

RPL에서 DIS 메시지를 이용한 Storing 노드 추가 및 Instance 평준화 기법 연구

A Study on Storing Node Addition and Instance Leveling Using DIS Message in RPL

배 성 현*, 윤 정 오*

Sung-Hyun Bae*, Jeong-Oh Yun*

Abstract

Recently, interest in IoT(Internet of Things) technology, which provides Internet services to objects, is increasing. IoT offers a variety of services in home networks, healthcare, and disaster alerts. IoT with LLN(Low Power & Lossy Networks) feature frequently loses sensor node. RPL, the standard routing protocol of IoT, performs global repair when data loss occurs in a sensor node. However, frequent loss of sensor nodes due to lower sensor nodes causes network performance degradation due to frequent full path reset. In this paper, we propose an additional selection method of the storage mode sensor node to solve the network degradation problem due to the frequent path resetting problem even after selecting the storage mode sensor node, and propose a method of equalizing the total path resetting number of each instance.

요 약

최근 사물들에 인터넷 서비스를 제공하는 IoT(Internet of Things) 기술에 관한 관심이 높아지고 있다. IoT는 홈 네트워크, 헬스케어, 재난 알림 등에서 다양한 서비스를 제공하고 있다. LLN(Low Power & Lossy Networks) 특징을 갖는 IoT는 빈번한 센서 노드의 손실이 발생한다. IoT의 표준 라우팅 프로토콜인 RPL은 센서 노드의 손실이 발생하면 전체 경로 재설정(Global repair)을 수행하여 데이터를 전송한다. 하지만 낮은 사양의 센서 노드로 인한 빈번한 센서 노드 손실은 잦은 전체 경로 재설정으로 인해 네트워크 성능 저하를 일으킨다. 본 논문에선 Storing mode 센서 노드 선정 이후에도 잦은 경로 재설정 문제로 인한 네트워크 성능 저하 문제를 해결하기 위한 Storing mode 센서 노드의 추가 선정 방법을 제안하고 각 Instance의 전체 경로 재설정 횟수의 평준화 방법을 제안하고자 한다.

Key words : RPL, DODAG(Destination Oriented Directed Acyclic Graph), Iot, LLN, DIS(DODAG Information Solicitation)

* Dept. of Aviations Information & Communication
KyungWoon University.

★ Corresponding author
E-mail : joyun@ikw.ac.kr, Tel : +82-54-479-1312

※ Acknowledgment

Manuscript received Aug. 23, 2018; revised Aug. 31, 2018;
accepted Sep. 5, 2018

This is an Open-Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. 서론

최근 IoT 기술은 홈 네트워크, 헬스케어, 재난알림 등 다양한 서비스를 제공하고 있다[1-2]. IoT 네트워크에서 사용되는 센서 노드는 수집한 감지 정보를 교환하기 위해 통신 모듈을 탑재하고 있다. 모든 사물에게 유무선의 인터넷을 제공하며 저 전력 낮은 사양이 특징이다[3-4].

이러한 IoT 네트워크는 이동성과 외부충격, 통신 환경 등 저 사양 센서 노드로 인해 네트워크 라이

프타임, 신뢰성, 안정성, 정확성 등에서 크게 성능 저하를 일으킬 수 있다[5].

특히 IETF(Internet Engineering Task force) working group은 위에서 언급한 IoT 네트워크의 특징을 고려하여 RPL(IPv6 Routing Protocol for Low-power and Lossy Networks)을 표준 라우팅 프로토콜로 선정하였다[6-7]. RPL의 상향경로는 하위 센서 노드가 싱크 노드에게 패킷을 전송하며 잦은 상위 센서 노드의 손실을 대비해 다중 경로를 형성한다. 그림 1에서와 같이 상향경로는 싱크 노드에서 루트 노드로 가는 경로를 말하고 반대로 하향경로는 루트 노드에서 싱크 노드로 가는 경로를 말한다.

IoT 네트워크에서는 센서 노드 간의 통신을 위해 싱크 노드를 중심으로 Multipoint-to-Point 전송을 위해 상향경로 DODAG(Destination Oriented Directed Acyclic Graph)를 형성한다. DODAG 형성은 싱크 노드가 DIO(DODAG Information Object) 메시지를 네트워크내의 모든 센서 노드에 플러딩 한다. DIO 메시지를 수신한 센서 노드는 상위 센서 노드 중 라우팅 메트릭 값이 가장 적은 상위 센서 노드를 부모 노드로, 그 외 센서 노드들은 예비 부모 노드로 선정하여 DAG(Directed Acyclic Graph)를 형성한다. 응용의 요구에 따라 싱크 노드가 센서 노드들에 패킷을 전송(Point-to-Multipoint)하는 RPL표준의 하향경로 형성은 다음과 같다.

상향경로를 위한 DIO메시지를 수신한 센서 노드들은 부모 노드 선정 후 자신의 주소 정보를 DAO(Destination Advertisement Object) 메시지에 기록하여 부모 노드에 전송한다. 자식 노드의 DAO 메시지를 수신한 부모 노드는 자신의 주소 정보를 누적 기록하여 부모 노드에 전달한다. 위 과정은 싱크 노드까지 반복되며 DAO 메시지를 수신한 싱크 노드는 라우팅 테이블에 주소 정보를 저장하여 하향경로를 형성한다.

