

# 자기 조직적 우수 피어 링 검색기법에서 입자 군집 최적화(PSO)를 이용한 적응적 우수 피어 비율 조절 기법

장 형 근<sup>†</sup> · 한 세 영<sup>††</sup> · 박 성 용<sup>†††</sup>

## 요 약

자기 조직적 우수 피어 링 검색 기법에서는 기존의 비구조적 피어-투-피어 시스템에서 성능이 우수한 피어들이 자기 조직적으로 우수 피어 링을 형성하고, 모든 피어가 키의 광고 및 검색에 이 링을 이용하게 함으로써 검색 성능을 향상 시켰다. 그러나 이 기법에서는 우수 피어의 비율을 고정된 값으로 유지하므로, 본 논문에서는 입자 군집 최적화(PSO)를 이용하여, 동적인 환경의 변화에 적응적으로 우수 피어의 비율을 최적의 값으로 변화시킴으로써, 기존의 자기 조직적 우수 피어 링 검색 기법의 성능을 더욱 향상시켰다. 시뮬레이션을 통하여 고정 우수 피어 링과 자기 조직적 우수 피어 링, 그리고 적응적 우수 피어 비율 조절 기법을 비교하여, 제안하는 기법이 검색 성공률을 월등히 향상시킴을 확인하였다.

키워드 : 입자 군집 최적화(PSO), 피어-투-피어 검색 기법, 자기 조직적 링 구조, 우수 피어

## Adaptive Control of Super Peer Ration using Particle Swarm Optimization in Self-Organizing Super Peer Ring Search Scheme

Hyunggun Jang<sup>†</sup> · Saeyoung Han<sup>††</sup> · Sungyong Park<sup>†††</sup>

### ABSTRACT

The self-organizing super peer ring P2P search scheme improves search performance of the existing unstructured peer-to-peer systems, in which super peers with high capacity constitute a ring structure and all peer in the system utilize the ring for publishing or querying their keys. In this paper, we further improves the performance of the self-organizing ring by adaptively changing its super peer ratio according to the status of the entire system. By using PSO, the optimized super peer ratio can be maintained within the system. Through simulations, we show that our self-organizing super peer ring optimized by PSO outperforms not only the fixed super peer ring but also the self-organizing super ring with fixed ratio in the aspect of query success rate.

Key Words : PSO, Peer-To-Peer Search Scheme, Self-Organizing Ring, Super Peer

### 1. 서 론

비구조적인 피어-투-피어 시스템으로는 Gnutella[1], FreeNet[2] 등이 있는데, 구조적인 피어-투-피어 시스템과 달리 위상을 구성하지 않으므로 특별한 라우팅 없이 임의의 경로 검색이나 플러딩을 이용하여 검색한다. 그러나  $O(N)$  스텝의 검색[3]을 하므로, 구조적인 피어-투-피어 시스템보다 검색 시간이 길고, 검색 성공률이 낮다. 특히 네트워크의 크기가 커질수록 성능의 저하는 현저히 드러난다. 이런 비

구조적인 피어-투-피어 시스템의 문제점을 개선하기 위해서 우수 피어를 이용한 계층적 피어-투-피어 시스템이 연구되어 왔다. 이러한 계층적 피어-투-피어 시스템으로는 KaZaA[4], JXTA[5, 6] 등이 있다. 계층적 피어-투-피어 시스템에서 우수 피어는 중앙 집중 서버처럼 동작하며 다른 우수 피어들과 연결하여 우수 피어들만의 네트워크를 이룬다. KaZaA나 Gnutella에서는 우수 피어들이 플러딩이나 임의의 경로 탐색, 복제 등을 이용하여 검색을 하는 반면, JXTA 등에서는 우수 피어들이 느슨하게 결합된 링 구조의 네트워크를 이루고 해시 함수를 이용하여 검색을 한다. 계층적 피어-투-피어 시스템에서 일반 피어는 한개 이상의 우수 피어와 연결이 되어 있으며 검색 시 우수 피어에게 메시지를 전달한다. 즉, 모든 검색은 어떤 형태이든 우수 피어들로 이루어

\* 이 논문은 2006년도 두뇌한국21사업에 의하여 지원되었음.

† 정 회 원 : (주)마로비전 뉴미디어연구소

†† 정 회 원 : 서강대학교 컴퓨터학과 박사과정

††† 정 회 원 : 서강대학교 컴퓨터학과 부교수

논문접수: 2006년 6월 1일, 심사완료: 2006년 9월 6일

어진 우수 피어 네트워크상에서 이루어진다.

이러한 계층적 피어-투-피어 시스템에서는 시스템 내의 전체 피어의 수에 비해 우수 피어 수의 비율이 어떻게 되는지, 그 비율을 어떻게 유지할 것인가가 중요한 문제이고, 이는 동적인 환경에서 전체 시스템의 성능에 큰 영향을 미친다[7]. 따라서 자기 조직적 우수 피어 링 검색기법[8]에서는 자기 조직적으로 우수 피어 링을 구성하고, 일정한 우수 피어 비율이 유지될 수 있도록 피어-투-피어 시스템의 크기에 따라 우수 피어의 개수가 적응적으로 늘어나거나 줄어든다. 또한 에이전트를 이용하여 주변 피어들의 정보를 수집하여 상대적으로 높은 능력을 가진 피어가 우수 피어가 되도록 함으로써, 시스템의 검색 능력을 효율적으로 유지할 수 있게 한다.

그러나 [9]에 의하면 피어-투-피어 시스템에서 시간에 따른 트래픽 및 이용자의 변화는 (그림 1)과 같이 참여와 떠남의 비율이 시간에 따라 달라지는 동적인 환경임을 알 수 있다. 이와 같은 변화가 잦은 환경에서 고정된 우수 피어의 비율을 유지하기 위해서는 우수 피어들 자체에 잦은 변화가 있게 되고 이는 피어뷰의 잦은 불일치를 야기 하여 검색 성공률이 낮아지게 한다. 이러한 부분을 개선하기 위하여 본 논문에서는 입자 군집 최적화(PSO)[10]를 이용하여 환경의 변화에 따른 적절한 우수 피어의 비율을 찾아 적용하고, 결과적으로 우수 피어의 변화를 억제함으로써 전체 피어-투-피어 시스템의 검색 성능을 향상시키고자 한다.

입자 군집 최적화(Particle Swarm Optimization, 이하 PSO)는 군집을 이루는 각 개체의 행동을 모방함으로써 큰 검색 공간에서 최적 또는 최적에 가까운 해를 찾는 효과적인 방법으로 알려져 있다[10]. 각 개체는 자신이 과거에 가졌던 값 중 가장 좋은 값을 저장하고, 이와 함께 모든 개체는 전체 개체들이 과거에 가졌던 값 중 가장 좋은 값을 공유한다. 각 개체는 이런 값들을 이용하여 다음 동작의 방향을 정한다. 한편, PSO는 일반적으로 목표 값이 고정되어 있지 않고 동적으로 변하는 상황에서는 최적 값을 잘 찾지 못하는 것으로 알려져 있으나, 각 개체가 저장하는 가장 좋은 값을 주기적으로 삭제하여 목표 값이 동적으로 변하는 상황에 적용시킬 수 있다[11]. 따라서 동적인 환경에 적응적이고,

통신비용을 최소화하며, 지역 정보만을 이용하여 자기 조직적 우수 피어 링의 적절한 우수 피어 비율을 찾기 위하여 이러한 PSO 알고리즘을 사용하였다.

