

SDN 환경에서 Deep Q-Learning을 이용한 QoS 인지 동적 라우팅 기법 연구

김기태[○] 이주연[#] 홍충선*
 경희대학교 컴퓨터공학과
 {glideslope[○], lly0419[#], cshong* }@khu.ac.kr

QoS-Aware Dynamic Routing Strategy in Software-Defined Network Environment using Deep Q-Learning

Kitae Kim[○] Juyeon Lee[#] Choong Seon Hong*
 Department of Computer Science and Engineering, Kyung Hee University

요 약

최근의 증가하는 네트워크 기기와 트래픽은 기존의 하드웨어 중심의 네트워크로의 관리는 한계가 있다. 따라서 더욱 유연하게 네트워크를 관리 할 수 있는 방안이 연구되었는데 그 중 하나가 바로 소프트웨어 정의 네트워크 SDN(Software-Defined Network) 이다. SDN의 가장 큰 장점은 중앙의 컨트롤러를 이용해 다양하고 유연한 서비스를 제공 할 수 있다는 점이다. 따라서 본 논문에선 위와 같은 장점에 착안하여 어플리케이션 별로 분류 된 각각의 패킷의 Quality of Service(QoS)를 보장하기 위하여 강화학습의 종류 중 하나인 Deep Q-Learning을 이용한 동적인 라우팅 방안을 제안한다.

1. 서 론

소프트웨어 정의 네트워크(SDN)는 기존의 Control Plane 과 Data Plane이 공존했던 네트워크 구조에서 이를 분리 하여 Control Plane을 중앙으로 집중시켜 소프트웨어를 통해 Data Plane에서의 통신 장비 및 통신 제어가 가능하다. 이러한 새로운 네트워크 구조는 오늘날의 다양한 모바일 기기와 클라우드 기반의 가상화 서비스가 등장함에 따라서 SDN은 미래 인터넷 구조로 각광 받고 있으며 이유는 다음과 같다. 첫 번째로 과거와는 다른 트래픽 패턴의 변화를 보여주고 있다. 과거에는 서버-클라이언트 간의 트래픽이 주를 이뤘다면 현재는 데이터센터간의 트래픽이나 서버간의 트래픽이 폭증하고 있으며 이러한 트래픽들을 제어하기에는 기존의 서버-클라이언트의 네트워크 구조는 실시간으로 급변하는 현재의 네트워크 상태를 실시간으로 제어하기에 바람직하지 않다는 것이다.[1] 소프트웨어로 Data Plane이 제어가 가능하다면 이를 이용해 폭증하는 다양한 종류의 트래픽의 각각의 QoS(Quality of Service)를 실시간으로 보장할 수 있다.

SDN에서 Flow Rule 테이블의 업데이트는 아래와 같이 진행된다. 컨트롤러에 연결 된 스위치로 Packet이 도착 하면 스위치는 라우팅 테이블을 검색하여 Match 되는 라우팅 규칙이 있다면 그에 상응하는 Action을 통해 패킷을 포워딩 한다.반면에 Packet Miss가 일어나는 경우 Northbound Protocol인 OpenFlow를 이용해 컨트롤러에게 Packet_In 메시지를 보내 라우팅 경로를 질의 한다.[2]

Packet_In 메시지를 받은 컨트롤러는 Dijkstra 알고리즘을 이용해 최단 경로를 계산해 해당 스위치의 Flow Rule 테이블을 업데이트 한다. 하지만 이러한 Dijkstra 알고리즘은 실시간 변화되는 각 경로의 대역폭이나 링크 속도를 고려하지 못하기 때문에 대량의 트래픽이 발생하는 IoT 환경이나 실시간 스트리밍 등과 같은 QoS가 중요시 되는 서비스를 제공할 시 네트워크에 혼잡이 발생할 수 있으며 효율적인 서비스를 제공하지 못한다는 단점이 있다.

따라서, 본 논문에서는 위와 같은 문제를 해결하기 위해 SDN컨트롤러에서 스위치의 Flow Rule 테이블을 업데이트 할 때 먼저 Packet Classification을 통해 Application 별로 서비스를 분류하고 각 패킷의 요구에 맞는 경로를 강화학습의 종류 중 하나인 Deep Q-Learning을 이용해 찾아주는 방법을 제안한다.

2. 관련연구

2.1 Deep Q-Learning

Deep Q-Learning은 Q-Learning에 기반한 강화학습(Reinforcement Learning)의 종류 중 하나이다.

