Adaptive Access Gateway Function Framework with P4 Accelerated User Plane

Ze-Yu Jin, Jen-Chun Ou, Li-Hsing Yen, Chien-Chao Tseng

Dept. of Computer Science, College of Computer Science, National Yang Ming Chiao Tung University, Hsinchu, Taiwan.

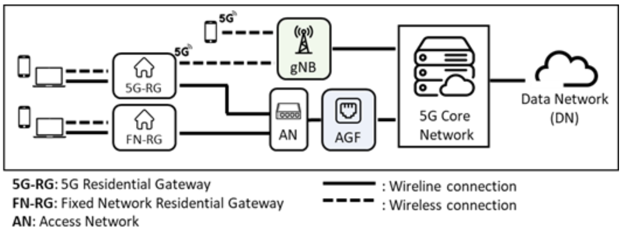

Fig. 1: 5G Architecture with AGF

***Abstract*— In order to facilitate the 5G Wireless and Wireline Convergence (5G-WWC) architecture, the Access Gateway Function (AGF) is defined as an interworking function between the 5G core (5GC) and fixed network residential gateways (FN-RG), utilizing a non-3GPP wireline access network. As a proxy, AGF enables communication between the FN-RG and the 5GC. This paper presents an adaptive architecture integrating different access gateway functional user planes. We have implemented a P4 switch-accelerated access gateway user plane with data stream bandwidth management to verify the functionality and adaptability of the adaptive architecture.**

Keywords—5G-WWC, AGF, Adaptive Framework, P4

I. INTRODUCTION

A recent trend in telecommunications is the convergence of the fifth-generation (5G) system with non-5G access networks. This enables user equipment (UE) to connect to a 5G system via access technologies not explicitly defined by 5G standards. To facilitate this integration, the 3rd Generation Partnership Project (3GPP) has defined support for three non-3GPP access networks: untrusted, trusted, and wireline [1]. The category of untrusted access networks encompasses systems such as IEEE 802.11 (Wi-Fi), whereas the category of trusted access networks pertains to secure wireless systems. The category of wireline networks constitutes part of the 5G wireless and wireline convergence (5G-WWC) [2] [3], which enables network operators to expand their service coverage and obviates the necessity of deploying a separate core network for wireline connections.

A joint initiative between the 3GPP and the Broadband Forum (BBF) has resulted in the development of 5G-WWC specifications. In collaboration, they define a novel network function designated the Access Gateway Function (AGF). As in Fig. 1, AGF facilitates connectivity to the 5G core (5GC) from both the 5G-capable residential gateway (5GRG) and the fixed network residential gateway (FN-RG). While the 5G-RG is capable of exchanging control plane messages with the 5GC via the gNodeB (gNB), fixed network residential gateways (FN-RGs), such as cable modems, lack the requisite 5G capability. Consequently, AGF assumes the role of a user equipment proxy for FN-RGs. In the control plane, AGF functions as a gNB, communicating with the Access and Mobility Management Function (AMF) in 5GC via the N2 interface. Furthermore, AGF serves as a user equipment (UE) for each FN-RG, communicating with the access and mobility management function (AMF) through the N1 interface. The delivery of packets between the FN-RG and 5GC in the user plane is facilitated by AGF through the N3 interface.

Our prior work in [4] proposed a framework that divides AGF into the AGF Control Plane (AGF-CP) and the AGF User Plane (AGF-UP), which aligned with 5GC in Control and User Plane Separation (CUPS). The authors also provided three AGF-UP implementations based on UERANSIM [5], gtp5g [6], and DPDK [7]. In this paper, we further restructure AGF-UP into two parts (AGF-UP-c and AGF-UP-u) following the CUPS principle and add an abstract layer in between to facilitate the adaption of any AGF-UP-u implementation. We. We showcase the efficiency of the proposed framework by providing an AGF-UP implementation based on Programming Protocol-independent Packet Processors (P4). We set up an experimental environment to test the latency and throughput between FN-RG and the data network (DN). Our contribution is twofold:

- Proposing a new framework that facilitates the adaption of any AGF-UP implementation by restructuring AGF-UP into two AGF-UP-c and AGF-UP-u.
- Showcasing an AGF-UP implementation based on P4 that is capable of bandwidth management under this framework.

