

모바일 애드-혹 네트워크를 위한 효율적인 서비스 디스커버리

정희원 강은영*

An Efficient Service Discovery for Mobile Ad Hoc Networks

Eunyoung Kang* *Regular Member*

요약

모바일 애드-혹 네트워크에서 현재 상황에서 원하는 시간에 원하는 서비스를 검색, 사용할 수 있는 서비스 디스커버리 기법은 중요 이슈이다. 본 논문에서는 DHT기반 서비스 디스커버리 기법과 P2P 캐싱 기법을 혼합하여 효율적인 서비스 디스커버리 기법을 제안한다. 제안한 기법은 DHT기반으로 CAN과 같이 2차원의 논리적 영역에 해시된 정보를 저장하고 이를 이용하여 서비스를 검색한다. 또한 서비스 검색을 빠르게 하기 위하여 P2P 캐시를 기반으로 이웃 노드의 정보를 사용하여 서비스 검색 성능을 향상시킨다. 제안된 기법은 노드의 부하를 가중 시키며 병목 현상을 일으키는 중앙 서버를 사용하지 않고 많은 메시지를 생성하는 브로드캐스팅 방식을 사용하지 않는다. 그러므로 네트워크에서 노드에게 부하를 균등하게 배분하며 서비스 디스커버리를 이용하여 서비스를 검색하는데 필요한 메시지수를 줄이고 평균 탐색 거리를 줄임으로서 전체 네트워크 로드와 응답 시간을 향상 시킬 수 있다.

Key Words : mobile ad-hoc, networks, service discovery, DHT, Peer-to-Peer

ABSTRACT

In mobile ad hoc networks, service discovery to search for an available service is an important issue. In this paper, we propose an efficient service discovery protocol that is combined a dht-based service discovery scheme and a P2P caching-based information spreading scheme. Proposed scheme store key information in hashed zone and search services using it's information. To search quickly a service, it use its physical neighbors information which collected by 1-hop hello message between a node and its physical neighbors. We do not use a central look up server and do not rely on flooding. Hence, it uniquely balanced all node's load in mobile ad hoc networks and reduced the number of messages exchanged, network load and response time. The simulation results show that our protocol outperforms DHT-based and flooding protocols.

1. 서론

모바일 애드-혹 네트워크(Mobile Ad-Hoc Network: MANET)는 기존에 설치된 유선망이나 기지국의 도움 없이 이동 노드들 사이에 자율적으로 구성되는 네트워크로서, 최근 산업 및 연구 단체에서 MANET

에 대한 관심이 높아지고 있다. 이는 MANET이 더 이상 단일의 독립적인 네트워크로서의 기능을 수행할 뿐 아니라 기존 인프라 기술과 접목하여 유비쿼터스 네트워크 구성이 가능했기 때문이다^[1]. 유비쿼터스 컴퓨팅 환경에서는 다양한 장치들이 상호 연결되어 네트워크상의 서비스를 자동으로 발견하고

* 동양공업전문대학 전기전자통신공학부(eykang1@unitel.co.kr)

논문번호 : KICS2009-02-071, 접수일자 : 2009년 2월 12일, 최종논문접수일자 : 2009년 9월 11일

서비스는 자신의 능력을 광고할 수 있는 서비스 디스커버리(service discovery)가 중요하다.

기존의 서비스 디스커버리 기법의 대표적인 프로토콜로는 SLP, UPnP, Hybrid 기법 등이 있다. 그러나 SLP는 유선 환경에서 중앙 룩업 서버를 유지하는 중앙 디렉터리 기반 서비스 디스커버리 기법으로 중앙 서버에 부하가 집중되고 병목 현상과 같은 문제점이 있어 동적이고 자원이 부족한 모바일 애드-혹 네트워크에는 적합하지 않다. UPnP, SLP2와 같은 프로토콜들은 802.11(a, b, or g)과 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) 브로드캐스팅을 통한 서비스 검색을 위한 Hybrid 기법들이 제안되었으나 여전히 플러딩 기법을 사용하여 서비스 검색에 필요한 많은 메시지가 생성되어 네트워크 비용이 크므로 자원이 부족하고 무선 환경인 모바일 애드-혹 네트워크에서는 효율적이지 않다.

