Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 110 (2017) 344–351

Procedia Computer Science

www.elsevier.com/locate/procedia

The 12th International Conference on Future Networks and Communications (FNC 2017)

A simulative model of a 5G Telco Operator Network

Giuseppe Faraci*, Alfio Lombardo

DIEEI, University of Catania, Viale A. Doria, 6, 95125, Catania, ITALY

Abstract

In the near future, an important milestone for the evolution of wireless technologies will be the deployment of 5G network, having the target of supporting very huge data rate generated by a very high number of devices. One of the main technological enablers in this evolution is the joint SDN/NFV paradigm, defined in the last years to support the softwarization process of the Telco Operator networks. Given the very hard quality of experience (QoE) and quality of service (QoS) requirements in some application scenarios, mainly in terms of end-to-end delay, a challenging activity is to realize tools that can support network architects in performance evaluation and network design.

With this in mind, this paper proposes a simulative tool for 5G networks, which is able to capture delay statistics due to both CPU load and transmission link congestions in NFVI-PoP nodes. The model is then applied to a case study to demonstrate how it can be applied for performance evaluation.

Peer-review under responsibility of the Conference Program Chairs.

Keywords: 5G, SDN, NFV, End-to-end delay, Network Model.

1. Introduction

In the last years, wireless technology has emerged as one of the most significant trends in networking. According to recent statistics, mobile wireless broadband penetration will exceed that of fixed wireline broadband networks, and by 2018, the global mobile traffic is expected to increase from 2.6 to 15.8 Exabyte[1].

To meet the request of this massive device connectivity, the fifth generation (5G) will be developed with efficiency, scalability, and versatility as main design objectives, with the final target of supporting 1000 times the

* Corresponding author. Tel.: +39 095 7382375
E-mail address: gfaraci@dieei.unict.it

1877-0509 © 2017 The Authors. Published by Elsevier B.V.
Peer-review under responsibility of the Conference Program Chairs.
10.1016/j.procs.2017.06.104

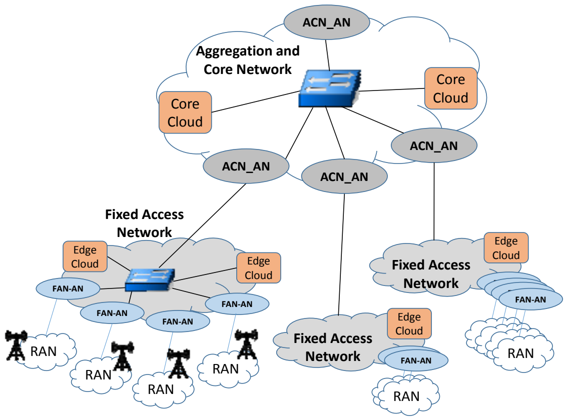


Figure 1: Reference 5G Network Architecture

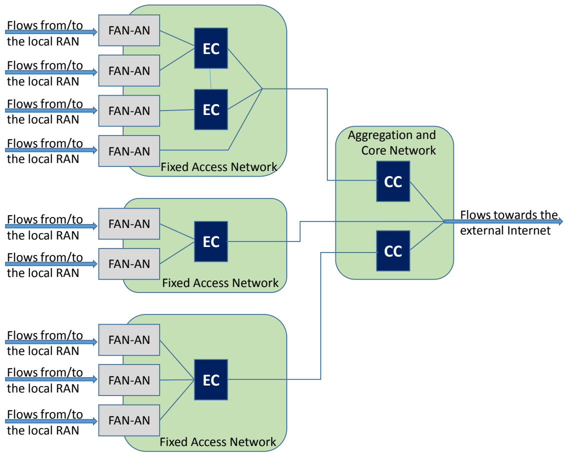


Figure 2: Reference 5G Network Architecture

current aggregate data rate and 100 times the user data rate, while enabling a 100 times increase in the number of currently connected devices, 5 times decrease of end-to-end latency, and 10 times increase of battery lifetime[2].

In this context, the joint SDN/NFV paradigm is an enabling technology to meet efficiency, scalability, and versatility objectives of 5G networks, while smoothing its coexistence with previous technologies[3].

Software Defined Networks (SDN)[4,5] aims at separating the control plane from the data plane, so providing the possibility of dynamically steering traffic flows according to some policy centrally implemented by the SDN Controller. Network Function Virtualization (NFV)[6,7], on the other side, inherits resource virtualization technologies from data center and cloud environments to implement network functions in software components called virtual network functions (VNFs), which are deployed on high-volume servers or cloud infrastructure instead of specialized hardware middle boxes[8,9,10]. Joining the above technologies allows the realization of a big data center distributed over the node of the Telco Operator network.

