지속적인 이동성을 갖는 이동 센서네트워크를 위한 통신 프로토콜

김 형 진[†] · 김 래 영[††] · 송 주 석[†††]

요 약

최근 Robomote, Robotic Sensor Agents(RSAs)와 같은 이동 센서의 등장으로 인해 이동 센서네트워크(MSN: Mobile Sensor Network)에 대한 연구가 활발히 진행되고 있다. 하지만 기존의 이동 센서네트워크에 대한 연구는 주로 기존의 고정 센서네트워크(SSN: Stationary Sensor Network)에서 발생하는 문제점인 Coverage Hole을 해결하는데 초점을 맞추고 있다. 이러한 연구들에서는 이동 센서들에게 부여된 이동 능력을 최대한 활용하지 못하는 단점을 안고 있다. 이를 해결하기 위해 이동 센서에게 지속적인 이동성을 부여함으로써 고정 센서네트워크에 비해 더 넓은 영역을 센싱하도록 제안한 연구가 있으나, 그 연구가 아직 초기 단계로써 이동 센서의 지속적인 이동으로 인한 싱크 노드로의 통신 경로 설정 및 데이터 전송 문제에 대해서는 논하고 있지 않다. 이에 본 논문에서는 지속적인 이동성을 갖는 이동 센서로 구성된 이동 센서네트워크 환경에서 효율적으로 경로 설정 및 데이터 전송을 가능하게 하는 통신 프로토콜을 제안한다. 제안하는 프로토콜에서는 이동 센서와 함께 고정 센서를 배치함으로써 고정 센서가 이동 센서를 대신하여 싱크 노드로 센싱 데이터를 전송하도록 한다. 시뮬레이션을 이용한 성능 평가를 통해 제안한 통신 프로토콜이 기존의 고정 센서네트워크에 비해 네트워크 커버리지 면에서 최대 40%, 트래픽 오버헤드 부분에서는 최대 76%의 성능을 향상시킴을 보인다.

키워드 : 이동 센서네트워크, 이동 센서, 통신 프로토콜, 네트워크 커버리지, 트래픽 오버헤드

Communication Protocol for Mobile Sensor Networks with Continuous Mobility

HyoungJin Kim[†] · LaeYoung Kim[††] · JooSeok Song[†††]

ABSTRACT

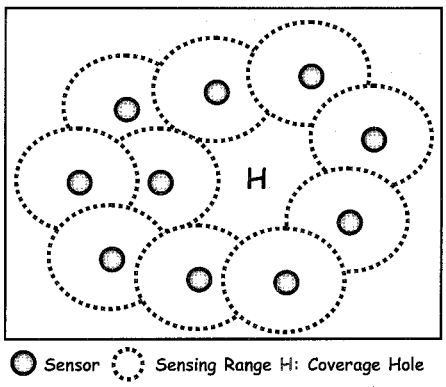
Mobile Sensor Network(MSN) is actively studied due to the advent of mobile sensors such as Robomote and Robotic Sensor Agents(RSAs). However, existing studies on MSN have mainly focused on coverage hole problem which occurs in Stationary Sensor Network(SSN). To address coverage hole problem, these studies make mobile sensors move temporarily so that they do not make the best use of the mobility of mobile sensors. Thus, a mechanism utilizing the continuous movement of mobile sensors is proposed to improve the network coverage performance. However, this mechanism is presently immature and does not explain how to make routing path and send data from mobile sensors to a sink node. Therefore, to efficiently make routing path and send data from mobile sensors to a sink node, we propose a communication protocol for mobile sensor network where mobile sensors continuously move. The proposed protocol deploys not only mobile sensors but also stationary sensors which send sensing data to a sink node instead of mobile sensors. Simulation results show that the proposed protocol improves the performance in terms of network coverage and traffic overhead, compared to conventional SSN protocols.

Key Words : Mobile Sensor Network, Mobile Sensor, Communication Protocol, Network Coverage, Traffic Overhead

1. 서 론

최근 Robomote[1], Robotic Sensor Agents(RSAs)[2]와 같이 이동성을 갖는 이동 센서(MS: Mobile Sensor)의 등장으로 인해 이동 센서네트워크(MSN: Mobile Sensor Network)에 대한 연구가 활발히 진행되고 있다. 이동 센서네트워크 환경에서는 이동 센서들의 위치 이동 능력으로 인해 고정 센서(SS: Stationary Sensor)로만 구성된 고정 센서네트워크(SSN: Stationary Sensor Network)에 비해 네트워크 커버리지(Network Coverage)를 확장할 수 있다는 장점을 갖고 있다[3-7]. 또한 이동 센서네트워크는 사람이 직접 탐지하기

※ 본 연구는 한국과학재단 특정기초연구(R01-2006-000-10614-0)지원으로 수행되었음.
† 준 회 원 : 연세대학교 컴퓨터과학과 석사
†† 준 회 원 : 연세대학교 컴퓨터과학과 박사
††† 종신회원 : 연세대학교 컴퓨터과학과 정교수
논문접수 : 2006년 10월 3일, 심사완료 : 2007년 2월 15일



(그림 1) 센서네트워크의 Coverage Hole 예

어려운 화산, 지진 등의 자연재해 현장과 군사 목적에도 효과적으로 사용할 수 있다[4].

