

논문 2010-05-20

무선메쉬 네트워크에서 우회정보를 이용한 다중경로 DSDV 라우팅기법

(Multi-Path DSDV Routing Protocol using Bypass Information in Wireless Mesh Network)

이도엽, 김중규*

(Do-Yup Lee, Jung-Gyu Kim)

Abstract : One of the most famous proactive routing protocols, DSDV (Destination -Sequenced Distance-Vector), is based on the minimum hop count. This paper suggests BIMP(Bypass Information Multi-Path) DSDV routing protocol that this protocol could route reliably as it can take the information of the bypass for the solution of the problem on EHMP DSDV.

In this paper, the protocol checks throughput, packet delivery delay, packet delivery ratio, number of management messages, packet transmit finish time, etc on the traffic compared to DSDV, HMP DSDV, and EHMP DSDV protocols.

Keywords : DSDV, EHMP, Wireless mesh network, Routing algorithm

I. 서론

무선 메쉬 네트워크에서 라우팅은 유비쿼터스 통신을 목표로 하는 4세대 이동통신 네트워크에서 중요한 요소 기술의 하나로 인식되고 있으며, 최근 활발한 연구 및 개발이 진행되어 오고 있다. 무선 메쉬 네트워크는 메쉬 라우터와 메쉬 단말로 구성 되는데, 메쉬 라우터는 메쉬 네트워크의 백본 역할을 수행하며 메쉬 단말간의 통신을 지원한다. 메쉬 라우터 중 일부는 메쉬 단말에게 인터넷 접속을 제공하기 위한 인터넷 게이트웨이 기능도 수행하게 되며, 이러한 메쉬 라우터는 게이트웨이 메쉬 라우터로 불린다.

메쉬 라우터는 이론적으로 이동성에 제약이 없으나 보통 이동성이 적은 상태에서 동작하는 것이 일반적이다. 메쉬 단말은 이동 애드 혹 프로토콜을 통해 다른 단말과 직접 통신을 하거나 메쉬 라우터를 통해 통신을 수행할 수 있다[11].

최근, 무선 메쉬네트워크에서 라우팅 프로토콜에 대한 연구가 활발히 진행되고 있으며, 무선메쉬 네트워크를 구현하기 위한 여러 요소 중 라우팅 기술은 데이터를 효과적으로 전달하기 위한 핵심기술로서 유사한 통신 특성을 가지는 이동 애드혹 네트워크의 라우팅 프로토콜을 적절히 변형하여 적용하는 방안이 논의되고 있다.

애드혹 네트워크에서 라우팅 기법을 분류하는 방법 중 하나로 데이터 전송을 위한 경로 설정 시점을 기준으로 경로가 미리 계산이 되어 있는 여부에 따라 proactive 기법과 reactive 기법으로 나눌 수 있다[6].

애드혹 네트워크에서 DSDV(Destination-Sequenced Destination-Vector)는 distance vector 방식의 대표적인 proactive 라우팅 프로토콜로서 라우팅 테이블을 교환함으로써 원하는 목적지로 가기 위한 거리 정보인 홉 수와 원하는 목적지로 가기 위한 다음 홉의 정보를 알게 된다. 그리고 네트워크의 정보가 변하기 때문에 라우팅 테이블을 갱신해야 하는데, 순서 번호가 큰 최근의 것을 우선으로 갱신하며, 순서 번호가 같을 때는 목적지까지 적은 홉 수를 가진 것으로 갱신해 준다. 라우팅 테이블을 주고 받을 때는 순서 번호를 이용해서 루프가 발생하는 것을 방지한다.

* 교신저자(Corresponding Author)

논문접수 : 2010. 06. 18., 수정일 : 2010. 08. 11., 채택확정 : 2010. 09. 02.

이도엽 : 드림소프트

김중규 : 대구대학교 정보통신공학부

메쉬 라우터는 이동성이 거의 없으므로 한 번 토폴로지가 형성되면 새로운 라우터가 진입하거나 떠나지 않는 한 그 토폴로지는 계속 유지된다. 또한, 무선 백본으로서 메쉬 라우터의 목적지는 네트워크 내의 모든 다른 라우터가 될 수 있으며, 데이터의 흐름 또한 많다. 그렇기 때문에 메쉬 라우터사이에 전체 네트워크의 토폴로지의 정보를 가지는 proactive 기법이 이점을 가진다.

그러나 무선메쉬 네트워크에 DSDV를 그대로 사용하기는 어려운데, 이는 DSDV가 하나의 다음 홉 정보만 가지고 있어 여분의 경로를 얻기 어렵고, 사용 경로에 문제가 발생하였을 때 복구하는 것에 많은 시간이 걸려 데이터의 손실이 크게 발생할 수 있기 때문이다. 또한 주기적인 라우팅 테이블 갱신은 부하가 클 뿐만 아니라 네트워크 변화를 빠르게 따라가기 어렵기 때문에 QoS를 만족시키기 위한 실시간 정보를 라우팅에 이용하기도 어렵다[1].

이를 해결하기 위한 방법으로 홉 간 다중경로와 잉여대역폭을 이용한 HMP(Hop-By-Hop) DSDV 방식과 최소 홉 수에서 N홉 이내의 홉 수를 가지는 모든 경로를 고려하는 방식인 EHMP(Enhance HMP) DSDV 라우팅 기법을 제안하였다. 그러나 HMP DSDV 라우팅 기법은 최소 홉 수를 가지는 복수의 라우팅 경로 정보만을 포워딩 테이블에 저장하여 이용함으로써 트래픽이 증가하는 경우 효과적으로 동작하지 않을 수 있고 EHMP DSDV는 우회한 이후의 잉여 대역폭을 알 수 없기 때문에 우회한 이후의 잉여 대역폭이 좋지 않을 수 있다.

본 논문에서는 EHMP DSDV의 우회 경로 이후 잉여 대역폭이 좋지 않을 경우 나타나는 단점을 개선하기 위해, 우회 경로 다음 홉 간의 경로 정보를 획득하고 이를 이용하여 신뢰성 있는 우회 정보를 가지는 방법을 제시하고 시뮬레이션을 통하여 그 성능을 비교 분석한다.

II. 관련연구

1. DSDV

DSDV(Destination Sequenced Distance Vector)는 기존 DV(Distance-Vector)방식의 라우팅 알고리즘에서 무한 카운트 문제를 근본적으로 해결한 라우팅 방식이다. 그 이전에도 이 문제를 해결하기 위한 여러 가지 방안들이 제시되었지만, DSDV 이전까지는 이러한 문제를 완벽하게 해결하

지 못하였다. 그러나 DSDV는 단순한 방식으로 이 문제를 해결하였고, DV방식의 라우팅 알고리즘에 드혹 네트워크에 적용할 수 있는 길을 최초로 마련했다는 점에서 의의를 가진다. DSDV에서 사용하는 라우팅 테이블은 기존 DV방식의 라우팅 테이블에 목적지, 다음 홉, 거리 외에 순서 번호, 설정 시간, 안정 데이터 항목이 추가된 것이다[5].

