

논문 2009-3-22

모바일 애드-혹 망에서 분산 캐시를 이용한 효율적인 P2P 서비스 방법

An Efficient P2P Service using Distributed Caches in MANETs

오선진*, 이영대**

Sun-Jin Oh, Young-Dae Lee

요 약 무선 모바일 애드 혹 망 기술과 P2P 서비스 기술의 급속한 발전과 더불어, 모바일 애드-혹 망에서의 P2P 서비스를 통합하려는 시도와 많은 통합 응용들의 개발이 활발히 이루어지고 있다. 하지만 모바일 애드-혹 망에서의 모바일 사용자들의 높은 이동성은 이들 사이에서의 안정적인 P2P 서비스를 구현하는데 많은 어려움을 낳는다. 본 논문에서 우리는 이러한 문제점을 해결하기 위해 모바일 환경에서 효율적인 파일 공유와 관리를 하고, 좋은 성능을 성취할 수 있도록 피어들 중간에 분산된 캐시를 두어 빈번하게 요청되는 파일에 대한 인기도를 고려한 파일 캐시를 이용한 효율적인 모바일 P2P 서비스 방법을 제안한다. 제안하는 방법은 분석적 모델을 이용하여 그 성능을 평가하고, 기존의 P2P 망에서의 DHT 기반 서비스 기법과 비교하였다.

Abstract With rapid growth of Mobile Ad-Hoc network(MANET) and P2P service technologies, many attempts for integration of MANET and P2P service and development of such applications are actively introduced recently. The implementation of stable P2P service, however, is very difficult challenge because of the high mobility of mobile users in MANET. In this paper, we propose an efficient mobile P2P service, which shares and manages multimedia data files efficiently in mobile environment, uses distributed caches to store files considering their popularities in order to achieve high performance. The performance of proposed P2P service is evaluated by an analytic model and compared with those of existing DHT based P2P service in peer-to-peer network.

Key Words : MANET, Mobile P2P Network, Distributed Cache

I. 서 론

모바일 애드 혹 망(MANETs)은 고정된 인프라가 없는 환경에서 무선 모바일 노드들로 구성된 망을 말한다. 지금까지의 모바일 애드-혹 망에 관한 연구는 이동성을 가진 모바일 호스트들 간에 높은 통신 효율성을 제공하기 위한 라우팅 프로토콜과 위치 서비스에 대한 연구가 주류를 이루고 있다. 하지만 최근에는 애드-혹 망 환경에

적합한 응용모델에 대한 관심이 높아지면서 UCC나 게임 등 P2P 기반 다양한 응용에 대한 연구가 활발히 진행되고 있다. Peer-to-Peer(P2P) 서비스는 중앙에 서버 없이 개인 간의 직접적인 연결을 통하여 디지털 자원을 효과적으로 공유하는 방식으로 기존의 클라이언트/서버 방식과 달리 관련 프로그램만으로 디지털 자원에 대한 손쉬운 공유와 교환이 가능하다. 하지만 애드-혹 망에서의 모바일 노드들은 높은 이동성을 가질 수 있으며, 이러한 노드들 간의 안정적인 P2P 서비스를 제공하는 것은 많은 도전을 요구한다. 분산 환경에서의 원활한 P2P 서비스를 위해 일반적으로 분산 해시 테이블(DHT)이 사용되는데

