Context-Aware Service Chaining Framework for Over-the-Top Applications in 5G Networks

Chang Ge, David Lake, Ning Wang, Yogaratnam Rahulan and Rahim Tafazolli
5GIC, Institute for Communication Systems, University of Surrey, Guildford, UK
Email: {C.Ge, D.Lake, N.Wang, Y.Rahulan, R.Tafazolli}@surrey.ac.uk

Abstract—**As over-the-top (OTT) applications such as videos dominate the global mobile traffic, conventional content delivery techniques such as caching no longer suffice to cope with mobile network users' requirements due to e.g., fluctuating radio conditions. In legacy 4G networks, mobile network operator (MNO) and OTT service providers (OSPs) are logically decoupled from each other, hence preventing them to share necessary context and enable in-network context-aware intelligence. The recently standardized 5G network architecture's softwarized and virtualized nature opens up new opportunities for flexible deployment of MNO- and OSP-operated network functions. In this work, we first extend the current 5G standard to enable third-party stakeholders to deploy their own user-plane functions (UPFs) within the MNO infrastructure. Based on this, we propose a service function chaining (SFC) framework within the 5G core network, which allows MNO to dynamically determine the optimal set of UPFs that each flow should traverse based on their real-time contexts. The proposed framework has been implemented in a testbed network. Through realistic experiments, we demonstrate that UPF deployment strategy plays a crucial rule in the resulting SFC performance, and our proposed scheme can achieve performance that is close to the benchmark. Furthermore, we establish recommendations on best practices of UPF deployment strategies in 5G network.**

I. Introduction

The global mobile data traffic, especially those originated from Over-the-Top (OTT) applications such as webpages and videos, has been skyrocketing in recent years [1]. Nowadays, it is common practice for OTT service providers (OSPs) to perform content caching and/or prefetching at locations within proximity of end-users, which aim to perform Quality-of-Experience (QoE) assurance at transport and application layers through reduced content access latency.

Although such techniques generally work well in fixed networks, there are distinct challenges regarding their applications in mobile networks. In legacy mobile networks (e.g., LTE-Advanced), due to the network system architecture design, the mobile network operator (MNO)'s network functions are logically isolated from OSPs' functions such as content caches. This means that the MNO is unaware of the OSP users' application context or requirements, and the OSP is unaware of its users' network context such as Radio Access Network (RAN) conditions. From the MNO's perspective, intuitively, it should treat the flows that traverse its network infrastructure differently based on their various application-level requirements and contexts, which are unknown to the MNO. From the OSP's perspective, e.g., in the case of video streaming, not all users require caching or prefetching operations depending on their backhaul network contexts [2], [3]. Therefore, we envisage that a holistic system-level approach is required for MNO and OSPs to not only share their knowledge on network and application aspects, but also enable network and content operations in a more context-aware manner.

The softwarized and virtualized nature of the 5G network system architecture [4] has opened up new opportunities in supporting flexible deployment and embed of native and third-party VNFs. In this paper, we take the current 3GPP 5G Phase 1 system architecture [4] as a starting point and extend it to address the challenges above. We first propose extensions to the definition of User-Plane Function (UPF) in the standard, which enables third-party stakeholders such as OSPs to deploy their own UPFs and embed intelligence (e.g., caching and prefetching) within the MNO-operated infrastructures as virtual network functions (VNFs). This is enabled by software-defined networking (SDN) and network function virtualization (NFV) in the 5G core network, where the MNO leases virtual computing and storage resources to OSPs. Based on this, we propose a system architecture where a number of UPFs with logically decoupled functionalities are defined, which remains compliant to relevant 3GPP standards.

Based on the proposed architecture above, we propose two key evolutions on the 5G network systems. First, we specify the methodology on sharing multi-dimensional context information between MNO and OSPs within the 5G core network and envisage the enabled scenarios. Second, based on the policies that are established with shared contexts, we propose a novel service function chaining (SFC) framework within the 5G network, where the MNO dynamically determines the optimal set of UPFs (including third-party UPFs) that each user traffic flow traverses. Such decision-making process not only considers MNO's knowledge on each user's network context (e.g., RAN conditions), but also takes into account OSPs' knowledge on each flow's specific application requirement and context. More specifically, such multi-dimensional context information are "translated" into network operating policies at the control-plane NFs, and are then disseminated and enforced at UPFs for SFC and traffic steering.

