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. The accuracy of the proposed analytical model is validated by conducting comprehensive simulation experiments. To illustrate its application, the proposed model is used the study the performance limits of datacentre networks.**

I. Introduction

In recent years demand for high performance computing has accelerated, driven by modern compute-intensive business and scientific applications. The imminent arrival of new, demanding use cases such as the Internet of Things stands to raise demand even further as billions more devices are connected to the internet [1]. Many of these new use cases are reliant on datacentres, large sites capable of providing tremendous amounts of computing power. With large datacentres containing tens of thousands of servers [2], management of a modern datacentre poses a variety of challenges. Key among them are high capital and operating expenditure and slow deployment of new services. Two technologies that will help solve these issues are Network Function Virtualisation (NFV) and SDN (Software Defined Networking).

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. In an NFV enabled network, virtual network functions (VNFs) are run on virtual machines on commodity hardware. These VNFs can be moved, scaled or destroyed on demand, allowing for flexible placement and allocation of resources and rapid 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 [3], [4]. 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 and routers. This centralises the networks intelligence, simplifying management and allowing for new and complex networking structures.

SDN and NFV are often considered complementary technologies [5], with the flexible placement enabled by NFV meshing well with the flexible routing allowed by SDN. Despite this, 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. [6] proposed a model of the reliability of a two layer hierarchical SDN controller. Azodolmolky et al. [7] 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. [8] 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.

As with SDN, NFV modelling has similarly had a narrow focus. Prados-Garzon et al. [9] 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. [10] analysed a single VNF in detail, modelling each queue in the packet processing process of a Linux x86 system. To the best of our knowledge, only Fahmin et al. [11] have considered both NFV and SDN, they modelled the performance of two methods of combining SDN and NFV in the network. However they consider a simplified network with only one switch and one VNF.

These aforementioned studies have several drawbacks. By considering NFV and SDN in isolation, existing models cannot model interactions between the technologies. Additionally the datacentre interconnection network is rarely considered.

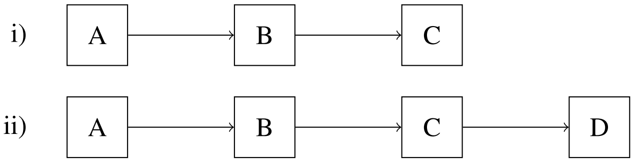

Fig. 1. Two service chains of different lengths represented with directed acyclic graphs. Packets must pass through each VNF in sequence. These and other services may exist in the network at the same time

Further current models assume only a single service will be provided and do not consider multiple NFV service chains or service chains composed of several network functions. To this end the key contributions of this paper are:

- An efficient analytical model of a joint NFV and SDN enabled datacentre network is proposed that considers the datacentre interconnection network
- Extensions for modelling of multiple NFV service chains of different lengths are derived
- Useful properties about the performance limits of an NFV and SDN enabled datacentre are derived

The rest 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. Section V explores the implications of the model and Section VI concludes the paper and examines future research directions.

to increase resource utilisation, reduce capital and operational expenditure, and enhance business agility and continuity

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. Service chains can be represented with Directed Acyclic Graphs (DAG), as in Figure 1, which encapsulate the dependencies between the VNFs. 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 [12]. The fat-tree topology (see Figure 2) 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.

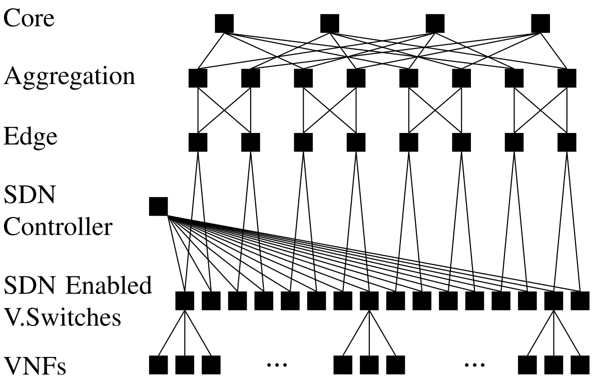

Fig. 2. An example SDN and NFV enabled fat-tree network with 4 ports for each hardware switch and 3 for the virtual switches.

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 routing table maintained by the controller containing instructions on where to send received packets. This routing 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. This architecture is representative of those used in industry, most notably a comparable architecture is used in VMWare's NSX software [13].

III. Analytical Model

In a datacentre each VNF, physical or virtual switch and the SDN controller contain a queue where packets are buffered before being processed. Subsequently the datacentre is modelled in the same way with each component modelled as an M/M/1 queue.

