

링크 교환을 이용한 무선 센서 네트워크용 체인 토폴로지 : LECSEN

신 지 수[†] · 서 창 진^{††}

요 약

무선 센서 네트워크(Wireless Sensor Network : WSN)에서 한정된 에너지를 가진 센서 노드의 동작 기간을 연장하기 위해서 LEACH와 PEGASIS, PEDAP 등의 대표적인 라우팅 방식이 제안되었다. 이들은 데이터가 완전 퓨전(perfect fusion)되는 환경에서 주기적으로 데이터를 수집하여 한 노드로 전송하는 convergecast 라우팅 방식을 사용한다. 그러나 convergecast와 에너지 분배를 동시에 다루는 과정에서 토폴로지에 관한 특성과 한계에 대한 연구는 거의 이루어지지 않았다.

이 논문은 한 번의 convergecast에 소비되는 총에너지의 관점에서 토폴로지를 다음과 같이 연구하였다. 우리는 주요 라우팅 토폴로지로서 최소 스패닝 체인(Minimum Spanning Chain : MSC)과 최소 스패닝 트리(Minimum Spanning Tree : MST), PEGASIS 체인, 제안하는 LECSEN 체인을 소개하거나 정의하였다. 우리는 MSC를 선형 프로그래밍(LP) 방식으로 풀었으며, MSC나 MST에 준하는 토폴로지를 만들기 위해서 LECSEN 체인을 제안하였다. Monte Carlo 방식의 시뮬레이션을 통해 토폴로지의 전체 길이와 각 링크 길이의 분포를 분석한 결과, 대부분의 WSN 환경에서 LECSEN은 MST에 필적할 만큼 에너지를 적게 소모하고, 각 센서 노드의 에너지 소비가 매우 균등하였다. 그러므로 우리는 LECSEN 체인이 WSN 라우팅에서 매우 유용하다는 사실을 확인하였다.

키워드: WSN, LECSEN, 링크 교환, 최소 스패닝 체인(MSC), 체인 토폴로지, Kruskal 알고리즘, Prim 알고리즘, PEGASIS, PEDAP, 선형 프로그래밍, 외판원의 최소동선 문제(TSP)

LECSEN : Link Exchanged Chain in Sensor Networks

Jisoo Shin[†] · Changjin Suh^{††}

ABSTRACT

In WSN(Wireless Sensor Network) many routing algorithms such as LEACH, PEGASIS and PEDEP consisting of sensor nodes with limited energy have been proposed to extend WSN lifetime. Under the assumption of perfect fusion, these algorithms used convergecast that periodically collects sensed data from all sensor nodes to a base station. But because these schemes studied less energy consumption for a convergecast as well as fairly energy consumption altogether, the minimum energy consumption for a convergecast was not focused enough nor how topology influences to energy consumption.

This paper deals with routing topology and energy consumption for a single convergecast in the following ways. We chose major WSN topology as MSC(Minimum Spanning Chain)s, MSTs, PEGASIS chains and proposed LECSEN chains. We solved the MSC length by Linear Programming(LP) and propose the LECSEN chain to compete with MST and MSC. As a result of simulation by Monte Carlo method for calculation of the topology length and standard deviation of link length, we learned that LECSEN is competitive with MST in terms of total energy consumption and shows the best with the view of even energy consumption at the sensor nodes. Thus, we concluded LECSEN is a very useful routing topology in WSN.

Key Words: WSN, LECSEN, Link Exchange, Minimum Spanning Chain, Chain topology, Kruskal algorithm, Prim algorithm, PEGASIS, PEDAP, Linear Programming, Travelling Salesman's Problem

1. 서 론

WSN은 컴퓨팅 능력과 무선통신 능력을 갖춘 센서 노드를 자연환경이나 오염지역 등에 산발적으로 설치하여 센서 노드가 수집한 정보를 센서 노드로 구성된 자율적인 망을 통하여 멀티 홉으로 원격지로 전송하여 센서 필드를 관리할

※ 본 연구는 숭실대학교 교내연구비 지원으로 이루어졌음.
† 준 회 원 : 숭실대학교 컴퓨터학과 박사과정
†† 종신회원 : 숭실대학교 컴퓨터학부 교수
논문접수 : 2008년 3월 3일
수 정 일 : 1차 2008년 5월 26일, 2차 2008년 6월 6일
심사완료 : 2008년 6월 11일

수 있는 네트워크이다.[1-7]

많은 WSN 환경에서 센서 노드는 재충전되지 않는 한정된 에너지를 가진다.[2,3,4] 이러한 환경에서 WSN의 동작 시간을 연장하려면 라우팅에 소요되는 에너지를 줄여야 하며 모든 노드에서 균형있게 에너지를 소모해야 한다. 에너지 균등 소모는 WSN을 구성하는 노드의 수가 감소하여 WSN의 감지능력이 떨어지는 단점을 예방하기 위해서 필요하다.

대표적인 유형의 WSN인 convergecast는 라운드라고 부르는 일정 주기마다 모든 센서 노드가 측정된 자료를 특정 노드인 기지 노드(Base Station : BS)로 전달한다. 센서 노드는 송신 에너지를 줄이기 위해서 수신 노드가 간신히 수신할 만큼의 최소 출력으로 데이터를 전송하며, 자신이 수신한 모든 데이터와 자신이 측정한 정보를 합하여 인접 노드로 전송한다.[2,3,4] 만일 라우팅이 효율적으로 이루어진다면 모든 노드는 자신의 데이터를 라운드 당 한 번씩 전송한다. 이러한 성질을 라우팅 경로 측면에서 보면 라우팅 경로는 모든 센서 노드를 포괄하는 스패닝 트리임을 알 수 있다. 또한 만일 데이터가 100% 축약될 수 있다면, 다시 말해서 완전 푸전되면, 모든 센서 노드는 동일한 길이의 패킷을 전송한다.[2,3,4,6,7] 이 논문은 완전 푸전하는 convergecast를 가정한다.

