

모바일 P2P 네트워크에서 피어의 연결성을 고려한 그룹 기반 캐시 공유 기법

Group-based Cache Sharing Scheme Considering Peer Connectivity in Mobile P2P Networks

김재구*, 윤수용*, 임종태*, 이석희**, 복경수*, 유재수*
충북대학교 정보통신공학부*, 동아방송예술대학교 뉴미디어콘텐츠과**

Jaegu Kim(kimjaegu@chungbuk.ac.kr)*, Sooyong Yoon(ysy@chungbuk.ac.kr)*,
Jongtae Lim(jtlim@chungbuk.ac.kr)*, Seokhee Lee(seoklee@dima.ac.kr)**,
Kyoungsoo Bok(ksbok@chungbuk.ac.kr)*, Jaesoo Yoo(yjs@chungbuk.ac.kr)*

요약

모바일 P2P 네트워크에서 사용자 요청을 효과적으로 처리하기 위해 캐시 공유 기법들이 연구되고 있다. 본 논문에서는 모바일 P2P 네트워크에서 피어의 연결성에 기반을 둔 클러스터를 구축하여 캐시를 공유하는 기법을 제안한다. 제안하는 기법은 연결성이 오랫동안 유지될 수 있는 피어들을 하나의 클러스터로 구성하여 캐시를 공유한다. 클러스터 내부에 있는 피어들의 캐시를 하나로 사용함으로써 데이터의 중복을 감소시키고 캐시 공간을 효율적으로 사용한다. 또한, 캐시 공간을 데이터 캐시와 임시 캐시 두 부분으로 분할하여 사용한다. 임시 캐시를 활용하여 클러스터의 토폴로지가 변경되거나 캐시 데이터 교체 시, 지연 시간을 감소시킬 수 있다. 일반적인 기법은 협력적 캐시 처리 절차를 통해 질의가 전달되지만 통신비용을 줄이기 위해 1-홉 피어 캐시확인과 클러스터 헤더로 가는 경로에 있는 피어의 캐시를 확인한다. 성능 평가 결과 제안하는 기법이 기존 기법에 비해 캐시 적중률도 높고 지연시간을 감소시키는 것을 확인할 수 있었다.

■ 중심어 : | 모바일 P2P | 공유 | 캐시 | 연결성 |

Abstract

Recently, cache sharing methods have been studied in order to effectively reply to user requests in mobile P2P networks. In this paper, we propose a cache sharing scheme based on a cluster considering the peer connectivity in mobile P2P networks. The proposed scheme shares caches by making a cluster that consists of peers preserving the connectivity among them for a long time. The proposed scheme reduces data duplication to efficiently use the cache space in a cluster. The cache space is divided into two parts with a data cache and a temporary cache for a cache space. It is possible to reduce the delay time when the cluster topology is changed or the cache data is replaced utilizing a temporary cache. The proposed scheme checks the caches of peers in a route to a cluster header and the caches of one-hop peers in order to reduce the communication cost. It is shown through performance evaluation that the proposed scheme outperforms the existing schemes.

■ keyword : | Mobile P2P | Share | Cache | Connectivity |

* 이 논문은 2012년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업(2012R1A1A2041898), 미래창조과학부 및 정보통신산업진흥원의 대학IT연구센터육성 지원사업/IT융합고급인력과정지원사업(NIPA-2014-H0301-14-1022), 교육부와 한국연구재단의 지역혁신인력양성사업(No.2013H1B8A2032298)과 미래창조과학부의 방송통신정책연구센터운영지원사업(KCA-2013-003)의 연구결과로 수행되었음

접수일자 : 2014년 08월 19일

심사완료일 : 2014년 09월 11일

수정일자 : 2014년 09월 11일

교신저자 : 유재수, e-mail : yjs@chungbuk.ac.kr

1. 서론

최근 GPS를 탑재한 스마트폰의 보급이 확산되면서 위치 기반 서비스(Location-Based Service: LBS)에 대한 연구가 활발히 이뤄지고 있다. 사용자는 모바일 기기를 통해 현재 위치에서 날씨, 교통 정보, 편의시설 등 다양한 정보를 실시간으로 얻을 수 있다. 위치 기반 서비스를 활용하여 다양한 정보를 효율적으로 찾아주는 질의 처리에 대한 많은 연구가 진행되고 있다. 대부분의 위치 기반 서비스는 클라이언트/서버(Client/Server: C/S) 구조로 되어 있다. C/S 방식은 서비스를 공급하는 서버와 서비스를 제공받는 클라이언트로 되어 있다. 하지만 C/S 방식은 서버에 문제가 발생하거나 병목 현상이 발생하는 경우에는 정상적인 서비스를 제공하지 못한다. 또한, 사용자의 증가에 따라 서버 부하를 증가시킬 뿐만 아니라 확장성, 신뢰성, 유연성에 문제가 발생하기 때문에 이러한 문제점을 해결하기 위해 각 피어끼리 자율적으로 필요한 자원을 공유하고 확장성이 높은 P2P(Peer-to-Peer) 구조가 제안되었다.

P2P 네트워크란 클라이언트나 서버란 개념이 없이 동등한 피어들이 서로 클라이언트와 서버 역할을 동시에 수행한다. 피어간 동등한 수평 연결망을 통해 피어들끼리 자원을 서로 나누고 공유하는 형태이다. 최근 모바일 기기 및 통신의 성능이 향상되면서 P2P 구조를 모바일 환경에 접목시킨 모바일 P2P(Mobile Peer-to-Peer) 네트워크가 연구되었다. 모바일 P2P 네트워크는 IEEE 802.11, 블루투스(bluetooth), UWB(Ultra Wide Band)와 같은 단거리 무선 통신을 사용하여 이동기기들 사이에 파일 공유, 콘텐츠 전송, 소셜 네트워크, 광고 및 추천 등을 제공한다. 유선 환경에서의 P2P 네트워크는 신뢰성을 보장하여 안정적으로 통신이 가능하지만 모바일 P2P 네트워크에서 연결이 수시로 변화하기 때문에 신뢰성이 보장되지 않으며 각각의 모바일 기기들은 한정된 전원 그리고 작은 저장 공간 문제를 가지고 있다[1]. 따라서 모바일 기기의 한계로 인해 유선 환경에서 사용되는 P2P 기술을 모바일 P2P 네트워크에 직접 적용하기 어렵다.