*정회원, 세명대학교 정보통신학부

**중신회원, (사)인터넷방송통신기술원

접수일자 2009.4.15, 수정완료 2009.06.05

이 방법에 기초한 현재 P2P 정보 공유 시스템은 수천의 사용자들 간의 폭넓은 자원과 수많은 정보교환의 요구를 만족시킬 수 있다. DHT의 예로는 Chord, Pastry, Tapestry, 그리고 CAN을 포함한다.^[1] 하지만 모바일 애드-혹 망에서의 P2P 서비스는 높은 이동성을 가진 노드들의 특성으로 인해 기존의 방법으로는 통신 단절이나 서비스 품질 저하 등 문제가 발생할 수 있다. 한편, 모바일 애드-혹 망에서 장치들은 인프라 구조 없이 통신 플랫폼으로 자기 조직화 할 수 있으며, 그러한 모바일 장치들은 DSDV, DSR, 그리고 AODV 등과 같은 라우팅 프로토콜을 사용하여 라우팅 정보를 구축할 수 있다. P2P망과 MANETs은 다소의 공통 특성을 가지며, MANETs 상에서 P2P 파일 공유를 적용하는 것은 매우 당연하다. P2P 망의 설계에서 오버레이 위상과 물리적 위상간의 잘 맞지 않는 것은 성능에 중대한 영향을 줄 수 있으며 이를 위상 미스매치라 한다.^[2]

본 연구에서 우리는 MANETs 상에서 P2P 기반 서비스의 효율적인 미디어 스트리밍에 초점을 맞춘다. 즉, 효율적인 모바일 위상을 적용시키기 위해 모바일 P2P 망에서 분산 캐시를 이용하여 빈번하게 요청되는 인기 있는 멀티미디어 스트림에 대한 파일 캐시를 통해 지역에서의 요청에 대한 가용성을 높이고, 보다 효율적인 파일 공유와 관리가 이루어지게 하는 협력적 캐싱 알고리즘을 제안한다. 우리의 목표는 피어들이 MANETs에서 요구하는 파일들을 효율적으로 관리하고, 검색하며, 스트리밍하는데 있다. 본 논문에서 제안한 P2P 서비스 기법은 분석적 모델을 이용하여 그 성능을 평가하고, 기존의 P2P 망에서의 DHT 기반 서비스 방법과 성능을 비교하였다.

본 논문의 구성은 다음과 같다. 2장에서는 모바일 P2P 망에서의 구조와 스트리밍에 대한 기존의 관련 연구를 살펴보고, 3장에서는 본 논문에서 제안한 모바일 P2P 망에서의 분산 캐시를 이용한 서비스 방법을 소개하고, 4장에서는 이와 관련된 구체적인 구현 알고리즘을 제안한다. 5장에서는 분석적 모델을 이용하여 제안한 기법의 성능을 평가하였고, 마지막으로 6장에서 향후 연구과제와 함께 결론을 맺는다.

II. 관련 연구

P2P 시스템에서 피어들은 오버레이 망에서 물리적 링

크에 의해 서로 연결되어진다. 여기서 각 논리적 링크는 물리적 링크에서 긴 경로에 대응될 수 있다. 오버레이 망과 물리적인 망간의 위상 미스매치는 오버레이 망을 효율적으로 구축하고 트래픽 오버헤드를 줄이기 위한 해결 방법으로 연구되었다. 초기에 오버레이 망은 하부 물리적인 망과 동일하게 구축되었다. 다수의 지름길은 오버레이 망에서 링크들의 범주에 따라 동적으로 생성되어진다. 그러한 지름길은 2개의 노드들을 오버레이 망에서 직접 연결되게 만들기 위하여 가상 링크를 생성하는데 사용된다.

Landmarking^{[3][4]}은 물리적 근접에 따라 DHT-기반 P2P 오버레이를 생성하기 위하여 사용되는 방법이다. Pastry에 기초한 RLM (Random Landmarking)은 각각이 오직 오버레이 ID인 무작위로 선택된 경계표시 (landmark) 키들의 집합을 선택한다. 물리적으로 가까운 이웃노드들은 오버레이 공간에서 서로 숫자상으로 가깝게 부여될 것이다.

S. Ratnasamy^[5]는 CAN (Content-Addressable Network)을 모바일 애드-혹 망에 적용하여 노드들이 빈번하게 결합하고 이탈하는 것을 허용하는 SAT-Match라 불리는 효율적인 방법을 제안하였다. 이 방법에서 모든 노드는 전체 공간에서 그것의 지역을 소유하는 동적 분할을 요구한다. 노드 A가 물리적으로 이웃하는 노드 B로 이동할 때, 그것은 각 노드가 부하 균등을 이루기 위하여 그 공간 절반에 대해 책임을 지게 될 그것의 오버레이 공간을 나누는 것을 노드 B에게 물어볼 것이다. 이 방법에서 오버레이 위상과 물리적 위상은 서로 거의 일치시킬 수 있다. 각 노드는 만일 그것이 주기적으로 새로운 지역으로 이동하는지 검사해야 한다.