The remainder of this paper is structured as follows. Sec. II briefs the backgrounds and related work. Sec. III presents the design and requirements. Sec. IV describes the method of implementation. Sec. V shows the experimental results, and the last section concludes the paper.

II. RELATED WORK

In our prior work [4], we have proposed a framework for the separation of AGF control and user plane, where three different implementations based on user-space, kernel-space, and kernel-bypass (DPDK) software were reported and tested. The results exhibited the superiority of the DPDK approach in terms of latency and throughput of user plane traffic.

In [8], Chen et al. proposed a bandwidth management mechanism for software-defined networking (SDN) switches. The mechanism boosted by P4 classifies user traffic into three different types based on QoS demands and traffic status. User packets pass through a two-level priority queue for possible packet disaggregation or drop. Experiments confirmed the effectiveness of the mechanism in limiting the maximum allowed rate and guaranteeing a minimum bandwidth for each traffic flow.

III. DESIGN OF NON-3GPP WIRELINE ACCSEE GATEWAY

Our architecture of AGF consists of AGF-CP, AGF-UP-c, and AGF-UP-u, as shown in Fig. 2.

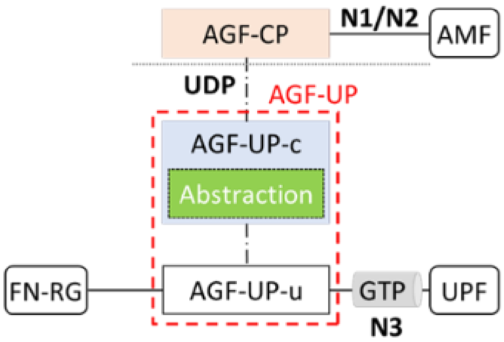

Fig. 2: AGF UP CUPS and Abstraction Layer

A. Functional Requirements

Implementation of AGF should provide the following functionalities:

- *Enabling FN-RG attachment.* AGF should provide IP connectivity for FN-RGs capable of dynamically attaching to it. This can be achieved via the Dynamic Host Configuration Protocol (DHCP).
- *Performing UE registration and authentication.* AGF should perform UE registration and authentication on behalf of FN-RGs. Once these procedures have been completed, 5GC will treat each FN-RG as a registered UE.
- *Establishing PDU sessions.* Given that the registered UE utilizes the Protocol Data Unit (PDU) session for the exchange of data between the UE and the UPF, AGF should request the establishment of a PDU session on behalf of FN-RGs.
- *Exchanging control messages with 5GC via the N2 interface.* This is necessary because AGF should act as a gNB.
- *Encapsulating/decapsulating data stream in N3*. To facilitate data delivery through the N3 interface, it is necessary to establish a tunnel between the gNB and UPF. This can be achieved using the GPRS Tunneling Protocol-User Plane (GTP-U) [9]. Given that AGF acts as a gNB, it is responsible for encapsulating user data from FN-RG and decapsulating user data from UPF using GTP-U.

Except for the last functionality, which is the primary objective of AGF-UP-u, all the functionalities above are managed by AGF-CP and AGF-UP-c.

B. AGF-CP

In our design, AGF-CP comprises Proxy UE and N2 modules. Proxy UE is the AGF module responsible for 5G user registration on behalf of FN-RG. Since FN-RG only uses IPoE/PPPoE to connect to AGF, AGF will use the function of Proxy UE to handle user registration on behalf of FN-RG. AGF must communicate with AMF in the 5G core network, so there is an N2 module in AGF for N2 interface message processing.

C. AGF-UP-c

AGF-UP-c includes the DHCP module, ARP module, Proxy UE Manager, and Adaptor. The main function of the DHCP Module is to process the DHCPv4 message that FN-RG uses to connect. When FN-RG uses DHCP Discover to connect, the DHCP Module notifies AGF-CP to create a Proxy UE representing the FN-RG, waits for the 5G core network for the assignment of an IP address, and then uses its IP to complete DHCPv4 four-way handshake. Since AGF is in an ARP messaging environment, the ARP module is responsible for processing these messages and sending out active ARP Requests, asking for the MAC Address of the device connected to it. The proxy UE manager is mainly responsible for creating and managing proxy UEs and managing PDU session information for UEs. After Proxy UE completes the user registration and PDU session request for the 5G core network on behalf of FN-RG, it will obtain the PDU session information used by FN-RG.