상-하향경로는 IoT 네트워크의 하향경로에서 센서 노드의 손실이 발생하면 이를 발견한 센서 노드가 전체 경로 복구(Global repair)를 위해 DIS 메시지를 부모 노드에 전달한다. DIS 메시지를 수신한 부모 노드 중 라우팅 테이블을 가진 부모 노드 또는 싱크 노드는 DIO메시지를 플러딩하여 하향경로 전체를 재설정한다.

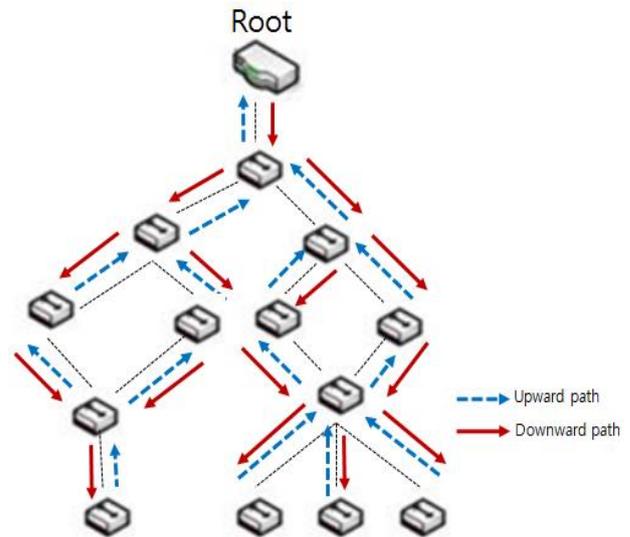


Fig 1. Upward path and Downward path.

그림 1. 상향경로와 하향경로

하지만 낮은 사양의 센서 노드를 사용하는 IoT 네트워크에서는 제한적인 MTU(Maximum Transmission Unit)사이즈로 인해 패킷 전달을 위한 홉의 수가 증가할수록 헤더의 길이가 길어져 throughput이 떨어지는 문제가 발생한다. 위의 문제 해결을 위해 표준 RPL에서는 중간센서 노드를 하향경로를 유지, 관리하기 위한 MOP(Mode of Operation)을 지원한다. MOP는 센서 노드의 동작 모드를 non-storing mode와 storing mode 두 가지로 나누고 있다[8-9]. IoT 네트워크에서 일반적인 동작을 하는 센서 노드를 non-storing mode라 하고 storing mode 센서 노드는 하향경로 전송을 할 때 라우팅 테이블을 이용하여 소스 라우팅하며 자식 센서 노드들의 손실이 발생하면 하위 경로 재설정(Local repair)을 수행한다[10]. IoT 네트워크에서 한 가지 응용이 아닌 여러 응용을 하나의 센서 노드가 수행할 수 있다. 이에 따라 RPL 표준은 하나의 센서 노드는 응용의 요구에 따라 각각 다른 여러개의 DODAG를 형성할 수 있으며 하나의 DODAG를 instance라 한다. IoT 네트워크에 사용되는 센서 노드는 낮은 사양이기 때문에 모든 센서 노드들은 제한된 메모리 크기로 인해 모든 instance에 대해 storing mode로 동작할 수 없다. 하향경로 전송에서 instance에 따른 storing mode로 동작하는 센서 노드의 선정은 하위 센서 노드 손실이 발생하면 경로 재설정 메시지 플러딩으로 인해 네트워크 처리량에 영향을 미친다. 따라

서 각 응용의 instance에 storing mode로 동작하는 센서 노드 선정은 경로 재설정 메시지 플러딩으로 인한 네트워크 throughput에 영향을 줄 수밖에 없다.

본 논문은 IoT 네트워크에서 각 instance에 따른 센서 노드 손실로 인해 경로 재설정을 위한 최적의 storing mode 노드 추가 선정 방법 연구 및 평준화된 경로 재설정 방법을 제안하고자 한다.

II. 본론

1. 관련 연구

가. RPL 상향경로 형성 방법

(1) RPL의 상향경로 형성

RPL에서는 그림 2에서와 같이 각 노드는 센서로부터 수집한 데이터 정보를 싱크 노드에 전달하기 위해 상향경로(MultiPoint-to-Point)를 형성한다. IoT 네트워크의 LLN 특징을 고려하여 Proactive 방식으로 DODAG를 형성한다. 싱크 노드는 IoT 네트워크에서 요구하는 응용에 따라 DODAG를 형성하기 위하여 라우팅 메트릭값을 이용한 DIO메시지를 네트워크 전체에 플러딩 하며 각 센서 노드는 전달된 DIO메시지를 다중의 경로로 수신한다.

