Remote Orchestration of NextG Services Across the Global Internet: An Experimental Study

Shalini Choudhury
Prasad Netalkar
WINLAB, Rutgers University
North Brunswick, NJ, USA
{shalini, pnetalka}@winlab.rutgers.edu

Frank Slyne
Diarmuid Collins
Sumaiya Binte Ali
Daniel Kilper
CONNECT Centre, Trinity College
Dublin, Ireland
{fslyne, dcollin5, alisu, dan.kilper}@tcd.ie

Ivan Seskar
Dipankar Raychaudhuri
WINLAB, Rutgers University
North Brunswick, NJ, USA
{seskar, ray}@winlab.rutgers.edu

***Abstract*—This paper presents results from an experimental study of next generation services delivered over the global Internet. The experimental platform used for this study is the "COSM-IC" federated testbed which connects the COSMOS 5G/6G edge deployment at Rutgers and Columbia Universities with other international testbeds, including the "Open Ireland" network at Trinity College, Dublin, the CQPD network in Brazil and the JGN testbed in Japan. The experiments reported here use two sites (COSMOS at Rutgers and Open Ireland at Trinity) to conduct experiments aimed at understanding the performance of emerging latency-critical applications from a remote server. Baseline application latency and throughput performance are reported for a file server in Open Ireland and a client in the COSMOS network, showing significant degradation due to the large RTT (112 ms) associated with international connections. Techniques for ameliorating these performance problems are introduced, first at the transport layer using improved versions of TCP, including TCP BBR and a new cross-layer protocol called mmCPTP. A second set of experiments was conducted to evaluate the effectiveness of Container Migration in which application code (service) is dynamically migrated to an edge cloud close to the target user. Significant performance gains in service response time are shown experimentally due to service migration using enhanced transport protocols on the high RTT international connections.**

***Index Terms*—Testbed, global experiments, 5G, edge cloud, service migration, TCP**

I. Introduction

The emergence of 5G networks is expected to result in an increase in time-critical and bandwidth intensive applications delivered from Internet servers to mobile clients [1]. Applications in this category include augmented and virtual reality involving large data transfer and machine learning computation for image or natural language processing in real-time. An important technical challenge associated with such ML-based mobile application scenarios is that of minimizing the network's end-to-end latency. Latency reduction is being addressed at various levels via the introduction of the URLLC (ultra-reliable low latency communication) as a 5G transmission mode, along with the provisioning of edge cloud computing capacity in the proximity of the mobile user. URLLC networking with collocated edge computing will certainly help with the delivery of real-time services, but there is still the basic problem of adapting the ubiquitous client-server model for Internet applications to work well for servers across the globe. Today's popular Internet service, such as video streaming, have addressed this performance problem with content-delivery networks (CDN) [2], which provide in-network caching to reduce service latency and network traffic. For services other than content delivery, the other option is to provision cloud computing capacity in the general region where the service is to be delivered. Both options are useful for established service providers with sufficient traffic volume to amortize the cost of CDN or cloud usage fees, but these solutions may be too costly for niche services catering to small numbers of users. The availability of edge-cloud infrastructure associated with 5G opens up another option for small-scale services by making it possible to migrate computing on demand from remote servers to closer to the target users.

This research was supported in part by NSF grant #2029295, Science Foundation Ireland, CONNECT Centre under Grant 13/RC/2077 P2.

TABLE I
ROUND TRIP TIME BETWEEN TRANSATLANTIC SERVERS.

UE Location	Server Location	RTT (ms)	Network Type
COSMOS, NY	Open Ireland, IE	112	Research
COSMOS, NY	AWS, IE	90	Public

Applications such as automatic license plate recognition (ALPR) [3] consider latency as the primary performance metric. Accessing similar applications remotely violates the latency threshold and degrades application goodput, defined as the number of useful (on time) bits per second delivered to user equipment (UE) running the application. Additionally, with high bandwidth delay product (BDP) links, attaining high throughput becomes particularly challenging since high bandwidth and delay make it difficult to saturate the link with data, leading to lower effective throughput.

