

Ad-hoc 센서네트워크를 위한 최적 라우팅 경로 설정 알고리즘

Optimal Routing Path Selection Algorithm in Ad-hoc Wireless Sensor Network

장인훈, 심귀보

중앙대학교 전자전기공학부

In-Hun Jang, Kwee-Bo Sim

Electrical & Electronic Engineering, Chung-Ang University, Seoul, Korea,

E-mail: kbsim@cau.ac.kr

요 약

다단계 호핑(multi-hop) 무선 센서네트워크의 라우팅 경로를 설정하는데 있어서 데이터 전송의 요구가 있는 경우에만 경로를 만드는 Demand-Driven 방식의 대표적인 방법이 DSR(Dynamic Source Routing)인데 라우트 레코드를 패킷에 실어 보내기 때문에 이 또한 센서 노드들의 수가 많아질수록 패킷이 무거워질 수밖에 없다. 본 논문에서는 DSR 프로토콜을 기반 하면서도 라우팅 테이블을 적절히 이용하여 노드 수 증가에 대해 고정된 패킷크기를 가지도록 하였으며 라우팅 비용함수를 적용하여 각 센서노드들의 전원 소모량, 이동성(Mobility), 네트워크 내에서의 Traffic, 거리(Hop) 등을 복합적으로 고려한 안정적이고 신뢰성 있는 최적의 라우팅 설정알고리즘을 제안한다.

1. 서론

지난 몇 년 동안 스마트센서의 제조기술의 발전과 함께 이를 이용하여 센서 네트워크를 어떻게 구성하고 응용할 것인가에 대한 문제가 커다란 이슈가 되어왔고, 많은 연구 성과를 이루어 왔다. 특히 블루투스, 지그비, RFID와 같은 저가의 저 전력 무선통신 기술의 실용화와 함께 그 관심은 더욱 더 높아지고 있다.

센서 네트워크는 기존의 유무선 네트워크와 다른 여러 가지 특징들을 가지고 있는데, 주요한 특징으로는 센서 네트워크를 구성하는 각각의 센서노드들의 수가 수십에서부터 수백, 수천 이상의 대규모로 구성되어질 수 있다는 점과, 배터리를 사용하기 때문에 저 전력으로 작동되어야 한다는 것이다. 또한 저 전력으로 작동하기 때문에 통신반경이 짧아 Base station과의 직접적인 통신이 아닌 다단계 호핑 방식(multi hopping)으로 통신을 하며 센서노드들의 이동 또는 fail 등으로 인해 네트워크 토폴로지가 빈번히 변한다는 것을 주요한 특징으로 들 수 있다[1,2].

특히 전원 소비를 최소화 하기 위한 노력들이 많이 있어 왔는데 센서노드에 있어서 무선 송수신을 하기 위한 전원소비가 데이터를 획득하고 처리하는데 드는 전원소비에 비해 거의 1000배 정도 더 들어간다고 알려져 있다[2]. 이러한 이유 때문에 전통적으로 data aggregation, join, average와 같이 데이터 자체에 대한 프로세싱을 통해 무선 송수신을 최소화 함으로써 전원소비를 줄이려는 시도가 많이 있어왔다[2,3,4]. 그러나 그 적용에 따라서는 데이터 프로세싱이 불가능한 경우가 있다. 대표적인 예로 메디컬 방면에 응용이 되어 환자뿐만 아니라 일반인의 건강상태를 감지하는 것이 각 센서노드들의 역할인 경우에는 aggregation, join, average 등과 같은 방법을 사용할 수 없다.

또 다른 방법으로 selection 순서의 재정의 그리고 위치정보를 이용한 선택적인 query 등과 같은 query optimization을 통해 네트워크 전체의 전원소비를 줄이려는 시도가 있어왔다. 특히 메타 데이터(Meta data)의 특성(Attribute)을 기준

으로 SRT(Semantic Routing Tree)를 구성하여 query processing을 하는 방법[2]이 제안되었으며, 위치정보를 가지고 지도검색과 같은 공간 데이터베이스의 검색에 널리 이용되는 R-Tree기반의 라우팅 트리(Routing Tree) 구성[5] 등이 연구되어져 있다.

