Modelling and Analysis of SDN and NFV Enabled Datacentre Networks

Joseph Billingsley*, Wang Miao*, Geyong Min*, Nektarios Georgalas† and Ke Li*
*Department of Computer Science, University of Exeter, UK
Email: {jb931, wang.miao, g.min, k.li}@exeter.ac.uk
†Research and Innovation, British Telecom, UK
Email: nektarios.georgalas@bt.com

***Abstract*—Consumer demand for better and faster online services requires datacentres to continually evolve to provide more powerful and flexible storage, processing and networking services. Software defined networking (SDN) and network function virtualisation (NFV) have been regarded as two key pillars for building next generation data centres. Analytical models provide a fast and cost effective approach to experiment with these new technologies. Although some interesting research findings have appeared in the literature regarding the performance of SDN and NFV in the datacentre, most work only considers these technologies in isolation which hardly reflects their practical deployment and cannot capture interaction effects between these two technologies. In order to achieve a deeper understanding of next generation datacentre networks, a comprehensive analytical model is developed in this work to investigate the performance of a datacentre network in the presence of multiple NFV service chains and a virtualised SDN implementation. The end-to-end latency is derived based on the developed model with different network parameters. Finally the accuracy of the proposed analytical model is validated by conducting comprehensive simulation experiments.**

I. Introduction

Emerging services such as Augmented and Virtual Reality, 4K video, and the Internet of Things will require incredible amounts of computational resources [1]. At the heart of many of these new use cases is the datacentre, providing the required volumes of processing, storage and networking resources. The traditional approach of 'scaling-up' a datacentre: acquiring more powerful yet more expensive equipment to meet demand is no longer tenable [2]. Faced with high capital and operating expenditure, service providers have been seeking technologies that allow for more efficient usage of available resources and simplify management of new and existing equipment. Increasingly, the solution to these problems has been virtualisation [3] and modern datacentres have embraced the concepts of network function virtualisation (NFV) and software defined networking (SDN).

Modern datacentres require components capable of providing functions such as load balancing, firewalls and intrusion detection systems. Traditionally these network functions would be provided by purpose engineered network hardware greatly hindering the network innovation. In an NFV enabled network, network functions are instead run on virtual machines on commodity hardware. These Virtual Network Functions (VNFs) can be moved, scaled or destroyed on demand, allowing for efficient placement and allocation of resources, significantly accelerating the deployment of new services.

Datacentres contain large interconnection networks that allow communication between servers. Software Defined Networking (SDN) allows for dynamic configuration of this network and the other datacentre components [4], [5]. A logically centralised SDN controller maintains a global view of the network. The controller provides instructions that describe how packets should be routed through the network to 'dumb' switches. This centralises the networks intelligence, simplifying management and allowing for complex and flexible networking structures.

SDN and NFV are often considered complementary technologies [6]; with NFV freeing services from particular servers and SDN separating services from switches, rapid deployment and configuration of services is possible. Unfortunately, whilst powerful technologies their great flexibility adds additional complexity to the design of data centres.

Analytical models can provide insight into datacentre design by formally defining the interactions between key parameters of the design such as the size of the datacentre, the supported services and the required performance. As SDN and NFV are complementary technologies and may often be deployed together, it is important to identify how their interactions can affect the performance of the datacentre. However existing research in modelling of both technologies has typically considered them in isolation.

Many methods of modelling SDN alone are available in the literature. Longo et al. [7] proposed a model of the reliability of a two layer hierarchical SDN controller. Azodolmolky et al. [8] also examine the two layer SDN controller but use network calculus to determine the worst case delay and the minimum buffer size required to prevent packet loss. Wang et al. [9] developed a more realistic SDN model by considering the bursty and correlated arrivals of packets and a high and low priority queue at an SDN enabled switch. These models focus solely on SDN, ignoring the particular interactions between SDN and the network it would be deployed on.

