International Journal of Research
Available at https://edupediapublications.org/journal
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 03 Issue 17
November 2016

Servay of big data network and SDN technology

Ahmed jaber Al-mansoori Shui Yu
email:(ajalma,shui.yu)@deakin.edu.au
school of information technology
Deakin University
Burwood, Australia

Abstract

This paper studies most recent developments in this dynamic research region of SDN. We first present a by and large acknowledged definition for SDN with the previously mentioned two trademark elements and potential advantages of SDN. We then harp on its three-layer engineering, including a framework layer, a control layer, and an application layer, and substantiate every layer with existing research endeavors and its related research regions. The study takes after that with an outline of the de facto SDN implementation (i.e., OpenFlow). At long last, we conclude this overview paper with some recommended open research challenges. The Traditional network makes it a troublesome undertaking to include and evacuate the network devices. The Static Nature of the routine network makes it hard to fulfill the dynamic registering and capacity desires of enormous server farms and grounds. Software Defined Networking is the new way to deal with networking. Software Defined Network model guarantees to improve the network design and asset administration. Software Defined Networking will supplant the present network and satisfy the business needs by means of Software instead of equipment. This paper presents the ideas of Software Defined Network which will arrange and deal with the network needs.
Keywords: Big data,Dynamic computing,Hardware, Networking,Software-Defined Networking (SDN).

Introduction

Big data become one of the hottest research topics in the last decade. The terms of big data refer to the tremendous amount of data that could be generated by sensors, social media stock exchange, etc., which cannot be handled and processed by a traditional database [1]. Big data process can be done by batch data processing which can be a good way to handle a significant amount of data by collecting and processing an input of data resulting output as a batch; Hadoop is one good example of Batch Processing. However, due to the accelerated development of the technology, the number of applications, smartphones, sensors, etc. that generate continues data streaming are rapidly increasing. These applications and sensors could be used in many fields like analysing of social media, surveillance and medical services, video annotation and financial analysis. These data cannot be stored or delayed for a long time, hence, unlike big data process, in Big Data Stream Processing (BDSP) data should be processed in real time or nearly real time, and the extracted information is importantly needed.

Mostly, the streamed data are crucial and needed to be evaluated in approximate real-time to obtain a valuable information before the data become valueless. For instance, the raw streamed date comes from health monitoring sensors should be processed at once and an action should take a place. Delay in the process may lead to serious issues. Many challenges facing big data stream processing in many terms. Many frameworks have been proposed to overcome this obstacle such as Yahoo S4!, Twitter Storm, Samza and so on. However, these platforms are still in developing stage, hence, ensuring QoS and performances are not guaranteed with such systems. Many restrictions were addressed in [9] regarding above mentioned platforms. Writing complex monitoring pipeline, requiring explicit deployment and separating the computation in a cloud-hosted cluster are such restrictions. Moreover, would those methods be proper to process the increment of data? CISCO claim that the data generation will exceed a Zettabyte of data in 2016. They also argue that the generated amount of data will exceed 2.1 Zettabytes by 2020, mostly (66%) from wireless devices and sensors hardware [2].

If we are questioning the present technology whether it would be suitable to deal with such increasing in data, the answer will be defiant no. With the before-mentioned volume of data, the process of designing or developing an application deals with BDSP will encounter many challenges like minimizing the overall computing and communication energy while ensuring the quality of services. The existent and new application that focus on BDSP should rely on robust network infrastructure to perform a better solution. Software Define Network (SDN) has become one of the biggest solutions to achieve better system supervision. Open Network Foundation (ONF) has defined SDN as a new network technique wherein the network control and data forwarding function are separated, and the network is controlled by programming wise. The effect is a notably dynamic, flexibility, cost-effective, and adaptable architecture that gives administrators unprecedented programmability, automation, and control. By using the features of SDN on a network, we can have adequate infrastructure for the application that deals with BDSP. Decoupling the control and data planes, ability to program the network and logically centralized the network are some features of SDN which can advance big data processing [3].