센서네트워크의 경우 보통 센서를 배치할 때 비행기 등을 이용하여 센서를 랜덤하게 뿌리기 때문에 (그림 1)에서 볼 수 있듯이 센서가 닿을 수 없는 Coverage Hole이 존재할 가능성이 높다. 따라서 대부분의 이동 센서네트워크와 관련한 기존의 연구들은 Coverage Hole 문제를 해결하기 위하여 이동 센서를 이용하는 방안을 연구하고 있다.

[4]에서는 센서들이 클러스터를 중심으로 네트워크 커버리지를 넓히는 방법을 제안하였는데, 각 클러스터의 중심인 클러스터 헤드(Cluster-head)가 Virtual Force Algorithm을 사용하여 자신의 클러스터에 속한 센서들에게 최적인 위치를 찾아내고 센서들을 해당 위치로 이동시킨다. 이러한 방법을 사용하여 각 클러스터의 커버리지를 넓힘으로써 전체 네트워크의 커버리지가 넓어지는 효과가 있다. [5]에서는 [4]에서 제안한 클러스터 중심 프로토콜이 실제 전쟁 상황이나 자연 재해로 인한 탐사 작업에서는 적용하기 어려운 문제점을 발견하고, 클러스터 중심에서 벗어나 센서들 스스로 자신의 최적의 위치를 찾아내는 방법을 제안하였다. 각 센서들은 자신의 최적 위치를 찾기 위해 두 가지 단계를 수행하는데 먼저 Voronoi Diagram을 이용하여 Coverage Hole을 찾아낸 후 Vector 기반, Voronoi 기반, Minimax 프로토콜을 이용하여 자신의 최적 위치를 찾아 이동한다. [4],[5]가 네트워크 설정 초기시의 센서 이동에 관한 방안을 제안한데 반해, [6]에서는 센서 고장 혹은 새로운 이벤트 발생 시 센서의 위치를 효율적으로 재이동하는 방법을 제안하였다. 특히 센서의 이동을 최소화하기 위해 특정 지역에 집중되어 있는 센서 그룹을 찾아내고 해당 그룹에서 목표 위치(즉, 실패 혹은 새로운 이벤트가 발견된 위치)로 갈 수 있는 경로를 찾은 뒤 그 경로에 속한 센서 전체가 부분적으로 움직여 센서가 목표 위치를 센싱할 수 있는 방법을 제안하였다. 그러나 이러한 기존의 연구들은 이동 센서들이 Coverage Hole을 채우기 위해 움직일 때를 제외하고는 고정 센서와 마찬가지로 대부분의 시간을 정지해 있고, 더욱이 대부분의 이동 센서들은 초기에 뿌려진 후, 한 번도 이동하지 않기 때문에 이동 센서의 장점을 최대한 살리지 못하는 단점을 안고 있다[7].

이러한 단점을 보완하기 위해, 즉 이동 센서의 이동성을 최대한 활용하기 위해 [7]에서는 모든 이동 센서를 랜덤한 방향과 속도로 지속적으로 움직이게 하는 방안을 제안하였다. 이 연구에서는 모든 센서들이 지속적으로 움직일 경우 고정 센서네트워크에 비해 더 넓은 영역을 센싱할 수 있음을 수학적으로 증명하였다. 또한, 침입자 탐지 시스템에 대한 방안을 제안하였는데, 이동 센서들이 일정한 주기로 균등하게 방향을 바꿔 움직이게 되면 짧은 시간 안에 침입자를 탐지할 수 있음을 증명하였다. 하지만 이 연구에서는 센서들의 지속적인 이동에 따른 싱크 노드(sink node)로의 경로 설정/재설정에 대해서는 논하고 있지 않다.

[4],[5],[6]의 경우 일부의 센서가 이동한 후에만 싱크 노드로의 경로를 재설정하면 되지만, [7]의 경우 센서들의 위치가 계속적으로 변하기 때문에 이동 센서들이 센싱한 데이터를 싱크 노드로 전송하기 위한 경로 설정에 어려움이 따른다.

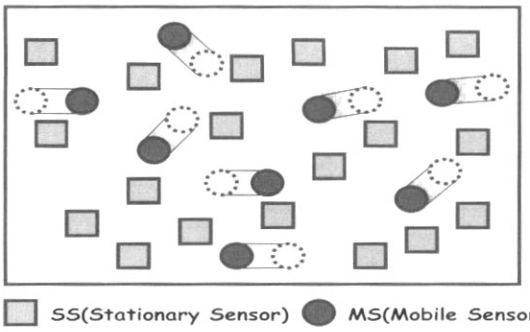
본 논문에서는 이처럼 이동 센서가 지속적으로 이동함으로써 발생하는 경로 설정 및 데이터 전송의 문제점을 해결하기 위하여 지속적인 이동성을 갖는 이동 센서네트워크를 위한 통신 프로토콜을 제안하고자 한다. 제안하는 방안에서는 이동 센서와 함께 고정 센서를 배치함으로써 고정 센서가 이동 센서를 대신하여 싱크 노드로 센싱 데이터를 전송하도록 한다. 이동 센서는 센싱한 데이터를 가장 가까운 위치에 있는 고정 센서에게만 전송하면 되고 싱크 노드로의 라우팅에는 참여하지 않는다. 고정 센서가 이동 센서로부터 받은 데이터를 고정 센서들 간에 설정된 경로를 이용하여 싱크 노드로 전달함으로써 이동 센서의 지속적인 이동에 따른 경로 설정 및 데이터 전송 문제를 해결할 수 있다. 지속적으로 움직이는 이동 센서의 경우 고정 센서에 비해 에너지가 더 빨리 소모되는 단점을 갖고 있으나 이는 태양열을 이용한 배터리 충전[8], 이동 센서가 직접 센서들에게 에너지를 전달해주는 배터리 충전[9], 진동을 이용한 배터리 충전[10] 등의 다양한 방식으로 해결이 가능하기 때문에 더 이상 큰 문제가 되지 않는다.