WSN 환경에서 우리는 최소 스패닝 트리(Minimum Spanning Tree : MST) 혹은 체인을 주요 토폴로지로서 사용한다. 체인은 일련의 노드를 한 줄로 연결한 토폴로지이다. 우리는 한 노드에 연결된 링크의 개수를 노드 차수(degree)라고 부른다. 체인의 양끝 노드는 노드 차수가 1이고 중간 노드는 2이므로 스패닝 체인은 모든 노드에서의 최대허용 노드 차수가 2인 스패닝 트리이다. 이하에서는 스패닝 체인과 스패닝 트리를 각각 체인과 트리로 약술하여 사용한다.

체인을 구성하는 모든 링크 길이의 합을 체인의 길이라고 부르며 최소 길이를 가진 체인을 최단 체인(Minimum Spanning Chain : MSC)이라 한다. MST는 에너지를 최소로 사용하는 토폴로지라는 점에서 중요하며, MSC는 MST에 비해서 총 에너지를 추가적으로 사용하지만 각 센서 노드에서 에너지를 균등하게 소모한다는 점에서 주목받고 있다. MSC는 외판원의 최소 동선 문제(Travelling Salesman's Problem : TSP)와 매우 유사하다. 즉 MSC의 양끝 노드를 연결하면 링이 되는데 TSP는 최단 링을 찾는 문제이다. MSC에서 양 끝단 노드를 잇는 링크의 길이를 0으로 가정한다면 MSC와 TSP는 동일하므로 MSC 문제는 TSP와 마찬가지로 NP-complete임을 쉽게 증명할 수 있다. 따라서 센서 노드의 수가 적정수준 이상일 경우에는 MSC를 찾기가 어렵다. 우리는 센서 노드의 수가 적을 때 MSC를 계산하는 방법과 센서 노드의 수가 많을 때 MSC를 추정하는 방법을 부록에서 소개한다.

체인 토폴로지의 유용성은 유명한 WSN 방식인 PEGASIS에서 찾을 수 있다. PEGASIS는 PEGASIS에서 정의한 PEGASIS 체인을 사용하며 매우 단순하게 동작한다.

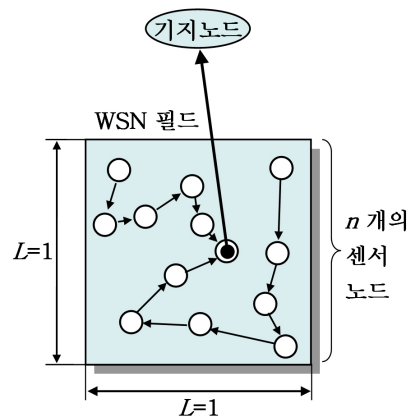
PEGASIS는 이전에 발표된 2단 트리를 사용하는 LEACH에 비해서 매우 우수한 성능을 보여주었으며, 이후 발표된 트리를 라우팅 경로로 사용하는 PEDAP에 비해서도 유사한 성능을 보여주었다.[2,4] LECSEN은 LEACH가 사용한 Prim 체인방식 대신에 더 우수한 Kruskal 체인을 생성하며 우리가 개발한 링크 교환 기법을 적용해서 추가로 체인의 길이를 줄여서 현격히 개선된 체인을 생성한다. 조약한 수준의 PEGASIS 체인 대신에 우리가 제안한 LECSEN 체인을 사용한다면 센서 네트워크의 수명을 많이 연장할 수 있다.

본 논문은 PEGASIS 체인과 MSC, MST, LECSEN 체인에 대한 에너지 소비 특성을 연구하였다. 우리는 토폴로지의 총 길이와 토폴로지를 구성하는 링크 길이의 표준편차를 성능평가 요소로 선정하였다. 총 길이는 한 라운드에서 소요되는 총 에너지에 비례하므로 WSN 수명을 가늠하며, 링크 길이의 표준편차는 각 센서 노드가 얼마나 균일하게 에너지를 소모하는가를 나타낸다.

본 논문의 구성은 다음과 같다. 2장에서는 WSN을 모델링하고 논문의 목표를 제시한다. 3장에서는 기존 연구에서 사용된 토폴로지인 MST와 PEGASIS 체인을 소개하고, 4장에서는 LECSEN 체인을 설명한다. 5장에서는 소개한 네 가지 토폴로지를 시뮬레이션을 통해 링크 길이와 표준편차를 구하고 결과를 분석한 후 6장에서 결론을 맺는다. 부록에는 MSC의 길이를 계산하고 추정하는 방법을 언급한다.

