

# Traffic Estimation 을 이용한 무선 메쉬 네트워크에서 효율적인 경로 설정 기법

정연호\*, 김형태\*, 박재형\*  
\*전남대학교 전자컴퓨터공학과  
e-mail : loven7942@lycos.co.kr

## An Efficient Routing Algorithm using Traffic Estimation Scheme in Wireless Mesh Networks

Younho Jung\*, HyungTae Kim\*, Jaehyung Park\*  
\*Dept. of Computer Engineering, Chonnam National University

### 요 약

본 논문은 무선 메쉬 네트워크의 다중 라디오 인터페이스와 다중 채널을 이용한 효율적인 멀티홉 무선 전송을 위한 네트워크 라우팅에 관해 다룬다. 이를 위한 라우팅 매트릭으로는 각 중간 노드의 대역폭 사용량 정보에 기반한 대역폭 예상 변화량을 사용하여 실제 데이터 전송이 진행될 때 기존 보다 안정적인 전송이 가능하도록 유도한다. 멀티 인터페이스 / 멀티 채널의 수의 제약이 존재하는 현실적인 경우를 고려하여 라우팅 매트릭을 정립하고 라우팅 알고리즘을 설계하였다. NS-2 시뮬레이터를 이용한 시뮬레이션 평가를 통해 트래픽 사용량 정보에 기반한 제안 알고리즘이 기존 MIMC 라우팅 알고리즘보다 개선된 성능을 보여준다.

### 1. 서론

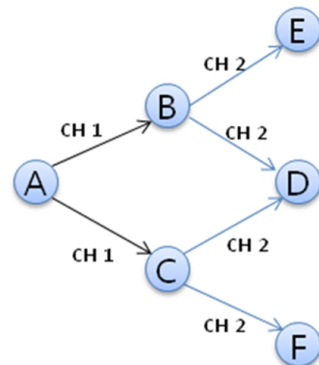
최근 스마트 폰과 태블릿 PC 의 빠른 보급으로 셀룰러 네트워크에서의 LTE [1] 와 더불어 데이터 네트워크에서는 와이파이 기술을 적용한 무선 네트워크 기술들이 빠르게 상용화되고 있다. 정부의 ALL-IP 융합 네트워크 인프라 구축 로드맵과 맞물려 무선 메쉬 네트워크[2]에 대한 관심이 높아지고 있다. 본 논문에서는 무선 메쉬 네트워크에서 데이터 전송 성능을 높이기 위해 트래픽 예상 기법을 적용하여 라우팅 알고리즘을 작성하고 성능평가를 수행한다.

### 2. 문제점

멀티채널과 멀티 인터페이스 무선 환경에서의 라우팅에 대한 연구는 라우팅 매트릭 연구 및 라우팅 기법 연구, 채널 할당 문제, 간접 인지 라우팅, 토폴로지 컨트롤 기법 등이 주요 연구 범위로서 연구가 이루어져 왔다. 본 논문의 주제도 라우팅 매트릭과 라우팅 기법에 관한 것으로 Multi-Interface / Multi-Channel (MIMC) 기반의 선행연구들[3,4,5]의 단점에서 출발한다. 즉, 현재 구현된 많은 라우팅 알고리즘들은 싱글 인터페이스 / 싱글 채널 환경(Single Interface Single Channel, SISC)을 기반으로 동작한다. SISC 기반의 라우팅 알고리즘은 채널 다양성 요소를 구분하지 못하여 멀티 인터페이스 / 멀티 채널 환경에서도 효

율적인 리소스 활용이 어렵다.