Based on the above motivation, this work attempts to investigate the impact/limitations of low latency and high bandwidth demanding applications from a remote server by setting up a global wide area experiment where the ALPR service is hosted in a remote server at the Open Ireland testbed in Ireland [4] and is accessed from the COSMOS testbed in USA [5],

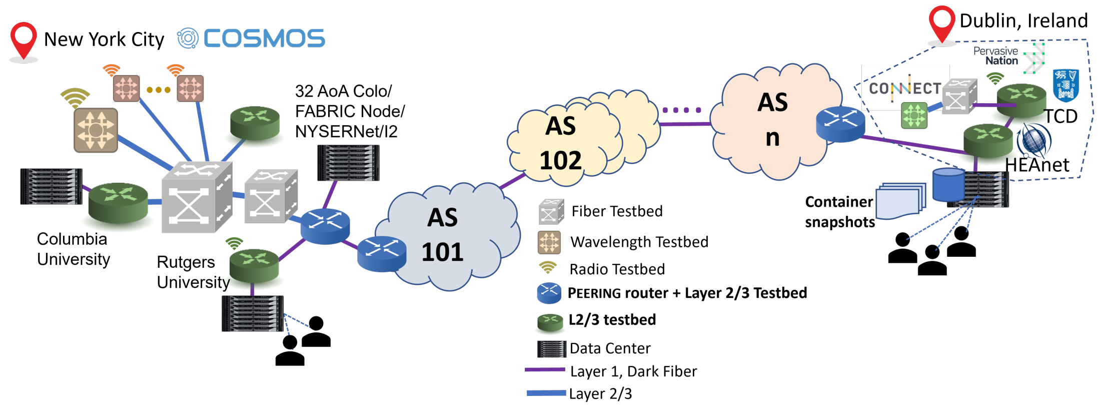

Fig. 1. COSMOS-Open Ireland Transatlantic Research Testbeds

demonstrating a link delay of 112 ms, as shown in Table I. The testbeds (seen in Figure 1) are connected physically through national research networks such as the HEAnet in Ireland, FABRIC in the USA, and the pan-European GÉANT network and offer a new transatlantic research partnership designed to allow researchers to conduct real-life, wide-area experiments. Specific contributions of this work are as follows:

- Analyze the performance of a low-latency use case within a wide-area network environment and aim to elucidate the interactions and potential performance bottlenecks that emerge in this context.
- Provide perspectives on enhancing network performance for latency-critical applications by exploring strategies like the implementation of a cross-layer transport protocol and service migration with edge clouds.
- In this study, we leverage COSMIC, an international experimental platform enabling peering between global research testbeds like COSMOS (USA) and Open Ireland (Ireland). This wide-area testbed serves as a realistic setting for our proposed examination of a global scale, low-latency use case.

The rest of the paper is organized as follows. Section 2 discusses related work and Section 3 reports the challenges of accessing applications remotely. Section 4 enlists approaches for seamless service orchestration, and Section 5 elaborates on the COSMIC testbeds. Sections 6 and 7 present the cosmic benchmark results and future work, respectively. Section 8 concludes the paper.

II. Related Work

TCP performs poorly in a number of scenarios, including delivery of files/services over a high BDP link [6]. Several congestion control algorithms have been proposed, particularly targeting high BDP links such as HighSpeed TCP [7], H-TCP [8], etc. Yet, they perform poorly over certain wireless channels [9] resulting in high latency and slow recovery. TCP BBR, recently proposed by Google [10] operates by estimating bottleneck bandwidth and RTT. It enters a state called "probeRTT" having a smaller congestion window (several kB) periodically to measure real-time RTT. However, TCP BBR approach poses challenges for streaming applications and internet traffic due to the variability in RTT.

