

무선 메시 네트워크 환경에서 오픈플로우 프로토콜을 이용한 QoS 지원 기법 설계

강용혁¹, 김문정^{2*}, 김수선³

¹극동대학교 글로벌경영학과, ²U1대학교 스마트IT학과, ³한양여자대학교 컴퓨터정보과

Design of QoS Supporting Mechanism using Openflow Protocol in Wireless Mesh Network Environments

Yong-Hyeog Kang¹, Moon Jeong Kim^{2*}, Su-Sun Kim³

¹Department of Global Business Administration, Far East University

²Department of Smart IT, U1 University

³Department of Computer Information, Hanyang Women's University

요약 무선 메시 네트워크는 무선 노드 간에는 다중 홉 라우팅을 통해 통신을 하며 인터넷과 연결된 게이트웨이를 통해 인터넷과 연결되는 네트워크이다. 이러한 네트워크는 부분적으로 기존 무선 접근 네트워크의 유선 백본 역할을 수행할 수 있어서, 다양한 응용이 가능한 확장 가능한 네트워크이다. 본 논문에서는 무선 메시 네트워크 환경에서 소프트웨어 정의 네트워킹 기술을 이용하여 QoS를 지원하는 기법을 설계한다. 제안 기법에서는 무선 메시 라우터의 자원 제약적인 상황을 고려하여 비용 효율적이고 상황 적응적인 QoS 지원 기법을 설계한다. 제안 기법은 QoS 지원 기법을 차별 서비스와 통합 서비스로 구분하여 설계하고, 소프트웨어 정의 네트워킹 기술인 오픈플로우 프로토콜을 이용하여 MPLS 기법과 RSVP를 통해 각각의 상황에 적합한 모델을 제시한다. 제안 기법의 타당성 검증을 위해 다양한 QoS 메트릭들에 대한 성능 평가 모델을 제시하였으며 제안 기법이 제시하는 QoS 지원 기법이 비용 효율적으로 동작하는 지를 제시한다.

Abstract Wireless mesh networks contain multi-hop routing protocols between wireless nodes and are connected to the Internet through a gateway. These networks perform a role as a backbone and are scalable for main applications. We propose the design of QoS supporting mechanisms for wireless mesh networks using software defined networking. Our proposed scheme is cost-effective and features adaptive QoS mechanisms for wireless router's resource constraints. The QoS mechanisms use software defined networking technology with Openflow protocol based on diffserv and intserv models with MPLS mechanism and RSVP respectively. A performance evaluation model is suggested to verify the validity of the proposed scheme using several QoS metrics of the wireless mesh networks.

Keywords : MPLS, Openflow, RSVP, Quality of Service, Software Defined Networking, Wireless Mesh Network

1. 서론

무선 메시 네트워크(WMN: Wireless Mesh Network)는 다양한 유비쿼터스 인터넷 접근을 제공하는 유망한 기술로 나타났고, 넓은 범위의 무선 커버리지를 가지며,

비용 효율적이고 빠른 배치 및 최소한의 하부 구조 요구를 통해 많은 응용에 적용될 수 있다[1]. 무선 메시 네트워크를 구성하는 네트워크 장비는 메시 라우터(mesh router), 메시 AP(mesh Access Point), 메시 클라이언트(mesh client) 타입이 있다. 무선 메시 네트워크는 전용

*Corresponding Author : Moon Jeong Kim(U1 Univ.)

Tel: +82-41-536-5722 email: tops@yd.ac.kr

Received June 12, 2017

Revised July 10, 2017

Accepted September 15, 2017

Published September 30, 2017

무선 메시 라우터를 통해 멀티홉 무선 백본을 구성한다. 무선 메시 라우터들은 멀티홉 라우팅을 통해 연결되며, 일부 라우터들은 인터넷을 연결하는 게이트웨이로 동작한다. 이러한 메시 네트워크의 유연성은 고속 홈네트워킹, 유비쿼터스 컴퓨팅 등 많은 응용들을 가능하게 한다.

