

블루투스 스캐터넷 형성 정책 : 모바일 환경에 대한 효과적인 지원

김수곤⁰ 김길용
부산대학교 컴퓨터공학과
(sgkim2, gykim)@hyowon.pusan.ac.kr

Bluetooth Scatternet Forming Policy : Effective Support of Mobile Environment

Su-gon Kim⁰ Gil-Yong Kim
Dept. of Computer Engineering, Pusan National University

요 약

블루투스는 저렴한 칩 가격과 간섭에 강한 특성을 바탕으로 휴대폰, PDA, 노트북 등 모바일 디바이스를 주요 대상으로 하는 좁은 범위의(short-range) 무선 ad-hoc 네트워크를 구성하기 위한 유력한 기술로 떠오르고 있다. 블루투스는 대부분의 무선 ad-hoc 네트워크 기술들이 단일 채널의 방송 기반(broadcast based) 방식에 바탕을 두고 있는 것과는 달리, 주파수 호핑(frequency-hopping)을 기반으로 하는 다중 채널 방식에 바탕을 두고 있다. 그러므로 각각의 채널을 서로 연결하여 스캐터넷(scatternet)이라고 하는 하나의 네트워크를 형성하게 되는데, 이 때 각 채널을 연결하는 방법에 따라 다양한 형태의 스캐터넷을 형성할 수 있다. 현재 발표되어 있는 스캐터넷 형성 정책들은 모바일 환경에 대한 고려가 부족하여, 노드들의 이동이 빈번한 모바일 환경을 효과적으로 지원하지 못하고 있다. 본 논문에서는 각 노드들이 빈번하게 스캐터넷에 추가되고 삭제되는 모바일 환경에서 정확하고 효과적으로 스캐터넷을 형성할 수 있는 정책을 제시한다.

1. 서 론

블루투스는 저렴한 칩 가격과 간섭에 강한 특성을 바탕으로 휴대폰, PDA, 노트북 등 모바일 디바이스를 주요 대상으로 하는 좁은 범위의(short-ranged) 무선 ad-hoc 네트워크를 구성하기 위한 유력한 기술로 떠오르고 있다.

IEEE 802.11과 같은 대부분의 무선 ad-hoc 네트워크 기술들은 단일 채널의 방송 기반(broadcast based) 방식에 바탕을 두고 있으므로, 네트워크의 모든 노드들이 CSMA 방식의 MAC 프로토콜을 통해 하나의 채널을 공유하게 되어 네트워크의 형태(topology)가 메시형(mesh topology) 하나로 결정된다. 그러나 블루투스의 경우 주파수 호핑(frequency-hopping)을 기반으로 하는 다중 채널 방식에 바탕을 두고 있기 때문에 [1] 각각의 채널을 연결하는 방법에 따라 다양한 형태의 네트워크가 형성될 수 있다. 즉, 하나의 통신 영역 안에 여러 개의 채널이 존재하므로, 각 노드들이 지리적으로 통신이 가능한 영역 안에 있는 경우에도, 서로 다른 채널에 있는 노드와 통신하기 위해서는 각 채널을 서로 연결해야만 한다. 이 때 각 채널들을 연결하여 형성된 네트워크를 스캐터넷(scatternet)이라 부르는데 [1], 채널들을 연결하는 방법에 따라 다양한 형태와 기능 및 성능을 갖는 스캐터넷이 형성될 수 있다.

현재 스캐터넷의 형성 정책과 관련된 연구는 그다지 많이 이루어져 있지 않은 상황이며, 발표된 정책들도 모바일 환경에 대한 고려가 부족하여, 노드들의 이동이 빈번한 모

바일 환경을 효과적으로 지원하지 못하고 있다. 하지만 블루투스 기술이 적용될 주요 대상 디바이스들이 모바일 디바이스라는 점을 감안하면 스캐터넷 형성 정책의 모바일 환경에 대한 지원은 반드시 고려되어야 하는 사항이다. 이에 본 논문에서는 각 노드들의 추가와 삭제가 빈번한 모바일 환경에서 정확하고 효과적으로 스캐터넷을 형성할 수 있는 정책을 제시한다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구로서 블루투스 연결 설정의 특징과, 모바일 환경에서 다른 스캐터넷 형성 정책들이 갖는 문제점들에 대해 살펴보고, 3장에서는 모바일 환경을 효과적으로 지원하는 스캐터넷 형성 정책을 제시한다. 그리고 4장에서는 본 논문의 결론을 맺고 향후 과제를 제시한다.

