AI-Assisted Knowledge-Defined Network Orchestration for Energy-Efficient Datacenter Networks

Wei Lu, Lipei Liang, Bingxin Kong, Baojia Li, Zuqing Zhu

***Abstract*—In this article, we discuss the design and implementation of a novel datacenter network (DCN) system, which utilizes a knowledge-defined network orchestration mechanism (NO-M) to operate a hybrid optical/electronic DCN (HOE-DCN) cost-effectively and energy-efficiently. The motivations behind the proposed HOE-DCN system are the urgent needs to address the scalability, energy and manageability issues in the existing DCN systems. To realize the knowledge-defined NO-M, we follow the principle of predictive analytics in human brain to design three artificial intelligence (AI) modules based on deep learning (DL) and make them operate collaboratively. The proposed HOE-DCN system is implemented in a network testbed, and we conduct experiments that involve both control and data plane operations to demonstrate its advantages. The experimental results show that the HOE-DCN simultaneously achieves high-performance service provisioning and improved energy-efficiency. Furthermore, by analyzing the pros and cons of the HOE-DCN system, we also point out several directions to work on in the future.**

***Index Terms*—Datacenter networks (DCNs), Network orchestration, Energy saving, Knowledge-defined networking (KDN), Artificial intelligence (AI).**

I. Introduction

DRIVEN by the happening revolution on information and communication technology (ICT), cloud computing is penetrating academia, industry, and government sectors rapidly across the world. In the new context, customers can rent IT and network resources from the cloud service providers (CSPs) in a “pay-as-you-use” manner, which can significantly reduce the investment in self-built infrastructure. Therefore, the amount of global IP traffic generated by clouds has been growing at a compound annual growth rate of 30% since 2015 and would reach 14.1 zettabytes (1 zettabytes = 10^{21} bytes) in 2020, and this actually represents more than 92% of the total traffic within/among datacenters (DCs) [1]. Consequently, being the fundamental infrastructure of cloud computing, DC networks (DCNs) are under great pressure to accommodate such tremendous traffic with sustainable technologies [2]. This means that the network orchestration mechanisms (NO-Ms) used to manage the virtual machines (VMs) and network connections in DCNs should try to not only increase resource utilization but also reduce energy consumption.

The challenges of realizing a highly efficient NO-M are mainly two-fold. On one hand, the NO-M needs to control and manage a large number of heterogeneous network elements, which at least include the servers for VM deployment and the top-of-rack (ToR), aggregation and core switches for traffic routing. This is to say, it has to jointly optimize the allocations of multi-dimensional IT resources (*i.e.*, CPU cycles, memory and storage) on servers and bandwidth resources on network links [3]. Note that, DCNs purely based on electronic packet switching would be unsustainable soon [4, 5]. Hence, hybrid optical/electronic DCNs (HOE-DCNs) that use both high-speed Ethernet switches and optical circuit switch(es) to connect ToR switches will soon become increasingly common for new DCN deployments [4, 5]. This, however, will further complicate the network management part in an NO-M.

On the other hand, both the computing tasks and the traffic in a DCN can be highly dynamic. For example, the ratio between inter- and intra-rack traffic could change over time when different types of computing tasks are running, while for the same reason, the ratio between mice and elephant flows would also be time-variant. Therefore, an NO-M can hardly be efficient without precise knowledge on the characteristics of computing tasks and traffic in the DCN. Moreover, to reduce the setup latency of services and minimize the mismatch between allocated and utilized resources, the NO-M needs to predict future computing tasks and traffic accurately [6].

For these two challenges, the first one can be addressed by designing a centralized NO-M based on software-defined networking (SDN) [7, 8], while the idea of knowledge-defined networking (KDN) [9] might be leveraged to resolve the second one. KDN is a new networking paradigm that incorporates SDN, telemetry, data analytics, and artificial intelligence (AI). Specifically, with the global view and full control over a network provided by SDN, the KDN controller can collect rich telemetry information, conduct AI-assisted data analytics, abstract knowledge via deep learning (DL), and make wise decisions to achieve highly efficient network orchestration.

In this article, we first review the recent trends of DCN development from the perspectives of architecture scalability, energy efficiency and management agility. Then, to address the revealed issues, we design an HOE-DCN system with knowledge-defined NO-M. By leveraging DL-based AI, our NO-M achieves precise prediction and intelligent decision making, and thus can effectively improve both the service provisioning performance and energy-efficiency of the HOE-DCN. Next, we describe how to implement our proposal in a real testbed, and present the experimental results to demonstrate its advantages. Finally, we summarize the article.