I. BIG DATA

A. Big data features

Across the last couple decades, data has grown on a massive scale in various fields. The term of big data is usually used to express gigantic datasets. Compared with traditional datasets, big data typically includes masses of unstructured data that need more real-time analysis. Recently, many companies and government are paying more attention and effort to big data. Nevertheless, people still uncertain about the definition of big data and how can we differentiate it with normal data. The popular open-source software framework Apache Hadoop had defined the term of big data as a dataset that can't be stored, managed and processed by general purpose computers. By the definition, we can sense that the Volume of data is an important characteristic of big data. Volume refers to the mass of data dataset that generates from different sources of data and processed by big data application. Moreover, the ripped increasing in the size of data and the need for collecting and processing data in the real-time is another issue; this is called as Velocity of data which is another feature of big data. The term Velocity in big data refers to the pace of generating data. The more data generated, the more challenges brings to the big data. Another significant feature of big data is the Variety which refers to the different types of data comes from various sources. With the developing in the of information technology many technologies had seen the light, Let's take into consideration the variety of data produced by mobile sensors, RIFD Tags GPS sensors, and cyber physics devices. What's more, the value of data, even though the value of the data is a major factor in both big and not big data. This feature relates to the complexity of applying the big data application for getting the results [28].Veracity which refers to the degree in which a leader trusts information to make a decision [4]. This term describes the "quality" of captured data. The quality here referred assurance and confidence as the generated data could vary widely on these metrics.

B. Challenges of big data

Given the guarantees of upgraded setup, enhanced execution, and empowered advancement, SDN is still in its outset. Numerous principal issues still remain not completely understood, among which institutionalization and reception are the most earnest ones. In spite of the fact that the ONF definition of SDN is most gotten one, OpenFlow supported by ONF is in no way, shape or form the main SDN standard and in no way, shape or form a develop arrangement. An open-source OpenFlow driver is still truant for SDN controller development, a standard north-bound API or an abnormal state Software dialect is as yet missing for SDN application development. A sound biological system consolidating network device sellers, SDN application developers, and network device shoppers, has yet to show up. SDN offers a stage for inventive networking methods, however the move from customary networking to SDN can be troublesome and excruciating. Regular concerns include SDN interoperability with legacy network devices, execution and security worries of unified control, and absence of specialists for specialized support. Existing deployments of SDN are frequently constrained to little testbed for research models. Models for research reason premature to offer confidence without a real world deployment.

C. Big data tools

With the ripped development in the computer network, companies can handle immense volumes of data using standard and cheap computers rather than using supercomputers. Researchers and industry field have proposed many techniques for distributed computing. To store, manage, access and analyse a big amount of data on a real time many companies like Google, Yahoo, Twitter and Facebook were motivated to come up with new methods to deals with the vast amount of the data they produce or deals with. As a good example of these methods MapReduce, Hadoop, and Big Table. These technologies lead to a new generation of data management and allow businesses to address one of the most fundamental problems, namely the capability to process massive amounts of data efficiently, cost-effectively, and quickly. Below a short discussion of these methods:

1. MAPREDUCE

MapReduce was originally designed and produced by Google. The main idea was to deploy an assortment of functions in batch mode wich handle vast amount of data. Map technique is a task distributor. It usual distributes the tasks over a large number of systems while maintaining placement to check the load and provide recovery from failures. After Mapping technique takes the place, Reducing function start to tack a place in the raw. Reduce function aggregates all the sub-distributed tasks back together to provide a result [5].

