수행되었음.

(ARQ), Cyclic Redundancy Check (CRC) 그리고 Forward Error Correction (FEC)을 사용한다. 블루투스 사용하는 하나의 마스터(Master)와 슬레이브(Slave) 기기들이 하나의 피코넷(Piconet)을 이루는데, 하나의 마스터에는 최대 7개의 슬레이브가 연결될 수 있다. 이 피코넷을 서로 연결한 형태의 멀티홉 블루투스 네트워크를 스캐터넷(Scatternet)이라고 한다. 이 스캐터넷에서 피코넷 간 연결을 해주는 디바이스를 게이트웨이 또는 브릿지 노드라고 부른다. 피코넷과 스캐터넷 그리고 브릿지 노드의 예가 그림 1에 표현되어 있다.

피코넷에서 슬레이브 기기들과 마스터 기기들 간의 통신을 위해서는 폴링(Polling) 방법에 의한 피코넷 스케줄링(Scheduling) 방법이 사용된다. 피코넷 스케줄링의 방법에는 라운드 로빈, 가중치 라운드 로빈 등이 있는데 저전력 모드인 스니프 모드나 홀드 모드를 이용해 스케줄링 할 수도 있다. 반면 스캐터넷에서의 스케줄링은 여러 피코넷간의 연결을 포함하므로 스케줄링 방법이 매우 복잡해진다.

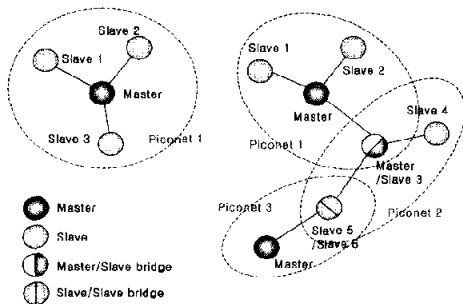


그림 1. 피코넷(왼쪽)과 스캐터넷(오른쪽)의 예.

이 가운데 Johansson 등은 피코넷이 서로 만나는 시점인 랑데부 포인트와 한 피코넷의 통신 구간인 랑데부 윈도우를 이용한 랑데부 포인트 IPS (Inter-Piconet Scheduling) 알고리즘^[4]을 제안했다. 이 연구를 발전시킨 Maximum Distance Rendezvous Point (MDRP) 알고리즘^[5]은 임의의 피코넷 마스터와 브릿지 노드 쌍을 위한 두 랑데부 포인트들 사이의 시간 주기를 나타내기 위해 주기적인 슈퍼프레임의 개념을 사용하였다. Zhang 등은 트래픽 변화에 적응적인 알고리즘^[6]으로 브릿지 노드와 연결된 노드의 트래픽 상태(버퍼)를 측정해 특정 노드의 폴링 횟수를 증가시켜 주거나, 스위치 테이블을 수정해 해당 노드에게 더 많은 기회를 주는

방식을 제안하였다. Baatz 등이 제안한 스케줄링 방식^[7]에서는 communication period를 나타내는 스니프 이벤트는 링크의 credit 값에 따라 변하는 가변길이를 갖는다. 이 연구를 발전시킨 APPD (Adaptive Presence Point Density) 알고리즘^[8]은 트래픽 로드가 낮거나 없는 링크에게 할당해 주던 스니프 슬롯을 스킵하는 방식이다.

위의^{[4],[5]}는 모든 슈퍼프레임에서 고정된 랑데부 포인트를 가지므로, 트래픽 변화에 적절하게 대응할 수가 없다.^[6]은 트래픽 변화에 적응하기 위한 스케줄링 알고리즘을 제시하고 있지만, 효율성이 떨어지고 오버헤드가 큰 단점을 가지고 있다.^{[7],[8]}은 스니프 모드를 이용해 파워 절약 효과를 얻으면서 고정된 스케줄링을 해주지만, 트래픽 적응 동적 스케줄링은 아니다. 이를 개선하기 위해^[9]에서는 효율적으로 동적인 트래픽 변화에 적응할 수 있는 랑데부 포인트를 이용한 스캐터넷 스케줄링 알고리즘을 제안하였다. 이러한 스캐터넷 스케줄링 방법은 피코넷 마스터와 브릿지 노드간의 효과적인 데이터 전송을 위한 것으로 임의의 피코넷 내에서 마스터와 슬레이브 간의 효과적인 데이터 전송까지 함께 고려하지 못했다. 따라서, 본 논문에서는 랑데부 윈도우를 사용한 트래픽 적응 동적 스캐터넷 스케줄링 알고리즘과 함께 피코넷 스케줄링에 스니프 모드를 사용해 전력 소비 절약과 효율적 데이터 전송이 가능한 트래픽 적응 동적 스케줄링 알고리즘을 제안하고 성능 분석을 통해 기존 방법에 비해 성능이 향상됨을 보인다.