W. Lu, L. Liang, B. Kong, B. Li and Z. Zhu are with the School of Information Science and Technology, University of Science and Technology of China, Hefei, Anhui 230027, P. R. China (email: zqzhu@ieee.org).

Manuscript received on April 10, 2018.

II. Review on Datacenter Networks (DCNs)

A. Architecture Scalability

Most deployed DCNs use electronic packet switches organized in hierarchical architectures to interconnect servers. However, they have sustainability and scalability issues due to the following two reasons. First of all, both the capital expenditure (CAPEX) from switches and the operational expenditure (OPEX) due to power consumption would increase rapidly, when the capacity of such a DCN is being expanded [4]. This will eventually make the DCNs unsustainable to support the future Mega DCs that can contain hundreds of thousands of servers and have tremendous bandwidth capacity requirements. Secondly, due to the best-effort nature of packet switching, it would be challenging for the DCNs to provide differentiated services to clients with various QoS requirements. On the other hand, by utilizing the tremendous bandwidth capacity in optical fibers, optical circuit switching (OCS) and optical packet switching (OPS) provide alterative solutions to flatten the DCN architecture for supporting large-scale DCs better. Nevertheless, the optical solutions are not perfect either, since the long switch configuration time of OCS would make the path setup latency unsuitable for delay-sensitive services and OPS is still not mature because of the lack of all-optical buffers and the complexity from optical signal processing [10].

Hence, the hybrid optical/electronic DCNs (HOE-DCNs) are recommended [4–6]. HOE-DCNs explore the advantages of both electronic packet switching (EPS) and OCS, and more importantly, they can enable a smooth transition from the traditional DCNs. As shown in Fig. 1(a), an HOE-DCN makes an EPS network work coordinately with an OCS network, *i.e.*, the ToR switches can be either interconnected with high-speed Ethernet switches (*i.e.*, to carry delay-sensitive and/or highly dynamic traffic) or an optical circuit switch (*i.e.*, to support bandwidth-intensive and/or long-lasting traffic). Note that, in reality, especially for a Mega DC, the HOE-DCN would have a more complex architecture than that in Fig. 1(a). This is because the available ports on a commercial OCS switch is usually limited (*i.e.*, in the magnitude of hundreds), and thus we cannot use a single OCS switch to interconnect all the ToR switches in a DC. Therefore, multiple OCS switches might be organized in a hierarchical manner (*e.g.*, the leaf-and-spine architecture) to realize a scalable HOE-DCN.

B. Energy Efficiency

It is known that DCs in the United States consumed approximately 70 billion kilowatt-hours (KWHs) in 2014, which accounts for $\sim$1.8% of the total electricity consumption of the country, and the number has been growing fast since then [11]. The energy consumption of a DC normally comes from two entities, which are the ICT equipment (*i.e.*, the servers, switches and other network elements in the DC) and the cooling and lighting facilities. Building DCs in cold places and/or using renewable energy resources can improve the energy efficiency of the latter, while there is also plenty of room to reduce the energy consumption of ICT equipment because the servers and switches in current DCs are usually in very low utilization (*e.g.*, 10%) [12].

The utilization on ICT equipment can be greatly improved with the virtualization technology that virtualizes the IT resources on servers and uses them for VM deployment. Hence, server load consolidation, *i.e.*, grooming VMs onto fewer servers and shutting down the idle ones, can be achieved for energy saving. Here, the overhead is that the associated VM migrations might generate huge volumes of traffic among servers and push the energy usage on related switches up. Therefore, the DC operator has to carefully steer the traffic to avoid unnecessary energy consumption [13, 14].

