

논문 2023-18-19

IoT 디바이스의 연결성 향상을 위한 동적 leaf 모드 기반의 INFRA-RPL

(INFRA-RPL to Support Dynamic Leaf Mode for Improved Connectivity of IoT Devices)

홍석원, 유성은*

(Seokwon Hong, Seong-eun Yoo)

Abstract : RPL (IPv6 Routing Protocol for Low-power Lossy Network) is a standardized routing protocol for LLNs (Low power and Lossy Networks) by the IETF (Internet Engineering Task Force). RPL creates routes and builds a DODAG (Destination Oriented Directed Acyclic Graph) through OF (Objective Function) defining routing metrics and optimization objectives. RPL supports a leaf mode which does not allow any child nodes. In this paper, we propose INFRA-RPL which provides a dynamic leaf mode functionality to a leaf node with the mobility. The proposed protocol is implemented in the open-source IoT operating system, Contiki-NG and Cooja simulator, and its performance is evaluated. The evaluation results show that INFRA-RPL outperforms the existing protocols in the terms of PDR, latency, and control message overhead.

Keywords : Internet of Things, IoT, RPL, Leaf mode, Dynamic leaf mode, INFRA-RPL

1. 서론

RPL (IPv6 Routing Protocol for Low-Power and Lossy Networks) [1]은 2012년 IETF [2] 워킹그룹에서 제안한 표준 (RFC 6550)으로, IPv6 기반 저전력 IoT [3] 무선 네트워크를 위한 라우팅 프로토콜이다. RPL에서는 세 종류의 제어 메시지 (DODAG Information Object, Destination Advertisement Object, DODAG Information Solicitation)를 이용하여 트리 형태의 비순환 그래프 (DODAG: Destination-Oriented Directed Acyclic Graph)를 구성하고 목적지 (Sink)로의 경로를 형성한다. 기존 RPL은 노드의 이동성을 고려하여 제작된 것이 아니기에 장치들이 정적인 환경에서는 제대로 동작하지만, 노드가 이동할 경우 DODAG을 재구축해야 하는 상황이 발생하며, 이는 패킷 손실률, 에너지 소모량과 연결 지연 시간을 상승시킬 수 있다.

하지만 현대 사회는 기술이 발달함에 따라 인간과 기계, 기계와 기계의 상호 작용이 더욱 중요해지고 있다. 기계의 경우 고정된 장소에서 이동하는 경우가 드물지만, 사람은 기계와 달리 생리 현상 및 다른 일을 수행하는 데 있어 장소를 이동하는 경우가 많고 그에 따라 이동성 지원에 대한 수요가 높다 [4, 5]. 물론, 기계에도 이동성을 보장하면 여러 장소에서 여러 정보를 수집할 수 있는 이점이 있다. 예를 들어 업무를 하는 사무실을 살펴보면 사람들은 보통 같은

자리, 같은 곳에서 업무를 본다. 하지만 때에 따라 사람은 기계와 다르게 같은 자리에 있을 수 없으며 어떤 이유로든 이동한다. 또 다른 예로, 축제의 현장을 보면 주변의 가로등, 스피커, 천막, 등 사물들은 모두 고정된 자리에 있지만 사람들은 계속해서 이동한다. 하루를 기준으로 기계는 한 장소에서 24시간 머무를 수 있지만 사람은 그렇지 않다. 우리 주변의 환경 대부분에서 기계나 사물들은 많은 경우 한 장소에 머무르지만, 사람은 여러 생리 현상을 해결하기 위해 이동을 할 수밖에 없는 것이 현실이다. 따라서 본 논문은 정적인 상태에 있는 기계와 이동하는 사람이 효율적으로 통신할 수 있도록 INFRA-RPL을 제안하고자 한다. INFRA-RPL은 기본 노드와 모바일 노드로 구성된다. 기본 노드는 기본적으로 기계를 포함한 사물에 설치되며 움직이지 않는다. 모바일 노드는 사람이 휴대하는 노드로서 이동성을 지원하는 노드이다. 기존의 RPL에서 모바일 노드가 이동하게 되면 이 모바일 노드의 초기 부모는 사라진 모바일 노드를 찾기 위해 제어 메시지를 보낸다. 그 과정에서 에너지 소모와 전송 지연을 겪게 된다. 하지만 본 논문의 INFRA-RPL에서는 사람의 이동성을 보장하면서 기계는 기계의 DODAG (통신을 위한 infrastructure)을 빠르게 재구축할 수 있도록 한다.

본 논문의 구성은 다음과 같다. 2장에서는 RPL의 개요와 관련 연구를 살펴보고 3장에서는 제안된 INFRA-RPL에 대하여 설명하고 4장에서는 성능평가를 진행한다. 5장에서는 본 논문을 요약하며 결론을 내린다.