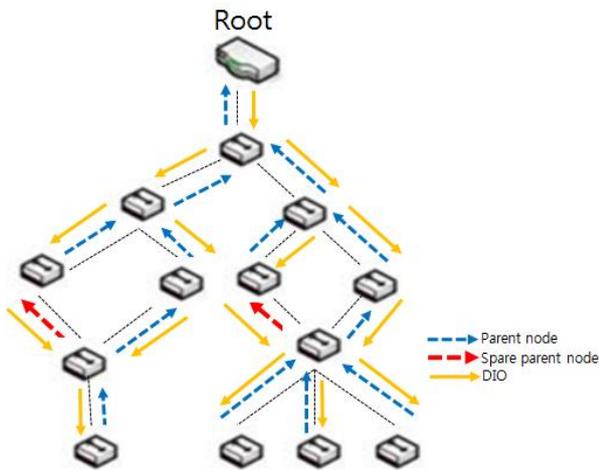


Fig 2. Upward path formation.
그림 2. 상향경로 형성

각 센서 노드는 라우팅 메트릭 값이 가장 적은 노드를 부모 노드로 선정하고 다른 상위 노드를 예비 부모 노드로 선정한다. DAG 형성과정을 반복하여 DODAG를 형성하며 싱크 노드까지 패킷을 전달할 수 있도록 한다. DODAG를 형성한 각 센서

노드는 수집된 데이터 패킷을 부모 노드를 통하여 싱크 노드에게 패킷을 전달한다.

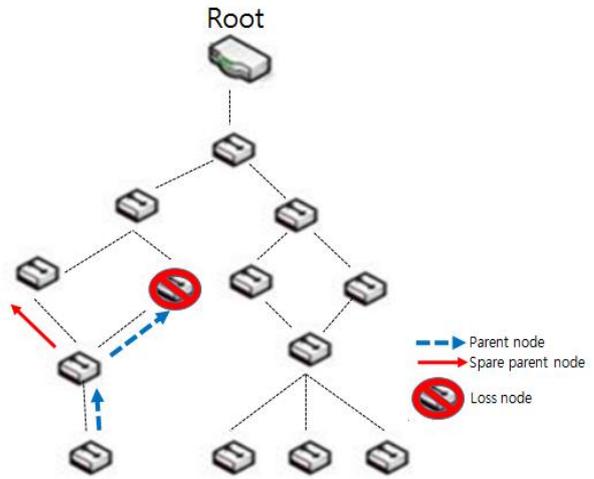


Fig 3. Upward path recovery.
그림 3. 상향경로 복구

(2) RPL 상향경로 복구

IoT 네트워크의 특징인 센서 노드의 이동성과 배터리 소진 등으로 인한 센서 노드의 손실로 인해 데이터 패킷을 전송하지 못하는 경우가 흔히 발생할 수 있다. 위와 같은 문제로 인해 네트워크의 토폴로지가 바뀌었을 경우 IoT 네트워크 내에서 감지한 데이터 패킷을 싱크 노드에게까지 전달하지 못하는 경우가 발생한다. 이러면 RPL 표준에서는 하위 노드가 상위 노드인 부모 노드의 손실이 감지되었을 때 DAG 형성과정 중 선정한 예비 부모 노드를 부모 노드로 선정하여 데이터 패킷을 싱크 노드에게까지 전달한다. 그림 3에서와 같이 상향경로에서 손실 node가 발생했을 때 여분의 부모노드를 선정하여 복구되는 것을 보여주고 있다.

나. RPL 하향경로

(1) RPL의 하향경로 형성

하향경로는 싱크 노드가 IoT 네트워크 내의 각 센서 노드들에 데이터 패킷을 전송하는(Point-to-Multipoint) 방식으로 이루어진 경로를 하향경로라 한다. RPL 표준에서 하향경로 설정 과정은 옵션으로 제공하고 있다[7]. 싱크 노드는 하향경로 형성을 위해 네트워크 내에 DIO메시지를 플러딩 전송한다. DIO메시지를 수신한 각 센서 노드는 DAO 메시지에 자신의 IP 주소 정보를 삽입한 후 DODAG의 미리 선정된 부모 노드

에 메시지를 전달한다. 상위 노드인 부모 노드는 하위 자식 노드에 수신한 DAO 메시지에 자신의 주소 정보도 포함해 위의 상위 부모 노드에 전달한다. 위의 과정을 반복 수행하여 싱크 노드에까지 전달하는 방법을 이용하여 싱크 노드는 DODAG 전체 노드의 주소 정보를 수집할 수 있다. 싱크 노드는 각 센서 노드에 데이터 패킷을 전달할 때 센서 노드의 주소 정보를 이용하여 패킷 헤더에 주소 정보를 포함해 데이터 패킷을 전송한다.

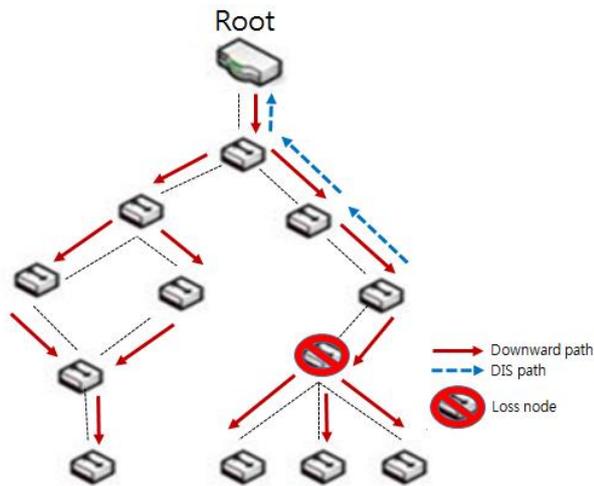


Fig 4. Downward path recovery.
그림 4. 하향경로 복구

(2) RPL의 하향경로 복구

하향경로 복구는 그림 4에서와 같이 상향경로 복구와는 반대로 상위 노드가 하위 노드의 손실을 파악하고 확인한다. 하위 노드의 손실을 확인한 상위 노드 즉, 부모 노드는 자식 노드의 존재 여부를 확인한다. 자식 노드가 존재할 경우 DIO 메시지를 브로드 캐스팅하여 하위 노드와의 경로를 재설정하고, 만약 자식 노드가 없다면 DIS 메시지를 상위 노드에 전송한다. 상위 노드는 DIS 메시지 수신 이후 위의 상위 노드에 다시 전송하는 과정을 반복 수행한다. 경로 재설정을 수행하며 DIS 메시지를 싱크 노드가 수신한다면 DODAG 전체의 네트워크 경로 재설정인 Global repair를 수행한다.