모바일 P2P 네트워크에서 모바일 기기들 사이에 데

이터 공유 및 질의 처리 성능을 향상시키기 위해 캐시를 사용하는 방법들이 사용되고 있다[2][3]. 캐시란 자원을 주고받을 때 자주 사용하는 자원을 캐시에 저장함으로써 디스크 I/O와 접근 시간을 감소시켜 요청되는 질의에 빠르게 응답하기 위한 전략을 말한다. 일반적으로 분산 데이터 처리에서 캐시는 자신의 로컬 캐시만을 사용하는 것이 아니라 이웃한 모바일 기기의 캐시를 공유하여 사용자의 요청을 처리하는 협력적 캐시를 사용한다. 최근 몇 년간 모바일 P2P 네트워크에서 협력적 캐시를 이용한 데이터 처리 기법에 대한 연구들이 진행되고 있다[4-6]. 협력적 캐시는 피어간 통신비용뿐만 아니라 배터리 소모를 감소시키고 캐시 적중률을 증가시킨다. 협력적 캐시 기법을 활용하지만 클러스터 기법을 활용하지 않고 주변 피어와 상관없이 독립적으로 캐시 교체 전략을 사용하게 되면 활용도가 높은 인기 데이터를 중복적으로 유지하게 된다. 각 피어마다 인기 데이터를 전부 가지고 있다면 동일한 데이터 접근은 용이하지만 피어마다 캐시 공간 활용도가 저하되며 다양한 데이터를 저장하지 못하게 된다. 이로 인해 캐시 적중률 감소와 저장 공간 낭비가 초래된다. 이러한 문제를 해결하기 위해 캐시 공유 기법을 사용한다. 캐시 공유 기법이란 피어들을 클러스터로 구성하여 각 캐시 공간을 하나의 캐시처럼 사용하는 기법이다. 클러스터에 포함되어 있는 피어들끼리 중복된 데이터를 제거하여 마치 여러 개의 캐시 공간을 하나의 커다란 캐시 공간처럼 사용한다. 또한 인기 있는 데이터를 피어마다 분배하여 가지고 있으므로 부하를 방지할 수 있다. 대표적으로 캐시를 공유하기 위한 연구들이 진행되었다[7][8].

본 논문에서는 모바일 P2P 네트워크에서 피어의 연결성을 고려한 그룹 기반 캐시 공유 기법을 제안한다. 제안하는 기법은 연결성이 오랫동안 유지될 수 있는 피어들을 하나의 클러스터로 형성하여 캐시를 공유한다. 클러스터 내부에 있는 피어들을 하나의 캐시로 사용함으로써 데이터의 중복을 감소시키고 캐시 사용 공간을 효율적으로 사용한다. 또한, 캐시 공간을 데이터 캐시와 임시 캐시 두 부분으로 분할하여 사용하고 임시 캐시를 활용하여 클러스터의 토폴로지가 변경되거나 캐시 데

이터 교체 시, 지연시간을 감소시킬 수 있다. 일반적인 기법은 협력적 캐시 처리 절차를 통해 질의가 전달되지만 통신비용을 줄이기 위해 1-홉 피어 캐시확인과 클러스터 헤더로 가는 경로에 있는 피어의 캐시를 확인한다.

본 논문의 나머지 구성은 다음과 같다. 2장에서는 본 논문과 관련된 기존 연구들을 기술하고 3장에서는 제안하는 캐시 공유 기법에 대해 기술한다. 4장에서는 제안하는 기법의 우수성을 입증하기 위해 성능 평가를 수행한 결과를 기술한다. 마지막 5장에서는 논문의 결론 및 향후 연구에 대해 기술한다.

II. 관련 연구

C. Chow, H. V. Leong[9]에서는 효율적인 클러스터 형성과 데이터의 활용도를 높이기 위해 피어의 이동 패턴과 유사한 데이터 선호도를 보여주는 집합으로 tightly-coupled group(TCG)을 정의한다. TCG 알고리즘을 통해 적합한 피어를 동적으로 발견하고 클러스터를 형성하고 유지하기 위해 제안된다. 협력적 캐시 제어 관리(cooperative cache admission control)와 협력적 캐시 교체(cooperative cache replacement)로 불리는 협력적 캐시 관리 프로토콜을 통해 데이터 접근성을 증가시키고 데이터 복제 본을 제어하도록 설계된다. 이 기법은 접근 대기 시간을 줄이고 캐시 적중률을 증가시킨다. 하지만 높은 전력 소모를 초래한다.

M. F. Caetano[10]에서는 각 피어들이 클러스터로 구성되어 있고 캐시 되어있는 데이터를 공통적으로 유지하는 목적으로 제안되었다. 캐시공간을 개인적인 캐시(private cache)와 공유된 캐시(shared cache) 두 부분으로 나누어 관리를 해준다. 개인적인 캐시는 피어가 개인적으로 데이터를 유지하는 공간이다. 공유된 캐시는 클러스터에 포함되어 있는 모든 피어가 공통적으로 데이터를 유지하는 공간이다. 헤더가 존재하여 인기 있는 데이터를 공유된 캐시에 공통적으로 유지 및 관리하여 캐시의 다양성과 캐시 적중률을 가능성을 증가시킨다.

Joy, Preetha Theresa[11]에서는 협력적인 캐시를 통

해 최근 접근 및 일관성 사이에 시간 간격과 크기 기반의 데이터 축출을 하는 E-LRU(Extended LRU) 캐시 교체 정책이 제안되었다. LRU는 교체 전략에 너무 적은 정보만 고려한다. 그래서 E-LRU는 데이터의 최근 참조된 시간 간격, 크기 그리고 일관성을 고려하여 교체 전략 사용한다. 또한 캐시공간을 가득 채우지 않고 임계점을 두어 크기를 조절한다. 캐시 공간에 빈 공간이 데이터 교체 시 발생하는 지연 시간을 감소시키는 용도로 사용된다. 캐시에 빈 공간을 두는 것은 버퍼와 같이 사용할 수 있다. 캐시가 가득 차 있는 상태에 데이터가 들어오면 교체 대상이 될 데이터를 찾고 교환까지 하는 시간에 발생하는 지연시간을 감소시킬 수 있다.

P. Victor Paul[12]에서는 서비스의 효율적인 검색을 가능하게하기 위하여 모바일 P2P 네트워크에 대해 service cache management(SCM)을 제안되었다. 모바일 P2P의 노드는 분산 방식으로 신장 트리의 집합으로 만들어진 분산 신장 트리라 불리는 통신 구조를 사용한다. 분산 신장 트리 구조를 사용하는 것은 모바일 P2P 네트워크 안에서 빠른 서비스 캐시와 검색을 가능하게 된다. 또한 시스템의 효율성을 향상시키기 위해, 분산 신장 트리 구조는 노드간의 최적의 경로를 식별하는 개미 집단 최적화를 사용한다. 향상된 가용성, 향상된 일관성, 확장성 및 증가된 적중률에 대한 캐시 서비스 검색 요구 시간을 감소시키는 관점에서 설명된다.

