

멀티 싱크 기반 IoT 네트워크에서 불필요한 경로 업데이트 문제와 두 가지 해결 기법

이성원*, 강현우*, 유홍석**, 정용환***, 김동균^o

Two Solutions for Unnecessary Path Update Problem in Multi-Sink Based IoT Networks

Sungwon Lee*, Hyunwoo Kang*, Hongsoek Yoo**, Yonghwan Jeong***, Dongkyun Kim^o

요약

최근 IoT(Internet of Things)에 대한 관심이 증가함에 따라, IoT의 특징을 고려한 새로운 프로토콜에 대한 연구 및 표준화가 진행되고 있다. 그 중 RPL(IPv6 for Low-Power Lossy Network)은 IoT를 위해 표준화된 라우팅 프로토콜이다. RPL은 싱크 노드로부터 네트워크 전체로 플러딩 되는 DIO(DODAG Information Object) 메시지를 사용하여 라우팅 경로를 생성 및 관리한다. 그러나, 대규모 네트워크에서는 네트워크 전체에 DIO 메시지를 전파하는 데 많은 시간이 소모될 뿐만 아니라 싱크 노드 주변에서 데이터 패킷의 병목 현상이 발생하게 된다. 이러한 문제를 해결하기 위해서 라우팅 오버헤드 및 병목 현상을 감소시킬 수 있는 멀티 싱크 기반 접근법이 널리 활용된다. 이 논문에서는 멀티 싱크 기반 IoT 네트워크에 RPL 프로토콜을 적용 시 발생할 수 있는 ‘불필요한 경로 업데이트 문제’를 정의하고, 선택적인 경로 업데이트를 위한 2가지 기법인 Routing Metric based Path Update Decision 기법과 Immediate Successor based Path Update Decision 기법을 제안한다.

Key Words : Internet of Things, Multi-sink, Routing Overhead, RPL

ABSTRACT

Recently, as interest in IoT (Internet of Things) increase, research and standardization of a new protocol which reflects the characteristics of IoT has progressed. Among them, RPL(IPv6 for Low-Power Lossy Network) is a standardized routing protocol for IoT. RPL utilizes DIO (DODAG Information Object) messages which is flooded from the sink node to the whole network for path establish and maintenance. However, in large scale networks, not only a long time is required to propagate the DIO message to the whole networks but also a bottleneck effect around the sink node is occurred. Multi-sink based approaches which take advantage of reducing routing overhead and bottleneck effect are widely used to solve these problems. In this paper, we define ‘unnecessary path update problems’ that may arise when applying the RPL protocol to the multi sink based IoT networks and propose two methods namely Routing Metric based Path Update Decision method and Immediate Successor based Path Update Decision method for selective routing update.

※ 본 연구는 대경권 지역산업연계 IT융합기술개발 및 산업계 지원사업의 결과물로 수행되었습니다.

♦ First Author : Kyungpook National University School of Computer Science & Engineering, swlee@monet.knu.ac.kr, 정회원

^o Corresponding Author : Kyungpook National University School of Computer Science & Engineering, dongkyun@knu.ac.kr, 종신회원

* Electronics and Telecommunications Research Institute, Daegu-Gyeongbuk Research Center, hwkang@etri.re.kr, 정회원

** Kyungwoon University, Department of Mobile Engineering, hsyoo@ikw.re.kr, 정회원

*** Kyungpook National University School of Computer Science & Engineering, 학생회원

논문번호 : KICS2015-09-277, Received September 1, 2015; Revised December 15, 2015; Accepted December 15, 2015

I. 서 론

최근 네트워크 인터페이스의 가격 저하 및 소형화에 대한 기술이 발전함에 따라 기계, 센서 등의 다양한 사물들에게 네트워크를 제공하는 무선 IoT에 대한 관심이 증가하고 있다. 이러한 IoT 단말들은 사람의 개입 없이 자율적으로 형성되어 스마트홈, 헬스케어, 홈 보안 등의 다양한 응용을 제공하고 있다. 하지만 이러한 소형화된 통신 인터페이스는 상대적으로 좁은 대역폭, 낮은 전송 신뢰성, 제한적인 배터리 용량으로 인한 노드 손실 및 낮은 연산 능력 등의 통신 제약 사항이 존재하므로 이를 고려한 새로운 프로토콜에 대한 연구 및 표준화가 진행되고 있다^{1,2)}.

그 중에서도 RPL (IPv6 for Low-Power Lossy Networks)은 무선 IoT를 대상으로 IETF (Internet Engineering Task Force)에서 표준화 중인 대표적인 라우팅 프로토콜이다³⁾. 현재 표준화된 RPL에서는 싱크 노드를 목적지로 하는 비순환성 방향 그래프인 DADOG (Destination Oriented Directed Acyclic Graph)를 생성하고, 각 센서 노드들이 DADOG 상의 부모 노드를 다음 Hop으로 하여 싱크 노드에게 패킷을 전달할 수 있는 기능을 제공한다. RPL에서 DADOG를 생성하는 기본적인 동작은 다음과 같다.

싱크 노드는 주기적으로 네트워크 전체로 플러딩 되는 DIO (DODAG Information Object) 메시지를 전송하고 이를 수신한 노드는 DIO 메시지를 전송한 이전 노드를 부모 노드 (parent node)로 설정한다. 부모 노드를 설정한 노드들은 수신된 DIO 메시지를 다시 하위노드들에게 브로드캐스트한다. 이후 다른 경로로 전송된 새로운 DIO 메시지가 수신될 경우 센서 노드는 최소의 라우팅 메트릭 (RPL 표준에서는 objective function) 값을 가진 DIO 메시지를 전송한 노드를 부모 노드로, 나머지 DIO 메시지를 전송한 노드들을 예비 부모 (immediate successor) 노드로 선정하여 라우팅 테이블에 기록한다. 이 과정에서 새로운 부모 노드가 선정될 경우 새로운 DIO 메시지를 다시 전송하여 자식 노드들에게 해당 사실과 새로운 라우팅 메트릭 값을 알려주는 라우팅 업데이트를 수행한다. 이후 부모 노드가 이동, 배터리 소진, 외부 충격 등으로 인해 손실될 경우 자신의 라우팅 테이블에 저장된 예비 부모 노드들 중 최소의 라우팅 메트릭 값을 가지는 노드를 새로운 부모 노드로 선정하여 패킷을 전송할 수 있다.

DADOG가 형성된 후 각 센서 노드는 싱크 노드에게 패킷을 전송할 수 있으나, 싱크 노드는 각 센서 노

드들에게 전송할 패킷의 다음 홉을 결정할 수 없는 bi-directional 한 경로가 형성된다. 실제 IoT 응용에 따라서는 싱크 노드의 패킷 및 제어 메시지 등이 센서 노드들에게 전달되어야 할 필요가 있으므로 RPL 표준에서는 하향 경로 생성 기능을 옵션으로써 제공하고 있다. 하향 경로를 설정하기 위해서 부모 노드를 설정한 센서 노드가 DAO (Destination Advertisements Object) 메시지를 싱크 노드에게 전달하며, 이 과정에서 DAO 메시지가 경유하는 모든 중간 노드의 주소 정보를 DAO 메시지에 기록한다. 싱크 노드가 이 DAO 메시지를 수신할 경우 해당 메시지에 기록된 주소 정보를 활용한 소스 라우팅을 수행한다.