The abstract layer mainly provides different implementations of the user plane that can be more quickly connected to the adaptive architecture of the implementation. To implement different user planes, we only need to develop their corresponding adapters to communicate with the user plane and convert the PDU Session Information stored in the common format into the information required by the user plane to complete the interconnection of the new user plane.

D. AGF-UP-u

AGF-UP-u concentrates on packaging, unpacking, and forwarding user traffic. Since the implementation of AGF-UP-u directly affects the service quality of AGF in terms of bandwidth, performance, and delay when serving FN-RG, this paper will use the P4 programming language with P4 switches for the implementation. In addition, the quality of service of the User Plane is provided to manage bandwidth for different quality of service flows.

IV. Programmable Switch Approach: P4 Switch

Our AGF implementation follows the methodology of the previous paper [4], retaining the Proxy UE module and N2 module from the original AGF-CP using UERANSIM. AGF-UP also separates the user plane from the control plane, dividing it into AGF-UP-c and AGF-UP-u to make the overall architecture more scalable. In AGF-UP-u, we use P4 Switch. P4 [10] is a programming language that can be used to design a network packet processing pipeline with the following merits.

- *Protocol Independence.* P4 programs only specify the process of pipeline processing and are not specific to a particular protocol, so they can be used to implement different protocol behaviors or specifications.
- *Target Independence.* Programmable chipset vendors provide compilers and drivers so that the pipeline defined by P4 programs can work properly in any vendor's switch.
- *Field Reconfigurable.* After the P4 program is deployed, users can still change the P4 program to meet different network requirements.

In the practical work above, AGF-CP and AGF-UP have been completely decoupled, but there is still information to be transferred between them, so we add AGF-UP-c core for information transfer between AGF-CP and AGF-UP-c. The architecture of the Programmable switch approach is shown in Fig. 3

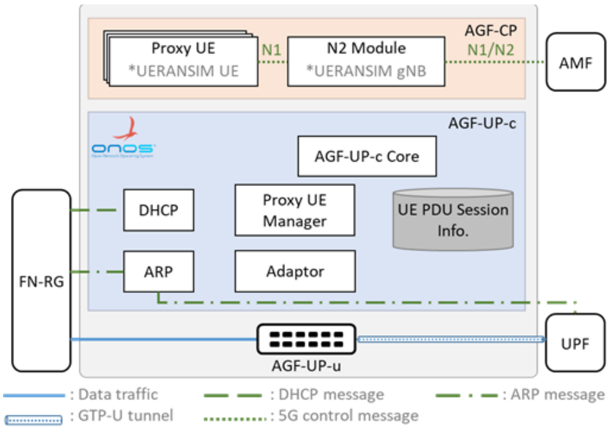

Fig. 3: Adaptive AGF Framework

A. *Implementation of AGF-CP*

- *Proxy UE.* The Proxy UE module in this work has been adapted from the UE Simulator in UERANSIM. Firstly, as this paper utilizes the P4 switch for acceleration, the module USER pertaining to the UE Simulator User Plane has been removed. Secondly, to ascertain the IP addresses assigned to the Proxy UE following registration with the 5G core network, After the UE registration and PDU session request are completed, the Proxy UE will actively report the IP assigned by the 5G core network to gNB Simulator.

- *N2 Module.* The N2 Module in this project has been modified from the gNB Simulator in UERANSIM. The N2 Module is also the primary information control module in AGF-CP. It is responsible for notifying AGF-UP-c of the information collected from IP and PDU session requests reported by proxy UEs upon completion. Consequently, a new module, FORWARDER, has been designed to deliver PDU session information to AGF-UP-c. A UDP socket facilitates communication between the AGF-CP (N2 Module) and the AGF-UP-c (AGF-UP-c Core).

