

센서네트워크에서 클러스터기반의 에너지 효율형 센서 스케줄링 연구

최 욱¹ · 이 용^{2*} · 정유진¹

Cluster-based Delay-adaptive Sensor Scheduling for Energy-saving in Wireless Sensor Networks

Wook Choi · Yong Lee · Yoojin Chung

ABSTRACT

Due to the application-specific nature of wireless sensor networks, the sensitivity to such a requirement as data reporting latency may vary depending on the type of applications, thus requiring application-specific algorithm and protocol design paradigms which help us to maximize energy conservation and thus the network lifetime. In this paper, we propose a novel delay-adaptive sensor scheduling scheme for energy-saving data gathering which is based on a two phase clustering (TPC). The ultimate goal is to extend the network lifetime by providing sensors with high adaptability to the application-dependent and time-varying delay requirements. The TPC requests sensors to construct two types of links: direct and relay links. The direct links are used for control and forwarding time critical sensed data. On the other hand, the relay links are used only for data forwarding based on the user delay constraints, thus allowing the sensors to opportunistically use the most energy-saving links and forming a multi-hop path. Simulation results demonstrate that cluster-based delay-adaptive data gathering strategy (CD-DGS) saves a significant amount of energy for dense sensor networks by adapting to the user delay constraints.

Key words : Energy Saving, Two Phase Clustering, Data Gathering, Scheduling, Sensor Network

요 약

다양한 응용에 적용될 수 있는 특성을 가진 무선 센서 네트워크는 적용되는 응용에 따라 데이터 리포팅 지연시간의 제한과 같이 요구사항이 다양하므로 각 응용별로 구분되는 알고리즘이나 프로토콜 설계 패러다임을 적용하여 에너지 효율을 최대화하고 네트워크의 생존기간을 최대화할 수 있어야 한다. 이 논문에서는 2단계 클러스터링(Two Phase Clustering : TPC) 방식을 이용하여 에너지 효율 데이터 수집을 제공하기 위한 새로운 알고리즘으로 지연시간 적응형 센서 스케줄링 방안을 제안한다. 이 논문의 궁극적인 목표는 센서들에게 응용 환경의 특성과 시간에 따라 변하는 특성을 갖는 지연시간에 대한 요구사항에 대하여 높은 적응성을 제공하여 네트워크의 생존기간을 늘리는 것이다. TPC 방식은 센서들이 직접 링크와 릴레이 링크의 두 가지 링크를 구성하도록 한다. 직접 링크는 제어 메시지나 시간에 민감한 센서 데이터들을 포워딩하는 데 사용된다. 릴레이 링크는 사용자의 지연시간 제한에 따라 데이터를 포워딩하는데 사용되며 이를 이용하여 센서들이 에너지-절약효과를 갖는 릴레이를 사용할 기회가 증가하도록 멀티홉 경로를 구성할 수 있도록 한다. 이 논문에서는 제안하는 CD-DGS 방식이 사용자의 지연시간 제한 요구사항에 잘 적응하여 센서 네트워크의 분포 밀도가 높은 경우에 상당한 비율의 에너지 효율을 보이는 것을 시뮬레이션 결과로 증명한다.

주요어 : 에너지 절약, 두단계 클러스터링, 데이터 수집, 스케줄링, 센서 네트워크

* 이 연구는 2009학년도 한국외국어대학교 교내학술연구비의 지원에 의하여 이루어진 것임.

2009년 5월 13일 접수, 2009년 8월 5일 채택

¹⁾ 한국외국어대학교 컴퓨터공학과

²⁾ 충주대학교 전자통신공학과

주 저 자 : 최 욱

교신저자 : 이 용

E-mail: yleehyun@cjnu.ac.kr

1. 서 론

최근 MEMS(Micro-Electro Mechanical Systems)와 단파 무선 통신 기술의 급속한 발전은 센서 네트워크가 점점 더 진화하여 실생활에 직접적인 영향을 줄 것이라고 기대된다. 무선으로 연결된 센서들이 자치적인 무선 센서 네트워크를 구성하여 물리적인 현상을 예측하고 설명하는 기본 정보를 제공하는 기반구조로서 동작한다는 의미에서 무선 센서 네트워크는 미래의 유비쿼터스 환경과 context-aware 컴퓨팅 환경을 실현하는 기술로서 고려되고 있다. 이런 센서 네트워크에 기반한 데이터 수집 시스템은 물리적으로 밀접한 통합 시스템들이 네트워크 컴퓨팅으로 처리되는 것을 의미하는 CPS(Cyber physical system)을 구현하기 위해 절대적으로 필요하다. 또한 생활환경을 보호하기 위해 효율적으로 컴퓨터 자원을 사용하는 그린 컴퓨팅의 목적에도 필요한 요소이다^{[1][2][3]}.

센서의 제한된 에너지 보유능력으로 인해 최근 무선 센서 네트워크용 프로토콜 설계에서 가장 강조되는 것은 센서 네트워크의 생존 기간을 최대화하기 위해 센서들이 에너지를 최대한 효율적으로 사용하도록 하는 것이다. 센서 네트워크는 대부분 정적이며 센서들이 밀도 높게 구성되어 있는 특징이 있다. 센서 네트워크의 밀도가 높다는 특징은 여러 센서들에서 중복된 데이터가 센싱되어 전송되는 문제를 가지며 이로 인해 불필요한 전력 소모가 발생하고 결국 네트워크의 생존기간을 감소시킨다. 센싱된 데이터를 전송 및 수신은 센서들의 동작중 가장 많은 양의 에너지 소모를 발생 시킨다. 따라서 중복된 데이터를 센싱하여 전송하는 상황을 제거하거나, 가장 에너지를 절약하는 통신 링크를 사용하거나, 통신 횟수를 최소화하는 등의 방법으로 많은 양의 에너지를 절약하여 네트워크의 수명을 연장하게 된다. 이러한 목표를 성취하기 위해, 네트워크는 제한된 에너지를 적절하게 사용하는 로컬 최적화를 이루도록 설계되어야 한다. 예를 들면 클러스터링^[4], 데이터 융합^[5], 에너지 사용 정도를 반영하여 균형을 맞추는 라우팅^{[6][7]} 등은 로컬 최적화를 구성하는 방법의 좋은 예가 된다. 이 논문에서 우리는 클러스터링 최적화 문제에 주목한다. 네트워크 클러스터링은 대용량의 네트워크에 대해 논리적으로 그 크기를 감소시켜 에너지 소모를 감소시키는 효율적인 설계 방법으로 계층구조의 네트워크에서는 데이터 전송의 수가 감소하고 제어정보 전송시 이 정보가 네트워크 전체에 걸쳐서 폭주하는 문제를 제거할 수 있도록 한다.

클러스터링은 센서들이 클러스터를 구성하고 유지하기

위해 일정량의 에너지 소모를 필요로 한다. 뿐만 아니라 클러스터 헤드의 역할은 클러스터 멤버와 비교하여 에너지 소모가 훨씬 많다. 네트워크 영역에서 이벤트 발생은 확률통계적으로 분석할 경우 모든 센서들이 동시에 데이터를 센싱하고 동시에 기지국으로 센싱한 데이터를 전송할 확률은 낮다고 생각된다^[8]. 그러므로 각 센서의 에너지 소모율은 동작에 따라 균일하지 않게 되고 클러스터 헤드의 순환은 센서의 잔존 에너지 수준에 따라 단위지역화될 필요가 있다. 이것은 불필요한 클러스터 헤드의 순환을 줄여 에너지를 절약하고 전력 고갈로 인한 센서 노드의 정지를 최소화할 수 있을 것이다.

이 논문에서는 멀티-홉 무선 센서 네트워크에서의 두 단계 클러스터링(Two Phase Clustering : TPC) 스킴을 제안한다. 이 방법은 지연시간 제한과 같은 사용자나 응용의 특정 요구사항을 만족시켜서 궁극적으로 네트워크의 생존기간을 연장시키기 위해 에너지 효율을 지원하는 지연시간 적응형 데이터 수집 플랫폼을 제공한다. 제안된 TPC 방식은 두 단계로 클러스터링을 통해 완성된다. 첫 번째 단계에서 센서들은 클러스터 헤드와 직접 링크를 구성하고 두 번째 단계에서 해당 조건에 부합하는 이웃 노드와 릴레이 링크를 구성한다. 단계 1에서 구성된 직접 링크는 제어 메시지나 시간에 민감한 센서 데이터들을 포워딩하는데 사용되며 단계 2에서 구성된 릴레이 링크는 사용자의 지연시간 제한에 맞게 적응하여 최대한 데이터를 포워딩하여 하는데 멀티홉 경로를 구성하는데 사용되며 센서들은 비례적으로 에너지-절약효과를 갖게 된다.

