Orchestrating Virtualized Core Network Migration in OpenROADM SDN-Enabled Network

Shunmugapriya Ramanathan$^{\#}$, Koteswararao Kondepu$^{\$}$, Tianliang Zhang$^{\#}$, Behzad Mirkhanzadeh$^{\#}$,
Miguel Razo$^{\#}$, Marco Tacca$^{\#}$, Luca Valcarenghi$^{\bullet}$, and Andrea Fumagalli$^{\#}$
$^{\#}$Open Network Advanced Research lab, The University of Texas at Dallas
$^{\$}$Indian Institute of Technology Dharwad, Dharwad, India
$^{\bullet}$Scuola Superiore Sant'Anna, Pisa, Italy
e-mail:sxr173131@utdallas.edu, k.kondepu@iitdh.ac.in

Abstract—**Optical network technology is one of the leading candidates for meeting the required backhaul transport layer latency and capacity requirements of 5G services. In addition, its physical layer programmability supports the execution of advanced methods that can improve 5G service reliability and SLA compliance in the face of equipment failure. While a number of such methods is addressed in the literature, including Virtual Network Function (VNF) fault-tolerant methods, a full proof of concept is yet to be reported.**

The study in this paper describes a testbed — along with its Software Defined Networking (SDN) and Network Function Virtualization (NFV) capabilities — which is used to experimentally showcase the key functionalities that are required by VNF fault-tolerant methods. The testbed makes use of OpenROADM compliant Dense Wavelength Division Multiplexing (DWDM) equipment to implement the programmable backhaul of a Next Generation Radio Access Network (NG-RAN) Non-standalone (NSA) architecture running 4G Evolved Packet Core (EPC) with the 5G next-generation NodeB (gNB). Specifically, the testbed is used to showcase the live migration of virtualized EPC components that is required to restore pre-failure VNF.

Index Terms—**Virtual EPC, Cloud-Native, Container, VM, OpenROADM, Live Migration.**

I. INTRODUCTION

Fifth-generation mobile networks are expected to support billions of devices with high data-rate, virtually no delay, and highly reliable connectivity [1]. Both Network Function Virtualization (NFV) and Software-Defined Networking (SDN) help achieve these objectives by enabling Cloud-Native Radio Access Network (C-RAN) architecture [2], [3]. Through the use of both NFV and SDN, radio network elements are implemented as Virtualized Network Functions (VNF), thus simplifying network programmability and reconfiguration. VNFs run as software components on top of either a Virtual Machine (VM) or Container virtualization framework[1], enabling network elements to be implemented in the Cloud and improving service flexibility.

The inherent geographically distributed C-RAN architecture also requires a high-speed and low-latency transport network. Barring lack of right-of-way access, optical fiber cables and Dense Wavelength Division Multiplexing (DWDM) represent the most desirable solution for the C-RAN backhaul due to their abundance of transmission capacity. Proprietary or open optical network solutions [4] are now offering physical layer SDN programmability that can be readily leveraged to achieve highly reliable transport connectivity in the backhaul.

While these platforms are widely beneficial, some reliability challenges remain open to be addressed [5]. For the VNFs, the Optical network architecture's reliability schemes generally consider the reliable Ethernet-over-DWDM transport network through dynamic rerouting [6]. However, the fault tolerance schemes need to consider recovering mobile application failures while maintaining application Quality Of Service (QoS) guarantees. It is believed that an integrated reliable system combining connection restoration and live migration is necessary to restore pre-failure VNF. VNF fault-tolerant methods have been widely discussed in the literature [7], [8], but a full proof of concept is yet to be reported in the telecommunication industry.

The contribution of this paper is to experimentally showcase some of the key functionalities that are required by VNF fault-tolerant methods. Specifically, a testbed is used to showcase the live migration of virtualized EPC components that is required to restore pre-failure VNF. The testbed makes use of OpenROADM compliant DWDM equipment to implement the programmable backhaul of a Next Generation Radio Access Network (NG-RAN) Non-standalone (NSA) architecture running 4G Evolved Packet Core (EPC) with the 5G next-generation NodeB (gNB). The NSA version of the 5G mobile communication comprises the New Radio (NR) and the NG-RAN, connected to the 4G EPC. The EPC functional components are the Mobility Management Entity (MME), the Home Subscriber Server (HSS), the Serving Gateway (S-GW), and the Packet Gateway (P-GW).