To mitigate the impact of high BDP links on conventional TCP, service migration [11] to an edge cloud server, geographically closer to the user, is performed [12]. Service migration can be classified into stateful and stateless migrations [13]. However, in either of the migration approaches, the container in the source node should record very small downtime [14].

Further, the COSMOS testbed has been used to demonstrate proof of concept experiments including smart city intersection [15], studying dynamic spectrum access [16] and virtual reality experiments [17]. Additionally, CONNECT Centre OpenIreland testbed was used in experiments to analyze the coexistence of digital coherent and analog radio over fiber signals [18] and generalized signal-to-noise ratio (GSNR) estimation error during service characterization [19]. This work includes early experimental results from a collaborative project between the COSMOS and Open Ireland teams.

III. Challenges in Remote Access of NextG Applications

Next generation applications require high throughput and low latency simultaneously. If applications are accessed remotely over a high BDP link, such as the transatlantic link, the TCP eventually becomes a performance bottleneck, and the low latency requirement is violated due to the wide area delay. This section lists the limitations of availing interactive and latency-sensitive services from remote servers.

TABLE II
Application Performance over High BDP Link (BW = 10 Gbps).

File Size (MB)	Response Time (sec)		
	TCP Cubic	TCP BBR	UDP
125	4.61	1.37	0.50
175	5.17	1.48	0.67
200	5.51	1.57	0.74
225	6.64	1.80	0.84

A. TCP performance over high bandwidth delay product link

Application response time: Consider the scenario where a user based in New York intends to access a file that is hosted on a remote server located in Ireland. The file sizes range between $\approx$ 125 to 225 MB. Table II lists the delay for the data transmitted over the high bandwidth transatlantic link between COSMOS and Open Ireland testbeds characterized by a high propagation delay of 112ms. The application response delay is in the order of seconds due to the high BDP link, which is a significant limitation of the TCP protocol. TCP slow start compromises the peak utilization of the high bandwidth transatlantic link for both TCP CUBIC and BBR congestion control mechanisms. Our observation indicates that UDP exhibits a faster response (in the order of milliseconds). However, this expedited response time comes at the cost of an increased packet loss rate (refer to Table III), indicating that while UDP may enhance time-sensitive data transmission, it might concurrently compromise data reliability due to its connectionless nature.

TABLE III
JITTER/LOSS MEASUREMENTS WITH UDP.

File Size (MB)	Jitter (ms)	Loss%
125	0.068	8.5
175	0.013	7.5
200	0.041	5.8
225	0.041	7.8

Jitter and loss: In the context of low latency interactive applications, jitter has a substantial impact on user experience and can even lead to physical discomfort or symptoms of motion sickness in users. From Table III, it can be seen that the remote server in Ireland sends a stream of files of size ranging between 125 MB - 225 MB to a server in the COSMOS testbed over the transatlantic link. The jitter recorded in this scenario varies between 13 - 68 microseconds which are quite low and tolerable for most next-gen applications' QoS requirement point of view. Other factors, such as overall network latency, packet loss rate, and application's tolerance for latency variations, also play a role. For real-time applications such as AR/VR or ALPR, it is desirable to have close to 0% packet loss. This is because packet loss directly leads to degraded quality of experience (QoE), causing video freeze, pixelation, or audio dropouts and significant disruptions to the user end experience. In this scenario, packet loss observed for file transmission ranges between 5.8% - 8.9%, which is $\approx$6x - 8x more than the acceptable packet loss rate for interactive applications. It can be stated that network infrastructures with multi-Gbps delivery capacity provide only a rate of a few Mbps over cross-country and intercontinental distances.

B. Transport protocol limitations

The transport protocol plays a key role in delivering data packets reliably and improving application performance. In the first set of experiments, we validate the performance of TCP on the COSMIC [20] testbed.

