

최소의 Forwarding Peer 선택을 통한 애드 혹 네트워크에서의 효율적 콘텐츠 분배 방법

강승석[†]

Efficient Content Sharing using the Selection of Minimum Forwarding Peers in an Ad Hoc Network

Seung-Seok Kang

ABSTRACT

Recent portable devices are so versatile that they have multiple communication channels and play several multi-media formats. Especially, many services are under development for users who connect Internet or nearby devices via WWAN (Wireless Wide Area Network) and/or WLAN (Wireless LAN). In case of paying the telecommunication cost proportional to the amount of data downloaded, it is necessary to reduce the cost by constructing a special ad hoc network in which each participating peer downloads a specific portion of the want-to-be-shared content over the payable WWAN channel and exchanges the remaining portion with other peers using the cost-free WLAN channel. If all peers participate in forwarding packets, some transmissions are redundant which results in the unnecessary consumption of bandwidth as well as the delayed content distribution time. In order to reduce the redundant transmission, this paper proposes both the excluding method which discourages some peers not to forward redundant packets, and the minimum cover set method in which only the minimum number of peers are in charge of forwarding packets. These two methods obviate redundant packet forwarding, and result in reduction of content distribution time by up to around 29%.

Key words : Content distribution, Ad hoc network, Benefit value, Minimum cover set, Peer excluding

요 약

최근의 휴대용 기기들은 통신과 멀티미디어 측면에서 다양한 기능들을 장착하고 있으며 특히 WWAN(Wireless WAN)과 WLAN(Wireless LAN) 등 두 개 이상의 채널을 이용하여 인터넷 혹은 주변기기와 접속할 수 있도록 다양한 서비스가 개발되고 있다. WWAN 채널이 내려 받은 자료의 양과 비례하여 요금을 부과하는 경우 해당 통신비용을 절감하기 위하여 peer라 하는 무선기들이 서로 공유하고자 하는 콘텐츠를 인터넷에서 내려 받을 때, 각 peer는 일부의 콘텐츠만 비용이 부과되는 WWAN을 통해 내려 받고 나머지를 비용이 부과되지 않는 WLAN을 통해 peer들이 애드 혹 네트워크를 구성하여 서로 교환함으로써 비용과 시간을 절감할 수 있다. 교환 과정에서 모든 peer가 forwarding에 참여하면 여러 peer가 동일한 패킷을 중복 전송하게 되는 경우가 발생된다. 그러나 이웃한 peer의 패킷 forwarding으로 인하여 특정 peer는 더 이상의 forwarding이 필요 없는 경우가 발생하는데, 본 논문에서는 forwarding이 필요 없는 peer를 선별하는 excluding 방식과 최소한의 peer로 forwarding을 수행하는 minimum cover set 방식등 두 가지 해결방식을 소개한다. 소개된 두 가지 방식을 이용하면 모든 peer가 forwarding 하는 방식에 비해 분배완료 시간을 더욱 단축할 수 있으며, 모의실험을 통해 최대 29% 정도 분배완료 시간이 단축되었다.

주요어 : 콘텐츠 분배, MANET, Benefit value, Minimum cover set, Peer excluding, Minimum forwarding

* 본 논문은 2009학년도 서울여자대학교 교내학술특별연구비의 지원을 받았음.

2009년 10월 1일 접수, 2009년 12월 9일 채택

[†] 서울여자대학교 정보미디어대학 컴퓨터학부

주 저 자 : 강승석

교신저자 : 강승석

E-mail; msukang@swu.ac.kr

1. 서 론

IT 환경이 계속 진보하고 사용자의 요구사항도 고급화됨에 따라 최근에 출시되는 노트북, PDA, Smart Phone, PMP 등은 다양한 통신기능과 멀티미디어 재생기능 등 많은 기능이 경쟁적으로 추가되며 해당 기능을 이용한 다양한 서비스가 개발되고 있다. 예를 들면, 근거리에서 있는 친구와 애드 혹 네트워크를 구성하여 다중 사용자 게임을 하거나 자료를 교환하기도 하고, 인터넷에 접속하여 흥미로운 음악, 영화 클립, 게임 등의 콘텐츠를 내려 받아 개인 혹은 주변의 사용자와 콘텐츠를 공유하기도 한다. 다양한 서비스를 제공하기 위해 많은 통신 기기들은 두 종류 이상의 네트워크를 접속할 수 있는 환경을 제공하고 있으며 이를 이용한 서비스가 개발되고 있다(Xiao 등, 2005; Kang 등, 2005). 접속 방식으로는 크게 802.11 과 블루투스 등을 이용한 Wireless LAN(WLAN)과 3G 등의 Wireless WAN(WWAN)이 있다.

WLAN은 주변에 access point가 있을 경우 이를 통해 인터넷을 접속하고 일반적으로 통신비용을 지불하지 않는 반면 특정 지역에 통신이 국한되는 등 사용지역이 제한적이다. 반면, WWAN은 광범위한 지역에서의 접속은 가능하지만 WLAN에 비해 속도가 느리며 비용을 지불한다. WWAN을 이용해 인터넷을 접속할 때 내려 받는 자료의 양에 비례하여 비용이 부과되는 경우 해당 비용을 줄이는 방안이 제안되었다(강승석, 2006). 예를 들어, 선생님과 학생들이 야외수업을 하는 경우 인터넷에 저장된 수업 자료를 내려 받거나, 친구들끼리 관심 있는 음악, 영화, 게임을 내려 받아 같이 즐긴다거나, 경기장에서 자신의 팀 성적과 선수의 정보를 내려 받는 것과 같은 상황이 있을 수 있다. 이 때 개별 사용자가 WWAN을 이용해 인터넷을 접속하면 내려 받는 자료의 양에 비례하여 비용이 부과된다. 이 비용을 줄이기 위해 여러 사용자가 하나의 콘텐츠를 공유하고자 할 때 휴대용 기기가 WWAN과 WLAN 모두 접근이 가능한 경우 각 기기는 인터넷에 저장된 콘텐츠의 일부만 WWAN을 통해 내려 받고 나머지 콘텐츠는 주변의 다른 사용자로부터 WLAN을 이용한 애드 혹 네트워크를 이용해 서로 교환하면 통신비용을 줄일 수 있다. Wimax나 Wibro등을 이용한 WWAN은 전 세계적으로 아직 개발 중이거나 서비스 범위가 제한적인 반면 3G 등을 이용한 서비스는 전 세계에서 보편적으로 사용되며 내려 받는 자료의 양에 비례해 사용금액이 정해진다. 이 경우 비용이 지불되는 WWAN 채널의 사용을 최소화하고 비용이 부과되지 않으면서 고속인 WLAN 채널

