Analysis of Traffic Engineering capabilities for SDN-based Data Center Networks

Pilimon, Artur; Kentis, Angelos Mimidis; Ruepp, Sarah Renée; Dittmann, Lars

Published in:
Proceedings of SDN NFV World Congress 2018

Link to article, DOI:
10.1109/SDS.2018.8370445

Publication date:
2018

Document Version
Peer reviewed version

Analysis of Traffic Engineering capabilities for SDN-based Data Center Networks

Artur Pilimon, Angelos Mimidis Kentis, Sarah Ruepp and Lars Dittmann
Department of Photonics Engineering
Technical University of Denmark
{artpil, agmimi, srru, ladit}@fotonik.dtu.dk

***Abstract*— In the recent years more and more existing services have moved from local execution environments into the cloud. In addition, new cloud-based services are emerging, which are characterized by very stringent delay requirements. This trend puts a stress in the existing monolithic architecture of Data Center Networks (DCN), thus creating the need to evolve them. Traffic Engineering (TE) has long been the way of attacking this problem, but as with DCN, needs to evolve by encompassing new technologies and paradigms. This paper provides a comprehensive analysis of current DCN operational and TE techniques focusing on their limitations. Then, it highlights the benefits of incorporating the Software Defined Networking (SDN) paradigm to address these limitations. Furthermore, it illustrates two methodologies and addresses the scalability aspect of DCN-oriented TE, network and service testing, by presenting a hybrid physical-simulated SDN enabled testbed for TE studies.**

***Keywords*—Data Center; SDN; Traffic Engineering.**

I. INTRODUCTION

Over the years, Data Centers (DC) and their supporting network infrastructure have gone through several important stages of architectural transformation. The scope of this transformation was to create a more economically feasible, reliable and energy-efficient communication environment, capable of accommodating ever growing data storage, processing and distribution demands.

Data Center Networks (DCN) can undoubtedly be regarded as the digital backbone of the global economy. DCN provide the business and mission critical communication for a vast range of entities (e.g. governments, business enterprises and private users). This diversity of the userbase translates into fundamentally different communication requirements in terms of reliability and service availability and Quality of Service (QoS) guarantees (bandwidth, latency, on-demand resource scaling, speed of failure recovery, etc.). As a result, Data Center businesses (large DC operators, wholesale DC solution providers and enterprise-level operators of DC facilities) are continuously challenged to look for new and more efficient ways of operating the existing DCNs, as well as defining novel strategies of building new DC facilities.

In this paper, we provide an analysis of traffic engineering (TE) capabilities for Software Defined Networking (SDN) based DCN environments and highlight the advantages and shortcomings of different strategies. The purpose of this study is to provide a condensed analysis of the most important DCN-oriented TE approaches, reported in a variety of other extensive surveys [1][2][3][4][5][6], but which cover a very broad range of SDN TE areas. In addition, we illustrate two methodologies for evaluating the benefits of traffic engineering capabilities in SDN controlled DCNs. The remainder of this paper is organized as follows: Section II explores and analyzes the TE capabilities offered by SDN in the context of DCNs. This includes overview of the common DC operation strategies, existing TE methodologies and their limitations, as well as discussion of the benefits of applying principles of SDN in TE. Section III discusses the need for TE in modern DCNs, and section IV highlights two methodologies for testbed evaluation of SDN TE features. Section V concludes the paper.

II. DATACENTER NETWORK OPERATION STRATEGIES

The initiatives for finding efficient ways of operating the existing DCNs, as well as defining novel approaches of building new DC facilities, have developed into three main strategies, widely adopted by the DC industry, namely:

1) DC resource overprovisioning;
2) DC resource usage optimization;
3) Building business-specific custom DC solutions.

The main features of each of these approaches are briefly highlighted as follows. The first strategy (**DC resource overprovisioning**) tackles the DC resource demand problem, should this be compute, storage or network infrastructure, by dimensioning the DCN for the "worst-case" scenario. In this case, the maximum possible all-to-all communication and service provisioning (including processing and storage) demands are taken as an input to the network dimensioning "formula". As a result, excessive redundancy of the allocated resources is supposed to handle the communication and service provisioning demands; however, there is a range of factors, which play a definitive role in making this strategy unfeasible, both from the economic and technical point of view. First, the potential Capital Expenditures (CAPEX) may be too high. Second, network performance degradation can occur even in an overprovisioned DCN environment. For example, traffic congestion can occur due to inefficient scheduling of the traffic flows or memory buffer limitations of the network devices. Therefore, some network links may be heavily loaded (leading to highly variable delays, jitter, packet loss) whilst other may be underutilized, creating imbalance of resource usage.

