

## 자기 조직적 링 구조를 이용한 P2P 검색기법

손재의 한세영<sup>0</sup> 박성용

서강대학교 컴퓨터학과

{andrew, syhan<sup>0</sup>, parksy}@sogang.ac.kr

### A Peer-to-Peer Search Scheme using Self-Organizing Ring

Jae-eui Son, Sae-Young Han<sup>0</sup>, Sungyong Park  
Dept. of Computer Science, Sogang University

#### 요 약

본 논문에서는 비구조적 피어-투-피어 시스템에서의 낮은 검색 성공률과 긴 검색 시간을 개선하기 위하여, 성능이 우수한 우수 피어들로 자기 조직적인 링 구조를 구성하고 광고 및 검색에 이용하는 검색 기법을 제안하였다. 우수 피어 링 구조는 동적인 환경에서 시스템의 상황에 따라 적응적으로 크기가 변하고, 에이전트를 이용하여 지속적으로 우수한 피어들이 링 구조에 참여하게 함으로써 검색 성공률을 높이고, 빠른 검색 시간을 유지할 수 있게 하였다.

#### 1. 서 론

피어-투-피어 시스템에서는 중앙 집중적인 서버의 도움 없이 분산되어 있는 각 피어들의 자원이나 서비스를 이용할 수 있는데, 크게 구조적인(structured) 피어-투-피어 시스템과 비구조적인(unstructured) 피어-투-피어 시스템으로 나뉜다.

구조적인 피어-투-피어 시스템에서는 각 피어들이 위상을 구성하고 분산 해시 테이블(distributed hash table)을 이용하여 검색 메시지를 전달한다. 이 시스템에서는 검색의 성공을 보장하고 상대적으로 빠른 검색 시간을 갖는다는 장점이 있으나, 특정 위상과 분산 해시 테이블을 유지하는 비용이 들고, 동적인 환경에서는 이 비용이 더욱 커진다. 비구조적인 피어-투-피어 시스템의 경우 특정 위상에 기반을 두지 않으므로 위상을 유지하기 위한 비용이 들지 않으나 플러딩(flooding)이나 임의 경로(random walk) 검색 방법을 사용하므로 검색의 성공을 보장하지 못하고 검색 시간도 구조적인 피어-투-피어 시스템보다 느리다[1].

한편 기존의 연구결과에 따르면 비구조적인 피어-투-피어 시스템을 이루는 피어들은 파워-로우 토폴로지(power-law topology)를 형성하고[2], 또한 각 피어들은 서로 다른 능력을 갖는다고 보고되고 있다[3].

본 논문에서는 이러한 연구 결과를 기반으로 하여 우수한 성능을 갖는 우수 피어들이 자기 조직적인 링 구조를 형성하고, 나머지 피어들은 자신의 키를 광고하거나 원하는 콘텐츠를 검색하는데 이 우수 피어 링을 이용하는 계층적 검색 기법을 제안하고자 한다.

제안된 기법에서는 각 피어들이 에이전트를 이용하여 서로의 능력에 대한 정보를 수집하고, 자신의 능력이 상대적으로 우수하다고 판단되면 자신을 우수 피어로 설정하고 링 구조에 참여한다. 이 링 구조는 피어-투-피어 시스템의 상황에 따라 적응적으로 크기가 변하므로, 링 구조를 유지하기 위한 불필요한 오버헤드를 줄이고 빠른 검색 시간을 유지한다.

제안 기법의 성능을 검증하기 위하여 기존의 임의 경로 탐색 방법에 한 홉 복제기법을 이용하는 Gnutella와 정적인 우수 피어 링 구조의 성능을 시뮬레이션을 통해 성공 검색 메시지 수, 평균 검색 시간, 오버헤드 등의 측면에서 비교하였다.

본 논문은 다음과 같이 구성된다. 2장에서는 제안하고자 하는 자기 조직적 우수 피어 링 구조를 이용한 검색 기법을 자세히 소개한다. 3장에서는 시뮬레이션 내용과 결과를 보이고 4장에서는

결론 및 향후 과제에 대해 논의하고자 한다.

