

# 무선 센서 네트워크에서 모바일 에이전트의 전망

배 인 한\*

## 1. 서 론

무선 센서 네트워크(Wireless Sensor Networks, WSNs)은 최근 몇 년 동안에 연구 커뮤니티에서 많은 주목을 끌고 있고, 많은 이론적이고 실용적인 과제들이 추진되었고, 그리고 실용적인 민간 응용의 수가 증가하고 있다. 넓은 지리적 지역에 수백 수천 개의 센서들을 배치하는 고비용과 WSN 태스킹의 애플리케이션 특정 특징에 기인하여 WSNs의 개발에서 출현하는 추세는 “한 번의 배치, 다수의 응용(One deployment, multiple applications)”이다[1]. 그러한 추세는 센서 노드들이 다수의 애플리케이션들을 처리하기 위하여 다양한 기능을 가질 것을 요구한다. 그러나 임베디드 센서들은 일반적으로 빠듯한 메모리를 가지는 만큼, 모든 가능한 애플리케이션을 실행하기 위하여 요구되는 프로그램들을 임베디드 센서의 로컬 메모리에 저장하는 것은 불가능하다. WSNs에서 새로운 프로그램을 동적으로 배치하기 위하여 모바일 에이전트의 사용은 이 문제를 해결하기 위한 효과적인 방법이 될 것으로 입증되고 있다. 모바일 에이전트는 모바일 에이전트 디스패치의

목적을 실현하기 위하여, 네트워크 환경에서 조건 변화에 대응하여 태스크를 자동적이고 지능적으로 수행하기 위하여 네트워크의 노드들 간에 이주하는 특별한 종류의 소프트웨어 또는 컴퓨터 프로그램이다. 모바일 에이전트들은 WSNs에서 효율적인 데이터 퓨전과 전달 촉진에서 특히 유용하다고 알려졌다[1-10].

그림 1(a)는 클라이언트-서버 패러다임에 따르는 일반적인 데이터 전달 방법을 보여준다. 여기서 이벤트 발생은 데이터를 수집하고 데이터를 각 싱크로 전송하기 위하여 주변 소스 노드들을 트리거 한다. 클라이언트-서버 패러다임에서, 데이터 흐름의 개수는 소스 노드들의 개수와 일반적으로 같아서 높은 대역폭과 에너지 소비를 불러일으킨다. 더욱이, 이 방법은 싱크에 근접한 노드들은 다른 노드들을 위하여 많은 데이터를 전송한다는 사실에 기인하여 그 네트워크에서 균형을 잃은 에너지 소비를 초래할 수 있다.

그림 1(b)에서 보여주는 모바일 에이전트 기반 방법에서, 싱크 노드는 소스 노드들을 하나하나씩 방문하기 위하여 모바일 에이전트를 목표 지역으로 전송한다. 감지된 데이터는 에이전트에 의해 감축, 수집되고 그 모바일 에이전트에 의해 지시된 싱크로 다시 보내고, 다수의 트래픽 흐름 대신에 하나의 트래픽 흐름을 생성한다. 그러나 코드 캐싱과 보안 문제와 같은 특별한 시나리오에서

※ 교신전자(Corresponding Author): 배인한, 주소: 경북 경산시 하양읍 금락로 5(712 702), 전화: 053)850-2742, FAX: 053)850-2750, E-mail:ihbae@cu.ac.kr

\* 대구가톨릭대학교 컴퓨터정보통신공학부 교수

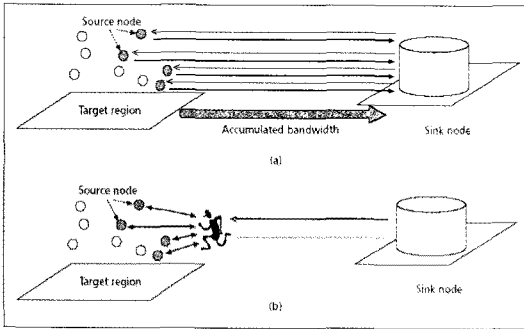


그림 1. WSNs에서 데이터 전달: (a) 클라이언트-서버 기반 패러다임; (b) 모바일 에이전트 기반 패러다임

모바일 에이전트의 사용은 다소의 단점이 있다.

이 논문은 WSNs에서 모바일 에이전트의 주요 응용과 그것들의 개발에서 연구 문제를 간략히 전망한다. WSNs에서 모바일 에이전트 기반 데이터 전달의 응용과 구조를 기술하고, 모바일 에이전트 순회 계획을 설명한다. 그리고 모바일 에이전트 미들웨어 시스템의 설계와 서로 통신하는 모바일 에이전트 협동을 논의한다.

