슈퍼피어를 이용한 모바일 P2P시스템을 위한 효율적인 플러딩 알고리즘

(Efficient Flooding Algorithm for Mobile P2P Systems using Super Peer)

강 소 영 † 이 광 조 †
(Soyoung Kang) (Kwangjo Lee)

양 성 봉 ††
(Sungbong Yang)

요 약 다양한 종류의 새로운 모바일 기기의 등장과 이들 기기들의 사용자의 증가에 따라 모바일 P2P 시스템과 관련된 연구가 활발히 진행되고 있다. 본 논문에서는 모바일 환경에서 double-layered 슈퍼 피어 시스템을 위한 새로운 검색 알고리즘을 제안한다. 제안한 검색 알고리즘에서는 전체 실험 영역을 그리드 셀로 분할하며 셀들은 같은 크기를 가지고 있다. 그리드는 모바일 기기의 통신 반경과 피어들의 수를 고려하여 적절히 구성된다. 제안한 검색 알고리즘은 방향 분할 플러딩으로서 검색시 셀들의 검색 방향을 포함하는 방법을 기반으로 한다. 이 방법은 성공적으로 네트워크의 부하를 줄였으나 낮은 검색 성공률을 보여준다. 성공률을 보다 개선하기 위해 슈퍼 피어를 위한 bridge-peer table과 n-way 검색을 활용하였다. 그 실험 결과 제안한 알고리즘은 기존의 double-layered 시스템에 비하여 평균 20~30%의 메시지 패킷수가 감소되었음을 보여주었다. 성공률 역시 double-layered 시스템에 비해 약 2~5% 개선되었다.

키워드 : P2P, Grid, Super-peer, Double-layer system

Abstract As the appearances of various new mobile devices and the explosive growth of mobile device users, many researches related to mobile P2P systems have been proceeded actively. In this paper, we propose a new search algorithm for the double-layered super peer system in the mobile environment. For the proposed search algorithm, we divide the entire experiment region into a grid of cells, each of which has the same size. The grid is configured properly by considering the communication range of a mobile device and the number of peers in the system. The proposed search algorithm is a partial flooding search method based on the directions of cells involved with the search. It reduces successfully the network traffic, but shows a low search hit ratio. To enhance the search hit ratio, we introduce a bridge-peer table for a super peer and utilize an n-way search. The experimental results show that the proposed algorithm made an average of 20~30% reduction in the number of message packets over the double-layered system. The success ratio was also improved about 2~5% over the double-layered system.

Key words : P2P, Grid, Super-peer, Double-layer system

1. 서 론

모바일 기기의 발전에 따라 다양한 핸드폰, PDA, 넷북 등이 보급되면서 모바일 P2P(Peer-to-Peer)환경에서 이를 사용하는 사용자 수도 역시 증가하고 있다. 모바일 기기는 이동성이라는 특성을 지니고 있으며, 각 모바일 기기는 통신 범위가 제한되어 있기 때문에 유선환경에서의 P2P와는 다른 통신환경이 제공되어야만 한다. 따라서 이를 반영한 모바일 P2P 시스템에 대한 연구가 진행되어 왔다. 냅스터(Napster)[1], 오픈냅(Open NAP)[2]이 대표적인 유선 P2P 시스템이라면, ORION(Optimized Routing Independent Overlay Network)[3]은 대표적인 모바일 환경을 위한 P2P 시스템이다.

본 논문은 [4]에 제시된 슈퍼 피어 P2P 시스템을 기반으로 하며, 슈퍼 피어간의 플러딩 알고리즘을 개선하여 메시지 패킷 오버헤드를 감소시켜 시스템 향상에 기여하는 데에 그 목적을 두고 있다.