본 논문의 구성은 다음과 같다. 2장에서는 기존의 라운드 로빈 피코넷 스케줄링과 스니프 모드를 사용한 피코넷 스케줄링 알고리즘에 대해 간단히 알아보고, 3장에서는 랑데부 윈도우와 스니프 모드를 사용한 정적 스케줄링 알고리즘에 대해 기술한다. 4장에서는 제안하는 랑데부 윈도우와 스니프 모드를 사용한 트래픽 적응 동적 스케줄링 알고리즘을 제시한다. 마지막으로, 5장에서 제안한 알고리즘에 대한 성능 분석을 다루고 6장에서 결론을 맺는다.

II. 피코넷 스케줄링 알고리즘

1. 라운드 로빈 피코넷 스케줄링

스케줄링에서 가장 기본적으로 사용되는 방식은 Round-Robin (RR)이며, 이 스케줄링 방식은 모든 링크에 순서대로 한번씩 전송 슬롯을 할당해주는

방식이다. 그림1의 피코넷1의 경우에 라운드 로빈 스케줄링의 예는 다음 그림 2와 같다.

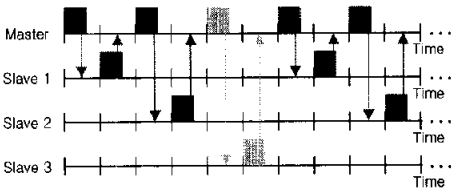


그림2. 라운드 로빈 스케줄링의 예.

라운드 로빈 방식과 같은 정적 피코넷 스케줄링 알고리즘은 단순하고 모든 슬레이브들에게 공평하게 슬롯을 할당해 준다는 장점을 갖는다. 하지만 링크의 트래픽 양과 관계없이 슬레이브들에게 동일한 기회를 주므로 폴링을 받은 슬레이브들에게 전송할 데이터가 없는 경우 슬롯이 낭비될 수 있다.

2. 스니프 모드를 사용한 피코넷 스케줄링

스니프 모드를 사용한 피코넷 스케줄링 알고리즘에서는 피코넷내의 각 슬레이브마다 Tsniff_attempt 간격을 두어 각 슬레이브는 이 구간동안에만 깨어나 마스터와 통신을 한다. 한 피코넷에서 모든 Tsniff_attempt 구간을 더한 시간 간격이 Tsniff이 되고 이 구간이 계속 반복되어 스니프 모드로 동작하게 할 수 있다 (그림 3 참조). 이러한 피코넷 스케줄링 방법은 전송 진력을 줄이면서 공정한 전송 기회를 각 슬레이브에게 줄 수 있다.

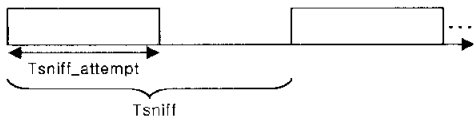


그림3. 스니프 모드의 동작.

III. 랑데부 윈도우와 스니프 모드를 이용한 정적 스케줄링 알고리즘

랑데부 포인트와 랑데부 윈도우의 개념을 사용하는 정적 스캐터넷 스케줄링 알고리즘에서는 브릿지 노드에 연결된 피코넷들을 위한 스케줄링의 한 주기를 슈퍼프레임이라고 부른다. 정적 스캐터넷 스케줄링 알고리즘의 브릿지 노드는 각각의 피코넷에게 라운드 로빈 방식으로 매 슈퍼프레임마다 동일한

랑데부 윈도우만큼의 슬롯을 할당해준다. 피코넷 P1, P2, P3의 세 피코넷을 스케줄링 하는 브릿지 노드가 가지고 있는 슈퍼 프레임의 예는 그림 4와 같다. 랑데부 포인트는 각각의 피코넷이 스케줄링에 참여한 시작 시점을 지정한다.