## 2. 모바일 에이전트 응용

WSNs에서 모바일 에이전트 적용의 이점은 주로 2가지 요소가 있다. 첫째, 원시 그대로 전송과 센서 노드의 대부분 에너지 소비를 일으키는 감지되는 데이터 위치로 데이터 처리 요소들을 이동시켜 대역폭 소비를 잠재적으로 감소시킬 수 있다. 이것은 많은 량의 데이터가 수집되고 싱크로 전달되어야 할 때 매우 매력적이다. 덧붙여, 모바일 에이전트 시스템은 다른 방법들과 비교되는 고도의 WSN 재구성 유연성을 제공하고, 그리고 협력적인 정보 처리를 용이하게 한다. 그러한 측면을 고려하여, 모바일 에이전트 기술의 사용이 효과적인 해결책이 될 수 있다고 증명된 3가지 일반적인 WSN 응용을 설명한다. 실제로, 라우팅과 데이터

퓨전과 같은 다수의 다른 WSN 문제들이 모바일 에이전트 시스템에 의해 효율적으로 해결될 수 있다.

### 2.1 비주얼 센서망

하드웨어 소형화에서 최근 발전은 이미지 응용을 위한 특수 주변기기 모듈의 사용을 지원하는 센서 장치들의 구현을 허용한다. 그러한 가볍고 값싼 이미지 하드웨어의 가용성은 센서 노드들로부터 비디오 스트림과 스틸 이미지의 검색을 가능하게 하는 비주얼 센서 네트워크 개발이 촉진되고 있다. 사실, 모바일 에이전트들은 비주얼 센서 네트워크에서 특히 유용하다고 알려져 있다[1]. 이미지 센서에 의하여 생성된 데이터의 량이 일반적으로 매우 많기 때문에, 전체 화면을 전송하는 것은 많은 대역폭과 에너지를 소비할 뿐만 아니라 만일 그 싱크가 그 화면에서 어떤 관심 영역(ROI, region of interest)만을 평가한다면 불필요할 수도 있다. 그림 2는 비주얼 센서 네트워크에서 모바일 에이전트의 응용을 보여준다. 여기서 이미지 세그멘테이션 코드를 운반하는 어떤 모바일 에이전트는 하나하나씩 이미지 센서들을 방문하기 위

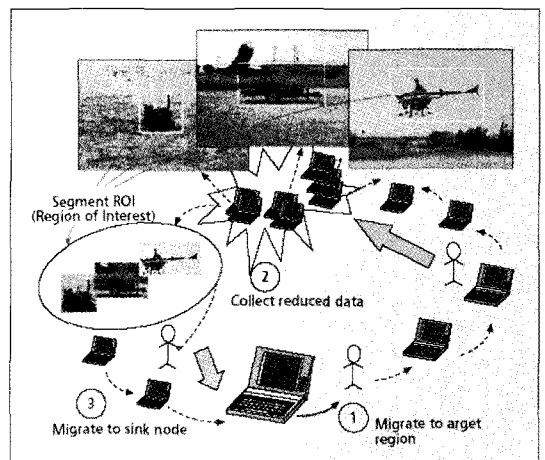


그림 2. 모바일 에이전트 기반 이미지 질의

하여 목표 지역에 파견되고, 그것들의 대응하는 ROI로부터 이미지 데이터를 수집한다. 즉, 목표 지역의 각 센서 노드에서 방대한 량의 이미지 데이터는 더 적은 이미지 데이터로 감축된다. 감지되고 있는 환경 주변 상황이 상당히 변화했을 때, 이미지 처리 코드 작업을 효율적으로 유지하기 위하여 다른 이미지 세그멘테이션 알고리즘을 실행하는 모바일 에이전트가 재구성 특수 이미지 센서들에 파견될 수 있다.

### 2.2 목표 추적

[2]에서, 모바일 에이전트들은 3변 측량술 (trilateration)이라 부르는 간단한 위치추정 알고리즘을 사용하여 어떤 목표물의 위치를 추적하기 위하여 그리고 위치 서버로 주기적으로 그 위치를 보고하기 위하여 사용되었다. 이 방법에서, 어떤 노드는 그림 3a(1)에서 보여진 것처럼, 목표물 위치를 예측하기 위하여 자신과 2개의 그것의 이웃 노드들로부터의 위치 측정 정보에 의존한다. 여기서, 3개의 원은 3개의 모바일 에이전트들로부터의 측정들에 기초한 가능한 목표물 위치이다. 하나는 부모 에이전트이고 다른 2개는 그 객체를 협력적으로 위치를 정하기 위하여 부모 에이전트에 의해 초대되어진 자식 에이전트들이다. 그림 3a(1)에서, 노드 A에 거주하는 부모 에이전트를 도와 목표물을 찾기 위하여 노드 B와 C에 에이전트들을 파견한다. 그 목표물이 노드 B로부터 떠날 때, B에서 수신되는 신호 수준은 줄어들 것이다. 그리고 그 신호 수준이 임계값 이하로 떨어지면, 그림 3a(2)에서 보여진 것처럼 B에서 자식 에이전트가 취소되고 새로운 자식 에이전트가 D로 파견되어진다. 목표물이 노드 C를 통과하면, 부모 에이전트가 C로 이주할 경우, 부모 에이전트는 추적에 실패할 것이다. 모든 이전 자식 에이전트들

이 호출되어지고, 그림 3a(3)에서 보여진 것처럼 새로운 부모 에이전트에 의해 새로운 자식 에이전트들이 노드 D와 E로 파견될 것이다. 그림 3a는 부모 에이전트가 단 2개의 자식 에이전트들과 협력하는 경우를 보여 주지만, 위치 정확성을 향상시키기 위하여 다 많은 에이전트들이 사용될 수 있다.

