

무선센서네트워크에서 에너지 효율 및 딜레이를 고려한 트리 재구축 알고리즘

이현, 염상길, 김동수, 추현승
 성균관대학교 정보통신대학

e-mail : {dusidu, sanggill2, dskim61, choo}@skku.edu

Rebuilding Tree Algorithm for Delay-Aware and Energy-efficient Data Aggregation in Wireless Sensor Networks

Hyun Lee, Sanggil Yeoum, Dongsoo Kim and Hyunseung Choo
 College of Information and Communication Engineering,
 Sungkyunkwan University

요 약

무선센서네트워크에서 센서 노드들은 정보를 수집 및 취합하기 위해 다양하게 사용되고 있다. 각 센서들은 배터리 전력을 사용하여 에너지 절약은 가장 중요한 이슈 중 하나이다. 현재까지 에너지 소모가 가장 큰 장거리 통신 시 에너지 절약 및 분산제어가 용이한 클러스터링의 데이터 병합 관련 분야는 꾸준히 관심을 받아오고 있다. 최근 이를 기반으로 데이터 병합 시 생기는 딜레이를 최소화하고, 에너지 소비량 도 고려한 다양한 알고리즘들이 제안되었다. 하지만 토폴로지 형성 시 데이터 병합 딜레이와 에너지 효율을 동시에 최적화하는 상황에서 장거리 노드 간 링크 생성 문제는 여전히 해결되지 않고 있다. 본 논문은 이러한 문제점을 해결하기 위해 노드 간 링크를 재구축하여 트리의 구조 유지하면서 링크들의 길이를 줄일 수 있는 트리 재구축 알고리즘을 제안한다.

1. 서론

무선센서네트워크에서는 특정 정보를 수집하기 위해 많은 센서 노드들을 사용한다. 각 센서 노드들은 배터리를 사용하며 인적이 드문 환경에서는 유지보수가 어렵기 때문에 에너지 절약은 무선센서 네트워크의 가장 큰 이슈 중 하나이다. 예전부터 무선센서네트워크에서는 분산 환경에서 제어가 용이하고, 에너지 소모가 가장 큰 장거리 통신에서 에너지를 절약 할 수 있는 클러스터링 관련 알고리즘들이 제안되고 있다. 하지만 지금까지 제안된 클러스터링 관련 기법들은 BS(Base Station)와 통신하는 헤드노드의 극심한 에너지 소비 및 노드 간 링크의 불균형으로 에너지 소비의 균형 문제가 해결되지 않고 있다.

분산 DADC(Delay-Aware Data Collection)기법[1]에서 이런 문제를 해결하고자 제안한 트리 구조로 네트워크를 구축하여 데이터 병합 딜레이를 최소화 했다. 하지만 에너지 효율 및 데이터 충돌에 대한 방안 등은 고려되지 않았다.

위 기법을 타겟으로 지연 최소화 및 에너지 효율 알고리즘인 DEDA(Delay-minimized Energy-efficient Data Aggregation) 기법[2]는 DADC 기법과 같은 네트워크 구조를 사용하며 특정 BackOff 기법을 이용해 노드 간 데이터 충돌과 전송거리를 일부 고려하였다. 또한 노드에 가중치를 두어 최종적으로 형성된 CH(Cluster Head)와 BS 사이의 거리, 클러스터 내 노드 간 거리를 최소화하여 에너지 효율을 높였다[2]. 하지만 클러스터 융합 시 노드 간 링크 길이의 불균형은 해결되지 않았고, 네트워크 구조를 구축하는 단계에서 생기는 오버헤드도 무시하지 못할 수준이었다.

본 논문에서는 이러한 문제를 해결하기 위해 구축된 HyperTree 트리를 재구축하여 노드 간 전송거리를 줄이고, 데이터 병합 시 딜레이는 최소화 시킬 수 있는 트리 재구축 알고리즘을 제안한다.

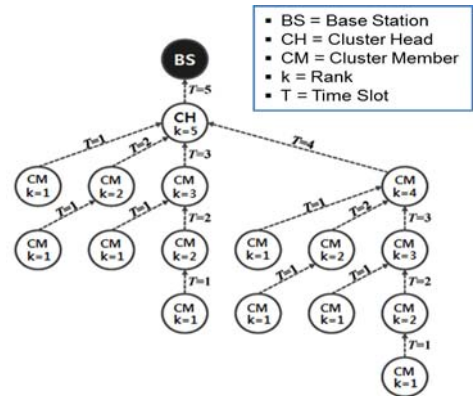
본 논문의 구성은 다음과 같다. 2장은 관련연구인 DADC와

DEDA 기법에 대해 언급하고, 3장에서는 제안 알고리즘을 기술한다. 마지막으로 본 논문의 결론을 4장에서 맺는다.