시뮬레이션 결과는 제안하는 방법이 클러스터 재구성 단계에서 LEACH^[14]와 비교하여 평균 전송 거리를 최대 75%까지 감소시킴으로서 클러스터 멤버들의 센싱 데이터 수집하는 데 소모되는 에너지의 절약효과가 매우 높음을 보여준다. 뿐만 아니라 클러스터 헤드의 부하도 클러스터 멤버의 수에 거의 영향을 받지 않는 수준이 되어 단위 시간당 클러스터 헤드의 순환 횟수가 훨씬 더 작음을 알 수 있다.

이 논문의 구성은 2장에서 관련 연구내용을 소개하고 3장에서 논문의 배경을 설명한다. 4장에서는 자세한 두단계 클러스터링과정을 설명하고 5장에서 TPC를 이용한 지연시간 적응형 에너지 절약 센서 스케줄링을 소개한다. 6장에서는 시뮬레이션 결과를 통하여 성능을 보여주고 7장에서 결론을 맺는다.

2. 관련 연구

일반적으로 클러스터링 스킴은 자기 선언이나 지명방식으로 클러스터 헤드(Cluster Head (CH))를 선출한다. 최초의 지원자가 CH가 되는 방식에 기반한 수동적인 클러스터링 방법이 제안되었다^[9]. 이 방식에서는 클러스터를 구성하고 유지하기 위해 별도의 제어 메시지를 사용하지 않는다. 대신에 데이터 패킷에 Piggyback 기술을 사용하는 데 이로 인해 클러스터링의 안정성과 구성 시간이 문제가 된다. 지명방식의 클러스터링은^[10]에서 제안되었고, 이 방법은 주파수 반경이 가장 큰 노드가 CH가 된다. 만일 어떤 노드가 자신보다 더 넓은 주파수 반경을 갖는 노드를 발견할 수 없다면, 자신이 CH가 된다. 선택된 CH는 자신을 지명한 노드들로 클러스터를 구성한다. 앞에 설명된 스킴은 클러스터 구성을 유지하는 주요요소로서 에너지 메트릭을 고려하지 않으므로 에너지 효율성의 특성을 갖지 못한다. [4]에서 제안된 균형있는 k-클러스터링 알고리즘은 min-weight matching 방법을 사용하여 노드의 수를 기준으로 센서 노드들을 균형잡힌 클러스터로 그룹화한다. 여기서 제안하는 알고리즘은 센서와 마스터 노드 사이에 최대의 거리를 공간적으로 최적화한다. 이런 특징이 각 마스터에 대한 로드가 균형을 갖도록 하지만, 센서들의 잔존 에너지는 클러스터가 구성되는 동안 고려되지 않는다.

[14]에 소개된 LEACH라는 클러스터링 스킴은 단일-홉 무선 센서 네트워크를 구성하기 위해 임의 선출 방식의 클러스터 헤드 순환 알고리즘을 적용하여 CH의 에너지 소모가 균등하게 분배되도록 한다. 몇 개의 CH가 임의로 선택되고 센서들은 가장 가까운 CH를 선택한다. CH가 기지국으로 수집한 데이터를 전송하여 네트워크의 에너지 소모를 줄일 수 있다. 그러나 필드에서 모든 센서들이 기지국을 포함한 임의의 센서와 통신할 수 있다고 가정할 경우 네트워크는 확장성이 떨어지게 된다. 더구나 LEACH에 의해 얻어진 에너지 효율의 향상 정도는 PEGASIS에 의해 제안된 것으로 모든 센서들이 체인으로 연결되고 매번 기지국으로 수집된 데이터를 보내기 위해 한 홉거리의 이웃노드와 순서대로 통신을 하는 것이다^[11]. 이 때 체인 구성은 모든 센서가 네트워크 구성에 대한 정보를 알고 있다는 가정 하에 잘 동작할 수 있으며, 센서들이 밀접하게 위치한 특정 지역에서는 효율적으로 협동하여 데이터를 수집할 수 없게 한다.

최근에는 HEED 라고 하는 확률적 클러스터링 알고리즘이 제안되었는데^[20], 이 알고리즘은 노드의 근접성과

노드의 잔여 에너지 수준의 복합 요인을 고려하여 주기적으로 CH를 선택하는 것이다. [19]에서 제안된 토폴로지 적응형 공간적 클러스터링(TASC) 방법은 노드의 밀도 변화에 따라 클러스터를 구성하는 것이다. [14]에서는 적응형 클러스터 기반의 데이터 수집 방법이 제안되었고 센서가 CH와 직접 통신하고 클러스터가 임의의 방법으로 주기적으로 생성되도록 하였다. 이러한 클러스터 구성은 데이터 싱크에 의해 트리거된다.

위에서 논의한 모든 방법들은 지연시간 적응형 클러스터의 구성과 시간에 따라 변하는 지연시간 제한이나 적용되는 응용의 종류에 따른 에너지 보존율의 향상을 고려하지 않아 에너지 효율성이 떨어진다(이 논문에서는 이런 클러스터링 방법을 단일홉 클러스터링 스킴(onehop clustering scheme: OCS)라고 부른다).

3. 논문의 배경

3.1 용어 설명 및 가정

무선 센서 네트워크는 많은 수의 센서와 한 개 이상의 기지국, 주파수 범위 내에서 센서간 혹은 센서와 기지국 간 직접 통신이 가능하도록 하는 무선 링크들로 구성된다. 우리는 무선 센서 네트워크를 undirected graph $G = (V, E)$ 로 정의하고, 여기서 V 는 노드들 (센서들과 기지국들)의 집합이며 E 는 에지들 (양방향 무선 링크들)의 집합이다. 각 $s_i \in V$ 는 반경 R 이내에서 최대 통신 주파수 반경을 가지고 s_i 의 주파수 영역은 A_{s_i} 로 정의한다. 수신자 s_i 와 송신자 s_j 사이의 거리는 $d(s_i, s_j)$ ($d(s_i, s_j) \leq R$)라 하고 $e(s_i, s_j) \in E$ 로 정의되는 에지가 s_i 와 s_j 사이에 존재한다. s_i 중에서 CH로 동작하는 노드는 특히 c_i 라 하고 집합 $C \subseteq V$ 는 CH의 집합으로 정의한다. 집합 C 내의 모든 CH들 중에서 $d(c_i, s_j)$ 가 최소인 경우 각 센서 $s_j \in V - C$ 는 c_i 에 속하게 된다. c_i 로 구성되는 CH의 클러스터 멤버 집합은 $\cup_{s_j \in C} M_{c_i} = V - C$ 인 M_{c_i} 로 정의한다. $N_{s_i} = \{s_j | d(s_i, s_j) \leq R, i \neq j\}$ 를 s_i 의 지역 이웃노드라 가정하자. 다음은 이 논문에서 사용하는 주요 정의들이다.

정의 1 : 직접 링크는 c_i 와 s_j 사이의 에지 $e(c_i, s_j) \in E$, $c_i \in C$, $s_j \in M_{c_i}$ 를 말한다. 각 s_j 마다 CH로 향하는 직접 링크가 있으며 이 직접 링크는 센싱 데이터 및 제어 데이터를 전송하는데 사용된다.

정의 2 : s_i 의 데이터 릴레이 포인트는 rp_{s_i} 로 표시되며 이 둘은 동일한 c_i 에 속하는 이웃관계이다. $W_{s_i} \subseteq N_{s_i}$ 가 다른 클러스터 멤버(CM)를 위한 데이터 릴레이 포인트로서 이미 선택된 이웃 노드들의 집합일 때, $T_{s_i} = \{s_j | d(s_i, s_j) < d(c_i, s_i)\}$, $s_j \in (N_{s_i} - W_{s_i}) \cap M_{c_i}$ 이다(예를 들면 rp_{s_i} 는 s_i 에 할당된 어떤 이웃 노드들과 공유되지 않는다). 그러므로 rp_{s_i} 는 $\min[d(s_i, s_j)], \forall s_j \in T_{s_i}$ 인 이웃노드 s_j 이다.

정의 3 : 데이터 릴레이 링크는 s_j 와 rp_{s_i} 사이의 에지 $e(s_j, rp_{s_i}) \in E$, $s_j \in M_{c_i}$ 이다. 이 링크는 센싱 데이터만을 전송하는 데에만 사용된다. M_{c_i} 내의 모든 센서 s_j 는 자신만의 rp_{s_i} 를 가질 수 있으며, c_i 로 센싱된 데이터를 전달하는 가장 긴 멀티홉 데이터 릴레이 패킷은 다음과 같다:

$$s_1 \rightarrow s_2 \rightarrow s_3 \rightarrow \dots \rightarrow s_k \rightarrow c_i, k = |M_{c_i}|$$

다음은 이 논문에서 사용되는 가정들이다.