소프트웨어 정의 네트워킹(SDN: Software Defined Networking) 기술은 컴퓨터 네트워킹 분야에 네트워킹 제어와 관리를 단순화하고 네트워킹 프로그래밍 가능성(Network Programmability)을 통해 네트워킹 분야에 혁신을 가져온 혁명적인 새로운 아이디어로 언급되고 있다 [2]. 포워딩 하드웨어와 제어 로직의 분리는 새로운 프로토콜 및 응용의 쉬운 배치를 허용하고 네트워킹 가상화와 관리를 용이하게 한다. 소프트웨어 정의 네트워킹은 미래 인터넷 개발의 유망한 트렌드가 되었다. 현재 연구들은 데이터 센터 네트워킹이나 캠퍼스 네트워킹 내에서 오픈플로우 스위치(openflow switch)를 이용하여 주로 플로우 관리, 보안, QoS(Quality of Service)에 집중하고 있지만, 실제 네트워킹에서 발생하는 문제를 해결하기에는 여전히 제한적이다[3].

본 논문에서는 무선 메시 네트워크의 다양한 응용의 서비스 품질을 위해 소프트웨어 정의 네트워킹 메커니즘을 이용하는 기법을 제안한다. 2장에서는 QoS 기법, 오픈플로우 기법 등 관련연구를 기술하고, 3장에서 무선 메시 네트워크 응용에서 QoS 지원을 위해서 오픈플로우가 어떻게 동작해야 하는지를 제시하고, 4장에서 제안 기법의 타당성을 분석하고, 5장에서 결론을 맺는다.

2. 관련연구

현재 인터넷이 제공하고 있는 서비스는 최선형(best-effort) 서비스이다. 최선형 서비스는 최대한 노력하지만 아무것도 보장하지 않는 서비스이며, 네트워크가 제공할 수 있는 가장 단순한 서비스이다. 최선형 서비스는 QoS를 요구하는 응용에는 부적합하기 때문에 통합 서비스(Integrated Service)와 차별 서비스(Differentiated Service)에 대한 연구가 진행되고 있다. 또한 그 외에 QoS 기법들에 대해 소개한다.

통합 서비스는 특정 플로우(flow)에 대한 자원 예약을 통해 자원을 보장하는 기법이며, 차별 서비스는 정책이나 프로비저닝(provisioning)이나 트래픽 우선순위의

조합을 통해 자원을 보장하는 기법이다[4].

2.1 통합 서비스(Integrated Service)

통합 서비스(intserv) 기법은 플로우 단위로 자원 예약(per-flow resource reservation)을 함으로써 보장 서비스(guaranteed service)와 제어된 부하 서비스(controlled load service) 모델을 선택할 수 있다. 보장 서비스 모델은 엄격한 수락 제어(admission control)와 공정 큐잉 스케줄링(fair queuing scheduling)을 통해 결정적인 최악 지연 상한치를 제공한다. 제어된 부하 서비스는 덜 엄격한 보장기법을 제공한다. RSVP(Resource reSerVation Protocol)은 응용 요구사항에 대한 시그널링과 경로를 따라 자원 예약을 설정하기 위해 표준화되었으나, 큰 규모(large-scale) 배치에는 적합하지 않다[5].

2.2 차별 서비스(Differentiated Service)

차별 서비스(diffserv) 기법은 최선형 서비스보다 더 좋은 서비스를 제공하지만 더 단순하고 확장 가능한 접근 방법이다. 이 기법은 플로우 단위(per-flow) 기반이 아니라 트래픽을 여러 개의 포워딩 클래스(class)로 구분하였다. 또한 트래픽의 양을 조절하여 자원 보장 정도를 조절한다. 트래픽의 구분은 SLA(Service Level Agreement)을 기반으로 하여 이루어지며, 네트워크 경계(edge) 단에서 트래픽 정책을 수행하여 부정행위하는 트래픽 소스를 막는다. 포워딩 클래스는 경계 단에서 패킷헤더에 인코딩되며, 인코딩 정보를 이용하여 트래픽 처리를 구분한다[4].

2.3 소프트웨어 정의 네트워킹 이용 기법

소프트웨어 정의 네트워킹을 이용한 QoS 기법들도 비디오 컨퍼런싱(video conferencing), WebTV 등의 스트리밍 미디어 응용을 위해 제안되었다. 트래픽 패턴에 따라 최적 경로를 선정하며, QoS 트래픽은 트래픽 우선순위에 의해 선점권을 갖는다. 컨트롤러를 이용하여 QoS 계약 관리, QoS 관리, 경로 계산, 경로 관리 및 네트워크 관리하는 기능을 수행한다. 또한 자원 모니터링과 자원 시그널링, 자원 예약 기능을 요구한다[3].