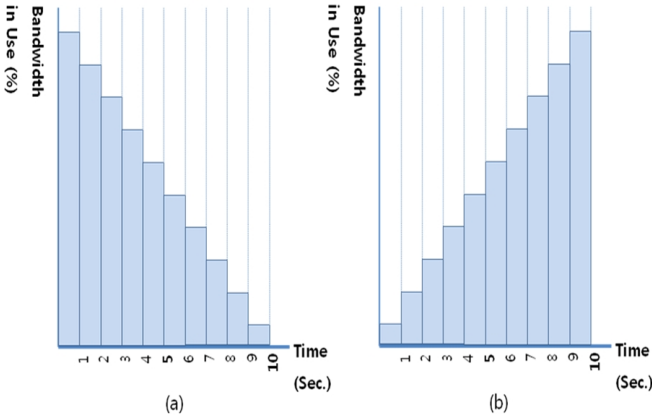
다른 문제점으로는 라우팅 알고리즘이 경로의 품질을 판단하는 시점(경로 탐색 과정)과 설정된 경로를 이용하여 데이터를 전송하는 시점에서 경로의 품질은 일치하지 않을 수 있다는 것이다. 이는 모바일 노드의 이동성에 의해 자신의 위치가 이동되어 설정된 경로를 재설정해야 하는 경우와, 이웃한 중계 노드의 이동 및 이탈로 새롭게 경로를 변경해야 하는 경우가 생기기 때문이다. 또한 새로운 트래픽 전달이 시작되는 경우, 링크 품질이 달라지기 때문에 경로 탐색 과정에서 판단한 링크의 품질은 데이터 전송과정에서는 일치하지 않는다.



(그림 1) MIMC 토폴로지 예

그림 1 은 MIMC 환경의 무선 토폴로지를 구성한 예이다. 소스 노드 A 에서 E 와 F 로 향하는 두 개의 데이터 트래픽의 전송량이 그림 2 의 경우처럼 한쪽은 감소하고 한쪽은 증가한다고 가정한다. 그리고 이 두 트래픽의 중간 노드를 공유하는 새로운 트래픽 전송을 시도하고자 할 때, 두 중간 노드(B 와 C)의 대역폭 사용량이 비슷하다고 가정하여 본다.

기존 라우팅 알고리즘은 새로운 트래픽에 대한 경로 탐색 / 응답 메시지가 먼저 도착하는 경로를 데이터 전송 경로로 설정한다. 이는 가장 단순한 기법이지만, 경로 탐색을 하는 순간의 각 경로마다의 전송 품질을 유추할 수 있는 방법이다. 다시 말해 먼저 도착하는 응답 메시지가 거처온 경로가 경로 탐색 순간에는 전송 품질이 더 좋다는 것을 의미한다. 하지만, 이 정보는 토폴로지 내의 각 트래픽의 전송량 변화가 거의 없는 경우이거나 전송 트래픽량의 차이가 큰 경우에만 어느 정도 정확한 방법이다.



(그림 2) 두 중간 노드의 사용중인 대역폭 비교

### 3. 제안 경로 설정 기법

본 논문에서는 기존 라우팅 알고리즘의 두 가지 문제점에 대해 가능한 접근방법으로 과거의 데이터 패킷(중간 노드들이 가지는 자신의 트래픽 전송량 정보)를 사용하는 라우팅 알고리즘 및 라우팅 매트릭을 다룬다.

제안 라우팅 알고리즘에서는 중간 노드들이 가지는 전송 중인 트래픽 량을 정확히 모니터링 하여 트래픽 량이 비슷한 경우에도 더 나은 경로를 선택할 수 있게 도와주고자 한다. 예를 들어 새로운 트래픽의 전송 시도 시작 시점이 5 초라면, 그림 2 의 두 중간 노드의 사용 중인 대역폭은 거의 비슷한 상황이 된다. 따라서 경로 탐색 / 응답 메시지가 소스 노드에 도착하는 시점은 거의 차이가 없게 되어, 기존 라우팅 알고리즘을 사용하면 도착 시점의 아주 미세한 순간의 차이로 전송 경로의 선택이 이루어지게 된다. 이런 경우에 기존 알고리즘은 때에 따라서 각기 다른 경로를 선택하게 되어 경로 선택에 있어 일관성이 없다.

