Efficient Resource-Aware Proactive Flow Rule Caching in Software-Defined Access Networks

Youngjun Kim
School of Computer Science and Eng.
Kyungnam University
Changwon, Korea
youngjun@kyungnam.ac.kr

Tae-Kook Kim
School of Computer Eng.
Pukyong National University
Busan, Korea
king@pknu.ac.kr

Yeunwoong Kyung
Div. of Info. & Comm. Eng.
Kongju National University
Cheonan, Korea
ywkyung@kongju.ac.kr

***Abstract*—As Software-Defined Networking (SDN) becomes increasingly critical for managing complex and dynamic network environments, efficient flow rule caching has emerged as a key challenge. Traditional reactive caching approaches introduce latency during handovers, impacting the Quality of Service (QoS) for delay-sensitive applications. Proactive caching methods address this by predicting mobile nodes' (MNs) future locations; however, they often face issues related to prediction accuracy and memory utilization. In this paper, we propose a efficient resource-aware proactive flow rule caching based on multi-agent reinforcement learning (MARL). Our approach dynamically predicts MN movement patterns, enabling the SDN controller to pre-install flow rules in a targeted and timely manner. A hierarchical multi-agent architecture is introduced to adjust caching strategies based on the mobility level of MNs, maximizing flow setup hit ratio (FSHR) while minimizing unnecessary TCAM occupancy. Proposed MARL-based caching strategy presents a scalable and efficient solution for flow management in SDN, particularly in environments requiring high mobility and low latency.**

***Index Terms*—Software-Defined Networking, Wireless communication networks, Flow Rule Caching, Quality of Service, Reinforcement Learning**

I. Introduction

With the increasing demand for flexible and efficient network architectures, Software-Defined Networking (SDN) has emerged as a transformative paradigm, enabling a software-centric approach to networking [1]. SDN separates the control plane from the forwarding plane, allowing a centralized controller to manage network functions and services with enhanced flexibility and scalability [2]. This decoupling simplifies packet processing in forwarding devices and supports network abstraction through programmable APIs, promoting innovation in application-driven network management.

One of the primary challenges in SDN is optimizing flow table management in forwarding devices. Given that each flow requires a unique entry in the flow table, efficient flow entry and lookup mechanisms are crucial for maintaining network performance. Ternary Content Addressable Memory (TCAM) is commonly employed for fast lookups; however, its high cost and limited capacity necessitate effective table management strategies [3].

This results was supported by "Regional Innovation Strategy (RIS)" through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(MOE)(2021RIS-003)

To address mobility in SDN environments, flow rule caching plays a significant role in maintaining seamless connectivity during handovers. Traditional reactive flow rule setup approaches can incur response delays as new flow rules are installed post-handover, which can impact the Quality of Service (QoS) for delay-sensitive applications [4]. In contrast, proactive flow rule caching techniques, which predict the mobile node's next location, aim to pre-install flow rules at potential handover destinations, thus reducing latency [5]. However, existing proactive strategies often rely on probabilistic models and face challenges in balancing prediction accuracy and resource utilization.

Reinforcement learning has been widely studied for optimizing network operations and enhancing various performance metrics [6]–[8]. Building on this foundation, this paper explores the application of reinforcement learning in network environments to dynamically optimize flow rule caching, enabling timely and efficient responses to mobile node movements in Software-Defined Networking (SDN). Specifically, we propose a resource-aware proactive flow rule caching mechanism based on multi-agent reinforcement learning (MARL). By leveraging multiple agents to predict mobile node movements and adjust caching strategies based on mobility patterns, our approach improves flow setup hit ratios and minimizes latency, particularly for latency-sensitive services [9]. Furthermore, the introduction of a hierarchical agent structure enables a tailored flow rule setup across multiple Access Devices (ADs), effectively balancing load and maximizing overall system efficiency.

II. Background

Flow tables are essential components in SDN, as they store flow entries that dictate how packets are processed. Each flow entry typically includes fields such as source and destination IP addresses, MAC addresses, and transport layer ports, along with associated actions and counters. Efficient management of flow tables is critical, as each flow requires a unique entry, leading to significant memory usage. TCAM is often used for fast lookups; however, TCAM is costly and has limited capacity, making efficient flow entry management essential to maintain network performance [4]. To address this, SDN controllers must efficiently manage flow rules to optimize table utilization, especially in large-scale or dynamic networks where the number of flows fluctuates rapidly.

Reactive flow rule setup is a traditional approach in SDN, where flow rules are installed on-demand. When a packet arrives that does not match any existing flow entry, the switch sends a packet-in message to the controller, which then calculates and installs the necessary flow rule. Although this method minimizes the memory usage of unused rules, it introduces latency, as each new flow requires a round-trip interaction with the controller. This latency is particularly problematic for delay-sensitive applications and mobile nodes (MNs) that frequently change their attachment points due to movement [5], [10].

Proactive caching, on the other hand, aims to pre-install flow rules based on predicted mobility patterns of MNs, thereby reducing latency during handovers. By caching flow rules in advance, the controller reduces response delays and improves QoS for latency-sensitive applications. However, proactive caching faces challenges related to prediction accuracy. If predictions are inaccurate, unused flow rules may occupy valuable TCAM space, leading to inefficiencies. Traditional proactive methods rely on probabilistic models, such as the Markov predictor, which can introduce errors, leading to unnecessary caching in incorrect ADs [9].