The second approach (**DC resource usage optimization**) is utilizing different optimization techniques, originating from research and industrial best practices, with the overall goal of increasing the efficiency of DCN's resource utilization, improving the QoS for services and applications in a shared DCN, as well as reducing the operational costs (OPEX). In this context, Traffic Engineering (TE) is an integral part of the

network optimization philosophy. As reported in [8], the average resource utilization of DC resources was ranging from 17.76% - 60% for computing resources, up to 77.93% for internal memory resources and up to 75.28% for disk storage. However, in most cases, the DCN resources are highly underutilized. By applying suitable TE techniques we can eliminate or alleviate the resource usage inefficiency and performance degradation threats, in addition to a potential reduction of OPEX. Adoption of the Software Defined Networking (SDN) paradigm [9] in the context of TE in DCNs is offering great improvements compared to the currently used TE solutions, and introduces new opportunities to fully exploit the potentials of modern DCN infrastructures.

The third strategy (**building business-specific custom DC solutions**) was derived by the web-scale service providers, such as Google, Amazon, Facebook, Netflix and alike. These service providers developed custom DCN designs with tailored TE solutions for their operational business needs, rather than trying to adapt industry standard solution and deal with arising scalability, performance and efficiency problems. Such solutions include, for example, Google's Jupiter architecture [10] or Facebook's Fabric [11] or custom routing platforms. However, these solutions cannot be directly reused in any other DCN environment, because they are built for specific use-cases, dictated by business demands.

III. The Need For Traffic Engineering in Modern Data Center Networks

According to the Global Cloud Index forecast [12], global Data Center IP traffic (intra-DC, DC-to-DC and DC-to-user) is expected to increase 3-fold by 2020, reaching 15.3 zettabytes. Other trends, which have significant effect on the concentration and distribution of the global DC workloads, as well as growth of intra-DC traffic volumes are [12]:

1) Growth of the hyper-scale DCs (estimated 53% of total DC traffic);
2) Virtualization and Cloud computing growth (up to 92% of workloads will be processed in Cloud DCs by 2020);
3) Rise of public and private Clouds;
4) Fast proliferation of new types of applications and services with diverse resource requirements (Big Data Analytics, Internet of Things, Multimedia content, Map-Reduce processing models).

To overcome the challenges imposed by these factors, flexible and highly efficient network resource management and TE approaches are required. In the following sub-sections, we discuss the key objectives of TE in DCNs (*A*) and evolution of TE approaches (*B*), limitations of the current TE approaches and why they cannot be effectively used in the SDN-based DCNs (*C*), as well as emphasize the main benefits and potentials of SDN-based TE to optimize the resource usage in DCNs (*D*) and outline some open research challenges in the context of SDN (*E*).

A. The objectives of TE techniques for DCNs

TE can be seen as an iterative process of performance optimization, carried out as a combination of monitoring and measurement techniques, static or dynamic adjustment of core relevant operational parameters. The ultimate goal of this process is to reach and sustain a carefully defined performance objective. It is important to distinguish the main performance objectives, usually pursued by TE (optimization mechanisms); however, integration of several TE objectives into one mechanism can be unfeasible, because certain TE goals can be mutually exclusive (e.g., network performance in terms of latency/throughput and energy efficiency) [1] [4][6]:

Minimization of network congestion: This is one of the most important performance objectives in communication networks, DCNs in particular. Congestion is one of the most significant problems, which directly affects other associated performance metrics, such as packet loss, latency and jitter. The problem affects the operational state of the DCNs more than that of a Wide Area Network (WAN), due to higher concentration of traffic with highly variable, bursty profiles and sub-second lifetime of vast majority of the flows within the DC (East-West traffic). This means that the critical performance tweaking decisions must be made in a very fast and dynamic on-line fashion. This performance metric can be optimized by utilizing the multipath redundancy of DCN architectures and spreading the traffic over the DCN infrastructure, relocating the resources for established flows (e.g., by disaggregation of large flows) or performing resource scheduling and access control (e.g., granting access only to uncongested resources of the network).