또한 위치정보와 각 센서노드의 센싱 범위, 통신 반경 등에 관한 정보를 이용하여 사전에 센서노드들을 잘 배치시키면 네트워크 자원을 좀 더 효율적으로 사용할 수 있기 때문에 센서의 효율적인 배치와 관련된 연구가 있어왔다[6,7].

이러한 방법들은 위치정보나 사전정보를 이용하여 실제 네트워크에 참여하는 센서노드들의 수를 줄여줌으로써 전체 네트워크 차원의 전원 소비효율을 높이려는 시도로 볼 수 있다.

본 논문에서는 다수의 센서노드들이 특정한 위치에 고정되어 있으며 또 다른 센서노드들은 이동성을 가지고 빈번하게 자신의 위치를 변경하는 것을 가정한다. 즉 고정된 센서노드들과 이동하는 센서노드들이 혼재되어 있는 환경의 Ad-hoc 무선 네트워크 환경을 가정한다. 이러한 환경에서 다단계 호핑 방식(multi hopping)의 센서네트워크 통신을 하는 경우 이동성을 가진 센서노드들이 많아질수록 네트워크의 경로를 구성하고 유지하는데 드는 비용은 점점 더 많아진다. 그렇기 때문에 라우팅 알고리즘은 네트워크 경로를 구성하고 유지하는데 있어 직접적인 데이터 송수신과 관련이 없는 송수신을 줄이고 traffic부하를 최소로 해야만 한다.

기존의 Ad-hoc routing protocol은 크게 Proactive routing protocol(Table-driven 방식)과 Reactive routing protocol(Demand-driven 방식)으로 나눌 수 있는데, 전자의 대표적인 방법이 DSDV(Destination-Sequenced Distance-Vector) Routing[8]이고 후자의 대표적인 방법이 DSDV기반의 AODV(Ad-hoc On-Demand Distance-Vector)[9]와 DSR(Dynamic Source Routing)[10,11]이 있다.

본 논문에서 제안하는 알고리즘은 DSR을 기반으로 하고 있으며, multi-hopping 방식의 무선 Ad-hoc 환경에서 좀 더 안정적이고 신뢰성 높은 라우팅 경로를 찾고 유지하도록 하여 결과적으로 전체 네트워크의 효율을 높이는데 기여하고자 한다.

이를 위해 2장에서는 대표적 Table-driven 방식인 Distance Vector Routing에 대해 간략히 서술하며 3장에서는 DSR의 기본 내용과 함께 본 논문에서 제안하는 라우팅 비용함수를 적용한 라우팅 방식을 다루겠다.

2. Distance-Vector Routing

센서 네트워크상의 한 노드에서 다른 노드로 어떤 데이터를 전달하기 위해서는 여러 단계의 센서노드를 거친다. Distance vector algorithms에 있어서 각 센서노드 사이의 거리는 어떤 값으로 표시되며, 두 센서노드간의 경로는 각 센서노드 사이의 거리의 합이 최소가 되는 경로로 정의한다. 실질적으로 두 센서 노드사이의 거리는 일반적으로 거쳐 가는 센서노드들의 수, 즉 hopping count가 된다. 각 노드는 목적지 노드에 대한 리스트, 목적지로 가기위해 될 다음 노드의 주소, 목적지까지의 hopping count 그리고 중복된 통신 요청(looping problem)을 회피하기 위한 sequence number를 항목으로 하는 routing table을 가지고 있다. 그리고 주기적으로 자신의 routing table을 이웃노드들에게 전달하여 전체적으로 최신의 routing table을 유지함으로써 각 목적지까지의 최단거리로 갈 수 있는 경로를 가지게 된다. 이처럼 실제 데이터의 주고받음에 대한 요청이 없는 경우에도 네트워크 경로에 대한 최신 정보를 유지하기 위해 주기적으로 routing table을 주고받는 통신 프로토콜을 proactive routing protocol 또는 table-driven 방식이라 하는데 이러한 경우에 필연적으로 필요 없는 통신 traffic이 발생하게 된다.

