

JXTA를 이용한 Peer-to-Peer 환경에서 Peer의 성향을 고려한 Peer 탐색 알고리즘의 연구

A Study on Effective Peer Search Algorithm Considering Peer's Attribute using JXTA in Peer-to-Peer Network

이종서*, 문일영**

Jong-Seo Lee*, Il-Young Moon**

요 약

최근 클라우드 컴퓨팅 시대에 접어들면서 분산 컴퓨팅 환경에서의 분산된 자원의 효율적인 검색이 중요하다. P2P(Peer-to-Peer) 네트워크에서 분산된 정보를 검색하기 위해서는 수많은 Peer를 순회하며 정보를 검색하기 때문에 네트워크의 규모 및 네트워크 상태에 따른 시스템의 부하와 많은 검색 시간을 소요하게 된다. JXTA(Juxtapose)는 P2P를 실현하기 위하여 셀 폰, PDA, 서버에 연결되어 있는 PC 등 어떤 다양한 디바이스도 커뮤니케이션 및 상호 협력을 위해 네트워크에 연결될 수 있도록 해 주는 P2P 프로토콜의 오픈 셋이라 할 수 있다. 본 논문에서는 P2P 오버레이 네트워크에서 JXTA를 이용한 효과적인 Peer 탐색 알고리즘을 제안한다. 기존 JXTA의 Peer 탐색 알고리즘은 DHT(Distributed Hashing Tables) 기반의 A Loosely-Consistent DHT 방식을 활용한다. 제안한 Peer 탐색 알고리즘은 WALK_LOOKUP 메시지의 수를 줄임으로써 시스템의 부하를 줄였고, 기존 JXTA-Sim의 Peer 탐색 성공률과 동등한 성능을 보임으로써, 이를 적용하여 향후 다른 네트워크상에서도 메시지의 수를 줄일 수 있는 최적화 모델을 찾아야 할 것이다.

Abstract

Searching distributed resource efficiently is very important in distributed computing, cloud computing environment. Distributed resource searching may have system overheads and take much time in proportion to the searching number, because distributed resource searching has to circuit many peers for searching information. The open-source community project JXTA defines an open set of standard protocols for ad hoc, pervasive, peer-to-peer computing as a common platform for developing a wide variety of decentralized network applications.

In this paper, we proposed peer search algorithm based on JXTA-Sim. original JXTA peer searching algorithm selected a loosely-consistent DHT. Our Lookup algorithm decreases message number of WALK_LOOKUP and reduce the network system overload. and we make a result of same performance both original algorithm and our proposed algorithm.

Key words : Peer-to-Peer, JXTA-Sim, Peer Search Algorithm

I. 서 론

소비자의 다양한 요구가 늘어나면서 사용자 중심

* 한국기술교육대학교 대학원 컴퓨터공학과(Department of Computer Eng., Korea University of Technology and Education)

** 한국기술교육대학교 컴퓨터공학부(School of Computer Eng., Korea University of Technology and Education)

· 제1저자 (First Author) : 이종서

· 투고일자 : 2011년 6월 23일

· 심사(수정)일자 : 2011년 6월 24일 (수정일자 : 2011년 8월 19일)

· 게재일자 : 2011년 8월 30일

의 유비쿼터스 컴퓨팅 환경으로 모습이 변화하였다. 개인 휴대용 장치를 통해 언제 어디서나 네트워크에 접속하여 자신이 원하는 서비스를 받고 있으며 또한, 사용자 중심의 고품질 맞춤형 서비스가 등장하고 있다. 또한 사용자마다 다수의 네트워크에 접속 가능한 단말을 가지고 있기 때문에 네트워크 접속에 따른 과도한 트래픽을 발생 시키기도 하는 문제점이 발생한다[1].

또한 분산 컴퓨팅, 클라우드 컴퓨팅과 같은 분산 컴퓨팅 환경을 일반 사용자들도 이제는 손쉽게 사용할 수 있게 되었다. 이러한 분산 네트워크 환경에서 사용자가 원하는 정보를 정확하고 쉽게 사용하기 위해서는 이에 걸 맞는 네트워크 환경이 필요하게 되었다. 1995년 5월 파일 공유를 위한 Napster 소개로 시작된 P2P(Peer-to-Peer) 네트워킹 기술은 분산 컴퓨팅, 인터넷전화(Skype)에 성공적으로 적용됨에 따라, 현재 가장 관심 있는 인터넷상 새로운 통신방식으로 떠오르고 있다. P2P 응용에 대한 사용 증가세는 WWW(World Wide Web) 보다도 빠른 성장을 보이고 있으며 모든 형태의 분산 자원 접근에 사용될 수 있으며 인터넷 기반 응용에 새로운 가능성을 제시하고 있다. 이러한 P2P 이용량을 살펴보면 인터넷 트래픽의 50% 이상(때로는 75% 이상)이 P2P 응용에 의하여 발생 된다고 한다.

