

인공지능 기반 네트워크 정책 도구 학습을 위한 KOREN Data추출 및 RouteNet 적용

조현준, 정혜선, 김경백

전남대학교

limdugmin2@gmail.com, albaneo0724@gmail.com, kyungbaekkim@jnu.ac.kr

Applying KOREN Data extraction and RouteNet for training artificial intelligence-based network policy tools

Hyeonjun Jo, Hyeeseon Jeong, Kyungbeak Kim

Chonnam Univ

요약

정부가 비대면 산업을 육성하며 사회간접자본을 디지털화하는 '디지털 뉴딜' 정책을 발표하면서 5G 융복합 사업에 대한 수요와 산업 현장의 스마트화를 시도하는 기업이 늘어나고 있다. 스마트화가 진행될수록 사용자 및 서비스의 급격한 증가로 인한 네트워크 슬라이스 정책 결정의 복잡도는 점점 더 상승하고 있다. 이를 효율적으로 해결하기 위해 인공지능 기반의 네트워크 정책 결정 도구를 사용하고 있다. 하지만, 인공지능 기반의 네트워크 정책 결정 도구 학습을 위해 네트워크 중단 간 발생한 Data(Delay, Bandwidth, Packet, Drop 등) 수집 어려움이 존재한다. 그래서 본 논문은 인공지능 기반의 네트워크 정책 결정 도구 학습에 필요한 네트워크 중단 간 발생한 Data(Delay, Bandwidth, Packet 수, Drop 수)를 네트워크 시뮬레이션 OMNeT++ 이용하여 Data 추출 방법을 설명하며, OMNeT++ 환경에서의 초 연결형 지능형 연구 개발망(KOREN) 네트워크를 구성하여 인공지능 기반의 네트워크 정책 결정 도구 학습을 위한 Data(Delay, Bandwidth, Packet 수, Drop 수) 추출 후, 선행 연구된 인공지능 기반의 네트워크 정책 결정 도구에 적용하여 초 연결형 지능형 연구 개발망(KOREN) 네트워크상의 중단 간 Delay 예측 후 OMNeT++ 환경에서 중단 간 Delay 비교를 수행하였다.

I. 서론

5세대에서는 서비스 유형에 따라서 각각 다른 기능과 구성을 갖는 전용의 네트워크가 필요하다.[1] 또한 새로운 문화 '언택트'가 탄생하면서 네트워크를 이용하는 사용자 및 서비스 증가는 급속도로 증가하고 있다.[2] 증가된 사용자 및 서비스 때문에 사용하는 네트워크 속도 저하가 발생할 수 있다. 속도 저하가 발생하거나 장애가 생길 경우 사용자 및 서비스 기업에게 많은 치명적인 결과가 발생할 수 있다.

물리적으로 하나의 네트워크를 통해 Device, Access, Transport, Core를 포함하여 중단 간 논리적으로 분리된 네트워크를 만들어 서로 다른 특성을 갖는 다양한 서비스들에 대하여 해당 서비스를 특화된 전용 네트워크를 제공해 주는 네트워크 슬라이스를 효율적이며 효과적으로 정책을 판단하는 인공지능 기반의 네트워크 정책 결정 도구가 필요하다.[3] 인공지능 기반의 네트워크 정책 결정 도구를 학습하기 위해서는 네트워크 중단 간 발생하는 네트워크 핵심성과지표[4]인 Data(Delay, Bandwidth, Packet 수, Drop 수 등)가 필요하다.[5] 하지만 네트워크 중단 간 발생한 실제 Data를 얻는 방법은 시간 및 비용이 많이 걸리는 단점이 존재하여, 현실적으로 어렵다. 이에 네트워크 시뮬레이터인 OMNeT++를 이용하여 실제 네트워크를 구성할 수 있으며, 인공지능 기반의 네트워크 정책 결정 도구 학습에 필요한 Data(Delay, Bandwidth, Packet 수, Drop 수)를 얻을 수 있다.[6]

초 연결 지능형 연구개발망(KOREN)[7] 이란 전국 10개 대도시 지역 서울, 수원, 판교, 대전, 전주, 광주, 대구, 부산, 제주, 춘천을 10Gbps~360Gbps로 연결하는 백본망을 구축 운영하여 대용량 트래픽 전송이 가능한 국내 유일의 통합연구 시험망이다.