3. DSR(Dynamic Source Routing)에 기반한 최적경로 설정 알고리즘

Proactive routing protocol과 반대되는 개념으로 필요시에만 라우팅 경로를 만드는 방식을 Reactive routing protocol이라 하며 이에 대표적인 방법으로는 AODV와 DSR이 있다.

본 논문에서는 상기의 DSDV에서와 유사한 table을 가지며 기본적으로 DSR 프로토콜을 따르지만 DSDV나 DSR의 거리 값(일반적으로 hop count) 대신에 네트워크 내에서 센서노드 자신의 라우팅 비용을 나타내는 척도(Measurement)를 사용하여 routing 경로를 설정하는 방법을 제안한다.

3.1 Route Discovery

DSR 프로토콜에서 메시지 전송이 필요한 경우 Route request를 네트워크상에 broadcasting하는 source routing을 통해 라우팅 경로를 찾게 되는데 패킷 헤더에 목적노드에 도달할 때까지의 지나간 노드를 명시하여 전송한다.

Route request를 받은 센서노드는 그 목적지가 자신이면 route reply를 전송하고 그렇지 아니면 route request를 브로드캐스팅 하는데 그 과

정은 그림 1 과 같다. 그림 1 과 2는 경로를 선택하는데 있어서 hopping count를 이용하여 최단 경로를 구하고 있다.

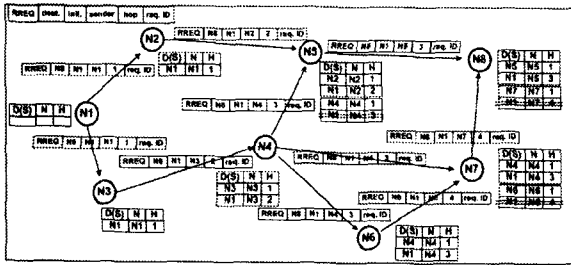


그림 1 DSDV의 Table을 가지는 DSR

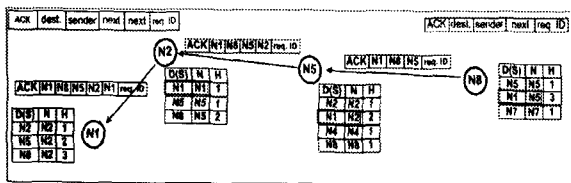


그림 2 거리(Hop-count)에 의한 라우팅 경로

우선 DSR에서는 그림 1, 2와 같이 hop-count를 기준으로 N1-N2-N5-N8로의 라우팅 경로가 생성되지만 본 알고리즘에서는 네트워크상에서의 라우팅 비용 정보를 추가하여 경우에 따라서는 N1-N3-N4-N5-N8이 최적 경로로 선택될 수도 있다. 예를 들어 Ad-hoc 네트워크 환경에서 노드 3은 고정되어 있지만 노드 2는 빈번히 이동한다면 이 경로는 오래가지 못해 다시 route request를 전 네트워크에 broadcasting해야만 할 것이다. 이처럼 mobility가 큰 노드는 가급적이면 경로선택에 있어서 제외하는 것이 전체적으로 더 나을 수도 있다. 또 다른 예로 우리가 자동차로 목적지까지 가는 경우에 있어서, 거리는 짧지만 차가 많아서 막히는 길과 먼 거리를 돌아가지만 막히지 않아서 시간을 절약할 수 있는 외곽도로가 있는 경우 종종 후자의 길을 택한다. 그러나 절약되는 시간에 비해 너무 많은 비용을 지불하게 된다면 막히더라도 짧은 길을 택할 것이다. 본 논문에서는 이러한 개념들을 기본으로 최적의 경로 선택 알고리즘을 다음과 같이 제안한다.