C. Management Agility

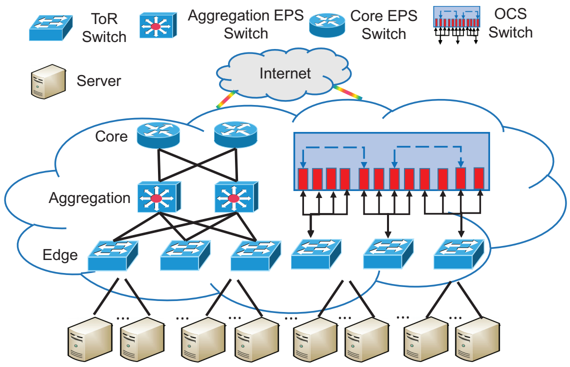
Traditional DCN management usually relies on vendor-dependent tools that can only be operated well by experts, and thus is error prone and difficult to operate and upgrade. Fortunately, SDN offers new opportunities to manage DCNs more easily, adaptively and reliably [7]. Under the paradigm of SDN, DCNs become more programmable, *i.e.*, the CSPs gain full control over the entire network system via a centralized and programmable orchestrator. There are several advantages due to this. Firstly, with a global view on its DCN, a CSP can manage the allocations of IT and bandwidth resources more effectively with the centralized orchestrator. Secondly, the orchestrator helps the CSP respond quickly to DCN status changes by collecting telemetry information proactively.

More promisingly, after gathering rich data about a DCN, we can incorporate the idea of KDN [9] in its management system by introducing AI-assisted data analytics. Hence, knowledge can be abstracted from the network data and used to achieve intelligent decision making [6]. Meanwhile, it is worth noting that there will be a tradeoff between the management agility and the complexity due to data collection and analytics. With the fast development of computing technologies/facilities and AI algorithms, the latter is becoming less and less problematic. For instance, one can easily improve the time efficiency of AI training by leveraging knowledge sharing among AI modules, parallel training, *etc*.

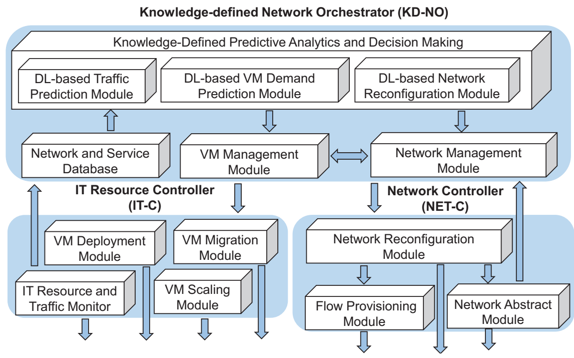
III. HOE-DCN with Knowledge-defined Network Orchestration

The HOE-DCN in Fig. 1(a) leverages a multi-tier EPS network and a flat OCS network to realize inter-rack interconnections. Here, the EPS network consists of edge, aggregation and core tiers. In the edge tier, the ToR switches connect the servers and organize them in racks. The switches in the aggregation and core tiers interconnect the ToR switches and bridge the communications among the servers in different racks. With packet-level switching granularity, the EPS network can support delay-sensitive and/or highly dynamic inter-rack traffic well, but may suffer from congestions caused by the oversubscription in the hierarchical structure.

The OCS network uses a reconfigurable OCS switch to interconnect the ToR switches, *i.e.*, the OCS switch can set up lightpaths with wavelength-level switching granularity to pump high-throughput traffic through. Therefore, bandwidth-intensive and/or long-lasting inter-rack traffic can bypass the



(a) Hardware architecture of HOE-DCN



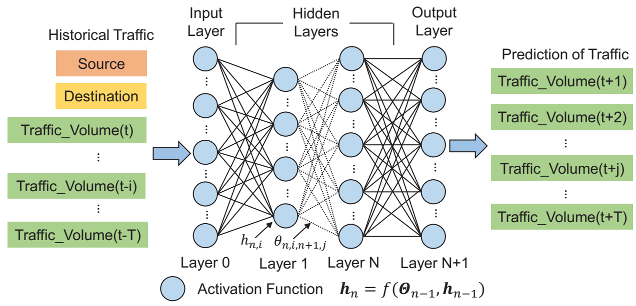
(b) Software modules in KD-NO

Fig. 1. Hybrid optical/electronic DCN (HOE-DCN) with a management system based on knowledge-defined NO-M.