- 동일성질의 센서들이 지리적으로 균일하게 분포되고 수신 노드까지의 거리에 따라 전송 파워를 조절하는 능력을 가진다^[11].
- 각 센서는 GPS나 다른 수단의 도움으로 네트워크 내에서 자신의 위치를 인식한다^[12].
- CH가 수집한 센싱 데이터는 기지국이 전송 주파수 거리 내에 있지 않는 경우 멀티홉 라우팅 경로를 통해 기지국으로 전달된다. 이 논문에서 제안하는 방법은 라우팅 계층 위에서 동작하는 클러스터링 프로토콜로서 수행되며 매홉간 데이터 전달시 마다 목적지까지와의 최소거리 유지하는 위치 기반의 라우팅 프로토콜이 라우팅 계층으로 수행된다고 가정한다^[13].

3.2 구체적인 문제 설명

밀도가 높은 경우 노드 인접성으로 인하여 발생하는 높은 데이터 중복성을 고려할 때, 센서들을 미리 중복된 데이터 수집을 제거하도록 설계하여, 불필요한 데이터 전송으로 인해 소모되는 에너지를 절약할 필요가 있다.

센서 네트워크의 응용-의존적 특성을 고려할 때 응용마다 주어진 작업을 성공적으로 수행하기 위해 필요한 지연시간은 각기 다양할 수 있다. 그러므로 센서들의 데이터 보고 시간 간격은 사용자나 특정 응용의 지연시간에 대한 요구사항을 만족하면서 에너지 절약을 최대화하기

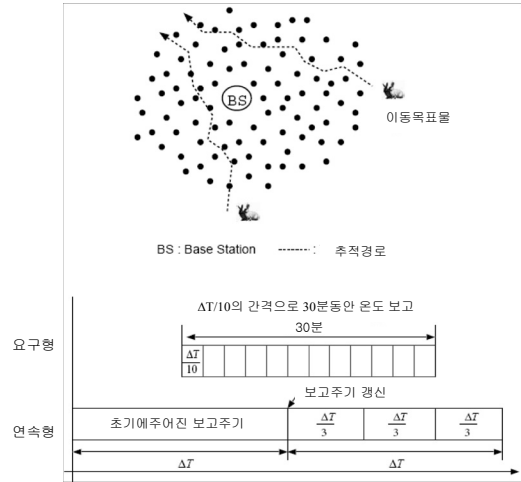


그림 1. 동적인 데이터 보고 시간 간격의 예

위해, 데이터 중복을 제거하는 능력을 융통성 있게 조절해야 한다. 예를 들면, 센서에 주어진 작업이 이동하는 목표물을 감시하는 것이라고 하자. 그림 1은 목표물을 감시하는 동안 데이터 보고 간격이 동적으로 변경되고 있음을 보여준다. 초기에 센서들은 매 ΔT 시간마다 센싱된 데이터를 전송한다. 사용자들은 이동 목표물의 속도에 따라 주기적인 보고 간격을 $\frac{\Delta T}{3}$ 로 설정한다. 이에 따라 센서들을 높은 에너지 소모가 필요하더라도 새로운 지연시간에 대한 요구사항을 만족시키기 위해 전송 간격을 조절한다. 또한 두 개의 서로 다른 데이터 전송 모델(요구형과 연속형 방식)이 공존할 수도 있으며 이런 방식들은 서로 다른 전송 지연 조건을 갖을 확률이 높다. 문제는 높은 데이터 중복성에 대한 제어 뿐 아니라 다양하게 변하는 지연 시간 적응형 보고 능력을 어떻게 센서에 제공하는 것이다. 본 논문에서 클러스터링을 이용하여 위의 문제를 해결하고자 한다.

4. 두 단계 클러스터링

센싱된 데이터 간의 관계 정도는 두 개의 센서 사이의 거리와 밀접한 관계가 있으며, 이로 인해 근접한 위치의 센서들은 중복된 센싱 데이터를 생성할 확률이 매우 높다. 이런 견지에서, TPC 방식은 먼저 단계 I에서 근접하게 위치한 센서들로 클러스터를 구성한다. 다음 단계 II에서 클러스터 노드들간에 연결구성을 재구성하여 에너지를 보존할 향상을 높이는 운영을 한다. 단계 I과 II 동안, 센서들은 전송 파워를 조정하지 않고 최대 주파수 반경을 유지

표 1. LEACH, PEGASIS와 TPC의 정성적 비교

파라미터	LEACH	PEGASIS	TPC
논리적 구조	Star	Chain	불완전한 Wheel
멤버 제어	효율	비효율	효율
에너지 효율	직접통신에 대한 보완	LEACH에 대한 보완	LEACH에 대한 보완, 그러나 PEGASIS 만큼 많지 않음
중복적 데이터 제거	효율	비효율	효율
단일/멀티-홉	단일홉	단일홉	단일/멀티홉
트래픽 분산	효율	비효율	효율
지연시간 적응성	고려안함	고려	효율적으로 고려

```

Procedure CF begin /*  $\forall s_i \in V$  */
1:  $t \leftarrow \text{RAND}(x)$ 
2: schedule a CH advertisement with t sec. delay;
3: if any CH advertisement is overheard for t sec.
4: then add the sender to K;
5: cancel the scheduled CH advertisement;
6: go to S2;
7: else proceed with the scheduled advertisement;
8: exit;
9: end-if
10: S2: collect senders of incoming CH advertisements in K;
11: choose a CH with  $\min_{c_i \in K} [d(s_i, c_i)]$ ;
12: send a join-request message to  $c_i$ 
End-Procedure
    
```

그림 2. 클러스터 구성 과정

하여 존재하는 모든 이웃 노드들을 알아낸다. TPC 방식의 주요 특징은 다음과 같다.

- 사용자 요구사항을 만족시키기 위해 센서를 효율적이고 융통성 있게 제어하여 최대의 에너지 절약효과 제공
- 지연시간 제한 조건이 없는 경우에는 CH의 로드가 CM의 수에 거의 영향을 받지 않음
- 클러스터 내부에서의 불필요한 CH 순환으로 인한 에너지 소모를 줄임.
- 이웃 노드 간에 잔존 에너지 수준 정보를 교환 없이 잔존 에너지 레벨에 기반한 CH 순환방식을 운영하여 제어 메시지로 인한 오버헤드를 감소시킴.

표 1은 일곱 개의 파라미터를 이용하여 TCP 방식을

LEACH^[14]과 PEGASIS^[11]방식과 비교한 것이다. 이 표로부터 TPC 방식은 다른 두 가지 방법에 비해 단일홉과 멀티홉 네트워크 둘다에 대해 센서가 사용자의 지연시간 요구사항에 적응하고 있음을 알 수 있다.

4.1 단계 I: 클러스터 구성

단계 I에서 네트워크는 클러스터로 나뉘어지고 각 클러스터별로 CH를 가지게 된다. 한 클러스터의 주파수 반경 내에는 오직 하나의 CH만이 있다고 가정한다. 무선 통신의 비용은 전송 거리에 비례하므로, 각 센서가 CH로 직접 센싱한 데이터를 포워드하는데 드는 에너지 소모를 최소화하기 위해서는 CH로의 거리가 자신의 CH를 선택하는 주요한 기준이 된다. 클러스터 구성(CF) 과정은 그림 2에 보여준다. 여기서 t 는 브로드캐스트 지연시간이며 $\text{RAND}(x)$ 는 0과 x 사이에 균일 분포 구성에 필요한 난수를 생성하기 위한 함수이다. K 는 CH advertisement overheard의 송신자를 저장하기 위해 사용되는 임시 집합이다. $d(s_i, c_i)$ 는 s_i 와 자신의 CH advertisement를 성공적으로 브로드캐스트하는 c_i (클러스터 헤드)사이의 거리이다. CF 과정은 주파수 반경 내에 유일한 CH가 생성되도록 한다(그림 5에서 3 - 9의 과정). 만약 여러 개의 CH advertisement가 존재할 경우에는 s_i 는 가장 가까운 것을 선택하게 된다(10-11의 과정). 같은 거리 내에 두 개 이상의 CH가 존재하는 경우에는 한 CH가 임의로 선택되고 나머지 노드와의 관계는 해제한다. 일단 하나의 CH가 결정되고 나면, 센서들은 CH로 join-request를 보내서 CM이 된다(12의 과정).