i 번째 센서노드를 SN_i 로 표시하면, SN_i 자신이 네트워크에 참여하여 통신을 하기 위해 지불해야 하는 비용을 P_i 라고 하면,

$$P_i = w_{11} \times Power_i + w_{12} \times Mobility_i + w_{13} \times Freq_i \dots (1)$$

w_{ij} 는 가중치이며, $Power_i$ 는 일반적인 센서네트워크에서 가장 큰 관심사가 되는 센서노드

SN_i 의 전원관련 정보이다. 자신의 power가 일정 수준 이하로 떨어지면 자신과 직접적인 관련이 있는 송수신은 유지하면서 단순 라우팅을 위한 송신은 off함으로써 다소 이기적인 통신을 할 필요가 있다. 또한 power의 감소율을 통해 그 노드의 부하정도를 알 수 있으며 이를 통해 부하가 적은 쪽으로 분산시킨다. $Freq_i$ 는 센서노드 SN_i 가 네트워크에 참여하는 빈도로서 power의 감소율로 대신할 수 있다. 이 값이 클수록 네트워크 collision이나 traffic 부하가 클 확률이 높다. 즉 상술한 '막히는 길'이 되는 것이다. $Mobility_i$ 는 SN_i 의 이동성을 나타내며 이 값이 클수록 네트워크상의 진출입이 빈번하므로 라우팅 경로로서는 신뢰성이 떨어지게 된다. 움직임이 심한 노드를 통한 경로보다는 움직임이 없는 노드를 통한 통신경로 설정을 택함으로써 link broken에 의한 route recovery를 줄이고 보다 안정적인 통신을 할 수 있다. 그 밖에 통신 에러의 발생빈도 또한 고려할 수 있다.

SN_i 에서 n 개의 센서노드들을 거쳐 SN_j 에 도달하는 경로에 있어서 예상되는 총 라우팅 비용 C_{ij} 는 다음과 같이 구할 수 있다.

$$C_{ij} = P_i + P_1 + P_2 + \dots + P_{n-1} + P_n + P_j \quad (2)$$

단순히 hop-count에 따른 기존 알고리즘으로는 그림 2와 같은 라우팅 경로가 설정되겠지만 라우팅 비용(Cost) 정보 C_{ij} 를 이용하여 이 값을 최소로 하는 경로를 택한다면 그림 3, 그림 4와 같은 라우팅 경로가 생성될 수도 있다.

여기서 C_{ij} 는 양의 값으로 계속 더해지는 값이기 때문에 노드를 많이 거칠수록 즉, 거리가 멀수록 커지므로 기존의 Hopping Count 값을 반영한다고 할 수 있다.

