SD-WAN을 위한 강화학습 기반 경로 설정 모듈 설계

홍남곽(*), 염성웅(*), 김경백(**)
(*) 전남대학교 인공지능융합학과, quachhongnam1995@gmail.com, yeomsw0421@gmail.com
(**) 전남대학교 인공지능융합학과, kyungbaekkim@jnu.ac.kr

Design of Reinforcement Learning based Flow Configuration Module for SD-WAN

Hong-Nam Quach(*), Sungwoong Yeom(*), Kyungbaek Kim(**)
(*)Chonnam National University, Department of Artificial Intelligence Convergence
(**)Chonnam National University, Department of Artificial Intelligence Convergence

요 약

In WANs, software-defined networking architectures offer the chance to build WANs with greater fault tolerance, scalability, and manageability. SD-WAN, through unified programmability, holds the promise of being a more cost-effective and simpler way to connect business divisions and corporate data centres. Many researchers have been drawn to the use of AI technology, which has been applied to many fields, and a great deal of research is underway. In particular, the reinforcement learning methodology, which received little attention except for the game field began to be followed in many areas, including the network industry. The SDN field also reflected the recent increase in research relating to reinforcement learning. Thus, we will therefore explore the ability of RL and try to apply RL to solve SD-WAN problems. In this paper, we present the idea of design is "Flow Configuration Module" to achieve low cost with enough bandwidth over an SD-WAN.

1. Introduction

Nowadays, businesses are increasingly mobile-centred, and business apps operate on several clouds; the traffic also increasing rapidly that traditional WAN architectures can not keep up with the lack of low latency, restricted security, and increased complexity that it can not adapt quickly to business requirements. Software-defined wide area network is regarded as the promising architecture of next-generation wide area network which holds the promise of being a more cost-effective and simpler way to connect business divisions and corporate data centres. Reinforcement learning (RL) has been interesting and efficiently used by many researchers in robots, video games and the area of computer networks. Implementing RL techniques in SDN routing showed an efficient to enforce routing to the dynamic network, so they can provide excellent levels of QoS while optimizing resource utilization[1]. We have tried to explore and build the module of experimenting with RL in solving a fascinating subject in SD-WAN connection selection to achieve the goal of using a low-cost network but still ensuring optimal bandwidth. We present the design of the Flow Configuration Module in this paper to achieve low costs with adequate bandwidth over an SD-WAN.

2. Background

2.1 An overview of SD-WAN

A specific SDN application that dynamically routes traffic through branches, data centers and clouds to achieve full WAN coverage is SD-WAN or software-defined wide-area network [2]. SD-WAN supports a software-oriented overlay architecture based on SDN technology over conventional WAN hardware. This simplifies the management of the network by transferring the control layer to the cloud, allowing network virtualization. SD-WAN tackles problems such as increasing bandwidth prices, lack of visibility and network control and more.

2.2 Reinforcement-learning(RL)

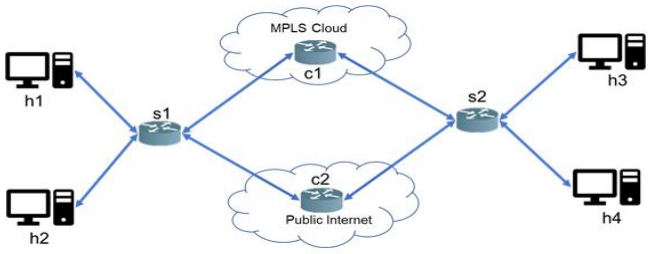
RL is a type of machine learning in which an agent learns how to make the most of the environment by experimenting with different options provided by the environment [3]. The agent learns to maximize the reward that he gets from the world by trying various actions. An agent communicates with the environment and checks various behaviors. Along with contact with the agent, the environment retains its state and goes through a different change of states and provides the reward in exchange. RL method has an objective, and through trial and error, the agent is learning to achieve the goal. By learning a set of optimal actions, the agent's goal is to maximize this incentive.

3. Configuration of RL Applied to SD-WAN

SD-WAN supports multiple transmissions and facilitates traffic load sharing across multiple WAN connections for easier WAN management. Switching traffic from the private MPLS cloud to a cheaper public internet infrastructure results in complex cost optimization. How this decision is made about when to make a path change and how often makes all the difference for an SD-WAN network. The mission of RL in this issue can be considered switching WAN connections and attempts to optimize the usage of bandwidth. This is achieved by switching to the Internet cloud once sufficient bandwidth is available and then switching back to MPLS when QoS constraints begin to fall. The aim is how to make such a link-switch such that maximum bandwidth is reached with minimal MPLS link usage the lowest cost.

