

모바일 P2P 네트워크에서 피어의 이동성을 고려한 연속적인 k -최근접 질의 처리

Continuous K -Nearest Neighbor Query Processing Considering Peer Mobilities in Mobile P2P Networks

복경수, 이현정, 박용훈, 유재수
충북대학교 정보통신공학과

Kyoung-Soo Bok(ksbok@chungbuk.ac.kr), Hyun-Jung Lee(hyunjung2004@nate.com),
Young-Hun Park(yhpark1119@gmail.com), Jae-Soo Yoo(yjs@chungbuk.ac.kr)

요약

본 논문에서는 MP2P(Mobile Peer-to-Peer) 네트워크 환경에서 이동하는 피어들을 대상으로 실시간으로 질의 결과를 갱신하는 연속적인 k -최근접 질의 처리 기법을 제안한다. 제안하는 기법은 질의 피어와 근접한 k 개의 피어를 효율적으로 모니터링하기 위하여 모니터링 영역(monitored region)을 생성하여 배포한다. 모니터링 영역은 주변 피어들의 이동 벡터를 이용하여 특정 시간 범위 내에서 최소 k 개의 질의 결과를 보장하기 위해 만들어진 영역이다. 제안하는 기법은 주변 피어들의 벡터를 이용하여 Monitoring Region을 계산하기 때문에 유지 시간을 증가시켜 모니터링 영역의 무효화에 따른 질의 재수행 비용을 감소시킨다. 제안하는 기법의 우수성을 입증하기 위해 기존 기법과 성능 평가를 수행한 결과 제안하는 기법이 기존 기법보다 우수한 성능을 보였다.

■ 중심어 : | 모바일 P2P | k -최근접 | 연속 질의 | 위치기반서비스 |

Abstract

In this paper, we propose a continuous k -nearest neighborhood query processing method for updating the query results in real-time over mobile peer-to-peer environments. The proposed method disseminates a monitoring region to efficiently monitor the k -nearest neighbor peers. The Monitoring Region is created to assure at least k peers as the result of the query within the time range using the vector of neighbor peers. In the propose method, the monitoring region is valid for a long time because it is calculated by the vector of neighbor peers of the query peer. Therefore, the proposed method decreases the cost of re-processing by monitoring region invalidation. In order to show the superiority of the proposed method, we compare it with the previous schemes through performance evaluation.

■ keyword : | Mobile P2P | k -Nearest Neighbor | Continuous Query | Location based Service |

* 이 논문은 2012년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 산학협력 선도대학(LINC)육성 사업임(No. 2012-B-0013-010112)

I. 서론

모바일 기기의 확산과 함께 단거리 무선 통신 기술의 발전으로 기존 P2P 시스템에 이동성이 결합된 모바일 P2P(MP2P : Mobile Peer-to-Peer) 네트워크에 대한 연구가 진행되고 있다[1]. 안정적인 대역폭과 신뢰성을 보장하는 유선 환경에서의 P2P 네트워크와 달리 MP2P 네트워크에서 각각의 피어들은 제한된 대역폭, 배터리, 컴퓨팅 능력, 저장 공간을 가지고 있다[2]. 또한, 피어의 이동성으로 토폴로지의 잦은 변화가 발생한다. 따라서 기존의 유선 환경에 연구된 기법들은 MP2P 네트워크에 그대로 적용하기 어렵다. 이러한 문제를 해결하기 위해 MP2P 네트워크에 대한 연구가 진행되었다[3-6]. MP2P는 소셜 네트워크, 파일 공유, 광고 등의 응용에 사용되고 있다[7][8].

일반적으로 MP2P에서 질의 처리는 피어간 통신 경로를 자율적으로 생성하고 피어들이 스스로 질의를 처리하는 분산 질의 처리 방식을 사용한다[9][10]. 최근 MP2P의 이동성으로 인해 질의 결과가 빈번히 변경됨에 따라 실시간으로 질의 결과를 갱신하는 연속적인 k -최근접 질의 처리 기법에 대한 연구가 진행되고 있다 [10-12]. [10]에서는 인접한 피어들이 대부분 동일한 질의 결과를 얻는다는 특성을 이용하여 질의 피어가 이동하는 환경에서 고정된 POI(Point of Interest)를 질의 결과로 산출하기 위한 기법을 제안하였다. 그러나 질의 결과로 고정 POI만을 산출하기 때문에 이동 피어를 대상으로 하는 환경에서는 적용될 수 없다. [11]에서는 QoS(Quality of Service)를 정의하여 질의 결과의 갱신 비용을 조정 할 수 있다. 그러나 정확한 질의 결과를 산출하는 환경에서는 메시지 전송 비용이 증가한다. [12]는 피어들 사이의 질의 정보를 유지하기 위해 네트워크를 그리드 형태로 분할하고, 각 셀마다 대표 피어를 선정하여 질의 처리를 수행하는 기법을 제안하였다. 선정된 대표 피어는 셀과 관련된 질의 정보를 자신의 로컬 테이블에 저장하고, 해당 셀을 벗어날 경우 다른 피어에게 셀에 대한 정보를 전달함으로써 중앙 서버에 의존하지 않고 질의 배포가 가능하다. 그러나 [12]는 이웃 피어들의 현재 위치만을 사용하기 때문에 Safe

Boundary의 유지 시간이 매우 짧다. 따라서 질의 배포 비용이 증가되는 문제점이 있다.

본 논문에서는 MP2P의 이동성을 고려하여 실시간으로 질의 결과를 갱신하는 연속적인 k -최근접 질의 처리 기법을 제안한다. 제안하는 기법은 최소한의 피어 정보만을 수집할 수 있도록 대표적인 질의 배포 기법인 DP(Dominant Pruning) 기법[13]을 기반으로 k 개의 객체 정보를 수집한다. 또한, 질의 피어는 자신과 가장 가까이 위치한 k 개 이상의 객체 이동 정보를 모니터링할 수 있도록 피어의 위치 정보와 벡터 정보를 통해 MR(Monitoring Region)을 생성하여 배포한다. 연속적인 질의 처리를 위해 질의를 매번 재배포하는 대신 질의 피어와 MR을 수신한 피어들의 협업을 통해 질의 결과를 효율적으로 갱신한다. 이러한 질의 처리 방식은 네트워크에 질의를 전달하는 비용과 질의 결과 갱신 비용을 감소시킬 수 있다.