To the MNO, the direct benefit of enabling context-aware SFC is that it gains more control and flexibility over the steering and management of traffic flows that traverse its network, which leads to improved efficiency on network resource utilization. Also from a business perspective, this creates the opportunity of achieving a win-win situation amongst end-users (with better-assured QoE), OSPs (with more satisfied users and potentially more subscribers), and MNO (with increased revenue through leasing virtual resources). However, the technical challenge is that an SFC that consists of VNFs generally experiences degraded throughput and latency performance when compared to an SFC whose NFs are not virtualized [5], [6]. Also, different virtualization techniques also exhibit distinct performance patterns under different application scenarios [6]. Therefore, we derive further guidelines on the virtualization and deployment techniques of SFCs (with third-party UPFs) in 5G networks through extensive experiments in a real testbed network.

Our key contributions in this work are as follows:

- Based on current 3GPP 5G standards, we propose a system architecture to enable third-party stakeholders (e.g., OSPs) to deploy their own-operated VNFs within the MNO's 5G network infrastructure. This further enables the sharing of multi-dimensional context information among MNO and OSPs. Such context sharing was not possible in legacy mobile networks and paves the foundation of collaboration among MNO and OSPs regarding user-plane traffic handling and QoE assurance etc.
- We propose the first 3GPP-compliant scheme to enable context-aware SFC within 5G core network, where the MNO dynamically determines the optimal set of UPFs (including third-party UPFs) that each traffic flow should traverse. The decision-making process benefits from the proposed context sharing framework and considers both RAN conditions and application contexts. It improves the effectiveness and flexibility of user-plane traffic management for the MNO. Furthermore, it allows the MNO to retain necessary control over the OSP-operated VNFs, which addresses a MNO's key security concern.
- We have implemented the proposed scheme and systematically evaluated its performance on packet forwarding latency and throughput in a 3GPP Rel. 15-compliant 5G core network, through which we have verified its low performance overhead. Furthermore, we have established practical guidelines regarding the provisioning and deployment of SFCs involving third-party UPFs as well as the virtualization techniques involved, so that the overall performance are not significantly degraded.

II. Background

We first present in Figure 1 a simplified version of the 5G CN point-to-point architecture that is recently standardized in 3GPP TS 23.501.[1] All user-plane traffic flows are handled by one or more UPFs in their data path between UE (User Equipment) and DN (Data Network, e.g., public Internet). A UE is attached to the 5G network at the first UPF it encounters through RAN and N3 interface, which uses GTP (GPRS Tunneling) Protocol [7]). Connectivity to its downstream UPFs (if applicable) is over N9 interface (which also uses GTP as of 5G Phase 1 [8]). The last UPF in the data path connects to DN over N6 interface (i.e., TCP/IP). 3GPP has defined multiple UPF functionalities such as UE RAN connectivity anchoring point, gateway, packet routing, forwarding and inspection etc. They can be logically co-located within a single UPF or distributed among multiple UPFs. Note that although GTP has been used in mobile networks since 3G, its lack of support for SFC is recognized as one of its weaknesses. Therefore, for 5G Phase 2 and beyond, standardization groups such as IETF DMM [9] etc. are studying replacement protocols for GTP, where candidates include SRv6 (Segment Routing IPv6) [10] and LISP (Locator Identifier Separation Protocol) [11].

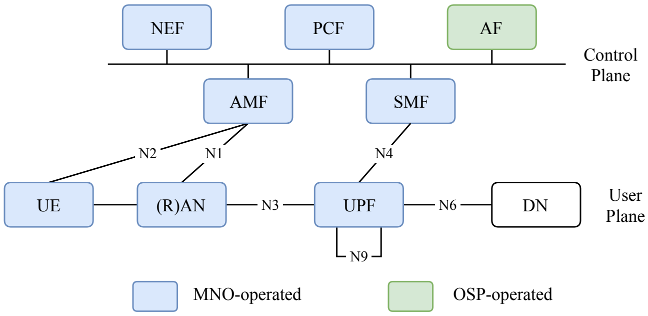


Fig. 1. Current 3GPP 5G system architecture (UPFs are operated by MNO only)

