논문 2008-45CI-6-10

클러스터 헤드의 변동 거리를 고려한 클러스터 헤드 선출 기법

(Cluster Head Selection Scheme Using Fluctuating Distance of Cluster Head)

김 진 수*, 최 성 용*, 한 승 진**, 최 준 혁***, 임 기 욱****, 이 정 현*****

(Jinsu Kim, Seongyong Choi, Seungjin Han, Junhyeog Choi, Keewook Rim, and Junghyun Lee)

요 약

기존의 클러스터 라우팅 방식은 클러스터 헤드를 선출하여 클러스터 내의 멤버 노드들로부터 정보를 수집하고 압축하여 기지국에 전송함으로써 에너지 효율을 높일 수 있는 대표적인 방식이다. 그러나 클러스터 형성 단계 중 매 라운드마다 셋업 단계에서 선출된 클러스터 헤드와 클러스터 내의 멤버 노드들 간의 빈번한 정보 교환으로 인해 발생하는 불필요한 에너지 소모는 클러스터 라우팅 방식이 해결해야 하는 과제이다. 본 논문에서는 셋업 단계에서의 선출된 클러스터 헤드와 기존의 클러스터 헤드 사이에 변경되지 않는 중첩된 영역에 속한 멤버 노드들을 계산함으로써, 중첩된 멤버 노드들의 셋업 단계에서의 불필요한 송수신 횟수를 줄여 정보 교환을 최소화하였다. 또한 최적의 클러스터 구성을 위해 상위 클러스터 헤드의 방향성을 고려하였다. 따라서 셋업 단계에서의 소모되는 에너지를 절약하여 안정 단계에서 효율적으로 사용함으로써, 에너지의 효율적인 사용과 전체적인 네트워크의 생존시간을 증가시키고자 하며, 전체 네트워크내의 멤버들에게 클러스터 헤드가 될 수 있는 균등한 기회를 주고자 하는 클러스터 헤드 선출 기법을 제안한다.

Abstract

Traditional cluster-based routing method is a representative method for increasing the energy efficiencies. In these cluster-based routing methods, the selected cluster head collect/aggregate the information and send the aggregated information to the base station. But they have to solve the unnecessary energy dissipation of frequent information exchange between the cluster head and whole member nodes in cluster. In this paper, we minimize the frequency of the information exchange for reducing the unnecessary transmit/receive frequencies as calculate the overlapped area or number of overlapped member nodes between the selected cluster head and previous cluster head in the setup phase. And besides, we consider the direction of super cluster head for optimal cluster formation. So, we propose the modified cluster selection scheme that optimizes the energy dissipation in the setup phase and reuses the saved energy in the steady phase efficiently that prolongs the whole wireless sensor network lifetime by uniformly selecting the cluster head.

Keywords: Wireless Sensor Network, Cluster-based Routing, Cluster Head Selection, Sleep Mode

* 학생회원, ***** 평생회원, 인하대학교 컴퓨터정보공학부
(Department of Computer Science & Information Engineering, Inha University)
** 정회원, 경인여자대학 정보미디어학부
(School of Information & Media, Kyungin Women's College)
*** 정회원, 김포대학 e-비즈니스과
(School of Management & Tourism, Kimpo College)
**** 평생회원, 선문대학교 컴퓨터정보학부
(Department of Computer & Information Science, Sunmoon University)
※ 본 연구는 지식경제부 및 정보통신연구진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음
(IITA-2008-C1090-0801-0020)
접수일자: 2008년10월10일, 수정완료일: 2008년10월30일

Ⅰ. 서 론

무선 센서 네트워크(WSN, Wireless Sensor Network)는 특정 현상을 모니터링 하고자 일정한 지역에 분포한 무선 노드들로 구성된 네트워크이다. 센서 노드들은 주변 현상을 인식하고, 측정된 데이터를 기지국(Base Station, BS)에 전송하며 기지국에서는 수집된 데이터를 분석한다. 이러한 무선 센서 네트워크에서의 가장 큰 제약사항은 제한된 자원을 들 수 있으며, 에너지를 효율적으로 사용하기 위한 대표적인 방법으로 데이터 통합을 통한 중복된 데이터들을 압축하여 통신비용을 줄이는 클러스터 기반의 라우팅 방법이 있다. 클러스터 기반의 라우팅 방법은 클러스터 헤드를 선출하여 클러스터 내의 멤버 노드들로부터 정보를 취합, 압축하여 기지국에 전송함으로써 에너지 효율을 높일 수 있는 대표적인 방식이다. 이러한 대표적인 클러스터 기반의 라우팅 방법들 중 LEACH(Low Energy Adaptive Clustering)[1]는 확률 함수를 이용하여 매 라운드마다 효율적인 클러스터 헤드를 선정할 수 있지만, 매 라운드마다 동일한 클러스터 헤드의 수를 보장할 수 없다는 단점이 있다. 이에 비해 LEACH-C[2]는 LEACH와 동일한 클러스터 기반이지만 기지국의 통제 하에 클러스터 헤드를 선정함으로써 클러스터 헤드의 수를 보장 받을 수 있는 장점이 있다. 그러나 센서 네트워크에 참여하는 각 센서 노드들의 정보를 기지국에 전송해야 하고, 또한 노드들의 거리를 계산하여 평균 거리가 가장 짧은 센서 노드를 찾아야 하는 NP-Hard Problem에 속한다. 이러한 문제를 해결하기 위해 Simulated Annealing Algorithm[3]을 사용하여 효율적으로 클러스터 헤드를 선정하지만, 최적의 값을 보장할 수 없고, 위치를 파악하기 위한 오버헤드가 많이 필요로 한다.