2.4 오픈플로우 이용 기법

오픈플로우를 이용하여 자동으로 QoS를 관리하는 기법도 제안되었다. QoS 플로우 컨트롤러가 QoS 플로우

를 관리하고 QoS 플로우를 모니터링한다. 오픈플로우 API와 QoS 플로우 API를 통해 오픈플로우 스위치와 상호작용한다. 멀티미디어 전송을 위해 오픈플로우 컨트롤러를 확장한 OpenQoS 기법도 제안되었다. 컨트롤러는 토폴로지 관리, 경로관리, 플로우 관리, 경로 계산, 호출 수락(call admission), 트래픽 정책 적용(policing)을 관리해야 한다[3].

2.5 오픈플로우기반 소프트웨어 정의 네트워킹

무선 메시 네트워크에서 오픈플로우기반 소프트웨어 정의 네트워킹을 이용하는 기법들이 제안 되고 있다. SD-WSN 기법은 데이터 플레인과 제어 플레인은 나누어졌지만, 데이터 플레인과 제어 플레인이 무선 자원을 공유할 때 발생하는 문제를 해결하기 위한 연구이다[6]. WMSDN 기법은 간헐적인 연결 상태인 노드들이 컨트롤러를 선택하는 데 소프트웨어 정의 네트워킹 기법을 적용하였다[7]. WMSDN 기법은 중앙 집중적인 SDN 컨트롤러를 통해 트래픽 공학을 수행하는 기법을 제시하였다[8].

3. 제안기법

본 논문에서 제안하는 기법은 무선 메시 네트워크 환경에서 소프트웨어 정의 네트워킹 기법을 사용하여 QoS를 제공하는 기법이다. 우선 소프트웨어 정의 네트워킹의 구현체인 오픈플로우 기법을 설명한 후 QoS를 위해서 오픈플로우를 수정하는 제안 기법을 설명한다.

3.1 오픈플로우 기법

오픈플로우 호환 스위치로는 오픈플로우 전용(Openflow-only)와 오픈플로우 하이브리드(Openflow-hybrid)의 2가지 유형이 있다[9]. 오픈플로우 전용 스위치는 오픈플로우 명령어만을 지원하며, 모든 패킷은 오픈플로우 파이프라인에 의해서 처리된다. 오픈플로우 하이브리드 스위치는 오픈플로우 명령어와 일반 이더넷 스위치 명령어, L2 이더넷 스위칭, L3 라우팅, ACL, QoS 처리를 지원한다.

오픈플로우 스위치의 컴포넌트 구조는 그림 1에서 보이는 바와 같다. 오픈플로우 스위치는 패킷 루업(lookup)과 포워딩을 수행하는 한 개 이상의 플로우 테이블과 한

개의 그룹 테이블 및 외부 컨트롤러와 연결되는 한개 이상의 채널로 구성된다. 컨트롤러는 오픈플로우 프로토콜을 통해 미리 또는 패킷에 대한 응답으로 플로우 테이블에 플로우 항목(flow entry) 추가, 갱신 및 삭제할 수 있다. 각 플로우 테이블에는 플로우 항목들의 집합이 저장되며, 각 플로우 항목은 매치 필드들, 카운터들 및 패킷이 매칭되었을 때 실행되는 명령어 집합으로 구성된다.

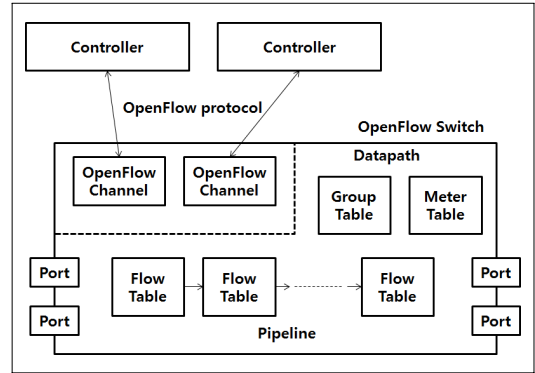


Fig. 1. Components of Openflow Switch

각 플로우 항목은 그림 2에서 보이는 바와 같다. 그림 2에서, 필드 매치 필드(Match Fields)는 패킷에 매칭하기 위한 필드이다. 필드 우선순위(Priority)는 매칭에 대한 우선순위를 나타내며, 필드 카운터(Counters)는 패킷이 매칭되었을 때 갱신된다. 필드 명령어들(Instructions)은 액션 집합(action set) 및 파이프라인(pipeline) 처리를 수정하기 위한 필드이며, 필드 타임아웃들(Timeouts)은 플로우가 소멸되는 최대 시간을 나타낸다. 필드 쿠키(Cookie)는 컨트롤러에 의해 사용되는 오페이크(opaque) 데이터 값을 나타내며, 필드 플래그들(Flags)은 플로우 항목을 관리하는 방법을 변경하는 필드이다.