2장에서는 본 논문에서 제안하는 지속적인 이동성을 갖는 이동 센서네트워크를 위한 통신 프로토콜에 대해서 자세히 살펴보며, 3장에서는 제안한 방식의 성능을 네트워크 커버리지와 트래픽 오버헤드 측면에서 평가한다. 마지막으로 4장에서는 결론을 맺는다.

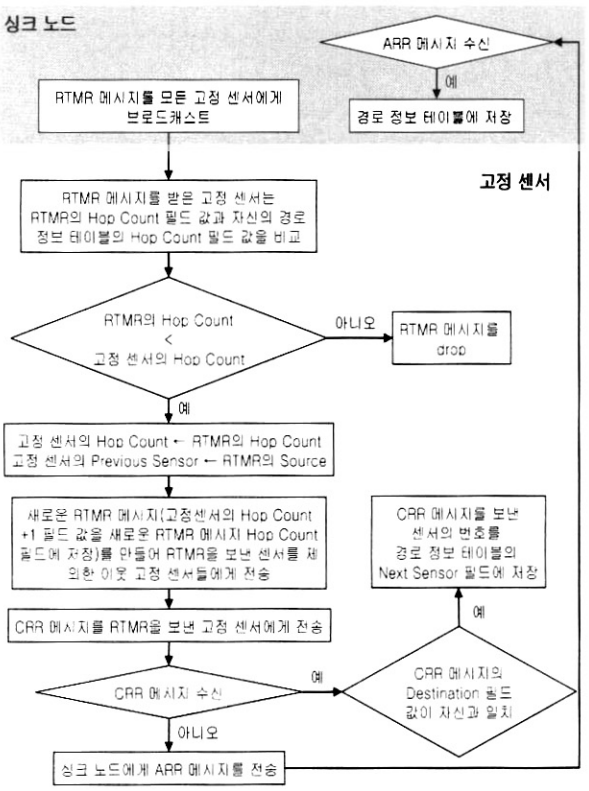
2. 지속적인 이동성을 갖는 이동 센서네트워크를 위한 통신 프로토콜

2.1 이동 센서네트워크의 배치(Deployment)

본 논문에서 제안하는 이동 센서네트워크는 일정한 속도로 랜덤한 방향으로 움직일 수 있는 이동 센서와 이동성 능력이 없는 고정 센서로 구성된다. 이동 센서는 센싱과 통신 기능을 수행하며, 고정 센서는 센싱과 통신 기능에 더해 싱크 노드로의 라우팅 기능도 수행한다. 따라서, 이동 센서의



(그림 2) 제안하는 이동 센서네트워크의 배치



(그림 3) CPMSN의 경로 설정 과정

지속적인 위치 이동으로 인하여 생기는 라우팅의 어려움을 고정 센서가 해결하여 준다. 본 논문에서는 모든 고정 센서가 하나 이상의 이웃 고정 센서와 통신이 가능한 범위에 있다고 가정한다. (그림 2)는 제안하는 이동 센서네트워크의 배치(deployment)를 보여준다.

2.2 경로 설정

센서네트워크를 구성하기 위해 센서들이 배치된 후, 고정 센서들 간의 경로 설정 과정이 수행된다. (그림 3)은 본 논문에서 제안하는 지속적인 이동성을 갖는 이동 센서네트워크를 위한 통신 프로토콜(CPMSN: Communication Protocol for Mobile Sensor Networks)의 경로 설정 과정을 보여주며, 이를 자세히 살펴보면 다음과 같다.

〈표 1〉 RTMR 메시지 구성 정보

Packet Field	Description
Type of Packet	패킷의 종류, RTMR 메시지는 1로 설정
Source	해당 RTMR 메시지를 전송한 이전 센서(또는 싱크 노드) (s_i)
Hop Count	싱크 노드에서 RTMR 메시지를 수신한 센서까지의 홉 수

〈표 2〉 고정 센서의 경로 정보 테이블

Table Field	Description
Previous Sensor	최적 경로의 이전 센서 주소 (s_i)
Next Sensor	최적 경로의 다음 센서 주소 ($s_j, \ldots, s_k$)
Hop count	최적 경로 홉 수

〈표 3〉 CRR 메시지 구성 정보

Packet Field	Description
Type of Packet	패킷의 종류, CRR 메시지는 2로 설정
Source	CRR 메시지를 보낸 센서 주소 (s_i)
Destination	해당 RTMR 메시지를 전송한 이전 센서(또는 싱크 노드) (s_j)