From Table IV, considering TCP Cubic, the E2E throughput (100 s interval) between COSMOS and Open Ireland testbed is observed to be 270 Mbps, whereas with TCP BBR the throughput is found to be 640 Mbps over a link delay of 112ms. A standard Linux package tool *tc* is used to limit the bandwidth and introduce any loss (at the intermediate router). Our topology consisted of two COSMOS nodes to emulate UE (User Equipment) and BS (Base Station) and the Open Ireland edge cloud node representing a remote file server, as shown in Figure 2. With just 0.1 % loss, TCP performs so poorly over this high BDP link, further motivating a need for a better transport protocol to overcome loss and fluctuations in the wireless/high BDP links.

To meet the stringent requirements of latency and bandwidth of next-gen applications, several strategies can be considered. For instance, one approach involves migrating the service closer to the end user's location, enabling them to access it from a local edge platform. This strategy aids in reducing latency, avoiding backhaul bandwidth bottleneck and improving the user experience. In addition, if the end-user connects via a wireless network characterized by variable bitrates and lossy channels, using a cross-layer aware transport protocol, such as the one discussed in [21] and [22], can further enhance performance. These strategies of service migration and cross-layer transport protocol not only optimize network resources but also ensure the application deadlines are met.

TABLE IV
TCP THROUGHPUT EVALUATION UNDER LOSS ON COSMIC TESTBED.

Transport Schemes (Throughput in Mbps)	Loss percentage		
	0%	0.1%	0.5%
TCP New Reno	206	4.89	3.98
TCP Cubic	270	9.42	2.64
TCP BBR	640	373	323

IV. APPROACHES FOR SERVICE ORCHESTRATION

As mentioned earlier, the use of CDN or commercial cloud services may be too expensive for a niche service delivered from a site across the globe. Therefore, to overcome this limitation, the proposed approaches are: (I) overcome the transport layer bottleneck due to the high BDP link by leveraging a cross-layer pull-based transport protocol or (II) push remote content and services at an edge cloud infrastructure closer to the end user's current location.

A. Cross-layer transport protocol

To support remote delivery of latency critical services, the transport protocol should support intermittent wireless channels, seamless mobility of end users, and better mechanisms to address congestion control and packet loss. Hence, to cater to these requirements, mmCPTP is used in this work [21]. The mmCPTP protocol is specifically designed to address the challenges associated with high BDP intermittent wireless channels. The design principle includes:

Pull-based flow and congestion control: In order to mitigate the impact of fluctuations and path switches, a unique pull mechanism is employed to retrieve packets from a file server located in close proximity to the user. Unlike traditional methods that involve probing, the packets are pulled based on feedback received from the lower layer, specifically the RLC/MAC buffer information. This approach eliminates the need for active probing and instead relies on the pull mechanism guided by lower-layer feedback.

Separate packet loss and congestion control: In order to address the challenges encountered by TCP for packet loss events and to prevent any negative impact on higher layers, a distinct mechanism is used for loss recovery and congestion control. The lower layers of the protocol stack, such as the link and MAC layers, handle loss and the transport layer is solely responsible for managing flow and congestion control.
Split connection: To reduce latency and improve throughput efficiency in situations where the content (file-server) experiences considerable round-trip time with the user, the link is split between wired and wireless segments using a ”proxy”. In doing so, the backhaul network with a potentially high BDP is effectively utilized, thereby preventing the sender from limiting the transmission rate due to frequent interruptions. Consequently, it also prevents end-to-end retransmissions and reduces latency.

B. Migration to edge infrastructure

Dynamically relocating application services closer to the user's geographical location is accomplished through container migration - a process by which an active container running an application service is seamlessly transferred from one host to another [12]. To migrate a container, this work adopts a three-phase approach based on Linux scripts:
1. Pre-copy: Prior to migrating the container hosting application, memory snapshots are created. Once the migration request is triggered, these snapshots are sent serially to the destination node. The snapshots are realized by Checkpoint/Restore In Userspace (CRIU), an open-source Linux tool that allows to freeze an application and checkpoint it to persistent storage as a collection of files. The container can resume its function in the source node after the snapshot is created.
2. State transfer: The above process continues until the container's minimum number of snapshots at different time stamps are checkpointed and sent to the destination node.
3. Post-copy: Once all the snapshots are received by the destination node, the container is restored with the most recent runtime state, following which the user starts receiving service from the geographically close destination node.