As shown in Figure 1, the control- and user-planes in 5G network are logically separated, and the user-plane operations are managed by policies that are established and updated by NFs at control-plane. The main control-plane NFs include AMF (Access and Mobility Management Function), PCF (Policy Control Function) and AF (Application Function). AMF specifies policies on radio resource management, access authentication and authorization, and mobility management etc. AF, which is operated by OSPs, provides contexts that are used to identify or classify each OSP's flows in the user-plane. These contexts are put into requests that are sent to PCF, where they are translated into user-plane policies that are interpretable by UPFs. During the process above, NEF (Network Exposure Function) acts as the bridge between AF and other contorl-plane NFs. The policies are sent to SMF for dissemination to relevant UPFs over N4 interface, which uses PFCP (Packet Filter Configuration Protocol [12]) in current 5G standards [8].

III. System Architecture Overview

Our proposed context-aware SFC framework is shown in Figure 2. There are two default UPFs that are at both ends of every flows' service function paths (SFPs), i.e., UPF (RAN) and UPF (Gateway). Between these two UPFs, we have defined a number of UPFs that *may* be involved in a flow's SFP depending on its contexts. The MNO-operated UPFs include UPF (OSP-Guard), UPF (PIT - packet inspection and treatment) and UPF (PEP - performance enhancement proxy) (more details in Section IV). The OSPs also deploy their own-operated UPFs with integrated intelligence, such as

[1] For simplicity, we omit some NFs and focus on the ones that are relevant to OTT applications.

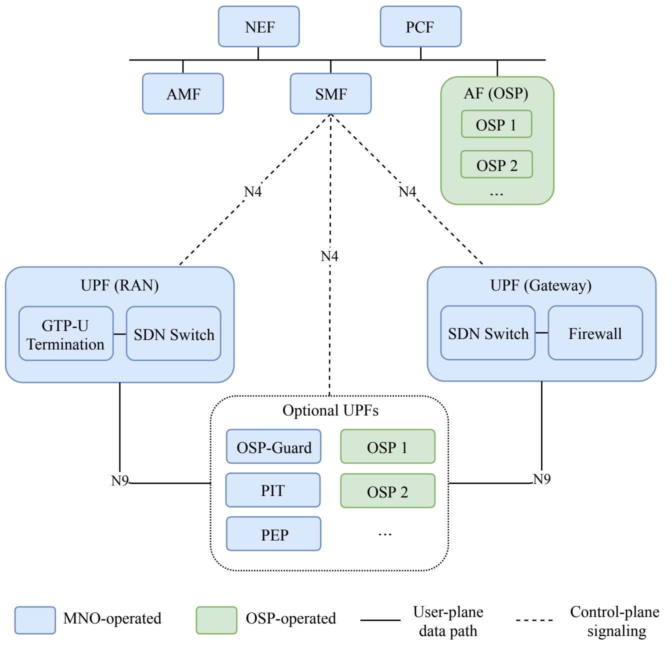


Fig. 2. Proposed SFC framework overview and architecture (note the OSP-operated UPFs)

application analytics or application performance enhancement logic (APEL). Typical APELs include TCP proxy, HTTP proxy [13], caching [3] etc. A UPF may include multiple APELs for different applications respectively.

UPF (RAN) acts as the anchoring UPF for all UEs and terminate their N3 interfaces. Its SDN switch component dynamically determines an SFP for each outbound flow based on its context while subjecting to traffic steering policies established at control-plane. **UPF (Gateway)** acts as the connection point between MNO's 5G core network and external DN. Flows that enter the core network from DN encounter UPF (Gateway) first, and its SDN switch component dynamically steer them towards different UPFs based on their policy-determined SFPs. UPF (Gateway) also contains a firewall component that examines inbound flows against MNO's security policies, as it is common practice for MNOs to impose stricter security on inbound flows than outbound ones. If a flow is deemed to be suspicious by the firewall, the MNO may perform additional actions on it, such as forwarding it to UPF (PIT) for (deep) packet inspection. The relevant UPFs are able to signal SMF to update relevant security policies (e.g., blacklist / whitelist) if necessary.