제안 라우팅 알고리즘은 그림 2 의 그래프 (a) 형태의 사용중인 대역폭을 가지는 중간 노드를 선택하도록 유도하여 경로 설정이 이루어진 이후 데이터 트래픽 전송과정에서 기존 라우팅 알고리즘 보다 원활한 전송이 가능하게 한다.

비교를 위한 계산은 아래와 같이 진행될 수 있다. 경로가 지닌 트래픽 사용량의 이전 값과 그 다음 값(최근 값)의 비교하여 기울기를 계산하고 최근 값에 곱하여 트래픽의 사용 경향을 알 수 있다. 이렇게 계산된 값은 경향(방향성)을 알 수 있는데 도움을 주지만, 가장 주의 깊게 살펴야 하는 값은 과거의 트래픽 사용정보보다는 최근의 트래픽 사용 값이다. 따라서 최근 값에 가중치를 주어야 하는데, 이는 2 의 자승의 형태로 다시 한번 곱해줌으로써 구할 수 있다.

$$\sum_{i=0}^n ((BW_i - BW_{i+1}) \times BW_i \times 2^{c-i})$$

(그림 3) 경로 품질 평가 수식

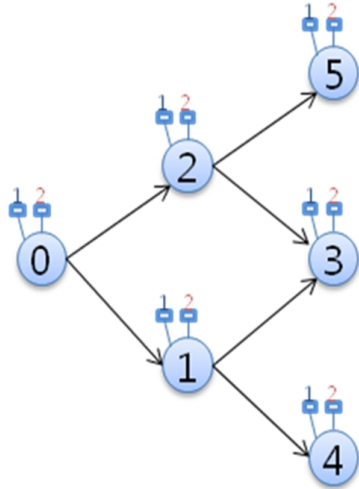
그림 3 의 수식은 제안 알고리즘이 현재와 가까운 정보에 가중치를 주는 방법이다. 이를 통해 수신된 여러 경로 탐색 메시지가 거처온 경로 중 더 나은 중간 노드들을 선택할 수 있다. 수식에서 n=4 이고, c 는 5 로 설정한다. BW 는 Bandwidth 를 상징하는데 이전 대역폭 정보에서 현재 대역폭 정보를 빼서 기울기를 구하고 여기에 현재 대역폭을 다시 곱하여 트래픽의 대역폭 사용추이를 계산한다. 여기에 2 의 c-i 승 값을 다시 곱하게 되면 최근 대역폭 값을 더 크게 만들어주어 최종 품질 값에 많은 영향을 주도록 유도한다.

예를 들어, 두 중간 노드의 사용중인 대역폭이 각각 1, 2, 3, 4, 5 로 증가하는 경우와 10, 9, 8, 7, 6 으로 감소하는 경우에 단순히 최소값을 계산하면 증가중인 중간 노드를 선택하지만, 위 수식을 통해 기울기와 최근 값에 가중치를 더 주게 되면 감소하는 중간 노드를 선택할 수 있게 된다. 증가중인 중간 노드의 경로 품질 값은  $((5-4)*5*(2^5)) + ((4-3)*4*(2^4)) + ((3-2)*3*(2^3)) + ((2-1)*2*(2^2)) + ((1-0)*1*(2^1))$  으로 계산되어 258 이 되고, 반면 감소중인 중간 노드의 품질 값은  $((6-7)*6*(2^5)) + ((7-8)*7*(2^4)) + ((8-9)*8*(2^3)) + ((9-10)*9*(2^2)) + ((10-11)*10*(2^1))$  으로 계산되어 -424 가 나온다. 따라서 사용 대역폭의 최근 값은 비슷하거나 오히려 크지만 계산에 의해 낮은 품질 값을 갖는 중간 노드를 선택하여 실제 전송이 진행되면 더 좋은 경로 품질을 보장해 줄 수 있게 된다.