Research on NFV modelling has also had a narrow focus. Prados-Garzon et al. [10] produced a detailed model of a single VNF which is composed of several VNF components and calculated the average response time of the VNF. Gebert et al. [11] analysed a single VNF in detail, modelling each queue in the packet processing pipeline of a Linux x86 system. To the best of our knowledge, only Fahmin et al. [12] have considered both NFV and SDN. However they consider a simplified network with only one switch and one VNF.

In this work a comprehensive analytical model is developed to investigate the performance of a datacentre network in the presence of multiple NFV services and a virtualised SDN implementation. The impact of multiple NFV service chains of varying lengths coexisting on the same physical network is considered as are interactions with the SDN controller.

The remainder of this paper is organised as follows. Section II discusses the details of the network architecture that is modelled in this work. In Section III we derive the analytical model for the network. Section IV validates the accuracy of this model with extensive simulation experiments. Finally Section V concludes the paper.

II. Network Architecture

With NFV a service is provided by forming several virtual network functions into a service chain where packets must pass through each of the VNFs in sequence. Different service chains may be composed of different numbers and types of VNF. Additionally many services may be provided by the datacentre simultaneously.

Service chains may be physically distributed over the datacentre. Communication between servers in the datacentre is provided by the interconnection network. The fat-tree or folded-Clos topology is currently the most common topology used for interconnection networks in datacentres [13]. The fat-tree topology (see Figure 1) is formed of three layers of switches: Core, Aggregation and Edge. Switches at the edge layer are additionally connected to servers. In an NFV enabled datacentre each of these servers contains a virtual switch which manages one or more VNFs.

The fat-tree topology is dependent upon the number of ports at each switch. We define k as the number of ports for each physical switch and k_{vsw} as the number of ports for each virtual switch. There are $(k/2)^2$ core switches. Each core switch connects to one switch in each of k pods. Each pod contains two layers (aggregation and edge) of $k/2$ switches. Each edge switch is connected to each of the $k/2$ aggregation switches of the pod. Each edge switch is connected to $k/2$ servers. Each server contains a virtual switch connected to k_{vsw} VNFs. This topology results in $n = (k^3/4) \cdot k_{vsw}$ VNFs.

In an SDN enabled datacentre an SDN controller provides centralised management, instructing the switches how to direct traffic to ensure it takes an efficient route to its destination. Each SDN enabled switch has a flow table maintained by the controller containing instructions on where to send received packets. This table may not be large enough to contain instructions for all possible destinations. If the local switch receives a packet that it does not have matching instructions for, it must request instructions from the controller. As a result a portion of the packets in the datacentre visit the controller. For this work we consider an SDN architecture where only the virtual switches connect to the SDN controller.

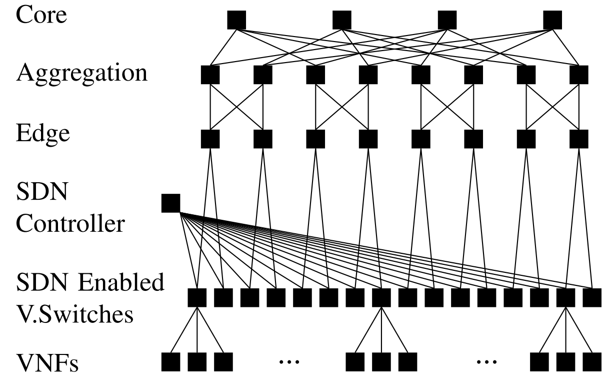

Fig. 1. An example SDN and NFV enabled fat-tree network with 4 ports for each hardware switch and 3 for the virtual switches.

This architecture is representative of those used in industry, most notably a comparable architecture is used in VMWare's NSX software [14].

III. Analytical Model

A. Assumptions

In a datacentre each VNF, physical or virtual switch and the SDN controller contain a queue where packets are buffered before being processed. To model the time a packet will wait at each queue we must consider three things: the probability distribution of the inter-arrival rate, the probability distribution of the service rate and the size of the queue.