*Corresponding Author (seyoo@daegu.ac.kr)

Received: Jul. 6, 2023, Revised: Aug. 8, 2023, Accepted: Aug. 14, 2023.
S. W. Hong: Daegu University (B.S. Student)

S. Yoo: Daegu University (Prof.)

* 이 논문은 2021년도 정부 (과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임 (No. 2020R1F1A1048179).

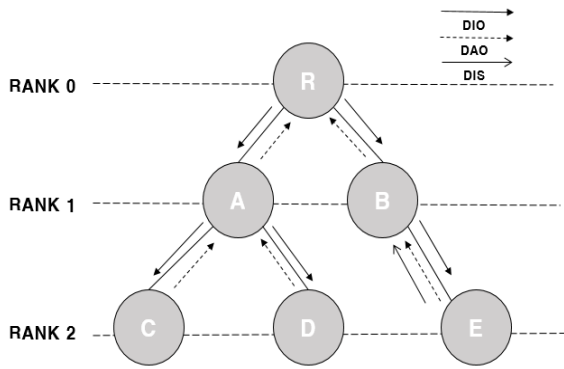


그림 1. DODAG
Fig. 1. DODAG

II. 관련 연구

본 장은 RPL을 소개하며 관련 연구를 분석하여 이들의 문제점을 설명한다.

1. RPL의 개요

RPL은 root 노드를 기반으로 하며 주변 노드에게 랭크(rank)를 부여하고 랭크를 기준으로 그림 1과 같은 DODAG 네트워크를 구축한다. 노드의 랭크 계산은 RPL에서 사용되는 목적함수 (Objective Function)에 의해 결정된다. RPL에서는 OF0 (Objective Function Zero)와 MRHOF (Minimum Rank with Hysteresis Objective Function)라는 두 가지 OF가 표준으로 되어있다.

RPL에는 그림 1과 같이 DIO (DODAG Information Object), DAO (Destination Advertisement Object), DIS (DODAG Information Solicitation), 세 가지 제어 메시지가 정의되어 있다. DIO 메시지는 DODAG 내의 노드들이 주기적으로 자신의 네트워크 정보를 알리기 위해 사용하는 메시지이며 각 노드는 DIO를 통해 이웃 노드들의 정보를 업데이트한다. 각 노드는 이웃의 정보를 기반으로 목적함수로 계산된 결과를 사용하여 부모와 자식 간의 관계를 설정한다. DAO 메시지는 root 노드에서 각 노드로 가는 메시지를 전송하기 위한 최선의 경로 (downward routes)를 확보하기 위해 사용된다. DODAG이 구축된 상태에서 각 노드들은 DAO를 사용하여 부모와 자식 간의 관계가 정의된 경로를 통해 root 노드까지 경유하는 모든 노드들의 주소 정보를 DAO 메시지에 기록한다. DIS 메시지는 DODAG내에 새로운 노드가 참여하거나, DODAG이 업데이트가 되었을 때 이상이 없는 노드가 DODAG에서 제외된 경우 상위 노드로 DIO 메시지를 요청하기 위해 사용된다. DIS를 받은 상위 노드는 자기 부모 노드를 거쳐 root로 DIS 메시지를 전달하여 DIO를 수신한 후, 하위 노드로 DIO를 전달하여 하위 노드 (새로운 노드)를 DODAG에 가입시킨다.

RPL은 Coniki-NG [6], TinyOS [7], OpenWSN [8] 등의 사물인터넷 오픈소스 운영체제에 구현되어 있다.

2. RPL mode

본 장에서는 RPL의 모드 (mode)에 대하여 설명한다. RPL의 모드는 저장 모드, 비 저장 모드 두 가지가 있으며 자식을 가지지 못하는 leaf 모드 또한 존재한다. 비 저장 모드에서는 노드들이 통신하기 위해서는 root 노드로 메시지를 전송하여 통신을 해야 한다. 하지만 저장 모드에서는 각 노드 간의 상호 작용이 가능하며 통신을 위해 root 노드까지 메시지를 전송할 필요가 없어진다. 비 저장 모드에서는 root만 모든 노드에 대한 정보를 알고 있지만 저장 모드에서는 노드들이 모든 주변 노드의 정보를 알고 있어야 하기 때문에, 에너지 소모율은 저장 모드가 비 저장 모드에 비해 높다. leaf 모드는 자기 자신이 자식을 가지지 못하는 모드를 말하며 DIO 수신과 DAO 전송만 가능하다. 만약 root 노드를 제외한 모든 노드가 leaf 모드라면 root의 통신 거리에 있는 노드를 제외하고는 네트워크 연결이 어렵다.