The Optical Network with the inter-operability support of multiple vendors in the context of SDN enables the cloud operators to deploy the C-RAN as vendor-agnostic white boxes in metro networks that significantly saves the CAPEX. OpenROADM Multi-Source Agreement (MSA) helps achieve the inter-operability at the southbound interface between the SDN controller that is fully compliant with YANG models [4]. With the aim of filling the gap between the reliability schemes and the QoS agreement for the mobile core network, we

---

[1] VMs concurrently and independently run on the same host compute hardware, while each provides distinct OS support to its guest application, namely each VNF. Docker makes use of OS-level virtualization to produce VNFs that run in packages called Containers.

978-3-903176-33-1

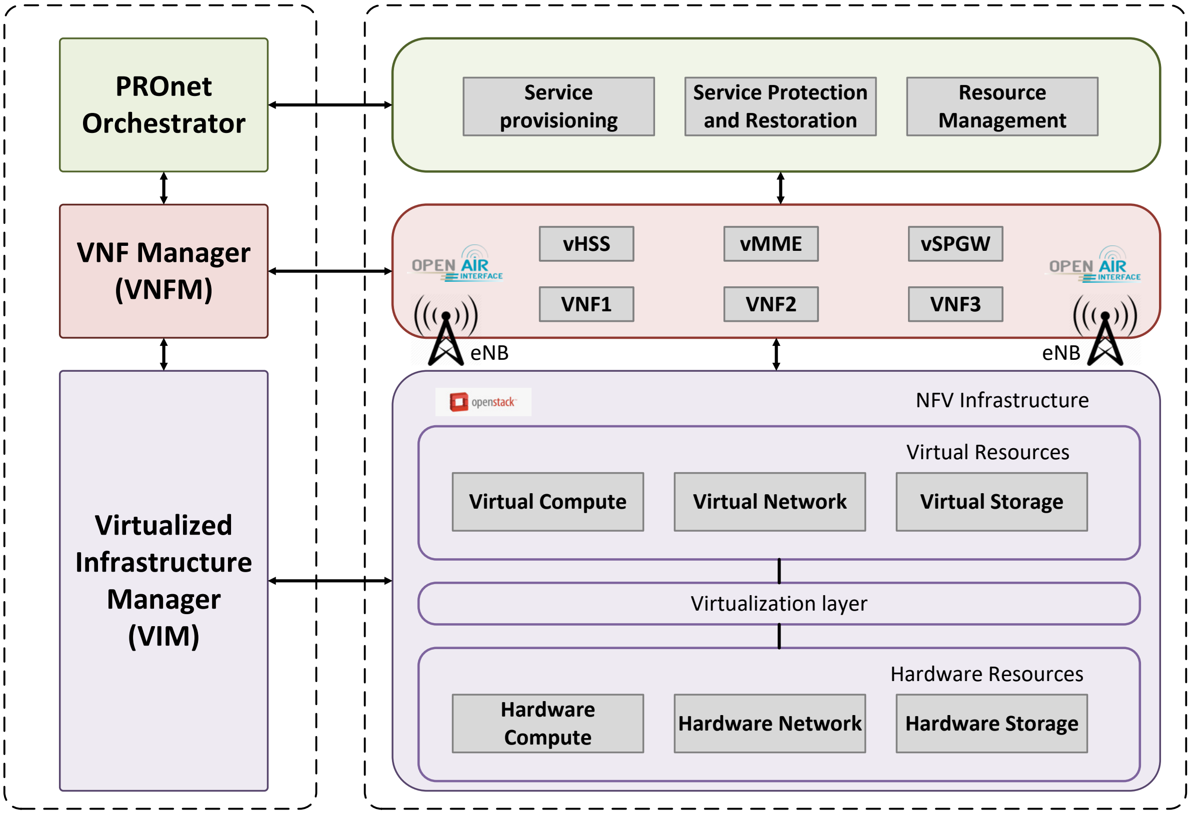

Fig. 1: Cloud based Architecture

propose to evaluate software-based fault tolerance – in terms of checkpoint and migration in the disaggregated elastic Optical Network orchestrated by the SDN controller.

II. Related Work

Providing resiliency for mobile network functions is a topic that has been broadly addressed. However, providing resiliency when mobile network functions are virtualized is a more recently defined challenge that involves different factors. For example, a VNF failure can be caused due hardware (including network elements) to software.