It is reasonable to expect that packets, that may come from different users or different sources, will be independently distributed from each other. Similarily, in an efficient system the time taken to service a packet should not be dependent on earlier packets. Hence we can consider the arrival and service rates of packets to rollow independant probability distributions.

In a real datacentre each queue must be bounded to within a certain size as to do otherwise would require infinite memory or storage. However in practice the buffers should be large enough that buffer overflows are rarely an issue. For the purposes of this work we will assume queues are effectively infinite.

Finally the placement of VNFs must be considered. An important consideration when placing a VNF is considering the impact on the wider network. So as to evenly distribute the load across the network we will assume that VNFs are uniformly distributed across the datacentre.

Following this reasoning, the following assumptions are made with regards to the construction of the network:

1) At each VNF, packets are generated according to an independent Poisson process with a mean rate of α packets a cycle. Furthermore, packet destinations are uniformly distributed across the VNFs.
2) Each physical/virtual switch, VNF and the controller services packets according to an independent Poisson process with a mean rate of μ_{sw}, μ_{vnf} and μ_{sdn} packets a second respectively.

3) The time taken for a packet to travel between datacentre components is negligible.
4) The SDN controller ensures packets take one of the shortest paths between the source and destination and that packets are evenly distributed over the switches in the datacentre.
5) Queues at each network component have infinite capacity.
6) Packets leaving a server will visit the SDN controller with probability p_{miss_route}.

B. Derivation of Model

Before extending the model to complex service chains we consider the base case where the datacentre provides only one service formed with a chain of two VNFs. Hence packets are only required to cross the network once. As a result of the network topology, packets sent between two VNFs only need to travel as high as their first common ancestor. As the packets will always take an efficient path, the average latency is dependant on the probability a packet must visit a certain layer switch and the waiting time incurred at each component on the path:

$$\begin{aligned} &Latency_{path}(\alpha, \mu_{sw}, \mu_{vnf}, \mu_{sdn}) \\ &= l_{vsw} \cdot p_{vsw} + l_{edge} \cdot p_{edge} \\ &+ l_{agg} \cdot p_{agg} + l_{agg} \cdot p_{core} \end{aligned} \tag{1}$$

where:

$$\begin{aligned} l_{vsw} &= w_{vnf} + w_{vsw} \\ l_{edge} &= l_{vsw} + w_{sdn} + w_{vsw} + w_{edge} \\ l_{agg} &= l_{edge} + w_{edge} + w_{agg} \\ l_{core} &= l_{agg} + w_{agg} + w_{core} \end{aligned} \tag{2}$$

and w_{vnf}, w_{sdn}, w_{vsw}, w_{edge}, w_{agg} and w_{core} represent the average time spent at a VNF, the SDN controller and virtual, edge, aggregate and core switches respectively. Similarly p_{vsw}, p_{edge}, p_{agg} and p_{core} represent the probability that the highest level switch a packet visits is a virtual, edge, aggregate or core switch respectively. We now deduce these values for arbitrary numbers of ports for the physical (k) and virtual (k_{vsw}) switches.

1) Probability of Highest Level: If the source and destination VNFs share the same virtual switch they will not need to visit a higher level switch. Hence the probability of a packet only visiting a virtual switch is the probability the destination is under the same virtual switch as the source:

$$p_{vsw} = \frac{k_{vsw} - 1}{n - 1} \tag{3}$$

where n denotes the total number of VMs in the datacenter.