본 논문의 나머지 구성은 다음과 같다. II장의 관련 연구에서는 기존에 제안된 연속적인 k -최근접 질의 처리 기법을 설명하고 기존 기법의 문제점을 기술한다. III장에서는 MP2P 환경에서 제안하는 질의 처리 기법을 기술하고 IV장에서는 기존에 제안된 연속 k -최근접 질의 처리 기법과의 비교 분석을 통해 제안하는 기법의 우수성을 입증한다. 마지막 V장에서는 결론과 향후 연구 방향을 기술한다.

II. 관련 연구

MP2P 환경에서 메시지를 전달하는 가장 일반적인 방법으로 플러딩(flooding) 방식을 사용한다. 플러딩은 통신 범위에 있는 모든 피어들에게 메시지를 전달하는 방법으로 홉 단위로 메시지를 전달하는 방법이다. 이러한 방법은 다중 경로를 통해 메시지가 중복적으로 전달되는 경우가 발생한다. 이러한 문제를 해결하기 위해 DP 기법이 연구되었다[13]. DP는 메시지를 전송할 때 자신의 이웃 정보를 함께 전송하여 이미 메시지를 받은 피어를 제외한 다른 피어들에게만 메시지를 보냄으로써 중복 메시지 전송을 감소시키는 방법이다.

SO-MKNN은 각 질의 피어들이 이전에 산출했던 질의 결과 정보를 공유하는 방법으로 고정된 POI에 대한 연속적인 k-최근접 질의 처리를 수행하는 기법을 제안하였다[10]. 지리적으로 근접한 위치에 있는 질의 피어들은 k-최근접 질의 결과가 부분적으로 동일할 수 있다는 특성을 이용하여 이웃 피어들 사이에 이전 질의 결과를 공유함으로써 연속 질의 처리 비용을 감소시킨다. SO-MKNN에서 피어들 사이에 공유되는 정보는 자신의 로컬에 저장하고 저장된 결과들의 정보를 이용하여 연속 질의를 수행한다. 이 과정에서 피어가 저장하고 있는 객체에 대한 정보량에 따라 kNN_{single} 또는 $kNN_{multiple}$ 을 수행한다. 이웃 피어로부터 수신한 피어 정보 중에 자신이 이미 알고 있는 k번째 피어 보다 근접한 위치에 k개의 피어가 존재할 경우 kNN_{single} 방법으로 질의를 처리한다. 그렇지 않은 경우에는 $kNN_{multiple}$ 방법으로 질의를 처리한다. 이 기법은 고정된 POI에 대한 k-최근접 질의를 처리하기 위해 제안된 기법으로 이동 객체를 질의 대상으로 하는 환경에서는 적용하기 어렵다.

DO-MKNN은 모든 피어가 질의 피어인 동시에 일반 피어가 되는 환경에서 QoS를 적용하여 연속적인 k-최근접 질의를 처리할 수 있는 DO-MKNN를 제안하였다[11]. QoS는 이전에 처리한 질의 결과를 현 시점에서 연속 질의를 처리할 때 산출될 질의 결과에 대한 커버리지(coverage)와 정확도(accuracy)를 나타낸다. DO-MKNN은 사용자가 원하는 QoS를 정의하여 상황에 따라 연속 질의 처리를 위한 처리 과정을 달리한다. 질의 처리는 최대 세 단계로 처리된다. 첫 번째 단계는 이전 질의 처리를 통해 얻은 질의 결과 객체들과 질의 피어의 최대 이동 속도 정보를 통해 각 피어들이 이동할 수 있는 영역 범위를 계산하고 QoS를 만족하는지 여부를 검사한다. 사용자가 원하는 수준의 QoS를 만족할 경우 이후의 과정이 필요 없이 질의 처리를 종료한다. 만약 첫 번째 단계를 통해 질의 결과를 산출하지 못할 경우에 두 번째 단계를 수행한다. 두 번째 단계는 질의 피어가 자신의 이웃 피어에게 질의를 배포하고 이웃 피어는 응답 메시지에 자신의 정보와 함께 자신의 이웃 피어 정보를 함께 전송한다. 질의 피어는 이웃 피어들

로부터 받은 정보를 통해 자신의 객체 테이블에 저장된 객체들의 위치 정보를 갱신한다. 만약 두 번째 단계 통해서도 k 개의 객체를 찾지 못할 경우 세 번째 단계를 통해 좀 더 넓은 범위로 홉 수를 증가시키면서 질의를 배포한다. 이 기법은 이동 피어의 정확한 위치 정보를 기반으로 질의 처리를 수행하는 대신 피어의 최대 이동 속도를 기반으로 QoS 를 만족하는 k 개의 피어들을 질의 결과를 산출한다. 따라서 정확한 질의 결과를 산출해야 하는 환경에서는 메시지 전송 비용이 증가한다는 문제점이 있다.

2차원 공간에서 연속적인 모니터링을 요구하는 질의 형태를 처리하기 위한 기법으로 SMQ(Spatial Monitoring Query) 가 제안되었다[12]. SMQ를 기반으로 처리되는 질의 유형은 S-RMQ, M-RMQ, S-KNNMQ, M-KNNMQ가 있다. SMQ는 질의 비용을 감소시키기 위해 전체 네트워크를 그리드 형태로 분할하고, 각각의 셀에 대한 질의 정보를 관리하는 대표 피어를 선정한다. 대표 피어는 셀에 들어오거나, 셀을 벗어나는 피어의 정보를 관리하기 때문에 질의를 배포하는데 소모되는 통신 비용을 최소화 할 수 있다. M-KNNMQ는 MP2P 환경에서 연속적인 k-최근접 질의를 처리하기 위한 기법으로 Safe-Boundary를 이용하여 질의 처리를 수행한다. Safe-Boundary는 연속적인 질의 처리를 수행하기 위해 질의 결과로 k 개의 객체를 보장하는 특정 모니터링 영역이다. M-KNNMQ를 수행하기 위해 세 개의 Safe-Boundary B_j, B_i, B_k 를 정의한다. 질의 피어가 B_j 을 지나가지 않고, 다른 피어들이 B_i 와 B_k 을 지나지 않는 한 이전 질의 결과가 여전히 유효하다고 판단한다. 질의 피어는 Safe-Boundary를 주변 피어에게 배포하여 특정 피어가 Safe-Boundary를 지나갈 경우 자신의 정보를 질의 피어에게 전달함으로써 k-최근접 질의 결과를 실시간으로 갱신한다. 그러나 Safe-Boundary는 이웃 피어들의 현재 위치만을 이용하여 계산되기 때문에 Safe-Boundary의 유지 시간이 매우 짧다. 또한, 피어 간에 메시지 통신이 실패하는 경우를 고려하지 않았기 때문에 네트워크 연결이 충분하지 않은 환경에 적용되기 어렵다는 단점이 있다.