2. 관련 연구

본 장에서는 블루투스 디바이스간 연결이 이루어지는 과정과 모바일 환경에서 기존의 스캐터넷 형성 정책에 갖는 문제점에 대해 기술한다.

2.1 블루투스 연결 설정

두 블루투스 디바이스가 연결되기 위해서 각 디바이스는 서로 다른 역할을 수행한다. 연결을 시도하는 디바이스는 마스터(master)가 되며, 다른 디바이스는 슬레이브(slave)가 된다. 마스터는 inquiry 상태(state)에서 통신 영역 내에 있는 다른 디바이스에 대한 정보

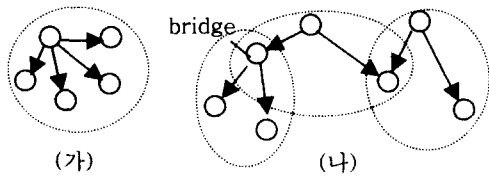


그림 1: (가) 피코넷 (나) 스캐터넷

(주소 및 클럭)를 수집하며, 슬레이브는 마스터가 자신을 발견할 수 있도록 inquiry scan 상태에서 정보를 제공한다.

그 후 마스터는 page 상태로 들어가서 수집한 정보를 이용하여 슬레이브에 대해 paging을 수행한다. 이 때 page scan 상태에 있는 슬레이브가 이에 응답하면 연결이 이루어진다. 이와 같은 방법으로 하나의 마스터는 7개 까지의 액티브 슬레이브(active slave)와 연결이 가능하며, 슬레이브들은 마스터를 통해서만 서로 통신이 가능하다. 이렇게 형성된, 그림 1의 (가)와 같은 형태를 갖는 네트워크를 피코넷(piconet)이라 하며, 하나의 피코넷이 하나의 채널이 된다.

하나의 피코넷은 8개의 노드만을 연결할 수 있기 때문에 [1], 하나의 통신 영역에 존재하는 더 많은 노드들을 연결하기 위해서는 여러 개의 피코넷을 연결하여 그림 1의 (나)와 같은 스캐터넷을 구성하게 된다. 이를 위해서 두 개 이상의 피코넷에 동시에 참여하면서 피코넷 사이의 패킷 전송을 담당하는 브릿지(bridge) 노드가 필요하게 되는데, 브릿지 노드를 배치하는 방법에 따라 다양한 스캐터넷을 형성할 수 있다.

2.2 Bluetree 와 BTCP

피코넷의 각 슬레이브를 새로운 피코넷의 마스터가 되게 하는 방식으로 네트워크를 확장해서 구성할 수 있는 트리 형태의 스캐터넷인 bluetree는 [2] 각 노드가 주위의 노드들에 대한 정보를 부팅 시에 수집하여 이미 가지고 있다는 가정으로 시작한다. 하지만 노드의 이동이 빈번한 모바일 환경에서는, 연결이 이루어지는 시점에서 각 노드들이 가지고 있는 정보가 부팅 시의 정보와 같다는 보장을 할 수 없으므로, 스캐터넷을 올바르게 구성할 수가 없게 된다.

리더 선출(leader election) 과정을 통해 주위의 노드들에 대한 정보를 하나의 노드에 모아주고, 이 노드가 조정자(coordinator)가 되어 각 노드에 적절히 임무를 부과하여 스캐터넷을 구성하는 BTCP (Bluetooth Topology Construction Protocol) [3]의 경우 동적으로 정보를 수집함으로써 [2]의 방법에 비해 모바일 환경을 보다 효과적으로 지원하고 있다. 하지만 [3]에서 설명하는 바와 같이 리더 선출 과정은 비교적 시간이 많이 걸리는 과정이므로, 리더 선출 과정 중에 주위 노드들의 이동이 발생할 수 있으므로, 조정자는 잘못된 정보를 이용하여 스캐터넷을 구성하게 될 수 있다.

3. 모바일 환경을 효과적으로 지원하는 스캐터넷 형성 정책

본 장에서는 모바일 환경을 효과적으로 지원하는 스캐터넷 형성 정책이 갖추어야 할 조건을 제시하고, 제시된 조건을 만족하는 정책을 제안한다.