2. WSN 모델링

이 절에서는 우리가 고려하는 (그림 1)과 같은 형상의 WSN을 소개한다. 가로, 세로가 모두 L 인 정사각형의 센서 필드 안에 원으로 표시된 13개의 센서 노드가 라우팅 경로를 체인으로 형성한 모습을 보여준다. 우리는 편의상 L 을 기본단위 1로 고정한다. BS는 센서 필드 밖에 위치하며, 센서 필드 내의 특정한 - (그림 1)에서는 검은 색으로 표시된 - 노드가 센서 필드 내의 모든 센서 노드가 발생한 데이터를 수집한 후에 BS로 최종 전송하여 한 라운드를 완료한다.[2,6,7]



(그림 1) WSN 환경

우리는 WSN을 다음과 같이 가정한다. 모든 센서 노드는 움직이지 않고 재충전되지 않는 에너지를 사용한다. 센서 노드는 BS가 계산하여 공표한 라우팅 경로를 따라서 측정값을 루트 노드로 전달한다. 전송 출력은 수신 노드와의 거리의 제곱에 정비례하며, 데이터는 완전 퓨전되어서 모든 센서 노드는 한 라운드에 한 번만 동일한 길이의 패킷을 전송한다. 본 논문은 언급한 가정 하에서 한 라운드 동안에 소비하는 총 전송 에너지의 합과 소요 에너지의 분산 정도를 노드 단위로 계산하고자 한다. 전송 에너지는 경로 손실 모델에 따라서 거리의 β 제곱에 비례한 항과 상수항의 합으로 구성된다. 본 논문은 스패닝 트리를 따라서 전송되는 총량을 비교하는데 모든 스패닝 트리에서 상수항의 합이 동일하기 때문에 상수항은 논의에서 제외한다. 또한 β 의 값으로서는 비교 논문에서 사용했던 값인 2를 동일하게 사용하고 있다. 그 결과 두 노드 i 와 j 를 연결하는 무선링크를 l_{ij} 라고 하고 노드 i 의 직교좌표를 (x_i, y_i) 라 하면 l_{ij} 의 링크값 C_{ij} 는 식 (1)과 같이 정한다.

$$C_{ij} = d_{ij}^2 = (x_i - x_j)^2 + (y_i - y_j)^2. \quad (1)$$

우리는 아래와 같은 논리를 통하여 문제를 추가적으로 단순화한다. 한 라운드의 총 전송 에너지는 최종 전달 노드로 측정값을 모으는 과정에서 소요되는 전송 에너지와 최종 전달 노드가 BS로 전송하는 전송 에너지의 합으로 구성된다. 후자는 BS와 센서 필드 사이의 거리, 센서 필드 넓이, 최종 노드 선정 방법과 같이 시스템상의 여러 문제를 복합적으로 고려해야 하므로 센서 필드 내의 토폴로지 연구에 집중하기 위해서 본 논문에서 제외하였다. 또한 전송 에너지에는 거리와 무관한 수신 에너지 항이 존재한다[2,3,4]. 수신 에너지는 모든 노드가 동일한 길이의 패킷을 한 번씩 전송하는 본 논문의 경우에는 토폴로지와 상관없이 모든 노드에서 동일한 값을 가지므로 이를 제외하였다.

본 논문은 다음과 같이 단순화된다. (그림 1)과 같은 WSN에서 모든 무선 링크의 링크값을 식 (1)에 의해서 파악한 상태에서 주요 토폴로지에 대해서 토폴로지의 총 길이와 각 노드의 소요 에너지 정도를 알아본다. 우리는 PEGASIS 체인과 본 논문에서 제안하는 LECSEN 체인, MST와 MSC를 주요 토폴로지로서 선정하였다.

3. 기존의 WSN 토폴로지

3.1 MST

PEDAP은 라우팅 토폴로지로서 MST를 사용한다.[4] MST는 스패닝 트리 중에서 총 길이가 가장 짧은 트리로서 Prim 방식과 Kruskal 방식에 의해서 구한다. 이 두 방식의 차이는 트리를 만드는 중간 과정에서 발생한다. Prim의 경우는 중간에 만들어진 부분-트리가 항상 하나로 연결되지만 Kruskal은 도중에 여러 개의 부분-트리를 허용하여서 소위 숲(forest)을 만든다. 이 두 방법은 체인을 만들 때에도 부분적으로 사용된다.

3.2 PEGASIS 체인

PEGASIS 체인은 Prim의 MST 방식에 최대허용 노드 차수($=2$) 조건을 추가하여 체인을 생성한다[2]. Prim의 MST 방식에서는 시작할 때에 루트 노드가 필요하다. PEGASIS 체인에서는 BS와 가장 먼 노드를 루트 노드로 정하였지만 본 논문은 BS를 정하지 않으므로 임의로 루트 노드를 선정한다.

PEGASIS 체인은 다음과 같은 방식으로 형성된다. 루트 노드를 $M(1)$ 노드라고 선정한다. $M(1)$ 과 가장 가까운 노드를 $M(2)$ 로 선정한다. $M(k)$ 가 선정된 상황에서 우리는 $M(k+1)$ 을 아직 선정되지 않은 노드 중에서 $M(k)$ 와 가장 가까운 노드로 선정한다. 우리는 $M(i)$ 을 선정할 때에 링크 $(M(i-1), M(i))$ 도 함께 선정한다.[2]

4. 제안하는 LECSEN 체인 토폴로지

4장에서는 우리가 제안하는 LECSEN 체인을 소개한다. LECSEN 체인은 두 단계에 걸쳐서 형성된다. 첫째 단계에서는 Kruskal의 MST 방식을 변형하여 초기 체인을 생성한다. 첫째 단계에서 생성한 체인은 체인 토폴로지라는 제한 때문에 최적화된 해가 아니어서 길이를 줄일 수 있다. 둘째 단계에서 링크 교환 방법으로 체인의 길이를 줄인다.