라우팅 정보는 목적지, 거리 그리고 순서 번호 항목만 전파한다. 추가된 항목 중 가장 중요한 것은 순서 번호이다 순서 번호는 보통의 경우 목적지 노드가 생성하며, 새로운 라우팅 정보를 전파할 때마다 1씩 증가한 다 즉, 각 노드는 현재 자신이 마지막으로 전파한 라우팅 정보의 순서번호를 기억하고 있다가, 다음 정보를 보낼 때는 이보다 1증가 한 번호로 전파하는 것이다 이정보는 각 노드가 가진 라우팅 테이블의 해당 엔트리에 기록되며, 다른 노드에 전파할 때에도 그대로 유지된다.

새로운 라우팅 정보 패킷을 받은 노드는 새로 받은 라우팅 정보와 자신의 테이블에 있는 정보를 엔트리별로 비교하면서, 새로 받은 패킷에 있는 순서 번호가 자신의 테이블에 있는 것보다 큰 경우나, 같은 순서 번호를 가지면서 더 가까운 거리를 나타내는 경우에만 엔트리를 갱신하고, 그렇지 않은 엔트리는 그대로 유지한다. 깨어진 링크는 거리가 ∞ 인 것으로 표시되고 링크가 깨어졌다는 것을 발견한 노드는 새로운 순서번호로 이를 네트워크 전체에 전파한다.

즉, 노드A와 B사이의 링크가 없어졌다는 것을 노드B가 발견했다면, 노드B는 자신이 가지고 있는 A에 대한 순서 번호보다 1 증가된 순서 번호로 자신과 A의 거리가 t 라는 사실을 전파하는 것이다. 이를 받은 노드는 자신의 라우팅 테이블에서 목적지가 A이고 다음 노드 항목이 B인 엔트리가 있는지 보고, 있으면 자신과 A 사이의 거리를 ∞ 로 기록한 후 다시 이를 전파한다. 이러한 과정이 반복됨으로써 노드B를 통해 A에 도달하는 노드들은 모두 자신과 A 사이의 거리를 ∞ 로 표시하게 된다. 만일 A가 네트워크에서 완전히 분리된 게 아니라면, A와 연결된 다른 노드를 통해 목적지 A에 관한 새로운 라우팅 정보가 다시 네트워크에 퍼지게 되므로 표시된 정보는 다시 유한한 새로운 값으로 바뀌게 된다. 이렇게 링크가 깨어짐으로 해서 어떤 노드와의 거리가 ∞ 가 되었음을 알리는 경우가, 목적지노드가 아닌 다른 노드에 의해 순서 번호가 매겨지는 유일한 경우이다.

설정 시간은 해당 엔트리가 언제 갱신되었는지

가 기록되는 항목이다. 일정 시간 동안엔 트리가 갱신되지 않으면 그 엔트리는 사라지게 되는데, 이는 노드가 재부팅되어 순서 번호가 리셋 되거나, 전송 오류로 인해 잘못된 순서 번호가 기록되었을 때의 오동작을 최소화하기 위한 것이다. 예를 들어 노드 A가 62433까지의 순서 번호를 전송하고 나서 재부팅(순간적인 정진이나, 고장 등의 이유로)된 경우를 생각해 보면, 재부팅 후 A는 자신이 몇 번까지의 순서 번호를 전송했는지 알 수 없으므로, 0부터 새로운 순서 번호를 시작하려고 할 것이다. 하지만, 이미 다른 노드들은 A로부터 62433까지의 정보를 받았으므로 이보다 작은 순서 번호를 무시할 것이기 때문에, A의 순서 번호가 62453보다 커질 때까지 A에 관한 정보는 계속해서 무시되게 된다. 일정 시간이 지나도록 갱신되지 않는 엔트리는 사라지도록 함으로써, 이러한 현상을 막을 수 있다[2].

2. HMP(Hop by hop Multi-Path) DSDV

HMP 라우팅 프로토콜 두 개의 라우팅 테이블을 사용하는데, 하나는 목적지까지 같은 홉 수를 가진 여러 개의 다음 홉 정보를 가지는 포워딩 테이블(Forwarding Table)이고, 다른 하나는 이웃 메쉬 라우터들 간에 잉여 대역폭 정보를 저장하고 네트워크의 토폴로지 변화 및 경로상의 문제를 감지하기 위한 네이버 테이블(Neighbor Table)이다[11].

목적지까지 같은 홉 수를 가진 여러 개의 다음 홉 정보를 알기 위해서 라우팅 테이블을 주고받을 때, 같은 순서번호를 가지더라도 다른 메쉬 라우터로부터 받은 라우팅 테이블이라면 그 정보를 받아 들인다. 그리고 포워딩 테이블 entry에 목적지는 같으나 다음 홉이 다르면 목적지까지 같은 홉 수를 가지면 새로운 entry로 보고 추가해 준다.

목적지와 다음 홉, 홉 수가 모두 같으면 순서번호가 큰 순으로 갱신한다. 그러나 목적지와 다음 홉이 같지만 홉 수가 더 작다면, 목적지와 다음 홉이 같은 다른 모든 entry를 삭제한 후 추가한다. 이 방식을 통해 추가적인 오버헤드 없이 목적지까지 최소 홉 수를 유지하는 여러 개의 다음 홉으로 포워딩 테이블을 구성할 수 있다.

메쉬 라우터는 이동성이 거의 없기 때문에 한번 형성된 토폴로지가 거의 변하지 않으므로 DSDV에서 수행하고 있는 주기적인 라우팅 테이블 갱신이 불필요해진다. 따라서 HMP DSDV 기법에서는 라우팅 테이블의 갱신을 새로운 메쉬 라우터가 참여하거나 떠나게 되는 즉, 토폴로지의 변화가 생긴 경우에만 수행하게 된다.