Given the very hard quality of experience (QoE) and quality of service (QoS) requirements in some application scenarios, like for example video surveillance[11,12], telelearning[13,14] and real-time multimedia content delivery[15,18], mainly in terms of end-to-end delay, a challenging activity in this future Internet scenario is to realize tools that can support network architects in performance evaluation and network design.

The target of this paper is to propose a simulative tool of a 5G network that is able to capture delays due to both CPU load and transmission link congestions in NFV Infrastructure Point-of-Presence (NFVI-PoP) nodes. At the best of our knowledge, this is the first model in the literature that is specific for a 5G network, and that considers the application of the SDN and NFV network paradigms.

The paper is structured as follows. Section 2 details the reference system, while the system model is described in Section 3. Section 4 introduces a case study to show how the proposed model can be applied. Finally, Section 5 draws some conclusions and describe some future work.

2. Reference System

The system we refer to in this paper is the 5G network architecture, as described in[19] and sketched in Fig. 1. It presents a hierarchical structure constituted by three different layers: the Radio Access Network (RAN), the Fixed Access Network (FAN), and the Aggregation and Core Network (ACN). Each RAN provides connectivity to mobile users, and is connected to a FAN through a FAN Access Node (FAN-AN). Moreover, each FAN is connected to the ACN through an ACN Access Node (ACN-AN).

According to the SDN/NFV paradigm, network functions run as virtual functions on NFVI Points of Presence (NFVI-PoPs), which are network nodes compliant with the NFV specifications[20,21].

Referring to Fig. 1, nodes working as NFVI-PoPs are the Edge Clouds located in the FANs, and the Core Clouds located in the ACN. In addition, according to the edge-computing approach[22,23], any access of the FAN network can be equipped to be NFVI-compliant, so working as NFVI-PoP.

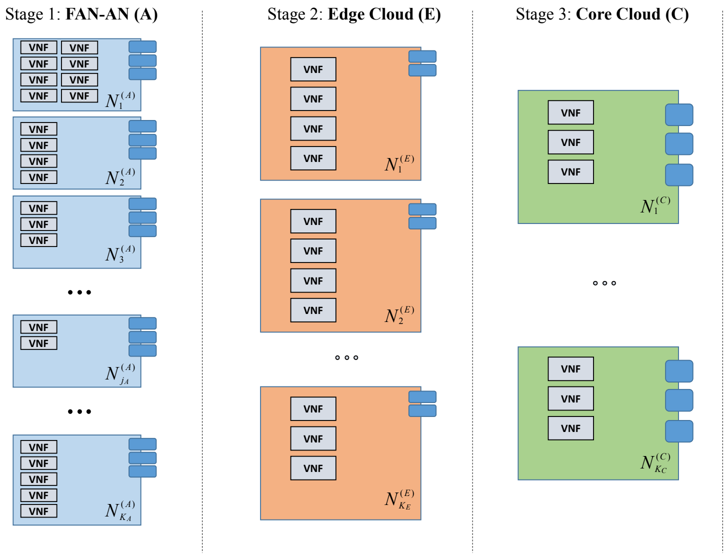


Figure 3: 5G Network Model

Therefore, the NFVI of a 5G network can be synthesized as shown in Fig. 2, where each rectangular block is a potential NFVI-PoP where some VNF can run. According to the NFV specifications, each flow can receive a Network Service (NS) that can be constituted by either one VNF or an ordered chain of them. In the following, without losing in generality, we will use the acronym VNF to indicate either a single virtual network function, or a chain of virtual network functions running on the same node. Moreover, we will use the term *flow* to indicate either a single flow or an aggregate of elementary flows originated from the same RAN, and requesting the same chain of VNFs.

The internal architecture of an NFVI-PoP is compliant with the NFV specifications. According to the indications received by the NFV Orchestrator (NFVO) of the Management and Orchestration (MANO) entity, which implements global management and orchestration policies, the Virtual Infrastructure Manager (VIM) interacts with the local hypervisor to assign computing, storage and network resources to the VNFs running on the node, in such a way that they respect some given performance requirements. For each running VNF, a VNF Manager (VNFM) and an Element Manager (EM) are instantiated, the former to perform lifecycle management functions, like instantiation, configuration, update/upgrade, out/in and up/down scaling, termination and performance result collections, and the latter for FCAPS (Fault, Configuration, Accounting, Performance and Security) management functionalities. Finally, an SDN Controller is in charge of implementing routing policies among the local elementary VNFs, in order to chain them to each other, and to other external VNFs.