을 사용하여 비용과 시간을 절감할 수 있다. 예를 들어, N명의 사용자가 있을 때 각 사용자는 내려 받고자 하는 파일을 1/N만 유료인 WWAN 채널을 통해 내려 받고 나머지(N-1)/N 부분은 다른 사용자를 통해 무료인 WLAN 채널을 이용해 서로 교환하여 전체 파일을 재구성할 수 있다.

본 논문은(강승석, 2006)에서 소개된 전체적인 메커니즘 중에서 각 무선기기가 내려 받은 일부의 자료를 애드 혹 네트워크에 참여한 다른 무선기기와 공유하는 분배과정을 효율적으로 수행하는 방법에 중점을 둔다. 세부적으로 각 무선기기를 peer라 하면, 각 peer는 전체 파일 중에서 지정된 부분을 WWAN 채널을 통해 내려 받은 상태에서 각자 지정된 부분을 내려 받은 다른 peer들과 WLAN 채널을 이용해 forwarding을 통해 분배과정을 수행한다. 기존에 소개된 연구는 모든 peer가 forwarding에 참여하여 콘텐츠를 분배하는 방식이었으나 이 경우 불필요하게 중복된 패킷전송이 발생하여 분배완료 시간이 늦어진다.

본 논문은 중복된 패킷 전송을 배제하여 분배완료 시간을 단축하기 위한 두 가지 방식을 제안한다. 첫 번째 방식은 자신과 이웃한 한 peer가 패킷을 forwarding할 때 자신의 모든 이웃에게 패킷이 전송될 수 있다면 해당 peer는 패킷을 forwarding하지 않도록 특정 peer를 배제시키는 excluding 방식이다. 두 번째 방식은 참여한 모든 peer에게 전송할 수 있는 최소한의 백본 peer 집합을 선택하여 해당 집합에 있는 peer들만 forwarding을 수행하도록 하는 MCS(minimum cover set) 방식이다. 특정 peer를 배제하는 방식과 특정 peer만을 선택하는 방식 중에서 후자의 방식이 좀 더 빠른 분배완료 시간을 보여주었다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구를 간략히 소개하고, 3장에서는 benefit 값을 이용한 기존의 방식과 excluding 및 MCS 방식을 소개하며, 4장에서는 두 방법을 사용하는 경우 다양한 조건을 이용해 분배완료 시간을 측정하는 모의실험 결과를 보이며 5장에서는 결론을 맺는다.

2. 관련연구

기존에 소개된 방식(강승석, 2006)은 불필요한 전송을 줄이기 위해 내려 받은 콘텐츠를 패킷 크기와 같은 세그먼트로 나누어 benefit 값을 매 세그먼트마다 유지한다. 모든 peer는 각 세그먼트별로 benefit 값을 저장하고 있는데 이 정수 값은 해당 세그먼트를 가지고 있지 않은 직접 연결된 이웃 peer의 수를 나타낸다. 그 값이 0인 경우 연

결된 모든 이웃 peer 들이 해당 세그먼트를 가지고 있다는 의미이므로 해당 세그먼트를 forward하지 않는다. 본 논문에서는 기존 방식에 추가하여 특정 peer가 저장하고 있는 세그먼트의 benefit값이 0이 아니라도 forwarding 과정이 필요 없는 노드인 경우 패킷을 forwarding하지 않도록 배제하거나, forwarding이 필요한 최소한의 peer를 지정하여 해당 peer만 forwarding함으로 분배완료 시간을 추가로 단축할 수 있도록 하였다. 센서 네트워크에서 이기적인 노드들을 관리하기 위해 크레딧을 사용하는 Sprite(Zhong 등, 2003)는 자신의 패킷을 다른 노드에게 전송하고자 할 경우 자신의 크레딧을 사용해야 하며 크레딧을 얻기 위해서는 다른 노드들의 패킷을 전송해 주어야 한다. 크레딧을 관리하는 CCS(Credit Clearance Service) 서버는 인터넷상에 존재하고 각 노드들로부터 크레딧 관련 정보를 받는다. 각 노드는 CCS를 접근하기 위한 채널과 주변 노드와의 패킷 교환을 위한 채널을 유지한다. Karbhari 등(2003)은 여러 송신자와 하나의 수신자로 이루어진 통신상에서의 fairness에 관한 연구를 진행하였다. SplitStream(Castro 등, 2003)은 하나의 송신자가 하나의 파일을 k개의 stripe로 나눈 후 개별 stripe를 k개의 multicast tree를 이용하여 여러 수신자에게 전송한다. 여러 경로로 패킷을 전송함으로 참여한 여러 peer들의 forwarding 부담을 경감시킬 수 있다. BitTorrent(Bittorrent)는 한 사용자가 URL로 지정한 파일을 특정 송신 서버로부터 내려 받을 때, 파일의 일부를 내려 받는 도중 다른 사용자가 같은 송신자로부터 같은 파일을 내려 받고자 하는 경우 해당 송신자의 부담을 줄이기 위해 해당 파일의 일부를 이미 내려 받은 사용자도 특정 부분을 송신자와 같이 upload하여 송신자의 부담을 줄인다. 최소의 forwarding peer 집합을 계산하는 Set Cover 문제는 NP-complete이지만(Evan, 1979) 다양한 heuristic 방법을 사용하여 최적에 가까운 유사 해법들이 사용되고 있으며 본 논문에서 제시한 MCS도 greedy 알고리즘을 적용한 유사해법이다.