2. BIG TABLE

Also produced by Google, the main idea of Big Table is to manage highly scalable data by distributed the storage system, in which data is combined into tables with rows and columns. To differentiate Big Table and Typical relation database model, big data has been designed to keep a vast amount of data crossed commodity servers. Opposite to traditional database, Big Table is a sparse, distributed, persistent multidimensional sorted map

3. Hadoop

By combining MapReduce and Big Table, Apache Hadoop was produced by Google also. Hadoop aims to parallelize data processing over computing nodes in the attempt of speeding computations and reduce latency. The main significant features of Hadoop is an extensive scalable distributed file system that can support petabytes of data and a massively scalable MapReduce engine that computes results in batches. Hadoop also inspired Yahoo Business and became the base of its computer design.

D. Big data processing platforms

Data processing can classify into three major kinds which are Interactive, Batch and Stream processing. Interactive processing is a significant method to obtain current patterns in various large datasets. As a good example Apache HBase and Google BigQuery. Batch processing process the collected offline data, where the data are not crucial and can wait and store, Hadoop is a good example of batch processing. As explained earlier, Hadoop uses the big table and MapReduce tools on top of Hadoop Distributed File System (HDFS). Stream processing is usually used to analyze streamed that can't wait or stored. Therefore, data are processed in real time to gain an immediate insight. Because of the spotlight on the stream processing due to its significant influence on the economy and daily process, many implementations had seen the light, we will discuss three of the biggest hit of the biggest companies:

Yahoo! S4 (Simple Scalable Streaming System): is an open source, general-purpose, platform; which is written in JAVA. Yahoo! S4 enable the users to develop applications that deals with an infinite stream of data. Yahoo! S4 is featuring: Decentralized, Scalable, Extensible, Cluster management and Fault-tolerance [6]. Another implementation is Twitter Storm. The Twitter storm is an open source, distributed computation framework mainly written in Clojure and Java and can be used by any programming language. Storm deals with an infinite stream of data in real time processing. The main features of Storm were also shown as Topologies, Streams, Data model and Stream groupings. Samza is produced as a distributed stream processing structure, which makes use of Apache Kafka and Apache Hadoop YARN technologies. Samza is mainly written in Scala and JAVA languages. It provides a fault tolerance, processor isolation, security, and resource management [7]. Samza Provide near real-time, an asynchronous computational framework for stream processing. The main features of Samza are durability, scalability, pluggable and process isolation. Below table is the most used platforms so far:

Platform name	support	Base structure	Prepose	Year
Hive 31	SQL	Hadoop		2008
Impala	Paralle processing SQL	Hadoop		2012
HAWQ	parallel processing SQL	Postgres database and the HDFS storage		2013
Big SQL	SQL	Hadoop		
HadoopDB project[1]	SQL	MapReduce and parallel databases.		
Presto	SQL	HDFS		
Polybase	SQL			
Dremel project	SQL	Hadoop		
	Graph Streem			
Apache Pig	SQL	Java program		2008
Jaql	CSV, TSV, XML.	JSON		2010
Gluster File System		Apache Hadoop		
Hortonworks Data Platform	RDBMS			2011
Lumify		Bigdata		2014
Zookeeper	SQL	Hadoop		2016
Cloudera	computer servers	Hadoop		
Sqoop	SQL	Hadoop		2015
Cassandra	SQL	Hadoop		2008

Table 1: list of platforms

E. Big data and clould

Although there still a questioning on what hardware and software should use with big data, cloud computing presented as a robust solution for big data. The flexible characteristics of the cloud computing present an efficient solution for many computer sciences queries. Cloud computing is a highly successful oriented service which has changed the way of practicing the computer infrastructure via using its key features which are (PaaS) Platform as a Service, (IaaS) Infrastructure as a Service, and (SaaS) Software as a Service. Moreover, with the developing of Big Data and its use, Cloud computing had adopted new trends to support Big data development which called Database as a Service, which offers a great feature that benefits consumers such us low upfront investment, pay-per-use and transfer of risks[8]. Despite the challenges of a technical issue in the cloud, the cloud is still a suitable platform for most scenarios. Many studies had taken a place to show that cloud is the best available solution for big data, Authors in focused on systems for supporting "heavy" applications, ad-hoc analytics and decision support. In researchers conducted in the area of big data processing in cloud environments advanced dynamic load balancing methods and data stream distribution for efficient data processing. While the work on points to the challenges and requirements for cloud computing systems regarding big data.