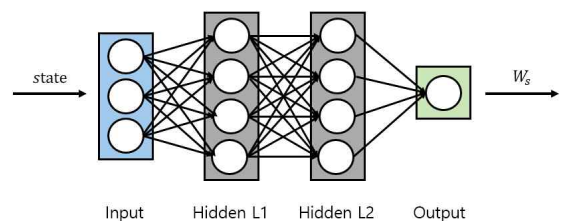


그림1. Deep Q-Network

이 논문은 2017년도 정부(과학기술정보통신부)의 재원으로 정보통신기술진흥센터의 지원을 받아 수행된 연구임(No.2015-0-00567, 유무선 통합 네트워크에서 접속 방식에 독립적인 차세대 네트워킹 기술 개발)

*Dr. CS Hong is the corresponding author

Deep Q-Learning은 Q-Learning과 다르게 테이블이 아닌 신경망(Neural network)을 이용하여 신경망의 출력이 Target Value와의 차이가 최소가 되도록(수식1.) 가중치(Weight)를 업데이트를 시키는 과정을 통해 학습한다. 이는 수 많은 상태(State)가 있을 때 적합한 모델이며 대규모 네트워크 구조에서 각 스위치의 상태 뿐 아니라 각 포트, 스위치 간 링크 상태, 대역폭을 고려하게 되면 수 많은 상태가 정의되므로 이에 적합하고 안정적인 모델을 만들어낼 수 있는 Deep Q-Learning을 사용하였다.

$$\min \sum_{t=0}^T [\hat{Q}(s_t, a_t | \theta) - (r_t + \gamma \max_a \hat{Q}(s_{t+1}, a | \bar{\theta}))]^2$$

수식1. Deep Q-Learning

2.2 Packet Classification

패킷 분류의 목적은 QoS 별로 분류 된 패킷을 통해 각각의 패킷에 적합한 최적화 된 경로를 SDN컨트롤러가 제공하기 위함이다. Packet 분류 기법으로는 각 패킷의 페이로드(Payload)를 조사하는 DPI(Deep Packet Inspection)등 이 있으나 모든 패킷의 페이로드를 검사함에 따라 빠르게 실시간 처리를 요하는 패킷에는 적합하지 않다. 따라서 본 논문에서는 기 제안하였던 결정트리(Decision Tree)기반의 패킷 분류 기법을 사용한다.[3]

3. 제안사항

3.1 Packet Classification

Deep Q-Learning을 통해 Routing 경로를 탐색하기 스위치로부터 Packet_In 메시지 형태로 Controller에 도착한 패킷은 Packet Classification 과정을 통해 Application 별로 분류된다. 본 논문에서의 분류는 아래의 표1과 같다.

Application	Category	Requirement
Youtube Live	Live Streaming	High Bandwidth
Youtube	Video	High Bandwidth
Skype	VoIP / Video	Low Latency
HTTP	Text	Default

표1. Packet Classification

미리 학습 된 결정트리를 통해 4가지의 어플리케이션이 분류가 되며 Category별로 필요한 Requirement가 정의된다. Default의 경우는 Dijkstra 알고리즘을 통해 경로를 제공한다. 이와 같이 정의된 Requirement를 통하여 강화 학습의 보상(Reward)이 계산된다.

3.2 Deep Q-Learning based Routing

이번 절에서는 앞서 분류된 각 패킷의 Requirement를 바탕으로 이러한 각 QoS를 만족시킬 수 있는 경로를 제공하는 Deep Q-Learning 알고리즘을 제안한다.

$$Reward = \begin{cases} \frac{1}{transmissiondelay} + \frac{linkspeed}{1000} & , packetclass = l \\ \frac{throughput + bandwidth}{1000} & , packetclass = b \end{cases}$$

수식2. Reward Function

수식2는 Packet Requirement에 따른 Reward Function으로 Low latency 를 요하는 패킷은 각 스위치포트의 Transmission Delay와 각 링크 간 Link Speed를 이용해 계산되며 고 대역폭을 요하는 패킷의 경우에는 링크 간 처리율과 대역폭을 통해 계산된다. Default의 경우에는 기존의 경로 탐색 알고리즘을 통해 경로를 제공받는다.