In [9], the 3GPP specifies different resiliency schemes for EPC components and how to handle failures with the help of Echo Request/Response timer messages. Such methods can be applied to both physical or virtual network functions. In [6], 5G fronthaul and backhaul protection and restoration mechanisms are evaluated in a programmable optical network.

In [10] approaches for recovering VNF through replication and migration of network functions when outages affect compute resources are presented. In addition, infrastructure network failures can be recovered directly at the network level, for example, by resorting to a SDN controller [11]. For instance, [7] presents a resiliency scheme for RAN functional split reconfiguration by orchestrating lightpath transmission adaptation. The VNF migration of virtualized Central Unit/virtualized Distributed Unit (vCU/vDU) (i.e., the gNB split functions) over WDM network using CRIU is briefly discussed in [12].

So far, no research work provides a detailed evaluation of NFV-SDN systems performing live migration of VM and Container supporting Core network (CN) functions in a programmable Optical Network. For the 5G system, resilience approaches need to handle both software and network failure while satisfying QoS during fault-recovery.

III. System Model and Rationale for Core Network Migration

Fig. 1 depicts the cloud-based architecture of the LTE Core Network considered in our study that is derived from the ETSI-NFV standards and takes into account the service provisioning and management of the CN components. The fronthaul and backhaul transport layer functionalities with built-in resilience mechanisms — Service Protection and Restoration — have been previously demonstrated in the PROnet (PRogrammable Optical network) testbed [6], [13], which has been more recently upgraded to operate with OpenROADM compliant equipment [14]. The protection mechanism is implemented at the Ethernet layer (1:1 or 1+1), while the optical circuit restoration mechanism is implemented at the DWDM layer. Overall, the resilience mechanism allows distributed application processes to overcome link failure by dynamically rerouting packets and optical circuits around the failed link — as shown by the red path when the green path is disrupted in Fig. 2. Beside maintain network connectivity between the CN components, this solution must also continue to guarantee specific Quality of Service (QoS) [15]. If the necessary channel capacity or latency is not achieved with the restored link, an additional complementing fault tolerance mechanism may be required, based on CN component relocation. The dotted green lines in Fig. 2 show the migration of CN components from Site C to Site B if the red path does not meet the minimum bandwidth and latency requirement after restoration. Once relocated to Site B, the CN components regain minimum bandwidth and latency requirements through the backhaul.

This study aims to integrate vEPC migration in PROnet SDN Orchestrator, intending to provide intelligent decision-making software to meet the QoS requirements during the link restoration. The design flowchart is shown in Fig. 3. When the path/link failure occurs, the desired link capacity is not always guaranteed during the restoration process. The objective is to select the core network migration if the desired QoS is not met by the link restoration process.

In our testbed the PROnet Orchestrator [13] acts as the NFV Orchestrator (NFVO). The Orchestrator supports and interfaces with Virtualized Infrastructure Managers (VIMs), i.e., OpenStack and Kubernetes. The NFV Infrastructure (NFVI) in our architecture consists of repurposed stampede compute nodes from Texas Advanced Computing Center (TACC). NFVI resources – compute, storage, and network – are managed and controlled by VIMs. The VNF Manager (VNFM) is responsible for instantiating and monitoring VNF instances. In the OpenStack cloud, Metal As A Service (MAAS) [16] is used to provision the compute nodes, and juju tool [17] is used to automate the software service deployment on the compute nodes. The VNFs considered are the core network functions - HSS, MME, and SPGW, running as either VM or Container. Our contemplated architecture has the RAN components — Distributed Unit and Central Unit (DU, CU) — that sit on the Edge cloud, forming a distributed NFV framework.