본 논문에서 제안하는 슈퍼 피어를 이용한 지역 균등 분할 P2P시스템에서는 실험 공간을 일정 크기의 셀로 나누어 그리드 공간을 만들고, 각 셀에 하나의 슈퍼 피어를 설정하여 하위의 서브 피어들을 관리하고 파일의 검색 및 전송을 슈퍼 피어가 전담하여 관리함으로써 불

· 본 연구는 한국과학재단(KOSEF) 일반 연구자 지원사업(2009-0073072) 지원으로 수행되었음
· 이 논문은 2009 한국컴퓨터종합학술대회에서 '슈퍼 피어 P2P시스템을 위한 효율적인 플러딩 알고리즘'의 제목으로 발표된 논문을 확장한 것임

† 학생회원 : 연세대학교 컴퓨터과학과
milkyway@cs.yonsei.ac.kr
kjlee5435@gmail.com
†† 종신회원 : 연세대학교 컴퓨터과학과 교수
sbyang@yonsei.ac.kr

논문접수 : 2009년 8월 14일
심사완료 : 2009년 11월 13일

Copyright©2010 한국정보과학회 : 개인 목적이나 교육 목적인 경우, 이 저작물의 전체 또는 일부에 대한 복사본 혹은 디지털 사본의 제작을 허가합니다. 이 때, 사본은 상업적 수단으로 사용할 수 없으며 첫 페이지에 본 문구와 출처를 반드시 명시해야 합니다. 이 외의 목적으로 복제, 배포, 출판, 전송 등 모든 유형의 사용행위를 하는 경우에 대하여는 사전에 허가를 얻고 비용을 지불해야 합니다.

필요한 중복되는 메시지 전송량을 줄인다. 그리드 공간의 각 셀에 인덱스를 지정하고, 이를 활용하여 슈퍼 피어가 검색할 방향이나 플러딩 방향을 설정한다. 여기에 슈퍼 피어간의 통신을 더욱 용이하고 안정적으로 할 수 있도록 슈퍼 피어는 bridge peer table 정보를 추가로 보유하도록 하였으며, n-way 탐색 방식을 부분적으로 수행하여 탐색 성공률을 높이고자 하였다.

본 논문의 구성은 다음과 같다. 2장은 관련 연구인 ORION과 슈퍼 피어를 이용한 P2P 시스템을 설명하고, 3장에서는 모바일 환경에서의 슈퍼 피어 P2P 시스템의 성능 향상을 위한 알고리즘들을 제안하며, 4장에서는 실험 결과를 보여준다. 마지막으로 5장에서 결론을 내린다.

2. 관련연구

2.1 ORION

ORION은 모바일 P2P 시스템 중에서 가장 대표적인 시스템으로서 라우팅을 위한 테이블과 파일 관리를 위한 라우팅 테이블을 이용하여 효과적으로 파일을 공유한다. 라우팅 구조는 AODV(Ad-hoc On-demand Distance Vector)[5] 라우팅 테이블과 유사한 형태를 지니고 있으며, 네트워크 링크가 생성되면 플러딩 방식으로 파일을 검색하게 된다. 플러딩 방식이란 자신에게 연결된 모든 피어들에게 메시지를 전달하는 것으로, 이 방식은 연결 된 피어 수가 많을수록 매우 많은 양의 메시지가 생성되므로 이로 인한 시스템의 부하가 심각한 수준이 될 수도 있다.

2.2 Mobile Super Peers Systems

[4]에서는 ORION에서의 피어들 간의 플러딩 방식의 단점을 개선시키기 위해 2계층 구조인 슈퍼 피어 시스템을 제안하였다. 이 시스템에서는 피어들을 상위 계층의 슈퍼 피어와 슈퍼 피어에 의해 관리되는 하위 계층의 서브 피어로 분류하고, 기존의 라우팅 테이블 구조를 변경하였다. 슈퍼 피어는 자신이 관리하고 있는 서브 피어의 정보(*id*, 주소, 파일 리스트)를 가지고 있으며, 파일의 검색 및 전송은 슈퍼 피어들이 맡아 관리한다. 서브 피어가 검색을 요청하면 우선 자신의 슈퍼 피어에게 질의를 보내게 되고, 슈퍼 피어는 자신이 가지고 있는 파일 리스트를 검색하여 보유하고 있으면 이 정보를 서브 피어에게 알려주게 된다. 만약 가지고 있지 않는 경우 슈퍼 피어는 다른 이웃 슈퍼 피어에게 질의 메시지를 전송하는 방식으로 검색을 수행한다. 따라서 서브 피어와 슈퍼 피어, 또는 슈퍼 피어간의 통신만 일어나게 되므로, ORION에서처럼 피어와 연결된 모든 피어들에게 메시지를 전달하게 되는 경우가 발생하지 않는다. [4]에서는 검색 시 발생되는 평균 패킷 수가 ORION과 비교하여 20% 이상 감소됨을 보였다.

