STO 기반 클러스터 헤더 선출 알고리즘

윤정현* · 이헌국 · 김승구**

충북대학교

STO-based Cluster Header Election Algorithm

Jeong-Hyeon Yoon* · Heon-Guk Lee · Seung-Ku Kim**

Chungbuk National University

E-mail : thera923@gmail.com / heonguklee@cbnu.ac.kr / kimsk@cbnu.ac.kr

요 약

본 논문은 센서 네트워크의 주요 문제점인 센서 노드의 이탈과 네트워크 변화에 따른 네트워크 수명 감소를 개선하기 위한 논문이다. 기존 Scalable Topology Organization(STO) 기반 지그비 트리 토폴로지 컨트롤 알고리즘은 전력 소모에 대한 방안을 고려하지 않아 네트워크의 수명이 비교적 짧았다. 이에 따라 매 라운드가 지날때마다 부모 노드를 새로 선출하고 네트워크 토폴로지를 새로 구성하는 클러스터 헤더 선출 기법을 도입하여 네트워크의 전체적인 수명을 연장하였다. 이에 대한 성능은 OMNet++ 시뮬레이터를 통해 동일한 실험 환경에서 기존의 STO 알고리즘과 제안하는 클러스터 헤더 선출 기법을 도입했을 때의 결과를 도출하였고 그 결과 네트워크 수명을 약 40% 증가시킬 수 있었으며 배터리 잔량 부분에서도 약 10%의 성능이 개선된 것을 확인 할 수 있었다.

ABSTRACT

This paper is about to improve the network life's reduction due to the deviation of sensor node and frequently change of network, the main problem of sensor network. The existing Scalable Topology Organization(STO)-based ZigBee Tree Topology Control Algorithm did not consider ways to consume power so the network lifetime is too short. Accordingly, per each round, electing a new parent node and consisting of the new network topology technique, The Cluster Header Selection, extending the network's overall lifetime. The OMNet++ Simulator yielded results from the existing STO Algorithm and the proposed Cluster Header Selection Technique in the same experimental environment, which resulted in an increase in overall network life by about 40% and an improvement of about 10% in performance in the remaining portion of the battery.

키워드

Sensor Network, ZigBee, Cluster, STO, Cluster Header

Ⅰ. 서 론

센서 네트워크는 물리 공간의 빛, 소리, 온도, 움직임과 같은 물리적 데이터를 센서 노드가 감지하고 이를 측정하여 중앙의 기본 노드로 데이터를 전달하는 구조를 가진 네트워크다. 이러한 센서 네트워크는 현재 시대의 흐름에 따라 작게는 수십에서 크게 수천, 수만 개에 이르는 센서 노드로 이루어지는데, 시간이 흐를수록 사용자가 원하는 응용 프로그램의 요구사항 기대치가 증가됨에 따라 배치되어야 할 센서 노드의 수는 더욱 더 늘어날 것으로 보인다. 이러한 센서 노드 수의 기하급수적인 증가와 외부의 환경적인 요인에 취약하다는 점, 또한 $1 미만의 값을 갖는 센서 노드의 특성상 비교적 취약한 하드웨어 설계방식에 의해 노드의 잦은 네트워크 이탈은 센서 네트워크에서 필수적으로 해결해야할 문제점들이다. 이에 따라 본

* speaker

** corresponding author

논문에서는 앞서 언급한 문제점들에 따라 발생하는 센서 네트워크의 특징인 빈번한 네트워크 변화 및 수명 연장에 대한 해결책을 제안한다.