다. Storing Mode

IoT 네트워크의 규모는 최근 점점 증가하는 추세이며, 이로 인해 DODAG의 센서 노드 수도 증가하고 있다. 센서 노드의 증가는 하향경로일 때, 싱크

노드로부터 목적지 센서 노드까지의 헤더에 많은 주소 정보를 포함해야 한다. 이는 제한적인 MTU (Maximum Transmission Unit)사이즈로 인해 많은 데이터 정보를 전송할 수 없게 된다. 주소 정보로 인해 헤더 치수가 커지기 때문에 네트워크 전체에 엄청난 에너지 및 대역폭 등 네트워크 자원의 소모가 일어나 처리량은 감소한다. 이와 같은 문제를 해결하기 위해 RPL 표준에서는 Storing mode를 지원한다. Storing mode란 노드의 손실 및 네트워크의 규모가 큰 경우 처리량 및 경로 재설정을 유지하고 관리하는 센서 노드이다. Storing mode의 MOP(Mode of operation)은 RPL 표준에서 정의하고 있다 [11-12]. MOP는 DIO메시지에 포함되어 전송되며 DODAG내의 센서 노드는 Non-Storing mode와 Storing mode로 나뉘어 동작한다. Non-Storing mode는 기존의 DODAG내의 싱크 노드 기반의 센서 노드로 동작하며, Storing mode로 동작하는 센서 노드는 싱크 노드와 같은 동작 모드를 지원한다.

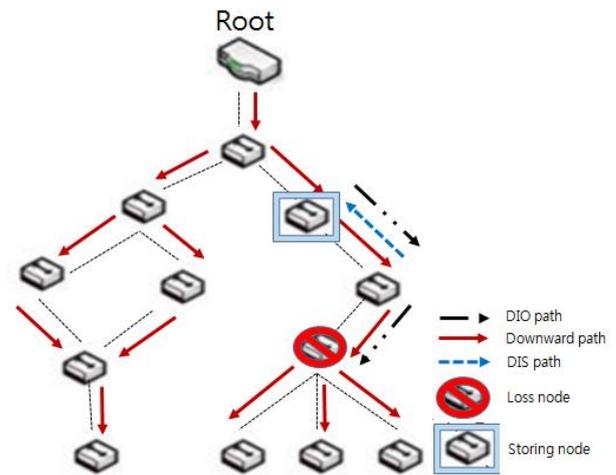


Fig 5. Downward path recovery in Storing mode.
그림 5. Storing mode의 하향경로 복구

DIO메시지에 포함된 MOP에 따라 DODAG의 하향 경로에서 Storing mode로 동작하는 센서 노드는 하위 센서 노드의 주소 정보를 갖고 있으며, 하향경로에서 센서 노드의 손실이 발생하면 DIS 메시지를 수신한 이후 DIS 메시지의 주소 정보를 하위 센서 노드와 비교 후 DIS 메시지를 싱크 노드까지 전송하지 않고 Storing mode 센서 노드가 스스로 DIO메시지를 브로드 캐스트하여 하위 노드의

경로 재설정을 수행한다. 이를 local repair라 하며, 하위 센서 노드에게 데이터 패킷을 전송할 경우 싱크 노드로부터 주소정보를 전달받아 Storing 노드 자신의 주소 정보와 비교 후 하위 센서 노드에 소스 라우팅으로 데이터 패킷을 전송한다. 그림 5는 Storing mode에서 손실 노드가 발생했을 때, DIS 메시지와 DIO메시지의 전송 방향과 복구과정을 보이고 있다.

라. Instance

최근 IoT 네트워크의 센서 노드는 여러 응용에 대해 센싱 및 데이터 수집이 가능하도록 설계되고 있다[13-14]. RPL표준에서는 싱크 노드는 각 Instance를 구분하기 위해 DIO 메시지에 RPL Instance ID와 라우팅 메트릭 값 설정 후 플러딩 한다. DIO메시지를 수신한 하위 센서 노드들은 RPL Instance ID를 확인하고 라우팅 메트릭 값에 따라 각 Instance별 부모 노드를 선정하여 데이터 전송을 수행한다. 그림 6에서와 같이 Instance는 같은 토폴로지 내에서 서로 다른 응용에 독립적 라우팅을 제공한다. 다른 경로를 통해 다수의 Instance 형성과정을 보여주고 있다.