III. 제안하는 기법

1. 개요

본 논문에서 제안하는 k-최근접 질의 기법은 연속적으로 질의를 처리하는 과정에서 최대한 질의를 재수행하지 않고 기존 질의 결과를 통해 질의 결과를 갱신할 수 있는 질의 처리 기법을 제안한다. 제안하는 질의 처리 기법에서는 질의 갱신 과정에서 메시지 전송 비용을 감소시키기 위하여 피어의 벡터 정보를 기반으로 피어의 향후 위치를 계산하고 연속 질의 처리를 수행하는 기법을 제안한다. 제안하는 연속 k-최근접 질의 처리 기법은 다음과 같이 크게 세 단계로 이루어진다. [그림 1]은 제안하는 질의 처리 기법에 대한 전체 처리 과정을 나타낸 것이다.

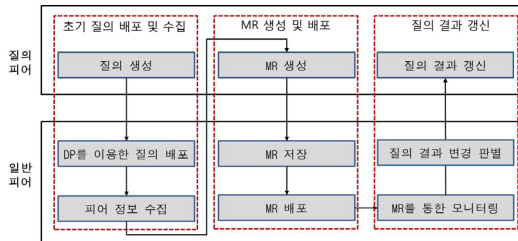


그림 1. 제안하는 질의 처리 절차

첫 번째 단계는 초기 질의를 배포하고 k 개 이상의 피어 정보를 수집한다. 질의 배포 및 수집 단계는 k 개의 결과를 보장하면서 불필요한 메시지 통신을 최소화하는 질의 배포를 수행한다. 연속 질의를 수행하는 과정에서 반복적인 질의 배포로 인한 통신 비용을 감소시키기 위해 k개의 피어 정보를 수집하기 위해 필요한 피어의 개수를 이웃 피어에게 전달한다.

두 번째 단계는 질의 결과 k개의 피어를 포함하는 영역을 보장하기 위한 영역인 MR을 생성하고 질의 갱신을 위해 MR을 이웃 피어들로 배포한다. 질의 피어는 초기 질의 배포를 통해 수집한 이웃 피어들의 벡터 정보를 이용하여 임의의 시간 동안 k개의 피어들을 포함하는 MR을 계산한다.

세 번째 단계는 피어간의 협업을 통한 질의 경과 갱신 단계이다. 이 단계는 피어들이 스스로 자신의 위치

정보를 이용하여 연속적인 질의를 처리한다. MR을 수신한 피어들은 자신이 질의 결과에 영향을 미치는지 여부를 주기적으로 계산하고 질의 결과에 영향을 미친다고 판단되면 자신의 정보를 질의 피어에게 전달한다. 질의 피어는 지속적으로 MR 내에 존재하는 피어의 수를 판별하여 k 개의 피어가 존재하지 않는 경우에만 질의를 재배포한다. 이로 인해 제한된 네트워크 환경에서 연속적인 k-최근접 질의를 수행하기 위한 통신 비용을 감소시킬 수 있다.

제안하는 기법에서는 모든 피어들이 질의 피어인 동시에 질의의 결과가 될 수 있다. 각 피어들은 기본적으로 질의 처리를 위해 필요한 자료 구조를 자신의 로컬에 저장하고 있다. 제안하는 기법에서는 질의 테이블(QT : Query Table), 질의 결과 후보 피어 테이블(QRCPT : Query Result Candidate Peer Table), 질의 결과 테이블(QRT : Query Result Table), 이웃 피어 테이블(OHT : One Hop Table)을 사용한다. QT는 자신의 생성한 질의 정보와 다른 피어들로부터 수신한 질의 정보를 저장하기 위해 사용된다. QRCPT과 QRT는 피어가 1개 이상의 질의를 생성할 때 동시에 생성한다. QRCPT은 질의 처리를 위해 피어 정보를 수집하는 과정에서 수신한 이웃 피어들의 정보를 저장한다. QRT는 질의 결과로 산출한 k개의 피어 정보를 저장한다. 질의 피어는 QRCPT에 저장된 피어들의 위치 정보를 이용하여 자신의 위치와 가장 가까이에 위치한 k개의 피어를 산출하여 QRT에 저장한다. OHT는 자신의 이웃 피어에 대한 정보를 저장한다. 제안하는 기법에서는 자신과 통신이 되는 피어를 파악하기 위해 주기적으로 1-홉 통신을 수행한다. 즉, 모든 피어들은 이웃 피어로부터 수신한 피어 정보를 저장한다.

2. 제안하는 질의 처리 기법

2.1 초기 질의 배포 및 수집

초기 질의 배포 및 수집 단계는 초기 질의 처리를 위해 질의 피어는 이웃 피어로부터 가장 근접한 k개 이상의 피어를 수집한다. 질의 피어는 k개의 피어 정보를 수집하기 위해 수집해야 할 피어의 개수를 이웃 피어에게 전달한다. k개의 피어 정보를 수집하기 위해 자신의 이

웃 피어수를 제외한 ck 를 이웃 피어로 전달한다. 이웃 피어들은 질의 피어와 동일한 방법으로 필요한 피어 개수를 수집할 때까지 점진적으로 질의 배포를 수행한다. 이를 통해 k 개의 피어 수집을 보장하면서 불필요한 메시지 통신을 최소화한다. 식 (1)는 k 개의 피어 정보를 수집한다고 할 때 이웃 피어에게 수집을 요청하는 피어의 개수 ck 을 나타낸다. 이때, i 는 자신의 이웃 피어수이다.

$$ck = k - i \quad (1)$$

[그림 2]는 $k = 5$ 일 때 피어 정보를 수집하는 과정을 나타낸다. 질의 피어 q 는 자신의 통신 반경에 있는 이웃 피어 a, b, c MSG_KNN_BROADCAST를 전달한다. MSG_KNN_BROADCAST는 질의 피어가 질의를 생성한 후 피어 정보를 수집하기 위해 배포하는 메시지를 나타낸다. MSG_KNN_BROADCAST에서 k 는 질의를 생성할 때 정의한 k 값에서 이웃 피어의 개수 3을 제외한 값인 2를 전달한다. MSG_KNN_BROADCAST를 수신한 a, b, c 는 q 와 같은 방법으로 수신한 k 에서 이웃 피어의 개수를 제외하고 k 를 수집하기 위해 필요한 피어의 수를 자신의 이웃 피어로 전달한다. 만약 수신한 메시지의 k 에서 이웃 피어의 개수를 제외한 수가 0일 경우 MSG_KNN_BROADCAST 배포를 중단하고 자신의 정보와 모든 자식 피어의 정보를 담은 MSG_KNN_RESULT를 질의 피어로 전송한다. MSG_KNN_RESULT는 MSG_KNN_BROADCAST를 수신한 피어들이 수집한 피어 정보를 질의 피어로 전달하기 위한 메시지이다.