본 논문에서는 모바일 애드-혹 네트워크를 위한 효율적인 서비스 디스커버리 프로토콜에 대하여 제안한다. 제안하는 기법은 분산 해시 테이블(Distributed Hash Table, DHT) 기반 서비스 디스커버리 기법과 P2P(Peer to Peer) 캐시 기법을 융합하여 두 기법의 장점을 활용함으로써 이웃 노드의 여러 정보(서비스 정보, 접속 정보, 노드 ID 정보)를 사용하여 서비스 검색 성능을 향상시킨다. DHT기반 서비스 디스커버리 기법은 중앙 서버의 부하 집중 현상을 방지할 수 있으나 중앙 인덱스 서버가 존재하지 않는 상태이므로 서비스의 검색을 위해 DHT기반 CAN^[2]과 같이 2차원의 논리적 영역에 해시된 정보를 저장하고 이를 이용하여 서비스를 검색한다. 그러나 DHT기반의 검색은 논리적인 이웃을 통한 검색이므로 본 논문에서는 서비스 검색을 빠르게 하기 위하여 물리적 네트워크 토폴로지 정보를 이용한다. 이는 서비스 요청 메시지가 중간 노드에서 캐시에 저장된 정보를 이용하여 서비스 제공자 및 랑데부 노드에게 도착하지 않고도 중간 노드에서 서비스 정보에 대한 응답을 할 수 있다. 또한, 제안된 기법은 분산 해시 테이블을 사용하므로 노드에게 부하를 균등하게 배분하며 플러딩을 사용하지 않는다. 그러므로 서비스 디스커버리를 이용하여 서비스를 검색할 때 발생하는 메시지 수를 줄이고 평균 탐색 거리를 줄임으로서 전체 네트워크 비용과 응

답 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

2.1 기존의 서비스 탐색 기법

서비스 디스커버리 기법은 크게 세 가지로 구분된다. 첫째, 중앙 집중식 디렉터리 기반 서비스 탐색 기법으로, 대표적인 프로토콜로는 Service Discovery Protocol(SLP)^[3], UDDI^[4], Jini^[5] 등이 있다. 이들은 중앙 룩업 서버를 유지하면서 이를 이용해 서비스 광고 및 서비스 탐색을 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 발생하기 때문에 자원이 부족한 모바일 디바이스와 무선 전송 기반인 모바일 애드-혹 네트워크에는 적합하지 않다. 둘째, 분산 디렉터리 기반 서비스 탐색 기법으로, 대표적인 프로토콜로는 확장성 있는 서비스 탐색 기법(Scalable Service Discovery, SSD)^[6], 애드-혹 네트워크에서 효율적인 P2P 시스템^[7] 등이 있다. 이 기법들은 분산된 디렉터를 지원하는 기법이다. 분산된 서비스 디스커버리에서는 물리적인 계층과 분산된 디렉터를 지원하는 대표 노드들로 구성된 가상 계층의 2계층으로 나타난다. 대표 노드는 이웃 노드들의 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 탐색하게 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 방식과 구조적 P2P인 DHT방식을 적용한 방식들이 있다. 플러딩 방식의 대표적인 프로토콜로는 Salutauin^[8], UPnP^[9], Hybrid기법^[10], ORION^[11],

Lower ID 기법^[14] 등이 있다. 구조적 DHT방식을 사용한 기법에는 MobiGrid^[12], 모바일 애드-혹과 P2P의 시너지 기법^[13]의 논문들이 있다. 플러딩 방식은 중앙 룩업 서버를 사용하지 않는 분산 방식으로 동작하여 중앙 집중식 보다는 모바일 애드-혹 네트워크에 적당하다. 그러나 플러딩 기법은 많은 수의 메시지를 생성하여 리소스가 제한적인 모바일 애드-혹 네트워크에서는 비효율적인 방법이다. 구조적 DHT 방식인 MobiGrid^[12]에서는 애드-혹 네트워크와 P2P 네트워크 두개의 네트워크가 스스로 조직적이며 분산된 유사성의 시너지를 이용하여 P2P와 모바일 응용을 통합한 아키텍처를 제안했다. 논리적 대 물리적 라우팅을 지적하여 이들의 통합이 민감한 이슈이다. 모바일 애드-혹과 P2P의 시너지 기법^[13]은 응용계층 P2P프로토콜 Pastry와 MANET라우팅 알고리즘을 대표하는 DSR을 통합함으로써 IP계층 라우팅 프로토콜을 제안하였고, Lower ID 기법^[14]에서는 노드의 ID와 이웃 노드의 캐시에 저장된 정보를 기반으로 하는 프로토콜을 제안하였다.