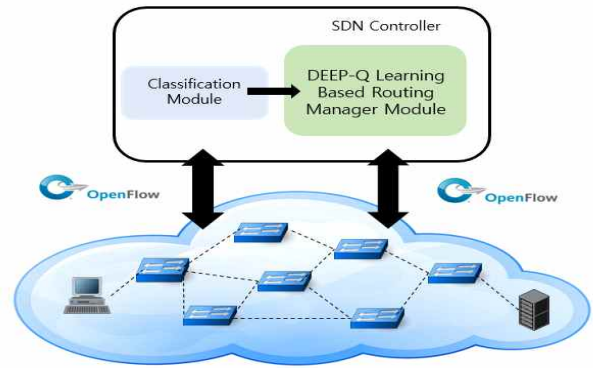


그림2. Proposed Structure

그림2는 제안한 두 개의 모듈이 설치된 SDN 컨트롤러로 구성된 네트워크 토폴로지의 예시이다. Server에서 왼쪽의 Host에게 패킷을 전달할 시 Packet_In 메시지를 통해 SDN컨트롤러로 전해지고 패킷은 2개의 모듈을 거치고 컨트롤러는 이에 맞는 경로를 경로의 스위치들에 업데이트 시킨다.

알고리즘1. Deep-Q Learning based on Packet Classification

- 1: Packet Arrived at Controller
- 2: **Class** = Classify(Packet)
- 3: Choose Q-Network for difference Class of Packet
- 4: Initialize Q-Network $\theta, \bar{\theta}, Q(\Phi(s_t), a; \theta)$
- 5: **For** Each Packet
- 6: Choose action randomly a_t
- 7: otherwise select $a_t = \text{argmax}_a Q(\Phi(s_t, a; \theta))$
- 8: Execute a_t , and observe **Reward, Next State**
- 9: Store $(\Phi_t, a_t, r_t, \Phi_{t+1})$ in **buffer**
- 10: Choose $(\Phi_t, a_t, r_t, \Phi_{t+1})$ randomly in **buffer**
- 11: Training Target Network $\bar{\theta}$ and θ
 $T = (r_t + \gamma \max_a \hat{Q}(s_{t+1}, a | \bar{\theta}))$
- 12: Compare T and $Q(\Phi(s_t), a; \theta)$
- 13: Update Weight in θ
- 14: Update $\bar{\theta} = \theta$
- 15: **Packet** choose the Route with highest **Reward**

위 의사 알고리즘은 각 패킷을 Requirement 별로 분류 후 Deep-Q Network를 학습 시킨 후 이를 바탕으로 최적

의 경로를 찾아내는 과정을 나타낸다. Reward의 값이 최대가 되는 Target 네트워크의 값과 실제 학습을 통한 Reward 값을 비교해 그 차이를 최소화할 목적으로 가중치(Weight) 업데이트를 통해 Q-Network 학습이 진행된다.[4]

4. 성능평가

본 논문에서는 Mininet을 이용해 같은 토폴로지를 구성하여 성능평가를 진행한다.

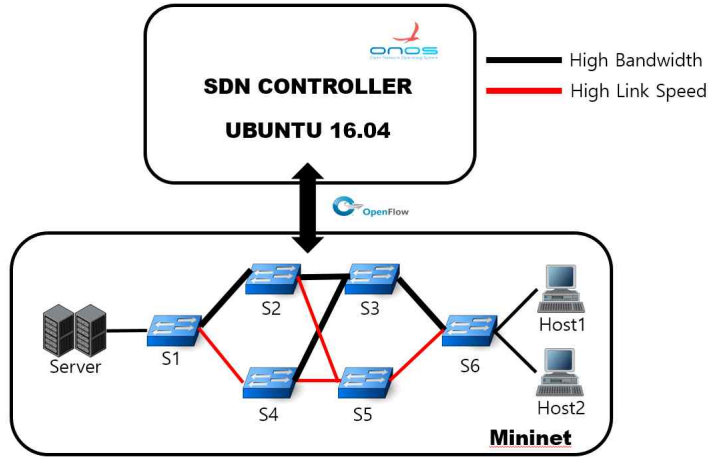


그림3. Simulation Topology

서비스에 맞는 경로 제공여부를 검증하기 위해 SDN 컨트롤러를 통해 Host들과 서버로 향하는 서로 다른 두 가지 경로1(S1-S2-S3-S6)에는 더 높은 대역폭을 할당하고 경로2(S1-S4-S5-S6)에는 빠른 링크속도를 할당해 Requirement가 각기 다른 두 개의 Youtube 패킷과 Skype 두 가지 패킷을 각각 Host1-Server, Host2-Sever로 전송 후 각 스위치를 지나는 Packet을 카운트하여 경로를 확인한다.[5]