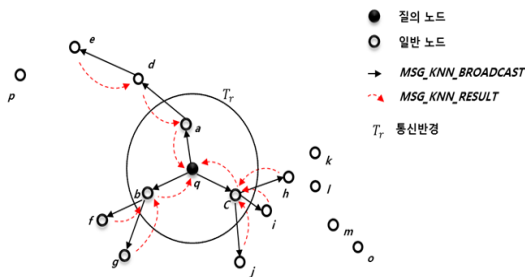


그림 2. $k = 5$ 일 때 초기 피어 정보 수집 과정

질의를 배포하는 과정에서 피어들이 이미 받은 질의를 다른 피어로부터 중복적으로 받는 경우가 발생한다. 제안하는 기법에서는 가장 처음에 수신한 메시지에 대해서만 MSG_KNN_RESULT를 응답하고 이후 받은 중복 메시지를 보낸 피어에게 MSG_REFUSAL_RESPONSE를 보내는 것으로 동일한 질의에 대한 중복적인 메시지 전달을 방지한다. [그림 3]은 동일한 질의에 대해 두 개 이상의 피어로부터 메시지를 받은 경우의 동작 과정을 나타낸 것이다. 피어 a 는 피어 q 의 k 피어 검출 메시지를 수신한 후 자신의 이웃 피어 b 와 c 에게 전달한다. 피어 b 와 c 는 수신한 k 에서 자신의 이웃 피어 수를 제외한 수가 2보다 크기 때문에 각각 자신의 이웃 피어인 d 로 MSG_KNN_BROADCAST를 전송한다. 이 때, 피어 d 는 b 가 전송한 MSG_KNN_BROADCAST를 수신한 후 피어 c 로부터 동일한 질의에 대한 중복 메시지를 전달받게 된다. 따라서 피어 d 는 b 에게 k 피어 MSG_KNN_RESULT를 전송하고 피어 c 에게 MSG_REFUSAL_RESPONSE를 전달한다.

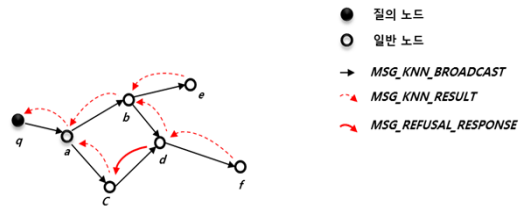


그림 3. 중복 메시지 처리 과정

[그림 4]는 초기 질의 배포 과정에서 MSG_KNN_BROADCAST를 수신 했을 때 처리되는 알고리즘이다. MSG_KNN_BROADCAST를 수신한 피어들은 메시지 식별자를 확인하여 이미 수신한 메시지인 경우 MSG_REFUSAL_RESPONSE를 전달한다. 기존에 수신하지 않은 메시지인 경우 QT에 질의를 저장하고 메시지의 k 에서 자신의 자식 피어의 개수를 제외한 수를 이웃 피어에게 전달할 ck 로 설정한다. 자신의 자식 피어가 없는 경우 질의 피어로 MSG_KNN_RESULT을 전송한다. 이때, 메시지 내의 result_candidate list에는 자신의 정보만을 저장한다. ck 가 0일 경우 result_

candidate list에 자신의 정보를 담아 질의 피어로 MSG_KNN_RESULT를 전송한다. 반대로 ck가 1이상 이라면 k값으로 계산된 ck을 지정하여 자식 피어로 MSG_KNN_BROADCAST를 전달한다.

```

receive_MSG_KNN_BROADCAST()
{
    if (MSG_ID가 이미 수신된 기록에 있는 경우)
        MSG_REFUSAL_RESPONSE 메시지 전송;
    QT에 수신한 질의 정보 저장
    ck = k - i;
    if (자식피어의 개수가 0인 경우) {
        MSG_KNN_RESULT 메시지 전송;
    }
    if (ck가 0인 경우) {
        질의피어로부터 가장 가까운 ck개의 피어 산출;
        산출된 피어 정보를 부모 피어에게 전송;
    }
    else {
        자식피어로 MSG_KNN_BROADCAST 메시지 전송;
    }
}

```

그림 4. MSG_KNN_BROADCAST 수신 시 알고리즘

MSG_KNN_RESULT을 수신한 피어는 응답 대기 테이블을 갱신한 후에 모든 자식 피어로부터 MSG_KNN_RESULT 메시지 혹은 MSG_REFUSAL_RESPONSE를 수신했는지 검사한다. 모든 자식 피어로부터 MSG_KNN_RESULT 혹은 MSG_REFUSAL_RESPONSE를 수신했다면 자신이 질의 피어가 아닌 경우 질의 피어와 자신이 수집한 피어들 사이의 거리를 계산한다. 계산된 결과를 통해 질의 피어로부터 가장 가까운 ck개의 피어를 선정하여 질의 피어로 MSG_KNN_RESULT를 전송한다. 만약 모든 자식 피어로부터 MSG_KNN_RESULT 혹은 MSG_REFUSAL_RESPONSE를 수신했는지 검사하여 이를 모두 수신하고 자신이 질의 피어라면 자신의 이웃 피어들로부터 수신한 피어 정보를 기반으로 자신과 가장 가까이에 위치한 k개 피어를 산출한다. 산출한 k개의 피어 정보는 QRT에 저장한다. k개 피어의 벡터 정보를 기반으로 특정 시간동안 질의 결과 유지를 보장하는 MR 계산하여 이웃 피어들로 MSG_MONITORING_REGION을 배포

한다.