Minimization of the end-to-end (E2E) delay: This is a critical parameter in the context of DCNs, since a dominant volume of traffic flows within a DCN are short-lived and small-sized ("mice flows"); as a result, a typical flow duration in a DCN may be a few orders of magnitude shorter than in a long-haul transport network. On the other hand, DCs may be hosting business critical applications and services with very stringent delay requirements. Thus, it is of uttermost importance to keep the upper bound of this metric as low as possible, e.g., by utilizing efficient flow scheduling/load balancing or routing algorithms, such as Constrained-Shortest Path First (CSPF), which uses E2E delay as a path selection constraint. This parameter is also correlated with another metric, widely used as a complementary QoS assessment parameter in multimedia transmission services, namely Quality of Experience (QoE).

Maximization of the Energy Usage Efficiency: The focus of this objective is on minimizing the energy consumption of the DCN infrastructure by applying a set of techniques, including flexible resource consolidation and workload migration to be able to power down the network (nodes, ports, line cards) and processing (servers, storage disks and arrays) elements, which have become idle. Other approaches may include application of advanced optical switching techniques within the DCN interconnect (intra-DCN) [13].

Optimization of the resource utilization: Bandwidth, packet buffer space as well as CPU (Central Processing Unit) processing capacity are critical resources, which must be used in the most efficient way. Unavailability of any of these DCN resources will directly impact the performance of the hosted applications and services, affected by the packet loss, increase in queueing delays and flow completion times. This can be achieved by, e.g., applying flexible flow scheduling mechanisms, such as Weighted constrained ECMP (Equal Cost Multi-Path) for weight-based flow distribution over

multiple available paths of the same cost, or by scheduling well characterized data transfers (e.g., backup flows, updates) at certain time of the day, when the availability of processing and transmission resources may be higher.

Minimization of the packet loss: This objective is a derivative of the global Congestion minimization objective, and an applied TE strategy largely depends on the detected root cause of the packet loss, since packets can be lost as a result of network congestion, protocol/algorithmic failures or queue management mechanisms (e.g., Drop-Tail or Active Queue Management) activated in the network devices.

B. The evolution of TE techniques and applicability in DCNs

Traditionally, considering data packet networks, there have been different TE approaches introduced, as described in RFC 3272 [14]. However, the communication networks were rapidly evolving and these techniques were not satisfying the new performance optimization requirements anymore, as highlighted by Awduche in [15].

One of the first successfully adapted TE strategies for the Internet traffic was a concept of overlay model/network, with the best example of this being IP over ATM (Asynchronous Transfer Mode) [16]. This was realized by the means of using virtual circuits and virtual path concepts, which allowed defining multiple virtual topologies over the physical infrastructure. However, the downside was the increased operational complexity of the networks, in addition to the fact that the circuits had to be pre-provisioned – limiting the flexibility of possible TE manipulations, especially for real-time applications and in the context of DCN scale factor.

Another important step in the evolution of TE was the introduction of the Shortest Path First (SPF) algorithms, which could take the Type of Service (ToS) [17] specifications into account when choosing routing paths. However, the limitation of this method led to unfair sharing of the network resources, where the SPF-based shortest paths were ending up being overloaded (congested), whilst other paths were severely underutilized. Next solution enabled better utilization of the multi-path redundancy in the infrastructures, with multiple sets of paths of equal cost (as can be found in DCN environments), called ECMP [18]. It allowed equal traffic splitting among multiple available shortest paths by using a flow hashing technique; this method is widely used (in different modifications) in DCN environments up to date.

Multi-Protocol Label Switching (MPLS) [7][19] emerged as a traffic forwarding architecture, with a goal of increasing the flexibility, performance and scalability of the network layer routing, where the core packet processing (classification, marking) functionality shifted to the edge of the MPLS domain, while performing fast packet switching based on label information. This introduced a whole range of more advanced TE capabilities, including the explicit routing, traffic disaggregation and aggregation by applying multi-tunneling (multiple levels of encapsulation).

Finally, a PCE-based (Path Computation Element) architecture was proposed by IETF for MPLS and Generalized MPLS (GMPLS) networks [20]. In this solution, there is a dedicated element (e.g. a server) responsible for path computation, supporting explicit routing (point-to-point and point-to-multipoint label switched paths, LSPs). Hence, it is being currently adapted and used for intra-domain, inter-domain and inter-layer TE applications, in Optically-switched networks in particular [21].