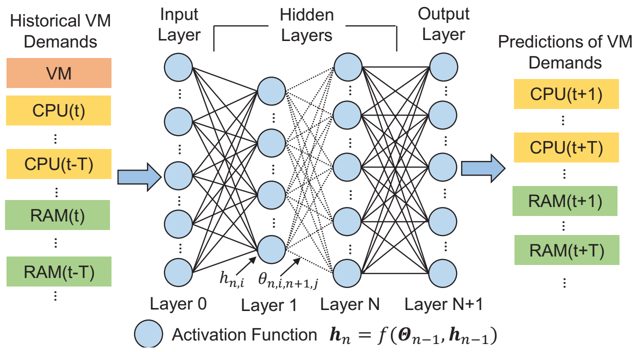
EPS network to avoid congestions. However, the typical configuration time of an OCS switch is in hundreds of milliseconds, which would introduce relatively long path setup latency. Hence, it would be critical to make the EPS and OCS networks cooperate well for accommodating various types of flows in the HOE-DCN, which motivates us to design a knowledge-defined network orchestration mechanism (NO-M).

Fig. 1(b) shows the design of our proposed management system to realize knowledge-defined NO-M for the HOE-DCN. Here, we design an IT Resource Controller (IT-C) and a Network Controller (NET-C) to manage the IT and bandwidth resources, respectively, and they are coordinated by a Knowledge-Defined Network Orchestrator (KD-NO). In the IT-C, the VM Deployment Module, VM Migration Module, and VM Scaling Module realize adaptive VM management in the servers, while the IT Resource and Traffic Monitor collects the statistics about IT resource usage and traffic periodically to send to the KD-NO. The NET-C consists of three modules. The Flow Provisioning Module accommodates flows according to their types, and routes their traffic to minimize the energy usage of switches. The Network Reconfiguration Module reconfigures the OCS switch to adapt to inter-rack traffic, while the Network Abstract Module collects the statistics about inter-rack traffic to forward to the KD-NO.

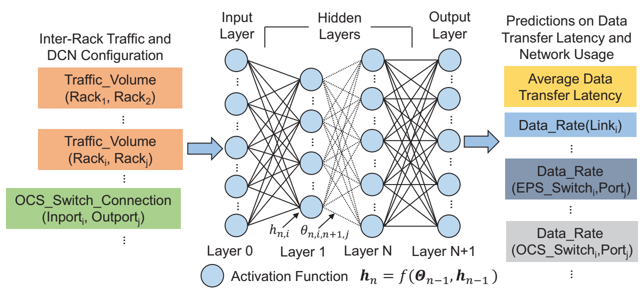
We follow the principle of predictive analytics in human brain to design the KD-NO [6], which makes three AI modules based on deep learning (DL) work collaboratively to achieve predictive analytics and decision making. Specifically, the DL-based Traffic Prediction Module and the DL-based VM Demand Prediction Module first predict future IT/bandwidth resource demands based on the historical network data stored in the Network and Service Database (*i.e.*, forecasting based on memory), and then the DL-based Network Reconfiguration Module takes the prediction results and determines the optimal HOE-DCN configuration intelligently (*i.e.*, decision making based on knowledge). Finally, the configuration is implemented by the VM Management Module and the Network Management Module, which talk with the IT-C and NET-C, respectively, for orchestrating the IT/bandwidth resources. Note that, the three DL-based AI modules work in a relatively independent way in our current design, but consolidating their functionalities may bring in more benefits. For example, higher prediction accuracy could be achieved if the traffic and VM demands are analyzed and predicted jointly. We will address this issue in our future work.



(a) DL-based Traffic Prediction Module



(b) DL-based VM Demand Prediction Module



(c) DL-based Network Reconfiguration Module

Fig. 2. Designs of AI modules based on deep neural networks (DNNs) to achieve knowledge-defined network orchestration.

IV. AI MODULES BASED ON DEEP LEARNING

We design the three AI modules in the KD-NO in Fig. 1(b) based on deep neural networks (DNNs) [15], and their structures are illustrated in Fig. 2. The three AI modules take the similar DNN structure, which contains an interconnected network of simple processing units (*i.e.*, neutrons) to explore the high-order correlation properties of input data. Each neutron contains a non-linear activation function that can take its current state $\mathbf{\Theta_{n-1}}$ and the outputs from other neutrons $\mathbf{h_{n-1}}$ as inputs to generate an output $\mathbf{h_n}$, for modeling highly nonlinear systems precisely. Each DNN consists of an input layer, multiple hidden layers, and an output layer. Here, the neutrons in each layer process the outputs from the previous layer with their activation functions, and propagate their outputs to the next layer through a set of links that can multiply weights on. The procedure is done when the processed data reaches the output layer, where the final outputs of the DNN are obtained. In general, by increasing the number of hidden layers, we can enhance the analyzing and predicting capability of a DNN at the cost of longer training time.