III. 제안하는 캐시 공유 기법

3.1 특징

기존 캐시 공유 기법에서는 클러스터에 포함된 각 피어들은 개별적인 교체 전략을 가지고 있다. 개별적인 교체 전략을 사용하면 클러스터 내부 피어들 또는 주변 피어들과 많은 중복된 데이터를 유지하게 된다. 이러한 경우 데이터 접근은 쉽겠지만 피어마다 캐시 활용도가 저하되고 다양한 데이터를 저장하기 어려워져 저장 공간 측면에서도 비효율적이다. 이러한 문제를 해결하기 위해 본 논문은 모바일 P2P 네트워크에서 피어의 연결성을 고려한 그룹 기반 캐시 공유 기법을 제안한다. 제

안하는 캐시 공유 기법은 클러스터에 포함되어 있는 피어들끼리 중복된 데이터를 제거하여 여러 개의 캐시 공간을 하나의 커다란 캐시 공간처럼 사용한다. 또한, 인기 있는 데이터는 피어마다 분배하여 가지고 있다. 이로 인해 다양한 데이터를 저장하여 효율적인 캐시 공간 사용할 수 있고 인기 있는 데이터 분배를 통한 피어마다 캐시 활용도를 높여 부하를 방지할 수 있다.

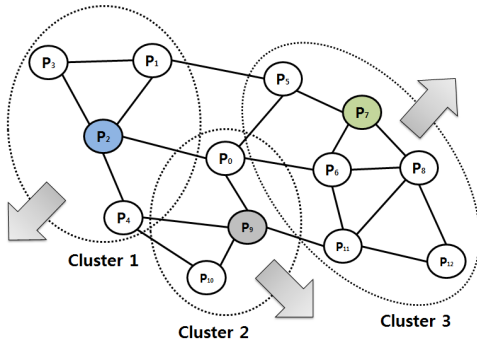


그림 1. 캐시 공유 환경

[그림 1]은 제안하는 캐시 공유 기법의 캐시 공유 환경을 나타낸다. 클러스터를 최대 n -홉으로 제한을 두어 소규모 클러스터를 많이 만드는 것을 목적으로 한다. 각 클러스터는 연결성이 지속적으로 유지될 수 있는 피어들끼리 클러스터를 형성한 상태이다. 연결성이란 이동 속도와 방향을 고려하여 각 피어가 1-홉 통신 가능한 상태로 얼마동안 유지할 수 있는지를 말한다. 각 피어는 X, Y 벡터 값을 가지고 있다. 연결성을 비교하기 위해 두 피어는 벡터 값과 통신 거리를 통하여 1-홉 통신 가능한 상태를 유지하는 시간을 구한다. 유지 시간은 연결성을 나타내고 이 연결성이 사용자자 정해둔 기준점 시간보다 높으면 각 피어는 서로 클러스터를 형성하거나 클러스터에 합류한다. 클러스터가 형성되면 중복을 제거하고 캐시 공유를 한다. 클러스터 헤더는 전체적으로 클러스터를 관리한다. 클러스터 피어 정보와 피어마다 데이터 정보를 가지고 있어 데이터를 요청할 때 클러스터 헤더를 통해 빠르게 데이터로 접근할 수 있게 해준다. Cluster 1과 2는 2-홉 구조로 구성되어 있고 Cluster 3은 3-홉 구조로 구성되어 있다. 원으로 표

시된 객체는 피어를 나타내고, P_2, P_7 그리고 P_9 는 클러스터 헤더 역할을 맡고 있는 피어이다. 그리고 모바일 기기 주위에 접속은 연결성이 유사한 피어들끼리 클러스터 형성한 것을 나타내고 피어 사이의 실선은 피어들 간에 통신이 가능한 상태를 의미한다. 마지막으로 화살표는 클러스터의 이동 방향을 나타낸다.

3.2 클러스터 생성 기법

제안하는 클러스터 생성 기법은 연결성이 지속적으로 유지될 수 있는 피어들끼리 클러스터를 형성한다. 먼저 피어는 비교 대상과 통신가능 상태가 되면 비교 대상이 무엇인지부터 판단한다. [그림 2]는 비교 대상이 클러스터 일 때 처리과정이다. [그림 2(a)]에서 P_0 가 클러스터에 연결되기 전에 먼저 클러스터가 n -홉 구조로 되어 있는지 판단한다. 클러스터가 n -홉 구조로 되어 있지 않다면 [그림 2(b)]와 같이 클러스터와 P_0 는 연결성을 판단한다. 하지만 n -홉 구조로 되어 있다면 P_0 가 클러스터에 연결 시 n -홉을 초과하는지를 판단한다.

[그림 3]은 P_0 가 클러스터에 연결실패 하는 경우를 보여준다. 만약 [그림 3(a)]와 같이 최대 홉 수를 초과한다면 P_0 는 클러스터에 연결 되지 못하고 그렇지 않다면 클러스터와 P_0 는 연결성을 판단한다. [그림 2(c)]와 같이 연결성을 판단할 때 P_0 와 클러스터에 있는 피어들 중 P_0 와 연결성이 가장 높은 피어와 비교를 하여 기준 값보다 높으면 클러스터에 연결이 되고 그렇지 않다면 [그림 3(b)]와 같이 연결 되지 못한다.

[그림 4]는 비교 대상이 클러스터가 없는 피어일 때 처리 과정을 보여준다. [그림 4(a)]에서 P_0 와 클러스터가 없는 피어는 비교는 서로 클러스터에 포함되어 있지 않기 때문에 바로 연결성을 비교한다. 연결성이 기준 값보다 높으면 [그림 4(b)]와 같이 두 피어는 클러스터로 형성하고 낮으면 [그림 4(c)]와 같이 연결에 실패하고 다른 피어나 클러스터를 찾는다.

클러스터가 형성되면 헤더를 선정한다. 1-홉 구조에서는 특별한 기준이 없기 때문에 무작위로 피어를 선택하고 2-홉 구조에서는 피어들 중간에 있는 피어가 헤더로 선정된다. n -홉 구조에서는 피어들 중간에 있고 피어들과 가장 통신을 많이 할 수 있는 피어로 선정한다.