C. Limitations of the traditional state-of-the-art TE approaches in the context of DCNs

As mentioned in Section I, the operational performance requirements in DCNs are fundamentally different compared to the classical transport/WAN environments, and the shortcomings of the traditional TE approaches are described as follows (more extensive general overviews can be found in [1][4][6]):

Non-optimal path computation algorithms: Available research results in [22] show that even when using MPLS-TE solution with CSPF algorithms for path computation and network resource management, increased latency was observed in numerous experiments. This was a result of combined impact of the CSPF algorithms used, auto-bandwidth scaling functionality (part of MPLS-TE framework), which led to continuous path changes due to automatic bandwidth adjustments on the LSPs (depending on the variations in traffic demands). In DCNs, where there might be millions of simultaneous mostly short-lived connections, this strategy would introduce delayed TE decisions, which would continuously affect the network state and potentially result in suboptimal paths being chosen.

Unbalanced traffic disaggregation (splitting) ratios: There are two ways to perform traffic splitting, either per-packet or per-flow. In the former, multiple performance problems may occur due to the inherent properties of the TCP (Transmission Control Protocol, being the dominant transport protocol in DCNs), such as performance drop due to packet reordering and subsequent false fast retransmissions and transmission window reductions [23]. In addition, using ECMP for traffic splitting may lead to congestion on the shortest paths, since the traffic is equally spread based on flow hashing in a uniform fashion; hence, e.g., large and small flows may collide on the shortest chosen paths.

Slow update rates of the TE Databases (TED): In this situation the contents of the TED do not reflect the network state in real-time, which may be a critical requirement for a DCN environment with fast changes in flow dynamics (flow arrival and completion, burstiness, etc.). Even though the current PCE-based TE methods can address the limitations of the traditional MPLS-TE, it is still facing a real-time data consistency problem [20], because this element entirely relies on the information stored in the TED for path computation and optimization.

Long convergence time of distributed protocols [4]**:** The problem of all the discussed so far approached is that they use in-band signaling for resource management, consuming network resources. Also, both MPLS-TE and PCE-based solutions rely on the RSVP-TE protocol (resource ReSerVation Protocol), which is an extra delay component, because once the path is computed by PCE, this information needs to be propagated to all the nodes of the path, and the resources are reserved after this reservation is confirmed by all these nodes. This aspect poses a direct scalability problem of this TE mechanism and leads to variable convergence times of

the network state information, affecting the number of flows that can be accommodated in a DCN within a short time unit.

D. The benefits of SDN in addressing TE challenges in DCNs

SDN, as a relatively new networking paradigm, introduced a highly flexible framework, allowing for tackling of the TE challenges, discussed in sub-section *C*. The key functional mechanisms of SDN, which enable intelligent TE decisions to optimize operational performance of DCNs, are outlined as follows [1][4][6]:

Separation of the control and forwarding planes: This feature is a definitive factor, enabling the network programmability by offloading the control decisions from the network devices to a logically centralized control unit, called an SDN controller. Such functional separation eliminates extra resource consumption in the data plane, while the core control logic is enforced to the data plane via out-of-band control channel, using an OpenFlow [24] protocol or other supported Southbound D-CPI (Data-Control Plane Interface) protocol [4]. This factor enhances the scalability of the control plane, while facilitating fast data plane forwarding of traffic flows.

Logically centralized control plane: This means that the control plane is presented as a logically centralized abstraction, but different implementation strategies are possible, including the clustering of multiple SDN controllers together. Clustering of the control plane is a very important aspect in terms of scalability, since it allows for virtual slicing of underlying network resources, where each virtual network domain can be controlled independently or in a hierarchical fashion. The benefit of this is that the SDN control has a unified view of all the network resources, operational state of the components and traffic load in real-time. Hence, a single centralized entity can utilize all this information to perform path computation and flow rule placement based on the defined policies for a corresponding traffic type. In the context of DCNs, scalability of the control plane is a critical factor to handle multi-granular TE decisions at sub-second time scales. Hence, cluster-based deployment of multiple SDN controllers with efficient coordination and network state synchronization techniques is a reasonable approach for DCNs.

High network programmability: This feature is of utmost importance, since it allows dynamic flow rule installation in the network. In addition, SDN provides also much finer levels of granularity (multiple protocol headers and individual fields can be used) in the flow rule definition. This enables application of more advanced TE policies by, e.g., exploiting a multi-table architecture of the flow processing pipeline (logical flow processing sequence). It is possible to define a complex ruleset by utilizing three types of flow processing tables available, such as the standard Flow tables, group tables (flow group processing actions) and meter tables (for traffic shaping/rate control). SDN provides a flexible network abstraction model through a well-defined open source Southbound interface (SBI), allowing to build complex control applications via an extensible API (Application programming interface). When compared to closed and proprietary interfaces or vendor-specific legacy network switching and routing devices, SDN allows for more efficient exploitation of multi-path redundancy of DCNs and performance isolation in multi-tenant architectures of Cloud DCNs to account for individual QoS requirements of common Cloud service models (SaaS, PaaS, IaaS, etc.).