본 논문에서는 네트워크 내의 노드 배치가 일정하지 않은 실제 환경에서 클러스터 헤드 선출시, 매 라운드마다 클러스터 헤드와 멤버 노드들 간의 정보 교환을 통해 발생하는 부하를 줄임으로써 에너지 소모를 최소화하기 위한 방법을 제안한다. LEACH와 같은 기존의 클러스터 헤드 선출은 매 라운드마다 전체 네트워크의 노드들의 확률 함수를 적용하여 수행하거나, LEACH-C처럼 기지국이 최적의 클러스터 헤드를 선출하였지만, 본 논문에서는 기존의 클러스터 헤드가 가진 정보와 각 멤버들로부터 획득한 정보를 이용하여 선출하는 방법을 제안한다. 즉, 각 멤버 노드들과의 거리(Distance), 에너지 잔여량(Energy), 밀도(Density), 그리고 상위 클러스터 헤드로부터 획득한 방향성(Direction)과 같은 정보 등을 이용하여 클러스터 헤드 후보 리스트를 작성하고, 이러한 후보 리스트 중에서 확률 함수를 이용하여 클러스터 헤드를 선출함으로써, 후보들끼리 경쟁을 통해 발생되는 빈번한 에너지 소모와 기존 클러스터내의 보존된 멤버들의 반복적인 송수신을 줄여 에너지를 효율적으로 사용한다. 다시 말해, 센서 노드 자신의 주위에 있는 센서 노드들은 같은 클러스터에 속할 가능성을 많이 가지고 있기 때문에, 기존의 클러스터 헤드는 선출된 클러스터 헤드에게 이전 클러스터 내에서 사용되었던 자신의 정보를 상속시켜, 기존 클러스터에 속했던 변동되지 않은 멤버 노드들은 매 라운드마다 셋업 단계에서 수행했던 불필요한 송수신 행위를 최소화하고, 새롭게 추가된 멤버 노드들만 기존의 수행 절차를 통해 전체 네트워크의 에너지 효율과 이에 따른 네트워크 생존 시간을 증대시키는 새로운 클러스터 헤드 선출 기법을 제안한다.

Ⅱ. 관련 연구

LEACH 프로토콜은 네트워크 노드들 간의 에너지 소모를 균등하게 함으로써 네트워크의 생존시간을 최대화하려 하였고, 클러스터 헤드는 식 (1)과 같은 확률함수에 의해 결정된다.

$$P_i(t)=\begin{cases}\dfrac{k}{N_{total}-k\times(r \bmod \frac{N_{total}}{k})} & : C_i(t)=1 \\ 0 & : C_i(t)=0\end{cases} \quad (1)$$

여기서 i는 노드의 식별자, t는 시간, N_{total}은 전체 노드의 수, k는 클러스터의 수, r은 라운드를 나타낸다. 그러나 클러스터 헤드를 선출하는 셋업 단계에서 멤버 노드들과의 빈번한 정보의 교환으로 인해 에너지 소비가 많이 발생하는 오버헤드를 가졌으며, 확률 함수에 의한 클러스터 헤드를 선출하다 보니 균등하지 않게 클러스터들이 형성될 수 있는 문제점을 가지고 있다. 비록 LEACH-C에서는 이러한 문제점을 해결하기 위해 Simulated Annealing Algorithm을 사용하여 효율적으로 클러스터 헤드를 선정하였지만 최적의 값을 보장할 수 없고, 노드 전체의 에너지와 자신의 현재 에너지를 고려하여 클러스터 헤드를 결정하기에는 추가적인 오버헤드가 필요하다는 단점을 가지고 있다.

HEED[4] 프로토콜은 클러스터 헤드를 선출하기 위해 각 멤버 노드 자신의 인자만을 이용한다는 점에서 선출

기법이 우수하다. 클러스터 헤드 선출 시 사용되는 식은 멤버 노드들이 가진 잔여 에너지를 이용하며, 식 (2)와 같은 확률 함수를 이용한다.

$$CH_{prob} = C_{prob} \times \frac{E_{residual}}{E_{\max}} \quad (2)$$

$E_{\max}$는 멤버 노드가 가진 초기 에너지, $E_{residual}$은 멤버 노드의 현재 잔여 에너지, C_{prob}는 전체 네트워크 노드 중 클러스터 헤드의 비율(예를 들어, 5%)을 의미한다. HEED 알고리즘은 초기의 C_{prob}와 절대값 $P_{\min}$(0.0001) 중 큰 값으로 시작하여 CH_{prob}이 1이 될 때까지 CH_{prob}에 2를 곱하거나 1이 된 이웃 노드들로부터 메시지를 수신할 때까지 반복하여 클러스터 헤드를 선출하도록 한다. 이것은 모든 노드들의 에너지를 알 필요 없이 오직 노드 자신의 인자만을 이용하여 클러스터 헤드를 선출할 수 있으며, 일정 반복 횟수를 수행한 후 종료되는 장점을 가지고 있다. 그러나 선출 확률 값에 2배씩 반복 수행함으로써 노드들이 가진 에너지의 양이 유사할 경우 클러스터 헤드의 수를 보장하지 않으며, 반복해서 ADV(Advertisement) 메시지들을 전송하기 때문에 노드들의 송수신에서 발생하는 에너지 소모에 커다란 오버헤드를 갖는 문제점을 가지고 있다.

ACE(Algorithm for Cluster Establishment)[5]는 노드의 연결도를 고정된 수만큼의 반복을 통해 최소 클러스터를 구성하려 하였다.

T. J. Kwon et al.[6]은 클러스터 구성에 따른 오버헤드가 없는 수동 클러스터링 (Passive Clustering)을 제안하였고, 이는 이전의 클러스터링 구성 방법이 일정 시간 주기로 클러스터 구성을 위한 필요한 메시지를 제거하기 위해 데이터 전송 시, 클러스터 구성 메시지를 데이터에 부가(Piggyback)하여 클러스터를 구성하고자 하였다.

Ⅲ. 3DE 알고리즘을 고려한 클러스터 헤드 선출

본 논문에서 제안된 3DE(Direction, Distance, Density, and Energy) 알고리즘을 이용하여 클러스터 헤드 선출 기법은 기존 클러스터 기반의 라우팅 방법에서 사용되는 두 단계인 셋업 단계와 안정(Steady-State) 단계 중 셋업 단계를 변형함으로써 에너지를 효율적으로 사용하고, 전체 네트워크의 클러스터의 수를 균등하게 분배하기 위해 방향성을 적용함으로써 전체 네트워크 생존시간을 증가시키고자 한다.

1. 변형된 셋업 단계

기존의 클러스터 기반의 방법에 사용되는 셋업 단계는 다음과 같다.

① 선출된 클러스터 헤드는 ADV(Advertisement) 메시지를 멤버 노드들에게 광고하고, ② ADV 메시지를 받은 클러스터 내 멤버 노드들은 수신된 ADV 메시지들 중 하나의 클러스터 헤드를 결정한 후, Join-REQ 메시지를 클러스터 헤드에 재전송한다. ③ 클러스터 헤드는 수신된 정보를 이용하여, TDMA 슬롯을 결정하고, 멤버 노드들의 목록을 작성하여, 그 결과를 모든 멤버 노드들에게 브로드캐스팅하고, 멤버 노드들은 이 정보를 수신하고, Steady-State 단계를 준비함으로써 셋업 단계를 종료한다.

