工學碩士 學位論文

무선 센서 네트워크에서 클러스터 기반 라우팅 프로토콜 연구

慶州大學校 一般大學院

컴퓨터電子工學科

文 俊 浩

2007年 6月

무선 센서 네트워크에서 클러스터 기반 라우팅 프로토콜 연구

指導敎授 趙 茂 昊

이 論文을 碩士學位 論文으로 提出함

2007年 6月

慶州大學校 一般大學院

컴퓨터電子工學科

文 俊 浩

文俊浩의 碩士學位論文을 認准함

審査委員長 _____ 印

審査委員 _____ 印

審査委員 _____ 印

慶州大學校 大學院

2007年 6月

목 차

Ⅰ. 서론 ····· 1

Ⅱ. 무선 센서 네트워크 계층 기존 연구 ····· 3

1. 개요 ····· 3
2. 네트워크 설계 ····· 3
   1) 네트워크 구조 ····· 3
   2) 라우팅 ····· 6
3. 라우팅 프로토콜 설계시 고려사항 ····· 9
   1) 센서 네트워크에서의 이동성 ····· 9
   2) 센서 노드 배치 ····· 10
   3) 전력 소모 ····· 10
   4) 데이터 전달 모델 ····· 10
   5) 센서 노드 성능 ····· 11
   6) 데이터 모음과 융합 ····· 11
4. 데이터 중심적(data centric) 프로토콜 ····· 12
   1) Flooding과 Gossiping ····· 12
   2) Sensor Protocol for Information via Negotiation (SPIN) 14
   3) 직접확산(Directed Diffusion) ····· 16
5. 계층적(Hierarchical) 프로토콜 ····· 18
   1) LEACH ····· 19
   2) TEEN ····· 20
   3) APTEEN (Adaptive TEEN) ····· 21

Ⅲ. 클러스터 트리 센스 네트워크 ····· 22
1. 개요 ····· 22
2. 클러스터 트리 네트워크의 형성 ····· 22
1) 클러스터 헤더 선택 과정 ····· 22
2) 단일 홉 클러스터 ····· 23
3) 멀티 홉 클러스터 ····· 24
3. 클러스터 트리 네트워크 관리 방안 ····· 26
1) 링크 상태 보고 ····· 26
2) 토플로지(Topology) 수정 ····· 28
3) 노드 고장으로부터 링크 회복 ····· 29
4. 수치해석 ····· 31

Ⅳ. 결론 ····· 35

참고문헌 ····· 36

초록 ····· 38

Ⅰ. 서 론

최근의 마이크로 전자 기기 시스템과 저전력 고집적 디지털 전자회로의 진보에 따라 마이크로 센서 개발이 급속도로 진전되었다. 이러한 센서들은 일반적으로 데이터 처리와 통신 기능을 장착하고 있다. 센서 회로는 센서 주변의 환경 변화를 감지하고 이러한 변화를 전기 신호로 변환한다. 센서는 수집된 데이터를 통상 무선 송신기를 통해 싱크(sink)로 직접 전송하거나 혹은 중간 수집기를 거쳐서 전송한다.

센서 노드들은 에너지 공급과 대역폭에 제한이 있다. 이러한 제한은 다량의 센서 노드들을 배치시키는 것과 더불어 센서 네트워크의 설계와 관리에 해결해야 할 문제점들이다. 이러한 문제 해결에는 네트워킹 프로토콜 스택의 모든 계층에서 에너지 절약에 대한 사항이 고려되어야 한다. 물리계층과 데이터 링크 계층과 관련된 에너지 문제들은 일반적으로 모든 종류의 센서 응용에서 공통적이다. 따라서 이 분야에 대한 연구는 시스템 차원의 전력 절약으로, 다이내믹 전압 조절, 무선 통신 하드웨어, 최소 의무 동작 시간(duty cycle) 문제, 시스템 분할(partitioning), MAC 프로토콜에서의 에너지 절약 고려 등이 있다. 네트워크 계층에서는 에너지 효율적인 경로를 설정하는 방법과 센서 노드에서 싱크로의 신뢰성 있게 데이터를 전달하는 방법을 찾아서 네트워크의 수명을 최대화 하는 것이 주된 목적이다. 센서 네트워크에서의 라우팅은 현재의 통신 방법이나, 무선 ad hoc 네트워크와 구별되는 몇 가지 특징으로 인해 도전해볼 만한 연구 분야이다. 첫 번째는 엄청난 수의 센서 노드들의 배치에 대한 전반적인 주소부여 방법이 가능하지 않다는 점이다. 따라서 고전의 IP 기반의 프로토콜을 센서 네트워크에 적용할 수가 없다, 두 번째는 일반적인 통신 네트워크와 반대로 거의 대부분의 센서 네트워크 응용은 센싱 된 데이터의 흐름이 다수 개의 영역의 소스 노드에서 특정 싱크 노드로 전송된다. 세 번째는 발생되는 데이터 트래픽이 아주 중복이 많이 발생하는데, 이는 어떤 현상의 주변에 있는 다수개의 노드들이 같은 데이터를 생성하기 때문이다. 이러한 중복은 에너지 절약과 대역폭 활

용도를 개선하기 위해 라우팅 프로토콜에 의해서 제거될 필요가 있다. 네 번째는 센서 노드들은 전송 전력이나, 동작 에너지, 처리 능력과 저장 용량 등에 아주 엄격하게 제한적이기 때문에 자원 관리를 조심스럽게 해야 한다.[1-2]

센서네트워크의 표준 규격에서는 개괄적인 라우팅 방안은 제시되어 있으나, 센서 노드와 싱크 노드 간에 멀티홉 무선 라우팅 프로토콜에 대한 구체적인 방안이 제시되어 있지 않다. 특히 홈 네트워크의 환경에서 지능형 서비스를 제공하기 위한 이동단말에 대한 빠른 정보전달이 무엇보다 중요하게 다루어져야 하므로, 홈 네트워크에 적합한 라우팅 프로토콜의 개발이 중요하다. 따라서 본 연구에서는 무선 홈 네트워크에서 구성되는 센서 네트워크의 다양한 접속 환경에서 최적화된 무선 라우팅 프로토콜에 대한 연구를 수행하고자 하며, 이의 최적화된 방안의 검증을 위해 프로토콜에 대한 성능을 분석하고자 한다.

Ⅱ. 무선 센서 네트워크 계층의 기존 연구

1. 개 요

무선 센서 네트워크의 네트워크 계층은 네트워크의 동작을 제어하고, 패킷을 라우팅 한다. 여하튼 네트워크 계층에서도 노드의 의무 동작 시간을 최소화시키기 위한 요구 사항은 네트워크 설계자에게 상당한 부담이 된다. 이러한 의무 동작 시간의 최소화 요구는 동기화(synchronization), 협상(negotiation), 협조(coordination)와 다른 네트워크 관련 기능에 제약 사항으로 작용한다.
멀티홉 무선 센서 네트워크는 인간의 관리가 필요 없는 자가구성(self organizing) ad hoc 네트워크이므로 네트워크 노드들 간의 물리적인 통신 링크는 어떠한 토폴로지 형태도 가질 수 있다. 따라서 네트워크의 물리적인 구조에 대해 언급하는 것은 무가치하며, 이 보다는 대규모화에 문제가 되는 논리적인 구조를 살펴보는 것이 더 중요하다.

2. 네트워크 설계

1) 네트워크 구조

논리적인 구조가 없는 완전한 평면(flat) 네트워크에서는 바로 인근의 네트워크 환경만 인식 가능하기 때문에 링크의 생성과 상실, 노드의 가입(association)과 해제(disassociation)등과 같은 네트워크를 조정하기 위해서는 통신 비용적인 측면에서 네트워크의 전 노드들이 협조해야 한다. 이러한 문제점의 예는 1973년 DRAPA PRENET 프로토콜에서 찾을 수 있다. DRAPA PRENET 프로토콜에서 각 네트워크 노드들은 매 7.5초마다 PROP (packet radio organization packet)에 자기 존재와, 자기 측면에서 인식한 네트워크 토폴로지 정보인 네이버 테이블(neighbor table)를 포함하여 송출한다. 인접 노드들로부터 수신한

PROP 정보는 라우팅 정보 같이 자기 자신의 PROP에 추가하여 전송한다 [3]. 모든 노드가 매 7.5초마다 네트워크 토폴로지 전체를 전송하게 되면 명백하게 대규모화는 어렵게 된다. 천 개 혹은 그 이상의 노드들이 존재하는 네트워크에서 이 방법으로의 구성은 전적으로 비실용적이다. 이러한 문제점은 LCA(Linked cluster architecture)로 해결할 수 있다.

LCA 네트워크는 노드 클러스터의 집합으로 구성되며, 각 노드는 최소한 어느 한 클러스터에 소속되어 있다. 모든 클러스터는 해당 클러스터 내에 있는 노드들을 지역적으로 제어하는 하나의 클러스터 헤더(cluster head, CH)를 가진다. 게이트웨이(gateway) 노드들은 클러스터간의 통신을 제공한다.

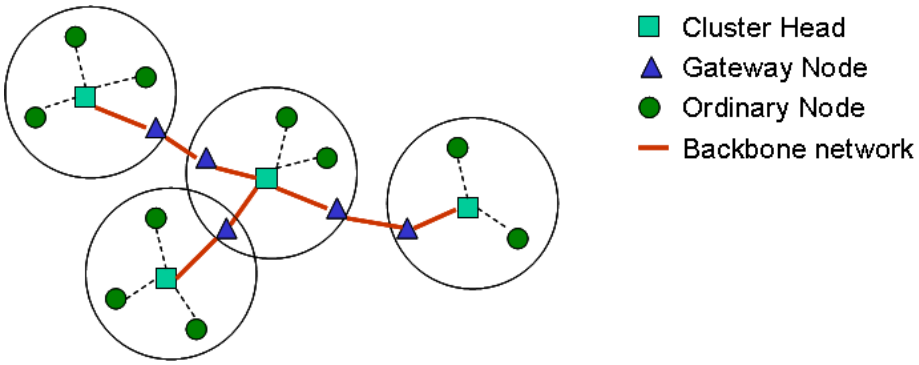


그림 1. LCA 네트워크 구조

TDMA를 사용하는 LCA에서 노드는 주기적으로 제어 채널 기간 동안에 분산 클러스터링 알고리즘을 사용하기 위해 통상적인 데이터 전송을 중지한다. 각 제어 채널 기간은 프레임1과 프레임2로 구성되고, 각 프레임은 다시 슬롯으로 나누어진다. 각 노드들은 프레임1과 2에서의 전송 슬롯을 배정 받아, 프레임1에서 각 노드들은 자기가 수신 가능한 노드 리스트를 배정된 슬롯에 브로드케스트하고, 나머지 슬롯에서는 다른 인접 노드들이 브로드케스트하는 것을 수신한다. 프레임2에서 각 노드들은 자신이 수신 가능한 전체 노드 목록을 가질 수 있고, 이 목록을 보고한다. 이후에 각 노드들은 클러스터 헤드, 일반 노드, 게이트웨이

선정 알고리즘을 수행한다.