네트워크상 존재하는 분산자원은 분산되어 네트워크상에 있으며 Peer 집합 내 각 Peer는 상대 Peer가 제공하는 자원을 이용하며, 대상 자원은 여러 가지 형태로 존재한다. 이동성을 기반으로 하는 유비쿼터스 환경에서 각 Peer는 해당주소가 동적으로 변화하며 상시 네트워크에 연결된다는 보장도 없다. 따라서 주소를 기반으로 하는 데이터 어드레싱은 더 이상 유효하지 않다. 이에 대한 대안으로 P2P 시스템은 콘텐츠 기반 라우팅 기능을 제공한다.

본 논문에서는 P2P 오버레이 네트워크 환경에서 사용자 성향 중심의 Peer 탐색 알고리즘을 연구한다. JXTA을 기반으로 한 JXTA-Sim의 Peer 탐색 알고리즘을 바탕으로 논문에서 제한하는 사용자 성향에 맞는 네트워크 토폴로지 구성과 P2P 네트워크상에서 사용자가 원하는 정보를 찾기 위한 신뢰성 있는 Peer 탐색 알고리즘 연구가 목적이다.

II. 관련 연구

2-1 Peer-to-Peer 오버레이 네트워크

오버레이 네트워크는 기존 물리적 네트워크를 바탕으로 그 위에 구성되는 또 다른 네트워크, 즉 기존의 네트워크 위에 별도의 Peer들과 논리적 링크들을 구성하여 이루어진 가상 네트워크이다. P2P 오버레이 네트워크 기술은 P2P기술에 오버레이 네트워크 개념을 접목시킨 기술로 기존의 Client / Server 개념과 달리 PC 들이 연결되어 자원을 공유하고 모든 참여자가 Server인 동시에 Client의 역할을 수행하는 특징을 갖는다.

다음 그림 1은 물리적 네트워크상에 존재하는 Peer들이 P2P 오버레이 네트워크상에 등록이 되어 각각의 Peer들 간의 가상 네트워크가 형성 되고, Sever의 도움 없이 다른 Peer들과 직접 정보를 공유하고 교환할 수 있다

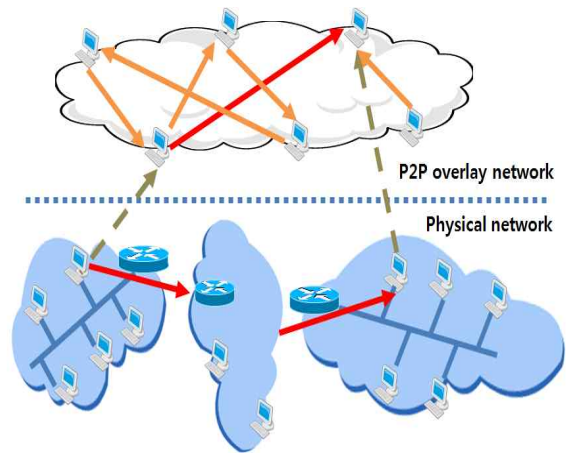


그림 1. P2P 오버레이 네트워크
Fig. 1. P2P Overlay network.

2-2 비구조적 P2P

본 방식은 Peer와 자원 간 연관성 설정이 없는 구조로써, 중앙 서버 또는 인접 Peer로 검색 쿼리 플러딩을 이용하여 자원 및 Peer를 탐색한다.

P2P 네트워킹 초기 모델은 자원을 보유한 Peer의 IP 주소를 중앙 서버에 의해서 관리하는 집중화된 형태로 시작되었다. 본 모델은 Napster에 의해서 널리

알려졌으나 공유 파일의 저작권에 따른 법적인 문제, 중앙 서버에 의한 확장성 제한 등으로 사용이 제한되었다. 이에 대한 대안으로 원하는 자원이나 Peer를 찾을 때 까지 중앙 장치 없이 인접 Peer로 검색 쿼리를 플러딩 시키는 방식인 순수 P2P(Pure P2P)모델이 널리 사용되었다. 그러나 본 모델은 플러딩에 따른 신호 트래픽이 과다 발생하는 문제점이 제기되었다. 이를 보완하기 위해 제안된 하이브리드 P2P(Hybrid P2P)모델에서는 시스템을 여러 개의 Super Peer를 이용한 계층적 구조를 구성하고 자원 검색 쿼리를 Super Peer간에 효율적으로 플러딩함으로써 신호 트래픽을 감소시킨다.