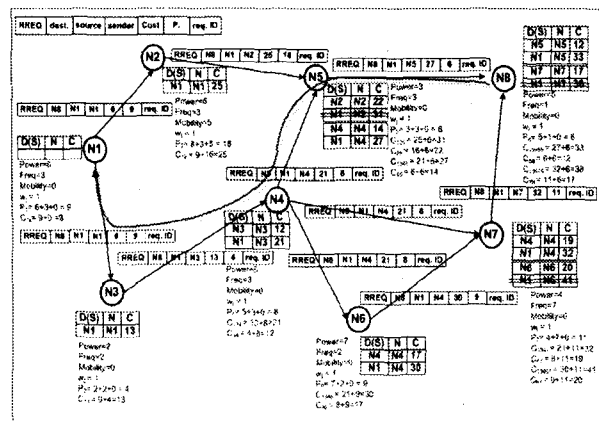


그림 3 Cost함수에 의한 Route Discovery

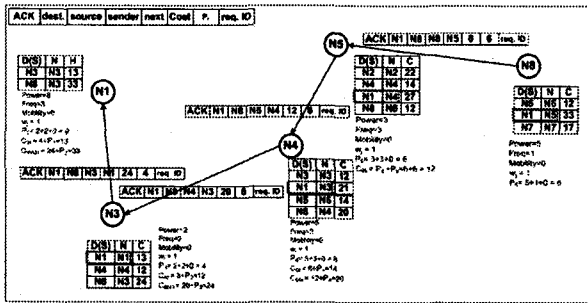


그림 4 Cost함수에 의한 최적 라우팅 경로 설정

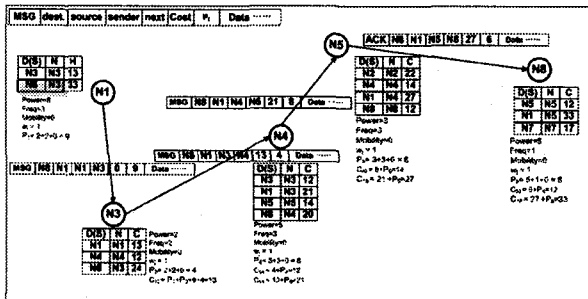


그림 5 메시지 전송

이 경우 라우팅 비용을 계산하기 위해 추가적인 송수신 데이터가 늘어나서는 안 되는데, 위에서 예들 든 파라메타 항목들은 주변 노드로부터 통신을 통해 전달된 정보가 아닌 노드 스스로 측정하고 계산한 내용이기 때문에 이로 인해 송수신 패킷의 크기가 커지지는 않는다. 또한 기존 DSR의 ACK. 메시지에 모든 경로를 실어 보내기 때문에 경로가 먼 경우 데이터가 커져 패킷이 커지는 것에 비해 본 제안의 경우는 데이터 길이가 일정하게 고정되어 네트워크 규모가 커지더라도 전체 패킷이 가벼워질 수 있다고 본다.

3.2 Route Maintenance

Route Maintenance는 기본적으로 3.1의 Route Discovery에서 찾아낸 경로를 잘 보관하고 유지하는 것에 관한 내용이다. 이를 위해서 각 센서노드들은 메모리에 학습한 경로를 저장하고 최신의 정보로 계속 업데이트해야 하며 때로는 캐시를 사용하여야 한다.

일반적으로 센서노드들의 하드웨어 사양 가운데 캐시나 메모리 크기는 그다지 크지 않기 때문에, 학습한 라우팅 경로를 저장하는 양은 센서네트워크의 크기가 커질수록 제한적일 수밖에 없다. 이 경우 메모리 및 캐시의 일반적인 최적화 알고리즘이 수행될 수 있다.

또한 C_{ij} 는 그 특성상 시간에 따라 변하는 값이기 때문에 이를 최신의 값으로 유지할 필요가 있으며, Link Broken이 발생하는 경우에는 이 또한 갱신해주어야 한다.

3.2.1 라우팅 비용(Cost) 정보 C_{ij} 의 갱신

C_{ij} 를 계산하기 위한 변수들, 즉 전원과 네트워크에의 참여도 등은 네트워크 환경이나 특정 목적에 따라 시간이 지나면서 계속해서 변하는 값들이다. 그러므로 지속적으로 이러한 값들을 갱신해주어야 할 뿐만 아니라 변하는 네트워크 환경에 더 잘 적응하기 위해서 구해진 라우팅 경로 또한 수명(Life Time)을 가질 필요가 있다.

그림 5에서 볼 수 있듯이 C_{ij} 는 메시지의 송수신이 이루어지는 것과 동시에 동적으로 계속 최신의 값으로 유지된다. 이를 위해서 각 센서노드들은 주기적으로 자신의 P_i 를 계산하고 메시지 전송시 그 정보를 실어 보낸다.

3.2.2 Link Broken발생과 경로 재구성

기본적으로 무선 네트워크는 유선에 의한 네트워크보다 통신신뢰가 떨어지는데, 센서 네트워크 상에 각 노드들은 전원문제에 의한 fail 및 이동 등에 의해서 빈번히 link broken이 발생한다. link broken의 발생뿐만 아니라 packet을 잃어버리거나 packet이 충돌을 일으켰을 때 이를 빠르게 찾아내고 재전송하기 위해 많은 경우 data link 레벨에서 hop-by-hop acknowledgement를 사용한다[11]. 이러한 경우, 통신루트의 유지보수는 패킷을 전송한 노드가 그 자리에서(at that hop) 다음 단계의 센서노드(hop)가 제대로 작동하고 있는지 아닌지를 결정함으로써 이루어진다.