A. Assumptions

For the derivation of the analytical model, 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:

$$\begin{aligned} &Latency_{pass}(\alpha, \mu_{sw}, \mu_{vnf}, \mu_{sdn}) \\ &\quad= (w_{vnf} + w_{vsw}) \cdot p_{vsw} \\ &\quad+ (w_{vnf} + w_{sdn} + 2w_{vsw} + w_{edge}) \cdot p_{edge} \\ &\quad+ (w_{vnf} + w_{sdn} + 2w_{vsw} + 2w_{edge} + w_{agg}) \cdot p_{agg} \\ &\quad+ (w_{vnf} + w_{sdn} + 2w_{vsw} + 2w_{edge} \\ &\qquad\qquad + 2w_{agg} + w_{core}) \cdot p_{core} \end{aligned} \tag{1}$$

where 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 settings of k and k_{vsw}.

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:

$$p_{vsw} = \frac{k_{vsw} - 1}{n - 1} \tag{2}$$

Whilst higher level switches cover more destinations, there may be shorter routes available to some of these destinations. The probability of a packet visiting at highest an edge switch 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:

$$p_{edge} = \frac{(k/2) \cdot k_{vsw} - k_{vsw}}{n - 1} \tag{3}$$

This same principle can be used to deduce the probability of visiting an aggregate or core switch:

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

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

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:

$$p_{sdn} = (1 - p_{vsw}) \cdot p_{miss\_route} \tag{6}$$

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 [14]:

$$f_w(\mu, \lambda) = \frac{1}{\mu - \lambda} \tag{7}$$

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:

$$\lambda_{vnf} = \alpha \tag{8}$$

Virtual switches can receive packets from three sources. All packets generated by VNFs on the server must pass through 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:

$$\begin{aligned} \lambda_{vsw} &= k_{vsw} \cdot \alpha \\ &\quad+ (n - k_{vsw}) \cdot \frac{k_{vsw}}{n - 1} \cdot \alpha \\ &\quad+ k_{vsw} \cdot p_{sdn} \cdot \alpha \end{aligned} \tag{9}$$

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

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:

$$\begin{aligned}\lambda_{edge} = (k/2) \cdot k_{vsw} \cdot \frac{(n - k_{vsw})}{n-1} \cdot \alpha \\ + (n - ((k/2) \cdot k_{vsw}) \cdot \frac{(k/2) \cdot k_{vsw}}{n-1} \cdot \alpha\end{aligned} \tag{10}$$

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:

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

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:

$$\lambda_{core} = p_{core} \cdot n \cdot \alpha \frac{1}{(k/2)^2} \tag{12}$$

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

$$\lambda_{sdn} = n \cdot p_{sdn} \cdot \alpha \tag{13}$$

By substituting the arrival rates (Equations 8 to 12) 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 switch.

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

$$w_{sdn} = (f_w(\mu_{sdn}, \lambda_{sdn}) + w_{vsw}) \cdot p_{sdn} \tag{14}$$

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 Services with Different Lengths: 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.

An important consequence of longer service chains is each packet persisting in the network for a longer period of time. Consider a situation where each VNF deterministically sends a packet to an adjacent VNF every second. Consider also that we have a service chains with three network functions so that packets will be required to cross the network twice. After one second all VNFs will have sent and received one packet. After two seconds all VNFs will have sent two packets, forwarding the packet received in the previous step and a new packet. It will have also received two packets, a packet with no VNFs left to visit and a packet with one VNF remaining. After three seconds one packet will be destroyed having completed the service, leaving one packet to be forwarded and one new packet created for each VNF. Effectively, each VNF is producing two packets per second on average.

We can extend this intuition to arbitrary length services. The longer the service grows, the longer messages will persist in the network leading to higher effective production rates. Following this intuition, the effective production rate for an arbitrary length service is:

$$\alpha_{eff} = \alpha \cdot (len(service_i) - 1) \tag{15}$$

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:

$$\alpha_{eff} = \alpha \cdot \sum_{i=1}^{ns} p(service_i)(len(service_i) - 1) \tag{16}$$

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

As each destination VNF in the service chain requires crossing the network, the average latency will be the sum of the time spent taking each path. Using the derivation for the case of a single pass through the network, the average latency for multiple services with variable length service chains is given by:

$$\begin{aligned}Latency = & Latency_{pass}(\alpha_{eff}, \mu_{sw}, \mu_{vnf}, \mu_{sdn}) \\ & \cdot \sum_{i=1}^{ns} p(service_i)(len(service_i) - 1)\end{aligned} \tag{17}$$

where $Latency_{pass}$ is given by Equation 1 and α_{eff} is given by Equation 16.