본 논문에서 제안하는 셋업 단계는 다음과 같다.

① 클러스터 내 멤버 노드들에게 기존의 클러스터 헤드(CH_{old})가 새로운 클러스터 헤드(CH_{new})가 선출되었음을 멤버들에게 알리고, ② 이를 수신한 멤버 노드들은 그 정보를 이용하여, 자신의 상태를 결정한다. 즉, 클러스터가 변경되지 않는 멤버 노드들은 셋업 단계가 종료될 때까지의 일정 시간 동안을 Sleep 상태가 되며, 삭제되는 멤버 노드들은 새로운 클러스터 헤드를 선택할 준비를 하고, ③ 선출된 클러스터 헤드는 멤버 노드들에게 ADV 메시지를 전송하고, 새로 추가된 멤버 노드들은 Join-REQ 메시지를 전송하고, ④ 수신된 정보를 이용하여 기존의 방법과 유사하게 TDMA 슬롯을 결정하고, 멤버 노드들의 목록을 작성하며, 그 결과를 새로 추가된 멤버 노드들은 이 정보를 수신하여, 안정 단계를 준비함으로써 셋업 단계를 종료한다. 여기서 제안된 방법은 셋업 단계에서 변경되지 않는 멤버 노드들이 셋업 단계가 종료되기까지 Sleep 상태가 되고, 정보 교환에 참여하는 송수신 노드들의 수 차이만이 있을 뿐 동일하게 적용된다. 또한 변경되지 않는 노드들은 기존 클러스터에서 사용된 정보들(TDMA slot과 CDMA spreading 등)을 사용하며, 이 정보는 상속을 통해 전달되고, 삭제된 멤버 노드들의 정보를 새로 추가된 멤버 노드들이 대체함으로써 재사용을 통한 효율을 증가시키고자 한다. 표 1은 기존의 셋업 단계와 제안된 셋업 단계의 각 단계에 필요한 에너지의 양을 수식으로 나타낸 것이다.

표 1에서 N은 클러스터내의 전체 노드 수, n은 클러스터 내에 새로 추가되는 멤버 노드 수를 나타내고, E_{Tx} 는 CH_{old}에서 CH_{new}로 상속에 필요한 정보를 전송하는 것이다. 기존의 셋업 단계에서 각 멤버 노드들

표 1. 셋업 단계에서 소모되는 에너지 비교
Table 1. Energy comparison with traditional setup phase and proposal setup phase.

step	기존의 셋업 단계 ($Setup_{trad}$)	제안된 셋업 단계 ($Setup_{prop}$)
1	$E_{Tx}+(N-1)E_{Rx}$	$E_{Tx}+(N-1)E_{Rx}$
2	$(N-1)E_{Tx}+(N-1)E_{Rx}$	$(N-n-1)E_{sleep}+E_{Tx}^{'}$ $+E_{Tx}+nE_{Rx}$
3	$E_{Tx}+(N-1)E_{Rx}$	$nE_{Tx}+nE_{Rx}$
4	-	$E_{Tx}+nE_{Rx}$

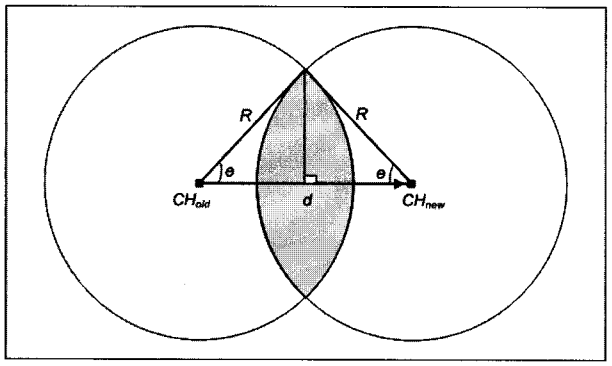


그림 1. 클러스터 헤드 변동에 따른 중첩된 영역
Fig. 1. The overlap area between CH_{old} and CH_{new}.

이 다른 클러스터 헤드로부터 ADV 메시지를 수신하였을 경우는 배제하였다. 주변의 클러스터 헤드로부터 수신된 ADV 메시지의 순서가 노드에 가장 가까운 클러스터 헤드이기 때문에, 본 논문에서는 수신된 ADV 메시지들 중 에너지의 크기가 큰 클러스터 헤드에 Join-REQ 메시지를 송신하는 기존의 방법과 달리, 먼저 도착한 ADV 메시지의 클러스터 헤드에 Join-REQ 메시지를 송신하여 지연시간을 최소화하였다. 표 1에서 사용된 송신(E_{Tx}) 및 수신(E_{Rx})에 필요한 세부적인 에너지의 양은 식 (3)과 식 (4)와 같이 결정된다[7].

$$E_{Tx}(l,d)=E_{Tx-elec}(l)+E_{Tx-amp}(l,d)$$
$$=\begin{cases} lE_{elec}+l\epsilon_{fs}d^2 & ,d<d_0 \\ lE_{elec}+l\epsilon_{mp}d^4 & ,d\geqq d_0 \end{cases} \quad (3)$$

$$E_{Rx}(l)=E_{Rx-elec}(l)=lE_{elec} \quad (4)$$

2. 최적의 에너지 소모를 위한 변경되지 않는 멤버 노드들의 수 계산

표 1에 보인 것과 같이, 기존의 방법과 제안된 방법 간의 셋업 단계에서의 에너지 소모를 줄이기 위한 최적화된 방법을 찾기 위해, 클러스터 헤드의 변동 거리와 변동 거리에 따른 변경되지 않는 멤버 노드들의 수를 이용하여 에너지 효율을 높이려고 한다. 제안된 셋업 단계에서의 에너지 소모량($\sum_{step=1}^{4} Setup_{prop(step)}$)이 기존의 셋업 단계에 사용된 에너지 소모량($\sum_{step=1}^{3} Setup_{trad(step)}$)보다 작거나 같은 경우를 정리하면 식(5)와 같다.

$$\sum_{step=1}^{3} Setup_{trad(step)} \geqq \sum_{step=1}^{4} Setup_{prop(step)}$$
$$\frac{(N-n-3)E_{Tx}}{(3n-2N+2)E_{Rx}}+\frac{(n-N+1)E_{Sleep}}{(3n-2N+2)E_{Rx}} \geqq 1 \quad (5)$$
$$\therefore n \leqq \left\lfloor \frac{3N-5}{4} \right\rfloor$$