3. 콘텐츠 분배 방법

본 장에서는 다수의 peer가 인터넷에 저장된 하나의 파일을 공유하고자 할 때, 각 peer가 미리 지정된 파일의 일부분을 WWAN을 이용해 내려 받은 상태에서 자신이 내려 받은 부분과 다른 peer가 내려 받은 부분을 WLAN 채널을 이용해 애드 혹 네트워크를 구성하여 서로 교환하는 방법을 소개한다. Per-peer 방식은 benefit 값을 이용해

패킷의 전송 여부를 결정하기 때문에 per-packet 방식보다 우수한 성능을 보여 준다(강승석, 2006). per-peer 방식은 모든 peer가 자신이 저장한 패킷 중에서 주변의 peer가 저장하지 않은 세그먼트가 있을 경우 패킷을 forwarding 하는 반면, 본 논문은 forwarding 할 필요가 없는 peer를 미리 선정해 해당 peer는 전송 순서가 되어도 forwarding 하지 않도록 하는 excluding 방법과, 이와 반대로 forwarding을 수행하는 peer를 최소한으로 지정하여 해당 peer들만 forwarding하도록 하는 Minimum Cover Set(MCS) 방식을 소개한다.

3.1 Per-Peer 방식

논문(강승석, 2006)에서는 여러 peer가 어떻게 애드 혹 네트워크를 구성하고 콘텐츠를 per-peer 방식으로 분배하는지에 대해 자세히 기술하였다. 본 장에서는 내려 받은 일부의 자료를 어떻게 다른 peer와 애드 혹 네트워크에서 서로 교환하는지에 대한 방법을 간략히 요약한다.

per-peer 방식은 어떤 peer가 공유하고자 하는 파일의 어떤 부분을 내려 받아야 하는지 인터넷 상의 서버를 통해 지정받게 되고, 각 peer는 지정된 부분을 WWAN 채널을 통해 내려 받는다. 또한 서버는 전체 애드 혹 네트워크 토폴로지를 파악하여 어떤 순서로 이웃 peer에게 자신이 내려 받은 세그먼트를 전송할 지 결정하는 스케줄링을 수행한다. 각 peer는 자신의 차례가 되면 benefit 값에 따라 보낼 세그먼트에 대한 forwarding 여부를 결정한다. 만약 이런 스케줄링을 하지 않고 peer가 임의로 자신의 콘텐츠를 전송할 경우 broadcast storm problem과 같은 문제가 발생하여 수많은 패킷 충돌이 발생되고 적절한 콘텐츠 교환 작업을 수행할 수 없게 된다.

그림 1은 7개의 peer로 구성된 애드 혹 네트워크를 보여주며 per-peer방식이 어떻게 작동되는지 본 그림을 통해 설명한다. 인터넷에 위치한 서버는 각 peer가 전송한

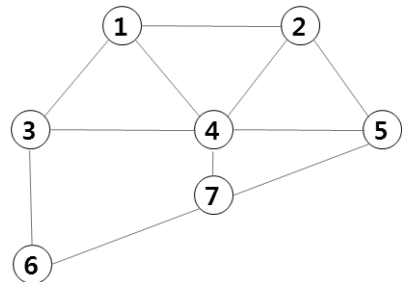


그림 1. 애드 혹 네트워크 토폴로지
Fig. 1. Ad hoc network topology

이웃 peer의 정보들을 바탕으로 전체 애드 혹 네트워크 토폴로지를 구성할 수 있으며, 이를 통해 이웃 peer의 수가 가장 많은 peer를 선택하고, 이전에 peer와 연결된 peer 중에서 이웃이 많은 peer 순으로 반복 선택하여 모든 peer의 전송 순서를 결정한다. 그림 1에서 서버가 결정할 수 있는 전송 순서로 peer 4를 먼저 선택하고 peer 7, peer 5, peer 2, peer 1, peer 3, peer 6 순으로 선택할 수 있으며 생성한 전송순서를 각 peer에게 알려준다. 서버는 위의 순서 이외에 다른 전송순서를 생성할 수도 있다. 만약 peer 6이 전송을 완료하면 다시 peer 4가 처음부터 전송을 반복하는 방식으로 작동된다.

공유하고자 하는 파일이 인터넷에 저장되어 있고, 그 크기가 패킷 7개의 크기와 같다고 간단히 가정해보면 각 peer는 패킷 하나 크기의 세그먼트를 WWAN 채널을 통해 내려 받는다. 각 peer는 세그먼트 별로 benefit 값을 지정하는데 이 값은 해당 세그먼트를 가지지 않은 이웃한 peer의 수로 정의된다. 예를 들어 peer 1, 2, 6이 저장한 세그먼트의 benefit 값은 2가 되고, peer 3, 5, 7이 저장한 값은 3이며, peer 4의 값은 5가 된다.

peer 4는 세그먼트(S4:V5)를 가지고 있다. 이 표현식에서 S는 세그먼트를 의미하고 따라오는 숫자는 세그먼트 번호를 의미한다. 즉, 각 peer는 자신의 ID와 같은 번호의 세그먼트를 WWAN 채널을 통해 내려 받았다고 가정한다. 표현식의 V는 benefit 값으로 해당 benefit 값만큼의 이웃한 peer들이 해당 세그먼트를 가지고 있지 않다는 의미이다. peer 4가 저장한 세그먼트의 benefit 값이 0이 아니기 때문에 peer 4가 전송을 완료하고 나면 peer 4의 세그먼트를 모든 peer가 소유하게 되므로 peer 4는 세그먼트(S4:V0)를 가진다.

