

# 트리 기반의 센서 네트워크에서 질의 중계를 통한 이동 객체의 연속적인 위치 획득 방안

김상대\*, 김천용\*, 조현중\*, 임용빈\*, 김상하°

## Continuous Moving Object Tracking Using Query Relaying in Tree-Based Sensor Network

Sangdae Kim\*, Cheonyong Kim\*, Hyunchong Cho\*, Yongbin Yim\*, Sang-Ha Kim°

### 요약

무선 센서 네트워크에서 이동 객체를 감지하기 위해 질의를 통해 위치를 획득하는 방법과 객체 감지 시 설정된 싱크로 보고받는 방법이 있다. 전자의 질의/응답 형태의 네트워크는 후자에 비해 사용자의 질의/응답에 따른 오버헤드가 있지만, 불필요한 정보 전달이 없으므로 에너지 효율적이다. 최근 질의/응답 형태의 연구들은 가상의 트리를 구성하여 질의를 하는 방법을 사용한다. 이동객체가 움직이는 네트워크에서 가상의 트리는 이동객체의 정보만을 트리에 가지고 있으며, 질의가 발생하면 트리에 저장된 정보를 참조하여 객체의 위치정보를 반환한다. 하지만 빠르게 움직이는 객체를 연속적으로 추적하는 경우, 다량의 질의 발생으로 인한 에너지 문제와 이동 객체의 속도에 따라 질의/응답 과정의 시간 지연으로 추적 정밀도 하락문제가 발생한다. 이러한 문제는 질의를 싱크에서만 하는 것으로부터 발생한다. 본 논문에서는 효율적인 이동 객체 추적을 위하여 싱크에서 시작된 질의를 각 리프노드가 이동 객체의 예상 경로에 중계함으로써 질의 경로를 줄이는 방안을 제시한다. 시뮬레이션을 통해 기존의 방법들에 비해 더 나은 에너지 효율 및 정밀성을 가진다는 것을 증명한다.

**Key Words** : query relaying, object tracking, routing, accuracy, wireless sensor networks(WSNs)

### ABSTRACT

In wireless sensor networks, there have been two methods for sensing continuously moving object tracking: user-query based method and periodic report based method. Although the former method requires overhead for user query rather than the latter method, the former one is known as an energy-efficient method without transferring unnecessary information. In the former method, a virtual tree, consisting of sensor nodes, is exploited for the user querying and sensor reporting. The tree stores the information about mobile objects; the stored information is triggered to report by the user query. However, in case of fast moving object, the tracking accuracy reduces due to the time delay of end-to-end repeated query. To solve the problem, we propose a query relaying method reducing the time delay for mobile object tracking. In the proposed method, the nodes in the tree relay the query to the adjacent node according to the movement of mobile object tracking. Relaying the query message reduces the end-to-end querying time delay. Simulation results show that our method is superior to the existing ones in terms of tracking accuracy.

※ 본 연구는 2013년도 충남대학교 학술연구비에 의해 지원되었습니다.

• First Author : Chungnam National Univ. Department of Computer Engineering, sdkim@cclab.cnu.ac.kr, 정희원

° Corresponding Author : Chungnam National Univ. Department of Computer Engineering, shkim@cnu.ac.kr, 종신회원

\* 충남대학교 컴퓨터 공학과 네트워크 연구실 ({sdkim, cykim, hcho, ybyim}@cclab.cnu.ac.kr, shkim@cnu.ac.kr)

논문번호 : KICS2014-02-065, Received February 24, 2014; Revised April 15, 2014; Accepted May 8, 2014

## I. 서 론

무선 센서 네트워크에서의 객체 추적은 관심지역에 분포된 센서들을 통하여 특정 대상의 움직임 포착 및 동선 파악 등에 활용되는 중요한 어플리케이션 중 하나이다. 예를 들어 동물들의 먹이사냥 경로 또는 계절별 이동 패턴 파악을 통한 천연기념물에 대한 연구를 수행하거나, 사무실이나 공장 등의 업무 환경에서 사람들의 이동 경로를 파악함으로써 좀 더 효율적인 동선을 만들어 내는데 활용할 수 있다. 하지만 무선 센서 네트워크는 무선 통신 능력과 제한된 계산 능력, 저장 공간 및 배터리를 가진 값싼 소형 센서들로 구성된 네트워크이므로, 다른 망들과는 다르게 많은 제약을 가지고 있다. 특히 재충전이 불가능하고 제한된 배터리에 따른 에너지 효율 문제는 무선 센서 네트워크에서 상당히 중요한 부분으로 여겨지고 있다.<sup>[1,2]</sup>

따라서 어플리케이션의 활용 방법에 맞춘 다양한 연구들이 진행되고 있으며, 이동 객체의 위치 추적을 위한 연구는 크게 두 가지 형태로 분류된다. 첫 번째 형태는 이동 객체의 위치 추적을 위해서 사용자의 질의를 통하여 정보를 수집하는 형태이고, 두 번째 형태는 미리 네트워크에 설정된 싱크로 무조건 보고를 함으로써, 사용자의 요구 없이 항상 이동 객체의 위치 정보를 수집하는 형태이다. 후자의 경우 미리 네트워크에 설정된 싱크로 위치정보가 보고됨으로써 사용자의 질의/응답을 위한 부가적인 오버헤드를 최소화 시킬 수 있는 장점이 있지만, 사용자가 원하지 않는 정보에 대해서도 지속적으로 불필요한 정보가 전달됨으로써 에너지 낭비 및 자원 낭비 등의 단점이 있다. 전자의 경우에는 이와 반대되는 장단점을 가진다. 따라서 최근 많이 연구가 되고 있는 사물 인터넷 분야에서도 불필요한 정보에 따른 에너지 낭비 및 자원 낭비를 최소화하기 위해 전자의 형태를 주로 사용한다.<sup>[3-4]</sup>

본 연구도 에너지 및 자원이 매우 제한적인 무선 센서 네트워크의 에너지 효율을 증대시키기 위하여 전자 형태의 네트워크를 기반으로 한다. 최근까지 객체 추적 연구에서 에너지 효율을 증대시키기 위하여 연구된 대표적 질의/응답형 네트워크는 이동 객체 추적을 위하여 일정한 구조체인 가상 트리를 구성하며, 각 이동 객체의 정보는 메시지-프루닝 방법(Message-pruning method<sup>[5]</sup>)을 통하여 가상 트리에 저장된다. 사용자로부터 질의가 발생하면 가상 트리에 저장된 객체들의 이동정보를 통하여 사용자의 질의를 리프노드까지 전달한다. 사용자의 질의를 전달받은 리프노드는 현재 이동객체의 위치를 반환한다.