RPL 등과 같이 IoT 네트워크를 대상으로 연구된 다양한 프로토콜들은 IPv6를 활용한 홉 네트워크 등 독립적이고 소규모인 네트워크인 6LoWPAN을 대상으로 최적화되었다. 하지만 최근에는 다수의 IoT 단말이 대규모의 네트워크를 구성하거나 하나의 노드가 다수의 센서/액추에이터를 장착하여 서로 다른 IoT 네트워크에 동시에 소속되어 다양한 응용을 저렴한 비용으로 제공하는 사례가 등장하고 있다^{4,5)}. 이러한 사례로는 드론, 노동자 및 공장 장비에 부착한 센서 노드들이 각각 IoT 네트워크를 구성하고, 구성된 네트워크들이 함께 동작하여 안전, 불량 및 물품 관리를 통합적으로 수행하는 스마트 팩토리 등이 해당한다. 대규모 IoT 네트워크에서는 센서 노드와 싱크 노드 사이의 경로 설정에 긴 지연 시간이 발생할 뿐 아니라 싱크 노드 주변에서 데이터 패킷이 집중되는 병목 현상이 발생하므로, 다수의 싱크 노드들을 설치하여 이러한 문제를 해결하는 새로운 RPL 프로토콜에 대한 연구가 진행되고 있다. 현재 연구되고 있는 멀티 싱크 기반 RPL 프로토콜들은 센서 노드들이 패킷을 전송하기에 최적인 싱크 노드를 선정함으로써 이러한 문제들을 해결하고 있다⁶⁾.

하지만 다수의 소규모 IoT 네트워크가 모여 생성된 대규모 IoT 네트워크에서는 최적인 싱크 노드를 선정한 후에도 제어 메시지 등 자신이 소속된 소규모 네트워크의 싱크 노드에게만 전송되어야 하는 패킷(DHCP 메시지 등)이 주기적으로 생성된다. 본 논문에서는 이를 싱크 지향적 패킷으로 정의한다. 각 노드는 싱크 지향적 패킷의 목적지인 싱크 노드로 향하는 경로를 보유하지 않은 경우 해당 패킷을 폐기하므로, 각 노드들은 각 싱크 노드를 목적지로 하는 경로를 최소 하나 이상 유지해야 한다. 따라서 멀티 싱크 환경에서 센서 노드는 새로 수신된 DIO 메시지의 라우팅 메트릭 값이 자신의 라우팅 테이블에 기록된 라우팅 메트릭 값

보다 더 크더라도 이 DIO 메시지를 자식 노드에게 전송하여 새로운 경로를 알려주어야 하는 상황이 발생한다(이러한 상황에 대한 자세한 설명은 III장에서 다루도록 한다).

본 논문에서는 멀티 싱크 환경에서 새로운 DIO 메시지가 수신될 때 자식 노드들에게 전송되는 DIO 메시지로 인해 라우팅 오버헤드가 불필요하게 증가될 수 있음을 지적하고 이러한 문제를 불필요한 경로 업데이트 문제라고 정의한다. 또한 새로운 DIO 메시지가 수신될 경우 라우팅 메트릭 값과 예비 경로의 유효성을 기반으로 자식 노드들에게 전송될 DIO 메시지가 유효한지를 예측하고, 해당 DIO 메시지 전송이 불필요하다고 판단될 경우 이를 폐기함으로써 경로 업데이트 문제를 해결하는 두 가지 방식을 제안한다.

본 논문의 나머지 구성은 다음과 같다. II장에서는 멀티 싱크 환경을 대상으로 연구된 기존 RPL 프로토콜에 대해 설명한 후 라우팅 오버헤드 감소 기법들에 대해 살펴본다. III장에서는 본 논문에서 가정하는 환경과 불필요한 경로 업데이트 문제에 대해 자세히 설명한 후 제안된 두 가지 해결 기법에 대해 설명한다. IV장에서는 NS-2 시뮬레이터를 사용하여 제안된 기법들의 성능을 평가하고, 그 결과에 대해 논의한다. 마지막으로 V장에서 결론을 맺는다.

II. 관련 연구

2.1 멀티 싱크 기반 라우팅 프로토콜

최근에는 RPL 프로토콜이 멀티 싱크 기반 네트워크에 적용될 경우 통신 효율을 최적화하기 위해 서로 다른 DADOG 간의 통신 기법, 싱크 노드 사이의 동기화 기법 및 관련 통신 방식 등이 제안되었다⁶⁾. 특히 이 논문에서는 서로 다른 DADOG 에 소속된 단말 간의 통신이 IoT 응용에서 요구될 수 있음을 지적하고 이를 지원하기 위해 특정 DADOG에서 다른 DADOG로 패킷을 전달하는 3가지 방식을 제안하고 있다.

첫 번째 방식은 그림 1-(a)와 같이 자신의 싱크 노드에게 패킷을 전달하면 이를 수신한 싱크 노드가 백본 등의 별도의 네트워크를 사용하여 해당되는 DADOG로 패킷을 전달하는 방식으로 일반적인 멀티 싱크 환경에서 다른 네트워크로 패킷을 전달하기 위해 사용되는 널리 알려진 방식이다.

하지만 이 방식은 싱크 노드가 다른 DADOG 의 단말 정보를 모두 공유해야 한다는 제약사항을 가지므로 그림 1-(b)와 같이 특정 싱크 노드가 자신과 연결된 모든 싱크 노드에게 패킷을 전송하고, 해당 패킷의 목적지 노드의 정보를 저장중인 싱크 노드만이 이를 수신하여 목적지까지 전달하는 방식을 제안한다.

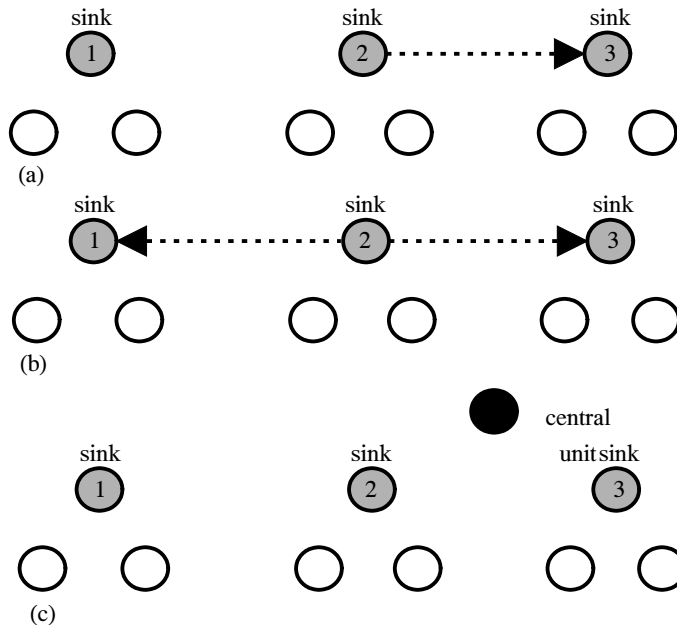


그림 1. 서로 다른 DADOG 간 통신 방식
 Fig. 1. Communication methods between difference DADOG

그러나 해당 패킷과 관계 없는 DADOG 의 싱크 노드에게도 패킷이 전송되어 IoT 네트워크의 주요 성능 저하의 원인인 병목 현상을 가중시킨다는 점을 고려하여 첫 번째 방식을 확장한 세 번째 방식을 제안한다.