Strategic migration obviates the need for the user to access the service from a distant location, thereby circumventing substantial propagation delay. The intention of this maneuver is to improve the QoE by reducing latency and enhancing the responsiveness of the application.

V. COSMOS and Open Ireland testbed Architecture

This paper investigates high-performance global connectivity through a federation of COSMOS with interconnections to Open Ireland. Wide area inter-continental experiments are conducted on the COSMOS and Open Ireland testbeds, with network traffic tunneled through the ASes.

A. COSMOS testbed

To support seamless orchestration of remote applications in the end-user network domain, COSMOS offers experimenters mmWave communications, wideband radio signal processing (500MHz or more), effective virtualization of radio resources, low latency front and back-haul, and tightly coupled edge cloud. COSMOS testbed is deployed in West Harlem (New York City) as part of the NSF Platforms for Advanced Wireless Research (PAWR) program. For replicating the remote service in a computing platform, COSMOS provides integrated edge cloud technology, including commodity CPUs/GPUs/FPGAs, for achieving computing speeds needed to support cloud radio access networks (C-RANs), network function virtualization (NFV), and low-latency cloud applications. Therefore in the context of remote service orchestration, the experimenter can select the edge cloud based on the application's compute requirement and consequently initiate migration to the selected edge cloud. COSMOS also integrates advanced optical switching technology based on wavelength division multiplexing (WDM) switch fabrics and radio over fiber (RoF) interfaces for low latency connections to edge and central clouds.

B. Open Ireland testbed

The OpenIreland testbed is an adaptable 5G and beyond radio, optical, and cloud testbed centered at CONNECT Centre, Trinity College Dublin. This testbed offers technologies, including virtualized 5G radio, optical transmission equipment, Cloud-RAN and Network Functions Virtualisation (NFV) technologies. The testbed encompasses indoor and outdoor 5G radio, cloud, and optical transmission equipment situated within Trinity College Dublin and extending through regional and metro dark fiber across Ireland. The testbed is equipped to support research towards the combination of SDR and radio slicing and physical layer approaches into open and coherent next-generation commercial networks. These activities, combined with experiment automation techniques, enable experiment reproducibility. The OpenIreland testbed extends optical capabilities and full reconfigurability to experimenters through the use of a large port count optical fiber switch, which manages the topology for any experiment. Optical and Wireless equipment is connected to a private computational cloud orchestrated primarily by technologies including MAAS (Metal as a Service), containers (Docker, LXD), cloud platforms (OpenStack), and operating systems (Ubuntu), enabling OpenIreland to easily build and deploy multiple isolated testbed environments supporting dynamic virtualized experiments.

VI. COSMIC experimental results

In this work, we conducted the following experiments to address remote orchestration of services:
(a) The first set of experiments was performed to address the impact of TCP bottleneck for accessing application over high BDP link. In this context, a cross-layer pull based transport solution mmCPTP was used that fully exploits the high bandwidth of the link and improves throughput performance.
(b) The second set of experiments includes pushing the service to an edge cloud close to the end user's location to address the high propagation delay. The container hosting the ALPR service is migrated from Open Ireland edge server to the