따라서 사용자는 네트워크 전체에 질의를 하지 않고, 싱크에게 질의를 하여 이에 관여하는 노드에게만 질의를 전달한다. 이는 불필요한 질의의 전달을 막고 에너지 효율을 증대시킬 수 있다. 하지만 각 센서 노드 단위의 가상 트리 구성은 현실적으로 어려운 부분이 있으며, 트리의 중간노드가 고장나거나, 방전이 되는 경우 망이 단절되는 문제가 생길 수 있다. 또한, 비교적 빠른 속도로 이동하는 객체를 연속적으로 추적하고자 할 경우, 추적의 정밀성 및 에너지 효율성 관점에서 문제가 발생할 수 있다. 즉, 사용자의 질의가 있어야만 응답을 수행함으로써 연속적인 추적을 위해서 지나치게 많은 질의가 발생하여 에너지 효율성이 저하되고, 싱크로부터 리프노드까지 질의가 전달되는 시간 지연으로 인하여 추적의 정밀성이 저하될 수 있다.

따라서 본 논문에서는 네트워크를 일점크기로 나누는 클러스터링을 통하여 각 클러스터 헤드노드를 이용한 가상 트리를 구축하며, 싱크에서 발생한 사용자의 질의를 이동 객체의 예상경로에 위치한 노드들에게 중계함으로써 사용자의 반복적인 질의를 피하는 방안을 제안한다. 이를 통하여 본 연구가 이동객체의 연속적인 위치 획득 과정에서 기존 연구들에 비해 높은 에너지 효율성 및 더 높은 정밀성을 보이는 것을 증명한다.

## II. 관련 연구

무선 센서 네트워크에서 이동 객체를 효율적으로 추적하기 위한 많은 프로토콜들이 제안되어 왔다. 이런 연구들은 사용자의 요구가 수반되어야 하는 주문(On-demand)형 시스템<sup>[5-7]</sup>과 사용자의 요구 없이 항상 객체의 위치 정보를 보고하는 선행(proactive)형 시스템의 두 가지 접근 방법으로 나눌 수 있다. 선행형 시스템<sup>[8]</sup>은 추적대상에 대한 정보 및 데이터 보고를 위한 싱크 노드의 위치 등에 대해서 네트워크가 전개 시 미리 시스템에서 지정해준다. 이런 방식의 시스템은 플러딩(Flooding)과 같은 선행 작업을 통하여 지속적으로 이동 객체에 대한 위치정보를 반환받는다. 기본적으로, 이런 추적 방식은 사용자의 질의가 반복적으로 전송할 필요는 없지만, 네트워크 전개 시 플러딩과 같이 사용자의 질의에 대한 정보를 네트워크 전체에 전달해야 하며, 질의의 내용이 변경되는 경우, 반복적으로 네트워크 전체에 변경된 질의를 다시 플러딩 해야 한다. 또한 사용자의 입장에서 더 이상 객체의 정보를 원하지 않는 경우에도 불필요하게 정보가 전달됨으로써 자원 낭비 등의 단점이 있다.

현재까지 연구된 대표적인 주문형 시스템의 경우, 일정한 구조체인 가상의 트리를 구성한다. 이 가상 트리는 프루닝 메시지 방법을 이용하여 가상 트리에 이동 객체에 대한 정보를 저장하고, 질의전달을 위해 플러딩 대신 사용자의 요구가 있을 때 구성된 가상의 트리에 저장된 정보를 따라 질의를 전달함으로써 지나친 자원의 낭비를 막는다.

메시지 프루닝 방법은 [5]에서 제안된 방법으로, 구축된 가상의 트리구조에 에너지 효율적으로 이동 객체의 정보를 저장하기 위한 방법이다. 가상 트리의 임의의 리프노드에서 이동 객체를 감지한 경우, 이동 객체의 정보를 싱크노드까지 전달하는 것이 아니라, 정보의 갱신이 필요한 수준까지만 업데이트를 수행한다. 또한 프루닝 메시지는 감지된 대상의 정보를 포함하는 것이 아니라 감지된 대상이 어느 위치에 있는지만을 보고한다. 그림 1을 통해 설명하면 가상 트리가 구축된 네트워크에서 리프노드 1이 이동객체를 최초로 감지하면, 이동객체에 대한 프루닝 메시지는 1의 모든 조상노드에게 전달되어, 조상노드 B에서는 리프노드 1에 이동객체가 존재함을 알고 있으며, 조상노드 A는 조상노드 B에 이동객체가 존재함을 알게 된다. 하지만 이동객체가 이동하여 리프노드 2에서 감지된다면, 리프노드 2의 프루닝 메시지는 자신의 조상노드 B에게만 전달되어, 조상노드 B는 이동객체가 리프노드 1에서 리프노드 2로 이동했음을 인지하게 된다. 이와 같은 방법으로 리프노드 3에서는 조상노드 A까지, 리프노드 4에서는 조상노드 C까지만 프루닝 메시지를 전달한다. 이를 통하여 에너지 효율적으로 이동 객체의 위치에 대한 정보를 가지고 가상의 트리에 저장시킨다. 주문형 시스템중 하나인 [5]은 에너지를 효율적으로 관리하기 위하여 네트워크를 가상의 계층 트리로 만든다. 이 가상 트리는 DAB(drain-and-balance)

정보를 이용하여, 이동 객체의 이동 패턴에 가중치를 부여하여 가중치가 높은 노드부터 연결해나가며 트리를 구성한다. 또한 이 가상 트리에서는 메시지 프루닝이라는 방법을 사용하여 낮은 비용으로 현재 이동 객체의 위치를 효율적으로 갱신한다.