여기서, E_{Tx}와 E_{Rx}의 차는 식(3)과 식(4)의 차($l\epsilon_{fx}d^2$)이고, 이것은 같은 클러스터 내의 짧은 거리($d<d_0$)이기 때문에 송수신하는 메시지의 비트수가 크지 않다. 새로 추가되는 노드들의 수가 정수이기 때문에, $\frac{E_{Tx}}{E_{Rx}} \approx 1$로 치환한 식으로 정리하였다. 또한 E_{Rx}에 비해 E_{Sleep}에 사용되는 에너지의 양이 아주 작아 ($E_{Rx} \gg E_{Sleep}$), $\frac{E_{Sleep}}{E_{Rx}} \approx 0$을 적용하면 식 (5)를 얻을 수 있다. LEACH와 LEACH-C의 경우에 프로그램에 세팅된 E_{Sleep}의 값은 0으로 적용하여 실험하였다. 식 (5)를 통해 기존의 방법보다 에너지 효율을 향상시킬 수 있는 변경되지 않는 멤버 노드들의 수(m; $m=N-n$)는 식 (6)과 같다.

$$m \geqq \left\lceil \frac{N+5}{4} \right\rceil \quad (6)$$

예를 들어, 클러스터 내 멤버 노드들의 수가 20개라 가정했을 경우, 변동되지 않은 멤버 노드의 수(m)가 7개 이상일 때, 본 논문에서 제안하는 방법이 에너지 효율을 향상 시킬 수 있다는 것을 의미한다. 물론 각 클러스터 내 멤버 노드들의 수는 매 라운드마다 상황에 따라 변동될 수 있기 때문에, m의 값은 클러스터 내의 멤버 노드들의 수에 따라 동적으로 결정된다.

3. 클러스터 헤드의 변동 거리에 따른 변경되지 않는 중첩된 부분의 멤버 노드 수 계산

한 라운드가 변경이 될 때마다 전체 센서 네트워크는 셋업 단계를 통해 클러스터를 재구성해야 한다. 그림 1

은 기존의 클러스터 헤드(CH_{old})가 새로운 클러스터 헤드(CH_{new})로 변동되었을 때, 변경되지 않는 중첩된 부분을 보여준다. CH_{old}와 CH_{new}가 동일한 클러스터 반경(R)을 가졌다고 가정했을 때, 겹치는 부분이 다른 클러스터로 변경되지 않는 멤버 노드들의 영역을 나타내며, 이러한 중첩된 영역($Area(R,d)$)을 계산하면, 다음 식 (7)과 같다.

$$
\begin{aligned}
&Area(R,D)\\
&=\left(\pi R^2\times\frac{acos(\frac{d}{2R})\times\frac{180}{\pi}}{360}-\frac{\frac{d}{2}\times sin(acos(\frac{d}{2R}))R}{2}\right)\times 4\\
&=\left(2R^2\times acos(\frac{d}{2R})-d\times sin(acos(\frac{d}{2R}))R\right) \qquad (7)
\end{aligned}
$$

여기서, θ는 각도(degree)이고, d는 CH_{old}와 CH_{new} 사이의 거리이고, R은 클러스터 헤드의 반경으로 새로운 클러스터 헤드의 반경과 기존의 클러스터 헤드의 반경은 같다고 가정하였다. 전체 네트워크의 멤버 노드들의 분포가 균등 분포라고 가정하였을 때, 식 (7)로부터 계산된 중첩된 영역을 하나의 노드가 분포될 수 있는 평균 영역($M^2/Node_{total}$)으로 나누면, 중첩된 영역에 분포될 수 있는 멤버 노드 수($Node_{overlap}$)를 구할 수 있고, 식은 다음과 같다.

$$Node_{overlap}=\frac{Node_{total}*Area(R,d)}{M^2} \qquad (8)$$

이때, M^2은 센서 네트워크의 전체 영역이며, $Node_{total}$ 은 센서 네트워크 내에 분포한 전체 노드수를 의미하고, $Node_{overlap}$은 식 (6)의 m과 같다.

클러스터 헤드들 사이의 변경되는 거리, 즉 CH_{old}와 CH_{new}의 거리(d)와 반경(R)이 주어지게 되면, 식 (7)과 식 (8)의 계산식은 즉시 계산 가능한 시간복잡도 $O(1)$을 갖는다.

4. 클러스터 내의 밀도와 Cost 결정

본 논문에서는 클러스터 헤드 결정에 영향을 끼치는 선출 비용인 Cost를 결정하기 위해 밀도(Density)의 개념을 사용하였다. 밀도는 한 클러스터 내에서 각 멤버 노드와 클러스터 헤드 사이의 거리에 따라 다르며, 각 멤버 노드로부터 일정 거리, 즉 클러스터 헤드가 포함할 수 있는 반경(R)에 속한 동일한 클러스터 내의 이웃 노드 수($Node_{neighbor}$)와 다른 클러스터에 포함된 외래 노드 수($Node_{foreign}$)의 비율을 말하며, 다음 식과 같이 정의하였다.

$$Density_i=\frac{Node_{neighbor}}{Node_{neighbor}+Node_{foreign}} \qquad (9)$$

여기서, 멤버 노드 i가 자신이 속한 클러스터 헤드에 가까우면 가까울수록 $Density_i$의 값은 커지고, 멀어지면 멀어질수록 작아지며, 이 값의 범위는 0과 1 사이로 정규화하였다. 시간이 경과함에 따라 멤버 노드들은 가지고 있던 모든 에너지를 소모하고, 생명 주기를 마치게 된다. 따라서 네트워크의 유지 시간이 경과함에 따라 생존 노드 수($Node_{alive}$)는 작아지고, 불능 노드 수($Node_{dead}$)는 커지기 때문에 밀도에 영향을 미친다. 그러나 기존의 클러스터링 방법[1~2, 4]에서와 같이 시간에 따른 생존 노드 수는 불능 노드가 발생한 후 일정 한계를 넘어서는 시점부터 급격히 감소하여 전체 네트워크의 수명 기간을 마치게 된다. 따라서 효율적인 에너지 관리를 위한 밀도 계산은 클러스터내의 생존 노드 수와 잔여 에너지가 각각 한계값 이상일 경우에 각 멤버 노드의 정보와 현재의 클러스터 헤드(CH_{new}) 정보를 식 (7)과 식 (8)에 적용하면 빠르게 계산된다. 한 라운드의 셋업 단계에서 클러스터 헤드 선출을 위해서는 전체 네트워크 내의 모든 멤버 노드들이 공평한 기회를 주고, 에너지 소모를 줄일 수 있는 멤버 노드들을 찾기 위해서 계산되어야 하는 것이 Cost이며, 각 멤버 노드들로부터 클러스터 헤드로 전송 받는 정보이다. LEACH-C인 경우, 평균 에너지 이상의 에너지를 가진 노드들 중에서 클러스터 헤드와 멤버 노드들 사이의 거리의 총합이 최소가 되는 노드를 Simulated Annealing Algorithm을 이용하여 결정하였다.

