

# Programmable Chip Based High Performance MEC Router for Ultra-Low Latency and High Bandwidth Services in Distributed Computing Environment

SeokHwan KONG<sup>†a)</sup>, Member, Saikia DIPJYOTI<sup>††</sup>, and JaiYong LEE<sup>†††</sup>, Nonmembers

**SUMMARY** With the spread of smart cities through 5G and the development of IoT devices, the number of services requiring firm assurance of high capacity and ultra-low delay quality in various forms is increasing. However, continuous growth of large data makes it difficult for a centralized cloud to ensure quality of service. For this, a variety of distributed application architecture researches, such as MEC (Mobile|Mutli-access Edge Computing), are in progress. However, vendor-dependent MEC technology based on VNF (Virtual Network Function) has performance and scalability issues when deploying a variety of 5G-based services. This paper proposes PRISM-MECR, an SDN (Software Defined Network) based hardware accelerated MEC router using P4 [3] programmable chip, to improve forwarding performance while minimizing load of host CPU cores in charge of forwarding among MEC technologies.

**key words:** MEC, ULL, VNF, SDN

## 1. Introduction and Problem Statements

In 5G, connected devices such as cars, drones, VR (Virtual Reality) headsets, and robots are becoming very diverse. The problem is that unlike smartphones, these devices require ultra-low latency from a few milliseconds to 10ms and require high-capacity bandwidth from hundreds of Mbps to several Gbps per stream [1]. In addition, transmitting large amounts of traffic over the backhaul requires a huge amount of backhaul capacity, causing cost problems.

Researchers have studied distributed application architectures to accommodate ultra-low latency and high-capacity services in mobile networks. MEC is a distributed edge technology that processes data near a device. For this, ESTI standards organizations are making various standards such as MEC and MEPM (Mobile Edge Platform Manager). The MEC platform consists of MEC router and MEC applications. MEP (Mobile Edge Platform) within the orchestrator sends traffic rules to the MEC router through Mp2 interface. The MEC router decapsulates all GTP headers from the global node-B (gNB), looks at the GTP internal target IP,

and sends them directly if traffic destination is local network or local MEC application server. Because private network traffic is not transferred to mobile core network, security of private network data traffic is also high. This reduces the cost of private network deployment by using low-cost MEC router instead of using expensive UPFs (User Plane Functions) and combine with programmable chips to provide additional features such as ultra-low latency time synchronization and edge computing telemetry.

MEC routers cannot be supported with existing networks because they require routing based on GTP Inner IP. The existing VNF was configured and used on a CPU basis for ease of implementation [6]. VNF-based MEC routers have problems with forwarding performance compared to traditional legacy systems due to frequent packet replication and CPU interrupts [2]. There are acceleration methods such as DPDK (Data Plane Development Kit), it has scalability problem because it requires more CPU cores for performance close to line rates [7]. And compared to programmable chips, it is also limited for future traffic control due to higher vendor dependencies and lower programmability. This paper proposes the PRISM-MECR which is a programmable chip based high performance MEC router for ultra-low latency services in distributed computing environment. It uses the P4 technology to make the GTP match and encap/decap actions in programmable chip controlling them in an SDN manner.

## 2. PRISM-MECR Architecture

PRISM-MECR consists of P4-based programmable data plane and Openflow [4] based control plane. Figure 1 shows the configuration of PRISM-MECR based MEC site in 5G environment. TOF (Traffic Offload Function) within MEPM generates a local break out rules and forwards rules through the Mp2 interface to the PRISM-MECR control plane. PRISM-MECR control plane converts rules into an Openflow messages and forwards it to P4-based PRISM-MECR data plane.

For PRISM-MECR data plane, we modified a switch.p4 which is a reference switch pipeline and implemented the prism-mecr.p4 to support Openflow 1.3 and GTP functions. To support GTP functions, we implemented the prism-mecr.p4 by utilizing the vendor extension function of Openflow. If the destination UDP port is 2123, it will be

Manuscript received January 7, 2020.

Manuscript revised March 28, 2020.

Manuscript publicized July 1, 2020.

<sup>†</sup>The author is with the UbiNet Lab., Department of Electrical and Electronic Engineering, Yonsei University, 134 Shinchon Dong, Seodaemun-Gu, Seoul, 120–749, Korea.