클러스터 구성과정 동안에, 각 센서 s_i 는 어떤 이웃 노드들이 CH로서 동작하는지, 어떤 노드가 클러스터 내에서 CM으로 동작하는지, 이웃노드가 어떤 클러스터 내에 위치하는 지에 대한 정보를 수집한다. CH advertisement나 join-request 메시지를 들으면, s_i 는 이 메시지에서 송신자의 위치 정보와 멤버십을 확인하고 이 정보를 자신의 이웃 노드의 집합인 N_{s_i} 에 저장한다. s_i 는 어느 이웃 노드가 같은 클러스터 내에 속하는 지, 어떤 노드들이 CH로 동작하는지, 그들에 대한 거리는 얼마인지를 N_{s_i} 에 질의하여 식별한다. 이렇게 수집된 정보는 CH 순환과 단계 II의 클러스터 재구성 과정에 사용된다.

4.2. 단계 II : 클러스터 재구성

단계 I에서 클러스터가 형성되고 나면, 각 클러스터는 CH가 지역적으로 중앙에 위치하고, 클러스터 노드들간에

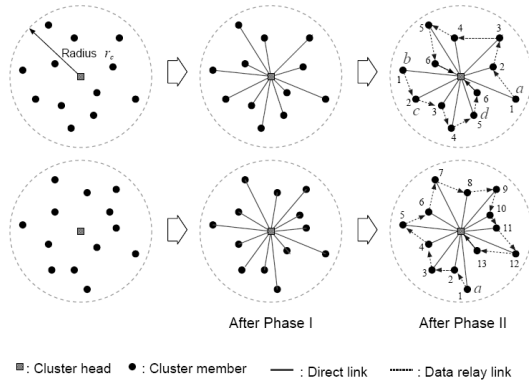


그림 3. 두 단계 클러스터링의 과정 설명

```

Procedure CR_CH begin /*  $\forall c_i \in C$  */
1: S1: if  $M_{c_i} - W_{c_i} = \phi$ 
2: then stop and exit;
3: else  $n \leftarrow 1$  and choose the farthest  $s_i \in M_{c_i} - W_{c_i}$ ;
4: add  $s_i$  to  $W_{c_i}$  and send  $m_{s_i}^{c_i}(n)$  to  $s_i$ ;
5: monitor the construction procedure;
6: end-if
7: while (monitoring)
8: if  $c_i$  overhears  $m_{s_j}^{s_j}(n)$  then add  $s_j$  to  $W_{c_i}$  end-if
9: if  $c_i$  receives  $m_{c_i}^{s_i}(n)$ 
10: then add  $s_i$  to  $W_{c_i}$ ;
11: go to S1;
12: end-if
13: end-while
End-Procedure
    
```

그림 4. CH를 위한 클러스터 재구성 과정

connectivity는 그림 3에서 보여주는 것처럼 스타형 구조가 된다. 단계 II에서는 이 구조에 추가적인 에지들의 집합을 추가하여 스타형 그래프의 변형을 가져오고 그 결과 클러스터 노드간의 connectivity를 재구성하게 된다. 이런 재구성의 목적은 지연 시간에 대한 제한 사항을 만족시키면서 모든 CM으로부터 센싱 데이터를 수집하는데 드는 에너지 소모를 최소화하는 것과 CH의 부하를 분산시키는 것이다. 구체적으로 말하자면, 직접 링크를 사용하여 센싱한 데이터를 직접 CH로 보내기 위해 소모되는 총 에너지 양은 상대적으로 짧은 데이터 릴레이 링크를 사용할 경우 감소될 수가 있다. 게다가, CH의 워크로드(예를 들면, CM들간의 직접 통신과 같은)는 데이터 릴레이 링크를 사용하는 CM의 수에 비례하여 감소할 수 있고 따라서 이로 인한 에너지 소모

```

Procedure CR_CM begin /*  $\forall s_i \in M_{c_i}$  */
1: monitor the construction procedure;
2: while (monitoring)
3: if  $m_{s_i}^{c_i}(n)$  or  $m_{s_j}^{s_j}(n)$  is received
4: then  $f_{ix} \leftarrow n$  as its forwarding index;
5: find data relay point  $rp_{s_i} \in (M_{c_i} - W_{s_i}) \cap N_{s_i}$ ;
6: if  $rp_{s_i}$  exists
7: then  $n \leftarrow +1$ ;
8: send  $m_{rp_{s_i}}^{s_i}(n)$  to  $rp_{s_i}$ ;
9: send  $m_{c_i}^{s_i}(n)$  to  $c_i$ ;
10: end-if
11: end-if
12: if  $m_{s_j}^{s_j}(n)$  or  $m_{c_i}^{s_j}(n)$  is overheard /*  $s_j = rp_{s_j}$  */
13: then add  $s_j$  to  $W_{s_i}$  if  $s_j \in M_{c_i} - W_{s_i}$ ;
14: end-if
15: end-while
End-Procedure
    
```

그림 5. CM을 위한 클러스터 재구성 과정

도 감소한다. 모든 CM들은 지연시간 제한이 민감하지 않은 경우에, CH로 센싱 데이터를 보내기 위해 직접 링크 대신에 데이터 릴레이 링크를 사용한다. 이들 릴레이 링크를 따라 전송되는 센싱 데이터는 각 데이터 릴레이 포인트에서 수집한다. 릴레이 링크는 데이터 릴레이 포인트를 찾고자 하는 CH의 요청으로 구성된다. 그러나, 클러스터내에서 센서의 분포밀도에 따라 어떤 CM은 데이터 릴레이 포인트를 찾는데 실패할 수도 있으며 이 경우 변형된 클러스터 노드간의 connectivity는 $0 \leq$ 데이터 릴레이 포인트의 수 $\leq |M_{c_i}|$ 와 같은, CH가 중심에 있는 휠 그래프의 서브 그래프가 될 수 있다.

CH c_i 는 시작점으로 M_{c_i} 중에서 가장 먼 CM, s_i 를 선택하는 것으로 클러스터 재구성을 시작한다(그림 3에서 센서 a). c_i 는 시작점 s_i 에 f_{ix} 로 표시되는 포워딩 인덱스를 부여하고 초기값 1을 세트한다. 이후 데이터 릴레이 포인트가 발견될 때마다 f_{ix} 의 값은 1씩 증가한다(예. 그림 3에서 CM 옆의 숫자). 반면에 CM이 데이터 릴레이를 찾는데 실패한다면 f_{ix} 는 다시 1로 세트된다. 실패의 경우에, CM은 CH에게 릴레이 포인트가 이용가능하지 않다는 것을 알리게되고 CH는 다시 새로운 시작점으로서 그동안 과정에 참가하지 않았던 남아있는 CM 중에서 가장 먼 노드를 선택하여 $f_{ix} = 1$ 로 한다(그림 3에서 센서 b).

클러스터 재구성에 필요한 과정은 그림 4과 그림 5에

보여주는 것처럼 c_i 를 위한 CR_CH와 모든 $s_i \in M_{c_i}$ 를 위한 CR_CM으로 이루어진다. 여기서 W_{c_i} 는 릴레이 링크를 설정하기 위해 데이터 릴레이 포인트를 찾는 과정을 이미 완료한 CM의 집합을 나타내며, s_i 중에서 $n = 1 + f_{ix}$ 인 $m_{s_j}^s(n)$ 은 s_i 에서 s_j 로 보내지는 릴레이 링크 설정 메시지를 나타낸다.

처음에 CM은 데이터 릴레이 포인트로 센싱 데이터를 포워딩한다. 데이터 릴레이 포인트는 자기의 센싱 데이터와 수신한 데이터를 수집하여 자신의 데이터 릴레이 포인트로 전송하며, 전송받은 데이터 릴레이 포인트도 같은 과정을 반복한다. 데이터 릴레이 포인트는 자신의 포워딩 시간에만 수집과정을 수행하고 데이터 패킷을 생성한 후 데이터 릴레이 포인트의 가용성에 따라 다음 데이터 릴레이 포인트나 CH로 전송한다. 그러므로 데이터 처리과정의 오버헤드는 릴레이의 수에 상관없이 동일하게 된다. 예를 들면, 그림 3에서 데이터 처리과정의 오버헤드는 센서 c와 d에서 동일하다. 센싱 데이터를 전송하기 위해 데이터 릴레이 링크나 직접 링크 중에 어느 것을 사용할지의 선택은 사용자나 응용이 정한 지연시간 제한에 따라 CH에 의해 제어된다. 지연시간 제한을 만족하는 릴레이의 최대 수는 R_{max} 로 표시하며, 각 CM에서 modulo 연산을 사용하여 결정된다. CH는 R_{max} 를 포함한 제어 메시지를 모든 CM에게 브로드캐스트하고 각 CM은 $f_{ix} \bmod R_{max}$ 를 계산한다. 그 결과가 0이면, CM은 데이터 릴레이 링크가 가용하더라도 수집한 데이터를 직접 링크를 통해 CH로 전송하고 그렇지 않으면 데이터 릴레이 링크를 사용한다. 이렇게하면, 기껏해야 n 개의 센서가 센싱 데이터를 릴레이하는데 참가하게 된다.

