Software-Defined Wide Area Network (SD-WAN): Architecture, Advances and Opportunities

Zhenjie Yang[1,2], Yong Cui[1*], Baochun Li[2], Yadong Liu[1] and Yi Xu[1]
[1] Department of Computer Science and Technology, Tsinghua University, Beijing, China
[2] Department of Electrical and Computer Engineering, University of Toronto, Toronto, Ontario, Canada
yangzj15@mails.tsinghua.edu.cn, cuiyong@tsinghua.edu.cn, bli@ece.toronto.edu, {liuyd17, yi-xu18}@mails.tsinghua.edu.cn

***Abstract*—Emerging applications and operational scenarios raise strict requirements for long-distance data transmission, driving network operators to design wide area networks from a new perspective. Software-defined wide area network, i.e., SD-WAN, has been regarded as the promising architecture of next-generation wide area network. To demystify software-defined wide area network, we revisit the status and challenges of legacy wide area network. We briefly introduce the architecture of software-defined wide area network. In the order from bottom to top, we survey the representative advances in each layer of software-defined wide area network. As SD-WAN based multi-objective networking has been widely discussed to provide high-quality and complicated services, we explore the opportunities and challenges brought by new techniques and network protocols.**

I. Introduction

The Internet has been a remarkable success over the past few decades. As one of the most important transmission mediums on the Internet, wide area network, such as inter-datacenter networks, enterprise networks, and carrier networks, has become the critical infrastructures of the information society [1]. Nowadays, the quick expanding of networks and the spring up of new applications and operational scenarios raise exacting requirements on wide area networks. For example, live video service providers require the latency from broadcasters to viewers to be less than 400 ms; Internet service providers hope to launch new businesses in their networks within several days. As most wide area networks were designed originally to work with the best-effort mentality, they do not provide any guarantees on service quality [2]. In addition, there are various brands of devices in carrier-grade networks, and each device is typically configured in a low-level vendor-specific manner, launching a new business on it usually requires several weeks to months and a lot of manpower [3]. Since the expenditure of building, managing, and debugging wide area networks is extremely high and traditional wide area networks have shown disadvantages on many aspects such as guaranteeing service quality and upgrading network easily, constructing wide area networks with new designs is quite necessary.

Software-defined wide area network is regarded as the promising architecture of next-generation wide area network, which offers network operators a new perspective to build network. Software-defined wide area network is proposed to apply software-defined techniques in networking connections covering a wide geographical area, and it achieves the purpose of software control using the philosophy that is different from software-defined network (SDN). Software-defined wide area network simplifies the connection building and managing between different sites, e.g., data centers in inter-datacenter networks and branch offices in enterprise networks, and provides the necessary flexibility, centralized control and monitoring with lower costs. Compared with conventional wide area networks, software-defined wide area network has two superiorities that are suitable for current markets. First, it provides an inherent programmatic framework for hosting control applications that are developed in a centralized way while taking into consideration the application-level requirements to guarantee the user-perceived quality of experience (QoE); Second, it is able to centrally define network policies and manage network traffic without requiring manual configuration at each device [4]. The former advantage enables it to provide service guarantees for specific applications, locations, and users, and the latter could simplify the network management tasks and accelerate network upgrades.

Despite software-defined wide area network shows great potential in implementing high-performance wide area networks, there is still a long way ahead to fully realize its talents in practice. In the past several years, lots of papers have been published to push software-defined wide area network towards the goal of wide deployment on the Internet [3]. To demystify software-defined wide area network, we survey the representative efforts made in the literature. Different from previous taxonomies, we analyze the market demands and the drawbacks of legacy wide area network, along with the rationality of software-defined wide area network. Besides presenting the remarkable solutions in software-defined wide area network, we attempt to facilitate the development of SD-WAN based multi-objective networking with emerging techniques.

First, we revisit the status and challenges of legacy wide area network. As new applications emerge and they have high expectations of user-perceived quality of experience, previous approaches that follow a best effort mentality are no longer good enough to serve these new applications [5]. Besides, enterprise networks expand fast and wide area networks need to be upgraded frequently. As network operators manually configure each of the vendor-specific devices, it slows down

* Yong Cui is the corresponding author.

the launch of new businesses much. Due to the high expenditure of bandwidth on wide area networks and the low link utilization with traditional traffic engineering techniques, wide area networks are also faced with a serious cost-efficiency problem.

Then, we introduce the architecture of software-defined wide area network and give the representative advances that push it towards the goal of wide deployment. Generally, software-defined wide area network is considered to have three layers, i.e., data layer, control layer, and application layer [3]. Such a layering method separates the control plane and data plane of wide area network and enables network operators to manage their networks flexibly and easily. Different from traditional wide area network, software-defined wide area network enables application developers and network providers to express their requirements. It translates specific requirements to compliant network configurations. In the order from bottom to top, we briefly introduce the representative progresses in each layer of software-defined wide area network.

Third, we introduce the SD-WAN based multi-objective networking and explore the possibility of applying new techniques, such as machine learning for networking and network function virtualization, on it. Inspired by the breakthroughs made by such techniques in various areas, we discuss the opportunities and challenges they may bring to the emerging networking.

The remainder of this paper is organized as follows. In Section II, we introduce the status and challenges faced by legacy wide area networks. In Section III, we introduce the architecture of software-defined wide area network. In the subsequent three sections, we survey the representative solutions in software-defined wide area network. We concentrate on the data layer in Section IV, the control layer in Section V and the application layer in Section VI. we discuss the opportunities and challenges that new techniques may bring to SD-WAN based multi-objective networking in section VII. Finally, we conclude our work in Section VIII.

II. Status and Challenges of Wide Area Network

As the emerging applications and growing businesses challenge legacy wide area networks in many aspects, network operators expect that quality-of-service (QoS) guaranteed transmission, easy network management, and high cost efficiency can be realized in next-generation wide area network.