인공지능 기반의 네트워크 정책 결정 도구인 Graph Neural Network(GNN) 모델 기반의 RouteNet은[8] 네트워크 Topology 상에서 Routing table과 네트워크 핵심성과지표 중 Delay, Bandwidth, Packet, Drop 학습하고 Routing table에 따른 Delay를 예측한다. 추후 본 연구에서의 End to End Delay 예측을 수행할 RoutenNet은 언급한 Graph Neural Network(GNN) 모델 기반의 RoutenNet을 통해 KOREN topology 상의 End to End delay를 예측한다.

본 논문에서는 OMNeT++환경에서의 인공지능 기반의 네트워크 정책 결정 도구를 학습하기 위한 네트워크 핵심성과지표 Data 중 Delay, Bandwidth, Packet 수, Drop 수 추출방법을 제안한다. 제안된 추출방법의 Data를 사용하여 KOREN topology 상 End to End Delay를 예측할 수 있다. 제안된 Data 추출 방법의 유효성을 검증하기 위해 RouteNet End to End Delay 예측 값과, OMNeT++환경에서의 End to End Delay 값을 비교한다.

II. 관련 연구

Graph Neural Network(GNN) 기반의 RouteNet은 네트워크 상에서 End to End 네트워크 핵심성과지표 중 Delay, Jitter를 예측하는데 사용할 수 있다. 특히 네트워크 Topology, Routing table 및 입력 트래픽 간의 복잡한 관계를 학습하고 모델링이 가능하여 매우 높은 정확도로 End to End Delay, Jitter 예측이 가능하다[9].

그림 1은 (A)NSFNET topology (B)모델의 성능을 알아 보기 위한 실험 결과이며, OMNeT++ 환경 NSFNET topology 상에서 발생한 네트워크 핵심성과지표 중 Delay, Bandwidth, Packet 수, Drop 수[10]를 학습

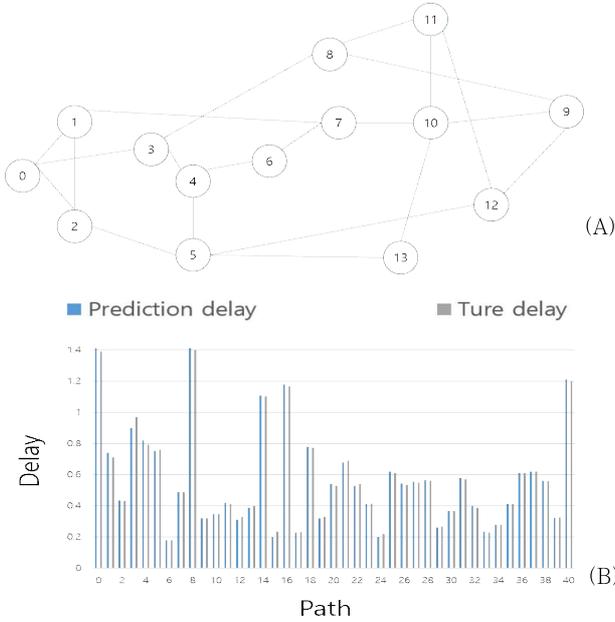


그림 1. (A) NSFNET topology (B) Delay 값 비교

후 End to End Path Delay 예측 값과 NSFNET 네트워크의 실제 측정 Delay 값을 비교하였다. 실험 결과 RouteNet의 예측 값이 네트워크 실제 측정값과 약 $1 * 10^{-2}$ 이하 오차범위가 발생하였다.

