

이동 agent가 있는 센서 네트워크에서 새로운 grant 기반의 접속 프로토콜

*김휴대, **조동호

한국과학기술원 전자전산학과 통신정보시스템연구실

*papin@comis.kaist.ac.kr, **dhcho@ee.kaist.ac.kr

New grant-based access protocol for Sensor Network with Mobile Agent

*Hyu-Dae Kim, **Dong-Ho Cho

CISLab, Dept. of EECS, KAIST

요약

센서 네트워크의 하나의 종류로 새롭게 제안된 SENMA(Sensor Network with Mobile Agent)에서 센서 노드는 일반적인 ad-hoc과 달리 오직 mobile agent와 통신이 가능하다. 센서 노드는 mobile agent가 근처에 접근하는 짧은 시간동안만 통신 가능 영역에 위치하기 때문에, SENMA의 접속 프로토콜은 매우 효율적으로 설계되어야 한다. 본 논문에서는 grant를 기반으로 하는 SENMA를 위한 효율적인 접속 방식을 제안한다. 수학적 분석과 시뮬레이션을 통해 제안한 방식이 전체 네트워크의 처리율을 향상시키고 에너지 소비를 줄임을 입증한다.

1. 서론

SENMA(Sensor Network with Mobile Agent)는 넓은 영역에 배치되는 센서 네트워크를 위한 구조의 하나로 제안되었다[1]. SENMA에는 두가지 종류의 노드가 존재한다. 먼저, 일반적인 센서 노드는 다른 센서 네트워크와 마찬가지로 주변 데이터의 수집을 담당하는 작은 장치이다. MA(Mobile Agent)는 센서 노드들이 수집한 정보를 취합하는 역할을 담당하며 일반적으로 작은 비행체나 자동차에 탑재되어 센서 노드 주위를 이동하며 센서 노드와 직접 통신을 수행한다.

SENMA에서는 일반적인 ad-hoc 방식과 달리 MA만이 센서 노드와 통신할 수 있다. 센서 노드는 MA가 보내는 beacon이 수신된 경우 MA에게 수집된 데이터의 전달을 시도하게 된다. 이런 단순한 통신 방식을 사용함으로써 불필요한 센서 노드간 통신을 하지 않고, 많은 기능을 MA에 구현하여 센서 노드의 전력소모를 줄이는 효과를 가져온다. 하지만, MA는 지속적으로 이동하기 때문에, MA가 센서 노드와 통신이 가능한 위치에 존재하는 시간은 그다지 길지 않다. 그러므로, 센서 노드의 수가 큰 경우, 동시에 다수의 노드가 접속 시도를 하게 되면 전체 시스템의 효율을 감소시키는 원인이 될 수 있다. 그러므로, 이런 상황을 충분히 고려한 효율적인 MAC(Media Access Control) 프로토콜이 적용되어야만 SENMA를 원활하게 운용할 수 있게 된다.

[3]에서는 SENMA에 클러스터의 개념을 도입하여 ad-hoc과 SENMA의 특징을 적절히 혼용하여 운용하는 방식에 대한 성능 분석을 수행하였다. [2]에서는 opportunistic ALOHA 방식을 이용한 random access 방식을 제안하였다. 이 방식은 채널의 상태 정보와 네트워크의 크기 정보를 MA가 충분히 알 수 있다고 가정하고, 시스템이 가질 수 있는 최대 throughput을 얻을 수 있도록 하였다.

하지만, MA는 계속해서 이동하고 수많은 센서 노드가 네트워크 상에 존재하기 때문에 데이터 전송을 시도하는 단말의 수나 MA와 각 노드간의 채널 정보 등을 정확히

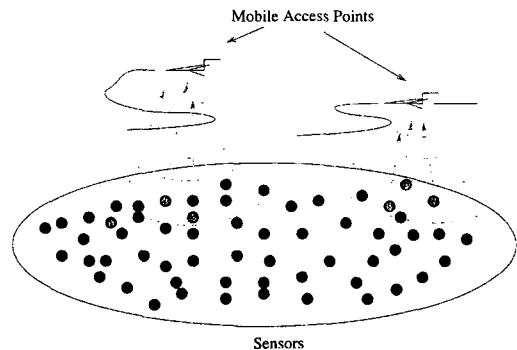


그림 1. SENMA 네트워크의 동작[2]

인식하기는 매우 어렵다. 이런 정보를 모르는 상태에서 센서 노드들이 데이터 전송을 시도한다면 충돌이 발생하여 처리율을 감소시키는 요인이 될 것이다. 그러므로, SENMA의 MAC 프로토콜은 간단하면서도 이런 상황에도 문제없이 잘 동작할 수 있도록 설계되어야 하며 처리율을 최대화할 수 있어야 한다.

본 논문에서는 grant를 기반으로 한 SENMA MAC 프로토콜을 제안한다. Grant란 센서 노드가 데이터를 전송할 수 있는 권한을 뜻하는 것으로, access request 전송을 통해 센서 노드가 먼저 요청을 하고, MA가 이 중에 몇개의 노드를 선택하여 grant를 전달해주는 것이다. 이 방식을 통해 동시에 전송되는 데이터 패킷의 양을 제어하여 불필요한 전송을 막고 처리율을 최대화할 수 있다. 또한, 전송 제어를 통