

모바일 애드-혹 네트워크를 위한 노드 ID 기반 서비스 디스커버리 기법

강은영*

Node ID-based Service Discovery for Mobile Ad Hoc Networks

Eun-Young Kang *

요 약

본 논문에서는 서비스 광고의 P2P 캐시 기법과 노드 ID를 기반으로 한 서비스 검색 기법을 혼합한 효율적인 서비스 디스커버리 기법을 제안한다. P2P 캐싱 광고 기법은 이웃 노드에 서비스 정보를 캐시 하기 때문에 빠르게 서비스 광고 정보를 확산시키며 서비스 검색 평균 홉 수를 적게 한다. 또한 노드 ID를 기반으로 한 서비스 검색은 모든 이웃 노드에게 메시지를 브로드캐스트하지 않기 때문에 네트워크 부하를 감소시켜 네트워크 전송 지연이 거의 발생하지 않는다. 제안하는 기법은 중앙 서버나 저장소를 사용하지 않으며 많은 메시지를 생성하는 플러딩 방식도 사용하지 않는다. 실험 결과는 제안하는 방식이 전통적인 플러딩 방식과 비교하여 이웃 노드의 적절한 선택으로 많은 메시지 수를 줄이고 평균 탐색 거리를 줄임으로서 전체 네트워크 로드와 응답 시간을 향상시킴을 보인다.

Abstract

In this paper, we propose an efficient service discovery scheme that combines peer-to-peer caching advertisement and node ID-based selective forwarding service requests. P2P caching advertisement quickly spreads available service information and reduces average response hop count since service information store in neighbor node cache. In addition, node ID-based service requests can minimize network transmission delay and can reduce network load since do not broadcast to all neighbor node. Proposed scheme does not require a central lookup server or registry and not rely on flooding that create a number of transmission messages. Simulation results show that proposed scheme improved network loads and response times since reduce a lot of messages and reduce average response hop counts using adaptive selective nodes among neighbor nodes compared to traditional flooding-based protocol.

▶ Keyword : 모바일 애드-혹 네트워크(Mobile Ad-hoc Networks), 서비스 디스커버리(Service Discovery), 파일 탐색(File Discovery), 피어-투-피어(Peer-to-Peer)

• 제1저자 : 강은영

• 투고일 : 2009. 10. 01, 심사일 : 2009. 10. 24, 게재확정일 : 2009. 12. 24.

* 동양공업전문대학 전기전자통신공학부 전임강사

I. 서론

모바일 애드-혹 네트워크(Mobile Ad-Hoc Network: MANET)는 기존에 설치된 유선망이나 기지국의 도움 없이 이동 노드들 간에 자율적으로 구성되는 네트워크다. 최근 무선 통신 장비 가격의 저렴화와 무선 이동 통신 기술의 발달로 언제 어디서나 컴퓨팅이 가능하면서도 통신비용이 저렴한 모바일 애드-혹 네트워크에 대한 관심이 높아지고 있다. 이는 모바일 애드-혹 네트워크가 더 이상 단일의 독립적인 네트워크로서의 기능을 수행할 뿐 아니라 기존 인프라 기술과 접목하여 유비쿼터스 컴퓨팅 환경이 가능하기 때문이다[1]. 이런 모바일 애드-혹 네트워크에서 사용자가 원하는 서비스나 파일 등의 자원을 찾고, 이를 사용할 수 있게 하는 서비스 디스커버리(service discovery)는 중요하다.

모바일 애드-혹 네트워크의 가장 큰 특성은 네트워크를 구성하는 이동 노드들이다. 이동 노드들의 자원은 (배터리, 프로세서, 메모리, 지원 장치, 전송 대역폭 등) 제한적이다. 모바일 애드-혹 네트워크에서 이동 노드는 서비스 제공자인 서버이자 서비스 요청 노드인 클라이언트이며 라우터로서도 동작한다. 그러므로 다른 노드를 대신하여 패킷을 전달하기도 한다. 또한 배터리로 동작함으로써 제한된 시간 제약을 받는다. 이동 노드는 메시지를 전송하고 수신할 때 많은 에너지를 소모한다. 그러므로 검색 과정 중에 생성되고 전송되는 전송 메시지수와 수신 메시지, 즉 네트워크 로드를 줄이는 것이 중요하다. 또한 모바일 애드-혹 네트워크는 무선 채널을 사용하므로 전송 거리와 전송 대역폭에 제약을 받는다. 특히 이동 노드는 제한된 무선 전송 거리를 가지기 때문에 목적지에 대한 정보는 이웃 노드 또는 전달 경로 상에 있는 중간 노드에 의해 결정된다. 모바일 애드-혹 네트워크에서 노드들의 이동으로 인하여 서비스 응답이 왔다 하더라도 중간 노드들이 이미 이동해버린 뒤라면 응답 메시지가 아무 소용이 없다. 그러므로 빠른 서비스 응답이 요구된다. 이런 이동 노드들의 특징과 무선 채널의 특징으로 동적으로 토폴로지가 변하는 모바일 애드-혹 네트워크에서 네트워크 전체에 메시지를 브로드캐스트하지 않고, 특정 노드를 중앙 서버로 사용하지 않으면서 알맞은 서비스를 검색하는 것은 서비스 디스커버리의 중요한 이슈이다. 모바일 애드-혹 네트워크에서 사용자가 원하는 서비스를 빠른 시간 내에 찾아내고 네트워크 로드를 줄여야 하는 기법이 서비스 디스커버리의 필수적인 문제이다.