본 논문에서는 Cost에 영향을 주는 주요 인자들로 밀도와 잔여 에너지량을 이용하여 결정하였다. CH_{old}가 획득한 정보들 중, 먼저 평균 에너지 이상을 가진 클러스터 내 노드들을 선별하여 리스트를 생성하고, 선별된 노드 리스트들 중에서 $Density_i$의 값이 한계값 이상인 노드들만 선별하여 최종 후보 리스트를 생성한다. 이때 사용된 한계값은 식(6)에서 획득한 값(m)이다.

5. 클러스터 헤드 선출 알고리즘(3DE algorithm)

클러스터 헤드의 선출에서 LEACH는 확률 함수를 이용하여 결정하였고, HEED 프로토콜은 개별 노드에서 자신의 잔여 에너지를 이용한 분산 처리를 통해 선

출하는 알고리즘을 제안하였다. 클러스터 헤드 선출을 위해 확률 함수나 분산된 노드들의 자율적인 선출을 통해 결정하면, 네트워크 내의 모든 멤버 노드들에게 균등하게 클러스터 헤드의 기회를 줄 수 있다는 장점은 있지만, 클러스터 헤드의 최적화된 위치를 선택하기가 어렵고, 비슷한 에너지를 가진 후보 클러스터 헤드들의 경쟁에 의한 멤버 노드들의 수신 빈도수가 많아져 막대한 에너지 낭비가 발생하게 된다. 따라서 본 논문에서는 클러스터 헤드와 멤버 노드들로부터 발생하는 정보들(*Cost, NodeID*, 잔여 에너지량 등)과, 거리, 밀도, 그리고 방향을 이용하여 후보 클러스터 헤드 집합을 구성한다. 방향은 계층적 클러스터 환경에서 이전의 클러스터 헤드가 새로운 클러스터 헤드를 선출하기 위해 상위 클러스터 헤드(SCH)의 방향을 참조하여 결정하는 하나의 인자이다. 변동 방향을 참조하는 이유는 클러스터에 포함되지 않는 멤버 노드들의 수를 적게 하고, 클러스터와 클러스터 사이의 중첩되는 영역을 가능한 최소화하기 위해서이다. 상위 클러스터 헤드가 다음 상위 클러스터 헤드의 방향을 결정하면, 결정된 방향 정보를 멤버 클러스터 헤드들에게 전송하고, 수신한 클러스터 헤드들(CH_{old})은 새로 선출할 클러스터 헤드 후보들 중 가장 비슷한 방향을 갖고, Cost가 작은 최종 후보 리스트들을 선택한다. 여기서, 전체 네트워크 내의 모든 멤버 노드들에게 클러스터 헤드가 균등하게 선택되도록 하기 위해, 가장 작은 Cost값을 갖는 멤버 노드를 선출하지 않고, 난수를 발생시켜 임의적으로 선택한다. 무작위로 선출하는 이유는 가장 높은 후보 클러스터 노드를 선출하면 균등한 선출이 아닌 한쪽으로 치우칠 가능성이 높게 되어, 상대적으로 클러스터 바깥쪽에 위치한 멤버 노드들이 클러스터 헤드로 선출될 가능성이 낮아진다. 따라서 특정 영역(클러스터 중심)에 속한 멤버 노드들만 선출되어 지역적 국부성이 발생되며, 바깥쪽의 멤버 노드들이 MTE 라우팅 기법[1]처럼 빠르게 불능 노드들이 발생되는 결과를 초래한다.

그림 2는 본 논문에서 제안한 클러스터 헤드 선출을 위한 3DE 알고리즘을 보여준다. 여기서, k는 k번째 클러스터의 번호를 의미하고, num_Cluster(k)함수는 k번째 클러스터에 속한 멤버 노드들의 수를 의미한다. 그리고 Vector()함수는 클러스터에 속한 노드와 현재 클러스터 헤드(CH_{old})를 가지고 벡터로 변환하고, 또한 이전 상위 클러스터 헤드(SCH_{old})와 선출된 상위 클러스터 헤드(SCH_{new})로부터 벡터를 만들어, Sim() 함수의

```
num_Candidate ← 0, cnt ← 0;
CandidateList ← ∅;
CostList ← ∅;
for(i=0; i<num_Cluster(k);i++)
  if(Sim(Vector(Node_i,CH_old),Vector(SCH_old,SCH_new))>=0)
      if(AvgEnergy(k) <= Energy(Node_i))
          CandidateList[num_Candidate++] ← Node_i;
for(i=0;i<num_Candidate;i++)
    if(numOverlap(CH_old,CandidateList[i]) >= m)
        CostList[cnt++] ← CandidateList[i];
if( cnt > 0){
    ClusterHead ← CostList[ rand() % cnt ];
    ModSetup();
}
else
    PrevSetup();
```

그림 2. 클러스터 헤드 선출을 위한 3DE 알고리즘
Fig. 2. 3DE algorithm for cluster head selection.

두 매개변수로 사용함으로써 두 벡터 사이의 사이각(θ)의 유사도를 측정하는 함수이다. 본 논문에서 설정한 두 벡터 간의 사이각의 한계값으로 $\pm 90°$(즉, $0 \leq cos(\theta) \leq 1$)로 설정하였다. AvgEnergy(k)함수는 k번째 클러스터의 평균 에너지를 구하는 함수이고, Energy()함수는 각 멤버 노드가 가진 잔여 에너지를 구하는 함수이다. numOverlap()함수는 클러스터 헤드와 후보 클러스터 헤드들과의 중첩된 영역에 속한 멤버 노드의 수를 구하는 함수이고, m은 식 (6)의 계산된 값이며, 만일 CostList가 공집합이면, 기존의 전통적인 셋업 단계(PrevSetup() 함수)를 수행하고, 공집합이 아니면 난수(rand()함수)를 발생시켜 클러스터 헤드를 선출한 후, Ⅲ장 1절에서 제안한 셋업 단계(ModSetup() 함수)를 수행하는 알고리즘을 나타낸다.