Whilst higher level switches cover more destinations, there may be shorter routes available to some of these destinations. The probability that the highest layer a packet visits is the edge layer is the probability the destination is under the same edge switch as the source, excluding those destinations that could be visited via a shorter route:

$$p_{edge} = \frac{(k/2) \cdot k_{vsw} - k_{vsw}}{n - 1} \tag{4}$$

This same principle can be used to deduce the probability of visiting the aggregate and core layers:

$$p_{agg} = \frac{(k/2)^2 \cdot k_{vsw} - (k/2) \cdot k_{vsw}}{n - 1} \tag{5}$$

$$p_{core} = \frac{n - (k/2)^2 \cdot k_{vsw}}{n - 1} \tag{6}$$

Finally, as the SDN controller will only be consulted if the destination VNF is on a different server to the source VNF, the probability of a packet visiting the controller is the probability of the destination being outside of the server and the virtual switch being unable to process it:

$$p_{sdn} = (1 - p_{vsw}) \cdot p_{miss\_route} \tag{7}$$

2) Calculation of Mean Waiting Time: As not every packet visits every layer but traffic is evenly distributed over the switches, the waiting time is the same at each component on a layer but can vary over layers. To determine the mean waiting time at each network component, each component is modelled as a M/M/1 queue where the mean waiting time is calculated with [15]:

$$f_w(\mu, \lambda) = \frac{1}{\mu - \lambda} \tag{8}$$

where μ is the service rate and λ is the arrival rate for a given component in the datacentre.

As destinations are evenly distributed over the VNFs, each VNF will receive an equal proportion of packets from every other VNF. Hence the arrival rate for each VNF is $(n-1) \cdot \frac{1}{n-1} \cdot \alpha$ which can be simplified to:

$$\lambda_{vnf} = \alpha \tag{9}$$

Virtual switches can receive packets from three sources. All packets generated by VNFs on the server must visit the virtual switch to reach any destination. Additionally, an equal portion of the traffic generated by VNFs on other servers will be intended for each of the VNFs under the virtual switch. Finally all of the traffic sent to the SDN controller must return to the virtual switch to reach higher level switches. Therefore the arrival rate at the virtual switch can be calculated as:

$$\begin{aligned} \lambda_{vsw} = &\, k_{vsw} \cdot \alpha \\ &+ (n - k_{vsw}) \cdot \frac{k_{vsw}}{n-1} \cdot \alpha \\ &+ k_{vsw} \cdot p_{sdn} \cdot \alpha \end{aligned} \tag{10}$$

Packets visiting the SDN controller do not affect the the arrival rate for higher level switches. While packets that are sent to the controller are not forwarded to higher level switches immediately, their absence is matched by packets returning from the SDN controller.

The arrival rate for the edge switches can be deduced in a similar way. The edge switch has more VNFs under it than the virtual switch. However packets that are intended for destinations on the same server as the source VNF do not need to visit the edge switch. Considering this, the arrival rate at the edge switch can be calculated as:

$$\begin{aligned}\lambda_{edge} = (k/2) \cdot k_{vsw} \cdot \frac{(n - k_{vsw})}{n-1} \cdot \alpha \\ + (n - ((k/2) \cdot k_{vsw}) \cdot \frac{(k/2) \cdot k_{vsw}}{n-1} \cdot \alpha\end{aligned} \tag{11}$$

Similarily, the aggregate switch allows access to more destinations than the edge switch. However destinations that share an edge switch with the source VNF can be visited in a more efficient manner. Additionally all traffic will be balanced between each aggregate switch in the pod. The arrival rate at the aggregate switch is hence:

$$\begin{aligned}\lambda_{agg} = \Big((k/2)^2 \cdot k_{vsw} \cdot \frac{(n - k_{vsw} \cdot (k/2))}{n-1} \cdot \alpha \\ + (n - (k/2)^2 \cdot k_{vsw}) \cdot \frac{(k/2)^2 \cdot k_{vsw}}{n-1} \cdot \alpha\Big) \cdot \frac{1}{k/2}\end{aligned} \tag{12}$$

As all VNFs are under each of the core switches the arrival rate at each core switch is the portion of traffic that must visit a core switch, split evenly between each of the core switches. Therefore the arrival rate at the core switch is:

$$\lambda_{core} = p_{core} \cdot n \cdot \alpha \frac{1}{(k/2)^2} \tag{13}$$

Finally, all VNFs will send a portion of the messages they produce to the controller. Therefore, the arrival rate at the SDN controller is:

$$\lambda_{sdn} = n \cdot p_{sdn} \cdot \alpha \tag{14}$$

By substituting the arrival rates (Equations 9 to 13) and service rates ($\mu_{sw}, \mu_{vnf}, \mu_{sdn}$) of each network component into $f_w(\mu, \lambda)$ we can calculate the average waiting time at each VNF and switch: w_{vnf}, w_{vsw}, w_{edge}, w_{agg}, w_{core}.