라우팅 에러가 발생하면 RERR(Route Error) 메시지를 생성한다. 예를 들어 link broken에 의한 RERR가 발생하면 그림 6과 같은 제한적인 local route discovery를 수행하기도 하는데, 주변 노드들의 루트 캐시 또는 메모리에 목적지까지의 경로가 있는지를 확인한다.

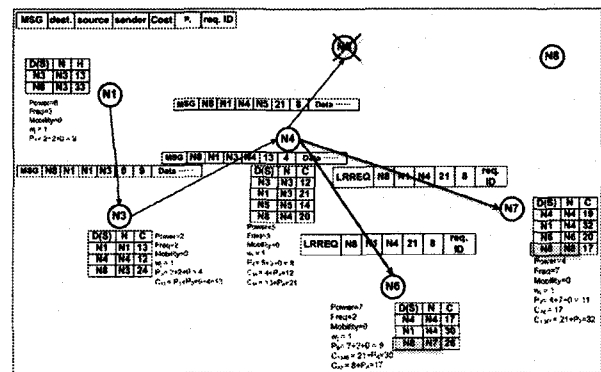


그림 6 Local Route Recovery

만약 주변 노드들로부터 목적지까지의 경로 정보를 얻지 못한다면 결국 최초의 노드로 RERR 메시지가 전달되며 최초의 노드는 3.1의 Route

Discovery를 통해 경로를 다시 설정 한다.

라우트 에러 패킷은 에러를 유발시킨 노드의 주소를 포함하고 있어서 최초의 노드로 RERR 메시지를 전달하는 과정에서 그 경로상에 있는 모든 노드들의 루트 캐시 또는 경로를 저장한 메모리로부터 링크오류를 일으킨 노드의 정보를 삭제하게 된다.

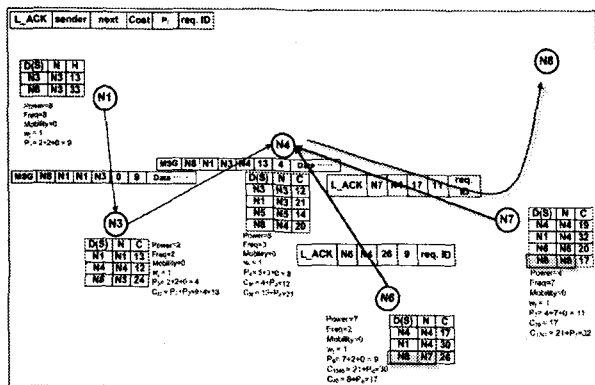


그림 7 복구된 경로를 통한 메시지 전송

만약 그림 7과 같이 주변 노드들의 메모리에 목적지까지의 경로 정보가 기록되어 있다면 그 경로를 새로운 경로로 하여 메시지 전송을 계속 수행한다. 이 경우 메시지 송수신 과정을 통해 새로운 경로 상에 있는 모든 센서노드들은 경로와 그에 따른 라우팅 비용 C_{ij} 를 갱신할 것이다.

3.2.3 Local Route Discovery

일반적으로 Route Discovery를 통해 각 노드에 학습된 네트워크 루트 정보는 시간이 지날수록 축적되며 어떤 두 노드 사이의 경로가 깨질 경우 그 주변 노드는 두 노드 사이를 잇는 또 다른 경로에 대한 정보를 가질 확률이 커진다.