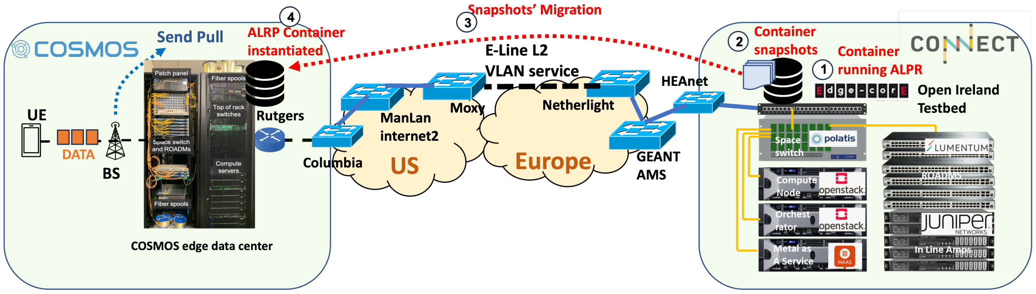

Fig. 2. Experimental set-up between COSMOS-Open Ireland testbeds

COSMOS edge cloud node improving the service response time and quality-of-experience. Figure 2 represents a wide-area experimental set-up between COSMOS and Open Ireland testbed to implement mmCPTP and perform service migration.

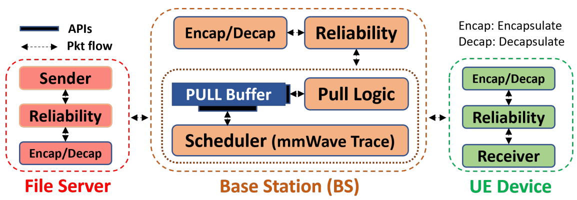

Fig. 3. Click element graph of mmCPTP [21].

A. Cross-layer transport protocol

To improve the network performance, mmCPTP using Click modular software is implemented as shown in Figure 3. The pull buffer at the BS periodically pulls the data based on buffer occupancy (PullLogic) from the file server and uses the Scheduler element to push data to the UE. Referring to Figure 2, the file server (edge server) is in the Open Ireland testbed, while UE and BS are located in COSMOS.

We evaluate the file transfer size over the mmWave channel on the COSMIC platform using the trace as shown in Figure 4, but max throughput is capped at 500 Mbps for a fair comparison between the protocols. The click implementation of mmCPTP is used and from Table V, we see that mmCPTP achieves a file transfer size close to the theoretical limit reporting a gain of 5.6x compared to TCP Cubic over highly intermittent mmWave channel. Future evaluation on the COSMIC includes integrating mmCPTP on the 5G NR SDR System (such as Amarisoft) and validating the E2E performance using outdoor field trials.

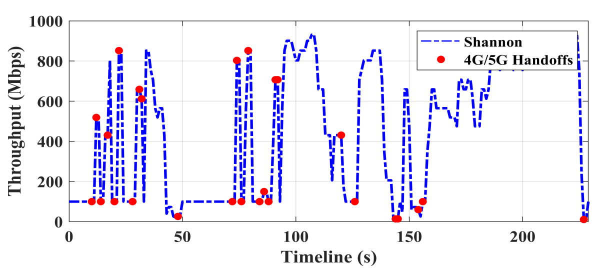

Fig. 4. [23] 4G/5G NR Shannon throughput

TABLE V
FILE TRANSFER OVER MMWAVE CHANNEL ON THE COSMIC TESTBED.

Transport Schemes	File transfer size
TCP New Reno	514 MB
TCP Cubic	1.67 GB
TCP BBR	5.53 GB
mmCPTP	9.46 GB
Theoretical limit	9.47 GB

TABLE VI
EVALUATING ALPR SERVICE MIGRATION (BW = 1 GBPS).

Container size	Transfer time	Container restore time
2.0 GB	20.75 sec	8.67 sec
2.9 GB	28.65 sec	11.39 sec
4.2 GB	41.55 sec	13.84 sec