4.3 지역내의 클러스터 헤드 순환방법

s_i 가 CH c_i 가 되고나면, 이 노드가 가진 에너지의 양은 E_{CH} 가 된다. 그러므로 CH가 정해진 양의 에너지를 모두 사용하거나, 에너지 고갈로 인해 단절이 일어날 상황이 감지되면 클러스터 헤드의 순환 과정이 일어나게 된다. 센서가 E_{CH} 의 에너지를 가진 CH로서 동작하는 시간은 데이터 전송 주파수에 따라 다양하다. 즉, 센서가 센싱하고 보고하는 이벤트 발생의 강도는 전체 네트워크 공간에 걸쳐 균일하지 않다. 그러므로 TPC 방식은 클러스터 헤드 순환을 단위지역으로 제한하여, 불필요한 클러스터 헤드 순환으로 인해 소모되는 에너지를 절약한다. CH는 $E_{CH} = 0$ 일 때, 서비스 완료 메시지, SC_{c_i} 를 브로드캐스트하여 로컬 클러스터 헤드 순환을 동작시킨다. SC_{c_i} 를 받

```

Procedure LCHR_CH begin /* for  $c_i$  */
1: if ( $E_{CH} = 0$ );
2: then broadcast a service completion message  $SC_{c_i}$ ;
3: collect senders of incoming CH advertisements into  $N_{c_i}$ ;
4: if (there is no CH advertisement overheard)
5: then perform its task independently until
6: a CH advertisement is received;
7: else choose a CH with
 $\min_{\forall c_j \in N_{c_i}} [d(c_i, c_j)]$ , for  $i \neq j$ 
8: send a join-request message to  $c_j$ 
9: end-if
10: end-if
End-Procedure
    
```

그림 6. CH를 위한 단위지역의 클러스터 헤드 순환 과정

```

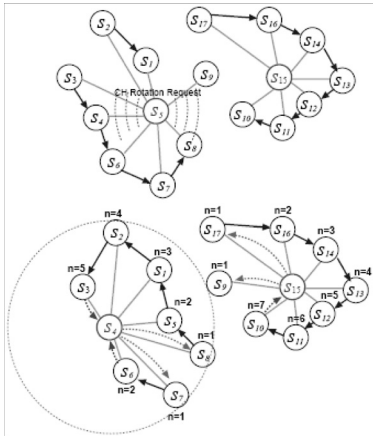
Procedure LCHR_CM begin /* for  $\forall s_i \in M_{c_i}$  */
1: mark sender  $c_i$  of  $SC_{c_i}$  in  $N_{s_i}$  as a non-cluster head;
2: if (there is no other cluster head overheard in  $N_{s_i}$ )
3: then  $t \leftarrow \frac{E_{IN} - E_{CT}}{E_{IN}}$ ; /*  $t$ : broadcasting delay */
4: schedule a CH advertisement with delay  $t$ ;
5: if (any CH advertisement is overheard for  $t$ )
6: then add the sender to  $N_{s_i}$ ;
7: cancel the scheduled CH advertisement;
8: go to S1;
9: else proceed the scheduled advertisement;
10: exit;
11: end-if
12: else choose  $s_j$  with  $\min_{\forall s_j \in N_{s_i}} [d(s_i, s_j)]$  as new CH;
13: end-if
14: S1: collect senders of incoming CH announcements into  $N_{s_i}$ ;
15: choose  $s_j$  with  $\min_{\forall s_j \in N_{s_i}} [d(s_i, s_j)]$  as new CH;
16: send a join-request message to  $s_j$ ;
End-Procedure
    
```

그림 7. CM을 위한 단위지역의 클러스터 헤드 순환 과정

으면 각 CM, s_i 는 잔존 에너지 수준에 따라 새로운 CH로서 자신을 선출하도록 시도한다. 그러나 CH로 동작하는 또다른 이웃노드가 있다면, s_i 는 선출과정에 참가하지 않고 이 이웃노드를 새로운 CH로 선택한다.

로컬 클러스터 헤드 순환 과정은 c_i 를 위한 LCHR_CH와 모든 $s_i \in M_{c_i}$ 를 위한 LCHR_CM으로 그림 6과 그림 7에 보여준다. 여기서 E_{IN} 과 E_{CT} 는 각각 s_i 의 초기와 현재의 에너지 수준을 표기한다. LCHR_CM 과정은 한 CH

단위지역의 클러스터 헤드순환(LCHR)



1. S_5 는 서비스완료를 공지하여 CH 순환을 시작한다.
2. 잔존에너지 수준을 가지고 계산한 지연시간 t 를 가지고, CH S_{15} 에 참가하고 있는 S_8 를 제외한 CM들은 자신이 CH의 역할을 하겠다고 공지하고 일단 셋업 메시지를 받는다.
3. S_2 의 공표가 상대적으로 가장 높은 잔존에너지 수준으로 인하여 가장 짧은 지연시간을 보이므로 새로운 CH가 된다.
4. S_4 는 단계 I과 단계 II를 수행한다.
5. S_{15} 는 S_8 의 참가로 인해 단계 II를 재수행한다.

그림 8. 단위지역 기반 클러스터 헤드 순환의 예

가 주파수 범위내에 또다른 CH와 충돌하지 않는 것을 보장하고(그림 6의 5-11과정) 클러스터 헤드 순환과정이 단위지역으로 제한되도록 한다. 처음 클러스터 구성과정과 달리, 센서가 같은 거리 내에 하나 이상의 CH를 발견할 경우 처음 자신이 CH임을 공표한 CH를 선택한다. 그 이유는 공표를 알기까지의 시간이 짧을수록 잔존에너지 자원이 더 크다는 것을 의미하기 때문이다. 클러스터 헤드 순환과정을 트리거한 CH 주변의 CH들은 LCHR로 인해 새로 참가하거나, 떠나는 멤버가 존재하게 되면 단계 II를 수행하게 된다. LCHR 과정의 예는 그림 8과 같다.

5. 지연시간 적응형 센서 스케줄링

데이터 보고 지연시간과 같은 사용자의 요구사항은 응용의 종류에 따라 다양하며 정적인 경우도 있고 동적인 경우도 있다. 동적인 경우에 사용자들은 환경적 상황이나 이미 전송된 센싱 데이터의 분석에 따라 제어 메시지를 보내서 센서의 행위(예를 들면, 센싱이나 데이터 보고)를 제어한다. 반면에 정적인 경우에는, 센서들은 초기에 주어진 임계값에 기반하여 데이터 보고 주기를 결정해야 한다.

시간이 민감한 환경에는 센서들이 에너지 소모가 상대적으로 많더라도 지연시간 제한요건을 만족하는데 더 높

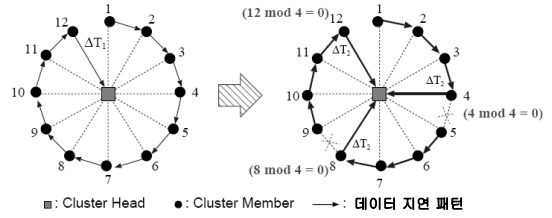


그림 9. 지연시간 적응형 에너지 절약 데이터 수집 예

은 우선순위를 두어야 한다. 반면에, 주기적인 센싱 데이터 보거나 네트워크 상태 보고와 같이 시간이 민감하지 않은 환경에서는 에너지 절약에 초점이 맞추어져야 한다. 그러므로 센서들은 시간 요건 변화와 상황에 따른 사용자 지연시간 요구사항에 융통성 있게 적응할 수 있어야 하며, 에너지 보존율을 극대화할 수 있어야 한다. TPC 방식은 센서들이 센싱 데이터를 단위지역내의 제어 센터, 즉, CH로 보고할 때, 데이터 릴레이 링크와 직접 링크의 두가지 데이터 포워딩 링크를 사용하여 에너지 보존율을 극대화한다.