2.2 MANET에서의 DHT 탐색 기법 활용

DHT기반 서비스 디스커버리는 각각의 서비스 디스크립션(Service Description, SD)을 해시(Hash)에 의해 Key를 만들고 각 노드는 오버레이 라우팅 테이블에 (Key, SD)의 쌍을 저장하고 유지한다. 서비스 광고는 그 영역을 담당하고 있는 해당 노드(랑데부노드, Rendezvous node)에 (Key, SD)의 쌍을 삽입 하여 수행된다. 네트워크에 참여하고 있는 노드들은 주기적으로 라우팅 정보를 교환하고, 오버레이 관리 방법으로써 라우팅 테이블을 갱신하고 유지한다. 서비스가 요청될 때, 오버레이 라우팅 테이블은 참조되고 서비스 요청 메시지는 랑데부노드를 찾음으로서 서비스 요청이 시작된다. 서비스 요청은 라우팅 테이블로부터 구축된 경로를 따라 탐색 과정으로 수행된다.

그림 1과 2는 기본적인 DHT-기반 CAN기법의 서비스 디스커버리 예를 나타낸다. 그림 1은 DHT 기반 CAN기법의 오버레이 네트워크를 나타내며, 그림 2는 실제 물리적 네트워크를 나타낸다. 그림에서 보듯이 논리적 오버레이 네트워크는 물리적 네트워크와는 다르게 구성되며, 논리적 이웃노드 역시 물리적 이웃 노드와는 다르다.

위의 예에서 노드 7은 서비스 “abc.mp3”를 제공하는 서비스 제공자이고, 노드 6은 “abc.mp3”라는 서비스를 요청하는 서비스 요청자라고 가정하자. 노

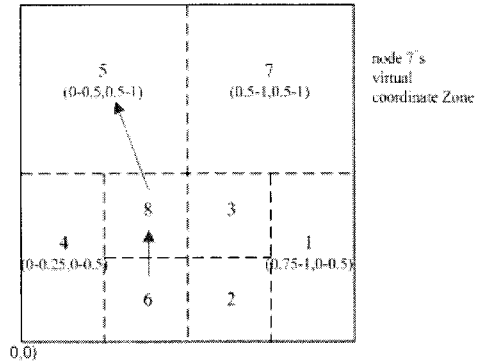


그림 1. CAN 오버레이 네트워크

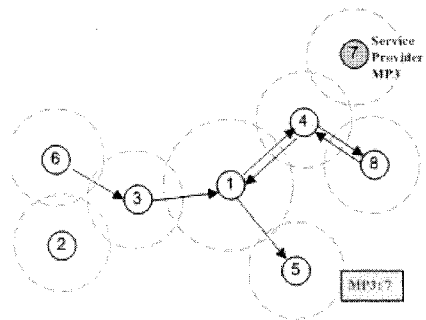


그림 2. 물리적인 네트워크

드 7은 서비스 제공자로서 “abc.mp3”라고 하는 파일 이름을 해시하여 얻은 값이 (0.4, 0.8)의 값을 갖는다면 DHT에서 그 값을 담당하는 랑데부노드 5의 DHT 테이블에도 이 서비스를 등록한다. 서비스 요청 노드 6은 원하는 서비스를 검색하기 위해서 “abc.mp3”를 해시하여 얻은 키 값(0.4, 0.8)으로 DHT테이블에서 랑데부노드 5에 이 서비스 등록 정보가 있다는 것을 안다. 노드 6은 (Key, SD)을 쌍으로 하는 서비스 요청 메시지를 랑데부노드 5로 전송한다.

모바일 애드-혹 네트워크상에서 노드들의 이동성이나 네트워크의 동적인 구성 등의 특징을 고려했을 때, 검색 응용 모델은 P2P와 유사한 특징을 가진다. 따라서 두 네트워크의 공통점을 잘 활용하여 모바일 애드-혹 네트워크에 적합한 P2P 검색 응용 모델에 대한 개발이 이루어져야 한다. 모바일 애드-혹 네트워크는 기존의 고정된 유선망과 달리 노드들이 자유롭게 이동하기 때문에 모바일 애드-혹 네트워크가 갖는 특성들을 고려하여야 한다.

Ⅲ. 제안하는 서비스 디스커버리 프로토콜

이 절에서는 모바일 애드-혹 네트워크에 적절한 서비스 디스커버리 프로토콜에 대하여 제안한다. 제안하는 기법은 DHT 기반 서비스 디스커버리와 P2P 캐시를 융합하여 두 기술의 장점을 활용하여 서비스 검색 성능을 향상시킨다.

