

피어의 특성을 고려한 Magic Square 확장

송동주⁰, 정일동, 손영성[†], 김경석^{*}
부산대학교 전자계산학과⁰, 한국전자통신연구원[†], 부산대학교 정보컴퓨터공학부^{*}
{djsong⁰, idjung, ysson, gimgs}@asadal.cs.pusan.ac.kr

Enhancement Architecture of Magic Square Considering Peer's Characteristics

Dong-joo Song⁰, Il-dong Jung, Young-sung Son[†], Kyongsok Kim^{*}
Dept. of Computer Science, Pusan National University⁰, Electronics and Telecommunications Research Institute[†]
Division of Computer Science and Engineering, Pusan National University^{*}

요약

통신기술과 인터넷의 급속한 발달로 Napster와 Gnutella와 같은 P2P시스템이 등장하였다. P2P시스템은 피어간의 정보와 데이터를 직접 공유하는데 널리 이용되고 있다. MagicSquare는 동적인 (Dynamic) 환경에서 작동할 수 있도록 설계한 P2P 프로토콜이다. MagicSquare는 피어의 컴퓨팅 능력을 고려하기 때문에 각 피어에 대한 자원의 낭비를 최소화할 수 있지만, P2P 네트워크에 참여하는 피어의 물리적인 특성은 고려하지 않았다. 본 논문에서는 피어의 저장 공간의 크기, 사용할 수 있는 네트워크 대역폭, 피어가 가지는 특징을 고려하여 MagicSquare를 확장한다.

1. 서론

컴퓨터가 많이 보급되고 통신 기술이 발달하면서 인터넷을 기반으로 하는 새로운 형태의 컴퓨팅 모델인 P2P 컴퓨팅 모델이 등장하였다. P2P 컴퓨팅 모델은 정보와 자원을 직접 공유하는데 많이 적용되고 있다. P2P 모델은 자원 저장 공간과 피어의 통신비용을 피어 각자에게 분산시키고, 노드 수에 상관없이 실시간으로 직접 통신을 가능하게 하는 확장가능한 (Scalability) 을 제공한다. Gnutella와 같은 완전 분산형 P2P 컴퓨팅 모델은 중앙의 관리 시스템이 없기 때문에 피어 각각이 통신에 대해서 독자적으로 처리하기 때문에 확장가능성과 신뢰성 (Reliability) 부분의 병목 현상을 제거할 수 있는 이점이 있다 [2].

지금까지 개발한 P2P 시스템은 각 피어가 같은 성능 (중앙 처리 장치, 저장 공간의 크기, 네트워크 대역폭 등) 을 가진다고 가정하고 있다. 이 가정은 시스템의 성능을 계량화하는데 쉽고 성능 분석을 간단하게 할 수 있기 때문에 학문적인 성능 평가에 유용하다. 하지만, 현실에서는 다른 피어보다 컴퓨팅 능력이 우수한 피어가 존재하기 때문에 현실에 부합되지 않는다. 각 피어의 컴퓨팅 능력이 다른 것을 고려해서 분산 해시 테이블을 구성한다면, P2P 시스템의 가용성 (Availability), 신뢰성 (Reliability), 안전성 (Security), 성능 (Performance) 을 향상시킬 수 있다 [3].

Magic Square는 동적인 (Dynamic) 환경에서 작동할 수 있도록 설계한 P2P 프로토콜이다 [4]. Magic Square는 피어의 컴퓨팅 능력을 고려하였기 때문에 피어의 자원의 낭비를 최소화할 수 있다. 하지만, P2P 네트워크에 참여하는 피어의 물리적인 특성을 고려하지 않았다. 본 논문에서는 네트워크 대역폭과 디스크 저장 공간, 피어의 특징을 고려해서 피어를 분류하고, Magic

Square에 적용하여 자원찾는 (lookup) 프로토콜을 확장한다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구를 살펴본다. 3장에서는 컴퓨팅 능력에 따라 피어를 분류한 후, Magic Square를 확장한다. 마지막으로 4장에서는 결론과 향후 연구 과제를 제시한다.

2. 관련 연구

2.1 자원 찾기 (Lookup)

P2P 시스템에서 핵심적인 기능은 필요한 자원을 찾는 것이다. 이는 데이터를 P2P 시스템에 저장하고 그 데이터를 찾는 것을 말한다. 아래와 같이 여러 가지 형태로 자원을 찾을 수 있다.