그에 비해, Xu 등 [3]은 목표물 추적을 위한 모바일 에이전트들의 다른 응용을 제안하였다. 그림 3(b)에서 보여진 것처럼, 새로운 목표물이 탐지된 후, 하나의 모바일 에이전트가 그 목표물의 로밍 경로를 추적하기 위하여 파견되어진다. 그 에이전트가 어떤 센서 노드로 이주할 때, 그 에이전트는 그 객체 인식 정확성을 점점 증가시키기 위하여 데이터를 수집한다. 성취된 정확성이 어떤 임계값을 초과하고 객체 인식의 요구조건을 만족하면, 그 모바일 에이전트는 추적 프로세스를 종료하고 싱크 노드로 수집된 결과를 반환한다. 즉, 불필요한 데이터 수집과 에이전트 이주와 관련된 오버헤드 발생을 피한다.

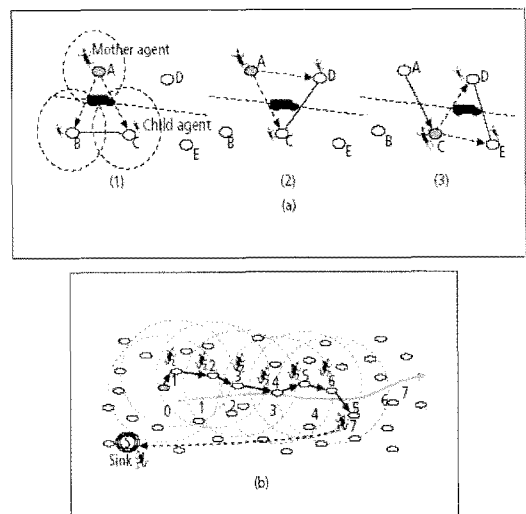


그림 3. 2가지 목표물 추적 응용의 예: (a) 지리적 위치 추적; (b) 객체 인식.

### 2.3 데이터 전송과 수집

무선 센서 네트워크에는 하나의 노드에서 다른 노드로 데이터의 라우팅, 전송, 수집을 위한 효율적인 메커니즘이 필요하다. [4]에서는 WSNs에서 P2P 멀티 에이전트 데이터 전송과 수집 시스템 구조를 제안하였다. 그 구조는 4가지 종류의 에이전트들: 인터페이스 에이전트, 질의 에이전트, 라우팅 에이전트, 그리고 데이터 획득 에이전트를 포함한다. 제안된 구조는 사용자들이 무선 센서 네트워크로부터 요구되는 정보를 효율적으로 감시하고 검색하는데 도움을 주기 위하여 설계되었다. 각 에이전트는 자치적이고, 지능적이고, 조정되고, 그리고 사용자의 요구를 실행하기 위하여 다른 에이전트들과 통신할 수 있다.

그림 4는 멀티 에이전트 데이터 전송 수집 시스템 구조에서 4가지 종류의 에이전트 기능을 보여준다. 이 구조에서, IA(interface agent)는 사용자 질의를 수신하고 QA(query agent)를 위한 SQL 비슷한 문을 생성한다. IA와 QA 둘 다 계산 집약적인 연산들을 요구하므로, 그것들은 자원이 풍부한 기지국에 위치되어진다. QA는 다른 사용자 질의를 수신하고, 최적화된 질의 실행 계획을 생성하고, 그리고 그 최적화된 계획을 RA(routing agent)에게 전송한다. RA는 에너지 효율적 데이

터 전송에 대한 책임이 있다. 여기서 최적화는 노드 밀도, 클러스터 구조, 가용 에너지와 같은 네트워크 조건에 기초한다. 센서 노드들은 제한된 에너지와 컴퓨팅 자원을 가짐에 따라, QA와 RA를 위한 피어들로 역할을 하는 프락시 에이전트들: PQA(proxy query agent)와 PRA(proxy routing agent)를 각각 생성한다. QA는 PQAs를 위한 질의 실행 계획을 제공하고, RA는 모든 PRAs를 위한 데이터 전송 계획을 제공한다. PQA는 센서 데이터를 획득하고, 여과하고, 그리고 구성하는데 책임이 있는 DAA(data acquisition agent)로부터 센서 데이터를 받는다.

### 3. 무선 센서망에서 모바일 에이전트 기반 데이터 전달 구조

WSN의 구조에 따라, 모바일 에이전트 기반 패러다임은 2가지 범주: 계층적 WSN 구조와 플랫폼 WSN 구조로 나누어진다. 계층적 WSN에서, 노드들의 역할과 기능은 같지 않다. 반면에 플랫폼 WSN에서 모든 노드들의 역할과 기능은 동일하다.