4.1 단계 1 : 초기 체인 생성

단계 1에서는 Kruskal의 MST 알고리즘에 최대허용 노드 차수($=2$)의 제한을 추가하여 초기 체인을 생성한다. 생성 알고리즘은 (그림 2)에 제시하였다. 제안하는 Kruskal 체인 알고리즘은 식 (1)을 사용하여 모든 링크의 링크값을 계산한다. 그 후 최대 허용 노드 차수가 2가 아닌 노드를 연결하는 링크 중에서 가장 짧은 링크를 체인에 포함시킨다. 즉, 알고리즘이 진행되는 동안 부분-체인이 형성되며 다음번 추가할 링크는 노드의 차수가 최대허용 노드 차수 미만인 노드를 잇는 링크 중에서 가장 짧은 링크이다. 단, 선택한 링크의 양끝 노드가 한 부분-체인의 양끝 노드인 경우는 연결하면 립을 형성하므로 제외한다. 모든 노드가 하나의 체인에 포함되면 단계 1은 종료된다. (그림 1)을 예로 설명해보자. (그림 1)에는 13 개의 센서 노드가 분포되어 있다. 제안하는 체인 알고리즘은 13×12 개의 링크 중에서 가장 짧은 링크를 선택하여 부분-체인을 형성한다. 그 다음 링크의 선택은 노드의 차수가 2 이하인 노드를 연결하는 최소 길이의 링크이며 이 때 립을 형성하는 링크는 선택하지 않는다.

Kruskal 체인방식은 Prim 체인방식보다 우수하다. 그 이유는 Prim은 중간 생성과정에서도 반드시 서로 연결된 부분-트리를 만들지만 Kruskal은 이러한 제한이 없어서 여러 개의 부분-트리를 생성과정에서 허용한다. 연결성에 대한 제한이 없어짐에 따라서 Kruskal 체인방식은 중간과정에서 링크를 선정할 때 더 많은 링크를 대상으로 링크를 선정할 수 있기 때문에 Prim 체인보다 짧은 체인을 만든다. 우리는 이 사실을 시뮬레이션을 통해 확인하였다.

```

main( )
{
  /* Initialization */
  A = {1, 2, ..., Nd} /* Set of nodes to constitute the chain */
  SH = ∅ /* Set of links L(i,j) to constitute the chain */
  Assign C(i)[j] = Cij /* using Eq. (1) */
  for (all i∈A) node[i].peer_leaf=i /* peer leaf node Id in the sub-chain */

  /* Chain Buildup */
  Repeat until A contains two elements { /* They are two leaf nodes in the chain. */
    Find i and j that minimize C(i)[j]
      such that ((i,j∈A) ∩ (i≠j) ∩ (node[i].peer_leaf≠j));
    chain_buildup(i,j);
  } /* SH contains all links that constitute the chain. */

chain_buildup(i,j) /* Include link (i,j) to the chain and update A and SH. */
{
  Put (i,j) to SH.
  node[node[i].peer_leaf].peer_leaf = node[j].peer_leaf;
  node[node[j].peer_leaf].peer_leaf = node[i].peer_leaf;
  If (node[i].peer_leaf≠i) remove i from A.
  If (node[j].peer_leaf≠i) remove j from A.
}
}

```

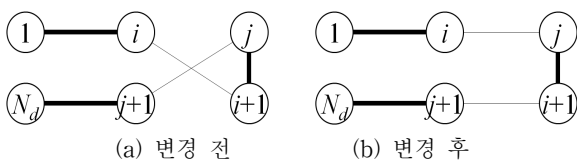
(그림 2) 제한적인 Kruskal 체인 알고리즘의 pseudo 코드.

4.2 단계 2 : 링크 교환 기법

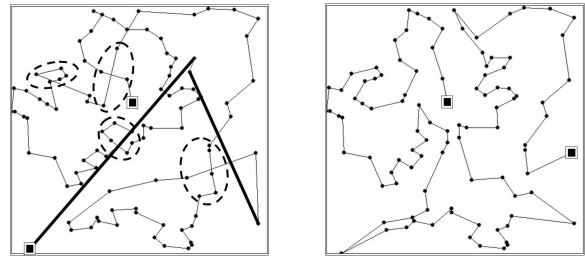
단계 2에서는 현재 체인을 구성하는 두 개의 링크를 새로운 두 개의 링크로 교환하여 (그림 3)과 같이 더 짧은 체인을 만든다. 이 작업을 링크 교환이라고 칭한다. (그림 3)에서 노드는 체인에서의 노드 순서에 따라 1부터 n까지의 일련번호로 표시하였다. 굵은 실선은 여러 개의 링크로 연결된 서브-체인이며 가는 실선은 단일 링크를 나타낸다. (그림 3)에서 보듯이 링크 교환이 발생하면 i+1부터 j까지의 서브-체인은 체인의 순서가 뒤바뀐다. 현재 체인을 구성하는 두 링크인 $l_{i,i+1}$ 과 $l_{j,j+1}$ 을 $l_{i,j}$ 와 $l_{i+1,j+1}$ 로 교환하려면

$$C_{i,i+1} + C_{j,j+1} > C_{i,j} + C_{i+1,j+1} \quad (2)$$

의 부등관계를 만족해야 한다. 식 (2)의 좌항은 삭제할 두 링크값의 합이며, 우항은 새로 대체할 두 링크값의 합이다. 식 (2)의 조건을 만족하면 체인은 새로운 두 링크를 대체하며 그 결과 새 체인의 총 길이는 축소된다.



(그림 3) 링크 교환 기법



(a) PEGASIS 체인 (b) LECSN 체인
(그림 4) PEGASIS 체인과 LECSN 체인의 비교.

