

블루투스 스캐터넷의 형태에 따른 성능 분석

강형구⁰ 강석진 차정근 고진환

경북대학교 정보통신학과

iekang⁰@inc.knu.ac.kr, time007@inc.knu.ac.kr, cha1210@inc.knu.ac.kr, jikoh@ee.knu.ac.kr

Bluetooth Scatternet formation and performance

Hyung-Koo Kang⁰ Suk-Jin Kang Jung-Keun Cha Jin-Hwan Koh

Dept. of Information and Communication Kyungpook National University

요 약

Bluetooth는 기기들간에 고정된 제어 장치를 갖지 않고 각각의 이동 기기가 스스로 piconet이라는 불리는 네트워크를 구성하게 되고, 이 piconet을 확장하여 scatternet을 구성하게된다. 이 scatternet의 형태는 Bluetooth 망 전체 성능에 큰 영향을 미치게 되는데, 본 논문에서는 지금까지 제시된 scatternet의 구성 방법들을 알아보고, scatternet을 구성하기 위해 piconet간에 공유되는 bridge 노드의 숫자를 2개까지 늘려서 원활한 통신과 안정적인 망 구성을 위한 방안을 제시한다. 또한, 성능 분석을 위하여 Bluetooth 전문 시뮬레이터인 IBM의 BlueHoc을 사용하여 bridge 노드수와 scatternet을 구성하는 piconet수의 변화에 따른 scatternet의 성능을 모의 실험을 통해 측정하여 보았다.

1. 서 론

Bluetooth piconet의 master는 자신의 piconet의 frequency hopping sequence를 결정하고 채널 접속을 컨트롤 한다. piconet에 속해있는 나머지 장치들은 master가 정한 hopping sequence에 동기화 되어있다. 채널은 TDD(time division duplex)방식을 사용해서 공유되는데, master와 slave에 각각의 시간 슬롯을 할당해서 서로 통신가능하게 한다. 따라서 Bluetooth 장치는 동시에 움직 하나의 piconet과 통신할 수 있는데, 각piconet은 master에 의해 결정되는 서로 다른 hopping sequence를 사용하므로, bridge 노드는 다른 piconet과 통신하기 위해 그 piconet의 hopping sequence로 스위칭하고 동기화를 시켜야한다[1][2].

이 bridge노드의 스위칭 지연시간은 서로 다른 piconet을 연결하기 위해서는 필연적인것이기기는 하지만 piconet간의 원활한 통신을 방해하는 요인이 된다[3][6].

그래서 본 논문에서는 piconet간의 원활한 통신을 위해 공유 하는 bridge 노드의 수를 변화하여 최적화시켜서 링크 확립에 많은 지연이 일어나게 하지 않으면서도 망을 좀더 안정적으로 구성하고 전체적인 처리량을 높일 수 있는 방법을 제시하고 IBM의 BlueHoc[7]을 사용한 모의 실험을 통하여 공유 노드수 변화에 따른 패킷 처리량을 측정하여 보았다.

2장과 3장에서는 Bluetooth의 링크가 어떻게 형성되는지와 기존의 연구들이 제시하는 방법의 장점 및 형태에 대해 설명하고, 4장에서 모의 실험을 통한 성능분석을 해보며, 5장에서 결론 및 향후 연구과제에 대해 제시한다.

2. Bluetooth 링크 형성 프로토콜

Bluetooth Baseband specification은 일대일 연결설정에 대해 정의하고 있는데, 먼저 송신 노드는 Inquiry 과정을 통해 주위에 IAC(Inquiry Access Code) 패킷을 보내고, 수신 노드가 듣게 되면 PAGE 과정을 통해 DAC(Device Access Code)를 다시 수신 노드에 보내게 된다[1][2].