OMNeT++는 C++ 기반으로 개발한 객체지향 이벤트 기반 시뮬레이터 엔진이며, 통신 프로토콜, 컴퓨터 네트워크, 멀티프로세서, 분산시스템 등 다양한 형태의 시뮬레이션을 위해 사용되는 시뮬레이터이다.[11] 또한 매우 큰 크기의 시뮬레이션을 빠른 시간에 수행할 수 있도록 메시지 전달 인터페이스(MPI)를 기반으로 한 병렬 수행을 지원하는 시뮬레이터이다.

III. KOREN topology 구성

1. OMNeT++에서 KOREN 네트워크 시뮬레이션 환경

실험에는 Window OS 용 OMNeT++ 시뮬레이터가 사용되었다. 그림 2는 KOREN 네트워크 구성 및 OMNeT++에서의 구현 환경이다.

2. Delay, Bandwidth, Packet 개수, Drop 개수 추출

KOREN topology는 10개의 Node로 구성되어 있으며, 각 Node 별 링크의 Bandwidth는 10Gbps, 100Gbps, 110Gbps, 360Gbps 총 12링크로 구성되어 있다. 하지만 링크가 1씩 존재한 Node는 링크가 파손 및 손실되면 통신할 수 없는 문제점 때문에, 기존의 KOREN topology에서 춘천과 대구, 대구와 부산, 전주와 제주 세 군데의 링크 10Gbps, 100Gbps를 추가하여 시뮬레이션 환경을 구성하였다.

시뮬레이션은 Routing table을 참조하여 시뮬레이션이 동작하며, 그림 3는 OMNeT++에서 인공지능 기반의 네트워크 정책 결정 도구를 학습하기 위한 Data 추출 동작 순서를 도식화하였다. End to End Bandwidth, Delay, packet 수, Drop 수를 얻기 위해 시뮬레이션 시간 동안 각 Node App(Node의 개수)에서 무작위로 생성된 Data packet은 최대 5.000byte 이하 모든 Node에게 전송한다, 각 Node에 존재하는 라우터에서 패킷 도착 시간 - 생성 시간, 총 생성 개수, 받은 개수를 저장 후 시뮬레이션이 끝나고, Statistic 파일에서 계산되어, 시뮬레이션 동안 발생 된 결괏값 [(1) End to End의 Bandwidth (2) End to End의 평균 Delay (3) End to End의 전송 Packet 수 (4) End to End의 Drop packet 수] 을 얻을 수 있다

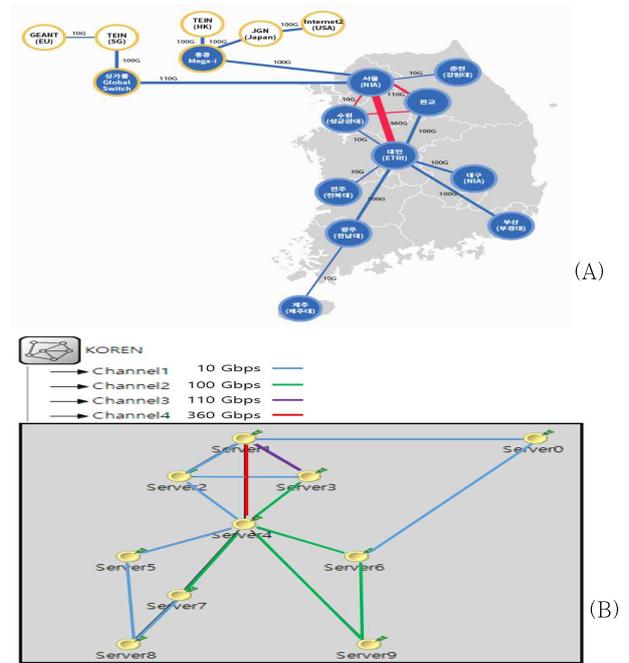


그림 2. (A) KOREN topology (B) OMNeT++ KOREN topology

3. KOREN 네트워크 DataSet

추출된 Data 들은 Result_(lambda).txt 형식으로 구성하였다. Result_(lambda).txt 파일은 네트워크 시뮬레이터인 OMNeT++에서 생성된 Node 들의 End to End Bandwidth, Delay, Num Packet, Drop Packet 을 포함한다. 추출된 DataSet의 lambda(트래픽 강도)는 시뮬레이션의 각 APP(Node의 개수)의 생성되는 Data packet의 비중과 lambda 변수 [9, 12, 15] 값을 곱해 최종 lambda 값이 설정되었다.