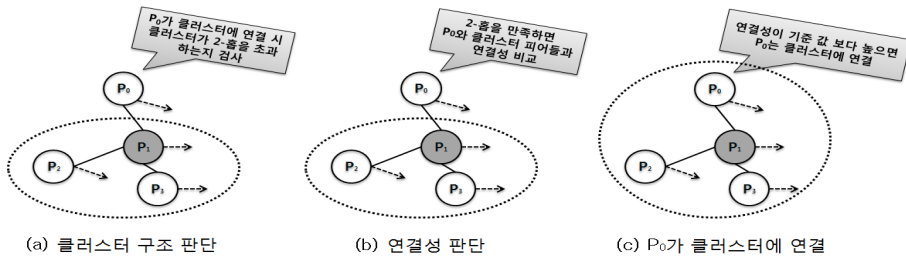


그림 2. P₀가 클러스터에 연결되는 과정(2-홉 구조 기준)

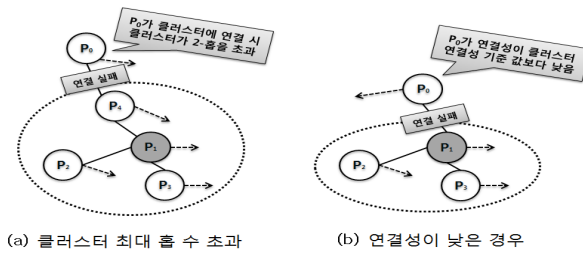


그림 3. P₀가 클러스터에 연결실패 하는 경우(2-홉 구조 기준)

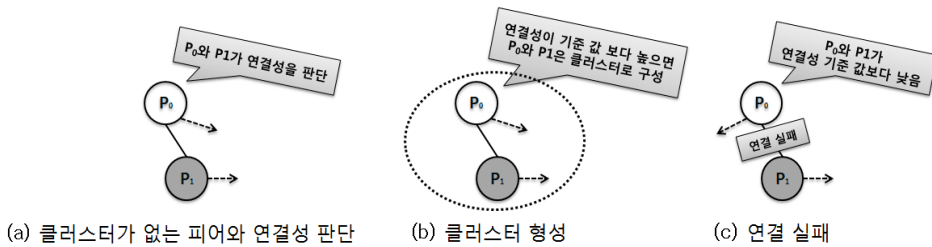


그림 4. P₀와 클러스터가 없는 피어와 연결되는 과정

하지만 클러스터가 피어에 연결될 때마다 헤더를 재선정하게 되면 추가적인 통신비용이 발생하기 때문에 기존 헤더는 그대로 유지하고 헤더가 클러스터와 연결성이 낮아지거나 통신을 수행하기 어려운 가장자리에 있을 경우 재선정을 해준다. 클러스터 헤더는 클러스터 내에 모든 피어를 관리하고 클러스터 헤더를 통해 통신을 수행한다. 또한, 클러스터에 대한 모든 정보를 수집하고 관리 유지한다. 클러스터에는 클러스터 헤더가 유지하는 헤더 테이블 그리고 각 피어가 유지하는 피어 테이블이 존재한다. 헤더 테이블은 $\langle D_{id}, P_{id}, R_v \rangle$ 구조로 되어 있으며 클러스터의 모든 데이터와 그에 따른

피어 정보를 저장 관리 한다. D_{id} 는 데이터 식별자, P_{id} 는 피어 식별자 그리고 R_v 는 클러스터 내부에서 헤더로부터 피어의 경로를 나타낸다. 피어 테이블은 $\langle D_{id}, D_v \rangle$ 구조로 되어 있고 피어의 데이터를 유지한다. D_{id} 는 데이터 식별자 그리고 D_v 는 데이터를 대표할 수 있는 값이다. 또한, 피어는 $\langle \text{vector}_X, \text{vector}_Y, P_{list}, C_{Header} \rangle$ 를 관리한다. vector_X 와 vector_Y 는 피어의 벡터를 의미한다. P_{list} 는 네트워크에서 1-홉 통신이 가능한 피어들의 목록이고 C_{Header} 는 자신이 포함되어 있는 클러스터 헤더를 의미한다. 1-홉 통신이 가능한 피어들을 관리하여, 질의 결과를 반환 할 때 클러스터 헤

더를 거치지 않고 전송하여 클러스터 헤더의 부하를 감소시킨다.

[그림 5]은 클러스터를 형성하는 과정을 흐름도로 나타낸다. 흐름도에서 보여주는 바와 같이 크게 3가지를 비교한다. 피어와 비교대상이 무엇인가, 클러스터 구조 형태는 어떤가, 연결성이 기존 값보다 높은가를 비교한다.

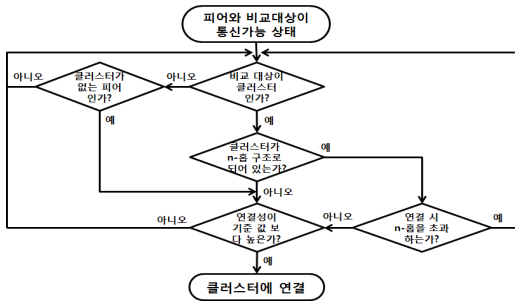


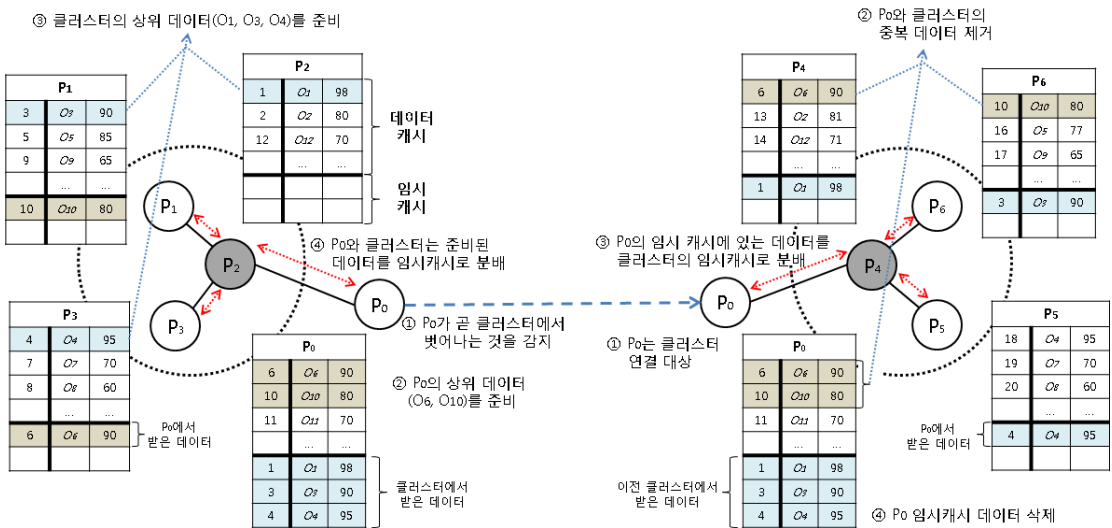
그림 5. 클러스터 형성 과정

3.3 캐시 공유 및 관리 기법

제안하는 클러스터 형성 기법을 통해 연결성을 고려하여 클러스터가 형성이 되면 클러스터에 있는 피어들

은 공유 캐시 전략을 수행한다. 클러스터에 포함되어 있는 피어들끼리 중복된 데이터를 제거하여 마치 여러 개의 캐시 공간을 하나의 커다란 캐시 공간처럼 사용하는 것처럼 사용한다. 또한 인기 있는 데이터를 피어마다 분배하여 가지고 있으므로 부하를 방지할 수 있다.