3. Trickle timer

Trickle timer [9]는 RPL에서 DODAG을 구축하기 위해 사용하는 timer이다. RPL에서는 root로부터 시작되는 DIO 메시지를 통해 각 노드들의 정보를 받아들이고 DODAG 내의 변경 사항에 대해 업데이트한다. 만약 변경 사항이 없음에도 불필요한 DIO 메시지를 계속 전송한다면 DODAG을 구성하는데 있어 에너지 소모율이 상승할 수밖에 없다. 따라서 RPL에서는 Trickle timer를 도입하여 DIO 전송 주기를 적응적으로 조절한다. 전송 간격 I 는 I_{min} 과 I_{max} 사이의 범위에서 설정되며 그 초깃값은 I_{min} 이다. 그 후 DIO 전송을 통해 DODAG 내의 통신이 정상적으로 이루어지지 않는다면 I_{min} 값을 유지하고 만약 통신이 성공적으로 이루어지면 I 의 값을 두 배로 증가시켜 I_{max} 값까지 증가시킬 수 있다.

4. OF (Objective Function)

앞 절에서 설명한 바와 같이 DODAG에서 노드의 랭크 계산은 해당 RPL에서 사용되는 OF에 의해 결정된다. IETF에서는 OF0 (Object Function Zero)와 MRHOF (Minimum Rank with Hysteresis Objective Function)라는 두 가지 OF를 RFC 문서로 표준화하였다. OF0에서 랭크는 DODAG에서 root 노드를 기준으로 홑 기반으로 계산되어 노드들이 최단 경로로 DODAG을 형성하고 MRHOF는 패킷을 오류 없이 전송하는 데 필요한 예상 전송 횟수인 ETX (Expected Transmission Count)를 기본으로 사용하여 랭크를 계산한다. 따라서 네트워크 품질 측면에서 신뢰성은 MRHOF가 우수하고 낮은 전력과 빠른 연결에 대해서는 OF0가 우수한 경향을 보인다.

5. 기존 연구

모바일 노드를 위한 RPL기반의 하이브리드 모드 라우팅 프로토콜 [10]에서는 노드의 이동성을 보장하기 위해 저장 모드와 비 저장 모드를 혼합한 RPL을 제안하였다. 하지만 이동

하는 모바일 노드가 여러 개인 경우 모바일 노드들도 서로 간에 부모 자식 관계를 맺을 수 있다. 하지만 두 모바일 노드가 서로 반대 방향으로 이동하는 경우, 통신이 빠르게 끊어지고 다시 DODAG을 구축해야 하기에 데이터 손실이 클 수밖에 없다.

EC-RPL [11]은 노드의 에너지 잔량을 파악하여 노드가 에너지를 모두 소모하여 끊어지기 전에 에너지가 낮은 노드를 DODAG에서 제외하고 새로운 DODAG을 형성한다. 하지만 이는 노드의 이동성을 고려하지 않고 만들어졌기에 노드가 이동하는 상황에서는 효율이 떨어진다.

OF-EC [12]에서는 초기 에너지에서 노드가 사용한 에너지를 빼서 노드의 잔여 에너지를 계산하고, 이 잔여 에너지를 통해 계산된 링크 값을 DIO 메시지 필드에 보냄으로써 잔여 에너지가 높은 노드를 선호 부모로 선택하도록 DODAG을 구축한다. 그러나 위 연구에서 잔여 에너지가 많은 노드의 링크 상태가 좋지 않다면 노드의 에너지를 빠르게 소진하여 네트워크의 수명이 빠르게 줄어드는 문제점이 있다.

RPL에서 이동성 향상을 위한 DIO 전송 간격 조절 연구 [13]에서는 노드의 이동성을 향상하기 위해 DIO전송 간격을 조절하여 이동하는 노드와도 통신하도록 한다. 하지만 노드의 이동 속력이 빠르면 DIO 전송이 많아지기에 에너지 소모는 커진다.

ADP-RPL (Adaptive RPL) [14]에서는 링크 값의 증가와 부모의 변경 시간을 바탕으로 링크 품질의 악화를 판단하여 링크 값에 반영하여 불안정한 노드를 부모 노드로 선택하지 않고 링크가 좋은 노드를 부모 노드로 선택하여 DODAG을 구축한다. 하지만 노드가 이동하는 상황에서는 링크가 자주 변동이 되며 품질이 악화되고 이에 따라 통신이 불안정해지게 불가피하게 패킷이 소모되는 단점이 존재한다.

위 연구들은 노드가 이동할 때, 새로운 DODAG을 형성해야 하기에 패킷 손실이 일어나기 쉽다. 하지만 I장에서 기술한 바와 같이 현대 사회에서는 사람과 기계, 사람과 사물의 상호 작용을 지원하기 위해서 이동성 지원은 불가피하다. 따라서 본 연구에서는 정적인 기계 (사물)와 이동하는 사람을 가정하여, 기계 (사물)의 INFRA는 유지하면서 이동하는 사람과의 상호작용을 지원하는 INFRA-RPL을 제안한다.