세 번째 방식은 그림 1-(c)와 같이 모든 라우팅 테이블을 저장할 수 있는 고성능의 중앙 처리 유닛을 두어서 싱크 노드는 자신의 DADOG에 존재하지 않는 노드를 목적지로 하는 패킷이 전달되면 이를 중앙 처리 유닛에게 전송하고, 중앙 처리 유닛은 이를 자신의 라우팅 테이블을 참조하여 해당되는 DADOG의 싱크 노드에게 패킷을 전송한다. 그러나 이 세 가지 방식 모두 싱크 노드를 이어주는 별도의 네트워크의 존재를 가정할 뿐만 아니라, 보안에 민감하여 백본 등 다른 외부 네트워크를 경유하지 말 것을 요구되는 패킷의 전송을 지원할 수 없다는 한계점이 존재한다.

이와 함께 DADOG 간 패킷 전송 방식 외에도 세 번째 방식에서 사용된 중앙 처리 유닛을 활용한 싱크 노드 사이의 동기화 기법 역시 연구되었다. 제안된 기법에서는 각 싱크 노드가 서로가 아닌 중앙 처리 유닛에게 요청/응답 메시지를 전달하는 서버-클라이언트 모델을 제안함으로써 네트워크 탐색 시간 및 동기화 지연 시간을 줄이고 각 싱크 노드의 메모리 부하를 감소시킬 수 있다.

2.2 라우팅 오버헤드 감소 기법

라우팅 오버헤드 감소 기법은 RPL 보다는 AODV 등의 on-demand 라우팅 경로 설정 과정 중에 발생할 수 있는 과도한 RREQ 메시지를 대상으로 연구되고 있다. 그러나 RPL에서 DIO 메시지 역시 RREQ와 마찬가지로 네트워크에 플러딩 된다는 점에서 이러한 라우팅 오버헤드 감소 기법들이 RPL에 응용될 수 있다. 제안된 기법에서 네트워크 내의 센서 노드들은 다중 경로를 통해 다수의 라우팅 메시지(RREQ 메시지)를 수신하게 되며, 이 때 이웃 노드의 수와 함께 자신이 전송한 라우팅 메시지를 수신한 이웃 노드가 해당 메시지를 다시 브로드캐스트하는 수를 카운트 한다^[7,8]. 이후 새로운 라우팅 메시지가 수신될 경우 자신이 하나의 라우팅 메시지를 전송 했을 때 발생하는 이웃 노드의 라우팅 메시지의 전송량이 이웃 노드의 수에 비해 일정 기준치(threshold) 보다 적으면 네트워크가 혼잡하여 라우팅 메시지를 정상적으로 처리할 수 없는 상황으로 판단하여 해당 라우팅 메시지를 폐기한다. 또한 라우팅 메시지를 폐기하는 대신 해당 메시지를 일정 시간 대기한 후 재전송을 하는 기법과 함께 라우팅 경로를 최대한 신속하게 형성할 수 있는 최적

의 재전송 대기 시간을 계산하는 기법을 제안한다.

이 외에도 RPL 프로토콜이 적용된 네트워크에서 예비 부모 노드의 협력적 통신(Co-operative communication)을 활용하여 패킷 손실 시 발생하는 재전송 오버헤드를 줄일 수 있는 방법을 제시하고 있다^[9]. 기존의 RPL에서는 그림 2-(a)처럼 부모 노드로 패킷을 전송한 센서 노드가 일정 시간 안에 ACK 메시지를 수신하지 못할 경우 부모 노드가 손실되었다고 판단하여 예비 부모 노드에게 해당 패킷을 재전송한다. 이 과정에서 센서 노드가 동일한 데이터 패킷을 부모 노드와 예비 부모 노드에게 각각 전송하게 되는데, ^[1]에서는 그림 2-(b)처럼 첫 번째 패킷이 전송될 때 예비 부모 노드가 이 패킷을 미리 overhearing 하도록 하여 손실이 발생했을 경우 데이터 패킷 대신 재전송 요청 메시지를 예비 부모 노드에게 전송한다. 예비 부모 노드는 재전송 요청 메시지를 수신할 경우 즉시 overhearing 해 둔 메시지를 싱크 노드를 향해 전송함으로써 데이터 패킷의 재전송 딜레이 및 오버헤드를 줄일 수 있다.

III. 본 론

3.1 가정

앞에서 언급한 것처럼 다수의 소규모 IoT 네트워크가 함께 동작하여 대규모 네트워크를 구성할 경우 발생하는 라우팅 오버헤드 및 병목현상을 감소시키기 위해 다수의 싱크 노드가 설치되므로 멀티 싱크를 고려한 RPL 프로토콜이 필요하다. 이 때 새롭게 형성된 대규모 네트워크 내에서는 기존 소규모 IoT 네트워크의 싱크 노드 또는 다수의 싱크 노드 중 특정 싱크 노드에게만 전달되어야 하는 패킷(네트워크 제어 메시지 또는 보안성에 민감한 패킷)과 싱크 노드에게 전달된 후 백본 네트워크를 경유하여 외부 서버로 전달되어야 하는 패킷(일반적인 데이터 패킷)이 혼재하게 된다. 본 논문에서는 특정 싱크 노드에게만 전달되어야 하는 패킷을 싱크 지향적(sink-oriented) 패킷, 외부 네트워크로 전달되어야 하는 패킷을 싱크 독립적(sink-independent) 패킷이라 정의한다. 이 때 싱크 독립적 패킷은 네트워크상 존재하는 임의의 싱크에게 전달되어도 성공적으로 외부 네트워크에게 전달될 수 있다고 가정한다. 또한 응용에 따라 하나의 센서 노드가 두 개 이상의 소규모 네트워크에 동시에 포함되어 복수의 싱크 지향적 패킷을 생성할 수도 있으며, 싱크 독립적 패킷 전송이 싱크 지향적 패킷의 전송에 비해 더 자주 요구됨을 가정한다.