또 다른 주문형 시스템인 [6]은 [5]의 확장으로써 마찬가지로 트리구조를 구성하지만 각 노드 간에 보로노이(Voronoi)그래프를 그리고, 각 노드 사이에 이동 객체의 이동정보를 기록하여 싱크 노드로부터 이동정보에 따른 가중치가 높은 노드부터 연결해나가며 트리를 구성한다.

그러나 [5]와 [6]은 비교적 빠른 속도로 이동하는 객체를 연속적으로 추적하고자 할 경우 추적의 정밀성 및 효율성 관점에서 문제가 발생할 수 있다. 즉, 싱크에서만 발생하는 사용자의 질의가 있어야만 응답을 수행함으로써 지나치게 많은 질의가 발생하는 비효율성뿐만 아니라 싱크에서 리프 노드까지 질의가 도달되는 시간 지연으로 인해 추적의 정밀성이 훼손될 수 있다.

### III. 제안 방안

본 논문은 다음의 사항들을 가정한다. 각 노드는 GPS<sup>[9]</sup>나 기타 위치 서비스들<sup>[10,11]</sup>을 통해 자신의 위치 정보를 얻을 수 있다고 가정한다. 또한 각 노드는 비코닝을 통해 자신의 한 홉 이웃의 정보를 얻을 수 있으며, 각 노드는 네트워크 전개 시에 수동 설정 또는 네트워크 전개 후에 자동 감지방안들<sup>[12,13]</sup>을 통하여 자신이 네트워크 경계에 위치하는지 여부를 알 수 있다. 또한 무선 센서 노드는 2차원 평면에 뿌려진 토폴로지를 고려한다.

#### 3.1 네트워크 모델

이 단원에서는 효율적인 이동객체 추적을 위한 네트워크 모델에 대해 설명한다. 제안된 방안은 네트워크에 가상 클러스터 구조<sup>[14-16]</sup>를 기반으로, 각 클러스터 헤드를 트리의 노드로 사용하는 가상 트리를 구성한다. 센서 노드들은 단일 계층 멀티 홉 클러스터로 구성되고, 클러스터는 균등하게 분포되어 있다고 가정한다. 다음의 두 절을 통해서 가상의 트리 구조를 생성하고 유지하는 방법에 대해 설명한다.

##### 3.1.1 가상 트리 구성

제안된 방안은 가상 트리 구조를 만들기 위해 HEED<sup>[16]</sup>의 클러스터링 방안을 활용한다. [16]은 센서

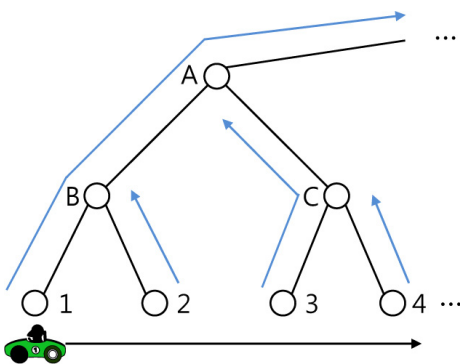


그림 1. 메시지 프루닝 방법  
Fig. 1. Message pruning method

네트워크에서 에너지 효율적이고 분산된 클러스터링 제안방안이다. 이 방안은 멀티 홉의 클러스터를 만들지만 노드의 위치에 관하여 어느 가정도 하지 않는다. [16]는 에너지와 통신비용을 고려하여 클러스터 헤드를 선택하고 클러스터를 생성한다. [16]에서 클러스터 생성을 위해 클러스터 헤드로 선택된 노드들은 클러스터 상수  $k$ -홉 이내의 모든 노드들에게 클러스터 헤드 광고 메시지를 보낸다. 만약 한 노드가 다른 클러스터 헤드로부터 1개 이상의 클러스터 헤드 메시지를 받는다면, 이 노드는 하나의 최소 홉 수로 연결된 클러스터 헤드를 자신의 클러스터 헤드로 선택하고, 클러스터 연결 메시지를 보낸다. 이 연결 메시지는 모든 클러스터 헤드로부터 받은 광고 메시지의 내용을 모두 포함한다. 그러면 이 클러스터 헤드는 인접한 클러스터들에 관한 정보를 얻을 수 있다. 이 때, 그 노드는 이웃 클러스터와의 통신을 중계하는 게이트웨이 (gateway) 노드 역할을 담당한다. 그리고 그 클러스터 헤드는 인접한 모든 클러스터 헤드들에게 광고 메시지를 보낸다. 그 이후에 모든 인접한 클러스터 헤드는 각각 알게 된다. 두 클러스터 간의 통신비용은 두 클러스터 간의 홉 수의 합으로 정의한다. 본 논문에서 모든 클러스터는  $k$ -홉 클러스터이다.

클러스터링이 완료되면, 이를 기반으로 데이터와 질의를 전달하기 위한 라우팅 경로로 사용하기 위한 가상 트리를 구축한다. 싱크노드 또는 네트워크 관리자에 의해 선택된 임의의 클러스터 헤드가 트리의 루트노드가 된다. 루트노드로부터 인접한 클러스터 헤드들 간의 통신비용을 기반으로 최소 비용 신장 트리를 구축하고, 각 트리의 구성노드들은 자신의 부모 및 자식노드에 대한 정보를 알고 있다.

그림 2를 통해 설명하면, 네트워크가 전개되면 [16]의 방법에 따라, 클러스터헤드로 선정된 노드들은 임

의  $k$ -홉 범위의 클러스터를 구성하고, 두 개의 백업 노드(Backup Node)를 갖는다. 클러스터링이 그림과 같이 완료되면, 각 클러스터 헤드간의 통신비용을 기준으로 최소 비용 신장트리를 구축한다.