B. Service Migration

Migrating ALPR container from Open Ireland: As seen in Figure 2, ALPR application is operational within a container located on the edge server in CONNECT Centre. When container migration is initiated, the first task is to capture snapshots of the running container. The collection of snapshots encompasses the entirety of the container's state. Followed by transferring snapshots from Open Ireland by tunneling them through the VLAN to the COSMOS edge cloud platform. This is the **transfer time** of the container. Finally, when all snapshots have reached the COSMOS edge server, the container is instantiated at the new location, incurring **container restore time**. The state of this container is subsequently resumed to match that of the original container in the Open Ireland edge node. From Table VI, it is seen that migrating containers of size 2 - 4.2 GB incurs a transfer time of 20.75 - 41.55 secs. Additionally, the container restore time has been recorded within a range of 8.67 - 13.84 seconds. The delay in transfer time principally originates from the link's high capacity and significant latency attributes. However, the incurred delay due to transfer time and container restore time is typically a one-time cost. Once the container is successfully migrated and launched in the new location closer to the user, the service response time is significantly reduced, as shown in Table VII.

Access ALPR from Open Ireland vs COSMOS server: Table VII draws a comparison of E2E delay when ALPR application is accessed remotely (Open Ireland server) vs. when the application is migrated near to the end user location and accessed locally (COSMOS edge server). The E2E latency includes communication and computing delays for Frames 1-

5. When remotely accessing the ALPR application, the user in the COSMOS region sends video frames 1-5 (sizes 166-187 KB) to the Open Ireland edge server, where Optical Character Recognition (OCR) is performed on the frames, and the result is sent back to the end user. For locally accessing the ALPR service from the COSMOS edge server, we emulate the UE by a COSMOS node with a link delay of 3 ms and bandwidth of 10 Gbps to the edge cloud in the COSMOS testbed. The UE sends video frames 1-5 as requests to the COSMOS edge server. The containerized ALPR application hosted on the COSMOS edge server runs OCR on the received frames and sends responses back to the end user. As seen from Table VII, the E2E delay recorded when accessing ALPR remotely clearly violates the 100ms latency threshold of the application, exhibiting service response delays of ≈300ms. Locally accessing the ALPR application in the end user's vicinity (in COSMOS) mitigates the high propagation delay, reducing the application response time approximately threefold to <100ms.

TABLE VII
END-TO-END (E2E) DELAY OF ACCESSING ALPR SERVICE FROM OPEN IRELAND (OI) VS COSMOS EDGE SERVER.

Request type	E2E delay at OI	E2E delay at COSMOS
Frame 1	307ms	84ms
Frame 2	296ms	56ms
Frame 3	310ms	90ms
Frame 4	305ms	87ms
Frame 5	292ms	51ms

VII. CONCLUSION AND FUTURE WORK

This paper analyzes the potential impact of accessing a latency-critical application from a remote location via global experiments conducted on the COSMIC platform. Benchmarking experiments were performed for accessing files over a high-bandwidth, high-delay link using both TCP and UDP protocols. Our findings showed that on a transatlantic link, TCP Cubic had a significant delay and was highly sensitive to a small packet loss of 0.1% in an intermediate router. In contrast, UDP significantly reduced the video access delay in the order of milliseconds but experienced a high packet loss rate. Strategies were introduced to improve these performance issues by utilizing TCP BBR and a novel cross-layer protocol named mmCPTP at the transport layer. Subsequently, a series of experiments were performed to assess the efficiency of service migration. This technique involved dynamically relocating the application to an edge cloud located near the target user. By leveraging mmCPTP the achieved file transfer size was 5.6x more than TCP CUBIC over mmWave channel in COSMIC. Enabling service migration to an edge compute platform closer to the end user's location, the application response time was reduced to less than 100ms.

In the future, we will extend this work to explore the implementation of EIR [24] to utilize (i) information about internal topology in terms of aNodes and vLinks (e.g., internal structure of the AS), and (ii) more metrics about speed and delay of each link, for informed interdomain routes based on criteria like high bandwidth, low latency, etc. The benefits of link performance-aware inter-domain routing with EIR motivates experimentation on cross-layer control. In particular, transmission bottlenecks at either edge network of the transatlantic link, might be addressed through adapting the capacity or routing of optical signals in the metro fiber networks, potentially across domains [25]. Through SDN control, EIR can be used for optical network adaptation to further speed up transfer times.

