

무선 센서네트워크에서 네트워크 생존성 향상에 관한 연구

이석우*, 신승중*, 류대현*, 안철주*
*한세대학교 IT융합학과
e-mail:dolbi14sec@gmail.com

A Study on Improvement of Network Survivability Wireless Sensor Networks

Seok-Woo Lee*, Seung-Jung Shin*, Dae-Hyun Ryoo*, Cheol-Joo Ahn*
*Dept of IT Convergence Science, Han-Sei University

요 약

모바일 로봇 센서가 개입하여 네트워크의 단절을 억제하여 모바일 로봇 센서를 통해 센서 네트워크의 생존시간을 향상시키는 것을 목표로 한다. 센서노드가 비교적 적은 희소한 환경에서는 배터리 소모 등의 다양한 원인으로 인하여 매개체 역할을 수행하는 브릿지 노드가 제거 될 경우 해당 브릿지 하부의 네트워크는 단절되어 더 이상 쓸모없게 되는 심각한 상황을 초래하기 쉽다. 본 논문에서는, 모바일 로봇 센서가 개입하여 단절된 브릿지를 효율적으로 찾아가 그 역할을 대행함으로써 네트워크의 생존 시간을 연장 시키는 것을 목적으로 하는 센서 네트워크를 설계하고 구현했다.

1. 서론

본 논문에서는 센서 네트워크에 로봇 센서를 개입시켜 네트워크의 생존시간을 향상하는 알고리즘을 제안하고 구현한다. 특히, 감시 지역이 넓은 경우(Sparse Sensor Network)는 센싱 지역을 구분하여 멀티 센싱을 수행한다. 그 중 Base Node(Sink)와 가장 인접한 중간 수집 노드(Bridge node)가 매체 역할을 수행하여 하부 센서들로부터 정보를 전달 받고, 중간 수집 노드들이 Base Node에 센싱 정보를 최종 전달하게 된다. 이때, 배터리 소모 등의 다양한 원인으로 중간 수집 노드(Bridge Node)가 제거될 경우 해당 Bridge 하부의 네트워크는 단절되는 심각한 상황을 초래하기 쉽다.

본 논문에서는 인간을 대신한 이동형 로봇이 Base Node와 협업을 통해 제거된 Bridge를 모니터링 하고, 만일 단절된 Base Node가 인지되면 해당 지역으로 찾아가 해당 Bridge역할을 로봇이 직접 대행하여 네트워크의 생존성을 향상시키는 시스템을 설계하고 구현하였다.

구현용 Mote의 Target Board는 128k의 비교적 넉넉한 메모리를 가진 ATmega128 칩셋이 탑재된 Micaz모터를 이용하였다. 운영체제와 응용프로그램은 크기가 작고 전력 소모가 적은 저전력 통신을 수행하는데 주로 사용되는 TinyOS 1.1.14와 NetC를 이용하였다.

2. 시스템 설계

2.1 전체 시스템 구성도

전체 시스템 구성도를 아래 그림 4와 5에서 표현하였다. 그림 4에서는 특정지역의 Bridge Node가 제거되어 해당 구역 전체 네트워크가 단절된 상황을 보여준다. 이동형 로봇은 네트워크를 무작위로 움직이다가 단절된 Bridge가 포착되면 Base Node와 협업하여 단절된 Bridge Node의 위치로 이동하게 된다. 그림 5에서는 로봇이 단절된 Bridge Node의 역할을 대행하여 네트워크가 정상으로 환원된 모습이다.