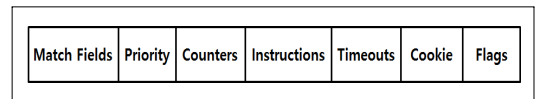


Fig. 2. Structure of Flow Entries in a Flow Table

오픈플로우 스위치는 큐잉 기법을 통해 제한된 QoS를 지원한다. 스위치에는 한개 이상의 큐가 출력 포트에 연결되어있다. 큐로 매핑된 패킷은 큐의 설정에 따라 처리된다. 본 논문에서는 이러한 큐잉 기법을 통해 QoS를 지원한다.

패킷이 도착했을 때 수행되는 동작과정은 그림 3에서와 같다. 0번 테이블부터 시작하여 매칭되는 테이블이 있으면 카운터를 갱신하고 명령어 집합(instruction set)을 실행한다. 명령어 집합의 실행은 액션 집합, 패킷 헤더, 매치 필드와 파이프라인 필드를 갱신하고, 필요시 패킷을 복제하는 작업이다. 오른쪽 마름모 그림은 파이프라인 처리가 있다면 m번 테이블로 가서 위의 과정을 반복 실행함을 의미한다. 파이프라인 처리가 없다면 액션 집합을 실행한다. 액션 집합 실행은 패킷 헤더, 매치 필드와 파이프라인 필드를 갱신하는 작업이다. 만일 그룹 액션이 있으면 해당 그룹에 대한 액션 집합을 실행하고, 만일 출력 액션이 있으면 출력 액션을 실행한다.

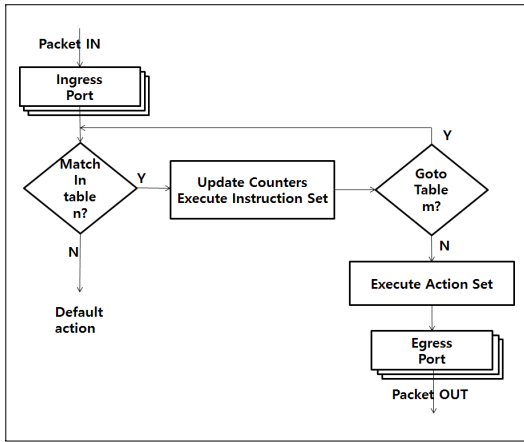


Fig. 3. Process of Packet in Openflow Switch

3.2 제안기법의 설계원리

본 논문의 제안 기법에서는 모든 매시 라우터가 오픈플로우 프로토콜 및 MPLS(Multi-Protocol Label Switching) 기법과 RSVP를 지원하며 QoS 큐잉을 지원한다고 가정한다. 본 논문에서 제안하는 기법을 위한 설계 원리는 다음과 같다.

- 1) 오픈플로우 프로토콜에서 QoS 기법을 우선적으로 적용하도록 한다.
- 2) 차별 서비스(diffserv)와 통합 서비스(intserv)를 분리해서 설계한다.
- 3) 차별 서비스는 MPLS 기법을 사용하여 패킷에 대한 라벨 스위칭을 수행하여 QoS를 제공한다.
- 4) 통합 서비스는 RSVP 기법을 이용하여 플로우에 대한 자원예약을 통해 QoS 기법을 지원한다.

5) QoS 지원 기법은 QoS 경로를 검색할 때 SDN 컨트롤러에 유지하는 정보를 이용하여 특정 QoS 메트릭에 최적의 경로를 구한다.

오픈플로우 프로토콜은 많은 네트워크 프로그래밍의 요구사항을 수용하여 너무 복잡하다는 단점이 있다. 제안기법은 오픈플로우에 QoS 기법을 우선적으로 적용하도록 수정한다. 또한, 제안기법은 QoS의 두 가지 유형을 분리함으로써 단순화한다. 차별 서비스 기법은 트래픽을 여러 개의 포워딩 클래스로 분리하는 기법이므로 MPLS의 라벨을 이용하여 FEC(Forward Equivalent Class) 기법을 사용한다[10]. 통합 서비스 기법은 플로우 단위로 처리되므로 플로우 설정을 위해 RSVP 기법을 사용하여 자원 예약을 수행한다. 마지막으로 네트워크 토폴로지에 대한 정보를 중앙 집중적으로 QoS 그래프로 유지 관리하여 QoS 경로 검색 시에 비용 효율적이며 빠르게 최적화된 QoS 경로를 검색하여 사용한다.