그림 3은 본 논문에서 제안된 방법을 이용하여 셋업 단계((a)~(c))부터 안정단계((d))까지의 과정을 보여주며, 여기서 ①에서 ⑤까지는 클러스터를 구성하는 단계에 사용되는 송수신을 보이고, ⑥에서 ⑧까지의 안정단계에서 수신된 정보를 BS까지 전송하는 예를 보여준다. 에너지의 효율을 최대한 높이기 위해 클러스터 형성 과정 중 셋업 단계에서 기존의 클러스터 헤드(CH_{old})가 미리 계산한 Cost 정보와 상위 클러스터 헤드의 방향성 (Θ)을 고려하여 선출한 클러스터 헤드(CH_{new})를 선포하고(a), CH_{new}의 정보를 받은 멤버 노드들은 Sleep 모드로 전환하거나, 새로운 클러스터 헤드에 포함될 준비를 하고, CH_{new}는 새로운 멤버 노드들을 포함시키기 위해 메시지를 교환한다(b). 그리고 (c)에서는 새로

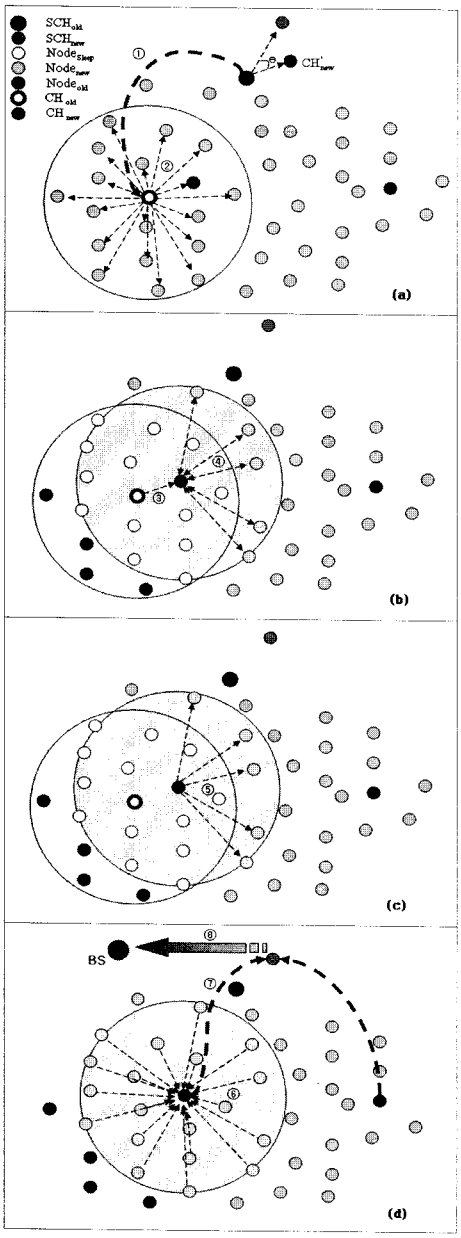


그림 3. 클러스터를 형성하는 셋업 단계와 안정 단계
Fig. 3. Setup phase and Steady-state phase for Cluster formation.

추가된 멤버 노드들에게 TDMA 스케줄링 정보를 전송하고, 안정 단계(d)로 전환하여 한 라운드를 수행한다.

Ⅳ. 실험 및 성능 평가

본 논문에서 제안한 알고리즘을 기존의 클러스터링 방법들과 성능을 평가하기 위해, NS-2[8] 시뮬레이터를 사용하여 구현하였고, 표 2는 시뮬레이션 환경에 사용된 매개변수와 그에 대응하는 값을 나타낸다. 라우팅 프로토콜의 다른 주류인 평면(flat) 라우팅 방법[9~10]처럼 일정 시간마다 BS가 브로드캐스팅하여 전체 네트워

표 2. 시뮬레이션에 사용된 매개변수와 해당 값
Table 2. Simulation parameters and values.

매개변수	값
Network grid	(0,0) ~ (100,100)
Base Station	(50, 175)
Threshold distance	75m
Cluster radius	25m
E_{elec}	$50nJ/bit$
ε fs	$10pJ/bit/m^2$
ε mp	$0.0013pJ/bit/m^4$
ε aggregation	$5nJ/bit/signal$
Data packet size	500 bytes
Broadcast packet size	25 bytes
Packet header size	25 bytes
Initial energy	2J/battery
Number of nodes	100
Number of clusters	5

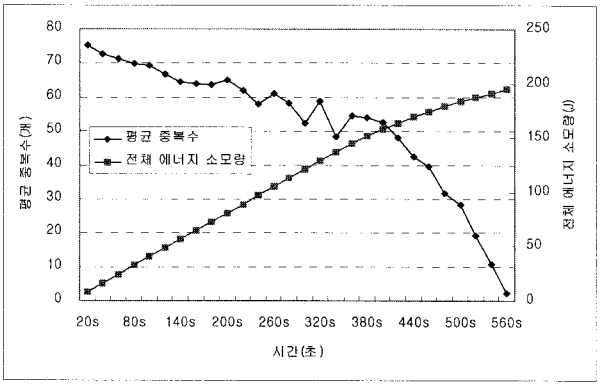


그림 4. 시간에 따른 변경되지 않는 평균 중복 노드 수와 전체 에너지량
Fig. 4. Average number of overlap nodes and total energy content.

크를 유지하듯이, 본 논문은 기존의 클러스터 기반의 방법처럼 매 라운드마다 20초를 설정하고 실험하였다.