각 피어는 캐시 공간을 데이터 캐시와 임시 캐시 두 부분으로 나누어 관리한다. 데이터 캐시는 데이터를 유지하는 캐시 저장 공간이다. 일반적으로 캐시 데이터를 관리하고 교체 전략이 수행하는 공간이다. 임시 캐시는 임시적으로 캐시 데이터를 저장하는 공간이고 캐시 공간을 일부를 고정적으로 차지한다. 피어가 클러스터에 연결되어 있는 동안 임시 캐시를 공간을 빈 공간으로 만든다. 그 이유는 캐시공간이 가득 차있는 상태에 빠르게 여러 개의 데이터가 들어오면 캐시는 교체 전략을 실시하여 새로운 데이터를 받고 기존 데이터를 제거한다. 하지만 데이터 흐름은 활발한 모바일 P2P 네트워크에서 일일이 데이터를 교체하게 되면 캐시는 데이터 교체 지연시간이 발생한다. 그러므로 임시 캐시의 빈 공간을 버퍼처럼 사용하여 데이터 교체 지연시간을 감소시킨다. 또한, 클러스터에서 피어 연결되거나 벗어나는 경우 데이터 교환 용도로 사용한다.



(a) 클러스터에서 P0가 벗어나는 상황

(b) 클러스터에 P0 연결

그림 6. P0가 클러스터에 연결 및 벗어나는 상황

[그림 6]은 피어 P_0 가 클러스터에 연결되거나 벗어나는 상황을 보여준다. [그림 6(a)]는 P_0 가 클러스터와 연결성이 약해져서 클러스터를 벗어나는 경우를 나타낸다. P_0 가 클러스터와 연결성이 약해질 때, P_0 는 곧 클러스터에서 벗어난다. 클러스터에 포함되어 있는 각 피어에게는 인기 있는 데이터가 분배가 되어 있다. P_0 가 클러스터를 벗어날 때, 아무런 데이터 교환이 없다면 P_0 와 클러스터는 더 이상 서로가 가진 인기 있는 데이터에 접근하지 못하기 때문에 손해가 발생한다. 그러므로 P_0 가 클러스터에서 벗어날 때, 클러스터에서 P_0 를 제외한 다른 피어의 상위에 분포되어 있는 O_1, O_3 그리고 O_4 와 같은 인기 있는 데이터를 P_0 의 임시 캐시에게 전달하고 P_0 는 자신의 상위에 분포되어 있는 데이터(O_6, O_{10})를 클러스터의 임시 캐시에 전달한다. 클러스터에서 벗어난 P_0 는 캐시 데이터가 손실 될 수 있기 때문에 다른 클러스터로 연결되기 전까지 임시 캐시를 빈 공간으로 만들지 않는다. 다른 클러스터로 연결되는 경우에 이전 클러스터로부터 받은 임시 캐시에 있는 인기 있는 데이터를 새로 연결된 클러스터에서도 활용할 수 있다.

[그림 6(b)]는 클러스터에 P_0 가 클러스터 생성 조건을 만족하여 클러스터에 연결되는 경우에 대해 나타낸다. P_0 가 클러스터에 연결되는 경우 P_0 는 클러스터 헤더에

게 자신이 가지고 있는 캐시 데이터 정보를 전송한다. P_0 의 정보를 전송을 받은 클러스터 헤더는 클러스터의 캐시 데이터 정보와 비교한 다음 중복되는 데이터가 있다면 P_0 캐시에 있는 중복되는 데이터를 삭제한다. 중복 데이터가 삭제되면 P_0 에 있는 임시 캐시 데이터를 클러스터에 존재하는 피어들의 임시 캐시로 전달한다. P_0 에 있는 임시 캐시 데이터를 클러스터에 있는 피어의 임시 캐시에 데이터 크기에 따라 골고루 분배 해준다. P_0 에 있는 임시 캐시 데이터를 분배하는 이유는 다른 클러스터에 있는 인기 있는 데이터를 새로 연결되는 클러스터에 분배하여 데이터 최신성 유지와 통신비용을 줄이기 위해서이다. 전달이 완료되면 P_0 의 임시 캐시에 있는 데이터는 삭제한다. P_0 로부터 데이터를 받은 클러스터 피어들은 데이터 캐시에 존재하는 데이터와 임시 캐시 공간에 있는 데이터를 대상으로 로컬 캐시 교체 전략을 사용하여 데이터 캐시에 저장할 데이터를 결정한다. 로컬 캐시 교체 전략을 완료하면 임시 캐시 공간에 있는 데이터를 삭제한다.