Advanced network monitoring and performance measurements: The SDN control framework can be used to collect (and calculate) fine-grained statistical performance data from the network devices by means of native SDN statistics polling or integration of traditional (e.g., SNMP, Simple Network Management Protocol) or more advanced (e.g., sFlow, NetFlow, or jFlow) network monitoring techniques. In this way it is possible to create a sophisticated network monitoring framework by combining the strengths of these solutions. In addition, the collected historical measurement data can be used as an input for use of ML/DL (Machine Learning/Deep Learning) techniques to define TE objectives based on the predicted performance evolution, so that timely localized TE adjustments can made instead of global dynamic TE actions, since the complexity, diversity and massive volume of DCN traffic profiles may hinder the practical feasibility of dynamic TE enforcement.

E. Open research questions in SDN in the DCN context

In general, there have been multiple initiatives proposed in the research community [1][4], targeting TE in SDN-based DCNs to address the issues of operational complexity in SDNs in path computation [25], improve routing [26][27] and resource utilization efficiency [28]. Most of these solutions have been tested in virtualized network environments and with different available open source SDN controllers (OpenDaylight, Floodlight, etc.). The main message in all works is that SDN offers more flexibility in testing of any developed TE methods, since the key logic is developed as a loadable module in a centralized framework. However, this flexibility comes at a price, and there are multiple challenges for the SDN-based TE as well. For example, the following major open questions have been identified by Akyildiz in [3]:

Scalability and availability: It has been noticed that under higher flow arrival rates at the flow table resources available in switches (Ternary Content Addressable Memory (TCAM), software tables), installation of flow rules was accompanied by increased latency and latency spikes. Thus, efficient flow management (load balancing) solutions are needed for the Control and Data planes.

Multiple flow tables: As noted, certain SDN-enabled switches have only a single flow table built around TCAM. This type of memory is space limited, and large volumes of flow may lead to huge rule sets to be installed, limiting large scale deployment capabilities. Hence, multi-table implementations will improve the situation.

Reliability: For large-scale SDN deployments, a cluster-based approach is needed with operational and backup controllers. However, there is no standardized protocol (e.g., like OpenFlow) that would provide any mechanism for coordination between the primary and backup controllers, therefore limiting the possibilities for fault tolerance of the control plane.

Other important issues are consistency of the topology updates (problems due to duplicate flow entries, stalled entries, etc.) and accuracy of traffic analysis (e.g., Big Data sampling challenges).

IV. Analysis of Traffic Engineering Properties in an SDN-enabled Testbed Environment

One of the challenges of testing limitations of TE capabilities in DCN is the lack of available large-scale testbeds. In this section, we illustrate two methodologies, based on experimental studies, which can be combined to enable realistic and scalable TE testing framework. In the first, we feature a testbed composed of HP Aruba DCN SDN switches, an SDN-enabled Polatis open-space optical circuit switch, and an external SDN controller. This testbed, was used for the final demonstration of EU FP7 COSIGN project [29]. In addition, we also present a hybrid (physical-simulated) testbed (for Proof-of-Concept please refer to [30]; a system without SDN capabilities is presented in [31]) for evaluating SDN TE characteristics at scale. We are using the hypercube structure illustrated in Fig. 1, due to its simplicity, flexibility and scalability properties.

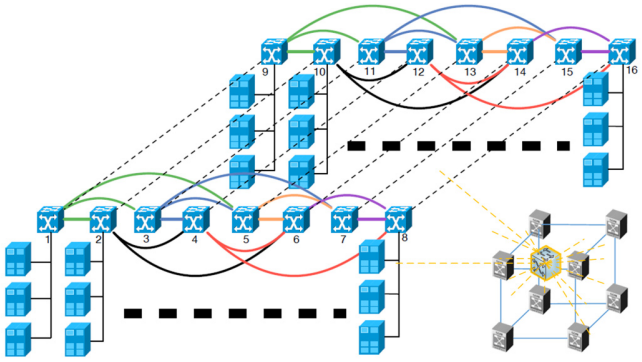

Fig. 1 Hypercube datacenter structure

The first study evaluates the latency figures, if SDN based TE is used for rerouting traffic via the Polatis switch to provide optical shortcuts in case of certain applications or network conditions. As a starting point multiple flows are routed via the shortest hypercube path. Once the threshold on the link reaches a specified or dynamically derived threshold (e.g. 70%), the traffic is re-routed through the optical shortcut. This TE feature is highly beneficial for low latency applications, load balancing and survivability in datacenters. The study steps are illustrated in Fig. 2.