For the Traffic Prediction Module, the DNN in Fig. 2(a) uses the historical statistics of traffic between a source-destination pair as the inputs to forecast its future traffic volumes. Note that, with the right inputs, the DNN can predict the traffic between either a VM pair or two racks. To train the DNN, we organize the collected traffic statistics as a time series, input it to the DNN for outputs, and update the DNN's parameters iteratively by performing gradient descent (GD) until the prediction loss[1] becomes satisfactory. The VM Demand Prediction Module in Fig. 2(b) works similarly, *i.e.*, its DNN is trained for predicting future IT resource usage on a VM (*i.e.*, usages on CPU, memory and storage) based on historical data.

When the modules in Figs. 2(a) and 2(b) have been trained, they work together to provide the DL-based Network Reconfiguration Module in Fig. 2(c) with the knowledge to make intelligent network orchestration decisions. Specifically, given the future IT/bandwidth resource demands, the DNN in Fig. 2(c) is trained to optimize the average data-transfer latency and the utilizations of EPS and OCS ports by selecting the best configuration for the HOE-DCN. Here, to create a training sample, we randomly generate certain IT/bandwidth resource demands and an HOE-DCN configuration, implement them in our HOE-DCN testbed, and collect the average data-transfer latency and link utilizations. Hence, the DL-based Network Reconfiguration Module is trained with practical data collected from a real network system, for making wise decisions.

V. EXPERIMENTAL DEMONSTRATIONS

A. System Implementation

We implement our proposal in a network testbed as shown in Fig. 3, and demonstrate its advantages experimentally. The data plane of our HOE-DCN prototype consists of servers organized in three racks. The servers in each rack are connected to a ToR switch through 1GbE ports, and the EPS inter-rack network is built with four aggregation/core switches organized in a hierarchical structure. Here, the ToR, aggregation and core switches are all packet-based OpenFlow switches, which are either hardware-based commercial ones or software-based OpenvSwitch running on Linux servers. Meanwhile, the ToR switches are also interconnected through a reconfigurable optical switch for the OCS inter-rack network. We distinguish the capacities in EPS and OCS inter-rack networks by letting the ToR switches connect to them with different types of ports. Specifically, the ToR switches connect to their aggregation switches through 1GbE ports, while they can also communicate with each other through 10GbE optical ports whose connectivity is configured by the optical switch.

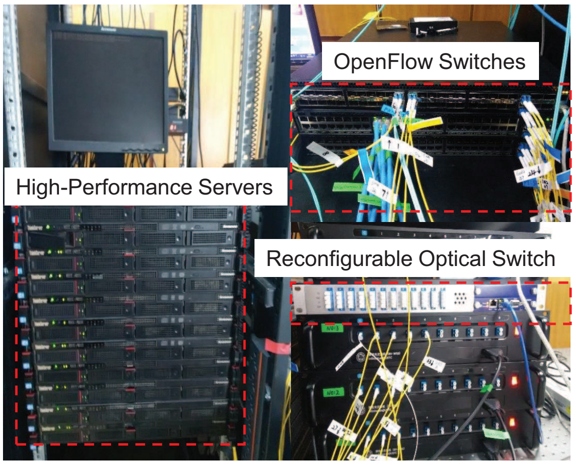


Fig. 3. Snapshot of our HOE-DCN prototype.

The control plane of the HOE-DCN is developed based on open-source softwares and implemented in commodity Linux servers. For the KD-NO in Fig. 1(b), we modify OpenStack to realize the IT-C, and the NET-C is implemented based on ONOS. Specifically, we develop the Flow Provisioning Module to support differentiated services and traffic consolidation in the EPS inter-rack network, program the Network Reconfiguration Module to configure the reconfigurable optical switch automatically, and realize the Network Abstract Module to collect network status in the HOE-DCN. These three modules are implemented by leveraging the internal APIs provided by ONOS. The three DL-based AI modules in Fig. 2 are realized based on TensorFlow. We have to admit that there is a gap between our HOE-DCN prototype and a practical HOE-DCN, but with all the required software/hardware components, it is good enough for the proof-of-concept demonstrations to show the advantages of our proposed knowledge-defined NO-M.