#### 2. 자기 조직적 우수 피어 링을 이용한 검색 기법

자기 조직적 우수 피어 링 구조를 이용한 검색 기법에서 시스템에 참여하는 피어들이 에이전트를 이용하여 다른 피어의 정보를 수집하고, 수집해온 정보를 이용하여 자신을 우수 피어로 선택할지를 결정하며, 우수 피어로 선택되면 우수 피어들이 형성한 링 구조에 참여하게 된다. 이 링 구조는 시스템의 환경에 따라 적응적으로 크기가 변하는데, 이를 위하여 각 우수 피어는 자신의 우수 피어 정보(피어 류)를 다른 우수 피어들과 일정화하기 위해 가시를 이용한다. 각 피어에서 광고나 검색 요청이 있을 경우, 메시지를 생성하고 임의 경로 검색을 이용하여 임의의 우수 피어에게 전달하면, 이 우수 피어는 해시 함수와 피어 류를 이용하여 그 광고가 저장되어야 할 우수 피어를 찾고, 해당 우수 피어에게 메시지를 전달한다.

제안하는 검색 기법은 크게 네 가지 알고리즘, 즉, 에이전트 인 구 및 분산 조절 알고리즘, 우수 피어 선택 알고리즘, 피어 류 일정화 알고리즘, 그리고 광고 및 검색 알고리즘으로 구성된다. 다음은 각 알고리즘에 대한 설명이다.

##### 2.1 에이전트 인 구 및 분산 조절 알고리즘

비구조적인 피어-투-피어 시스템에서 각 피어는 로컬 정보만을 가지므로, 에이전트를 이용하여 다른 피어들의 정보를 수집한다. 에이전트의 인구가 많을 경우 많은 양의 정보를 수집하므로 보다 정확하게 자신이 상대적으로 우수한 피어인지를 판단 할 수 있지만 오버헤드가 커지고, 반면 에이전트 인구가 적을 경우는 오버헤드는 적지만 판단이 부정확해진다. 한편, 에이전트가 시스템에 골고루 분포되어야 전체적인 시스템의 정보를 얻어올 수 있다. 따라서 오버헤드를 줄이고 검색능력을 향상시키는 적절한 에이전트 인 구 및 분산 조절 알고리즘이 필요하다.

에이전트의 인구를 조절하기 위해서는 먼저 에이전트를 생성하고 소멸시킬지를 결정해야 한다. 한 피어에 에이전트가 이전(migration)되면, 피어는 그 에이전트가 수집한 정보를 받고, 그 에이전트에게 자신의 정보를 추가하여 다른 피어로 이전시킨다. 한 에이전트가 이전된 후 새로운 에이전트가 최소 시간( $\alpha$ ) 이전 에 이전되어 오면, 그 피어 주변에 에이전트가 많은 것이므로 그 에이전트를 소멸시킨다. 만일 반대로 최대 시간( $\beta$ ) 이후에 새로운

에이전트가 이전되어 오면, 주변에 에이전트가 적은 것이므로 새로운 에이전트를 특정 확률로 하나 더 생성한다. 그 알고리즘은 (그림 1)과 같다.

또한, 에이전트가 시스템 전체에 골고루 분포하도록 하기 위해서는, 에이전트를 이전시킬 이웃 피어를 잘 선택해야 한다. 에이전트 이전 시간 간격이 클수록 에이전트가 이전 될 필요가 높으므로 에이전트 이전 시간 간격이 크고, 그 이웃 피어의 개수가 작을 이웃 피어로 에이전트를 이전시킬 확률을 높게 해야 한다.

```
//At each peer
if( there exist agent in this peer )
  At = current_tick - latest tick when an agent was arrived;
  if( At < α )
    delete this agent;
  else if( At > β )
    move this agent;
    if( uniform random variable R < v )
      generate new agent;
      move this agent;
  else
    move this agent;
```

(그림 1) 에이전트 생성 및 소멸 알고리즘

2.2 우수 피어 선택 알고리즘

각 피어는 에이전트가 수집해온 임의적인 정보를 이용하여 전체 피어-투-피어 시스템의 평균을 추정하고[4], 그 추정된 값을 이용하여 자신이 우수 피어인지를 판단한다. 즉, 자신의 능력이 추정된 평균에 비해서 상수( $\rho$ )배 이상이면 자신을 우수피어로 설정한다.