II. BIG DATA NETWORK

A. Software Define Network

1. SDN Overview of SDN

International Journal of Research
Available at https://edupediapublications.org/journal
p-ISSN: 2348-6848
e-ISSN: 2348-795X
Volume 03 Issue 17
November 2016

Developing new trends in Information and Communication Technologies domain, in specifically, big data, and cloud computing is demanding a high bandwidth of a computer network and dynamic management. Moreover, Cloud computing raise the need for flexibility of computer networks. Matter fact remote access of resources and storage operations are increasing rapidly in the cloud computing, subsequently, efficient access to these sources throughout the network has come to be essential to fulfil the needs of latest computing needs. In order of satisfying these requirements, both academic and commercial field had tried to advance computer network infrastructure. Nevertheless, this increase in network infrastructure would produce an increasing in complexity regarding network size, heterogeneity, and management. Alternatively, many ideas were proposed to obtain better control over the network. as an example of these attempts Name Data Network (NDN), Programmable Network (PN), HTTP as the narrow waist, and Software Define Network. Distinctly, SDN considered as a promising solution after receiving addition attention from bother industry and research community in past few years. The fundamental concept of SDN is to separate the control plane from the data plane and providing efficiency and flexibility in the network resources and operations, by using software programs and special tools (switches and routers). By using SDN, new solutions for particular obstacle can be easily addressed in a form of software such as resource management, network virtualization, and network security. Furthermore, by the sense of obtaining an instant network status, SDN provides a real-time centralized administration of a network based on both instantaneous network status and User policies [9]. Feature mentioned above will advantage the network optimization and configurations which will improve network performance. Moreover, the main characteristics of SDN that support the network with can be summed as the following:

1) Enhancing Configuration: SDN provides a proper solution to the problem of network configuration. Due to the heterogeneity in network device companies and configuration interfaces, typical network configuration involves some degree of manual processing. This hand operated configuration considered as a complicated process. By using SDN over a network, it is likely to configure network devices from any point in the network using software controller

2) Improving Performance: In network services, one of the important goals is to maximize the implementation of the network infrastructure. Furher, due to complex technologies that use in the network, optimizing a network performance consider as difficult progress. Current approaches usually focus on optimizing performance of a subset of networks or the quality of user experience for some network services. SDN offers an opportunity to enhance network performance by providing a feedback control information exchanged between different layers of the network and centralized management with a global network view. As such, many challenging performance optimization problems would become manageable with properly designed centralized algorithms. For instance, Data traffic scheduling, end-to-end congestion control, energy efficient operation and Quality of Service (QoS) support.

2. SDN architecture

Basically SDN consist of three layers named infrastructure layer, control layer and application layer discussed below:

Infrastructure layer: Also known as the data plane, similarly to a traditional network, is composed of a set of networking devices such as switches and routers in the data plane. Functions of these switching devices are: first, they are responsible for collecting network status, storing them temporally in local devices and sending them to controllers. Second, they are responsible for processing packets based on rules provided by a controller. Control layer: Also known as the control plane, it bridges the application layer and the infrastructure layer, via its two interfaces, the south-bound interface for downward interacting with the infrastructure layer and the north-bound interface for upward interacting with the application layer [10]. The south-bound interface is specific functions that allow controllers to access functions provided by switching devices. The functions may include reporting network status and importing packet forwarding rules. The north-bound interface provides service access points in various forms, for example, an Application Programming Interface (API). East/Westbound is another interface in the control layer that is not support any standards. The main function of East/Westbound is that enabling communication between groups or federations of controllers to synchronize state for high availability.