Note that the UPFs shown in Figure 2 are purely logical and are not bound to any hardware. The MNO or OSP may deploy or migrate them flexibly among network hardware. For example, UPF (RAN) and UPF (Gateway) can be collocated within the same hardware (e.g., SDN switch) with their own sets of inbound/outbound rules. Also, the MNO- and OSP-operated UPFs can be collocated within the same virtualization host, or they can be distributed among multiple hosts. As we will show in Section V, different deployment methods have significant impacts on the overall SFC performance.

In order to establish the policies above (e.g., flow classification, traffic steering etc.), the control-plane NFs need knowledge on multiple aspects, which include **network** context (each user's radio resource condition and mobility etc.), which are directly reported from UE and RAN to AMF over N2 and N1 interfaces respectively; **application-specific** context (an OSP's application-related information, e.g., server IP address, content popularities etc.), which are reported from the OSP's AF; and **user/session-specific** context, e.g., real-time QoE status which is reported from the OSP's UPF to its AF. As shown in Figure 2, 5G network's control-plane has a bus architecture, which makes it straightforward for a NF to share its knowledge with other NFs. For example, the OSP's AF can learn a user's RAN context from AMF via NEF, and AMF can learn a user's application context from that user's OSP's AF. Here, we envisage a few example scenarios which are only enabled through MNO-OSP context sharing.

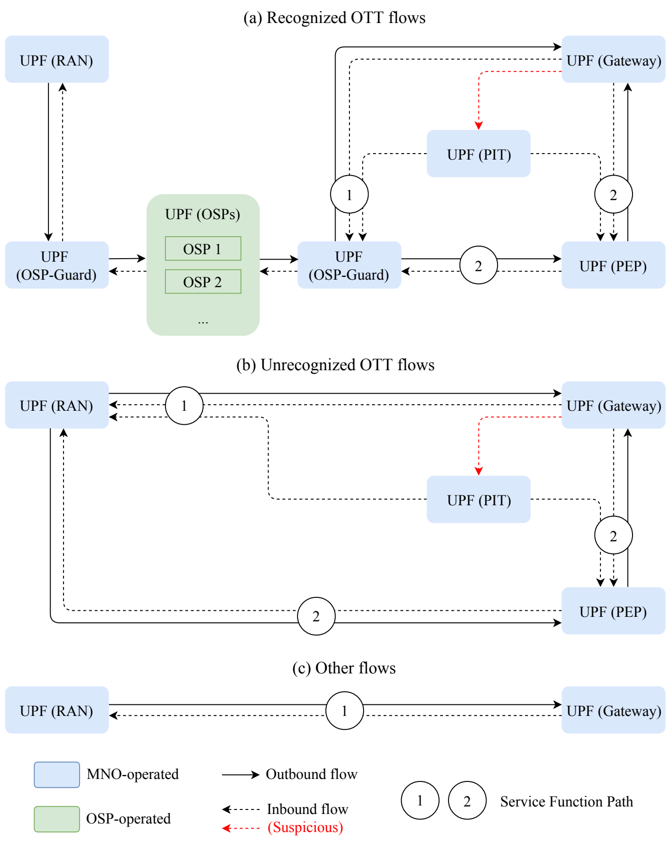
First, an OSP provides its server IP addresses (and port numbers) and the corresponding application type (e.g., webpage, video etc.) through AF to PCF, which uses them to establish policies that are disseminated to UPFs via SMF. For example, at UPF (RAN) and UPF (Gateway), policies are needed for flow classification and recognizing whether a flow belongs to any specific OSP. Furthermore, PCF needs to take these contexts into account to build SFP selection policies that are enforced at the two UPFs above, so that they can determine the specific set of UPFs that each flow traverses. More details on these policies' establishment are described in Section IV.

Second, the MNO provides (via AMF) each user's RAN context to relevant OSPs' AFs. AF will use this to gain insight into the user's RAN history, which enables e.g., prediction of the user's RAN downloading capability in the near future. The AF can then update its caching, prefetching or (video quality) adaptation policies accordingly at the OSP's UPFs. Note that such operations can be both in-band (i.e., through PCF and SMF) and out-of-band (i.e., directly between AF and UPF).