MSG_REFUSAL_RESPONSE을 수신한 피어는 모든 자식 피어로부터 MSG_KNN_RESULT 혹은 MSG_REFUSAL_RESPONSE를 수신했는지 검사한다. 모든 자식 피어로부터 MSG_KNN_RESULT 혹은 MSG_REFUSAL_RESPONSE를 수신했다면 자신이 질의 피어가 아닌 경우 질의 피어와 자신이 수집한 피어들 사이의 거리를 계산한다. 질의 피어와 자식 피어들로부터 받은 피어들 사이의 거리를 계산하여 질의 피어로부터 가장 가까운 ck개 피어를 산출한 후 질의 피어로 MSG_KNN_RESULT를 전송한다. 만약 모든 자식 피어로부터 MSG_KNN_RESULT 혹은 MSG_REFUSAL_RESPONSE를 수신했는지 검사하여 이를 모두 수신하고 자신이 질의 피어라면 자신의 이웃 피어들로부터 수신한 피어 정보를 기반으로 자신과 가장 가까이에 위치한 k개 피어를 산출한다. 산출한 k개의 피어 정보는 QRT에 저장한다. k개 피어의 벡터 정보를 기반으로 특정 시간동안 질의 결과 유지를 보장하는 MR 계산하여 이웃 피어들로 MSG_MONITORING_REGION을 배포한다.

2.2 MR 생성 및 배포

질의 피어는 초기 질의 배포를 통해 수집한 이웃 피어들의 벡터 정보를 이용하여 임시 시간 동안 최소 k개 피어를 포함하는 영역으로 MR을 계산하고 이웃 피어들로 배포한다. 질의 피어는 연속 질의를 처리하는 과정에서 MR 내에 k개의 객체가 존재한다면 질의를 재수행하지 않고 질의 갱신이 가능하다. 따라서 효율적인 질의 갱신이 가능하다. (식 2)는 n 시간 동안 k개 피어를 보장하는 질의 Q의 MR인 $MR_n(Q)$ 을 나타낸다. 이때, t_c 는 현재 시간, t_{c+n} 은 n만큼의 미래 시간, O_i^k 는 전 단계에서 수집된 객체 중의 t_i 시점에 질의 피어로부터 k번째 근접한 객체를 나타낸다.

$$MR_n(Q) = MAX\{Dist(Q, O_i^k), t_i \in [t_c, t_{c+n}]\} \quad (2)$$

MR은 이전의 질의 결과와 상관없이 k개의 피어를 포함하는 영역을 나타낸다. 질의 피어는 MR을 산출한

후 주변 피어들에게 MR 을 배포하여 연속적인 질의 결과를 산출한 최소한의 피어 정보만을 수집한다. [그림 5]는 질의 피어와 수집한 피어들의 벡터 정보를 이용하여 $k = 3$ 일 때, t_5 시간 동안 연속 질의를 처리하기 위해 생성한 상대적 거리를 나타낸다. 질의 피어는 수집한 각 피어들의 벡터 정보를 통해 향후 위치 정보를 예측하여 t_5 시간 동안 3개 이상의 피어를 포함하는 영역을 MR으로 산출한다.

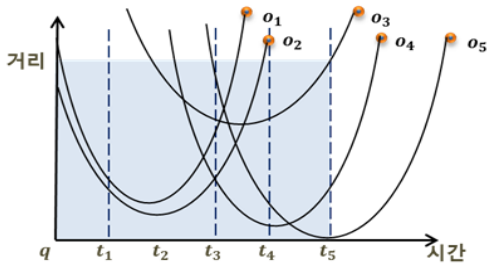


그림 5. 질의 피어와 피어들 사이의 상대거리

[그림 6]은 상대적 거리를 기반으로 산출된 MR을 나타낸 것이다. 기존 기법의 경우 수집한 피어들의 현재 위치만을 이용하여 Safe Boundary S_r 을 산출한다. 제안하는 기법에서는 수집한 피어들의 벡터 정보를 이용하여 피어들의 향후 위치에 대한 예측이 가능하므로 특정 주기 동안에 k 개의 피어를 모니터링 할 수 있는 영역인 MR을 생성한다. 따라서, 피어들의 이동이 잦은 환경에서 MR 내에 k 개의 피어가 존재한다면 질의 재배포를 수행하지 않는다.

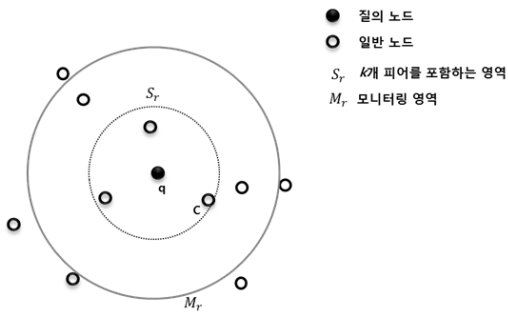


그림 6. 거리를 기반으로 산출된 MR

질의 피어는 MR을 생성한 후 MR을 이웃 피어들에게 MSG_MONITORING_REGION 메시지를 배포한다. MSG_MONITORING_REGION은 질의 피어가 질의에 대한 MR을 배포하는 메시지이다.

MSG_MONITORING_REGION를 수신한 피어들은 메시지에 포함된 질의 식별자를 확인하여 자신의 QT에 저장되지 않은 질의인 경우 질의를 저장하고 자신의 위치가 MR 내에 있는지 계산한다. 만약 MR 내에 있다면 자신이 질의에 영향을 미친다고 판단하여 MSG_UPDATE_REQUEST 메시지를 질의 피어로 전송한다. MSG_UPDATE_REQUEST는 각 피어들이 자신의 위치가 질의 결과에 영향을 미친다고 판단하는 경우 질의 피어로 자신의 위치정보를 전달하기 위한 메시지이다. 질의 피어는 MR 내의 피어들로부터 MSG_UPDATE_REQUEST 메시지를 수신 한 후 질의 결과를 산출한다.