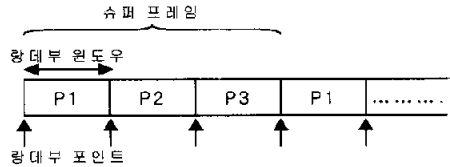


그림4. 슈퍼프레임 및 랑데부 윈도우, 랑데부 포인트 예.

브릿지 노드가 포함된 피코넷에서 슬레이브들에 대한 피코넷 스케줄링은 다음과 같이 할 수 있다. 즉 슈퍼 프레임에서 스캐터넷 스케줄링에 할당된 부분을 제외한 슬롯들을 Tsniff으로 본다. Tsniff 슬롯은 피코넷에 속한 브릿지를 제외한 슬레이브들에게 동일하게 제공되는데 이 슬롯들이 Tsniff_attempt가 된다. 그림1의 스캐터넷의 피코넷1에서 스니프 모드를 이용한 피코넷 스케줄링의 예는 그림 5와 같다.

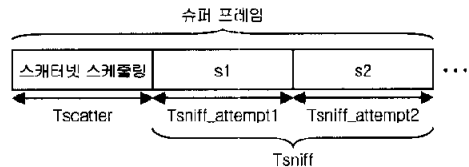


그림5. 스니프 모드를 이용한 피코넷 스케줄링의 예.

IV. 랑데부 윈도우와 스니프 모드를 이용한 트래픽 적응 동적 통합 스케줄링 알고리즘

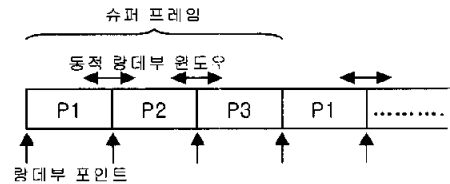


그림6. 동적 랑데부 윈도우, 랑데부 포인트 예.

량데부 윈도우를 이용한 동적 스캐터넷 스케줄링 알고리즘^[9]의 기본 개념은 슈퍼프레임마다 주기적인 량데부 윈도우를 트래픽 변화에 따라 가변시켜 주는 것이다. 제안된 알고리즘은 슈퍼프레임 단위로 실행되며, 새로운 슈퍼프레임이 시작되는 처음 슬롯에서 량데부 포인드와 량데부 윈도우 값을 새롭게 결정하게 된다. 동적 스캐터넷 스케줄링 알고리즘을 이용해 P1, P2, P3의 세 피코넷을 스케줄링하는 브릿지 노드의 예는 그림 6과 같다. 동적 스캐터넷 스케줄링에 의한 량데부 윈도우 값의 변화는 스니프 모드를 이용한 피코넷 스케줄링에서 Tsniff 값의 변화를 가져오게 된다.

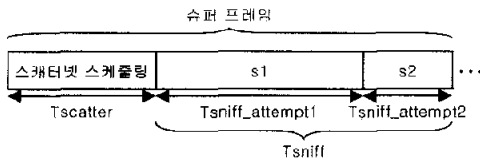


그림 7. 스니프 모드를 이용한 동적 피코넷 스케줄링의 예.

스니프 모드를 이용한 동적 피코넷 스케줄링은 그림 4에서처럼 피코넷에 속한 슬레이브들에게 고정적으로 할당되던 Tsniff_attempt 간격이 각 슬레이브의 트래픽 양에 따라 다르게 할당된다. 한 피코넷에 브릿지를 제외한 두개의 슬레이브가 존재하고 슬레이브1에 트래픽이 더 많을 경우의 Tsniff_attempt 할당의 예를 그림 7에 나타내었다.