제안 기법을 위해 오픈플로우 프로토콜에서 QoS를 우선 적용하도록 시스템을 설계한다. QoS를 지원하기 위한 기법은 크게 오픈플로우 테이블 중 첫 번째 테이블을 이용하는 기법과 큐잉 기법이 있다.

3.3 차별 서비스를 위한 일반적인 구조

차별 서비스를 위한 QoS 지원 기법의 일반적인 구조는 그림 4에서 보이는 바와 같다. 그림 4에서 보이는 바와 같이, 차별 서비스를 위한 기법은 MPLS 기법을 우선 적용하는 것으로 경계 단에서 레이블링(labeling)을 위해 오픈플로우 컨트롤러가 스위치들의 QoS 정보를 그래프로 유지하여 QoS 그래프를 이용하여 QoS 경로로 사용될 LSP(Label Switched Path)를 구하고 스위치와의 통신을 통해 LDP(Label Distributed Protocol)을 수행한다. LDP를 수행한 후에는 LSP 경로에 있는 모든 라우터들이 0번 테이블에 QoS를 지원하기 위한 정보가 설정되어 레이블 스위칭을 수행한다. IP 계층에서도 QoS를 위해 IP 헤더에 DSCP (Differentiated Services Code Point)가 있으며, 시그너링이 필요 없고 플로우마다 상태정보를 유지할 필요가 없어서 통합 서비스 보다는 단순하고 확장성이 있는 기법이 있다. 하지만, IP 헤더를 있는 봐야 하는 데에서 L3 계층에서 라우팅을 수행하는 것이며, MPLS는 L2 계층에서 수행되는 스위칭으로 IP 계층 라우팅보다 빠르다. MPLS 라우터는 MPLS 헤더에 있는 정보를 이용하여 클래스 기반 스위칭을 하며 이를 통해

차별 서비스를 지원할 수 있다. 즉, MPLS 헤더에 있는 CoS/Exp 필드나 Label 필드를 통해 큐잉 처리를 결정한다.

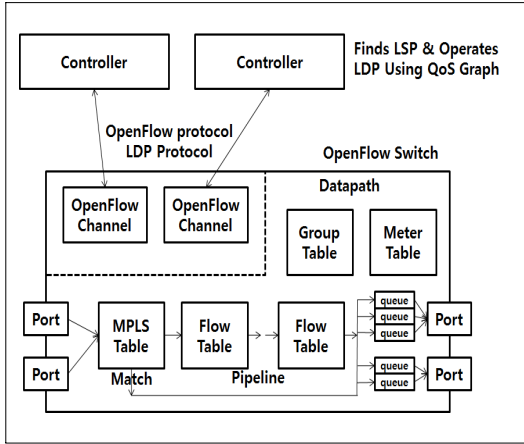


Fig. 4. Design for DiffServ using Openflow Protocol

제안 기법에서는 SDN 컨트롤러에 오픈플로우 스위치들에 대한 QoS 그래프를 유지한다. 이 QoS 그래프를 통해 LSP를 구한 후 LSP에 해당하는 오픈플로우 스위치들에 LDP를 동작시켜 레이블 스위칭 기능을 탑재한 후, 이후 패킷 처리에서는 레이블 스위칭이 가능하도록 설정한다. 차별 서비스를 위한 오픈플로우 스위치의 구성으로 첫 번째 테이블을 이용하여 레이블 스위칭을 수행하여 빠르게 처리하도록 하며 큐잉 기법을 통해 클래스 기반 라우팅을 수행한다. 이 과정에서 LDP는 기존 방식처럼 LDP를 LSP 상에 있는 스위치들이 수행하는 방법이 있을 수 있으며, 컨트롤러가 직접 LSP 상에 있는 노드들에게 설정하는 방법이 있다. 본 논문에서는 모든 라우터가 오픈플로우를 수행하므로 SDN 컨트롤러에 의해 수행한다고 가정한다. 이러한 방식은 단순하지만 컨트롤러와 LDP를 수행하는 스위치들 간의 통신이 필요하며 스위치들 간의 LDP 메시지 전송과 비교했을 때 메시지 양이 많아질 수 있는 단점이 있다.