이웃 메쉬 라우터 사이에 주기적으로 대역폭 정보를 주고받아 포워딩 테이블과 별도로 네이버 테이블을 만들어 이용하는 방식을 제안한다. 네이버 테이블은 포워딩 테이블과는 다르게 네트워크 전체의 정보를 유지하고 있는 것이 아니라, 자신과 한 홉 관계에 있는 이웃 메쉬 라우터 사이의 정보만을 유지한다. 네이버 테이블의 정보는 목적지까지 걸리는 홉 수가 같은 여러 다음 홉 중에 잉여대역폭이 큰 메쉬 라우터를 다음 홉으로 정하여 데이터를 보내는 데에 이용된다. 잉여 대역폭은 원래의 링크 용량에서 그 링크를 거쳐 간 대역폭을 빼서 계산한다. 잉여 대역폭이 크다는 것은 데이터를 처리 할 수 있는 능력이 더 있다고 보았으며, 목적지까지 최소 홉 수를 유지하는 다음 홉 중에서 잉여 대역폭이 가장 큰 것을 다음 홉으로 선택함으로써 재전송을 줄이는 효과와 MAC 계층에서 경쟁을 줄이는 효과 및 bottleneck 라우터의 발생을 억제하는 효과를 얻는다[3].

3. EHMP(Enhanced HMP) DSDV

HMP DSDV 프로토콜은 트래픽이 증가할 경우 최소 홉 수를 가지는 라우팅 경로로 트래픽이 집중되는 현상으로 인하여 최소 홉 수를 가지는 복수의 라우팅 경로 정보만을 이용할 경우 효율적으로 동작하지 않을 수 있다.

EHMP DSDV 라우팅 프로토콜에서는 기존의 HMP DSDV 라우팅 프로토콜이 가지는 단점을 보완하기 위하여, 라우터간의 최소 홉 수를 가지는 경로 이외에도 최소 홉 수에서 임계치 값만큼의 홉 수를 더한 값 이하의 홉 수를 가지는 모든 라우팅 테이블을 관리하여 다중경로와 잉여대역폭 정보를 이용하여 목적지까지의 라우팅 경로를 선택하는 라우팅 방식을 제안하였다. EHMP DSDV 프로토콜에서는 메쉬 라우터가 이웃 메쉬 라우터와 라우팅 정보를 교환하는 방식 및 포워딩 테이블 관리 방식은 HMP DSDV와 유사하나 최소 홉 수에 해당하는 라우팅 경로뿐만 아니라 최소 홉 수에 임계치 N 홉을 더한 값 이하의 홉 수를 가지는 라우팅 경로에 대해서 그 경로 정보를 저장하여 준다는 점에서 차이가 있다. EHMP DSDV 라우팅 프로토콜은 트래픽이 증가하였을 때 홉 수를 증가시키더라도 트래픽이 집중되지 않는 경로로 데이터를 분산하여 특정 경로로 트래픽이 집중되는 현상을 줄일 수 있으며 전체적인 네트워크 성능 관점에서 더 효율적일 수 있다. EHMP DSDV 프로토콜은 DSDV 라우팅 프로토콜이나 HMP DSDV 라우팅 프로토콜에 비하

여 더 많은 정보와 처리량을 필요로 하는 단점을 가지지만 mesh 라우터는 충분한 전원공급이 가능하며 충분한 연산능력을 가지기 때문에 큰 문제점을 가지지 않는다. EHMP DSDV 라우팅 프로토콜에서는 그림 1과같이 발신라우터와 착신라우터간의 라우팅 경로를 설정할 때 발신라우터에서 최소 홉 수를 가지는 12번으로 가는 경로나 8번으로 가는 경로를 선택하지 않고 대역폭이 높은 2번 경로를 선택하여 트래픽이 집중되는 곳을 우회하는 것을 알 수 있다.

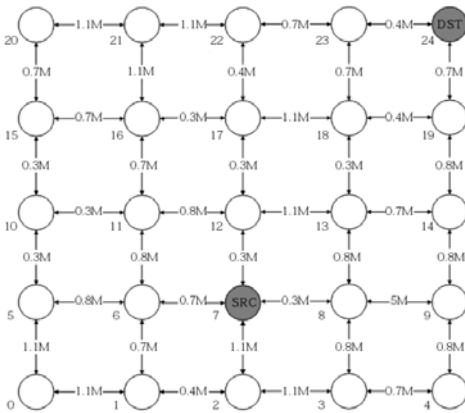


그림 1. Mesh 네트워크 토폴로지
Fig. 1. Mesh network topology

III. 제안한 기법 (BIMP DSDV)

1. BIMP DSDV

본 연구에서 제안한 기법인 BIMP (Bypass Information Multi-Path) DSDV 라우팅 기법은 기존의 DSDV 기법을 바탕으로 하여 EHMP DSDV 기법의 단점인 우회 한 이후의 잉여 대역폭을 알지 못하는 것을 한 홉 우회, 두 홉 우회, 최소 홉 수를 가지는 두 경로와 같은 세 가지 방식을 통하여 그 성능을 향상시켰다. EHMP DSDV 라우팅 기법을 기반으로 하기 때문에 다중경로를 가지고 포워딩 테이블뿐만 아니라 잉여 대역폭 정보를 가지는 네이버 테이블을 가지고 있다. 그리고 BIMP DSDV 라우팅 기법을 적용하기 위해 기존의 잉여 대역폭 정보를 지니는 네이버 테이블에 우회 홉 수를 알 수 있는 Bypass 항목을 추가한다.

제안한 BIMP DSDV는 EHMP DSDV의 우회 경

로 이후 잉여 대역폭이 좋지 않을 경우 나타나는 단점을 개선하기 위해, 우회 경로 다음 홉 간의 경로 정보를 획득하고 이를 이용하여 신뢰성 있는 우회 정보를 가지도록 보완하였다. 신뢰성 있는 우회 정보를 가지는 라우팅 기법으로 사용되는 방법에는 크게 3가지가 있다.

첫 번째 그림2와 같이 한 홉을 우회 하는 경우이다. 무선 mesh라우터 A에서 E로 데이터를 보내기 위해 경로 설정을 할 때 라우터 B는 라우터 D를 통하는 최소 홉 수를 가지는 경로와 라우터 C를 통하여 한 홉 우회 하는 총 두 가지 경로를 가지는 경우이다.

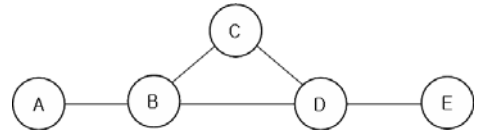


그림 2. 한 홉 우회
Fig. 2. 1-Hop detour

두 번째는 그림 3과 같이 두 홉을 우회 하는 경우이다. 무선 mesh라우터 A에서 F로 데이터를 보내기 위해 경로 설정을 할 때 라우터 B는 라우터 D를 통하는 최소 홉 수를 가지는 경로와 라우터 C, E를 통하여 두 홉 우회 하는 총 두 가지 경로를 가진다.