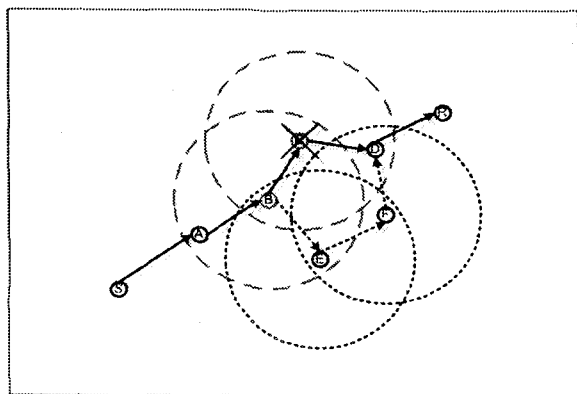


그림 8 이웃노드에 의한 Route Recovery

그림 8은 S-A-B-C-D-R로 이루어지는 경로가 C에 의해 깨지면서 B의 이웃 노드 E의 메모리에

학습되어 있던 R로의 경로를 통해 S에서 R로의 경로가 복구되는 경우를 보이고 있다. 그림 8에서 B의 통신 반경 내에 있는 E는 R까지 가는데 있어서 B와는 다른 경로를 가지고 있음을 알 수 있으며 이를 통해 S-A-B-E-F-D-R의 새로운 경로가 생성됨을 볼 수 있다.

이러한 경우를 가정하여 본 논문에서는 다음과 같은 새로운 local route discovery를 제안한다.

- 그림 6은 노드4에서 노드5로 패킷을 전송할 때 노드6과 노드7이 노드4의 통신 반경에 있는 경우로 노드5에 의해 link broken이 발생한 경우의 예로 이러한 경우, 노드4에서 노드5로 패킷을 전송할 때 노드6과 노드7은 이를 엿본다.
- 노드6과 노드7의 라우팅 저장 메모리에 노드4로부터 전송되는 패킷의 목적지가 있는 경우 이의 계속적인 엿보기를 위한 쓰레드를 생성시킨다.
- 일정 시간이 지나 link broken error가 발생된다면 노드6과 노드7은 자신의 메모리에서 노드5에 대한 라우팅 정보를 삭제한다.
- 노드6과 노드7은 자신의 메모리에서 목적지까지의 비용 정보를 포함한 link 복구용 패킷을 노드4로 전송한다.
- 노드4는 특별한 local route discovery를 주변 노드에 전송하지 않고서도 주변 노드로부터 새로운 경로에 대한 정보를 얻고 경로를 갱신한다.

제안된 local route discovery는 기본적으로 송수신에 소비되는 네트워크 비용보다는 노드 내에서 이루어지는 프로세스 비용이 훨씬 적다는 것을 가정하고 있으며, 이웃 노드의 수가 이웃 노드를 엿보는 프로세스의 부하가 지나치지 않을 정도란 것을 가정하고 있다.

3.3 Power Awareness

Power awareness문제는 센서네트워크 문제에 있어서 가장 기본적이고 중요한 문제로서 지금까지 매우 많은 연구가 있어 왔는데 본 논문의 참고자료의 대부분이 이를 다루고 있을 정도이다.

본 논문에서 각 센서노드는 자신의 현재 전원 상태를 계속적으로 모니터링하며 잔류전원이 적어질수록 라우팅 비용을 점점 크게 하여 네트워크상에서 라우터로서의 역할을 줄이도록 한다. 또한 전원상태가 기준 값 이하로 떨어지면 BAT. LOW 이벤트를 발생시키며, 이 이벤트가 발생하면 다른 노드들의 데이터를 단순 라우팅하는 라우터

로서의 역할은 하지 않고 자신의 센싱 데이터만을 송신하는 이기적인 송수신 전략을 세우게 된다. 또한 단위시간 당 전원의 감소율을 계산하여 네트워크 내에서 routing에 참여하는 빈도를 예측할 수 있으며 이 값을 traffic 부하로 생각할 수 있다. 결과적으로 이 값을 라우팅 비용 함수의 파라미터로 이용하면 traffic 부하를 분산시킬 수 있는 라우팅 경로를 생성시킬 수 있다.

4. 결론 및 향후과제

지금까지 제안한 라우팅 경로설정 알고리즘은 자신의 잔류전원, 네트워크상의 traffic 부하, 이동성(mobility), hop count를 이용한 거리를 종합적으로 평가한 Cost 함수를 이용함으로써 기존의 DSR과 비교하여 좀 더 안정적이고 지속적인 라우팅 경로를 택할 수 있다. 특히 본 논문에서는 위치가 고정되어 있는 센서노드와 이동성을 가진 센서노드가 공존하는 경우를 예를 들어 설명하였으며, 예의 경우 제안된 알고리즘은 link broken을 일으키기 쉬운 이동성을 가진 센서노드가 라우팅 경로에 참여할 수 있는 확률을 떨어뜨림으로써 결과적으로 네트워크 전체의 효율을 떨어뜨릴 수 있는 reroute discovery를 줄일 수 있음을 보여주었다.