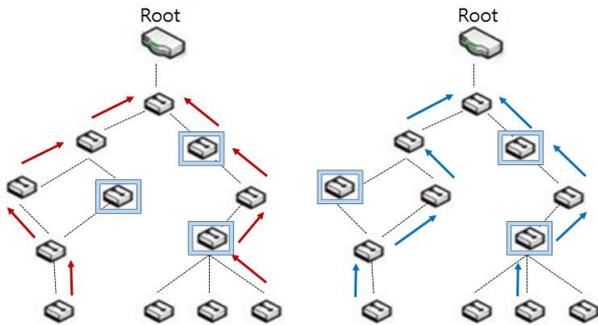


Fig 6. Multiple instances in the same topology.
그림 6. 동일 토폴로지에 다수의 Instance 형성

2. 제안기법

IoT 네트워크에서 RPL의 표준에서 정의하고 있는 하향경로 복구 방법은 Storing mode 노드의 유무에 따라 Global repair와 Local repair로 나눌 수 있다. IoT 네트워크에서 하향경로로 데이터 전송 도중 하위 센서 노드의 손실이 일어날 때 DIS 메시지의 플러딩하여 하향경로 재설정을 수행한다. Storing mode 노드 역시 손실이 일어날 경우, 혹은 Storing mode 노드에 소속되어 있는 하위 센서 노

드의 잦은 손실이 발생하면 Storing mode 노드의 하위 센서 노드들 또한 잦은 DIS 메시지 플러딩과 경로 재형성(Local repair)을 수행한다[15]. 싱크 노드가 전송하는 데이터 패킷, 컨트롤 패킷 등 상위 노드로부터 전송되어지는 패킷 수신이 어려울 수 있다.

위의 문제 해결 방법으로는 기존 IoT 네트워크에서 DODAG형성 이후 싱크 노드가 전송하는 DIO 메시지를 통한 Storing mode 노드 선정 방법에 추가적인 Storing mode 노드 선정이 필요하다. 즉, 기존 RPL 표준에서 정하는 각 Instance에 따른 Storing mode 노드 선정 방법에 있어 Storing mode 노드를 추가 선정하는 방법을 제안하고자 한다. 또한, IoT 네트워크 노드의 Low power Lossy Networks 및 다수 Instance의 소속되어 동작하는 특징으로 인한 메모리 공간 부족 및 에너지 소비 문제를 고려하여 Storing mode 노드를 선정하여야 한다.

가. Local repair를 위한 Storing mode 추가 기법

(A 기법)

RPL에서 Sink 노드를 제외한 모든 센서 노드는 하위 노드의 손실을 감지하면 DIS 메시지를 상위 노드에 전달하여 경로 재설정을 수행한다. 이때, Sink 노드와 같이 동작하는 Storing mode 노드로 선정된 센서 노드의 하위 노드에 있는 센서 노드 손실이 발생하면 하위 전송되는 DIS 메시지를 수신한 Storing mode 노드는 하위 노드의 손실을 확인할 수 있다. 이후 Storing mode 노드는 DIO 메시지를 이용하여 경로 재설정인 Local repair를 수행한다.

이러한 문제 해결 방법으로는 손실된 노드의 부모 노드는 다른 Instance에서 Storing mode 노드로써 동작 여부와 홉-카운트에 따른 숫자를 DIS 메시지 필드에 포함하여 전송한다. DIS 메시지를 수신한 부모 노드는 홉-카운트에 숫자를 증가시켜 상위 부모 노드에 전송한다. 하위 노드로부터 DIS 메시지를 수신한 Storing mode 노드는 가진 주소 정보를 홉-카운트 숫자 필드와 비교하여 DODAG에 손실된 노드의 위치와 부모 노드의 응용에 따른 Instance에서의 Storing mode 노드로써 동작하는지를 파악할 수 있다.

(1) 손실된 노드의 상위 노드가 다른 Instance에

서 Storing mode 노드로 동작할 경우, DIO메시지 필드에 추가로 Storing mode 노드로 추가 선정될 노드는 DIS 메시지를 생성한 노드의 부모 노드가 선정되며, 추가 선정된 노드 역시 만약 다른 Instance에서 Storing mode 노드로 선정되어 동작하고 있다면 위의 부모 노드에 DAO 메시지를 통해 Storing mode 노드 역할을 요청한다.

(2) 반대로 손실된 노드의 부모 노드가 다른 Instance에서도 Non-Storing mode 노드로 동작한다면 DIO메시지 수신하여 주소 필드 확인 후 추가적인 Storing mode 노드로 동작한다.

나. Global repair를 위한 Storing mode 노드 추가 기법(B 기법)

Instance 내에서 Storing mode 노드로 동작하는 노드가 존재한다면 Global repair는 자주 일어나지 않는다. 하지만 싱크 노드로부터 첫 번째로 동작하는 Storing mode 노드 혹은 그사이의 노드에 손실이 일어난다면 DIS 메시지는 Sink 노드에 전달된다. 센서 노드 손실로 인해 Sink 노드에 전달되는 DIS 메시지는 Local repair와는 또 다른 정보를 포함시켜 전달된다. Local repair에서는 기존의 Storing mode 노드로 동작하는 노드의 손실에 대한 repair에 대해 고려할 필요가 없었기 때문이다. 하지만 Global repair에서는 Non-Storing mode 노드와 Storing mode 노드 모두의 손실을 고려하여야 한다. 이로 인해 노드 손실이 발생하면 DIS 메시지를 생성한 노드의 주소를 포함하여 전송하여야 한다.