A. Best-Effort Mentality vs. QoS-Guaranteed Transmission

Wide area network is one of the most significant networks in the Internet world. It usually transfers data over long-distance. Typically, it is designed to provide best-effort delivery and does not provide any guarantee for application requirements [2], [5]. In wide area networks, data traffic is delivered with vulnerable physical links and network devices in harsh conditions. Failures on links and devices occur frequently, which severely affects the data transmission performance and the user-perceived QoE [4]. Besides, traditional wide area networks use distributed protocols to select routing paths for packets. As the popular protocols, such as OSPF [6], update link weights with the granularity of tens of seconds, they are not sensitive enough to the sudden failures on wide area networks, which further deteriorates the network performance [7]. Best-effort mentality under these circumstances is hard to provide satisfactory service.

New applications and operational scenarios raise exacting requirements on data transmissions over wide area networks. For example, low network latency in the context of cloud game is required to guarantee the interactions between players and the user experience; Telemedicine also depends on low-latency networks to achieve real-time operations, which is often required to be within several to hundreds of milliseconds. Such new applications and scenarios are not compatible with the outdated best-effort mentality used by traditional wide area network [2].

B. Manual Configuration vs. Easy Management

To satisfy the need for growing business, enterprises run more and more branch offices in different geographic locations and scale out their networks every now and then. ISP is faced with the demands of launching new businesses in their networks at times. Over many years of developments, there are hundreds of thousands of network devices on these wide area networks and each device should be configured in a low-level vendor-specific manner [3]. To upgrade the network successfully, network operators generally have to configure these devices manually with a long time, which slows down business development. Even worse, the rapid growth of the network together with the changing networking conditions result in network operators constantly performing manual changes to network configurations, thereby compounding the complexity of the configuration process and introducing additional configuration errors [3]. As managing traditional networks is often a cumbersome and error-prone task [1], every time the enterprises or Internet service providers change their network architectures, they need to appropriate many network operators and a long time but cannot guarantee the accuracy.

Traditional wide area network is being faced with the challenges of flexible and quick network upgrades. The changing application requirement further aggravates such situation. For example, some Internet festivals, such as the Black Friday, require e-commerce providers to provisionally increase bandwidth to deal with the predictably increasing data traffic. Network operators have to pay a great deal of time and arduous labor for twice to add bandwidth before these events and remove it later. With the proliferation of Internet services, such events become popular and network operators look forward to easy paradigms for network management.

C. Low Link Utilization vs. High Cost Efficiency

The cost efficiency of wide area networks also troubles network providers. As is well-known, bandwidth on wide area networks is an expensive resource, with an amortized annual cost of 100s of millions of dollars to provide 100s of Gbps to Tbps of capacity over long distances [8]. Along with the quick growth of traffic volume on the Internet, network operators have to install much more bandwidth capacity to satisfy the transmission requirements. According to a report published by Cisco [9], IP traffic across the world will rise more than threefold between 2017 and 2022, which means that the cost of wide area network bandwidth will increase several times in the next few years. As the growth of bandwidth on wide area networks has been decelerating for many years, the contrast between the demand and supply of it should be well considered and addressed [10], [11].

Despite the bandwidth on wide area networks is a scarce and valuable resource, it has not been fully utilized in practice. The average utilization of even the busier links on inter-datacenter wide area networks is just 40-60% [8]. There are two reasons leading to the inefficiency. First, link failures and device failures are prevalent in wide area network. In order to mask such failures, wide area networks are often over-provisioned. Second, the lack of coordination among the services that use the same network. Commonly, services send traffic whenever they want and however much they want, which causes a high spike of bandwidth usage [8]. To avoid congestion and packet loss inside networks, network operators have to overprovision to handle traffic peaks, thus the network is under-subscribed on average. The low network utilization prevents network providers from getting the full return from their investments, against the goal of pursuing high cost-efficiency.

III. Architecture of Software-Defined Wide Area Network

The static and inflexible architecture of legacy wide area network is inapposite to cope with today's increasingly dynamic networking trends and meet the QoE requirements of modern users. Software-defined wide area network has been widely discussed to replace the legacy network. The main vision of software-defined wide area network is to simplify networking operations in wide area networks, optimize wide area network management and introduce innovation and flexibility as compared to legacy wide area network architectures [3]. In the following, we give an overview of the logical and physical architectures of software-defined wide area network.

A. Logical Architecture

As shown in Fig. 1, there are three layers from bottom to top in software-defined wide area network, including data layer, control layer, and application layer [3]. The functions of data layer can be classified into bandwidth virtualization and data forwarding. Generally, there are several kinds of networks in a wide area network, e.g., multiple protocol label switching fabric, Internet, 4G and so on. To fully utilize the bandwidth resources, bandwidth virtualization combines different network links serving one location into a resource pool available for all applications and services. Data forwarding consists of a distributed set of forwarding network elements (mainly switches) in charge of forwarding packets using the bandwidth provided by bandwidth virtualization [3]. Both of them receive commands from upper-layer network controller through interface protocols such as OpenFlow [12].