2. 관련 연구

2.1 DADC(Delay-Aware Data Collection)

클러스터링 네트워크는 여러 개의 클러스터들로 나뉘지며 각각의 클러스터는 하나의 CH(Cluster Head)를 갖고, 나머지 노드들은 CM(Cluster Member)가 된다. 데이터 수집은 데이터 전송횟수를 줄이기 위해 CH에서 수행된다. 그림 1은 DADC에서 제안한 트리구조이다. 본 논문 예서는 이 트리를 HyperTree라고 명명했다. HyperTree트리를 간단히 설명하면



(그림 1) DADC에서 제안한 트리구조

먼저 노드의 수(N)는 2^n 개로 한정한다. 각 노드는 1부터 n-1의 랭크(k)를 보유하며 랭크가 k인 한 노드는 1부터 k-1의 랭크를 보유한 자식노드를 갖게 된다.

DADC기법은 분산 환경에서 Bottom-Up Approach를

사용 하며 식 1과 같은 에너지 모델을 사용한다. 여기서

$$E_{TOT}(N) = C_1 + C_2 + C_3 \sum_{i=1}^N d_i^2 \quad (1)$$

C_1, C_2, C_3 는 상수이며 각각 C_1 은 마이크로 컨트롤 유닛, 센서보드의 소모에너지와 데이터를 전송받은 때의 소모에너지를 뜻하며 C_2 는 전송할 때의 전기회로에너지, C_3 은 전송할 때의 증폭에너지, d 는 거리이다. 즉 이 에너지 모델에서 에너지 소비량은 데이터 전송 거리의 제곱에 비례한다. DADC에서 사용하는 Bottom-Up Approach는 최초 지정된 범위에서 노드 간 통신 후 링크를 형성한다. 지정된 범위 내에 더 이상 노드의 수가 같은 클러스터가 없을 경우 범위를 넓혀간다. 전체 범위 내에 더 이상 같은 규모의 클러스터가 없는 경우, CH가 BS와 커넥션을 형성한다.

2.2 DEDA (Delay-minimized Energy-efficient Data Aggregation)

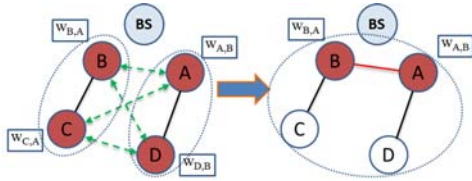
DEDA 기법[2]는 DADC와 같은 트리구조를 사용하며 가정 사항과 환경 또한 동일하다. DEDA는 데이터 전송 시 식 2과 같이 BackOff시간을 두어 충돌 문제를 보완했다. 여기서 C 는 상수이며 R 은 BackOff 시간이 같은 변수 간 우선순위를 두기 위한 랜덤 변수이고, d_{A-BS} 는 노드 A부터

$$t_{backoff_A} = \frac{C}{d_{A-BS}} + R \quad (2)$$

BS까지의 거리이다. 식 2에 의해 각 노드는 BackOff 시간을 기다린 후 자신과 가장 가까운 노드와 링크를 형성한다. 클러스터 내 노드의 수가 2개를 넘어가면 식 3를 이용하여 노드 간 가중치를 주고, 이를 이용하여 두 개의 CH를 선정한다.

$$W_{A,B} = (1-\alpha)d_{A,B}^2 + \alpha(d_{A,BS}^2 + d_{B,BS}^2) \quad (3)$$

선정된 두 CH는 그림 2과 같이 같은 노드의 수를 가진 다른 클러스터의 CH들과 서로 통신하여 거리를 계산하고, 가장 짧은 거리를 갖는 두 CH가 Chief가 되어 링크를 형성한다.



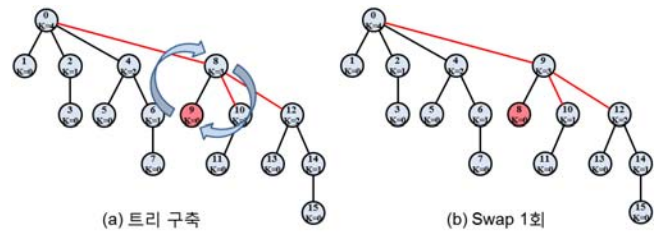
(그림 2) Link Weight Function 예시

3. 제안 알고리즘

기존에 제안된 DEDA기법은 BackOff를 사용하여 데이터 동시 전송 시 생기는 충돌을 예방하고, 주변 노드 및 BS와의 거리를 고려한 가중치를 구하여 노드 간 링크를 형성하므로 데이터 전송거리의 총 합은 Bottom-Up Approach 기법보다 항상 되었다. 하지만 Link Weight Function을 이용한 클러스터 융합 시 동일한 레벨의 클러스터 간 링크를 형성 가능한 가정 때문에 거리가 먼 CH 간 링크가 형성되는 경우가 많다.