이제 peer 7은 세그먼트(S4:V1, S7:V3)를 가진다. peer 4의 전송이 완료되어 peer 7의 전송차례가 되면 peer 7은 peer 4로부터 전송 받은 세그먼트 S4와 WWAN 채널을 통해 내려 받은 세그먼트 S7 등 두 개의 세그먼트를 가지게 된다. peer 4의 S4 전송 후에도 peer 6은 S4를 수신하지 못했으므로 peer 7이 소유한 S4에 대한 benefit 값은 1이 된다. 또한 S7의 경우 이웃한 어떤 노드도 해당 세그먼트를 가지고 있지 않기 때문에 benefit 값은 3이 된다. 이런 정보는 애드 혹 네트워크의 이웃 2-hop 정보를 이용해 세그먼트를 수신하는 시점에 계산할 수 있다(Lim 등, 2000; Peng 등, 2001; Williams 등, 2002). peer 7은 두 개의 세그먼트에 해당하는 benefit 값이 모두 0보다 크므로 해당 세그먼트를 모두 전송한다. peer 7의 전송을 완료하면 지정된 전송순서에 따라 peer

5가 전송을 수행한다. peer 5는 세 개의 세그먼트(S4:V0, S5:V3, S7:V1)를 가진다. S4는 peer 4를 통해 전달받았고 S5는 WWAN을 통해 내려 받았으며, S7은 peer 7을 통해 전달 받았다. 이 중에서 S5의 benefit 값은 3이므로 forwarding이 필요하며, S7은 peer 2로의 전달을 위해 benefit 값이 1이라 forwarding을 해야 하지만, S4의 benefit 값은 0이라 forwarding이 불필요하다. 그 이유는 peer 4의 S4 전송으로 peer 5와 이웃한 모든 peer가 S4를 가지고 있기 때문이다. 결국 peer 5는 S5와 S7만 forwarding 하면 된다. 이와 같은 방식으로 모든 peer가 자신의 전송 순서가 되면 해당 세그먼트의 benefit 값에 따라 세그먼트를 forwarding 하게 되고 결국 참여한 모든 peer가 전체 세그먼트를 가지게 되어 해당 콘텐츠를 공유하게 된다.

3.2 Excluding 방식

3.1 절은 2-hop 정보를 이용한 per-peer 방식으로 모든 peer가 forwarding에는 참여하되 benefit 값이 0인 경우 해당 세그먼트를 전송하지 않도록 하여 불필요한 패킷의 forwarding수를 줄이는 방법을 소개하였다. 본 절은 이에 추가하여 각 peer가 forwarding 그룹인지 아닌지 판단하여 forwarding 하지 않아도 되는 peer를 선별하여 이 peer들은 자신의 전송 순서가 되어 benefit 값이 0보다 큰 세그먼트가 있더라도 forwarding 을 하지 않도록 excluding peer group을 지정하는 방법을 소개한다. 이를 통해 모든 peer가 forwarding에 참여하는 방식보다 전체적인 분배 완료 시간을 감소시킬 수 있다.

Excluding 방식을 이용해 패킷 forwarding에서 제외되는 peer를 찾는 알고리즘은 다음과 같다. 각 peer는 2-hop 정보를 이용하여 아래의 알고리즘을 수행한다.

- 단계 1. 자신을 p라 할 때, 각 peer는 자신의 이웃 peer (1-hop) 집합 $N(p)$ 와 각 이웃들의 이웃 peer (2-hop) 집합 $N(q)$ 를 계산한다. 이때 $q \in N(p)$ 이다. 본 연산은 애드 혹 네트워크에서 일반적으로 수행할 수 있는 연산이다.
- 단계 2. 다음 조건을 만족하는 peer q가 있는지 검사한다. $N(p) \subseteq N(q) \cup \{q\}$, $q \in N(p)$
- 단계 3. 단계 2의 조건을 만족하는 q가 하나 이상 존재하면 해당 peer p는 forwarding 그룹에서 제외된다.

제외된 peer는 비록 자신의 전송순서가 되었을 때 WWAN 채널을 통해 내려 받은 패킷 이외에 애드 혹 채

널을 통해 수신한 어떤 패킷도 forwarding을 하지 않는다. 이를 위해 excluding 그룹에 포함된 peer들은 WWAN을 통해 내려 받은 세그먼트만 benefit 값이 이웃한 peer의 수에 해당하는 값으로 설정되고, 애드 혹 채널을 통해 수신된 모든 패킷의 benefit 값은 0으로 설정한다. 그림 1에서 Excluding 방법을 통해 제외되는 peer 들은 {1, 2, 5}이며 나머지 모든 peer들은 forwarding을 수행한다. 결국 모든 peer가 forwarding에 참여하는 방식은 peer 5의 전송 순서일 때, S5와 S7을 전송하지만, excluding 방식에서는 peer 5가 S7을 forward하지 않으므로 세그먼트 교환 시간이 단축되어 전체적인 세그먼트 분배완료 시간이 감소된다.

3.3 MCS(Minimum Cover Set) 방식

3.2 절에서 소개한 excluding 방식은 참여한 모든 peer 중에서 일부의 peer는 자신이 해당 패킷을 전송하지 않더라도 자신과 직접 이웃한 하나 이상의 peer가 자신의 모든 이웃에게 해당 패킷을 전송할 수 있다면 자신은 전송을 하지 않도록 배제시킴으로 전체적인 전송시간을 감소시킬 수 있었다. 본 excluding 방식은 애드 혹 네트워크에서 2-hop의 정보를 이용해 각 peer가 분산 알고리즘을 적용하여 연산이 가능하다. 이와는 달리, MCS 방식은 애드 혹 네트워크에 참여한 전체 peer와 연결된 최소한의 백본 peer 집합을 찾아서 해당 집합에 있는 peer만 forwarding을 수행할 수 있도록 하는 방식이다. 최소한의 peer가 포함된 집합을 찾는 문제는 NP-Complete 이므로 본 절에서는 이에 근접한 greedy 방식을 이용하여 작업을 수행하는 최소 peer 집합을 찾는 greedy 알고리즘을 소개한다.