B. *Implementation of AGF-UP-c.*

- *DHCP Module.* This module is responsible for handling packet-in and emulating DHCP server-like behavior. When a DHCP Discover message is received, indicating that a new FN-RG wants to connect, the DHCP Module retrieves its MAC Address and uses the SHA-256 hash algorithm to obtain the hash value of the MAC address. The hash value is used to establish the SUPI of the Proxy UE. When the Proxy UE registers the 5G core network user and gets the assigned IP, DHCP will use this IP to pack the DHCP Offer message and send it back to FN-RG. When it receives the DHCP Request, it will respond with the corresponding DHCP Ack message to complete the DHCP 4-Way Handshake process.
- *ARP Module.* This module is a Packet-in processor that handles ARP-related behavior. When this module receives an ARP Request, the module responds with an ARP Reply. This module also sends an ARP Request for network topology information.
- *AGF-UP-c Core.* To communicate between AGF-UP-c and AGF-CP, a UDP socket is used to receive PDU Session Information from AGF-CP (N2 Module). When AGF-UP-c Core receives the PDU Session Information, it parses it and converts it into a Java language object. It then notifies and delivers a new PDU Session Information object to the Proxy UE Manager.
- *Proxy UE Manager.* This module is used to manage Proxy UE instances and store PDU Session Information. It has two main functions. Firstly, when the DHCP Module detects a new FN-RG connection, the Proxy UE Manager helps it create and manage a Proxy UE instance. Next, when the Proxy UE completes the registration, the AGF-UP-c Core notifies and sends the PDU Session Information object of the Proxy UE to the Proxy UE Manager to keep this object. The Proxy UE Manager then informs the Adaptor that there is new PDU Session Information so that the Adaptor can update the Flow Rule of AGF-UP-u.
- *Adaptor.* The main function is to convert the updated PDU Session Information into the corresponding data format required by the user plane and transfer it to the user plane when the Proxy UE Manager finishes storing and notifying the PDU Session Information. Take the P4 Accelerated user plane as an example. The P4 Adaptor will convert the PDU Session Information into the Flow Rule used by P4, install it on P4 through P4Runtime, and then manage the Flow Rule on it.

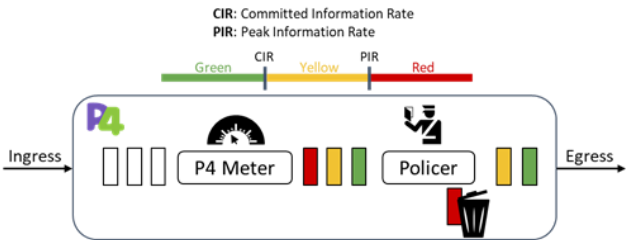

Fig. 4: Classifying packets into three different colors

C. *Implementation of AGF-UP-u*

Our design uses a P4 switch to accelerate the data transfer from AGF-UP-u. The P4 Pipeline used in this paper is based on the Tofino Native Architecture (TNA) [11] architecture. We use P4 Meter [8] to classify traffic packets and then apply rate limiting or bandwidth guarantee operations accordingly. For packet classification, the trTCM parameter Peak information Rate (PIR) is used for bandwidth limiting and (Committed Information Rate (CIR) for bandwidth guaranteeing, as shown in Fig. 4. With this setup, P4 meter classifies packets into three categories, each represented by a different color, and saves the results as metadata for the packets. When a flow's packets are classified as green, the current bandwidth allocated to that flow is lower than its CIR. Therefore, P4 meter tries to provide guaranteed bandwidth by setting a high forwarding priority for these packets. When the packet is yellow, the allocated bandwidth is between the CIR and PIR (or these parameters are not specified). In this case, P4 meter provides the best-effort packet transmission for these packets by allocating only the remaining bandwidth with a low forwarding priority. For red packets, the current bandwidth allocated to the traffic has exceeded the PIR. Therefore, P4 meter drops these packets directly.

We designed three tables in the P4 pipeline responsible for QoS behavior.

1) UL/DL QoS Determination Table. In 5G, QFI will be used to classify data streams for quality of service (QoS). UL/DL QoS Determination Table will use normal 5 Tuples or Inner 5 Tuples to identify different data streams and distribute their corresponding QFIs for QoS enforcement in the subsequent table.

2) QoS Classification Table. According to different QFI values, packets will be handed over to different P4 Meters for packet flow rate indication, and the packets will be marked with colors using the P4 Meter.

3) QoS Policer Table. Observing the colors marked on the packets, the packets are assigned to the corresponding Priority Queue to achieve the bandwidth guarantee and limiting behaviors.

D. DPDK AGF Integration

To test the adaptive AGF architecture's adaptability, we separated the DPDK AGF user plane in [4]. Then, we add a DPDK adapter to the abstract layer of the Adaptive AGF architecture and a DPDK agent that can communicate with the DPDK adapter on top of the DPDK AGF user plane architecture, as shown in Fig. 5. The DPDK AGF user plane can use our Adaptive AGF as a control architecture to provide full AGF functionality.