III. INFRA-RPL

INFRA-RPL은 RPL에서 기본적으로 제공하는 기본 노드와 이동성을 가진 leaf 모드인 모바일 노드로 나누어 기존 기본 노드의 DODAG은 그대로 형성하도록 하고 모바일 노드의 제한적인 이동성을 보장한다. 기존 RPL의 leaf 모드로 설정된 노드는 부모 노드가 부득이하게 고장이 나거나 통신이 불가능한 상황일 경우, 그림 2과 같이 네트워크로부터 단절된다. 하지만 같은 상황에서 제안하는 INFRA-RPL에서는 그림 3과 같이 leaf 모드인 모바일 노드 (E)를 기본 모드로 바꾸어 부모와 연결이 끊긴 C와 D노드에게 연결성을 제공한다.

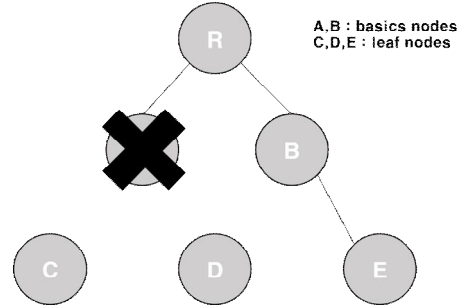


그림 2. 기존 leaf mode
Fig. 2. Existing leaf mode

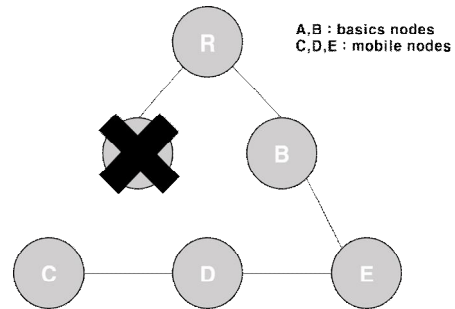


그림 3. INFRA-RPL
Fig. 3. INFRA-RPL

```

if(No DIO received during trickle timer period){ //DIS transmission
    print("parent loss parent_loss_flag = 1")
    parent_loss_flag = 1
    dis_send(parent_loss_flag)
}
if(parent_loss_flag in received DIS == 1){ //DIS reception
    print("received parent_loss_flag = 1, leaf_mode: disable")
    leaf_mode = 0
}
else{
    print("received parent_loss_flag = 0, leaf_mode: enable")
    leaf_mode = 1
}
    
```

그림 4. INFRA-RPL 의사 코드
Fig. 4. INFRA-RPL pseudo code

1. INFRA-RPL의 구조

INFRA-RPL에서 자식을 받아들일 수 있는 모드를 기본 모드, 그렇지 못한 경우 leaf 모드라 부른다. 기본 노드는 기본 모드로 동작하며 이동성이 지원되지 않는다. 모바일 노드는 기본적으로 (default) leaf 모드로 설정되지만, 네트워크 상황에 따라 기본 모드로 동작할 수 있다. 모바일 노드 중 leaf 모드로 동작하는 모바일 노드를 leaf 노드라 부른다.

INFRA RPL의 구조는 그림 4와 같다. RPL이 Trickle timer 주기로 DIO를 전송하는 동안 모바일 노드 (M∈{C, D, E}, 그림 3)에 DIO가 수신되지 않는다면 parent_loss_flag를 1로 설정한 후 이를 DIS메시지에 포함시켜 주변의 노드로 DIS를 전송한다. DIS를 수신한 모바일 노드(E)가 leaf 모드일

경우 leaf 모드의 값을 0으로 바꾸어 기본 모드로 바꾸고 DIS를 전송한 모바일 노드 (M)를 자식으로 받아들인다. 만약 DIS를 수신한 노드가 기본 노드라면 모드를 바꿀 필요 없이 그대로 자식으로 받아들인다. 기존의 정적인 leaf 모드를 사용하는 RPL의 경우에는 leaf 노드의 부모 노드가 사라지고 주변의 노드들이 모두 leaf 모드로 동작한다면, 부모를 잃은 leaf 노드는 통신을 할 수 있는 방법이 없다. 하지만 INFRA-RPL은 부모가 사라진 상황에서 주변의 leaf 노드(E, 그림 4)가 동적으로 모드 전환을 하여 연결성을 지원할 수 있다는 장점이 있다.

IV. 성능 평가

본 장에서는 기존 RPL과 INFRA-RPL의 성능을 평가한 시뮬레이션 모델을 설명하고 그 결과를 분석한다.

1. 실험 환경

실험 환경은 표 1과 같이 Contiki-NG OS의 Cooja 시뮬레이터를 사용하였다. Contiki-NG는 저전력 무선통신 IoT를 위한 오픈소스 운영체제이다. Cooja는 네트워크를 시뮬레이션하기 위한 시뮬레이터로 Contiki-NG에서 제공하는 코드를 하드웨어에 flashing 하기 전에 시뮬레이터로 가상환경을 생성하여 문제점을 파악할 수 있다. Device는 Cooja Mote를 사용하며, Topology는 그림 5와 같이 설정하고 응용계층에서 1분마다 10바이트를 root 노드에게 전송하도록 하고, 다음에 설명하는 세 가지 실험을 진행한다.