이 때, 각 피어는 표본 평균을 이용하여 모평균을 추정하는데, 각 피어의 표본 평균은 서로 다를 수 있으므로, 각 피어가 추정된 평균을 수정시키는 작업도 필요하다. 따라서 각 피어는 두 종류의 업데이트를 수행한다. 첫째는 다른 피어들의 능력치로 표본 평균을 구하고, 둘째는 이웃 피어와 자신의 표본 평균을 이용해서 표본 평균을 일정하게 맞춘다. 그 알고리즘은 (그림 2)에 나타내었다. 따라서 에이전트가 수집하는 정보는 각 피어의 능력치 뿐만 아니라 각 피어가 계산한 표본 평균도 포함한다.

```
// At each peer
if( uniform random variable R < ε )
  LOOP( # of collected data )
    sum += collected_peer_situation;
  Estimate_mine = sum / # of collected data;
else
  Estimate_mine =
    0.5Estimate_mine + 0.5Estimate_adjacent_peer;
```

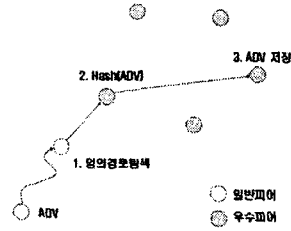
(그림 2) 우수 피어 업데이트 알고리즘

2.3 광고 및 검색 알고리즘

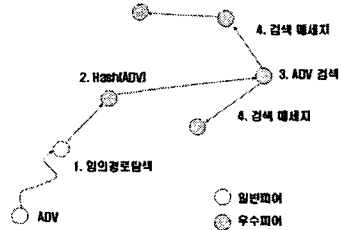
본 논문에서 제안하는 검색 및 광고 프로토콜에서는 우수 피어 링 구조를 활용한다. 광고를 하려는 피어는 광고 메시지를 생성하고 임의 검색 방식으로 임의의 우수 피어에게 전달한다. 광고 메시지를 받은 우수 피어는 해시 함수와 피어 뷰를 이용하여 광고가 저장될 우수 피어를 찾아 광고를 저장하게 한다 (그림 3 참조). 동적인 환경에서는 우수 피어들이 생성되거나 소멸되어 광고된 메시지가 사라질 수 있으므로, 주기적으로 광고할 메시지를 다시 생성한다.

검색의 경우도, 검색을 하려는 피어는 검색 메시지를 생성해서, 임의 검색 방식으로 임의의 우수 피어에게 전달하고, 메시지를 받은 우수 피어는 해시 함수로 광고가 저장되어있는 우수 피어를 찾아 검색 메시지를 전달한다. 만약 일시적인 불일치로 인하여 해당 우수 피어에 광고가 저장되어있지 않으면, 그 검색 메시지를 피어 뷰 상의 이웃 우수 피어들에게 전달하여, TTL이 0이 될 때까지

지 우수 피어 사이에 검색이 일어 나도록한다 (그림 4 참조).



(그림 3) 광고 알고리즘



(그림 4) 검색 알고리즘

2.4 피어 뷰 일정화 알고리즘

우수 피어들이 해시 함수를 사용할 때 다른 우수 피어에 대한 정보인 피어 뷰를 사용하는데, 어떤 우수 피어에서 정확한 피어 뷰를 갖고 있지 않을 경우, 광고 나 검색 과정에서 잘못된 우수 피어로 해상되므로 검색 시간이 길어진다. 따라서 정확한 피어 뷰를 갖는 것이 이 시스템의 성능에 중요한 영향을 미친다.