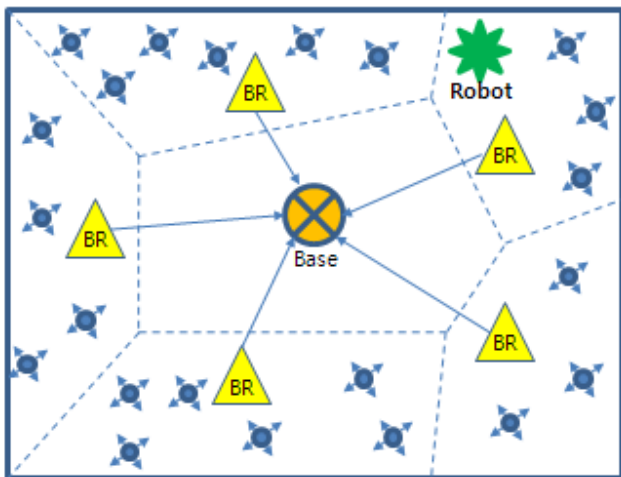


그림 1 멀티 센싱 구조

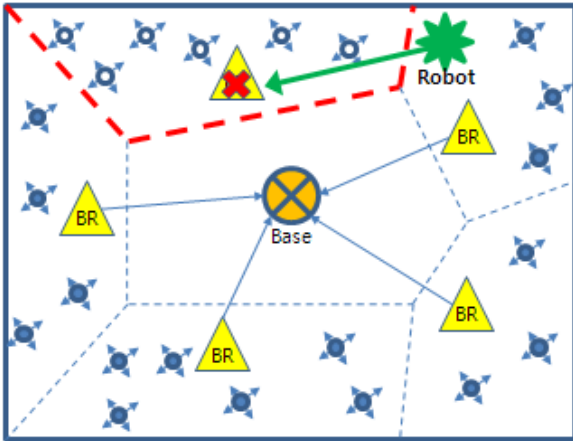


그림 4 특정지역 BRN단절 및 Robot의 이동

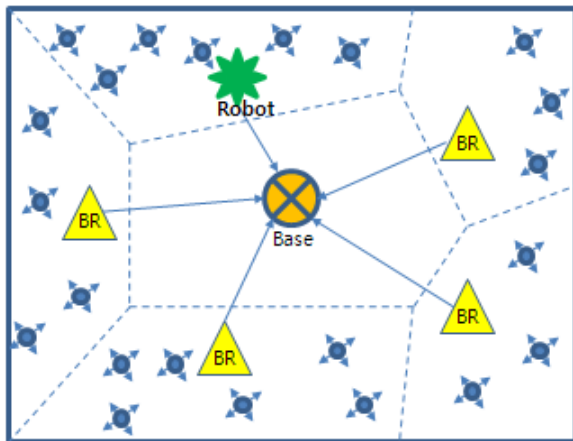


그림 5 Robot에 의한 단절BR 대행

2.2 노드의 형태와 기능 정의

2.2.1 Base Node

Base Node(BN) 는 인접한 Bridge Node의 위치정보를 기억한다. BN은 BRN 및 그 하부의 노드들로부터의 센싱된 정보를 Command Center에 Dump하는 임무를 부여한다. BN은 Bridge Node나 End Node에 비해 다양한 정보와 컨트롤 정보 등을 저장해야 하고 하부 노드 및 로봇 노드를 제어 하므로 충분한 메모리 확보가 필요하고 비교적 많은 전력을 소모하게 된다. 따라서, BN은 상시적인 전원을 공급 받는 것을 전제로 한다.

2.2.2 Bridge Node

BRidge Node(BRN)는 주변에 산재된 End Sensor Node들로부터 Sensing 정보를 전달 받는 Cluster역할을 수행한다. 따라서, 하부의 End Sensor Node들에 비해 비교적 충분한 메모리를 확보하고 있으며, End Node들로부터 전달받은 정보들을 누적하다가 Threshold를 초과하면 Base Node에 Dump한다. Head선출은 클러스터 기반으로 선출되며, 주로 특정 지역을 기준으로 Area를 구분하여 각 노드 중에 Base Node와 가장 근접한 Node가 Bridge Node 대상이 된다.

2.2.3 End Node

End Node(EN)들은 센싱 Value를 Bridge Node에 전송한다. 만일, Bridge Node가 응답이 없으면 배터리 보호를 위해 즉시, Sleep Mode로 전환한다.

3.2.4 Mobile Node

Mobile Node (Robot)는 네트워크를 무작위로 움직이며 단절된 BRN 정보를 Base Node로부터 전해 받는다. 만일, Base Node로부터 단절된 Bridge Node와 Routing 정보를 전달 받게 되면 단절된 Bridge가 위치했던 곳으로 직접 찾아가서 해당 Bridge 역할을 수행하여 Network의 연속성을 연장하는 주된 역할을 수행한다.