Application layer: It mainly consists of the end-user business applications that consume the SDN communications and network services. Through the programmable platform provided by the control layer, SDN applications can access and control switching devices at the infrastructure layer. Example of SDN applications could include dynamic access control, seamless mobility and migration, server load balancing, and network virtualization.

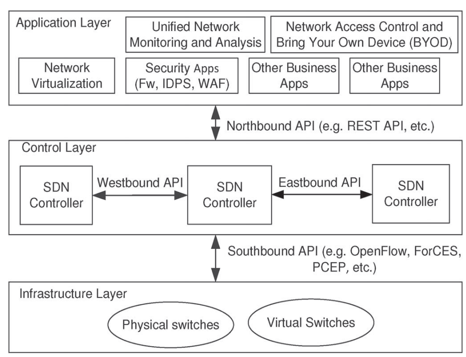

Figure 1: SDN architecture

SDN PROPOSAL:

SDN initially utilized in server farms. It has helped organizations change the network structure and necessities according to needs. Brought together Control of open stream design permits the network manager to program the network conduct. When we attempt to execute SDN network we require centering challenges of SDN [11].

Controller Scalability:

SDN controller is the cerebrum of a network. SDN network implementation requires defining various controllers needed for network and their area. The Network may require a solitary controller or chain of command of controller in control arrange. "at the point when the network scales up in the quantity of switches and the quantity of end has, the SDN controller can turn into a key bottleneck." The idleness increments amid transmission of data between numerous nodes and the single controller. The quantity of switches streams and transmission capacity increaser's number of solicitations will be pending to the controller, which may not ready to prepare. The SDN controller [NOX] examines said that it can deal with 30 k demands. For SDN network, a principle test is various controller need and their localization.

Convergence and Management :

SDN OpeFlow design was initially developed for big business grounds networks that help researchers to analyze their conventions. In other way, SDN designed for the little network like the private network. Be that as it may, to attempt and extend this design to substantial networks requires go to a few issues, for instance, the issue of Interdomain (Routing between two networks) [12].

Security :

Security is a principle worry in networking to detect and counteract peculiarities. Networks for the most part comprise of host based and network based security instrument which networks to detect interruption be a piece of their network or outside network. Current security arrangements are hard to oversee, costly, intricate, resolute. Programmable SDN requires keen security models in light of the fact that SDN frameworks handle by the network chairman who is designed the network according to necessities through Software. Security should be developed in design to ensure the controller safely.

Controller Flexibility:

OpenFlow proposed brought together controller is revered to the network. SDN permit network executive viably program a network with Software running on a focal controller. A breakdown of the controller can contrarily trade off an adaptability of the entire network. SDN network needs to concentrate on to define the best approach to handle the controller disappointment. As of now developing SDN strategy concentrate on independent the control arrange from the information arrange and provide programmable interfaces to satisfy the business needs [13].

Cost Efficiency:

Software defined networking offers awesome chances to expand effectiveness while in the meantime diminishing expenses and unpredictability. Today's distributed computing demands are detonating, and therefore less vitality utilization and high security networking is needed. Contrasted with customary networks, Lombardo et al. identify high potential for element allotment of network capacities over network nodes, however the way toward testing, testing and propelling is still too tedious and is not good with business needs. Casado et al. say now that present networks are excessively costly and excessively convoluted, making it impossible, making it impossible to oversee [14].