A visit to the SDN controller requires waiting at two queues. When a packet is sent to the controller it will first wait at the controller and then at a virtual switch when it returns. The additional waiting time incurred by the SDN controller is therefore:

$$w_{sdn} = (f_w(\mu_{sdn}, \lambda_{sdn}) + w_{vsw}) \cdot p_{sdn} \tag{15}$$

By substituting the probabilities of the different paths and the mean waiting times at each component into Equation 1, we can determine the average latency in the network for the case of a single pass through the network.

3) Multiple NFV Services with Different Length Chains: Existing research into NFV modelling has only considered the case of a single service requiring a single pass through the network. However in practice, datacentres may provide several services with different length service chains.

As service chains increase in length, packets will persist in the network for longer. Each packet a VNF receives that has not completed it's service will eventually be forwarded on to another VNF. At the same time it is also producing new packets. Hence, we can model this as the VNF effectively producing more packets. As packets only persist for the length of the service, the effective production rate is given by:

$$\alpha_{eff} = \alpha \cdot (len(service_i) - 1) \tag{16}$$

where len is the number of network functions that compose a given service and $service_i$ is the service being modelled.

Furthermore, if several services of different lengths were supported by the network, the average time a packet persisted in the network is dependent on the average service chain length. As not all services may produce packets at the same rate, if a given packet has probability $p(service_i)$ of belonging to $service_i$, the expected service length determines the effective production rate:

$$\alpha_{eff} = \alpha \cdot \sum_{i=1}^{ns} p(service_i)(len(service_i) - 1) \tag{17}$$

where ns is the number of different services and $\sum_{i=1}^{ns} p(service_i) = 1$.

The network must be crossed to visit each VNF in the service chain. The end-to-end latency will be the sum of the time spent taking each path. Using the derivation for the case of a single crossing of the network, the average latency for multiple services with variable length service chains is given by:

$$\begin{aligned}Latency = &Latency_{path}(\alpha_{eff}, \mu_{sw}, \mu_{vnf}, \mu_{sdn}) \\ &\cdot \sum_{i=1}^{ns} p(service_i)(len(service_i) - 1)\end{aligned} \tag{18}$$

where $Latency_{path}$ is given by Equation 1 and α_{eff} is given by Equation 17. For convenience, pseudocode for the entire process is given in Algorithm 1.

Algorithm 1 Calculation of Average Latency

1: Calculate $p_{vsw/edge/agg/core/sdn}$ (*Equations 3 - 7*)
2: Calculate $\lambda_{vnf/vsw/edge/agg/core/sdn}$ (*Equation 9 - 14*)
3: Calculate $w_{vnf/vsw/edge/agg/core/sdn}$ (*Equations 1, 15*)
4: Calculate effective prod. rate: α_{eff} (*Equation 17*)
5: Calculate one path latency: $Latency_{path}$ (*Equation 1*)
6: Calculate average latency: $Latency$ (*Equation 18*)