이와 같이 스캐터넷의 트래픽 특성, 피코넷의 트래픽 특성에 따라 동적인 스케줄링을 사용하면 부족한 무선 슬롯을 효율적으로 사용하는 것이 가능하고 피코넷 별로 혹은 슬레이브마다 필요로 하는 성능 요구 조건을 만족시켜 줄 수 있다.

량데부 윈도우와 스니프 모드를 이용한 동적 통합 스케줄링 알고리즘의 순서도는 그림 8과 같다. 이 알고리즘에서는 슈퍼프레임의 처음 슬롯에서 각 트래픽의 양(버퍼)을 측정해 한계 값과 비교해 레벨 값을 결정한다. 여기서, 버퍼는 브릿지와 직접 연결된 링크의 여러 버퍼 중에서 브릿지로 전송될 패킷이 담긴 버퍼를 말한다. 그 후에는 각 버퍼의 레벨 값을 비교해 레벨이 큰 쪽의 량데부 윈도우를 증가시키고, 작은 쪽은 감소시키는 것이다. 이런 방법으로 슈퍼프레임내의 량데부 포인트의 위치를 바꿔준다. 단, 레벨이 같은 버퍼의 량데부 윈도우는 변하

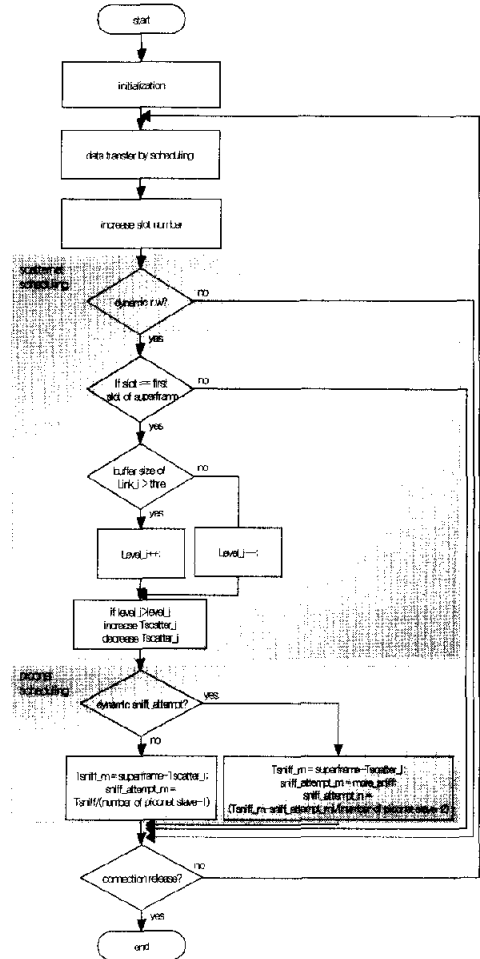


그림 8. 제안 동적 통합 스케줄링 알고리즘의 순서도.

지 않는다. 피코넷 스케줄링에서 사용되는 Tsniff와 Tsniff_attempt 간격은 변화된 량데부 윈도우 값을 기반으로 다시 계산된다.

V. 성능 분석

제안된 트래픽 적용 동적 통합 스케줄링 알고리즘과 기존 방법의 성능을 비교하기 위해 시뮬레이션을 수행하였다. 시뮬레이션에 사용된 토폴로지는 그림 9와 같은 스캐터넷 구조를 갖는다.

전체 시뮬레이션 실행시간은 100초이고 구현된 스캐터넷에서는 4개의 트래픽을 갖는다. 즉, 1) 슬

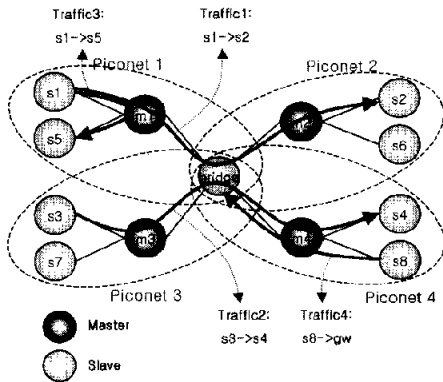


그림 9. 시뮬레이션에 사용된 토폴로지.