IV. Validation

To verify the accuracy of the analytical model, a discrete event simulator has been built using OMNeT++ [15] 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.

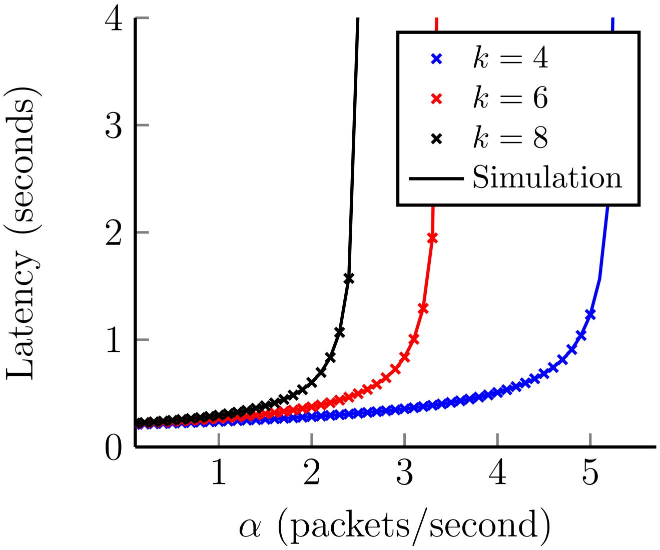

Fig. 3. Latency predicted by the model and simulation for different numbers of ports (k).

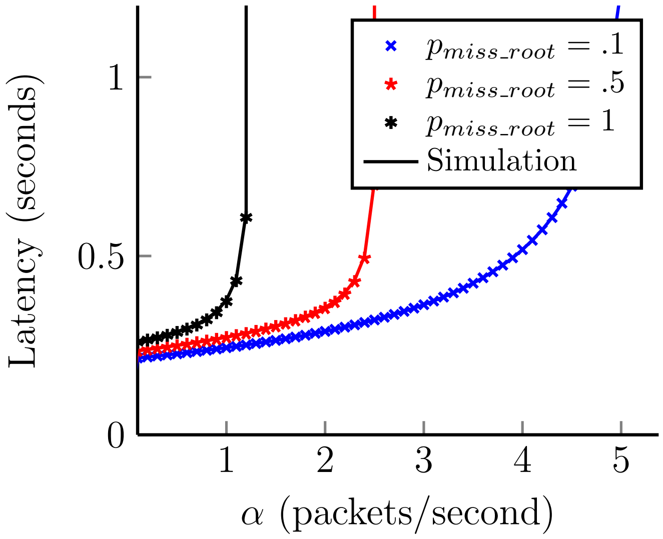

Fig. 4. Latency predicted by the model and simulation with different proportions of packets routing via the SDN controller (p_{miss_route}).

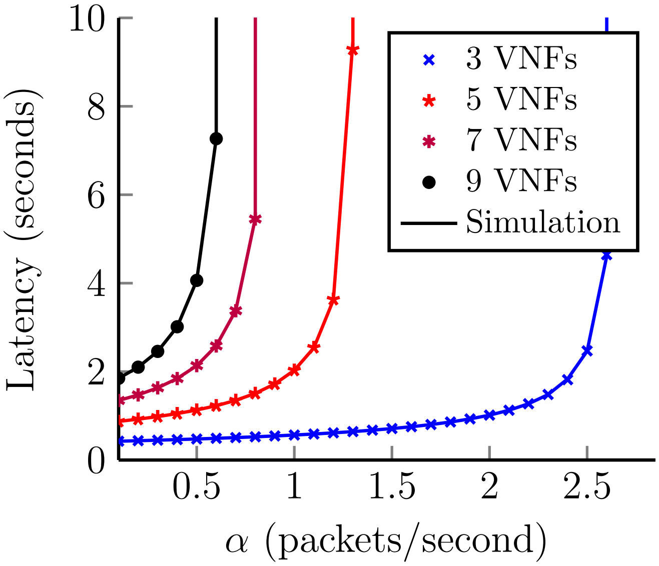

Fig. 5. Latency predicted by the model and simulation for different length service chains.