### 4. 시뮬레이션 평가

제안 라우팅 매트릭의 시뮬레이션을 위해 본 연구에서는 MIMC 라우팅 알고리즘의 소스코드[6]를 기본으

로 NS-2 시뮬레이터[7]를 이용하여 시뮬레이션을 진행하였다. 시뮬레이션을 위해 그림 4의 6개 노드를 지닌 토폴로지를 가지고 실험하였으며, 노드 인터페이스 및 채널의 제약이 있는 경우를 가정하였다.



(그림 4) 시뮬레이션 토폴로지

즉, 그림 4의 각 노드 위 안테나 모양으로 나타낸 것은 물리적인 무선 인터페이스를 상징하며, 그 위 숫자는 할당된 채널의 번호를 의미한다. 즉 채널과 인터페이스 수가 여러 개인 MIMC 네트워크를 구성하고 있지만, 전송 시에는 전송 트래픽의 수가 이를 초과하므로 채널 및 인터페이스의 제약을 가진다.

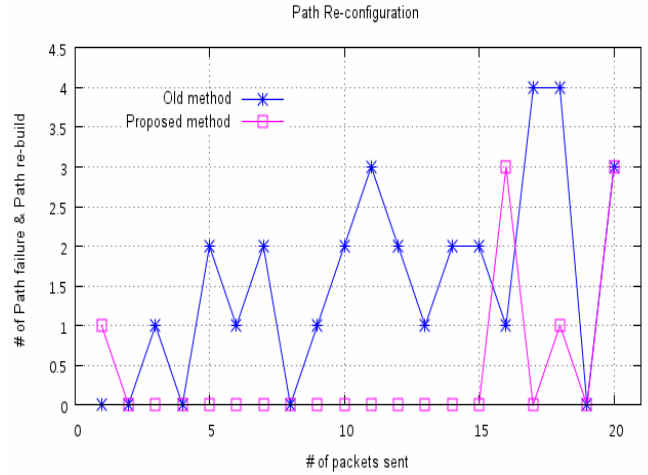
평가 항목으로는 경로 단절 및 재설정 횟수, 전송 완료된 트래픽 패킷 수, 전송 지연 시간을 계산하여 성능을 평가하였다. 시뮬레이션 시나리오는 기존 전송 중인 트래픽은 2개가 존재하고, 트래픽 A는 노드 0 - 2 - 5를 거치는 경로이고, 트래픽 B는 노드 0 - 1 - 4를 지나는 경로이다. 여기에 새로운 트래픽 C가 노드 0에서 1 또는 2를 거쳐 3에 이르는 전송을 시뮬레이션 시작 10초부터 시작한다. 이때 각각의 물리적 환경에 따라 어떠한 결과를 내는지 확인하도록 한다.

<표 1> 시뮬레이션 환경설정

Routing	MIMC, 제안알고리즘
Mac Type	Mac/802_11
Interface Queue Type	Queue/DropTail/PriQueue
Interface Length Type	50
Antenna Type	Antenna/OmniAntenna
Propagation Type	Propagation/TwoRayGround
Physical Type	Phy/WirelessPhy
Channel	Channel/WirelessChannel
Traffic	FTP Traffic

표 1은 시뮬레이션에 적용된 환경설정 값들을 나타낸다. 시뮬레이션에서는 무선 수신 가능 경계 값인

RxThresh 값을 175m로 설정하여 오버헤어링으로 인한 데이터 전달 가능성을 줄이고, 라우팅 매트릭에 의한 라우팅의 영향을 측정하고자 하였다.