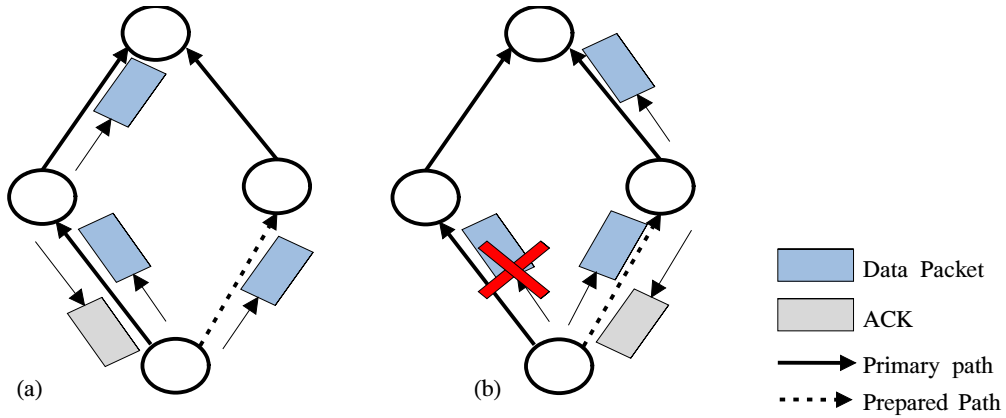


그림 2. 재전송 오버헤드 감소 기법
Fig. 2. Retransmission Overhead Reduction Scheme

이와 같은 멀티 싱크 환경에서 각 센서 노드는 여러 싱크 노드로부터 전송된 다수의 DIO 메시지를 수신할 수 있다. 이 때 각 센서 노드들은 수신된 DIO 메시지와 자신의 라우팅 테이블 정보를 비교함으로써 아직 경로가 형성되어 있지 않은 싱크 노드로부터 전송된 DIO 메시지인지를 구분한다. 만약 아직 경로가 형성되어 있지 않은 싱크 노드로부터 수신된 DIO 메시지일 경우 해당 경로 정보를 라우팅 테이블에 추가함으로써 싱크 지향적 패킷의 전송할 수 있는 경로를 형성한다.

만약 경로가 형성되어 있는 싱크 노드로부터 다른 경로로 전송된 DIO 메시지가 수신되었을 경우, 센서 노드는 해당 DIO 메시지의 라우팅 메트릭 값과 라우팅 테이블에 저장된 해당 싱크 노드로 향하는 경로의 라우팅 메트릭 값을 비교한다. 만약 새로 수신된 DIO 메시지에 기록된 라우팅 메트릭 값이 더 크다면 이는 이미 저장된 경로에 비해 비효율적인 경로를 통해 전달되었다는 의미이므로 해당 DIO 메시지를 전송한 노드를 예비 부모 노드로써 관리하되, 해당 DIO 메시지는 더 이상 전달되지 않고 폐기한다.

3.2 불필요한 경로 업데이트 문제

위에서 가정한 방식으로 동작하는 RPL이 구현된 경우 각 센서 노드는 자신이 저장하고 있는 전체 라우팅 테이블의 라우팅 메트릭 값에 따라 가장 효율적이라 판단되는 경로와 싱크 노드를 경유하여 외부 백본 네트워크로 패킷을 전달할 수 있다. 또한 자신이 소속되어 있는 네트워크의 싱크 노드로 향하는 경로를 지속적으로 유지함으로써 싱크 지향적인 패킷인 제어 메시지를 전달할 수 있다. 또한 라우팅 메트릭 값 비교를 통해 비효율적인 경로로 전송된 DIO 메시지를

폐기하여 라우팅 오버헤드를 줄일 수 있다.

이 밖에도 특정 싱크 노드로부터 전송된 DIO 메시지에 기록된 라우팅 메트릭 값이 라우팅 테이블에 기록되어 있는 라우팅 메트릭 값에 비해 좋을 경우 센서 노드는 다음과 같이 동작한다. 우선 새로 수신된 DIO 메시지가 더 좋은 경로를 통해 전송되었으므로 센서 노드는 해당 메시지를 전송한 노드를 새로운 부모 노드로 선정하고 이전 부모 노드는 예비 부모 노드로써 저장한다. 이후 업데이트된 DIO 메시지를 다시 자식 노드들에게 브로드캐스트한다.

이러한 업데이트된 DIO 메시지를 전송하는 과정을 경로 업데이트라 정의하며, 단일 싱크 기반 네트워크에서는 저장된 라우팅 테이블의 중 최소의 라우팅 메트릭 값(RM_{best}) 보다 낮은 라우팅 메트릭 값이 기록된 DIO 메시지가 수신될 때 마다 수행된다. 이러한 경로 업데이트 과정은 DIO 메시지의 전송 오버헤드를 발생시킬 수 있으나, 자식 노드들이 효율적으로 패킷을 전송할 수 있는 경로를 제공하는 것이 중요하므로, RM_{best} 보다 낮은 라우팅 메트릭 값을 가진 DIO 메시지가 수신될 때 수행된다.

하지만 가정한 멀티 싱크 환경에서는 싱크 독립적 패킷 외에도 싱크 지향적 패킷의 전송 역시 발생하므로, 수신된 DIO 메시지의 라우팅 메트릭 값은 현재 저장되어 있는 라우팅 테이블의 최소 라우팅 메트릭 값인 RM_{best} 뿐 아니라 이를 전송한 특정 싱크 노드로 향하는 경로들과 따로 비교되어야 한다. 다시 말하여, 수신된 DIO 메시지가 전송된 경로가 싱크 독립적인 패킷을 전달하는 현재 저장된 라우팅 테이블 상 최적의 경로가 아니더라도 이 메시지를 전송한 싱크 노드 i 를 목적지로 하는 싱크 지향적인 패킷을 전송하기에

는 최적의 경로일 수 있다. 이 경우에서 싱크 노드 i 를 목적지로 하는 싱크 지향적 패킷을 전송하기 위한 효율적인 경로를 자식 노드들에게 알려주기 위해 경로 업데이트를 수행해주어야 한다.

하지만 앞에서 언급한 것처럼 싱크 지향적 패킷은 싱크 독립적인 패킷에 비해 느린 속도로 생성되므로 이러한 경로 업데이트를 통해 생성되는 경로가 굉장히 드물게 사용되거나 심지어는 아예 사용되지 않을 수도 있다. 또한 자식 노드들이 이미 더 좋은 경로를 보유하고 있어서, 업데이트된 DIO 메시지가 수신되더라도 부모 노드를 변경하지 않을 수 있다. 뿐만 아니라 자식 노드들은 이미 전송된 DIO 메시지를 통해 해당 싱크 노드 i 를 목적지로 하는 경로를 하나 보유하고 있으므로, 굳이 추가적인 DIO 메시지 오버헤드를 감수하면서 경로 업데이트를 해주지 않더라도 싱크 지향적인 패킷은 비효율적인 경로를 경유해서나마 싱크 노드에게 전달될 수 있다. 따라서 이러한 상황에서 수행되는 업데이트된 DIO 메시지 전송은 네트워크의 대역폭을 소모할 뿐 아니라 자식 노드들이 DAO 메시지를 새로 생성하여 싱크 노드에게 전송하게 함으로써 싱크 노드 주변의 병목 현상을 발생시켜 throughput을 감소시키는 불필요한 경로 업데이트 문제를 발생시킨다.