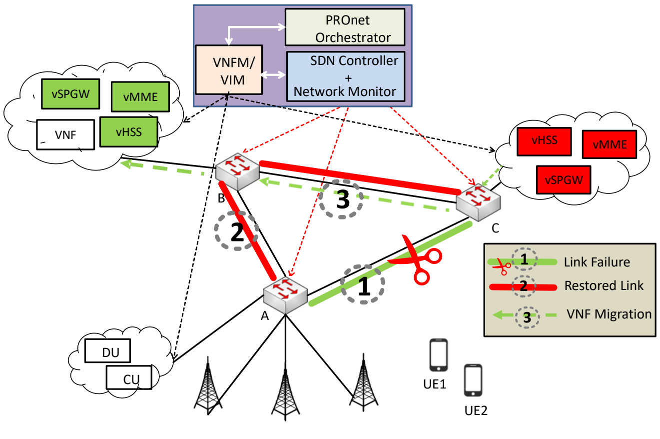

Fig. 2: Live migration motivation scenario

IV. Experimental Setup

Fig. 4 shows the block diagram of the PROnetOpenROADM testbed configuration. USRP B210 — software-defined radio — acts as the RF frontend with 2.6 GHz frequency coverage [18]. The User Equipment (UE) is deployed on a dedicated server with the B210 connectivity. This testbed is used to investigate Kernel-based Virtual Machine (KVM) for VM migration and Checkpoint/Restore In Userspace (CRIU) [19] for docker container migration. Two racks of stampede compute nodes are connected through an optical transport (the backhaul) network consisting ofmprising of OpenROADM compliant equipment.

The optical transport network consists of two OpenROADM nodes provided by Ciena (6500) and Fujitsu (1FINITY) for routing lightpaths between the two racks or compute sites. Transmission and reception of Ethernet client signals across the optical transport network are realized by deploying OpenROADM compliant Fujitsu (1FINITY) T300 100G Transponder and Juniper ACX6160-SF Transponder for the tenant network, and Fujitsu (1FINITY) F200 1G/10G/100G Switchponder and ECI Apollo OTN OpenROADM switchponder for the management network. The optical equipment is controlled by the open-source optical network controller TransportPCE version 2.0.0, which is an application running on OpenDaylight version 6.0.9. Also shown in Fig. 4, the programmable optical network (PROnet) Orchestrator coordinates automatic resource provisioning in an Ethernet-over-WDM network.

The virtualized EPC software components (HSS, MME, SPGW) are first executed on the left rack (Rack 1). Once triggered, the live migration of either the VM or Container that supports one of these EPC components takes place over a dedicated optical circuit (lightpath) that is dynamically created between the two racks to form a temporary high-speed connection in the management network to expedite the migration procedure between racks.

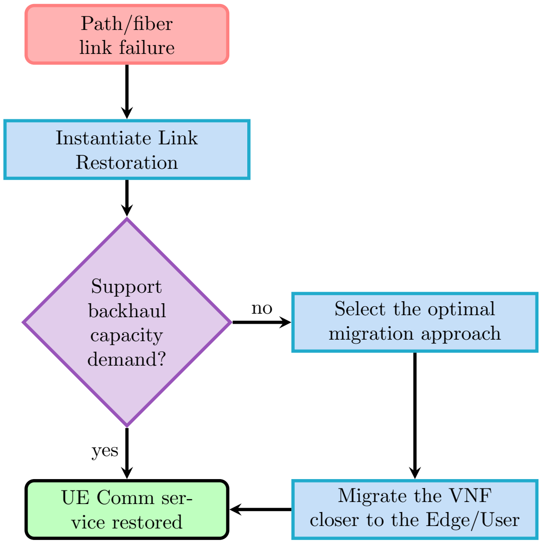

Fig. 3: An example of link restoration flow

OpenFlow [20], [21] enabled switches (Juniper QFX5120 and Dell N3048p) — controlled by the PROnet Orchestrator — are used to interconnect the compute nodes in the two racks and also to route packets (in both management and tenant networks) to the assigned transport optical equipment. The PROnet Orchestrator was recently upgraded with two additional features [14]: a RESTCONF interface to work with the TransportPCE northbound API, which relies on the