Sink 노드는 DIO메시지 필드의 주소 정보를 자신의 주소 정보와 비교하여 손실이 일어난 노드가 Non-Storing mode 노드인지 혹은 Storing mode 노드인지 확인한다.

(1) 손실이 발생한 노드가 Non-storing mode 노드라면 Local repair와 같은 동작인 DIO 메시지 필드에 포함된 홉-카운트 정보와 주소 정보를 비교하여 손실이 발생한 노드의 부모 노드를 Storing mode 노드로 동작시킨다. Storing mode 노드로 선정된 노드가 Local repair와 같이 다른 Instance의 Storing mode 노드로 동작한다면 부모 노드가 Storing mode 노드로 동작한다.

(2) 노드 손실이 발생한 노드가 Storing mode 노드로 동작한 노드라면 상위 부모 노드로부터 받는 주소 정보는 하위 노드인 Storing mode 노드로 동

작하는 주소가 마지막 주소이다. 주소 정보 중 마지막 주소 정보의 노드가 손실이 발생한다면 그 노드의 주소 정보를 DIO 메시지에 포함해 전송하여 Storing mode 노드의 손실을 Sink 노드에 전달할 수 있다. Sink 노드는 자신의 주소 정보와 비교하여 Storing mode 노드의 손실이 발생하였을 경우 손실된 노드 이후의 주소 정보 또한 손실된 것으로 파악하고 Global repair를 수행한다. Local repair와 Global repair는 Non-Storing mode로 동작하는 노드의 경로 재설정과 Storing mode 노드로 동작하는 노드의 손실이 발생하면 DIS 메시지를 수신하여 repair 하는 센서 노드의 성격에 따라 동작 방법이 다르다.

다. DIS 메시지를 이용한 각 Instance의 Global repair 평준화 기법(C 기법)

앞에서는 하나의 Instance 내 Global repair와 Local repair에서 추가적인 Storing mode 노드 추가 기법을 제안하였다. 이번 세션에서는 DIS 메시지 카운트를 이용한 DODAG에서 각 Instance의 Global repair평준화를 위한 방법을 제안한다. DODAG에서 Sink 노드는 여러개의 Instance를 DIO 메시지의 Instance ID를 통해 구분한다. Sink 노드는 각 Instance의 DIS 메시지를 수신할 경우 Instance ID를 확인하여 Global repair를 수행한다. Global repair는 각 Instance의 응용 및 Storing mode 노드의 개수 및 선정 방법에 따라 횟수가 다르다. Sink 노드는 각 Instance의 Global repair 횟수를 DIS 메시지 카운트를 이용하여 확인할 수 있고, 평균 Global repair의 횟수를 구할 수 있다. 평균 Global repair 횟수를 이용하여 각 Instance의 Global repair 횟수와의 비교를 통해 임의의 Storing mode 노드를 선정 및 교체할 수 있도록 하는 방법이 필요하다. 평균의 Global repair보다 높은 Instance의 경우 Storing mode 노드의 추가 선정 및 Non-Storing mode 노드의 Storing mode 노드 전환이 필요하다.

3. 성능평가

가. 시뮬레이션 환경

앞에서 제안한 기법인 (A),(B),(C)의 기존 RPL 동작에서의 타당성 및 성능 실험을 위해 NS-2(Network Simulator -2)를 사용하여 실험에 임하였다. 성능 실험을 위해 400여 개의 노드를 랜덤하게 배치하여

네트워크를 구성하였으며, IoT의 저 사양 특성에 따라 MTU의 크기는 500byte로 설정하였다. 각 노드의 전송 거리는 100m, 노드의 배치 공간은 3km × 3km로 하였으며 최소 싱크로부터 7홉 이상 떨어진 노드를 Storing mode 노드로 설정하였다. 또한, 각 노드는 0.5~15%까지의 확률로 패킷 전송의 손실을 유발하였으며 네트워크 내의 Instance는 3~8개 랜덤하게 설정하여 반복 실험하였다.

나. 시뮬레이션 결과

본 시뮬레이션에서는 하나의 Instance에서 Global repair와 Local repair에 소비되는 네트워크 내의 총 에너지 소비량과 하나의 DODAG내에서 Instance의 수에 따른 Global repair와 Local repair에 소비되는 시간, 마지막으로 Sink 노드의 Storing mode 노드 추가 기법을 NS -2 시뮬레이션을 이용하였다[16].

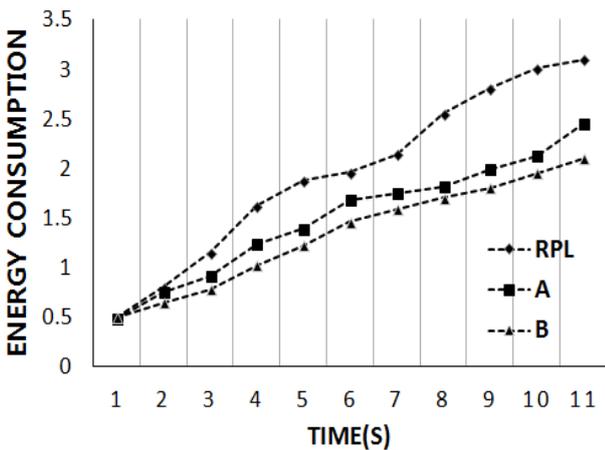


Fig 7. Total network energy consumption per hour(A, B).
그림 7. 시간당 네트워크 전체 에너지 소모량(A, B 기법)

그림 7에서는 Global repair와 Local repair에 그리고 패킷 전송 등 네트워크 전체에서 소비하는 에너지 소비량을 나타내었다.