I. Stoica^[6]는 모바일 애드-혹 망에 Chord를 제안하였다. 2개의 Chord 시스템을 하나의 큰 링으로 조합하기 위하여 또는 한 개의 Chord 시스템을 더 작은 시스템들로 분할하기 위하여 적절한 임계치가 생성된다. 여기에서 망은 더 작은 블록들로 분할되어지는 가상 2-차원 좌표 공간으로 가정한다.

III. 시스템 설계

1. 모바일 P2P 시스템 모델

그림 1은 기존 네트워크 응용 프로그램의 대표적인 구

조인 클라이언트/서버 구조와 P2P 방식의 구조를 보여준다. 그림에서 보는 바와 같이 클라이언트/서버 구조는 클라이언트와 서버 각각 그 기능과 역할이 명확히 구분되어 서버는 많은 클라이언트들에게 서비스를 제공하는 역할을 하고, 반면에 클라이언트는 서버로부터 제공되는 서비스를 제공받는 역할을 수행한다. 따라서 여기에서 클라이언트들 간에는 직접적인 데이터 교환이 이루어지지 않으며 반드시 서버를 통해서만 데이터 교환이 이루어질 수 있다. 데이터의 전달 방식은 먼저 클라이언트의 요청에 의해 서버가 응답하는 방식으로 이루어진다. 한편 P2P 서비스 구조에서는 망을 구성하고 있는 각 피어들이 서비스를 제공하는 서버의 역할을 하는 동시에 서비스를 이용하는 사용자의 역할도 같이 수행하는 응용프로그램 구조이다. 데이터의 흐름을 비교해 볼 때 클라이언트/서버 구조는 주로 서버에서 클라이언트로의 데이터의 흐름이 대부분을 차지하는 반면, P2P 구조는 피어들 서로 간에 데이터를 교환하기 때문에 어느 한 피어에 트래픽이 집중되거나 방향성을 가지지 않고 피어들 사이에 서로 간 자유로운 데이터 교환이 이루어진다.

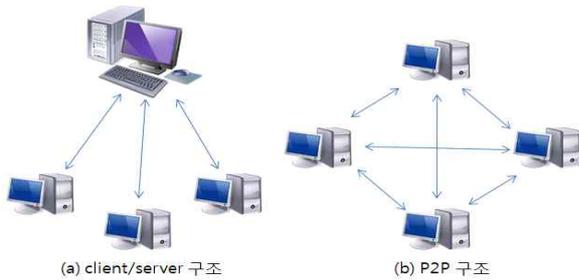


그림 1. 구조 비교
Fig. 1. Comparison of Structures

이 논문에서 우리는 위성 망처럼 다른 망과 인터페이스를 갖는 다소의 모바일 노드들이 있는 무선 모바일 애드-혹 망 환경을 고려한다. 모바일 노드들은 다른 노드들이 외부 망과 통신을 허용하기 위한 게이트웨이로 서비스 한다. 데이터를 제공하는 데이터 서버는 MANETs 내부 또는 외부에 존재할 수 있으며 각 모바일 노드들은 그 데이터 서버들로부터 데이터를 읽을 수 있다. 하지만 데이터의 소스만이 그 데이터를 기록하거나 갱신하는 특권을 갖는다. 이렇게 함으로써 데이터가 분산되더라도 데이터의 중복성 문제를 해결할 수 있다. 각 모바일 노드들은 자신의 로컬 저장장치에 데이터의 사본을 캐시할 수 있다. 우리는 모바일 노드 상의 캐시와 데이터 서버 저장

소 간의 데이터 일관성을 유지하고 로컬 캐시의 수를 제한하기 위하여 TTL 방법이 사용된다고 가정한다. TTL의 기본 개념은 각 데이터 아이템에 어떤 TTL 값이 할당되고, 그 캐싱 기간 동안 로컬 캐시의 데이터 아이템이 참조될 때마다 TTL 값을 재설정하며 그 TTL 값이 초과되지 않는 한 그 데이터 아이템은 유효한 것으로 고려되어진다. TTL 값이 만료되면 로컬 캐시에 저장되었던 데이터 아이템은 자동 삭제된다.