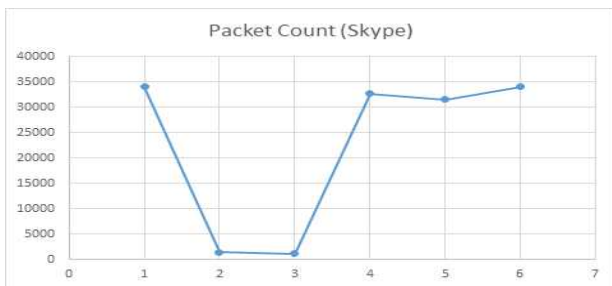


그림4. 시뮬레이션 결과(Skype)

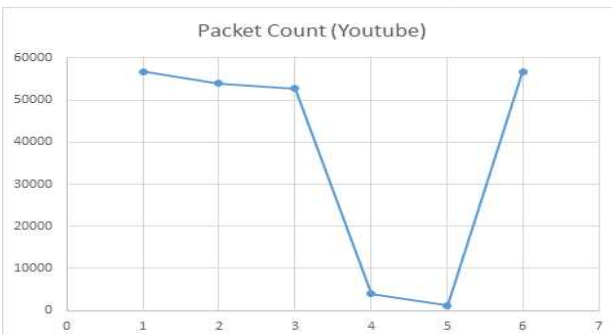


그림5. 시뮬레이션 결과(Youtube)

그림4 는 Server에서 Host1로 패킷을 Skype 패킷을 전송할 때의 각 스위치에 통과하는 Packet의 를 측정한 그래프이다. 그림을 통해 S1,S4,S5,S6의 경로로 많은 패킷이 전달되었음을 알 수 있으며 그림5는 Server에서 Host2로 Youtube 패킷을 전송할 때 각 스위치에 통과하는 Packet의 개수를 측정한 그래프이며 그 결과 S1,S2,S3,S6으로 많은 패킷이 전달되었음을 알 수 있다. 결론적으로 각 패킷에 Requirement에 맞는 경로로 많은 양의 패킷이 전달되었음을 알 수 있다. 다만 첫 번째의 경우 S2,S3 두 번째의 경우 S4,S5를 지나는 패킷이 관찰되었는데 이는 분류, 경로탐색 알고리즘의 학습 모델의 정확도가 좀 더 향상되어야 함을 나타낸다.

5. 결론 및 향후 연구

본 논문에서는 Deep Q-Learning을 이용한 애플리케이션 별 QoS Requirement에 따른 효율적인 라우팅방안을 제시했다. 시뮬레이션 결과 서로 다른 요구사항을 가진 두 패킷이 전송에 효율적인 경로를 컨트롤러로부터 할당 받아 전송이 됨을 확인하였다. 하지만 제안한 구조는 트래픽 분류의 정확도와 학습 된 Q-Network에 정확도에 큰 영향을 받을 수 있음을 알 수 있었다. 따라서 향후 연구에서는 조금 더 정확한 모델을 만들기 위해 Reward Function을 개선시키고 더 많은 노드들과 데이터를 이용해 실험을 진행할 계획이다.

참고 문헌

- [1] 유재성, 김우성, 윤찬현 “SDN/Openflow 기술동향 및 전망” ,KNOM Review, Vol. 15, No. 2 2012
- [2] SDN/NFV 포럼 공저 “오픈 소스와 오픈 스탠더드로 알아 보는 SDN과 NFV” SDN/NFV 포럼 2016
- [3] 김기태, 홍충선 “SDN 환경에서 효율적인 트래픽 분류를 위한 Feature Selection 기반 Multi Classification 기법 연구” , 2017년 한국 컴퓨터 종합 학술대회(KCC 2017).
- [4] Demis Hassabis, Shane Legg et al, “Human-level control through deep reinforcement learning” , Nature Letter 2015.
- [5] Satadal Sengupta, Harshit Gupta, Niloy Ganguly, Bivas Mitra, Pradipta De, Sandip Chakraborty, CRAWDAD dataset iitkgp/apprtraffic (v. 2015-11-26), traceset: apptraffictraces, downloaded from <http://crawdad.org/iitkgp/apprtraffic/20151126/apprtraffictraces>, <http://doi.org/10.15783/C77S3W>, Nov 2015.