제안기법의 성능을 검증하기 위해서 고정된 우수 피어를 사용하는 알고리즘과 정적인 우수 피어의 비율을 갖는 알고리즘을 성공한 검색 메시지 개수, 평균 검색 시간, 그리고 오버헤드 측면에서 제안 알고리즘과 비교하였다. 또한 PSO 파라미터가 시스템의 검색 성능에 미치는 영향을 보이기 위하여 파라미터 변화에 따른 성능 변화를 분석하였다.

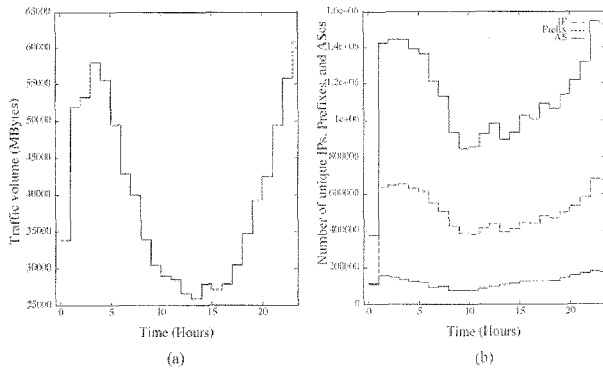
본 논문의 구성은 다음과 같다. 2장에서는 고정된 우수 피어 비율을 가진 자기 조직적 우수 피어 링 검색기법을 소개하고, PSO 기법에 대해 설명한다. 3장에서는 자기 조직적 우수 피어 링에서 PSO를 이용한 적응적 우수 피어 비율 조절 기법을 설명한다. 4장에서는 시뮬레이션을 통해 기존의 고정된 우수 피어 링과 고정 비율의 우수 피어 링을 제안하는 기법과 성능을 비교, 분석하고, PSO 파라미터가 시스템의 검색 성능에 미치는 영향을 확인한다. 마지막으로 5장에서는 논문의 결론과 향후 과제에 대해 논의한다.

## 2. 관련 연구

### 2.1 자기 조직적 우수 피어 링 검색기법

자기 조직적 우수 피어 링 검색기법에서는 시스템의 변화에 따라 우수 피어의 개수도 적응적으로 변화하여 기존의 고정된 우수 피어 링 알고리즘의 문제점을 보완한다. 피어-투-피어 시스템에서 각 피어는 지역적 정보만을 가지고 있으므로 에이전트를 이용하여 다른 피어의 정보를 수집하고, 수집해온 정보를 이용하여 자신이 우수 피어인지를 결정하며, 우수 피어로 판단되면 우수 피어들이 형성한 링 구조에 참여하게 된다[12]. 시스템에 계속 새로운 피어가 참여하거나 떠남이 일어남에 따라 계속 상대적으로 성능이 우수한 피어가 우수 피어로서 링을 구성하게 되므로, 우수 피어 링 구조는 시스템의 크기와 피어들의 능력 분포에 따라 적응적으로 그 크기가 변한다.

한편, 각 우수 피어는 가십 프로토콜(Gossip protocol)을 이용하여 다른 우수 피어에 대한 정보인 피어뷰(peer view)를 다른 우수 피어들과 동기화(synchronization)한다[13]. 따라서 한 피어가 자신의 데이터를 광고하고자 하는 경우, 광고 메시지를 생성하고 임의의 경로 검색 통해 임의의 우수 피어에게 그 메시지를 전달하면, 그 메시지를 받은 우수 피어가 해시 함수와 피어뷰를 이용하여 해당 광고가 저장되어야 할 우수 피어를 찾아 메시지를 전달하고, 해당 우수 피어에서는 광고를 저장하게 된다. 검색 요청에 대해서도 비슷한 과정을 거치게 된다. 즉, 검색 요청을 할 피어는 검색 메시지 생성하여 임의의 경로 검색을 통해 임의의 우수 피어에게 전달하고, 그 우수 피어 역시 해시 함수와 피어뷰를 이용하여 그 광고를 저장하고 있을 것으로 추정되는 우수 피어를 찾아 검색 메시지를 전달한다. 검색 메시지를 받은 해당 우수 피어는 자신이 저장하고 있는 정보들 중 요청한 정보가



(그림 1) 시간에 따른 피어-투-피어 시스템의 트래픽 및 이용자 변화

있으면 검색을 요청한 피어에게 응답 한다. 만일 동적인 환경에서 피어뷰가 일시적인 불일치 상태에 있어 검색 요청을 받은 우수 피어가 해당 정보를 가지고 있지 않다면, 링 구조상에서 양방향의 이웃 우수 피어에게 검색 요청을 전달하는 우수 피어 경로 검색(super peer walk)을 수행하며, 이는 우수 피어 TTL이 0이 될 때까지 진행된다.

자기 조직적 우수 피어 링 구조는 지역 정보를 이용하여 동적인 환경에 대하여 우수 피어의 개수가 적응적으로 변하게 하였다. 하지만 전체 피어의 개수에 대한 우수 피어 개수의 비율이 동적인 환경에 최적화 되지 못하고 정적으로 고정되어 있다. 이 경우 피어-투-피어 시스템에 피어의 참여나 떠남이 지속적으로 많은 경우 우수 피어의 비율이 정해져 있기 때문에 우수 피어의 개수 변동이 심해지고, 이는 피어뷰 불일치에 따르는 시스템 성능의 저하는 유발한다.

따라서 다음 장에서 동적인 환경에서 최적의 우수 피어 비율을 결정하고 그에 따라 우수 피어를 결정하는 적응적 우수 피어 비율 기법을 설명하고자 한다. 이 기법에서는 결과적으로 시스템의 변동에 따르는 우수 피어의 변동을 지연 시킴으로써 전체 시스템의 검색 성능을 향상 시키고자 한다.

### 2.2 PSO(Particle Swam Optimization)

PSO는 효율적으로 최적 또는 최적 근처의 해를 찾는 방법으로 알려져 있다[10]. PSO는 모든 개체가 검색 공간  $X$ 와 목적 함수  $f: X \rightarrow R$ 에 대하여

$$x^* \rightarrow \min_{x \in X} f(x) \tag{식 1}$$

인  $x^*$ 를 찾는 것을 목표로 한다. 이를 위해 다차원 검색 공간에서 개체들이 사회적 행동을 통해서 해를 찾는다.  $D$ -차원 검색 공간에서 개체  $i$ 는  $X_i = (x_{i1}, x_{i2}, \dots, x_{iD})$ 로 표현된다. 개체  $i$ 가 갖는 각 차원에 대한 속도는  $V_i = (v_{i1}, v_{i2}, \dots, v_{iD})$ 로 표현된다. 각 개체는 자신의 과거 위치 중 가장 좋은 위치  $P_i = (p_{i1}, p_{i2}, \dots, p_{iD})$ 를 저장한다. 또한 모든 개체는 전체 개체가 과거에 가졌던 위치 중 가장 좋은 위치  $P_g = (p_{g1}, p_{g2}, \dots, p_{gD})$ 를 공유한다. 개체  $i$ 는 현재의 위치  $X_i$ 와 자신의 가장 좋았던 위치  $P_i$ , 전체 개체의 가장 좋았던 위치  $P_g$ 를 통합하여, <식 2>와 <식 3>과 같이 다음 단계의 속도와 위치를 조절한다.

$$v_{id} = \omega \times v_{id} + \eta_1 \cdot rand() \cdot (p_{id} - x_{id}) + \eta_2 \cdot rand() \cdot (p_{gd} - x_{id}) \tag{식 2}$$

$$x_{id} = x_{id} + v_{id} \tag{식 3}$$

이 때, 상수  $\eta_1, \eta_2$ 는 개체 자신의 의견과 개체 전체의 의견 중 어떤 것에 더 큰 비중을 둘 것인가에 의해 정해진다. 반복을 통해 모든 개체는  $x^*$ 를 찾아가게 된다.