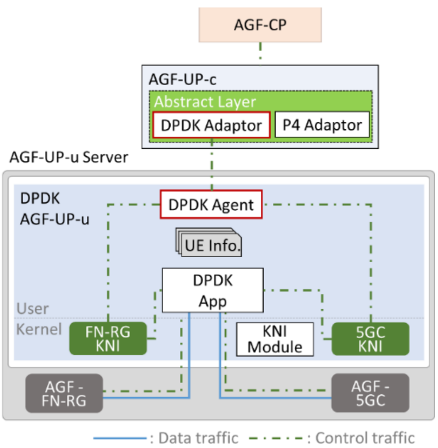

Fig. 5: DPDK Integration

E. FN-RG Attachment Workflow

The FN-RG connecting to the 5G core network for Internet access via AGF includes several steps (see Fig. 6).

1) Detecting FN-RG attachment. The FN-RG sends a DHCP Discover message to request a connection, which the DHCP Module receives in the AGF (Step 1). The DHCP Module informs the Proxy UE Manager about the new FN-RG requesting a connection and informs the AGF-UP-u implementation to which the FN-RG is currently connected (Step 2).

2) Deploying Proxy UE. When Proxy UE Manager gets the information about the new FN-RG connection, Proxy UE Manager retrieves the MAC Address of the FN-RG and uses the SHA-256 hash algorithm to obtain the hash value. The first ten digits of the hash value are used as the SUPI of the FN-RG (Step 3). Then, the Proxy UE Manager uses this SUPI to create a Proxy UE instance for the FN-RG (Step 4).

3) Establishing PDU Session. After creating the Proxy UE instance, the Proxy UE initiates user registration and PDU session requests for the 5G core network (Steps 5 and 6).

4) Recording and managing PDU session information. After the 5G core network registration, the N2 Module packages the PDU session information (including uplink and downlink TEID, uplink and downlink IP, and UE IP) into JSON format. It sends it to the AGF-UP-c Core, which retrieves the information, saves it into Java objects, and passes it to the Proxy UE Manage for management (Step 7,8).

5) Installing and Managing Flow Rules. The Proxy UE Manager informs the Adaptor that new PDU Session information is available after it has acquired and saved the new PDU Session information object (Step 9). Afterwards, the Adaptor will reads the new PDU session information (Step 10). The information is then converted to the flow rule used by the underlying AGF-UP-u and the configuration installed (Step 11).

6) Assignment of FN-RG IP. The Proxy UE Manager the IP address obtained by the Proxy FN-RG to the DHCP module (Step 12). Once the IP address has been obtained, the DHCP module uses this address to complete the four-way handshake with the FN-RG (Step 13).

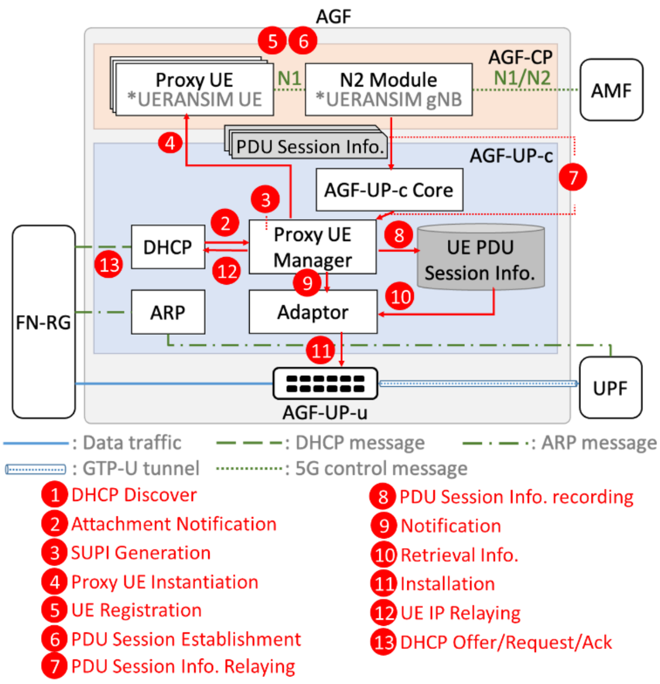

Fig. 6: FN-RG Attachment Workflow

V. Performance Evaluation

We test the performance of the adaptive AGF architecture, the P4-accelerated user plane, and the integration of DPDK AGF. We used free5GC [12] as the 5GC with P4-UPF to ensure the bandwidth of the user plane in the 5GC. We adopt two test environments: The Environment (A) cuts out three