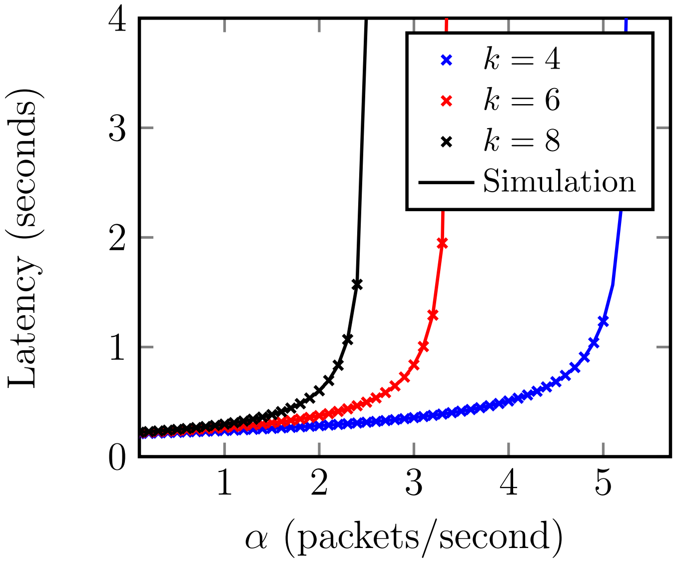

Fig. 2. Latency predicted by the model and simulation for different numbers of ports (k).

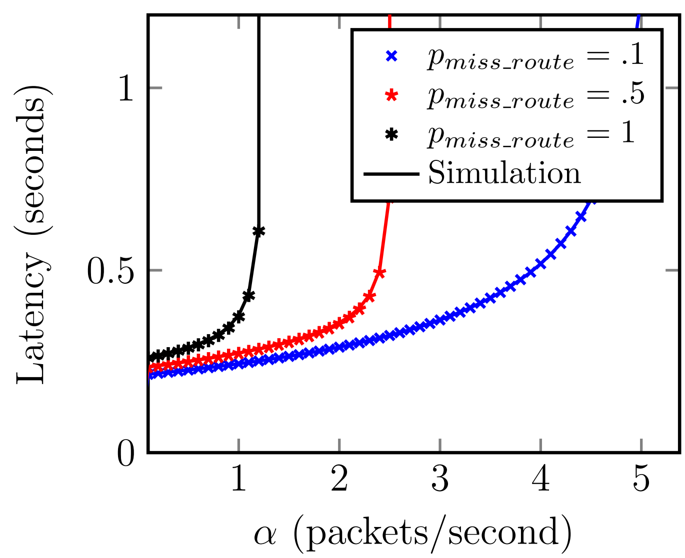

Fig. 3. Latency predicted by the model and simulation with different proportions of packets visiting the SDN controller (p_{miss_route}).

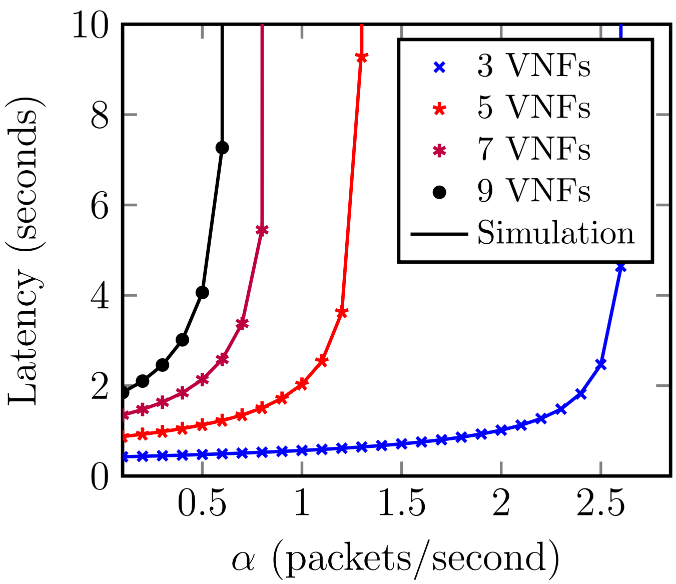

Fig. 4. Latency predicted by the model and simulation for different length service chains.