기존의 대표적인 서비스 디스커버리 기법으로는 SLP, UPnP, 제한된 플러딩 기법 등이 있다. 그러나 SLP는 중앙

서버를 유지하는 중앙 디렉터리 기반 서비스 디스커버리 기법으로 서비스는 중앙 서버에 광고와 등록을 한다. 디스커버리 프로토콜은 서비스가 등록된 중앙 서버를 검색하는 기법이다. 이런 프로토콜들은 중앙 서버에 부하가 집중되고 서버에 병목 현상이 발생하는 등 문제점이 있어 동적이고 자원이 부족한 모바일 애드-혹 네트워크에는 적합하지 않다. UPnP, SLP2와 같은 프로토콜들은 802.11과 같은 무선 네트워크 기술을 도입하여 모바일 디바이스를 사용할 수 있도록 하였다. 그러나 아직은 모바일 환경보다는 유선 환경에 더 적합하도록 되어 있고 근본적인 문제 해결은 이루어지지 않았다. 또한 최근에 모바일 애드-혹에서 제한된(2홉) 플러딩을 통한 서비스 검색을 위한 기법들이 제안되었으나 여전히 플러딩 기법을 사용하여 많은 메시지를 생성하여 네트워크 비용이 크므로 자원이 부족하고 무선 환경인 모바일 애드-혹 네트워크에서는 효율적이지 않다.

본 논문에서는 모바일 애드-혹 네트워크를 위한 서비스 디스커버리 프로토콜을 제안한다. 제안하는 기법은 노드의 ID 기반 서비스 검색과 Peer to Peer (P2P) 캐시를 통한 이웃 노드의 여러 정보(노드 ID 정보, 서비스 정보)를 사용하여 서비스 검색 성능을 향상시킨다. P2P 캐시 기법은 이웃 노드에 서비스 정보를 캐시하기 때문에 빠르게 서비스 광고 정보를 확산시키며, 서비스 정보 제공으로 서비스 검색을 신속하게 하며 서비스 검색 평균 탐색 거리(홉 수)를 적게 한다. 또한 노드 ID를 기반으로 한 서비스 검색은 모든 이웃 노드에게 메시지를 브로드캐스트하지 않기 때문에 서비스 검색 메시지를 줄여 네트워크 전송 지연이 발생하지 않고 네트워크 부하를 감소시킨다. 제안된 기법은 브로드캐스트로 메시지를 전달하지 않고 중앙 디렉터리를 사용하지 않아 분산 환경인 모바일 애드-혹 네트워크에 적절하며 주고받는 메시지 수를 줄이고, 평균 탐색 거리를 줄임으로서 전체 네트워크 비용과 응답 시간을 향상시킬 수 있다.

본 논문의 구성은 다음과 같다. 2장에서는 기존의 서비스 검색을 위한 여러 가지 기법들을 살펴보고, 3장에서는 제안한 서비스 디스커버리에 대하여 설명하고, 4장에서는 성능 평가와 그 결과를 나타낸다. 마지막 5장에서는 결론을 맺는다.

II. 관련 연구

네트워크에 속해 있는 임의의 노드는 다른 노드에서 제공하는 서비스를 사용하기 위해서는 사용자가 원하는 서비스를 어떤 노드에서 제공하는지를 알아야 하며 이를 위해서 서비스 검색을 한다. 이런 서비스 디스커버리 과정은 먼저 자신이

제공하는 서비스를 다른 사용자가 사용할 수 있도록 하여야 하며 이를 위해서는 서비스 광고를 한다. 또한 원하는 서비스를 찾기 위한 검색 과정이 실행되며 이 과정에서 원하는 서비스를 찾았을 때 서비스 응답을 한다. 이런 일련의 과정을 통하여 서비스 디스커버리가 이루어진다.

서비스 디스커버리 기법은 크게 세 가지로 구분된다. 첫째, 중앙 집중 형 디렉터리 기반 서비스 탐색기법으로, 대표적인 프로토콜로는 Service Discovery Protocol(SLP)(2), UDDI(3), Jini(4) 등이 있다. 이들은 중앙 룩업 서버를 유지하면서 이를 이용해 서비스 광고 및 서비스 검색을 수행한다. 이러한 중앙 집중 방식은 중앙 룩업 서버 역할을 하는 특정 노드를 두어야 하므로 특정 기반 시설 없이 동작해야 하고 네트워크 특성이 동적인 모바일 애드-혹 네트워크에는 적합하지 않다. 또한 서버를 두더라도 서버에서 병목현상이 발생하며, 서버 노드의 이동성 문제로 노드들이 많은 대규모의 모바일 애드-혹 네트워크에는 적합하지 않다. 둘째, 분산 디렉터리 기반 서비스 탐색 기법으로, 대표적인 프로토콜로는 확장성 있는 서비스 탐색 기법(SSD, Scalable Service Discovery)(5), 애드-혹 망에서 효율적인 P2P 시스템(6), 모바일 애드-혹 네트워크에서 분산 해쉬 테이블 기반의 서비스 디스커버리 기법(7) 등이 있다. 이 기법들은 분산된 디렉터리를 지원하는 기법이다. 분산된 서비스 디스커버리에서는 물리적인 계층과 분산된 디렉터리를 지원하는 대표 노드들로 구성된 가상 계층의 2계층으로 구분한다. 대표 노드는 이웃 노드들의 서비스 광고 정보를 모집하고 디렉터리에서 이웃 노드들의 정보를 유지한다. 서비스 검색은 현재 노드가 속해 있는 대표 노드에게서 서비스를 검색하고, 이 대표 노드에 원하는 서비스가 없을 경우 가상 계층에 있는 다른 대표 노드에게 원하는 서비스를 검색함으로써 전체 네트워크를 검색하게 된다. 가상 계층은 대표 디렉터리들로서 활동하는 모바일 애드-혹 네트워크의 노드들의 부분집합으로 구성된다. 그러나 대표 노드들이 계속적으로 변하고 이를 선출하고 유지하기 위한 오버헤드가 크다. 셋째, 디렉터리가 없는 분산 서비스 디스커버리 기법은 멀티캐스트 및 플러딩 방식을 사용하는 방식이 있다. 플러딩 방식의 대표적인 프로토콜로는 Salutation(8), UPnP(9), Hybrid기법(10), ORION(11) 등이 있다. 플러딩 방식은 중앙 룩업 서버를 사용하지 않고 분산 방식으로 멀티캐스트나 플러딩 방식을 사용하여 서비스를 광고하고 서비스를 탐색하므로 중앙 집중식 보다는 모바일 애드-혹 네트워크에 적합하다. 그러나 플러딩 기법은 많은 수의 메시지를 생성하여 통신 시에 많은 전력을 소모하게 되어 자원이 부족한 모바일 노드들에 적합하지 않으며, 네트워크 부하가 많아져 모바일 애드-혹 네트워크에서는 비효율적인 방법이다.