<sup>††</sup>The author is with kulcloud, Changup Ro, Seongnam-Si, GyeongGiDo, 120–749, Korea.

<sup>†††</sup>The author is with the School of Electrical and Computer Engineering UNIST, Ulsan, Rep. of Korea.

a) E-mail: kongseokhwan@yonsei.ac.kr

DOI: 10.1587/transinf.2020PAL0001

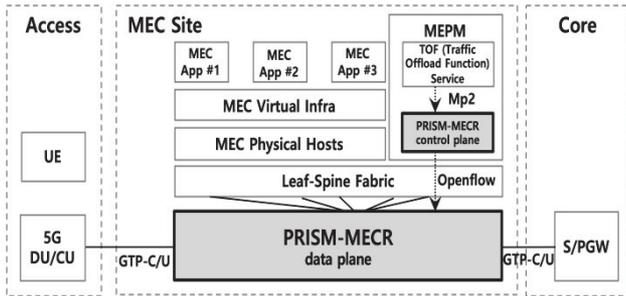


Fig. 1 PRISM-MECR location

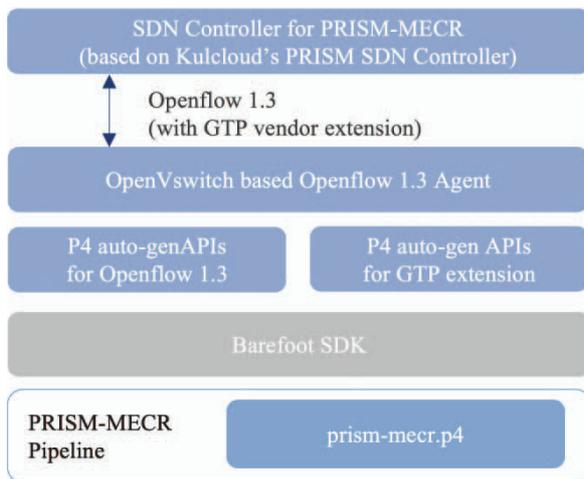


Fig. 2 PRISM-MECR internal overview

matched with GTP-C traffic and if it is 2125, it will be matched with GTP-U traffic. For GTP traffic, pipelines were configured to parse outer IP fields and inner IP fields. After matching GTP traffic, packet headers should be modified in a form that can be interpreted by MEC application servers, because the destination IP is tunneled to the IP of the UPF rather than the IP of the MEC application server. For this, we implemented the decap/encap function of GTP headers, allowing packets to be sent with the IP and MAC of the MEC application server via GTP decap.

When `prism-mecr.p4` is compiled by the p4 compiler, APIs are automatically generated to control the table in that pipeline. To integrate the control plane, we modified the Openflow agent application of the Open Vswitch-2.10.0 and linked it with that APIs. The user can use the vendor extension field of the SDN controller to set up the GTP match and encap/decap actions associated with PRISM-MECR. To implement GTP extension function in a control plane, we modified the Kulcloud's PRISM SDN controller which is a carrier-grade commercial SDN controller [5]. PRISM-MECR can provide virtualized MEC routers through port and VLAN based slicing.

### 3. Evaluation and Future Work

For the proposed PRISM-MECR data plane, we used the

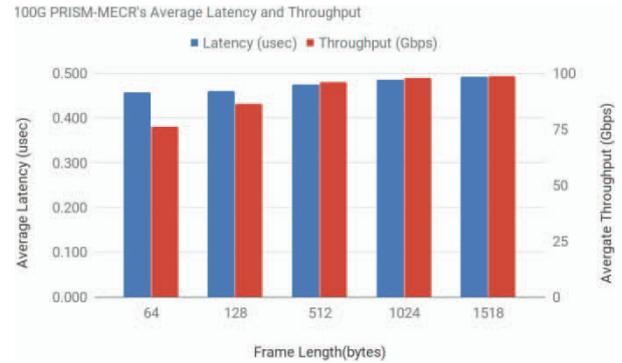


Fig. 3 PRISM-MECR forwarding performance

Barefoot Wedge 100BF-32X switch. The wedge 100BF-32X is a white box switch that consists of 100G QSFP28 32 ports and supports the P4 language. The `prism-mecr.p4` uses Barefoot's SDE 8.5 version and the p4-16 language. The control plane of PRISM-MECR is equipped with 3.3GHz Intel Xeon E3-1230 V2 CPU and 8G RAM memory. We connected control plane and 100BF-32X switch through the 1G management interface. For performance measurements, the 100BF-32X switch connects one input port and one output port to the Spirent N2X 100G traffic generator.