링크 교환 기법은 두 개의 링크를 선정한 상태에서 수행된다. 우리는 이 중 긴 링크를 관심 링크라고 부른다. 관심 링크와 다른 한 개의 링크를 고르는 방법은 아래와 같다. 모든 링크를 링크값의 역순으로 정렬한 후에 긴 링크부터 하나씩 관심 링크로 고정하고 관심 링크보다 작은 링크에 대해서 링크값의 역순으로 선정하여 식 (2)의 관계를 확인한다. 만일 관심 링크에 대해서 식 (2)를 만족하는 링크를 찾을 수 없다면 정렬된 순서에서 현재의 관심 링크의 다음 번 링크를 새로운 관심 링크로 선정한다. 만일 관심 링크에 대해서 식 (2)의 조건을 두 개 이상의 링크가 만족한다면 링크 교환으로 길이 축소 효과가 더 큰 경우를 선택하여 링크 교환을 수행하고 링크를 재 정렬한다. 링크 교환이 일어난 다음에는 이전 관심 링크보다 길이가 짧은 링크 중에서

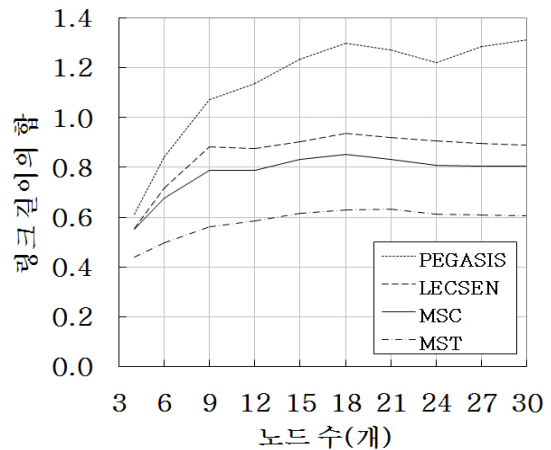
길이가 가장 긴 링크를 새로운 관심 링크로 선정한다. 이러한 동작은 가장 작은 링크를 관심 링크로 지정할 때까지의 반복된다. 가장 긴 링크를 관심 링크로 선정할 때부터 가장 짧은 링크를 관심 링크로 선정할 때까지의 기간을 한 주기라고 칭할 때에 한 주기 내내 링크 교환이 발생하지 않으면 동작이 종료된다.

(그림 4)는 100개의 노드로 구성된 WSN에서 PEGASIS 체인과 본 논문이 제안하는 LECSEN 체인이 얼마나 상이하게 형성되는지를 보여주는 대표적인 예이다. (그림 4)에서 체인의 양끝 노드는 큰 크기의 정사각형으로 표시하였다. 우리는 링크 교환 기법을 통해 PEGASIS의 두 가지 문제를 해결한다. 첫째, PEGASIS와 LECSEN 체인은 모두 체인을 생성할 때 탐욕적인 방법으로 링크를 선택하기 때문에 마지막에 형성되는 링크들이 매우 큰 링크값을 가지는 경우가 빈번하며 이는 우리가 가정하는 환경에서 에너지 소비에 큰 영향을 미친다. 그러나 LECSEN은 링크 교환을 수행하여 이를 해결한다. (그림 4.(a))에서는 굵은 선으로 표시된 두 개의 링크가 다른 링크에 비해서 매우 길이가 긴데 이들은 가장 마지막과 세 번째 마지막으로 선정된 링크이다. PEGASIS의 체인은 추가적인 작업 없이 형성된 체인을 그대로 사용하기 때문에 길이가 긴 링크에 연결된 노드들은 에너지를 극심하게 소비한다. 둘째, 비록 MSC에서 링크가 가끔 교차되기도 하지만 이차원 평면에서 체인의 교차는 체인의 성능을 나타내는 척도로 활용할 수 있다. (그림 4.(a))의 PEGASIS 체인에서 굵은 실선으로 표시된 두 개의 링크는 많은 짧은 링크들과 교차하며 또한 네 개의 점선으로 표시한 원형에서도 서로 교차하는 링크 쌍이 있다. 교차된 링크들의 체인의 순서를 적절히 바꾸면 대부분 체인의 길이를 크게 개선할 수 있다. 반면에 링크 교환 기법을 통한 (그림 4.(b))의 LECSEN 체인은 교차하는 링크 쌍이 없으며 육안으로는 개선할 여지를 찾기 어려울 정도로 정교하다.

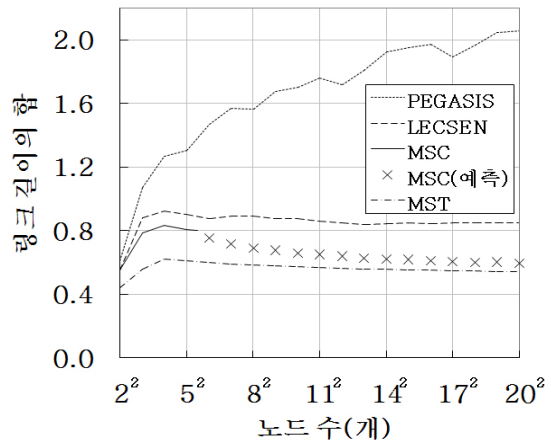
5. 성능 평가

본 절에서는 3장에서 서술한 PEGASIS 체인과 MST, 4장의 LECSEN 체인, 부록에서 LP를 사용하여 해를 구한 MSC를 다양한 노드 범위에 대해서 길이와 표준편차를 비교분석한다. 이를 위해서 주어진 n 에 대해서 임의로 분포한 100 개의 노드 셋을 선정하여 Monte Carlo 방식에 의한 시뮬레이션을 시도하였다. LP의 해는 ILOG CPLEX 9.0 시스템을 사용하여 풀었다.