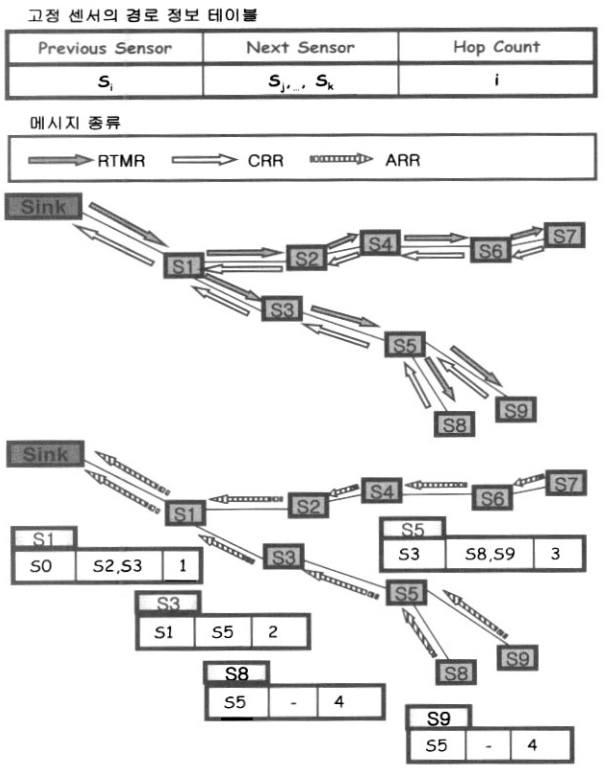
단계 1. 통신 경로를 만들기 위해 싱크 노드는 RTMR (Request To Make Routing) 메시지를 만들어 모든 고정 센서들에게 브로드캐스트한다. 여기에서 RTMR 메시지는 해당 RTMR 메시지를 전송한 이전 센서(또는 싱크 노드)를 나타내는 Source 필드, 싱크 노드로부터 RTMR 메시지를 수신한 센서까지의 홉 수를 나타내는 Hop Count 필드로 구성된다. 싱크 노드는 RTMR 메시지의 Source, Hop Count 필드를 각각 0, 1로 초기화하여 보낸다. <표 1>은 RTMR 메시지의 구성 정보를 나타낸다.

단계 2. RTMR 메시지를 받은 고정 센서는 RTMR 메시지에 있는 Hop Count 필드 값과 <표 2>와 같은 자신의 경로 정보 테이블에 저장되어 있는 Hop Count 필드 값을 비교한다. 이때 RTMR 메시지의 Hop Count 필드 값이 더 작으면 단계 2-1을 수행하고, 그렇지 않으면 단계 2-2를 수행한다. 고정 센서가 가지고 있는 경로 정보 테이블의 Previous Sensor, Next Sensor, Hop Count 필드의 초기 값은 각각 -1, -1, ∞ 이다.

단계 2-1. 고정 센서는 수신한 RTMR 메시지의 Source 필드 값과 Hop Count 필드 값을 자신의 경로 정보 테이블의 Previous Sensor와 Hop Count 필드에 각각 저장한다. RTMR 메시지에 대한 응답으로 CRR(Cover Routing Reply) 메시지를 생성하여 RTMR 메시지를 보낸 고정 센서에게 전송한다. <표 3>에서 볼 수 있듯이 CRR 메시지는 해당 CRR 메시지를 전송한 센서를 나타내는 Source 필드, RTMR 메시지를 전송한 이전 센서(또는 싱크 노드)를 나타내는 Destination 필드로 구성된다. 단계 3을 수행한다.

단계 2-2. 고정 센서 자신의 경로 정보 테이블을 변경 없이 그대로 유지한다. 단계 4를 수행한다.

단계 3. 고정 센서는 Source, Hop Count 필드 값이 각각 자신의 노드 번호, 자신의 경로 정보 테이블의 Hop Count 필드 값에 1을 더한 값으로 설정된 RTMR 메시지를 만들어 자신에게 RTMR 메시지를 보낸 고정 센서를 제외한 이웃 고정 센서들에게 전송한다.



(그림 4) CPMSN의 경로 설정 과정 예

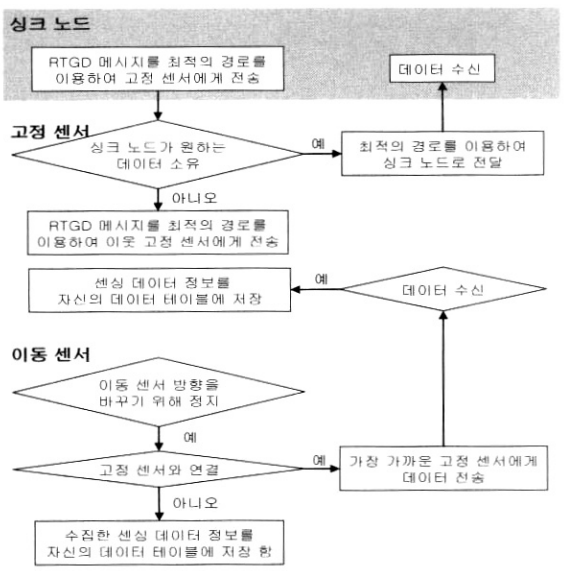
단계 4. 새로운 RTMR 메시지를 보낸 고정 센서는 일정시간 동안 CRR 메시지를 기다린다. CRR 메시지를 받은 고정 센서는 CRR 메시지의 Destination 필드 값이 자신의 노드 번호와 일치할 경우, CRR 메시지를 보낸 고정 센서의 번호를 자신의 경로 정보 테이블에 있는 Next Sensor 필드에 저장한다. 여러 개의 센서들로부터 CRR 메시지를 받은 경우에는 CRR 메시지의 Destination 필드 값이 자신의 노드 번호와 일치하면 이를 보낸 센서들의 번호를 모두 Next Sensor 필드에 저장한다.