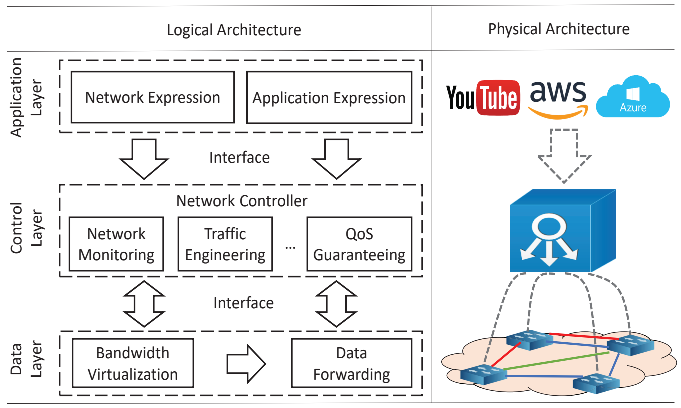

Fig. 1. Software-defined wide area network: logical and physical architecture

There are many network functions in the control layer [4]. Such network functions are implemented and managed independently [13]. Decoupling these functions enables network operators to develop, modify, debug and remove arbitrary one of them at a low cost while not affecting others. In addition to working independently, network functions can be connected or chained together to create manifold services, and increase the flexibility of software-defined wide area network [14]. For example, network monitoring provides a global network view to traffic engineering, with which the latter computes an optimal scheduling solution to execute in the network [15], [8]. QoS guaranteeing takes charge of satisfying application requirements during data transmission [16].

Application layer enables network providers and application developers to declare their specific requirements for network through network expression and application expression, both of which are able to translate high-level requirements expressed almost in natural language into compliant network configurations [17]. As more and more applications come up with multi-dimensional requirements, and sometimes such requirements are conflictive, it is necessary to customize network policies while taking into account the application characteristics [18]. For example, live video streaming service expects high bitrate and low latency to satisfy users, while such objectives conflict with each other. With application expression, application developers can declare their strategies about handling the stubborn requirements and carry them out in underlying wide area network. Similar to application expression, network expression is designed for reporting networking requirements, such as cost-efficient networking [19] and multi-objective networking [16]. Application layer enables network providers and application developers to be more involved in controlling the network.

B. Physical Architecture

We present the physical architecture of software-defined wide area network on the right side of logical one in Fig. 1. In the data layer, there are a set of SDN switches interconnected with each other by physical links [3]. A network controller is in charge of these devices. Typically, the network controller is a server or a cluster, depending on the network size and complexity [20]. Various network functions are charged by the network controller. On top of the network controller are the specific applications. Application developers and network providers can express their requirements to the network controller, and the network controller will transform them into compliant policies and configurations. Generally, there are more than one network controllers distributed in different sites, with one selected as the master controller and others as backup controllers [21]. When the master controller fails, one of the backup controllers will take over immediately [3].

IV. Advances in Data Layer

In this section, we present the representative advances in the data layer.

A. Bandwidth Virtualization

In a wide area network, there exist several kinds of networks, e.g., the public Internet and the dedicated inter-datacenter wide area network [8]. In the data layer, bandwidth virtualization is able to combine the links serving one location into a resource pool available for all applications and services. Through the network controller, it can centrally manage and automatically configure the branch devices, including various Internet access links and dedicated links. Network controller keeps track of the resource usage and provides intelligent routing functions to ensure the efficiency of data transmission.

VIA [22] is a representative work that leverages the public Internet and private overlay networks to improve the quality of Internet telephony calls. By analyzing a data set of millions of calls from Skype, Jiang et al. identify the call quality problems. Taking into account the emergence of private backbones in recent years to connect globally distributed data centers, which can serve as a readily available infrastructure for a managed overlay network, they propose VIA, which uses data center servers as relay nodes to mitigate the bad effects caused by the poor performance of the public Internet. It is able to cut the incidence of poor network conditions for calls by more than 45%. As VIA is derived from data-driven methods, it needs to aggregate sufficient data for building model, then uses a prediction-based method to select relay nodes for calls. Due to data collection is slow, it updates the model every several hours and cannot quickly adapt to changes in operating conditions, such as network jitters.

B. Interface Protocol

In software-defined wide area network, the separation of control layer and data layer requires data layer devices to be connected to the network controller via an open vendor-agnostic southbound interface [3]. There are two classical interfaces, OpenFlow [23] and Forwarding and Control Element Separation [24].

OpenFlow is the most common southbound interface. It is standardized by the Open Networking Foundation. It describes the interaction of one or more control servers with OpenFlow-compliant switches. An OpenFlow controller installs flow table entries in switches so that these switches can forward traffic according to these entries. Forwarding and Control Element Separation (ForCES) is another option for the southbound interface. It has been standardized by the Internet Engineering Task Force since 2004. In ForCES, forwarding devices are modeled using logical function blocks that can be composed in a modular way to form complex forwarding mechanisms. Each logical function block provides a given functionality. These logical function blocks model a forwarding device and cooperate with each other to form complex network devices. Control elements use the ForCES protocol to configure the interconnected logical function blocks to modify the behavior of the forwarding elements.

V. Advances in Control Layer

In this section, we focus on the advances about network controller and network functions.

A. Network Controller

As applications pose particular challenges to designers of SD-WAN because of the limited scaling-out ability of central controllers, pervasive link failure and disrupted connectivity between the control and data planes in wide area networks, and the prolonged network consistency caused by the varying propagation delays through long distances, the design of network controller faces strict requirements [4]. Many studies have been conducted in recent years to address the three main challenges faces by network controller in software-defined wide area network, i.e., scale-out behavior, network consistency, and failure resiliency. We broadly classify them into three categories and introduce them one by one.

1) Scalaibility: Onix [25] is a logically centralized and physically distributed controller platform for large-scale production networks. It supports state synchronizing and distributed computing schemes among the controller nodes in the same cluster [20]. It uses multiple interconnected controllers sharing a global network-wide view and allowing for the development of centralized control applications to guarantee the scalability and flexibility [3]. Its general API for the flexible development of control applications allows them to make trade-offs among consistency, durability, and scalability. Taking into account the large-size network statistics/state information in a large-scale network, a cluster working as a network controller will be a possible resort [20].