기존의 RPL은 기본 노드를 12개 준비하였다. INFRA-RPL은 root 노드 1개 (①)와 기본 노드 4개 (②-⑤), 모바일 노드 8개 (⑥-⑬)를 준비하였다. RPL (leaf)는 기존 RPL실험에서 12개의 노드 중 8개를 leaf 노드로 바꾼 것이며 실험 환경은 INFRA-RPL과 같이 설정하였다. 모두 같은 시드 (seed)와 같은 TX RX 성공율로 설정하였고, 설정 기준은 100%로 해두면 모든 패킷이 송수신되기 때문에 TX 100% RX 80%로 설정하였다. 또한 갑자기 노드가 고장이 나서 통신이 중단될 수 있는 환경을 만들기 위해 시뮬레이션 시간으로 30분이 지난 후 그림 6과 같이 기본 노드 중 하나 (③)를 제거하고 30분을 더 진행하였으며 모든 실험의 시간은 동일하게 진행하였다 (실험 1). 각각의 실험을 다른 시드로 10회씩 실행하였고 평균을 내어 실험 결과를 나타내었다.

표 1. 시뮬레이션 환경

Table 1. Simulation Environment

OS/Simulator	Contiki-NG/Cooja
Device	Cooja Mote
Duty Cycle	ContikiMAC
NWK	RPL-light
APP	sending 10B every 60s over UDP
Radio medium	UDGM(distance loss)
TX	100%
RX	80%

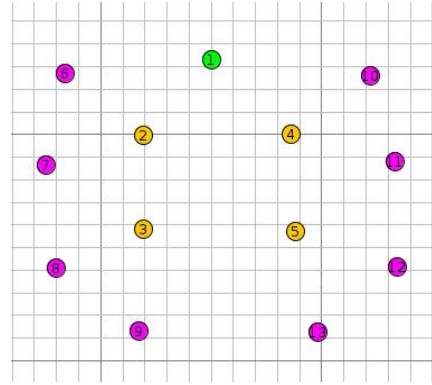


그림 5. 시뮬레이션 토폴로지

Fig. 5. Simulation topology

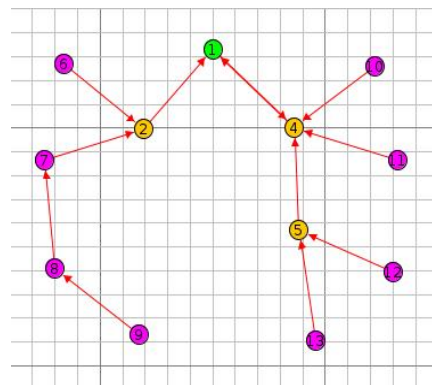


그림 6. 기본 노드 제거 후 시뮬레이션 토폴로지 (30분 후)

Fig. 6. Simulation topology after a basic node removal (30 minutes later)

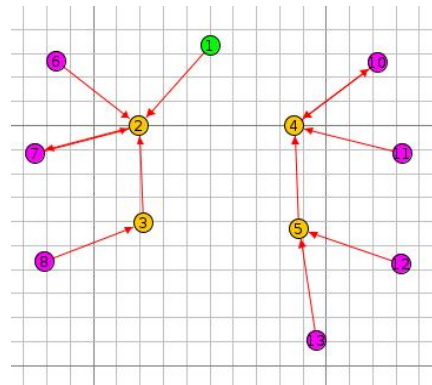


그림 7. 모바일 노드 제거 후 시뮬레이션 토폴로지 (30분 후)

Fig. 7. Simulation topology after a mobile node removal (30 minutes later)

OF는 OF0를 사용하였다. OF0를 사용한 이유는 MRHOF는 홉 기반이 아닌 패킷을 오류 없이 전송하는 데 필요한 예상 전송 횟수인 ETX를 사용하여 OF0보다 부모 변경이 많기 때문이다. MRHOF에서 모바일 노드가 ETX에 따라 다른 모바일 노드를 부모로 선택할 경우 두 노드 사이는 통신이 불가능하기 때문에 DODAG을 구축하는 데 있어 에너지 소모를

과 전송 지연 시간 (latency)이 늘어나고 이에 따라 통신이 불가능한 상황이 이루어질 수 있다. 이는 현장에서 매우 치명적으로 다가올 수 있다. 긴급한 상황에 통신이 제대로 되지 않으면 사고로 이어져 많은 인명 피해가 발생할 수 있다.

기존의 RPL과 본 연구에서 제안한 INFRA-RPL의 비교를 위해 실험을 하나 더 진행하였다. 본 실험 (실험 2)은 그림 5와 같이 Topology를 구축한 후 30분이 지난 후 모바일 노드 중 하나 (⑨)를 제거 (그림 7)한 후 다시 30분이 지날 때 각각의 PDR (Packet Delivery Ratio)을 비교해 보았다. 실험을 2가지로 진행한 이유는 첫 번째 실험은 고정된 기계 (기본 노드)가 고장 난 상황을 가정한 것이고, 두 번째 실험은 기존의 상황에서 사람이 이동한 상황 (모바일 노드 단절)을 가정한 것이다.