모바일 애드-혹 네트워크상에서 동작하는 노드의 이동성과

네트워크 토폴로지의 동적 특징을 고려했을 때, 검색 응용 모델은 P2P와 유사한 특징을 가진다(12). 따라서 두 네트워크의 공통점을 잘 이용하여 모바일 애드-혹 네트워크에 적합한 P2P 검색 응용 모델에 대한 개발이 이루어져야 한다(13). 모바일 애드-혹 네트워크는 기존의 고정된 유선망과 달리 노드들이 자유롭게 이동하기 때문에 모바일 애드-혹 네트워크가 갖는 특성들을 고려하여 검색이 수행되어야 한다.

III. 제안하는 서비스 디스커버리 프로토콜

이 장에서는 모바일 애드-혹 네트워크에 효율적인 서비스 디스커버리 프로토콜에 대하여 제안한다. 제안하는 기법은 노드 ID 기반 서비스 디스커버리뿐만 아니라 P2P 캐시를 기반으로 이웃 노드의 여러 정보(노드 ID 정보, 서비스 정보)를 사용하여 서비스 검색 성능을 향상시킨다.

3.1 정보 캐싱을 통한 이웃 노드 정보 활용

모바일 애드-혹 네트워크에서 각 노드는 이웃 노드의 정보를 통하여 서비스 디스커버리의 검색 효율성을 증대 시킬 수 있다. 본 논문에서는 물리적인 이웃 노드들의 여러 정보를 캐싱하여 이를 서비스 디스커버리에 활용하는 것이다. 라우팅 프로토콜의 hello 메시지는 무선 전송 범위 내에 있는 이웃들에게 주기적으로 1-홉 브로드캐스트 한다. 각 노드의 이웃노드 서비스 정보를 알기 위하여 hello 메시지에 노드의 ID, 서비스 정보를 포함하여 전송한다. 그러므로 따로 이웃노드의 정보를 얻기 위한 메시지를 사용하지 않는다. 이웃노드로부터 받은 메시지를 기반으로 이웃노드의 존재와 정보를 인접노드 리스트에 저장한다. 이웃노드로부터 일정기간 hello 메시지를 받지 못하면 이웃노드가 이동하여 더 이상 이웃하지 않음을 의미하므로 인접노드리스트에서 그 노드는 제외하면 된다.

3.2 노드 ID 기반 메시지 전달 기법

노드 ID 기반 서비스 디스커버리의 기본 아이디어는 각 노드는 메시지를 보낼 때 자신의 모든 이웃노드들에게 메시지를 브로드캐스트하지 않고 이웃 노드들 중 자신의 노드 ID보다 작은 노드 ID를 가진 노드에게만 메시지를 전달한다. 이는 자기가 가지고 있는 파일 정보에 대하여 다른 노드에게 파일 정보를 광고할 때 사용될 수 있으며, 또한 사용자 노드가 필요한 파일을 찾기 위한 탐색을 위한 요청 메시지를 보낼 때도 사용될 수 있다. 그림 1은 노드 ID 기반 서비스 디스커버리 기법에 대한 예 이다.

