

Article

# Latency-Based Dynamic Controller Assignment in Hybrid SDNs: Considering the Impact of Legacy Routers

Jie Yuan <sup>1,\*</sup>, Erxia Li <sup>2</sup>, Chaoqun Kang <sup>2</sup>, Fangyuan Chang <sup>2</sup>, Tingting Yuan <sup>3</sup>  and Xiaoyong Li <sup>1,\*</sup>

<sup>1</sup> Key Laboratory of Trustworthy Distributed Computing and Service, Beijing University of Posts and Telecommunications, Ministry of Education, Haidian District, Beijing 100876, China

<sup>2</sup> China Electric Power Research Institute, Haidian District, Beijing 100192, China

<sup>3</sup> DIANA Team, Institut National de Recherche en Automatique et Informatique (INRIA), Paris 75020, France

\* Correspondence: yuanjie@bupt.edu.cn (J.Y.); lixiaoyong@bupt.edu.cn (X.L.)

Received: 23 June 2019; Accepted: 26 July 2019; Published: 28 July 2019



**Abstract:** Software-defined networking (SDN) is a modern network architecture, which separates the network control plane from the data plane. Considering the gradual migration from traditional networks to SDNs, the hybrid SDN, which consists of SDN-enabled devices and legacy devices, is an intermediate state. For wide-area hybrid SDNs, to guarantee the control performance, such as low latency, multi SDN controllers are usually needed to be deployed at different places. How to assign them to switches and partition the network into several control domains is a critical problem. For this problem, the control latency and the packet loss rate of control messages are important metrics, which have been considered in a lot of previous works. However, hybrid SDNs have their unique characters, which can affect the assignment scheme and have been ignored by previous studies. For example, control messages pass through Legacy Forwarding Devices (LFDs) in hybrid SDNs and cause more latency and packet loss rate for queuing compared with SDN-enabled Forwarding Devices (SFDs). In this paper, we propose a dynamic controller assignment scheme in hybrid SDNs, which is called the Legacy Based Assignment (LBA). This scheme can dynamically delegate each controller with a subset of SFDs in the hybrid SDNs, whose objective is to minimize average SFD-to-control latency. We performed some experiments compared with other schemes, which show that our scheme has a better performance in terms of the latency and the packet loss rate.

**Keywords:** hybrid SDN; dynamic controller assignment; wide area network

## 1. Introduction

Software-defined networking (SDN) [1,2] is an attractive networking paradigm that improves network performance by logically centralized control. However, due to economic and operational constraints, SDN-enabled devices may not be able to be fully deployed at a time, particularly, in large scale networks [3]. During the gradual migration period, there exists a new type of network, which is called a hybrid SDN [4,5]. This kind of network consists of Legacy Forwarding Devices (LFDs) and SDN-enabled Forwarding Devices (SFDs). LFDs can find paths and form their own routing table. SFDs are under the control of SDN controllers, which needs the guide from controllers, such as how to establish their routing tables.

In SDN architecture, SDN controllers [6,7] are essential components, which is an application that can manage flows to improve network performance. In a large SDN or a hybrid SDN, only one SDN controller may be not sufficient for numerous reasons: latency, load balance, scalability, and reliability [8,9]. Especially in a wide area network (WAN), one controller usually cannot cover

all the areas within the limited capacity and SFD-to-controller latency. Hence, several controllers [10] need to be deployed to guarantee the performance of network control. Besides, to greatly improve the control performance, SDN controllers can be located at different places [11] in SDN (or hybrid SDN) WANs.

There are two ways of communication between SDN switches and SDN controllers, namely in-band and out-of-band communication [12]. In the first case, controllers are usually directly connected with SFDs, and control messages share the same links with other data. In the second case, there is usually an independent network for the control message. For WANs, in-band communication is more suitable, because out-of-band communication with long distance needs huge deployment costs [13]. In the hybrid SDN WAN, the control messages may pass through the LFDs. Control messages are transmitted among the data packets on the same physical channel. In LFDs, which are out of control of SDN controllers, control messages need to queue with data packets. In SFDs, the control messages can be forwarded under the control of SDN controllers with a higher priority and without queuing with data packets.

A good controller assignment policy is vital to guarantee a good network management, which has been well studied. Nevertheless, an important scenario, the hybrid SDN, has been ignored in the previous studies. To optimize the assignment of controllers in hybrid SDN WANs, three important aspects should be considered. First, LFDs have impacts over the control messages latency and loss rate, especially when the legacy devices are in a bad situation, such as traffic congestion. Second, processing PACKET\_IN messages from SFDs is the most significant part of the working load of controllers. Because of the limitation on the controllers' capacity, the heavy-load of controllers brings more failure probability [14]. Thus, the capacity of controllers needs to be taken into consideration. Third, in hybrid SDNs, there are new paths generated by the SDN devices partially deployed, which can be used in control messages forwarding.