풀고자 하는 문제를 적용하고자 하는 곳은 동적인 환경의 피어-투-피어 시스템이다. 이것은 목적 함수  $f$ 가 끊임없이 변하여 찾고자 하는  $x^*$ 이 끊임없이 변함을 뜻한다. 따라서 과거의 가장 좋은 위치가 현재 찾고자 하는  $x^*$ 와 연관이 없

다. 원래 PSO는 이런 변화를 빠르게 탐지할 수 없으나, 이러한 문제에 대해 [11]에서는 과거의 경험을 기억하지 않음으로써 동적인 환경에 적응하게 하였다. 즉 과거의 가장 좋은 위치를 나타내던 벡터  $P$ 를 주기적으로 현재의 위치인 벡터  $X$ 로 대체하여 새로운 목적 함수  $f$ 에 대한 새로운  $x^*$ 을 찾게 한다.

다음 장에서 이와 같은 PSO를 자기 조직적 우수 피어 링 검색기법에 적용하는 방법을 설명하고, 그를 이용하여 우수 피어의 비율을 조절하는 기법에 대하여 설명하고자 한다.

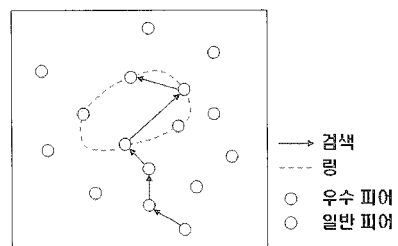
### 3. PSO를 이용한 적응적 우수 피어 비율 기법

#### 3.1 PSO의 적용

자기 조직적 우수 피어 링 검색 기법을 사용하는 피어-투-피어 시스템에서 각 우수 피어는 각자 피어뷰(peer view)를 유지한다. 피어뷰란 각 우수 피어가 알고 있는 다른 우수 피어들을 식별자 순으로 정렬한 우수 피어의 목록이다. 피어뷰는 우수 피어들 사이에 가십 프로토콜을 이용해 일정화 된다.

우수 피어를 이용한 검색은 (그림 2)와 같다. 즉, 검색 메시지  $q$ 는 임의의 경로 탐색(random walk)을 통해 임의의 우수 피어  $i$ 까지 도달한 후  $i$ 의 피어뷰 중 해시 함수를 통해 우수 피어  $j$ 로 전달된다. 그러나 피어뷰가 일정화 되지 못한 경우 검색 메시지에 대한 광고가 실제로  $j$ 에 없을 수 있다. 이 경우  $j$ 는 자신의 이웃 우수 피어인  $j-1$ 과  $j+1$ 에게  $q$ 를 전달한다.  $j-1$ 이나  $j+1$ 의 경우도 자신에게  $q$ 에 대한 광고가 없는 경우 TTL이 0이 아닌 동안 자신의 이웃에게  $q$ 를 전달한다.

이때 일반 피어에서 생성된 검색 메시지가 우수 피어에 도달할 때까지의 홉 수를  $d_n$ 이라 하고 우수 피어 링 상에서 해당 광고를 찾을 때까지의 홉 수를  $d_s$ 라 하면, (그림 2)의 예에서  $d_n$ 은 3,  $d_s$ 는 2가 된다. 이러한 우수 피어 링을 이용한 검색 기법의 성능을 높이기 위해서는  $d_n$ 과  $d_s$ 를 줄이는 것이 중요하다.  $d_n$ 을 줄이기 위해서는 우수 피어의 개수가 많아져야 한다. 그러나 이 경우 우수 피어간의 피어뷰가 불일치하기 쉬우므로  $d_s$ 는 오히려 늘어나는 경향이 있다. 본 연구에서는 시스템 전체의 평균  $d_n$ 과  $d_s$ 를 줄임으로써 검색 성능을 높이고자 한다. 즉 우수 피어의 개수가  $N$  개이고 우수 피어  $i$ 가 알고 있는  $d_{ni}, d_{si}$ 를  $d_{ni}, d_{si}$ 라 할 때, 다음의 <식 4>을 최소화해야 한다.



(그림 2) 검색 홉 수

$$\frac{1}{N} \sum_{i=0}^N (\alpha \cdot d_m + \beta \cdot d_{is}) \quad <식 4>$$

이를 위해 PSO의 스키마를 이용하였다.

적응적 우수 피어 비율 조절 기법은 개체 집합  $X = \{x_1, x_2, \dots, x_N\}$ ,  $x_i = \{d_m, d_{is}\}$ 와 목적 함수 <식 5>를 갖는 PSO 문제로 볼 수 있다.

$$f(x_i) = \alpha \cdot d_m + \beta \cdot d_{is} \quad <식 5>$$

각 우수 피어가 개체에 해당하고, 우수 피어  $i$ 의 위치  $x_i$ 는 우수 피어  $i$ 가 알고 있는  $d_m, d_{is}$ 인 ( $d_m, d_{is}$ )로 정의한다. 이 때  $d_m, d_{is}$ 는 우수 피어  $i$ 가 최근 수집한  $k$ 개의 검색 메시지로부터 구한 평균값이다. 목적 함수의 값은 각 개체가 알고 있는  $d_m, d_{is}$ 에 의해 결정되는데, 이때  $d_m, d_{is}$ 는 자신이 가지고 있는 우수 피어 비율  $\rho_i$ 에 대한 함수로 나타낼 수 있다. 즉 개체  $i$ 의  $d_m, d_{is}$ 는 다음과 같이 나타낼 수 있다.

$$d_m = f_{1i}(\rho_i)$$

$$d_{is} = f_{2i}(\rho_i)$$

동적인 네트워크 환경에서  $f_1, f_2$ 는 끊임없이 변하므로, 같은 우수 피어 비율을 가졌다 하더라도 다른  $d_m, d_{is}$  값을 가질 수 있다. 목적 함수는  $f = \alpha \cdot f_1 + \beta \cdot f_2$  이므로 이 또한 끊임없이 변화한다. 피어-투-피어 시스템에서  $f_1, f_2$ 는 정의하기가 쉽지 않지만 검색 메시지를 통한  $d_m, d_{is}$ 를 알 수 있으므로  $f_1(\rho), f_2(\rho)$ 의 값을 알 수 있다.

$p_i$ 는 우수 피어  $i$ 가 최근 1번 내에 가졌던 가장 좋은 위치로 정의하며  $p_i$ 의 위치를 나타내는  $d_m, d_{is}$ 는  $p_m, p_{is}$ 로 나타낸다. 본래의 PSO와 달리 과거의 경험 중 가장 좋은 위치가 아닌 1번 이내의 값으로 제한하는 것은 목적 함수가 끊임없이 변화하는 동적인 환경이기 때문이다.