3.1 이웃 노드 정보 캐싱을 통한 DHT기반 서비스 디스커버리 기법의 개선

모바일 애드-혹 네트워크에서 각 노드는 이웃 노드의 정보를 통하여 서비스 디스커버리의 검색 효율성을 증대시킬 수 있다^[5]. 본 논문에서는 물리적인 이웃 노드들의 여러 정보를 캐싱하여 이를 서비스 디스커버리에 활용 하는 것이다. AODV의 HELLO 메시지는 무선 전송 범위 내에 있는 이웃들에게 주기적으로 1-홉 브로드캐스트 한다. 각 노드의 이웃노드 정보를 알기 위하여 HELLO 메시지에 노드의 ID, 접속 정보, 서비스 정보를 포함하여 전송한다. 그러므로 따로 이웃노드의 정보를 얻기 위한 메시지를 사용하지 않아 추가적인 오버헤드가 발생하지 않는다. 이 메시지를 받은 노드들은 자신의 캐시에 저장한다. 이 캐시를 이웃 노드 정보 캐시(NIC, Neighbor node Information Cache)라 한다. 그림3은 그림2의 네트워크에 이웃 노드 정보 캐시를 노드 아래에 {} 기호로 표현하였다. 예를 들면 노드 6의 NIC에 저장된 정보는 노드 3인 {3}이고 노드 1의 NIC에 저장된 정보는 노드 3, 노드 4, "abc.mp3" 정보를 가진 노드 5의 정보로 {3,4,5-abc.mp3:7}이다. 이렇게 저장된 정보들은 다른 노드들이 서비스 검색을 할 때 유용하게 사용될 수 있다. 서비스를 필요로 하는 노드가 서비스를 제공하는 서비스 제공자나 랑데부노드에게 까지 도달하지 않고도 캐시에 저장된 정보를 이용하여 응답하게 함으로써, 노드가 여행하는 경로의 수를 줄여 전체 네트워크 로드를 줄일 수 있기 때문이다.

3.2 서비스 검색 기법

제안하는 기법은 DHT-기반 서비스 디스커버리와 이웃 노드 정보 캐시를 융합한 방식이다. 그림 3에서 노드 6은 해당 서비스 "abc.mp3"를 검색하기 위해서 자신의 로컬 정보 캐시와 이웃 노드 정보 캐시를 차례로 서비스를 검색한다. 자신의 로컬 정보들에서 해당하는 서비스가 없다면 "abc.mp3"를 제시하여 얻은 키 값(0.4, 0.8)으로 DHT 테이블에서

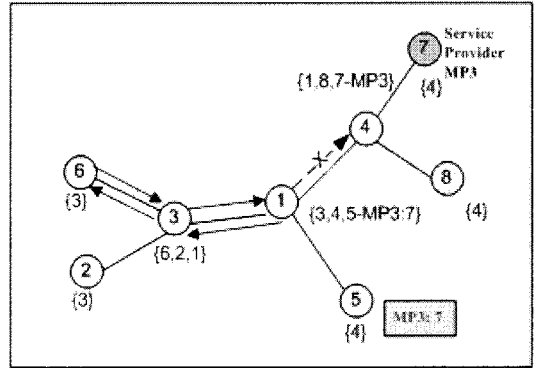


그림 3. 제안하는 서비스 디스커버리 기법

랑데부노드 5에 이 서비스 등록 정보가 있다는 것을 안다. 노드 6은 (Key, SD)을 쌍으로 하는 서비스 요청 메시지를 전송한다. 노드6은 DHT 테이블을 이용하여 노드 5까지의 서비스 요청 경로를 찾는다. 이 때 경로는 n6-n8-n5이다. 그러나 DHT 기반 오버레이 라우팅은 논리적 이웃에게 전달되며 물리적 토폴로지와는 독립적이다. 즉, 물리적 네트워크를 전혀 고려하지 않았다. 그러므로 실제 경로는 n6-n3-n1-n4-n8-n1-n5의 경로를 거친다.