본 논문에서 고려하는 각 노드의 시스템 구조는 안정적이고 효율적인 P2P 서비스를 위하여 사용자의 요청이나 응용을 위한 인터페이스 계층을 두고, 데이터 아이템을 저장 관리하는 로컬 캐시 관리를 위한 미들웨어 계층이 있으며, 다른 모바일 노드들과 통신을 위한 하부 네트워크 스택을 사용한다. 그림 2는 본 논문에서 고려하는 시스템 모델의 구조를 보여준다.

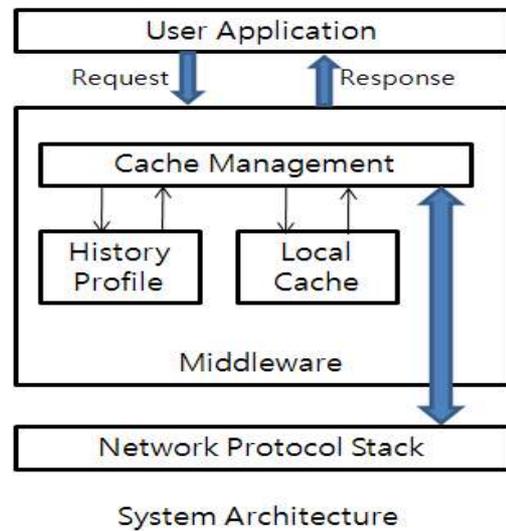


그림 2. 시스템 구조
Fig. 2. System Structure

2. 분산 캐시 기반 P2P 서비스

모바일 노드는 사용자의 응용으로부터 데이터 요청을 받고 캐시 결정 알고리즘을 사용하여 그 요청을 해결한다. 캐시 관리 방법은 어떤 모바일 장치의 국부 캐시에 무슨 데이터를 위치시킬 것인지 삭제할 것인지를 결정한다. 캐시 결정 알고리즘은 서비스의 안정성, 서비스 품질, 네트워크 대역폭의 최소 비용 등을 고려하여 어떤 데이터 요청에 대해 캐싱을 할 것인지를 결정한다.

캐시 결정은 3단계에 걸쳐 수행이 되는데 우선 이전에 수신한 데이터 요청들의 히스토리 프로파일을 관리하고,

그 프로파일에 기초하여 사용자 응용으로부터 데이터 아이템에 대한 요청이 들어오면 자신의 로컬 캐시에 해당하는 데이터 아이템이 존재하는지를 확인하여 존재하는 경우 로컬 캐시로부터 서비스를 한다. 자신의 로컬 캐시에 존재하지 않는 경우에는 유지하고 있던 히스토리 프로파일에 기반한 사용자 요청에 대한 가까운 데이터 아이템을 캐시하고 있는 모바일 노드를 결정하게 된다.

히스토리 프로파일을 사용하여 사용자 요청이 해결되지 않으면 바로 n-홉 거리의 이웃 노드들에게 해당하는 데이터 아이템이 존재하는지를 조회하기 위해 비용에 기초하여 적절한 플러딩 범위를 결정하는 적응적 플러딩 방법을 사용한다. 이 방법으로 요청주변의 최근접 캐시를 발견할 수 있는 가능성을 가질 뿐 아니라 플러딩을 통해 이웃에서 어떤 공지로 서비스 할 수 있고 전체 망을 클러스터로 효율적으로 세그먼트 할 수 있기 때문에 관리 하는 캐시 콘텐츠를 공유하고 협력할 수 있다.