$p_g$ 는 전체 개체가 가진 가장 좋은 위치를 뜻한다. 이 값은 각 우수 피어가 지역 정보를 이용하여 추정한다. 따라서 우수 피어  $i$ 가 알고 있는  $p_g$ 는  $p_{ig}$ 로 나타낸다. 우수 피어  $i$ 는 자신이 추정한 가장 좋은 위치를 다른 피어와 교환하여  $p_{ig}$ 를 업데이트 한다.  $p_{ig}$ 는 주기적으로  $p_i$ 로 대체된다. 이때  $p_{ig}$ 의 위치를 나타내는  $d_m, d_{is}$ 는  $p_{igm}, p_{igs}$ 로 나타낸다.

$v_i$ 는 개체  $i$ 의 위치를 변화시키는 속도이다. 그러므로 본문에서는 우수 피어  $i$ 의 위치 즉 ( $d_m, d_{is}$ )를 변화시키는 우수 피어의 비율  $\rho_i$ 로 정의 된다.

본래의 PSO에서는 각 개체의 위치  $x_i$ 는 <식 3>에 의해 정해진다. 즉 바로 직전의 위치에서 당시 속도를 더하여 새로운 위치를 구한다. 그러나 본 문제에서 각 개체는 자신이 구한 속도에 대한 피드백(feed-back)을 바로 받을 수 없고, 뿐만 아니라 자신이 구한 속도는 다른 개체의 위치에도 영향을 끼친다. 따라서 각 개체의 위치는 <식 3>에 의해 구할 수 없고, 정보 수집을 통해 얻은 ( $d_m, d_{is}$ )로 정의한다.

우수 피어를 이용한 비구조적 피어-투-피어 시스템에서 적응적으로 우수 피어의 비율을 조절하기 위하여 다음과 같

이 PSO가 적용된다.

$$v_i = \omega \times v_i + \eta_1 \cdot rand() \cdot (p_i - x_i) + \eta_2 \cdot rand() \cdot (p_{ig} - x_i) \quad <식 6>$$

$$p_i - x_i = \alpha \cdot (p_m - d_m) + \beta \cdot (p_{is} - d_{is}) \quad <식 7>$$

$$p_{ig} - x_i = \alpha \cdot (p_{igm} - d_m) + \beta \cdot (p_{igs} - d_{is}) \quad <식 8>$$

### 3.2 우수 피어 비율 조절 기법

본 논문에서는 기존의 자기 조직적 우수 피어 링 검색 기법에서 검색 메시지의 흡 수를 이용하여 우수 피어의 비율을 조절하고자 하는데, 이를 위해 크게 다음의 세 가지 과정을 거친다.

#### ◦ 자료수집 및 확산

우수 피어의 비율을 정하기 위해 기존에 수집한 자료 외에 검색 메시지의  $d_m, d_{is}$ 에 대한 자료 수집이 필요하다. 이는 검색 메시지를 통해 이루어지며 이를 통해 각 우수 피어는  $x_i, p_i, p_{ig}$ 를 추정한다. 이러한 추정을 더욱 정확히 하기 위해 우수 피어 간에 가십 프로토콜을 통해 피어뷰를 교환할 때 이 정보도 교환한다. 또한 새로 정해진 비율을 확산시키기 위해 기존의 에이전트를 이용한다.

#### ◦ 적응적 우수 피어 비율 계산

우수 피어는 수집된 정보를 바탕으로 PSO를 이용하여 새로운 우수 피어 비율을 계산한다. PSO가 동적인 환경에 적용하기 위해서는 과거의 경험을 주기적으로 삭제하는 기법을 이용한다.

#### ◦ 우수 피어의 선택과 강등

각 피어는 최근에 업데이트된 우수 피어 비율에 의해 자신의 능력치를 고려하여 자신이 스스로 우수 피어가 될 것인지 일반 피어가 될 것인지를 정한다.

#### 3.2.1 자료 수집 및 확산

본 논문에서 제안한 개선된 비구조적인 피어-투-피어 시스템에서는 세 가지 형태로 정보를 교환한다. 첫째 전체 피어에 대하여 에이전트들이 이전(migration) 되면서 피어들의 정보를 모으거나 일정화된 정보를 확산하는 역할을 한다. 둘째 검색 메시지를 통해 우수 피어는  $d_m, d_{is}$  정보를 업데이트 한다. 셋째 각 우수 피어는 가십 메시지를 통해 자신이 가진 피어뷰와 함께 자신이 수집한  $d_m, d_{is}$ 에 대한 정보를 다른 우수 피어와 교환한다.

에이전트는 피어-투-피어 시스템에서 피어들의 상태를 수집하는 도구로 사용된다. 즉, 피어-투-피어 시스템에 에이전트들이 이전되면서 피어들의 능력치에 대한 정보를 모으고, 또한 모든 피어들의 정보를 현재 이전된 피어에게 알려주는 역할을 한다. 각 피어는 이러한 지역적인 정보를 통해 전체 시스템의 능력치를 추정하기 위해 자기 조직적 링 구조 기법의 추정 이론을 그대로 사용하였다. 에이전트는 능력치에 대한 자료 수집 외에 우수 피어들이 계산한 새로운

우수 피어 비율을 확산하는 역할을 한다. 즉, 에이전트는 이전 된 피어에서 자신이 가진 우수 피어 비율의 업데이트 시간과 그 피어가 가진 우수 피어 비율의 업데이트 시간을 비교하여, 만약 자신이 가진 우수 피어 비율의 업데이트 시간이 더 최근이라면 피어에게 새로운 우수 피어 비율을 업데이트 시킨다. 그 반대라면 에이전트는 피어로부터 새로운 우수 피어의 비율을 업데이트 받는다.

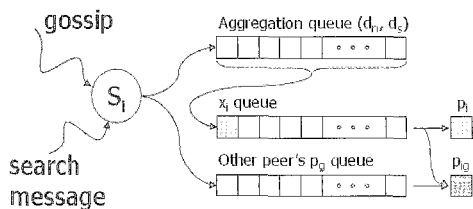
우수 피어는 새로운 우수 피어의 비율을 계산하기 위해  $d_n, d_s$ 에 대한 정보를 수집해야 한다. 이를 위해 검색 메시지를 이용한다. 일반 피어에서 생성된 검색 메시지는 임의 경로 탐색을 통해 우수 피어를 찾을 때까지의 홉 수  $d_n$ 을 업데이트한다. 우수 피어에 도착한 검색 메시지는 우수 피어 상에서 이동하는 동안 방문하는 모든 우수 피어에게  $d_n$ 을 업데이트한다. 이후 검색 메시지가 우수 피어 상에서 검색이 성공할 때까지의 홉 수  $d_s$ 를 구한다.  $d_s$ 는 검색 메시지가 찾고자 하는 광고가 있는 우수 피어에 업데이트된다.

우수 피어 간에는 가십을 통해서 피어뷰를 일정화 하는데, 이 때 가십 메시지에 각 피어의  $x_i$ 와  $p_{ij}$ 를 포함시켜 일정화 한다. 이를 통해 우수 피어는 더 정확한  $d_n, d_s$  및  $p_{ij}$ 를 추정 할 수 있다.

### 3.2.2 적응적 우수 피어의 비율 계산

본 문제의 가장 큰 목적은 PSO를 이용하여 우수 피어의 비율을 구하는 것이다. 환경을 반영한 새로운 우수 피어의 비율은 PSO의 개체인 우수 피어 상에서 계산된다. 이를 위해 각 개체는  $p_i, x_i, p_{ij}$  값을 정하는 것이 필요하다.