3.4 통합 서비스를 위한 일반적인 구조

통합 서비스를 위해서 제안기법은 RSVP와 자원상태 모니터링을 사용한다. 자원상태 정보 모니터링을 통해 자원의 사용량을 측정하여 컨트롤러에게 보고하면, 컨트롤러는 그 상태 정보를 전체 네트워크 그래프로 유지하고 있다가, 자원 사용 요청이 올 경우 RSVP를 컨트롤러

에 있는 전체 네트워크 그래프에 수행하여 가능한 최적의 경로를 찾고 해당 경로 상에 있는 오픈플로우 스위치에게 자원예약 및 플로우에 대한 처리를 설정한다. 통합 서비스 기법을 위한 오픈플로우 스위치의 구성은 그림 5에서 보이는 바와 같다.

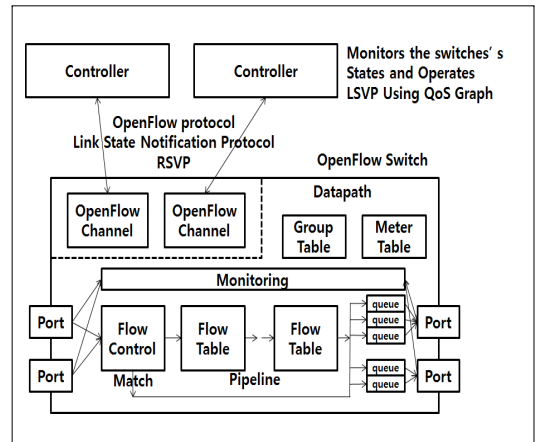


Fig. 5. Design of IntServ using Openflow Protocol

그림 5에서 보이는 바와 같이, RSVP나 LDP는 SDN 컨트롤러에 의해 중앙 집중적으로 처리될 수 있으며 설정을 위해 컨트롤러와 경로에 있는 스위치들과의 통신이 필요하다. 하지만 중앙 집중적으로 처리할 수 있어서 QoS 측면에서 비용 효율적으로 최적화하는 것이 가능하다. 즉, 제안기법은 컨트롤러에 의해 중앙 집중적으로 LSP 경로나 RSVP 경로를 검색함으로써 다른 기법에 비해 최적화된 경로를 찾을 수 있다. 무선 메시 네트워크를 그래프로 표현하여 유지하고 있으므로 그래프를 통해 최적의 경로를 찾을 수 있으며 경로를 유지함으로써 관리를 용이하게 할 수 있다.

4. 타당성 분석

4.1 표기

제안기법의 QoS 그래프에서 사용되는 표현은 표 1과 같다.

Table 1. Notations in QoS Graph G (V, E)

Notation	Description
P_{nm}	simple path is sequence($v_0, e_1, v_1, \dots, e_k, v_k$), where $v_0 = m, v_k = n, e_j = (v_{j-1}, v_j), v_i \in V, e_i \in E$
BW_{e_i}	Bandwidth of edge e_i
AD_{e_i}	Average link delay of e_i
AD_{v_i}	Average queueing delay of v_i
UD_{e_i}	Upper Delay time of e_i
UD_{v_i}	Upper Delay time of v_i
LD_{e_i}	Lower Delay time of e_i
LD_{v_i}	Lower Delay time of v_i

표 1에서 보이는 바와 같이, QoS 그래프 상에서는 에지(edge)에 전송 지연 시간 및 대역폭을 유지하고 있으며 노드(vertex)에는 큐잉 지연 시간을 유지하고 있다.

4.2 QoS 메트릭

제안기법의 QoS 메트릭(metric)으로 중요한 지연시간(delay time), 대역폭(bandwidth) 및 지터(jitter) 값은 표 2에서와 같이 표현할 수 있다. $MMBW_{nm}$ 과 $MIND_{nm}$ 은 각각 노드 m과 노드 n사이의 최대 가용 대역폭과 최소 지연시간을 의미한다. $MINJ_{nm}$ 은 노드 m과 노드 n사이의 최소 지터 값으로 노드 m과 노드 n사이의 경로들 중에서 최대 지연시간과 최소 지연시간의 차이가 가장 작은 값이다.