본 논문에서는 각 우수 피어들의 피어 뷰의 일관성을 맞추기 위해 가십(gossip) 알고리즘을 사용하였다[5]. 한 우수 피어에서 자신이 알고 있는 우수 피어의 수가 N개라고 가정하면, 일정 시간마다 자신의 피어 뷰에서  $\log(N) + c$  개의 피어 뷰를 선택하여 가십 메시지를 생성하고, 임의의  $\log(N) + c$  개의 우수 피어를 선택하여 메시지를 전달한다. 가십 메시지를 받은 우수 피어는 가십 메시지에 있는 우수 피어들의 정보를 자신의 피어 뷰로 업데이트한다. (그림 5)에서 해당 알고리즘을 나타내었다.

```
// At each super peer
size_gossip = log(size of peer view) + c;
LOOP(size_gossip)
  select peer randomly at its own peer view;
  if( selected peer is alive )
    make gossip message with selected peer information;
  else
    delete this peer information from own peer view;
LOOP(size_gossip)
  select peer randomly at its own peer view;
  send gossip message to selected peer;
```

(그림 5) 가십 메시지 생성 및 전달 알고리즘

3. 성능평가

본 장에서는 제안한 알고리즘의 성능을 한 홑 복제 기법을 이용한 임의 경로 선택 방법인 Gnutella와 정적인 우수 피어 링과 비교 하였다. 동적인 환경에서 성공 검색 메시지 수, 평균 검색시간,

그리고 오버헤드 측면에서 이들을 비교하였는데, 이를 위하여 이벤트-구동(event-driven) 시뮬레이터를 이용하였다. 시뮬레이션을 위한 파라미터는 (표 1)과 같다.

(표 1) 시뮬레이션 파라미터

Topology	Power-Law Topology
Power-law topology alpha	2.4~2.6
# of node	4000~10000
Node arrival distribution	Poisson distribution
Node online period distribution	Exponential distribution
Node capacity distribution	Normal distribution
Agent population alpha ( $\alpha$ )	2
Agent population beta ( $\beta$ )	80
Agent population gamma ( $\gamma$ )	0.1
Super peer selection row ( $\rho$ )	0.01
Query generation	Poisson distribution
Advertisement generation	Poisson distribution
TTL	32
Super peer walk TTL	5

(그림 6)에서 (그림 8)은 실험 결과를 보여주고 있다. (그림 6)의 성공 검색 메시지 수에서 제안 알고리즘이 Gnutella에 비해 월등히 좋은 성능을 보이고 있음을 확인할 수 있다. 여기서 정적인 우수 피어 링의 경우는 제안 기법보다 더 좋은 성능을 보이는데, 이는 정적인 우수 피어 링의 경우 우수 피어들의 변화가 없어 피어 뷰가 일정화 된 채로 유지되는 이상적인 상황을 가정하였기 때문이다. (그림 7)의 평균 검색시간의 경우 역시 제안 기법과 정적인 우수 피어 링이 Gnutella에 비해 우수한 성능을 보이는데, 이는 링을 구성하는 우수 피어 만을 광고나 검색에 이용하는 방법이 모든 피어를 광고나 검색에 이용하는 Gnutella에 비해 성능이 우수함을 나타낸다. 또한 제안 기법이 정적인 우수 피어 링에 비해서도 검색시간이 빠름을 볼 수 있는데, 이는 동적인 상황에서 항상 성능이 좋은 피어를 우수한 피어로 선택하기 때문이다. 한편 (그림 8)에서 제안 기법에서 월등히 오버헤드가 많은 것을 볼 수 있는데, 에이전트를 이전 하거나 우수 피어들의 피어 뷰 일정화 등을 위한 통신비용이 증가하기 때문이다. 전체적으로 제안 기법이 기존의 복제 기법에 비해 오버헤드는 많지만 검색 성공률이 월등히 높고, 검색 시간을 상당히 단축시킨다.