The second test environment consists of a hybrid real-simulated setup, that allows linking of real networking equipment with a simulation model, in order to evaluate large scale SDN TE capabilities, that otherwise may not be possible due to CAPEX limitations [30]. The setup is shown in Fig. 3, with a model to real world connectivity exemplified in Fig. 4.

Implementation of a Hypercube-based simulation model consists of network switches, supporting Open Flow 1.3.0 protocol. The simulation model is connected to the real physical hardware (network equipment, servers) and an external SDN controller via specialized virtual System-in-the-Loop (SITL) gateways linked to the physical network interface(s) of a workstation/server, which the simulation environment is running on.

The applied simulation-based approach brings many benefits when evaluating the scalability of networks, datacenters and communication systems in general. The primary advantage is the real-time operation mode of the simulation model, meaning that realistic behavior and timing of processes within the simulated SDN-enabled network nodes are preserved. It is critical for these settings to be real-time to obtain accurate information of packet conversion times and buffering latencies at the boundary points between the real and simulated parts of the hybrid simulation model. The second advantage refers to the support of SDN principles via the implementation of the OpenFlow protocol in the simulated nodes.

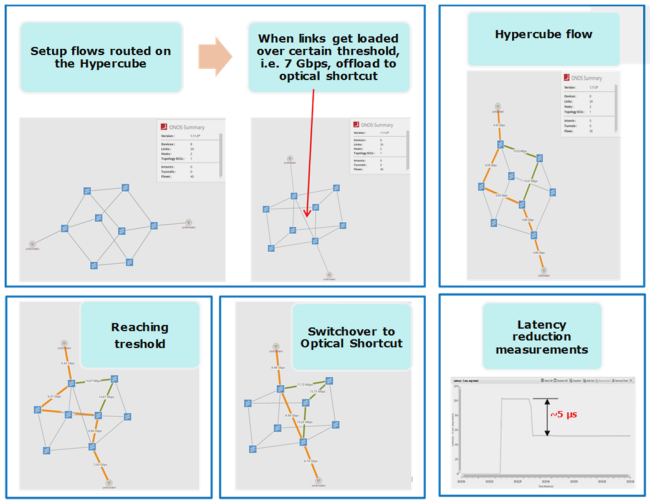

Fig. 2 Latency reduction study: SDN-based TE

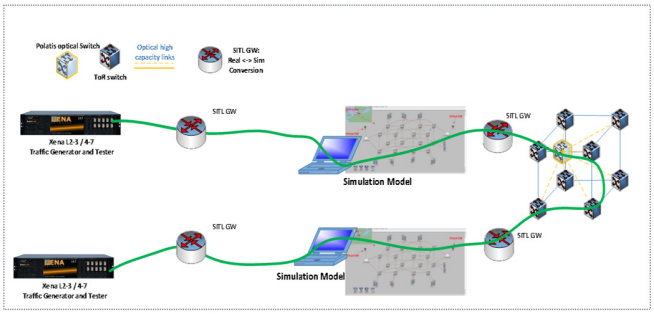

Fig. 3 Hybrid real-simulated SDN-enabled DCN testbed setup

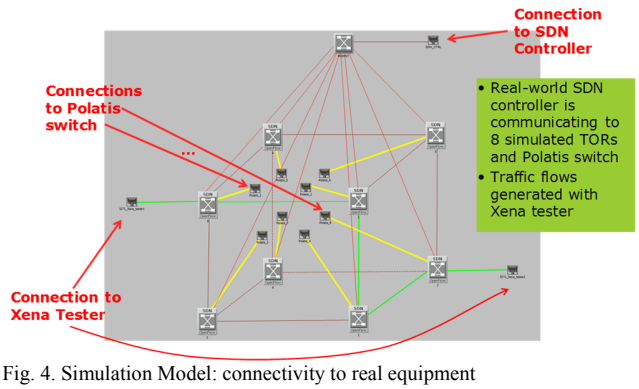

Fig. 4. Simulation Model: connectivity to real equipment

These features provide substantial flexibility in terms of reconfiguration, as the control logic only needs to be modified in the SDN controller instead of every individual simulated node, placement of additional real or hybrid components, as well as an increase of the scale of the network interconnect by

extending the simulation model. This approach is highlighted in Fig. 5, emphasizing that the scalability property can be achieved using advanced simulation and modeling techniques, which would allow us to assess SDN TE solutions in a repeatable manner with a greater degree of control over the infrastructure, configuration and traffic generation.