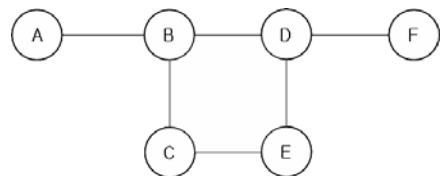


그림 3. 두 홉 우회
Fig. 3. 2-Hop detour

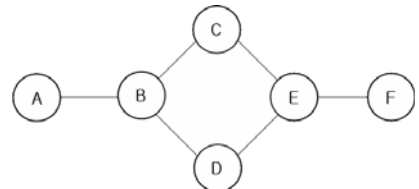


그림 4. 최소 경로를 가지는 두 경로
Fig. 4. Two routes with shortest path

세 번째는 그림 4와 같이 최소 홉 수를 가지는

두 가지 경로가 존재하는 경우이다. 이것은 그림 3과 유사 하지만 우회 경로를 가지지 않고 최소 경로이면서 그림 3의 방식을 응용하여 효율적인 라우팅 경로를 설정 할 수 있다.

이제 우회정보를 이용하여 한 홉 우회, 두 홉 우회, 최소 경로를 가지는 두 경로 방법에 대해 각각 자세히 설명한다. 아래 그림 5, 그림9, 그림10은 시작 라우터에서 목적지 라우터까지 최대 우회 홉 수는 두 홉이다.

2. 우회 정보를 이용한 한 홉 우회

BIMP DSDV 라우팅 기법에서 우회 정보를 이용한 한 홉 우회 경로 설정은 그림 5에서 라우터 A에서 라우터 E로 데이터를 보내기 위해 경로 설정을 할 때 라우터 B가 신뢰성 있는 우회 정보를 획득하기 위해서는 C-D 구간의 잉여 대역폭 정보를 얻을 수 있어야 한다.

C-D 구간의 잉여 대역폭 정보를 얻기 위해서는 라우터 D의 경우 B-D, C-D, D-E 구간 즉 라우터 D가 가지는 전체 잉여 대역폭 정보를 라우터 B에게 제공한다. 라우터 C의 경우도 라우터 C가 가지고 있는 전체 잉여 대역폭 정보를 B에게 제공한다.

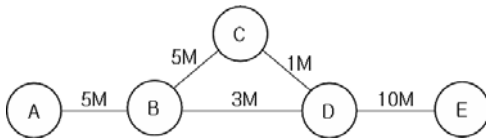


그림 5. 한 홉 우회
Fig. 5. 1-Hop detour

라우터 B는 라우터 C, D에게 받은 잉여 대역폭 정보를 가지고 네이버 테이블에 추가 하거나 갱신하게 되는데 포워딩 테이블과 동일한 엔트리가 존재할 경우 네이버테이블의 잉여 대역폭 정보를 갱신한다. 동일한 엔트리가 존재하지 않을 경우에는 라우터 C, D의 각각의 엔트리들을 비교하여 중복되는 경로가 존재 할 때에만 네이버 테이블에 우회 정보를 추가 하게 된다. 표1과 표2는 라우터 B의 포워딩 테이블과 네이버 테이블로, 네이버 테이블에는 Bypass 항목에 우회 경로가 아니면 0으로 설정하고, 한 홉 우회 할 경우 1, 두 홉 우회 하는 경우 2로 설정하게 된다. 라우터 D의 경우에도 같은 방법으로 B-C 구간의 잉여 대역폭 정보를 저장한다.

라우팅 테이블을 이용하여 데이터를 보낼 때는 라우터 B의 포워딩 테이블을 참조한다. 라우터 B는

라우터 E로 보내기 위해서 라우터 D, C 두 가지 경로를 선택할 수 있는데 라우터 D로 보내는 경우 홉 카운트는 2이고 C로 보내는 경우 홉 카운트는 3이다. 이 경우 D는 최소 홉 카운트 2를 가지고 있기 때문에 라우터 B테이블의 Bypass 항목의 0의 엔트리 중 D를 선택하고 C를 통해서 보내는 경우 한 홉 우회하기 때문에 Bypass 항목의 1의 엔트리 중 C-D구간의 잉여 대역폭과 C구간 중에서 대역폭이 낮은 C-D 구간을 선택한다. 선택된 D와 C-D 구간 중 잉여 대역폭이 높은 경로로 데이터를 넘겨주게 된다. 만약 C구간과 D구간의 대역폭이 같을 경우 낮은 홉 수를 가지는 경로를 선택한다.

표 1. 라우터 B의 포워딩 테이블
Table 1. Forwarding table of router B

Dest	Next	HopCnt
A	A	1
C	C	1
C	D	2
D	D	1
D	C	2
E	D	2
E	C	3

표 2. 라우터 B의 네이버 테이블
Table 2. Neighbor table of router B

Next	R_Band	Bypass
A	5	0
C	5	0
D	3	0
C-D	1	1

그림 6은 포워딩 테이블을 갱신하기 위한 방법인 포워딩 테이블 갱신 알고리즘을 나타낸 것이고,

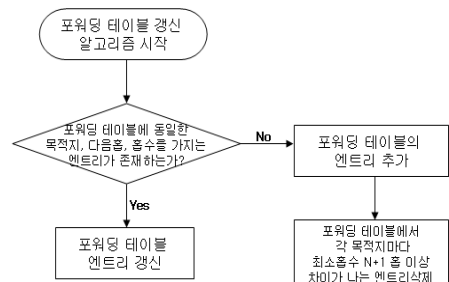


그림 6. 포워딩 테이블 갱신 알고리즘
Fig. 6. Updating algorithm for forwarding table

그림 7은 네이버 테이블을 갱신하기 위해 사용하는 알고리즘을, 그림 8은 경로선택 알고리즘을 나타낸 것이다.

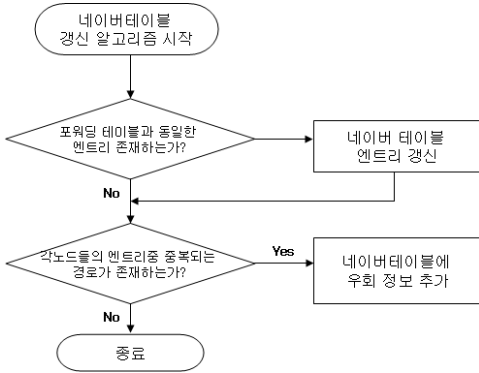


그림 7. 네이버 테이블 갱신 알고리즘
Fig. 7. Updating algorithm for neighbor table

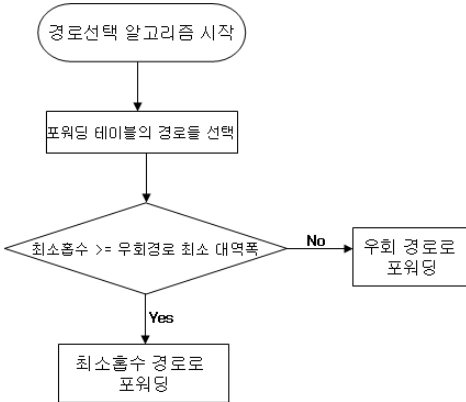


그림 8. 경로 선택 알고리즘
Fig. 8. Routing algorithm

3. 우회 정보를 이용한 두 홉 우회

BIMP DSDV 라우팅 기법에서 그림 9와 같은 경우, 라우터 A에서 F로 데이터를 보낼 때 A에서 포워딩 된 데이터를 라우터 F로 보내기 위해서는 라우터 D로 포워딩 하는 경우와 라우터 C로 포워딩 하여 두 홉만큼 우회하는 경우가 있다. 라우터 B에서 신뢰 성 있는 우회 정보를 가지기 위해서는 라우터 B에서 C-E 구간과 D-E 구간의 대역폭을 알아야 한다.