이 두 가지 방법에 의해서도 사용자 요청이 해결되지 않으면 비로소 기존의 P2P 망에서의 분산 해시 테이블을 사용하여 데이터 아이템을 가지고 있는 데이터 서버를 찾아 데이터 아이템을 요청한다. 이때 스트리밍되는 데이터 아이템은 현재의 모바일 노드 주변에는 존재하지 않는 데이터이므로 그 데이터 아이템의 인기도를 조회하여 높은 인기도를 가진 경우 캐시할 것을 결정하고 자신의 로컬 캐시에 저장한 후 TTL 값을 부여한다. 이렇게 캐시된 데이터 아이템은 추후 사용자의 요청이 주변에서 있을 경우 우선적으로 스트리밍 서비스를 하게 되며 이때 마다 TTL 값을 재설정하여 유지 한다. TTL 값이 만료될 때까지 이 데이터 아이템에 대한 주변의 요청이 없는 경우에는 로컬 캐시로 부터 이 데이터를 삭제함으로써 분산 캐시의 숫자를 제어할 수 있다.

IV. 구현 알고리즘

본 논문에서 제안한 모바일 P2P 망에서 분산 캐시를 이용한 서비스 구현 알고리즘은 그림 3에서 보여 주는 바와 같다. 그림에서 보는바와 같이 클라이언트의 요청에 대한 처리는 4단계에 의해 최소의 비용으로 서비스 된다.

그림 4는 모바일 P2P 망의 예를 보여준다. 그림에서 DS는 데이터 서버를, C는 캐시노드를, R은 요청노드를 의미한다. 사용자로부터 요청이 들어오면 우선 자신의

로컬캐시에 해당 데이터 아이템이 있는지 확인하고 서비스한다. 로컬 캐시에 존재하지 않으면 히스토리 프로파일에 근거하여 이전에 접근한 캐시 노드를 질의하며, 질의 내용이 없으면 n-홉 거리의 이웃 노드에 RREQ를 통해 질의 요청을 하게 된다. 그림 5는 이렇게 하여 주변 캐시노드를 발견하여 연결된 상태를 보여준다. 주변 캐시노드 역시 존재하지 않으면 그림 6에서 보는바와 같이 비로소 DHT를 가진 데이터 서버에 요청하고 지정된 노드로부터 데이터 아이템을 받게 되며 이때 인기도에 따라 자신의 로컬 캐시에 저장한 후 자신을 이 후 서비스를 위한 캐시노드로 설정하게 된다.

```

//=====캐시 결정 알고리즘=====//

모바일 노드가 사용자 응용 요청 수신

if(exist in Local Cache) then
    Service(Stream)           // 로컬 저장소에서 서비스

else if(exist in History Profile) then
    Send(RREQ);               // 해당 노드에 요청
    Receive(RREP);            // 해당 노드로부터 응답
    Service(Stream);          // 최근접 서버로부터 서비스

else Adaptive-Flooding(RREQ) // 이웃 노드에 적응적 플러딩
    if(Receive(RREP)) then // 이웃 노드로부터 응답
        Calc_Cost(RREP);    // 노드 선정 비용 계산
        Send(RREQ);         // 이웃노드에 데이터 요청
        Receive(RREP);      // 이웃 노드로부터 응답
        connect(PeerID);    // 이웃 노드와 연결
        Service(Stream);    // 이웃 노드로부터 서비스
    else
        Search(DHT);        // 분산 해시테이블 검색
        Get(Server ID);     // 서버 ID 획득
        Send(RREQ);         // 서버에 데이터 요청
        Receive(RREP);      // 서버로부터 응답
        Calc_Pop(Stream);   // 데이터의 인기도 계산
        connect(PeerID);    // 서버와 연결
        Service(Stream);    // 서버로부터 서비스
        cash_store(Stream); // 스트림 저장, 캐시노드
        timer(TTL);         // TTL 타이머 작동
    endif
    
```