3.3 구현 알고리즘

3.3.1 Base Node 알고리즘

Base Node는 BRN으로부터 정보를 받음과 동시에 자신에 인접한 BRN List인 Linked_BR_Nodes[]를 Update한다. Base Node는 Update시에 Linked_BR_Nodes[]와 비교하여 LOST된 BRN을 인지하면 Lost_BR_Nodes[]에 Add하고 관리한다. 만일, 10번 이상 수신 누락시 Add하게 되며, 다시 Lost BRN으로부터 정보가 전달되면 Lost_BR_Nodes()에 Delete한다.

3.3.2 Robot Sensor 알고리즘

(1) 단절BRN 여부 탐지 모드 (Patrol Mode) :

MN은 네트워크를 무작위로 움직인다. MN이 EN 혹은 BRN을 인지하면 곧바로 단절된 노드 존재 여부를 알기 위해 Request_Order_From_BN를 요청한다. 해당 신호를 받은 EN 혹은 BRN은 지체하지 않고 BN에 Request_Order_From_BN를 재전송한다.

BN은 Request_Order_From_BN로부터 MN이 네트워크에 도달했음을 인지하고 Lost_BR_Nodes[]를 조회하여 Lost BRN (LBRN)이 있으면, LBRN중 현재 MN위치에서 가장 인접한 혹은 도달이 용이한 LBRN 라우팅 정보를 계산하여 BRN과 EN을 통해서 넘겨준다. 이때, BN은 MN에 LBRN이 관리하던 EN List정보도 함께 전송한다.

BN으로부터 손실된 Node 정보를 얻은 MN은 곧바로 LBRN SEEK모드로 변경된다. 만일, 손실된 Node가 없으면 단절된 Node를 찾기 위해 네트워크를 Patrol하게 된다.

(2) 단절BRN SEEK 모드 (Moving Mode) :

BN으로부터 LBRN ID/라우팅 정보/LBRN의 EN List 3가지 정보를 획득한 MN은 모드를 변경하여 해당위치로 이동하게 된다. BN이 전송한 라우팅 정보를 이용하여 LBRN위치로 이동한 MN은 인접한 ED들로부터 수신을 감지하면 LBRN의 EN List와 비교작업을 수행한다. 이때, MN은 지속적으로 EN으로부터 ID수신을 하며 EN List의 90% 이상의 범위에 도달할 때까지 움직이며 최적의 BRN 위치를 탐색한다. 탐색이 완료되면 LBRN delegation 모드로 자동 변경된다.

(3) 단절BRN Delegation 모드

탐색을 완료한 MN은 Delegation Mode로 전환하고 자신의 ID를 LBRN의 ID로 변경한다. 인접한 EN에게 자신이 BRN임을 알리면 EN은 Awake 모드가 되어 다시 센싱 정보를 전달한다. 이미 MN이 기존의 BRN의 ID를 소유하고 있으므로 Destination을 변경을 하지 않아도 된다. MN은 그때부터 LBRN과 동일한 임무를 수행하게 된다. 위임된 MN으로부터 센싱 정보를 수신한 BN은 Lost_BR_Nodes[]에서 Delete하고 Linked_BR_Nodes[]에 Add하여 Normal Routine을 수행하게 된다.

3. 시스템 구현 및 결과

3.1 개발 환경

3.1.1 하드웨어 플랫폼

개발용 하드웨어 플랫폼으로는 ATmega128 칩셋을 사용한 Micaz 모드를 사용하였으며, 개발용 Tool로는 Atmel사 mkII보드와 윈도우용 AVR Studio4를 이용하여 센서보드의 Rom에 Burning 하였다.

| MCU | Device | Type | Flash Memory /SRAM |
|----------------|------------|-----------------|---------------------------|
| | ATmega128L | 7.3728Mhz | 128kb/4kb |
| RF Transceiver | Device | Radio Frequency | Max. Data Rate(kbits/sec) |
| | nanoLOC | 2.4GHz | Maximum 2Mbps |
| Flash memory | Device | Connecion | Size |
| | AT45DB041B | SPI | 512kb |

표 1 개발용 센서 노드 Spec.