- 단계 1. n개의 peer가 있을 때 각 peer를 $p_i (1 \leq i \leq n)$ 라 하고 $U = P = \{p_1, p_2, \dots, p_n\}$, $C = \emptyset$, $F = \emptyset$ 으로 설정한다. 이 때, C는 현재까지 선택된 백본 peer에 연결된 peer들의 집합을 의미하고 F는 백본 peer에 포함된 forwarding peer의 집합이다.
- 단계 2. $p_k \in P$ 라 하고 $N(p_k)$ 를 p_k 에 이웃한 peer들의 집합이라 할 때, $N(p_k) \subseteq P - C$ 이면서 $N(p_k)$ 의 peer 수가 최대인 $k (1 \leq k \leq n)$ 값을 찾는다.
- 단계 3. 연산 $P = P - \{p_k\}$, $F = F \cup \{p_k\}$, $C = C \cup N(p_k) \cup \{p_k\}$ 를 수행한다.
- 단계 4. 만약 $C \equiv U$ 이면 forwarding 그룹인 MCS는 F 이고 알고리즘은 종료된다.
- 단계 5. 그렇지 않으면 단계 2부터 다시 반복한다.

각 peer는 자신의 이웃한 peer들에 대한 정보를 인터넷에 위치한 관리자 서버에게 전달하면 관리자 서버는 이를 마스터 서버에게 전달하고, 마스터 서버는 개별 peer의 정보를 종합하여 MCS 알고리즘을 수행 후 각 peer에게 그 결과를 알려줌으로 각 peer는 자신이 forwarding 그룹에 속해있는지를 알 수 있다. 자신이 forwarding 그룹에 속해 있는 경우, 자신이 저장한 세그먼트의 benefit 값에 따라 forwarding을 수행한다. MCS 알고리즘을 그림 1의 토폴로지에 적용할 경우 forwarding 그룹은 {4, 7} 혹은 {4, 3}이 되며, 3.2 절과 비교할 경우 excluding 그룹은 {1, 2, 3, 5, 6}이 된다. 이렇게 되면 excluding 방식의 경우 peer 3과 peer 6은 benefit 값이 1 이상인 세그먼트에 대해 forwarding을 수행하지만, MCS 방식의 경우 peer 3과 peer 6도 forwarding을 수행하지 않으므로 세그먼트 분배완료 시간은 더욱 감소된다.

4. 시뮬레이션 결과

본 시뮬레이션은 모든 peer들이 forwarding에 참여하거나, forwarding 하지 않아도 되는 peer를 제외시킨 후 각 peer가 내려 받은 세그먼트들을 애드 혹 네트워크를 통해 서로 교환하는 excluding 방식 및 MCS 방식의 성능을 다양한 모의실험을 통해 비교하였다.

기존의 알고리즘과 본 논문에서 제안한 알고리즘의 성능 비교를 위해 ns2(ns2) 시뮬레이터를 이용해 모의실험을 수행하였다. 시뮬레이션 모델은 기본적으로 peer에 해당하는 모바일 노드들과 노드들이 무선통신을 위해 공유하는 2 Mbps로 최대 250 미터까지 전송되는 802.11 MAC 프로토콜 및 노드들 사이에 교환되는 패킷으로 구성되어 있다. 패킷의 크기는 내려 받은 세그먼트의 크기와 동일하게 설정하였다. 모든 통신은 broadcast 방식만 사용되므로 노드들 사이에 TCP나 UDP 등의 전송 프로토콜은 사용되지 않는다. 모의실험을 위하여 Excluding 방식을 수행하는 에이전트와 MCS 알고리즘을 수행하는 에이전트를 참여한 노드에서 작동할 수 있도록 설정하였다. 노드 사이의 통신은 ns2에서 기본적으로 사용하는 전송방식을 이용하며 송신 및 수신된 패킷은 해당 알고리즘을 수행하는 에이전트에 의해 처리하도록 하였다.

시뮬레이션에 참여한 peer는 최소 5에서 최대 15로 설정하였으며, 특별한 언급이 없는 경우 모의실험에서는 10개의 peer를 사용하였다. 각 peer가 세그먼트를 전송하기 위해 512, 1024, 2048 byte 세 가지 크기의 패킷을 사용

하였으며, 특별한 언급이 없는 경우 512 byte 크기의 패킷을 기본으로 사용하였다. 본 논문에서는 패킷의 크기와 세그먼트의 크기를 같은 것으로 처리하였다. 각 peer가 공유를 목적으로 인터넷에서 내려 받은 파일의 크기는 작은 경우 128 KByte에서 최대 1024 KByte 크기로 설정하였다.

사용된 모의실험 영역은 작은 경우 가로 세로 모두 200 미터인 공간을 이용하였고 큰 경우 평방 600 미터의 공간에서 실험을 진행하였다. 특별한 언급이 없는 경우 모의실험은 평방 400 미터 공간을 사용하였다. 각 peer가 자신의 전송 순서가 되면 최소 100개의 패킷을 전송할 수 있으며 최대 3200개의 패킷을 전송할 수 있도록 조정하였다. 소개되는 그래프의 각 값은 60개의 서로 다른 애드혹 네트워크 토폴로지에서 실행하였을 때 모든 세그먼트를 참여한 모든 peer에게 분배 완료한 시간의 평균값이다. 또한 각 peer는 항상 최소한 하나 이상의 다른 peer와 연결되어 있으며 고립된 peer는 없다고 가정하고 모의실험을 진행하였다.