### 3.1.2 가상 트리 유지

서론에 언급했듯이, 무선 센서 네트워크에서 센서 노드들은 재충전이 어렵고, 비교적 제한된 에너지를 가지고 있기 때문에 고장이나 방전되기 쉽다. 각 클러스터 헤드들은 각 클러스터의 대표이자 가상 트리의 노드의 역할을 수행하므로 다른 센서 들에 비해 고장 나거나, 방전될 위험이 비교적 크다. 따라서 비교적 높은 클러스터 헤드들의 고장에 의해 가상 트리 구조가 정상적인 역할을 수행하지 못할 경우를 대비하여, 각 클러스터 헤드들은 백업 노드(Back-up node)들을 미리 선택한다.

클러스터 헤드는 미리 자신의 전송반경 이내의 노드들 중에 두 노드를 선택하여, 클러스터 헤드의 백업 노드로 사용한다. 백업노드들은 클러스터 헤드가 가지고 있는 라우팅 정보와 클러스터 헤드의 연결 상태를 함께 저장하고 있다. 이 백업노드들은 주기적으로 클러스터 헤드의 에너지 상태 및 고장여부를 확인한다. 만약 클러스터 헤드가 정상적으로 동작하지 않거나, 에너지 잔량이 일정 기준(Threshold)이하로 떨어지게 되면, 이 두 노드 중 에너지가 더 많이 남아 있는 노드가 새로운 클러스터 헤드로 선출된다. 선출된 새로운 클러스터 헤드는 인접 클러스터에게 자신이 새로운 클러스터로 선출되었음을 알린다.

예를 들어, 그림 2에서 기존의 클러스터 헤드 A가 고장 나거나 방전이 되어 클러스터 헤드의 역할을 정상적으로 수행하지 못하게 되는 경우, 백업 노드 1, 2 중에 에너지 잔량이 더 많은 백업 노드 1이 새로운 클러스터 헤드로 선정되어 이웃 클러스터 헤드 B와 C에게 클러스터 헤드 갱신 메시지를 전달하여 백업 노드 1이 기존의 클러스터 헤드 A를 대신하여 선출되었음을 알린다.

### 3.2 객체 추적 방법

이 단원에서는 이전 단원에서 구성한 가상 트리를 기반으로 네트워크에서 이동 객체를 감지했을 때에 대해 설명한다. 사용자는 특정 이동 객체의 정보를 얻기 위해 루트 노드에게 질의를 하게 된다. 이 질의에는 추적 대상, 추적 기간에 대한 정보가 포함되어 있다. 이 질의는 설정된 추적 기간 동안 각 센서들에 의해 중계되어, 사용자의 반복적인 질의를 피하며, 이동

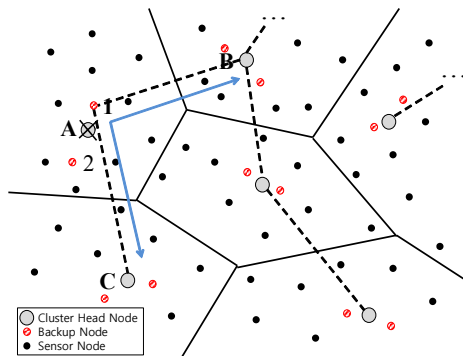


그림 2. 가상 트리 구축 방법  
Fig. 2. Virtual tree construction method

객체가 감지되는 즉시 위치정보를 반환한다.

### 3.2.1 이동 객체 등록

본 논문에서 이동객체는 여러 개일 수 있고, 어디로든지 움직일 수 있다. 따라서 사용자의 질의 없이 객체의 이동을 감지하는 즉시 무조건 이동객체의 위치 정보를 싱크노드에게 보고하는 것은 매우 비효율적인 방법이다.

따라서, 이동 객체가 네트워크 내에 진입하게 되면, 이를 감지한 여러 센서 노드들 중 Source localization<sup>[17]</sup>방법에 의해 선정된 센서 노드는 자신이 포함된 클러스터의 헤드에게 이동 객체를 감지했음을 알린다. 클러스터 헤드는 이동 객체가 자신의 클러스터에 있음을 메시지 프루닝을 통하여 루트노드까지 전달하게 되며, 이동 객체가 다른 클러스터로 이동한 경우에 이를 감지한 클러스터 헤드 또한 메시지 프루닝을 통하여 업데이트가 필요한 수준까지 감지정보를 전달한다. 가상 트리의 각 노드는 자신의 하위노드로부터 전달받은 프루닝 메시지를 통하여 각 이동객체의 감지정보를 포함하고 있으며, 각 객체의 이동에 따라 가장 최근에 전달받은 프루닝 메시지의 정보를 저장한다. 이에 대해 간략하게 나타내면 다음과 같다.

이동 객체 등록 - 네트워크에 정상적으로 가상트리가 구축되었고, 구축된 네트워크에 이동 객체가 진입하는 시나리오를 가정

- 1 이동객체가 특정 클러스터 내에 진입
- 2 while(1)
- 3 이동객체를 감지한 소스 노드들은 [15]을 통해 응답할 노드를 선정
- 4 선정된 노드는 객체를 클러스터 헤드에 보고
- 5
- 6 switch(각 클러스터 헤드)
- 7 case 클러스터 내 노드로부터 보고 :
- 8 상위 클러스터 헤드로 프루닝 메시지 전달
- 9 case 자식 클러스터 헤드 노드로부터 보고 :
- 10 상위 클러스터 헤드로 프루닝 메시지 전달
- 11 객체의 위치정보 저장(자식 클러스터 헤드)
- 12 case 다른 자식 클러스터 헤드 노드로부터 보고 :
- 13 자신의 상위클러스터로 프루닝 메시지 전달
- 14 객체의 기존 정보 삭제 및 새 정보 저장
- 15 end switch
- 16 end while