각 노드들은 인접한 이웃 환경에 대한 정보만 저장하고 유지하면 되기 때문에, LCA 네트워크는 대규모화 할 수 있다. 그러나 LCA를 무선 센서 네트워크에 적용시에 다음과 같은 문제점들이 있다. 첫번째는 TDMA 시스템으로 global clock에 의존한다. GPS를 사용하는 초기 응용에서는 문제가 되지 않으나, 무선 센서 네트워크에 적용하면 가격과 GPS 수신을 위한 전력이 문제가 된다. 또한 실내인 경우는 GPS 수신이 불가하다. 더 근본적인 문제는 이웃하지 않은 클러스터의 검출이다. LCA에서 각 클러스터는 자신과 모든 인접 클러스터를 인식하는 일종의 클러스터 링크 리스트를 형성한다. 위치에 대한 추가적인 알고리즘이 없이는 바로 인접 외부에 있는 클러스터에 대해 정보를 전달할 방법이 없다. 주어진 클러스터에서 발생한 트래픽은 인접 클러스터 보다 더 멀리 전해질 수는 없다.

이 문제는 멀티 클러스터, 이동통신, 멀티미디어 무선 네트워크에서 Bellman Ford 알고리즘을 사용해서 해결하였다. 제어 채널 기간 동안 각 노드는 라우팅 리스트를 브로드캐스트 하는데, 라우팅 리스트에는 각 목적지 별 경로의 다음 노드와 홉수를 나타낸다. 그러나 인접 외부 클러스터의 라우팅 문제를 해결했지만, 제어 채널 기간에 다시 큰 테이블을 전송해야 하는 문제점으로 돌아가게 된다.

이웃하지 않은 클러스터 발견 문제는 네트워크 백본(backbone)에 클러스터헤드를 위치시키는 계층적 트리(hierarchical tree) 형태로 클러스터 위치시킴으로 해결된다. 계층적 라우팅은 라우팅 테이블을 효율화 시키며, 각 노드의 복잡성은 네트워크 크기에 무관하게 나타난다. 클러스터 트리 네트워크에서 네트워크 직경이 최소화되는 클러스터 형성의 최적화 문제는 계속적인 연구 분야이다. 클러스터 트리 네트워크에서 트리 루트를 향하는 클러스터 트리 라우팅을 사용한다 [4].

클러스터 트리를 무선 센서 네트워크에 적용시의 문제점은 클러스터 헤더 백본에 있는 노드들은 전력 소모가 높다. 이 노드들은 메시지 트래픽의 상당 양을 전달하기 때문에 의무 동작 시간이 커지게 되고, 다른 노드들 보다 짧은 배터리

수명을 가지게 된다. 이 문제는 LEACH(Low Energy Adaptive Clustering Hierarchy) 프로토콜으로 해결한다 [5-7]. LEACH는 네트워크 노드들에게 클러스터 헤드 기능을 랜덤하게 회전시켜, 에너지 소모를 균등하게 분배시킨다. 매 회마다 모든 네트워크 노드들이 선택한 랜덤 값에 따라 스스로 클러스터 헤드를 선택하고, 선택된 경우 CSMA MAC 프로토콜을 이용하여 다른 노드들에게 이 사실을 알린다. 각 노드들은 수신된 신호 세기를 근거로 클러스터 헤드에게 연결요구를 CSMA MAC 프로토콜을 사용하여 알린다.

LEACH는 클러스터 헤더 전력 소모 문제를 해결하였지만, 랜덤으로 클러스터를 발생시키는 과정에서 주어진 특정 노드의 범위에서 클러스터 헤더가 선발되지 않을 때 발생하는 네트워크 분할 가능성은 해결하지 못 하였다. 더 심각한 문제점은 라우팅 관련 문제인데, 대규모 네트워크에서 라우팅 문제점을 단순화시키기 위해 클러스터를 사용하는데, 클러스터 헤더가 수시로 변경되는 경우에 네트워크 노드들이 어떻게 라우팅 정보를 적절히 관리할 수 있을 지가 불분명해지는 문제점이 있다.

2) 라우팅

Ad hoc 무선 네트워크에서 라우팅 프로토콜에 대한 연구는 활발한 연구 분야이다. 이 연구들의 많은 분야가 노드들이 이동성을 가정으로 한 경우에 대한 라우팅 문제를 해결하기 위한 것이다. 노드 이동성은 라우팅 알고리즘을 매우 복잡하게 한다. 네트워크 설계자는 노드 이동성에 따라 경로 수정이 빈번해지는 문제와 배터리 수명을 최대화하기 위한 낮은 의무 동작 시간이 요구되는 상반되는 요구사항을 해결해야 하기 때문이다. 이에 대한 좋은 예는 IETF의 MANET 워킹 그룹의 연구로, 인터넷 기반의 이동 ad hoc 네트워킹에 대한 표준안을 작성하는 것이 목표이다. 이 범주의 프로토콜은 DSDV(Destination Sequenced Distance Vector), TORA(Temporally ordered Routing Algorithm), GSR(Global State Routing), AODV(Ad hoc On demand Distance Vector), FSR(Fisheye State Routing), HSR(Hierarchical State Routing), LANMAR(Landmark Ad hoc Routing) 등이다 [3].

무선 센서 네트워크의 노드들은 거의 정지 상태라고 가정하고, 라우팅 프로토콜에서 이동성 추적이나 이에 따른 링크 관리와 관련된 오버헤더도 줄일 수 있어 노드의 전력 소모도 줄일 수 있다. GRAd (Gradient routing) 프로토콜이 한 예이다. GRAd 프로토콜을 사용하는 노드는 라우팅 테이블을 사용하지 않고, 그 대신에 코스트(cost) 테이블을 관리한다. 코스트 테이블에는 노드가 각 목적지로 메시지를 전송하는데 요구되는 전체 코스트(홉수)를 나타낸다. 메시지 송출을 원하는 노드는 코스트 테이블에서 목적지에 해당하는 코스트를 포함시켜 단순히 메시지를 방송한다. 방송되는 메시지를 수신한 모든 노드들 중에 명시된 코스트 보다 낮은 코스트로 전달할 수 있는 노드만이 메시지를 전달 가능하고, 다른 모든 노드들은 메시지를 무시한다. 이 방법으로 메시지는 목적지를 향해 코스트 경사(cost gradient)를 미끄러져 간다.

이 방법의 장점은 GRAd에 사용되는 코스트 정의에 전력소모 인식 기능을 통합하면 아주 전력이 적게 소모되는 동작이 가능하다. 단점은 GRAd 발신 노드가 어떤 경로를 선택할 지를 사전에 알 수 없기 때문에 “on demand” 라우팅 방법으로 분류되지만 테이블 근거(table based) 방법이기도 하다. 따라서 네트워크의 모든 예상 목적지에 대한 코스트를 테이블에 저장하기 때문에, 대규모 네트워크에서는 심각한 저장 용량을 요구하게 된다. 또한 각 목적지에 대한 코스트를 발견해야 하는데, 초기에 플러딩(flooding)으로 수행된다. 센서 네트워크에서 싱크(sink)가 하나만 있으면 모든 노드들의 목적지가 하나가 되어 심각한 문제는 안 된다. 만약 목적지가 여러 개가 되면 코스트 발견 과정이 배터리 수명측면에서 비용이 커지게 된다.

Dominating set을 사용하는 라우팅 방안이 제안되었다. 만약 네트워크의 모든 노드들이 주된 몇 개의 노드나 그 이웃에 있는 경우에 이 주된 노드들을 dominating set이라 한다. Dominating set을 라우팅 목적으로 사용하는 방안은 라우팅 문제를 소규모의 주된 서브네트워크에서 발생하는 문제로 단순화 시키는 장점이 있다. 또한 오직 주된 서브네트워크에 포함되어 있는 노드들만이 라우팅 정보를 저장하면 된다.

네트워크 수명을 최대화시키기 위해서 트래픽 라우팅은 단지 절대적인 전력 소

모를 최소화시키는 라우팅 보다는 각 노드의 에너지 소모가 각자의 에너지 보존에 비례하여 균형 있게 소모될 수 있어야 한다. 이 원칙은 BEE(battery energy efficient) 프로토콜에 적용하여, 네트워크의 모든 노드들의 전력소모를 평균적이 되게 시도한다. BEE는 각각의 예상 경로에 대해 노드의 전력 소모와 배터리 동작을 고려한 코스트 함수(cost function)를 배정하고, 가장 낮은 코스트를 갖는 경로를 선택한다. 그러나 가능한 경로의 집합이 커지게 되면 이러한 경로를 저장하는 테이블을 가지는 것은 저비용 네트워크에서 심각한 지장이 되어 불가능하게 된다.

무선 센서 네트워크는 ad hoc 토폴로지와 자가 구성 특성이 있기 때문에 메시지 전달에 있어 생물학적인 모델이 도입되었다. 주목 받은 모델은 개미의 통신 방법에 근거한 것이 있다. 개미는 그들이 움직이는 지면에 페르몬(pheromone)을 흘림으로 통신한다. 예를 들어, 개미가 음식을 발견하면 다른 개미들이 음식을 찾아갈 수 있게 페르몬 흔적을 남기면서 둥지로 복귀한다. 페르몬은 확산되고, 증발하는데, 확산은 흔적을 넓게 하고, 증발은 가용한 시간 한계를 만든다. 개미는 확산되고, 다른 개미들의 페르몬에 의해서 확대된 흔적은 좁은 것보다 더 중요한 것으로 인식된다. 또한 증발된 흔적은 더 이상 사용되지 않는다. 개미는 중앙 집중화된 제어 없이, 즉 분산 제어 시스템(distributed control system) 모델인 개개의 비교적 간단한 행동에 근거하여 복잡한 조직화된 행동을 만들어 낸다. 개미 통신 방법은 자가 구성과 중앙 집중화된 제어가 필요 없다는 점에서 무선 센서 네트워크에 적합하다. 개미 둥지를 발신 노드로, 음식을 목적지 싱크 노드로, 개미는 메시지로 가정 한다. 또한 노드 간의 지면을 중간 노드들의 집합으로 가정 한다. 이 모델에서 목적지 싱크가 메시지(개미)를 받기 위해서는 먼저 존재를 광고하는 메시지 전송한다. 즉, 개미는 음식을 발견하고 둥지로 돌아간다. 광고 메시지는 노드 노드들 간을 무작위로 유니케스트 방법으로 전달된다. 광고 메시지를 받은 각 노드들은 발신지와 메시지를 받은 노드를 페르몬처럼 타임 스템프(time stamp)와 함께 테이블에 저장한다. 메시지의 “도달에 요구되는 홉수”를 증가 시킨 후 무작위 인접 노드에게 전달한다. 각 노드들은 타이 스템프 나이를 증가시켜 오래된 것은 삭제한다. (증발됨) 만약 발

신 노드가 메시지를 전송하면, 무작위로 이웃에게 메시지 전송한다. 수신한 페르몬 테이블을 점검하여 목적지 싱크로부터 광고 메시지를 받은 것이 있는지 확인한다. 만약 받지 않았으면 무작위 이웃에게 메시지 전송한다. 만약 받았으면, 광고메시지를 받은 노드로 메시지를 전송 한다. 따라서 메시지는 페르몬 흔적 상에 있게 되며, 메시지는 목적지 싱크 노드로 전달된다. 이 모델은 페르몬 흔적을 확장시키는 확산(diffusion) 효과와 한정된 네트워크 크기와 같은 요소를 포함하면 더 보강된다. 확산은 각 노드들은 자신의 페르몬 테이블 내용을 이웃에게 방송함으로서 된다. 개미 모델에서 가장 큰 제약 점은 음식(food)과 둥지(nest)들이 소수만이 존재하고, 대부분은 포함하지 않아야 한다. 통신 응용에서도 소수의 발신지와 싱크가 있을 때 잘 작동한다. 만약 모든 노드들이 발신지와 싱크이면, 갑자기 많은 광고메시지가 송출되게 되고, 아주 많은 양의 에너지가 소모하는 조건이 된다. 모든 개미가 모든 지역에서 음식을 발견하게 되면 개미 통신 시스템이 얼마나 열악하게 동작할지는 예상 가능하다.