그림 3. 분산 캐시 기반 P2P 서비스 알고리즘
 Fig. 3. P2P Service Algorithm based on Distributed Cache

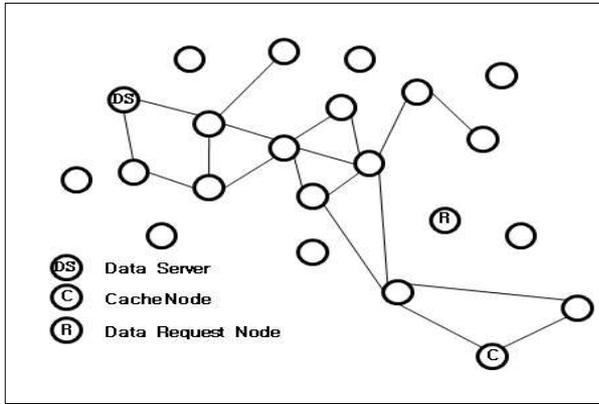


그림 4. 모바일 P2P 망 예
 Fig. 4. Example of Mobile P2P Network

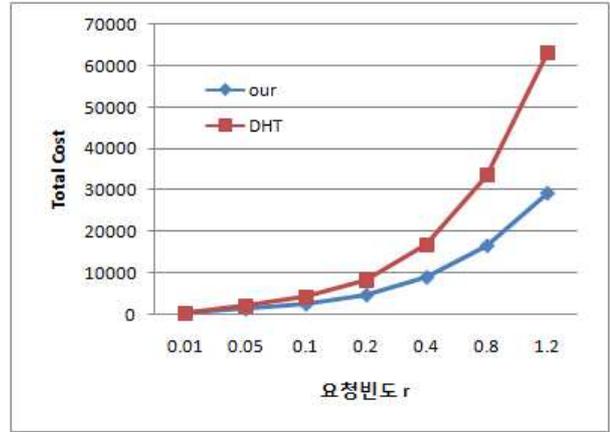


그림 7. 요청빈도에 대한 순수 서비스 비용
 Fig. 7. True Service Cost for Request Frequency r

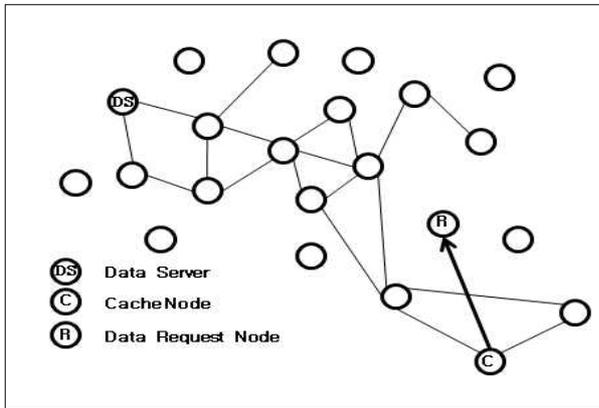


그림 5. 캐시노드로의 질의
 Fig. 5. Query to Cache Node

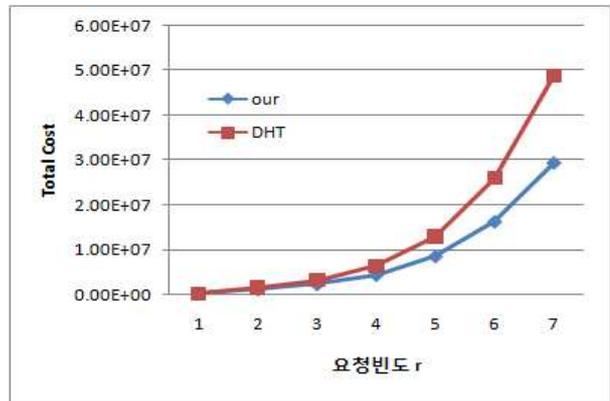


그림 8 요청빈도에 따른 총 서비스 비용
 Fig. 8. Total Service Cost for Request Frequency r

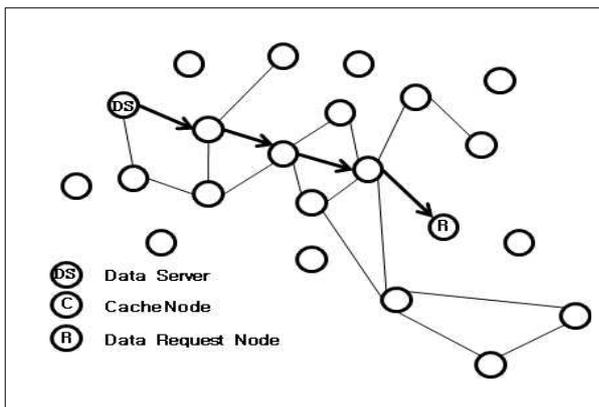


그림 6. 데이터 서버로부터 질의
 Fig. 6. Query to Data Server

V. 성능 평가

이 장에서는 본 논문에서 제안하는 분산 캐시 기반 모바일 P2P 서비스 알고리즘에 대한 성능을 분석적 모델을 이용하여 평가하고 이를 DHT-기반 P2P 서비스 기법과 비교하였다. 성능 평가는 일정 지역의 모바일 P2P 망에서 클라이언트들의 요청 빈도에 대한 P2P 서비스를 위한 총 소요 비용 항목으로 평가하였다.

성능 평가를 위해 지역 A 내의 P2P 망을 구성하는 노드의 수를 100개로 가정하였고, 이들 노드들 중 1개의 데이터 서버 노드가 존재하며, 수개의 캐시노드가 운영되고 있다고 가정한다. 본 논문에서 성능 평가를 위해 사용한 파라미터는 표 1과 같다.