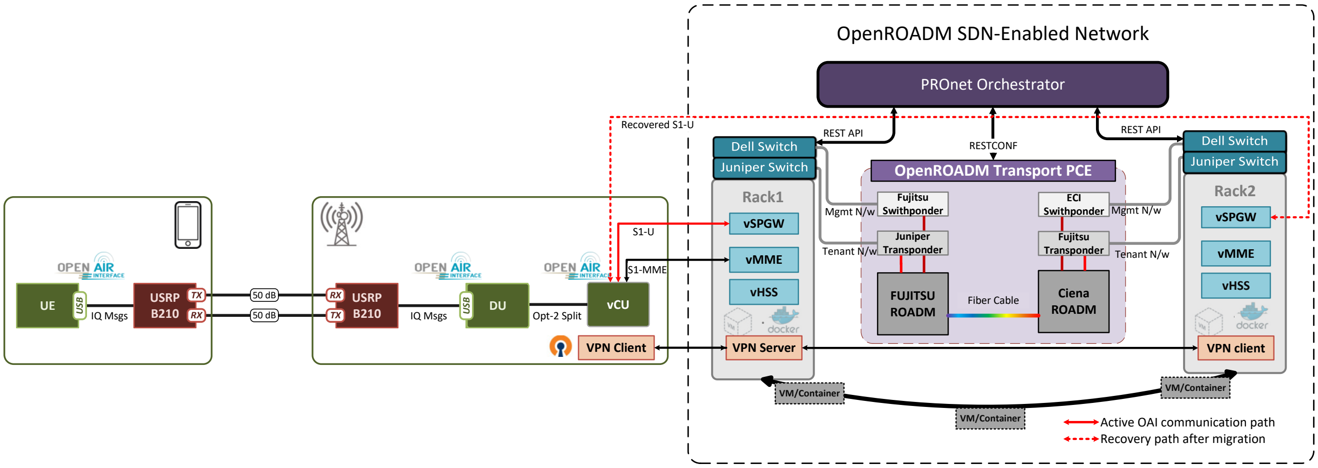

Fig. 4: Experimental Testbed

OpenROADM Service Model, and a REST API to work with OpenStack.

With these two upgrades, the PROnet Orchestrator offers a single point of control and coordination of the compute and network resources in the described experimental setting. For example, to enable experimentation with varying backhaul network round trip delays, the PROnet orchestrator is instructed to create lightpaths in the OpenROADM network with varying end-to-end propagation distances, i.e., a few meters – considered as short distance, 25 km, and 50 km. During the migration process, the PROnet Orchestrator first triggers the creation of the management lightpath between the two racks and then initiates the migration of one of the EPC virtual components. The migration procedure is carried out through the OpenStack dashboard when using VMs and through shell script commands when using Containers.

TABLE I: System configuration details.

Description	**OpenROADM**
Nodes	1 control node, 1 network manager, and 8 compute nodes on each rack
Product	Dell DCS8000Z
CPU	2 Intel Xeon E5-2680 processors@2.7GHz (16 cores, 2 threads/core)
Intel Architecture	Sandy Bridge
Memory	RAM: 32GB, Disk: 256GB Flash Storage
OpenStack - Management N/W	1G - 10G (Flexponder) - 1G
OpenStack - Tenant N/W	40G - 100G (Transponder) - 40G
Avg. CPU Utilization	< 10 %

The system configuration details of the testbed are reported in Table I. The UE, RAN, and the CN software modules are envisioned using the OpenAirInterface (OAI) software. RF with the DAC/ADC functionality resides on the B210 board. Option-2 split of design model is selected between the DU and CU modules that allows the centralization of the PDCP layer. The PROnet Orchestrator facilitates the migration process of the virtualized EPC components running on the compute nodes. For the VM migration, the hypervisor does the process of copying memory changes on the source and destination nodes. For the Container migration, the Checkpoint and Restore services are executed on the individual nodes. The version details of OAI software with VM, CRIU, and docker packages installed are shown in Table II.

TABLE II: Module version for the experimentation

Module	**Software version**
B210 Radio	UHD_3.14.0.0-release
RAN - UE,DU,CU	v2019.w25
Core - HSS,MME,SPGW	v0.5.0-4-g724542d
QEMU version	v3.1.0
Libvirt version	v5.0.0
Docker Container	v19.03.12
CRIU	v3.12

TABLE III: OpenStack flavors for the experimentation

No	**Flavor Name**	**vCPUs**	**RAM [MB]**	**Disk [GB]**
1	Small	1	2048	20
2	Medium	2	4096	40

V. EXPERIMENTAL EVALUATION

The PROnet testbed in Sec. IV is used to run the experiments. The experiments are conducted to assess the fault handling of the backhaul network. Both the VM PreCopy [22] and the Container StopandCopy [23] methods of migration are considered when migrating the CN components. The ability of the testbed to maintain connectivity between the CN components and the mobile end-user is assessed along with each component's migration time.

A. Migration Time Evaluation

Migration Time measures the time involved in migrating a VNF from the source node to the destination node. The two racks (compute sites) in Fig. 4 are connected using a dedicated Optical circuit, creating a high-speed connection between the compute racks. Once the migration procedure is initiated, the top-of-the-rack Ethernet switches are configured using the OpenDaylight controller to route data flow between the two racks. The CN migration is then initiated, moving the vEPC from its primary server in Rack 1 to its secondary server in Rack 2. The VPN configuration is updated in the secondary server to restore the mobile network backhaul communication. Fig. 5 reports the migration time of both VM and Container running the virtualized CN instances.