Table I: Server Specification

Machine	CPU	Memory	OS	Note
FN-RG/DN Emulator	Intel Xeon E5-2630	128GB	Ubuntu 18.04.1 LTS (Kernel 5.4.0-87-generic)	Environment (A) NIC: Intel-X710 10Gbps
FN-RG/DN Emulator	8 Armv8 A72 cores (64-bit)	16GB		Environment (B) NIC: NVIDIA BlueField 100GbE Dual-Port
free5GC Virtual machine	Intel Xeon CPU E5-2630 @8 vCPU	8GB	Ubuntu 18.04.1 LTS (Kernel 5.4.0-87-generic)	NIC: PCIe Passthrough
ONOS Virtual machine	Intel Xeon CPU E5-2630 @8 vCPU	8GB	Ubuntu 18.04.1 LTS (Kernel 5.4.0-87-generic)	NIC: PCIe Passthrough

Table II: Switch Specification

Switch	Hardware	Software
P4-AGF	Inventec D10056	Barefoot SDE 9.3.0
P4-UPF	Inventec D10056	Barefoot SDE 9.3.0

namespaces from each server, simulating three FN-RGs and three DNs. All connections were SFP+10G fiber cables. The Environment (B) in Fig. 7 used two 100GbE ports on the NVIDIA DPU as the FN-RG and DN end to form a ring topology. Table I and Table II show the specifications of the servers and switches used in the experiments, respectively.

A. *End-to-End Latency*

We used Environment (A) to perform latency tests on P4-AGF to study the efficiency of P4-AGF in performing encapsulation and decapsulation. We pinged the DN server from the FN-RG server, sending 100 ICMP ping requests at 0.2-second intervals, and recorded all round trip times (RTTs). The results are shown in Table III. P4-AGF's processing efficiency can keep a small time cost and has a relatively low mean deviation, which can be recognized as its stable and fast performance.

B. *Throughput*

Next, we measured the throughput of P4-AGF in Environment (B). The following two subsections report and discuss uplink and downlink performance.

1) Uplink. The FN-RG side of the DPU used TRex [12] to generate UDP traffic with different payload sizes and sent it to the P4-UPF after being encapsulated by the P4-AGF (*Fig. 8*). In the Bit Rate section, it can be observed that the RX could not reach 100Gbps when the packet was small due to the DPU limitation, while the packet could reach 100Gbps in Tx after the encapsulation of the P4-AGF, which indicates that the experimental situation reached the maximum load of the network cable. It can also be observed that the 104.3 Mpps observed at Tx with zero payload was the highest performance for processing packets within the observable range of the P4-AGF. When looking at the graph of Packet Rate, since the experiment reached the maximum load of the network line, not all incoming packets could be processed, so the Rx value was larger than the Tx.

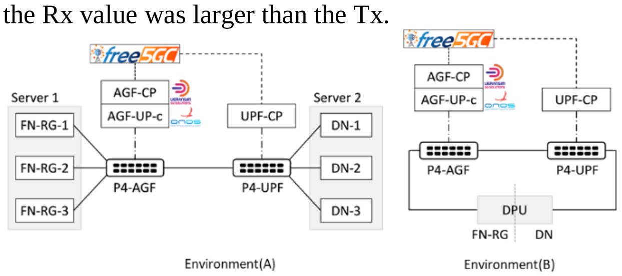

Fig. 7: Environment Setup

2) Downlink. The DN side of the DPU used TRex to generate UDP traffic with different payload sizes and sent it to the P4-AGF after being encapsulated by the P4-UPF (Fig. 9). For the Bit Rate graph, because the packets sent to P4-AGF were sealed, the Rx could reach 100Gbps. The Tx decreased due to the unsealing of the packets. In the Packet Rate graph, we can see that both Tx and Rx had the same number of packets per second, which means that all the downstream packets were processed, and the observed maximum processing rate was 104.2Mbps.