네트워크의 수명을 연장하기 위한 방법으로는 여러 가지가 있다. 보내야 할 데이터의 패킷 사이즈를 줄이거나, 데이터 패킷의 발생량을 최소화하거나 송신 전력의 소비를 줄이거나 중계 역할을 맡은 노드의 부하를 줄여주어야 한다. 이러한 해결책을 이용하여 네트워크의 수명을 늘리는데 성공한 사례는 대표적으로 SigFox가 있다. SigFox는 데이터의 패킷 사이즈를 총 12 Byte로 줄이고 UNB(Ultra Narrow Band) 대역을 이용하여 제한된 전원으로도 장기간 노드를 사용할 수 있게 개발되었다. 또한 하루에 노드가 데이터를 보낼 수 있는 패킷 수를 140개로 제한하여 이 부분에서도 전력 소비를 줄이는 것에 성공하였다. 노드의 부하를 줄이는 방법으로는 데이터를 발생시켜 보내는 역할만 맡은 엔드 노드를 제외한 중계 역할을 맡은 부모 노드나 클러스터 헤더 노드의 적절한 교체가 있다. 말단에 존재하지 않고 다른 노드가 발생시킨 데이터를 코디네이터까지 중계해줘야 할 의무가 있는 노드들은 주변의 노드 수에 따라 크게 영향을 받으며 이에 따라 여러 가지 문제점이 발생한다. 1홉 단위로 연결되어 자신이 중계해줘야 할 노드의 수가 많아지면 그만큼 해당 노드는 데이터를 받을 때의 전력 소모와 보낼 때의 전력 소모까지 더 소비되어 다른 노드들보다 훨씬 더 빠른 배터리 방전 현상이 찾아오게 된다. 이처럼 하나의 노드에 부하가 집중됨에 따라 결국 배터리가 완전히 소진되어 네트워크를 지속시킬 수 없게 되면 이는 곧 네트워크의 붕괴에 이르게 된다. 이러한 부모 노드의 적절한 교체 기법이 없는 네트워크는 사용자 및 사용자 응용 프로그램의 요구사항을 만족시킬 수 없다.

II. 본 론

본 논문에서 설명할 기술은 전체 트리 네트워크에서의 로드 밸런싱과 빈번한 네트워크 변화에 따른 해결책으로, 네트워크의 수명 연장을 위한 기술이다. 클러스터가 형성된 후 클러스터 내의 클러스터 멤버 노드들은 발생한 데이터를 코디네이터에게 직접 보낼 수는 없다. 보내야 할 데이터가 발생한 노드는 자신이 속한 클러스터 헤더를 통해서 데이터를 보내야 한다.

이에 따라 클러스터 헤더는 자신과 1홉 단위로 연결된 자식 노드들이 발생시키는 데이터를 코디네이터에게 중계해야 할 의무가 있으며, 클러스터 헤더에 연결된 자식 노드들의 수가 많아질수록 클러스터 헤더의 부하가 급격히 커진다. 코디네이터에게 보내야 할 데이터가 많아질수록 클러스터 헤더의 전력 소비는 자신에게 속한 노드들의 수에 비례해 더욱 커지며 적절한 전력 소비 기술이 없다면 이는 곧 전체 트리 네트워크의 붕괴에 이르게 된다. 따라서 클러스터 헤더의 교체 전략이 필수적으로 따라야 한다. 기존의 클러스터 형성 프로토콜인 STO[1] 기반 지그비 트리 토폴로지 컨트롤 알고리즘에서는 네트워크 전체의 확장성만 고려했지만 클러스터의 설정과 클러스터 헤더의 선출에 있어서는 별도의 알고리즘이나 기법을 제시하지 않았다. 이에 따라 앞서 언급한 클러스터 멤버 노드의 네트워크 이탈에 따른 전체 네트워크 변화에 의해 발생하는 문제점들을 해결하기 위해 본 문단에서는 클러스터 헤더 선출 기법을 제안한다.

클러스터 헤더 선출의 과정은 다음과 같다. 처음 토폴로지 구성 시에는 STO 기반 지그비 트리 토폴로지 컨트롤 알고리즘에 기반하여 RSSI 값만을 이용하여 클러스터 헤더를 선출하지만 토폴로지 구성이 완료된 후 한 라운드를 진행한 뒤에는 클러스터 헤더 선출 기법을 이용해 새로운 클러스터 헤더를 선출한다. 한 라운드가 지나 새로운 클러스터 헤더를 선출한 뒤, 기존의 클러스터 멤버 노드들은 속해있던 클러스터에서 벗어나 새로운 클러스터 헤더를 찾는다. 이에 따라 클러스터는 새로운 클러스터 헤더와 클러스터 멤버 노드로 재구성된다. 이러한 과정에 따라 클러스터 헤더로 선출되었던 노드는 라운드가 지나도 과중한 에너지 소비와 부담에서 벗어나는 것이 가능하다. 또한 후술할 우선순위에 따른 선출 과정을 겪기 때문에 네트워크에 참여하는 모든 노드가 각자 균등하게 에너지 소비를 함으로써 특정한 하나의 노드만 에너지가 부족하게 되는 문제점은 크게 발생하지 않는다.