3.4 캐시공유 환경에서 질의 처리

제안하는 캐시 공유 환경에서 질의 처리 기법은 질의

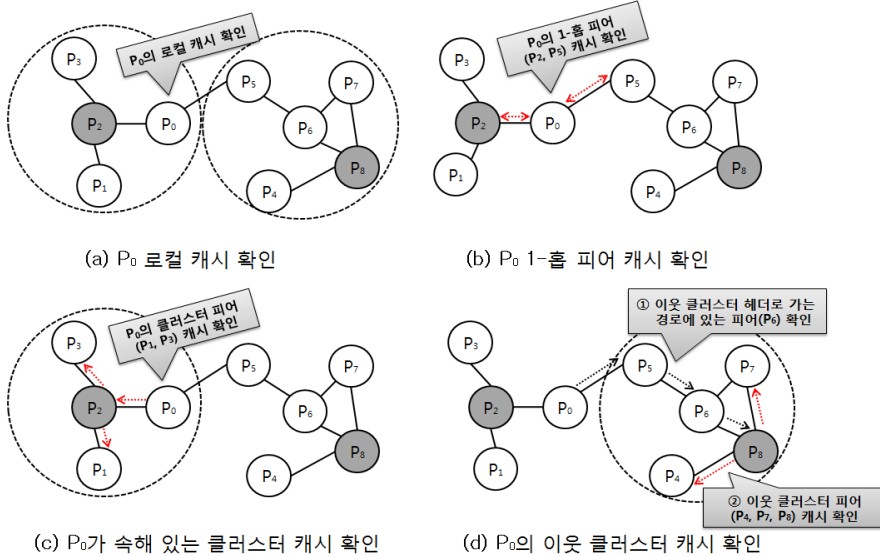


그림 7. P_0 의 질의 처리 과정

를 요청한 피어가 질의를 요청하고 어떠한 흐름으로 질의가 피어들에게 전달하는 과정에 대해 설명한다. 일반적인 기법은 협력적 캐시 처리 절차를 통해 질의가 전달되지만 통신비용을 줄이기 위해 1-홉 피어 캐시확인 과 클러스터 헤더로 가는 경로에 피어의 캐시를 확인하는 절차를 추가했다. [그림 7(a)]에서 P₀가 질의를 요청하면 자신의 로컬 캐시에서 일치하는 데이터를 검색한다. 질의를 만족하면 데이터를 P₀에 반환한다. [그림 7(b)]에서는 로컬 캐시 적중 실패 시 P₀는 1-홉에 있는 피어들의 캐시를 확인한다. 제안하는 기법에서는 자신이 속해 있는 클러스터를 먼저 확인 하는 것이 아니라 1-홉에 있는 피어들을 확인한다. 그 이유는 질의에 만족하는 데이터가 1-홉 안에 존재하지만 클러스터 헤더로 질의를 전달한 후 클러스터에 일치하는 데이터를 찾고 질의가 온 경로에 따라 반환해주는 경우가 발생할 수 있다. 하지만 1-홉에 있는 피어를 먼저 확인하는 작업을 한다면 위와 같은 문제를 해결하여 통신비용을 감소시킬 수 있다. 이와 같이 클러스터 헤더로 전송하는 과정에서도 지나가는 피어도 확인 작업을 하고 데이터가 없다면 다시 질의를 전송한다. [그림 7(c)]에서는 1-홉에 있는 피어와 클러스터 헤더로 가는 경로에 캐시 적중 실패 시 클러스터 캐시 확인한다. 질의를 받은 클러스터 헤더는 헤더 테이블에 가지고 있는 데이터 식별자와 질의를 비교한다. 질의가 적중하면 클러스터 헤더는 질의를 가지고 있는 피어에게 데이터를 요청하고 요청 메시지를 받은 피어는 데이터를 찾아서 P₀에게 반환한다. [그림 7(d)]에서는 클러스터 캐시 적중 실패 시 클러스터 헤더는 질의를 이웃 클러스터 헤더에게 전달한다. 이웃 클러스터 헤더로 가는 경로 피어에 데이터가 존재 시 P₀에게 데이터를 반환하고 없을 시 이웃 클러스터 헤더에게 전달이 되고 클러스터 질의처리는 동일하게 처리한다.

[그림 8]은 [그림 7]에서 설명한 캐시 공유 환경에서 질의 처리 기법을 흐름도로 나타낸다. 흐름도에서 보여주는 바와 같이 질의처리는 크게 4가지 절차로 진행된다. 로컬 캐시 확인, 1-홉 피어 캐시 확인, 클러스터 캐시 확인 그리고 이웃 클러스터 캐시 확인 순으로 진행된다.

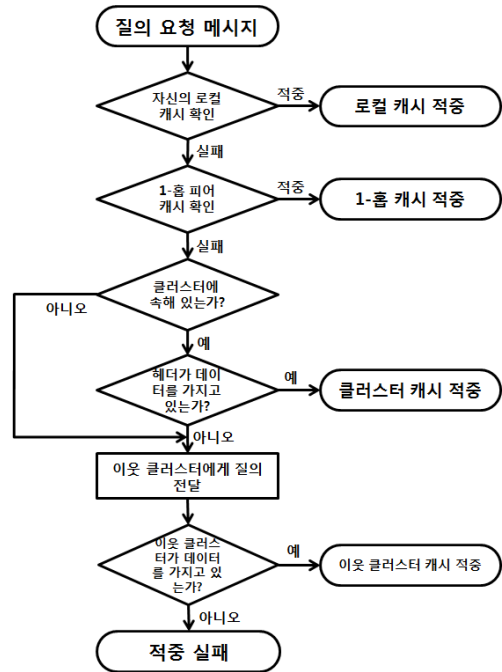


그림 8. 제안하는 협력적 캐시 질의 처리 절차

IV. 성능평가

성능 평가는 Intel(R) Core(TM) i7 프로세서, 4G 메모리를 가진 Window 7 운영체제를 사용하는 PC에서 수행하였다. 실험은 500×500에서 피어의 개수를 200, 데이터 개수를 500, 캐시 크기를 30개 그리고 클러스터 크기를 2-홉으로 기준을 잡고 데이터 개수, 캐시 크기 그리고 피어의 개수를 변화시켜가며 캐시 적중률과 지연시간을 측정했다. 피어의 위치와 데이터는 모두 무작위로 생성하였다. 제안하는 기법의 우수성을 확인하기 위해 GroCoca(GCC)[9]와 CBCCA[10]를 비교하여 성능 평가를 수행했다. GCC는 피어의 이동 패턴과 유사한 데이터 선호도를 보여주는 집합으로 tightly-coupled group(TCG)를 사용한 클러스터 형성 기법을 사용하고 CBCCA는 공유된 캐시를 유지하며 클러스터 전체가 인기 있는 데이터를 공유할 수 있는 기법을 사용한다. 제안하는 기법은 피어의 이동 패턴을 통해 연결성이 오랫동안 유지될 수 있는 피어들끼리 클러스터를 형성하

고 각 피어들의 캐시 공간 전체를 공유하는 기법을 사용한다.

[그림 9]은 데이터 수에 따른 캐시 적중률과 지연시간을 나타낸 것이다. 데이터 개수를 500에서 900까지 변화시켜 캐시 적중률과 지연시간의 변화량을 측정했다. 네트워크상에 데이터 개수가 많아지면 데이터가 다양해지는 것을 의미한다. 캐시에 유지하는 데이터양은 제한되어 있기 때문에 데이터 개수가 증가함에 따라서 캐시 적중률이 낮아지는 것을 볼 수 있고 그만큼 데이터 교체도 많이 수행된다. 데이터 개수가 많아질수록 CBCCA는 GCC와 비슷한 적중률을 보인다. CBCCA는 클러스터에서 공통으로 캐시를 유지하기 때문에 데이터 개수가 많아질수록 클러스터에 데이터의 다양성이 떨어지므로 캐시적중률도 급격히 감소하는 것을 볼 수 있다. 그로인해 데이터 교체가 빈번히 발생하게 되어 지연시간도 증가하는 것을 볼 수 있다. 제안하는 기법이 GCC, CBCCA에 비해 캐시 적중률이 각각 약 16%, 9% 향상되었으며 지연시간이 약 50%, 16%가 감소되는 성능을 나타내었다.