3. 라우팅 프로토콜 설계시 고려사항

센서 네트워크의 응용에 따라 서로 다른 구조와 설계 목표와 제약이 고려되어 왔다. 라우팅 프로토콜의 성능은 구조적인 모델과 아주 밀접하게 연관되어 있지 때문에 다음에 구조적인 문제점과 설계에서의 주안점에 대해 살펴본다 [8].

1) 센서 네트워크에서의 이동성

센서 네트워크에서 라우팅의 세 가지 주요 구성요소는 센서 노드, 싱크와 검출된 이벤트(event)이다. 통상적으로 센서 노드들은 정지 상태로 가정된다. 싱크와 클러스터는 필요에 따라 이동할 수도 있다.

검출된 이벤트는 동적 혹은 정적이 될 수 있다. 표적 검출이나 추적 응용에서는 이벤트는 동적이며, 산불 방지를 위한 산림 감시는 정적인 이벤트이다. 대부분의 응용에서 동적인 이벤트는 주기적인 보고와 이의 결과로 싱크로 라우팅 되

는 중요한 트래픽을 발생시킨다. 정적 이벤트의 감시는 네트워크를 트래픽이 발생 했을 때에만 보고하는 반응적(reactive) 모드로 동작한다.

2) 센서 노드 배치

센서 노드의 배치는 사전 구성적(deterministic)이거나 자가구성적(self organizing)이다. 사전 구성적 방법에서는 센서들이 수동적으로 배치되며, 데이터는 사전에 결정된 경로로 따라 라우팅 된다. 자가 구성적 방법에서는 센서 노드들이 무작위로 산개되어 ad hoc 형태로 네트워크를 구성하게 된다. 이렇게 배치된 네트워크에서 싱크나 클러스터 헤더의 위치는 에너지 절약 측면에서 아주 중요하게 된다. 센서 노드의 분포가 균등하게 되지 않으면 에너지 효율적인 네트워크 동작에서 최적 클러스터 형성 문제는 중요하게 된다.

3) 전력 소모

네트워크 형성 과정에서 경로의 설정 과정은 에너지 절약에 아주 많이 영향을 받는다. 왜냐하면 무선의 전송 전력은 거리에 비례하게 되고, 장애물이 있으면 더 많은 전력이 요구된다. 이에 비해 멀티홉 라우팅은 직접 전송에 비해 적은 에너지가 소모 된다. 그러나 멀티홉 라우팅은 토플로지 관리나 매체 액세스 제어에 심각한 오버헤더를 초래한다. 직접 라우팅은 모든 센서 노드들이 싱크 가까이 있으면 아주 잘 동작할 것이다. 대부분의 센서들이 관심 대상 영역에 무작위로 흩어져 배치되므로 멀티홉 라우팅은 불가피 하게 된다.

4) 데이터 전달 모델

센서 네트워크의 응용에 따라 싱크로의 데이터 전달 모델은 연속적(continuous), 이벤트 발생적(event driven), 질의 중심적(query driven) 복합적(hybrid) 나누어진다. 연속적 전달 모델은 센서 노드가 주기적으로 데이터를 전송한다. 이벤트 발생적과 질이 중심적 모델은 데이터 전송이 이벤트가 발생하거나 싱크에 의해서 질의가 발생되었을 때 시작된다. 어떤 네트워크에서는 이러한 방법들이 복합적으로 사용된다.

라우팅 프로토콜은 에너지 소모 최소화와 경로의 안정성에 있어 데이터 전달 모델에 크게 영향을 받는다. 생태 서식 감시를 위한 응용에서는 감시 데이터가 싱크로 계속해서 전송되어야 한다. 이러한 경우에 최적인 라우팅 방법은 계층적 라우팅 방법이다. 왜냐하면 이러한 응용에서 중복해서 발생되어 싱크로 전송되는 데이터는 통합될 수 있기 때문에 메시지 수를 줄일 수 있고 에너지를 줄일 수 있다.

5) 센서 노드 성능

센서 네트워크에서 센서 노드들의 성능은 서로 다를 수 있다. 초기 모델에서는 모든 센서 노드들이 계산 통신 전력에서 동일한 특성을 가지는 것으로 가정되었다. 그러나 노드는 전송, 검출, 통합 등으로 특정 기능용으로 전용될 수 있다. 이는 한 노드가 세 가지 기능 모두를 동시에 수행하게 되면 에너지를 급속히 소모하기 때문이다.

특정 계층적인 프로토콜에서는 클러스터 헤더는 일반 노드들과는 다른 기능을 가질 수도 있다. 클러스터 헤더는 일반 노드에 비해 전력, 통신 대역, 메모리 등에서 더 우수한 기능을 가질 수 있다. 이러한 경우에 싱크로의 전송과 데이터의 통합에 대한 부담은 클러스터 헤더에 부담 지울 수 있다.

6) 데이터 모음과 융합

센서 노드들은 중복된 많은 데이터를 생성할 수 있다. 다수 개의 노드들에서 발생되는 이러한 유사한 데이터는 모아 질 수 있고, 전송 회수는 감소될 수 있다. 데이터 모음(aggregation)은 중복을 제거하는 suppression, min, max, average 등과 같은 함수를 사용해서 서로 다른 노드들로부터 발생되는 데이터를 조합할 수 있다. 통신이 연산에 비해 적은 에너지를 소모하기 때문에, 데이터 모음을 통해 근본적인 에너지 절약을 얻을 수 있다. 어떤 경우에는 노드들이 노이즈를 제거하고 신호를 정밀하게 강화 시키는 기술을 통해 신호를 더 정확하게 만드는 것을 데이터 융합(fusion)이라 한다.

4. 데이터 중심적(data centric) 프로토콜

센서 네트워크의 많은 응용에서 배치되는 노드의 수가 엄청나기 때문에 각 노드별로 고유한 식별 주소를 배정하는 것은 불가능해 보인다. 이러한 고유 식별 주소 배정의 문제점과 센서 노드의 무작위 배포로 인해 특정 센서 노드 집단을 선택하여 질의 하는 것은 어렵게 된다. 따라서 데이터는 배치된 지역 내에 있는 모든 노드들에 상당량이 중복된 채로 전송되게 된다. 이것은 전력 소모 절약 측면에서는 아주 비효율적이기 때문에 라우팅 프로토콜이 특정 센서 노드들을 선택할 수 있고, 데이터를 중계하는 과정에서 데이터 모음을 수행할 수 있어야 한다. 이러한 요구 사항은 기존에 네트워크 계층에서 주소 가능한 노드들 간에 경로를 형성하는 주소 기반(address based)의 라우팅과는 다른 데이터 중심적(data centric) 라우팅을 개발했다.

데이터 중심적 라우팅에서는 싱크(sink)는 특정 지역으로 질의(query)를 보내고, 선택된 영역에 위치한 센서들로부터의 데이터를 기다린다. 데이터는 질의를 통해서 요청되기 때문에, 속성기반(attribute based) 네이밍이 데이터 특성을 규정하기 위해 요구된다. SPIN (Sensor Protocol for information via negotiation) 프로토콜은 최초의 데이터 중심적 프로토콜이며, 노드들 간에 협상을 통해 중복 데이터를 제거하기 하여, 에너지를 절약하는 프로토콜이다. 이후에 직접확산(Directed Diffusion)이 개발 되었으며, 데이터 중심적 라우팅의 획기적인 발전이 된다 [8-9].

1) Flooding과 Gossiping

Flooding과 gossiping은 센서 네트워크에서 어떠한 라우팅 알고리즘이나 토플로지 관리도 요구되지 않으면서 데이터를 중계하는 두 가지의 고전적인 방법이다 [10].

Flooding에서는 데이터나 관리 패킷을 받은 각 노드는 이웃의 모든 노드들에게 브로드캐스팅 한다. 이 과정은 패킷이 목적지에 도달하거나, 패킷이 도달할 수 있는 최대 홉수에 도달할 때까지 계속 된다.

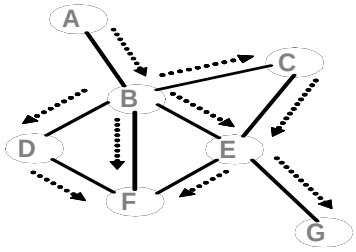


그림 2. Flooding

비록 Flooding은 구현하는데 아주 용이하지만, Implosion, Overlap, Resource Blindness 등과 같은 단점이 있다. Implosion은 중복된 메시지가 동일 노드에 전달되는 경우이다. 그림 3에서 노드 A에서 Flooding으로 데이터를 이웃들에게 전송하면, 노드 D는 궁극적으로 두개의 중복된 복사본을 불필요하게 받게 된다. Overlap은 두개의 노드가 동일 지역을 중복 감시하는 경우에, 하나의 사건을 동시에 검출하여 전송하게 되므로 인접 노드들은 중복 메시지를 받게 되는 것이다.

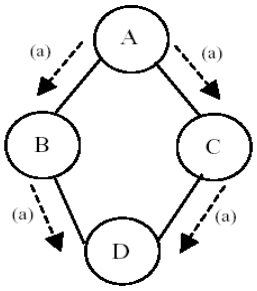


그림 3. Implosion

그림 4에서 노드 A와 B는 지역적으로 중복되게 담당하고 있는 영역에서 이벤트가 발생하게 되면, 노드 C는 한 사건에 대해 두개의 메시지를 받게 된다.
Resource blindness는 에너지 제약을 고려하지 않고 전송하여 에너지를 많이 소모하는 단점을 말한다.

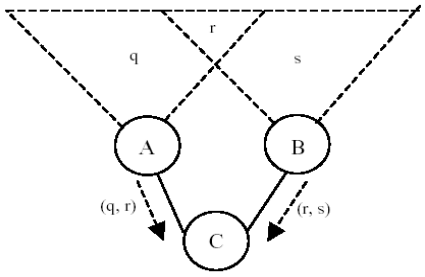


그림 4. Overlap

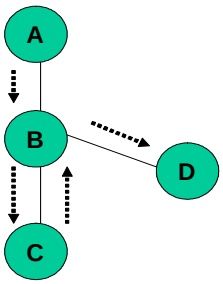


그림 5. Gossiping

Gossiping은 flooding을 개선한 버전으로 각 노드는 수신한 데이터를 브로드캐스트하는 대신에 무작위로 선정된 하나의 이웃 노드로 전송하고, 그 이웃 노드는 다시 무작위로 선정된 이웃에 전송하는 방안으로 계속 된다. 장점은 어떠한 노드에서도 하나의 메시지만 존재하므로 Implosion이 방지되지만, 단점은 데이터 전달에 지연이 많이 발생하게 된다는 점이다.