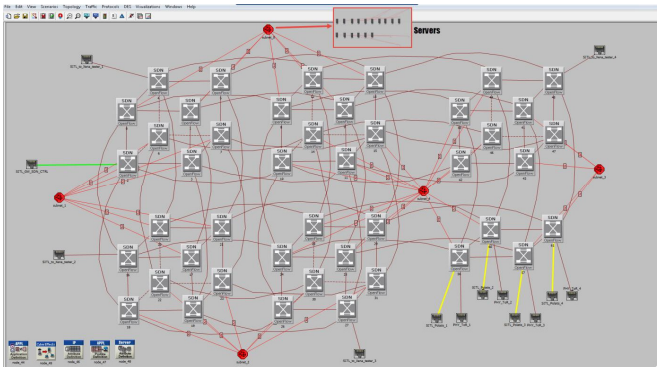

Fig. 5 A simulation model using an Incomplete Hypercube structure with 44 nodes and links to external DCN equipment [31]

V. CONCLUSIONS

In this work we have discussed the key limitations of the traditionally used TE approaches, taking into consideration the operational performance requirements in DCNs, due to completely different traffic profiles, with burstiness, critical bandwidth or ultra-low delay requirements, in addition to the sub-second lifetime of the dominant volume of all the traffic flows. These limitations include non-optimal path computation algorithms, slow TE Database convergence times, which is a critical requirement for the real-time control plane decisions. SDN-based TE solutions are discussed and the main benefits are distinguished with regards to how this new paradigm can greatly benefit the highly virtualized DCNs challenged by the new scalability, resource usage and operational efficiency requirements. The existing limitations of SDN-based approach are summarized as well.

We furthermore present two methodologies that can be used to evaluate traffic engineering capabilities in SDN. One is based on available hardware, whereas the other allows for large-scale datacenter evaluation by combining real equipment and a simulation environment.

REFERENCES

[1] I. F. Akyildiz, A. Lee, P. Wang, M. Luo, and W. Chou, "A roadmap for traffic engineering in software defined networks," *Comput. Networks*, vol. 71, pp. 1–30, June 2014.

[2] W. Xia, P. Zhao, Y. Wen, and H. Xie, "A Survey on Data Center Networking (DCN): Infrastructure and Operations," *IEEE Comm. Surv. and Tutor.*, vol. 19, no. 1. pp. 640–656, 2017.

[3] I. Akyildiz, A. Lee, P. Wang, M. Luo, and W. Chou, "Research challenges for traffic engineering in software defined networks," *IEEE Network*, vol. 30, no. 3, pp. 52–58, June 2016.

[4] A. Mendiola, J. Astorga, E. Jacob, and M. Higuero, "A Survey on the Contributions of Software-Defined Networking to Traffic Engineering," *IEEE Comm.. Surv. Tutor.*, vol. 19, no. 2, pp. 918–953, May 2017.

[5] J. Zhang, F. Ren, and C. Lin, "Survey on transport control in data center networks," *IEEE Netw.*, vol. 27, no. 4, pp. 22–26, Aug. 2013.

[6] M. Noormohammadpour and C. S. Raghavendra, "Datacenter Traffic Control: Understanding Techniques and Trade-offs," *IEEE Comm.. Surv. Tutorials*, vol. PP, no. 99, pp. 1–34, Dec. 2017.

[7] D. Awduche, J. Malcolm, J. Agogbua, M. O'Dell, and J. Mcmanus, "Requirements for Traffic Engineering Over MPLS," RFC 2702 1999.

[8] X. Sun, N. Ansari, and R. Wang, "Optimizing Resource Utilization of a Data Center," *IEEE Commun. Surv. Tutorials*, vol. 18, no. 4, pp. 2822–2846, Nov. 2016.

[9] P. A. Morreale, *Software Defined Networking: Design and Deployment*. CRC Press, 2014, pp. 1-186.

[10] A. Singh *et al.*, "Jupiter Rising: A Decade of Clos Topologies and Centralized Control in Google's Datacenter Network," in *Proc. ACM SIGCOMM*, pp. 183–197, London, UK, 2015.