For all three CN components, VM migration time almost doubles that of the Container regardless of the flavor type. During the VM migration, the C-RAN core functions are still operational in the primary server. The extra time required by the VM migration is due to the Gigabytes (GB) of VM disk image that must be migrated along with the memory page. In contrast the Container Metadata size of the CN components is measured in Megabytes (MB). Each CN component has a slightly varying migration time based on the memory usage and storage size requirements. For example, the HSS Container migration time is more than that of MME and SPGW because of the storage size requirement. The HSS Metadata – size 173 MB – mainly stores the user database information consisting of a larger size than the MME and SPGW components. Similarly, the intense memory usage by the SPGW VM – with uplink and downlink data transfer – causes a slightly higher migration time than the HSS and MME VM instances.

Two OpenStack image flavor types are considered as shown in Table III. In OpenStack, flavors represent the compute, memory, and storage capacity reserved for a VNF. Based on the application processing requirement, the flavor is selected for the VNF. The impact of flavor type on the CN migration behavior is quantified in Fig. 5. Most notably, the VM and Container migration times are differently affected by the flavor type. The VM Medium flavor requires a modest extra migration time compared to the VM Small flavor because of its increased image size. Transferring a larger image from the source host to the destination host takes extra time – magnified when the network round trip time is large. Conversely, the Container Medium flavor requires less migration time compared to the Container Small flavor. Container metadata size almost remains the same irrespective of a flavor change. The improved CPU core configuration helps expedite both checkpoint and restoration executions for the Container based CN components.

B. UE Service Recovery time Evaluation

UE Service Recovery time measures the time interval during which the UE (mobile) connectivity is temporarily disconnected from the mobile network due to the CN component migration. In the normal UE attach procedure, a GTP tunnel is established between the end-user and the vEPC. Here, the user service disruption – in the data plane – is measured by monitoring the UE ICMP traffic with the EPC tunnel address. Fig. 6 shows the UE service recovery time captured in both VM and Container environments. Core Network Container migration has certain requirements, and the design strategy to achieve such requirements at the GTP interfaces is reported in [24]. The red line in Fig. 6 denotes the VNF running in the primary server, and when the link failure is artificially introduced, the migration is initiated. The UE data plane service is temporarily interrupted for around 2.3 seconds in the Container environment and approximately 5.4 seconds in the VM environment. But the required QoS – in terms of bandwidth – is restored once the VNF is migrated to the secondary server. The potential downside of this migration process is the burst loss of data packets. However, the UE remains connected irrespective of this service disturbance. This migration approach is acceptable in the backhaul network as long as stringent real-time transport layer requirements are not imposed. More in general, the downtime value is influenced by various migration approaches [23], [25]. By accounting for the experienced downtime of each migration approach the PROnet Orchestrator can invoke the approach that is most suitable at the time of the CN component migration.