3. System Model

In the following, for the sake of conciseness, we will refer to an NFVI-PoP as a node. Moreover, in order to simplify notation but without losing in generality, we refer to the network shown in Fig. 3, where we assumed that each flow receives a VNF at each crossed NFVI-PoP. Mathematical symbols used in the sequel are listed in Table I. Let us indicate the generic stage in the flow path as *p*, where $p \in \{A, E, C\}$, where *A* stands for FAN-AN stage, *E* for Edge Cloud stage, and *C* for Core Cloud stage. Let K_p be the number of NFVI-PoP nodes at the generic *p*-th stage. Therefore, the generic *j*-th node at the *p*-th stage can be referred to as $N_j^{(p)}$. Let $M_{p,j}$ be the number of VNFs running on the node $N_j^{(p)}$, and $F_{j,b}^{(p)}$ is the b-th VNF running on the node j, with $b \in [1, M_{p,j}]$.

Each flow ϕ, generated from a mobile terminal in a RAN, follows a given network path, identified by the following sequence:

$$S(\phi) = \{n^{(A)}(\phi), f^{(A)}(\phi), n^{(E)}(\phi), f^{(E)}(\phi), n^{(C)}(\phi), f^{(C)}(\phi), y(\phi)\} \tag{1}$$

where $f^{(p)}(\phi)$ indicates the VNF $p \in \{A, E, C\}$ in the chain assigned to the flow ϕ, while $n^{(p)}(\phi)$ is the node where it is running. The term $y(\phi)$ indicates the output NIC of the last node in the chain, while the output NICs of the nodes in the stages A and E are determined by the successive nodes in the sequence $S(\phi)$.

Focusing on a generic stage $p \in \{A, E, C\}$, let us indicate the set of all the possible sub-chains from the stage *p* to the last stage *C*, also considering the output NICs of the last-stage nodes, as $\Im^{(p)}$. We have:

Table I: List of mathematical symbols

Symbol	Meaning	Symbol	Meaning
A, E, C	FAN-AN, Edge Cloud and Core Cloud stages	$N^{(p)}(S_p)$	Generic node at the stage p for the sub-chain S_p
p, j	Generic stage in the flow path and generic node	$F^{(p)}(S_p)$	VNF used in the node $N^{(p)}(S_p)$
K_p	No. of NFVI-POP nodes at the p-th stage	$\mu^{(P,PoP)}$	Processor rate of the node [packets/s]
$N_j^{(p)}$	Generic j-th node at the p-th stage	$\mu^{(NIC,l)}$	Output transmission rate of the l-th NIC of the node, expressed in bits/s
$M_{p,j}$	Number of VNFs running on the node $N_j^{(p)}$	Ψ	Set of SCGs starting from a certain node
$F_{j,b}^{(p)}$	b-th VNF running on the node j	σ	Generic SCG in Ψ
$S(\phi)$	Sequence of nodes and VNFs crossed by the flow ϕ	$\mu^{(P,\sigma)}$	Processor rate assigned to the virtual CPU dedicated to σ
$f^{(p)}(\phi)$	p-th VNF in the chain assigned to the flow ϕ	$\mu^{(NIC,\sigma)}$	Tx bandwidth assigned to the virtual NIC used by the flows of σ to leave the node
$n^{(p)}(\phi)$	The node where $f^{(p)}(\phi)$ is running	$K^{(VNF)}$	Size of each queue associated to each VNF
$y(\phi)$	Output NIC of the last node in the chain	$K^{(NIC)}$	Queue size associated to the virtual NICs
$\mathfrak{I}^{(p)}$	Set of all the possible sub-chains from the stage p to the last stage C	$N_{NotEmpty}^{(VNF)}(t)$	Number of VNF queues that have at least one packet at the generic instant t
Ω_p	Number of possible sub-chains starting from the stage p to the last stage C	$\Lambda_\sigma^{(OUT)}(n)$	Output trace for the virtual NIC associated to σ
S_p	Generic sub-chain belonging to $\mathfrak{I}^{(p)}$	$\underline{\Lambda}_\sigma^{(IN,p)}(n)$	Traces entering the $M_{p,j}$ VNF queues associated to the virtual CPU assigned to the considered SCG σ
$\sigma_{S_p}^{(p)}$	Sub-chain companion group (SCG) for the sub-chain S_p	$\aleph_p$	Set of all the SCGs that, starting from the stage p, have the same path of σ

$$\mathfrak{I}^{(p)} = \begin{cases} \left\{\left(N_{j_A}^{(A)}, F_{j_A,b_A}^{(A)}, N_{j_E}^{(E)}, F_{j_E,b_E}^{(E)}, N_{j_C}^{(C)}, F_{j_C,b_C}^{(C)}, NIC_h\right)\right\} & \text{if } p = A \\ \left\{\left(N_{j_E}^{(E)}, F_{j_E,b_E}^{(E)}, N_{j_C}^{(C)}, F_{j_C,b_C}^{(C)}, NIC_h\right)\right\} & \text{if } p = E \\ \left\{\left(N_{j_C}^{(C)}, F_{j_C,b_C}^{(C)}, NIC_h\right)\right\} & \text{if } p = C \end{cases} \quad (2)$$

$$\forall j_A \in [1, K_A], \forall j_E \in [1, K_E], \forall j_C \in [1, K_C], \quad \forall b_A \in [1, M_{A,j_A}], \forall b_E \in [1, M_{E,j_E}], \forall b_C \in [1, M_{C,j_C}], \quad \forall h \in [1, L]$$