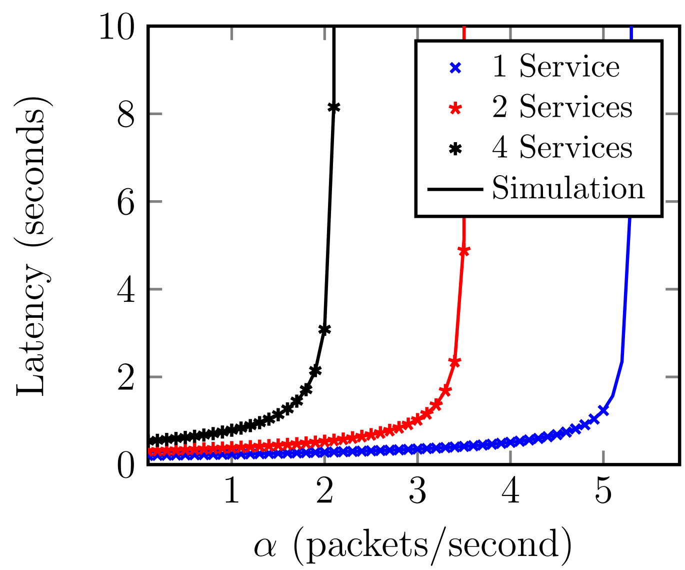

Fig. 5. Latency predicted by the model and simulation for several services with different length service chains.

IV. Validation

To verify the accuracy of the analytical model, a discrete event simulator has been built using OMNeT++ [16] to simulate a NFV and SDN enabled datacentre network. Each simulation experiment was run until the network reaches its steady state where further network cycles do not change the collected statistics appreciably.

Comprehensive simulation experiments were conducted to validate the performance of the proposed analytical model under different network configurations. However for the sake of specific illustration only a selection of tests are presented here and the results comparison between the analytical model and simulation experiments are presented in terms of the average end-to-end latency.

In practice a datacentre can contain on the order of tens of thousands of servers [17], with each switch supporting 1 to 100Gbits/s traffic a second. Unfortunately it is computationally expensive to simulate a datacentre at this scale, especially for a large number of tests. Additionally, the results do not produce a better comparison than smaller instances of the problem. Hence a scaled down version of a typical datacentre is modelled with the following parameters:

- $k = 4$, $k_{vsw} = 2$ and $p_{miss\_route} = 0$
- The service rate of the switches and SDN controller are

set to be 40 packets per second ($\mu_{sw} = 40$, $\mu_{sdn} = 40$)
- The service rate of the VNFs is set to be 20 packets per second ($\mu_{vnf} = 20$)
- Services are selected with equal probability
- Case I: The network holds one service with two VNFs
- Case II: The network holds multiple services and the number of VNFs in the ith service chain has a length of $i + 1$

Figures 2 to 5 depict mean message latency predicted by the model plotted against those provided by simulation experiments for a range of parameter settings. For the model, results are only shown where the network is in a steady state, i.e. where the arrival rate is lower than or equal to the service rate for all queues. The figures demonstrate that the simulation results closely match those predicted by the model. The tractability and accuracy of the analytical model make it suitable for analysis of next generation NFV and SDN enabled datacentre networks.

V. Conclusion

Emerging services will place intense demand on the datacentre. To provide a good service whilst remaining economically viable, modern datacentres are exploring the potential of SDN and NFV to provide efficient allocation of resources and simplify management. Whilst these are often considered complementary technologies, previous analytical models in the literature have typically considered them in isolation. Further previous work on this topic has not considered the importance of the interconnection network or how multiple services with different length service chains may affect performance.

In this paper we have presented a comprehensive analytical model capable of modelling an SDN and NFV enabled datacentre. Extensions are derived that accurately model how the presence of multiple services with varying length service chains impacts the datacentre. The resulting comprehensive model provides insight into the interactions between SDN and NFV that can be applied in the planning and design of datacentres.

References

[1] J. G. Andrews, S. Buzzi, W. Choi, S. V. Hanly, A. E. Lozano, A. C. K. Soong, and J. C. Zhang, "What will 5g be?" *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1065–1082, 2014.

[2] A. Vahdat, M. Al-Fares, N. Farrington, R. N. Mysore, G. Porter, and S. Radhakrishnan, "Scale-out networking in the data center," *IEEE Micro*, vol. 30, no. 4, pp. 29–41, 2010.