노드에 대한 클러스터 헤더 선출 우선순위를 P_c라 하고 배터리 잔량을 B_r이라고 한다면 우선순위 P_c는 아래의 수식과 같으며 새로운 클러스터 헤더는 기존의 방식과 다르게 RSSI 값만 이용하지 않고 노드들의 배터리 잔량과 트래픽 발생 패턴(a)을 분석하여 선출한다.

$$P_c = a^* \frac{RSSI}{-88dBm} + (1-a)^* \frac{B_r}{100\%}$$

위 수식에 의해 계산된 각 노드들의 우선순위 P_c에 따라 새로운 클러스터 헤더가 선출되면 클러스터 헤더는 기존의 클러스터를 해제하고 새로운 클러스터를 형성한다. 클러스터 헤더로 선출되니 노드는 주변 노드들에게 자신이 클러스터 헤더가 되었음을 알리는 패킷을 브로드캐스팅하고 이를 수신한 클러스터 멤버 노드가 될 노드들은 다른 클러스터 헤더에게서 받은 패킷

의 RSSI를 각각 비교하여 가장 가까운 클러스터 헤더의 클러스터에 참여하여 클러스터 멤버 노드가 되어 클러스터를 형성하게 된다.

III. 성능 평가

본 논문에서 제안하는 클러스터 헤더 선출 기법의 성능을 실험하기 위해 OMNet++ 시뮬레이터 프로그램을 사용하였다. 해당 프로그램을 통해 실제 마이크로 그리드와 유사한 환경에서 STO 기반 지그비 트리 토폴로지 컨트롤 알고리즘을 이용한 네트워크를 구현하였다. OMNet++는 C++기반의 라이브러리와 프레임 워크로 구성되어 있으며 주로 네트워크를 구축하여 시뮬레이션을 진행할 수 있는 프로그램이다. 실험 평가를 위해 사용된 프레임 워크는 inet을 기반하여 구현되었다. OMNet++는 이클립스 기반 IDE 그래픽 런타임 환경 및 기타 여러 도구를 사용자가 사용하기 쉽도록 제공하고 있다[2].

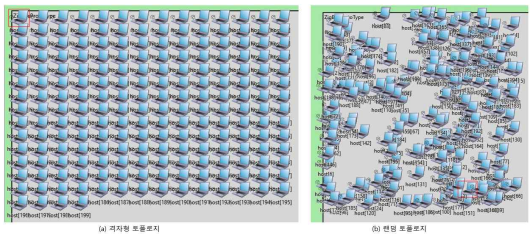

그림 1. OMNet++ 시뮬레이터 프로그램의 실험 환경 (a) 격자형 토폴로지 (b) 랜덤형 토폴로지

그림 1과 같이 노드가 배치된 환경은 200m x 200m로 총 40,000 제곱미터 환경에서 총 200개의 노드를 (a) 격자형 및 (b) 랜덤으로 배치하여 실험을 진행하였으며 노드를 격자형으로 배치했을 때와 임의로 배치했을 때의 실험 결과 모두 거의 똑같았다. 또한 각 노드가 데이터를 송수신 할 때 사용하는 패킷의 사이즈를 1kbps, 10kbps, 50kbps로 설정하여 데이터 속도가 끼치는 영향의 정도 또한 실험하였다.