그림 2는 평방 400 미터 영역에서 참여한 peer의 수를 증가시켰을 때 모든 peer가 전체 파일의 분배를 완료할 때 까지 경과한 시간을 나타낸다. 분배하고자 하는 파일의 크기는 1 MByte 이고 512 byte 크기의 패킷을 이용해서 서로 교환하며 각 peer는 자신의 전송 순서가 되면 최대 100개의 패킷을 이웃 peer에게 전송한 후 지정된 다음 peer에게 전송 기회를 넘긴다. 그림에서 “Forward-All”은 모든 peer가 forwarding에 참여하는 방식을 의미하고, “Forward-Excluding”은 excluding 방식을 사용한 결과를 나타낸다. 또한 “Forward-MinCover”는 MCS 방법을 이용하여 선택된 최소의 forward peer만 전송하였을 때의 결과를 나타낸다. 5개의 peer가 참여한 경우 총 60개의 네트워크 토폴로지 중에서 57개는 세 방식 모두 같은 결과를 보였고 excluding 방식과 MCS 모두 3개의 토폴로지에서만 더 나은 성능을 보였다. 10개의 peer가 참여한 경우, excluding 방식은 평균 15%, MCS의 경우 평균 19% 정도 분배 시간이 감소되었다. MCS와 excluding 방식만을 비교할 때, 5개의 peer가 참여한 경우 60번의 실험 모두 excluding 방식과 MCS는 같은 결과를 보여주었고, 6개의 peer가 참여한 경우 2번의 결과만 달랐다. 10개의 peer인 경우 16번의 실험 결과에서 MCS가 일찍 분배를 완료하였다. 대부분의 경우 MCS가 excluding 방식보다 더 빠르게 전체 콘텐츠의 분배를 완료하였다.

그림 3은 평방 500 미터 모의실험 영역에서 참여한 peer의 수를 10개에서 15개 까지 증가시킨 경우 각 방식에서 1 MByte 파일을 서로 교환 완료한 평균시간을 나타낸다.

그림 3에서 peer의 수가 10 이상일 때 평방 400 미터가 아닌 500 미터 영역을 사용한 이유는 좁은 영역에 많은 peer가 밀집해 있는 것 보다 더 넓은 영역에 peer들이 공간을 골고루 확보한 경우 각 방식의 성능평가가 명확히 대비되었기 때문이다.

그림 3도 512 byte 크기의 세그먼트를 하나의 패킷에 저장하여 전송하였고 한 번에 최대 100개의 패킷을 전송하였다. 그림 3에 따르면 두 방식 모두 참여한 peer의 수가 증가할수록 세그먼트 분배완료 시간이 증가하였다. 참여한 peer가 10일 때 excluding 방식은 “forward-all” 방식에 비해 10%의 분배완료 시간이 단축되었고 MCS의 경우에는 16%의 시간이 단축되었다. 또한 참여한 peer가 15였을 때 excluding의 경우 18%, MCS의 경우 29%의 시간이 단축되었다. 이런 결과는 peer의 수가 증가할수록

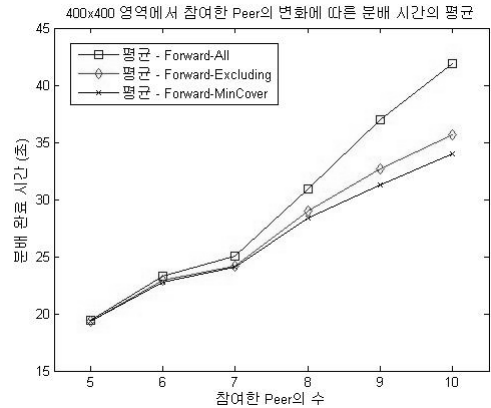


그림 2. 참여한 peer의 수에 따른 분배 완료시간
Fig. 2. number of peers vs. distribution time

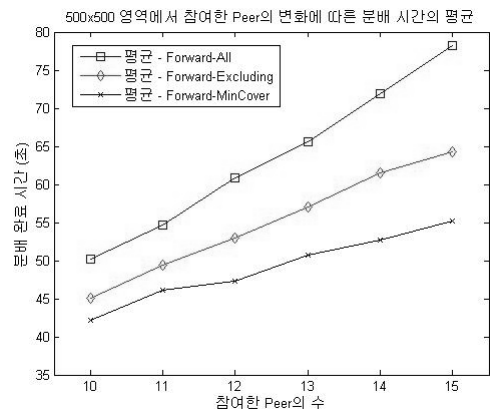


그림 3. peer의 수에 따른 분배 완료시간(평방 500m)
Fig. 3. number of peers vs. distribution time

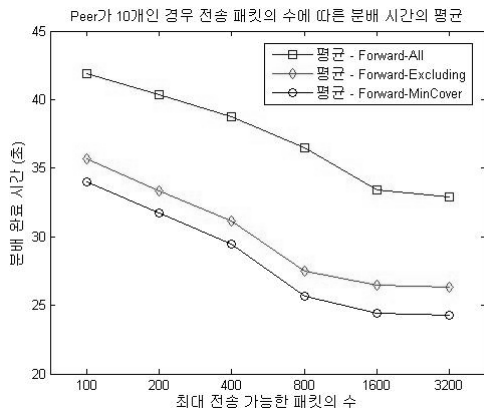


그림 4. 전송 가능한 패킷 수에 따른 분배완료 시간
Fig. 4. maximum packet transmission count vs. distribution time

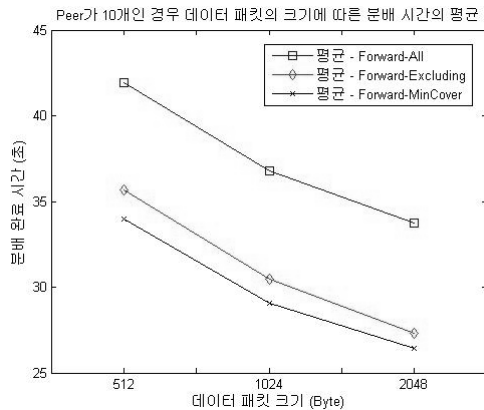


그림 5. 패킷 크기에 따른 분배완료 시간
Fig. 5. packet size vs. distribution time

모의실험 영역에 peer들이 골고루 분포하게 되고 이에 따라 다른 peer의 평균 접근 hop 수가 증가되어 forwarding 수도 증가하기 때문에 전체적인 분배완료 시간도 증가된다. 이 경우 forwarding peer의 수를 줄이는 방법을 사용함으로써 전체적인 분배완료 시간을 감소시킬 수 있다. 60 회의 모의실험에서 10개의 peer가 참여한 경우 MCS가 excluding 방식보다 27회에 걸쳐 빨리 분배를 완료하였으며, 15개의 peer가 참여한 경우에는 MCS가 51회 실험에서 MCS가 더 빠른 분배 완료 결과를 나타내었다. 이는 forwarding을 위한 hop 수가 증가할수록 MCS가 excluding 방식보다 우수한 성능을 보임을 알 수 있다.