2. PDR

첫 번째 실험에서 PDR 을 기존 RPL과 INFRA-RPL을 비교하였다. PDR은 (전체 받은 패킷/전체 보낸 패킷)*100의 값으로 각 노드가 보낸 데이터가 root 노드에게 얼마나 전달이 되었는지 측정한 것이다. 그림 8은 기존 RPL과 INFRA-RPL의 PDR을 비교 결과이다. RPL의 PDR은 75.25%로 나왔고, RPL (leaf)의 경우 82.43%, INFRA-RPL의 PDR은 85.46%가 나왔다. RPL (leaf)에서는 그림 5에서 가장자리에 있는 8개의 모바일 노드를 leaf 모드로 설정함으로써 제어 메시지 전송량이 줄어, 전체적인 PDR이 개선되었다. 제안된 INFRA-PDR이 기존 RPL보다 약 10.21% 높은 것을 알 수 있으며 RPL (leaf)보다는 3.03% 높다. 이는 INFRA-RPL이 leaf 모드를 동적으로 변경해 다른 모바일 노드에게 네트워크 연결을 제공하여 높은 PDR성능을 보인 것으로 판단할 수 있다.

두 번째 실험에서도 PDR을 비교하였다. 그림 9는 기존 RPL과 INFRA-RPL의 PDR을 비교 결과이다. RPL은 85.48% 나왔고, RPL (leaf)는 88.4% INFRA-RPL은 88.56가 나왔다. 제안된 연구가 기존 RPL보다 약 3.08% 높은 것을 알 수 있으며 RPL (leaf)와는 유사한 성능임을 확인할 수 있다.

3. Latency

본 절에서는 실험에 대한 Latency를 살펴본다. Latency는 패킷을 목적지 노드로 전달하는데 걸린 시간을 측정한 것으로 Latency = (패킷 수신 시간) - (패킷 송신 시간)으로 계산할 수 있다. 그림 10은 실험 1의 실험 결과이다. 기존 RPL은 311.9ms, RPL (leaf)는 295.7ms, INGRA-RPL은 292.4ms로 측정되었으며 제안된 INFRA-RPL이 기존 RPL 보다는 19.5ms, RPL (leaf)보다는 3.3ms 더 짧은 latency를 가지는 것으로 측정되었다.

그림 11은 실험 2에서 측정된 latency를 비교한다. 기존 RPL의 경우 292.9ms, RPL (leaf)의 경우 268.3ms, INGRA-RPL의 경우 268.4ms가 나왔으며, 제안된 INFRA-RPL이 기존 RPL 보다는 24.5ms 더 빠른 것으로 나왔다. 이를 통해 기존의 RPL보다 제안한 RPL의 성능이 더 뛰어난 것을 알 수 있다.