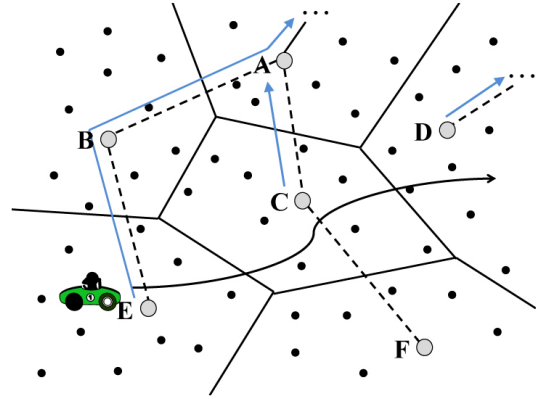


그림 3. 이동 객체 등록  
Fig. 3. Moving object registration

상기의 과정을 그림 3을 통하여 설명하면, 이동객체가 최초로 클러스터 헤드 E의 클러스터에서 감지된 경우, 프루닝 메시지를 통하여 클러스터 헤드 E는 자신의 상위노드 B에게 자신의 클러스터에 이동객체가 있음을 알리고, 클러스터 헤드 B는 자신의 하위노드에 이동객체가 있음을 상위노드 A에게 알린다. 이를 반복적으로 수행하여 싱크 또는 사용자가 지정한 루트노드에게 이동객체의 위치를 클러스터 범위로 알린다. 이 때, 이동객체가 화살표를 따라 이동하게 되어, 클러스터 헤드 C의 클러스터로 이동한 경우, 클러스터 헤드 C는 자신의 클러스터에서 이동객체가 감지되었으므로, 자신의 상위노드 A에게 이동객체가 감지되었음을 알린다, 하지만, 클러스터 헤드 A의 입장에서 아직 자신의 하위노드에 이동객체가 존재하므로 프루닝 메시지를 더 이상 상위노드로 전달하지 않는다. 마지막으로 이동객체가 클러스터 헤드 D의 클러스터로 이동하게 되면, 클러스터 헤드 D는 자신의 상위노드로 프루닝 메시지를 전달하며, 갱신이 필요한 상위노드까지 프루닝 메시지가 전달된다.

### 3.2.2 이동 객체 추적

사용자는 네트워크에서 움직이고 있는 이동 객체의 이동경로를 획득하기 위하여 루트노드에 질의를 하게 된다. 최초로 사용자의 질의를 전달받은 루트노드는 추적 대상, 추적 기간 등의 정보를 포함한 질의를 트리를 통해 전달한다. 트리의 중간 노드들은 사용자가 추적하고자 하는 기간 동안 사용자의 질의를 보관하고 있으며, 질의를 전달받은 리프노드는 자신이 관리하는 클러스터 내에 질의를 플러딩하여 추적 기간 동안 이동 객체의 이동을 보고받고 이를 루트노드에게 반환한다. 또한 이동 객체의 현재 위치를 반환한 노드

들의 위치를 통하여 현재 이동 객체가 어느 방향으로 이동하고 있는지 파악하여 자신의 클러스터를 벗어나 다른 클러스터로 이동하려는 경우, 미리 해당 클러스터 헤드에게 사용자의 질의를 전달하여 미리 해당 클러스터의 노드들에게 사용자의 질의를 전달한다. 이를 통하여 사용자는 반복적인 질의를 하지 않고, 각 리프 노드의 질의 중계를 통해 이동 객체의 연속적인 이동 경로에 대한 정보를 얻을 수 있다. 이를 간략하게 나타내면 다음과 같다.

```

1 while(1)
2   if(사용자 질의의 유효시간 만료 이전)
3     이동객체를 감지한 소스 노드들은 [15]을 통해
      응답할 노드를 선정
4     선정된 노드는 이동 객체의 정확한 위치 정보를
      클러스터 헤드에 보고
5     클러스터 헤드는 트리를 통해 루트에게 전달
6     클러스터 헤드는 이동 객체에 대해 보고한 노드
      의 정보를 순서대로 저장
7     if(이동 객체가 클러스터 경계 쪽으로 이동)
8       해당 경계 방향의 클러스터 헤드에게 사용자의
      질의를 전달
9     end if
10    if(이웃 클러스터 헤드에게 질의를 전달받음)
11      전달받은 질의를 관할 클러스터 내에 플러딩
12    end if
13  end if
14 end while
    
```

상기의 과정을 그림 4를 통해 설명하면, 현재 이동 객체가 클러스터 1에서 클러스터 2로 이동한다고 가정하자. 여기서 최초로 이동 객체를 감지한 클러스터 노드 A, B, C는 [17]을 통하여 이동 객체의 위치 정보를 보고할 대표노드 A를 선출하고, 선출된 클러스터 노드 A는 이동객체의 현재 위치를 반환한다. 이동 객체가 이동함에 따라서, 이동 객체를 감지한 B, C, D 또한 대표노드 D를 선출하고, 클러스터 노드 D는 이동객체의 현재 위치를 반환한다. 클러스터 1에서는 이동 객체가 이동을 함에 따라 A, D, E, F가 이동 객체에 대한 위치 정보를 클러스터 헤드 1에게 보고한다. 이를 전달받은 클러스터 헤드 1은 자신의 상위노드로 각 정보를 전달함과 동시에 이동객체가 클러스터 2방향으로 이동하고 있음을 알게 된다. 따라서 클러스터 헤드 1은 이동 객체가 클러스터 2로 이동하기 전에 클러스터 헤드 2에게 사용자가 전달한 질의를 전달해주

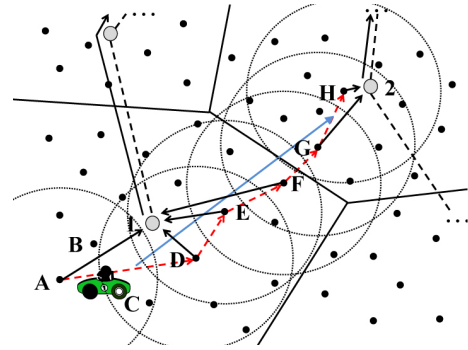


그림 4. 이동 객체 추적  
Fig. 4. Moving object tracking

고, 이를 전달받은 클러스터 헤드 2는 클러스터 2의 구성 노드들에게 사용자의 질의를 플러딩하여 이동 객체가 클러스터 2로 이동하기 전에 미리 클러스터 2의 구성노드들에게 사용자의 질의를 전달한다. 그림4에서 이동객체가 F를 지나 클러스터 2에서 감지되면, 이를 감지한 클러스터 노드 G는 클러스터 헤드 2에게 즉각적으로 위치 정보를 반환하고, 클러스터 헤드 2는 구축된 트리를 따라 싱크노드 또는 루트노드에게 위치정보를 전달하여준다.