Let us indicate the number of possible sub-chains starting from the stage p to the last stage C as Ω_p, which represents the cardinality of $\mathfrak{I}^{(p)}$. It can be calculated as follows:

$$\Omega_C = \sum_{j=1}^{K_C} M_{C,j} \cdot L \qquad \Omega_E = \sum_{j=1}^{K_E} M_{E,j} \cdot \Omega_C \qquad \Omega_A = \sum_{j=1}^{K_A} M_{A,j} \cdot \Omega_E \quad (3)$$

Now, let us consider a generic stage p, and let S_p be the generic sub-chain belonging to $\mathfrak{I}^{(p)}$. Let us define a *sub-chain companion group* (SCG) for the sub-chain S_p, $\sigma_{S_p}^{(p)}$, as the set of flows sharing the same node at stage p, and the same sub-chain starting from the stage $p+1$, that is:

$$\sigma_{S_A}^{(A)} = \left\{ \phi \text{ such that: } \begin{cases} n^{(A)}(\phi) = N^{(A)}(S_A), \\ \left(n^{(E)}(\phi), f^{(E)}(\phi)\right) = \left(N^{(E)}(S_A), F^{(E)}(S_A)\right), \\ \left(n^{(C)}(\phi), f^{(C)}(\phi)\right) = \left(N^{(C)}(S_A), F^{(C)}(S_A)\right) \\ y(\phi) = NIC(S_A) \end{cases} \right\}$$

$$\sigma_{S_E}^{(E)} = \left\{ \phi \text{ such that: } \begin{cases} n^{(E)}(\phi) = N^{(E)}(S_E) \\ \left(n^{(C)}(\phi), f^{(C)}(\phi)\right) = \left(N^{(C)}(S_E), F^{(C)}(S_E)\right) \\ y(\phi) = NIC(S_E) \end{cases} \right\} \qquad \sigma_{S_C}^{(C)} = \left\{ \phi \text{ such that: } \begin{cases} n^{(C)}(\phi) = N^{(C)}(S_C) \\ y(\phi) = NIC(S_C) \end{cases} \right\} \quad (4)$$

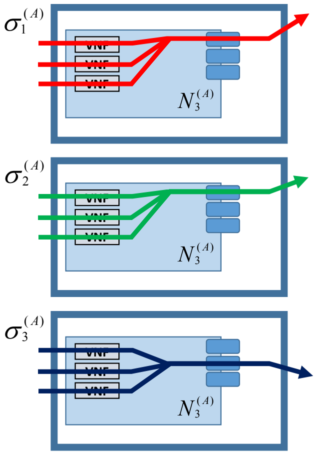


Figure 4: Model of an NFVI-PoP as seen by the flows of a generic SCG

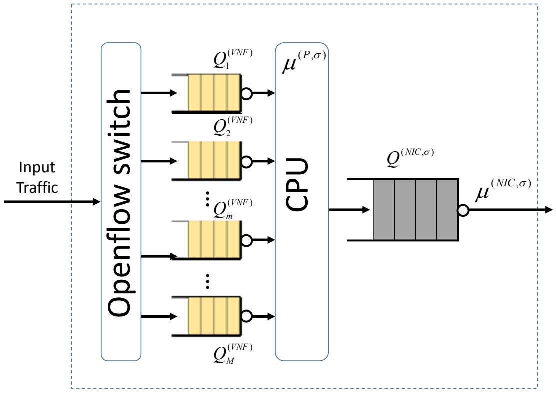


Figure 5: Model of an NFVI-PoP as seen by the flows of a generic SCG

where, according to the chain definition in (1), $N^{(p)}(S_p)$ is the generic node at the stage p for the sub-chain S_p, while $F^{(p)}(S_p)$ is the VNF used in that node. It is easy to prove that all the SCGs defined at the stage p are a partition of the set $\Im^{(p)}$.