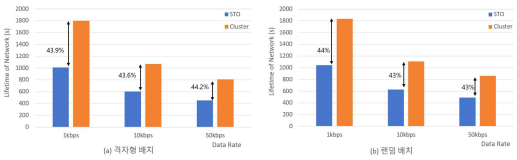

그림 2. 네트워크 구성 후 첫 번째 방전 노드가 발생하는 시간 (a) 격자형 토폴로지 (b) 랜덤형 토폴로지

첫 번째 실험은 기존의 STO 기반 지그비 트리 토폴로지 컨트롤 알고리즘을 이용한 네트워크 구성 기법과 클러스터 헤더 선출 기법을 도입했을 때, 배터리 잔량이 0이 되는 노드가 발생하는 시간을 측정하였다. 그림 1과 같이 데이터 속도가 높을수록 노드가 소모하는 전력의 양도 비례하여 커지므로 STO 알고리즘과 클러스터 헤더 선출 기법 모두 50kbps 실험에서 빠른 배터리 소모 결과를 나타내었다. 하지만 3가지 실험 모두 클러스터 헤더 선출 기법을 도입했을 때가 기존의 STO 알고리즘 보다 약 40% 정도의 효율적인 결과를 나타내었다. 시간이 지나도 부하가 집중되는 부모 노드의 적절한 교체가 없는 STO 알고리즘에서는 자식 노드가 데이터를 많이 보낼수록 혹은 데이터 속도가 빠를수록 중계해줘야 할 데이터가 많다는 의미가 되므로 그만큼의 배터리 소모가 커지며 배터리 잔량이 0이된 부모 노드가 네트워크에서 이탈하게 되면 이는 곧 네트워크 전체의 붕괴를 뜻한다. 이와 반대로 클러스터 헤더 선출 기법을 도입하면 부모 노드가 적절한 시기에 교체되므로 하나의 부모 노드에 집중되는 부하를 다른 노드들에게 분배하는 것이 가능하다. 이는 결국 기존의 STO 알고리즘 보다 클러스터 헤더 선출 기법을 도입한 네트워크가 훨씬 더 긴 네트워크 수명을 갖는 것이 가능하다.

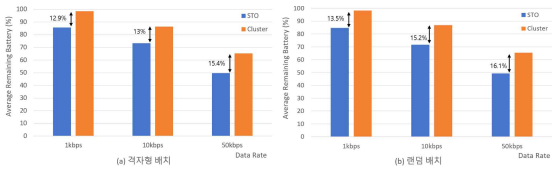

그림 3. 네트워크 구성 후 동일한 시간이 지났을 때 모든 노드 배터리 잔량의 평균 (a) 격자형 토폴로지 (b) 랜덤형 토폴로지

두 번째 실험은 네트워크 구성 후 STO 알고리즘과 클러스터 헤더 선출 기법을 도입했을 때 동일한 시간이 지났을 때의 모든 노드의 배터리 잔량 평균을 나타낸 것이다. 네트워크 동작 시간은 400초로 3가지 실험 모두 동일하다. 1kbps 실험에서 STO 알고리즘은 첫 번째로 죽는 노드가 비교적 빠르게 발생했지만 남은 노드들의 배터리 잔량 평균은 클러스터 헤더 선출 기법을 도입했을 때와 약 10% 정도의 차이만을 보였다. 10kbps와 50kbps 실험에서도 10%를 웃도는 비슷한 결과가 도출되었다. 네트워크 구성 후 동일한 시간이 지났을 때 모든 노드의 배터리 잔량 평균이 더 높다는 것은 그만큼 네트워크의 수명이 STO 알고리즘 보다 클러스터 헤더 선출 기법을 도입한 네트워크가 더 길다는 것을 의미한다.

Ⅳ. 결론

본 논문에서는 기하급수적으로 증가하는 노드의 수에 따라 발생하는 센서 네트워크의 문제점인 네트워크 수명 연장을 위한 해결책을 제시하고 있다. 기존의 STO 기반 지그비 트리 토폴로지 컨트롤 알고리즘은 네트워크에 대한 확장성만 고려하여 전력 소비에 대한 방안은 전무했다. 이에 따라 본 논문에서 제안한 클러스터 헤더 선출 기법을 도입하여 네트워크의 전체적인 전력 소모를 줄임으로써 네트워크의 수명을 기존 STO 알고리즘보다 약 40% 증가시킬 수 있었으며 평균 배터리 잔량을 STO 알고리즘보다 약 10% 향상시킬 수 있었다.

Acknowledgement

본 연구는 한국전력공사의 2016년 선정 기초연구 개발과제 연구비에 의해 지원되었음 (과제번호: R17XA05-69)

References

[1] Heonguk Lee, “ZigBee Tree-Based Scalable Topology Organization Algorithm for Microgrid Facility Management”, The Journal of Korean Institute of Communications and Information Sciences, pp.1446-1460.

[2] A. Varga and R.Hornig, “An Overview of the OMNeT++ simulation environment”, Simutolls 2008, Marseille, France, Mar.2008.