표 2에서 보이는 바와 같이, 지연 시간과 지터는 큐잉 시간과 노드간의 전송 지연 시간과 관련되지만, 지연 시간은 상대적으로 시간이 적게 걸리는 큐잉 지연보다는 전송 지연 시간에 주로 영향을 받고, 지터 값은 같은 경로를 이용할 경우 경로상의 지연 시간은 거의 유사하므로 노드에서 수행되는 큐잉 지연시간에 의해 결정될 것으로 파악된다. 이러한 메트릭을 통해 그래프 알고리즘을 통해 최적의 QoS 경로를 찾을 수 있다. $MIND_{nm}$ 인 경우는 다익스트라(Dijkstra)의 소스에서 목적지까지의 최소경로 검색 알고리즘을 통해 $O(|V|^2)$ 시간 복잡도 안에 QoS 경로를 찾을 수 있다[11]. 다른 메트릭도 최소 신장트리를 찾는 프림(Prim) 알고리즘과 유사한 그래프 알고리즘으로 $O(|V|^2)$ 시간 복잡도 안에 찾을 수 있을 것이다.

Table 2. QoS Metrics in QoS Graph G (V, E)

QoS Metric	Description
$MMBW_{nm}$	Maximum available Bandwidth between m and n For all $p_k \in P_{nm}$, $MAX \{ MINBW_{p_k} \mid MINBW_{p_k} = \min(BW_{e_i}), \text{ for all } e_i \in p_k \}$
$MIND_{nm}$	Minimum of average delay times between m and n For all $p_k \in P_{nm}$, $MIN \left\{ D_{p_k} \mid D_{p_k} = \sum_{v_i \in V_{p_k}} AD_{v_i} + \sum_{e_j \in E_{p_k}} AD_{e_j} \right\}$ where V_{p_k} and E_{p_k} are the sets of vertices and edges in p_k respectively
$MINJ_{nm}$	Jitter value between m and n, Minimum of differences between Upper Delay time and Lower Delay time of $p_k \in P_{nm}$ For all $p_k \in P_{nm}$, $MIN \{ U_{p_k} - L_{p_k} \}$ where $U_{p_k} = \sum_{v_i \in V_{p_k}} UD_{v_i} + \sum_{e_j \in E_{p_k}} UD_{e_j}$, $L_{p_k} = \sum_{v_i \in V_{p_k}} LD_{v_i} + \sum_{e_j \in E_{p_k}} LD_{e_j}$, V_{p_k} and E_{p_k} are the sets of vertices and edges in p_k respectively

4.3 제안기법의 타당성 분석

제안기법은 다음과 같은 장점을 가진다.

- 1) QoS 그래프를 이용한 최적 QoS 경로 검색
- 2) 중앙집중처리로 인해 최소 비용 경로 검색
- 3) QoS 그래프와 모니터링을 통해 QoS 경로 유지 및 관리 용이

표 3에서 보이는 바와 같이 제안기법은 QoS 경로를 찾는 시간이나 메시지 오버헤드는 기존 기법에 비해 좋다. 즉, QoS 경로를 찾는 데 사용되는 메시지 오버헤드도 없으며, 최소지연시간인 $MIND_{nm}$ 을 찾는 QoS 경로 탐색 시간도 $O(|V|^2)$ 의 CPU 시간만 필요하다. 또한, QoS 경로를 관리하는 것도 오픈플로우 프로토콜을 사용하면 단순화 된다.

Table 3. Comparison between Schemes

	our proposed scheme	pre-existing schemes
Message overhead of finding QoS path	X	Message flooding across the entire network
Time of finding QoS path	$O(V ^2)$ CPU time	Round trip time of QoS path
Overhead of Maintaining QoS graph	Message overhead between Openflow switch and SDN controller	X
Management of QoS path	simple	difficult

QoS 그래프를 유지 관리하기 위해서 토폴로지의 변화가 있을 때마다 지속적으로 오픈플로우 스위치와 SDN 컨트롤러와의 메시지 교환이 발생한다. 무선 메시 네트워크가 급변하는 토폴로지가 아닌 경우에는 제안 기법이 충분히 타당한 기법이 된다.

5. 결론

본 논문에서는 오픈플로우 프로토콜을 이용하여 무선 메시 네트워크 환경에서 QoS를 제공하는 기법을 제안했다. 제안 기법은 컨트롤러에 의해 중앙 집중적으로 최적의 QoS 경로를 찾을 수 있으며 오픈플로우 프로토콜을 이용하여 설정 및 유지할 수 있다. 제안 기법을 위해 오픈플로우 스위치들과 컨트롤러와의 메시지 교환이 필요하지만 QoS 목표를 달성하는 장점을 비해 네트워크 위상의 변화가 적은 네트워크인 경우 상대적으로 작은 오버헤드를 갖는다.