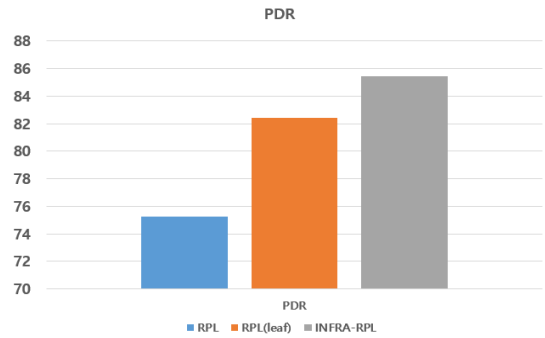


그림 8. 실험 1에 대한 PDR
Fig. 8. PDR for Experiment 1

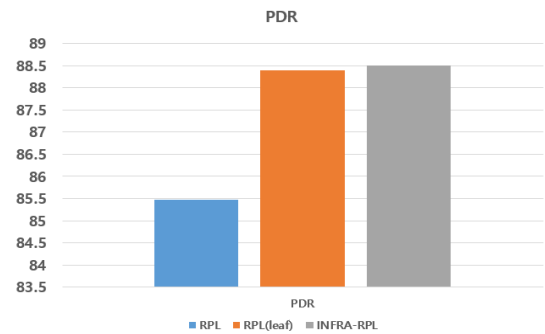


그림 9. 실험 2에 대한 PDR
Fig. 9. PDR for Experiment 2

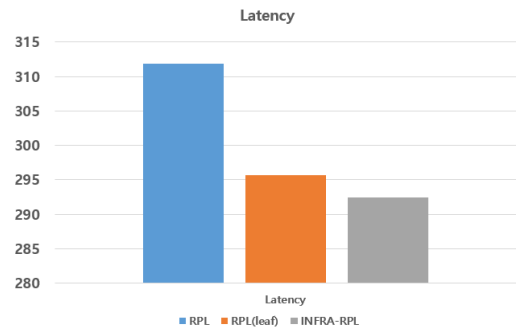


그림 10. 실험 1에 대한 Latency
Fig. 10. Latency for Experiment 1

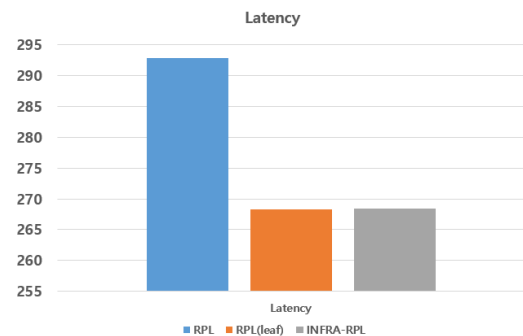


그림 11. 실험 2에 대한 Latency
Fig. 11. Latency for Experiment 2

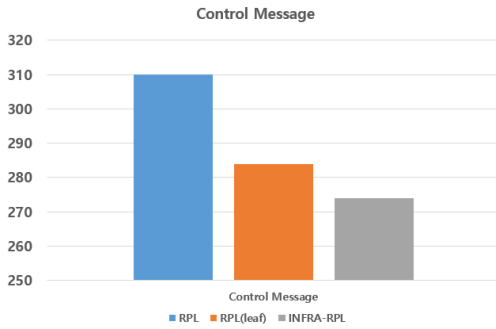


그림 12. 실험 1에 대한 제어 메시지 개수

Fig. 12. the number of control messages for Experiment 1

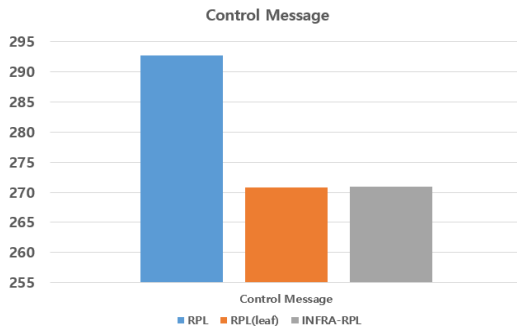


그림 13. 실험 2에 대한 제어 메시지 개수

Fig. 13. the number of control messages for Experiment 2

4. Control Message Overhead

본 절에서는 각 실험에서 사용된 Control Message의 개수를 통해 에너지 효율성을 유추해본다. Control Message는 DODAG를 구축함에 있어 서로 전달되는 제어 메시지이다. 그림 12는 실험 1에 대한 결과로 기존 RPL에서 310.1개, RPL (leaf)에서 283.9개, INFRA-RPL은 274개의 제어 메시지가 사용되었으며, 제안한 RPL이 기존 RPL보다는 36.1개, RPL (leaf)보다는 9.9개 적은 제어 메시지를 사용한 것을 알 수 있다.

그림 13은 실험 2에 관한 결과로 기존 RPL은 292.7개, RPL (leaf)는 270.8개, INFRA-RPL은 270.9개의 제어 메시지를 사용하며, 제안한 RPL이 기존 RPL보다는 더 적은 제어 메시지를 사용하였다. 이를 통해서 제안된 INFRA-RPL이 다른 비교 대상의 프로토콜보다 이동성이 있는 환경에서 에너지 효율적임을 알 수 있다.

5. 실험 결과 분석

실험 1 같은 경우는 기본 노드 (고정 기계)를 제거하여 기존의 DODAG에서 연결되어있는 기본 노드와 모바일 노드의 연결이 끊기면 모바일 노드는 새로운 부모를 찾기 위해 DIS를 송신해야 한다. 주변 노드가 leaf 모드일 경우 이 DIS에 대해 응답하지 않아 모바일 노드가 네트워크에 연결되지 못한다. 하지만 INFRA-RPL의 경우 연결이 끊긴 노드는 자신의 parent_loss_flag를 1로 변경하여 DIS를 통해 다른 노

드로 parent_loss_flag를 전송한다. 이를 수신한 모바일 노드가 leaf 모드로부터 벗어나 DIS에 대해 응답하여 연결이 끊긴 노드에게 연결성을 제공한다. RPL에서는 모든 노드가 기본 노드이기 때문에 랭크값이 변동되고 모바일 노드와 기본 노드의 차이가 없기에 하나의 노드가 사라지면 DODAG를 재구축하는 데 시간이 걸리고 제어 메시지의 손실이 크다. 그래서 실험에서는 PDR과 지연 시간, 제어 메시지 개수 측면에서 가장 낮은 성능을 보인다. RPL (leaf)에서는 모바일 노드가 leaf 모드를 사용하기에 기본 노드 (고정된 노드)가 사라지면 모바일 노드는 더 이상 통신을 할 수 없고 기존 DODAG에서 사라진 부모를 찾기 위한 제어 메시지의 손실이 크다. 실험 2에서는 가장 끝에 있는 노드가 사라지기에 실험1 보다는 제어 메시지의 손실이 적어 전반적으로 실험1보다 성능이 더 나은 것을 볼 수 있다.