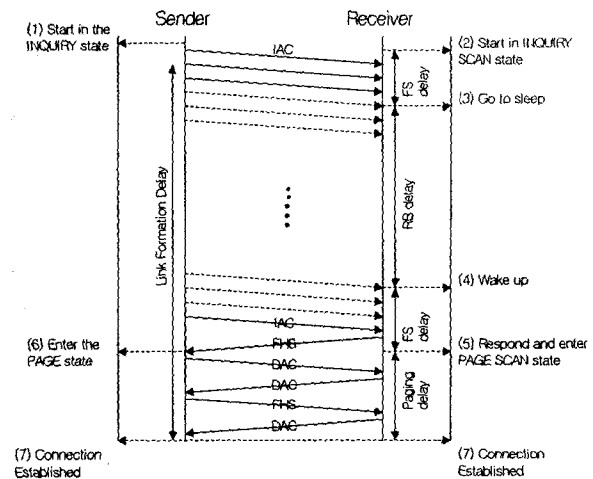


그림 1. Bluetooth 링크 형성[2]

노드들 간에 서로 다른 hopping sequence를 사용하. 로 Inquiry의 경우에는 빠른 frequency hopping sequence, Inquiry Scan의 경우에는 느린 hopping sequence를 사용하여 서로를 발견한다. <그림 1>에서 같이 링크 형성시에는 sender의 전송을 receiver가 들

때 까지의 지연인 FS(Frequency Synchronization) 지연과 receiver가 회답할 때 까지의 RB(Random Backoff)지연이 발생한다[2].

3.이중 bridge 노드를 이용한 scatternet 구성

이 장에서는 먼저 기존의 연구들을 살펴보고, 뒤에 본 논문에서 제시하는 방안을 기술하겠다.

3.1 TSF(Tree scatternet formation)

TSF라 불리는 이 구조는 노드들에게 연결되어 있는 동안 maseter/slave 역할을 할당한다. 메시지의 라우팅과 스케줄링을 간단하게 하는 tree 구조로 노드를 연결하는데, 라우팅 루프에 대해 고려할 필요가 없고, 두 노드사이의 통신을 위해 하나의 경로만 존재 하게 되므로 라우팅이 간단해진다. 노드는 그 위치에 따라 유일한 주소가 할당되고, 이 주소로 목적지를 찾아가므로 패킷전달이 간단하게 된다. 또 tree 구조로는 링크의 숫자와 bridge 노드당 평균 piconet 숫자를 줄일 수 있다. 링크의 숫자를 줄이는 것은 TDD 시스템에서 전송 슬롯에 대한 경쟁을 줄인다. 그리고, bridge 노드당 평균 피코넷 숫자를 줄이는 것은 bottleneck 현상을 피할수 있게 한다. 망 구성시에도 노드 숫자에 제한이 없다. 하지만, 어떤 노드가 이탈시 링크를 다시 확립해야하고, 망이 커지면 통신을 위해 거치는 노드수가 많아 질 수가 있는 단점이 있다 [4].

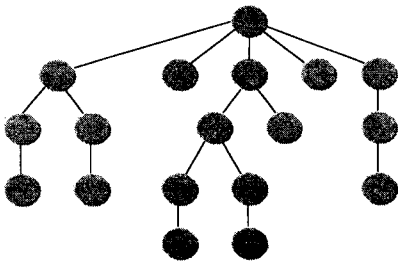


그림 2. TSF scatternet 구조

3.2 BTCP(Bluetooth Topology Costruction Protocol)

BTCP방식은 분산된 위상 구조를 가지며, 노드들 중에서 leader 뽑아서 나머지 노드들의 역할을 결정하고 망 형성을 컨트롤 하게 된다. 그 특징으로는 다음과 같다.[2]

- bridge 노드는 오직 두개의 piconet만을 연결한다. (각 piconet간의 통신을 위한 스위칭 시간에 의해 야기되는 overload를 줄인다.)
- piconet의 수를 최소화하여 scatternet을 구성한다. (piconet의 수를 줄여서 네트워크의 컨트롤을 쉽게한다.)
- master 노드는 bridge노드를 통하여 다른 모든 master 노드와 연결된다.
- 두개의 piconet은 오직 하나의 bridge 노드만을 공유한다.(piconet간의 연결 확립을 쉽게한다.)