우수 피어  $i$ 는 검색 메시지와 가십으로 수집한 정보를 통해  $x_i$ 를 구한다. 검색 메시지가 우수 피어를 방문하며 업데이트한  $d_n, d_s$ 은 시간 순으로 최고  $k$ 개까지 저장된다. 이들의 평균을 우수 피어의 위치  $x_i$ 로 삼는다. 이때의 위치는 각 우수 피어 상에 저장되며 그 중 최근  $l$ 개의 위치 중 가장 좋은 위치를  $p_i$ 로 삼는다.  $p_{ij}$ 는 가십을 통해 받은 정보를 기반으로 각 우수 피어가 각자 유추한다. 그러므로 실제 우수 피어  $i$ 가 알고 있는  $p_{ij}$ 는 우수 피어  $j$ 가 알고 있는  $p_j$ 인  $p_{ij}$ 와 다르다. 우수 피어  $i$ 는 가십을 통해 자신의 위치  $x_i$ 와 자신이 유추한 시스템의 가장 좋은 위치  $p_{ij}$ 를 다른 우수 피어에게 전한다. 마찬가지로 다른 우수 피어로부터 그들의 위치와 그들이 유추한 시스템의 가장 좋은 위치를 업데이트 받는다. 우수 피어  $i$ 는 가십을 통해 얻은 정보로부터 새로운  $p_{ij}$ 를 유추한다.  $p_{ij}$ 는 (그림 3)과 (그림 4)에서 나타난 바와 같이 주기적으로  $p_i$ 로 대체된다. 이렇게 업데이트



(그림 3) 정보 업데이트

```

if( uniform random variable < ε )
    Pg = Pi;
else
    LOOP(# of gossip data)
        Pg = min(Pg, gossip_Pg);
    
```

(그림 4)  $P_g$  업데이트 알고리즘

```

super_peer_percentage = α * super_peer_percentage
    + η1 * rand() * position_to_cost(Pi, Xi)
    + η2 * rand() * position_to_cost(Pg, Xi)
percent_update_time = tick
    
```

(그림 5) 우수 피어 비율 업데이트

```

return α * (a.dn - b.dn) + β * (a.ds - b.ds)
    
```

(그림 6) position\_to\_cost(a, b)

된 데이터를 통해 각 우수 피어는 (그림 5)와 (그림 6)에서 나타난 것과 같이 <식 6>, <식 7>, 그리고 <식 8>을 이용해 새로운 우수 피어 비율을 계산한다. 새로운 우수 피어의 비율은 에이전트를 통해서 일반 피어에게 확산된다.

### 3.2.3 우수 피어의 선택과 강등

우수 피어의 선택과 강등을 위해서 경계 변수인 *threshold\_variable*을 이용한다. *threshold\_variable*은 각 피어가 알고 있는 시스템 전체의 능력치에 대한 표본 평균과 우수 피어 비율  $\rho$ 에 의해 정해진다. 각 피어는 자신의 능력치가 *threshold\_variable*보다 크면 *selected\_count variable*을 증가시키고, 그 값이 *SELECTION\_THRESHOLD*를 넘게 되면 자신을 우수 피어로 선택한다. 반대로 우수 피어의 능력치가 *threshold\_variable* 보다 크지 않으면 *selected\_count variable*을 감소시키고, 그 값이 *DEMOTION\_THRESHOLD* 보다 작아지면 일반 피어로 강등 된다. 우수 피어의 선택과 강등 알고리즘은 다음의 (그림 7)에서 나타내었다.

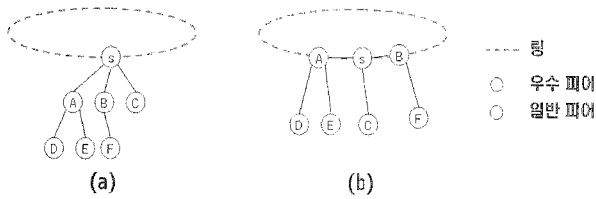
피어가 우수 피어로 선택되면 참여 메시지를 보낸다. 이 메시지는 임의 경로 탐색을 통해 가까운 우수 피어에게 전달된다. 참여 메시지를 받은 우수 피어는 해당 피어를 자신의 피어뷰에 추가 하고 해당 피어에게 자신이 갖고 있는 피어뷰를 전달한다. 우수 피어들은 가십을 통해서 새로운 우수 피어의 존재를 피어뷰에 추가한다. (그림 8)은 일반 피어가 우수 피어로 선택 된 후의 모습을 보여준다.

```

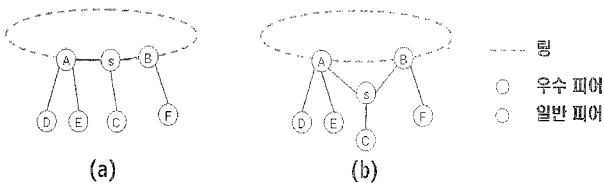
if (node_situation > threshold_variable)
    selected_count++;
else
    selected_count--;
if (node.RDV == TRUE)
    if (selected_count < DEMOTION_THRESHOLD)
        demotion();
else
    if (selected_count > SELECTION_THRESHOLD)
        selection();
    
```

(그림 7) 우수 피어 선택 및 강등

또한 우수 피어가 일반 피어로 강등되면 우수 피어뷰와 자신이 가지고 있는 광고 메시지에 대한 정보를 모두 삭제한다. 피어뷰 외의 간선에 대해선 그대로 유지하기 때문에 일반 피어로 강등되었다 해도 새로운 간선을 만들지 않는다. (그림 9)는 우수 피어가 일반 피어로 강등 되는 모습을 보여준다.



(그림 8) 우수 피어 선택 (a) 선택 전, (b) 피어 a,b 선택 후



(그림 9) 우수 피어의 강등 (a) 강등 전, (b) 피어 s 강등 후

#### 4. 성능평가

본 장에서는 제안한 알고리즘의 성능을 성공한 검색 메시지 수, 평균 검색 시간, 그리고 오버헤드의 측면에서 고정된 우수 피어 링과 자기 조직적 우수 피어 링(우수 피어 비율 고정)과 비교 하였다. 또한 PSO 파라미터가 시스템의 성능에 어떻게 영향을 미치는지를 실험을 통해 살펴보았다. 이를 위하여 이벤트-구동(event-driven) 시뮬레이터를 이용하였다.

##### 4.1 시뮬레이션 환경

본 실험에서는 이벤트-구동 시뮬레이터를 구현하여 사용하였는데, 각 시뮬레이션 타임에 시뮬레이션 되는 모든 피어들은 자신의 이벤트 큐를 확인하고 발생된 이벤트에 맞는 동작을 수행한 후 이벤트를 생성하고 다음 시뮬레이션 타임으로 진행한다.

한편, 비구조적인 피어-투-피어의 네트워크 토폴로지는 파워-로우 토폴로지(power-law topology)로 되어있다고 연구되어 있으므로, 그에 따라 토폴로지를 구성하였다[14]. 또한, 기존의 연구[3]에 따르면 비구조적인 피어-투-피어 시스템에 참여하는 각 피어의 성능은 서로 다르므로, 본 시뮬레이션에서는 각 피어의 능력치를 [15]에 따라 <표 1>과 같이 설정하고, 피어의 능력에 따라 검색 메시지를 처리하는 시간을 다르게 두었다. 실험을 위한 기타 환경 및 PSO 파라미터는 <표 2>에 나타내었고, 나머지 에이전트 관련 파라미터는 [8]에서와 같다.