2-3 구조적 P2P

본 방식에서는 분산 인덱싱을 제공하는 DHT(Distributed Hashing Tables)로 콘텐츠와 Peer 정보들을 공통의 단일 주소 공간으로 매핑하여 콘텐츠 저장 및 검색이 이루어지는 분산 구조의 콘텐츠-어드레싱 기반 데이터 저장 기법을 제시하고 있다. 본 기법에서는 최대 검색 횟수를 $O(\log N)$ 으로 데이터 검색이 가능하기 때문에, 검색 효율에 영향 없이 Peer 개수를 임의로 증가 시킬 수 있다. 또한, 각각의 Peer에서 검색을 위해 필요한 정보의 양이 기존의 중앙 서버 기반의 P2P 보다 적으며, 검색을 위한 통신 오버헤드가 플러딩 기반의 P2P 방식보다 적다. 따라서 비구조적 P2P에 비하여 상대적으로 대규모 네트워크에 적용이 가능하다. 그러나 비구조적 P2P에서 다양한 데이터의 속성 값을 이용하여 복잡한 쿼리가 가능했던 반면에 DHT를 사용함으로써 인하여 특정 키 값만을 사용한 검색을 함으로써 쿼리가 단순화되는 단점이 있다.

DHT에서 어떤 항목을 찾아갈 때 해시 테이블을 이용하는데, 중앙 시스템이 아닌 각 Peer들이 이름을 값으로 맵핑하는 기능을 하는 방식이다. 부하가 집중되지 않고 분산된다는 큰 장점이 있어, 극단적으로 큰 규모의 Peer들도 관리할 수 있다.

2-4 JXTA 플랫폼

JXTA는 API(Application Program Interface)대신 프로토콜에 기반을 두고 있으므로, 어떠한 언어나 운영체제 하드웨어 및 전송 프로토콜과도 잘 어울려 동작한다[2]. 네트워크에 접속이 가능한 장치라면 사실상 어떠한 것이라도 JXTA Peer가 될 수 있다.

JXTA에서는 기반이 되는 네트워크가 반드시 TCP/IP일 필요가 없기 때문에 TCP/IP, HTTP, Bluetooth, Home-PNA 등을 거쳐서도 전달될 수 있다. 다른 네트워크상에 존재하는 동등 개체도 JXTA 프로토콜을 이용하면 쉽게 연결될 수 있다.

JXTA는 다른 네트워크 프로토콜의 최상단에 존재하는 네트워크 프로토콜이다. JXTA의 자바 바인딩에는 TCP와 HTTP가 있는데, Bluetooth 같은 네트워크 표준같이 놓여 질 수 있다고 보면 된다. 처음에 TCP 소켓을 열고 그 소켓을 통해서 JXTA 프로토콜 메시지를 전송한다. 즉, 프로토콜의 상세한 내용은 숨기고 쉬운 JXTA 언어 바인딩을 이용해서 보낸다는 뜻이다.



그림 2. JXTA 소프트웨어 구조
Fig. 2. JXTA Software Structure.

그림 2는 JXTA 구조를 나타낸 것이다. JXTA는 JXTA Core, JXTA Services, JXTA Application 등 크게 세 개의 계층구조를 갖는다. JXTA Core 계층은 Peer, Peer group, 발견, 통신, 모니터링, 보안 등과 같이 P2P 네트워크를 위해 일반적으로 필요한 기능들을 캡슐화 하여 제공해 주는 JXTA Platform이라 할 수 있다. JXTA Services 계층은 P2P 네트워크를 운영하기 위해 절대적으로 필요한 기능들을 제외한, P2P 환경에서 일반적으로 필요하거나 요구되는 네트워크 서비스를 제공해 주기 위한 계층이다. 예를 들면, 검색 및 인덱싱, 디렉토리, 저장 시스템, 파일 공유, 분산 파일 시스템, 자원의 집합, 그리고 렌팅(renting), 어그리게

이션(aggregation), PKI 서비스 등이 있다. JXTA Applications 계층은 P2P 인스턴스 메시지, 엔터테인먼트 콘텐츠 등의 관리와 배달, P2P 이메일 시스템, 분산 경매 시스템 등과 같은 JXTA 기반의 애플리케이션을 포함할 수 있다.

2-5 JXTA-Sim 구성요소

JXTA-Sim은 JXTA Peer 탐색 알고리즘의 시뮬레이션에 위한 시뮬레이터이다[3]. 또한, PlanetSim을 기반으로 PlanetSim의 구조를 따라서 설계 되었다[4]. JXTA-Sim을 구성하기 위하여 PlanetSim의 component를 수정하여 확장하였다. 다음 그림 3은 JXTA-Sim의 구조이다[5].