IV. User-Plane Service Function Chaining

A. *Flow Classification Criteria*

For each outbound flow, UPF (RAN) begins by examining its 5-tuple (source IP address & port number, destination IP address & port number, and the layer-4 protocol in use) and checking it against flow classification policies that were established by PCF. For example, it may examine the flow's destination port and check whether it is HTTP traffic (e.g., port 80 or 443). It may also examine e.g., the flow's packet arrival pattern to infer whether it belongs to an OTT application. If a flow is identified to be OTT traffic, UPF (RAN) further examines its destination IP address and checks whether it matches any known address that has been provided by any OSP through its AF. Depending on whether a match is found, the flow falls into either "recognized" or "unrecognized" categories. For inbound flows, UPF (Gateway) follows a similar approach as above and classifies inbound flows. As such, we establish three flow categories: a) recognized OTT flows; b)


(a) Recognized OTT flows

(b) Unrecognized OTT flows

(c) Other flows

Fig. 3. SFC templates based on flow categories

unrecognized OTT flows and c) other flows. For each category, we define an SFC template that is shown in Figure 3, where each SFC template includes a set of SFPs that flows may take.

B. Recognized OTT Flows

When an OTT flow's IP address belongs to an OSP that is recognized by the MNO, its SFP template is illustrated in Figure 3(a). By default, UPF(RAN) steers an outbound flow towards its respective OSP's UPF, where the flow will be handled by the OSP's APELs. Besides UPF(OSP), the flow's SFP may or may not involve UPF(PEP). For example, if an OSP's UPF already improves transport-layer performance, there is no need for that OSP's flows to traverse another PEP. The OSP can choose to share its UPF's performance enhancement measures through its AF to other control-plane NFs, so that the latter can take this into account when establishing traffic steering policies.

For inbound flows, since the flows are likely to come from trusted sources, it is expected that UPF(PIT) is excluded from most flows' SFPs. This is especially the case for encrypted traffic. Meanwhile, the SDN switch in UPF(Gateway) performs security inspection, marks suspicious flows and redirecting it to UPF(PIT).

All UPF(OSPs) are encompassed and monitored by two UPF(OSP-Guards). When OSPs rent virtual resources from MNO to deploy their UPFs, they need to establish service level agreements (SLA) regarding their performance and behavior. UPF(OSP-Guard) monitors each UPF(OSP)'s outbound traffic for SLA violations such as excessive number of parallel TCP connections or an abnormally high number of bursty short-lived sessions, which may be caused by poorly written APELs or a malicious UPF with compromised security. When a UPF(OSP-Guard) detects a UPF(OSP) violates its SLA, it may warn the OSP, block malicious traffic or even send feedback to AMF to terminate the UPF(OSP). Hence, UPF(OSP-Guard) can detect and isolate any misbehaving UPF(OSP)'s traffic and prevent it from compromising the performance and security of the rest of core network.

Note that based on Figure 3(a), for performance considerations, we recommend that each OSP should deploy at most one UPF(OSP), and the MNO should provision a virtualization host that is dedicated for third-party UPFs. These ensure that the UPFs benefit from advanced packet processing techniques such as DPDK (Data Plane Development Kit), which are verified by experiment results in the next section.

C. Unrecognized OTT Flows

In some cases, although a flow is classified as OTT traffic, its IP address is not recognized by the MNO regarding which OSP is it for. This can happen if the flow's OSP did not provide its application-specific context through its AF. For these flows, we include two representative UPFs in their SFPs, namely UPF(PEP) and UPF(PIT).

UPF(PEP) is a performance enhancement proxy that is operated by the MNO, and it can work at transport or application layer. At application layer, if the flow is unencrypted traffic, UPF(PEP) can perform typical caching operations. At transport layer, it can be a TCP proxy that provides generic TCP performance improvement, which also works for encrypted flows since it only needs TCP header information. The two techniques above complement each other and work on a best-effort basis. Note that UPF(PEP) is not necessarily in *every* flow's SFP. For example, the MNO may decide that only OTT sessions with large sizes (e.g., video files, not webpages) need to traverse UPF(PEP). These traffic steering policies are synchronized between UPF(RAN) and UPF(Gateway) through SMF, which ensures that each flow follows the same SFP on outbound and inbound directions.