lambda 값 9, 12, 15를 설정한 이유는 동일 Routing table을 사용하지만 발생하는 트래픽의 비율을 다르게 하여, 큐잉 딜레이 발생하는 비율[15%, 25%, 50%] 설정하기 위해 9, 12, 15를 사용하였다. 시뮬레이션할 때 동일한 매개 변수(Routing table, lambda)를 사용하여 500번의 실험을 수행하였으며, 얻어진 Result_(lambda).txt 파일의 각 행은 Node[0-9] 10x10 행렬 Routing table에 의해 End to End에서 발행한 [(1) Bandwidth : End to End의 대역폭(kbps), (2) Delay : End to End에서 전송된 패킷에 대한 평균 Delay(s), (3) Numpacket : End to End에서 전송된 패킷 수(Number), (4) Drop : End to End에서 삭제된 패킷 수(Number)]이다.

Routing table은 Shortest Path로 구현하였으며, 총 10개의 Routing table을 사용하여 학습 및 예측에 사용할 KOREN topology DataSet을 구성하였다.

4. RouteNet 적용

학습할 때 사용한 DataSet은 Shortest Path Routing table 8개에서 발생한 [(1) End to End의 Bandwidth (2) End to End의 Delay (3) End to End의 Packet 수 (4) End to End의 Drop packet 수], DataSet를 사용하였으며, 2개의 Shortest Path Routing table에서 발생하는 End to End의 Delay를 예측한다.

5. RouteNet End to End Delay 예측 및 비교

10개의 Node에서 발생하는 End to End Delay를 예측하기 위해 표 1의 Routing table을 사용하였으며, 그림 4-5는 Routing table을 사용하여 RouteNet에 적용하여 얻은 End to End Delay 예측값 이다. 그림 6-7는 Route

Net을 사용한 10개의 Node에서 발생하는 End to End Delay 예측값과 네트워크 시뮬레이터인 OMNeT++ 상에서 얻어진 End to End Delay 비교 수행한 결과이다. 비교 수행한 결과 RouteNet의 예측 값이 OMNeT++ 측정값과 약 1.5×10^{-1} 이하 오차 범위가 발생하였다.