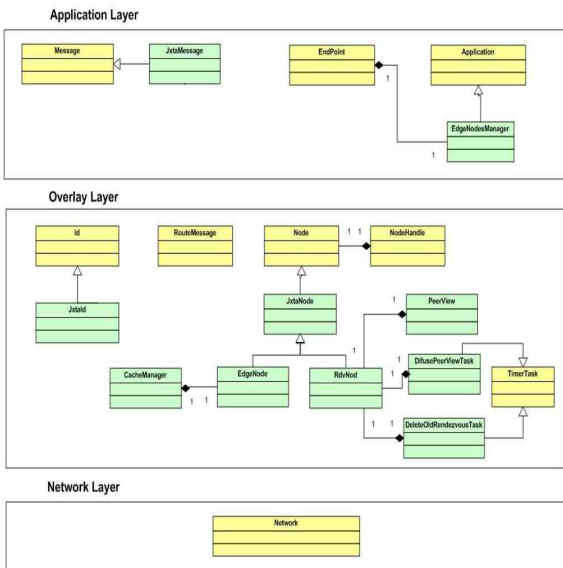


그림 3. JXTA-Sim 구조
Fig. 3. JXTA-Sim Structure

Application 계층에서는 JxtaMessage, EdgeNodesManager를 확장 하였고, Overlay 계층에서는 Jxtald, JxtaNode, EdgeNode, RdvNode, CacheManager, PeerView, DifusePeerViewTask, DeleteOldRendezvousTask를 확장하여 JXTA-Sim에 추가하였다.

노드 컴포넌트는 2가지 노드 타입을 지원하는데 Edge Nodes와 RDV Nodes이다. 두 노드 중 Edge는 published, discovered 한 메시지를 Cache Manager에

저장한다. RDV는 다른 RDV와의 연결을 지속하기 위하여 RPV를 필요로 하는데 PeerView를 통한 테이블 생성하고 새롭게 갱신한다. DifusePeerView와 DeleteOldRendezvous는 RDV의 RPV를 관리하며 오래된 데이터를 삭제한다.

III. Peer 탐색 알고리즘

3-1 JXTA-Sim Peer 탐색 알고리즘

JXTA-Sim Peer 탐색 알고리즘으로 A loosely-consistent DHT 방식을 사용하고 있다[6]. Edge는 Adv(Advertisement)를 publishing, discovering 한다. 이런 Adv는 Cache Manager를 통하여 local cache에 저장한다. RDV는 Adv를 저장하지 않지만 대신에 각각의 Adv의 index를 저장한다.

RDV는 RDV간의 연결을 유지하기 위하여 RPV 리스트를 사용한다. 각각의 RDV들은 다른 Rendezvous들의 존재에 대해 알지 못할 수도 있다. 이러한 이유로 JXTA-Sim에서는 이를 A loosely-consistent DHT라고 부른다.

```
function getReplicaPeer(expression, rpv)
    var Longint digest, sizeOfSpace, sizeOfHashSpace
    var Integer pos
    var JxtaID pid

    digest := jxtaHash(expression)
    sizeOfSpace := length(rpv)
    sizeOfHashSpace := shiftLeft(1, 8 * length(digest))
    pos := (digest * sizeOfSpace) / sizeOfHashSpace

    pid := rpv[pos]

    return pid
}

{
    where:
        jxtaHash(expression) returns the digest of the given expression
        shiftLeft(n, m) := floor(n * 2m)
}
```

그림 4. JXTA getReplicaPeer 함수
Fig. 4. JXTA getReplicaPeer Function

Edge는 Adv를 publishing 할 때 Edge가 연결된

RDV를 이용한다. JXTA가 제공하는 indexing service를 사용하고 Adv를 indexing 한다. 이러한 서비스를 SRDI(Shared Resource Distributed Index)라고 부른다. Rendezvous는 publish request 요청을 받고 Adv를 저장하기 위하여 다음 그림 4의 getReplicaPeer Function을 이용하게 된다. Expression은 Adv의 키로서 고유 Id 값 및 Adv의 이름을 나타내고 RPV를 호출하며, hash function을 이용하여 sizeOfHashSpace 값을 얻게 된다. Replication은 replication distance(얼마나 많은 Peer가 index를 저장하는지 나타내는 값)값으로 만약 1이면 index는 RPV 리스트 안에 해당 RDV가 있는 값에 위, 아래로 있는 RDV에게만 저장하게 된다. 여기서 위, 아래 존재하는 RDV를 chosen rendezvous라고 한다.