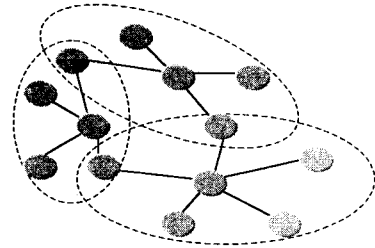


그림 3. BTCP scatternet 구조

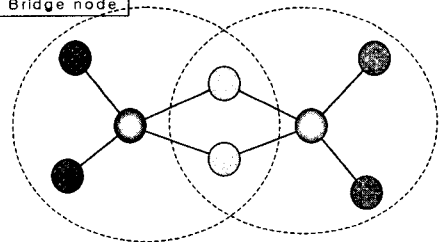
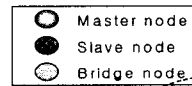
BTCP는 연결 지연을 줄이는데 중점을 두고 있지만, 링크 확립에 큰 시간지연이 일어 나지 않는다면, 이중의 bridge 노드를 사용하면 망의 효율을 더 높이게 된다.

3.3 제시 방안

본 논문에서 제시하는 방식은 분산된 scatternet 구조를 가지면서 주어진 환경에 따라 bridge 노드의 수를 2개로 하여 안정적인고 원활한 통신을 가능하게 하는 scatternet 형태를 제시한다.

3.3.1 1개 또는 2개의 bridge 노드

scatternet 구성시 각 piconet의 통신 가능범위에 공유되는 노드가 1개 이면 단순히 bridge노드를 하나로 해서 scatternet을 형성한다. 하지만 각 piconet이 공유할수 있는 노드의 수가 2개 이상이라면, 2개의 노드를 공유하여 scatternet을 형성한다. 2개의 bridge노드를 공유하는 이유는 piconet간의 원활한 통신과 안정성을 위해서이다. bridge 노드는 piconet간의 동기를 위한 스위칭으로 인해 시간지연이 생기고 처리량이 많을 경우에는 bottleneck 현상이 일어 날수 있다. 또 만약 bridge 노드에 문제가 생기거나 노드가 이동하게 되면 링크를 다시 확립해야 한다. 그런데 2개의 노드를 공유하면 이런 문제들을 해결하거나 줄일수 있다. 물론 2개의 노드를 공유하면 1개의 노드를 공유할 때 보다 링크의 숫자가 늘어나서 링크 확립 시간이 늘어나지만 이것은 크게 문제가 되지 않는다.



(a) 2개의 bridge 노드

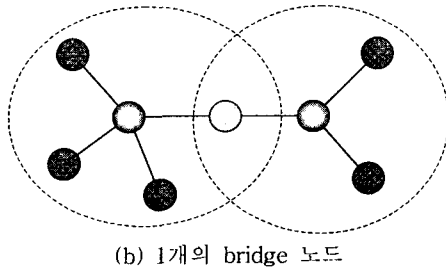


그림 4. bridge 수에 따른 링크의 확립

<그림 4>의 (a)와 (b)를 비교해 보면 알수 있듯이 한개의 bridge 노드를 공유할 때 보다 한번의 링크만 더 확립시키면 2개의 bridge 노드를 형성할 수 있다.

하지만 bridge 노드의 수를 하나씩 늘릴때 마다 링크의 확립도 늘어나고, 라우팅도 복잡해진다. 다수의 bridge 노드를 공유하는 것은 상당히 비효율적이 되고, 링크의 수가 많아져서 scatternet 형성에 큰 지연을 초래하고 망을 복잡하게 한다.

4. 성능 평가

본 논문에서는 시뮬레이션을 위해 IBM의 블루투스 시뮬레이터인 BlueHoc을 사용하였다[7].