3.1. Intra-PoP Resource Sharing

According to a multi-tenant approach, flows crossing the same NFVI-PoP but not belonging to the same SCG, do not have to interfere with each other. Instead, flows belonging to the same SCG will compete on the same resources. Therefore, let us consider a generic NFVI-PoP (refer to Fig. 4 where the node $N_3^{(A)}$ is considered as an example, and some SCGs are shown). Let $\mu^{(P,PoP)}$ be the whole processor rate of the node, expressed in packet/s, and $\mu^{(NIC,l)}$ be the total output transmission rate of its l-th NIC, expressed in bit/s. According to some guidelines received by the Orchestrator, the local Hypervisor subdivides the processor rate to the SCGs crossing the same node, while the SDN controller assigns a portion of the total output rate of each NIC to the flows crossing it.
Now, considering a generic NFVI-PoP, let Ψ be the set of SCGs starting from it. If we indicate the generic SCG in Ψ as σ, and indicating the NIC used by the flows belonging to σ to leave the considered node as l, let $\mu^{(P,\sigma)}$ be the amount of processor rate assigned to the virtual CPU dedicated to σ, and $\mu^{(NIC,\sigma)}$ be the transmission bandwidth assigned to the virtual NIC used by the flows of σ to leave the considered node. Of course, we have:

$$\sum_{\forall\sigma\in\Psi} \mu^{(P,\sigma)} = \mu^{(P,PoP)} \quad \text{and} \quad \sum_{\forall\sigma\in\Psi} \mu^{(NIC,\sigma)} = \sum_{\forall l} \mu^{(NIC,l)} \tag{5}$$

Consequently, the flows belonging to a generic SCG σ see the considered node as dedicated to them. Thanks to this property, an NFVI-PoP node, from the viewpoint of the flows belonging to a given SCG σ, can be modeled as shown in Fig. 5. Input traffic coming from nodes of the previous stage, or directly from the sources if the considered NFVI-PoP node is at the stage A, enter the OpenFlow switch that, according to the rules received by the NFV Orchestrator, redirects flows to the requested VNF.
As shown in Fig. 5, a set of queues are associated to the VNFs for the considered SCG σ in order to buffer packets waiting for processing by the virtual CPU[24], and one queue to the virtual NIC associated to σ for packets waiting for transmission on the output link. Let M be the number of VNFs that are running in the node for the SCG σ, and therefore the number of associated queues, in the following referred to as *VNF queues*. Let $K^{(VNF)}$ and $K^{(NIC)}$ be the size of each queue associated to each VNF and to the virtual NIC, respectively, representing the maximum number of packets that each queue can contain.

1 for each stage $p \in \{A, E, C\}$
2 for each $j \in [1 : K_p]$, that is, for each node $N_j^{(p)}$ in the stage *p*
3 for each SCG $\sigma \in \Psi_j^{(p)}$, where $\Psi_j^{(p)}$ is the set of all the SCGs starting from the node $N_j^{(p)}$
4 for each VNF $b \in [1 : M_{p,j}]$ in the node $N_j^{(p)}$
5 if *p*==A then $\Lambda_{b|\sigma}^{(IN,A)}(n) = \Lambda_{RAN}^{(\sigma,b)}(n)$; end
6 if *p*==E then $\Lambda_{b|\sigma}^{(IN,E)}(n) = \sum_{\lambda \in \aleph_E} \Lambda_\lambda^{(OUT,A)}$; end
7 if *p*==E then $\Lambda_{b|\sigma}^{(IN,C)}(n) = \sum_{\lambda \in \aleph_C} \Lambda_\lambda^{(OUT,E)}$; end
8 end
9 [$\Lambda_\sigma^{(OUT,p)}(n)$, $\delta_{[b]}^{(P,p,j,\sigma)}$ $\forall b$, $\delta^{(NIC,p,j,\sigma)}$]= NFVI_POP_model($\Lambda_{b|\sigma}^{(IN,p)}(n)$ $\forall b$, $\mu^{(P,\sigma)}$, $\mu^{(NIC,\sigma)}$)
10 for each $\phi \in \sigma$
11 $\delta(\phi) = \delta(\phi) + \delta_{[f^{(p)}(\phi)]}^{(P,p,j,\sigma)} + \delta^{(NIC,p,j,\sigma)}$
12 end
13 end
14 end
15 end

Figure 6: Algorithm to evaluate per-flow end-to-end delay

The total processor rate $\mu^{(P,\sigma)}$ of the virtual CPU assigned to the SCG σ is shared among all the active VNFs used by σ according to a round robin policy. If we indicate the number of VNF queues that have at least one packet at the generic instant *t* as $N_{NotEmpty}^{(VNF)}(t)$, each VNF queue is served with a portion of processing rate equal to $\mu^{(P,\sigma)} / N_{NotEmpty}^{(VNF)}(t)$.