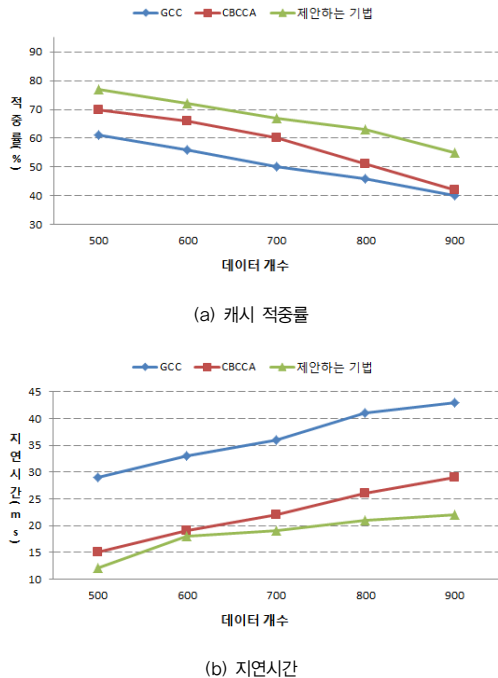


그림 9. 데이터 수에 따른 성능 평가

[그림 10]은 캐시 크기에 따른 캐시 적중률을 나타낸 것이다. 캐시 크기(데이터 개수)를 10에서 50까지 변화시켜서 캐시 적중률과 지연시간의 변화량을 측정했다. GCC와 CBCCA는 캐시 크기가 작을 때 제안하는 기법에 비해 낮은 적중률과 지연시간을 보인다. CBCCA는 캐시 캐시를 공유하기 때문에 크기가 증가할수록 클러스터가 유지라는 데이터가 다양해지고 많은 양을 보유한다. 제안하는 기법은 캐시 전체를 공유하기 때문에 캐시 크기가 작아도 클러스터 전체로 보면 커다란 캐시가 되기 때문에 다양한 데이터를 유지하기 때문에 높은 적중률을 나타내고 그로인해 교체 전략이 빈번히 발생하지 않아 낮은 지연시간을 나타낸다. 제안하는 기법이 GCC, CBCCA에 비해 캐시 적중률이 각각 약 21%, 10% 향상되었으며 지연시간이 약 56%, 20%가 감소되는 성능을 나타내었다.

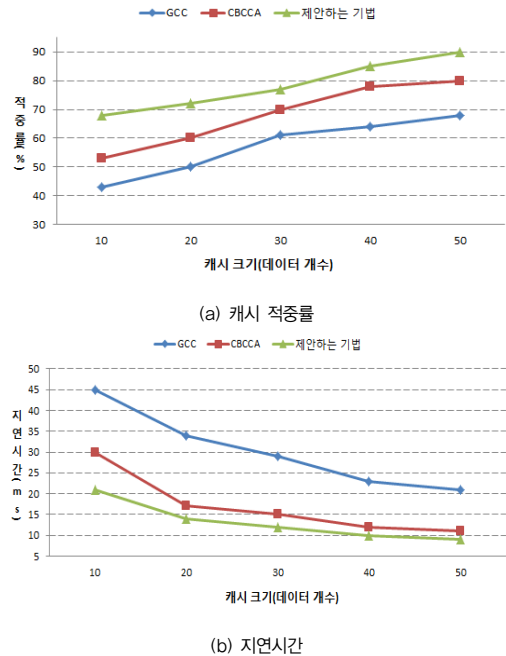


그림 10. 캐시 크기에 따른 성능 평가

[그림 11]은 피어 개수에 따른 캐시 적중률을 나타낸 것이다. 피어 개수를 50에서 400까지 변화시켜서 캐시 적중률과 지연 시간을 측정했다. 고정된 환경에서 피어

개수를 변화 시키는 것을 예를 들면 사람이 적은 외곽 지역과 변화가의 비교라고 볼 수 있다. 사람들이 적은 외곽 지역에는 클러스터를 형성하기 어렵기 때문에 기존 기법과 제안하는 기법은 비슷한 적중률 보인다. CBCCA와 제안하는 기법은 피어 개수가 증가함에 따라 클러스터를 형성하는 피어들이 많아지고 공유되는 양이 많아진다. 그러므로 피어수가 많아질수록 캐시 적중률의 상승폭이 크게 증가한다. 그에 비해 GCC는 낮은 상승폭을 보인다. 제안하는 기법이 GCC, CBCCA에 비해 캐시 적중률이 각각 약 21%, 9% 향상되는 성능을 나타내었다.

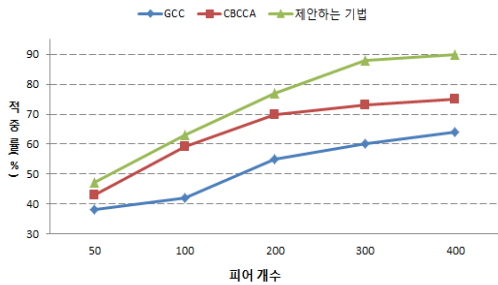


그림 11. 피어 개수에 따른 성능 평가

V. 결론 및 향후 연구

본 논문에서는 모바일 P2P 네트워크에서 피어의 연결성을 고려한 그룹 기반 캐시 공유 기법을 제안하였다. 제안하는 기법은 연결성이 오랫동안 유지될 수 있는 피어들을 하나의 클러스터로 형성하여 캐시를 공유한다. 클러스터 내부에 있는 피어들을 하나의 캐시로 사용함으로써 데이터의 중복을 감소시키고 캐시 사용 공간을 효율적으로 사용한다. 또한, 캐시 공간을 데이터 캐시와 임시 캐시 두 부분으로 분할하여 사용한다. 데이터 캐시는 일반적으로 캐시 데이터를 관리하고 교체 전략이 수행하는 공간이고 임시 캐시는 고정적으로 빈 공간을 만들어 놓아서 클러스터의 토폴로지가 변경될 때 인기 있는 데이터를 임시적으로 저장하여 데이터 교환 용도로 사용한다. 또한 캐시 데이터 교체 시, 빈 공

간을 버퍼처럼 사용하여 지연시간을 감소시킬 수 있다. 질의가 전달되는 통신비용을 감소시키기 위해 일반적인 기법에 1-홉 피어 캐시확인과 클러스터 헤더로 가는 경로에 있는 피어의 캐시를 확인 절차를 추가했다. 성능평가 결과 제안하는 기법이 기존 기법보다 우수한 성능을 나타내는 것을 확인했다. 향후 연구로는 제안하는 캐시공유 기법을 실질적인 모바일 P2P 네트워크에 적용할 예정이다.