JXTA-Sim Peer 탐색 알고리즘에서는 Fall-back 현상으로 인한 네트워크 트래픽이 증가하게 되고 검색 시간이 증가하게 되는 문제점이 발생한다. Fall-back 현상을 해결하기 위하여 Max Walk Num Hops 지정하여 일정 수 이상으로 Fall-back을 방지하였다. 그러나 WALK_LOOKUP 메시지를 보내면 언젠가는 원하는 정보를 찾을 수 있기 때문에 무조건적인 Fall-back 현상을 막는 것은 네트워크의 Peer의 수가 무수히 많은 경우에 탐색 성공률이 떨어지는 현상을 보인다.

3-2 Peer 성향을 고려한 Peer 탐색 알고리즘

본 논문에서는 2가지 방법을 제안한다. 첫째 네트워크를 형성할 때 각각의 Peer의 속성을 추가하여 Edge가 Join할 때 같은 속성을 가진 RDV에 Join을 하고, Peer를 탐색하기 위한 Adv를 탐색할 때 속성을 기준으로 탐색을 한다. 여기서 속성이란 예를 들어 파일 공유 P2P 서비스 상에서 Peer가 원하는 파일의 속성(음원, 영화, 어플리케이션 등)을 말한다.

둘째 Peer 탐색 시에 Fall-back 현상으로 인한 메시지의 수를 줄이기 위하여 Discovery Message에 방문한 모든 Peer의 ID를 기록하고, RDV에 해당 메시지가 도착하였을 때 방문한 Peer의 정보를 먼저 확인하여 이전에 방문한 Peer이면 더 이상의 WALK_LOOKUP 메시지를 종료하여 해당 Peer에게 탐색 실패 메시지를 전송한다.

그림 5의 네트워크 토폴로지 Sequence Diagram을 보면 네트워크에 Peer가 Join할 때 각각의 Peer들의

성향이 결정된다.

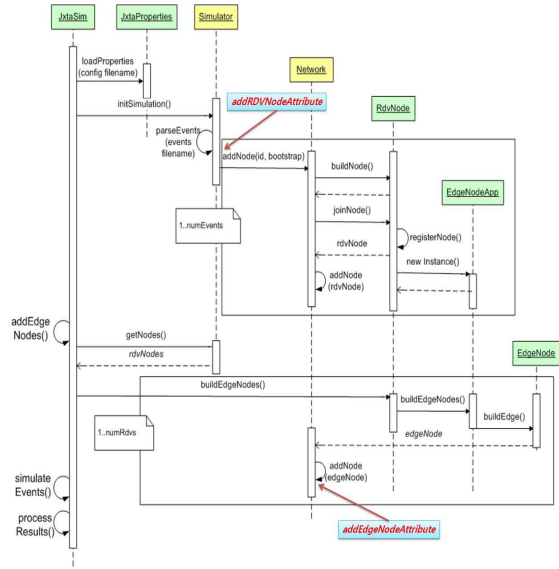


그림 5. 네트워크 토폴로지 Sequence Diagram
Fig. 5. Network Topology of Sequence Diagram.

JXTA-Sim의 PeerView 클래스에 RDV의 Id와 성향 정보를 추가한다. 또한 Discovery Message에 방문한 RDV의 정보를 추가하였다.

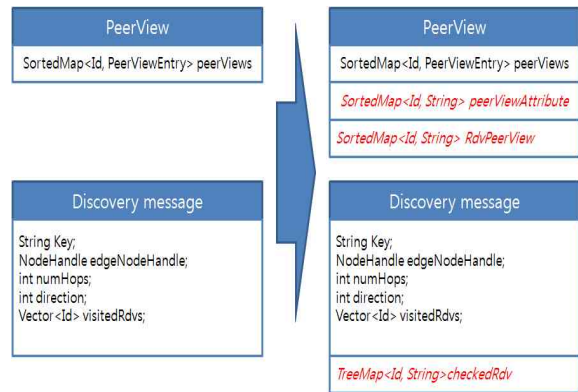


그림 6. 추가한 정보
Fig. 6. Additional information.

그림 7은 본 논문에서 제안한 Peer 탐색 알고리즘 Sequence Diagram이다. WALK_LOOKUP 메시지의 수를 줄이기 위하여 기존의 Peer 탐색 과정에서 findAdvertisement() 안에서 Peer의 속성을 추가한 RPV를 검사하여 Adv의 속성과 같으면 RdvPeerView 맵에 저장하였다. 이 맵은 getSucc(), getPred()에서 RdvPeerView 맵의 정보를 검색하여 Adv의 속성과