만약, 일정시간 안에 CRR 메시지를 받지 못하면 고정 센서는 자신의 위치가 통신 경로의 끝이라고 간주하고 싱크 노드에게 ARR(Assure Routing Reply) 메시지를 전송한다. 이 때 ARR 메시지는 해당 ARR 메시지를 생성한 센서를 나타내는 Source 필드와 해당 ARR 메시지를 생성한 센서의 Hop 수를 나타내는 Hop Count 필드로 구성되며, ARR 메시지임을 표시하기 위해 Type of Packet 필드는 3으로 표시한다. 이 ARR 메시지는 이미 완성된 최적의 통신 경로를 따라 싱크 노드까지 전송된다.

단계 5. 싱크 노드가 ARR 메시지를 받으면 ARR 메시지에 담겨진 경로 정보를 싱크 노드 자신의 경로 정보 테이블에 저장한다. 싱크 노드의 경로 정보 테이블은 통신 경로의 마지막 센서를 나타내는 Destination 필드, 해당 경로의 최대 Hop 수를 나타내는 Hop Count 필드로 구성된다.

(그림 4)는 경로 설정 과정의 예를 보여준다.

2.3 데이터 전송(Data Forwarding)



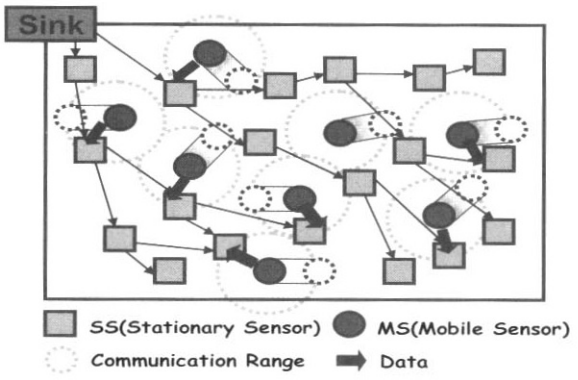
(그림 5) CPMSN의 데이터 전송 과정

〈표 4〉 RTGD 메시지 구성 정보

Packet Field	Description
Type of Packet	패킷의 종류, RTGD 메시지는 4로 설정
Query	싱크 노드의 요청 정보
Source of SS	RTGD 메시지를 보낸 고정 센서 주소

센서들이 센싱한 데이터를 수집하기 위해 싱크 노드는 RTGD(Request To Get Data) 메시지를 통해 센서들에게 데이터를 요청한다. 여기서 데이터는 싱크 노드가 얻고자 하는 정보로 특정 지역의 온도, 습도 등을 예로 들 수 있다. RTGD 메시지는 경로 설정 과정을 통해 설정된 통신 경로를 따라 고정 센서들에게 전송되며, 이동 센서에게는 전송되지 않는다. 이는 고정 센서와 이동 센서 사이의 지속적인 통신이 어렵기 때문이다. (그림 5)는 CPMSN의 데이터 전송 과정을 보여주며 <표 4>는 RTGD 메시지의 구성 정보를 보여준다.

이동 센서는 RTGD 메시지를 못 받기 때문에 싱크 노드가 원하는 데이터를 가지고 있어도 해당 데이터를 보내 줄 수 없게 된다. 이를 해결하기 위해 이동 센서는 주기적으로 움직임의 방향을 바꿀 때마다 고정 센서와 통신 연결의 유, 무를 확인하고 연결될 경우 이동 센서가 이동하면서 수집한 센싱 데이터를 고정 센서에게 전송한다. 따라서, 고정 센서는 자신이 수집한 센싱 데이터와 함께 이동 센서로부터 받



(그림 6) 이동 센서가 고정 센서에게 센싱 데이터를 전송하는 예

은 데이터도 자신의 데이터 테이블에 저장한다. 이때 데이터가 수집된 위치를 기준으로 분류하여 저장한다. 각 데이터는 수집된 시각이 함께 기록되며 각 데이터마다 타이머를 두어 타이머가 종료될 때까지 싱크 노드로부터 해당 데이터에 대한 전송 요청이 없으면 데이터를 삭제한다. 이는 고정 센서의 메모리를 절약하기 위함이다. (그림 6)은 이동 센서가 고정 센서에게 센싱 데이터를 전송하는 예를 보여준다.

고정 센서는 RTGD 메시지를 받으면 자신이 센싱한 데이터와 이동 센서로부터 받은 데이터가 저장된 데이터 테이블의 모든 센싱 정보를 검색하여 싱크 노드가 원하는 데이터를 가지고 있는지 확인한다. 만약, 해당 센싱 데이터를 가지고 있으면 최적의 경로를 이용하여 싱크 노드에게 전송한 후, 이 데이터를 테이블에서 삭제한다.

3. 성능 평가

본 장에서는 제안한 프로토콜인 CPMSN의 성능을 NS2를 이용한 시뮬레이션을 통해 네트워크 커버리지와 트래픽 오버헤드 측면에서 평가하였다. 네트워크 커버리지의 경우 이동 센서가 없이 고정 센서만으로 구성된 SSN과 비교하였다. 싱크 노드와 센서 사이의 통신 경로를 설정하는 과정과 소스에서 싱크 노드에게 데이터를 전달하는 과정에서 생기는 트래픽 오버헤드의 경우 SSN에서 사용하는 대표적인 통신 프로토콜인 Flooding 및 MCF(Minimum Cost Forwarding) [11]와 비교하였다. Flooding은 일정한 통신 경로 없이 각각의 센서들이 자신에게 메시지를 전송한 센서를 제외한 모든 이웃 센서들에게 메시지를 전송하는 가장 기본적인 통신 프로토콜이다. MCF는 비용(cost)을 중심으로 통신 경로를 설정한 뒤 메시지를 주고받는 방법으로 통신 경로 설정 시에는 Flooding과 비슷한 방법을 사용하고 통신 경로가 설정된 뒤에는 해당 통신 경로를 통해 메시지를 주고받는 방법이다.