All inbound flows are subject to security inspection at UPF(Gateway)'s firewall. During this process, a flow may be considered to be potentially harmful due to e.g., irregular packet arrival patterns or abnormally-high packet frequency. These flows, which are highlighted in red in Figure 3(b), are redirected to UPF(PIT) for additional *inspection* such as checking the flow's IP address against third-party IP reputation systems or deep packet inspection etc. Afterwards, UPF(PIT) will either decide the flow is safe and forward it onwards to the next-hop UPF, or mark the flow as malicious and perform *treatment* on it (e.g., terminating or applying throttling policies on the flow). UPF(PIT) is also able to send feedback to SMF regarding the flow source's safety or trustworthiness, so that relevant NFs in the control-plane can update their security policies accordingly.

D. Other Flows

As illustrated in Figure 3(c), all non-OTT flows' SFPs involve UPF(RAN) and UPF(Gateway) only. Such flows do not generally require high throughput or network intelligence, and basic UPF functions would suffice. Note that these flows (e.g., ICMP or DNS queries etc.) are generally short-lived and are often used in malicious Denial-of-Service attacks. It is the two UPFs' responsibilities to carry out basic security checks and perform actions (if necessary) on these flows while subjecting to the MNO's security policies. For example, UPF(Gateway)'s firewall and SDN switch can perform flow-based inspections. If a flow is identified to be malicious, it can be treated by UPF(Gateway). Furthermore, UPF(Gateway) may send feedback to SMF at control-plane to report the flow, where SMF can update its user-plane policy to block all future flows from/to the same IP address.

V. Performance Evaluation

As a proof-of-concept, we have implemented the proposed SFC framework in a local testbed as illustrated in Figure 4, which realizes two SFC templates in Section IV, i.e., recognized OTT and other flows. UPF(RAN) and UPF(Gateway) classify all flows with port numbers 80 or 443 into OTT flows. All uncategorized flows' SFPs include UPF(RAN) and UPF(Gateway) only. For OTT flows, their SFPs also include two UPF(OSP-Guards) with iptables firewalls, and an UPF(OSP) that is a transparent HTTP proxy.

We have carried out experiments under various scenarios. We use Cisco TRex load generator[2]'s provided *sfr_full* packet trace bundle to generate realistic Internet traffic with mixed packet sizes, which contains a mixture of UDP (36.6%) and TCP (63.4%) traffic. All TCP traffic are formed by HTTP/HTTPS flows. We use TRex's rate multiplier feature to linearly scale the generated load and create two load scenarios (light load and heavy load). Approximately 0.95Gbps and 4.4Gbps of overall traffic are generated under the two scenarios respectively. DPDK is enabled on the TRex machine.

We evaluate the following deployment schemes of two UPF(OSP-Guards) and UPF(OSP):

- **Bare-metal (BM)** - each UPF is deployed in a dedicated *physical* server, i.e., no virtualization is involved.
- **Distributed VM (VM-DIS)** - each UPF is deployed in a dedicated virtual machine (VM), and the three VMs are deployed in three separate OpenStack hosts.
- **Collocated VM (VM-CO)** - similar to VM-DIS, except the three VMs are collocated within one OpenStack host.

It is worth noting that in the three schemes above, all OpenStack hosts run OVS-DPDK to accelerate its packet forwarding. However, the acceleration takes place at the host's *physical* network port only. In the VM-CO scheme, traffic between UPF(OSP-Guard) and UPF(OSP) is forwarded *within* the OpenStack host, and hence is handled by conventional OVS and is not accelerated by DPDK.