Table III: Latency Test

MIN.	0.110 ms
AVG.	0.139 ms
MAX.	0.174 ms
MDEV.	0.024 ms

C. *Bandwidth Guarantee and Limiting*

We tested the performance of P4-AGF combined with the bandwidth management of P4-TINS [8] in Environment (A). The experiment delivered three types of traffic from three sets of FN-RGs and DNs within 100 seconds. Non-Guaranteed TCP (N-TCP) traffic was delivered using iperf3 [13] for 0 to 100 seconds. We used iperf3 to send Guaranteed TCP (G-TCP) traffic with corresponding Guaranteed Flow Bit Rate (GBR) and Maximum Flow Bit Rate (MBR) settings to limit and guarantee its bandwidth from 20 to 80 seconds. We used iperf2 [13] to send Background UDP (B-UDP) traffic, which is more aggressive in bandwidth contention; it is used as background traffic to verify bandwidth assurance behavour. The time range is 40 to 60 seconds.

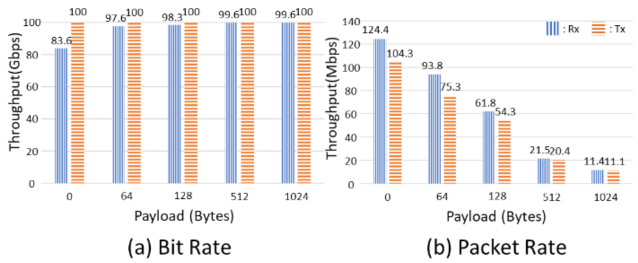

Fig. 8: Uplink Throughput

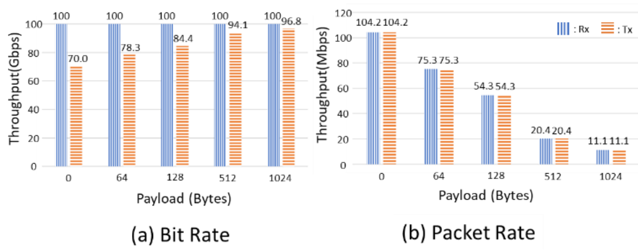

Fig. 9: Downlink Throughput

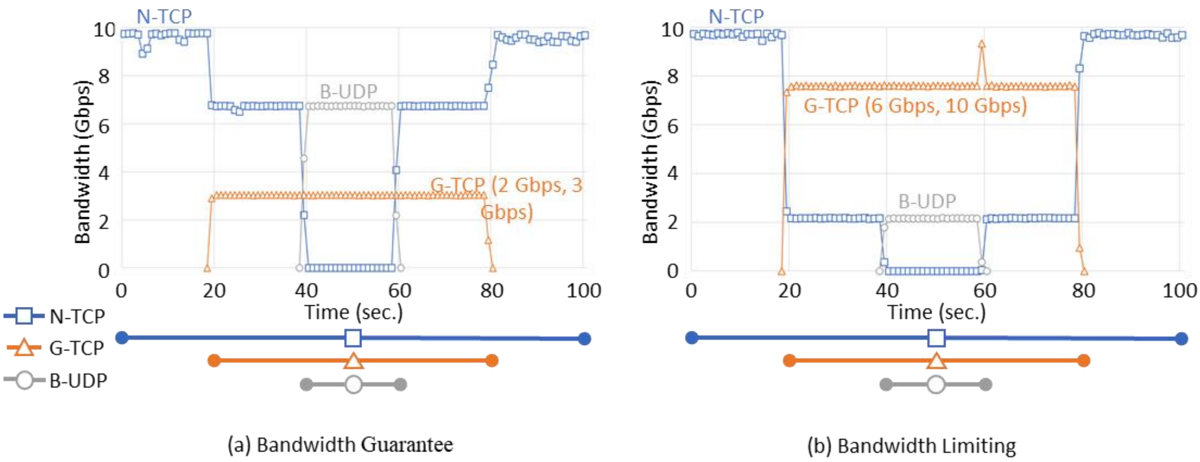

Fig. 10: Bandwidth Test

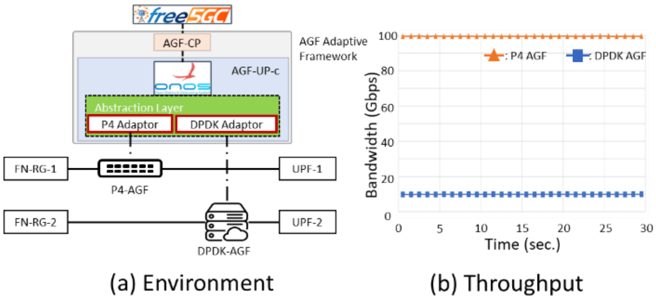

Fig. 11: User Plane Integration

1) *Bandwidth Guarantee*

We set GBR to 6Gbps and MBR to 10Gbps. The bandwidth guarantee can be observed at higher GBR settings. Experimental results are shown in Fig. 10, showing that G-TCP could obtain the guaranteed traffic stably regardless of whether it coexisted with N-TCP or DP. After 40 seconds of B-UDP transmission, the original bandwidth of N-TCP was taken by B-UDP due to its weak competitiveness.