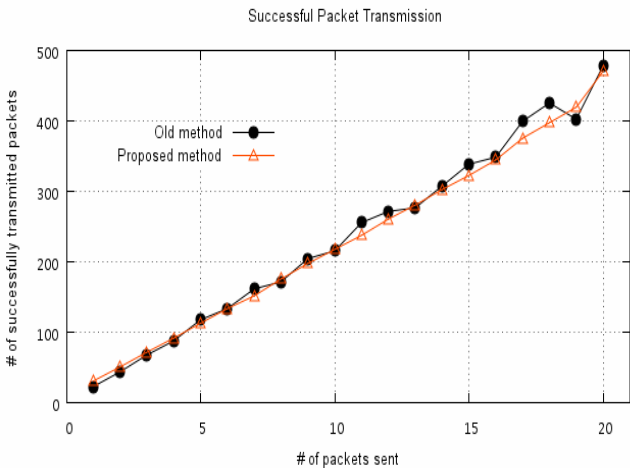


(그림 5) 경로 재설정 횟수

인터페이스와 채널의 수를 제한한 경우의 시뮬레이션 결과를 살펴본다. 즉, 노드 별 인터페이스와 채널의 수를 각각 2개로 고정된 경우이다. 노트북이나 모바일 가젯들이 가질 수 있는 현실적으로 가장 가까운 형태가 아닐까 사료된다. 노드별로 인터페이스 수와 채널의 수를 동일하게 맞춘 것은 인터페이스 수보다 채널의 수가 많을 경우 발생하는 연쇄적인 채널 변경 반응 (chain effect 또는 chain reaction)에 대한 처리를 수행하여야 하는데, 이는 논문의 주제에서 벗어나 있고, 해당 문제가 발생하는 특정 환경에만 시뮬레이션을 고정해야 하기에 논외로 두기로 한다.

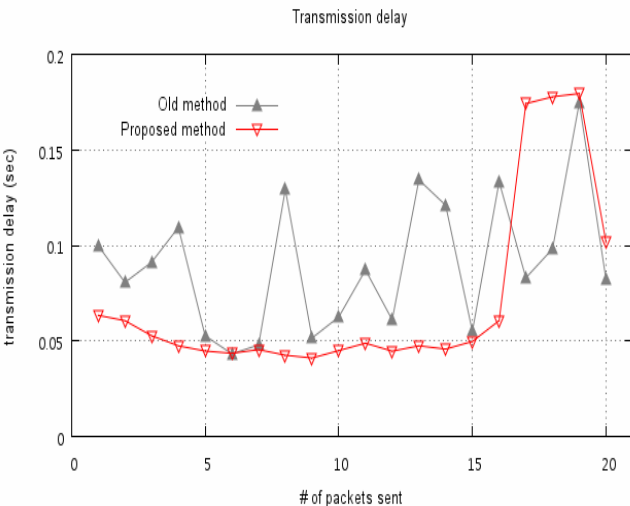
그림 5 ~ 7까지의 x축은 트래픽 C (10초에 새로 전송을 시작한 트래픽)을 0.5초 간격으로 몇 개를 보내는지를 x축에 나타낸 것이다. 즉, x=7인 경우는 트래픽 C가 10초 이후부터 시뮬레이션 종료(23초)까지 0.5초당 7개씩의 패킷을 보냈다는 뜻이 된다. 실험 결과 대략적으로 0.5초당 패킷 수 13개 이하까지는 전송 대기 시간이 크지 않고, 이 이상이 되면 이미 전송 중인 두 트래픽 A, B의 영향으로 소스 노드가 트래픽 C를 전송 시작하기 전 기다리는 시간이 계속적으로 늘어나게 된다.

그림 5는 채널 / 인터페이스의 제약이 있는 무선 MIMC 환경에서의 경로 재설정 횟수를 표시한 것이다. 트래픽의 사용량에 기반한 제안 라우팅 알고리즘은 거의 대부분의 구간에서 기존 라우팅 알고리즘에 비해 월등한 성능을 보여주고 있다. 의미 있는 구간인 트래픽 C의 전송 패킷 수 13개 이하 영역에서 경로 단절과 경로 재설정이 거의 일어나지 않는 안정적인 모습을 보여준다.



(그림 6) 전송 성공한 트래픽 C의 패킷 수

그림 6은 트래픽 C가 전송한 FTP 패킷이 목적지 노드(노드 3)에 몇 개가 도착하는지를 표시한 그래프이다. 도착한 패킷의 수는 패킷 전송 성공률과는 전혀 관련이 없다. 이는 FTP 서비스 자체가 신뢰적 전송을 통하여 한번 보낸 패킷에 대해서는 책임을 지기 때문이다. 결과를 살펴보면 기존 알고리즘과 제안 알고리즘 모두 특별한 차이를 보이지 않는다. 따라서 이 그래프만으로 성능의 차이를 비교하는 것은 무리가 따른다.