2) Sensor Protocol for Information via Negotiation (SPIN)

SPIN은 초기 데이터 중심적 라우팅 알고리즘을 사용하는 방법 중의 하나이다 [11]. SPIN의 배경적 개념은 상위 레벨 서술자를 사용하여 데이터를 명명하는 meta data를 사용하는 것이다. 전송하기 전에 데이터 광고 방법을 사용하여 센서들 간에 meta data를 교환하는 것이 핵심 기능이다. 각 노드는 새로운 데이터를 수신하면, 이웃에게 이것을 광고한다. 만약 이웃이 이에 대해 관심이 있으면, 즉 그런 데이터를 가지고 있지 않은 노드는 요구 메시지를 보내어 데이터

를 수신한다. SPIN의 meta data 협상 방법은 중복 데이터를 보내는 Flooding이나, 센싱 영역의 overlap, resource blindness와 같은 고전적인 문제들을 해결하여, 에너지 절약에 효율적이다. meta data 포맷에는 표준이 없고, 응용에 의해 결정되므로 응용에 의존적이 된다. SPIN에는 데이터를 교환하기 위해 3가지 메시지가 정의되어 있다. ADV는 센서가 특정 meta data를 광고하기 위한 것이고, REQ는 특정 데이터를 요청하기 위한 것이며, DATA는 실제 데이터를 운반하기 위한 것이다.

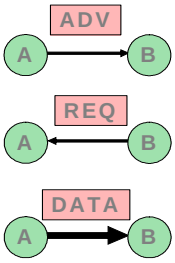


그림 6. ADV, REQ, DATA 메시지

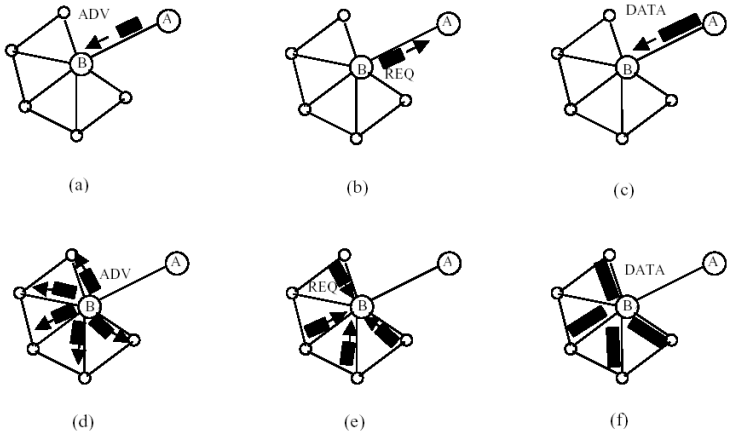


그림 7. SPIN 프로토콜

그림 7에서 (a)는 노드 A에서 노드 B로 검출한 데이터를 광고(ADV) 한다. (b)에서 노드 B가 노드 A로 데이터를 요청(REQ)을 보냄으로 응답한다. (c)에서

노드 B가 노드 A로부터 요구한 데이터(DATA)를 받는다. (d)에서 노드 B가 이웃들에게 광고(ADV)를 보낸다. (e)에서 각 이웃들이 요청(REQ)을 보냄으로 응답한다. (f)에서 노드 B가 이웃들에게 데이터(DATA)를 보낸다.

SPIN의 장점은 각 노드는 단일 홉의 이웃들만 알면 되기 때문에 토플로지 변경이 지역에 국한된다. SPIN은 에너지 소모관점에서 Flooding에 3.5배 감소되고, 중복 데이터를 반으로 감소시킨다. 그러나 만약 발신 노드와 목적지 노드 사이의 노드들이 협상 데이터에 관심이 없으면 중계하지 않으므로 SPIN은 데이터 전달을 보장하지 못하게 된다. 따라서 SPIN은 일정 간격으로 신뢰성 있는 데이터 패킷의 전달이 요구되는 침입 감시와 같은 응용에는 적합하지 않다.

SPIN 계열의 프로토콜들은 다음과 같다. Point to Point 네트워크에서 SPIN PP는 point to point 통신 프로토콜이고, SPIN EC는 SPIN PP 방법에서 만약 low energy threshold에 도달하면 해당 노드는 라우팅 프로토콜 참여를 줄이는 방법이다. SPIN BC는 브로드케스터 네트워크를 위한 방안이고, ADV를 받은 후에 REQ를 보내기 전에 랜덤 시간을 설정하여 기다리므로 다른 노드가 REQ를 송출하는 것을 기다리는 방식이다. SPIN RL은 SPIN BC에 각 노드들은 특정 시간 동안 요청한 데이터가 수신되는 가를 추적하여 수신이 안 된 경우는 다시 요청하는 방식이다.

3) 직접확산(Directed Diffusion)

집적확산은 질의의 수가 이벤트의 수 보다 많다고 예상되는 경우를 위해 개발한 방식이다[12-13]. 즉, 요구기반(On demand basis)의 데이터 중심적이다. 집적확산은 센서 노드들을 통해 데이터의 네이밍(naming) 방법을 사용하여 데이터를 확산 시키는 것이 주된 개념이다. 이러한 네이밍 방법을 사용하는 주된 이유는 에너지를 절약하기 위해 네트워크 계층의 라우팅의 불필요한 동작을 제거하기 위한 것이다. 집적확산은 데이터에 대해 속성 값(attribute value) 쌍을 사용하고, 요구기반으로 이 쌍을 사용하여 질의한다. 질의를 만들기 위해 개체(object), 간격(interval), 기간(duration), 지리적인 지역(geographic area) 등의 이름을 사용하는 속성 값 쌍의 목록으로 인터레스트(interest)를 정의한다. 인터

레스트는 싱크에 의해서 이웃을 통해 방송된다. 인터레스트를 받은 각 노드는 나중에 사용하기 위해 캐싱(caching)하여 둔다. 캐싱된 인터레스트는 수신된 데이터의 인터레스트 값과 비교된다. 인터레스트 항목에는 몇 개의 gradient 항목이 있다. Gradient는 인터레스트를 수신한 이웃을 통해 응답하기 위한 경로를 나타낸다. Gradient는 수신된 인터레스트 항목에서 추출된 데이터율, 기간, 종료시간 등으로 특징된다. 인터레스트와 gradient를 활용하면, 싱크와 발신 노드 간의 경로가 설정될 수 있다. 몇 개의 가능한 경로 중에서 강화(reinforcement)에 의해서 하나의 경로가 선택된다. 싱크는 선택된 경로를 통해 원래의 인터레스트 메시지를 다시 전송하는데, 이때는 간격이 작아져서 그 경로를 통해서 발신 노드까지 데이터를 더 자주 보낼 수 있게 강화 된다.

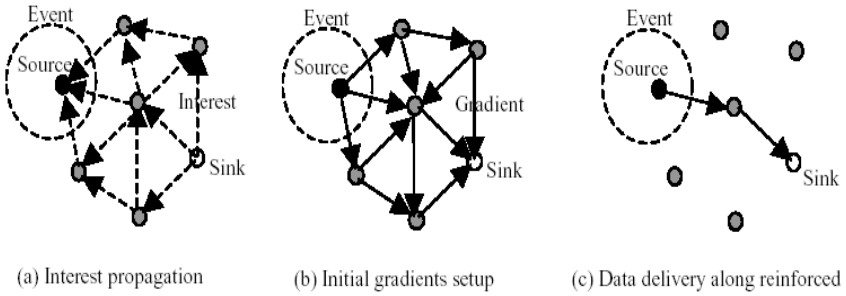


그림 8. 집적확산 프로토콜 과정

경로 복구도 집적확산에서 가능하다. 발신 노드와 싱크 사이의 경로가 단절이 되면 새로운 경로나 우회 경로가 찾아져야 한다. 이를 위해 집적확산에서는 다른 경로들 중에서 강화를 재실시 한다.

집적확산은 요구기반(On demand) 데이터 질의 방법이란 관점에서 SPIN과 차이가 있다. 집적확산에서는 싱크가 특정 데이터가 가용한 가를 flooding에 의해서 센서 노드들에게 질의한다. SPIN에서는 센서가 관심 있는 노드들에게 광고를 통해 질의한다.

집적확산 장점은 neighbor to neighbor로 통신이 이루어지는 데이터 중심적(data centric) 방식이기 때문에 노드를 위한 식별 주소부여 방법이 필요 없다는 점이다. 각 노드들은 데이터 센싱 외에도 모음(aggregation)과 캐싱(caching)을

할 수 있다. 캐싱 기능은 에너지 절약과 지연 측면에서 대단한 장점을 가진다. 추가적으로 요구기반 이므로 에너지 절약 측면에서 상당히 효율적이며, 네트워크 토플로지에 대한 관리가 요구되지 않는다.

집적확산 단점은 모든 센서 네트워크 응용에 적용할 수는 없다는 것이다. 싱크로 연속적으로 데이터 전달이 요구되는 응용에서는 질의에 의해 요구하는 방식은 적합하지 않다. 즉, 환경 감시와 같은 응용을 위한 라우팅 프로토콜로는 적합하지 않다. 네이밍 방법이 응용에 의존적인 것과 데이터와 질의의 일치 여부를 확인 하는 과정은 센서 노드에서 추가적인 오버헤더로 작용하는 단점이 있다.

5. 계층적(Hierarchical) 프로토콜

센서 네트워크에서도 다른 통신 네트워크와 마찬가지로 네트워크 규모의 대형화가 가능한 가하는 속성은 매우 중요하다. 단일 계층(tier)의 네트워크는 센서 노드들이 네트워크에 추가되어 증가하게 되면 게이트웨이 노드의 부하가 가중되게 된다. 이러한 과부하는 통신 지연을 초래하게 되고, 사건 발생을 추적하는데 지장을 초래하게 된다. 또한 통상적으로 센서 노드의 한정적인 전력으로 인해 통신 능력이 장거리 통신에 적합하지 않기 때문에 단일 계층으로 대상 지역이 넓고 많은 노드들을 서비스 하는 규모로 커지게 하는 것은 불가능하다. 이러한 추가되는 노드들에 의한 부하를 감당하고, 광범위한 지역의 서비스를 가능하게 하기 위해서 네트워크를 클러스터링 하는 방안에 대해 연구되기 시작 했다. 클러스터 네트워크의 주된 목적은 클러스터 내에서 멀티홉 통신과 클러스터 헤더에서의 데이터 모음과 통합을 통하여 싱크로 가는 메시지 수를 줄임으로 센서 노드의 에너지 소모를 효율적으로 유지 감소시키는 것이다.

클러스터는 센서들의 에너지 보존과 클러스터 헤더에 근접해서 센서 노드들이 존재한다는 근거로 형성된다. LEACH(Low Energy Adaptive Clustering Hierarchy)는 센서 네트워크에서 계층적 라우팅을 사용하는 첫번째 프로토콜중의 하나이다 [5].

1) LEACH

LEACH (Low Energy Adaptive Clustering Hierarchy)는 싱크(또는 베이스 스테이션)는 고정되어 있고, 센서 노드로부터 멀리 떨어져 있으며, 모든 센서 노드들은 동질성이며, 에너지 제한적이라고 가정한다. 센서 노드들과 싱크와의 통신은 데이터 처리보다 에너지가 많이 소모된다. 클러스터를 사용하므로 해서 소수의 노드들 즉, 클러스터 헤더(Cluster head, CH)들 만이 싱크로 장거리 통신을 수행하고, 반면에 클러스터의 다른 노드들은 근거리 전송이 가능하게 한다. LEACH는 고전적인 클러스터링 알고리즘에 비해 적응적 클러스터링과 클러스터 헤더의 순환을 통해 우수한 성능을 가진다. 클러스터 헤더는 각 노드의 에너지 소모를 균등하게 유지하기 위해 특정 시간마다 무작위로 순환된다. 각 클러스터에서 데이터에 대한 연산을 수행하므로, 싱크로 전송되는 데이터양을 줄일 수 있다. 이는 통신이 연산보다 에너지를 많이 소모하므로 에너지 소모의 절감을 기본적으로 많이 얻을 수 있다. 최적의 클러스터 헤더 숫자는 전체 노드 수의 5% 정도로 예측된다. 또한 각 노드들은 클러스터 헤더가 될 동일한 가능성을 가지고 있다. 그리고 클러스터 헤더 간의 통신은 허용하지 않는다.