3.2. Simulative model of the network as a whole

In this section, we describe the algorithm to evaluate end-to-end performance of the whole network shown in Fig. 1 and synthesized in Fig. 2. The algorithm is presented in Fig. 6. As shown in Fig. 6.a, it is run, stage-by-stage, for each node of the generic considered stage. More specifically, for each node $N_j^{(p)}$, we consider the generic SCG σ, and run the simulative tool *NFVI_POP_model*, a Matlab tool that is able to simulate the queueing system model shown in Fig. 5. This tool allows us to calculate the mean delay suffered in the node by each flow belonging to σ, and the output trace, $\Lambda_\sigma^{(OUT)}(n)$, for the virtual NIC associated to σ. To this purpose, we need the set of traces $\underline{\Lambda}_\sigma^{(IN,p)}(n)$ that enter the $M_{p,j}$ VNF queues associated to the virtual CPU assigned to the considered SCG σ. As expressed at the lines 5, 6 and 7 of the pseudo-code in Fig. 6.a, if we are considering the FAN-AN stage, the input traces $\underline{\Lambda}_\sigma^{(IN,p)}(n)$ are the flows generated by the mobile nodes that are present in the RAN connected to the considered access node. Instead, for the Edge Cloud stage and the Core Cloud stage, we can calculate them by summing the output traces of the nodes of the previous stage. More specifically, for an SCG σ at the stage *E*, we have to sum all the flows belonging to the SCGs λ defined at the stage *A* that, starting from the stage *E*, have the same path of σ, that is, all the output flows calculated by the *NFVI_POP_model* tool considering the all the SCGs λ belonging to the following set:

$$\aleph_E = \left\{ \lambda \in \Im^{(A)} \middle| \left(N^{(E)}(\lambda), N^{(C)}(\lambda), F^{(C)}(\lambda), NIC(\lambda) \right) = \left(N^{(E)}(\sigma), N^{(C)}(\sigma), F^{(C)}(\sigma), NIC(\sigma) \right) \right\} \quad (6)$$

Likewise, for the stage *C*, we have to consider all the SCGs λ belonging to the following set:

$$\aleph_C = \left\{ \lambda \in \Im^{(E)} \middle| \left(N^{(C)}(\lambda), NIC(\lambda) \right) = \left(N^{(C)}(\sigma), NIC(\sigma) \right) \right\} \quad (7)$$

Finally, once the *NFVI_POP_model* tool has calculated the delay suffered by each flow in the VNF queue and in the NIC queue of each crossed NFVI-PoP, as specified at lines 10-12 of the pseudo-code, we can calculate the per-flow end-to-end delay, and the amount of delay suffered to wait VNFs, and the amount of delay suffered to wait transmission on the output links.

4. Case Study

In this section, we consider a case study to apply the model described so far. More specifically, let us refer to the network shown in Fig. 7, constituted by two fixed access networks, each connected to three RANs. One Edge Cloud is present in each FAN, and one Core Cloud is present in the ACN. We assumed that the number of flows generated from the six RANs are 174, 124, 97, 75, 44 and 112. As far as the VNF placement policy, we decided to run each VNF in the root of the sub-trees containing all the flows requesting the VNF. For example, if a given VNF is

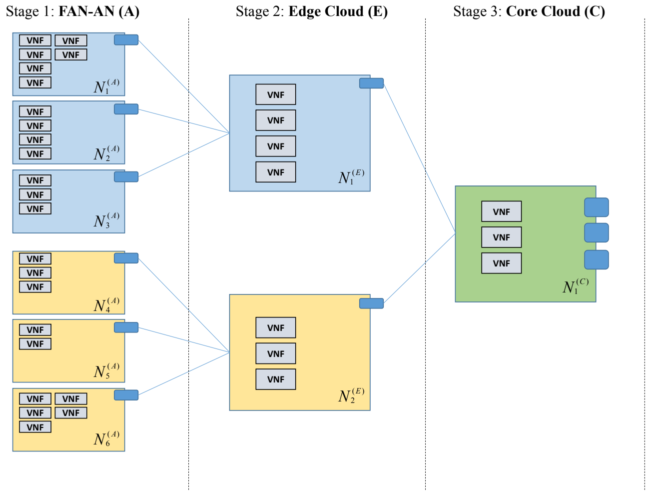


Figure 7: Network Topology Considered in the Case Study

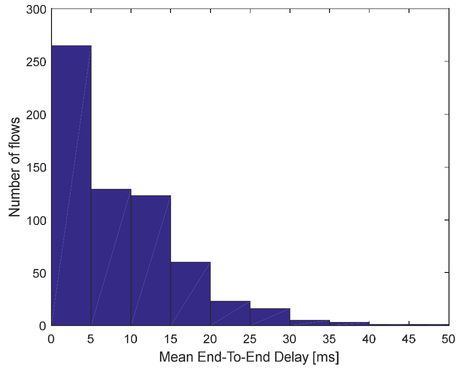
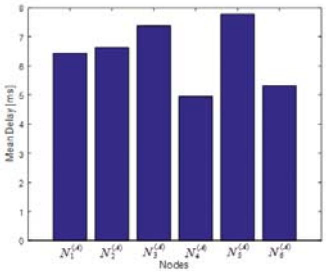
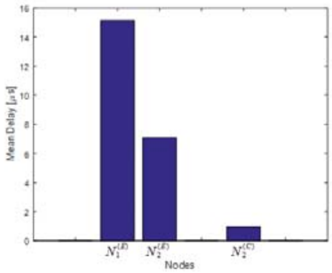