Network Security

Network security is a remarkable piece of cyber security and is gaining considerations. Traditional Network security rehearses deploy firewalls and proxy servers to ensure a physical system. Because of the heterogeneity in system applications, ensuring select gets to by true blue system applications involves implementation of a system wide strategy and repetitive arrangement of firewalls, proxy servers, and different devices. In this viewpoint, SDN offers a helpful stage to unify, consolidation and check strategies and setups to ensure that the implementation meets required assurance in this manner preventing security ruptures proactively. Also, SDN provides better approaches to detect and defend assaults responsively. Capacity to gather organize status of DN permits examination of activity examples for potential security dangers. Assaults, for example, low-rate burst assaults and Distributed Denialof-Service (DDoS) assaults, can be detected just by analyzing movement design. In the meantime, SDN provides automatic control over activity streams. Thus, activity of interest can be unequivocally coordinated to Intrusion Prevention Systems (IPSs) for Deep Packet Inspection (DPI). On the off chance that assaults are detected, SDN can install bundle forwarding standards to switching devices to obstruct the assault movement from entering and propagating in a system [15]. Incorporated control of SDN allows progressively quarantine of traded off hosts and authentication of genuine hosts in light of information obtained through requesting end has, requesting a Remote Authentication Dial In User Service (RADIUS) server for users' authentication information, tainting activity or framework scanning during enrollment. Finally, SDN is further fit for providing immediate and fine-grained control over systems, and offers chances to actualize novel security insurance methodologies. For instance, Jafarian et al. develop Moving Target Defense (MTD) in view of effective control of SDN [16]. A virtual IP is connected with every host for information transmission and is arbitrarily changed with high capriciousness and rate while a genuine IP of the host is static. Controllers will indicate interpretation between the virtual IP and genuine IP while maintaining setup integrity. As another illustration, Mendonca et al. introduce AnonyFlow, an in system endpoint anonymization benefit designed to provide security to users. AnonyFlow performs interpretation between AnonID, Network IP and Machine IP using an implementation of SDN.

Software Defined Networking: Effect – Features

Software defined networking comprises of a few new elements contrasted with conventional Networks. The most widely recognized and frequently specified element in the writing is the likelihood of decoupling the forwarding plane from the information plane, leading to a few deliberation layers [17]. The study has progressive when comparing tight customary system models. This situation brings about a programmability of the information plane, a customization of systems and incorporated control decisions because of a worldwide view over the entire system. The study describing the OpenFlow convention as an increase of system perceivability. Kirkpatrick likewise perceives the given API as a component, delivered by programming defined networking, to handle applications (like email or phone applications) effortlessly over the entire network45. Dynamic, demand-based system division and use are moreover defined as pivotal key elements of programming defined networking. An extra advancement of the header information by using dynamic streams as opposed to static routing altogether diminishes the overhead on per byte exchange. Dely et al. give an account of a SDN-based design for optimizing handover components in remote LANs48. Vissicchio et al. condense SDN by defining such elements as another engineering, which provides the likelihood of controlling the entire system structure with the advantages of an innovative and enhanced administration [18].

CONCLUSION AND FUTURE WORK

SDN has developed as a way to enhance programmability within the system to support the dynamic nature of future system works by separating network control and forwarding capacities. SDN has been utilized in server farms furthermore helped organizations to alter arrange setup and structure. SDN guarantees to change over today's conventional static system into adaptable, versatile, programmable based system with intelligence which progressively design arrange assets. Programming defined networking is viewed as a transformative outlook change, yet at the same time confronts a few challenges. The secured territories of challenges and impacts provide a general perspective of what may back off further development and what is conceivable when the innovation is integrated effectively. The study witness broad action around SDN sooner rather than later. Emerging themes requiring further research are, for instance: the movement way to SDN, extending SDN towards transporter transport systems, acknowledgment of the system as-an administration cloud computing worldview, or programming defined situations (SDE). Accordingly, we might want to get criticism from the networking/SDN people group as this novel worldview advances, to make this a "live archive" that gets redesigned and enhanced in view of the group input. We have set up a github for this reason, and we invite our readers to join us in this public exertion. Certain absences of know-how, combined with high intricacy with regards to integration into conventional systems, are main purposes behind a delayed dissemination of the innovation. Moreover, the broke down papers for the most part describe programming defined networking on an exceptionally detailed scientific and innovative premise, making it hard for ventures and associations to survey if the innovation can have a particular business affect. In any case, the unfaltering increase of users and prerequisites leave providers confronted with the need to rethink the utilization of Network Technologies innovations in order to remain focused and gainful.