그러나 제안된 우리의 기법에서는 논리적인 이웃만을 사용하지 않는다. 우리는 논리적 오버레이 네트워크에 있는 목적지를 향해 빨리 도달하기 위해 물리적인 이웃 노드들을 통해 수집된 이웃 노드들의 정보를 이용한다. 그림 3에서와 같이 서비스 요청 메시지를 받은 노드 1은 이웃 노드 정보 캐시(NIC)를 이용할 수 있다. 자신의 캐시에 저장된 정보가 노드 3, 4, 5에 대한 정보를 알고 있으므로 랑데부 노드 5까지 도착하지 않고도, 서비스 정보와 IP 주소를 포함한 서비스 응답 메시지를 서비스 요청 노드에게 보낼 수 있다. 그러므로 실제 경로는 n6-n3-n1이 된다. 이는 DHT기반 서비스 디스커버리의 경로인 n6-n3-n1-n4-n8-n1-n5보다 효율적이다. 또한 제안하는 기법은 서비스 검색의 효율성 증대를 위해 다른 추가적인 메시지를 보내지 않고 Hello 메시지를 활용한다는 것이다.

3.3 서비스 광고 및 서비스 요청

모바일 애드-혹 네트워크는 무선망이기 때문에 유선망과 달리 각 노드가 사용할 수 있는 평균 대역폭이 제한적이고, 노드들이 이동하는 동적 네트워크를 가진다. 따라서 전체 네트워크의 성능을 향상시키기 위해서는 전송 메시지 수를 최소화하여야 한다.

본 논문에 제시하는 서비스 디스커버리 기법은 노드가 특정 서비스를 요청하는 경우 해당 노드는 먼저 자신의 로컬 정보를 먼저 검색한다. 이때, 서비스를 자신의 로컬 캐시(LSC)에서 발견하였다면, 이는 자신이 바로 서비스를 제공할 수 있는 서비스 제공자임을 의미한다. 요청한 서비스가 이웃 노드 정보 캐시(NIC)에서 검색되었다면 등록된 서비스 정보를 사용한다. 이는 서비스제공자가 자신에게 인접해 있음을 의미한다. 만약 등록된 서비스가 로컬 캐시 정보에 없다면, DHT-기반 서비스 디스커버리 기법으로 서비스 요청을 한다. 이는 찾고자하는 해시 키 값과 서비스 디스크립션을 포함하는 서비스 요청 메시지를 만들어 DHT에서 해시 키 값을 담당하는 노드들에게 전송한다. 서비스 디스커버리에 관한 알고리즘은 알고리즘 1에 나타내었다.

알고리즘 1 서비스 탐색 기법

```

Service Request(Key, S)
/*
Input: Key=Hashed Key, S: Service Description
LSC: Local Service Cache, NIC: Neighbor node
Information Cache
DHT: Distributed Hash Table
*/
//요청 service가 local 에 있으면 서비스 제공자임
if S ∈ LSC
    reply Service Response Message;
//요청 서비스가 이웃에 있으면 서비스 응답 메시지 보냄
else if S ∈ NIC
    reply Service Response Message
//논리적 이웃 확인, 서비스 담당 구역으로 전송
else If S ∈ DHT
    forward Search(Key, S)
        to logical neighbor node
end if.
    
```

3.4 서비스 응답

서비스 요청 메시지의 중간 과정에서 요청한 서비스가 로컬 캐시, NIC에 발견 되거나, 서비스 제공자에서 발견되어지면, 노드는 발견한 서비스의 정보를 바탕으로 서비스 응답 메시지를 만든다. 서비스 응답 메시지는 서비스 요청 메시지에서 지정한 서비스 디스크립션(SD), 그리고 이 서비스를 제공할

수 있는 서버의 IP 주소 정보를 포함한다. 수신된 서비스 응답 메시지 중 서비스의 파워(여유시간)값이 가장 큰 값을 가지는 서비스제공자를 선택하고, 선택된 서비스제공자에게 서비스를 호출한다.

3.5 캐시 갱신 기법

강한 캐시 일관성을 유지하는 기존의 연구들이 있다¹⁵⁾. 그러나 모바일 애드-혹 네트워크에서 대역폭과 전력 제한 때문에 강한 캐시 일관성을 유지하는 것은 너무 많은 오버헤드가 있다. 그러므로 캐시 일관성을 유지하기 위하여 본 논문에서는 시간 기법을 기반으로 한다. 데이터 엔트리의 서비스 가능 시간 값이 초과되지 않았다면 적절한 값으로 간주한다. 만일 서비스 가능 시간 값이 초과 되었다면 서비스가 부적합하므로 그 데이터 엔트리는 테이블에서 제거된다. 만일 지나가는 메시지가 같은 데이터 엔트리를 가지고 있다면 새로운 서비스 가능 시간 값을 최근 값으로 변경시켜 가장 최근의 정보를 유지한다.