B4 [15], [26] is Google's globally-deployed software-defined inter-datacenter wide area network solution. It connects the data centers in different locations with a two-level hierarchical control framework. At the lower layer, each data center site contains a network controller and it hosts local control applications managing site-level traffic. There is a logically centralized traffic engineering server at the upper layer. It enforces high-level traffic engineering policies that are mainly aimed at optimizing bandwidth allocation between competing applications across the different data-center sites [3]. With the consideration of the characteristics of Google's traffic and inter-datacenter wide area network, custom switches are designed to support B4. Benefiting from the hierarchical management and high-performance custom devices, B4 has good scalability and is able to manage increasing inter-datacenter traffic flexibly. In B4, the network controller dynamically reallocates bandwidth for shifting application demands and also provides dynamic rerouting in the case of link or switch failures. It is able to drive links to near 100% utilization in practice, much higher than the conventional wide area network traffic engineering solutions.

DISCO [27] and SDX [28] work with inter-domain wide area networks. DISCO consists of multiple controllers and operates in multi-domain heterogeneous environments. Each controller in DISCO manages its own domain and interacts with others to provide end-to-end network services. DISCO uses a lightweight control channel to share summary network-wide information between controllers. Despite it is possible to be adapted in the Internet-scale networks, it has limited reliability to deal with geographically-distributed network failures. SDXes interconnect participants of different domains via a shared software-based platform. It is derived from the idea of deploying software-defined techniques at Internet exchange points. The SDX architecture consists of a SDX controller handling both SDX policies and BGP routes, conventional Edge routers, and an OpenFlow-enabled switching fabric. As the SDX controller is the central element in the SDX architecture, both security and reliability issues should be considered in practice.

2) Consistency: Strict scalability and failure resiliency requirements on software-defined wide area network makes the use of a distributed control plane almost inevitable. How to keep the control plane logically centralized is a hard problem because of the infeasibility to update the entire network atomically while maintaining full network operation [4]. Fortunately, recent efforts have been made to update the network in a consistent manner, which can be classified into two categories: plan-based consistency and state-based consistency.

Microsoft has designed software-driven wide area network, i.e., SWAN, as its software-defined wide area network solution to improve the efficiency, reliability, and fairness of their inter-datacenter wide area network [8]. It adopts a plan-based approach to achieve consistent updates in network controllers. To avoid congestion during network updates, SWAN computes a multi-step congestion-free transition plan. Each step involves one or more changes to the state of one or more switches, but irrespective of the order in which the changes are applied, there will be no congestion. Similar works such as [29] use the plan-based network updates to guarantee the network consistency in data centers. Despite they work well for the already-planned updates, such solutions do not adapt to runtime variabilities and can still cause inconsistencies while rolling out an update [4].

Dionysus [30] and ANU[1] [31] are state-based consistency solutions. Dionysus [30] uses heuristics to pick a path through a dependency graph containing the required update steps in real time. It predetermines valid orderings of updates and then heuristically applies them based on runtime behavior of the switches and network. Specifically, it starts by computing a dependency graph of the steps to reach a consistent final state, then a scheduler inside Dionysus selects a path through this graph to update policies while maintaining consistency and correctness. Other than updating network based on the runtime states, ANU proposes a novel abstraction for network updates. It provides two distinct consistency levels, per-packet and per-flow, and present general mechanisms for implementing them in software-defined network using switch APIs like OpenFlow [23].

3) Reliability: Failures such as device failures and link failures are common in wide area networks and network performance will be severely degraded if they are not handled properly. As failed link can be removed immediately after it occurs failure, many researchers concentrate on the reliability of devices (controllers and switches) in wide area networks and provide many solutions. ONOS [21] and Hyperflow [32] are two representative failover systems. When HyperFlow [32] has discovered a controller failure, it reconfigures the affected switches and redirects them to another nearby controller instance. Nevertheless, ONOS [21] guards against controller instance failures by connecting each SDN switch to more than one controller. When the master controller fails, backup ones will take over instantly. It also incorporates additional recovery protocols for healing from lost updates due to such controller crashes.

B. Network Functions

There are also some representative works to promote the control layer of software-defined wide area network by implementing various network functions. In this part, we focus on networking monitoring, traffic engineering, and QoS-guaranteeing, which are closely related to application requirements.

1) Network Monitoring: Monitoring is an important concept in network management as it helps network operators to determine the behavior of a network and the status of its components. Traffic engineering, QoS guaranteeing, and anomaly detection also depend on networking monitoring for decision making. As software-defined networking is becoming increasingly popular for network provision and management tasks, network monitoring should be placed in a significant place and gets more attentions [33].

In the past several years, OpenFlow [23] has emerged as the most common southbound interface in SDN and SD-WAN. It provides a flow level statistics collection mechanism from the data plane and exposes a high-level interface for per flow and aggregate statistics collection, which can be used by network functions to monitor network status without being concerned about the low-level details. In order to keep the switch design simple, this statistics collection mechanism is implemented as a pull-based service, i.e. network applications

[1] ANU is the abbreviation of "abstractions for network update".

and network controller periodically query the switches about flow statistics. Since the frequency of polling the switches determines monitoring accuracy and network overhead, the trade-off between accuracy and overhead puts network operators in a bind. PayLess [34] is a network monitoring framework for software-defined networking platforms that take advantages of OpenFlow. It provides an abstract view of the network and a uniform way to request statistics about the resources. Moreover, it is developed as a collection of pluggable components. Interactions between these components are abstracted by well-defined interfaces. Hence, one can develop custom components and plug into the PayLess framework. To the best of our knowledge, PayLess is the first monitoring framework for software-defined networks. It provides generic RESTful API to network applications and controllers to obtain network status easily. Also, it becomes possible for components to be customized without affecting others in Payless framework.