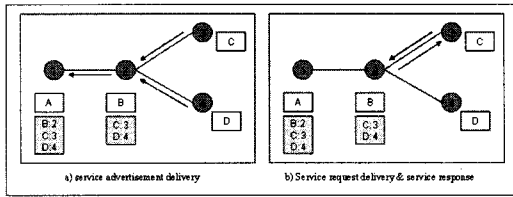


그림 1. 노드 ID 기반 서비스 디스커버리
Fig. 1. Node-ID based Service Discovery

그림 1(a)에서 네트워크에 참여한 각 노드는 자신이 가지고 있는 자신의 서비스 정보를 네트워크에 있는 다른 노드에게 광고한다. 노드 1은 A, 노드 2는 B, 노드 3은 C, 노드 4는 D의 파일을 각각 가지고 있다고 가정하자. 노드 3은 자신의 파일 정보를 자신의 노드 ID보다 작은 ID를 가진 이웃노드 2에게 광고 메시지를 전달한다. 이를 전달 받은 노드 2는 자신의 인접노드리스트에 파일 C와 그것의 소유 노드 주소 3에 대한 정보를 저장한다. 또한 노드 4도 자신의 ID보다 작은 ID를 가진 이웃노드 2에게 광고 메시지를 전달한다. 이 때 노드 2는 자신의 인접노드리스트에 파일 D와 그의 소유 노드 주소 4에 대한 정보를 저장한다. 마찬가지로 노드 2도 자신의 ID 보다 작은 ID를 가진 이웃노드 1에게만 광고 메시지를 전달한다. 이런 광고 과정을 되풀이함으로써 가장 작은 노드 ID를 가진 노드 1은 더 이상 광고 메시지를 전달하지 않는다. 이 때 노드1은 많은 파일 정보를 제공할 수 있다.

파일 정보를 검색하기 위한 요청 메시지도 이런 정책을 따른다. 예를 들면 그림 1(b)에서 노드 3은 파일 D를 찾기 위하여 자신의 노드 ID보다 작은 이웃노드 2에게 요청 메시지를 보낸다. 노드 2에는 파일 D에 대한 정보를 가지고 있으므로 바로 파일 D에 대한 검색을 성공적으로 마칠 수 있다. 이 기법은 중앙 디렉터리 구조 없이 분산된 환경에서 이웃 노드와의 협력을 통하여 제공되는 메시지 정보를 이용하여 낮은 ID를 가진 노드가 많은 정보를 제공한다. 이는 분산된 환경에서 플러딩하지 않고 낮은 ID를 가진 노드에게 메시지를 전달하여, 서비스 디스커버리에 사용되는 메시지의 수를 줄이고 메시지의 전달을 줄여 네트워크 로드에도 효율적이다.

3.3 서비스 검색

노드가 특정 서비스를 원할 때 해당 노드는 먼저 자신의 로컬 정보를 먼저 검색한다. 이때, 서비스를 자신의 로컬 캐시에서 발견하였다면, 이는 자신이 바로 서비스를 제공할 수 있는 서비스 제공자임을 의미한다. 만일 서비스가 자신의 로컬 캐시에서 발견되지 않았다면, 인접노드리스트에서 원하는 서비스가 있는지 확인한다. 이 때 서비스가 발견되면 이는 이웃노드가 서비스 제공자임을 알 수 있다. 만약 인접노드리스트에서도 서비스가 발견되지 않으면, 서비스 이름과 자신의

주소를 포함한 서비스 검색 메시지를 만들어 서비스 검색 요청을 시작한다. 이는 찾고자하는 서비스 검색 메시지를 만들어 노드 ID 기반 정책에 따라 선택된 이웃 노드들에게 전송한다.

만약 서비스를 원하는 노드가 일정 시간이 지난 후에도 원하는 서비스를 발견하지 못했을 경우, 다시 메시지 번호를 증가시켜 서비스 검색 메시지를 다시 보내는 재검색을 한다. 재검색 시에는 전략 기법을 반대로 하여 메시지를 전송한다. 즉, 처음 메시지 전송 시 노드 ID가 낮은 이웃 노드에게 메시지를 전송하였다면 이번에는 반대로 노드 ID가 큰 이웃 노드에게만 메시지를 전달함으로써, 확실성과 성공률을 동시에 높인다. 낮은 노드 ID 기법은 높은 노드 ID를 가진 노드에게만 메시지를 전달하는 높은 노드 ID기법과 동일한 기법이다. 서비스 요청과 디스커버리에 관한 알고리즘은 그림2에 나타내었다.

```

=====
Service_Request(S)
/* S: 서비스 이름, 요청 노드ID
LSC: Local Service Cache, NIC: Neighbor node
Information Cache */
//요청 service가 local 에 있으면 서비스 제공자임
if S ∈ LSC
    reply Service Response Message:
//요청 서비스가 이웃에 있으면 응답 메시지 보냄
else if S ∈ NIC
    reply Service Response Message:
endif
//노드 ID가 작은 이웃 노드에게 서비스 요청 메시지 전송
for each 이웃 노드 n in 인접노드리스트 엔트리
{
    if 서비스 검색 요청 메시지가 처음 전송하는
        메시지 이면
            If 요청노드ID >= n.인접노드ID
                forward Service_Request(S)
                to neighbor node n
            else
                return
        else // 메시지전송이 재전송인 경우
            if 요청노드ID < n.인접노드ID
                forward Service_Request(S)
            else
                return
        endif
    endif
}
=====

```

그림 2. 서비스 디스커버리 알고리즘
Fig. 2. Service Discovery Algorithm

3.3 서비스 검색

서비스 검색 메시지의 중간 전달 과정에서 요청한 서비스가 발견되거나, 서비스 제공자에서 발견되어지면, 노드는 발견한 서비스의 정보를 바탕으로 서비스 응답 메시지를 만든다. 서비스 응답 메시지는 서비스 검색 메시지에서 지정한 서비스, 그리고 이 서비스를 제공할 수 있는 서버의 IP 주소 정보를 포함한다. 수신된 서비스 응답 메시지 중 서비스의 여유 시간 값이 가장 큰 값을 가지는 서비스제공자를 선택하고, 선택된 서비스제공자에게 서비스를 호출한다.