Figure 8. Histogram of the mean end-to-end delay



(a) Mean delay in the Stage-A nodes



(b) Mean delay in the Stage-E and Stage-C nodes

Figure 9: Mean delay in all the NFVI-PoPs

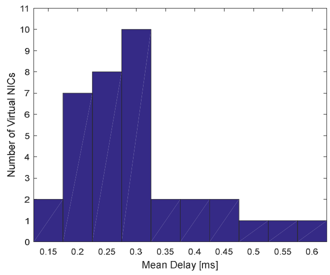


Figure 10:Histogram of the mean delay in the NIC queues

requested by flows all belonging to the same RAN, that VNF is placed on the relevant FAN-AN. Instead, if a VNF is requested by flows belonging to RANs connected to the same FAN, that VNF is placed on the Edge Cloud of that FAN. Otherwise, it is run in the Core Cloud.

Association of VNFs (let us remember that we use the term VNF also to refer to a composite network service) to each flow has been done randomly, but in such a way that three different VNFs are requested by each flow. Accordingly, thirty-three different VNFs have been requested, and their placement is as shown in Fig. 7.

Since no 5G traces are available today, the traffic trace for each flow has been generated with a switched batch Bernoulli process (SBBP) modulated by an underlying Markov model[25,26]. More in deep, we considered the time slotted with a slot duration of 250 µs, and we have randomly calculated the number of packets per slot for each flow, by using a three-state SBBP model characterized by the following 3x3 transition probability matrix for the underlying Markov chain: $P=[0.9960, 0.0035, 0.0005;\ 0.0040, 0.9543, 0.0417;\ 0.0040, 0.3160, 0.6800]$, where each semicolon indicates the end of a row.

Two states of the underlying Markov chain are OFF with two different mean duration, and therefore are characterized by no packet emission. In the third state, emission occurs with a mean value randomly chosen in the set of integer values {1,..,6}. The size of each packet is randomly calculated with a mean value of 1024 bytes. The buffer dimension of all the queues is of 1000 packets. The CPU processing rates and the NIC transmission rates have been randomly calculated in such a way that the utilization coefficient, defined as the ratio between the average arrival rate and the average service rate, has been randomly calculated in the interval [0.85, 0.95]. Finally, we assumed that all flows have the same importance, and therefore the virtual CPU portion and the virtual NIC rate assigned to each SCG are decided proportionally to their average input rate.

Fig. 8 presents the histogram of the mean delay calculated for all the flows. From this figure, we can observe how the mean delay is distributed over the considered flows. For example, we can note that 265 flows over the 626 considered ones suffer a mean delay less than 5 ms. In Fig. 9, showing the mean delay suffered in all the nodes of

the network, averaged over all the considered flows, we can observe how the most part of the delay is suffered on the FAN-AN nodes, while the delay in both the Edge Clouds and the Core Clouds are more little.
Finally. Fig. 10 shows the histogram of the mean delay suffered in all the NIC queues.

5. Conclusions and Future Work

The target of this paper is to define a simulative model of a 5G network, constituted with a three-level tree structure, that is, from the bottom to the upper layer, the RANs, the FANs, and the Telco Operator Core network. Assuming an SDN/NFV paradigm that runs VNFs as software tools in the servers located in the FAN Access Nodes, in the Edge Clouds and in the Core Cloud, the proposed model is able to evaluate end-to-end delay as composed by the delay suffered in the VNF queues and the delay experienced in the NIC queues of all the NFVI-PoP nodes in the network. As a future work, we are planning to introduce a Markov-based model within the NFVI-PoP model tool, in such a way that the whole model will be completely analytical.

Acknowledgements

This work was partially supported by the EU INPUT project.