3.1 네트워크 커버리지 (Network Coverage)

제안한 CPMSN의 네트워크 커버리지에 대한 성능을 측정하기 위한 시뮬레이션 토폴로지 구성은 $30\times30m^2$의 면적에 이동 센서와 고정 센서의 수가 전체 센서에서 각각 25%와 75%를 차지하도록 하였으며 랜덤하게 위치하도록 하였다. 또한, 이동 센서의 움직임을 효율적으로 시뮬레이션하기 위해 이동 센서는 일정한 속도(0.5m/s)로 움직이게 하였고 동일한 방향으로 5m를 이동한 뒤에는 움직이는 방향이 랜덤하게 바뀌도록 설정하였다. SSN의 경우에도 $30\times30m^2$ 면적의 시뮬레이션 토폴로지를 사용하였다.

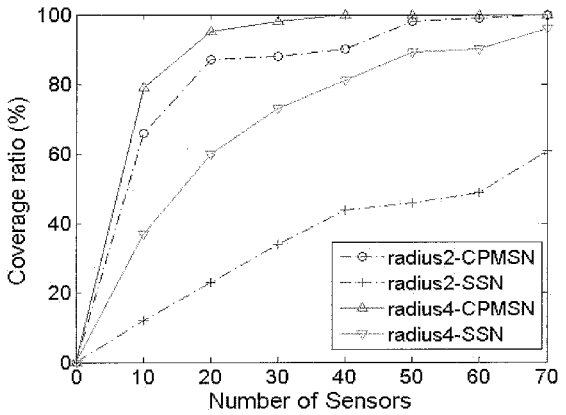
(그림 7)은 네트워크를 구성하는 센서의 개수와 센싱 반경에 따라 CPMSN과 SSN이 네트워크의 전체 면적에서 몇 퍼센트의 영역을 센싱할 수 있는지(즉, 네트워크 커버리지)를 나타내는 그림이다. SSN과 제안하는 CPMSN에 대하여 센서의 센싱 반경을 2m와 4m로 설정하여 시뮬레이션을 수행하였다. (그림 7)에서 볼 수 있듯이 고정 센서로만 구성된 SSN의 경우, 센서의 수가 늘어날수록 네트워크 커버리지 비율은 커진다. 또한, SSN에서 동일한 수의 센서를 사용하면, 센서의 센싱 반경이 더 큰 경우에 더 넓은 영역을 센싱할 수 있음을 볼 수 있다. 하지만 SSN은 센싱 반경이 4m인 센서 50개로 89%의 영역만을 센싱할 수 있지만 본 논문에서 제안하는 CPMSN은 센서의 센싱 반경이 2m와 4m일 때 50개의 센서로 각각 전체 네트워크의 98%와 100%를 센싱할 수 있음을 보여준다. 이는 제안한 CPMSN이 SSN과 비교하여 더 작은 개수의 센서를 가지고도 이동 센서의 이동성으로 인해 더 넓은 영역을 센싱할 수 있음을 보여준다.

(그림 8)은 CPMSN에서 이동 센서의 이동 속도에 따른 네트워크 커버리지를 나타내는 그림이다. SSN과 제안하는 CPMSN에 대하여 센서의 센싱 반경을 2m로 설정하여 시뮬레이션을 수행하였다. (그림 8)에서 볼 수 있듯이 초반의 CPMSN은 이동 센서 속도가 빠를수록 더 적은 센서의 수로 전체 영역을 센싱할 수 있지만 센서 개수가 50개부터는 0.5m/s, 0.7m/s, 1m/s의 이동 속도와 무관하게 $30\times30m^2$ 영역 전체를 센싱할 수 있음을 알 수 있다. 이는 0.5m/s의 속도만으로도 같은 개수의 센서를 이용하여 넓은 영역을 센싱할 수 있음을 알 수 있다. 또한, 0.2m/s의 이동 속도를 갖는 CPMSN도 SSN보다 더 넓은 영역을 센싱할 수 있음을 알 수 있다.

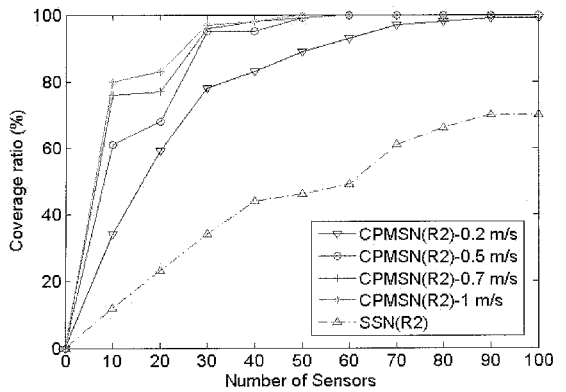
3.2 트래픽 오버헤드 (Traffic Overhead)

트래픽 오버헤드를 측정하기 위한 시뮬레이션 토폴로지는 네트워크 커버리지를 측정할 때와 마찬가지로 $30\times30m^2$의 면적에 50개의 센서를 사용하였고 센서 노드간의 메시지 충돌은 없다고 가정하였다. 제안한 CPMSN은 이동 센서와 고정 센서의 수가 전체 센서에서 각각 25%와 75%를 차지하도록 하였다. Flooding과 MCF의 경우 데이터 전송(data forwarding) 단계에서 인접한 센서간의 유사한 정보의 중복 전달로 인한 에너지 낭비를 줄이기 위해 제안된 방식인 데이터 병합(data aggregation)을 사용하도록 시뮬레이션 하였다.