표 1. 성능평가를 위한 매개변수
Table 1. Parameters for Performance Evaluation

Parameters	Values
지역 A 내의 노드 수 n	100 개
P_{LC}	0.05 - 0.07
P_{HP}	0.15 - 0.2
P_{Cache}	0.3 - 0.43
P_{DS}	0.3 - 0.5
t	1000

여기서 P_{LC} 은 클라이언트가 요청한 스트림에 대한 데이터 아이템이 로컬 캐시에 존재할 확률, P_{HP} 는 히스토리 프로파일의 근거하여 이전에 접근한 캐시노드에 존재할 확률, P_{Cache} 는 데이터 아이템이 n-홉 거리의 이웃 노드에 존재할 확률, 그리고 P_{DS} 는 DHT를 가진 데이터 서버에 의해 데이터 아이템을 찾게 될 확률을 각각 나타낸다. 또한 t 는 총 시뮬레이션 시간을 의미한다. 여기서 r 을 클라이언트 노드로부터의 요청 빈도를, Avg_{hop} 을 평균 hop 수를, 그리고 C_{stream} 을 비디오에 대한 스트리밍 비용이라 가정하면 모바일 P2P망에서 분산캐시를 기반으로 서비스하는 총 비용은 수식 (1)과 같다.

$$\begin{aligned}
 C_{Total} = & P_{LC} \times rt (C_{init} + C_{stream}) \\
 & + P_{HP} \times rt \times Avg_{hop} (C_{HP} + C_{stream}) \\
 & + P_{Cache} \times rt \times Avg_{hop} (C_{Cache} + C_{stream}) \\
 & + P_{DS} \times rt \times Avg_{hop} (C_{DS} + C_{stream})
 \end{aligned} \quad (1)$$

여기서 C_{init} , C_{HP} , C_{Cache} , C_{DS} 는 각각 초기 설정을 위한 메시지 교환 비용을 의미한다.