이러한 문제를 해결하기 위해서는 해당 경로 업데이트가 유효한지를 판단하는 선택적인 경로 업데이트를 수행해야 한다. 본 논문에서는 경로 업데이트의 유효성을 판단하는 두 가지 방식을 제안한다.

3.3 라우팅 메트릭 기반 경로 업데이트 결정 (Routing Metric based Path Update Decision, RM-PUD) 기법

제안된 첫 번째 방식인 라우팅 메트릭 기반 경로 업데이트 결정(Routing Metric based Path Update Decision, RM-PUD) 방식에서는 그림 3 각 경로 정보의 라우팅 메트릭 값을 비교하여 경로 업데이트를 통해 자식 노드들이 새로운 경로를 형성할 가능성을 판단하여 그림 3과 같은 알고리즘으로 경로 업데이트 수행 여부를 결정한다.

RM-PUD 방식에서 센서 노드는 싱크 노드 i 로부터 전송된 새로운 DIO 메시지를 수신할 경우, 해당 메시지에 기록된 라우팅 메트릭 값인 $RM_{DIO(i)}$ 와 자신의 라우팅 테이블에서 싱크 노드 i 로 향하는 경로들 중 최소의 라우팅 메트릭 값인 $RM_{RTB(i)}$ 을 비교한다. 이 때 앞에서 가정한 것처럼 만약 $RM_{RTB(i)}$ 가 $RM_{DIO(i)}$ 보다 작다면 해당 DIO 메시지가 비효율적인 경로로 전

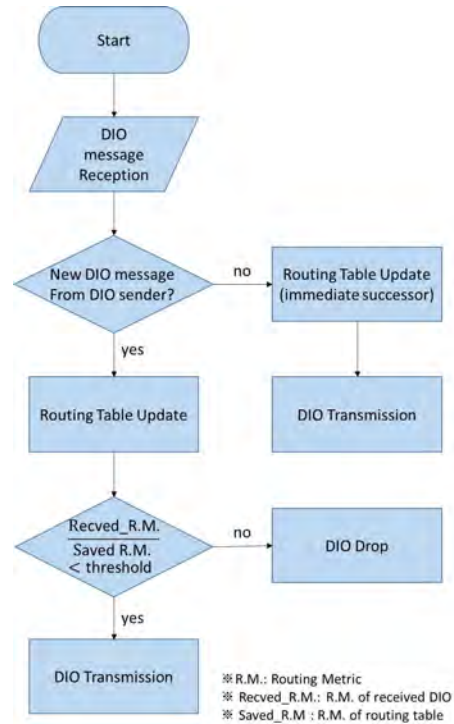


그림 3. RM-PUD 기법의 논리 흐름도
Fig. 3. Flow Diagram for RM-PUD

송되었다는 의미이므로 경로 업데이트를 하지 않고 DIO 메시지를 폐기한다. 반대로 $RM_{RTB(i)}$ 가 $RM_{DIO(i)}$ 보다 크다면 센서 노드는 수식 (1)을 활용하여 경로 업데이트를 결정한다.

$$\frac{DIO(i)}{RTB(i)} < Threshold \Rightarrow Routing Update \quad (1)$$

수식 (1)에 따르면 센서 노드는 싱크 노드 i 로 부터 전송된 DIO 메시지와 라우팅 테이블에 저장된 라우팅 메트릭 값의 비율이 일정 Threshold 이상 차이가 날 경우 해당 경로 업데이트를 통해 제공되는 새로운 경로 정보가 자식 노드들이 새로운 부모 노드를 선정할 수 있을 것이라 판단하여 경로 업데이트를 수행한다. 이 때 사용되는 Threshold 값은 아래 수식 (2)를 통해 계산된다.

$$Threshold = \frac{RM_{DIO(i)}}{RM_{RTB(best)}} \times \alpha \quad (2)$$

수식 (2)에서 $RM_{RTB(best)}$ 는 현재 저장되어 있는 라

우팅 테이블에서 싱크 노드에 관계없이 가장 라우팅 메트릭 값이 적은 경로의 라우팅 메트릭 값이다. α 는 네트워크 파라미터로 주어지며 두 라우팅 메트릭 값의 비율에 따른 보정값이다. 큰 α 가 주어질수록 경로 업데이트가 수행될 수 있는 라우팅 메트릭 값의 차이 역시 증가한다. 수식 (2) 에 따르면 새로 수신된 DIO 메시지의 라우팅 메트릭 값과 현재 저장된 가장 좋은 경로의 라우팅 메트릭 값의 차이가 작을수록 Threshold가 감소하여 경로 업데이트가 자주 수행된다. 이는 현재 저장된 라우팅 테이블 중 최소의 라우팅 메트릭 값을 가지는 경로가 손실되었을 경우 해당 라우팅 업데이트를 통해 제공되는 경로가 예비 부모 노드로써 동작하여 싱크 독립적인 패킷 전송에 활용될 수 있기 때문이다.

즉, 센서 노드는 새로 수신된 DIO 메시지가 전송된 경로가 이전에 전송된 DIO 메시지가 전송된 경로(라우팅 테이블에 저장된 경로) 에 비해 좋을수록 싱크 지향적인 패킷 전송에 활용될 가능성이 높으며, 싱크 노드에 관계없이 라우팅 테이블에 기록된 가장 좋은 경로에 비해 차이가 적을수록 향후 해당 경로가 손실되었을 때 싱크 독립적인 패킷 전송에 활용될 가능성이 높으므로 해당 라우팅 업데이트를 수행한다.

3.4 예비 부모 노드 기반 경로 업데이트 결정 (Immediate Successor based Path Update Decision, IS-PUD) 기법

두 번째로 제안한 기법인 예비 부모 노드 기반 경로 업데이트 결정(Immediate Successor based Path Update Decision, IS-PUD) 기법은 경로 업데이트 과정에서 전달되는 DIO 메시지가 경로 정보를 자식 노

드들에게 전달해 주는 기능 외에도 자식 노드들이 예비 부모 노드를 설정할 수 있는 기능을 제공한다는 점을 고려하여 경로 업데이트 유무를 결정하는 기법이다.

예를 들어, 그림 4와 같은 DIO 메시지 전송 시나리오에서 노드 A는 노드 B를 경유하는 경로 1을 통해 라우팅 메트릭 값(아래 예시에서는 ETX 값)이 4인 DIO 메시지를 수신하고, 이를 자식 노드인 노드 D에게 전송한다. 이후 노드 C를 경유하는 경로 2를 통해 라우팅 메트릭 값이 3.9인 DIO 메시지를 수신한다. 이 때, 노드 A는 노드 B를 새로운 부모 노드로 선정하고 자신보다 라우팅 메트릭 값이 높아진 노드 B를 부모 노드에서 삭제한다. 이후 노드 A는 경로 업데이트 수행 유무를 결정해야 하는데, 앞에서 설명한 RM-PUD 기법이 적용된다면 두 DIO 메시지의 라우팅 메트릭 값 차이가 크지 않아 경로 업데이트를 수행하지 않을 것이다.