두 구간의 잉여 대역폭을 알기 위해서는 포워딩 테이블에서 하나의 목적지로 가는 구간 중 홉 카운트의 수가 2가 되는 두 개의 경로가 발생할 경우

이를 이용하여 우회 경로 정보를 얻을 수 있는데 라우터 B에서는 라우터 E로 가기 위한 엔트리의 홉 카운트가 둘 다 2가 된다.

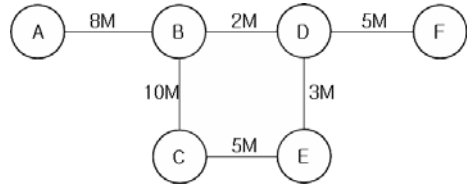


그림 9. 두 홉 우회
Fig. 9. 2-Hop detour

표 3. 라우터 B의 포워딩 테이블
Table 3. Forwarding table of router B

Dest	Next	HopCnt
A	A	1
C	C	1
C	D	3
D	D	1
D	C	3
E	D	2
E	C	2
F	D	2
F	C	4

표 4. 라우터 B의 네이버 테이블
Table 4. Neighbor table of router B

Next	R_Band	Bypass
A	8	0
C	10	0
D	2	0
C-E	5	1
D-E	2	1
C-E-D	3	2
D-E-C	2	2

라우터 E의 경우에도 B로 가는 경로의 엔트리가 둘 다 2가 된다. 앞서 설명한 한 홉 우회 하는 경우 서로 중복되는 하나의 경로 정보를 얻었지만 두 홉 우회 하는 경우 라우터 E가 가지는 C-E, D-E 구간의 경로 정보를 라우터 B는 얻어야 한다. 따라서 두 홉 우회 하는 경우에는 목적지 경로가 2가 되는 복수의 엔트리를 가지는 라우터에게 잉여 대역폭 정보를 보내게 한다. 즉, 라우터 E는 라우터 B로 가는 포워딩 테이블에서 홉 카운트가 2가 되는

두 개의 경로를 가지고 있기 때문에 라우터 E의 모든 구간의 잉여 대역폭 정보를 보내게 한다.

라우터 E로부터 잉여 대역폭 정보를 수신한 라우터 B는 네이버 테이블의 C-E 구간과 D-E 구간을 추가하고 C, C-E, E-D 구간과 D, D-E, E-C 구간에서 각각 잉여 대역폭이 가장 낮은 값을 각각 추가한다. 이 두 구간은 C-E-D 구간과 D-E-C 구간으로 두 홉을 우회 하는 구간 중 가장 낮은 잉여 대역폭을 기록 하고 Bypass 항목에 두 홉 우회를 의미하는 2로 설정을 한다. 라우팅 테이블을 이용하여 라우터 B를 통하여 라우터 F로 데이터를 포워딩 할 때 라우터 B의 라우팅 테이블에서 라우터 D와 C를 통하는 경우 두 가지가 있다. C로 포워딩 하는 경우 최소 홉 수이고 D로 포워딩 하는 경우는 두 홉 우회 하여 보내는 경우이다.

포워딩 테이블에서 두 개의 엔트리를 이용하여 네이버 테이블의 잉여 대역폭을 참조한다. 이때 목적지 F가 가지는 최소 홉 수만큼 HopCnt 값을 빼주면 D는 0이 되고 C는 2가 된다. 여기서 0과 2는 라우터 B테이블에 Bypass 항목과 비교하여 같은 값을 지니고 있는 항목과 비교하면 된다. 즉 Bypass 0인 A, C, D 중 하나를 선택하고 Bypass 값이 2인 C-E-D 와 D-E-C 중 하나를 선택한다. 최소 홉 카운트를 가지는 D와 우회 정보를 가지는 C-E-D 구간 중 잉여 대역폭이 높은 C-E-D 구간으로 즉 C로 데이터를 포워딩 하면 된다.

4. 최소 홉 수를 가지는 두 경로

최소 홉 수를 가지는 두 가지 경로에서는 우회 정보를 이용한 두 홉 우회하는 방법으로 응용할 수 있으며 네이버 테이블 갱신 방법도 이전 방식과 일치한다. 라우터 B는 라우터 E로부터 C-E, D-E 구간의 잉여 대역폭 정보를 받아서 네이버 테이블에 추가한다.

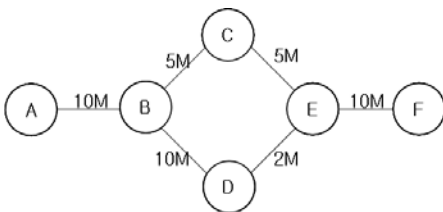


그림 10. 최소 홉 수를 가지는 두 경로
Fig. 10. Two routes with minimum hop

그림 10의 경우 라우터 B에서 라우터 F로 데이터를 포워딩하기 위해서 포워딩 테이블의 목적지 F

를 참조 하게 되는데 목적지 F의 다음 홉은 C, D이고, 홉 카운트는 둘 다 3이다. 여기서 목적지 E의 다음 홉은 F와 같은 C, D이고 홉 카운트는 둘 다 2홉이다. 따라서 우회 정보를 이용한 두 홉 우회 방식에서 E는 두 가지 경로를 가지는 경우이기 때문에 최소 홉 수를 가지는 두 가지 경로로 인지하고 네이버 테이블의 Bypass 항목이 1인 엔트리와 비교하여 잉여 대역폭이 높은 경로로 포워딩하게 된다. 만약 C-E 구간과 D-E 구간의 대역폭이 같을 경우 C구간과 D구간 중 대역폭이 높은 경로를 비교하여 포워딩한다.