2) Traffic Engineering: Distributed protocols like OSPF/IS-IS are the dominant routing solutions on the Internet [7]. Their easy-to-implement feature helps them make success in the early days of the Internet. However, our networks have grown rapidly in size and complexity in the past years, which magnifies the drawbacks of these routing protocols. For example, even the best link weight settings may lead to routing that deviates significantly from the optimal routing assignments [35]. Besides that, poor resource utilization resulting from OSPF bothers network operators. They are forced to overprovision their networks to handle peak traffic. As a result, most network links run at just 30-40% utilization on average [7]. Software-defined wide area network brings encouragement to traffic optimization in wide area networks and firstly used in inter-datacenter networks [8], [15], [36].

SWAN [8] is a SD-WAN based centralized traffic engineering solution designed by Microsoft and used to transfer data between data centers. It addresses the shortcomings of today's primarily used decentralized TE practice and makes a strong point for using a global network view and software-defined networking in order to solve the global and network-wide problem of traffic engineering. It uses fine-grained policy rules and treated traffic according to their priorities in order to carry more high-priority traffic (e.g., interactive traffic) while maintaining fairness among services of the same class. The global network view is used to find globally optimized bandwidth to path assignments through the network. Fine-grained control is used to make and enforce bandwidth reservations on a per-application basis. To achieve higher utilization, SWAN takes 5 minutes as a period and predicts the interactive traffic in next 5 minutes, so that the left bandwidth can be used by background traffic. By doing so, the need for over-subscription is drastically reduced, and resources are used more effectively.

Similar to SWAN, B4 [15] is Google's SD-WAN based traffic engineering solution deployed in inter-datacenter wide area network. B4 adopts a two-level hierarchical control plane to execute the global traffic scheduling. The network controller dynamically reallocates bandwidth for shifting application demands and also provides dynamic rerouting in the case of link or switch failures. Different from the centralized controller that does global allocation based on linear programming, which is complicated and time-consuming, B4 uses fast and easy-to-implement heuristics instead to allocate bandwidth. It is able to drive links to near 100% utilization in practice, which is comparable to the capability of SWAN [8].

3) QoS Guaranteeing: As new applications, such as video streaming services, have growing expectations of user-perceived QoE, they require networks to provide QoS-aware services. Software-defined wide area network has been studied to achieve the above objective in wide area networks.

For video streaming over the Internet, despite content delivery networks and adaptive bitrate algorithms are useful to improve user-perceived video quality, the root causes of congestion problems have not been well addressed, said by Nam et al. [16]. To pinpoint a bottleneck and improve video QoE, they propose a software-defined networking platform from the point of view of over-the-top video service providers. It is designed to monitor network conditions of streaming flow in real time and dynamically change routing paths using multi-protocol label switching traffic engineering to provide reliable video watching experience. In [37], a QoS-aware adaptive routing (QAR) is proposed in the designed multi-layer hierarchical architecture of software-defined wide area network. Specifically, the distributed hierarchical control plane architecture is employed to minimize signaling delay in large software-defined networks via the three-levels design of controllers, including the super controllers, domain controllers, and slave controllers. QAR algorithm is proposed with the aid of reinforcement learning and QoS-aware reward function. It achieves a time-efficient, adaptive, QoS-provisioning packet forwarding services on the data plane.

VI. Advances in Application Layer

In this section, we briefly introduce the representative progresses about network expression and application expression in application layer.

A. Network Expression

As the Internet usually depends on more than one networks (e.g., carrier networks and inter-datacenter wide area networks) to provide end-to-end data transmission, properly connecting these networks is necessary for high-quality end-to-end data transmission. Espresso [38] is Google's SDN-based Internet peering edge routing infrastructure, which runs in peering edge routers that connect external peers and data centers. Espresso's design greatly accelerates deployment of new networking features at peering edge. According to three years of historical data, Espresso's entire control plane has been updated more than 50 times more frequently than traditional peering routers. Benefiting from its programmability, Espresso shows high feature velocity in practice, which is able to promote the evolution of the Internet.

DEFO [17] is proposed to simplify network management while preserving high robustness and scalability in carrier-grade networks. Different from conventional network management, it uses declarative and expressive approaches to control forwarding paths. It is a two-layer architecture separating connectivity and optimization tasks. Even more amazing, DEFO is able to translate high-level goals expressed almost in natural language into compliant network configurations. Evaluated with real and synthetic traces, DEFO improves the state of the art from many aspects. Besides that it achieves better trade-offs for classic goals covered by previous works, it also supports a larger set of goals such as fine-grained traffic engineering and service chaining. As for the operation time, it can optimize large carrier networks in a few seconds, considerably reducing the time required to upgrade networks.

As more businesses are moved to clouds, inter-datacenter bandwidth becomes an ever more valuable and congested resource. Cost-efficient networking is expected by most cloud providers, while the traditional separation between the economic and engineering aspects makes it difficult to steer customer demand to lightly loaded paths and times, which is important for managing costs and providing service guarantees. To achieve this objective, Pretium [19] combines dynamic pricing with traffic engineering for inter-datacenter wide area networks to schedule traffic economically. In Pretium, users specify their required rates or transfer sizes with deadlines, and a price module generates a price quote for different guarantees on these requests. The price quote is generated using internal prices which are maintained and periodically updated by Pretium based on history. Evaluated by the researchers, Pretium is able to achieve up to 80% of the social welfare of an offline oracular scheme, significantly outperforming usage-based pricing alternatives.

B. Application Expression

Besides network providers, application developers raise strict requirements for networks to guarantee service quality. In software-defined wide area network, some solutions are proposed to support priority classification for traffics and multi-objective network optimization.

1) Priority classification: As different services raise different requirements, handling traffics differently may bring more benefits in practice. To carry more traffic and support fair sharing, many solutions have been proposed. One important idea is to set priority for different traffic.