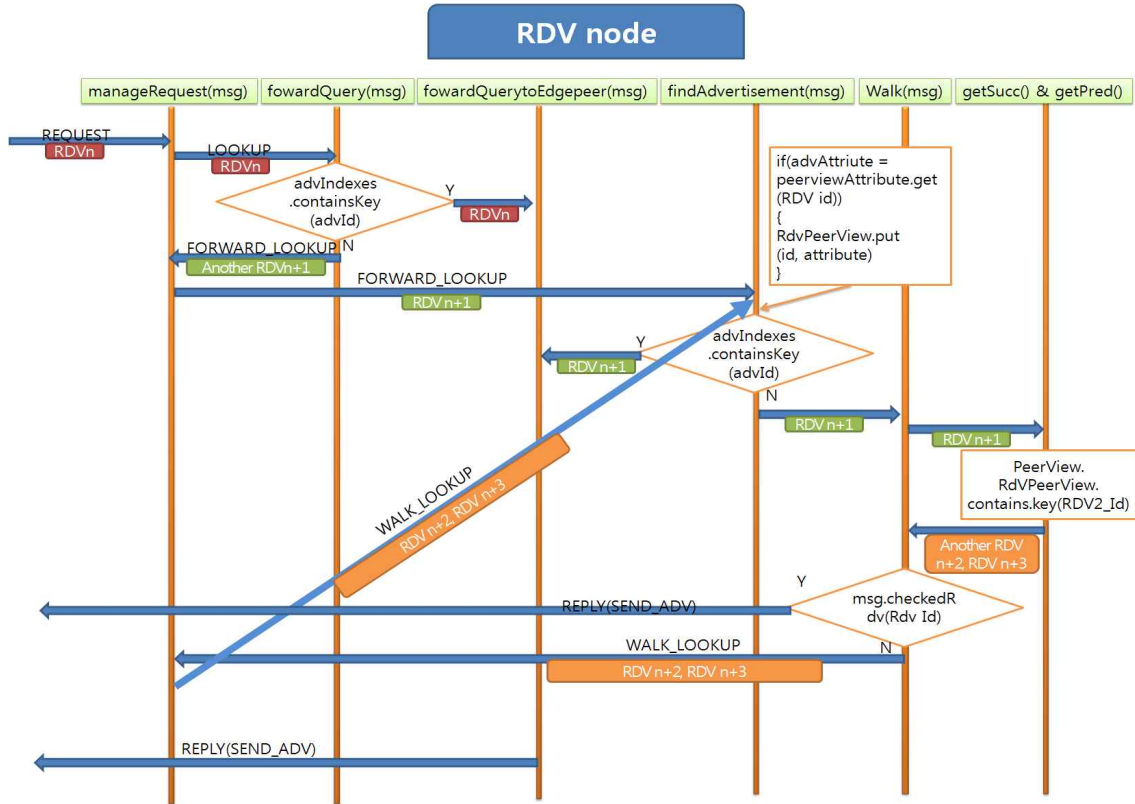


그림 7. 제안한 Peer 탐색 알고리즘 Sequence Diagram
 Fig. 7. Proposed Peer Search Algorithm Sequence Diagram.

RDV 속성이 같은 다음 목적지 Peer를 다시 Walk()으로 보내준다. Walk()안에서는 목적지 후보 Peer가 방문했던 Peer인가를 검사한 후 방문하지 않은 RDV라면 WALK_LOOKUP 메시지를 보내 탐색 과정을 진행한다. 그러나 방문했던 Peer라면 Walk() 안에서 Adv를 찾을 수 없다고 Adv를 탐색 요청한 Edge에게 탐색 실패 메시지를 보내준다.

은 RDV에 Edge가 연결된다. 각각의 RDV들은 A loosely-consistent Distributed Hash을 사용하여 각각의 RDV들 간의 정보를 유지한다.

IV. 알고리즘의 시뮬레이션 및 성능 평가

4-1 시뮬레이션 모델 및 토폴로지

본 논문에서는 JXTA-Sim을 기반으로 논문에서 제안하는 Peer 탐색 알고리즘의 성능을 평가하기 위해 다음과 같은 토폴로지 및 Edge, RDV, RDV의 속성을 추가 하였다.

Edge와 RDV는 네트워크에 참여하게 될 때, Peer의 성향에 맞는 속성을 부여 받으며, Edge의 속성과 같

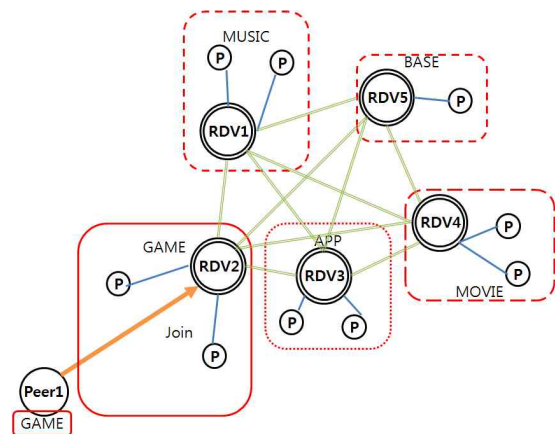


그림 8. Peer 성향을 고려한 네트워크 토폴로지
 Fig. 8. Considering Peer's Attribute Network Topology.