(그림 9)는 CPMSN, Flooding, MCF의 프로토콜에 대하여 총 20번의 라운드(round)에서의 매 라운드 당 트래픽 오버헤드를 측정한 그림이다. 여기서 라운드는 싱크 노드로 데이터가 전송되기까지의 과정으로, Flooding의 경우 각 센서 노드들이 통신 경로를 저장하지 않기 때문에 매 라운드마다 통신 경로를 설정한 후 데이터를 전송하는 반면,



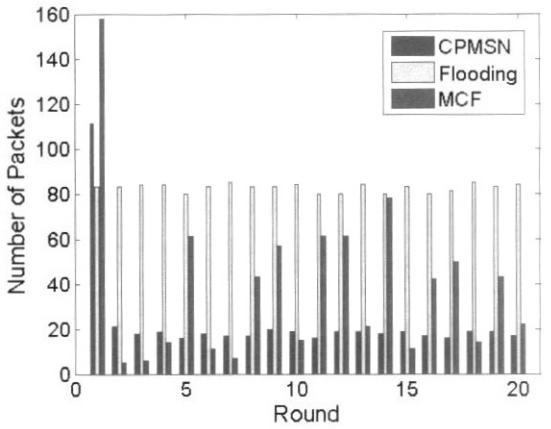
(그림 7) 센서의 개수와 센싱 반경에 따른 네트워크 커버리지 (%)



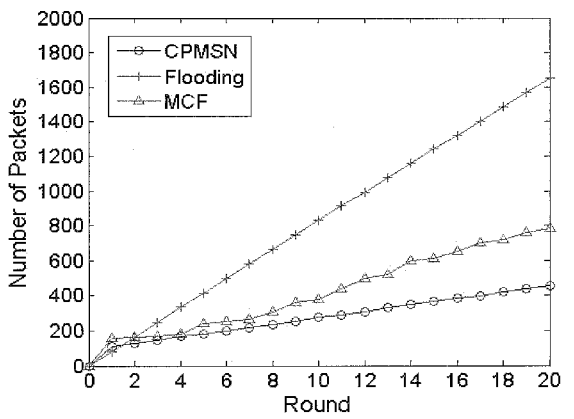
(그림 8) 이동 센서의 속도에 따른 네트워크 커버리지 (%)

CPMSN과 MCF는 첫 번째 라운드에서만 통신 경로를 설정하며 두 번째 라운드부터는 첫 번째 라운드에서 설정된 통신 경로를 이용하여 싱크 노드로의 데이터 전송만을 수행한다. 트래픽 오버헤드는 센서 노드들 간에 그리고 센서 노드와 싱크 노드가 주고받는 모든 패킷의 수로 나타내었다.

(그림 9)의 결과를 살펴보면 첫 번째 라운드에서는 CPMSN과 MCF가 Flooding 보다 더 높은 트래픽 오버헤드를 나타내지만, 두 번째 라운드부터는 Flooding이 매 라운드마다 통신 경로 설정을 수행하기 때문에 트래픽 오버헤드가 가장 높음을 알 수 있다. CPMSN의 경우 이동 센서는 통신 경로 설정에 참여하지 않으므로 MCF 보다 초기 통신 경로 설정에 따른 트래픽 오버헤드가 더 낮음을 볼 수 있다. 또한, 제안한 CPMSN은 매 라운드에서, 첫 번째 라운드에서 설정된 최적의 통신 경로를 이용하므로 데이터 전송에 필요한 트래픽만을 사용한다. 이에 반해, MCF는 데이터 전송 시, 데이터를 가지고 있는 센서 노드와 싱크 노드 간의 거리에 비례하여 트래픽을 사용하므로 매 라운드마다 트래픽 오버헤드가 달라짐을 볼 수 있다. 결과적으로 CPMSN은 데이터를 가지고 있는 센서 노드와 싱크 노드 간의 거리에 상관없이 낮은 트래픽 오버헤드를 보여주는 반면, MCF는 데이터를 가지고 있는 센서 노드와 싱크 노드간의 거리가 멀수록 더 높은 트래픽 오버헤드를 일으킴을 알 수 있다.



(그림 9) 매 라운드 당 트래픽 오버헤드



(그림 10) 누적 트래픽 오버헤드

(그림 10)은 라운드가 진행됨에 따른 누적 트래픽 오버헤드를 보여주는 그림이다. 앞에서 언급했듯이 Flooding의 경우 매 라운드마다 경로 설정을 수행하므로 라운드가 진행됨에 따라 트래픽 오버헤드가 급격하게 증가한다. 초기에 CPMSN은 MCF와 비교하여 비슷한 수준의 트래픽 오버헤드를 나타내지만 라운드가 진행 될수록 CPMSN은 MCF 보다 낮은 트래픽 오버헤드를 나타낸다. 따라서, 본 논문에서 제안한 CPMSN은 Flooding, MCF와 비교하여 트래픽 오버헤드가 높지 않으면서도 네트워크 커버리지 성능은 더 우수함을 알 수 있다.