To measure the performance of PRISM-MECR's GTP matching function, we set up rules to match the GTP inner source IP & destination IP and send them to the output port decapsulating GTP header. Traffic generated about 100 pairs per frame length (64, 128, 512, 1024, 1518 bytes) to measure the latency and throughput. The average value is calculated by repeating 100 times. As shown in Fig. 3, PRISM-MECR provides up to 98.7 Gbps throughput and 0.458 usec latency in best case. This means that PRISM-MECR provides the maximum line speed of the hardware chip without packet loss, and the 0.458 usec latency means that the transfer delay within the MEC can be guaranteed with only a 0.03% transfer delay margin in the ultra-low latency service, which requires a 2ms end-to-end transfer delay. After rules are set by MEC router, all forwarding takes place in the data plane, so there is no performance degradation due to the increase in the number of MEC routers in the control plane. Instead, performance will be affected by the size of the forwarding table on the network chip. Rules can be set up 120K for Barefoot wedge's Tofino chips. For example, if 1K MEC router rules are required per user, a maximum of 120 MEC routers can be operated without performance degradation based on the Tofino chip.

### 4. Conclusion

In this paper, we describe the PRISM-MECR that can be applied in distributed computing environment which requires ultra-low latency forwarding performance and high programmability. This implemented using P4 programmable chip and SDN manner. We have demonstrated that PRISM-MECR can support 0.458 usec latency (with 64 bytes frame)

and 98.7Gbps bandwidth (with 1518 bytes frame) using traffic generator. It can cover with 0.03% margin of end to end ultra-low latency service. In the future work, we will develop a standard interface to integrate completely with MEPM and improve new functions such as hardware accelerated time synchronization, load balancing, and NAT for MEC.

#### References

- [1] N. Ansari and X. Sun, "Mobile Edge Computing Empowers Internet of Things," *IEICE Trans. Commun.*, vol.101-B, no.3, pp.604–619, 2018.
  - [2] N. Pitaev, M. Falkner, A. Leivadreas, and I. Lambadaris, "Characterizing the performance of concurrent virtualized network functions with OVS-DPDK, FD. IO VPP and SR-IOV," *Proc. ACM/SPEC Int. Conf. Perform. Eng.*, pp.285–292, 2018.
  - [3] P. Bosshart, D. Daly, G. Gibb, M. Izzard, N. McKeown, J. Rexford, C. Schlesinger, D. Talayco, A. Vahdat, G. Varghese, and D. Walker, "P4: Programming protocol-independent packet processors," *SIGCOMM Comput. Commun. Rev.*, vol.44, pp.87–95, July 2014.
  - [4] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "Openflow: enabling innovation in campus networks," *ACM SIGCOMM Computer Communication Review*, vol.38, no.2, pp.69–74, 2008.
  - [5] PRISM SDN, <http://www.kulcloud.com/en/prism>.
  - [6] I. Sarrigiannis, E. Kartsakli, K. Ramantas, A. Antonopoulos, and C. Verikoukis, "Application and network VNF migration in a MEC-enabled 5G architecture," *Proc. 23rd IEEE Int. Workshop Comput. Aided Modeling Design Commun. Links Netw. (CAMAD)*, pp.1–6, Nov. 2018.
  - [7] D. Firestone, A. Putnam, S. Mundkur, D. Chiou, A. Dabagh, M. Andrewartha, H. Angepat, V. Bhanu, A. Caulfield, E. Chung, H.K. Chandrappa, S. Chaturmohta, M. Humphrey, J. Lavier, N. Lam, F. Liu, K. Ovtcharov, J. Padhye, G. Popuri, S. Raïndel, T. Sapre, M. Shaw, G. Silva, M. Sivakumar, N. Srivastava, A. Verma, Q. Zuhair, D. Bansal, D. Burger, K. Vaid, D.A. Maltz, and A. Greenberg, "Azure accelerated networking: SmartNICs in the public cloud," *15th USENIX Symposium on Networked Systems Design and Implementation (NSDI 18)*, pp.51–66, 2018.
-