만약 이동 객체가 예상된 경로의 클러스터가 아닌 다른 방향의 클러스터로 이동하는 경우, 이동 객체를 감지한 클러스터는 이동 객체의 감지를 알리는 메시지 프루닝을 수행하게 된다. 이 때, 사용자의 질의를 저장하고 있는 중간 노드가 프루닝 메시지를 받고 이동 객체가 예상 경로대로 움직이지 않았다고 판단, 저장하고 있는 사용자의 질의를 현재 이동객체가 있는 클러스터로 사용자의 질의를 중계해 줌으로써, 사용자의 질의 재전송을 막고 가상의 트리구조에서 사용자의 질의를 전달하여준다.

#### IV. 성능 분석

이 장에서는 시뮬레이션을 통하여 이동 객체 추적을 위하여 가상의 구조를 생성하고, 사용자의 질의를 반복적으로 전달하여 이동 객체의 위치를 획득하는 연구인 [5], [6]과 사용자의 질의를 중계하는 제안된 프로토콜의 성능을 평가한다. 먼저 시뮬레이션 환경과 성능평가 항목에 대해 서술하고, 이후 [5], [6]과 비교를 통해 제안방안의 성능을 평가한다.

##### 4.1 시뮬레이션 환경과 성능 평가 항목

우리는 제안된 프로토콜을 네트워크 시뮬레이터 Qualnet 4.0<sup>[18]</sup>으로 구현하였다. 기본적으로 소스 노

드와 목적지 노드를 포함한 200개의 센서 노드들을 1000m x 1000m 필드에 임의적으로 분산시켰고, 실험에 따라 크기를 임의로 조정하였다. 센서 노드의 전송범위는 30m이며, 센서 노드의 전송, 수신 그리고 대기 전력 소비비율은 각각 33, 24, 그리고 0.03mW이다. 노드의 성능은 MICA specification<sup>[19]</sup>을 참조하였다. 센서 노드들은 우리의 클러스터 구성 방법으로 1-계층의 멀티 홉 클러스터로 구성되었으며, 클러스터 크기인 k를 2로 한다. 모든 클러스터 헤드는 백본 기반의 가상 인프라 구조(Backbone-based Virtual Infrastructure, BVI)를 이룬다. 네트워크의 중앙에 있는 한 클러스터를 BVI의 루트 노드로 선정한다. 기본적으로 이동 객체는 임의의 위치에서 임의의 방향으로 같은 속도로 계속 이동하고 있다. 우리가 제안한 프로토콜의 성능을 분석하고 비교하기 위한 성능평가 항목과 용어들을 정리하면 다음과 같다.

**에너지 소비량**은 제어메시지, 데이터 패킷의 송수신을 위해 네트워크 내의 전체 센서 노드들이 소비하는 에너지의 총합으로 정의한다.

**오차거리**는 사용자가 이동객체에 대한 위치정보를 수집하였을 때, 사용자가 이동객체의 위치를 알게 된 시점에 수집된 위치정보와 실제 이동객체와의 거리차이로 정의한다.

**객체속도**는 이동객체가 얼마나 빠르게 이동하는지를 나타낸다. 객체가 빠를수록 오차거리가 증가하게 된다.

**네트워크의 크기**는 센서 노드들이 설치된 네트워크의 크기를 나타낸다. 네트워크의 크기가 클수록 오차거리가 증가한다.

#### 4.2 네트워크 크기에 따른 시뮬레이션 결과

그림 5는 이동 객체의 이동속도를 고정하고 네트워크의 크기에 따른 에너지 소비량을 측정한 결과 그래프이다. 프로토콜 모두 네트워크의 크기가 커질수록 에너지 소비량이 증가하는 모습을 보여준다. 이것은 네트워크의 크기가 커질수록 가상 구조를 구성하는데 많은 에너지를 소비할 뿐 만 아니라, 프루닝 메시지를 통해 이동객체의 정보를 저장하고, 사용자의 질의를 전달 및 위치정보 반환에 더 많은 전달 홉 수를 거쳐가기 때문이다. 상기의 그래프에서 [5]는 모든 노드들을 리프노드로 사용하며, 트리를 논리적인 가상트리로 구성하였기 때문에 네트워크의 크기가 커질수록 가장 많은 에너지 소비량을 보여주며, [6]은 실제 노드들을 이용하여 트리를 구성하였기 때문에 [5]에 비해 비교

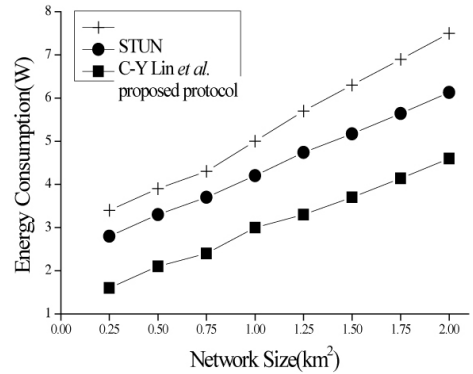


그림 5. 네트워크 크기에 따른 에너지 소비량  
Fig. 5. Energy consumption versus network size

적 적은 에너지 소비량을 보여준다. 하지만 두 프로토콜 모두 이동 객체를 연속적으로 추적하기 위하여 반복적으로 사용자의 질의를 전달하기 때문에, 사용자의 질의를 반복적으로 전달하지 않고 노드들 간에 전달해주는 제안방안에 비해 에너지 소비량이 많은 것을 보여준다.