그림 9는 데이터 릴레이 링크와 직접 링크를 사용하여 센싱 데이터 보고 간격이 다양한 경우에 어떻게 적응하는지를 예로서 설명한다. 이 예에서 초기 데이터 보고 간격은 $\Delta T_1 = 30$ 초이다. 이 초기 간격은 모든 CM이 릴레이 링크를 통해 센싱 데이터를 포워드할 수 있을 만큼 충분히 커서 최대 에너지 절약 효과를 낼 수 있도록 결정한다. 센서들이 주어진 업무를 수행하는 동안, 사용자는 전달된

센싱 데이터를 분석하여 새로운 보고 간격, 예를 들면 $\Delta T_2 = 12$ 초인 요구사항을 CH로 보낸다. 지연시간 제한요건을 받은 CH는 이 제한요건을 만족할 수 있는 릴레이의 최대 수를 계산한다. 이 결과를 4-릴레이(예, $R_{max} = 4$)라고 하자. CH는 4-릴레이 제어 메시지를 브로드캐스트한다. 다음 CM은 $f_{ix} \bmod 4$ 의 결과에 따라 직접 링크나 데이터 릴레이 링크를 통해 자신의 센싱 데이터를 전달한다. $f_{ix} = 4, 8, 12$ 인 센서들은 modulo 계산의 결과가 0이므로 자신이 센싱한 데이터 (이전 흡으로부터 받은 센싱 데이터와 수집된)를 데이터 릴레이 링크 대신 직접링크를 사용하여 CH로 전달한다.

6. 시뮬레이션 결과

CD-DGS 데이터 수집방식의 기본과정으로 제안된 TPC 방식 중, 먼저 단계 II에서의 에너지 보존과 CH의 워크로드 감소 효과에 대하여 다루기로 한다. 비록 TPC

와 OCS가 다른 네트워크 플랫폼에서 설계되었더라도, 클러스터 구성 단계 I은 OCS를 멀티홉 센서 네트워크로 항상 변경한 버전이므로 단계 II를 실행하여 얼마나 많은 에너지가 절약될 수 있는지를 측정하여 단계 I에서 얻어지는 절약효과와 비교하고자 하며 이것은 TPC 방식과 OCS의 성능비교 결과를 제공하게 된다.

6.1 성능 평가 파라미터

이 절에서는 단계 I과 단계 II에서 다음 두 가지 메트릭을 측정하여 TPC 방식의 에너지 보존 능력을 평가한다 :

- 평균 전송 거리 : 무선 통신에서의 에너지 소모는 수신단까지의 거리에 비례하므로 데이터 전송에 소모하는 에너지 양에 대한 인덱스로 평균 전송 거리를 고려한다.
- 잔존 에너지 수준 : 센서는 자신이 센싱한 것을 때 데이터 보고 라운드마다 CH로 보고한다. 얼마나 많은 에너지가 절약되었는지를 보여주기 위해 1000번의 보고 라운드마다 각 센서의 잔존 에너지 수준을 측정한다.

이 논문에서는 TPC 방식을 JAVA Thread 기반으로 구현하여 시뮬레이션을 수행하였다. 이 실험에서는 동일한 성질의 센서들을 500m×500m의 네트워크 공간에 다양한 밀도를 가진 균등 분포로 설치하였다. 전송거리를 측정하기 위해 다음의 네 가지 밀도, 1센서/200m², 1센서/300m², 1센서/400m²와 1센서/500m²가 사용되고, 에너지 소모를 측정하기 위해 1센서/350m²가 사용된다. $r_c = 50m$ 와 $r_c = 100m$ 인 두 가지 통신 주파수 범위가 사용된다. 단계 I과 단계 II가 완료된 후에, 센서는 이벤트에 대한 센싱을 시작하고 3초의 데이터 보고 주기마다 CH로 센싱 데이터를 전송한다(예. 지속적인 데이터 보고 모델). 센서는 센싱 데이터를 1000번 생성하도록 설정된다. CM으로부터 센싱 데이터를 수집하는데 소모되는 에너지를 측정하기 위해 [14]에서 적용한 것과 동일한 에너지 모델로 다음을 사용한다 : radio electronics energy 50nJ/bit, radio amplifier energy 100pJ/bit와 512 비트 크기의 데이터 패킷. 지연시간 적용형 데이터 수집방법의 실험을 위해 5-릴레이 제어 메시지를 사용한다. 각 네트워크 밀도에 대해 다섯 번의 실험을 수행하여 결과를 도출하였다.

6.2 전송 거리의 효과

앞에서 설명한 것처럼 전송거리는 CM이 센싱 데이터

표 2. 평균 전송 거리에 대한 실험 결과

센서 밀도	평균 CH수		평균전송거리(m) (단계 I, 단계 II)	
	$r_c=50m$	$r_c=100m$	$r_c=50m$	$r_c=100m$
1센서/500m ²	58.2	18.8	(23.3, 14.4)	(45.8, 17.7)
1센서/400m ²	61.2	18.4	(23.1, 13.4)	(47.8, 16.0)
1센서/300m ²	63.0	19.4	(23.0, 12.2)	(46.2, 14.3)
1센서/200m ²	65.2	19.0	(23.9, 10.7)	(46.8, 11.9)

표 3. 전송 거리 감소 비율에 따른 실험 결과

센서 밀도	평균 CH수		단계 I과 단계 II의 평균 전송 거리 감소 비율	
	$r_c=50m$	$r_c=100m$	$r_c=50m$	$r_c=100m$
1센서/500m ²	58.2	18.8	38.3%	61.4%
1센서/400m ²	61.2	18.4	41.6%	66.3%
1센서/300m ²	63.0	19.4	47.0%	68.9%
1센서/200m ²	65.2	19.0	55.1%	74.5%

를 포워드하기 위해 소모하는 에너지의 양에 비례한다 [6][15][16]. 그러므로 단계 II에서 구성된 데이터 릴레이 링크를 사용하여 TPC가 얼마나 많은 전송거리를 감소시키는지를 측정할 필요가 있다. 모든 CM이 단계 II 과정 수행에 대한 이득을 보는 것이 아니므로, 전송거리 감소 비율(예. $\frac{\text{직접 링크의 평균 길이} - \text{릴레이링크의 평균 길이}}{\text{직접 링크의 평균 길이}}$)은 각 CM의 데이터 릴레이 포인트의 가용성에 따라 결정된다. 즉, 단계 II에서의 전송 거리는 어느 CM도 데이터 릴레이 포인트를 발견하지 못한 최악의 경우에는 단계 I에서와 같다. 그러나 일반적으로 센서가 네트워크에서 밀접하게 분포하기 때문에 이것은 아주 드문 경우이다.

네트워크 밀도가 증가함에 따라 CM은 자신의 주파수 범위 내에 있는 이웃 노드가 늘어나게 되고, 이웃노드간 거리가 감소하게 되므로, 데이터 릴레이 포인트를 발견할 가능성이 증가하게 된다. 더구나 주파수 범위는 네트워크 밀도가 높을 때 감소 비율에 영향을 끼치는 주요 요인이 된다. 이것은 주파수 범위가 커질수록, 이웃노드의 수가 많아지기 때문이다. 표 2와 표 3의 실험 결과는 네트워크 밀도와 주파수 범위의 크기가 어떻게 전송 거리 감소 비율에 영향을 주는지를 보여준다. 이 표에서는 단계 II에서의 평균 전송거리가 밀도와 주파수 범위 면적의 증가에 따라 눈에 띄게 감소하는 것을 관찰할 수 있다. 특히, TPC는 각 CM의 평균 전송 거리를 단계 II의 수행으로 최대 74.5%까지 감소시키며 이것은 상당한 양의 에너지 절약효과를 의미한다. 반면에 단계 I에서는 평균 전송 거

리가 밀도에 의해 영향을 받지 않는다. 그 이유는 CM이 클러스터 영역(CH의 주파수 범위)내에 균일하게 분포되어 있고, 모든 CM이 직접 CH로 센싱 데이터를 전송하기 때문이다. 전송 파워가 전송거리에 비례하므로 각 센서의 센싱 데이터 전송에 필요한 전송거리가 감소되었음을 보여주는 것은 단계 II를 수행하는 것이 얼마나 많은 에너지가 절약되는 지를 보여줄 수 있는 방법이 된다.