표 5. 라우터 B의 포워딩 테이블
Table 5. Forwarding table of router B

Dest	Next	HopCnt
A	A	1
C	C	1
C	D	3
D	D	1
D	C	3
E	C	2
E	D	2
F	C	3
F	D	3

표 6. 라우터 B의 네이버 테이블
Table 6. Neighbor table of router B

Next	R_Band	Bypass
A	10	0
C	5	0
D	10	0
C-E	5	1
D-E	2	1
C-E-D	2	2
D-E-C	2	2

IV. 시뮬레이션 및 성능 분석

제안한 BIMP DSDV 라우팅 기법의 성능 평가를 위해 NS-2 시뮬레이터를 이용하여, 제안한 BIMP DSDV 라우팅 기법과 기존의 DSDV, HMP DSDV, EHMP DSDV 라우팅 기법과 성능을 비교 평가하고, 제안한 방법의 성능 우수성을 입증하였다 [12].

1. 실험 환경

본 연구에서 사용된 시뮬레이션 변수는 표 7과 같다. 메쉬 라우터의 수는 그림 11, 그림 12와 같은 5x5, 7x7 메쉬 토폴로지를 가정했는데 이는 최소 홉 수가 6홉과 9홉이기 때문에 최소 홉 수에 따른 성능 차이가 있는지 확인하기 위함이다.

표 7. 시뮬레이션 파라미터
Table 7. Simulation parameters

매개변수	값
Number of mesh routers	25, 49
MAC	IEEE 802.11b
Traffic type	CBR(constant bit rate)
Packet type	UDP, TCP
Packet size	512byte
Packet rate	200packets/s
Total bandwidth	11Mbyte
Hello message period	2s
Simulation time	1000s

패킷사이즈는 512 byte이며, 200 packets/s로 패킷 생성률을 설정하였고, 시뮬레이션 시간동안 일정하게 트래픽을 생성하도록 하였다. 패킷사이즈와 패킷 생성률을 이와 같이 설정한 이유는 트래픽을 발생시키는 메쉬라우터의 수가 4일 경우, 각 라우팅 기법들이 최대한의 처리량을 낼 수 있는 값이기 때문이며, 트래픽을 발생시키는 메쉬라우터의 수가 증가할수록 성능의 차이를 쉽게 알 수 있기 때문이다.

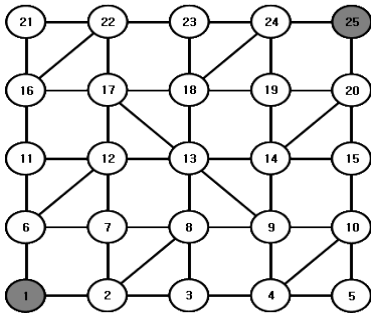


그림 11. 5x5 메쉬 토폴로지
Fig. 11. 5x5 Mesh topology

그림 11에서 라우터 25와 그림 12에서 라우터 49를, 인터넷에 접속하기 위한 게이트웨이 역할을 수행하는 게이트웨이 메쉬 라우터로 가정한다. 메쉬 라우터 1과 게이트웨이 메쉬 라우터간의 통신 및 트래픽을 발생시키는 다른 메쉬 라우터간의 통신은

TCP와 UDP를 사용하여 평가하였고, 잉여 대역폭 정보를 교환하기 위한 HELLO 메시지의 전송주기는 2초로 한다.

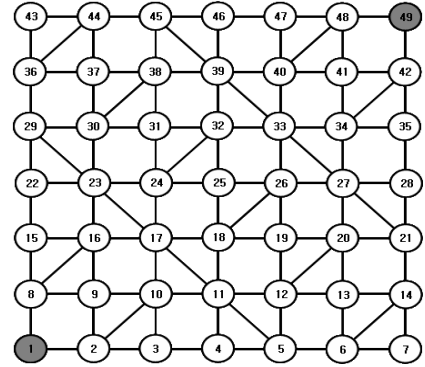


그림 12. 7x7 메쉬 토폴로지
Fig. 12. 7x7 Mesh topology

트래픽을 발생시키는 임의의 메쉬 쌍의 노드 수를 변화시키면서 시뮬레이션을 각각 1,000초 동안 수행하였으며, 트래픽 발생이 시작하는 시간은 100초이고 트래픽발생이 끝나는 시간은 200초이다. 트래픽 발생을 100초에서 시작하는 이유는 각 라우터에서 초기 라우팅 테이블을 완성하기 위한 시간을 주기 위해서이다.

2. 실험 결과 및 분석

그림 13과 그림 14는 우회 홉 수에 따른 메쉬 네트워크 라우터에서의 처리량을 측정된 결과인데, 우회 홉 수의 값이 고정된 환경에서 통신에 참여하는 메쉬 라우터의 수를 변경시켜 가면서 그 성능을 측정된 것이다.

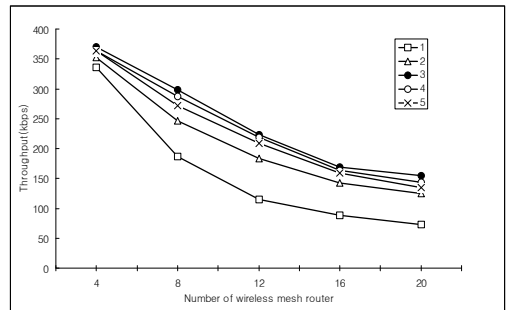


그림 13. 우회 홉 수에 따른 처리량(5x5, TCP)
Fig. 13. Throughput of detour hop number (5x5, TCP)

통신에 참여하는 메쉬 라우터 수가 증가하면 채널 경쟁으로 인한 충돌이 더 자주 발생되어 처리량은 감소하게 되는 것을 알 수 있다.

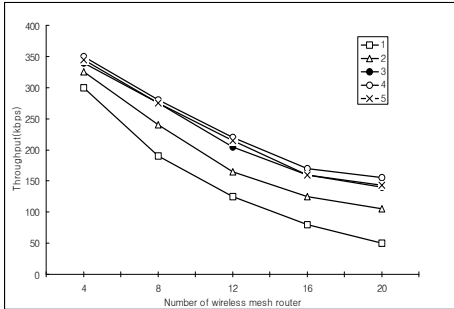


그림 14. 우회 홉 수에 따른 처리량(7x7, TCP)
Fig. 14. Throughput of detour hop number

여기서 실험한 우회 홉 수란 출발 라우터에서 목적지 라우터까지 경로선택을 하는 동안 총 우회할 수 있는 홉 수이며 한 라우터에서 최대 우회할 수 있는 홉 수는 두 홉이다.