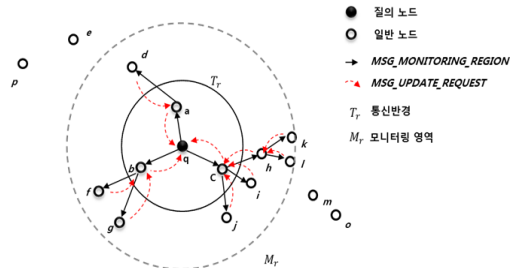


그림 7. MR 배포 및 피어 정보 수집 과정

[그림 7]은 질의 피어가 MSG_MONITORING_REGION 메시지를 배포하고 MSG_UPDATE_REQUEST 메시지를 수신하는 과정을 나타낸다. M_r 은 계산된 MR을 나타낸다. 질의 피어 q 는 MR 내에 위치한 자신의 이웃 피어 a, b, c 에게 MSG_MONITORING_REGION 메시지를 전송한다. a, b, c 는 메시지를 수신한 후에 질의 피어로 MSG_UPDATE_REQUEST 메시지를 전송하고 자신의 이웃 피어 중에 MR 내에 위치한 피어가 존재할 경우 수신한 MSG_MONITORING_REGION 메시지를 전달한다. MSG_MONITORING_REGION 메시지를 수신한 모든 피어는 자신이 MR 내에 있는지 판단하여 질의 피어로 MSG_UPDATE_REQUEST 메시지를 전달

하는 것으로 질의 피어가 질의를 결과를 산출할 수 있도록 한다. MSG_MONITORING_REGION 메시지를 수신한 피어는 QT에 수신한 질의를 저장하고 자신이 MR 내에 있다면 질의 피어로 MSG_UPDATE_REQUEST 메시지를 전달한다. 자신의 이웃 피어가 MR 내에 있는지 검사하여 이웃 피어의 위치가 수신한 MR 내에 있을 경우 해당 피어로 MSG_MONITORING_REGION 메시지를 전달한다.

2.3 협업을 통한 질의 결과 갱신

연속적인 질의를 재수행하는 대신 질의에 미칠 수 있는 각 피어들이 스스로 자신의 정보를 질의 피어로 전달하도록 한다. 제안하는 기법에서는 다음의 두 가지 상황에서 피어간의 협업을 통해 질의 결과를 갱신한다. 첫 번째 경우는 MSG_MONITORING_REGION을 수신한 피어 중에 벡터가 변경되는 경우이다. 벡터가 변경된 피어는 자신의 정보를 질의 피어로 전송하여 질의 피어가 질의를 재배포 하지 않고도 질의결과를 갱신할 수 있도록 한다. 두 번째 경우는 새로운 피어가 MR 내에 진입한 경우이다.

[그림 8]은 MR 내에 있는 피어 c의 벡터가 변경된 경우를 나타낸다. 피어 c는 주기적으로 자신의 벡터가 변경되었는지 검사한다. 만약 벡터가 변경되었다면 자신이 질의 결과에 영향을 줄 수 있다고 판단하여 질의 피어로 MSG_UPDATE_REQUEST를 전달한다. 이때, 메시지는 자신의 이웃 중에 질의 피어와 가장 가까운 피어를 통해 반복적으로 전달한다.

MSG_UPDATE_REQUEST를 수신한 질의 피어는 수신한 피어 정보를 바탕으로 질의 결과를 갱신한다.

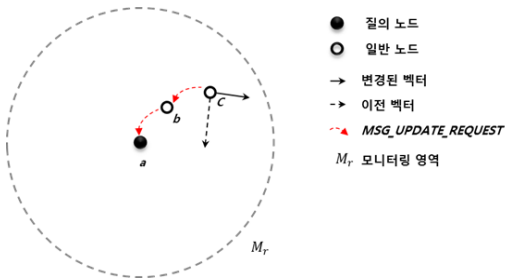


그림 8. MR 내 피어의 벡터가 변경된 경우 처리 과정

[그림 9]는 기존에 MR 내에 존재하지 않았던 피어 c가 MR로 이동한 경우이다. 제안하는 기법에서는 자신의 이웃 피어를 파악하기 위해 주기적으로 1-홉 통신을 한다고 가정하고 있다. 피어 c는 이동 후 자신의 이웃인 피어 b로 1홉 피어 알림 메시지(MSG_HELLO)을 전달한다. 메시지를 수신한 피어 b는 자신의 QT에 저장된 MR 내에 피어 c가 속한 질의의 하나라도 있는 경우에 MSG_QUERY_DELIVER 통해 해당 질의들의 목록을 피어 c로 전달한다. 피어 c는 수신한 질의 목록에 있는 각각의 질의 피어들로 질의 MSG_UPDATE_REQUEST를 전송하여 질의 피어가 질의 결과를 갱신할 수 있도록 한다.

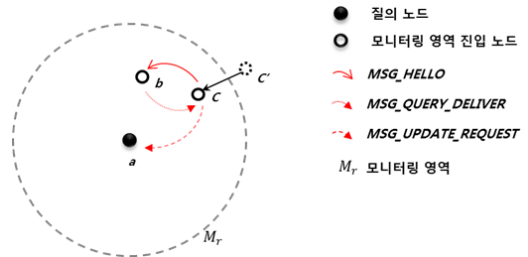


그림 9. 새로운 피어가 MR 내에 진입한 경우

IV. 성능평가

연속적인 k-최근접 질의 기법의 우수성을 입증하기 위해 M-KNNMQ 기법과 성능 평가를 수행한다. 피어의 수, 질의 개수, 벡터 최고 이동 속도, k, 벡터 변경 피어, 통신 반경을 변경해 가면서 질의 처리에 소모되는 메시지 수를 측정하였다. 기존에 연구된 SO-MKNN은 고정된 POI를 질의 대상으로 하며 DO-MKNN 기법은 정확한 질의 결과 대신 QoS를 만족하는 피어들을 질의 결과로 하기 때문에 정확한 질의 결과를 산출하는 환경에는 적용하기 어렵다. 따라서 모든 피어들이 이동하는 환경에서 질의 결과의 산출이 가능한 M-KNNMQ와 비교 평가를 수행하였다. 성능 평가를 수행하기 위해 Intel Core i3 CPU 3.07GHz 와 4G RAM을 사용하는 컴퓨터에서 JDK 1.7을 사용하여 구현하였다. [표 1]은

성능 평가에 사용되는 파라미터를 나타낸 것이다.