첫째, Napster처럼 데이터의 이름과 위치를 매핑해주는 중앙의 데이터베이스를 사용하는 형태이다. 이 형태는 확장가능성이 떨어지고, 중앙 서버의 고장 (failure) 에 취약하다는 단점이 있다. 둘째, Gnutella와 같이 데이터를 찾는 방송 (broadcast) 을 해서 데이터를 가진 피어가 데이터를 필요로 하는 피어에게 제공하는 형태이다. 이 방법은 방송 메시지가 계속 전달 (forward) 되어 루프를 가지지 않도록 해야 하며, 네트워크 대역폭을 낭비하는 문제가 있다. 셋째, 분산 해시 테이블을 사용하는 것이다. 이 방법은 최근에 많이 연구되고 있으며, 각 피어에 저장된 부분 라우팅 테이블의 정보를 조합해서 데이터를 찾아가는 형태이다 [5,6,7].

분산 해시 테이블을 이용해서 P2P 시스템을 구현하면 피어에 자원을 저장하고 찾아내는 방법을 명확하게 정의해야 한다. P2P 시스템에 저장된 자원을 항상 찾을 수 있도록 피어의 특징을 적절하게 반영할 필요가 있다.

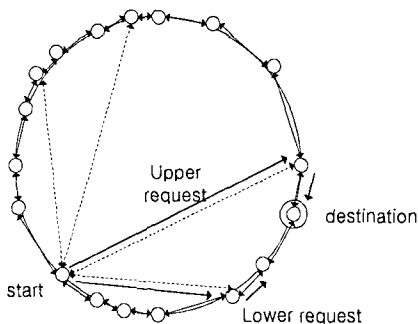
2.2 컴퓨팅 능력을 고려한 P2P 시스템

P2P 네트워크에 참여한 대부분의 피어는 1개 혹은 2개의 피어에 연결되어 있고, 일부의 피어는 대부분의 피어에 연결되어 있다. 따라서 어떤 피어에서 5개의 피어만 거치면 60% 이상의 피어에 연결할 수 있다 [8]. 다른 피어와의 연결을 많이 가진 피어를 슈퍼 피어로 볼 수 있으며, 슈퍼 피어는 인터넷에서 라우팅의 기반이 된다. 따라서, 연결을 많이 가진 피어를 적절하게 이용하면 라우팅 경로의 길이를 줄일 수 있을 것이다.

보통 슈퍼 피어는 다른 피어보다 컴퓨팅 능력 (네트워크 대역폭, 중앙 처리 장치의 성능 등) 이 뛰어나거나, P2P 시스템에 오래 접속되어 있어서 다른 피어에게 서비스를 안정적으로 제공할 수 있을 것이라고 다른 연구에서 제시되었다 [8,9]. 하지만 컴퓨팅 능력에 따라 피어를 분류하고 그 결과를 P2P 프로토콜에 반영하지 않았고 실제세계의 P2P 프로토콜에 적용된 바가 없다.

2.3 Magic Square

분산 해시 테이블을 사용한 P2P 프로토콜에 대한 이전 연구에서는 피어의 컴퓨팅 능력이 다른 것을 고려하지 않았고, 피어의 수가 늘어날수록 각 피어가 관리하는 라우팅 테이블을 유지하는 비용이 커지는 문제를 해결하지 못했다. 이 문제들을 해결하기 위해서 Magic Square라는 P2P 프로토콜을 제안하였다 [4]. [그림 1] 은 Magic Square의 동작 모습을 보인다.



[그림 1] Magic Square의 동작

Magic Square는 피어가 스킵리스트 (skiplist) 를 사용해서 라우팅 테이블을 관리하기 때문에 자원을 탐색할 경우 스킵리스트 탐색 시간과 같아진다. 따라서 자원 탐색을 스킵리스트의 시간 복잡도 (Time Complexity) 인 $\log(N)$ 개의 메시지를 사용한다. 게다가 컴퓨팅 능력이 뛰어난 피어가 다른 피어와의 연결을 더 많이 관리하고, 자원도 더 많이 저장한다. 따라서 메시지의 수가 $\log(N)$ 보다 더 적을 것으로 보인다.

표의 형태가 아니라 리스트의 형태로 라우팅 테이블을 관리하기 때문에 피어의 P2P 네트워크에 추가와 삭제가 쉽다. 따라서 라우팅 테이블 관리의 비용을 줄일 수 있을 것이다.

Magic Square는 피어의 컴퓨팅 능력을 고려하여 라우팅 테이블을 구성하지만, 다양한 네트워크 환경을 고

려하지 않았다. 본 논문에서는 다양한 네트워크 환경을 가정한다.