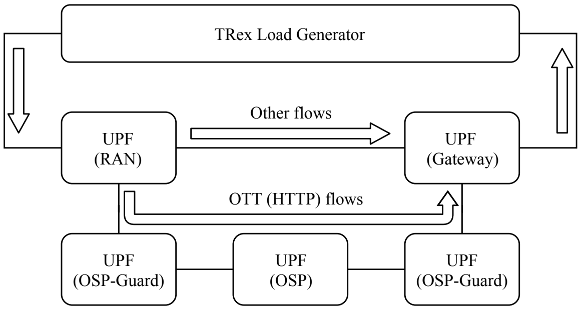


Fig. 4. Proof-of-concept testbed architecture

We first evaluate the schemes' throughput performance by examining the throughput of traffic that is successfully forwarded by the SFC framework. The 97-percentile results are plotted in Figure 5. In the light load scenario, VM-DIS and BM schemes have very similar throughput performance (with average of 931Mbps and 931.1Mbps). VM-CO achieved significantly lower mean throughput (209Mbps), which means only 22.4% of generated traffic was successfully forwarded. This is caused by OVS bridge's performance limitation and meets our expectation [5], [6]. A similar pattern is observed in the heavy load scenario, where BM and VM-DIS achieved mean throughput of 4.42Gbps and 4.4Gbps respectively, while VM-CO achieved only 943.9Mbps. These results verify that the VM-DIS deployment scheme matches bare-metal servers' SFC forwarding performance and does not cause any throughput bottleneck. In contrast, when UPFs are deployed under VM-CO scheme, almost 80% of the traffic is dropped due to OVS performance limitations.

We then evaluate the schemes' latency performance, i.e., the time duration that a packet spends in its SFP. The 97-percentile results are plotted in Figure 6. In the light load scenario, BM, VM-DIS and VM-CO produced similar latency performance with mean latencies of 0.11, 0.12 and 0.1ms. This is expected as the generated load is well within OVS's forwarding capability, so OVS does not cause any latency bottleneck. In the heavy load scenario, BM, VM-DIS and VM-CO achieved mean latencies of 0.15, 0.18 and 0.24ms and jitters of 0.05, 0.1 and 0.11ms. These results show that VM-DIS experiences slightly increased latency and jitter under heavier network traffic. This is because packets are still handled by VM's Linux kernel, and heavier traffic load leads to higher VM CPU usage and hence higher latency. The results also verify that under VM-CO scheme, OVS causes significant latency bottleneck when the network traffic exceeds its forwarding capability (˜1.5Gbps) [5], [6].

To summarize, the experiments above firstly demonstrated that our proposed SFC framework is able to achieve bare-metal-like throughput and latency performance in an NFV environment. Second, we have verified that the deployment method of UPFs in a virtualization network plays a crucial rule on the SFC performance. Specifically, we establish the follow-

[2] https://trex-tgn.cisco.com/

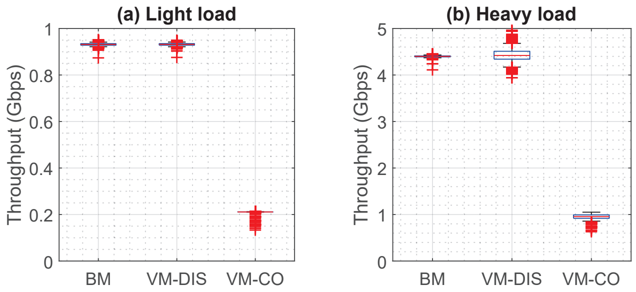


Fig. 5. SFC throughput performance

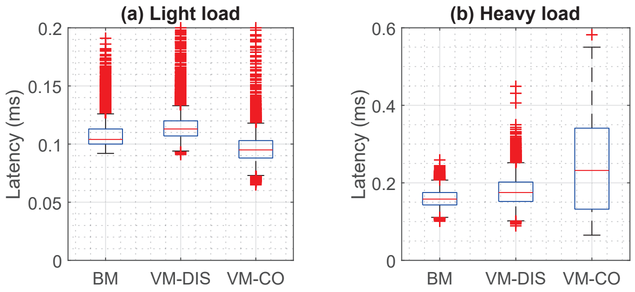


Fig. 6. SFC latency performance

ing recommendations on UPF deployment best-practices:

- In an OTT flow's SFP, UPF(OSP) should be deployed within a separate virtualization host (as in VM-DIS scheme). This assures the SFP's forwarding performance. From MNO's perspective, it may choose to deploy UPF(OSPs) from all third-party OSPs within the same virtualization host for e.g., centralized management. This ensures that any OTT flow's SFP's UPF(OSP) is deployed in an isolated host.
- In some cases, a UPF may not require high throughput (e.g., less than 200Mbps), but requires low forwarding latency for lots of bursty packets (such as ICMP or DNS queries). In these cases, OVS performance does not become the bottleneck, which means multiple UPFs can be collocated in the same virtualization host.