그림에서 보는 것과 같이 총 네트워크 내의 전체 에너지 소비량은 시간에 따라 증가하지만, 그 증가량은 처음과 달리 제안기법에서 특히 Global repair 제안기법에서 많은 소비량이 줄어든 것을 알 수 있다. 이와 같은 결과는 Storing mode 노드의 추가 또한 에너지 소비량을 줄일 수 있지만, Global repair의 횟수에 따라 전체 네트워크 에너지 소비량이 줄어들 수 있다는 것을 알 수 있다.

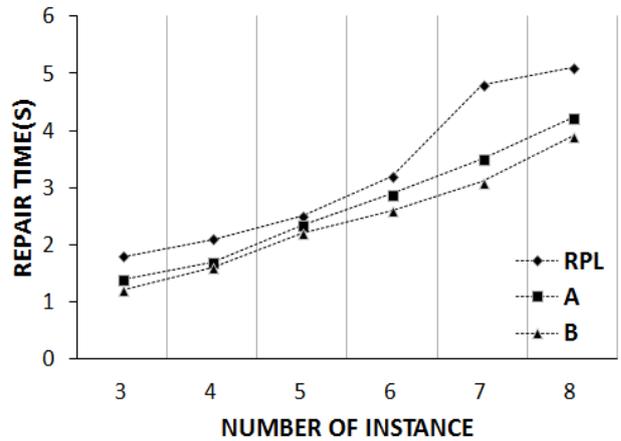


Fig 8. Global repair, Local repair Consumption time by number of Instance(A, B).
그림 8. Instance 수에 따른 Global repair, Local repair 소비시간(A, B 기법)

그림 8에서는 하나의 DODAG에서 Instance의 수에 따라 A 기법과 B 기법 그리고 표준 RPL에서의 Global repair 혹은 Local repair에 소비하는 시간의 차이를 보여주고 있다. 제안기법 A와 B는 Instance의 수가 증가함에 따라 repair에 소비되는 시간 선형적으로 증가하고 그 증가 폭 역시 두 기법이 비슷한 것을 알 수 있다. 하지만 RPL의 표준에서는 repair에 소비하는 시간의 증가 폭이 선형적이지 않으며, 특정 Instance의 개수부터는 현저하게 많은 시간을 Global 혹은 Local repair 등 repair 하는 시간을 소비하는 특징을 보였다.

마지막으로 제안하는 기법 C는 Instance의 수에 따른 특정한 변화를 보여주지 않아 시간에 따른 각 Instance 별 Global repair 횟수를 나타내었다.

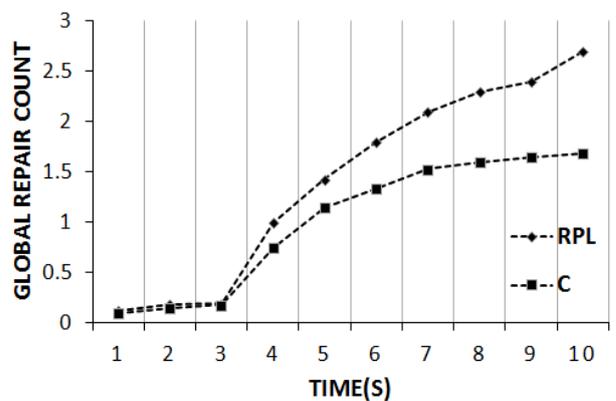


Fig 9. Sink node average Global repair count(C).
그림 9. Sink 노드 평균 Global repair 횟수(C 기법)

이 횡수는 시간에 따라 각 Instance 별 노드 손실로 인한 Global repair 횡수의 차이를 보이는 만큼 기법 C를 적용할 경우 일정 시간 이후에도 총 Instance의 Global repair의 평균 횡수가 많이 늘어나지 않음을 알 수 있다. 이는 평균 Global repair 값을 기준으로 Storing mode 노드를 조절하기 때문에 시간이 지남에 따라 최대 Global repair 수와 최저 Global repair 수의 차이가 크지 않기 때문으로 분석된다.

III. 결론

표준 RPL에서 제공하고 있는 MOP로써 Storing mode 노드는 Sink 노드의 부하를 줄이고 여러 응용이 하나의 DODAG에서 동작할 때 패킷의 전송 및 노드 손실로 인한 repair기능을 제공하고 있다. 하지만 표준에서 제공하는 MOP는 뚜렷한 기준 없이 응용에 따라 사용자에게 설정하도록 하고 있다. 본 논문에서는 이러한 MOP 설정 방법에 있어 표준의 동작을 해치지 않고 여러 응용이 존재할 경우 노드의 손실로 인해 repair에 소비되는 자원을 줄일 방법을 제안하였다. 이미 설정된 MOP의 Storing mode 노드의 추가적인 선정 방법을 제안하고 있으며 노드의 손실에 따라 더욱 효과적인 Global repair와 Local repair를 수행할 방법을 제안했다. 또한, 여러 개의 Instance가 동작할 경우 각 Instance의 응용에 따라 노드 손실이 일어날 때 repair에 소비되는 자원을 줄이려는 방법을 제안했으며, 시뮬레이션 결과를 통해 입증을 도출하였다. 하지만 C 기법은 여러 Instance 중 응용의 우선순위를 고려하지 않았기에 다음에는 우선순위를 고려한 여러 Instance의 repair 자원을 줄이는 방법을 고안할 예정이다.