REFERENCE:

1. Nunes, B. A. A., Mendonca, M., Nguyen, X. N., Obraczka, K., & Turletti, T. (2014). A survey of software-defined networking: Past, present, and future of programmable networks. *IEEE Communications Surveys & Tutorials*, *16*(3), 1617-1634.
2. Jain, R., & Paul, S. (2013). Network virtualization and software defined networking for cloud computing: a survey. *IEEE Communications Magazine*, *51*(11), 24-31.
3. Chen, M., Jin, H., Wen, Y., & Leung, V. C. (2013). Enabling technologies for future data center networking: a primer. *Ieee Network*, *27*(4), 8-15.
4. Kreutz, D., Ramos, F. M., Verissimo, P. E., Rothenberg, C. E., Azodolmolky, S., & Uhlig, S. (2015). Software-defined networking: A comprehensive survey. *Proceedings of the IEEE*, *103*(1), 14-76.
5. Hu, F., Hao, Q., & Bao, K. (2014). A survey on software-defined network and openflow: from concept to implementation. *IEEE Communications Surveys & Tutorials*, *16*(4), 2181-2206.
6. Xia, W., Wen, Y., Foh, C. H., Niyato, D., & Xie, H. (2015). A survey on software-defined networking. *IEEE Communications Surveys & Tutorials*, *17*(1), 27-51.
7. Yang, M., Li, Y., Jin, D., Zeng, L., Wu, X., & Vasilakos, A. V. (2015). Software-defined and virtualized future mobile and wireless networks: A survey. *Mobile Networks and Applications*, *20*(1), 4-18.
8. Jansen, S., & Cusumano, M. A. (2013). Defining software ecosystems: a survey of software platforms and business network governance. *Software Ecosystems: Analyzing and Managing Business Networks in the Software Industry*, *13*.
9. Jarraya, Y., Madi, T., & Debbabi, M. (2014). A survey and a layered taxonomy of software-defined networking. *IEEE Communications Surveys & Tutorials*, *16*(4), 1955-1980.
10. Gringeri, S., Bitar, N., & Xia, T. J. (2013). Extending software defined network principles to include optical transport. *IEEE Communications Magazine*, *51*(3), 32-40.
11. Channegowda, M., Nejabati, R., & Simeonidou, D. (2013). Software-defined optical networks technology and infrastructure: Enabling software-defined optical network operations [Invited]. *Journal of Optical Communications and Networking*, *5*(10), A274-A282.
12. Sezer, S., Scott-Hayward, S., Chouhan, P. K., Fraser, B., Lake, D., Finnegan, J., ... & Rao, N. (2013). Are we ready for SDN? Implementation challenges for software-defined networks. *IEEE Communications Magazine*, *51*(7), 36-43.
13. Kim, H., & Feamster, N. (2013). Improving network management with software defined networking. *IEEE Communications Magazine*, *51*(2), 114-119.
14. Nunes, B. A. A., Mendonca, M., Nguyen, X. N., Obraczka, K., & Turletti, T. (2014). A survey of software-defined networking: Past, present, and future of programmable networks. *IEEE Communications Surveys & Tutorials*, *16*(3), 1617-1634.

15. Kreutz, D., Ramos, F. M., Verissimo, P. E., Rothenberg, C. E., Azodolmolky, S., & Uhlig, S. (2015). Software-defined networking: A comprehensive survey. *Proceedings of the IEEE*, *103*(1), 14-76.
16. Drutskoy, D., Keller, E., & Rexford, J. (2013). Scalable network virtualization in software-defined networks. *IEEE Internet Computing*, *17*(2), 20-27.
17. Tuttlebee, W. H. (1999). Software-defined radio: facets of a developing technology. *IEEE Personal Communications*, *6*(2), 38-44.
18. McKeown, N. (2009). Software-defined networking. *INFOCOM keynote talk*, *17*(2), 30-32.