3.1 모바일 환경을 효과적으로 지원하기 위해 갖추어야 할 조건

노드들의 이동이 빈번한 모바일 환경에서는 스캐터넷을 형성하고 있는 중에도 노드의 이동이 발생할 수 있으며, 이 경우 이전에 수집한 정보는 쓸모없게 되어, 올바른 스캐터넷의 구성이 불가능하게 된다. 따라서 정확한 스캐터넷을 구성하기 위해서는 특정 노드의 정보를 수집하는 시점(inquiry)과 그 노드와의 연결을 시도하는 시점(paging) 사이의 간격이 최소화 되어야 하며, 새로운 노드의 추가와 삭제가 발생하더라도 스캐터넷을 계속 유지할 수 있는 방법이 있어야 한다. 그러므로 스캐터넷 형성 정책은 스캐터넷을 노드의 추가 및 삭제가 계속해서 발생하는 동적인 것으로 인식하고 이를 효과적으로 처리할 수 있는 것이어야 한다.

3.2 모바일 환경을 효과적으로 지원하는 스캐터넷 형성 정책

본 논문에서는 그림 2와 같은 링형 스캐터넷의 형성 정책을 제시한다. 링의 body를 형성하는 노드들은 모두 마스터-슬레이브 브릿지가 되며, 각 body 노드에 leaf 노드들이 연결되어 있는 형태이다. 링형 스캐터넷에서는 각 body 노드들이 자신이 마스터인 피코넷에서 슬레이브인 leaf 노드들에 대한 정보를 관리하는 것만으로 라우팅이 가능하므로, 라우팅 정보 업데이트의 부담이 적어, 노드의 추가, 삭제가 빈번한 모바일 환경에 매우 적합하다.

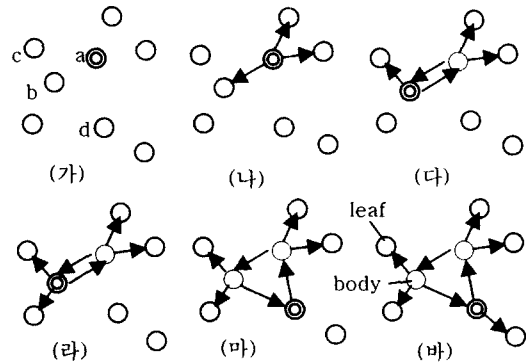


그림 2: 링형 스캐터넷 형성 과정 (◎ 는 Inquiry 수행중인 노드, → 의 꼬리는 마스터, 머리는 슬레이브)

본 논문의 링형 스캐터넷은 조정자(Coordinator) 노드를 중심으로 형성되며, 조정자 노드는 노드들의 추가 및 삭제에 따른 스캐터넷의 재구성을 담당한다. 각 노드는 디바이스의 처리 능력에 따라 7 이하의 범위에서 서로 다른 MAX_SLAVES 값(연결 가능한 최대 슬레이브 개수)을 설정할 수 있는데, 다음의

두 절에서 모든 노드의 MAX_SLAVES 값을 3이라 가정하고 그림 2와 3을 통해 스캐터넷의 형성 및 재구성 방법을 설명한다.

3.2.1 노드의 추가를 통한 링형 스캐터넷 형성 과정

노드의 추가를 통한 링형 스캐터넷 형성 과정은 그림 2와 같다.

- (가) 스캐터넷을 형성하고자 하는 하나의 노드만 조정자 노드가 되어 inquiry를 수행하고, 다른 노드들은 inquiry scan과 page scan 이 가능한 상태로 통신 영역 내에 있다. a 가 조정자.
- (나) Inquiry의 결과로 수집한 주위 노드들의 정보를 이용하여 즉시 paging을 수행하여 연결하는 방법으로 leaf 노드들을 추가해 나간다.
- (다) 연결된 leaf 노드의 개수가 MAX_SLAVES와 같은 상황에서, 다른 노드 c가 inquiry의 결과로 발견되면, 조정자인 a는 자신의 leaf 노드 중 하나인 b에게, 조정자와 새로 발견된 노드의 정보를 전송하고, b를 마스터-슬레이브 브릿지가 되게 하여 링의 body 노드로 만든다. 이후 b는 paging을 통해 조정자와 c에 연결되고, a는 inquiry를 멈추고, b가 inquiry를 수행하게 된다.
- (라) b는 계속 inquiry를 수행하여 주위 노드들(새로 통신 영역으로 이동해 온 노드들 포함)의 정보를 수집하고, 곧 바로 paging을 수행하여 연결하는 방식으로 leaf 노드들을 추가해 나간다.
- (마) b의 leaf 노드 개수가 MAX_SLAVES와 같은 상황에서 다른 노드 d가 발견되면, b는 a와의 연결을 끊고 d와 연결한 후, d에게 자신이 저장하고 있는 조정자의 정보를 전송하고 d를 마스터-슬레이브 브릿지로 만들어 body 노드가 되게 한 후, d가 paging을 통해 조정자와 바로 연결할 수 있도록 한다. b는 inquiry를 멈추고, d가 inquiry를 수행한다.
- (바) (라),(마)의 과정을 반복하여, 스캐터넷을 확장해 나간다.