#### 3.1 계층적 센서망의 구조

계층적 WSN에서, 어떤 모바일 에이전트의 작업은 단순화되어진다. [5]에서, 2가지 하이브리드 모바일 에이전트 기반 방법들이 클러스터 된 WSNs에서 제안되었다. 인트라클러스터(intra-cluster) 방법에서, 각 클러스터 헤드는 데이터를 모으기 위하여 모든 클러스터 멤버들을 하나하나 방문하는 하나의 모바일 에이전트를 파견한다. 어떤 모바일 에이전트가 그것의 대응하는 클러스터 헤드로 반환될 때, 그것은 누적된 데이터를 처리 센터(예, 싱크 노드)로 전송한다. 반면에, 인터클러스터(inter-cluster) 방법은 어떤 클러스터 내에 모바일 에이전트 작업은 포함되지 않는다. 대신

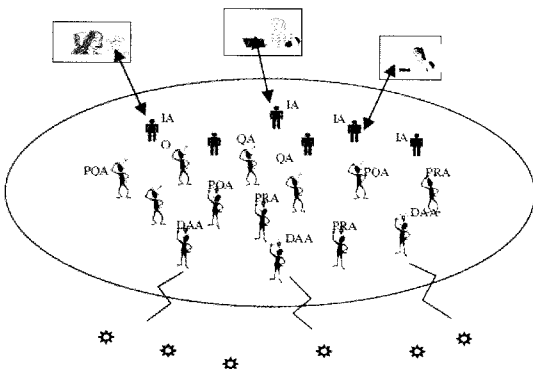


그림 4. 멀티 에이전트 데이터 전송 수집 시스템 구조

에, 어떤 모바일 에이전트가 클러스터 헤드들과 처리 센터 간을 이동한다.

인트라클러스터 방법은 하나의 클러스터내의 노드의 개수는 많고, 그러나 클러스터 헤드의 개수는 작은 시나리오를 선호한다. 이것에 비해, 인터클러스터 방법은 하나의 클러스터내의 노드의 개수가 상대적으로 작을 때 더 효율적이다.

어떤 계층적 WSN에서 특별한 구조를 설립하고 관리하는 것은 많은 제어 오버헤드를 요구할 수도 있다. 이러한 문제는 광범위한 센서 응용을 위하여 적합할 수도 있는 플랫폼 WSN 구조로 해결될 수 있다.

### 3.2 플랫폼 센서망의 구조

MADSN(mobile agent-based distributed sensor network) [6]은 계층적 WSN과 플랫폼 WSN 둘 다에서 채택될 수 있는 첫 번째 제안된 모바일 에이전트 패러다임이다. 그림 5(a)에서 보여진 플랫폼 WSNs에서처럼, 싱크는 목표 지역에서 데이터를 직접 모으기 위하여 특정한 개수의 모바일 에이전트들을 파견한다. [5]에서, 소스 노드들은 싱크 근처에 있다고 가정한다. 소스 노드들이 그 싱크로부터 훨씬 멀리 있는 경우에, 모바일 코드를 운반하는 각 에이전트는 첫 번째 소스 노드를 방문하기 전에 장거리를 이동할 수 있다. 싱크로부터의 모바일 코드 전송의 큰 오버헤드는

불필요할 뿐만 아니라 모바일 에이전트 사용의 이점을 약화시킨다. 따라서 [7]에서 MAWSN (mobile agent-based WSN)이라 부르는 이상적인 구조를 제안하였다.

그림 5(b)에서 보여진 것처럼, 부모 에이전트라 부르는 단지 하나의 모바일 에이전트가 싱크에서 목표지역으로 파견되어진다. 부모 에이전트는 항모와 유사하다. 초기에, 그 부모 에이전트는 그 모바일 코드를 목표지역 안에 있는 어떤 노드를 운반하고 그곳에 체재한다. 그 다음에, 부모 에이전트는 목표 지역 내의 센서 노드들에서 특정 작업을 실행하기 위하여 다수의 자식 에이전트들을 실행한다. 특정 목표로 가고 있는 항공기와 유사한 자식 에이전트는 데이터 수집과 집적 라운드에서 하나하나씩 소스 노드들의 배정된 집합을 방문할 것이고, 그 수집된 데이터를 부모 에이전트나 싱크로 직접 운반한다. 데이터 수집의 새로운 라운드를 초기화하기 위하여, 새로운 자식 에이전트들은 요구되는 데이터 보고 율에 따라 부모 에이전트에 의해 주기적으로 실행되어진다. 지연을 줄이기 위하여, 다수의 자식 에이전트들이 병렬로 작업을 수행하기 위하여 부모 에이전트에 의해 실행될 수도 있다. MAWSN에서, 모바일 에이전트의 사용은 3 단계 정보 중복 감축[7]이 가능하다.

- 노드 단계: 각 소스 노드에서 생성된 원시 데이터는 자식 에이전트 지원 로컬 처리에 의해 감축되어진다. 그 다음, 관련 정보만이 싱크로 전송되어진다.
- 자식 에이전트 단계: 노드들이 지리적으로 근접해 있고 그것들의 측정치 높은 수준의 상관관계를 보일 때 이 단계의 감축이 나타난다, 자식 에이전트가 이벤트 주변 소스 노드들을 하나하나씩 방문할 때 하나의 이벤트로부터 센서 데이터의 중복성을 줄이기 위하여 데이터 집적

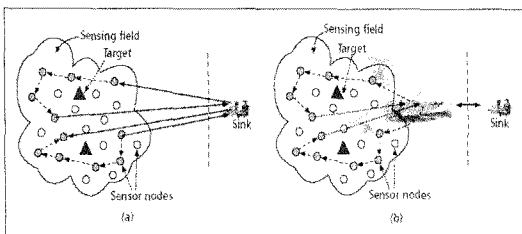


그림 5. 플랫폼 센서 네트워크에서 모바일 에이전트 기반 구조: (a) MADSN; (b) MAWSN.