SWAN [8] is an software-defined wide area network system that boosts the utilization of inter-datacenter networks while meeting policy goals expressed by applications. In SWAN, traffics are classified into three categories, including interactive, elastic and background, according to the characteristics of the services that generate them. For example, queries and response in search engines are highly sensitive to packet loss and delay, and they should be scheduled with the highest priority, thus the data from such services is regarded as interactive traffic. As the services that copying all the data of a service from one data center to another data center for long-term storage has no explicit deadline or a long deadline, and they can be scheduled with a lower priority, data from such services is regarded as background traffic. In practice, SWAN allows interactive traffic to preempt bandwidth. It is sent as soon as possible to meet the strict service level agreements. For other traffic, such as elastic and background traffic, SWAN computes how much traffic each service can send and configures the networks data plane to carry that traffic while taking into account the fairness among similar services. By this means, SWAN is able to fully use the bandwidth and guarantee the performance of different services simultaneously. Besides the way that sets priority classes based on the type of traffic, the other is based on a function to capture the relative priority. For example, B4 [15] uses a function to capture the relative priority. It uses bandwidth function which calculates each application's traffic by administrator-specified static weights in their datacenter.

2) Multi-objective optimization: As user-perceived QoE is critical for Internet videos and bottlenecks could occur anywhere in the delivery system to affect user experience, a robust bitrate adaptation algorithm is necessary to ensure good QoE [18]. For applications such as Internet videos, user experience is usually related to many factors, such as bitrate, latency and jitter rate. Optimizing them together brings great challenges to network operators. Even worse, many of them are contending with each other, e.g., high bitrate vs. low latency. Balancing the weights of these metrics and achieving multi-objective optimization worth in-depth study.

MPC [18] is a representative work focusing on multi-objective optimization for Internet videos. It employs model predictive control algorithms to select bitrates that are expected to maximize QoE over a horizon of several future chunks based on throughput estimates and buffer occupancy. It uses different weighting parameters corresponding to video quality variations, rebuffering time and startup delay, and develops a principled control-theoretic model to reason about a broad spectrum of strategies. Said by the authors, MPC is able to outperform the industry reference player by more than 60% in terms of median QoE. As it relies heavily on accurate throughput estimates which are not always available. When throughput predictions are incorrect, the performance of MPC will degrade significantly. Despite drawbacks exist, MPC provides insights into the field of multi-objective optimization in networks.

VII. Future Perspectives: Opportunities and Challenges

In this section, we consider to use emerging techniques such as machine learning for networking and network function virtualization, and new transport protocols to facilitate the development of SD-WAN based multi-objective networking.

A. Multi-Objective Networking

Along with the fast development of the Internet, new applications and operational scenarios emerge and raise exacting requirements on networks. Except for data transferring, services have other requirements on networks, such as high throughput, low latency, and high stability [2]. Multi-objective networking has been widely discussed by industry and academia and the SD-WAN based multi-objective networking is of great potential.

Low-latency networking is one kind of multi-objective networking, as new applications and operational scenarios place exacting requirements on latency [2]. For example, players of cloud games expect low-latency data transmission over networks to enjoy the interaction with each other [2]. Virtual reality (VR), as a computing-intensive application, expects low-latency data transmission between cloud servers and local devices to improve the rendering efficiency and user experience [39]. In [2], Zuo et al. present a survey of network latency and approaches to reduce latency, particularly the delays resulting from the protocol design and functionalities. They summarize the factors impacting delay from different layers of the network architecture and review some state-of-the-art solutions to reduce latency at each layer. Lai et al. [39] present Furion, a VR framework that enables high-quality immersive mobile VR on todays mobile devices and wireless networks, which separates foreground interactions and background environment, and employs a split renderer architecture running on both the phone and the server to achieve low-latency transmission. Despite such methods can mitigate the bad effects caused by higher latency in networks on some scenarios, they are not fit for all cases. With the proliferation of cloud applications, data transmission between remote servers and local devices will become more and more popular. Achieving low-latency networking requires network operators to eliminate long round-trip time and get clouds much "closer" to users, which is essential for realizing multi-objective networking.

B. Machine Learning for Networking

Recent years, machine learning methods have been used to solve networking problems and show great promise. Mao et al. [40] build a system, Pensieve, to generate adaptive bitrate algorithms using reinforcement learning techniques, which outperforms the best state-of-the-art scheme, with improvements in average QoE of 12-25%. Compared with conventional solutions, Pensieve is able to automatically learn adaptive bitrate algorithms that adapt to a wide range of environments and QoE metrics. Except for Pensieve, Chen et al. [41] use deep reinforcement learning methods to conduct traffic optimization in data center-scale networks. It can achieve up to 48% reduction in average flow completion time over existing solutions. Despite network status varies over time and machine learning for networking shows great promise in handling changing environments, how to balance the effectiveness and generalization ability of machine learning models is a hard problem in the context of networking. In addition, setting the right optimization targets is a critical but challenging task for machine learning models [42].

C. Network Function Virtualization

Network function virtualization provides a new method to build IT applications. Different from the traditional network functions that depend on custom hardware, virtual network functions can run as software in commercial devices [13]. There are many benefits of coupling network function virtualization and software-defined wide area network: (1) Capital expenditure reduction. Network operators can lower network costs by running virtual network functions on cheap commercial devices other than expensive customized hardware [14]. (2) Operation expenditure reduction. Network function virtualization automates the entire end-to-end provisioning process. A network operator can spin-up the appropriate at their required devices with prerequisite configuration and capacity with only one click of a button. Comparing with manual operations in vendor-specific devices, network function virtualization reduces the manpower, distinct provisioning, and management systems required [14]. (3) Service agility and flexibility. The adoption of microservices and service chaining in network function virtualization makes it easy to add new features and functionality [43]. Together with flexible open service environments and software-defined networking techniques, network function virtualization transforms the service evolution process and reduces the time required to launch new services from years to months [44]. Although network function virtualization has many good characteristics and shows great potential in complementing software-defined wide area network, its performance under off-the-rack techniques is not comparative to that achieved by custom hardware.