먼저 <그림 5>는 2개의 piconet으로 구성된 scatternet에서 bridge 노드의 수를 2개까지 변화시켜 패킷 처리량을 측정한 결과이다. 시간이 지날수록 1개 보다는 2개의 bridge 노드를 사용한 piconet의 패킷 처리량이 더 높아진다는 사실을 알 수 있다. <그림 6>은 piconet의 숫자를 3개 이상으로 늘려 scatternet을 구성한 다음 패킷 처리량의 변화를 알아보았다. 다수의 piconet으로 구성된 scatternet에서도 1개 보다는 2개의 bridge노드를 사용하는 것이 패킷 처리량 측면에서는 유리하지만 전체적으로 원활한 통신을 위해서는 더 작은 수의 piconet으로 구성된 scatternet이 더 유리함을 알 수 있다.

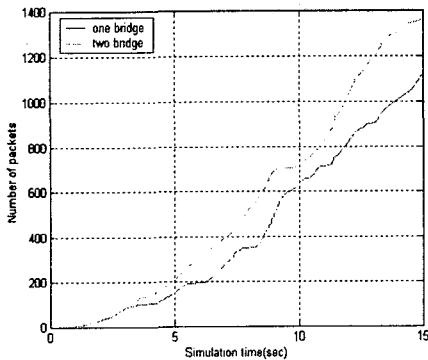


그림 5. bridge 노드 수에 따른 패킷 처리량의 변화

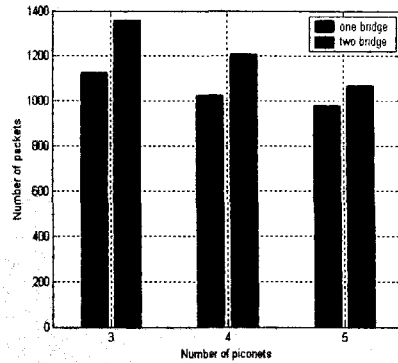


그림 6. piconet수의 변화에 따른 패킷 처리량 변화

5. 결론 및 향후 연구과제

Bluetooth piconet의 결합된 형태인 scatternet은 그 형태에 따라 망 전체적인 성능에 큰 영향을 미치게 된다. 따라서 그 형태는 scatternet 구성에서 큰 비중을 차지하게 된다. 본 논문에서는 기존에 제시된 방법과는 조금 다른 망 구성 형태인 2개 까지의 bridge 노드를 사용하는 방법을 제시한다. 이 형태는 1개의 bridge보다 더 원활한 통신을 가능하게 하고 안정적이면서 처리량면에서도 높은 결과를 보임을 알아보았다. 아직도 scatternet 구성 형태는 여러 가지 문제점들이 있다. 그래서 라우팅 방법이나 스케줄링등 여러 가지 효율성을 위한 요소들을 더 고려하여 더 나은 scatternet 구성 형태와 거기에 따른 시뮬레이션으로 성능을 평가하는 연구가 계속 진행되고 있다.

6. 참고 문헌

- [1] Bluetooth SIG, *Bluetooth Baseband Specification Version 1.0B*, <http://www.bluetooth.com>
- [2] Theodoros Salonidis, Pravin Bhagwat, Leandros Tassioulas, and Richard LaMaire, "Distributed topology construction of Bluetooth personal area networks," in *Proc. IEEE INFOCOM*, Anchorage, AK, April 2001.
- [3] Gy. Miklos, A. Racz, Z. Turanyi, A. Valko, and P. Johansson, "Performance aspects of Bluetooth scatternet formation," in *Proceedings of The First Annual Workshop on Mobile Ad Hoc Networking and Computing*, 2000.
- [4] Godfrey Tan, Allen Miu, John Guttag, Hari B., "Forming Scatternets from Bluetooth Personal Area Networks", *MIT Technical Report, MIT-LCS-TR-826*, October 2001.
- [5] G.V. Zaruba, S. Basagni, I. Chlamtac, "Bluetrees - scatternet formation to enable Bluetooth-based ad hoc networks", *IEEE International Conference on Communications (ICC)2001*, pp.273-277.
- [6] Manish Kalia, Sumit Garg, Rajeev Shorey, "Scatternet Structure and Inter-Piconet Communication in the Bluetooth System", *IEEE National Conference on Communications 2000*, New Delhi.
- [7] <http://oss.software.ibm.com/developerworks/projects/>