이 수행되어진다.

- 부모 에이전트 단계: 다수의 자식 에이전트들이 부모 에이전트로 반환된 후, 부모 에이전트는 다양한 자식 에이전트에 의해 수집된 데이터에 존재하는 중복성을 더 감축할 수 있다.

#### 4. 순회 계획

순회(itinerary)는 모바일 에이전트 이동 동안에 따르는 경로로 정의되어진다. 순회 계획은 싱크에 의해 또는 모바일 에이전트에 의해 자동적으로 해결될 수 있는 다음 2가지 문제를 포함한다.

- 모바일 에이전트에 의해 방문되어질 소스 노드들의 집합의 선택
- 에너지 효율적인 방식으로 소스 방문 순서의 결정

모바일 에이전트가 선택된 소스 노드들을 방문하는 순서는 에너지 소비에 대한 중요한 영향을 줄 수 있다. 최적 소스 방문 순서를 찾는 것은 NP-완전 문제이다. 방문 계획은 다음과 같이 분류될 수 있다.

- 정적 계획: 에이전트 순회는 그 에이전트가 파견되기 전에 싱크 노드에 의해 완전히 결정되어진다.
- 동적 계획: 모바일 에이전트는 현재 네트워크 상태에 따라 방문될 소스 노드들과 이동 라우트를 자체적으로 결정한다.
- 하이브리드 계획: 방문될 소스 노드들의 집합은 싱크에 의해 결정되고, 소스 방문 순서는 모바일 에이전트에 의해 동적으로 결정되어진다.

##### (1) 정적 계획

모바일 에이전트가 전송되기 전에 정적 계획은 현재 광역 네트워크 정보를 사용하여 디스패처에서 효율적인 에이전트 경로가 유도된다. 2가지 방

법: LCF(local closest first)와 GCF(global closest first)가 제안되었다[8]. 두 알고리즘 모두 디스패처에 최근접한 동일한 노드에서 시작한다고 가정하고, LCF는 현재 노드와 가장 짧은 거리를 갖는 다음 노드를 검색한다. 반면에 GCF는 그 디스패처에 가장 가까운 다음 노드를 검색한다. 소스 노드들이 싱크와 비슷한 거리를 갖는 다수의 클러스터들을 형성한다면, GCF는 그러한 클러스터들 간의 순회 변동에 기인하여 지그재그 라우팅을 일으킨다. 이 문제는 LCF에 의해 해결될 수 있는데, 광범위한 시나리오에 적합할 수도 있다.

[9]에서, WSNs에서 모바일 에이전트 순회 계획을 위한 유전 알고리즘이 제안되었다. 탐색 공간을 줄이기 위하여 각 노드는 반복해서 방문될 수 없다고 가정한다. 비록 유전 알고리즘을 사용하여 광역 최적화를 이룰 수 있지만, 에너지 공급이 제한되는 센서 노드에 적합한 경량 해결책은 아니다.

##### (2) 동적 계획

싱크에서 수집된 광역 정보는 에이전트 이동에 따른 네트워크상의 변화에 기인하여 진부하게 될 수도 있기 때문에, 어떤 정적 순회는 어떤 동적 WSN에서 준최적이 될 수도 있다. 반면에, 동적 계획은 모바일 에이전트가 그것의 이동 경로의 각 체재에서 방문할 다음 노드 결정을 가능하게 한다. 동적 에이전트 라우트의 도출은 이동 비용과 이동 정확도의 이점 간의 상반관계를 고려해야 한다. [3]에서, 제안된 동적 계획 방법은 최대 가용 잔여 에너지를 가지고, 에이전트 이동을 위해 최소 에너지 소비를 요구하고, 최대량의 정보를 이익을 제공하는 센서 노드를 찾는다. 모바일 에이전트가 객체를 인식하기 위하여 방문되어야 하는 노드의 개수를 줄일 수 있도록 모바일 에이전트는 정확도를 점진적으로 증가시키는 센서 노드로 이

동을 항상 시도해야 한다.

### (3) 하이브리드 계획

하이브리드 계획에서, 소스 방문 집합의 선택은 정적이다. 반면에 소스 방문 순서의 결정은 동적이다. [1]에서, MADD(mobile agent-based directed diffusion)라 부르는 하이브리드 계획 방법이 제안되었다. MADD에서, 단일 목표 지역 내의 소스들이 관심 이벤트를 탐지하면, 그것들은 탐사 패킷을 싱크로 개별적으로 플로드 한다. 그러한 탐사 패킷들에 기초하여, 그 싱크는 어떤 모바일 에이전트에 의하여 방문될 소스들을 정적으로 선택하지만, 그 모바일 에이전트가 소스 방문 집합 내의 노드들 간의 이동하는 소스 방문 순서를 자체적으로 결정한다. 그 결과, 그 모바일 에이전트는 MADD로 목표 센서들 간의 비용이 효율적인 경로를 따라 간다.