[11] A. Roy, H. Zeng, J. Bagga, G. Porter, and A. C. Snoeren, "Inside the Social Network's (Datacenter) Network," in *SIGCOMM Comput. Commun. Rev.*, vol. 45, no. 4, pp. 123–137, 2015.

[12] Cisco, "Cisco Global Cloud Index : Forecast and Methodology, 2015–2020," *White Paper*. pp. 1–29, 2016.

[13] A. Pilimon, A. Zeimpeki, A. M. Fagertun, and S. Ruepp, "Energy Efficiency Benefits of Introducing Optical Switching in Data Center Networks," in *Proc. ICNC*, Santa Clara, CA, USA, 2017, pp. 891–895.

[14] D. Awduche, A. Chiu, A. Elwalid, I. Widjaja, and X. Xiao, "Overview and Principles of Internet Traffic Engineering," RFC 3272, 2002.

[15] D. O. Awduche, "MPLS and traffic engineering in IP networks," *IEEE Commun. Mag.*, vol. 37, no. 12, pp. 42–47, Dec. 1999.

[16] R. Cole, D. Shur, and C. Villamizar, "IP over ATM: A Framework Document," RFC 1932, 1996.

[17] K. Nichols, S. Blake, F. Baker, and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers," RFC 2474, 1998.

[18] C. Hopps, "Analysis of an Equal-Cost Multi-Path Algorithm," RFC 2992, 2000.

[19] E. Rosen, A. Viswanathan, and R. Callon, "Multiprotocol Label Switching Architecture," RFC 3031, 2001.

[20] A. Farrel, J.-P. Vasseur, and J. Ash, "A Path Computation Element (PCE)-Based Architecture," RFC 4655, 2006.

[21] T. Tsuritani, M. Miyazawa, S. Kashihara, and T. Otani, "Optical path computation element interworking with network management system for transparent mesh networks," in Proc. *OFC/NFOEC*, San Diego, CA, USA, 2008, pp. 1–10.

[22] A. Pathak, M. Zhang, Y. C. Hu, R. Mahajan, and D. Maltz, "Latency inflation with MPLS-based traffic engineering," in *Proc. Acm Sigcomm Conference*, Berlin, Germany, 2011, pp. 463–472.

[23] K. C. Leung, V. O. K. Li, and D. Yang, "An overview of packet reordering in transmission control protocol (TCP): Problems, solutions, and challenges," *IEEE Trans. Parallel Distrib. Syst.*, vol. 18, no. 4, pp. 522–535, Apr. 2007.

[24] Open Networking Foundation, "OpenFlow Switch Specification 1.4.0 (Wire Protocol 0x05)," 2013.

[25] L. A. Rocha, "Framework for Traffic Engineering of SDN Data Paths," *Adv. Appl. Sci.*, vol. 1, no. 2, pp. 37–45, Oct. 2016.

[26] L. Davoli, L. Veltri, P. L. Ventre, G. Siracusano, and S. Salsano, "Traffic Engineering with Segment Routing: SDN-Based Architectural Design and Open Source Implementation," *Proc. - Eur. Work. Softw. Defin. Networks, EWSDN*, Bilbao, Spain, pp. 111–112, 2015.

[27] K. T. Dinh, S. Kukliński, W. Kujawa, and M. Ulaski, "MSDN-TE: Multipath Based Traffic Engineering for SDN," in *ACIIDS 2016. Lecture Notes in Computer Science*, 2016, vol. 9622, pp. 630–639.

[28] M. Khoshbakht, M. Tajiki, and B. Akbari, "SDTE : Software Defined Traffic Engineering for Improving Data Center Network Utilization," *Int. J. Inf. Commun. Technol. Res.*, vol. 8, no. 1, pp. 15–26, Mar. 2016.

[29] "COSIGN: Combining Optics and SDN In next Generation data centre Networks," Public Deliverable 5.5, 2017. [Online]. Available: http://www.fp7-cosign.eu

[30] S. Ruepp, A. Pilimon, J. Thrane, M. Galili, M. Berger, and L. Dittmann, "Combining Hardware and Simulation for Datacenter Scaling Studies," in Proc. *ONDM*, Budapest, Hungary, 2017, pp. 1–6.

[31] A. Pilimon and S. Ruepp, "A Hybrid Testbed for Performance Evaluation of Large-Scale Datacenter Networks," *accepted for presentation at ICNC 2018, Maui, HI, USA*.