IV. 성능 평가

이번 장에서 본 논문에서 제안하는 서비스 프로토콜의 성능을 평가한다. 본 논문에서 제안하는 이웃 노드와의 상호 협동을 통한 DHT기반 프로토콜(Cooperative DHT, CoDHT)을 기존의 플러딩 방식(Flood), DHT 기반의 방식(DHT)과 비교한다.

4.1 실험 환경

이동 애드-혹 네트워크에서 서비스 디스커버리의 성능평가를 위해 대표적인 네트워크 시뮬레이션 도구인 NS2 v2.32¹⁶⁾을 사용하였다. 실험 환경은 1500m x 1500m의 네트워크 영역에 이동하는 무선 노드들이 랜덤하게 분포된 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 이동 속도를 3(m/s)로 이동하며, 이동 중 일정 정지 시간(2초)만큼 정지한 후 임의로 선택된 목적지점까지 이동 유니폼 속도의 속도로 이동한다. 이동 노드들의 이동 시나리오는 Random way-point 방식이 사용된다. 라우팅 프로토콜로 AODV를 사용하였다. 노드의 수는 20개 ~ 100개로 변화를 주어가며 실험하였다. 표 1은 알고리즘을 위한 시뮬레이션에 사용된 인자의 값들을 보인다.

표 1. 시뮬레이션 매개변수

Item	Value
노드 수	20 ~ 100
네트워크 크기 (x, y)	1500m X 1500m
서비스 요청 간격	5 (초)
라우팅 프로토콜	AODV
평균 이동 속도	3 (m/s)
평균 최대 이동 속도	5 (m/s)
이동 패턴	Random way-point
Networkinterface	Phy/WirelessPhy
MAC	Mac/802_11
실행시간	250 (초)
무선 전송 범위	250 (m)
서버율	30 (%)

4.2 성능 평가 기준

모바일 애드-혹 네트워크의 가장 큰 특성은 네트워크를 구성하는 이동 노드들이다. 이동 노드들의 자원(배터리, 프로세서, 메모리, 지원 장치, 전송 대역폭) 제한적이다. 모바일 애드-혹 네트워크에서 이동 노드는 이동 컴퓨팅 기능을 가진 서비스 제공자인 서버이자 서비스 요청 노드인 클라이언트이며 라우터로서도 동작한다. 그러므로 다른 노드를 대신 하여 패킷을 전달하기도 한다. 또한 제한된 배터리로 동작함으로써 에너지 잔량에 따라 제약을 받는다. 이동 노드는 메시지를 전송하고 수신할 때 많은 에너지를 소모한다. 애드-혹 망에서의 효율적인 P2P 시스템^[7]의 연구에서 보면 패킷을 전송할 때 소비되는 에너지를 1이라 하면 패킷을 받을 때는 0.59 배의 에너지를 소모하며 휴지 상태에서도 에너지를 소비한다. 그러므로 검색 과정에 의해 생성되는 전송 메시지와 수신 메시지, 즉 네트워크 로드를 줄이는 것이 중요하다. 또한 모바일 애드-혹 네트워크는 불안정한 링크 특성을 갖는다. 이동 노드들은 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 응답이 요구되며, 이는 서비스 요청 노드로부터 응답 노드까지의

평균 응답 거리가 짧아야 한다. 본 논문에서는 성능 평가 기준으로 검색 성공률, 평균 검색 거리, 네트워크 로드 3가지가 사용된다. 검색 성공률: 검색 성공률은 서비스 검색 요청을 위해 생성된 서비스 요청에 대해 서비스 검색을 성공한 질의의 비율이다. 평균 검색 거리는 검색에 소요된 평균 홉 수로 구한다. 네트워크 로드는 서비스 광고 메시지 수와 서비스 검색 메시지 수를 합하여 구한다.

4.3 성능 평가 결과

본 절에서는 제안한 방식의 실험 결과를 제시하고 분석한다. 노드의 수는 20개, 40개, 60개, 80개, 100개로 변화를 주어가며 실험하였고 성능 평가 결과 값은 10번의 성능 평가 결과를 평균하였다.