## 5. 미들웨어 계층 설계

모바일 에이전트들은 미들웨어로 흔히 개발되어진다. 미들웨어는 운영체제와 고급 컴포넌트들 간의 틈을 메우기 위하여 그리고 애플리케이션의 개발과 전개를 용이하게 하기 위하여 컴퓨팅 시스템에서 폭넓게 사용되어진다. 자원이 풍부한 유선 네트워크를 위하여 설계된 모바일 에이전트 미들웨어는 폭넓은 다양한 작업을 수행하기 위하여 충분한 기능을 쉽게 제공할 수 있다. 그러나 WSN에서 가혹한 하드웨어와 에너지 제한은 실용적인 무선 센서 네트워크 모바일 에이전트 미들웨어 시스템(WMS) 설계에서 다수의 기술적인 문제들을 만들어 낸다. WMS는 모바일 에이전트들에게 애플리케이션 특정 태스크들을 지원하기 위한 플랫폼을 제공하고, WSNs에서 빠른 애플리케이션 개발을 가능하게 한다. 자원 제한, WSNs에서 그

것의 응용을 보장하기 위한 충분한 기능성의 수준 유지에 기인하여 요구사항에서부터 단순한 설계에 이르기 까지 상반관계가 발생한다. SensorWare [10]과 Agilla [11]은 WMS 설계에서 선구자적인 프레임워크들이다.

재구성 유연성을 유지하면서 WMS를 단순화하는 목표를 이루는 한 가지 방법은 WMS 언어 구조를 증가시키는 것이다. 특히 에이전트의 태스크 제어 부분을 더 표현력이 풍부하게 만드는 것이다. 이 방법은 에이전트 코드의 크기를 줄일 수 있을 뿐만 아니라 코드 인터프리터 자체의 설계를 단순화하고, 고유한 함수 라이브러리의 사용을 촉진시킨다. 결국, WSN 애플리케이션을 구현하는 프로그래머는 에이전트가 현장에서 사용할 있는 이미 존재하는 노드 기능, 에이전트의 저수준 잡일을 제거하는 것 그리고 에이전트의 크기를 줄이는 것에 의존할 수도 있다.

WMS의 태스크 제어 기능은 의도된 애플리케이션의 요구사항을 만족시켜야 한다. WMS 설계는 태스크 제어를 위해 고수준의 코드 추상화를 사용하는 코스 그레인드(coarse-grained)와 코드 추상화가 저수준에 있는 파인 그레인드(fine-grained)으로 분류될 수 있다. 예를 들어, 고수준 언어 구조는 SensorWare [10]에서 사용되었다. 여기서 하나의 코드 표현은 다수의 저수준 태스크들을 포함할 수 있다. 반면에, Agilla [11]은 대부분 저수준 코드 명령을 사용한다.

WSN 노드들의 자원 제한에 의존하여, 태스크 제어는 그림 6(a)에서 요약된 것처럼 분산 태스크 조정 또는 데이터의 로컬 처리를 위하여 설비될 수 있다. 파인 그레인드 태스크 제어는 고수준의 프로그램 실행 적응성을 제공하지만 프로그램이 길어질 수 있다. 코스 그레인드 모델은 태스크 제어와 재구성에서 덜한 적응성의 대가로 WMS 복

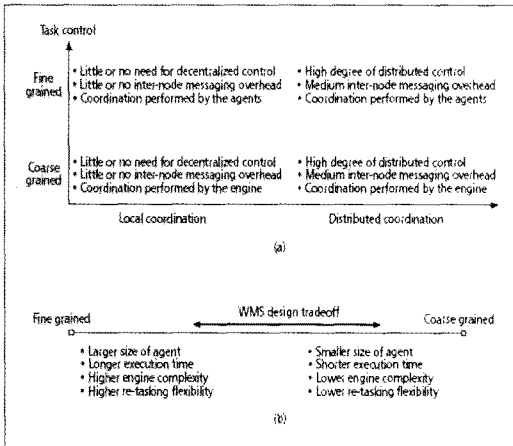


그림 6. WMS 프로그래밍 가이드라인: (a) 태스크 제어 모듈; (b) 상반관계 맵

잡도와 모바일 에이전트의 크기는 떨어질 것이다. 그림 6(b)에서 보여진 것처럼, 적절한 WMS 설계 상반관계는 센서 노드들의 기능에 따라 가능한 한 에너지를 절약하면서 애플리케이션 한정된 지연과 재구성 요구사항 만족시키기 위한 시도를 해야 한다.