3. 피어의 컴퓨팅 능력을 고려한 Magic Square

3.1 문제 정의

P2P 시스템은 데스크톱 컴퓨터만으로 구성하지 않고, 여러 종류의 컴퓨터로 구성한다. 특히 컴퓨팅 능력이 좋지 못한 컴퓨터 (PDA, 핸드폰 등) 를 컴퓨팅 능력이 우수한 피어와 똑같이 취급하면 P2P 시스템의 가용성, 신뢰성, 성능을 보장하기 어렵다. 따라서, P2P 시스템에 참여하는 피어의 컴퓨팅 능력을 고려해서 시스템에서 활용해야 한다. P2P 시스템의 성능을 개선하기 위해서는 피어의 컴퓨팅 능력이나 특징에 따른 분류 규칙이 필요하다.

본 논문에서는 다음을 가정한다.

- 1) P2P 시스템에 참여하는 컴퓨터의 종류가 다양하다. (고정 네트워크 (Fixed network) 뿐만 아니라 Ad Hoc 네트워크에 연결되어 있는 피어도 참여할 수 있다.).
- 2) 속도가 느린 네트워크는 라우팅 경로가 아주 동적이거나, 피어가 네트워크에 참여하는 시간이 아주 짧다. 즉, 피어가 네트워크에 빈번한 접속단질이 발생한다.
- 3) Ad Hoc 네트워크에 연결되어 있는 피어는 Ad Hoc 네트워크의 특징을 가지기 때문에 사용할 수 있는 대역폭이 작고, 피어 간 연결이 불안정하다.
- 4) 피어의 컴퓨팅 능력은 사용가능한 네트워크 대역폭, 파일 저장 공간의 크기로 결정된다. 가까운 미래에는 대부분의 중앙 처리 장치의 성능이 아주 우수할 것으로 보이며, 모델을 간단하게 하기 위해서 논의하지 않는다.

본 논문에서는 Magic Square에 컴퓨팅 능력에 따른 피어 분류 규칙을 추가하고, Ad Hoc 네트워크에 연결되어 있는 피어의 성능을 고려하여 프로토콜을 확장한다.

3.2 컴퓨팅 능력에 따른 피어의 분류

P2P 네트워크를 인터넷에서 여러 가지 형태의 네트워크로 확장하기 위해서는 네트워크의 특징과 네트워크에 참여하는 피어의 특징을 고려해야 한다. 피어의 컴퓨팅 능력에 따라 분류하면 다음과 같다.

- ① 네트워크 속도가 느리고 저장 공간이 작은 피어

Ad Hoc 네트워크에서 작동하는 PDA나 핸드폰 등이 좋은 보기이다. 네트워크 속도가 느리기 때문에 메시지 라우팅에 참여하면 메시지를 전달하는 시간이 많이 걸린다. 라우팅에 자주 참여하지 않으면 시스템의 성능을 향상시킬 수 있을 것이다. 그리고 저장 공간이 충분치 않기 때문에 거리가 가까우면서 (피어의 ID가 비슷한) 충분한 컴퓨팅 능력을 가진 피어에 저장하는 것이 좋을 것이다. 지역성 (locality) 를 가지고 데이터를 저장하므로 데이터에 대한 요청을 다시 한 번 전달 (forward) 하는 방법이 필요하다.

- ② 네트워크 속도가 느리고 저장 공간이 큰 피어

Ad Hoc 네트워크에 참여하는 노트북이나 모델을 사용하는 데스크톱 컴퓨터가 대표적인 보기이다. 데이터를 저장하는 능력은 우수하지만 네트워크가 불안정하기 때문에 피어가 네트워크에 존재하지 않을 수가 있다. 따라서 라우팅 성능이 좋지 못할 수 있다.

③ 네트워크 속도가 빠르고 저장 공간이 작은 피어

저장 공간은 작지만 네트워크가 안정적이기 때문에 라우팅에 적극적으로 참여하는 것이 좋다. 단 저장 공간이 부족해서 인접한 다른 피어에 데이터를 저장하고 요청을 전달해서 서비스를 제공할 수 있다.

④ 네트워크 속도가 빠르고 저장 공간이 큰 피어

이제까지 가정한 피어의 일반적인 형태이며, 서비스를 제공하면서 서비스를 받는 피어이다. 이런 경우에는 개량화된 피어의 컴퓨팅 능력에 맞게 레벨을 할당한다.