캐시 일관성을 정확하게 유지하는 기존의 연구들이 있다 [14]. 그러나 모바일 애드-혹 네트워크에서 이동성과 전력 제한 때문에 캐시 일관성을 정확하게 유지하는 것은 너무 많은 오버헤드가 있다. 그러므로 캐시 일관성을 유지하기 위하여 본 논문에서는 *ttl* 메커니즘을 기반으로 한다. 캐시에 저장된 데이터 엔트리의 서비스 가능 시간 값이 초과되지 않았다면 적절한 값으로 간주한다. 만일 서비스 가능 시간 값이 일정 시간을 초과 하였다면 서비스가 부적합하므로 그 데이터 엔트리는 테이블에서 제거된다. 만일 지나가는 메시지가 같은 데이터 엔트리를 가지고 있다면 새로운 서비스 가능 시간 값을 최신 값으로 수정하여 가장 최근의 정보를 유지한다.

IV. 성능 평가

이 장에서는 본 논문에서 제안하는 서비스 프로토콜의 성능을 평가한다. 본 논문에서 제안하는 프로토콜, 낮은 노드 ID (Lower ID, LD)를 기존의 플러딩 방식(Flood)과 비교한다.

4.1 시뮬레이션 환경

모바일 애드-혹 네트워크에서 서비스 디스커버리의 성능평가를 위해 대표적인 네트워크 시뮬레이션 도구인 NS2 v2.32 [15]를 사용하였다.

실험 환경은 1000m x 1000m의 네트워크 영역에 이동하는 이동 노드들이 랜덤하게 분포되며 노드 ID도 랜덤하게 부여된 모바일 애드-혹 네트워크를 구성하였다. 또한 시뮬레이션 초기에 몇몇 노드들이 랜덤하게 서비스를 제공하는 서비스 제공자로서 선택된다. 서비스 제공자가 제공하는 서비스도 역시 랜덤하게 선택된다. 각 노드들은 평균 속도로 이동하며, 이동 중 일정 정지 시간만큼 정지한 후 임의로 선택된 목적지점까지 일정 속도로 이동한다. 이동 노드들의 이동 시나리오는 Random way-point 방식이 사용된다. 라우팅 프로토콜로 AODV를 사용하였다. 표 1은 알고리즘을 위한 시뮬레이션에 사용된 인자의 값들을 보인다.

표 1. 시뮬레이션 인자
Table 1. Simulation Parameters

인자	값
노드 수	20 ~ 100
네트워크 크기 (x, y)	1000m X 1000m
서비스 생성 시간 (초)	5
라우팅 프로토콜	AODV
이동 속도 (m/s)	2 ~ 10
정지 시간	2
이동 패턴	Random way-point
Networkinterface	Phy/WirelessPhy
채널	Channel/WirelessChannel
실행시간 (초)	250
무선 전송 범위 (m)	100 ~ 500
서버율 (%)	50

모바일 애드-혹 네트워크에서 이동 노드들은 배터리를 사용하고 노드들이 자유롭게 이동하며, 무선 채널을 사용하므로 전송 거리에 제약을 받는다. 특히 메시지를 송신할 때, 많은 에너지를 소비하므로 서비스 검색에 필요한 메시지 수, 즉 네트워크 로드와 평균 검색 거리가 중요한 평가기준이 된다. 그러므로 본 논문에서는 성능 평가 기준으로 네트워크 로드, 평균 탐색 거리, 검색 성공률, 탐색 요청 메시지 수로 한다. 네트워크 로드는 서비스 광고 메시지 수와 서비스 검색 메시지 수 서비스 응답 메시지를 합한 것이다. 평균 검색 거리는 서비스 요청 노드에서 검색을 시작하여 서비스 응답 메시지를 보낸 노드와의 거리 즉 평균 홉 수를 말한다. 검색 성공률은 전체 서비스 검색 중 서비스 검색을 성공한 비율이다.

4.2 성능 평가 결과

본 절에서는 제안한 방식의 실험 결과를 제시하고 분석한다.

4.2.1 노드 수에 따른 성능 평가 영향

노드의 수는 20개, 40개, 60개, 80개, 100개로 변화를 주어가며 실험하였다. 노드의 이동 속도는 10m/s 로 설정하고, 무선전송범위는 250m, 전체 노드 중 서비스를 제공하는 서비스 제공자, 즉 서버의 수는 50%로 하여 실험하였다.

그림 3은 노드 수에 따른 서비스 탐색 요청 메시지 수를 보여준다. 노드가 원하는 서비스를 찾기 위해서 사용한 서비스 검색 요청 메시지 전송량을 보인다.