### 6. 다수의 에이전트들 협동

모바일 에이전트는 하나의 처리 단위로 또는 주어진 작업을 성취하기 위하여 협동할 수 있는 컴포넌트들의 분산 집합으로 작동할 수 있다. 에이전트 협동을 위한 수단 제공에 대한 요구는 WMS 설계에서 중요한 고려사항이다. 왜냐하면 이 통신 메커니즘은 WSN에서 에너지 소비를 줄일 수 있다고 증명되었기 때문이다[2, 10, 11]. 덧붙여, 멀티-에이전트 협동은 라우팅과 같은 다른 WSN 관련 작업을 수행하는 에이전트 코드의 구현을 용이하게 한다. 본질적으로, 정보 공유는 에이전트들이 다른 에이전트들이 이미 무엇을 학습했는지 아는 것, 더 빠른 태스크 완료 시간, 잠재적인 대역폭 절약, 그리고 에너지 보존을 가능하게

한다.

에이전트들에 의해 교환되는 정보의 종류는 간단한 부울 값으로부터 복잡한 전처리된 데이터까지 광범위할 수도 있다. 이 데이터는 요구되는 결과가 얻어질 때 까지 다른 에이전트들에 의해 나중에 다시 처리될 수 있고 현재 분산 태스크의 전개에서 의사 결정을 위한 참조로 역할을 할 수 있다. 에이전트에 의해 사용되는 인기 있는 통신 메커니즘은 튜플 공간(tuple space)이다. 튜플 공간은 시간 분리 방식으로 참조하고 수정하기 위하여 다른 에이전트를 위하여 그 노드에 저장된 전형적인 정보이다. 예를 들어, 어떤 에이전트는 어떤 값들 (예, <string agentName, double time, bool foundResult>)을 표현하는 3가지 변수들의 집합을 저장할 수 있는데, 그 시맨틱 의미는 미리 정의되어지고 다른 에이전트들은 어떻게 액세스하는지를 알고 있다.

그림 7은 Agilla 모델을 보여준다. 각 노드는 동시에 4개의 에이전트들을 지원한다. Agilla는 에이전트들이 병렬로 그리고 독립적으로 실행을 허용하는 문맥 교환을 자동적으로 처리한다. Agilla는 에이전트 간 조정을 용이하게 하는 2가지 컴포넌트: 튜플 공간과 지식 리스트(acquaintance list)를 제공한다. 두 공간 모두 미들웨어에 의해 각 노드에서 관리된다. 어떤 노드의 튜플 공간은 로컬 에이전트들에 의해 공유되고

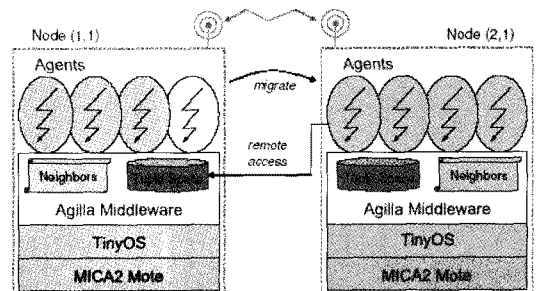


그림 7. Agilla 모델



원격에서 액세스 가능하다. 튜플 공간은 하나의 에이전트가 하나의 튜플을 삽입할 수 있고, 그리고 다른 에이전트가 나중에 읽거나 어떤 템플릿을 통한 패턴 매칭을 사용하여 그것을 제거하는 분리된 방식의 통신을 제공한다.

튜플 공간의 사용은 WMS 구조에서 일정한 량의 메모리와 데이터 처리 오버헤드를 분명히 도입함으로 시스템 오버헤드 대 성능 이익 간의 상반관계 분석이 요구된다. 반면에, 직접 에이전트 통신은 엄격한 시간제한이 있거나 메모리 또는 대역폭과 같은 자원의 많은 사용을 원하지 않을 때 필요할 수도 있다.

### 7. 결 론

일반적으로, 에이전트 시스템은 WSNs를 위한 다른 종류의 문제 해결 기법들에서 발견된 단점을 효율적으로 극복할 수 있다. 여기서 대역폭과 전력 소비가 주요 관심사이다. 이 논문에서, 애플리케이션, 구조, 순회 계획, 미들웨어 시스템, 그리고 다수의 에이전트 협동에 관하여 WSNs에 모바일 에이전트 시스템 전개에 속하는 주요 측면을 조사하였다. 그림 8(a)에서, 다수의 모바일 에이전트 시스템 구성요소들에 관한 핵심 설계 문제를 요약하였다. 그림 8(b)에서, 핵심 설계 컴포넌트들에 관한 기존 연구들을 분류하였다. 애플리케이션에 관해서는, 어느 정도의 대역폭 절약은 많은 데이터가 에이전트 코드에 의한 명령으로 로컬에서 처리될 때 또는 태스크 자율성을 용이하게 하는 프로그램 가능한 방식이 요구될 때 달성될 수 있다. WSN 응용은 에이전트의 태스크를 효율적인 수행하기 위하여 모바일 에이전트 시스템에 의해 사용되는 통신 매커니즘의 유형에 직접 영향을 받는다.