한편, DHT를 기반으로 하는 P2P 서비스 기법의 클라이언트의 요청에 대한 서비스 비용은 수식 (2)와 같다.

$$C_{DHT} = rt \times Avg_{hop} (C_{init} + C_{stream}) \quad (2)$$

이들 비용 모델을 기반으로 클라이언트 요청 빈도에 따른 P2P 망에서의 서비스 비용에 대한 성능 평가 결과는 다음의 그림과 같다. 그림 7은 클라이언트들의 스트림 요청 빈도에 따른 순수 스트리밍 처리 비용의 관계를 보여주고, 그림 8은 클라이언트들의 스트림 요청에 대한 처

리 비용과 미디어 스트리밍 비용을 합한 총 서비스 비용의 관계를 분석적 방법에 의해 성능 평가한 결과이다. 그림에서 보는 바와 같이 순수 비용이나 총 비용 모두 본 논문에서 제안하는 분산 캐시 기반 P2P 서비스 방법이 분산 해시 테이블 기반의 기존 P2P 서비스 방법에 비해 우수한 성능을 보임을 알 수 있다. 또한, 클라이언트들의 스트림 요청 빈도가 높아짐에 따라 그 격차는 뚜렷하게 차이가 남을 알 수 있다. 그 차이는 최대 53.7%까지 서비스 비용을 줄일 수 있음을 알 수 있다.

VI. 결 론

자원관리에 대한 P2P 파일 공유 시스템의 분산 특징에 기인하여, P2P 구조는 모바일 애드-혹 망에 자연스럽게 적용될 수 있다. 본 논문에서, 우리는 무선 모바일 애드-혹 망에서 P2P 기반 효율적으로 파일을 관리하고 검색하기 위한 모바일 P2P 망에서 분산캐시를 이용한 효율적인 P2P 서비스 방법을 제안하였다. 제안한 방법은 MANETs에서의 높은 이동성을 가진 모바일 노드들 간의 안정적인 P2P 서비스를 위해 데이터 서버와 피어들 사이에 분산 캐시를 두어 피어들의 데이터 접근성을 향상시키고, 피어들의 질의에 따라 운행되어야 하는 홉 수를 줄일 수 있다. 또한 질의를 처리하는 과정을 4단계로 분리하여 보다 효율적인 미디어 스트리밍이 이루어지도록 하였고 피어들 사이에 분산 캐시의 수를 제한하기 위해 TTL을 채택하였다. 분석적 방법에 의한 성능평가 결과 기존의 DHT 기반 P2P 서비스 방법 보다 제안하는 P2P 서비스 방법이 우수함을 알 수 있다. 특히 클라이언트들의 요청 빈도가 높을수록 더 우수한 성능을 보임을 알 수 있다. 향후 연구 과제는 제안한 알고리즘을 모의실험으로 성능평가 하는 것이다.

참 고 문 헌

- [1] L. B. Oliveira, I. G. Siquenira, and A. A. Loureiro, "Evaluation of Ad-hoc Routing Protocols under a Peer-to-Peer Application," IEEE WCNC 2003, March 2003.
- [2] G. Ding and B. Bhargava, "Peer-to-Peer File Sharing over Mobile Ad-hoc Networks,"

- International Workshop on Mobile Peer to Peer Computing 2004 (IWMP2P'04).
- [3] T. Zahn, J. Schiller, "Random landmarking in mobile, topology-aware peer-to-peer networks," In proceedings of 10th IEEE International Workshop on Future Trends of Distributed Computing Systems, (FTDCS 2004), pp. 319-324, 26-28 May 2004
- [4] T. Zahn, J. Schiller, "MADPastry: A DHT Substrate for Practicably Sized MANETs," In proceedings of the 5th workshop on Applications and Services in Wireless Networks (ASWN2005), Paris, France, June 2005.
- [5] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Schenker, "A Scalable Content-Addressable Network," In proceedings of SIGCOMM, August 2001.
- [6] I. Stoica, R. Morris, D. Karger, M. Kaashoek, and H. Balakrishnan, "Chord: A Scalable Peer-to-Peer Lookup Service for Internet Applications," In proceedings of SIGCOMM, 2001.

저자 소개

오 선 진(정회원)
제 6권 2호 참조

이 영 대(중신회원)
제9권2호참조