Edge가 생성하는 Adv는 Edge의 속성을 가지게 되며, Edge는 자신이 연결된 RDV를 활용하여 Adv를

publish 하게 되며, 각각의 RDV는 publish한 Adv의 Id와 해당 Edge의 데이터를 저장하게 된다.

4-2 성능평가

JXTA-Sim의 Peer 탐색 알고리즘과 Peer의 성향을 고려한 Peer 탐색 알고리즘의 성능 평가를 하였다. 다음 표 1은 시뮬레이션 환경변수이며 RDV의 수를 증가 시키면서 Peer 탐색 알고리즘의 성능 평가를 실시하였다.

표 1. 시뮬레이션 환경변수
Table 1. Environment variable of Simulation.

Parameter	Value
MAX_EDGES_PER_RDV	5
MAX_WALKER_HOPS	5
MAX_PEERVIEW_SIZE	100
REPLICATION_DISTANCE	1
CACHE_SIZE	10
RPV_ENTRY_DEFAULT_EXPIRATION	800
PEERVIEW_REFRESH_INTERVAL	500000
PEERVIEW_ENTRIES_FLUSH_INTERVAL	200
SIMULATION_STEP	1000
NUM_PUBLISHERS	10
NUM_SEARCHERS	10
RDVS_IN	0
RDVS_OUT	0

본 논문의 목표는 Fall-back 현상으로 인한 WALK-LOOKUP 메시지수의 줄이고, 기존 JXTA-Sim Peer 탐색 알고리즘의 최악의 탐색 상황을 속성을 중심으로 Peer 탐색을 시도하여 개선한다.

다음 그림 9는 기존 알고리즘과 제안한 알고리즘의 Peer 수의 증가에 따른 Peer 탐색 성공률 나타낸 그림이다.

Peer 수의 변화에 따른 시뮬레이션은 총 100개의 Peer 탐색 메시지를 표본으로 성공률을 측정 하였다. JXTA-Sim Peer 탐색 알고리즘과 제안한 Peer 탐색 알고리즘의 성능은 동등한 성능을 보여주고 있다. RDV의 수가 증가 할수록 Peer 탐색 성공률은 낮아지는 것은 Peer의 RPV의 크기는 한정 되어 있고 RPV안에 모든 RDV 정보를 저장할 수가 없기 때문에 성공률은 낮아진다.

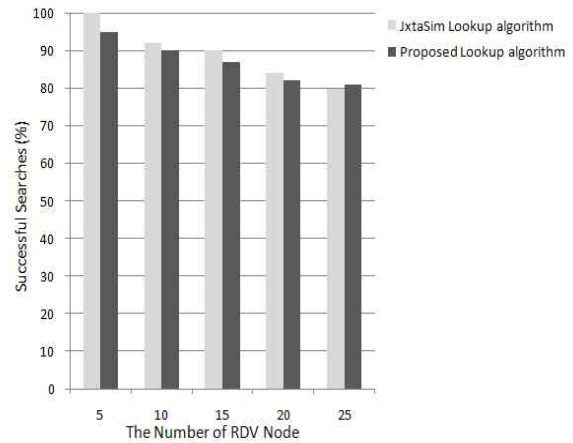


그림 9. Peer 수에 따른 Peer 탐색 성공률
Fig. 9. Successful Searches depending on numbers of peer.

다음 그림 10은 시뮬레이션에서 보내진 WALK_LOOKUP 메시지의 수를 측정하여 JXTA-Sim Peer 탐색 알고리즘과 제안된 Peer 탐색 알고리즘의 성능 평가를 하였다.

JXTA-Sim의 Peer 탐색 알고리즘의 WALK_LOOKUP 메시지 수 보다 제안한 Peer 탐색 알고리즘의 메시지수가 더 적게 보내지는 것을 확인 하였다.

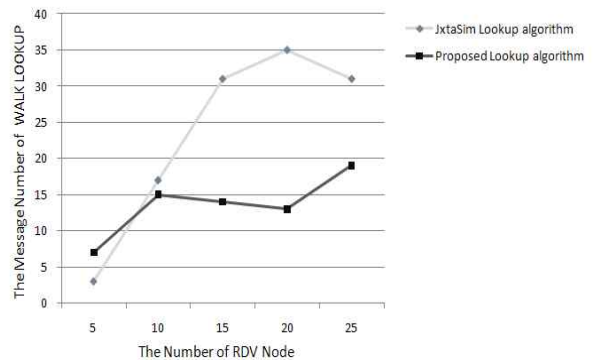


그림 10. Peer 수에 따른 WALK_LOOKUP 메시지수
Fig. 10. The Number of WALK_LOOKUP message depending on numbers of peer.