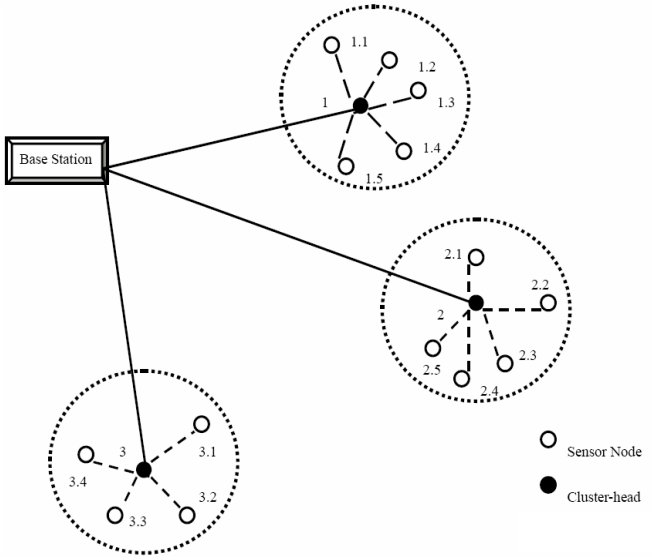


그림 9. LEACH의 계층적 클러스터링

LEACH의 장점은 싱크(베이스 스테이션)로 전송되는 정보의 양을 줄이기 위해 라우팅 프로토콜에 데이터 모음(aggregation)이나 융합(fusion) 기능을 사용하는 것이다. 또한 클러스터 헤더를 무작위로 순환시킴으로 센서 노드들의 에너지 소모를 분산시켜 네트워크의 수명을 연장시키는 것이다. LEACH의 단점은 클러스터 헤더가 직접 전송할 수 있는 거리의 단일 홉 라우팅을 사용하기 때문에 대규모의 영역에 배치되는 네트워크에는 사용할 수 없다. 클러스터링을 역동적으로 변경하는 개념은 클러스터 헤더의 변경, 관리 등과 같은 추가적인 오버헤더가 있어, 에너지 소모에서 얻은 이득을 상쇄시킬 수 있다. 또한 싱크에서 멀리 떨어진 클러스터 헤더는 많은 에너지를 소모할 것이며, 근접해 있는 클러스터보다는 빨리 배터리가 소모되는 문제점이 있다.

2) TEEN

TEEN(Threshold sensitive Energy Efficient sensor Network protocol)은 센싱되는 속성이 온도와 같이 급속하게 변화에 반응하는 계층적 프로토콜로 설계되었다 [14]. 시간적인 측면의 반응이 중요한 경우에는 네트워크는 반응적(Reactive) 모드로 동작하게 된다. 데이터 중심적 방법을 사용하는 계층적 프로토콜이다. 네트워크 구조는 근접한 노드들이 클러스터를 형성하고, 이러한 과정이 싱크에 도달하기 까지 두 단계까지 진행된다.

클러스터가 형성되고 나면 클러스터 헤더는 노드들에게 두 개의 한계값(threshold)을 브로드캐스터 한다. Hard threshold는 센서노드의 송출기를 동작시켜 클러스터 헤더로 송출하기 위한 최소한의 값으로, 송출되는 메시지 수를 상당히 감소시킬 수 있다. Soft threshold는 한번 hard threshold에 도달한 노드가 다시 데이터를 송출하기 위해서는 soft threshold와 같거나 클 때에 송출할 수 있다. 따라서 soft threshold는 센싱 된 속성 값이 같거나 조금 변화가 있을 때는 메시지 수를 줄일 수 있다. TEEN의 장점은 Hard와 soft threshold를 조정해서 메시지 전송 수를 조정할 수 있다. TEEN의 단점은 만약 한계값에 도달하지 못하면 어떠한 데이터로 받아볼 수 없기 때문에 주기적인 보고가 요구되는 응용에는 적합하지 않다.

3) APTEEN (Adaptive TEEN)

APTEEN은 TEEN과 동일하게 동작하면서, 어떤 노드가 정해진 시간 동안 데이터를 송출하지 않으면 강제적으로 센싱하여 데이터를 송출하게 하는 방식이다 [15]. 네트워크 구조는 TEEN과 동일하다. 싱크(베이스 스테이션)이 클러스터를 형성할 때, 클러스터 헤더는 속성과, 한계값, 모든 노드들의 전송 스케줄, count time을 방송한다. Count time은 노드에 의해서 보내지는 연속적인 두 개의 보고간의 최대 시간을 나타낸다. APTEEN의 장점은 사전적(Proactive)인 모드와 반응적(reactive) 모드로 동작이 가능하다는 것이다. 에너지 소모는 count time과 한계값을 변경하여 조정할 수 있다. APTEEN의 주된 단점은 count time과 한계값 구현하는데 추가적인 복잡성이 요구된다. 또한 다단계의 클러스터 형성이 오버헤더를 요구하고 복잡해짐에 따라 에너지 소모도 증가하게 되는 점이다.

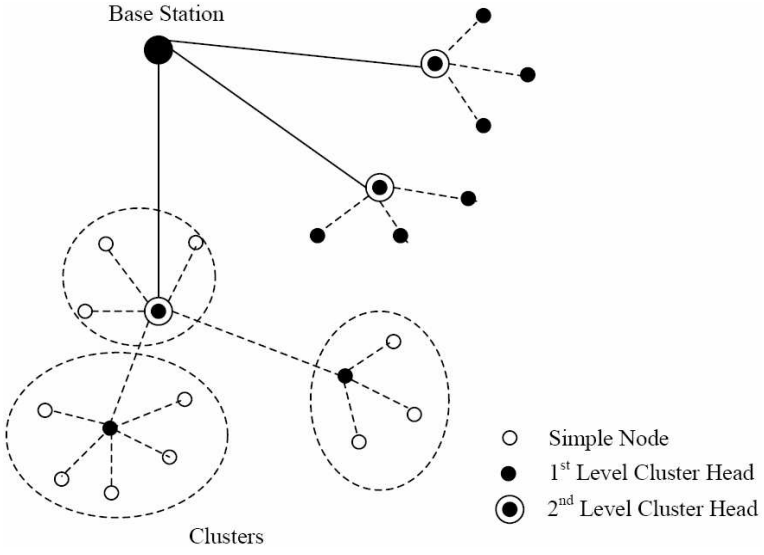


그림 10. TEEN 및 APTEEN의 계층적 클러스터링

Ⅲ. 클러스터 트리 센서 네트워크

1. 개 요

클러스터 기반의 계층적 트리 구조는 대형화가 용이하면서 라우팅 프로토콜이 단순하다는 점에서 장점을 가진다. 즉, 로컬 클러스터를 형성함으로써 인접한 지역에서 발생한 사건에 대한 유사한 정보를 클러스터 헤드로 전송하고 클러스터 헤드가 데이터 모음을 수행하여 보다 에너지 효율적인 라우팅을 가능하게 한다. 또한 요청된 질의에 대한 클러스터 헤드에 의한 전달로 비효율적인 플러딩 되는 질의들을 막을 수 있다.

센서 노드들은 클러스터 헤드를 선택하고 자가 구성 방법으로 클러스터를 형성한다. 클러스터 형성 과정에서 클러스터 헤더는 각 멤버 노드들에게 고유한 노드 식별 번호를 부여한다. 자가 구성된 클러스터들은 Designated Device(DD)에 의해서 서로 연결 된다. DD는 다른 노드들보다 우수한 계산 능력과 대용량 메모리를 보유한 특별한 디바이스이다. 대부분의 응용에서 DD는 센서 네트워크와 인터넷 사이에서 게이트웨이 역할을 수행한다. DD는 각 클러스터에 고유한 클러스터 식별번호를 부여한다 [4].

2. 클러스터 트리 네트워크의 형성

1) 클러스터 헤더 선택 과정

클러스터 형성은 클러스터 헤드 선택으로부터 시작 된다. 어느 한 노드의 동작이 시작되면 다른 노드로부터 송출되는 HELLO 메시지를 기다린다. 만약 HELLO 메시지를 어느 기간 동안 기다린 후에도 들을 수 없으면 인접에 클러스터 헤더가 없는 것으로 간주하고 자신이 클러스터 헤드가 되어 HELLO 메시지를 송출 한다. 이 새로운 클러스터 헤더는 일정 시간 동안 이웃으로부터 응답을 기다린다. 만약 어떠한 연결 요구도 받을 수 없으면 다시 일반 노드로 돌아

가 HELLO를 수신하게 된다.

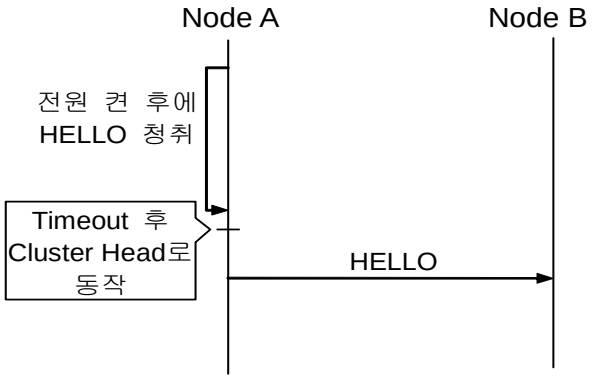


그림 11. 클러스터 헤더 시작

클러스터헤드를 선택할 때에 전송범위, 전원 능력, 계산 능력 또는 위치 정보 등과 같은 각 노드의 파라미터를 근거로 선택될 수 있다.

2) 단일 홉 클러스터

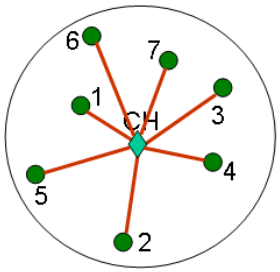


그림 12. 단일 홉 클러스터 구조

어느 한 노드가 클러스터 헤드로 선택되어 주기적으로 HELLO 메시지를 방송하게 된다. 여기에는 클러스터 헤드의 MAC 주소와 클러스터 헤드 식별 번호를 나타내는 노드 ID(NID) 0번이 포함되어 있다. 이 HELLO 메시지를 받은 노드는 클러스터 헤드로 연결 요구(CON REQ) 메시지를 송출한다. 이 메시지를 받

은 클러스터 헤드는 요구한 노드를 위한 NID 번호가 포함된 연결응답(CON RES) 메시지로 응답한다. 이 NID 는 클러스터 내에서 유일해야 하며, 클러스터 헤드가 유일한 NID를 할당하는 책임을 담당하게 된다. NID를 할당 받은 노드는 클러스터 헤드로 ACK 메시지로 응답하여 링크 연결을 완료한다. 이후에 각자의 네이버 리스트(neighbor list)에 상대방을 parent와 child로 설정하는데, 클러스터 헤드는 새로이 추가된 노드를 child로 설정하고, 새로이 추가된 노드는 클러스터 헤드를 parent로 설정한다.