[3] W. V. Heddeghem, S. Lambert, B. Lannoo, D. Colle, M. Pickavet, and P. Demeester, "Trends in worldwide ICT electricity consumption from 2007 to 2012," *Computer Communications*, vol. 50, pp. 64–76, 2014.

[4] H. Kim and N. Feamster, "Improving network management with software defined networking," *IEEE Communications Magazine*, vol. 51, no. 2, pp. 114–119, 2013.

[5] S. Hares and R. White, "Software-defined networks and the interface to the routing system (I2RS)," *IEEE Internet Computing*, vol. 17, no. 4, pp. 84–88, 2013.

[6] J. Matías, J. Garay, N. Toledo, J. Unzilla, and E. Jacob, "Toward an sdn-enabled NFV architecture," *IEEE Communications Magazine*, vol. 53, no. 4, pp. 187–193, 2015.

[7] F. Longo, S. Distefano, D. Bruneo, and M. Scarpa, "Dependability modeling of software defined networking," *Computer Networks*, vol. 83, pp. 280–296, 2015.

[8] S. Azodolmolky, P. Wieder, and R. Yahyapour, "Performance evaluation of a scalable software-defined networking deployment," in *Second European Workshop on Software Defined Networks, EWSDN 2013, Berlin, Germany, October 10-11, 2013*. IEEE Computer Society, 2013, pp. 68–74.

[9] W. Miao, G. Min, Y. Wu, H. Wang, and J. Hu, "Performance modelling and analysis of software-defined networking under bursty multimedia traffic," *TOMCCAP*, vol. 12, no. 5s, pp. 77:1–77:19, 2016.

[10] J. Prados-Garzon, P. Ameigeiras, J. J. Ramos-Muñoz, P. Andres-Maldonado, and J. M. López-Soler, "Analytical modeling for virtualized network functions," in *2017 IEEE International Conference on Communications Workshops, ICC Workshops 2017, Paris, France, May 21-25, 2017*. IEEE, 2017, pp. 979–985.

[11] S. Gebert, T. Zinner, S. Lange, C. Schwartz, and P. Tran-Gia, "Performance modeling of softwarized network functions using discrete-time analysis," in *28th International Teletraffic Congress, ITC 2016, Würzburg, Germany, September 12-16, 2016*, T. Hoßfeld, B. L. Mark, S. G. Chan, and A. Timm-Giel, Eds. IEEE, 2016, pp. 234–242.

[12] A. Fahmin, Y. Lai, M. S. Hossain, Y. Lin, and D. Saha, "Performance modeling of SDN with NFV under or aside the controller," in *5th International Conference on Future Internet of Things and Cloud Workshops, FiCloud Workshops 2017, Prague, Czech Republic, August 21-23, 2017*. IEEE Computer Society, 2017, pp. 211–216.

[13] Cisco, "Cisco global cloud index: Forecast and methodology, 2016-2021," 2018, [Online; accessed 2018-05-04]. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/global-cloud-index-gci/white-paper-c11-738085.html

[14] VMware, "Network virtualisation and security platform - nsx," 2018, [Online; accessed 2018-05-04]. [Online]. Available: https://www.vmware.com/uk/products/nsx.html

[15] L. Kleinrock, *Theory, Volume 1, Queueing Systems*. Wiley-Interscience, 1975.

[16] A. Varga and R. Hornig, "An overview of the omnet++ simulation environment," in *Proceedings of the 1st International Conference on Simulation Tools and Techniques for Communications, Networks and Systems & Workshops, SimuTools 2008, Marseille, France, March 3-7, 2008*, S. Molnár, J. R. Heath, O. Dalle, and G. A. Wainer, Eds. ICST/ACM, 2008, p. 60.

[17] J. Hamilton, "Aws re:invent 2016," 2016, [Online; accessed 2018-05-04]. [Online]. Available: https://www.youtube.com/watch?v=AyOAjFNPAbA