3.1.2 소프트웨어 플랫폼

OS는 센서 네트워크용으로 설계된 TinyOS를 사용하였으며 ver. 1.1.14에서 포팅 하였다. 개발 언어는 nesC를 이용하였으며 응용프로그램은 TinyOS에서 제공되는 UART통신, Ad-hoc라우팅과 멀티 홉을 주로 이용하였다.

3.2 각 Node별 Task 및 LED 상태

각 Sensor Node별 Task, Mode 및 LED의 상태를 표로 정리하였다.

(G:Green, R:Red, Y:Yellow, Tog:Toggle)

| Node | Task | Mode / LED State |
|---------|------------------------|-----------------------------------|
| End | Conn. Ready (Power On) | Sleeping / All Off |
| | Near & Bridge Conn. | Wake Up / G-Off, R-Off, Y-Tog |
| Bridge | Conn. Ready (Power On) | Sleeping / All Off |
| | Base Conn. | Wake Up / G-Off, R-Off, Y-Tog |
| Base | Conn. Ready (Power On) | Sleeping / All Off |
| | Bridge Conn. | BR Receive / G-Tog, R-Off, Y-Off |
| | Bridge & Robot Conn. | R Found / G-Tog, R-Off, Y-Tog |
| M Robot | Bridge Dis-Conn. Found | BR Dis-Conn. / G-Tog, R-On, Y-Tog |
| | Base Conn. | BR Receive / G-Off, R-On, Y-Off |
| M Robot | Seeking & Moving | Patrol, Move / G-On, R-On, Y-On |
| | Reset & Delegation | Delegation / G-Tog, R-Off, Y-Off |