참고 문헌

- [1] D. T. Ahmed and S. Shirmohammadi, "Design Issues of Peer-to-Peer Systems for Wireless Ad Hoc Networks," Proc. International Conference on Networking, p.26, 2007.
- [2] 북경수, 이현정, 박용훈, 유재수, "모바일 P2P 네트워크에서 피어의 이동성을 고려한 연속적인 k-최근접 질의 처리", 한국콘텐츠학회논문지, 제12권, 제8호, pp.47-58, 2012.
- [3] 북경수, 조미림, 유재수, "모바일 P2P 네트워크에서 효율적인 콘텐츠 검색을 위한 데이터 배포 기법", 한국콘텐츠학회논문지, 제12권, 제8호, p.37, p.46, 2012.
- [4] C. Y. Chow, H. V. Leong, and A. T. S. Chan, "Cache signatures for peer-to-peer cooperative caching in mobile environments," Proc. International Conference on Advanced Information Networking and Applications, pp.96-101, 2004.
- [5] C. Y. Chow, H. V. Leong, and A. T. S. Chan, "Distributed group based cooperative caching in a mobile broadcast environment," Proc. International conference on Mobile data management, pp.97-106, 2005.
- [6] Y. W. Ting and Y. K. Chang, "A novel cooperative caching scheme for wireless ad hoc networks: Group caching," Proc. International

Conference on Networking, Architecture and Storage, pp.62-68, 2007.

- [7] L. Fan, P. Cao, J. Almeida, and A. Z. Broder, "Summary cache: a scalable wide-area web cache sharing protocol," IEEE/ACM Transactions on Networking, Vol.8, No.3 pp.281-293, 2000.
- [8] S. Chen, P. B. Gibbons, M. Kozuch, V. Liaskovitis, A. Ailamaki, G. E. Blelloch, B. Falsafi, L. Fix, N. Hardavellas, T. C. Mowry, and C. Wilkerson, "Scheduling threads for constructive cache sharing on CMPs," Proc. Annual ACM symposium on Parallel algorithms and architectures, pp.105-115, 2007.
- [9] C. Chow, H. V. Leong, and A. T. S. Chan, "GroCoca: Group-based peer-to-peer cooperative caching in mobile environment," IEEE Journal on Selected Areas in Communications, Vol.25, No.1, pp.179-191, 2007.
- [10] M. F. Caetano and J. L. Bordim, "A cluster based collaborative cache approach for MANETs," Proc. International IEEE Conference on Networking and Computing, pp.104-111, 2010.
- [11] Preetha Theresa Joy and K. Poulouse Jacob, "A key based cache replacement policy for cooperative caching in mobile ad hoc networks," Proc. International Advance Computing Conference, pp.383-387, 2013.
- [12] P. V. Paul, D. Rajaguru, N. Saravanan, R. Baskaran, and P. Dhavachelvan, "Efficient service cache management in mobile P2P networks," Future Generation Computer Systems, Vol.29, No.6, pp.1505-1521, 2013.

저 자 소 개

김 재 구(Jaegu Kim)

준회원



- 2013년 2월 : 충북대학교 정보통신공학과(공학사)
- 2013년 3월 ~ 현재 : 충북대학교 정보통신공학부 석사과정

<관심분야> : 모바일 P2P 네트워크, 위치 기반 서비스, 클라우드 컴퓨팅, 데이터베이스 시스템 등

윤 수 용(Sooyong Yoon)

준회원



- 2013년 2월 : 충북대학교 정보통신공학과(공학사)
- 2013년 3월 ~ 현재 : 충북대학교 정보통신공학부 석사과정

<관심분야> : 데이터베이스 시스템, 모바일 P2P 네트워크, 위치 기반 서비스, 센서네트워크, 빅데이터 등

임 종 태(Jongtae Lim)

정회원



- 2009년 2월 : 충북대학교 정보통신공학과(공학사)
- 2011년 2월 : 충북대학교 정보통신공학과(공학석사)
- 2011년 3월 ~ 현재 : 충북대학교 정보통신공학과 박사과정

<관심분야> : 데이터베이스 시스템, 시공간 데이터베이스, 위치기반 서비스, 모바일 P2P 네트워크, 빅데이터 등

이 석 희(Seokhee Lee)

정회원

컴퓨터정보통신연구소 교수



- 1994년 2월 : 충북대학교 정보통신공학과(공학사)
- 1998년 2월 : 충북대학교 정보통신공학과(공학석사)
- 2001년 2월 : 충북대학교 정보통신공학과(공학박사)

- 2009년 3월 ~ 2010년 2월 : 캘리포니아주립대학교 방문교수

<관심분야> : 데이터베이스시스템, 빅데이터, 센서네트워크 및 RFID, 소셜 네트워크 서비스, 분산 객체컴퓨팅, 바이오인포매틱스 등

- 2000년 3월 ~ 현재 : 동아방송예술대학교 뉴미디어콘텐츠과 부교수

<관심분야> : 정보검색, 데이터베이스, 이동객체, SNS, 센서네트워크 등

북 경 수(Kyungsoo Bok)

정회원



- 1998년 2월 : 충북대학교 수학과(이학사)
- 2000년 2월 : 충북대학교 정보통신공학과(공학석사)
- 2005년 2월 : 충북대학교 정보통신공학과(공학박사)

- 2005년 3월 ~ 2008년 2월 : 한국과학기술원 전산학과 Postdoc

- 2008년 3월 ~ 2011년 2월 : (주)가인정보기술 연구소

- 2011년 3월 ~ 현재 : 충북대학교 정보통신공학과 초빙부교수

<관심분야> : 데이터베이스 시스템, 위치기반서비스, 모바일 P2P 네트워크, 소셜 네트워크 서비스, 빅데이터 등

유 재 수(Jaesoo Yoo)

종신회원



- 1989년 2월 : 전북대학교 컴퓨터공학과(공학사)
- 1991년 2월 : KAIST 전산학과(공학석사)
- 1995년 2월 : KAIST 전산학과(공학박사)

- 1995년 3월 ~ 1996년 8월 : 목포대학교 전산통계학과(전임강사)

- 1996년 8월 ~ 현재 : 충북대학교 정보통신공학부 및