그림 4는 무작위로 생성한 100개의 멤버노드들을 가진 10개의 테스트 집합을 LEACH-C의 클러스터 헤드를 선출하는 과정에서 측정한 변경되지 않은 평균 멤버 노드 수를 보여주며, 이러한 누적된 멤버 노드의 수가 에너지 소모를 최적화할 수 있도록 사용된다. 그림 4에서 일정 시간동안 변경되지 않는 평균 멤버 노드의 수가 큰 차이가 없다가 불능 노드의 수가 급증하는 420초 지점부터 급격히 감소하며, 이때의 평균 에너지 소모량이 약 1.64J이었다. 따라서 본 논문에서는 Ⅲ장 4절에서 설명한 불능노드와 생존 노드의 관계를 이용한 밀도 계산에서 한계값으로 0.8(1.64J/2J)을 설정하고 수행하였다. 실험에 사용된 10개의 샘플들 중 하나의 결과는 전

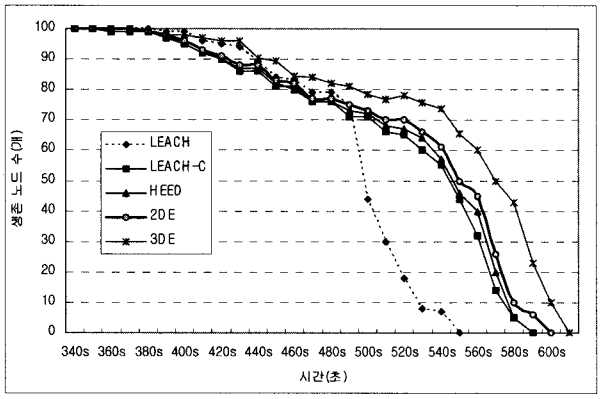


그림 5. 시간에 따라 변경되는 생존 노드의 수의 비교
Fig. 5. Comparison of number of alive nodes according to the time variables.

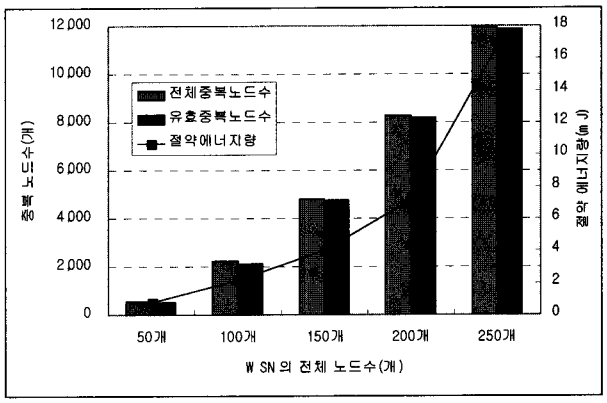


그림 6. 무선 센서 네트워크의 전체 멤버 노드수에 따른 중복된 노드수와 에너지의 비교
Fig. 6. Comparison of number of overlap nodes and saving energy content according to the number of total nodes in Wireless Sensor Network.

체 2,116개의 변경되지 않은 멤버 노드가 있었으며, 이 중에서 유효한 멤버 노드의 수는 2,073개로 표 2의 매개변수를 적용하였을 때, 총 3,612회 이상의 송수신을 동시에 할 수 있는 에너지의 양이 절약되었다.

그림 5는 시간에 따라 변경되는 생존 노드의 수를 비교한 그래프이다. 그림 5에서 보이는 것과 같이 LEACH, LEACH-C, HEED, 2DE(3DE에서 방향 제외), 그리고 3DE의 불능 노드가 처음 발생하는 시점은 각각 390초, 360초, 380초, 380초, 그리고 380초였고, 마지막 생존 노드가 발생하는 시점은 각각 547초, 580초, 590초, 613초, 그리고 619초에 발생하였다. 이것은 전체 센서 네트워크의 생존 시간을 3DE가 LEACH, LEACH-C, HEED, 그리고 2DE와 비교하여, 각각 13.2%, 6.7%, 4.9%, 1% 정도 증가되었다. 기존의 LEACH나 LEACH-C에 비해 전체 멤버 노드들의 셋업 단계에서 사용되는 불필요한 에너지를 감소시킴으로써, 전체 에너지 효율을 높일 수 있었다. 또한 HEED는 클러스터 헤드를 선출하기 위해 이웃 노드들과의 잦은 메시지 교환을 통해 정보를 송수신하여 불필요한 에너지를 낭비하였고, 본 논문에서 제안한 3DE는 방향 정보를 이용하여 전체 센서 네트워크가 균형을 맞춰 고립되는 센서 노드들을 최소로 하고, 또한 HEED와 다르게 거리에 따라 이웃 노드들의 개수를 파악함으로써 에너지를 효율적으로 사용할 수 있었다. 비록 LEACH에 비해 3DE는 기지국과의 정보 교환 또는 클러스터 내의 정보 교환으로 인한 부담으로 먼저 불능 노드가 발생하지만 시간이 흐름에 따라 생존 노드가 많이 분포되어 중첩된 멤버 노드들의 누적된 에너지로 안정 단계에서 효율적으로 사용됨을 알 수 있다.

그림 6은 동일한 크기($100m \times 100m$)를 가진 네트워크 안에서 3DE 알고리즘을 이용하여 클러스터를 구성하였을 경우, 멤버 노드 수에 따른 전체 중복수와 유효 중복수를 나타낸 것이다. 유효 중복수만이 실제 에너지 절약을 위해 사용되었으며, 전체 멤버 노드수가 증가함에 따라 유효한 멤버 노드들의 중복수가 증가됨을 알 수 있다. 특히 전체 멤버 노드수가 50개와 100개의 유효 중복수가 약 308% 증가, 100개와 150개인 경우, 약 165% 증가, 150개와 200개인 경우, 129% 증가, 그리고 200개와 250개인 경우가 107% 증가되었고, 이것은 전체 센서 네트워크를 구성하는 노드의 수가 증가하거나, 초기 에너지의 양을 증가시키면 중복되는 에너지의 양을 절약하여 에너지 효율을 최대화할 수 있음을 보였다.

클러스터를 구성하기 위해서 셋업 단계에서 CH_{old}가 CH_{new}에게 삭제된 멤버 노드들의 정보를 제공하는 시간과 CH_{new}가 그림 3 (b)에서 보인 것처럼 추가될 멤버 노드들에게 ADV 메시지를 재전송하는데 지연시간이 발생한다. 그러나 기존의 클러스터 방식과 달리 클러스터 내의 모든 멤버 노드들이 클러스터 헤드에게 Join-REQ 메시지를 전송하지 않고, 단지 새로 추가된 멤버 노드들에게만 전송하기 때문에 처리 시간이 감소하며, 매 라운드에 비해 클러스터를 구성하는 셋업 시간이 너무 짧기 때문에 기존의 방식과 비교하여 많은 지연 시간과 같은 복잡도는 증가하지 않는다. 왜냐하면 본 논문에서는 클러스터를 형성하는데 사용되는 중첩된 노드들의 수를 판별하여 새로운 방법을 적용할 것인지, 기존의 방법을 사용할지를 3DE 알고리즘에 의해 선택하기 때문에 에너지의 소모를 절약할 수 있기 때문이다.