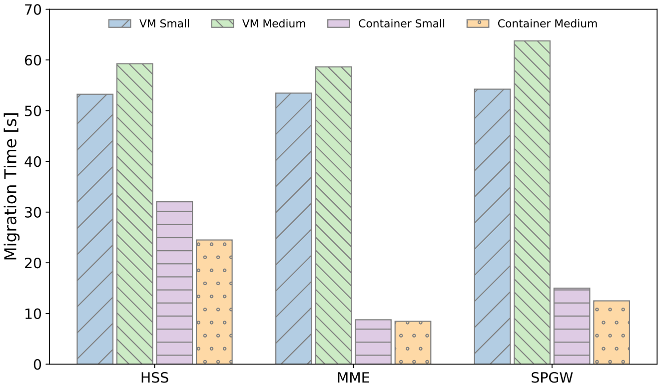

Fig. 5: Migration time analysis in OpenROADM

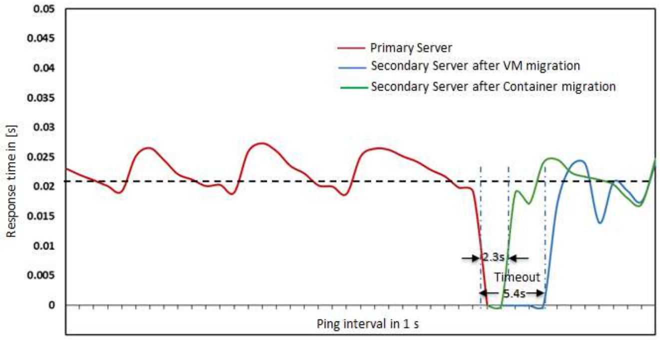

Fig. 6: Service Recovery time analysis in OpenROADM

C. Effect of Transport Network Propagation

In this experiment, the length of the lightpath connecting the primary and secondary rack is varied to study the impact of the rack geographical location on the CN component's migration time. The PROnet Orchestrator instantiates the CN component (i.e., MME) migration over three distinct lightpaths: one of a few hundreds of meters, 25 km, and 50 km, respectively as in Fig. 7. In addition to the fiber propagation delay – 5 microseconds per km – the experiment accounts for the delay introduced by the switchponder and transponder pairs used in the management and tenant network, respectively. The increase in MME migration time is noticeable for the VM medium flavor due to additional time required to migrate VM disk image along with the memory pages. Only a modest extra time is required to complete all four migration types when using a longer lightpath, thus proving that these solutions can scale geographically.

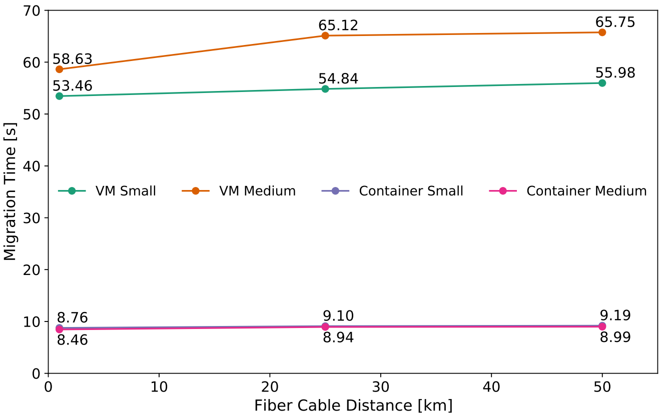

Fig. 7: Migration time influenced by lightpath length

VI. Conclusion

This paper experimentally evaluates an NFV enabled mobile network comprising a backhaul fiber optics transport network that is built with the latest OpenROADM compliant equipment (from multiple vendors) and SDN control technology. Through the single point of coordination provided by the PROnet Orchestrator module — for joint control of the backhaul optical layer, Ethernet layer, and compute resources — live migration of three EPC components — virtualized through either VM or Container technology — is experimentally achieved without causing UE disconnection. These experimental data represent an initial batch of results that can be applied to identify best practice in the context of link restoration, in which EPC components are migrated to a secondary site and the optical physical layer is reconfigured to guarantee QoS during fault-recovery.

Acknowledgment

This work is supported in part by NSF grants ACI-1541461, CNS-1531039T, and CNS-1956357, partially funded by the EC H2020 5GPPP 5Growth project (Grant 856709), and partially funded by the SGNF project (“Reliability Evaluation of Virtualised 5G").

References

[1] ITU, “5G - Fifth generation of mobile technologies.” [Online]. Available: https://www.itu.int/en/mediacentre/backgrounders/Pages/5G-fifth-generation-of-mobile-technologies.aspx

[2] SDx Central, “NFV Report Series Part 1: Foundations of NFV: NFV Infrastructure and VIM,” SDN Central Market Report, 2017.

[3] V. Nguyen, A. Brunstrom, K. Grinnemo, and J. Taheri, “SDN/NFV-based mobile packet core network architectures: A survey,” *IEEE Communications Surveys Tutorials*, vol. 19, no. 3, pp. 1567–1602, 2017.

[4] M. Birk, O. Renais, G. Lambert, C. Betoule, G. Thouenon, A. Triki, D. Bhardwaj, S. Vachhani, N. Padi, and S. Tse, “The openroadm initiative,” *J. Opt. Commun. Netw.*, vol. 12, no. 6, pp. C58–C67, Jun 2020.

[5] P. Rost, I. Berberana, A. Maeder, H. Paul, V. Suryaprakash, M. Valenti, D. Wübben, A. Dekorsy, and G. Fettweis, “Benefits and challenges of virtualization in 5G radio access networks,” *IEEE Communications Magazine*, vol. 53, no. 12, pp. 75–82, 2015.

[6] S. Ramanathan, M. Tacca, M. Razo, B. Mirkhanzadeh, K. Kondepu, F. Giannone, L. Valcarenghi, and A.Fumagalli, “A programmable optical network testbed in support of c-ran: A reliability study,” *Photonic Network Communications*, vol. 37, no. 3, p. 311–321, 2019.