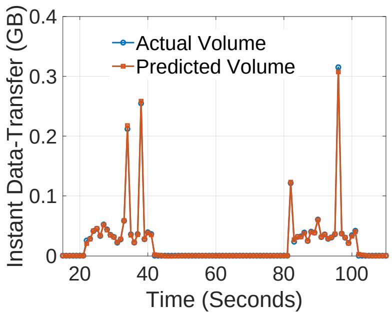
In the experiments, we deploy 9 VMs in the three racks randomly, generate traffic among the VMs with the DCT2GEN tool, and implement the corresponding data transfers with iPerf. To emulate traffic congestion in the testbed, we apply an upper limit on the data-rate of each switch's ports. Specifically, on each ToR switch, the data-rate of a 1GbE port that connects to an aggregated EPS switch is limited below 300 Mbps, while each optical 10GbE port connecting to the OCS optical switch has a peak data-rate of 700 Mbps.

[1]Here, the prediction loss is defined as the mean square error of the predicted traffic volumes to the actual ones.

B. Experimental Results

In the experimental demonstrations, the HOE-DCN testbed leverages the KD-NO to realize AI-assisted VM migration, VM scaling, and HOE-DCN reconfiguration, where all the control and management operations on the HOE-ECN are automatic to verify the intelligence of KD-NO. Note that, to adapt to the dynamics in the HOE-DCN, we train the DL-based AI modules with the online scheme, *i.e.*, the training data sets are updated consistently based on real collected data. Specifically, the modules are first trained in their initializations and then get trained again when their training data sets have been updated with new data. The first training always takes the longest time but can be considered as a part of the offline initialization, while each subsequent training converges much faster since only a portion of the training data set gets updated. Meanwhile, we optimize the DNNs in the AI modules to reduce the time complexity of training.

1) AI-Assisted VM Migration: Fig. 4 shows the experimental results of AI-assisted VM migration. Specifically, the KD-NO analyzes the traffic prediction from the DL-based Traffic Prediction Module and invokes automatic VM migration to consolidate the traffic among VMs for energy saving. In the worst case (*i.e.*, the first training in the initialization), the prediction loss of the DL-based module can be brought down to 0.0196, which corresponds to a deviation of ≤ 20 MBytes on data-transfer volume, after 200 rounds of training. The training takes 148 seconds. Fig. 4(a) shows the predicted and actual volumes of instant data-transfer within a second between two racks[2]. We observe that over 100 seconds, the curves of the predicted and actual traffic basically overlap with each other, and there is almost no visible difference between them. Therefore, the results in Fig. 4(a) verify that the DL-based Traffic Prediction Module can forecast the volumes of both intra-rack and inter-rack data-transfers accurately.



(a) Predicted and actual traffic volumes

	Rack 1	Rack 2	Rack 3
Rack 1	0	1.8489	0.0060
Rack 2	0.0044	0.0218	0.8262
Rack 3	0.0042	0.9130	0.0506

(1) without VM Migration (GB)

	Rack 1	Rack 2	Rack 3
Rack 1	0	0.7935	1.0614
Rack 2	0.0042	0.0002	0.0090
Rack 3	0.0044	0.9130	1.7802

(2) with VM Migration (GB)

(b) Data-transfer volumes for with and without VM migration

Fig. 4. Experimental results of AI-assisted VM migration.

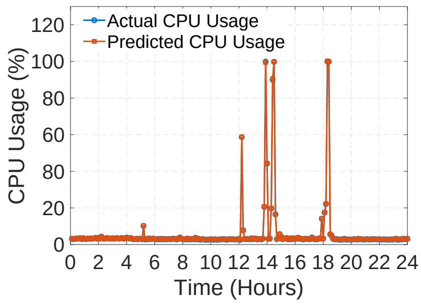
Based on the accurate traffic prediction, the KD-NO organizes the VMs in the optimal scenario for energy saving with automatic VM migration. Fig. 4(b) compares the data-transfer volumes among the racks for with and without the AI-assisted VM migration, which indicates that through VM migration, the proposed KD-NO converts certain traffic from inter-rack to intra-rack and consolidates the inter-rack traffic more. This improves not only the service provisioning performance but also the energy efficiency of the HOE-DCN.