이 같은 조건에서는, 우회 홉 수가 1에서부터 증가 할수록 성능이 더 좋아 지게 되어, 5x5 메쉬 토폴로지에서는 우회 홉 수가 3이 될 때, 7x7 메쉬 토폴로지에서는 우회 홉 수가 4가 될 때 최적의 성능을 가지게 된다. 그러나 우회 홉 수가 최적의 성능을 나타내는 3과 4보다 더 큰 경우에는 우회 홉 수가 증가하더라도 성능에는 큰 차이가 없음을 알 수 있다. 본 연구에서는 기본적으로 우회 경로의 값을 그림 11의 환경에서는 우회 홉 수를 3으로 정하고, 그림 12의 환경에서는 우회 홉 수를 4로 설정하여 시뮬레이션을 수행하였다.

그림 15와 그림 16은 통신에 참여하는 메쉬 라우터의 수에 따른 처리량으로, 그림에서 보듯이 통신에 참여하는 메쉬 라우터의 수가 적을 경우, 모든 기법에서 우수한 성능을 나타내는 것을 알 수 있다. 왜냐하면 트래픽이 부하가 적은 경우에는 DSDV 프로토콜로도 충분한 성능을 제공할 수 있기 때문이다. 그러나 패킷을 교환하는 메쉬 라우터의 수가 증가하는 경우 DSDV 방식과 HMP DSDV 방식보다, EHMP DSDV 방식과 BIMP DSDV 방식이 더 좋은 성능을 가지는 것을 알 수 있는데, 이는 DSDV 기법과 HMP DSDV 기법이 최소 홉 수를 가지는 특정 경로로만 집중되어 트래픽이 증가하는 경우 성능이 감소하기 때문이다.

통신에 참여하는 메쉬 라우터의 수가 12 이상이

되는 경우에는 성능의 차이가 더 커지는 것을 알 수 있는데, 이것은 다중 경로를 통해 트래픽을 분산시켜 충돌을 줄여 줄 수 있기 때문이다. 그리고 EHMP DSDV 기법 보다 BIMP DSDV 기법이 성능이 더 우수한 것을 알 수 있는데, 이것은 우회한 이후의 잉여 대역폭을 알 수 있으므로, 우회한 이후의 대역폭이 나쁘지 않으면 성능의 차이는 없지만 그렇지 않을 경우에는 잉여 대역폭이 좋은 경로로 트래픽을 분산시킬 수 있기 때문이다.

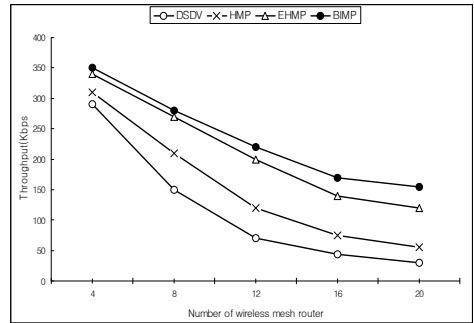


그림 15. 트래픽 증가에 따른 처리량(7x7, TCP)
Fig. 15. Throughput of increasing traffic

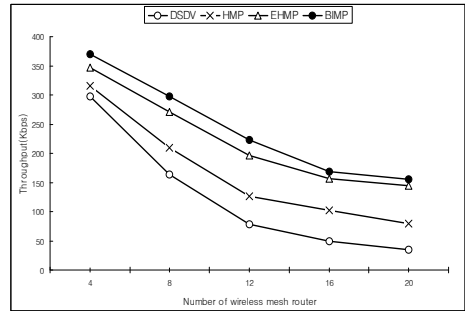


그림 16. 트래픽 증가에 따른 처리량(5x5, TCP)
Fig. 16. Throughput of increasing traffic

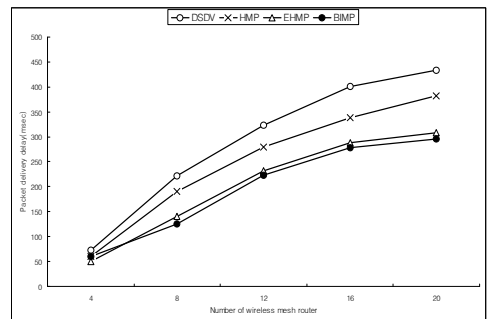


그림 17. 패킷 전달지연(5x5, TCP)
Fig. 17. Packet transfer delay(5x5, TCP)

그림 17은 통신에 참여하는 메쉬 라우터의 수에 따른 패킷 전달지연 시간을 나타낸 것이다. 메쉬 라우터의 수가 증가하면, 전달지연 시간도 증가하는 것을 알 수 있다. 그 이유는 크게 두 가지로, 첫째는 MAC 계층에서 채널 경쟁으로 인한 충돌이 더 자주 발생되어 전달지연 시간이 증가하는 것이고, 두 번째는 트래픽의 증가에 기인한 TCP의 흐름제어로 전달지연 시간이 증가하기 때문이다.

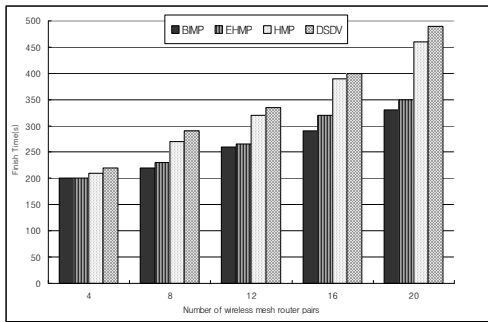


그림 18. 패킷 전송 종료시간(5x5, TCP)
Fig. 18. Packet transfer over time(5x5, TCP)

그림 18은 통신에 참여하는 메쉬 라우터의 수에 따른 패킷 전송 종료시간을 나타낸 것인데, 그림과 같이 통신에 참여하는 메쉬 라우터의 수가 증가 할 수록 패킷 전달지연 시간도 증가 하는 것을 알 수 있다. 통신에 참여하는 메쉬 라우터의 수가 적을 경우에는 전송 종료 시간인 200초에서 패킷전달이 종료되는 것을 알 수 있지만, 통신에 참여하는 메쉬 라우터의 수가 많을 경우에는 패킷 전송 종료 시간이 크게 증가하는 것을 알 수 있다.

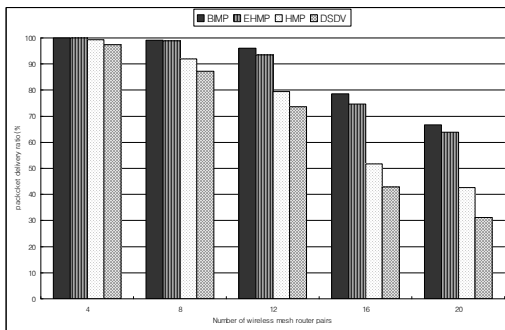


그림 19. 트래픽에 따른 패킷 전달율(5x5, UDP)
Fig. 19. Packet rate by traffic(5x5, UDP)

그림 19는 UDP 환경에서 통신에 참여하는 메쉬 라우터의 수에 따라 트래픽에 따른 패킷 전달율을 그래프로 나타낸 것이다. TCP 환경에서는 트래픽이 증가 할수록 패킷 전달 지연 시간이 증가하는 것을 알 수 있다.