첫째로 (그림 5)에서 스페닝 트리/체인의 총길이를 살펴보자. MSC는 노드 30까지 구하였으며, 그 이후는 추정치 $\hat{W}(k^2)$ ($k=6,7,\dots,20$)를 사용하여 (그림 5.(b))에서 'x'로 표시하였다. (그림 5.(a))는 노드 규모가 작은 경우의 결과값이고 (그림 5.(b))는 대규모인 경우일 때이다. (그림 5)에서 모든 노드 규모 n 값에서 트리/체인의 길이는 항상 MST가 가장 짧고 MSC, LECSEN 체인, PEGASIS 체인의 순서를 유지한다. 트리는 체인에 비해서 최대허용 노드 차수를 제한받

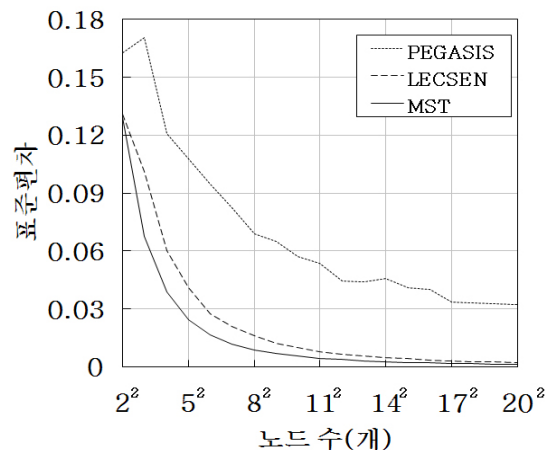


(a) $n \leq 30$ 일 때



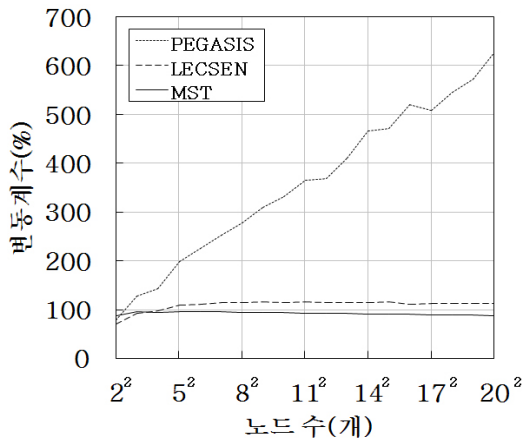
(b) $n \leq 400$ 일 때

(그림 5) 여러 토폴로지의 노드 수(n)에 따른 총 길이 비교



(그림 6) 여러 토폴로지의 링크 길이에 따른 표준편차 비교

지 않으므로 길이가 짧다. PEGASIS 외의 세 토폴로지는 n 이 18 또는 21에 이르기까지 총길이가 증가하다가 그 이후 구간에서는 약간씩 감소한다. MSC와 LECSEN 체인의 길이는 n 이 증가함에 따라서 MST 길이와 거의 일정한 비율을 유지하는 반면에 PEGASIS 체인 길이는 MST와 상대적으로 볼 때에 꾸준히 증가한다.



(그림 7) 여러 토폴로지의 링크 길이에 따른 변동계수 비교

둘째로 스패닝 트리/체인을 구성하는 링크들 사이에서 길이의 차이가 얼마나 발생하는가를 살펴보기 위해서 (그림 6)에서 구성 링크의 표준편차를, (그림 7)에서 변동계수를 각각 알아보았다. 추정치를 이용한 MSC의 경우는 표준편차를 계산할 수 없으므로 MSC는 이곳에서는 제외한다. (그림 6)의 표준편차는 모든 경우에서 노드 n 이 증가할수록 줄어든다. 이는 링크간의 길이의 정도가 균일해졌다기보다는 노드 n 이 증가할수록 링크 길이가 짧아져서 길이의 편차도 따라서 작아졌기 때문이다.

노드의 밀도를 유지한 채 노드 수와 센서 필드의 면적을 늘린 경우의 결과를 확인하기 위해서 우리는 변동계수를 계산하여 (그림 7)을 그렸다. 변동계수란 표준편차를 링크 길이의 평균으로 나눈 값이다. 그 결과 PEGASIS는 노드 수가 증가함에 따라서 링크 길이 분포가 상대적으로 분산됨을 알 수 있으며, LECSSEN과 MST는 노드 수와 관계없이 상대적으로 링크 분포가 일정하며 다만 LECSSEN이 MST에 비해서 20-30% 정도 더 넓게 분산되어 있다.

아래에서는 시뮬레이션 결과가 WSN에서 어떻게 활용되는가를 언급한다. 우선, 토폴로지의 총 길이는 모든 노드에서 발생한 실측 데이터를 전달하기 위해서 소비하는 에너지에 비례한다. 단지 거리에 비례하지 않는 고정항을 추가적으로 고려한다면 (그림 5)에서 곡선들은 동일한 값만큼 상향조정해야 한다. 또한 각 노드의 소요 에너지 분포는 (그림 6) 혹은 (그림 7)에서 각 노드의 노드 차수가 추가로 고려되어야 한다. 즉 인근 노드가 많은 노드는 적은 노드에 비해서 수신할 횟수가 많으므로 인근 노드 수에 따라서 수신 에너지가 추가되기 때문이다. 특히 센서필드가 작거나 센서노드를 땀곡하게 설치할 경우에는 송신 에너지가 미미하기 때문에 노드 차수가 소비 에너지에 지대한 영향을 미친다.

지금까지 언급한 상황을 정리하면 LECSSEN 체인은 총 에너지 소모 측면에서는 MSC와 MST에 필적할 수준이며 각 노드별 에너지 사용의 균등정도에 대해서는 MST보다 우수하다. 또한 모든 경우 LECSSEN 체인은 PEGASIS 체인과는 현격하게 우수하다. 이를 종합해 보면 WSN 토폴로지로서 이

상적인 토폴로지는 MSC이며, 실제적으로는 MST와 LECSSEN이 권고된다.