그림 4는 10개의 peer가 평방 400 미터 영역에서 1 MByte 파일을 공유하기위해 512 byte 크기의 패킷을 이

용해 한 번에 전송 가능한 패킷 수의 변화에 따라 세 방식의 분배완료 시간의 차이를 보여준다. forwarding 하는 peer는 자신의 전송 차례가 되었을 때, 이웃한 peer가 가지고 있지 않은 세그먼트를 전송하는데 한 번의 전송 순서 동안에 더 많은 수의 세그먼트를 전송하게 되면 전체 파일의 교환완료 시간을 단축할 수 있다. 100개의 세그먼트를 전송하는 경우 excluding 방식은 모든 peer가 전송하는 경우보다 15%(6.219초)의 시간을 단축하였고 MCS의 경우 19%(7.927초)의 교환완료 시간을 단축하였다. 또한 2048개 이상의 세그먼트를 한 번에 전송하게 한 경우 excluding 방식은 20%(6.547초) 정도의 분배완료 시간이 단축하였으며 MCS의 경우는 26%(8.594초) 시간을 단축하였다. 전송 가능한 패킷의 수가 증가함에 따라 단축 시간의 백분율 증가율이 절대 시간보다 증가하는 이유는 “forwarding-all” 방식의 분배완료 시간이 감소하였기 때문이다.

그림 5는 전송되는 패킷의 크기를 변경하였을 때 두 방식의 분배완료 시간을 비교하였다. 10개의 peer가 참여하여 1 MByte 파일을 패킷의 크기에 맞추어 세그먼트로 분할 후 분배하였다. 패킷의 크기가 512 byte인 경우 2048개의 세그먼트가 만들어지며 각 peer당 대략 204개를 나머지 peer들과 서로 교환하게 된다. 패킷의 크기가 1024 byte인 경우 1024개의 세그먼트가 필요하고 2048 byte 크기의 패킷을 사용하면 512개의 세그먼트가 사용된다. 적은 수의 세그먼트가 사용될수록 패킷을 통해 전달되는 전송횟수도 줄어들기 때문에 분배완료 시간은 감소하게 된다. 512 byte 패킷을 사용하는 경우 excluding 방식은 이를 사용하지 않은 경우보다 15%(6.219초)의 성능 향상을 보였고 MCS 방식은 19%(7.927초)의 성능 향상을 보였다. 또한 2048 byte 크기의 패킷을 사용하면 excluding 방식의 경우 19%(6.424초)의 분배완료 시간이 감소하였으며 MCS의 경우 22%(7.294초)의 시간이 감소하였다.

그림 6은 참여한 peer의 수가 10일 때 공유할 파일의 크기가 다른 경우 세 방식의 성능 차이를 보여준다. 모의 실험은 400 평방미터 영역에서 진행되었고 전송되는 패킷의 크기는 512 byte이며 각 peer는 한 번에 최대 100개의 패킷을 전송한다. 전송할 세그먼트의 수는 peer의 수와 패킷의 크기, 그리고 peer가 공유할 파일의 크기에 따라 결정된다. 예를 들어 10개의 peer가 참여하고 512 byte 크기의 패킷을 사용한다고 가정할 때, 공유할 파일의 크기가 512 KByte 인 경우 1024개의 세그먼트를 각 peer가 대략 102개 씩 나누어 전송하게 되며, 1024 KByte 크기의 파일인 경우 해당 파일은 2048개의 세그먼트로 나누

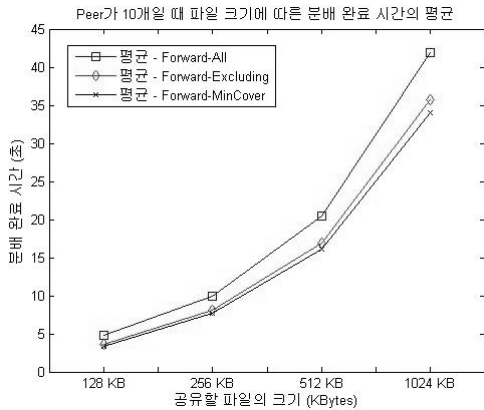


그림 6. 파일 크기에 따른 분배완료 시간
Fig. 6. content file size vs. distribution time

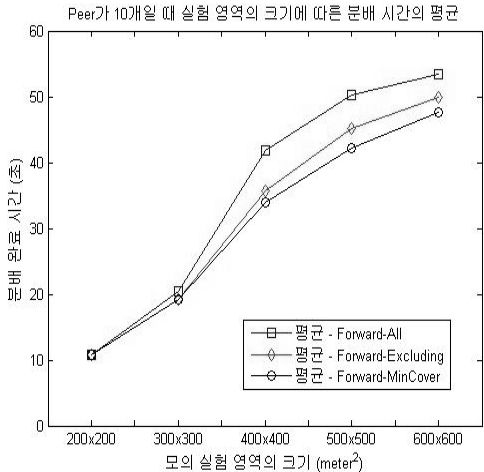


그림 7. 실험영역 크기에 따른 분배완료 시간
Fig. 7. simulation area vs. distribution time

어지고 각 peer는 대략 204개의 세그먼트를 담당한다. 전송할 세그먼트의 개수가 증가하면 분배완료 시간도 증가한다. 파일 크기가 128 KByte부터 2배씩 증가할 경우 분배완료 시간은 “forward-all”의 경우 각각 2.09, 2.06, 2.05 배 증가하였으며, excluding 방식의 경우 2.21, 2.11, 2.11 배 증가되었다. 또한 MCS의 경우 2.24, 2.12, 2.11배 증가되었다.