<표 1> 피어의 능력치 분산

Capacity level	percentage of nodes
1X	10%
10X	55%
100X	30%
10000X	5%

<표 2> 시뮬레이션 파라미터

Topology	Power-Law Topology
Power-law topology $\alpha$	2.4~2.6
# of node	15000~32000
Node arrival distribution	Poisson distribution
Node online time distribution	Exponential distribution
Contents Distribution	Uniform Distribution
Query generation distribution	Uniform distribution
Advertisement generation distribution	Uniform distribution
TTL	20
TTL over super peer ring	5
PSO $\omega$	1
PSO $\alpha$	0.1
PSO $\beta$	0.18
PSO $\eta_1$	0.0002
PSO $\eta_2$	0.0004

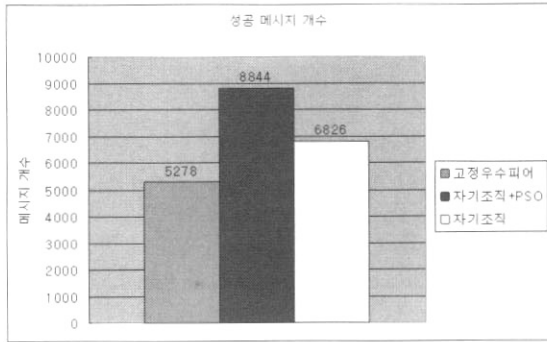
##### 4.2 성능 측정 및 분석

본 절에서는 동적인 환경에서 제안 기법을 통해 우수 피어의 비율이 어떻게 변하는지를 살펴보고 이에 대한 성능측정을 하였다. 피어-투-피어 시스템이 피어의 참여와 떠남이 있는 상황에서 고정된 우수 피어를 사용하는 알고리즘과 자기 조직적 우수 피어 링 그리고 제안 알고리즘인 PSO를 이용한 자기 조직적 링 구조를 비교하였다. 또한 PSO의 파라미터인  $\alpha, \beta$ 의 값에 따라 시스템이 어떤 모습을 갖는지 비교해 보았다.

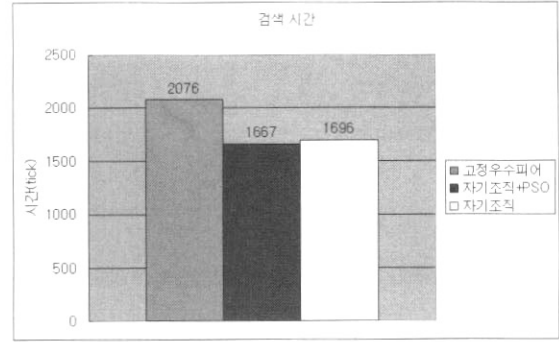
###### 4.2.1 동적인 환경에서의 성능 측정

(그림 10)의 성공한 검색 메시지 수 그래프에서, PSO를 이용한 자기 조직적 우수 피어 링이 고정된 우수 피어 비율의 링보다 약 30% 정도 성능이 향상되었음을 볼 수 있다. 피어의 개수 변화가 (그림 13)과 같이 변할 때, 제안하는 알고리즘에서 우수 피어의 비율은 (그림 14)에서와 같이 시스템의 변화 속도에 따라 우수 피어의 비율이 적응적으로 변한다. 즉, PSO를 사용하는 우수 피어 링의 경우는 시스템의 변화를 바로 우수 피어 개수에 반영하지 않고, 과거에 가장 성능이 좋았던 우수 피어 비율을 저장하였다가 한동안 적용함으로써, 결과적으로 (그림 15)에서와 같이 우수 피어 개수가 증가하는 속도를 고정된 비율의 우수 피어 링에 비해 상대적으로 늦춘다. 따라서 피어뷰의 불일치 정도가 상대적으로 적을 것이고, 이는 성공률 측면에서 유리하다.

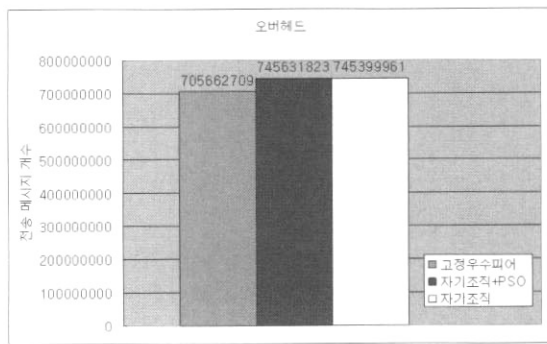
한편, 우수 피어의 비율의 조절이 토폴로지의 구성에는



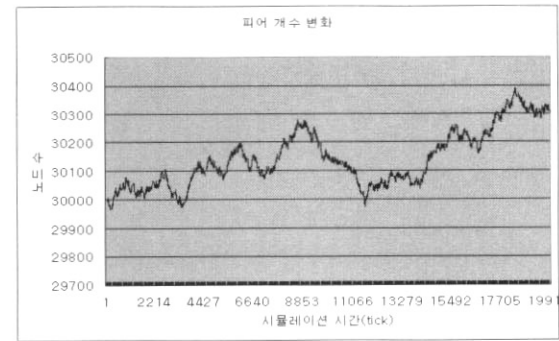
(그림 10) 성공 메시지 개수



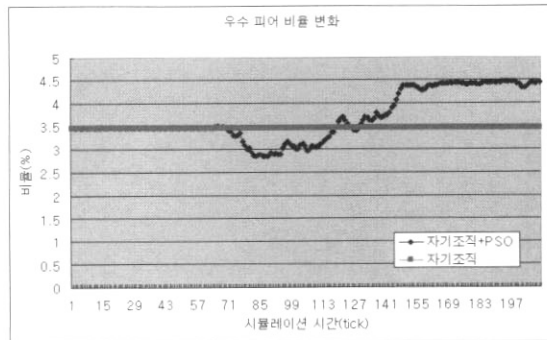
(그림 11) 검색 시간



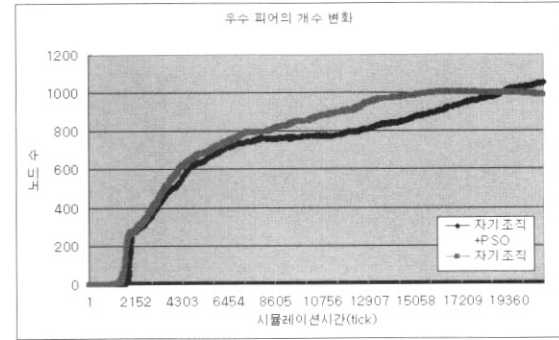
(그림 12) 오버헤드



(그림 13) 피어 개수 변화



(그림 14) 우수 피어 비율 변화



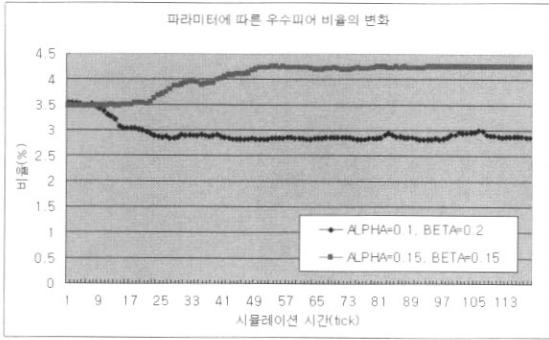
(그림 15) 우수 피어의 개수 변화

영향을 미치지 않으므로 (그림 11)에서 보는 바와 같이 검색 시간의 경우는 고정된 비율을 사용하는 경우와 차이가 없다. 또한, PSO를 이용한 우수 피어 링의 경우 기존 우수 피어 링 구조에서 사용했던 에이전트와 가십 통신만을 이용하고 추가적인 통신비용이 거의 발생하지 않으므로, (그림 12)에서 보는 바와 같이 둘 사이의 오버헤드의 차이는 없다.