6. 결 론

WSN과 같이 에너지의 제약을 받는 환경에서는 에너지를 고르게 소모하여 모든 센서 노드가 오랜 시간 동안 동작하는 것이 중요하다. 이러한 동작을 목표로 본 논문은 두 가지 연구에 치중하였다. 첫째는 발표된 WSN 라우팅 방식에서 사용하는 주요 토폴로지를 비교하는 것이며 둘째는 새로운 뛰어난 토폴로지를 찾는 일이다.

첫 번째 목표를 위해서 우리는 MSC와 MST, PEGASIS 체인을 선정하였으며, 토폴로지의 총 길이와 토폴로지를 이루는 링크 길이의 표준 편차를 Monte Carlo 방법을 이용한 시뮬레이션으로 측정하였다.

두 번째 목표를 위해서 LECSSEN이란 토폴로지 생성 알고리즘을 제안하였다. LECSSEN 체인은 초기 체인을 정교하게 만든 후 링크 교환 기법을 통해서 추가적으로 길이를 줄인다. 시뮬레이션 결과 LECSSEN 체인은 PEGASIS 체인보다 월등하게 우수하다. 또한 LECSSEN 체인의 길이는 우리가 확인한 모든 노드 범위에 대해서 MST 길이의 두 배 내로 수렴함을 밝혔다. 또한 LECSSEN 체인은 대부분의 WSN 환경에서 각 노드에서의 에너지 소모 분포가 매우 고르다는 점을 알게 되었다.

이러한 근거로 우리는 LECSSEN 체인이 WSN에서 매우 유용한 토폴로지임을 확인하였다. 향후에 우리는 LECSSEN을 사용하여 기존의 모든 라우팅 방식보다 평균 수명과 초기 노드 사망 시점이 증가된 개선된 라우팅 방식을 완성할 계획이다.

감사의 글

본 연구를 위해 LP package 사용에 도움을 주신 한양대학교의 '이동호' 교수님과 연구실 학생분들께 감사드립니다.

참 고 문 헌

- [1] Y.Li, T.Newe, "Wireless Sensor Networks Selection Of Routing Protocol For Applications," Australian Telecommunication Networks and Application Conference, Melbourne, December 2006.
- [2] S. Lindsey and C. S. Raghavendra, "Pegasis: Power-efficient gathering in sensor information systems," in IEEE Aerospace Conference, March 2002.
- [3] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-Efficient communication protocol for wireless micro sensor networks," in 33rd Annual Hawaii International Conference on System Sciences, pp.3005-3014, 2000.
- [4] Huseyin Ozgur Tan and Ibrahim Korpeoglu, "Power

Efficient Data Gathering and Aggregation in Wireless Sensor Networks,” SIGMOD Record, Vol.32, No.4, Dec. 2003.

- [5] Minh-Long Pham, Daeyoung Kim, Yoonmee Doh, Seong-eun Yoo, “Power Aware Chain Routing Protocol for Data Gathering in Sensor Networks,” Intelligent Sensors, Sensor Networks and Information Processing Conference, December 2004.
- [6] Nahdia Tabassum, Quazi Ehsanul Kabir Mamun, Yoshiyori Urano, “COSEN: A Chain Oriented Sensor Network for Efficient Data Collection,” Information Technology: New Generations 2006, Third International Conference, April 2006.
- [7] Kemei Du, Jie Wu, Dan Zhou, “Chain-Based Protocols for Data Broadcasting and Gathering in the Sensor Networks,” Proceedings of Parallel and Distributed Processing Symposium, April 2003.
- [8] C.E.Miller, A.W.Tucker, R.A.Zemlin, “Integer Programming Formulation of Traveling Salesman Problems,” Journal of the ACM, 7(4):326-329, October 1960.
- [9] J. B. Kruskal, “On the Shortest Spanning Subtree of a Graph and the Traveling Salesman Problem,” Proceedings of the American Mathematical Society, 7, pp.48-50, 1956.

부 록

MSC는 1장에서 소개한 바와 같이 NP-complete 문제이므로 적정범위까지만 해를 구할 수 있다. 우리는 부록 A에서 LP를 이용하여 30-노드까지의 범위에서 MSC를 찾았다. 부록 B에서는 30-노드 이상의 규모에서 추정치를 얻는 방법을 제시한다.

부록 A. LP에 의한 MSC 길이 계산

이곳에서는 n 개의 센서 노드의 좌표가 주어졌을 때 MSC를 구하는 문제를 LP를 이용하여 풀어본다. 우리는 유명한 NP-complete 문제인 TSP의 LP 풀이법들[8,9] 참조하였다. 단, PEGASIS에서는 시작 노드를 미리 정하는 관례를 고려하여 임의의 노드인 1번 노드를 시작 노드로 미리 정한 후에 계산한다.

WSN은 무선 환경이므로 임의의 두 노드를 연결할 수 있다. 부울값을 가지는 플로우 변수 f_{ij} 는 번호가 i 인 노드 바로 다음에 번호가 j 인 노드를 선택할 경우에 1이며 그 이외에는 0이다. 이는 체인을 따라서 1 단위의 플로우가 발생하여 MSC로 선정된 링크를 따라서 1의 플로우가 통과하는 것으로 이해할 수 있다. f_{ij} 가 1일 때에 우리는 노드 i 는 1의 출력 플로우가 발생하고 노드 j 는 1의 입력 플로우가 발생한다고 표현하자. 우리는 한 노드만으로 구성된 루프는 허용하지 않다. 또한 번호가 1인 노드를 시작 노드로 설정하였기 때문에 노드 1에서는 입력 플로우가 발생하지 않고 1 단위의 출력 플로우가 발생한다. 따라서

$$f_{ii} = f_{i1} = 0 \text{ for } 1 \leq i \leq n, \tag{3}$$

$$\sum_{j=2}^n f_{1j} = 1 \tag{4}$$

가 성립한다. 시작 노드가 아닌 모든 노드는 항상 1 단위의 입력 플로우가 발생하므로

$$\sum_{i=1}^n f_{ij} = 1 \text{ for } 2 \leq j \leq n \tag{5}$$

의 관계를 가진다. 마지막으로 중간 노드에서는 1 단위의 입력과 1 단위의 출력 플로우가 발생하여서 식 (6)의 합은 0이며, 최종 노드는 1 단위의 입력 플로우만 발생하여서 식 (6)의 합은 1이다. 불행하게도 우리는 최종 노드를 모르므로 최선의 조건은 식 (6)이다.