레이브 1에서 슬레이브 2로 2) 슬레이브 3에서 슬레이브 4로 3) 슬레이브 1에서 슬레이브 5로 4) 슬레이브 8에서 브릿지로의 트래픽이 존재한다. 스케줄링의 성능을 보기 위해서 각 트래픽은 Poisson process에 의해 발생된다.

트래픽 변화에 대한 적응을 확인하기 위해 트래픽1에서의 평균 패킷 발생률을 125에서 250 (packets/second)으로 변화시키면서 시뮬레이션을 수행했고 트래픽2와 4에서는 125(packets/second)으로 그리고 트래픽3에서는 100(packets/second)으로 패킷을 발생시킨다. 모든 패킷은 1슬롯 패킷이라고 가정한다. 제안된 알고리즘에 대해 기초 시뮬레이션 결과 최적값을 시뮬레이션의 기본 파라미터로 했고 값은 표 1과 같다.

표 1. 시뮬레이션에 사용된 파라미터.

Parameter	Value
양대부 윈도우(static인 경우)	12 슬롯
Tsniff(static인 경우)	36 슬롯
Tsniff_attempt(static인 경우)	18 슬롯
버퍼 threshold	70 패킷
양대부 윈도우 변화율	4 슬롯
최소 양대부 윈도우	2 슬롯

그림 10, 11에서는 Tsniff_attempt와 트래픽1의 트래픽 발생률 변화에 따른 지연시간과 전송율을 보여준다. more_sniff이 0일 경우가 양대부 윈도우와 스니프 모드를 사용한 정적 스케줄링 알고리즘에 해당되며, 그 이외는 트래픽 적응 동적 통합 스

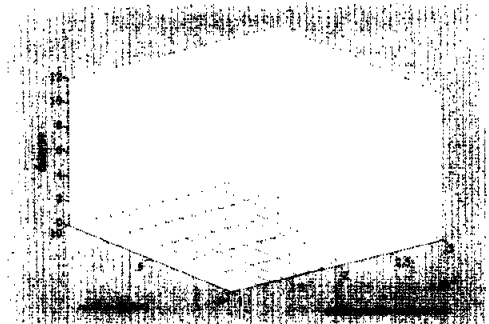


그림 10. 양대부 윈도우와 스니프 모드를 사용한 정적 스케줄링 알고리즘(more_sniff=0인 경우)과 동적 스케줄링 알고리즘에서 Tsniff_attempt와 트래픽 발생률 변화에 따른 지연시간(트래픽1).

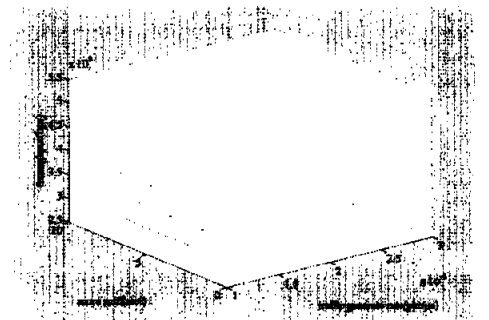


그림 11. 양대부 윈도우와 스니프 모드를 사용한 정적 스케줄링 알고리즘(more_sniff=0인 경우)과 동적 스케줄링 알고리즘에서 Tsniff_attempt와 트래픽 발생률 변화에 따른 전송율(트래픽1).

케줄링 알고리즘에 해당된다. more_sniff을 증가시키에 따라 지연시간은 감소하고 전송율은 증가함을 확인 할 수 있다. 트래픽 발생률이 많아도 할당되는 Tsniff_attempt값이 증가함에 따라 지연시간을 감소시키면서 전송율의 향상을 이룰 수 있다.

양대부 윈도우와 스니프 모드를 사용한 트래픽 적응 동적 통합 스케줄링 알고리즘의 성능 평가를 위해서, 라운드 로빈 스케줄링, 양대부 윈도우와 스니프 모드를 사용한 정적 스케줄링 알고리즘과의 성능을 비교하였다. 라운드 로빈 방식과 정적 스케줄링 방식 역시 같은 환경에서 시뮬레이션 했고, 세



그림12. 기존 라운드로빈, 정적 알고리즘과 제안 동적 통합 스케줄링 알고리즘의 지연시간과 전송율 비교(트래픽1).