(그림 7) 트래픽 C의 전송 지연 시간

그림 7은 트래픽 C의 전송 지연 시간을 나타내는 그래프이다. 의미 있는 구간인 트래픽 C의 0.5초당 패킷 전송 수 13개 이하의 모든 구간에서 기존 라우팅 알고리즘보다 더 나은 성능을 보여주고 있다. 또한 주목해야 할 점은 인터페이스 수 / 채널 수의 제약이 있는 환경에서 제안 라우팅 알고리즘이 대단히 고른 성능을 보였다는 점이다. 이는 3개의 결과 그래프 모두에 걸쳐 나타나며 결과적으로 제안 라우팅 알고리즘을 통해 MIMC 무선 네트워크의 전반적인 전송 성능과 경로 안정성에 좋은 영향을 미쳤다고 평가할 수 있다.

## 5. 결론

본 논문은 ALL-IP 융합 네트워크 인프라 구축에 핵심이 되는 IP 기반의 무선 데이터 네트워크인 WMN에서의 커버리지 확대 및 서비스 안정성에 도움이 될 수 있는 다중 라디오 인터페이스와 다중 채널을 이용한 멀티홉 무선 네트워크의 라우팅에 관해 다루었다. 논문에서 여러 종류의 무선 기기들(AP, 노트북, 모바일 가젯 등)이 멀티홉 전송을 하는 환경을 가정하고, 멀티 인터페이스 / 멀티 채널의 수의 제약이 존재하는 현실적인 경우를 고려하여 라우팅 매트릭을 정립하고 라우팅 알고리즘을 설계하였다. 라우팅 매트릭으로는 각 중간 노드의 대역폭 사용량 정보에 기반한 대역폭 예상 변화량을 추정된 값을 사용하여 실제 데이터 전송이 진행될 때 기존보다 안정적인 전송이 가능하도록 유도하였다. NS-2 시뮬레이터를 이용한 시뮬레이션을 통해 트래픽 사용량 정보에 기반한 제안 알고리즘이 멀티 인터페이스의 수 / 멀티 채널의 수와 관계없이 모든 환경에서 기존 MIMC 라우팅 알고리즘보다 개선된 성능을 보여준다.

## 참고문헌

- [1] 3GPP Long Term Evolution, [http://en.wikipedia.org/wiki/3GPP\\_Long\\_Term\\_EvEvolution](http://en.wikipedia.org/wiki/3GPP_Long_Term_EvEvolution)
- [2] Wireless mesh network, [http://en.wikipedia.org/wiki/Wireless\\_mesh\\_network](http://en.wikipedia.org/wiki/Wireless_mesh_network)
- [3] Sung-Hee Lee; Young-Bae Ko; Youg-Geun Hong; Hyoun-Jun Kim; , "A new MIMC routing protocol compatible with IEEE 802.11s based WLAN mesh networks," *Information Networking (ICOIN), 2011 International Conference on*, vol., no., pp.126-131, 26-28 Jan. 2011
- [4] IEEE 802.11s, [http://en.wikipedia.org/wiki/IEEE\\_802.11s](http://en.wikipedia.org/wiki/IEEE_802.11s)
- [5] HWMP Specification, <https://mentor.ieee.org/802.11/public/06/11-06-1778-01-000s-hwmp-specification.doc>
- [6] Raniwala, A.; Tzi-cker Chiueh; , "Architecture and algorithms for an IEEE 802.11-based multi-channel wireless mesh network," *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, vol.3, no., pp. 2223- 2234 vol. 3, 13-17 March 2005
- [7] MIMC routing source code, NS-2의 이해와 활용 (홍릉출판)
- [8] The Network Simulator - ns-2, <http://www.isi.edu/nsnam/ns/index.html>