그림 4는 각 프로토콜별로 노드 수에 변화를 주어가며 검색 성공률을 모의 실험한 결과이다. 본 논문에서 제안한 방식(CoDHT)이 DHT 기법보다 서비스 검색 성공률이 높고, 플러딩 기법보다는 성공률이 낮음을 보인다. 그러나 대부분의 노드 수에서 성공률은 90%이상을 보인다. 만일 서비스가 광고될 때 서비스에 대한 정보가 네트워크상에 효율적으로 이웃 노드 캐시 테이블에 저장되면, 검색 성공률이

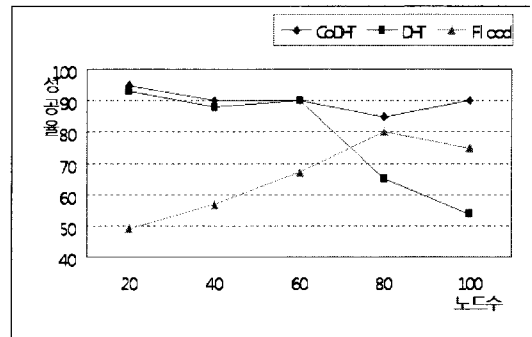


그림 4. 성공률

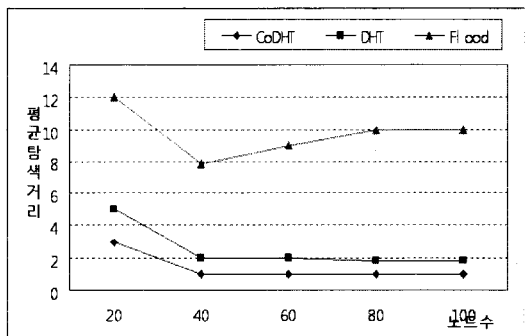


그림 5. 평균 탐색 거리

높아질 것이다.

그림 5는 노드 수에 따른 평균 탐색 거리를 모의 실험한 결과이다. 대부분의 경우 이웃한 노드의 로컬 캐시에 있는 정보를 이용하여 서비스를 찾게 되므로 0.5 ~ 2홉 사이에서 원하는 서비스를 찾는다. 0홉에서 찾는다든 것은 본인이 가지고 있는 로컬에서 원하는 정보를 찾는다든 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 DHT기법과 플러딩 기법보다 평균 탐색 거리가 가까워 가장 빠른 검색 결과를 보여 좋은 성능을 보인다. 이는 서비스 요청 노드가 서비스 요청 메시지를 보낸 후 서비스 응답 메시지를 받을 때까지의 거리로 거리가 짧을수록 검색에 빨리 성공했으며, 서비스 검색에 이용 되는 메시지 수도 작아 질 수 있다.

그림 6은 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 중복 되게 받은 메시지는 다시 보내지 않고 폐기 하며, 새로 받은 메시지만 탐색 요청을 위해 전달한다. DHT기반 서비스 디스커버리 기법은 원하는 서비스가 저장되어 있는 랭데부노드를 찾아 서비스 검색을 시작한다. 그러므로 서비스 검색 메시지는 플러딩 하지 않고 원하는 노드를 찾아 서비스 검색을 하므로 유니캐스트 방식을 사용한다. 그러나 이 유니캐스트 방식의 검색은 물리적인 네트워크 토폴로지를 고려하지 않은 논리적인 검색이다. 그림 6에서 나타내듯이 노드수가 증가 할수록 DHT 기법과 플러딩 기법에서는 네트워크 로드가 기하급수적으로 늘어나나, 제안하는 기법은 노드수가 늘어나더라도 네트워크 로드

가 증가하는 율은 적어 좋은 성능을 나타냈다.

이는 서비스 제공자가 서비스 광고 메시지를 보낸 후, 이를 받은 d-hop 이웃 노드들은 이를 캐시에 저장하여 서비스 검색 요청을 받았을 때, 유용하게 사용될 수 있어 네트워크 로드를 줄일 수 있다.

V. 결론

모바일 애드-혹 네트워크 환경은 이동성을 가지는 수많은 노드들로 구성되며 중앙 관리 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 DHT 기반 서비스 검색 기술들은 논리적인 이웃 노드를 통하여 메시지를 전달함으로써 실질적인 전달 경로인 물리적 네트워크 토폴로지를 고려하지 않아 먼 거리를 탐색할 수 있다. 본 논문에서는 모바일 애드-혹 네트워크에서 이웃 노드 정보 캐싱을 통하여 물리적 네트워크 토폴로지를 고려한 효율적인 DHT-기반 서비스 디스커버리 방식을 제안한다. 제안하는 기법은 DHT 기반 서비스 디스커버리와 P2P 캐시를 융합하여 두 기술의 장점을 활용하여 서비스 검색 성능을 향상시킨다. 이는 서비스 요청 메시지가 중간 노드에서 캐시에 저장된 정보를 이용하여 서비스 제공자 및 랭데부 노드에게 도착하지 않고도 중간 노드에서 서비스 정보에 대한 응답을 할 수 있다. 또한, 제안된 기법은 분산 해쉬 테이블을 사용하므로 노드에게 부하를 균등하게 배분하며 브로드캐스팅을 사용하지 않는다. 그러므로 서비스 디스커버리를 서비스 검색을 위해 이용하는 메시지 수를 줄이고 평균 탐색 거리를 줄임으로서 전체 네트워크 비용과 응답 시간을 항상 시킬 수 있다. 성능 평가 결과는 기존의 플러딩 기법과 DHT기법보다 제안하는 기법이 성능이 우수함을 보인다.