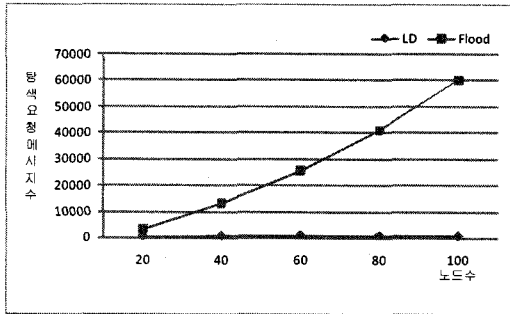


그림 3. 탐색 요청 메시지 수 노드수별
Fig. 3. The number of service request messages

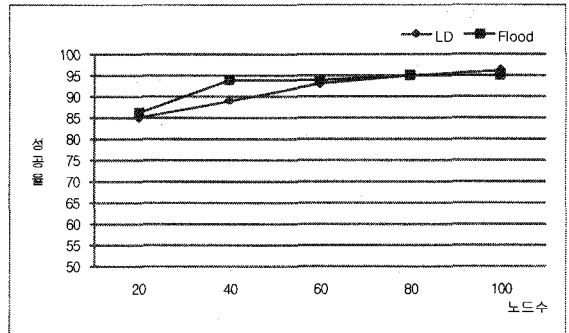


그림 6. 성공률 - 노드수별
Fig. 6. Success ratio

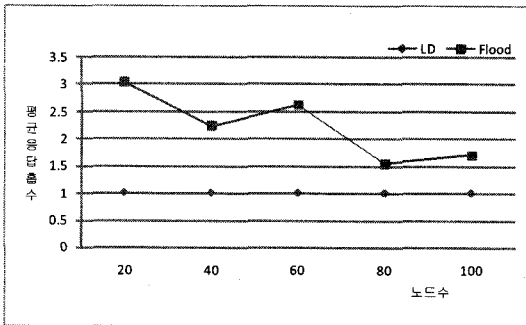


그림 4. 평균 응답 홉 수 노드수별
Fig. 4. The number of average response hop

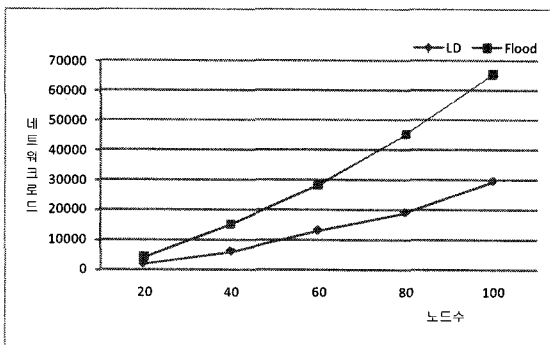


그림 5. 네트워크 로드 - 노드수별
Fig. 5. The network load

그림에서 보듯이 제안한 기법은 전체 탐색 요청 메시지 수가 노드수가 늘어나더라도 거의 증가하지 않으나 플러딩 기반 서비스 디스커버리 기법은 노드의 수가 늘어날수록 기하급수적으로 서비스 탐색 메시지수가 늘어난다. 노드수가 늘어날수록 메시지는 전달되지 못하고 지연되거나 폐기되는 경우가 생겨 네트워크에 많은 부담을 준다.

그림 4는 노드 수에 따른 평균 탐색 거리를 실험한 결과이다. 대부분의 경우 이웃한 노드의 로컬 캐시에 있는 정보를 이용하여 서비스를 찾게 되므로 0.5 ~ 2홉 사이에서 원하는 서비스를 찾는다. 0홉에서 찾는다라는 것은 본인이 가지고 있는 로컬에서 원하는 정보를 찾는다라는 것을 의미한다. 그림에서 나타내듯이 제안하는 기법들은 대부분 2홉 이내에 있는 이웃 노드에서 원하는 서비스를 찾고 플러딩 기법에서는 10홉이 넘는 거리에서 원하는 서비스를 찾는다. 본 논문에서 제안한 방식이 플러딩 기법 보다 평균 탐색 거리가 가까워 가장 빠른 검색 결과를 보여 좋은 성능을 보인다. 이는 서비스 요청자가 서비스 요청 메시지를 보낸 후 서비스 응답 메시지를 받을 때까지의 거리로 거리가 짧을수록 검색에 빨리 성공했으며, 주고받는 메시지 수도 작아 질 수 있다.

그림 5는 노드 수에 따른 네트워크 로드를 보인다. 플러딩 기반 서비스 디스커버리 기법은 서비스 광고 메시지를 보내지 않는다. 플러딩 기법은 네트워크에 있는 모든 도달 가능한 노드에게 탐색 요청 메시지를 보낸다. 플러딩 기법에서 중복되게 받은 메시지는 다시 보내지 않고 폐기 하며, 새로 받은 메시지만 탐색 요청을 위해 전달한다. 그림 5에서 나타내듯이 노드수가 증가 할수록 플러딩 기법에서는 네트워크 로드가 기하급수적으로 늘어나나, 제안하는 기법은 노드수가 늘어나더라도 네트워크 로드가 증가하는 율은 적어 좋은 성능을 나타냈다. 이는 서비스 제공자가 서비스 광고 메시지를 보낸 후, 이를 받은 d-hop 이웃 노드들은 이 정보를 저장하여 서비스

검색 요청을 받았을 때, 유용하게 사용될 수 있어 네트워크 로드를 줄일 수 있다.

그림 6은 각 프로토콜별로 노드 수에 변화를 주어가며 검색 성공률을 모의 실험한 결과이다. 본 논문에서 제안한 방식(LD)이 플러딩 기법과 거의 유사한 성공률을 보임을 알 수 있다. 대부분의 노드 수에서 성공률은 90%이상을 보인다. 만일 서비스 광고 시 서비스에 대한 정보가 네트워크상에 효율적으로 캐시 된다면, 검색 성공률이 높아질 것이다.

4.2.1 이동 속도에 따른 성능 평가 영향

노드의 이동 속도를 2, 4, 6, 8, 10 m/s로 변화를 주어가며 실험하였다. 노드의 수는 80개로 하고, 무선전송범위는 250m, 전체 노드 중 서비스를 제공하는 서비스 제공자, 즉 서버의 수는 50%로 하여 실험하였다.