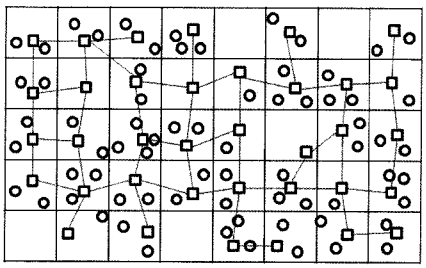

그림 1 지역 균등 분할 시스템에서 슈퍼 피어와 서브 피어의 구성 예

2.2.1 지역 균등 분할 시스템

지역 균등 분할 시스템은 서비스 지역을 일정 크기로 분할하여 각 지역마다 하나의 슈퍼 피어를 할당한다. 각 슈퍼 피어는 동일 지역 내에 존재하는 서브 피어들을 관리하며, 슈퍼 피어들끼리 서로 직접 통신을 한다. 빈 지역이거나 통신거리가 닿지 않는 경우에는 이웃 지역에 있는 슈퍼 피어와 연결이 되지 않을 수도 있다. 또한 통신범위 내에 있더라도 최대거리에 있는 피어가 누락되는 경우도 있으므로 공간을 분할할 때 통신범위보다 작게 지역의 크기를 정한다.

그림 1은 지역 균등 분할 시스템에서 슈퍼 피어와 서브 피어가 어떻게 구성되는지 보여주고 있다. 사각형은 슈퍼 피어를 표시하고, 원은 서브 피어를 나타낸다. 또한 실선들은 통신 가능한 거리상의 슈퍼 피어들 간의 연결을 보여준다.

지역 균등 분할 시스템은 각각의 피어가 자신의 위치를 파악할 수 있어야 하며, 이를 위해 기기의 GPS 기능이 요구된다. 피어의 이동에 의해 위치가 변경되는 경우 지역의 중심 위치와 가깝게 위치한 피어를 슈퍼 피어로 재설정하며, 새로운 피어가 추가 되는 경우 이웃 지역에서 왔을 확률이 높으므로 이웃 지역의 슈퍼 피어로 하여금 이를 알려 서브 피어 목록에서 삭제하도록 한다.

3. 그리드 분할 방식에서의 플러딩 알고리즘

본 논문에서 제안하는 시스템은 슈퍼 피어 시스템을 바탕으로 하는 계층 구조 시스템을 기반으로 하며, 실험 영역을 그리드로 분할하고, 탐색을 방향에 근거하여 효율적으로 탐색을 수행한다. 본장에서는 그리드 기반 탐색 알고리즘을 제안하며, 이를 위해 실험 영역의 분할 방법, bridge-peer table, 방향에 근거한 플러딩 탐색 방법 및 n-way 탐색 방식을 설명한다.

3.1 그리드 분할 방식 시스템

서비스 지역을 일정한 크기로 분할하고, 각 영역을 셀이라고 칭하며, 각각의 셀에 고유한 id를 할당한다. 영역을 분할할 때 셀의 크기를 설정하는 것은 시스템 성능에

중대한 영향을 미칠 수 있다. 즉, 셀의 크기가 작으면 많은 수의 셀이 만들어지고 이와 더불어 슈퍼 피어 수도 증가하게 되므로 슈퍼 피어 시스템에서 이루어지는 다중 전송된 메시지 수를 감소시키기 어렵다. 반대로 셀의 크기가 크면, 셀의 수는 감소하나 통신거리로 인한 제한 때문에 슈퍼 피어 간의 연결이 제대로 이루어지지 않고, 또한 슈퍼 피어가 자신의 서브 피어들을 제대로 관리하지 못하게 되므로, 검색 실패율이 증가되는 요인이 된다. 따라서 실험 영역과 통신거리에 비례한 적절한 셀의 크기를 찾아야 최적의 결과를 얻을 수 있으며, 피어의 이동 속도 역시 셀 크기를 정하는 데에 고려되어야 한다. 본 논문의 실험에서는 다양한 크기를 실험한 결과 전파거리의 약 80%가 적절한 셀의 크기인 것으로 판명되었다.