2) *Bandwidth Limiting*

We set GBR to 2Gbps and MBR to 3Gbps, and the experiment results are shown in Fig. 10. We deliberately set MBR lower to show the effect of traffic limitation. We can see that the traffic was maintained at the upper limit of 3Gbps after B-TCP started, and B-UDP occupied the bandwidth of N-TCP because of its bandwidth aggressiveness after B-UDP started sending.

D. *User Plane Integration*

The experimental environment is shown in Fig. 11 (A), which Environment (B) expanded to control DPDK AGF's connection and observe the two sets of AGF sending traffic simultaneously. From Fig. 11 (B). we can observe that when sending traffic simultaneously, the adaptive AGF architecture completed the registration of the two FN-RGs normally and made each user plane reach its maximum performance.

VI. Conclusions

This paper proposes an adaptive access gateway function and implements the user plane accelerated by P4. In the adaptive architecture part, an abstraction layer concept is designed so that the functional modules of the management plane can be adjusted according to the implementation of the user plane, maximizing the shared functional modules. Based on the P4 programming language and the P4 switch, a high-performance user plane with individual data stream bandwidth management is realized.

VII. Acknowledgment

This work was supported in part by the National Science and Technology Council, Taiwan, under grants MOST 110-2221-E-A49-044-MY3, MOST 110-2221-E-A49-064-MY3, NSTC 112-2622-8-A49-019, NSTC 113-2218-E-011-008, NSTC 113-2218-E-011-009 and NSTC 113-2221-E-A49-179.

VIII. References

[1] "Access to the 3GPP 5G Core Network (5GCN) via non-3GPP access networks 3rd generation partnership project-3GPP," 3GPP, TS 124 502, V17.5.0, Release 17, May. 2022.

[2] "Wireless and wireline convergence access support for the 5G System (5GS)," 3GPP, TS 23.316, V16.4.0, Release 16, 3GPP, Jul. 2020.

[3] 5G Wireless Wireline Convergence Architecture, TR-470, issue: 02, BBF, Mar. 2022.

[4] Pai-Hui Wang, Hung-Chang Tsao, Li-Hsing Yen, and Chien-Chao Tseng, "Design and Implementation of non-3GPP Wireline Access Gateway for 5G Wireless and Wireline Convergence," *IEEE VTC2024-Spring*, Singapore, June 2024..

[5] "GitHub: aligungr/UERANSIM: Open source 5G UE and RAN (gNodeB) mplementation," https://github.com/aligungr/UERANSIM, accessed: 2023-05-10.

[6] "gtp5g: 5G compatible GTP kernel module," https://github.com/free5gc/gtp5g, accessed: 2023-09-16.

[7] "Data plane development kit," https://www.dpdk.org/, accessed: 2023-09-17.

[8] Yan-Wei Chen, Chi-Yu Li, Chien-Chao Tseng, and Min-Zhi Hu, "P4-TINS: P4-Driven Traffic Isolation for Network Slicing With Bandwidth Guarantee and Management," IEEE Transactions on Network and Service Management, vol. 3, no. 19, pp. 3290-3303, Mar. 2022.

[9] "Universal mobile telecommunications system (UMTS); LTE; 5G;general packet radio system (GPRS) tunnelling protocol user plane (GTPv1-U)," 3GPP, TS 29.281, V15.7.0, Release 15, 3GPP, Jan 2020

[10] "P4 Language and Related Specifications," https://p4.org/specs/, accessed: 2023-07-10.

[11] "Tofino," https://barefootnetworks.com/products/brief-tofino/, accessed: 2023-06-23.

[12] "free5GC, " https://free5gc.org/, accessed: 2024-06-01.

[13] "TRex," https://trex-tgn.cisco.com/, accessed: 2023-10-13.

[14] "iPerf - The ultimate speed test tool for TCP, UDP and SCTP," https://iperf.fr/, accessed: 2023-06-03.