데이터 전송거리의 제곱에 비례하는 에너지 모델을 사용하는 만큼 노드 간 링크의 길이를 줄이는 것은 매우 중요하다. 트리 구축 시 생기는 먼 노드 간 링크의 수를 줄이고자 트리 재구축 알고리즘을 제안한다.

제안 알고리즘은 HyperTree를 구축 후 구축된 트리를 재구축함으로써 링크의 길이를 줄인다. 교대 방법은 부모/자식 노드 간 역할교대이다. 그림 3과 같이 부모/자식 노드 간 교대는 특정 노드 u 의 형제노드 중 u 보다 큰 랭크(k)를 보유한 노드들을



(그림 3) 부모/자식 노드 간 교대

u 의 자식노드로 재편한다. 만약 u 의 부모노드 $Pr(u)$ 가 root가 아니면 u 가 조부모노드 $Gr(u)$ 의 자식노드가 되고, $Pr(u)$ 는 u 의 자식노드가 된다. 또한, 노드 u 의 랭크가 0이 아닐 경우 AVL트리의 로테이션과 같이 Swap 한다. 노드 간 역할교대 시 링크는 $(Pr(u), Gr(u))$ 와 $(Pr(u), w)$ 만 변경된다. 또한 부모/자식 간 랭크도 변경 된다. 노드 간 역할 변경 전/후의 링크 길이 제곱의 합을 뺀 값을 $\delta[u]$ 라고 표현하며, 이 값이 음수인 경우 전체 Tree Cost를 줄일 수 있다. 부모/자식 노드 간 교대 시 $\delta[u]$ 는 식 4를 이용하여 계산한다. 식 4에서 $d(a,b)$ 는 a 와 b 의 거리를 뜻하며 w 는 같은 부모를 가지며 랭크가 u 보다 큰 노드를 뜻한다.

$$\delta[u] = -d(Pr(u), Gr(u)) + (d(Gr(u), u) - \sum_{\substack{w \in ch(Pr(u)) \\ rank(w) > rank(u)}} (d(Pr(u), w) - d(u, w))) \quad (4)$$

트리 재구축 알고리즘을 사용하면 기존의 트리구축 기법만 사용하는 것 보다 노드 간 링크의 길이를 단축시킬 수 있고, 에너지모델 특성 상 각 노드의 에너지소모 또한 줄일 수 있을 것이다. 기존 트리 구조를 유지하면서 에너지 효율을 늘릴 수 있다는 점이 제안하는 알고리즘의 가장 큰 장점이다.

4. 결론

본 논문에서는 기존 HyperTree 구축 알고리즘들의 문제점인 장거리 노드 간 링크 형성으로 인한 과도한 에너지 소비를 해결하고자 트리 재구축 알고리즘을 제안한다. 이 알고리즘은 HyperTree를 만들 수 있다면 어떤 알고리즘을 사용하는지 여부와 상관없이 적용 가능하다. 또한 구축된 HyperTree의 구조를 유지한 채 재구축함으로써 데이터 병합 딜레이는 최적화 되고, 노드 간 링크의 길이를 줄여 노드의 에너지 소비를 줄일 수 있다.

ACKNOWLEDGEMENT

본 연구는 미래창조과학부의 차세대정보·컴퓨팅기술 개발사업(2010-0020727), 미래창조과학부 (한국산업기술 평가관리원)의 산업융합 원천기술개발사업 (정보통신) [10041244, 스마트TV2.0 소프트웨어 플랫폼] 및 미래창조과학부가 지원한 2014년 정보통신·방송(ICT) 연구개발사업의 연구 결과로 수행되었음.

참고문헌

- [1] C. Cheng, C. K. Tse, and M. Lau, "A Delay-Aware Data Collection Network Structure for Wireless Sensor Networks," IEEE Sensors Journal, 2011
- [2] Huu Nghia Le, Vyacheslav Zalyubovskiy, Hyunseung Choo, "Delay-minimized Energy-efficient Data Aggregation in Wireless Sensor Networks," Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), 2012