그림 6은 네트워크의 크기에 따른 오차거리를 나타내는 그래프이다. 10분간 임의로 이동하는 이동객체의 이동 경로를 추적하고, 사용자가 위치정보를 전달받는 시점에서 실제 이동객체의 위치와의 차이의 평균을 계산했다. 네트워크의 크기가 커질수록, 이동객체의 속도가 빨라질수록 사용자가 이동 객체의 위치정보를 전달받는 시점에서 사용자가 전달받은 위치정보와 실제 이동객체의 위치와의 차이가 생기게 된다. [5]는 모든 노드들을 리프노드로 사용하여, 트리를 논리적인 가상 트리로 구성하였기 때문에 네트워크의 크기가 커질수록 많은 전달 홉 수를 가지게 되므로 가장 큰 오차를 가진다. [6] 또한 위치정보를 획득하기

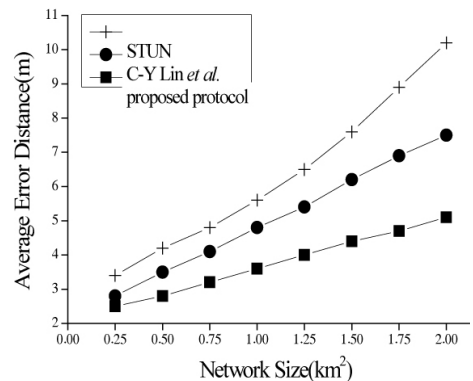


그림 6. 네트워크 크기에 따른 평균 오차거리  
Fig. 6. Average error distance versus network size

위해서 매번 사용자의 질의를 트리를 따라 전달해야 하기 때문에 이동 객체의 위치 정보에 대한 정밀성이 제안방안에 비해 떨어지게 된다.

### 4.3 이동객체의 속도에 따른 시뮬레이션 결과

그림 7은 이동객체의 이동속도에 따른 시뮬레이션 결과이다. 프로토콜 모두 이동객체의 속도가 빨라질수록 사용자가 이동객체의 위치정보를 전달받는 시점에서 실제 이동객체의 위치와의 오차가 생긴다. 이는 각 노드가 사용자의 질의를 전달받고, 위치정보를 다시 사용자에게 전달하는 과정 중에 이동객체가 계속 이동하고 있기 때문에 발생하는 오차로, 이 오차가 크면 클수록 사용자가 받는 정보의 정밀성이 떨어지게 된다. 따라서 사용자가 보다 정밀성이 높은 위치정보를 얻기 위해서 네트워크는 사용자에게 적은 홉 수를 통해 전달해야 한다.

하지만 [5]의 경우 모든 노드들을 리프노드로 사용하여 트리를 논리적인 가상트리로 구성하였기 때문에, 실제로 이동객체를 감지한 리프노드가 싱크에게 데이터를 전달하는 과정에서 각 중간노드들의 위치가 실제 싱크와 더 멀어지는 경우가 발생하기 때문에, 실제 노드들로 트리를 구성하는 [6]이나, 클러스터를 형성하여 클러스터 헤드를 트리로 구성하는 제안방안에 비해 추적정보의 정밀성이 매우 떨어지게 된다. 또한, 제안방안은 [6]에 비하여 비교적 높은 정밀성을 가지게 된다. 이것은 제안방안이 [6]과는 다르게 위치추적을 위한 사용자의 질의를 루트노드가 반복적으로 질의를 하는 것이 아니라, 사용자의 질의를 전달받은 리프 클러스터 헤드노드가 이동객체의 이동방향에 따라 사용자의 질의를 중계해 줌으로써, 사용자가 리프노드에게 질의를 하는 과정이 생략되기 때문이다.

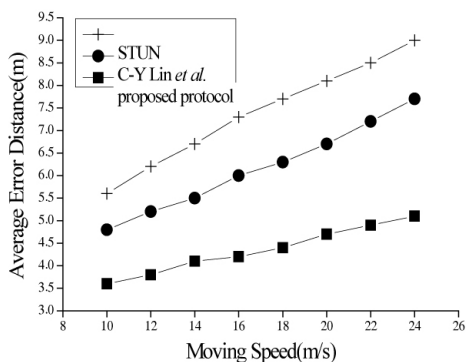


그림 7. 속도에 따른 평균 오차거리  
Fig. 7. Average error distance versus moving speed

### 4.4 시간에 따른 시뮬레이션 결과

그림 8은 시간에 따른 에너지 소비량을 나타낸 시뮬레이션 결과이다. 에너지 소비량은 사용자의 질의 전달, 각 센서 노드들의 제어 메시지 등에 의한 메시지 전달에 가장 큰 영향을 받는다. 따라서 센서 노드들의 메시지 중계가 늘어날수록 시간에 따른 에너지 소비량은 증가한다. [5]는 에너지 소비를 줄이기 위해서 프루닝 메시지방법을 사용하였지만, 논리적인 가상 트리를 구성하여 데이터를 전달하기 때문에 메시지를 전달하는 전달 홉 수가 [6]이나 제안방안에 비하여 많다. 따라서 가장 높은 에너지 소비량을 보이며, [6] 또한 모든 센서들이 트리 구성에 참여하였기 때문에, 이동 객체의 이동에 따른 프루닝 메시지 발생이 빈번하게 일어나고 사용자의 질의를 매번 전달해야하기 때문에 제안방안에 비하여 비교적 에너지 소비량이 높다.

## V. 결 론

본 논문에서 우리는 질의 중계를 통한 이동 객체의 연속적인 위치 획득 방안을 제안하였다. 각 센서 노드는 초기화 단계에서 클러스터링 방안을 활용하여 네트워크를 효율적으로 분산시킨 후, 각 클러스터 헤드를 이용하여 가상의 트리를 구성한다. 제안방안은 이동객체 추적을 위하여 반복적인 질의를 피하기 위하여, 각 클러스터에서 이동 객체의 이동방향을 예측하여 이동 방향 클러스터로 사용자의 질의를 중계해주는 방법을 통하여 사용자의 반복적인 질의를 피하고, 이동 객체 추적 정보에 대한 정밀성을 높인다. 시뮬레이션 결과 제안 방안은 사용자의 질의를 중계해 줌으로써 기존의 객체 추적 연구들에 비해 높은 에너지 효율성을 가지고 보다 정밀한 위치 정보를 획득할 수 있다.