그림 7은 참여한 peer의 수가 10일 경우 모의실험 영역의 크기를 변화시킬 때 세 방식의 분배완료 시간의 평균값을 보여준다. 공유할 파일의 크기는 1 MByte 이고 패킷의 크기는 512 byte이며 한번에 peer는 최대 100개의 패킷을 전송할 수 있도록 설정하였다. 모의실험 결과

에 따르면 평방 200 미터의 경우 세 방식은 모두 같은 결과를 가진다. 그 이유는 이 영역에서는 한 peer가 어느 위치에 있더라도 자신이 내려 받은 세그먼트를 전송하면 다른 모든 peer들이 자신의 전송 영역 범위에 있으므로 어느 peer도 forwarding 할 필요가 없기 때문이다. 그러나 모의 실험 영역의 크기가 커짐에 따라 세 방법의 forwarding 방식에는 차이가 발생한다.

forward-all 방식의 경우 몇몇 노드가 패킷을 중복해서 forwarding하게 되는 반면, excluding 방식과 MCS 방식의 경우 forwarding 하지 않거나 반드시 해야 하는 노드들을 미리 정하기 때문에 모든 peer가 forwarding 하는 방식과 비교할 때 중복된 전송을 피할 수 있어 분배완료 시간을 단축시킬 수 있다. 모의실험 영역이 커지는 경우 각 peer의 평균 전송 hop 수가 증가하게 되어 forwarding 수가 늘어나기 때문에 분배완료 시간이 증가한다.

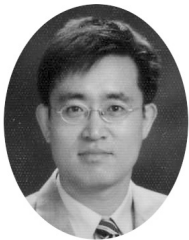
5. 결론

본 논문은 WWAN 채널을 통해 인터넷에 저장된 콘텐츠를 내려 받을 때 지불하는 통신비용의 절감을 위해 해당 콘텐츠를 공유하고자하는 다른 peer와 협력하여 해당 콘텐츠의 일부만 WWAN을 통해 내려 받고 나머지는 WLAN을 통해 다른 peer와 콘텐츠를 교환하여 비용과 시간을 절감하는 효율적인 방식을 제안하였다. 기존의 benefit 값을 이용해 모든 peer가 forwarding에 참여하는 알고리즘은 benefit 값만을 적절히 이용해 패킷의 forwarding 횟수를 줄여 콘텐츠의 분배완료 시간을 감소시킬 수 있는데, 본 논문은 여기에 추가하여 토폴로지의 구성상 forwarding을 하지 않아도 되는 peer들을 미리 배제하는 excluding 알고리즘을 실행하여 해당 peer가 forwarding을 수행하지 않도록 peer를 배제하거나, 최소의 forwarding peer를 선택하여 해당 peer만 forwarding을 수행하도록 하면 패킷의 전송 횟수가 추가적으로 감소하여 분배완료 시간이 모든 peer가 forwarding 하는 방식에 비해 최고 29% 정도 감소하는 것을 모의실험을 통해 확인하였다.

참고 문헌

- Xiao, Y., Leung, K., Pan, and Y., Du, X. (2005), “Architecture, mobility management, and quality of service for integrated 3G and WLAN network”, Wireless Communications and Mobile Computing, Vol. 5, pp. 805-823.
- Kang, S., and Mutka, M (2005), “A mobile peer-to-peer

- approach for multimedia content sharing using 3G/WLAN dual mode channels”, *Wireless Communications and Mobile Computing*, Vol. 5, pp. 633-645.
3. 강승석 (2006), “무선 P2P 네트워크에서 Peer 사이의 효율적 자료 분배 방법에 대한 연구”, *한국 시뮬레이션학회 논문지*, Vol. 15, No. 4, pp. 97-105.
 4. Zhong, S., Chen, J., and Yang, Y. (2003), “Sprite: A Simple, Cheat-Proof, Credit-Based System for Mobile Ad Hoc Networks”, *IEEE Infocom 2003*, pp. 1987-1997.
 5. Karbhari, P., Zegura, E., and Ammar, M. (2003), “Multi-point-to-Point Session Fairness in the Internet”, *IEEE Infocom 2003*, pp. 207-217.
 6. Castro, M., Druschel, P., Kermarrec, A., Rowstron, A., and Singh, A. (2003), “SplitStream: High Bandwidth Multicast in Cooperative Environments”, *ACM SOSP 2003*, pp. 298-313.
 7. BitTorrent, The BitTorrent File Distribution System, “<http://www.bittorrent.org/>”.
 8. Evan, S. (1979), “Graph algorithm”, Science Press, 1979, pp. 204-209
 9. Lim, H., Kim, C., and Kim, C. (2000), “Multicast Tree Construction and Flooding in Wireless Ad Hoc Networks”, *ACM MSWiM 2000*, pp. 61-68.
 10. Peng, W., and Lu, X.(2001), “AHBP: An Efficient Broadcast Protocol for Mobile ad hoc Networks”, *Journal of Science and Technology (JCST) - Beijing, China*, Vol 16, No. 3, pp. 114-125.
 11. Williams, B., and Camp, T.(2002), “Comparison of Broadcasting Techniques for Mobile Ad Hoc Networks”, *IEEE MobiHoc 2002*, pp. 194-205.
 12. ns2, The network simulator, “<http://www.isi.edu/nsnam/ns>”.



강 승 석 (msukang@swu.ac.kr)

1992 고려대학교 이과대학 전산과학과 학사
 1998 Michigan State University 전산학 석사
 2004 Michigan State University 전산학 공학박사
 현재 서울여자대학교 컴퓨터학부 조교수

관심분야 : Ad Hoc Network, RFID, Mobility, QoS, Anonymity, Multimedia Streaming, Ubiquitous Computing