$$0 \leq \sum_{j=2}^n f_{ij} \leq 1 \text{ for } 2 \leq i \leq n. \tag{6}$$

추가적으로 노드 1을 제외한 노드들의 부분집합으로 구성되는 루프를 제거하기 위해서 우리는 부울 변수 u_i ($1 \leq i \leq n$)를 정의하여 아래와 같이 $(n-1)^2$ 개의 관계식을 추가하여야 한다. (증명은 [8]을 참조.)

$$u_i - u_j + n f_{ij} \leq n - 1 \text{ for } i, j \text{ s.t. } (i \neq j, 2 \leq i, j \leq n). \tag{7}$$

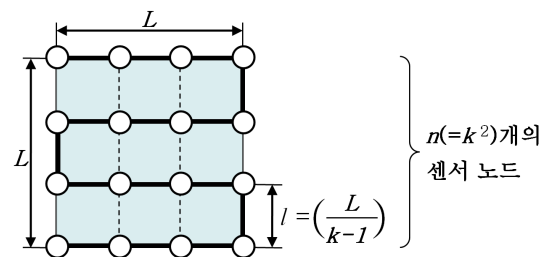
식 (3)에서 (7)까지의 관계로 우리는 노드 1에서 시작하는 모든 종류의 체인을 정의하였다. 선택된 체인의 총 길이 $W(n)$ 은 식 (8)로 표현한다.

$$W(n) = \sum_{i=1}^n \sum_{j=2}^n C_{ij} f_{ij}. \tag{8}$$

시작 노드가 정해진 n -노드 MSC의 총 길이를 구하려면 식 (3)에서 식 (7)까지의 선형 관계식을 모두 만족하고 식 (8)의 값을 최소로 하는 솔루션을 찾아야 한다. 우리는 이를 LP로 풀었다. 우리가 사용한 변수 f_{ij} 와 u_i 는 모두 부울 변수이다. 부울 변수는 LP로 풀기 어려워서 우리는 30-노드 크기까지 최단 체인을 LP 방법으로 구하였다.

부록 B. MSC의 길이 추정

본 절에서는 31 개의 노드 이상으로 구성된 WSN에서의 MSC 길이를 추정하는 방법을 설명한다. 분석을 위해서 (그림 8)과 같이 n 개의 노드를 $L \times L$ 의 센서 필드에 규칙적으로 배열시키자. 이 경우에 n 은 거듭제곱 꼴($=k^2$)로만 존재할 수 있다. (그림 8)에서 센서 필드의 경계선에 노드를 위치시



(그림 8) 노드가 규칙적으로 배열된 WSN에서 구성된 체인

켜 링크 길이를 확대한 이유는 규칙적인 배열에서는 실제의 랜덤 배열에 비해서 체인의 평균 길이가 짧은 현상을 보상하기 위해서이다.

그림 8에서 보듯이 인접한 두 노드의 직선거리는 $\left(\frac{L}{k-1}\right)$ 이며, 체인을 구성하는 링크값은 모두 $\left(\frac{L}{k-1}\right)^2$ 이다. 또한 (k^2-1) 개의 링크로 구성된 체인의 최소길이는 $\frac{(k+1)}{(k-1)}L^2$ 이다. 이 값은 식 (8) $W(n)$ 의 추정값이다. 이를 $\hat{W}(n)$ 로 표시하자. 우리는 $W(n)$ 는 $\hat{W}(n)$ 와 서로 비례관계가 있다고 가정하여 $c \cdot \hat{W}(n)$ 로 단순화 하였다. 그 결과 식 (9)을 얻었다.

$$\hat{W}(n) = c \cdot \frac{k+1}{k-1} L^2 \quad (9)$$

노드 수 n 이 25일 때 LP를 통하여 얻은 MSC 길이인 $W(25)$ 를 입력하면 c 의 값을 구할 수 있다. 우리는 c 값을 대입한 식 (9)를 이용하여 $\hat{W}(k^2)$ ($k=6,7,8,\dots$)를 유추하였다.



신 지수

e-mail : jsshin@networking.com
 2003년 숭실대학교 컴퓨터학부(학사)
 2005년 숭실대학교 대학원
 컴퓨터학과(공학석사)
 2005년~현 재 숭실대학교 대학원
 컴퓨터학과 박사과정

관심분야 : Sensor network, Carrier Ethernet, Routing theory 등



서 창진

e-mail : cjsuh@ssu.ac.kr
 1982년 서울대학교 제어계측학과(학사)
 1984년 서울대학교 대학원 제어계측학과
 (공학석사)
 1985년~1990년 한국전자통신연구원
 선임연구원

1996년 ECE(Electric and Computer Engineering), Univ. of Massachusetts at Amherst(공학박사)

1997년~현 재 숭실대학교 컴퓨터학부 교수

관심분야 : Sensor network, Carrier Ethernet, Routing theory, BGP