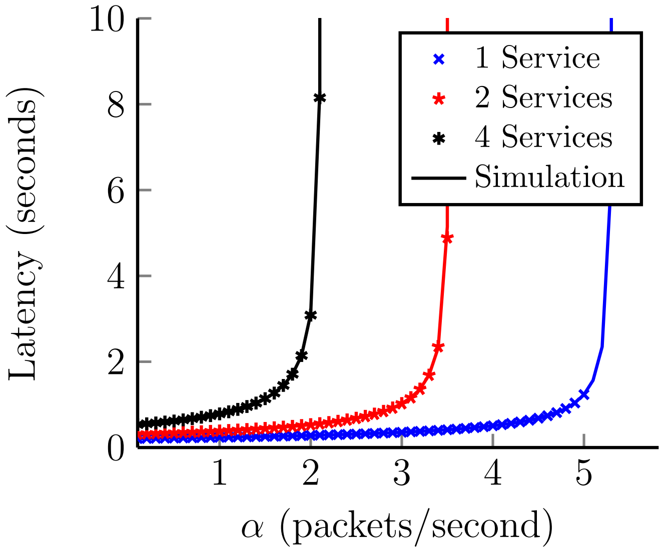

Fig. 6. Latency predicted by the model and simulation for several services of different lengths.

Numerous validation experiments were performed for several combinations of network sizes, service lengths, number and probability of selection, and p_{miss_route}. To remain concise latency results are presented for selected cases. For all cases, where not otherwise specified, the following parameter settings are used:

- $k = 4$, $k_{vsw} = 2$ and $p_{miss\_route} = 0$
- All switches and the SDN controller have the same service rate of 40 packets per second ($\mu_{sw} = 40$, $\mu_{sdn} = 40$)
- The VNFs have a service rate of 20 packets per second ($\mu_{vnf} = 20$)
- The network holds one service with two VNFs
- Services are selected with equal probability

Additionally, when multiple services in the same network are considered (as in Figure 6) each service is assigned a length the same as its index plus one. This ensures the tests have different average lengths.

Figures 3 to 6 depict mean message latency predicted by the model plotted against those provided by a discrete event simulator 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. Performance Analysis

Having validated its accuracy, the analytical model can now be used to deduce properties of SDN and NFV enabled networks. In particular it is useful to determine which layer will receive the most traffic so as to identify likely bottlenecks. We first determine the proportion of traffic the edge switch receives compared to the other switches and simplify the resulting expression:

$$\frac{\lambda_{edge}}{\lambda_{agg}} = 2(k^2 - 1) \geq 1 \tag{18}$$

$$\frac{\lambda_{edge}}{\lambda_{core}} = \frac{k(k^2 - 1) - 2}{k(k^2 - 1 - k/2)} \geq 1 \tag{19}$$

These equations show that the edge switches receives more traffic than the aggregate and core switches when $k > 2$.

Similarily from the definition of the arrival rates for the edge switches (Equation 10) and the VNFs (Equation 8) it is clear that edge switches will also receive more traffic than the VNFs.

The portion of traffic that visits the SDN controller is dependent on the parameter p_{miss_route}. The minimum value of p_{miss_route} that will cause the SDN controller to receive a higher traffic rate than the edge switches can be found when $lambda_{sdn} = lambda_{edge}$. Rearranging and simplifying the resultant equation gives:

$$p_{req\_sdn\_miss} = \frac{k_{vsw} \cdot k \cdot (n - k_{vsw} \cdot \frac{2k+4}{8})}{n \cdot (1 - \frac{k_{vsw}-1}{n-1})} \tag{20}$$

a value of $p_{miss\_route} > 1$ indicates that there is no setting of this parameter which can cause the SDN controller to receive more traffic than an edge switche.

The same technique can be used to calculate the minimum value of $p_{miss\_route} > 1$ for a virtual switch to receive more traffic than an edge switch.

$$p_{req\_vsw\_miss} = \frac{k}{n - k_{vsw}} \cdot \left(n - k_{vsw}\left(\frac{k}{4} + \frac{1}{2}\right)\right) - 2 \tag{21}$$

VI. Conclusion

Management of modern, large scale datacentres suffer from high capital and operating expenditure and slow deployment of new services. Two technologies that can improve this situation by increasing the flexibility of the datacentre and simplifying management are SDN and NFV. 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 did not consider the importance of the interconnection network or how the presence of more than one service or different length services may impact the datacentre.

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. Finally useful properties pertaining to the performance of the network are derived from the mathematical model. These show that the edge switches, virtual switches and SDN controller can receive disproportionately more traffic than the other components in the datacentre.

References

[1] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, “Internet of things (iot): A vision, architectural elements, and future directions,” *Future Generation Comp. Syst.*, vol. 29, no. 7, pp. 1645–1660, 2013.

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

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

[4] 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.

[5] 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.

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

[7] 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.

[8] 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.

[9] 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.

[10] 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.

[11] 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.

[12] 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

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

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

[15] 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. [Online]. Available: https://doi.org/10.4108/ICST.SIMUTOOLS2008.3027