3.3 피어 분류 결과를 Magic Square에 적용

앞 절에서 피어에 대해 분류한 결과를 Magic Square에 적용하면 다음과 같다.

① 네트워크 속도가 느리고 저장 공간이 작은 피어

라우팅에 자주 참여하지 않도록 레벨 1로 할당하고, 자원은 다른 피어에 저장한다. Magic Square는 분산해서 테이블을 이용해서 피어의 ID와 자원의 ID가 가까운 쪽으로 메시지를 전달하기 때문에 자원을 요구하는 메시지를 받으면 로컬 데이터베이스에 표시된 피어에 받은 메시지를 전달 (forward) 를 하는 기능을 프로토콜에 추가해야 한다. 대부분의 경우에는 컴퓨팅 능력이 우수한 피어에 자원의 복사본을 가지고 있으므로 컴퓨팅 능력이 낮은 피어에 메시지가 전달되는 경우가 거의 없을 것이다.

② 네트워크 속도가 느리고 저장 공간이 큰 피어

라우팅에 자주 참여하지 않도록 레벨 2로 할당한다.

③ 네트워크 속도가 빠르고 저장 공간이 큰 피어, 작은 피어

라우팅에 자주 참여하도록 동적으로 레벨을 할당한다. 다음은 피어의 컴퓨팅 능력을 계산하는 식이다.

$$P = c_1 BW * c_2 ST$$

(c_1 , c_2 : 상수, BW: 네트워크 대역폭, ST: 저장 공간)

4. 결론 및 향후연구과제

P2P 컴퓨팅 모델은 인터넷에서 자원을 저장하고 관리하는 비용을 피어 각각이 부담을 하기 때문에 컴퓨팅 자원을 효율적으로 사용할 수 있다. 사용자에게 만족스런 성능을 제공하기 위해 각 피어에 자원을 균일하게 저장하고 빠르게 찾을 수 있어야 한다.

Magic Square는 각 피어의 컴퓨팅 능력을 고려해서 라우팅 테이블을 구성하는 P2P 프로토콜이다. 본 논문

에서는 Magic Square에서의 피어의 저장 공간의 크기, 사용할 수 있는 네트워크 대역폭, 피어의 물리적인 특성을 고려하여 피어를 분류하였고, 그 결과를 Magic Square의 라우팅 테이블에 적용하였다. 따라서 Magic Square의 컴퓨팅 자원을 보다 효율적으로 활용할 수 있을 것으로 보인다.

본 연구는 피어의 컴퓨팅 능력을 계량화시키는 초기 단계이다. 앞으로 피어의 컴퓨팅 능력을 계량할 수 있는 일반적인 컴퓨팅 능력 측정 모델이 필요할 것으로 보이며, 성능을 평가할 수 있는 모델도 필요할 것이다.

관련연구

[1] Matei Ripeanu, "Peer-to-Peer Architecture Case Study: Gnutella Network", in proceedings of 2001 IEEE International Conference on Peer-to-Peer Computing, 2001

[2] M. K. Ramanathan, V. Kalogeraki, J. Pruyne, "Finding Good Peers in Peer-to-Peer Networks", Proc. of IPDPS'02

[3] H. Balakrishnan, M. F. Kaashoek, D. Karger, R. Morris, I. Stoica, "Looking Up Data in P2P Systems", Communications of the ACM, Feb 2003, Vol. 46, No. 2

[4] 박선미, 정일동, 손영성, 김경석, "Magic Square : 노드의 능력을 고려한 자원 탐색 프로토콜" 정보과학회 춘계학술대회(심사중), 2003

[5] I.Stoica, R.Morris, D.Karger, F.Kaashoek, and H.Balakrishnan. Chord: A scalable peer-to-peer lookup service for internet applications. In Proceedings of ACM SIGCOMM'01, Aug.2001

[6] S.Ratnasamy, P.Francis, M.Handley, R.Karp, and S. Shenker. A scalable content-addressable network. In Proceedings of ACM SIGCOMM'01. Aug.2001

[7] A. Rowstron and P.Druschel. Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems. In IFIP/ACM Middleware 2001, Heidelberg, Germany, Nov.2001

[8] R. Albert, H. Jeong, A. Barabasi, "Error and attack tolerance of complex networks", Nature406, July 2000

[9] A.Oram, "Peer-to-Peer: Harnessing the power of Disruptive Technologies", O'Reilly Books, 2001

[10] William Pugh, "Skip Lists : a probabilistic alternative to balanced Trees", communications of the ACM, 33권 6호 pp.668-676, June 1990