하지만 이 경로 업데이트가 수행된다면 노드 B는 노드 A로부터 자신보다 라우팅 메트릭 값이 낮은 새로운 DIO 메시지를 수신하게 되어 노드 A를 예비 부모 노드로써 라우팅 테이블에 저장하게 된다(노드 A가 노드 B의 새로운 부모 노드로 설정될 수도 있다). 향후 노드 B의 부모 노드인 노드 E가 손실될 경우, 노드 B는 예비 부모 노드인 노드 A를 통해 싱크 노드로의 패킷 전송을 재개할 수 있다.

이와 같은 예비 부모 노드 선정 기능을 제공하기 위해서는 새로 수신된 DIO 메시지의 라우팅 메트릭 값이 자신의 이전 부모 노드의 라우팅 메트릭 값에 비해 더 좋을 경우 경로 업데이트를 수행해야 한다. 하지만 이러한 경로 업데이트 과정에서 전송되는 DIO 메시지는 노드 B뿐만 아니라 노드 D에게도 전달되며, 이로 인해 노드 D와 그 후손 노드들에서 불필요한 라우팅 업데이트가 수행될 수 있다. 뿐만 아니라 노드 B가 노드 A를 예비 부모 노드를 선정하더라도 동일한 조상 노드인 노드 F가 손실이 발생할 경우 예비 경로를 사용하더라도 싱크 노드에게 전달되지 못한다. 따라서 제안된 IS-PUD 기법에서는 그림 5와 같은 알고리즘에 따라 해당 경로 업데이트가 제공하는 예비 경로가 얼마나 경로 손실을 대처하기에 유용한 경로인지 판단하여 경로 업데이트 수행 유무를 결정한다.

DIO 메시지를 통해 새롭게 제공되는 경로가 기존 경로를 전달한 DIO 메시지와 유사한 경로를 통해 전송될수록 두 경로가 공유하는 조상 노드의 수가 증가한다. 두 경로가 공유하는 조상 노드에서 손실이 발생할 경우 예비 경로 역시 패킷을 전송할 수 없게 되므로, 두 DIO 메시지가 공유하는 조상 노드가 적을수록

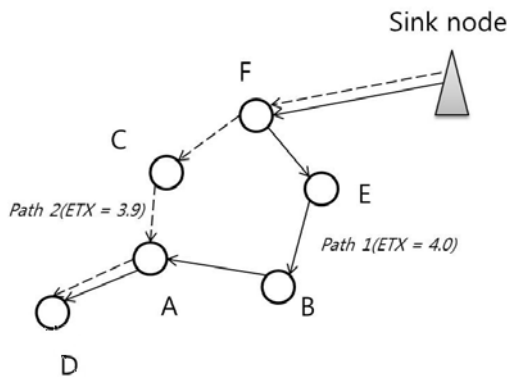


그림 4. 경로 업데이트를 통한 예비 부모 노드 설정 시나리오
Fig. 4. Path Update based Immediate Successor Decision Scenario

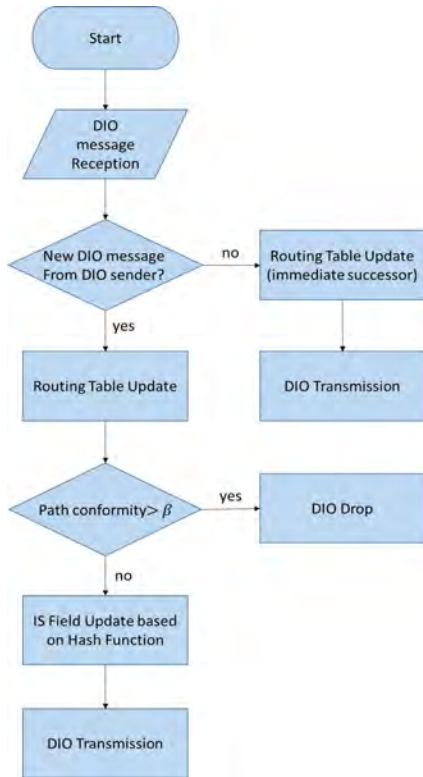


그림 5. IS-PUD 기법의 논리 흐름도
Fig. 5. Flow Diagram for IS-PUD

해당 경로 업데이트가 유효하다고 판단할 수 있다. 제안하는 IS-PUD 기법에서는 두 DIO 메시지가 공유하는 조상 노드의 수를 경로 일치도로 정의하며, 센서 노드는 수신된 두 DIO 메시지의 경로 일치도가 일정 이하일 경우에만 경로 업데이트를 수행한다.

두 DIO 메시지의 경로 일치도를 판단하기 위해서는 센서 노드가 두 DIO 메시지가 경유한 모든 중간 노드들을 파악해야 하지만, DIO 메시지가 경유한 모든 중간 노드들의 정보를 기록하는 것은 DIO 메시지의 길이를 지나치게 증가시키는 문제가 있다. 따라서 IS-PUD 기법에서는 DIO 메시지의 n 비트의 IS(Immediate Successor) 필드를 DIO 메시지 옵션 필드에 추가한다. 이후 DIO 메시지를 수신한 센서 노드는 자신의 주소를 입력으로 하여 0에서 n 사이의 결과를 가지는 임의의 해쉬 함수(Hash Function)를 실행한 결과 값과 동일한 번호의 IS 필드를 1로 설정한다. 이후 경로 일치도를 판단하고자 하는 센서 노드는 두 DIO 메시지의 IS 필드를 XOR 연산시킨 후 1의 결과를 가진 IS 필드의 수를 카운트 한 D_{HF} 를 계산한다. 센서 노드는 D_{HF} 값이 β 이상일 경우에만 경로 업데이트를

수행하며, 이 때 β 값은 네트워크 파라미터로 주어진다.

만약 두 DIO 메시지가 같은 조상 노드를 경유한 경우 두 DIO 메시지의 IS 필드의 값이 모두 1로 설정되므로 XOR 연산 시 0의 결과값을 가진다. 반대로 하나의 DIO 메시지만이 특정 조상 노드를 경유한 경우 해당 조상 노드를 경유한 DIO 메시지의 필드값은 IS 1, 해당 조상 노드를 경유하지 않은 DIO 메시지의 IS 필드값은 0이므로 XOR 연산한 D_{HF} 값을 1 증가시키게 된다.