We first need to design an environment for the RL. We use the Open-AI gym, a toolkit that provides agents with a sandbox to try to learn optimum behaviour for a personalized environment [4]. Any gym is a virtual environment in which an agent can try various actions and obtain a reward. We must simulate a WAN network between a so-called 'branch office' and 'headquarters' for SD-WAN. Two alternative routes exist in this simplified network, one via the dedicated MPLS circuit and the other via the public Internet. For its simplicity and suitability for SDN applications, we selected Mininet as the option [5]. We have two R1 and R2 routers reflecting the branch edge router and head quarter edge respectively in the following configuration. The central router C1 describes C2 represents the direction of the MPLS and the public Internet. The aim is to send traffic from the branch of host H1 to the headquarters of host H2. We also submit additional traffic from host N1 to N2, and this simulates the Internet-only route for other traffic.

The second, we build the observation (or state) of the environment. At any point in time 'tick', we have selected the following as the state of the experiment. Please note that these are calculated at the start of the experiment and at every 'step' where the agent applies any 'action'.



(Figure 1) The topology of experiment.

1) Current bandwidth − The amount of bandwidth that every route obtains, say MPLS or public internet,
2) Available bandwidth − This is how much Internet cloud bandwidth is available. The MPLS route is presumed to have adequate bandwidth to satisfy any demand.
3) Link Id − The connection that transports the traffic at any moment.

The third, we will déign the action between MPLS and the Internet for this agent to select. Connection 0 (Internet) and Link 1 (MPLS) are two discrete acts. For every action and its state transformation, there is no stochasticity involved. It is deterministic that one connection is shifted to another.

Next, the reward is a crucial piece of architecture that reinforcement learning algorithms rely heavily on in the manner in which the reward is constructed. It is important to design the incentive, keeping the objective of the system in mind. The target here for SD-WAN is to optimize internet path bandwidth usage, which is a cheaper choice. Here are some of the values that are taken into account when the incentive is designed:

1) SLA must be maintained at any expense, so current achieved bandwidth should always surpass the limit specified by the SLA. As bandwidth falls below the SLA limit, the incentive should be structured to punish such lapses heavily.
2) The MPLS scheme is not cost-effective so that incentives can prevent its use.
3) The flip side of the above is that the Internet circuit should be more widely used.
4) The aim is to maintain traffic flow, regardless of whatever option is made. So 'tick' should carry a minimum positive reward every time.

Finnally, experiments in reinforcement learning have the idea of an episode, which means when the experiment or the game gets over by one of the party winnings. In this experiment of SD_WAN, there is no such thing of wine. The transfer of data can go on indefinitely, but for the sake of the experiment, an artificial limit of MAX-TICKS has been designed. When the number of time 'ticks' exceeds this limit, the episode is thought to be over. There is another way the episode can end, and that is by error. Whenever the SLA bandwidth is not met for successive two-time 'ticks', it is decided that the episode has ended and an agent has to start afresh all over again.

4. Conclusion

Through the implementation of different machine learning algorithms, SD-WAN has great potential to gain, and reinforcement learning is most successful in solving some of its core issues. In this paper, we present the idea of design is "Flow Configuration Module" to achieve low cost with enough bandwidth over an SD-WAN. The future work will be to apply advanced algorithms such as Deep Reinforcement Learning or A3C to solve this SD-WAN problem.

Acknowledgment

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Science, ICT & Future Planning(NRF-2017R1A2B4012559).

참고 문헌

[1] Quach, Hong-Nam, Sungwoong Yoem, and Kyungbaek Kim. "Survey on Reinforcement Learning based Efficient Routing in SDN."

[2] Yang, Zhenjie, et al. "Software-defined wide area network (SD-WAN): Architecture, advances and opportunities." 2019 28th International Conference on Computer Communication and Networks (ICCCN). IEEE, 2019.

[3] Sutton, Richard, S., Barto, Andrew, G.: Introduction to reinforcement learning. Machine Learning, 16(1), 285--286 (2005).

[4] Zamora, Iker, et al. "Extending the openai gym for robotics: a toolkit for reinforcement learning using ros and gazebo." arXiv preprint arXiv:1608.05742 (2016).

[5] Kaur, Karamjeet, Japinder Singh, and Navtej Singh Ghumman. "Mininet as software defined networking testing platform." International Conference on Communication, Computing & Systems (ICCCS). 2014.