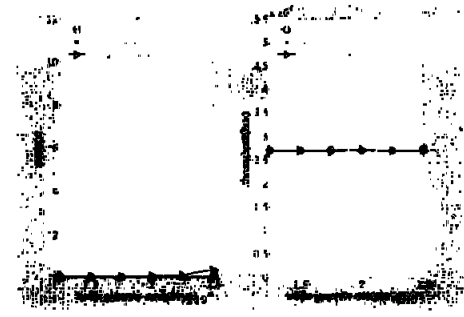


그림13. 기존 라운드로빈, 정적 알고리즘과 제안 동적 통합 스케줄링 알고리즘의 지연시간과 전송율 비교(트래픽2).

스케줄링 방식을 비교한 결과를 그림 12, 13, 14, 15에 나타내었다.

그림에서 볼 수 있듯이 기존의 라운드 로빈과 정적 스케줄링 알고리즘은 동일한 성능을 보여주고 있다. 이는 두 알고리즘이 동일하게 정적 스케줄링 방식을 사용하기 때문이다. 그러나 이 경우에도 제안된 정적 스케줄링 방법에서는 스니프 모드를 사용함으로써 전력 소비를 최소화 할 수 있다. 그림 12의 결과에서 모든 링크에 같은 양의 대역폭을 할당해주는 기존의 두 방식보다는 동적인 랑데부 윈도우를 이용해 트래픽이 많은 트래픽1쪽에 더 많은 대역폭을 할당해 주는 것이 더 효율적임을 확인할 수 있다(지연시간은 최대 11s에서 0.2s로 감소, 전송율은 42Kbps에서 54Kbps로 증가). 그림 14의 트래픽3은 스니프 모드를 이용해 트래픽이 많이 불러

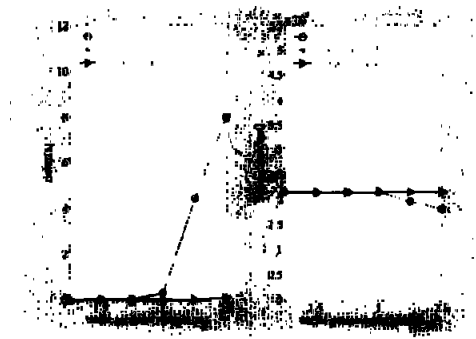


그림14. 기존 라운드로빈, 정적 알고리즘과 제안 동적 통합 스케줄링 알고리즘의 지연시간과 전송율 비교(트래픽3).

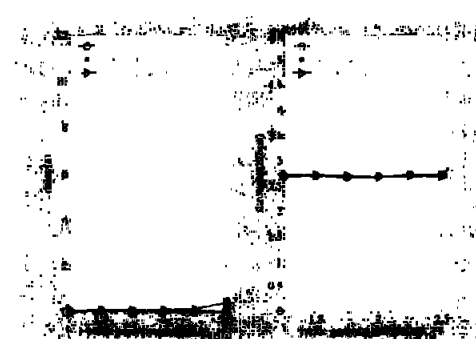


그림15. 기존 라운드로빈, 정적 알고리즘과 제안 동적 통합 스케줄링 알고리즘의 지연시간과 전송율 비교(트래픽4).

는 슬레이브 쪽 링크로 더 많은 Tsniff_attempt 간격을 할당해준 결과이다. 그림에서 볼 수 있듯이, 지연시간의 현저한 감소와 트래픽 변화에 전송율이 영향을 받지 않음을 알 수 있다(지연시간은 최대 8s에서 0.1s이하로 감소, 전송율은 22Kbps로 유지). 즉, 트래픽1과 트래픽3에 각각 트래픽 적응 동적 랑데부 윈도우와 동적 스니프 모드를 사용해 성능 향상을 가져올 수 있음을 확인할 수 있다. 반면 트래픽2와 4의 경우에는 기존의 방법에 비해 성능 저하가 거의 없음을 확인할 수 있다.