본 논문의 목표인 WALK_LOOKUP 메시지의 수를 줄이기 위하여 Peer간에 주고받는 Discovery 메시지에 방문한 RDV의 Id값들을 메시지에 추가하였다. 메시지가 RDV에 방문하였을 때 먼저 이 Peer가 방문했

던 Peer인가를 검사하고 WALK_LOOKUP 절차를 진행 하게 되어, NH 값이 무한정 증가 하여 Fall-back 현상으로 인한 메시지 증가를 줄일 수 있다.

V. 결 론

본 논문에서는 P2P 오버레이 네트워크에 대한 전반적인 내용과 플랫폼과 네트워크 디바이스에 독립적인 환경을 지원하는 JXTA 네트워크 환경에서 분산된 자원을 효율적으로 탐색하고 사용자 성향에 맞는 맞춤형 탐색 알고리즘을 설계하였다. 기존 A loosely-consistent DHT 방식에서 발생하던 Fall-back 현상으로 WALK_LOOKUP 메시지가 발생하는 수를 줄이고 사용자 성향에 맞는 맞춤형 Peer 탐색 알고리즘을 제안하여 Peer 탐색 횟수를 줄일 수 있었다.

제시한 사용자 성향 중심의 네트워크 모델에서 얻을 수 있는 효과로는 사용자의 성향별로 그룹을 나누어 Super peer 역할을 하는 RDV가 자신의 Edge들을 관리함으로써 해당 사용자의 성향에 맞는 Peer 탐색을 할 때 보다 빨리 Peer를 탐색 할 수 있었다.

향후 연구 방향은 첫째, 플랫폼과 단말의 종류에 독립적인 JXTA의 특징을 활용하여 제안한 알고리즘을 기반으로 실제 네트워크상에서 테스트 배드를 구축하여 성능 평가를 해야 할 것이다. 둘째, WALK_LOOKUP 메시지 수를 줄이기 위하여 논문에서 제안한 알고리즘과 Max Num Hops와의 최적화 되는 값을 찾아 실제적으로 본 논문에서 제안하는 사용자 성향에 맞는 네트워크 토폴로지 외의 다른 네트워크상에서도 메시지의 수를 줄일 수 있는 최적화 모델을 찾아야 할 것이다.

감사의 글

이 논문은 한국기술교육대학교 교육연구진흥비지원 프로그램의 (일부) 지원에 의하여 수행되었음.

참 고 문 헌

- [1] Zhang Dong, Wu chunming, Xiong Wei, Jiang ming, "Research on virtual network design based on virtual router," Future Information Networks, *ICFIN2009*, pp.66~69, Oct. 2009.
- [2] JXTA, "Project homepage," <http://www.java.net/projects/community/148>
- [3] Mohamed, B. T., Abdelaziz, M., and Pouyoul, E. "Project jxta : A loosely-consistent dht rendezvous walker," 2003.
- [4] P. Garcia, C. Pairet, R. Mond'egar, J. Pujol, H. Tejedor, and R. Rallo, "PlanetSim: A New Overlay Network Simulation Framework," *In Proc. SEM'4*, pp 123~136, September 2004.
- [5] JXTA-Sim, "JXTA-SIM A simulator for evaluation the JXTA Lookup Algorithm," <https://www.scss.tcd.ie/publications/tech-reports/reports.09/TCD-CS-2009-47.pdf>
- [6] B. Traversat, M. Abdelaziz, and E. Pouyoul, "Project JXTA : A Loosely-Consistent DHT Rendezvous Walk," <http://www.jxta.org/project/www/docs/jxta-dht.pdf>

이 종 서 (李鍾瑞)



2009년 8월 : 한국기술교육대학교
인터넷미디어공학과(공학사)
2009년 9월 ~ 현재 : 한국기술교육대학교
컴퓨터공학과(공학석사)
관심분야 : P2P, 무선 네트워크

문 일 영 (文日永)



2000년 2월 : 한국항공대학교 항공통신
정보공학과 (공학사)
2002년 2월 : 한국항공대학교 항공통신
정보공학과 (공학석사)
2005년 2월 : 한국항공대학교 정보통신
공학과 (공학박사)
2004년 ~ 2005년 : 한국정보문화진흥원

선임연구원

2005년 3월 ~ 현재 : 한국기술교육대학교 컴퓨터공학부
부교수

관심분야 : 무선 인터넷 응용, 무선 인터넷, 모바일 IP