In this paper, we propose a dynamic controller assignment scheme in hybrid SDNs, which is called the Legacy Based Assignment (LBA). This scheme can dynamically delegate each controller with a subset of SFDs in the hybrid SDNs, whose objective is to minimize average SFD-to-controller latency. Firstly, we analyze the influence of LFDs of hybrid SDNs in SFD-to-controller latency, which is caused by routing queuing in LFDs. Secondly, we formulate the dynamic controller assignment problem to be a 0–1 Integer Linear Programming. In the formulation, in addition to the impact of LFDs, the capacity limitation of the controllers and the available routing paths generated by the SDN partial deployment are also considered. We performed some experiments compared with other schemes, which show that our scheme has a better performance in terms of the latency and the packet loss rate. The rest of the paper is organized as follows. Section 2 gives a brief survey on related works about the SDN controller assignment problem. Section 3 formulates controller assignment problem. Section 4 shows the performance evaluation of our approach.

## 2. Related Work

To assign multiple controllers to control SFDs is a vital issue in a large-scale SDN to provide good network management, which has been well studied.

First, many studies have been done on static controller assignment in pure SDN, which fails to consider the dynamic network environment, such as network traffic. The average latency and the worst latency of control messages were used to indicate the performance of the network in [15]. In [16], spectral clustering was used to partition a large network into several small domains, each of which is under the control of one or a group of SDN controllers. The K-Critical was proposed in [17] to decide the controller assignment based on a robust control tree. In [18], a controller assignment scheme was proposed based on community detection algorithms, which limited the spread of maximum number of label hops and maximum difference of nodes number between control domains to meet the load balancing and reliability requirements. In [19], a framework for Pareto-based optimal controller placement, was proposed to get optimal placements with respect to different performance metrics.

However, these static assignment schemes between the switches and the controllers result in long and varied controller response times [20]. To solve this problem, several methods for dynamic SDN controller assignment were proposed, which considered different factors. Firstly, the average controller response time is one of the most important factors. The study in [21] proposed dynamic SDN controller assignment scheme based on the stable matching transfers to minimize the average response time. The work in [22] leveraged a learning automaton (LA) approach to solve the controller placement problem to minimize the overall propagation latency. Secondly, the cost of maintenance on the cluster of controllers is considered in some works. For example, Wang et al. [23] tried to minimize the total cost caused by response time and maintenance on the cluster of controllers. Thirdly, the load balance of controllers can be considered in this problem, which can affect the latency of response. ElastiCon [24], an elastic distributed controller architecture, was proposed to address the load imbalance problem, in which the controller pool is dynamically grown or shrunk according to traffic conditions. Furthermore, some works considered the energy consumption of SDN controller. For example, Ruiz-Rivera et al. [25] proposed a controller assignment scheme to save the energy to switch off the maximum number of controller-switch links subject to the following constraints: delay, link and controller load. Besides, some works considered other factors, such as the recovery of failure and path selection. For example, the work in [26] considered the control-plane recovery performance to assign controllers to switches and choose the routing of controller-switch control paths. The study in [27] proposed a low-complexity master assignment (LCMA) algorithm for the controller problem to minimize the average flow setup latency with consideration of the number of assigned master controllers along a path.

Considering the gradual migration of SDNs, hybrid SDNs are inevitable and important states, which have some special characteristics compared with pure SDNs and legacy networks. Most of previous studies on the topic of hybrid SDNs focused on traffic engineering and SDN migration schemes [28,29]. Nevertheless, as one kind of SDNs, the controller assignment is also critical to hybrid SDNs, which has almost been ignored in the previous studies. A few studies noticed the importance of controllers in hybrid SDNs. In [30], the controller placement problem for hybrid SDN/legacy networks over a period of migration time is firstly introduced and formulated. However, the research on the controller assignment is still very scarce.

### 3. Problem Description and Formulation

In this section, the problem of controller dynamic assignment in the hybrid SDNs is described and formulated, whose objective is to minimize the average latency between SFDs and their controllers. The parameters and variables used in our model are summarized in Table 1. The problem description and formulation are introduced in the following part.

**Table 1.** List of notations used. SFDs: SDN-enabled Forwarding Devices. SDN: Software-defined networking. LFDs: Legacy Forwarding Devices.

Parameter	Meaning
$S$	The set of SFDs in a hybrid SDN.
$ S $	The number of SFDs.
$LN$	The set of LFDs in a hybrid SDN.
$C$	The set of SDN controllers.
$E$	The set of physical links between forwarding devices.
$P$	The location of controllers in networks.
$B_c$	The capacity of SDN controllers.
$A$	The set of paths between controllers and SFDs, whose SFD-to-controller latency is in the constraints.
$D$	The maximum latency constraint of control messages.