표 1. 성능평가 파라미터

| 파라미터 | 값 |
|-------|---------------|
| 실험 크기 | 1000m × 1000m |
| 피어 수 | 500 ~ 2000 |
| 질의 개수 | 20 ~ 100 |

피어 수에 따른 질의 처리 비용을 평가하기 위해 질의 개수는 20개, 백터 최고 이동 속도는 20m/sec, k는 15개, 백터 변경 피어의 수는 전체 피어의 20%, 통신반경은 60m로 고정하고 피어의 수를 변경하면서 피어 수에 따른 메시지 수를 평가하였다. [그림 10]은 피어의 개수에 따라 질의를 처리하기 위해 배포되는 메시지 수를 나타낸다. 피어의 수를 변화시켜가며 성능을 평가해 본 결과 제안하는 기법이 M-KNNMQ 보다 약 220% 우수한 성능을 나타냈다. 기존 연구에서는 단 하나의 피어라도 Safe-Boundary를 지나는 경우 질의 재배포를 수행한다. 제안하는 기법의 경우 질의 피어가 이동하는 경우에도 MR 내에 k개의 피어가 존재한다면 질의를 재배포 하지 않기 때문에 질의 재배포 비용을 감소시킨다.

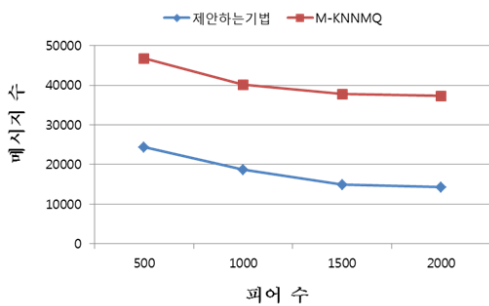


그림 10. 피어 수에 따른 메시지 수

k의 수에 따른 질의 처리 비용을 평가하기 위해 피어 수는 1000개, 질의 개수는 20개, 백터 최고 이동 속도는 20m/sec, 백터 변경 피어의 수는 전체 피어의 20%, 통신반경은 60m로 고정하고 k의 수를 변경하면서 메시지

수를 평가했다. [그림 11]은 k의 변화에 따라 질의를 처리하기 위해 배포되는 메시지 수를 나타낸다. k의 수를 변화시켜가며 성능을 평가해 본 결과 제안하는 기법이 M-KNNMQ 보다 약 170% 우수한 성능을 나타냈다. 기존 기법의 경우 현재 결과 객체만을 위한 영역을 배포하기 때문에 결과 객체의 이동에 따른 질의 재배포를 수행한다. 따라서 k가 증가할수록 질의 처리에 필요한 메시지 수는 제안하는 기법과 비교해 최대 약 2배의 메시지 비용이 발생한다는 것을 확인하였다.

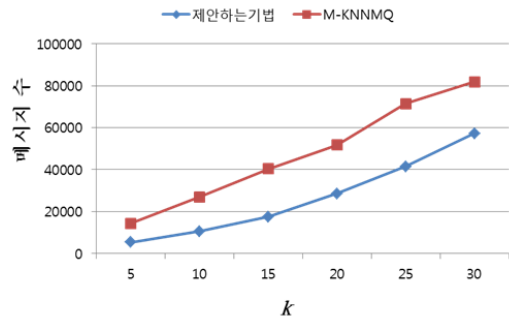


그림 11. k 수에 따른 메시지 수

질의 수에 따른 질의 처리 비용을 평가하기 위해 피어 수는 1000개, 백터 최고 이동 속도는 20m/sec, k는 15개, 백터 변경 피어의 수는 전체 피어의 20%, 통신반경은 60m로 고정하고 질의 수를 변경하면서 질의 수에 따른 메시지 수를 평가했다. [그림 12]는 질의 수에 따라 질의를 처리하기 위해 배포되는 메시지 수를 나타낸다. 질의의 개수를 변화시켜가며 성능을 평가해 본 결과 제안하는 기법이 M-KNNMQ 보다 약 230% 우수한 성능을 나타냈다. 제안하는 기법에서 MR은 단지 k개 피어의 존재를 보장하는 영역으로 사용되며 이 영역 내에 새로운 피어가 진입하는 경우 단지 MSG_UPDATE_REQUEST 메시지만 발생하게 된다. 즉, 기존에 배포된 MR이 여전히 유효하기 때문에 질의를 재배포하지 않는다. 따라서 질의 수의 증가에 따라 비례적으로 메시지 수가 증가하는 기존 기법과 달리 제안하는 기법은 메시지 수가 반비례적으로 증가한다.

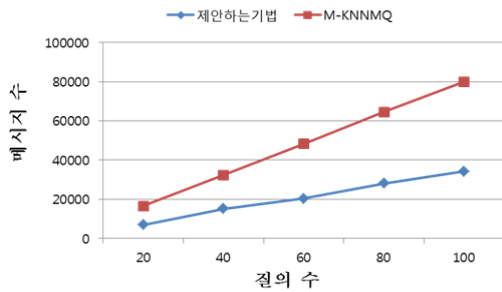


그림 12. 질의 수에 따른 메시지 수

피어의 이동 변화에 따른 질의 처리 비용을 평가하기 위해 피어 수는 1000개, 질의 개수는 20개, k는 15개, 벡터 변경 피어의 수는 전체 피어의 20%, 통신반경은 60m로 고정하고, 최대 벡터 값을 변경하면서 메시지 수를 평가했다. [그림 13]은 피어들의 최대 벡터 값에 따라 질의를 처리하기 위해 배포되는 메시지 수를 나타낸다. 피어들의 최대 벡터 값을 변화시켜가며 성능을 평가해 본 결과 제안하는 기법이 M-KNNMQ 보다 약 300% 우수한 성능을 나타냈다. 제안하는 기법의 경우 피어의 벡터 정보를 이용하여 MR 을 생성하기 때문에 질의를 재배포하지 않고 효율적인 질의 결과 갱신을 수행한다. 따라서 피어들의 최고 이동 속도가 증가하여도 질의를 처리하기 위해 발생하는 메시지 양이 감소한다.