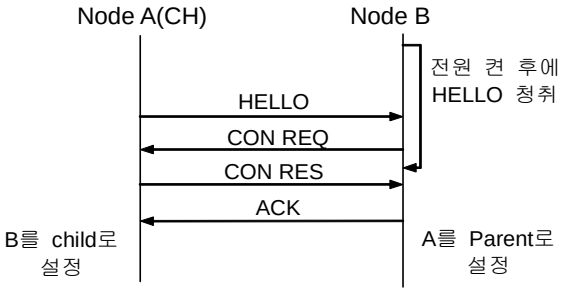


그림 13. 클러스터 헤더 선택 및 노드 추가

만약 모든 노드들이 클러스터 헤드의 서비스 영역인 1홉 내에 위치하게 되면 접속 형태(topology)는 성형(star)이 된다. 클러스터 내에 존재할 수 있는 노드 수는 방송과 임시 NID를 제외하면 최대 254개가 된다.

3) 멀티 홉 클러스터

클러스터는 각 노드들이 다중 연결을 지원하면 멀티 홉 구조로 확장할 수 있다. 비록 통신 지연은 증가하게 되지만 서비스 커버리지는 하나의 클러스터 서비스 영역에서 확장될 수 있다.

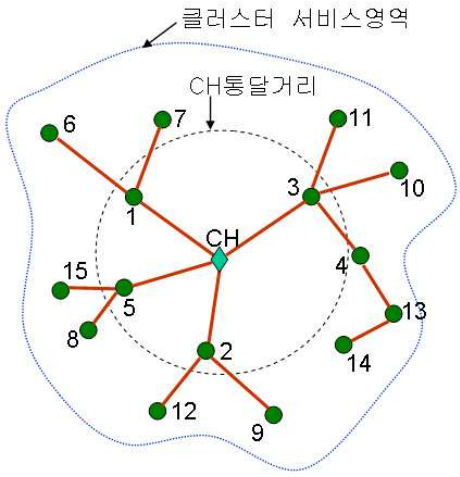


그림 14. 멀티 홉 클러스터 구조

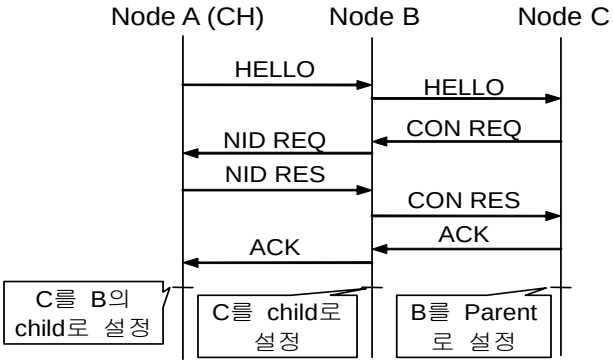


그림 15. 멀티 홉 클러스터 형성 절차

그림 15에서와 같이 노드 B는 클러스터 헤드와의 링크가 설정된 후에 클러스터 헤드로부터 전송되는 HELLO 메시지를 중계하게 된다. 노드 C가 노드 B로부터 HELLO를 받게 되면 연결 요구 메시지를 송출한다. 노드 B는 클러스터 헤드로 노드 C의 NID 요구(NID REQ)하고, 새로운 NID 할당(NID RES) 받아 노드 C로 응답한다. 링크가 연결된 후에는 노드 B를 노드 C의 parent로, 노드 B는 노드 C를 child로, 클러스터 헤드는 C를 B의 child로 설정한다. 이후에 노드 C는 이웃을 위하여 HELLO 메시지를 중계하게 된다. 만약 어느 노드가 여러 개의 HELLO 메시지를 받게 되면 가장 빨리 도착한 메시지에 응답한다. 이 경우에

클러스터 헤더로의 경로가 최적이 아닐 수 도 있지만, 추후에 최적화가 이루어진다.

만약 클러스터 헤더가 NID를 할당할 식별번호를 다 사용하거나 다른 제약에 의해서 새로운 노드의 연결 요구를 거절해야 하는 경우에는 NID를 254로 사용한다. NID 254를 받은 새로운 노드는 클러스터 식별번호 (CID)를 저장하고 한동안 이 CID에 속한 노드들로 연결 요구를 전송하지 않는다.

3. 클러스터 트리 네트워크 관리 방안

1) 링크 상태 보고

클러스터 헤더는 주기적으로 HELLO 메시지를 멤버 노드들에게 전송한다. 멤버 노드는 HELLO 메시지를 받게 되면 그들의 이웃을 위해 다시 방송하게 된다. 각 노드는 HELLO 메시지가 방송될 때 그들의 이웃 노드 정보를 인식하여 네이버 리스트에 기록한다. 특정 시간 동안 이웃 노드에 대한 HELLO 정보가 갱신되지 않으면 목록에서 제거된다. 만약 다른 클러스터에 속한 노드로부터 HELLO 메시지를 받으면 클러스터 번호를 네이버 리스트에 저장한다.

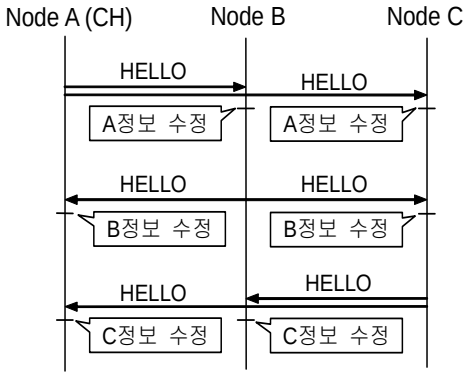
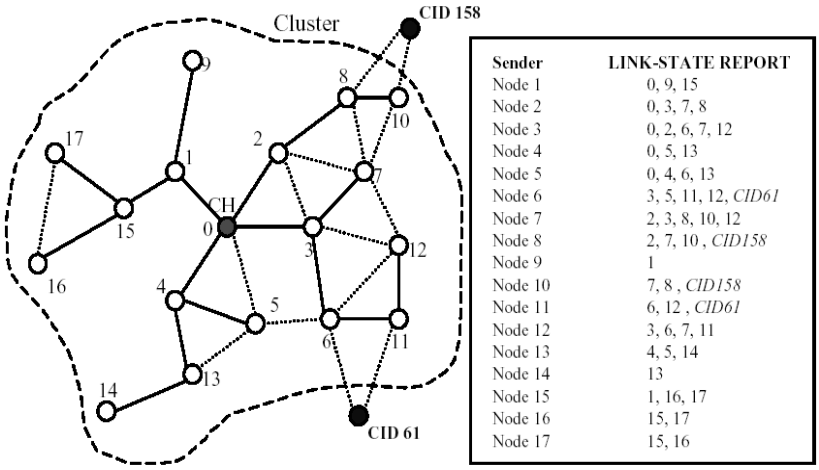


그림 16. HELLO 메시지에 의한 네이버 리스트 관리

모든 노드들은 클러스터 헤더로 링크상태보고(Link State Report)를 사용하여 네이버 리스트를 주기적으로 또는 요청에 의해서 보고한다.



Sender	LINK-STATE REPORT
Node 1	0, 9, 15
Node 2	0, 3, 7, 8
Node 3	0, 2, 6, 7, 12
Node 4	0, 5, 13
Node 5	0, 4, 6, 13
Node 6	3, 5, 11, 12, *CID61*
Node 7	2, 3, 8, 10, 12
Node 8	2, 7, 10 , *CID158*
Node 9	1
Node 10	7, 8 , *CID158*
Node 11	6, 12 , *CID61*
Node 12	3, 6, 7, 11
Node 13	4, 5, 14
Node 14	13
Node 15	1, 16, 17
Node 16	15, 17
Node 17	15, 16

그림 17. 링크 상태 보고

링크 상태 보고는 클러스터 트리의 최종 말단에서 시작되며, 클러스터 헤더로 가는 경로의 중간에 있는 노드들은 이 메시지를 중계하면서 직접 통신이 안 되는 트리 하부 노드들에 대한 정보를 얻게 된다. 또한 중간 노드들은 여러 child 들로부터 수신한 정보를 종합하여 링크상태보고 메시지를 parent로 송부한다. 이때 트리 하부의 통달 거리가 벗어나는 노드들은 링크상태보고를 보고한 child 노드를 표시하여 라우팅시 트리를 따라서 찾아갈 수 있게 단순화 시킨다. 이러한 과정은 링크 상태 보고가 최종 클러스터 헤더에 도달할 때까지 반복하여, 클러스터 내에 존재하는 어떠한 노드라도 최종적으로 클러스터 헤더에서는 라우팅 경로를 알 수 있게 되게 있다. 표 1은 라우팅 테이블의 일부를 나타내고 있다.

표 1. 라우팅 테이블

Source Node	Parent	Destination node				
Node 17	15	15	16			
		15	16			
Node 16	15	15	17			
		15	17			
Node 15	1	1	16	17		
		1	16	17		
:						
Node 9	1	1				
		1				
:						
Node 1	0	9	15	16	17	
		9	15	15	15	
Node 0	-	1	9	15	16	17
		1	1	1	1	1

각 노드는 직접 통신이 가능한 이웃 노드들과는 바로 통신할 수 있다. 만약 노드의 통신 범위는 벗어나지만 라우팅 테이블에 존재하는 목적지는 트리를 따라 전송 가능하다. 라우팅 테이블에 존재하지 않는 노드와 통신하기 위해서는 parent로 중계를 요청하여 목적지로 메시지 중계를 요청한다.

2) 토플로지(Topology) 수정

클러스터 헤더는 모든 노드들로부터 송출되는 링크 상태 보고를 근거로 각 멤버 노드들과의 최적 경로를 주기적 계산하여, 토플로지 수정 메시지를 통해 각 노드들에게 통보한다. 최적 경로 선정 방법은 가장 작은 홉수를 가지는 경로를 근거로 한다. 만약 동일한 홉수를 가지는 다수개의 경로가 존재하면 parent 노드의 식별번호가 가장 최소의 NID를 가지는 경로를 선택한다. 클러스터의 멤버 노드는 parent 노드가 변경되어 명시된 토플로지 수정 메시지를 수신하게 되면 메시지에 명시된 것과 같이 parent 노드를 변경한다.