V. 결 론

본 논문에서는 클러스터링 방법에서 각 라운드마다 수행해야 하는 셋업 단계를 상황에 맞게 선택하고, 전체 네트워크의 멤버 노드들이 고립되지 않고, 각 클러스터에 균등하게 분포시키기 위해 방향성을 고려하여 클러스터 헤드를 선출하는 변형된 셋업 방법을 제안하였다. 클러스터 헤드가 선출되었을 때, 대부분의 라운드에서 이전의 클러스터 헤드와 선출된 클러스터 헤드 사이에 중복된 멤버 노드들의 불필요한 정보교환을 최소화하여 에너지 효율성을 극대화하였다.

향후 연구 과제로 클러스터에서 삭제될 노드들을 이전 클러스터 헤드가 정보를 획득하여 선출된 클러스터 헤드에게 전송함으로써 지연 시간이 추가되어 셋업 단계의 구축 시간을 증가시켰다. 따라서 셋업 단계의 지연 시간을 최소화하고, GPS의 사용과 같은 제약사항을 극복할 수 있는 연구가 필요하며, 센서 네트워크 내의 멤버 노드들 간의 이동성까지 고려한 클러스터 헤드 선출 기법에 관한 연구가 필요하다.

참 고 문 헌

[1] W. R. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan,"Energy-Efficient Communication Protocol for Wireless Microsensor Networks," Proc. 33rd Hawaii Int'l. Conf. Sys. Sci., Jan. 2000.

[2] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An Application-Specific Protocol Architecture for Wireless Microsensor Networks," IEEE Trans. Wireless Commun., Vol. 1, no.4, Oct. 2002, pp. 66~70.

[3] T. Murata and H. Ishibuchi, "Performance evaluation of genetic algorithms for flowshop scheduling problems," Proc. 1st IEEE Conf. Evolutionary Computation, Vol. 2, pp. 812817, June 1994.

[4] O. Younis and S. Fahmy, "Distributed Clustering in Ad-hoc Sensor Networks: A Hybrid, Energy-Efficient Approach," IEEE INFOCOM 2004, Mar. 2004.

[5] H. Chan and A. Perrig, "ACE: An Emergent Algorithm for Highly Uniform Cluster Formation", In 2004 European Workshop on Sensor Networks, pp154-171, Aug. 2004.

[6] T. J. Kwon, M. Gerla, V. K. Varma, M. Barton, R. Hsing, "Efficient Flooding with Passive Clustering-An Overhead-Free Selective Forward Mechanism for Ad Hoc/Sensor Networks", Proceedings of the IEEE, Vol. 91, no. 8, pp. 1210-1220, Aug. 2003.

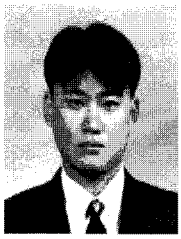
[7] T. Rappaport, Wireless Communications: Principles & Practice. Englewood Cliffs, NJ: Prentice-Hall, 1996

[8] UCB/LBNL/VINT Network Simulator ns (Version 2). http://www.isi.edu/nsnam/ns/, 1998

[9] J. Kulik, W.R. Heinzelman, H. Balakrishnan, "Negotiation-based Protocols for Dissemination in Wireless Sensor Networks," ACM Wireless Networks, Vol.8, Issue 2/3, pp.169-185, March-May, 2002.

[10] Chalermek Intanagonwiwat, Ramesh Govindan, Deborah Estrin, John Heidemann, "Directed Diffusion for Wireless Sensor Networking," IEEE/ACM Transactions On Networking, Vol.11, No.1, pp. 2-16, Feb., 2003.

저 자 소 개



김 진 수(학생회원)

1998년 인천대학교 전자계산 공학과 학사 졸업
2001년 인하대학교 전자계산 공학과 석사 졸업
2003년 인하대학교 컴퓨터정보 공학과 박사수료

<주관심분야 : 유비쿼터스 컴퓨팅, 무선 센서 네트워크, 데이터마이닝, 정보검색>

최 성 용(학생회원)

1993년 인하대학교 통계학과 학사 졸업
2001년 인하대학교 통계학과 석사 졸업
2003년 인하대학교 컴퓨터정보 공학과 박사수료

<주관심분야 : 유비쿼터스 컴퓨팅, 무선 센서 네트워크, 데이터마이닝, 정보검색>

한 승 진(정회원)

1990년 인하대학교 전자계산학과 학사
1992년 인하대학교 전자계산 공학과 석사
2002년 인하대학교 전자계산 공학과 박사
1992년~1996년 대우통신 종합연구소
1996년~1996년 한국전산원 초고속사업단
1996년~1998년 SKTelecom 디지틀사업본부
2002년~2004년 인하대학교 컴퓨터공학부 강의조교수
2004년~현재 경인여자대학 정보미디어학부 조교수

<주관심분야 : USN, MANET, Mobile Computing, 임베디드 시스템, Security>

최 준 혁(정회원)

1990년 경기대학교 전자계산학과 학사
1995년 인하대학교 전자계산 공학과 석사
2000년 인하대학교 전자계산 공학과 박사
1997년~현재 김포대학 e-비즈니스과 부교수
2001년~2002년 한국전자통신연구원 컴퓨터소프트웨어연구소(초빙연구원)
2003년~현재 특허청 특허출원 심사자문위원
2003년~현재 김포발전연구소 소장

<주관심분야 : 정보검색, 유전자 알고리즘, 신경망, USN, 임베디드 시스템, 전자상거래 보안>

임 기 욱(평생회원)

1977년 인하대학교 전자공학과
1987년 한양대학교 전자계산학 석사
1994년 인하대학교 전자계산학 박사
1977년~1988년 한국전자통신 연구소 시스템소프트웨어 연구실장
1989년~1996년 한국전자통신연구원 시스템연구부장, 주전산기(타이컴)Ⅲ,Ⅳ 개발사업 책임자
2001년~1999년 한국전자통신연구원 컴퓨터소프트웨어 연구소장
2000년~현재 선문대학교 컴퓨터정보학부 교수

<주관심분야 : 실시간데이터베이스시스템, 운영체제, 시스템구조>

이 정 현(평생회원)

1977년 인하대학교 전자공학과
1980년 인하대학교 전자공학과 석사
1988년 인하대학교 전자공학과 박사
1979년~1981년 한국전자기술 연구소 시스템 연구원
1984년~1989년 경기대학교 전자계산학과 교수
1989년 1월~현재 인하대학교 컴퓨터공학부 교수

<주관심분야 : 자연어처리, HCI, 음성인식, 정보검색, 고성능 컴퓨터구조>