6.3. 에너지 절약 효과와 클러스터 헤드의 워크로드

이 절에서는, 각 센서가 1000번의 센싱 데이터 보고를 한 후에 잔존 에너지 수준의 분포를 보여준다. 각 전송에 대한 에너지 소모량을 측정하기 위해, [14]에서 소개한 에너지 모델을 사용하였다. 일반적으로 자유 공간에서의 저주파 무선 통신에 대해, 전자적인 에너지 뿐 아니라, 송신자와 수신자사이의 통신 거리도 에너지 소모를 측정하는 주요 요인으로 고려된다. [14]의 에너지 모델에서는 짧은 주파수 통신 환경에서 ℓ 비트 메시지를 s_i 에서 s_j 로 전송하는데 소모되는 총 에너지를 $E_{tx}(d(s_i, s_j))$ 로 표시하고, s_j 가 메시지를 수신하는데 소모되는 총에너지를, E_{rx} 로 표시하며 다음과 같이 계산한다.

$$\begin{aligned} E_{tx}(d(s_i, s_j)) &= \ell_{msg} \times (E_c + E_a \times d(s_i, s_j)^2), \\ E_{rx} &= \ell_{msg} \times E_c \end{aligned} \quad (1)$$

여기서 E_c 는 메시지 송수신에 필요한 전자적 에너지이고, E_a 는 s_i 와 s_j 사이에서 신뢰성있는 데이터 전송을 위한 $d(s_i, s_j)$ 에 비례하는 증폭 전송 에너지이다. 이 모델에 기반하여, 단계 I의 수행 후에 클러스터내의 모든 CM으로부터 센싱 데이터를 수집하는데 드는 총에너지 소모량, E_{tot}^I 은 다음과 같이 구할 수 있다.

$$E_{tot}^I = E_{CH}^I + \sum_{M_i} E_{CM}^I, \quad (2)$$

여기서 $E_{CH}^I = \sum_{M_i} E_{rx} + E_{tx}(d(c_i, s_j))$ 이고, $E_{CM}^I = s E_{tx}(d(s_i, c_i))$ 이다. 단계 II 수행 후에 클러스터 내 센싱 데이터 수집에 필요한 에너지 소모(E_{CH}^II)는 다음과 같이 구할 수 있다.

$$E_{CH}^II = \sum_k E_{rx} + E_{tx}(d(c_i, s_j)), \quad (3)$$

여기서 k 는 rp_s 대신에 CH로 센싱 데이터를 포워드하

는 CM의 수로서 사용자 지연시간 제한요건에 따라 다양하다; s_j 는 CH가 최종적으로 CM으로부터 수집한 센싱 데이터를 기지국(만약 기지국이 바로 도달할 수 있다면, $s_j = s_s$ 이다)으로 보고할 때 거치는 바로 다음 홉이며 각 CM의 에너지 소모량, E_{CM}^II 는 다음과 같이 주어진다.

$$E_{CM}^II = \begin{cases} E_{tx}(d(s_i, c_i)) & \text{if } s_i \text{ does have } rp_{s_i} \\ E_{rx} + E_{tx}(d(s_i, c_i)) & \text{if } f_{ix} \bmod n = 0 \\ E_{tx}(d(s_i, rp_s)) & \text{if } f_{ix} \bmod n = 1 \\ E_{rx} + E_{tx}(d(s_i, rp_s)) & \text{if } 1 < f_{ix} \bmod n \leq n-1 \end{cases} \quad (4)$$

따라서 위의 모델은 단계 II의 수행후에 한 클러스터 내에서 센싱한 데이터 수집에 필요한 총에너지 소모량, E_{tot}^II 이 각 CM 내에서 rp_{s_i} 의 가용성에 따른다는 것을 보여준다. 우리는 이 모델을 단계 I과 단계 II를 대한 실험에서 에너지 소모량을 측정하기 위해 사용하며 이에 따라 각 센서에서의 잔존 에너지 수준이 계산된다. 에너지 소모의 전체적인 경향을 보여주기 위해, 각 클러스터로부터 수집한 잔존 에너지 수준을 분류하였다.

지연시간 제한 데이터 수집을 위해 센싱 데이터 릴레이의 최대 수를 5(예. 5-릴레이)로 설정하였다. 주파수 범위 반경, $r_c = 50m$ 이 그림 10과 그림 12에서 사용되고, $r_c = 100m$ 이 그림 11과 그림 13에서 사용된다. 그림 10과 그림12는 각각 단계 I(CH로의 직접 통신)과 지연시간 제한요건이 있을 때와 없을 경우에 단계 II에서 각 센서의 잔존 에너지 수준을 비교한 결과이다. 앞에서 언급한 것과 같이, 주파수 범위는 에너지 소모에 비례하는 전송 거리에 영향을 끼치는 요인이다.

그림 11은 그림 10의 결과와 비교하여 에너지 절약효과가 훨씬 더 크다는 것을 보여준다. 지연시간 제한요건

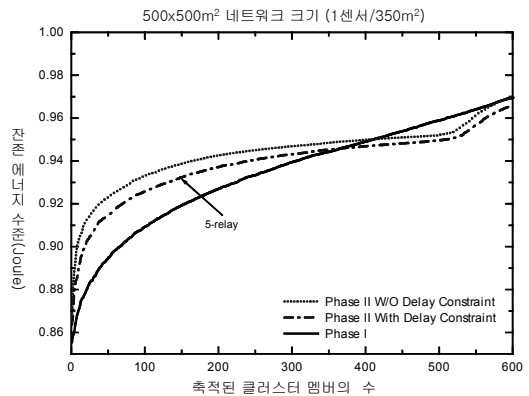


그림 10. 클러스터 멤버의 잔존 에너지 수준 분류($r_c=50$)

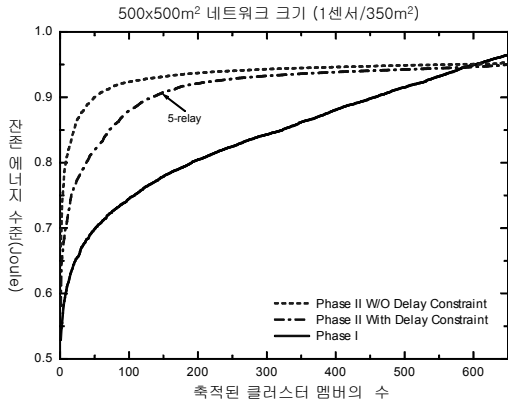


그림 11. 클러스터 멤버의 잔존 에너지 수준 분류($r_c=100m$)

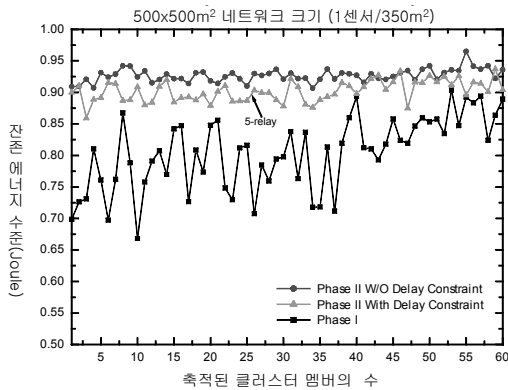


그림 12. 클러스터 헤드의 잔존 에너지량 ($r_c = 50m$)

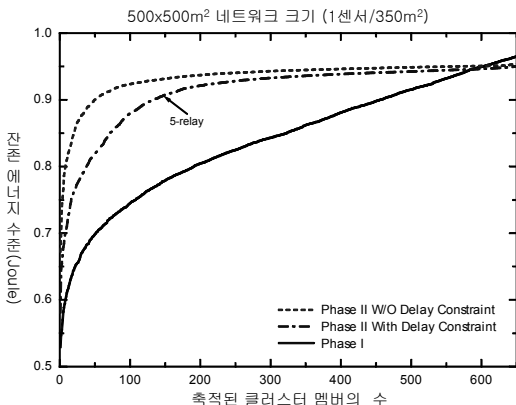


그림 13. 클러스터 헤드의 잔존 에너지량 ($r_c = 100m$)

(5-릴레이)을 만족시키기 위해, 어떤 CM은 데이터 릴레이 포인트가 있더라도 더많은 에너지 소모가 필요한 직접 링크를 사용해야 하는 경우가 있다. 그러므로 그림 10과

그림 11은 데이터 보고 지연시간과 에너지 절약 사이에 trade-off가 있음을 보여준다.

단계 I에서 CH로의 직접통신을 사용하는 것과 달리, 단계 II에서 이웃 CM의 릴레이 포인트로 선택된 대부분의 CM들은 릴레이 데이터를 수신하기 위해 일정량의 여분의 에너지소모가 필요하고 이로 인해 CH와 가까운 위치에 있는 센서들은 단계 I에서의 경우보다 약간 더 많은 에너지를 소모하게 된다. 이런 결과는 그림 10과 그림 11의 교차점에서 보여준다.