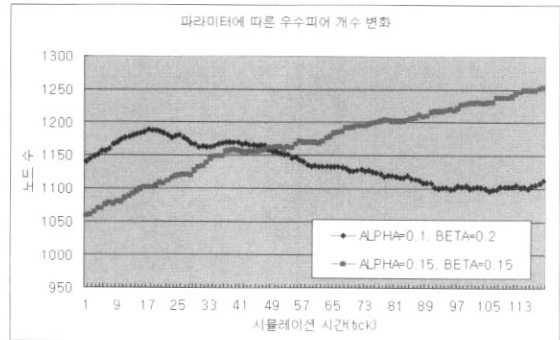
고정된 우수 피어를 사용하는 경우는 우수 피어가 전체 시스템의 변화와 무관하게 그 숫자와 역할 면에서 고정되어 있다. 따라서 시스템의 크기가 증

가하는 상황에서 우수 피어의 수가 증가하지 않으므로 상대적으로 적은 수의 우수 피어로 부하가 집중되어 성능의 하락을 가져오고, 시스템의 크기가 감소하는 상황에서는 시스템 내에 불필요하게 많은 수의 우수 피어가 링을 구성하

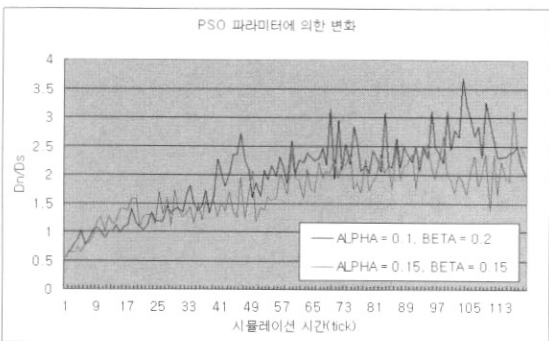
게 되어 실제 검색에 필요한 통신비용에 비해 링 유지에 필요한 통신비용이 상대적으로 커져서 비효율적인 시스템이 된다. 또한, 적응적 우수 피어 링의 경우는 지속적으로 각 피어의 능력치를 모니터링 하여 능력이 큰 피어가 우수 피어가 되도록 하는데 반해, 고정된 우수 피어 링의 경우 한번 우수 피어로 결정 되면 성능이 더 우수한 피어들이 시스템에 참여하더라도 링 구조에 참여할 수 없다. 따라서 우수 피어가 검색을 처리하는 성능이 적응적 우수 피어 링에서 더 좋다. 따라서 (그림 10)의 성공한 메시지 수나, (그림 11)의 검색 시간에서 고정 우수 피어 링이 다른 두 가지 적응적 우수 피어 링에 비해 그 성능이 떨어짐을 볼 수 있다. 그러나 고정된 우수 피어 링의 경우 에이전트를 사용하지 않으므로 (그림 12)에서 오버헤드가 더 적음을 볼 수 있다.



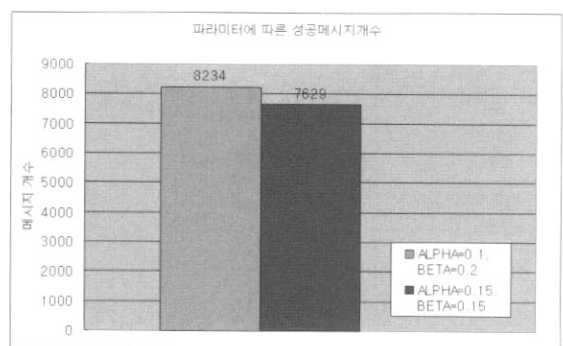
(그림 16) 파라미터에 따른 우수피어 비율 변화



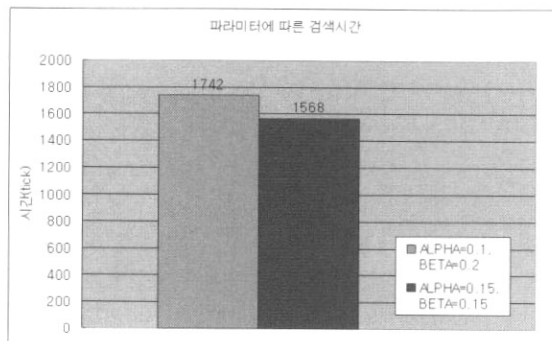
(그림 17) 파라미터에 따른 우수피어 개수 변화



(그림 18) 파라미터에 따른  $d_n, d_s$ 의 변화



(그림 19) 파라미터에 따른 성공메시지 개수



(그림 20) 파라미터에 따른 검색시간

#### 4.2.2 PSO 파라미터에 따른 변화

우수 피어의 비율은 <식 6>, <식 7>, <식 8>에 의해 결정된다. 본 실험에서는 <식 7>과 <식 8>에서  $\alpha, \beta$ 의 변화에 따라 우수 피어의 비율 변화와 시스템 전체 평균  $d_n, d_s$ 의 변화를 살펴보고자 한다. 파라미터  $\alpha, \beta$ 는  $\alpha/\beta$ 가 작을수록  $d_s$ 을 최소화 하는 방향으로 우수 피어의 비율을 변하게 하고  $\alpha/\beta$ 이 클수록  $d_n$ 을 최소화 하는 방향으로 우수 피어의 비율을 변하게 한다.

피어의 개수 변화는 앞의 (그림 13)과 같으며  $\alpha, \beta$ 에 대해  $\alpha=0.1, \beta=0.2$  즉  $d_s$ 를 감소시키는 경우와  $\alpha=0.15, \beta=0.15$  즉  $d_n$ 을 감소시키는 경우를 비교하였다. 이때 우수 피어 비율의 변화와 우수 피어 개수의 변화는 (그림 16), (그림 17)과 같다. 이를 통해  $\alpha/\beta$ 이 작을수록 우수 피어의 비율이 커

지고 우수 피어의 개수가 늘어남을 알 수 있다. 이는  $d_n$ 을 줄이기 위해 우수 피어의 개수를 늘렸기 때문이다. 반대의 경우 우수 피어의 비율은 작아지고 이에 따라 우수 피어의 개수는 감소하였는데 이는  $d_s$ 를 감소시키기 위해 우수 피어의 개수를 감소시켰기 때문이다. (그림 18)은 시간에 따른  $d_n/d_s$ 값을 나타내는데,  $\alpha=0.1, \beta=0.2$ 인 경우  $d_s$ 의 감소를  $d_n$ 의 감소보다 중시하고  $\alpha=0.15, \beta=0.15$ 인 경우  $d_n$ 의 감소를  $d_s$ 의 감소보다 중시함을 알 수 있다.  $\alpha=0.1, \beta=0.2$ 인 경우 우수 피어의 개수 변화가 적기 때문에 (그림 19)에서 보는바와 같이 검색 성공률에서 좀 더 우수하며,  $\alpha=0.15, \beta=0.15$ 인 경우는 우수 피어의 개수를 늘려  $d_n$ 을 감소 시켰기 때문에 (그림 20)에서 보는바와 같이 검색 시간에서 좀 더 우수함을 알 수 있다.