표 2 각 노드별 Task 및 LED상태

3.3 구현 결과

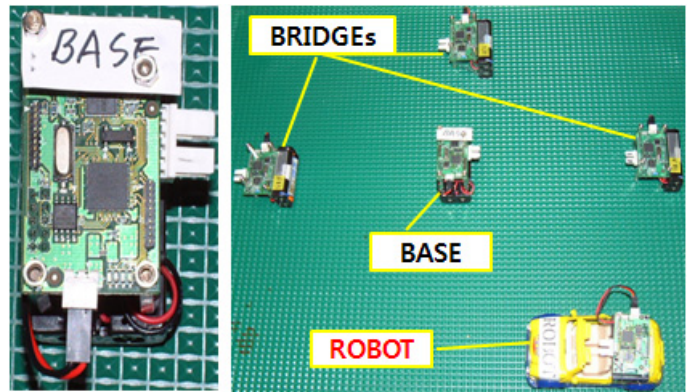


그림 6 구현Mote와 각 노드의 구성

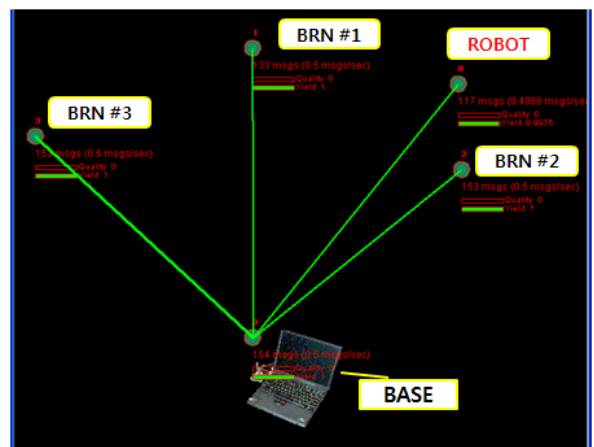


그림 7 정상 구동중인 센서 네트워크

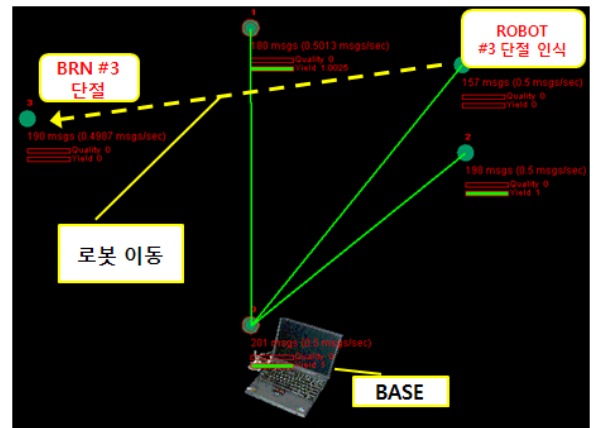


그림 8 1개 BRN 단절 및 ROBOT의 단절노드 인식

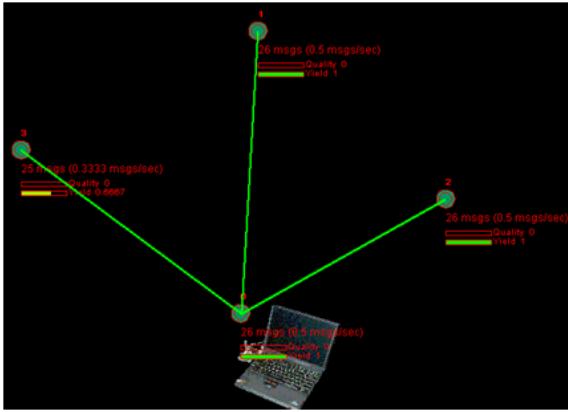


그림 9 ROBOT에 의한 네트워크 복원

4. 결론 및 향후 연구과제

센서 네트워크 기술은 유비쿼터스 컴퓨팅의 핵심을 차지하고 있으며 군사, 환경, 산업, 가정 등 다양한 분야에서 활용중이며 현재도 연구가 진행되고 있다. 특히, 센서 네트워크의 특성상 제한된 자원과 제한된 에너지를 가진 환경으로 인하여 네트워크의 수명을 연장하기 위한 연구는 빼놓을 수 없는 분야이기도 하다.

본 논문에서는 네트워크의 수명을 연장하기 위하여 이동형로봇을 매체로 활용하였다. 로봇센서노드가 Base Node와 협업을 통해 제거된 Bridge를 모니터링 하고, 만일 단절된 Base Node가 인지되면 해당 지역으로 찾아가 해당 Bridge역할을 로봇이 직접 대행하여 네트워크의 생존성을 향상시키는 시스템을 설계하고 구현하였다.

본 논문에서 Base Node와 로봇센서노드가 Bridge Node가 다른 노드들을 컨트롤 하기위해 에너지와 메모리를 사용하는 문제와 로봇센서노드가 단절된 Bridge Node를 최단거리를 찾아가는 라우팅 문제는 해결해야 할 과제이다.

참고문헌

- [1] Wireless Sensors and Intergrated Wireless Sensor networks (Technical Insights), Frost & Sullivan, 2002.
- [2] Y. Yu, R. Govindan, and D. Estirin, Geographical and Energy Aware Routing, Dept. Computer Science of UCLA, 2001.
- [3] M Weiser, "Some Computer Science Issues in Ubiquitous Computing," Comm. ACM, Vol36, No7, 1993.
- [4] W.B Heinzelman, A.P Chandrakasan, and H. Balakrishnam, "An application-specific protocol architecture for wireless microsensor networks, Wireless Comm. IEEE, Vol. 1, No. 4, 2002.
- [5] W.B Heinzelman, "Application-Specific Protocol architecture for wireless microsensor networks, Ph.D. dissertation, Mass. Inst. Tech., Cambridge, 2000
- [6] Wen-Hwa Liao and Yu-Chee Lee, "A Lightweight

Localization Scheme in Wireless Sensor networks," Wireless and Mobile Communications, July 2006.

[7] R. Stoleru, J.A. Stankovic, and S.H. Son, "Robust node localization for wireless sensor networks," Proc. of the 4th workshop on Embedded networked sensor, pp. 48-52, 2007.

[8] Boukerche, A, Oliveira, H.A.B, Nakamura, E.F, and Loureiro, A.A.F, " Localization systems for wireless sensor networks," Wireless Communications, IEEE, Vol 14, pp 6-12, Dec 2007.

[9] S. Simic and S. Sastry, "Distributed localization in wireless ad-hoc networks," UC Berkeley, Tech. rep. UCB/ERL M02/26, 2002.