[7] K. Kondepu, A. Sgambelluri, N. Sambo, F. Giannone, P. Castoldi, and L. Valcarenghi, “Orchestrating lightpath recovery and flexible functional split to preserve virtualized ran connectivity,” *J. Opt. Commun. Netw.*, vol. 10, no. 11, pp. 843–851, Nov. 2018.

[8] S. Cherrared, S. Imadali, E. Fabre, G. Gössler, and I. G. B. Yahia, “A survey of fault management in network virtualization environments: Challenges and solutions,” *IEEE Transactions on Network and Service Management*, vol. 16, no. 4, pp. 1537–1551, 2019.

[9] 3GPP TS 23.007, “Technical Specification Group Core Network and Terminals; Restoration procedures,” Release 16, Mar. 2020.

[10] F. Carpio and A. Jukan, “Improving reliability of service function chains with combined VNF migrations and replications,” *CoRR*, vol. abs/1711.08965, 2017.

[11] A. Giorgetti, A. Sgambelluri, F. Paolucci, F. Cugini, and P. Castoldi, “Demonstration of dynamic restoration in segment routing multi-layer sdn networks,” in *Optical Fiber Communications Conference and Exhibition (OFC)*, 2016, pp. 1–3.

[12] J. Feng, J. Zhang, Y. Xiao, and Y. Ji, “Demonstration of containerized vDU/vCU migration in wdm metro optical networks,” in *Optical Fiber Communication Conference (OFC) 2020*, 2020, p. Th3A.4.

[13] B. Mirkhanzadeh, A. Shakeri, C. Shao, M. Razo, M. Tacca, G. M. Galimberti, G. Martinelli, M. Cardani, and A. Fumagalli, “An sdn-enabled multi-layer protection and restoration mechanism,” *Optical Switching and Networking*, vol. 30, pp. 23 – 32, 2018.

[14] B. Mirkhanzadeh, S. Vachhani, B. G. Bathula, G. Thouenon, C. Betoule, A. Triki, M. Birk, O. Renais, T. Zhang, M. Razo, M. Tacca, and A. Fumagalli, “Demonstration of joint operation across openroadm metro network, openflow packet domain, and openstack compute domain,” in *2020 Optical Fiber Communications Conference and Exhibition (OFC)*, 2020, pp. 1–3.

[15] L. Valcarenghi, F. Cugini, F. Paolucci, and P. Castoldi, “Quality-of-service-aware fault tolerance for grid-enabled applications,” *Optical Switching and Networking*, vol. 5, no. 2, pp. 150–158, 2008.

[16] Canonical, “MAAS.” [Online]. Available: https://maas\.io/docs/snap/2.9/ui/about-maas

[17] Canonical Team, “Juju.” [Online]. Available: https://juju.is/docs

[18] Ettus, “Ettus B210 Radio board.” [Online]. Available: https://www.ettus.com/all-products/ub210-kit/

[19] CRIU Community, “Checkpoint/Restoration In UserSpace (CRIU),” 2019. [Online]. Available: https://criu.org/.

[20] OpenFlow, “OpenFlow Switch Specification,” April 2013. [Online]. Available: http://www.opennetworking.org/wp-content/uploads/2013/04/openflow-spec-v1.3.1.pdf

[21] OpenNetwork, “Open Networking Foundation.” [Online]. Available: www.opennetworking.org/

[22] A. Choudhary, M. Govil, G. Singh, L. Awasthi, E. Pilli, and D. Kapil, “A critical survey of live virtual machine migration techniques,” *J. Cloud Comput.*, vol. 6, no. 1, 2017.

[23] C. Puliafito, C. Vallati, E. Mingozzi, G. Merlino, F. Longo, and A. Puliafito, “Container migration in the fog: A performance evaluation,” *Sensors*, vol. 19, no. 7, 2019.

[24] S. Ramanathan, K. Kondepu, M. Tacca, L. Valcarenghi, M. Razo, and A.Fumagalli, “Container migration of core network component in cloud-native radio access network,” in *International Conference on Transparent Optical Networks (ICTON)*, Jul. 2020, pp. 1–6.

[25] A. Shribman and B. Hudzia, “Pre-copy and post-copy VM live migration for memory intensive applications,” in *18th international conference on Parallel processing workshops*, Aug. 2012, pp. 539–547.