실험 2의 경우에는 RPL (leaf)와 INFRA-RPL의 차이가 크지 않음을 알 수 있는데 두 실험 모두 leaf 노드가 사라지는 것이기에 PDR과 Latency, 제어 메시지 오버헤드의 차이가 거의 없는 것을 알 수 있다.

V. 결론

현대 사회는 기계와 인간의 상호작용이 더욱 중요해지고 있는 시대로 IoT (Internet of Things)의 중요성도 높아지고 있다. 사람의 경우 이동하는 경우가 많아서, 사람이 휴대하는 장치에 이동성을 지원할 필요가 있다. 본 논문에서는 기존 RPL에서 leaf 모드를 사용하는 노드가 자식을 가질 수 없어 긴급한 상황에서 연결성을 지원하지 못하는 문제를 개선한 INFRA-RPL을 제안하고 구현하여 그 성능을 평가하였다. 제안된 연구에서는 leaf 노드가 특정 상황에 자식을 가질 수 있도록 지원하여 부모를 잃은 노드가 빠른 속도로 다시 네트워크에 연결할 수 있도록 한다. 본 연구에서는 제안된 프로토콜을 Cooja네트워크 시뮬레이터에서 평가하였으나 추후 실제 테스트베드에 구현하여 추가적인 성능 평가를 진행할 예정이다.

References

- [1] <https://www.rfc-editor.org/rfc/rfc6550>
- [2] https://www.ietf.org/?pk_campaign=ietf-google-ads&pk_source=google&pk_medium=cpc&pk_content=ietf-brand&clid=Cj0KCQiAxbefBhDfARIsAL4XLRomAgFriXnPdau_XdKQ0UEiLx26tASqEfu20VVJBefYpci03VmmYRIaAuOVEALw_wcB
- [3] J. Lee, S. Yoo, "An Extensible Smart Home IoT System Based on Low-power Networks," IEMEK J. Embed. Sys. Appl., Vol. 13, No. 3, pp. 133-141, 2018 (in Korean).
- [4] H. Jung, S. Yoo, "GRU-based Adaptive Data Rate Control Algorithm Considering the Mobility of LoRa Devices," Journal of Institute of Control, Robotics and Systems, Vol.28, No. 6, pp. 640-647, 2022 (in Korean).

[5] H. Jung, S. Yoo, "FCR-ADR Algorithm to Improve the Network Connectivity of Mobile LoRa Devices," Journal of Institute of Control, Robotics and Systems, Vol. 28, No. 9, pp. 838-845, 2022 (in Korean).

[6] <https://github.com/contiki-ng/contiki-ng/wiki>

[7] <http://www.tinyos.net/>

[8] <https://openwsn.atlassian.net/wiki/spaces/OW/overview>

[9] <https://www.rfc-editor.org/rfc/rfc6206>

[10] W. S. Son, "An RPL-based Hybrid Mode Routing Protocol for Supporting Mobile Nodes", 2015 (in Korean).

[11] J. D. Jung, S. W. Hong, Y. S. Kim, S. Yoo, "EC-RPL to Enhance Node Connectivity in Low-Power and Lossy Network", Journal of the Korea Industrial Information Systems Research, Vol. 27, No 6, pp. 41-49, 2022 (in Korean).

[12] H., Lamaazi, N. Benamar, "OF-EC: A Novel Energy Consumption Aware Objective Function for RPL Based on Fuzzy Logic," Journal of Network and Computer Applications, Vol. 117, pp. 42-58, 2018.

[13] Y. J. Shin, S. U. Seol, "The DIO Interval Adjustment to Enhance Mobility in RPL" The Journal of Korean Institute of Communications and Information Sciences Vol. 23, No.12, pp. 1679-1686, 2019 (in Korean).

[14] J. Lee, S. Yoo, "Adaptive ADP-RPL Avoiding Unstable Nodes in Low Power IoT Networks," Journal of Institute of Control, Robotics and Systems Vol. 26, No. 2, pp. 92-99, 2020 (in Korean).

Seokwon Hong (홍 석 원)

2023 Artificial Intelligence from Daegu University (B.S.)



Field of Interests: Internet of Things, Sensor Networks
 Email: sw071497@daegu.ac.kr

Seong-eun Yoo (유 성 은)

2003 Electronics and Computer Engineering from Hanyang University (B.S.)



2005 Information and Communications Engineering from KAIST (M.S.)

2010 Information and Communications Engineering from KAIST (Ph.D.)

2010~School of Artificial Intelligence, Daegu University (Prof.)
 Field of Interests: Internet of Things, Embedded Systems
 Email: seyoo@daegu.ac.kr