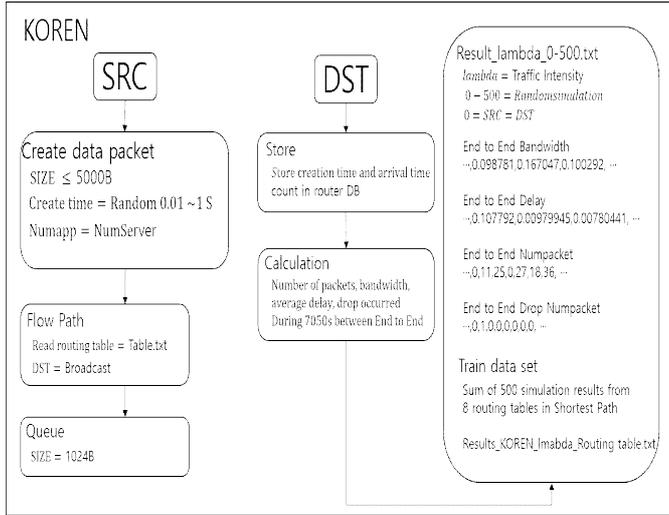


그림 3. KOREN Data추출 순서도

표 1. End to End Delay 예측 하기 위해 사용된 Routing table

Routing Table1	Routing Table2
-1.0.0.0.0.1.0.0.1,	-1.0.0.0.0.1.1.1.1.1,
0,-1.1.3.2.2.0.2.2.2,	0,-1.1.1.2.2.0.2.2.0,
0.0,-1.1.2.2.1.2.2.1,	0.0,-1.1.1.1.2.2.2.2,
0.0.1,-1.2.2.0.2.2.0,	1.0.1,-1.2.2.2.2.2.2,
1.1.0.2,-1.3.1.4.4.5,	6.1.0.2,-1.3.6.4.4.5,
0.0.0.0.0,-1.0.1.1.0,	0.0.0.0.0,-1.0.1.1.0,
1.1.0.0.0.0,-1.0.0.2,	1.1.0.0.0.0,-1.0.0.2,
0.0.0.0.0.1.0,-1.1.0,	0.0.0.0.0.1.0,-1.1.0,
1.1.1.1.1.0.1.1,-1.1,	1.1.1.1.1.0.1.1,-1.1,
1.0.0.0.0.0.1.0.0,-1	1.1.0.0.0.0.1.0.0,-1

IV. 한계점

초 연결 지능형 연구개발망(KOREN) topology 환경을 구축하여 Route Net 학습에 필요한 네트워크 핵심성과지표 중 End to End Delay, Bandwidth, Packet 수, Drop 수 추출하였으며, 추출된 Data를 이용하여 GNN 기반의 RouteNet의 적용하여, End to End Delay 예측값과 OMNeT++ End to End Delay 값을 비교 검증할 수 있었다.

비교 검증의 한계점인 실제 KOREN topology 환경에서 발생한 Delay 값 비교 검증을 수행하지 못하였으며, 학습을 위한 시뮬레이션의 Data와 실제 Data의 차이에 대한 성능 평가하지 못하였다.

V. 결론

5G 융복합 사업에 대한 수요와 산업 현장의 스마트화를 시도하는 기업이 점차 늘어남에 따라, 사용자 및 서비스가 급격하게 증가하였다. 이에 복잡도가 상승된 네트워크 슬라이스 정책 결정을 해결하기 위한 방법인 인공지능 기반의 네트워크 정책 결정 도구를 사용하여 해결하고 있다. 인공지능이 정확한 정책 결정을 하기 위해 실제 네트워크상 발생하는 핵심

성과지표 Data로 학습하는 것이 좋다. 하지만 네트워크상 실제 Data를 얻는 방법은 현실적으로 많은 애로사항이 존재한다.

본 논문은 인공지능 기반의 네트워크 정책 결정 도구인 RouteNet을 학습하기 위한, 필요 네트워크 핵심성과지표 Data중 (1) End to End의 Bandwidth (2) End to End의 Delay (3) End to End의 Packet 수 (4) End to End의 Drop packet 수를 네트워크 시뮬레이션인 OMNeT++를 통해 얻는 방법을 제안하였다. 또한 얻은 Data를 이용하여 선행 연구된 GNN기반의 RouteNet의 적용하여 End to End Delay 예측값과 OMNeT++ 환경의 End to End Delay 값을 비교 검증을 수행하였다.

본 연구를 기반으로 네트워크 시뮬레이션 도구를 이용하여 초연결 지능형 연구 개발망(KOREN)을 유연하면서 즉각적으로 서비스 구성·운영 가능하게 만들기 위해 SDN 환경의 KOREN topology 환경을 구축하여, 인공지능 기반 네트워크 정책 결정 도구를 학습하기 위한 필요한 네트워크 핵심성과지표 Data 추출 후 적용할 것이다.

ACKNOWLEDGMENT

"본 연구는 과학기술정보통신부 및 정보통신기획평가원의 대학ICT연구센터육성지원사업의 연구결과로 수행되었음"(IITP-2021-2016-0-00314*)