그리드 분할 방식 시스템에서는 고유한 셀 id와 이를 이용한 주변 셀 및 슈퍼 피어가 위치한 방향을 활용한다. 슈퍼 피어는 자신이 속한 셀 id와 서브 피어 목록, 파일 리스트, 연결 가능한 8 방향 (N, NW, W, SW, S, SE, E, NE)에 해당되는 이웃 셀에 관한 정보를 가진다. 이러한 방향 정보 이용은 파일 및 피어 검색 시 원하는 방향으로의 통신을 가능하게 한다. 슈퍼 피어는 자신이 관리하고 있는 셀에 있는 피어의 진출입의 상황이 발생하면 최종 갱신 시간과 피어들의 속도와 이동 방향에 따라 해당 이웃 슈퍼 피어에게만 통보하여 변경된 서브 피어 정보를 갱신할 수 있도록 한다. 따라서 주기적으로 수행되는 P2P 시스템 전체 갱신과 별도로 실시간 부분 갱신이 가능하게 된다.

3.2 Bridge-peer Table

그리드 분할 방식 시스템에서는 피어의 분포 상태와 이동성에 따라 공간의 일부분에 아무 피어도 없는 빈 셀이 발생할 수도 있다. 이러한 빈 셀로 인하여 파일 탐색 성공률이 저하될 수 있다. 따라서 이를 보완하기 위한 방법으로 *bridge-peer table*이라는 새로운 정보를 각 슈퍼 피어가 보유하도록 하였다. bridge-peer란 빈 이웃 셀이 있는 방향에 가까이 있는 서브 피어로서 빈 셀 건너편에 위치한 셀의 슈퍼 피어와의 연결을 가능하게 하는 중계자 역할을 담당하는 피어이다.

그림 2에서와 같이, 만일 S 방향에 빈 셀이 있다면 서브 피어를 이용하여 건너편 셀의 슈퍼 피어와 연결될 수 있는지를 파악하고, 연결 가능한 서브 피어를 발견하면 이를 bridge-peer로 설정 후 bridge-peer table에 저장하게 된다.

3.3 위치에 따른 방향 분할 플러딩

기존의 슈퍼 피어시스템에서는 피어가 파일 탐색 시작 신의 슈퍼 피어에게 먼저 파일의 보유 여부를 물어보게 된다. 이 때 슈퍼 피어는 자신이 가지고 있는 파일 리스트와 서브 피어 목록을 검색한다. 만일 파일을 가지고

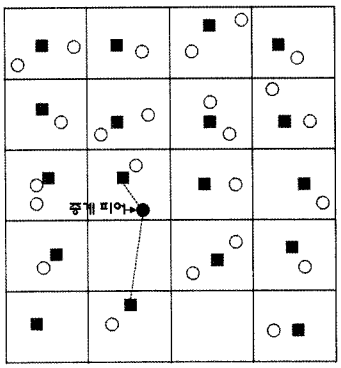

그림 2 슈퍼 피어와 중계 피어

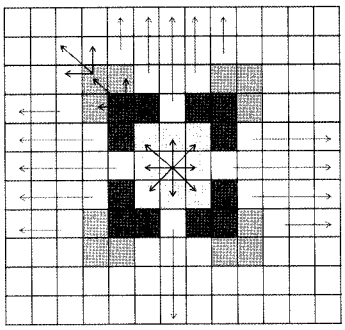

그림 3 위치에 따른 분할 플러딩

있지 않다면 이웃 슈퍼 피어들에게 파일 탐색을 요청하게 되는데, 이 때 자신과 연결된 모든 슈퍼 피어에게 메시지를 전송하게 된다. 본 논문에서는 이러한 경우의 슈퍼 피어들 사이의 플러딩을 보완하기 위해, 메시지를 특정 방향의 슈퍼 피어들에게만 보내는 방식을 제안한다.

그림 3에서와 같이 중앙에 위치한 슈퍼 피어가 자신의 서브 피어가 찾는 파일을 가지고 있지 않다면 8방향의 이웃 셀의 슈퍼 피어들에게 질의 메시지를 전송한다. 그 다음엔 메시지를 전달받은 셀의 위치에 따라 이웃 셀과 겹치지 않게 정해진 방향의 셀 영역으로만 메시지를 전송한다. 즉, 각 모서리 부분에 해당되는 셀은 그림 3에서와 마찬가지로 기본적으로 3개의 이웃 셀의 슈퍼 피어에게 메시지를 전송하며, 그 외의 셀들은 전달받은 방향의 이웃 셀로만 메시지를 전달한다.