D. New Transport Protocols

As applications are often limited by the use of TCP as the underlying transport, which introduces extra latency due to its handshake mechanism, to achieve low network latency without sacrificing security and reliability, developing new transport protocols such as Quick UDP Internet Connections (QUIC) to achieve low network latency is of great promise [45], [46]. QUIC is defined on top of UDP. It aims to reduce connection latency by sending data directly when establishing a connection in the best case. It also provides multiplexing features optimized for HTTP/2 and richer feedback information that might allow for new congestion control approaches. Different from TCP that is built in the system kernel and brings difficulties to protocol modification, QUIC can be easily implemented and updated in user space [46]. As most applications are developed on the basis of legacy transport protocols and such protocols are widely used in practice, there is still a long way ahead to replace them with new transport protocols.

VIII. Conclusion

Software-defined wide area network is worth studying and it is of critical importance for the next-generation wide area network. In this paper, we introduced the status and challenges that legacy wide area networks face. As software-defined wide area network is regarded as the promising architecture of next-generation design of wide area network, we presented the logical and physical architectures of it and briefly surveyed the representative progresses made to improve it. Inspired by the breakthroughs made by emerging techniques, including machine learning for networking and network function virtualization, along with new transport protocols, we discussed the opportunities and challenges that they may bring to SD-WAN based multi-objective networking. We hope that the analysis in this paper helps to drive the software-define wide area network forward.

Acknowledgements. This work is supported by National Key R&D Program of China under Grant 2017YFB1010002, NSFC (no. 61872211) and CSC (no. 201806210244).

References

[1] K.-T. Foerster, S. Schmid, and S. Vissicchio, "Survey of consistent software-defined network updates," *IEEE Communications Surveys & Tutorials*, 2018.

[2] X. Zuo, Y. Cui, M. Wang, T. Wang, and X. Wang, "Low-latency networking: Architecture, techniques, and opportunities," *IEEE Internet Computing*, 2018.

[3] F. Bannour, S. Souihi, and A. Mellouk, "Distributed sdn control: Survey, taxonomy, and challenges," *IEEE Communications Surveys & Tutorials*, 2018.

[4] O. Michel and E. Keller, "Sdn in wide-area networks: A survey," in *IEEE Fourth International Conference on Software Defined Systems (2017)*.

[5] J. Jiang, V. Sekar, I. Stoica, and H. Zhang, "Unleashing the potential of data-driven networking," in *Springer International Conference on Communication Systems and Networks (2017)*.

[6] J. Moy, "Ospf version 2," Tech. Rep., 1997.

[7] N. Michael and A. Tang, "Halo: Hop-by-hop adaptive link-state optimal routing," *IEEE/ACM Transactions on Networking*, 2015.

[8] C.-Y. Hong, S. Kandula, R. Mahajan, M. Zhang, V. Gill, M. Nanduri, and R. Wattenhofer, "Achieving high utilization with software-driven wan," in *ACM SIGCOMM (2013)*.

[9] V. Cisco, "Cisco visual networking index: Forecast and trends, 2017–2022," *White Paper*, 2018.

[10] K. Hsieh, A. Harlap, N. Vijaykumar, D. Konomis, G. R. Ganger, P. B. Gibbons, and O. Mutlu, "Gaia: Geo-distributed machine learning approaching lan speeds." in *USENIX NSDI (2017)*.

[11] A. Vulimiri, C. Curino, P. B. Godfrey, T. Jungblut, J. Padhye, and G. Varghese, "Global analytics in the face of bandwidth and regulatory constraints." in *USENIX NSDI (2015)*.

[12] W. Braun and M. Menth, "Software-defined networking using openflow: Protocols, applications and architectural design choices," *Future Internet*, 2014.

[13] Z. A. Qazi, C.-C. Tu, L. Chiang, R. Miao, V. Sekar, and M. Yu, "Simple-fying middlebox policy enforcement using sdn," in *ACM SIGCOMM (2013)*.

[14] A. Gember-Jacobson, R. Viswanathan, C. Prakash, R. Grandl, J. Khalid, S. Das, and A. Akella, "Opennf: Enabling innovation in network function control," in *ACM SIGCOMM (2014)*.

[15] S. Jain, A. Kumar, S. Mandal, J. Ong, L. Poutievski, A. Singh, S. Venkata, J. Wanderer, J. Zhou, M. Zhu *et al.*, "B4: Experience with a globally-deployed software defined wan," in *ACM SIGCOMM (2013)*.

[16] H. Nam, K.-H. Kim, J. Y. Kim, and H. Schulzrinne, "Towards qoe-aware video streaming using sdn," in *IEEE Global Communications Conference (2014)*.

[17] R. Hartert, S. Vissicchio, P. Schaus, O. Bonaventure, C. Filsfils, T. Telkamp, and P. Francois, "A declarative and expressive approach to control forwarding paths in carrier-grade networks," *ACM SIGCOMM (2015)*.

[18] X. Yin, A. Jindal, V. Sekar, and B. Sinopoli, "A control-theoretic approach for dynamic adaptive video streaming over http," in *ACM SIGCOMM (2015)*.

[19] V. Jalaparti, I. Bliznets, S. Kandula, B. Lucier, and I. Menache, "Dynamic pricing and traffic engineering for timely inter-datacenter transfers," in *ACM SIGCOMM (2016)*.

[20] S. Liu and B. Li, "On scaling software-defined networking in wide-area networks," *Tsinghua Science and Technology*, 2015.

[21] P. Berde, M. Gerola, J. Hart, Y. Higuchi, M. Kobayashi, T. Koide, B. Lantz, B. O'Connor, P. Radoslavov, W. Snow *et al.*, "Onos: towards an open, distributed sdn os," in *ACM HotSDN (2014)*.