VI. Conclusion

In this work, we have proposed an SFC framework that is based on the recently standardized 5G network architecture. We have firstly extended existing standards to enable third-party stakeholders such as OSPs to deploy their own UPFs and embedded in-network intelligence, as well as necessary context-sharing operations between MNO and OSPs to enable context-aware operations. Next, we define three flow categories and specify how the flow classification takes place at MNO-operated UPFs. Three SFC templates were defined for these categories, and we have described in details the set of UPF paths within each SFC template that a flow may traverse, which is determined by MNO in real-time based on the flow's contexts.

We have implemented the proposed SFC framework in a virtualized testbed network. We have demonstrated the framework's traffic handling capability, which matches a non-virtualized system with the same SFC templates. Furthermore, we have verified that the deployment method of UPFs have crucial impacts on the resulting SFC's performance. For optimal SFC performance, each UPF should be deployed within a separate virtualization host. Meanwhile, same-chain UPFs that are collocated within the same host can only handle a limited amount of traffic. This leads to our recommendations on UPF deployment, i.e., for each OSP, its own-operated UPF should be deployed within a separate host and should be isolated from other MNO-operated UPFs in the chain.

VII. Acknowledgments

This work is funded by EPSRC NG-CDI project (EP/R004935/1). The authors would also like to acknowledge the support of the University of Surrey's 5G Innovation Centre (5GIC) (http://www.surrey.ac.uk/5gic) members for this work.

References

[1] "Cisco VNI Mobile Forecast (2016-2021)." [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.html

[2] C. Ge, N. Wang, G. Foster, and M. Wilson, "Toward QoE-assured 4K video-on-demand delivery through mobile edge virtualization with adaptive prefetching," *IEEE Transactions on Multimedia*, vol. 19, no. 10, pp. 2222–2237, Oct 2017.

[3] C. Ge, N. Wang, W. K. Chai, and H. Hellwagner, "QoE-assured 4K HTTP live streaming via transient segment holding at mobile edge," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 8, pp. 1816–1830, 2018.

[4] 3GPP. TS 23.501 - system architecture for the 5G system. [Online]. Available: http://www.3gpp.org/DynaReport/23501.htm

[5] R. Kawashima, H. Nakayama, T. Hayashi, and H. Matsuo, "Evaluation of forwarding efficiency in NFV-nodes toward predictable service chain performance," *IEEE Transactions on Network and Service Management*, vol. 14, no. 4, pp. 920–933, Dec 2017.

[6] N. Pitaev, M. Falkner, A. Leivadeas, and I. Lambadaris, "Characterizing the performance of concurrent virtualized network functions with OVS-DPDK, FD.IO VPP and SR-IOV," in *Proc. ICPE*. Berlin, Germany: ACM, 2018, pp. 285–292.

[7] 3GPP. TS 38.300 - NR and NG-RAN overall description. [Online]. Available: http://www.3gpp.org/dynareport/38300.htm

[8] ——. TR 29.891 - 5G system - phase 1 CT WG4 aspects. [Online]. Available: http://www.3gpp.org/dynareport/29891.htm

[9] S. Homma, T. Miyasaka, S. Matsushima, and D. Voyer, "User Plane Protocol and Architectural Analysis on 3GPP 5G System," IETF, Internet Draft, Oct. 2018. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-hmm-dmm-5g-uplane-analysis-02

[10] C. Filsfils, P. C. Garvia, J. Leddy, D. Voyer, S. Matsushima, and Z. Li, "SRv6 Network Programming," IETF, Internet-Draft, Oct. 2018. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-filsfils-spring-srv6-network-programming-06

[11] D. Farinacci, V. Fuller, D. Meyer, and D. Lewis, "The Locator/ID Separation Protocol (LISP)," RFC 6830, Jan. 2013. [Online]. Available: https://rfc-editor.org/rfc/rfc6830.txt

[12] 3GPP. TS 29.244 - interface between the control plane and the user plane nodes. [Online]. Available: http://www.3gpp.org/dynareport/29244.htm

[13] P. Qian, N. Wang, B.-H. Oh, C. Ge, and R. Tafazolli, "Optimization of webpage downloading performance with content-aware mobile edge computing," in *Proc. MECOMM*. Los Angeles, USA: ACM, 2017, pp. 31–36.