IV. 성능 분석

4.1 실험 환경

본 실험에서는 NS-2 시뮬레이터를 사용하여 제안한 기법들의 성능평가를 수행하였다. 시뮬레이션을 위해 2100×2100 크기의 토폴로지를 700×700 크기로 9등분하고, 분할된 각 구역의 가운데에 하나의 싱크 노드를 설치하여 총 9개의 싱크 노드를 설치하였다. 전체 네트워크에 총 n 개의 센서 노드를 랜덤하게 설치하였으며 각 센서 노드의 최대 전송거리는 150m로 설정하였다. 싱크 노드를 제외한 센서 노드들 중 임의의 50개의 노드들이 총 100개의 패킷을 생성하여 전송되되, 1/20의 확률로 자신의 구역에 인접한 구역의 싱크 노드에게 전송되어야 하는 싱크 지향적 패킷을 생성하고 나머지의 경우에는 싱크 독립적 패킷을 전송한다. RM-PUD 기법의 α 값은 0.75, IS-PUD 기법의 β 값은 5로 각각 설정하였으며 16bit의 IS 필드를 DIO 메시지에 추가하였다. 또한 IS-PUD에 사용될 해쉬 함수로는 자신의 IP 주소를 16으로 나눈 나머지 값을 사용하도록 설정하였으며 무분별한 DIO 메시지 플러딩을 막기 위해 각 DIO 메시지에 10의 TTL 값을 삽입하였다.

실험에 사용한 MAC 프로토콜로는 IEEE 802.11을 사용하였다. 모든 RPL의 최대 라우팅 테이블 크기는 7로 설정하였으며 ETX를 라우팅 메트릭으로 사용하였다. 또한 3%의 랜덤 패킷 손실율을 설정하였으며 100개의 패킷을 전송한 후 임의의 50개의 노드가 손실되도록 설정하였다.

RM-PUD 기법을 적용한 RPL, IS-PUD 기법을 적용한 RPL, 두 기법을 모두 적용한 RPL과 아무 기법도 적용하지 않은 RPL을 대상으로 싱크 지향적 패킷 전송률, DIO 메시지 전송 오버헤드 및 싱크 지향적 패킷을 위한 평균 예비 경로의 수를 각각 비교하였다. 이 때 아무 기법도 적용하지 않은 RPL의 경우 DIO

메시지를 전송한 노드와 관계없이 가장 라우팅 메트릭 값이 좋은 경로들만을 최대 라우팅 테이블만큼 저장하고, 기존 라우팅 테이블에 저장된 라우팅 메트릭 값보다 적은 라우팅 메트릭 값을 가진 DIO 메시지가 수신될 때 마다 경로 업데이트를 수행한다. 또한 두 기법을 모두 적용한 RPL은 두 기법 모두 경로 업데이트를 수행하도록 결정될 때에만 경로 업데이트를 수행한다.

4.2 실험 결과

그림 6은 네트워크에 설치된 센서 노드의 수에 따라 변화하는 싱크 지향적 패킷 전송률을 비교한 그래프이다. 이 실험에서 아무런 기법도 적용하지 않은 RPL의 경우 싱크 노드에 관계없이 라우팅 테이블에 저장하므로, 인접한 구역의 싱크 노드로부터 전송된 DIO 메시지가 우연히 수신되어 해당 경로를 저장중인 상태에서만 싱크 지향적 패킷 전송이 가능하다. 따라서 다른 기법에 적용될 경우에 비해 매우 낮은 싱크 지향적 패킷 전송률을 보이며, 센서 노드의 수가 많아짐에 따라 자신이 소속된 구역의 싱크 노드로 전송되는 경로의 수가 증가하여 이웃 구역에 위치한 싱크 노드로 향하는 경로가 라우팅 테이블에서 삭제되므로 싱크 지향적 패킷 전송률이 감소한다.

또한 선택적 경로 업데이트 기법이 적용된 RPL의 경우 전체적으로 높은 비슷한 싱크 지향적 패킷 전송률을 보이며, IS-PUD 기법만 적용된 RPL이 상대적으로 높은 싱크 지향적 패킷 전송률을 가진다. 이는 IS-PUD 기법이 싱크 지향적 패킷 전송을 위한 예비 부모 노드를 설정하여 부모 노드 손실에 강인하기 때문이다. 이 외에도 센서 노드의 수가 200개 이상일 때 성능이 크게 증가하는데 이는 경로의 수가 증가함에 따라 충분한 예비 경로가 생성되므로 부모 노드의 손실이 발생하더라도 패킷 전송을 지속할 수 있기 때문이다.

그림 7은 네트워크에 설치된 센서 노드의 수에 따라 변화하는 DIO 메시지 전송 오버헤드를 비교한 그래프이다. 이 실험에서 아무런 기법도 적용하지 않은 RPL의 경우 라우팅 테이블에 저장된 최소 라우팅 메트릭 값보다 큰 DIO 메시지가 수신된다면 해당 메시지를 폐기하므로 하므로 낮은 DIO 메시지 전송 오버헤드를 보인다. 선택적 경로 업데이트 기법이 적용된 RPL은 싱크 지향적 패킷 전송을 위한 경로를 생성하기 위해 높은 DIO 메시지 전송 오버헤드를 보이며, 특히 IS-PUD 기법만이 적용될 경우 가장 높은 DIO 메시지 전송 오버헤드를 보이는데, 이는 경로 일치도

가 충분히 크다면 라우팅 메트릭 값의 차이가 없더라도 경로 업데이트를 수행하기 때문이다.