4. 결 론

본 논문에서는 지속적인 이동성을 갖는 이동 센서로 구성된 이동 센서네트워크 환경에서 효율적으로 경로 설정 및 데이터 전송을 가능하게 하는 통신 프로토콜을 제안하였다. 제안하는 프로토콜에서는 이동 센서와 함께 고정 센서를 배치함으로써 고정 센서가 이동 센서를 대신하여 싱크 노드로 센싱 데이터를 전송하도록 한다. 이동 센서로부터 데이터를 전달받은 고정 센서는 고정 센서들 간에 설정된 경로를 이용

하여 싱크 노드로 전달함으로써 이동 센서의 지속적인 이동에 따른 경로 설정 및 데이터 전송 문제를 해결할 수 있다. 시뮬레이션을 이용한 성능 평가를 통해 제안한 CPMSN이 기존의 고정 센서네트워크에 비해 네트워크 커버리지와 트래픽 오버헤드 성능에서 우수함을 보여주었다.

향후 연구 과제로, 센서의 에너지 측면을 고려한 이동 센서의 효율적인 이동 방향에 관해 연구하겠다.

참 고 문 헌

[1] K. Dantu, M. Rahimi, H. Shah, S. Babel, A. Dhariwal and G. S. Sukhatme, "Robomote: Enabling Mobility in Sensor Networks," IPSN 2005, pp.404-409, April 2005.

[2] E. M. Petriu, T. E. Whalen, R. Abielmona and A. Stewart, "Robotic Sensor Agents: a New Generation of Intelligent Agents for Complex Environment Monitoring," IEEE Instrumentation & Measurement Magazine, Vol.7, No.3, pp.46-51, Sept., 2004.

[3] T. L. Chin, P. Ramanathan, K. K. Saluja and K. C. Wang, "Exposure for Collaborative Detection using Mobile Sensor Networks," IEEE Mobile Adhoc and Sensor Systems Conference 2005. pp.743-750, Nov., 2005.

[4] Y. Zou and K. Chakrabarty, "Sensor Deployment and Target Localization based on Virtual Forces," IEEE INFOCOM 2003, Vol.2, pp.1293-1303, March 2003.

[5] G. Wang, G. Cao and T. L. Porta, "Movement-assisted Sensor Deployment," IEEE INFOCOM 2004, Vol.4, pp.2469-2479, March 2004.

[6] G. Wang, G. Cao, T. L. Porta and W. Zhang, "Sensor Relocation in Mobile Sensor Networks," IEEE INFOCOM 2005, Vol.4, pp.2302-2312, March 2005.

[7] B. Liu, P. Brass, O. Dousse, P. Nain and D. Towsley, "Mobility Improves Coverage of Sensor Networks," ACM MobiHoc 2005, pp.300-308, May 2005.

[8] K. Lin, J. Yu, J. Hsu, S. Zahedi, D. Lee, J. Friedman, A. Kansal, V. Raghunathan and M. Srivastava, "Heliomote: Enabling Long-Lived Sensor Networks Through Solar Energy Harvesting," ACM SenSys 2005, pp.309-309, Nov. 2005.

[9] M. Rahimi, H. Shah, G. S. Sukhatme, J. Heideman and D. Estrin, "Studying the Feasibility of Energy Harvesting in a Mobile Sensor Network," IEEE ICRA 2003, Vol. 1, pp. 19-24, Sept., 2003.

[10] Y. Ammar, A. Buhrig, M. Marzencki, B. Charlot, S. Basrour, K. Matou and M. Renaudin, "Wireless Sensor Network Node with Asynchronous Architecture and Vibration Harvesting Micro Power Generator," ACM Joint sOc-EUSAI 2005, pp. 287-292, Oct., 2005.

[11] F. Ye, A. Chen, S. Lu and L. Zhang, "A Scalable Solution to Minimum Cost Forwarding in Large Sensor Networks," IEEE ICCCN 2001, pp. 304-309, Oct., 2001.

김 형 진

e-mail : hyoung@emerald.yonsei.ac.kr

2004년 연세대학교 컴퓨터산업공학부 학사

2007년 연세대학교 컴퓨터과학과 석사

관심분야 : 무선 네트워크, 네트워크 보안

김 래 영

e-mail : leon@emerald.yonsei.ac.kr

1994년 이화여자대학교 전자계산학과 학사

1994년~2000년 삼성전자

2002년 연세대학교 컴퓨터과학과 석사

2006년 연세대학교 컴퓨터과학과 박사

현재 University of Illinois at Urbana-Champaign Post-Doc.

관심분야 : 유·무선통신, 정보보호

송 주 석

e-mail : jssong@emerald.yonsei.ac.kr

1976년 서울대학교 전기공학과 학사

1979년 한국과학기술원 전기전자공학석사

1988년 Univ. of California at Berkeley, 컴퓨터과학 박사

1988년~1989년 Assistant Professor in Naval Postgraduate School

1989년~현재 연세대학교 컴퓨터과학과 정교수

2006년 한국정보보호학회 회장 역임

관심분야 : 유·무선통신, 정보보호