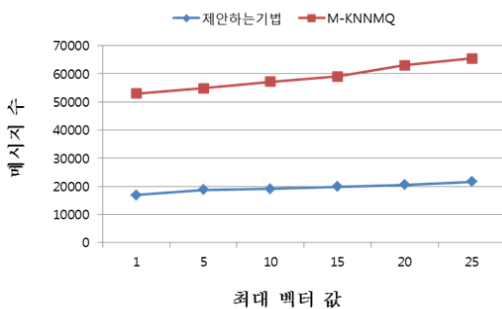


그림 13. 최대 벡터 값에 따른 메시지 수

V. 결론 및 향후 연구

본 논문에서는 MP2P 환경에서의 이동성을 고려하여

실시간으로 질의 결과를 갱신하는 연속적인 k-최근접 질의 처리 기법을 제안하였다. 제안하는 기법에서는 피어의 벡터 정보를 이용하여 효율적인 질의 결과 갱신이 가능하도록 하였다. 또한 질의 피어 중심적으로 질의 처리를 수행하는 대신 피어들과의 협력적인 질의 처리를 수행함으로써 질의를 갱신 비용을 감소시켰다. 제안하는 기법의 우수성을 보이기 위해 기존의 연속적인 k-최근접 질의 처리 기법인 M-KNNMQ 와의 성능 평가를 수행한 결과 제안하는 기법이 기존의 기법보다 우수한 성능을 보였다.

향후 연구로 질의 처리를 수행하는 과정에서 중복 메시지 수신을 감소시키기 위한 연구를 수행할 예정이다.

참고 문헌

- [1] Y. Luo, O. Wolfson, and B. Xu "Mobile Local Search via P2P Database," Proc. IEEE International Interdisciplinary Intersociety Conference on Portable Information Devices, pp.1-6, 2008.
- [2] D. T. Ahmed and S. Shirmohammadi, "Design Issues of Peer-to-Peer Systems for Wireless Ad Hoc Networks," Proc. International Conference on Networking, p.26, 2007.
- [3] R. Haw, C S. Hong, and D. S. Kim, "Group P2P Network Organization in Mobile Ad-Hoc Network," Proc. Asia-Pacific Network Operations and Management Symposium, pp.477-480, 2009.
- [4] K. S. Bok, D. W. Kwak, and J. S. Yoo, "A Resource Discovery with Data Dissemination over Unstructured Mobile P2P Networks", KSII TRANSACTIONS ON INTERNET AND INFORMATION SYSTEMS, Vol.6, No.3, pp.815-834, 2012.
- [5] 광동원, 북경수, 강태호, 여명호, 유재수, 조기형, "콘텐츠 검색 서비스를 위한 효율적인 이동 P2P 구조," 한국콘텐츠학회논문지, 제9권, 제1호,

pp.30-44, 2009.

- [6] 광동원, 북경수, 박용훈, 정근수, 최길성, 유재수, “이동 P2P 환경에서 효율적인 데이터 전송을 이용한 피어 색인 기법,” 한국콘텐츠학회논문지, 제10권, 제9호, pp.26-35, 2010.
- [7] M. Li, E. Chen, and Phillip C. Sheu, “A Chord-Based Novel Mobile Peer-to-Peer File Sharing Protocol,” Proc. Asia-Pacific Web Conference, pp.806-811, 2006.
- [8] Z. Chen, H. T. Shen, Q. Xu, and X. Zhou, “Instant Advertising in Mobile Peer-to-Peer Networks,” Proc. International Conference on Data Engineering, pp.736-747, 2009.
- [9] 임종태, 박용훈, 북경수, 유재수, “이동 P2P 네트워크 환경에서의 효율적인 연속 범위 질의 처리 기법,” 정보과학회논문지 : 컴퓨터의 실제 및 레터, pp.36-330, 2012.
- [10] W. Ku and R. Zimmermann, “Nearest Neighbor Queries with Peer-to-Peer Data Sharing in Mobile Environments,” Pervasive and Mobile Computing, Vol.4, No.5, pp.775-788, 2008.
- [11] C. Chow, F. Mohamed, and L. Hong, “On Efficient and Scalable Support of Continuous Queries in Mobile Peer-to-peer Environments,” IEEE Transactions on Mobile Computing, Vol.10, No.10, pp.1473-1487, 2011.
- [12] P. Galadames, K. Kim, and Y. Cai, “A Generic Platform for Efficient Processing of Spatial Monitoring Queries in Mobile Peer-to-Peer Networks,” Proc. International Conference on Mobile Data Management, pp.1-10, 2010.
- [13] W. Lou and J. Wu, “On Reducing Broadcast Redundancy in Ad Hoc Wireless Networks,” IEEE Transactions on Mobile Computing, Vol.1, No.2, pp.11-123, 2002.

저 자 소 개

북 경 수(Kyoung-Soo Bok)

정회원



- 1998년 2월 : 충북대학교 수학과 (이학사)
- 2000년 2월 : 충북대학교 정보통신공학과(공학석사)
- 2005년 2월 : 충북대학교 정보통신공학과(공학박사)

- 2005년 3월 ~ 2008년 2월 : 한국과학기술원 전산학과 Postdoc
 - 2008년 3월 ~ 2011년 2월 : (주)가인정보기술 연구소
 - 2011년 3월 ~ 현재 : 충북대학교 정보통신공학과 초빙부교수
- <관심분야> : 데이터베이스 시스템, 자료저장 시스템, 위치기반서비스, 모바일 P2P 네트워크, 센서네트워크 및 RFID 등

이 현 정(Hyun-Jung Lee)

정회원



- 2009년 2월 : 배재대학교 전자상거래학과(전자상거래학사)
- 2010년 9월 ~ 현재 : 충북대학교 정보통신공학과 석사과정

- <관심분야> : 무선 센서 네트워크, 데이터베이스, 위치기반서비스 등

박 용 훈(Yong-Hoon Park)

정회원



- 2005년 : 호원대학교 정보통신공학과 및 건축공학과(공학사)
- 2007년 : 충북대학교 정보통신공학과(공학석사)
- 2011년 : 충북대학교 정보통신공학과(공학박사)

- 2011년 3월 ~ 현재 : 충북대학교 Postdoc
- <관심분야> : 데이터베이스 시스템, 정보검색, 시공간 데이터베이스, 센서 네트워크 및 RFID

유재수(Jae-Soo Yoo)

중신회원



- 1989년 2월 : 전북대학교 컴퓨터 공학과(공학사)
 - 1991년 2월 : 한국과학기술원 전산학과(공학석사)
 - 1995년 2월 : 한국과학기술원 전산학과(공학박사)
 - 1995년 3월 ~ 1996년 8월 : 목포대학교 전산통계학과 전임강사
 - 1996년 8월 ~ 현재 : 충북대학교 정보통신공학과 교수
- <관심분야> : 데이터베이스시스템, 정보검색, 센서네트워크 및 RFID, 멀티미디어 데이터베이스, 분산객체컴퓨팅 등