3.2.2 노드의 삭제 시 스캐터넷 재구성 과정

Body 노드의 leaf 노드들이 삭제되는 경우에는 별도의 재구성이 필요치 않다. body 노드의 현재 leaf 노드 개수를 확인하여 MAX_SLAVES보다 적으면 body 노드가 inquiry를 수행하여, MAX_SLAVES와 개수가 같아질 때까지 leaf 노드를 추가해 나가다가, 개수가 MAX_SLAVES와 같아지면 다시 inquiry 수행을 중지하도록 한다.

하지만, body 노드들이 삭제되는 경우에는 링이 끊어지게 되므로 스캐터넷의 재구성이 필요하다. body 노드의 삭제 시 스캐터넷 재구성 과정은 그림 3과 같다.

- (가) 연결된 링에서 두 body 노드가 삭제된다.
- (나) body 노드의 삭제와 동시에 body 노드의 앞, 뒤에 연결되어 있던 노드들은 이것을 감지하게 되고, 이 노드들은 즉시 paging을 통해 조정자 노드에 연결하여, 삭제된 노드와 자신에 대한 정보를 제공한다. 이와 같은 방법으로 조정자 노드는 끊어진 링

을 복원할 수 있는 정보를 수집하게 되고, 이 정보를 이용하여 스캐터넷을 재구성한다.

- (다) 조정자 노드는 해당 노드에게 연결해야 할 대상 노드의 정보를 제공하고, 노드들은 조정자 노드에서 얻은 정보를 이용해 paging을 수행하여 연결한다. 이런 방법으로 링이 복구되면, 각 노드는 조정자 노드와의 연결을 끊는다. 삭제된 body 노드의 leaf 노드들은 통신 영역에 새로 들어온 노드와 같이 inquiry scan, page scan 가능한 상태로 다시 연결되기를 기다린다.

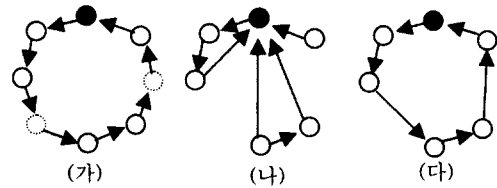


그림 3: body 노드 삭제 시 스캐터넷 재구성 과정 (● 는 조정자 노드, 단순화를 위해 각 body 노드에 연결된 leaf 노드들은 도시하지 않았음)

4. 결론 및 향후 과제

본 논문에서는 모바일 환경을 효과적으로 지원하는 링형 스캐터넷 형성 정책을 제시하였다. 제시된 정책은 주위 노드들에 대한 정보 수집(inquiry)후 즉시, 연결(paging)을 시도하여, 잘못된 정보를 이용하여 스캐터넷 구성을 시도하게 될 위험을 최소화 하였으며, 스캐터넷 형성 중에도 노드의 추가 및 삭제가 발생할 수 있으므로, 조정자 노드를 이용하여, 이에 효과적으로 대처할 수 있도록 하였다.

향후 본 논문에 제시된 정책을 보다 구체화하여, 하나의 프로토콜로 구현하고, 이것을 기존의 블루투스 프로토콜 스택과 조화시키기 위한 연구가 더 이루어져야 할 것이다.

5. 참고 문헌

- [1] Bluetooth SIG, " Bluetooth Specification Version 1.1" <http://www.bluetooth.com>
- [2] Zaruba G.V. , Stefano Basagni, Imrich Chlamtac, " Bluetrees - Scatternet formation to enable bluetooth-based ad hoc networks" , Communications, 2001 IEEE
- [3] Salonidis T. , Pravin Bhagwat, Leandros Tassioulas, Richard LaMaire " Distributed topology construction of bluetooth personal area networks" , INFOCOM 2001, IEEE