하지만 영역에 분포된 피어들의 위치에 따라 빈 셀들이 발생하며, 전달되어야 할 방향에 bridge-peer가 없을 수도 있다. 또한 여러 개의 빈 셀들이 연속적으로 위치하는 경우도 있다. 이때에는 그림 4와 같이 빈 셀의 위치에 따라 이웃 셀들의 메시지 전달 방향을 추가시킨다. 그림 4는 6가지의 경우를 보여주고 있으며, 나머지 6가지의 경우는 그림 4의 경우에 대칭되므로 생략되었다.

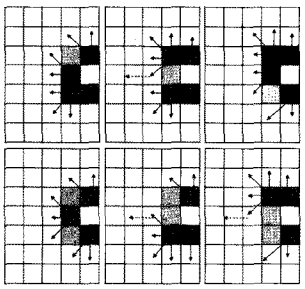

그림 4 이웃 셀의 상황에 따른 메시지 전달 방향 추가

3.4 n-way 검색

위치에 따른 방향 분할 플러딩 방식이 슈퍼 피어들 간의 중복된 메시지 전송을 감소시키는데 크게 기여는 하나, 이웃한 셀이 비어있는 경우, 특히 여러 개의 빈 셀이 연결된 형태로 존재하거나 또는 통신 거리와 셀의 크기에 대한 제약으로 인해 슈퍼 피어들이 분할 검색 방향을 따라 연결되어 있지 않으며, 이와 다른 방향의 여러 셀의 슈퍼 피어들에 의해 연결된 경우에는 검색되지 못하는 피어가 있을 수도 있다. 이를 보완하기 위하여, 그림 5와 같이 4-way 방향 또는 8-way 방향으로 검색을 실시할 경우, 보다 많은 셀들을 방문함으로써 파일 탐색 성공률을 높일 수 있다.

n-way 검색 알고리즘은 그림 5와 같이 n 방향으로 질의 메시지를 전달하고 전달받은 셀들은 다시 또 n 방향으로 메시지를 전달한다. 단, n은 4 또는 8이다. 따라서 그림에서 보여주듯이 동일한 메시지를 같은 셀에 중복하여 보내거나 받게 되는 단점을 가지나, 모든 셀들을 방문하게 되므로 앞서 제시한 방향 분할 플러딩이 놓칠 수 있는 위치의 셀들을 모두 검색할 수 있다. 8-way는 4-way보다 더 많은 메시지를 보내게 되므로, 4-way보다 성공률은 좀 더 높으나, 보다 더 많은 수의 패킷이 발생된다. 결론적으로 n-way 검색 알고리즘은 모든 셀들을 검색하여 성공률을 높이지만, 중복되는 패킷의 발생으로 방향 분할 플러딩보다 패킷 오버헤드가 훨씬 크게 된다. 다음은 위에서 언급된 탐색 기법들을 적용한 탐색 알고리즘이다.

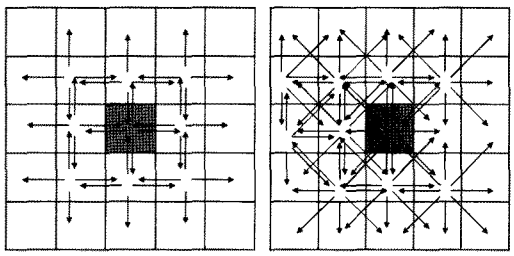

그림 5 4-way 검색과 8-way 검색

알고리즘 1. 제안한 플러딩 알고리즘

[1] 각 슈퍼 피어는 비어있는 이웃 셀에 대한 bridge-peer table을 생성한다.
[2] 8 방향의 이웃 셀의 슈퍼 피어에게 질의 메시지를 전달한다. 메시지를 전달받은 슈퍼 피어는 메시지의 전달 방향에 따라서 이웃 셀에 메시지를 전송한다. 이 때 빈 셀이 있는 경우, 빈셀의 이웃 셀이 대신 전달해야할 셀에 메시지를 전송하도록 한다.
[3] 3개 이상의 연속된 이웃 셀들이 비어 있는 경우거나 주변에 비어있는 셀만 있는 경우에는 n-way 검색을 실행한다. 단, n은 4 또는 8이다.