참고 문헌

- [1] koreaScience. 5G 망에서의 Network Slicing 요구사항 및 제공 구조. June, 2016.
- [2] Cisco. 2020 글로벌 네트워킹 트렌드 보고서. 2020.
- [3] ETRI. 네트워크와 AI 기술 동향. 2020.
- [4] Zhiyao Xie et al. "RouteNet: Routability prediction for Mixed-Size Designs Using Convolutional Neural Network". IEEE, Conference Paper, Nov, 2018.
- [5] M. Ruiz et al. "End-To-End KPI Analysis in Converged Fixed-Mobile Networks". IEEE, July, 2020.
- [6] A. Varga, "The omnet++ discrete event simulation system," in Proceedings of the European Simulation Multiconference (ESM), 2001.
- [7] KOREN. 미래네트워크선도시험망(KOREN) 이용규정, 1028. (<http://www.koren.kr/kor/>)
- [8] Krzysztof Rusek et al. "RouteNet: Leveraging Graph Neural Networks for network modeling and optimization in SDN". IEEE, Journal on, Vol.38, pp. 2260-2270. June, 2020.
- [9] Hyeseon Jeong, Seongwoong Yeom, Kyungbaek Kim, "지능형 네트워크 관리를 위한 RouteNet 분석", 2020 한국다지털콘텐츠학회.
- [10] "Knowledge-defined networking repository," <https://github.com/knowledgedefinednetworking/Papers/wiki/RouteNet:-Leveraging-GNN-for-network-modeling-and-optimization-in-SDN>, 2019.
- [11] OMNeT++ (<https://omnetpp.org/documentation/>).