References

[1] Z. Sheng, S. Yang, Y. Yu, A. V. Vasilakos, J. A. Mccann and K. K. Leung, "A survey on the ietf protocol suite for the internet of things: standards, challenges, and opportunities," *IEEE Wireless Communications*, vol.20, no.6, pp. 91-98, December 2013. DOI:10.1109/MWC.2013.6704479

[2] P. Thulasiraman, "RPL routing for multigateway AMI networks under interference constraints,"

IEEE International Conference on Communications, Budapest, 2013. DOI:10.1109/ICC.2013.6655272

[3] E. Ancillotti, R. Bruno and M. Conti, "The role of the RPL routing protocol for smart grid communications," *IEEE Communications Magazine*, vol.51, no.1, pp.75-83, January 2013. DOI:10.1109/MCOM.2013.6400442

[4] <http://www.postscapes.com/internet-of-things-hardware/#iot-boardcomparison/view-hardware-details3/>

[5] D. Gadde and M. S. Chaudhari, "Survey on routing protocol for low-power and lossy networks," *IEEE International Conference on Computational Intelligence and Computing Research*, Madurai, 2015. DOI:10.1109/ICCIC.2015.7435775

[6] T. Winter and ROLL Design Team, "RPL: Routing protocol for low power and lossy networks," *Netw. Working Group Internet Drafts*, July 2009.

[7] E. Aljarrah, M. B. Yassein and S. Aljawarneh, "Routing protocol of low-power and lossy network: Survey and open issues," *International Conference on Engineering & MIS*, Agadir, 2016. DOI:10.1109/ICEMIS.2016.7745304

[8] M. C. R. Anand and M. P. Tahiliani, "mRPL++: Smarter-HOP for optimizing mobility in RPL," *IEEE Region 10 Symposium*, Bali, 2016. DOI:10.1109/TENCONSpring.2016.7519374

[9] Mayzaud, A., Sehgal, A., Badonnel, R., Chrisment, I. and Sch'önw'alder, J, "Using the RPL Protocol for Supporting Passive Monitoring in the Internet of Things," *IEEE/IFIP Network Operations and Management Symposium*, April, 2016. DOI:10.1109/NOMS.2016.7502833

[10] S. Elyengui, R. Bouhouchi and T. Ezzedine, "A comparative performance study of the routing protocols RPL, LOADng and LOADng-CTP with bidirectional traffic for AMI scenario," *International Conference on Smart Grid and Clean Energy Technologies*, Offenburg, 2015. DOI:10.1109/ICSGCE.2015.7454267

[11] Wei Gan, Zhiqiang Shi, Chen Zhang, Limin Sun and D. Ionescu, "MERPL: A more memory-

efficient storing mode in RPL,” *IEEE International Conference on Networks*, Singapore, 2013.

DOI:10.1109/ICON.2013.6781985

[12] Ko, J., Jeong, J., Park, J., Jun, J. A., Gnawali, O. and Paek, J, “DualMOPRPL: Supporting Multiple Modes of Downward Routing in a Single RPL Network,” *ACM Transactions on Sensor Network*, vol. 11, no. 2, pp. 1-20, February 2015.

DOI: 10.1145/2700261

[13] S. Yessad, N. Tazarart, L. Bakli, L. Medjkoune-Bouallouche and D. Aissani, “Balanced energy efficient routing protocol for WSN,” *International Conference on Communications and Information Technology*, Hammamet, 2012.

DOI:10.1109/ICCITechnol.2012.6285817

[14] S. S. Wang, H. T. Liu and Y. D. Chen, “Link quality aware routing protocol for low-power and lossy networks,” *IEEE Wireless Communications and Networking Conference*, Istanbul, 2014.

DOI:10.1109/WCNC.2014.6952816

[15] J. Guo, P. Orlik, K. Parsons, K. Ishibashi and D. Takita, “Resource Aware Routing Protocol in Heterogeneous Wireless Machine-to-Machine Networks,” *IEEE Global Communications Conference*, San Diego, CA, 2015.

DOI:10.1109/GLOCOM.2015.7417203

[16] NS3, ”<http://www.nsnam.org/>,” 2016.

Jeong-oh Yun (Member)



1989 : BS degree in Electronics Engineering, Kyungpook National University.

1991 : MS degree in Electronics Engineering, Kyungpook National University.

2002 : PhD degree in Electronics Engineering, Kyungpook National University.

1998~Present : Professor, KyungWoon University.

BIOGRAPHY

Sung-hyun Bae (Member)



2000 : BS degree in Electronics and electrical Engineering, Kyungpook National University.

2003 : MS degree in Electronics Engineering, Kyungpook National University.

2005 : PhD course completion in Electronics Engineering, Kyungpook National University.

2017~Present : Assistant Professor, KyungWoon University.