4. 실험결과

제안된 알고리즘을 기반한 시스템을 기존 시스템과 성능 비교를 위해 다양한 실험을 수행하였고, 실험 환경은 [4]에 제시된 환경을 따랐다. 입력으로 텍스트 파일에 피어의 id, 주소, 피어의 위치, 파일 리스트를 기록하고, 텍스트 파일을 한 줄씩 읽어가며 피어에 해당 정보를 저장하도록 하였다. 이러한 텍스트 파일을 10개 만들어 기존의 시스템과 성능 비교하였다. 실제 도시 환경에서 많은 사용자들이 일정 지역 내에서 모바일 기기를 이용하여 P2P 통신을 할 때를 가정하여 1000m × 1000m 크기의 실험 공간에 피어의 수 100~1000으로 변화시켜 가면서 실험하였다. 통신범위도 모바일 환경을 가정하여 120m로 설정하였다. 셀의 사이즈에 대해서도 다양하게 실험해본 결과 $100m^2$ 가 슈퍼 피어 설정과 이웃 셀의 슈퍼 피어와의 연결 및 생성되는 셀의 개수를 고려했을 때 가장 좋은 결과를 보였기 때문에 본 실험에서 셀의 사이즈를 $100m^2$로 설정하였다.

본 논문에서 제안된 탐색방식들과 bridge-peer table 활용을 다양하게 적용하여 아래와 같이 6가지 방법으로 실험하였다.

(1) 방향 분할 플러딩만을 적용
(2) 방향 분할 플러딩 + bridge-peer table 탐색
(3) 방향 분할 플러딩 + bridge-peer table + 4-way 탐색
(4) 방향 분할 플러딩 + bridge-peer table + 8-way 탐색
(5) 4-way 방향 탐색 기법만을 적용
(6) 8-way 방향 탐색 기법만을 적용

그림 6은 검색 시 전달되는 총 패킷의 수를 10개의 테스트 파일로 실험한 결과 값의 평균을 나타낸 그래프이다. 그래프가 보여주는 것과 같이 방향 분할 플러딩과 방향 분할 플러딩 + bridge-peer table 검색 방법이 다른 실험결과에 비해 현저하게 적은 패킷 수를 보여주었다. 하지만 그림 7의 평균 성공률 그래프가 보여주듯이 두 방법은 탐색 성공률 면에서 다른 알고리즘들에 비해 현저히 떨어지는 단점을 가지고 있다. 이는 정해진 방향으로만 메시지를 전달하므로 빈 셀들이 있는 방향은 건

너편에 피어가 있는 셀이 있음에도 메시지 전송을 중지하기 때문이다. 방향 분할 플러딩 + bridge-peer table + 4-way와 방향 분할 플러딩 + bridge-peer table + 8-way 검색을 보면 기존의 슈퍼 피어 P2P 시스템, 즉, 4-way에 비하여 패킷 오버헤드가 크게 감소됨을 보이고 있다. 그림 7에서는 6가지 방식에 대한 평균 탐색 성공률을 보여주고 있다. 그림에서 보듯이 n-way 탐색을 활용한 경우는 그렇지 않은 경우보다 높은 성공률을 보였다. 8-way가 4-way 검색 방법보다 성공률이 약간 높은 이유는 8방향에 대한 탐색이 확실히 이루어지기 때문이다. 하지만 8-way는 그만큼 많은 메시지의 생성으로 인하여 평균 패킷 수가 더 증가된다.

방향 분할 플러딩 + bridge-peer table + 4-way/8-way 탐색 방식의 경우 패킷 오버헤드가 약 21~33% 감소한 것을 알 수 있으며, 성공률 역시 기존의 슈퍼 피어 P2P시스템에 비하여 약 2.5~5% 증가하였다. 또한 피어의 수의 증가하더라도 그에 비례하여 패킷의 수가 크게 증가하지 않음을 확인할 수 있다.