또한 알고리즘을 위해 사용된 라우팅 비용(Cost) 함수의 경우, 네트워크의 크기나 목적에 따라 파라미터를 임의로 추가할 수도 있고 각 파라미터의 weight를 적절하게 조절하고 선택할 수 있기 때문에 상황에 맞게 좀 더 유연한 라우팅 경로를 설정할 수 있다.

향후 과제로는 경로가 결정된 이후에 센서노드의 추가 등으로 인하여 더 좋은 경로가 생긴 경우에 자동으로 이를 감지하고 경로를 업데이트하는 방법에 대한 추가적인 연구가 필요하겠다. 그리고 Cost 함수의 파라미터로서 좀 더 적절한 항을 찾아내고 이를 더 지능적으로 구현하는 것에 대한 연구가 추가적으로 요구된다.

감사의 글 : 이 논문은 서울시·중소기업청의 연구비지원에 의한 2005년도 중앙대학교 산학연 컨소시엄사업에 의해 수행되었습니다. 연구비지원에 감사드립니다.

5. 참고문헌

[1] Ian F. Akyildiz, W. Su, Y. Sankarasubramanian, and E. Cayirci, "A Survey on Sensor Networks," IEEE Communications Magazine, pp.102-114, Aug. 2002.

[2] S. Madden, M. J. Franklin, and J. M. Hellerstein, "The Design of an Acquisitional Query Processor for Sensor Networks," Proc. ACM SIGMOD, ACM Press, pp.491-502, 2003.

[3] S. Madden, M. J. Franklin, and J. M. Hellerstein, and W. Hong., "TAG: A Tiny AGgregation Service for AD-Hoc Sensor Networks," Proc. Symp. OSDI, 2002.

[4] Johannes Gehrke, Samuel Madden, "Query Processing in Sensor Networks," IEEE CS and IEEE ComSoc., pp.46-55, January-March 2004.

[5] Murat Demirbas, Hakan Ferhatosmanoglu, "Peer-to-Peer Spatial Queries in Sensor Networks," 3rd International Conference on Peer-to-Peer Computing (P2P 2003), pp.32-39, 1-3 Sep. 2003.

[6] Y. Zou and K. Chakrabarty, "Coverage-Oriented Sensor Deployment", Distributed Sensor Networks, R. R. Brooks and S. S. Iyengar (eds.), CRC Press, pp.453-481, 2005.

[7] S. S. Iyengar, A. Tandon, Q. Wu, Eungchun Cho, N. S. V. Rao, V. K. Vaishnavi, "Deployment of Sensors : An Overview," R. R. Brooks and S. S. Iyengar (eds.), CRC Press, pp.483-504, 2005.

[8] C. E. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing(DSDV) for Mobile Computers," comp. Commun. Rev., pp.234-244, Oct 1994.

[9] C. E. Perkins and E. M. Royer. "Ad hoc On-Demand Distance Vector Routing." Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications, New Orleans, LA, pp. 90-100, Feb. 1999.

[10] David B. Johnson. Routing in Ad Hoc Networks of Mobile Hosts. Proceedings of the Workshop on Mobile Computing Systems and Applications, pp. 158-163, IEEE Computer Society, Santa Cruz, CA, Dec. 1994.

[11] David B. Johnson, David A. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks," Mobile Computing, edited by Tomasz Imielinski and Hank Korth, Kluwer Academic Publishers, 1996.

[12] Xu Li; Zheng Baoyu, Yang Zhen, "Power and mobility-aware adaptive dynamic source routing in MANET," TENCON 2004. 2004 IEEE Region 10 Conference Volume B, 21-24 Nov. 2004 pp. 652 - 655 Vol. 2