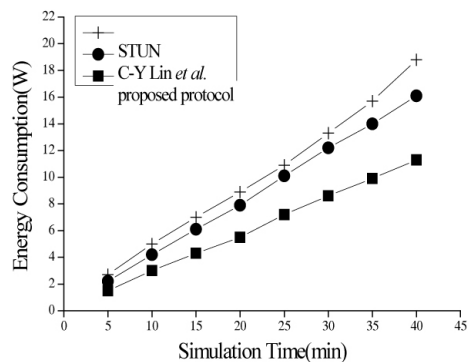


그림 8. 시간에 따른 에너지 소비량  
Fig. 8. Energy consumption versus simulation time



References

[1] I. F. Akyildiz, *et al.*, "A survey on sensor networks," *IEEE Commun.*, vol. 40, no. 8, pp. 102-114, Aug. 2002.

[2] E. B. Hamida and G. Chelius, "Strategies for data dissemination to mobile sinks in wireless sensor networks," *IEEE Trans. Wirel. Commun.*, vol. 15, no. 6, pp. 31-37, Dec. 2008.

[3] B. Anggorojati, *et al.*, "Efficient and scalable location and mobility management of EPCglobal RFID system," *WPMC*, pp. 1-6, Atlantic City, NJ, Jun. 2013.

[4] M.-C. Chung, *et al.*, "RFID Object Tracking with IP Compatibility for the Internet of Things," *GreenCom*, pp. 132-139, Besancon, Nov. 2012.

[5] H. T. Kung and D. Vlah, "Efficient location tracking using sensor networks," *WCNC*, pp. 1954-1961, New Orleans, LA, USA, Mar. 2003.

[6] C.-Y. Lin and Y.-C. Tseng, "Structures for in-network moving object tracking in wireless sensor networks," *BROADNETS*, pp. 718 - 727, San Jose, California, USA, Oct. 2004.

[7] E. S. Lee, *et al.*, "On-demand geographic routing protocol in wireless sensor networks," *J. KISS*, vol. 15, no. 7, pp. 495-499, Jul. 2009.

[8] G. Wang, M. Z. A. Bhuiyan, J. Cao, and J. Wu, "Detecting movements of a target using face tracking in wireless sensor networks," *IEEE Trans. Parallel and Distributed Syst.*, vol. 25, no. 4, pp. 939-949, Mar. 2013.

[9] J. Hightower and G. Borriello, "Location systems for ubiquitous computing," *IEEE Computer*, vol. 34, no. 8, pp. 57-66, Aug. 2001.

[10] E. S. Lee, *et al.*, "Sink location dissemination scheme in geographic routing for wireless sensor networks," *J. KICS*, vol. 34, no. 9, pp. 847-856, Sept. 2009.

[11] N. Bulusu, J. Heidemann, and D. Estrin, "GPS-less low cost outdoor localization for very small devices," *IEEE Personal Commun. Mag.*, vol. 7, no. 5, pp. 28-34, Oct. 2000.

[12] W. S. Lee, H. S. Hong, and S. H. Kim, "Cluster-based continuous object prediction algorithm for energy efficiency in wireless sensor networks," *J. KICS*, vol. 36, no. 8, pp. 489-496, Aug. 2011.

[13] J.A. Bondy and U.S.R. Murty, *Graph theory with applications*, North-Holland: Elsevier, 1976.

[14] O. Younis, M. Krunz, and S. Ramasubramanian, "Node clustering in wireless sensor networks: Recent Developments and Deployment Challenges," *IEEE Network Mag.*, vol. 20, no. 3, pp. 20-25, May/June. 2006.

[15] S. Bandyopadhyay and E. Coyle, "An energy-efficient hierarchical clustering algorithm for wireless sensor networks," in *Proc. IEEE INFOCOM*, pp. 1713-1723, San Francisco, California, USA, Apr. 2003.

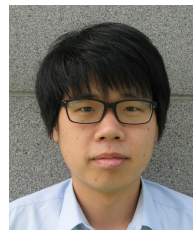
[16] Younis and S. Fahmy, "HEED: a hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks," *IEEE Trans. Mobile Computing*, vol. 3, no. 4, pp. 366-379, Oct./Dec. 2004.

[17] E. Masazade, R. Niu, P. K. Varshney, and M. Keskinoz, "Energy aware iterative source localization schemes for wireless sensor networks," *IEEE Trans. Signal Process.*, vol. 58, no. 9, pp. 4824-4835, Sept. 2010

[18] Scalable Network Technologies, Qualnet, [online]. Available: <http://www.scalable-networks.com>

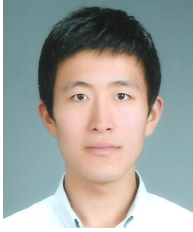
[19] J. Hill and D. Culler, "Mica: A wireless platform for deeply embedded networks," *IEEE Micro*, vol. 22, no. 6, pp. 12-24, Nov./Dec. 2002.

김상대 (Sangdae Kim)



2013년 2월 : 충남대학교 컴퓨터 공학과 졸업  
 2013년 3월~현재 : 충남대학교 컴퓨터공학과 석박사통합과정 <관심분야> Wireless Sensor Networks, IoT 등

김 천 용 (Cheonyong Kim)



2013년 2월 : 충남대학교 컴퓨터공학과 졸업  
2013년 3월~현재 : 충남대학교 컴퓨터공학과 석사과정  
<관심분야> Wireless Sensor Networks, IoT 등

임 용 빈 (Yongbin Yim)



2010년 2월 : 충남대학교 컴퓨터공학과 졸업  
2010년 3월 : 충남대학교 컴퓨터공학과 석박사통합과정  
<관심분야> Internet Routing Wireless Sensor Networks 등

조 현 중 (Hyunchong Cho)



2013년 2월 : 전북대학교 전자정보공학부 컴퓨터공학과 졸업  
2013년 3월~현재 : 충남대학교 컴퓨터공학과 석박사통합과정  
<관심분야> Wireless Sensor Networks, IoT 등

김 상 하 (Sang-Ha Kim)



1980년 : 서울대학교 학사  
1984년 : University of Houston 석사  
1989년 : University of Houston 박사  
1992년~현재 : 충남대학교 전기정보통신공학부 교수  
<관심분야> Internet Routing, Wireless Sensor Networks, MANET, 4G, Mobility, Multicast 등