REFERENCES

[1] Nasrallah et al. "Ultra-low latency (ull) networks: The ieee tsn and ietf detnet standards and related 5g ull research," IEEE Communications Surveys & Tutorials, 2018
[2] Su et al. "Drafting behind akamai: Inferring network conditions based on cdn redirections," IEEE/ACM transactions on networking, 2009.
[3] Chang et al. Automatic license plate recognition. IEEE transactions on intelligent transportation systems, 5(1):42–53, 2004.
[4] Raychaudhuri et al. Challenge: Cosmos: A city-scale programmable testbed for experimentation with advanced wireless. In Proceedings of the 26th Annual International Conference on Mobile Computing and Networking, pages 1–13, 2020.
[5] Open ireland testbed (connect), funded by science foundation ireland www.openireland.eu, 2023-06-27.
[6] Yorozu et al. "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987.
[7] Sally Floyd. Rfc3649: Highspeed TCP for large congestion windows, 2003.
[8] Leith et al. "H-TCP: TCP for high-speed and long-distance networks," In Proceedings of PFLDnet, 2004
[9] Zhang et al. " Will tcp work in mmwave 5g cellular networks?" IEEE Communications Magazine, 2019.
[10] Cardwell et al. "BBR: Congestion-based congestion control: Measuring bottleneck bandwidth and round-trip propagation time," Queue, 2016.
[11] Puliafito et al. "Container migration in the fog: A performance evaluation," Sensors, 19(7):1488, 2019.
[12] Govindaraj et al. "Container live migration for latency critical industrial applications on edge computing," In 2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA), volume 1, pages 83–90. IEEE, 2018.
[13] Machen et al. "Live service migration in mobile edge clouds," IEEE Wireless Communications, 25(1):140–147, 2017.
[14] Mirkin et al. "Alexey Kuznetsov, and Kir Kolyshkin. Containers checkpointing and live migration," In Proceedings of the Linux Symposium, volume 2, pages 85–90, 2008.
[15] Kostić et al. "Smart city intersections: Intelligence nodes for future metropolises," Computer, 2022.
[16] Stojadinovic and Netalkar et al. "A spectrum consumption model-based framework for dsa experimentation on the cosmos testbed," In Proceedings of the 15th ACM Workshop on WiNTECH, 2022.
[17] Choudhury et al. "Experimental evaluation of virtual reality applications running on next-gen network scenarios with edge cloud assistance," In Proceedings of the 16th ACM Workshop on WiNTECH, 2022.
[18] Slyne et al. "Experimental demonstration of network convergence with coherent and analogradio-over-fibre signals for densified 5.5 g/6g small cell networks," arXiv preprint arXiv:2302.04915, 2023
[19] Kaeval et al. "Employing channel probing to derive end-of-life service margins for optical spectrum services," Journal of Optical Communications and Networking, 15(7):C63–C73, 2023.
[20] Cosmos interconnecting continents (COSM-IC). https://www.cosmos-lab.org/cosmos-interconnecting-continents-cosm-ic/, 2023.
[21] Netalkar et al, Jiachen Chen, and Dipankar Raychaudhuri. "mmCPTP: A cross-layer pull based transport protocol for 5g mmwave networks," In NOMS IEEE, 2023.
[22] Ajay Bakre and BR Badrinath. I-tcp: Indirect tcp for mobile hosts. In Proceedings of 15th ICDCS, pages 136–143. IEEE, 1995.
[23] Narayanan et al. "Lumos5g: Mapping and predicting commercial mmwave 5G throughput," In Proceedings of the ACM Internet Measurement Conference, 2020.
[24] Mukherjee et al. "Eir: Edge-aware inter-domain routing protocol for the future mobile internet," Computer Networks, 127:13–30, 2017.
[25] Li et al. "tsdx: Enabling impairment-aware all-optical inter-domain exchange," Journal of Light-wave Technology, 36(1):142–154, 2018.