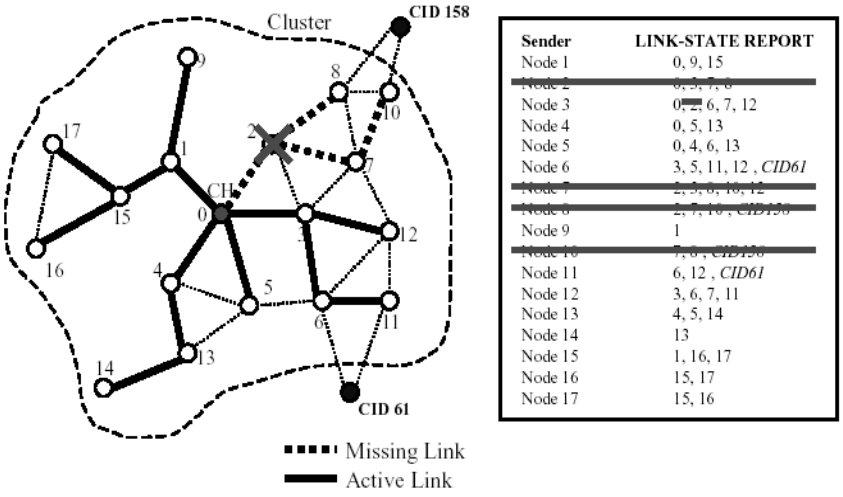
표 2. 토플로지 수정

Node	Parent
1	0
2	0
3	0
4	0
5	0
6	3
7	2
8	2
9	1
10	7
11	6
12	3
13	4
14	13
15	1
16	15
17	15
CID61	6
CID158	8

3) 노드 고장으로부터 링크 회복

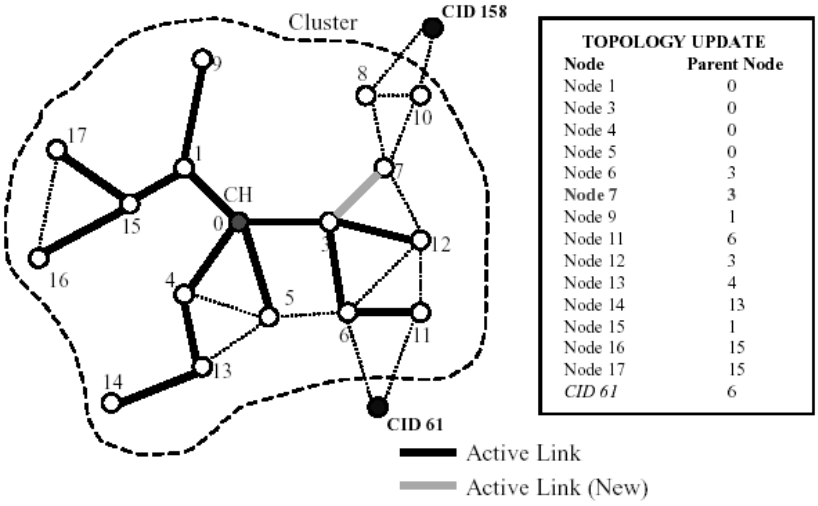
만약 멤버 노드 중에 하나가 고장이 발생하여 통신이 불가한 경우에는 클러스터 트리 경로는 재구성되어야 한다.

그림 18에서 2번 노드가 동작이 중단되면 노드 2, 7, 8, 10으로부터 링크상태보고가 수신되지 않는다. 클러스터 헤더는 보고되지 않는 노드와 각 노드에서 송출되는 링크상태보고의 네이버 목록을 참조하여 고장 난 노드를 검출한다. 이후에 노드 3번을 통해 1단계로 토플로지 수정 메시지를 송출하여 노드 7번의 링크를 회복시킨다. 2단계로 노드 7번을 통해 8번 10번의 링크를 회복시킨다. 그림 19와 그림 20 참조.



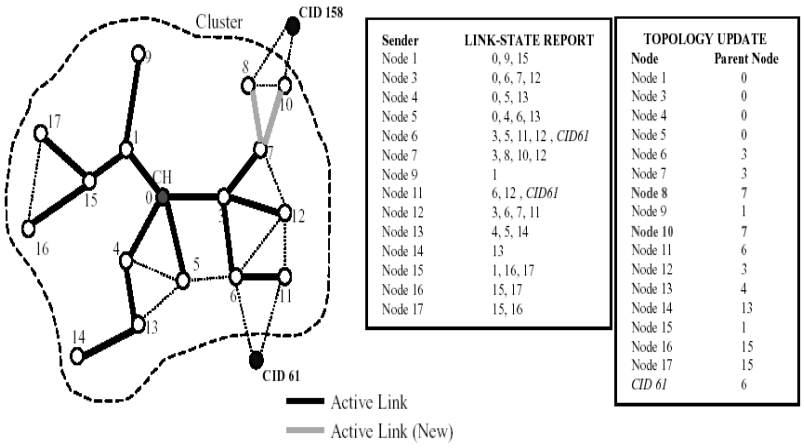
Sender	LINK-STATE REPORT
Node 1	0, 9, 15
~~Node 2~~	~~0, 5, 7, 8~~
Node 3	0, 2, 6, 7, 12
Node 4	0, 5, 13
Node 5	0, 4, 6, 13
Node 6	3, 5, 11, 12 , *CID61*
~~Node 7~~	~~2, 3, 8, 10, 12~~
~~Node 8~~	~~2, 7, 10 , CID158~~
Node 9	1
~~Node 10~~	~~7, 8 , CID158~~
Node 11	6, 12 , *CID61*
Node 12	3, 6, 7, 11
Node 13	4, 5, 14
Node 14	13
Node 15	1, 16, 17
Node 16	15, 17
Node 17	15, 16

그림 18. 2번 노드가 동작이 중단된 경우



TOPOLOGY UPDATE	
Node	**Parent Node**
Node 1	0
Node 3	0
Node 4	0
Node 5	0
Node 6	3
Node 7	**3**
Node 9	1
Node 11	6
Node 12	3
Node 13	4
Node 14	13
Node 15	1
Node 16	15
Node 17	15
CID 61	6

그림19. 1단계 7번 노드의 링크 회복



Sender	LINK-STATE REPORT
Node 1	0, 9, 15
Node 3	0, 6, 7, 12
Node 4	0, 5, 13
Node 5	0, 4, 6, 13
Node 6	3, 5, 11, 12 , *CID61*
Node 7	3, 8, 10, 12
Node 9	1
Node 11	6, 12 , *CID61*
Node 12	3, 6, 7, 11
Node 13	4, 5, 14
Node 14	13
Node 15	1, 16, 17
Node 16	15, 17
Node 17	15, 16

TOPOLOGY UPDATE	
Node	**Parent Node**
Node 1	0
Node 3	0
Node 4	0
Node 5	0
Node 6	3
Node 7	3
Node 8	7
Node 9	1
Node 10	7
Node 11	6
Node 12	3
Node 13	4
Node 14	13
Node 15	1
Node 16	15
Node 17	15
CID 61	6

그림 20. 2단계 8번 및 10번 노드의 링크 회복

만약 클러스터 헤더에 문제가 발생하면 HELLO 메시지의 송출이 중단되어 모든 멤버들이 알게 된다. 이 경우에는 클러스터 초기 형성과 동일한 방법으로 재구성이 이루어 져야 한다.

4. 수치해석

멀티 홉 클러스터 네트워크에서는 중간 노드들에서의 데이터 모음과 통합을 통하여 클러스터 헤드로 가는 메시지 수를 줄임으로 센서 노드의 에너지 소모를 감소시키는 것이다. 그림 21은 개별 전송과 데이터 모음의 차이를 나타내고 있다. 노드 레벨(Node Level)은 클러스터 헤드로부터의 홉수를 나타내며, 클러스터 헤더는 레벨이 0이고, 클러스터 헤더의 child는 레벨 1이 된다.

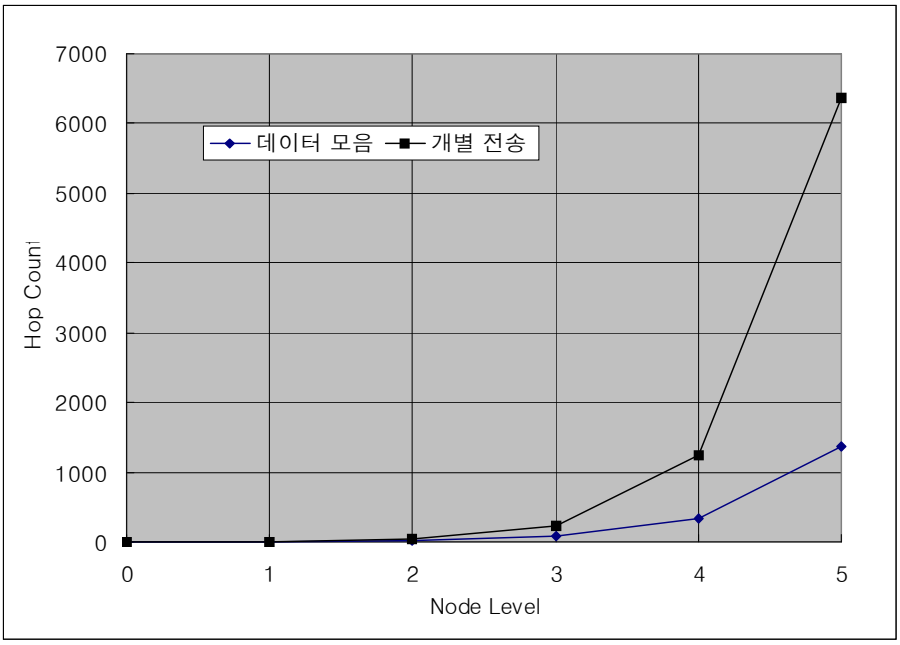


그림 21. 개별 전송과 데이터 모음의 홉수 비교

그림 21은 한 노드에서 child를 4개씩 가지면서, 노드 레벨 5까지 각 노드에서 클러스터 헤드로의 개별 전송할 때의 전체 홉수와 데이터 모음을 수행했을 때의 홉수를 비교하여 나타내고 있다. 개별 전송에 비해 데이터 모음을 수행 했을 때의 홉수가 약 1/5로 감소함을 나타낸다. 데이터 모음에 요구되는 전력에 비해 데이터 전송에 요구되는 전력이 상대적으로 아주 큰 점을 고려하면 멀티홉 클러스터 네트워크에서의 데이터 모음은 센서 노드의 에너지 절약에 매우 중요하다는 것을 나타내고 있다.

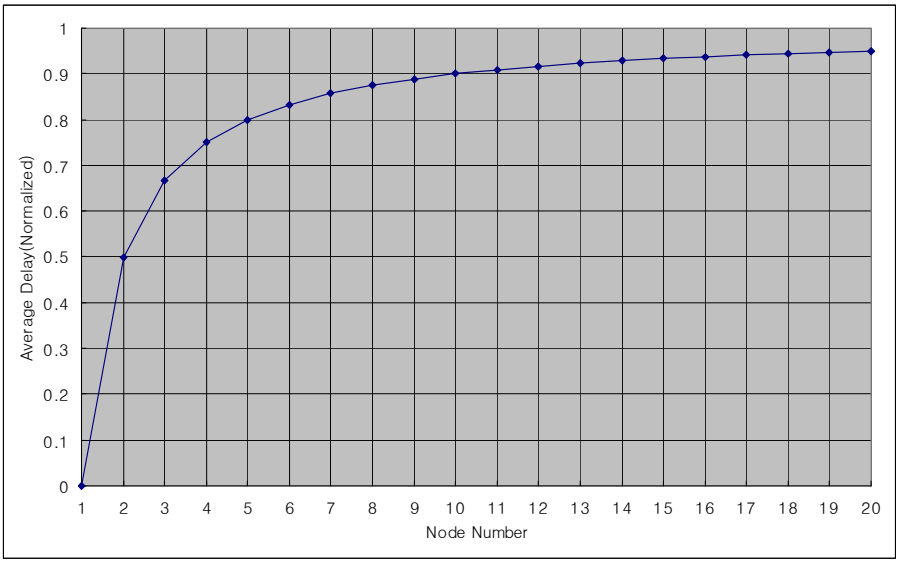


그림 22. 정규화된 데이터 모음의 평균지연

데이터 모음을 수행하기 위해서는 필연적으로 지연이 수반된다. 그림 22는 센서 노드 수에 따라 데이터 모음을 위해 중간 노드에서 기다리는 평균 지연을 정규화 시킨 것이다. 노드의 개수가 증가될수록 기다리는 시간이 길어지는 것을 나타내고 있다. 거의 99.9%가 비활성 상태에 있는 센서 노드의 특성을 고려하면 데이터 모음에 지연을 최소화 시킬 수 있는 방안이 연구되어야 한다.