그림 8은 선택적 경로 업데이트 기법이 적용된 세 가지 RPL에서 경로 지향적인 패킷 전송을 위해 형성

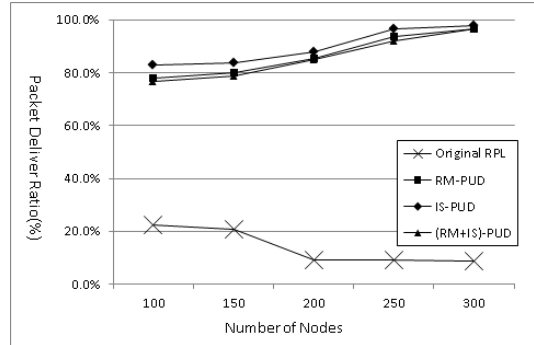


그림 6. 싱크 지향적 패킷 전송률
Fig. 6. Packet Deliver ratio of Sink Oriented Packets

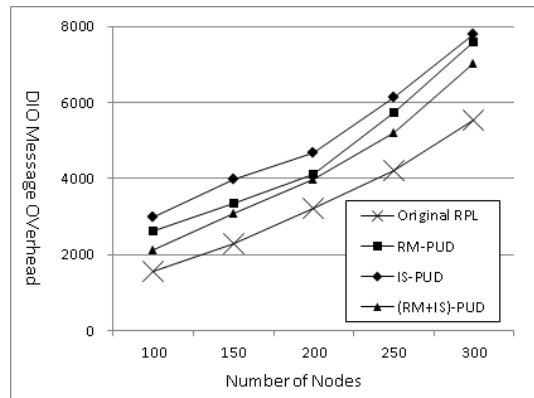


그림 7. DIO 메시지 오버헤드
Fig. 7. DIO Message Overhead

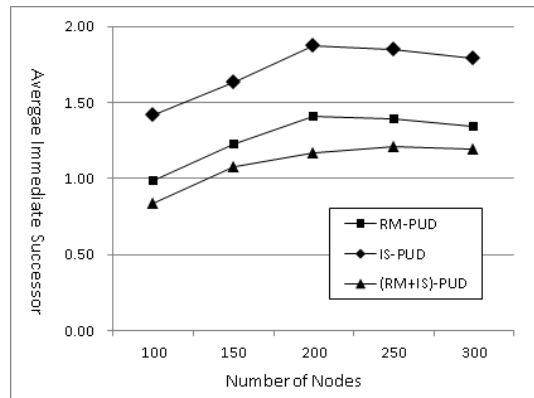


그림 8. 평균 예비 부모 노드
Fig. 8. Average Immediate Successor

된 경로의 평균 예비 경로의 수를 비교한 그래프이다. IS-PUD가 적용된 RPL 은 라우팅 메트릭 값에 관계 없이 경로 일치도를 기반으로 경로 업데이트를 수행하므로 RM-PUD만이 적용된 RPL 또는 두 기법이 모두 적용된 RPL 에 비해 많은 수의 예비 부모 노드를 설정함을 확인할 수 있다. 또한 노드의 수가 200개 이상일 경우에는 예비 부모 노드의 수가 오히려 조금 감소하는데, 이는 자신이 소속된 지역의 싱크 노드로 향하는 경로들이 라우팅 테이블에 저장되면서 싱크 지향적인 경로의 예비 경로를 저장할 라우팅 테이블이 부족해지기 때문이다.

V. 결 론

본 논문에서는 다수의 IoT 네트워크가 협력하여 대규모 IoT 네트워크를 형성했을 때 RPL 라우팅 프로토콜에서 센서 노드들이 불필요한 DIO 메시지를 전송할 때 발생하는 불필요한 경로 업데이트 문제를 정의하고 이를 해결하는 두 가지 기법을 제안하였다. 첫 번째 기법인 RM-PUD에서는 각 DIO 메시지의 라우팅 메트릭 값을 사용하여 경로 업데이트를 통해 제공될 새로운 경로가 지식 노드들에게 얼마나 유용한지를 측정하여 경로 업데이트 수행 유무를 결정한다. 두 번째 기법인 IS-PUD은 지식 노드들에게 전달되는 예비 경로 정보가 경로 손실을 대처해 줄 수 있는지를 나타내는 경로 일치도를 계산하여 경로 업데이트 수행 유무를 결정하는 기법이다. 실험을 통해 제안된 기법들이 싱크 지향적 패킷 전송을 위한 경로를 생성하지 않는 기존 RPL 에 비해 패킷 전송률을 최대 87% 향상시킬 수 있음을 확인하였다.

또한 IoT 네트워크의 규모가 거대해질수록 IS-PUD 기법에서 같은 해쉬 값을 가지는 노드의 빈도가 증가하므로 IS 필드의 길이 역시 증가하게 된다. 향후 증가된 IS 필드가 네트워크에 미치는 영향에 대한 분석과 함께 최적의 IS 필드 길이를 결정하는 새로운 경로 일치도 판단 기법에 대해 연구할 것이다.

References

[1] S. Yang, S. Lee, H. Rho, and W. Son, "Mobile sink based energy efficient path setup method for wireless sensor networks," *J. KICS*, vol. 39C, no. 11, pp. 1068-1077, Nov. 2014.

[2] S. Lee, H. Yoo, and D. Kim, "Residual

energy-aware duty-cycle scheduling scheme in energy harvesting wireless sensor networks," *J. KICS*, vol. 39B, no. 10, pp. 691-699, Oct. 2014.

[3] T. Winter, Ed., *RPL: IPv6 routing protocol for low power and lossy networks*, Retrieved Aug., 20, 2015, from <https://tools.ietf.org/id/draft-ietf-roll-rpl-19>

[4] H. Park and T. Kwon, "Military routing scheme using content-centric mobile Ad hoc networks," *J. KICS*, vol. 39, no. 4, pp. 334-342, Apr. 2014.

[5] Y. Xu, A. Saifullah, Y. Chen, C. Lu, and S. Bhattacharya, "Near optimal multi-application allocation in shared sensor networks," *MobiHoC 2010*, pp. 181-190, Chicago, USA, Sept. 2010.

[6] D. Carels, N. Derdaele, E. De Poorter, W. Vandenberghe, I. Moerman, and P. Demeester, "Support of multiple sinks a via virtual root for the RPL routing protocol," *EURASIP J. Wireless Commun. Netw.*, vol. 2014, no. 91, pp. 1-23, May 2014.

[7] A. R. Bhuyar and V. T. Gaikwad, "A review on reducing routing overhead in mobile ad hoc network using probabilistic rebroadcast mechanism," *Int. J. Comput. Sci. Inf. Technol.*, vol. 5, no. 1, pp. 390-393, Jun. 2014.

[8] R. R. Ravi and V. Jayanthi, "Energy efficient neighbor coverage protocol for reducing rebroadcast in MANET," *Procedia Comput. Sci.*, vol. 47, pp. 417-423, Apr. 2015.

[9] S. Gormus, F. Tosato, Z. Fan, Z. Bocus, and P. Kulkarni, "Opportunistic RPL for reliable AMI mesh networks," *Wireless Networks*, vol 20, no. 8, pp. 2147-2164, Oct. 2014.

이 성 원 (Sungwon Lee)



2010년 2월 : 경북대학교 컴퓨터공학과 졸업
2012년 2월 : 경북대학교 컴퓨터학부 석사
2012년 3월~현재 : 경북대학교 컴퓨터학부 박사과정
<관심분야> 무선 센서 네트워크, IoT, D2D

정 옹 환 (Yonghwan Jeong)



2014년 8월 : 영남대학교 컴퓨터공학과 졸업
2014년 9월~현재 : 경북대학교 컴퓨터학부 석사과정
<관심분야> IoT, D2D, 무선 센서 네트워크

강 현 우 (Hyunwoo Kang)



2005년 : 경북대학교 컴퓨터공학과 공학사
2009년 : 경북대학교 전자전기 컴퓨터학부 공학석사
2009년 2월~현재 : 한국전자통신연구원 대경권연구센터 선임연구원

<관심분야> 컴퓨터통신, 무선 센서 네트워크

김 동 균 (Dongkyun Kim)



1994년 2월 : 경북대학교 컴퓨터공학과 졸업
1996년 2월 : 서울대학교 컴퓨터공학과 석사
2001년 2월 : 서울대학교 컴퓨터공학과 박사
2009년 3월~현재 : 경북대학교 컴퓨터학부 정교수

<관심분야> 컴퓨터통신, 이동인터넷, 모바일 애드혹 네트워크, 센서 네트워크, 무선메쉬 네트워크, IoT

유 흥 석 (Hongseok Yoo)



2005년 7월 : 경북대학교 컴퓨터공학과 졸업
2008년 2월 : 경북대학교 전자전기 컴퓨터학부 석사
2014년 2월 : 경북대학교 컴퓨터학부 박사
2014년 3월~현재 : 경운대학교

IT에너지대학 모바일공학과 조교수

<관심분야> 근거리통신망, IoT, CPS, 지능형교통시스템, D2D