Table 1. Cont.

Parameter	Meaning
$d_p$	The latency of path $p$ .
$\eta_e$	Propagation latency of per kilometer of link $e \in E$ .
$d_e$	The distance between the source and the destination of link $e$ .
$\mu_v$	The service efficiency of the LFD $v$ .
$T_\alpha$	The propagation latency of one available path $\alpha$ .
$m_v$	The estimated queue length of the LFD $v$ .
$\lambda_v$	The message processing rate of the LFD $v$ .
$Q_v$	The queuing latency of $v$ .
$R_\alpha$	The SFD-to-controller latency of the path $\alpha$ for successful transmission.
$\delta_v$	The messages loss rate of the LFDs $v$ .
$L_\alpha$	The SFD-to-controller latency of the path $\alpha$ .
$F$	The maximum tolerant time for the control message re-sending.
$l(u)$	The load of the SFD $u$ .
$\omega(u, c)$	The minimum latency between SFD $u$ and a potential controller $c$ .
Variables	Meaning
$\pi_{uc}$	Binary variables, which denote whether the SFD $u$ is assigned to the controller $c$ or not.

### 3.1. Description of SDN Controller Assignment Problem

The hybrid SDN WAN consists of SFDs (for example, OpenFlow switches) and LFDs (for example, Open Shortest Path First (OSPF) switches), as shown in Figure 1. It is assumed that multi-controllers have been deployed in a large-scaled hybrid SDN WAN with different locations. The network is divided into multiple control domains, and each of them is charged by only one SDN controller. In WAN, the controllers are usually co-located with regular network nodes, such as SFDs and LFDs, and the in-band model is used for communication between SFDs and controllers.

The way of establishing the control relationship between controllers and SFDs is described as followed. Firstly, the controllers decide the set of SFDs in their charge, which is the controller assignment problem. Then, controllers send messages to their SFDs to establish the control relationship. After establishing the relationship, SFDs can find their target controllers to send control requests. As for which paths should be used for routing control messages, it depends on the flow table of SFDs, which is established according to instructions of its charged controller.

A hybrid SDN can be denoted by  $G(S, LN, C, E, P)$ , where  $S$  is the set of SFDs,  $LN$  is the set of LFDs,  $C$  is the set of controllers,  $E$  is the set of physical links in the network, and  $P$  is the location of controllers. The number of SFDs is  $|S|$  and the number of LFDs is  $|LN|$ . The major traffic of switch-controller communications includes two types of control messages: (1) PACKET\_IN; and (2) PACKET\_OUT or FLOW\_MOD [31]. Processing PACKET\_IN messages is the most significant part of the working load of controllers [14]. An SDN controller has a limited capacity, which is determined by its access bandwidth, processor, memory size and so on. To avoid the overload of the controllers, a limited number of SFDs is allowed to be assigned to one specific controller. The set of controllers is denoted by  $C = \{B_1, B_2, \dots, B_{|C|}\}$  where  $B_c$  is the capacity of the controller  $c$ , and  $|C|$  is the number of controllers.  $P$  is the location of controllers, which is denoted by  $P = \{p_1, \dots, p_{|C|}\}$ . Figure 1 illustrates  $S = \{4, 11, 12, 14, 15, 17, 19, 20, 26, 29, 31\}$ ,  $LN = \{1, 2, 3, 5, 6, 7, 8, 9, 10, 13, \dots, 35\}$ ,  $C = \{200 \text{ Mb/s}, 100 \text{ Mb/s}\}$ , and  $P = \{29, 15\}$ .

In the scenario of partial deployment of SFDs, there exist multiple available paths for SFD-to-controller packets forwarding. An available path that is loop-free is generated by the SFD routing configuration under the control of its assigned controller. For example, in Figure 1, the shortest path from Controller 1 to SFD 31 is  $29 \rightarrow 7 \rightarrow 33 \rightarrow 31$ . SFDs can choose the node in next hop under the control of controller, and node 29 can choose the next hop from  $\{7, 33, 5, 19, 23, 9, 12, 10, 2\}$ . Thus, path  $29 \rightarrow 5 \rightarrow 25 \rightarrow 1 \rightarrow 31$  is another path that can be used for control messages from Controller 1 to SFD 31. However, because of the constraints of latency, not all available paths can be used for

control message forwarding. The set of all latency-constrained paths between the controllers and the SFDs is denoted by  $A = \{p | d_p < D\}$ , where  $d_p$  is the latency of path  $p$  and  $D$  is the maximum latency constraint of control messages.