향후 과제로는 WSNs를 위한 모바일 에이전트

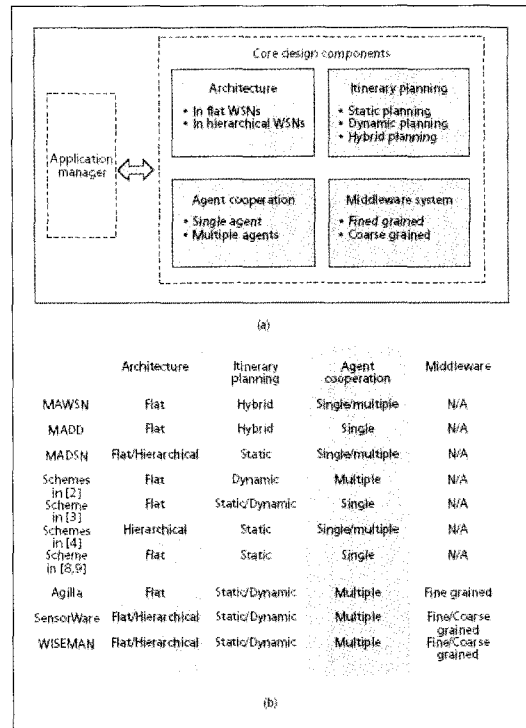


그림 8. WSNs에서 모바일 에이전트 기반 시스템 설계의 개요: (a) 핵심 설계 컴포넌트; (b) 기존 프로토콜의 분류와 비교

미들웨어에서 센서 정보를 퓨전하고 추론하는 컴포넌트에 관한 연구와 센서 데이터 조합을 통하여 고 수준의 동적 상황 정보를 유도하는 방법에 관한 연구 등이 있다.

### 참 고 문 헌

[1] M. Chen *et al.*, "Mobile Agent-Based Directed Diffusion in Wireless Sensor Networks," *URASIP J. Advances in Sig. Processing*, Vol. 2007.

[2] Y. Tseng *et al.*, "Location Tracking in a Wireless Sensor Network by Mobile Agents and Its Data Fusion Strategies," *Comp. J.*, Vol. 47, No. 4, pp. 448-60, July 2004.

[3] Y. Xu and H. Qi, "Mobile Agent Migration Modeling and Design for Target Tracking in

- Wireless Sensor Networks," *Ad Hoc Networks*, Vol. 6, No. 1, pp. 1-16, Jan. 2007.
- [ 4 ] E. Shakhshuki, S. Hussain, A. R. Matin, A. W. Matin, "P2P Multi-agent data Transfer and Aggregation in Wireless Sensor Networks," *Int. Conf. on Mobile Ad Hoc and Sensor Systems*, pp. 645-649, 2009.
- [ 5 ] Y. Xu and H. Qi, "Distributed Computing Paradigms for Collaborative Signal and Information Processing in Sensor Networks," *Int'l. J. Parallel and Distrib. Comp.*, Vol. 64, No. 8, pp. 945-59., Aug. 2004.
- [ 6 ] H. Qi, Y. Xu, and X. Wang, Mobile-Agent-Based Collaborative Signal and Information Processing in Sensor Networks," *Proc. IEEE*, Vol. 91, No. 8, pp. 1172-83, Aug. 2003.
- [ 7 ] M. Chen *et al.*, "Mobile Agent Based Wireless Sensor Networks," *J. Comp.*, Vol. 1, No. 1, Apr. pp. 14-21, 2006.
- [ 8 ] H. Qi and F. Wang, "Optimal Itinerary Analysis for Mobile Agents in Ad Hoc Wireless Sensor Networks," *Proc. IEEE ICC '01*, pp. 147-53, July 2001.
- [ 9 ] Q. Wu *et al.*, "On Computing Mobile Agent Routes for Data Fusion in Distributed Sensor Networks," *IEEE Trans. Knowledge and Data Eng.*, Vol. 16, No. 6, pp. 740-53, June 2004.
- [10] A. Boulis, C. Han, and M. Srivastava, "Design and Implementation of a Framework for Efficient and Programmable Sensor Networks," *Proc. ACM MobiSys '03*, pp. 187-200, May 2003.
- [11] C. Fok, G. Roman, and C. Lu, "Mobile Agent Middleware for Sensor Networks: An Application Case Study," *Proc. IEEE IPSN '05*, Apr. 2005.
- [12] S. Gonzalez, S. Vuong, and V. C. M. Leung, "A Mobile Code Platform for Distributed Task Control in Wireless Sensor Networks," *Proc. ACM MobiDE*, pp. 83-86, June 2006.
- [13] I. F. Akyildiz, T. Melodia, and K. Chowdhury, "A Survey on Wireless Multimedia Sensor Networks," *Comp. Networks*, Vol. 51, No. 4, pp. 921-60, Mar. 2007.



배 인 한

- 1984년 경남대학교 전자계산학과(공학사)
- 1986년 중앙대학교 대학원 전자계산학과(이학석사)
- 1990년 중앙대학교 대학원 전자계산학과(공학박사)
- 1996년~1997년 Department of Computer Science and Engineering, The Ohio State University(Post- Doc)
- 2002년~2003년 Department of Computer Science, Old Dominion University (Visiting Professor)
- 2009년~현재 Department of Computer Science, Old Dominion University (Visiting Professor)
- 1989년~현재 대구가톨릭대학교 컴퓨터정보통신공학부 교수
- 관심분야 : 센서 네트워크, 스마트 응용, 시맨틱 모바일 멀티미디어, 미들웨어 등