2) AI-Assisted VM Scaling: Then, we try to verify the performance of the DL-based VM Demand Prediction Module. Specifically, the experiments consider the scenario in which the KD-NO analyzes VM demand prediction from the DL-based module and invokes automatic VM scaling to improve a VM's performance on service provisioning. In the worst case, the prediction loss of the DL-based VM Demand Prediction Module goes below 1.2596, which corresponds to a deviation of $\leq 1.5\%$, after 8000 rounds of training. The training takes 15.5 hours. Fig. 5(a) compares the predicted and actual instant CPU usages of a VM, which confirms that the change of a VM's CPU usage over time can be predicted precisely too.

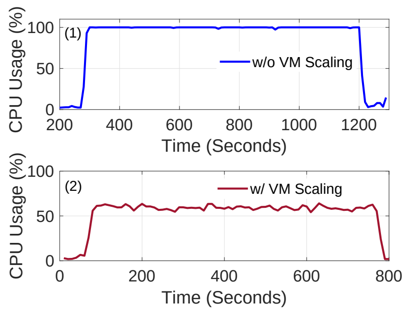
With the assistance from the VM Demand Prediction Module, the KD-NO can invoke VM scaling automatically and timely to ensure the performance of a VM. Fig. 5(b) shows the CPU usages for with and without the AI-assisted VM scaling. Without the VM scaling, the VM's CPU usage can be as high as 100% for $\sim$15 minutes to accomplish a computing task, while the VM scaling reduces the VM's CPU usage to 60% and makes it finish the task within $\sim$$12.5$ minutes.

3) AI-Assisted HOE-DCN Reconfiguration: Finally, we try to verify the performance of the DL-based Network Reconfiguration Module. In the worst case, the module needs to be trained for 4×10^4 rounds, which takes 955 seconds, for achieving an accuracy of 98.82% on the decision making to reconfigure the HOE-DCN based on the inputs from the DL-based Traffic and VM Demand Prediction Modules. Then, the experiments use the DL-based modules for automatic HOE-DCN reconfiguration, whose advantages can be confirmed with the results in Figs. 6(a) and 6(b). The results in Fig. 6(a) suggest that our proposed KD-NO significantly reduces the average data-transfer latency of the flows in the HOE-DCN, which improves the performance of service provisioning. More importantly, Fig. 6(b) indicates that the KD-NO actually uses less switch ports (*i.e.*, less energy consumption) to achieve the reduced data-transfer latency. Hence, unlike the traditional network orchestration systems that can only trade data-transfer latency for energy efficiency, our KD-NO can leverage intelligent decision making to optimize both of them simultaneously. This is a really promising observation and suggests that our KD-NO opens up new opportunities to realize green DCNs.

[2]Note that, in the KD-NO, we assign a DL-based Traffic Prediction Module to each rack-pair (*e.g.*, *Rack* 1-*Rack* 1 or *Rack* 1-*Rack* 2), and the results in Fig. 4(a) are the worst ones from all these modules, for fair comparisons.

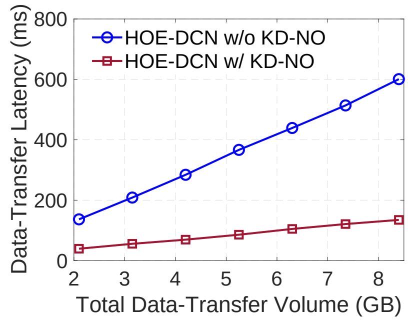


(a) Predicted and actual CPU usages of a VM

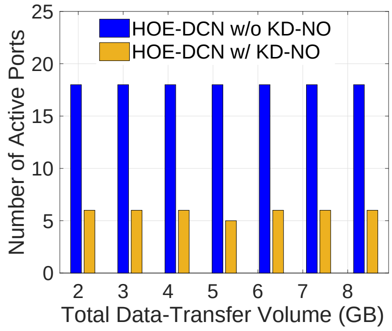


(b) CPU usages for with and without VM scaling

Fig. 5. Experimental results of AI-assisted VM scaling.



(a) Average data-transfer latency



(b) Number of active switch ports

Fig. 6. Experimental results of AI-assisted HOE-DCN reconfiguration.

VI. Discussion and Conclusions

We discussed our efforts on realizing an HOE-DCN system with knowledge-defined NO-M. Specifically, we designed three DL-based AI modules to assist the knowledge-defined NO-M, and implemented them in a real network testbed for prototyping our design. The experimental results demonstrated that our HOE-DCN prototype can achieve predictive analytics and intelligent decision making and thus realize automatical management, to improve not only the performance of service provisioning but also the system's energy-efficiency.