그림 7은 이동 속도별에 따른 탐색 요청 메시지 수를 나타낸다. 그림에서 보듯이 제안한 기법들은 전체 탐색 요청 메시지가 이동 속도가 증가할수록 탐색 요청 메시지가 줄어든다. 이는 이동 속도가 증가할수록 이웃하고 있는 노드수가 늘어나므로 서버에 있는 서비스정보가 이웃 노드에게 전파되어 캐시 됨으로서 쉽게 찾을 수 있음을 의미한다. 플러딩-기반 서비스 디스커버리 기법은 노드의 제안한 기법들에 비해 현저히 많은 탐색 요청 메시지가 발생하며 이는 네트워크 로드에 많은 부담을 준다.

그림 9는 이동 속도에 따른 네트워크 로드를 보여준다. 모바일 노드의 이동 속도가 변하더라도 네트워크 로드는 크게 변하지 않는다. 이는 모바일 노드의 이동 속도가 더 빠를수록 이웃노드에 저장되는 캐시 정보가 늘어남으로서 네트워크 로드가 조금 줄었다. 제안하는 기법은 이동 속도에 크게 영향을 받지 않고, 이동하는 모바일 환경에서도 잘 적응됨을 보인다.

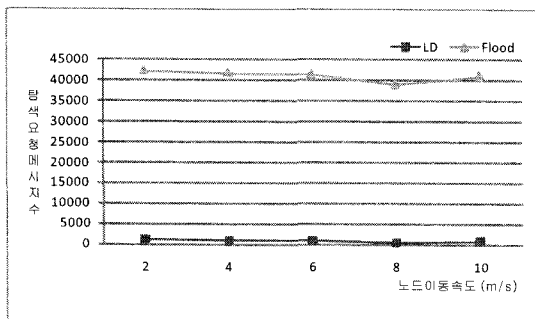


그림 7. 탐색 요청 메시지 수 - 노드 이동 속도 별
Fig. 7. The number of service request messages according to node speed

그림 8은 이동 속도에 따른 평균 응답 홉 수를 나타낸다. 그림에서 보듯이 플러딩 기반의 서비스 디스커버리는 평균 1.5배이상 더 많은 거리를 탐색하게 된다.

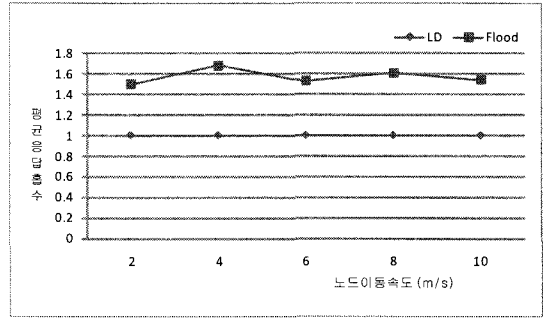


그림 8. 평균 응답 홉 수 -노드 이동 속도 별
Fig. 8. The number of average response hop

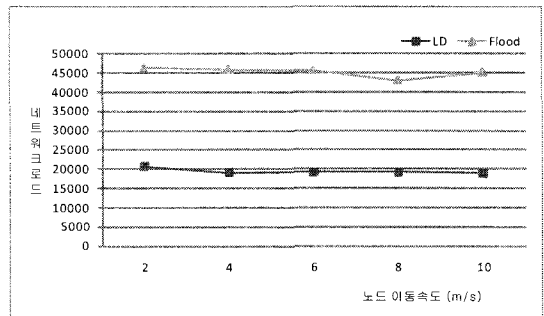


그림 9. 네트워크 로드 - 노드 이동속도별
Fig. 9. Network Load

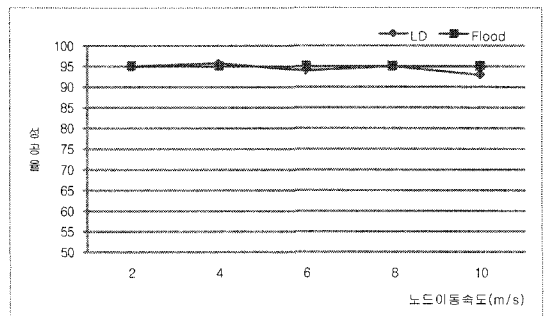


그림 10. 네트워크 로드 - 노드 이동속도별
Fig. 10. Network Load

그림 10은 각 프로토콜별로 노드 이동 속도에 변화를 주어가며 검색 성공률을 모의 실험한 결과이다. 본 논문에서 제안한 방식(LD)이 플러딩 기법과 거의 유사한 성공률을 보임을 알 수 있다. 대부분의 노드 수에서 성공률은 95%이상을 보인다.