[22] J. Jiang, R. Das, G. Ananthanarayanan, P. A. Chou, V. Padmanabhan, V. Sekar, E. Dominique, M. Goliszewski, D. Kukoleca, R. Vafin *et al.*, "Via: Improving internet telephony call quality using predictive relay selection," in *ACM SIGCOMM (2016)*.

[23] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "Openflow: enabling innovation in campus networks," *ACM SIGCOMM Computer Communication Review*, 2008.

[24] A. Doria, J. H. Salim, R. Haas, H. Khosravi, W. Wang, L. Dong, R. Gopal, and J. Halpern, "Forwarding and control element separation (forces) protocol specification," Tech. Rep., 2010.

[25] T. Koponen, M. Casado, N. Gude, J. Stribling, L. Poutievski, M. Zhu, R. Ramanathan, Y. Iwata, H. Inoue, T. Hama *et al.*, "Onix: A distributed control platform for large-scale production networks." in *USENIX OSDI (2010)*.

[26] C.-Y. Hong, S. Mandal, M. Al-Fares, M. Zhu, R. Alimi, C. Bhagat, S. Jain, J. Kaimal, S. Liang, K. Mendelev *et al.*, "B4 and after: managing hierarchy, partitioning, and asymmetry for availability and scale in google's software-defined wan," in *ACM SIGCOMM (2018)*.

[27] K. Phemius, M. Bouet, and J. Leguay, "Disco: Distributed multi-domain sdn controllers," in *IEEE Network Operations and Management Symposium (2014)*.

[28] A. Gupta, L. Vanbever, M. Shahbaz, S. P. Donovan, B. Schlinker, N. Feamster, J. Rexford, S. Shenker, R. Clark, and E. Katz-Bassett, "Sdx: A software defined internet exchange," *ACM SIGCOMM Computer Communication Review*, 2015.

[29] H. H. Liu, X. Wu, M. Zhang, L. Yuan, R. Wattenhofer, and D. Maltz, "zupdate: Updating data center networks with zero loss," in *ACM SIGCOMM (2013)*.

[30] X. Jin, H. H. Liu, R. Gandhi, S. Kandula, R. Mahajan, M. Zhang, J. Rexford, and R. Wattenhofer, "Dynamic scheduling of network updates," in *ACM SIGCOMM (2016)*.

[31] M. Reitblatt, N. Foster, J. Rexford, C. Schlesinger, and D. Walker, "Abstractions for network update," *ACM SIGCOMM (2012)*.

[32] A. Tootoonchian and Y. Ganjali, "Hyperflow: A distributed control plane for openflow," in *Proceedings of the 2010 internet network management conference on Research on enterprise networking (2010)*.

[33] P.-W. Tsai, C.-W. Tsai, C.-W. Hsu, and C.-S. Yang, "Network monitoring in software-defined networking: A review," *IEEE Systems Journal*, 2018.

[34] S. R. Chowdhury, M. F. Bari, R. Ahmed, and R. Boutaba, "Payless: A low cost network monitoring framework for software defined networks," in *IEEE Network Operations and Management Symposium (2014)*.

[35] B. Fortz and M. Thorup, "Increasing internet capacity using local search," *Computational Optimization and Applications*, 2004.

[36] A. Mendiola, J. Astorga, E. Jacob, and M. Higuero, "A survey on the contributions of software-defined networking to traffic engineering," *IEEE Communications Surveys & Tutorials*, 2017.

[37] S.-C. Lin, I. F. Akyildiz, P. Wang, and M. Luo, "Qos-aware adaptive routing in multi-layer hierarchical software defined networks: A reinforcement learning approach," in *IEEE International Conference on Services Computing (2016)*.

[38] K.-K. Yap, M. Motiwala, J. Rahe, S. Padgett, M. Holliman, G. Baldus, M. Hines, T. Kim, A. Narayanan, A. Jain *et al.*, "Taking the edge off with espresso: Scale, reliability and programmability for global internet peering," in *ACM SIGCOMM (2017)*.

[39] Z. Lai, Y. C. Hu, Y. Cui, L. Sun, and N. Dai, "Furion: Engineering high-quality immersive virtual reality on today's mobile devices," in *ACM MobiCom (2017)*.

[40] H. Mao, R. Netravali, and M. Alizadeh, "Neural adaptive video streaming with pensieve," in *ACM SIGCOMM (2017)*.

[41] L. Chen, J. Lingys, K. Chen, and F. Liu, "Auto: Scaling deep reinforcement learning for datacenter-scale automatic traffic optimization," in *ACM SIGCOMM (2018)*.

[42] M. Wang, Y. Cui, X. Wang, S. Xiao, and J. Jiang, "Machine learning for networking: Workflow, advances and opportunities," *IEEE Network*, 2018.

[43] C. Sun, J. Bi, Z. Zheng, H. Yu, and H. Hu, "Nfp: Enabling network function parallelism in nfv," in *ACM SIGCOMM (2017)*.

[44] H. Hawilo, A. Shami, M. Mirahmadi, and R. Asal, "Nfv: State of the art, challenges and implementation in next generation mobile networks (vepc)," *arXiv preprint arXiv:1409.4149*, 2014.

[45] A. Langley, A. Riddoch, A. Wilk, A. Vicente, C. Krasic, D. Zhang, F. Yang, F. Kouranov, I. Swett, J. Iyengar *et al.*, "The quic transport protocol: Design and internet-scale deployment," in *ACM SIGCOMM (2017)*.

[46] Y. Cui, T. Li, C. Liu, X. Wang, and M. Kühlewind, "Innovating transport with quic: Design approaches and research challenges," *IEEE Internet Computing*, 2017.