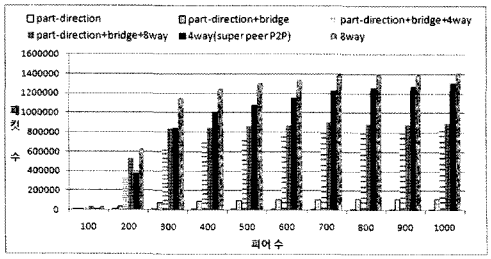

그림 6 평균 패킷 수

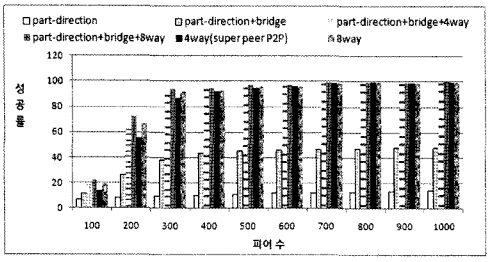

그림 7 탐색 성공률

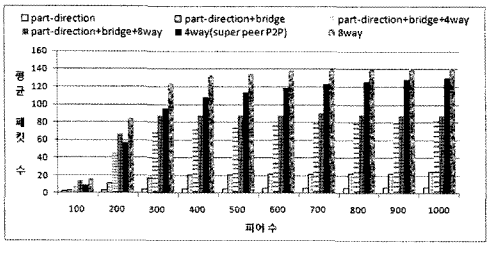

그림 8 하나의 메시지로 인해 발생된 평균 패킷수

그림 8은 각 알고리즘별 검색 시 하나의 검색 메시지가 실행될 때 발생되는 평균 패킷 수를 측정하여 6가지 방식을 피어 수 변화에 따라 그래프로 나타낸 것이다. 피어의 수가 증가할수록 하나의 검색 메시지 당 발생하는 패킷의 수도 증가하게 되는 것을 알 수 있다. 실험 결과 성공률과 패킷 오버헤드를 고려하였을 때 방향 분할 플러딩 + bridge-peer table + 4-way 방식이 기존의 슈퍼 피어 P2P 시스템에 비하여 패킷 오버헤드를 21% 감소시켰고 성공률이 약 5% 증가시켰으므로 실험한 방식들 중 가장 성능이 우수하다고 할 수 있다.

5. 결 론

본 논문에서는 기존의 슈퍼 피어를 이용한 모바일 P2P 시스템의 네트워크 트래픽을 줄이고 성공률을 향상시키기 위해 실험 영역을 그리드로 분할하고, 셀의 방향을 이용하여 부분적인 플러딩을 기반한 탐색 방법을 제안하였고, 탐색 성공률 향상을 위해 중계 피어를 활용하였으며, 트래픽의 증가를 약간 희생하나 보다 높은 탐색 성공률을 위해 부분적인 n-way 탐색 방식을 활용하였다.

실험 결과와 같이 전체적으로 평균 패킷 수와 성공률을 비교해 보았을 때, 방향 분할 플러딩과 bridge-peer table, 그리고 4-way 탐색을 결합한 알고리즘이 가장 우수한 것을 알 수 있었다. 인접한 셀에 대한 선택적이고 중복을 배제한 플러딩 방법과 패킷 수를 줄이면서 성공률을 높이기 위한 4-way 기법이 시스템 향상에 기여하였다고 판단된다. 향후에는 피어의 이동 벡터와 군집 형태에 따라 지역적인 그리드 공간 생성과 새로운 슈퍼 피어의 선정 방법 개발로 성공률을 더 높이는 연구를 수행하고자 한다.

참 고 문 헌

[1] The Napster home page, http://www.napster.com/
[2] The openNap home page, http://opennap-ng.sourceforge.net/
[3] A. Klemm, C. Lindemann, and O. Waldhorst, "A Special-Purpose Peer-to-Peer File Sharing System for Mobile Ad hoc Networks," *Proc. of the Vehicular Technology Conference*, vol.4, pp.2758-2763, Oct. 2003.
[4] Jung-Suk Han, Jin-Woo Song, Kwang-Jo Lee, and Sung-Bong Yang, "Mobile Peer-to-Peer System using Super Peers for Mobile Environments," *Journal of KIISE : Computing Practices and Letters*, vol.14, no.3, pp.286-290, May. 2008. (in Korean)
[5] C. Perkins, E. Royer, and S. Das, "Ad hoc On-Demand Distance Vector(AODV) Routing," http://www.ietf.org/interne t-drafts/draft-ietfmane-taodv-11.txt, *IETF Internet Draft*, June. 2002.