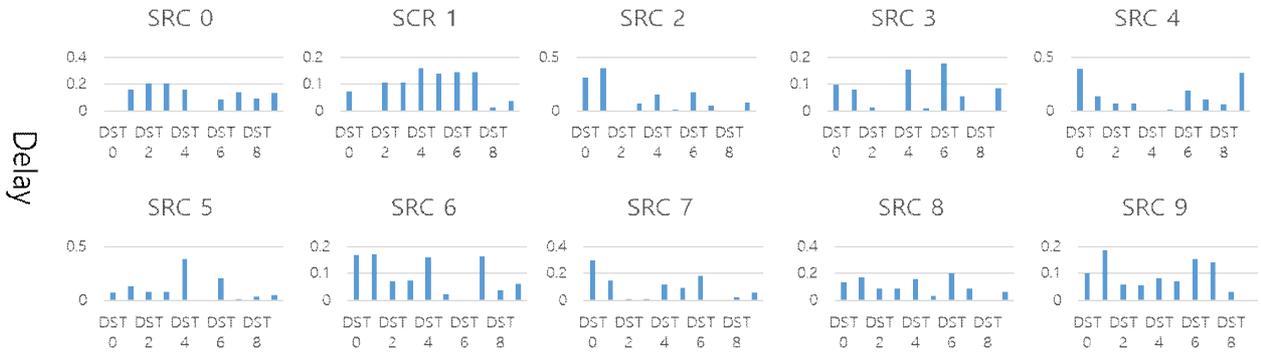


그림 4. Routing table 1 평균 Delay 예측

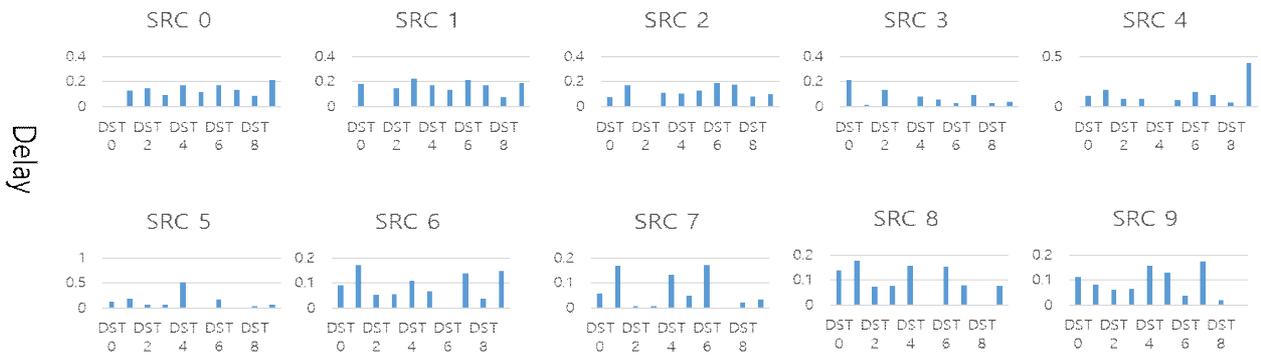


그림 5. Routing table 2 평균 Delay 예측

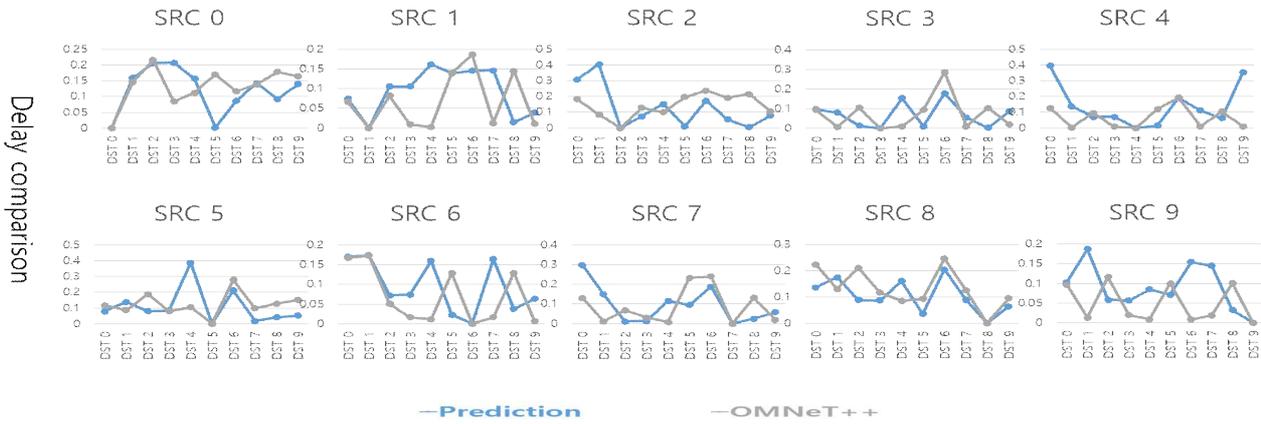


그림 6. Routing table 1 Delay 비교

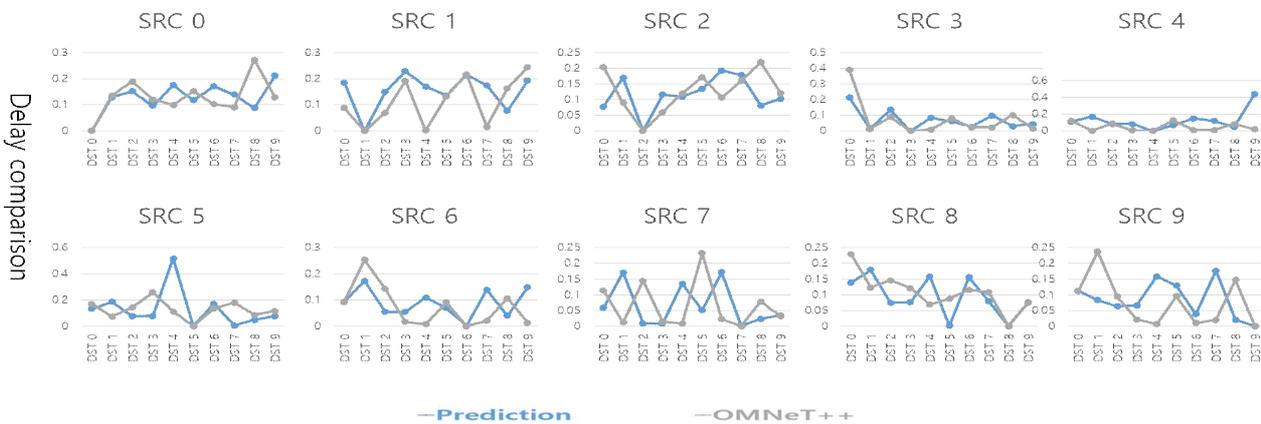


그림 7. Routing table 2 Delay 비교