그림 12와 그림 13은 단계 I과 단계 II 사이에 CH의 잔존 에너지 수준에서 큰 차이를 보여준다. 그 결과는 CM이 CH와 직접 통신하는 OCS와 같은 프로토콜과 비교하여 TPC 방식에서의 클러스터 헤드 순환 횟수가 훨씬 더 작다는 것을 의미한다. 이것은 단계 II에서 대부분의 CM이 데이터 릴레이 포인트로 자신이 센싱한 데이터를 포워드하여 CH와 직접 통신하는 센서의 수가 매우 작아지기 때문이다. 따라서 CH는 더 적은 양의 센싱 데이터를 수신하게 되고 이것은 CH의 전체적인 워크로드가 CH와 직접 통신하는 CM의 수에 따라 감소한다는 것을 의미한다. 그 결과, 단계 II에서의 모든 CH의 잔존 에너지 수준이 서로 크게 다르지 않음을 그림 12와 그림 13에서 보여준다. 주파수 반경이 클수록, 클러스터내의 센서의 수도 많아지고 이것은 CM이 데이터 릴레이를 발견할 확률을 매우 크게 한다. 그러므로 그림 12는 단계 I과 비교하여 단계 II에서의 에너지 절약효과가 주목할만하게 향상되었음을 보여준다. 5-릴레이/지연시간 제한요건에 대한 단계 II의 그래프는 에너지 절약효과와 데이터 보고 지연시간 사이에 trade-off가 있음을 보여준다.

8. 결론

이 논문에서는 클러스터 기반의 지연시간 적응형 데이터 수집 방법(CD-DGS)을 소개하고 이를 위해 적용한 두 단계 클러스터링 (TPC) 방법을 설명하였다. 이 논문의 목표는 네트워크 생존시간을 극대화하기 위해 시간/상황에 따라 변하는 사용자의 지연시간 제한 요건에 대한 높은 적응성을 보이는 에너지 효율형 센서스케줄링을 제공하는 것이다. 이 목표를 달성하기 위해 제안된 방법은 두단계 네트워크 클러스터링이다. 클러스터를 구성하는 단계 I에서는 네트워크가 클러스터로 나뉘어지고 클러스터 내의 각 CM과 CH사이에는 직접링크를 형성하고 단계 II에서는 에너지 절약을 위해 각 CM이 클러스터 내에서 CH보다 더 가까운 이웃노드를 찾아 릴레이 링크를 형성한다.

직접 링크는 제어 메시지 전달이나 시간에 민감한 센서 데이터 수집에 사용되며 릴레이 링크는 에너지 소모를 최소화하기 위해 지연시간 제한 요건이 없는 데이터 수집을 위해 사용된다. 센서 데이터 수집에 대한 직접 링크와 릴레이 링크의 사용은 사용자/응용의 데이터 보고 지연시간 요구사항에 따라 융통성 있게 조절될 수 있다. 시뮬레이션 결과는 센서가 사용자의 시간에 따른 변화와 응용에 따른 지연시간 제한요건에 적응하도록 하는 CD-DGS가 고밀도 센서 분포인 경우 상당한 양의 에너지 절약효과를 보여줌을 나타낸다. 이런 적응성은 CH의 워크로드를 감소시키고 CM으로부터 센싱 데이터를 수집하는 동안 클러스터 유지에 드는 오버헤드도 감소를 유도한다.

향후 CH에서 사용자의 데이터 보고 시간 요구사항에 따라 최대 릴레이수를 계산하는 방법을 개발하여 제안된 TPC 방식에 포함시키는 연구를 진행할 계획이다.

참 고 문 헌

1. I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci (2002), Wireless Sensor Networks: a Survey. *Computer Networks*, Vol. 38, No. 4, pp. 393-422.
2. A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson (2002), Wireless Sensor Networks for Habitat Monitoring. In *Proceedings of ACM Workshop on Wireless Sensor Networks and Applications*, pp. 88-97.
3. G. J. Pottie and W. J. Kaiser (2000), Wireless Integrated Network Sensors. *Communications of the ACM*, Vol. 43, No. 5, pp. 51-58.
4. S. Ghiasi, A. Srivastava, X. Yang, and M. Sarrafzadeh (2002), Optimal Energy Aware Clustering in Sensor Networks. <http://www.mdpi.net/sensors>, Vol. 2, No. pp. 258-269.
5. C. Intanagonwivat, R. Govindan, and D. Estrin (2000), Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks. In *Proceedings of ACM MOBICOM*, pp. 56-67.
6. W. Choi, S. K. Das, and K. Basu, (2004), Angle-based Dynamic Path Construction for Route Load Balancing in Wireless Sensor Networks. In *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 2474-2479.
7. X. Hong, M. Gerla, and H. Wang, (2002), Load Balanced, Energy-Aware Communications for Mars Sensor Networks. In *IEEE Aerospace*, Vol. 3, pp. 1109-1115.
8. A. Scaglione and S. D. Servetto, (2002), On the Interdependence of Routing and Data Compression in Multi-Hop Sensor Network. In *Proceedings of ACM Mobile Computing and Networking*, pp. 140-147.
9. M. Gerla, T. J. Kwon, and G. Pei, (2000), On Demand Routing in Large Ad Hoc Wireless Networks with Passive Clustering. In *Proceedings of IEEE WCNC*, Vol. 1, pp. 100-105.
10. J. Gao, L. L. Guibas, J. Hershberger, L. Zhang, and A. Zhu (2001), Discrete Mobile Centers. In *Proceedings of ACM Computational Geometry*, pp. 188-196.
11. S. Lindsey, C. Raghavendra, and K. M. Sivalingam, (2002), Data Gathering Algorithm in Sensor Networks Using Energy Metrics. *IEEE Transactions on Parallel and Distributed Systems*, Vol. 13, No. 9, pp. 924-932.
12. N. Bulusu, J. Heidemann, and D. Estrin, (2000), GPS-less Low Cost Outdoor Localization for Very Small Devices. *IEEE Personal Communications*, Vol. 7, No. 5, pp. 28-34.
13. I. Stojmenovic, (2002), Position-Based Routing in Ad Hoc Networks. *IEEE Communications*, Vol.40, No. 7, pp. 128-134.
14. W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, (2002) An Application-specific Protocol Architecture for Wireless Microsensor Networks. *IEEE Transactions on Wireless Communications*, Vol. 1, No. 4, pp. 660-670.
15. F. Zhao and L. Guibas, (2004), *Wireless Sensor Networks: An Information Processing Approach*. Morgan Kaufmann.
16. S. Banerjee and A. Misra, (2002), Minimum Energy Paths for Reliable Communication in Multi-hop Wireless Networks. In *Proceedings of ACM Mobile Adhoc Network Symposium (MOBIHOC)*, pp. 145-156.
17. A. Nasipuri and K. Li (2002), A Directionality based Location Discovery Scheme for Wireless Sensor Networks. In *Proceedings of ACM Workshop on WSNA*, pp. 105-111.
18. Younis, O and Fahmy, S. (2004) Efficient, Distributed Clustering Approach for Ad Hoc Sensor Networks. *IEEE Transactions on Mobile Computing*, Vol. 3, No. 4, pp. 366-379.
19. R. Virrankoski and A. Savvides (2005) TACS: Topology Adaptive Spatial Clustering for Sensor Networks. In *Proceedings of IEEE Int'l Conference on MASS*, pp. 1-10.
20. M. Lotfinezhad, B. Liang, and E. S. Sousa (2007) Adaptive Cluster-Based Data Collection in Sensor Networks with Direct Sink Access. *IEEE Transactions on Mobile Computing*, Vol. 7, No. 7, pp. 884-897.



최 욱 (twchoi@hufs.ac.kr)

2005 텍사스 주립대 (알링톤) 컴퓨터공학과 석사/박사
2005~2007 삼성전자 통신연구소 책임 연구원
2007~2009 한국마이크로소프트 R&D 센터 수석연구원
2009~현재 한국외국어대학교 컴퓨터공학과 조교수

관심분야 : Wireless Networking & Mobile Computing, Wireless Ad Hoc/Sensor Networks, Ubiquitous Service Development



이 용 (yleehyun@cjnu.ac.kr)

1997 연세대학교 컴퓨터공학과 석사
2001 연세대학교 컴퓨터공학과 공학박사
2001~2003 한국정보보호진흥원 선임연구원
2004~2005 코벨대학교 방문연구원
2005~2007 삼성전자 통신연구소 책임연구원
2007~현재 충주대학교 전자통신공학전공 조교수

관심분야 : Mobile and Wireless Security, Wireless Sensor Network, Wireless Mesh Network



정 유 진 (chungyj@hufs.ac.kr)

1997 서울대학교 컴퓨터공학과 학사/석사/박사
1997~1999 서울대학교 컴퓨터신기술연구소 특별연구원
1999~2001 부산대학교 컴퓨터및정보통신연구소 기금교수
2001~현재 한국외국어대학교 컴퓨터공학과 교수

관심분야 : Wireless Sensor Network, Embedded System, Computer Security, Algorithm, Bioinformatics