향후 연구 과제로는 제안 기법을 실제적으로 검증하기 위해서 메시 네트워크 환경에서 시뮬레이션을 통한 QoS 성능 평가를 수행하는 것이며, 실제 제안 기법을 오픈플로우 스위치에 구현하기 위해 필요한 링크 상태 모니터링 기법 및 SDN 컨트롤러와의 통신 기법에 대한 연구가 필요하다.

References

- [1] S. Basagni, M. Conti, S. Giordano, and I. Stojmenovic, *Mobile Ad Hoc Networking: Cutting Edge Directions*, 2nd ed, pp. 275-276, John Wiley & Sons, 2013. DOI: <https://doi.org/10.1002/9781118511305>
- [2] S. Azodolmolky, *Software Defined Networking with OpenFlow*, pp. 1-2, Packt Publishing, 2013.
- [3] F Hu, *Network Innovation through OpenFlow with SDN: Principles and Design*, pp. 42, 251-258 CRC Press, 2014. DOI: <https://doi.org/10.1201/b16521>
- [4] G. Ash et al., *Network Quality of Service: Know It All*, pp. 1-10, Morgan Kaufmann Publishers, 2009.
- [5] A. Mankin et al., "Resource reservation protocol (RSVP)", IETF, RFC 2208, Sept. 1997.
- [6] H. Huang, P. Li, S. Guo, W. Zhuang, "Software-defined wireless mesh networks: architecture and traffic orchestration", *IEEE Network*, vol. 29, no. 4, pp. 24-30, July-August, 2015. DOI: <https://doi.org/10.1109/MNET.2015.7166187>

- [7] S. Salsano, G. Siracusanano, A. Detti, C. Pisa, P. L. Ventre, N. Blefari-Melazzi, "Controller selection in a Wireless Mesh SDN under network partitioning and merging scenarios", submitted paper, 2014, Available From: <http://arxiv.org/abs/1406.2470>.
- [8] A. Detti, C. Pisa, S. Salsano, N. Blefari-Melazzi, "Wireless Mesh Software Defined Networks (wmSDN)", *IEEE 9th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, 2013. DOI: <https://doi.org/10.1109/wimob.2013.6673345>
- [9] Open Networking Foundation, *OpenFlow Switch Specification 1.5.1*, 2014, Available From: <https://www.opennetworking.org/images/openflow-switch-v1.5.1.pdf>
- [10] U. Black, *MPLS and Label Switching Networks*, Prentice-Hall, 2000.
- [11] E. Horowitz, S. Sahni, D. Mehta, *Fundamentals of Data Structures in C++*, 2nd ed., Silicon Valley Publishers Group, 2006.

강 용 혁(Yong-Hyeog Kang)

[정회원]



- 1996년 2월 : 성균관대학교 정보공학과(공학사)
- 1998년 2월 : 성균관대학교 정보공학과(공학석사)
- 2003년 8월 : 성균관대학교 전기전자및컴퓨터공학과(공학박사)
- 2003년 3월 ~ 현재 : 극동대학교 글로벌경영학과 교수

<관심분야>

보안컴퓨팅, 사물인터넷, 클라우드컴퓨팅

김 문 정(Moon Jeong Kim)

[정회원]



- 1998년 2월 : 성균관대학교 정보공학과(공학사)
- 2000년 2월 : 성균관대학교 전기전자및컴퓨터공학과(공학석사)
- 2005년 2월 : 성균관대학교 전기전자및컴퓨터공학과(공학박사)
- 2006년 7월 ~ 2007년 11월 : 고려대학교 정보보호대학원 연구교수
- 2007년 12월 ~ 2009년 2월 : 성균관대학교 정보통신공학부 연구교수
- 2009년 3월 ~ 현재 : 유원대학교 스마트IT학과 교수

<관심분야>

이동컴퓨팅, 무선에드-혹네트워크, 유비쿼터스컴퓨팅

김 수 선(Su-Sun Kim)

[정회원]



- 1984년 2월 : 서울과학기술대학교 컴퓨터공학과(공학사)
- 1986년 8월 : 한양대학교 산업대학원 컴퓨터공학과(공학석사)
- 1997년 2월 : 아주대학교 대학원 컴퓨터공학과(공학박사)
- 1991년 3월 ~ 현재 : 한양여자대학교 컴퓨터정보과 교수

<관심분야>

이동컴퓨팅, 모바일시스템, 멀티미디어, QoS