트래픽이 많은 쪽의 지연시간이 현저히 감소(최대 트래픽1은 98.18%, 트래픽3은 98.75% 감소)하면서 트래픽이 적은 쪽의 지연시간과 전송률은 큰 차이가 없음을 확인할 수 있다. 그 효과는 트래픽 발

생물이 증가함에 따라 더욱 효과적으로 알 수 있다. 즉, 전체적인 성능을 향상시킬 수 있음을 보였다.

VI. 결 론

본 논문에서는 양테부 포인트와 양테부 윈도우를 이용한 동적 스케터넷 스케줄링 알고리즘과 함께 동적 스니프 모드를 이용한 통합 스케줄링 알고리즘을 제안하였다. 제안된 알고리즘을 검증하기 위하여, 스케터넷 환경에서의 성능을 시뮬레이션을 수행하였다. 그 결과, 라운드 로빈 방식이나 양테부 윈도우와 스니프 모드를 사용한 정적인 스케줄링 알고리즘 방식보다 제안 알고리즘이 우수한 성능을 보이는 것을 확인할 수 있었다. 본 논문에서 제안한 방법은 기존 방법에 비해 복잡하지 않으면서 다양한 스케터넷 환경에서 효과적인 자원분배를 할 수 있고 스니프 모드를 이용해서 전력 소비를 절약할 수 있다. 또한 다양한 트래픽별 성능 요구 조건을 만족시키기 위해 사용될 수 있다.

참 고 문 헌

[1] Bluetooth Special Interest Group, www.bluetooth.com.
 [2] Bluetooth SIG, "Specification of the Bluetooth System," Ver.1.1b, Feb. 2001.
 [3] Chatschik Bisdikian, "An Overview of the Bluetooth Wireless Technology," IEEE Communications Magazine, pp. 86-94, Dec. 2001.
 [4] Per Johansson, Manthos Kazantzidis, Rohit Kapoor and Mario Gerla, "Bluetooth: An Enabler for Personal Area Networking," IEEE Network, pp. 28-37, Sep./Oct. 2001.
 [5] Per Johansson, Manthos Kazantzidis, Rohit Kapoor and Mario Gerla, "Rendezvous Scheduling in Bluetooth Scatternets," Proc. of IEEE ICC, pp. 318-324, 2002.
 [6] Wenshenge Zhang and Guohong Cao, "A Flexible Scatternet-wide Scheduling Algorithm for Bluetooth Networks," Proc. of Performance, Computing and Communications Conference, pp. 291-298, 2002.
 [7] Simon Baatz, Matthias Frank, Carmakuhi,

Peter Martini and Christoph Scholz, "Adaptive Scatternet Support for Bluetooth Using Sniff Mode," Proc. of IEEE Local Computer Networks, pp. 112-120, 2001.

[8] Simon Baatz, Matthias Frank, Carmakuhi, Peter Martini and Christoph Scholz, "Bluetooth Scatternets: An Enhanced Adaptive Scheduling Scheme," Proc. of IEEE INFOCOM, vol.2, pp. 782-790, 2002.
 [9] Sae-rom Park and Tae-jin Lee, "Traffic-Adaptive Dynamic Scatternet Scheduling for Bluetooth Systems," 한국통신학회 추계 종합 학술대회, 2002.

박 새 롬(Sae-Rom Park)

정회원



2000년 2월 : 동국대학교 정보통신공학과 공학사
 2002년 3월~현재 : 성균관대학교 정보통신공학부 석사과정

<주관심분야> 무선 LAN/PAN, 블루투스, ad-hoc 네트워크

이 태 진(Tae-Jin Lee)

정회원



1989년 2월 : 연세대학교 전자공학과 공학사
 1991년 2월 : 연세대학교 전자공학과 공학석사
 1995년 12월 : University of Michigan, Ann Arbor, EECS (M.S.E.)

1999년 5월 : University of Texas, Austin, ECE (Ph.D.)

1999년 8월~2001년 2월 : 삼성전자 중앙연구소 책임연구원

2001년 3월~현재 : 성균관대학교 정보통신공학부 조교수

<주관심분야> 통신 네트워크 성능 분석 및 설계, 무선 LAN/PAN, ad-hoc 네트워크, 무선 통신 시스템