멀티 홉 클러스터 네트워크에서 해결해야 할 또 하나의 중요한 문제점은 클러스터 헤드의 에너지 소모 문제이다. 클러스터 헤드는 멀티 홉 네트워크 내의 모든 통신이 집결되고, 처리되므로 일반 노드에 비해 에너지 소모가 많다. 이를 해결하기 위해 클러스터 헤드를 특정 시간 마다 무작위로 순환 시키는 LEACH(Low Energy Adaptive Clustering Hierarchy) 방안이 제안되어 있다. LEACH 방안은 각 노드에 에너지 소모를 균등하게 배분하는 것은 해결했지만, 클러스터 헤더가 무작위로 선정됨에 따른 문제점들을 안고 있다. 멀티홉 클러스터 네트워크에서는 헤드를 중심으로 최적화되게 트리 형태로 형성되어야 지연이 최소화 될 수 있다. 만약 무작위로 선정되는 클러스터 헤드가 기존 헤더를 중심으로 최적화로 형성되어 있는 상태에서 트리의 가장 말단 노드를 선정하게 되면 최악의 지연이 발생하는 결과를 초래한다.

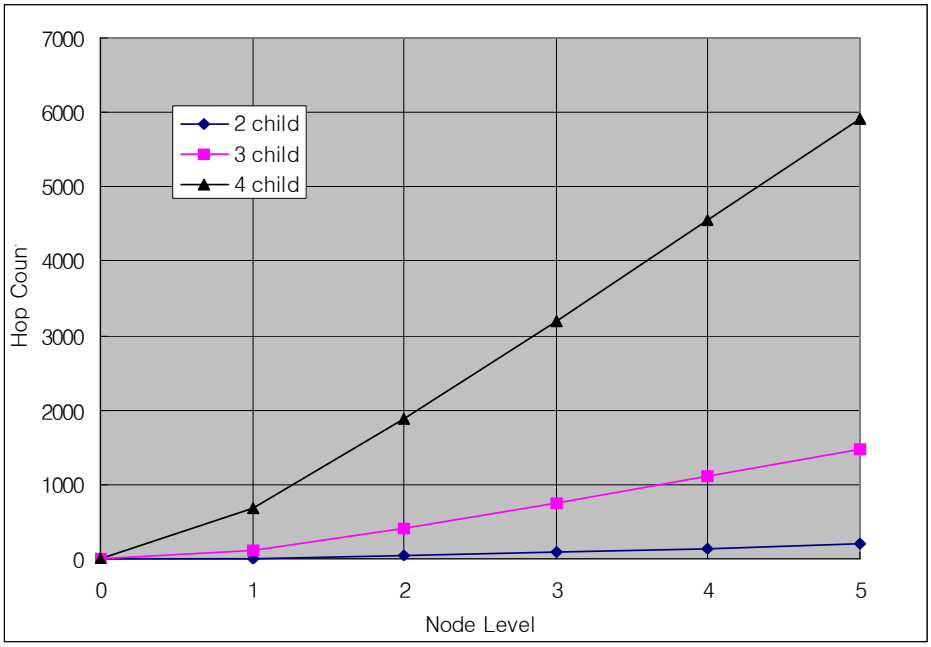


그림 23. 클러스터 헤더 변경에 따른 추가되는 홉수

그림 23은 무작위로 선정되는 클러스터 헤드의 위치에 따라 추가되는 홉수를 나타내고 있다. 4개의 child를 가지는 경우에 노드 레벨 5, 즉, 헤드로부터 5홉 하위의 노드가 선정되면 약 8.7배의 홉수가 늘어나게 된다. 이러한 결과는 LEACH에서 얻을 수 있는 4~8배의 효과를 상쇄시키는 결과를 초래한다.

Ⅳ. 결 론

무선 센서 네트워크는 필요한 경우에 임의의 형태로 형성되는 ad doc 네트워크이며, 네트워크의 형성이나 유지 관리가 인간이 최소한으로 관여되어야 하는 자가 구성 네트워크이다. 이러한 네트워크의 각 구성 센서노드들은 일회용으로 무작위로 배포할 수 있게 값이 저렴하면서도 다양한 접속 가능한 상황에서 아주 우수한 배터리 수명을 유지할 수 있어야 한다. 이러한 구현된 센서 노드들은 에너지 공급과 대역폭에 제한이 있다. 이러한 문제 해결에는 네트워킹 프로토콜 스택의 모든 계층에서 에너지 절약에 대한 사항이 고려되어야 한다.

본 논문에서는 클러스터 센서 네트워크에서 네트워크 관리를 효과적으로 지원할 수 있는 방안에 대해 살펴보았다. 클러스터 네트워크는 계층적 트리 구조로 대규모로 구성하기가 용이하면서 라우팅 프로토콜이 단순하다는 점에서 장점을 가진다. 또한 중간 노드에서의 데이터 모음을 통한 통신 메시지 수를 줄임으로 센서 노드의 전력을 크게 절약할 수 있지만, 중간 노드에서 메시지 전달 지연이 커지는 단점이 있다. 그리고 LEACH 방안에서 제안된 것과 같이 클러스터 헤드가 무작위로 선정되면 센서 노드들의 전력 소모는 균등하게 배분되지만, 만약 클러스터 헤드가 지역적인 말단에 선정되는 경우에는 최대 8.7배의 홉수가 늘어나게 되어 이에 따른 메시지 지연이 크게 증가하는 문제점이 있다.

앞으로의 연구 방향은 클러스터 트리 네트워크에서 트리 형성의 최적화 방안과 메시지 지연이 최소화되는 클러스터 헤더의 에너지 균등 소모 방안에 대해 지속적인 연구가 계속되어야 한다.

[참고 문헌]

[1] Ian F. Akiyldiz et al., "A survey on Sensor Networks", IEEE Communications Magazine, pp. 102-114, Aug. 2002.

[2] K. Sohrabi, et al., "Protocols for self-organization of a wireless sensor network," IEEE Personal Communication, Vol. 7, No. 5, pp. 16-27, Oct. 2000.

[3] Edgar Callaway, "Wireless Sensor Networks : architecture and protocols", Auerbach Publications, 2004.

[4] Ed Callaway, "Cluster Tree Network", IEEE P802.15 Working Group for WPANs, April 2001.

[5 W. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless sensor networks," in the Proceeding of the Hawaii International Conference System Sciences, Hawaii, January 2000.

[6] W. Heinzelman, et al., "Energy-Scalable algorithms and protocols for Wireless Sensor Networks", in the Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP '00), Istanbul, Turkey, June 2000.

[7] W. Ye, J. Heidemann and D. Estrin, "An Energy-Efficient MAC Protocol for Wireless Sensor Networks", in the Proceedings of IEEE Infocom 2002, New York, NY, June 2002.

[8] K. Akkaya and M. Younis, "A Survey of Routing Protocols in Wireless Sensor Networks, " in the Elsevier Ad Hoc Network Journal, Vol. 3/3 pp. 325-349, 2005.

[9] 배정숙, 김성희, "무선 센서 네트워크에서의 라우팅 프로토콜", 한국전자통신연구원 주간기술동향, 1140권호, 2004.04.

[10] S. Hedetniemi and A. Liestman, "A survey of gossiping and broadcasting in communication networks," Networks, Vol. 18, No. 4, pp. 319-349, 1988.

[11] W. Heinzelman, J. Kulik, and H. Balakrishnan, "Adaptive protocols for information dissemination in wireless sensor networks," in the Proceedings of the 5th

Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'99), Seattle, WA, August 1999.

[12] C. Intanagonwiwat, R. Govindan and D. Estrin, "Directed diffusion: A scalable and robust communication paradigm for sensor networks", in the Proceedings of the 6th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'00), Boston, MA, August 2000.

[13] D. Estrin, et al., "Next century challenges: Scalable Coordination in Sensor Networks," in the Proceedings of the 5th annual ACM/IEEE international conference on Mobile Computing and Networking (MobiCom'99), Seattle, WA, August 1999.

[14] A. Manjeshwar and D. P. Agrawal, "TEEN : A Protocol for Enhanced Efficiency in Wireless Sensor Networks," in the Proceedings of the 1st International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing, San Francisco, CA, April 2001.

[15] A. Manjeshwar and D. P. Agrawal, "APTEEN: A Hybrid Protocol for Efficient Routing an Comprehensive Information Retrieval in Wireless Sensor Networks," in the Proceedings of the 2nd International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile computing, Ft. Lauderdale, FL, April 2002.

무선 센서 네트워크에서 클러스터 기반 라우팅 프로토콜 연구

문 준 호

慶州大學校 大學院

컴퓨터電子工學科

指導敎授 趙 茂 昊

(초록)

클러스터 기반의 계층적 트리 구조는 대형화가 용이하면서 라우팅 프로토콜이 단순하다는 점에서 장점을 가진다. 본 논문에서는 무선 센서 네트워크에서 클러스터 트리 네트워크 관리를 효과적으로 지원할 수 있는 방안과 라우팅 프로토콜의 데이터 모음과 지연에 대해 분석한다. 수치 해석의 결과에서 보면 중간 노드에서의 데이터 모음을 통한 통신 메시지 수를 줄임으로 센서 노드의 전력 소모를 크게 절약할 수 있지만, 이에 따른 메시지 전달 지연이 커짐을 알 수 있었다. 또한 클러스터 헤드를 무작위로 선정하게 되면 메시지 지연이 매우 많이 증가 될 수 있는 것이 확인되었다.

A Study on Cluster tree Routing Protocol in Wireless Sensor Networks

Moon Jun Ho

Department of Computer and Electronic Engineering
The Graduate School
Gyeongju University

(Supervised by Professor Cho, Moo Ho)

(Abstract)

A hierarchical tree structure of clusters has advantages for the network design due to its scalability and simple routing protocol. In this paper, the cluster tree routing protocol is studied for the wireless sensor network. From the numerical analysis results, the data aggregation in the intermediate nodes reduces the number of communication message and saves the energy of sensor nodes, but it may result in increased data traffic latency. And also the random selection of cluster head can increase the relaying hops very high.

감사의 글

이 논문이 나오기까지 도와주신 주위의 많은 분들께 감사의 마음을 드립니다.

자상하신 성품으로 부족한 저를 제자로 받아 주시고, 석사 학위 과정동안 좋은 논문이 될 수 있도록 많은 가르침과 정성으로, 학문뿐만 아니라 바른 인생의 길로 인도하여 주신 조무호 지도 교수님께 머리 숙여 진심으로 깊은 존경과 감사를 드립니다.

바쁘신 가운데서도 학위 논문 심사 과정동안 관심과 충고로 인도하여 주신 심사위원 임찬호 교수님과 고택범 교수님께 또한 감사드리고, 학문의 길을 알차게 마무리 할 수 있도록 격려를 보내준 어머님과 장인어른, 장모님께도 감사드리며, 사무실에서 격려를 아끼지 않았던 명균이형에게도 서면으로나마 감사드립니다.

그리고, 항상 사랑스런 아내 선화와 아들 서진이에게 이 기쁨을 함께 하며, 사랑의 말을 전합니다.