### 5. 결론 및 향후 계획

기존의 우수 피어를 이용한 피어-투-피어 시스템의 성능 향상을 위하여 동적인 환경에 따라 적응적으로 우수 피어의 개수를 조절하고, 더 나아가 능력이 큰 피어가 우수 피어 링에 참여할 수 있도록 하는 자기 조직적 우수 피어 링을 이용한 검색기법이 제안되었다. 그러나 이 연구에서는 고정된 우수 피어 비율을 가정하므로, 본 논문에서는 환경의 변화에 따라 변화하는 최적의 우수 피어 비율을 PSO 기법을 이용하여 지속적으로 찾아내고, 그 비율을 유지할 수 있도록 하는 적응적 우수 피어 비율의 우수 피어 링 검색 기법을 제안하였다.

이를 위하여 PSO 최적화 기법을 사용하여 우수 피어까지의 거리와 우수 피어 상에서 검색 거리를 최소화하는 최적의 우수 피어 비율을 찾는다. 고정된 비율을 유지하기 위하여 시스템의 상황 변화에 따라 바로 우수 피어의 개수가 변화하게 하지 않고, 최적의 우수 피어까지의 거리와 우수 피어 내에서의 거리를 유지할 수 있도록 우수 피어 비율을 변화시킴으로써, 우수 피어 자체의 변화를 지연시키고 우수 피어들 사이의 피어뷰 불일치로 인한 성능 저하를 줄인다.

이를 위하여 각 우수 피어는 검색 메시지를 통해 우수 피어까지의 거리와 우수 피어 상에서의 거리를 추측하고 이를 통해 우수 피어의 비율을 계산한다. 한편 더 정확한 값을 추측하기 위해 우수 피어 간에 가십 프로토콜을 통해 정보를 교환하며, 계산된 우수 피어 비율을 확산하기 위해 에이전트를 이용한다. 그러나 PSO를 위한 거리 정보 전달을 위하여 기존의 우수 피어 링 검색기법에서의 에이전트와 가십 프로토콜만을 이용하므로 추가적인 통신비용의 증가는 거의 없다.

시뮬레이션을 통하여 제안하는 적응적 우수 피어 비율의 검색 기법이 기존의 자기 조직적 우수 피어 링 검색 기법에 비해서 검색 시간이나 오버헤드 측면에서는 거의 변화가 없으나 검색 성공 메시지 수에서 크게 향상되었음을 볼 수 있었다.

본 논문에서는 자기 조직적 우수 피어 링을 이용한 검색 기법에서 PSO를 이용하여 최적의 우수 피어 비율을 유지하도록 하는 기법을 제안하였는데, 일반적인 계층적 우수 피어 네트워크에서 적용할 수 있도록 확장하는 것이 필요하다.

### 참 고 문 헌

[1] The Gnutella Protocol Specification v0.4, [http://www9.limewire.com/developer/gnutella\\_protocol-0.4.pdf](http://www9.limewire.com/developer/gnutella_protocol-0.4.pdf)  
 [2] Freenet Website. <http://freenet.sourceforge.net>.  
 [3] Y. Chawathe, S. Ratnasamy, L. Breslau, N. Lanham and S. Shenker, "Making Gnutella-like P2P Systems Scalable," Proceedings of ACM SIGCOMM 2003.

[4] KaZaA File Sharing Network Homepage, <http://www.kazaa.com/>  
 [5] B. Traversat, A. Arora, M. Abdelaziz, M. Duigou, C. Haywood, J-C. Hugly, E. Pouyoul and B. Yeager, "Project JXTA 2.0 Super-Peer Virtual Network," <http://www.jxta.org/project/www/docs/JXTA2.0protocols1.pdf>  
 [6] B. Traversat, M. Abdelaziz and E. Pouyoul, "Project JXTA: A Loosely-Consistent DHT Rendezvous Walker," <http://www.jxta.org/project/www/docs/jxta-dht.pdf>  
 [7] L. Xiao, Z. Zhuang and Y. Liu, "Dynamic Layer Management in Superpeer Architectures," IEEE Transactions on Parallel and Distributed Systems, Vol.16, No.11, November, 2005.  
 [8] 손재의, "자기 조직적 링 구조를 이용한 비구조적인 피어-투-피어의 검색기법", 서강대학교 대학원, 컴퓨터학과, 2005.  
 [9] S. Sen and J. Wang, "Analyzing Peer-To-Peer Traffic Across Large Networks," IEEE Transactions on Networking, Vol.12, No.2, April, 2004.  
 [10] J. Kennedy and R. Eberhart, "Particle Swarm Optimization," IEEE International Conference on Neural Networks (Perth, Australia), IEEE Service Center, Piscataway, NJ, IV:1942-1948, 1995.  
 [11] A. Carlisle and G. Dozier, "Adapting Particle Swarm Optimization to Dynamic Environments," Proceedings of the 2001 International Conference on Artificial Intelligence, 2001.  
 [12] K. A. Amin and A. R. Mikler, "Dynamic Agent Population in Agent-Based Distance Vector Routing," In Proc. Second International Workshop on Intelligent Systems Design and Application (ISDA), 2002.  
 [13] A. J. Ganesh, A-M Kernmarrec and L. Massoulie, "Peer-to-Peer Membership Management for Gossip-Based Protocols," IEEE Transactions on Computers, 2003.  
 [14] L. A. Adamic, R. M. Lukose, A. R. Puniyani and B. A. Huberman, "Search in Power-Law Networks," Physical Rev. E, Vol.64, No.4, 2001.  
 [15] S. Saroiu, K. P. Gummadi and S. D. Gribble, "Measurement Study of Peer-to-Peer File Sharing Systems," Proceedings of Multimedia Computing and Networking 2002.



장 형 군

e-mail : [hugry.jang@gmail.com](mailto:hugry.jang@gmail.com)  
 2004년 서강대학교 컴퓨터학과(공학사)  
 2006년 서강대학교 컴퓨터학과(공학석사)  
 2006년 3월~현재 (주)바로비전 뉴미디어 연구소 제작중  
 관심분야: 피어투피어 컴퓨팅, 분산처리 시스템

### 한 세 영



e-mail : syhan@sogang.ac.kr  
1991년 포항공과대학교 수학과(이학사)  
2003년 서강대학교 정보통신대학원  
(공학석사)  
2004년~현재 서강대학교 컴퓨터학과  
박사과정

1996년~2006년 (주)이엔지 기술본부

관심분야: 피어투피어 컴퓨팅, 분산처리 시스템

### 박 성 용



e-mail : parksy@sogang.ac.kr  
1987년 서강대학교 컴퓨터학과(공학사)  
1994년 미국 Syracuse University 대학원  
(공학석사)  
1998년 미국 Syracuse University(공학박사)  
1998년~1999년 미국 Bell Communication  
Research 연구원

1999년~현재 서강대학교 컴퓨터학과 부교수

관심분야: Autonomic Computing, Peer to Peer Computing,  
High Performance Cluster Computing and System