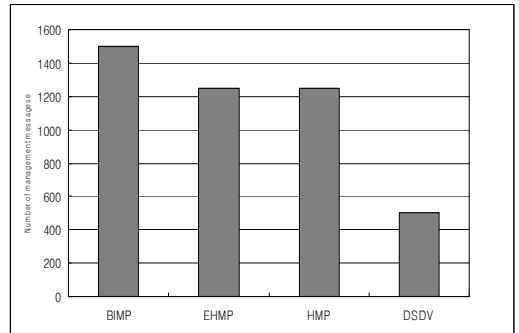


그림 20. 관리 메시지 수(5x5, TCP)
Fig. 20. Number of management messages

그림 20은 5x5 메쉬 토폴로지에서 시뮬레이션 시간동안 발생하는 관리 메시지의 수를 나타낸 것이다. 제안한 기법인 BIMP DSDV 기법에서는 관리 메시지의 수가 가장 큰 것을 알 수 있는데, 이것은 EHMP DSDV 기법과 EHMP DSDV 기법과 같이 포워딩 테이블 정보와 잉여 대역폭 정보를 전달 하지만, 제안한 기법에서는 두 홉 우회와 최소 경로를 가지는 두 경로와 같은 경우, 이웃 메쉬 라우터가 아닌 또 다른 메쉬 라우터에게 잉여 대역폭 정보를 전달하기 때문에 관리 메시지의 수가 증가한다.

5. 결론

무선 네트워크 기술의 발전에 힘입어 무선 메쉬 네트워크로 유선통신망이 없는 산악지역이나 오지, 재난지역에서도 인터넷 서비스가 가능할 전망이다. 이에 따라 무선 메쉬 네트워크에서 라우팅은 매우 중요한 요소기술이다. 무선메쉬 네트워크에서 라우팅 기술은 데이터를 효과적으로 전달하기 위한 핵심기술로, 유사한 통신 특성을 가지는 이동 애드혹 네트워크의 라우팅 프로토콜을 네트워크 특성에 맞게 변형하여 적용하는 방안이 논의되고 있다.

본 연구에서는 무선메쉬 네트워크에서 우회정보를 이용한 다중경로 DSDV 라우팅 기법을 제안하였다. 본 연구에서 제안하는 기법은 기존의 DSDV 기

법을 바탕으로 하여 EHMP DSDV기법의 단점인 우회 한 이후의 잉여 대역폭을 알지 못하는 것을 한 홉 우회, 두 홉 우회, 최소 홉 수를 가지는 두 경로와 같은 세 가지 방식을 통하여 성능을 향상시켰다.

제안한 BIMP DSDV 기법과 기존의 DSDV, HMP DSDV, EHMP DSDV 기법의 성능 비교를 위해 트래픽에 따른 처리량, 패킷 전달 지연, 패킷 전달율, 관리메시지 수, 패킷전송 종료시간 등의 관점에서 비교하였다. 시뮬레이션을 통해 제안한 BIMP DSDV 프로토콜은 네이버 테이블의 증가와 라우팅 테이블정보를 주고받는 관리 메시지가 일부 증가하는 단점은 있지만, EHMP DSDV 기법에서 트래픽이 우회 한 이후의 대역폭을 알지 못하여 발생하는 문제점을 해결하여, 전체적으로 기존의 프로토콜에 비해 처리량 및 패킷 전달률 측면에서 더 우수한 성능을 가지는 것을 알 수 있다.

참 고 문 헌

[1] 김광식, 조무호, "특허정보로 바라본 WMN(무선 메쉬 네트워크) 기술", IT 기획 시리즈, pp. 28-41, 2010.
 [2] C. E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers", Proceedings of SIGCOMM, pp. 234-244, 1994.
 [3] 오은주, 임재성, "무선 Mesh 네트워크에서 Mesh라우터 간에 성능향상을 위한 홉 간 멀티 패스(Hop-by-Hop Multipath) 라우팅 기법", 한국통신학회 추계 학술대회 논문집, 2005.
 [4] 이성용 정윤원, "무선 메쉬 네트워크에서 홉 수와 잉여 대역폭을 이용한 개선된 다중경로 DSDV 라우팅 프로토콜", SK Telecom. Review, 제17권 제6호, pp. 1066-1077, 2007.
 [5] 김영안, 박철현, 홍충선, "무선 메쉬네트워크 환경에서 효율적인 다중 홉 전달 기법", 한국통신학회 논문지, Vol.31, No.10B, pp. 872-882, 2006.
 [6] I. F. Akyildiz and W. Wang, "Wireless mesh networks: a survey", Computer Networks, Vol.47, issue 47, pp. 445-487, 2005.
 [7] W. H. Tam, Y. C. Tseng, "Joint multi-channel link layer and multi-path routing design for wireless mesh networks", Proceedings of INFOCOM, pp. 2081-2089, 2007.

[8] M. Jiangrui, T. Kun, and . . . tian, "Joint routing and channel assignment in multi-radio wireless mesh networks", Proceedings of ICC, Vol.8, pp. 3596-3601, 2006.
 [9] L. Ma and M. K. Denko, "A routing metric for load-balancing in wireless mesh networks", Proceedings of AINAW, Vol.2, pp. 409-414.
 [10] W. Jiang, S. Liu, Y. . hu and . . . hang, "Optimizing routing metrics for large-scale multi-radio mesh networks", Proceedings of WiCom, pp. 1550-1553, 2007.
 [11] 이성희, 고영배, "무선 메쉬 네트워크 표준화 동향", 정보통신기술 제20권 제1호 pp. 80-94, 2007.
 [12] The Network Simulator ns-2.

저 자 소 개

이 도엽



2008년 : 대구대학교 정보통신공학과 학사.
 2010년 : 대구대학교 정보통신공학과 석사.
 현재, 드림소프트 연구원.

관심분야 : 임베디드하드웨어, 센서네트워크.
 Email : shine411@dreamsystem.co.kr

김 중규



1984년 : 연세대학교 전자공학과 학사.
 1986년 : 연세대학교 전자공학과 석사.
 1992년 : 연세대학교 전자공학과 박사.

현재, 대구대학교 정보통신공학부 교수.
 관심분야 : 컴퓨터네트워크, RFID/USN, 인터넷프로토콜.
 Email : jgkim@daegu.ac.kr