References

1. S. Abdelwahab, *et al.*, "Network function virtualization in 5G," in IEEE Communications Magazine, vol. 54, no. 4, pp. 84-91, April 2016.
2. J. G. Andrews et al., "What Will 5G Be?," in IEEE Journal on Selected Areas in Communications, vol. 32, no. 6, pp. 1065-1082, June 2014.
3. C. Lee, *et al.*, "Enhanced LTE handover scheme using NFV for LTE handover delay reduction," 2016 IEEE ICCE-Asia, Seoul, 2016.
4. White paper on "Software-Defined Networking: The New Norm for Networks", available at https://www.opennetworking.org/.
5. Kreutz, D.; Ramos, F.M.V.; Esteves Verissimo, P.; Esteve Rothenberg, C.; Azodolmolky, S.; Uhlig, S., "Software-Defined Networking: A Comprehensive Survey," in Proceedings of the IEEE , vol.103, no.1, pp.14-76, Jan. 2015.
6. White paper on "Network Functions Virtualisation", available at http://portal.etsi.org/NFV/NFV_White_Paper.pdf.
7. Network Functions Virtualisation – White Paper #3, "Network Operator Perspectives on Industry Progress," available at https://portal.etsi.org/ Portals/0/TBpages/NFV/Docs/NFV_White_Paper3.pdf.
8. V. Sekar *et al.*, "The middlebox manifesto: Enabling innovation in middlebox deployment", Proc. ACM HotNets-X, 2011.
9. A. Lombardo, C. Panarello, D. Reforgiato, G. Schembra, "Measuring and modeling Energy Consumption to design a Green NetFPGA Giga-Router," in Proc. of IEEE Globecom 2012, Anaheim, California, USA, 3-7 December 2012.
10. J. Sherry et al., "Making middleboxes someone else's problem: Network processing as a cloud service", ACM CCR, vol. 42, no. 4, 2012.
11. A. Lombardo, et al., "Multipath Routing and Rate-Controlled Video Encoding in Wireless Video Surveillance Networks," Multimedia Systems, Volume 14, Number 3, pp. 155-165.
12. X. Fu and B. l. Guo, "Framework for Distributed Video Surveillance in Heterogeneous Environment," 2008 International Conference on Intelligent Information Hiding and Multimedia Signal Processing, Harbin, 2008, pp. 826-829.
13. G. Maraviglia, et al., "Synchronous Multipoint E-Learning Realized on an Intelligent Software-Router Platform over Unicast Networks: Design and Performance Issues," Proc. ETFA 2007, Patras, Grece, Sept. 25-28, 2007.
14. A. Handojo, D. S. Frederik and J. Andjarwirawan, "Broadcast learning system using multicast address," 2010 2nd International Conference on Education Technology and Computer, Shanghai, 2010.
15. A. Lombardo, G. Schembra, "Performance evaluation of an Adaptive-Rate MPEG encoder matching IntServ Traffic Constraints," IEEE Transactions on Networking, vol. 11, no. 1, pp. 47-65, February 2003.
16. N. Wang *et al.*, "Traffic Engineered Multicast Content Delivery Without MPLS Overlay," IEEE Trans. on Multim., v. 9, n. 3, April 2007.
17. G. Busà, A. Lombardo, M. Barbera, G. Schembra, "CLAPS: A Cross-Layer Analysis Platform for P2P video Streaming," Proc. IEEE ICC 2007, GLASGOW, Scotland (UK), 24-28 June 2007.
18. T. T. Nguyen and C. Bonnet, "Efficient Multicast Content Delivery over a Distributed Mobility Management Environment," 2013 IEEE 78th Vehicular Technology Conference (VTC Fall), Las Vegas, NV, 2013.
19. 5G PPP Architecture Working Group, "View on 5G Architecture," July 1st, 2016.
20. ETSI NFV GS NFV 003, Network Function Virtualization (NFV) Terminology for Main Concepts in NFV, v1.2.1, Dec 2014.
21. ETSI NFV GS, Network Function Virtualization (NFV) Management and Orchestration, NFV-MAN 001 v0.8.1, Nov 2014.
22. W. Shi, *et al.*, "Edge Computing: Vision and Challenges," in IEEE Internet of Things Journal, vol. 3, no. 5, Oct. 2016.
23. A. Lombardo, M. Barbera, C. Panarello, G. Schembra, "Active Window Management: an efficient gateway mechanism for TCP traffic control", Proc. IEEE ICC 2007, GLASGOW, Scotland (UK), 24-28 June 2007.
24. H. Choi, S. Han, S. Park and E. Yang, "A CPU provision scheme considering virtual machine scheduling delays in Xen virtualized environment," TENCON 2009 - 2009 IEEE Region 10 Conference, Singapore.
25. L. Galluccio, et al., "An analytical framework for the design of intelligent algorithms for adaptive-rate MPEG video encoding in next generation time-varying wireless networks," IEEE Journal on Selected Areas in Communications, Vol. 23 No. 2, February 2005.
26. M. Barbera, A. Lombardo, G. Schembra, M. Tribastone, "A Markov Model of a Freerider in a BitTorrent P2P Network," Proc. IEEE Globecom 2005, St. Louis, MO, USA, 28 Nov. – 2 Dec. 2005, pp. 985-989.