V. 결론

모바일 애드-혹 네트워크는 이동성을 가지는 수많은 노드들로 구성되며 중앙 관리 기반 없이 무선 링크로 연결된 이동 노드들의 자율적 시스템으로 구성되는 네트워크이다. 이러한 네트워크 환경에서 이웃한 노드와의 협력을 통하여 서비스 탐색이 제대로 이루어지기 위해서는 모바일 애드-혹 네트워크에 적합한 서비스 검색 기술이 필요하다. 그러나 기존의 서비스 검색 기술들은 중앙 룩업 서버를 사용하여 서비스 정보를 등록한 후 검색하거나 전체 네트워크에 메시지를 풀러딩하는 기법들을 사용하기 때문에 네트워크 로드와 중앙 서버 유지라는 문제를 가지고 있다. 본 논문은 모바일 애드-혹 네트워크에서 노드 ID-기반 서비스 디스커버리와 이웃노드 정보 캐싱을 혼합한 서비스 디스커버리 방식을 제안한다. 제안하는 기법은 노드 ID-기반 서비스 디스커버리의 장점을 그대로 유지하면서 이웃 노드의 정보를 캐시 함으로써 노드가 서비스 정보를 요청 할 때 여행해야할 경로를 줄여 서비스 검색 성능을 향상시킨다. 즉, 서비스 검색 시 서비스 요청 메시지가 중간 노드에서 캐시에 저장된 정보를 이용하여 서비스 제공자에게 도착하지 않고도 중간 노드에서 서비스 정보에 대한 서비스 장을 할 수 있다. 그러므로 서비스 검색을 위해 이용하는 메시지 수를 줄이고 서비스 디스커버리의 탐색 거리를 줄임으로서 전체 네트워크 비용과 응답 시간을 줄일 수 있다. 또한 제안된 기법은 중앙 룩업 서버를 사용하지 않으며 브로드캐스팅을 사용하지 않는다. 성능 평가 결과는 기존의 풀러딩 기반 서비스 디스커버리보다 제안하는 기법의 성능이 우수함을 보인다.

제안하는 기법은 정보의 활용성을 확대하기 위해 데이터를 캐시에 저장하여 저장에 대한 오버헤드가 있다. 그러나 이웃한 노드들의 정보를 극히 제한적으로 저장함으로써 많은 저장 공간을 필요로 하지는 않는다. 향후에는, 실시간 모바일 데이터의 일치성을 연구 할 예정이다.

참고문헌

- [1] C. Toh, "Ad Hoc Mobile Networks: Protocols and Systems," Prentice Hall, 2002.
- [2] E. Guttman, "Service Location Protocol: Automatic Discovery of IP Network Services," IEEE Internet Computing, Vol.3, pp.71-80, Jul-Aug. 1999.
- [3] "Universal Description Discovery and Integration Platform," http://www.uddi.org/pubs/Iru_UDDI_technical_White_Paper.pdf, Sept, 2000
- [4] J. Waldo, "The Jini architecture for network-centric computing," Communications of the ACM, Vol.42, No. 7, pp. 76-82, July 1999.
- [5] F. Sailhan and V. Issarny, "Scalable Service Discovery for MANET," IEEE PerCom, pp.235-244, Hawaii, U.S.A, Mar. 2005.
- [6] 최현덕, 박호현, 우미애, "애드혹 망에서 효율적인 P2P 시스템," 한국통신학회, 제 32권, 제 4호, 200-207쪽, 2007년 4월.
- [7] 정재훈, 이승학, 김남기, 윤현수, "모바일 애드-혹 네트워크에서 분산 해쉬 테이블 기반의 서비스 탐색 기법," 한국정보과학회, 제 35권, 제 1호, 91-97쪽, 2008년 2월
- [8] The Salutation Consortium Inc., "Salutation Architecture Specification Part 1, version 2.1 Edition," <http://www.salutation.org>, 1999
- [9] R. John, "UPnP, Jini and Saluation - A Look at Some Popular Coordination Frameworks for Future Network Devices," technical report, California Software Labs, <http://www.cswl.com/whitepaper/tech/upnp.html>, 1999
- [10] R. Moreno-Vozmediano, "A hybrid mechanism for resource/service discovery in ad-hoc grids," Future Generation Computer Systems, Vol.25, Issue 7, pp. 717-727, July, 2009
- [11] A. Klemm, C. Lindermann, and O. Waldhorst, "A Special purpose peer-to-peer file sharing system for mobile ad hoc networks," IEEEVC 2003, pp. 2758-2763, Orlando, Florida, USA, Oct. 2003.
- [12] A. Datta, "MobiGrid: P2P overlay and MANET rendezvous - A data management perspective," CAiSE 2003 Doctorial-Symposium, Klagenfurt, Austria, June 2003.
- [13] Y. Hu, S. Das, and H. Pucha, "Exploiting the synergy between peer-to-peer and mobile ad hoc networks," HotOS-IX 2003, pp.27-42, Hawaii, U.S.A, May 2003.

- [14] G. Gao, "Proactive Power-Aware Cache Management for Mobile Computing Systems," IEEE Trans. Computer, Vol. 51, No. 6, pp. 608-621, June 2002.
- [15] The network simulator ns-2, <http://www.isi.edu/nanam/ns/>
- [16] 신성윤, 박기홍, 이양원, 이종찬, 이진관, 장혜숙, "이동 Ad-hoc 통신을 위한 지능형 거리추정 클러스터방식," 한국컴퓨터정보학회논문지, 제14권 제11호, 105-111 쪽, 2009년 11월.
- [17] 웬중휴, 오상윤, "Ad-Hoc 무선 환경의 발행/구독 시스템을 위한 구독주제 유사도 기반의 이벤트 라우팅 알고리즘," 한국컴퓨터정보학회논문지 제14권 제10호, 11-22쪽, 2009년 10월.

저자 소개



강 은 영

1987: 숙명여자대학교 학사
 1999: 숙명여자대학교 석사.
 2009: 성균관대학교 박사.
 2009.09 - 현재:
 동양공업전문대학 전기전자통신공학부
 전임강사
 관심분야: 분산컴퓨팅, MANET, 파
 일탐색, 서비스디스커버리,
 데이터마이닝