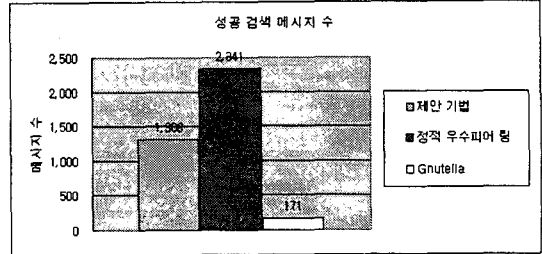
4. 결론 및 향후 과제

본 논문에서는 비구조적 피어-투-피어 시스템에서 성능이 우수한 피어들이 스스로 자기 조직적으로 우수 피어 링을 구성하고 광고 및 검색에 참여하게 함으로써 시스템의 검색 성공률 및 검색 시간을 개선하고자 하였다.

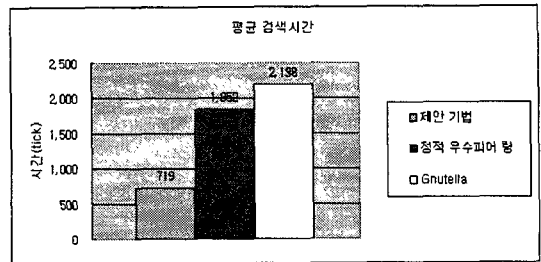
제안 기법에서는 에이전트를 이용하여 지속적으로 다른 피어들의 정보를 수집하므로, 지속적으로 피어들이 시스템을 떠나고 참여하는 동적인 상황에서 항상 성능이 우수한 피어가 우수 피어 링을 구성하게 하여 빠른 검색시간을 유지할 수 있었다. 또한 가십을 이용하여 우수 피어들의 피어 뷰를 일정화 함으로써 동적인 상황에서 우수 피어의 피어 뷰 불일치에 따르는 성능 저하를 최소화 하였다. 한편 시스템의 상황에 따라 링의 크기가 적응적으로 변하므로, 링 구조를 유지하기 위한 불필요한 오버헤드를 최소화 하였다.

본 논문에서는 시스템의 평균 성능에 비해 일정 비율 이상의 성능을 가진 피어를 우수피어로 선택하도록 하였는데, 시스템 변화의 정도에 따라 우수 피어의 비율 자체를 적응적으로 변화시킬

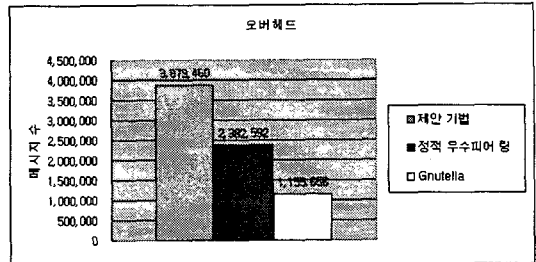
로써 시스템의 성능을 유지시키는 알고리즘의 개발이 더 필요하다.



(그림 6) 성공 메시지 수



(그림 7) 평균 검색 시간



(그림 8) 오버헤드

참고 문헌

- [1] Chawathe, Y., Ratnasamy, S., Breslau, L., Lanham N., and Shenker, S., "Making Gnutella-like P2P Systems Scalable", Proceedings of ACM SIGCOMM, 2003.
- [2] Adamic, L.A., Lukose, R.M., Puniyani A.R., and Hunezman, B.A., "Search on Power-Law Networks", Physical Rev. E, vol. 64, no. 4, 2001.
- [3] Saroiu, S., Gummadi, P.K., and Gribble, S.D.A., "Measurement Study of Peer-to-Peer File Sharing Systems". Proceedings of Multimedia Computing and Networking, 2002.
- [4] Bawa, M., Garcia-Molina, H., Gionis, A. and Motwani, R. "Estimating Aggregates on a Peer-to-Peer Network", Technical Report, Computer Science Department, Stanford University, 2003.
- [5] Ganesh, A. J., Kermarrec A.M., and Massoulié, L., "Peer-to-Peer Membership Management for Gossip-Based Protocols", IEEE Transactions on Computers, 2003.