Besides the advantages, we hope to point out that the proposed HOE-DCN system can still be improved from a few perspectives. For instance, the current knowledge-defined NO-M only predicts traffic and VM demands, which is still low-level knowledge for DCN operation and thus not good enough. Our future work will try to incorporate more intelligence in it and make it application-drive, *i.e.*, enabling it to predict the actual applications/services that will be run in the HOE-DCN. Hence, high-level knowledge can be obtained to make the network system more efficient, in terms of both system architecture and operation procedure. This is because for making intelligent decisions, high-level knowledge would be much more useful than low-level knowledge. Take human brains as an example, it is obvious that to recognize a people, the high-level knowledge about objects and faces would be much more useful than the low-level knowledge about sharps and colors. Moreover, the application-driven scheme would be helpful to address the issues on resiliency and security.

Acknowledgments

This work was supported in part by the NSFC Projects 61701472 and 61771445, CAS Key Project (QYZDY-SSW-JSC003), NGBWMCN Key Project (2017ZX03001019-004), China Postdoctoral Science Foundation (2016M602031), and Fundamental Research Funds for the Central Universities (WK2100060021).

References

[1] "Cisco global cloud index: Forecast and methodology, 2016-2021," Cisco, (Accessed on Feb. 1, 2018) [Online], Available: https://www.cisco.com/c/en/us/solutions/service-provider/visual-networking-index-vni/index.html.

[2] P. Lu *et al.*, "Highly-efficient data migration and backup for big data applications in elastic optical inter-datacenter networks," *IEEE Netw.*, vol. 29, no. 5, pp. 36–42, Sept./Oct. 2015.

[3] W. Fang *et al.*, "Joint defragmentation of optical spectrum and IT resources in elastic optical datacenter interconnections," *J. Opt. Commun. Netw.*, vol. 7, no. 4, pp. 314–324, Mar. 2015.

[4] N. Farrington *et al.*, "Helios: a hybrid electrical/optical switch architecture for modular data centers," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 40, no. 4, pp. 339–350, Oct. 2010.

[5] K. Chen *et al.*, "OSA: an optical switching architecture for data center networks with unprecedented flexibility," *IEEE/ACM Trans. Netw.*, vol. 22, no. 2, pp. 498–511, Apr. 2014.

[6] W. Lu *et al.*, "Leveraging predictive analytics to achieve knowledge-defined orchestration in a hybrid optical/electrical DC network: Collaborative forecasting and decision making," in *Proc. of OFC 2018*, pp. 1–3, Mar. 2018.

[7] N. Feamster, J. Rexford, and E. Zegura, "The road to SDN: an intellectual history of programmable networks," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 44, no. 2, pp. 87–98, Apr. 2014.

[8] Z. Zhu *et al.*, "Demonstration of cooperative resource allocation in an OpenFlow-controlled multidomain and multinational SD-EON testbed," *J. Lightw. Technol.*, vol. 33, no. 8, pp. 1508–1514, Apr. 2015.

[9] A. Mestres *et al.*, "Knowledge-defined networking," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 47, no. 3, pp. 2–10, Jul. 2017.

[10] Z. Zhu *et al.*, "RF photonics signal processing in subcarrier multiplexed optical-label switching communication systems," *J. Lightw. Technol.*, vol. 21, no. 12, pp. 3155–3166, Dec. 2003.

[11] "United States data center energy usage report," Lawrence Berkeley National Laboratory, (Accessed on Feb. 1, 2018) [Online], Available: https://escholarship.org/uc/item/84p772fc.

[12] A. Greenberg, J. Hamilton, D. Maltz, and P. Patel, "The cost of a cloud: research problems in data center networks," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 39, no. 1, pp. 68–73, Jan. 2009.

[13] P. Lu, Q. Sun, K. Wu, and Z. Zhu, "Distributed online hybrid cloud management for profit-driven multimedia cloud computing," *IEEE Trans. Multimedia*, vol. 17, no. 8, pp. 1297–1308, Aug. 2015.

[14] J. Wu, S. Guo, J. Li, and D. Zeng, "Big data meet green challenges: big data toward green applications," *IEEE Syst. J.*, vol. 10, no. 3, pp. 888–900, Sept. 2016.

[15] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.