III. Proposed Scheme

To address the limitations of traditional reactive and proactive caching methods, we propose a timely proactive flow rule caching mechanism based on MARL. In this scheme, multiple agents represent the states of MNs and predict their movement patterns, enabling dynamic and adaptive caching strategies. Each agent, representing an MN, monitors the previous AD and time spent there to estimate the MN's next likely destination. By leveraging this information, the controller can install flow rules just in time for the MN's arrival, reducing unnecessary TCAM occupation while ensuring low-latency handovers.

Additionally, we introduce a hierarchical agent architecture, wherein a higher-level controller agent manages the mobility levels of MNs. Based on the mobility level, the controller adjusts the proactive caching strategy, determining whether to cache rules in one or multiple ADs. For highly mobile nodes, rules are pre-installed in multiple nearby ADs to maximize the flow setup hit ratio (FSHR), ensuring that MNs can quickly connect to new ADs without delay. For less mobile nodes, caching is restricted to the most probable AD to conserve TCAM resources.

Maximizing FSHR is essential to reduce latency in SDN, especially in mobile environments. The proposed scheme enhances FSHR by setting up flow rules on multiple ADs, taking into account the MN's mobility patterns. Timely proactive scheme, which sets up flow rules by considering not only the expected locations but also the mobility patterns, including movement timings, further optimizes FSHR. By leveraging MARL, the system dynamically predicts and adjusts flow rule placements, maximizing hit ratios and reducing latency, thus improving the overall QoS in software-defined mobile access networks.

IV. Conclusion

In this paper, we proposed a efficient resource-aware proactive flow rule caching leveraging MARL to address the limitations of conventional proactive caching methods in SDN. By predicting mobile nodes' movement patterns and caching flow rules in advance, our approach minimizes handover latency and enhances QoS for latency-sensitive applications. The proposed hierarchical agent structure, which dynamically adjusts caching strategies based on mobility levels, further optimizes FSHR and conserves valuable TCAM resources.

Our approach offers several key advantages. Firstly, the MARL model enables adaptive caching by monitoring the mobility behavior of MNs and pre-installing flow rules only when necessary, reducing unnecessary cache occupancy. Secondly, by proactively setting up flow rules in multiple (ADs for highly mobile nodes, the system ensures quick connectivity during handovers, significantly improving FSHR and reducing network latency.

While the proposed mechanism demonstrates promising results, several areas remain for future exploration. One potential area is enhancing the accuracy of mobility prediction models to further reduce prediction errors and prevent resource wastage. Additionally, expanding the model to consider more complex network conditions, such as varying traffic loads and heterogeneous wireless environments, could improve robustness and applicability in real-world scenarios. Lastly, integrating security mechanisms to protect flow rules and controller communications against malicious attacks would strengthen the resilience of the proposed framework.

In conclusion, the proposed MARL-based caching approach provides a scalable and efficient solution for flow rule management in SDN, especially for dynamic and mobile environments. As SDN continues to evolve, this approach offers valuable insights and methodologies for optimizing network performance, paving the way for next-generation intelligent network architectures that can seamlessly support diverse and demanding applications.

Acknowledgment

This results was supported by ”Regional Innovation Strategy (RIS)” through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(MOE)(2021RIS-003)

References

[1] D. Kreutz, F. M. V. Ramos, P. E. Veríssimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, “Software-defined networking: A comprehensive survey,” *Proceedings of the IEEE*, vol. 103, no. 1, pp. 14–76, 2015.

[2] B. A. A. Nunes, M. Mendonca, X.-N. Nguyen, K. Obraczka, and T. Turletti, “A survey of software-defined networking: Past, present, and future of programmable networks,” *IEEE Communications Surveys & Tutorials*, vol. 16, no. 3, pp. 1617–1634, 2014.

[3] K. Benton, L. J. Camp, and C. Small, “Openflow vulnerability assessment,” in *Proceedings of the Second ACM SIGCOMM Workshop on Hot Topics in Software Defined Networking*, ser. HotSDN ’13. New York, NY, USA: Association for Computing Machinery, 2013, p. 151–152. [Online]. Available: https://doi.org/10.1145/2491185.2491222

[4] A. Sallahi and M. St-Hilaire, “Optimal model for the controller placement problem in software defined networks,” *IEEE Communications Letters*, vol. 19, no. 1, pp. 30–33, 2015.

[5] S. Bera, S. Misra, and M. S. Obaidat, “Mobi-flow: Mobility-aware adaptive flow-rule placement in software-defined access network,” *IEEE Transactions on Mobile Computing*, vol. 18, no. 8, pp. 1831–1842, 2019.

[6] G. Kim, Y. Kim, and H. Lim, “Deep reinforcement learning-based routing on software-defined networks,” *IEEE Access*, vol. 10, pp. 18 121–18 133, 2022.

[7] Y. Kim and Y. Kyung, “Aoi-aware retained message policy in mqtt-based iot networks,” *IEEE Sensors Journal*, vol. 24, no. 21, pp. 35 809–35 819, 2024.

[8] Y. Kyung, J. Sung, H. Ko, T. Song, and Y. Kim, “Priority-aware actuation update scheme in heterogeneous industrial networks,” *Sensors*, vol. 24, no. 2, 2024. [Online]. Available: https://www.mdpi.com/1424-8220/24/2/357

[9] Y. Zeng, B. Ye, B. Tang, S. Lu, F. Xu, S. Guo, and Z. Qu, “Mobility-aware proactive flow setup in software-defined mobile edge networks,” *IEEE Transactions on Communications*, vol. 71, no. 3, pp. 1549–1563, 2023.

[10] Y. Kim, J. Park, and Y. Kyung, “Mobility-aware hybrid flow rule cache scheme in software-defined access networks,” *Electronics*, vol. 11, no. 1, 2022. [Online]. Available: https://www.mdpi.com/2079-9292/11/1/160