제안하는 기법은 정보의 활용성을 확대하기 위해 데이터를 캐시에 저장한다. 이는 이동 노드의 저장 공간을 사용함으로써, 저장에 대한 오버헤드가 있다. 그러나 이웃한 노드들의 정보를 극히 제한적으로 저장함으로써 많은 저장 공간을 필요로 하지는 않는다. 향후, 실시간 모바일 데이터의 일치성을 연구할 예정이다.

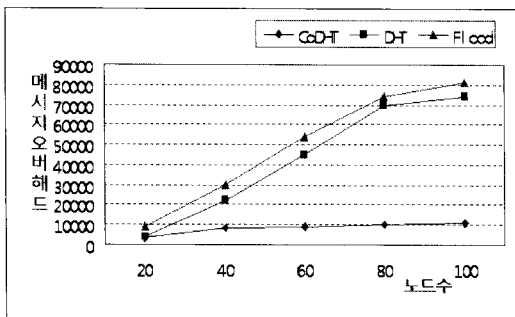


그림 6. 네트워크 로드

참 고 문 헌

[1] C.K. Toh, Ad Hoc Mobile Networks: Protocols and Systems, *Prentice Hall*, 2002

[2] S. Rtnasarny, P. Francis, M. Hnadkey, P. Karp, S. Schenker, "A saclable content addressable network", Proceedings of the 2001 conference on Applications technologies architectures and protocols for computer communications, NY, ACM Press(2001), pp. 161-172

[3] E. Guttman, "Service Location Protocol: Automatic Discovery of IP Network Services", *IEEE Internet Computing*, Vol. 3, pp. 71-80, 1999

[4] "Universal Description Discovery and Integration Platform", http://www.uddi.org/pubs/Iru_UDDI_technical_White_Paper.pdf, Sept, 2000

[5] K. Arnold, et al., "The Jini Architectural Overview: Technical White Paper", *Communications of the ACM*, 40, pp. 71-79, May 1997.

[6] F. Sailhan and V. Issarny, "Scalable Service Discovery for MANET", Proc. of IEEE PerCom 2005, pp. 235-244, 2005

[7] 최현덕, 박호현, 우미애, "애드-혹 망에서의 효율적인 P2P 시스템", *한국통신학회*, vol. 32, no. 4, pp. 200-207, April, 2007

[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 Coordiantion 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*, online 4 march 2008

[11] A. Klemm, C. Lindermann, and O.P. Waldhorst, "A Special purpose peer-to-peer file sharing system for mobile ad hoc networks", Proc. of IEEE VTC2003, pp.2758-2763, 2003

[12] A. Datta, "MobiGrid: P2P overlay and MANET rendezvous - A data management perspective", Proc. of CAiSE 2003, 2003

[13] Y.C. Hu, S.M. Das, and H. Pucha, "Exploiting

the synergy between peer-to-peer and mobile ad hoc networks", Proc. of *HotOS-IX 2003*, pp. 37-42, May 2003

[14] E. Kang, Y. Im, and U. Kim, "Adaptive Neighbor Selection for Service Discovery in Mobile Ad Hoc Networks", NPC 2008, LNCS 5245, pp. 13-23, 2008

[15] G. Gao, "Proactive Power-Aware Cache Management for Mobile Computing Systems", *IEEE Trans. Computer*, vol. 51, no. 6, pp. 608-621, June 2002

[16] The network simulator ns-2, <http://www.isi.edu/nanam/ns/>

강 은 영 (Eunyoung Kang)

정회원



1987년 2월 숙명여자대학교 전산학과 학사

1999년 8월 숙명여자대학교 전산학과 석사

2009년 2월 성균관대학교 전기전자및컴퓨터 공학박사

2009년 9월~현재 동양공업전문대학 전임강사

<관심분야> 분산컴퓨팅, 모바일 애드-혹 네트워크, 파일탐색, 서비스 디스커버리