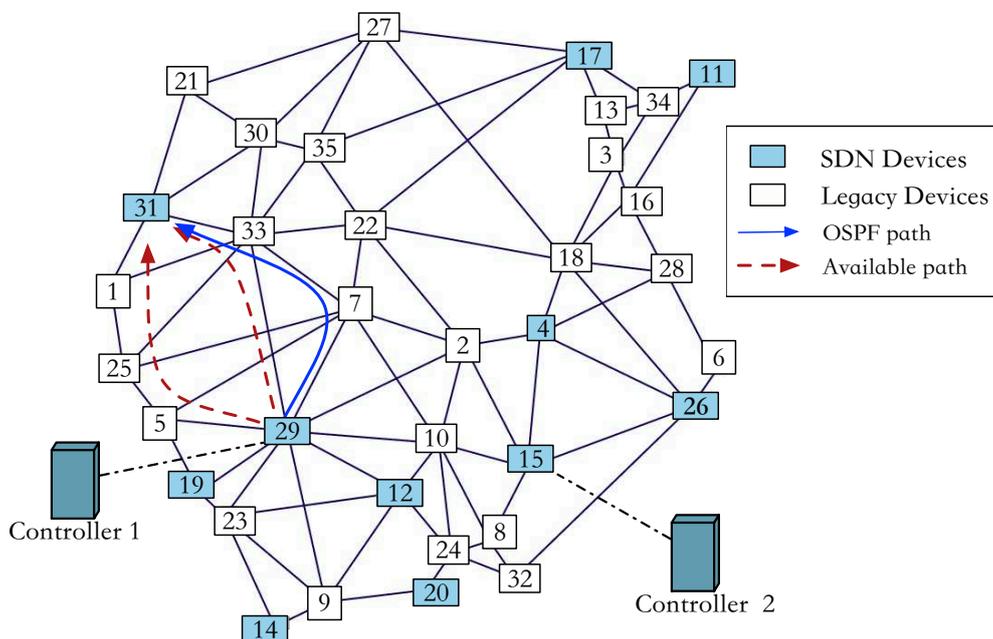


Figure 1. A hybrid SDN network with two controllers. SDN: Software-defined networking.

In the hybrid SDN WANs, each controller manages a portion of the SFDs and creates a control tree rooted at each controller, as shown in Figure 2. It is a control tree for Figure 1 when two controllers are placed at SFD 29 and 15. Seven SFDs  $\{4, 12, 14, 19, 20, 29, 31\}$  are under the control of Controller 1, and four SFDs  $\{11, 15, 17, 26\}$  are under the control of Controller 2. Besides, the chosen paths for SFD-to-controller communication are also shown in the control tree. For example, SFD 11 uses path  $11 - 16 - 18 - 26 - 15$  to communicate with its controller. Controllers 1 and 2 use the path  $29 - 2 - 15$  to communicate with each other.

The process of creating the control tree consists of four phases. Firstly, the controllers announce their presence. Secondly, they choose the control paths that are used for the control messages forwarding from all the latency-constrained available paths. Thirdly, based on the selected control messages forwarding paths, the capacity of controllers and the PACKET\_IN load of SFDs, the matches between controllers and SFDs are decided. The assignment of SFDs to controllers is denoted by:

$$\tau = \{\pi_{uc} | u \in S, c \in C\}, \tag{1}$$

where  $\pi_{uc}$  is a binary that denotes whether SFD  $u$  is assigned to controller  $c$  or not. Finally, the control paths are established between the SFDs and its assigned controller.

The latency of control messages is an important performance indicator of a hybrid SDN WAN. It describes the time taken for the control messages to travel across the network between the controller and the SFDs. The control message latency consists of processing latency, transmission latency, propagation latency, and queuing latency. The processing latency is the time taken by routers to process the header of the control messages which can be ignored in WANs. The propagation latency is the time taken for the control messages to be forwarded over a path which is one of the most important latencies in the WANs. The queuing latency is the time taken for the control messages to be queued in the routing queues. In this scenario, the control messages are transmitted among the data packets that form the actual payload on the same physical channel. The queuing latency in LFD and SFD is different. If the arrival rate of packets is faster than the process rate of the LFD, they are put into

a First-In-First-Out (FIFO) queue. The queuing latency that control messages experience in the queue is increased. The dropping rate of control messages is also increased if the LFD is congested. On the other hand, at the SFDs, the control messages can be forwarded with high priority without queuing with data packets. The size of control message is relatively small compared to that of the data packets. If the PACKET\_IN message is sent by an SFD to seek how to route a flow, it is encapsulated with the first 128 B of the data packet. Thus, the queuing time of control messages in SFDs can be ignored.

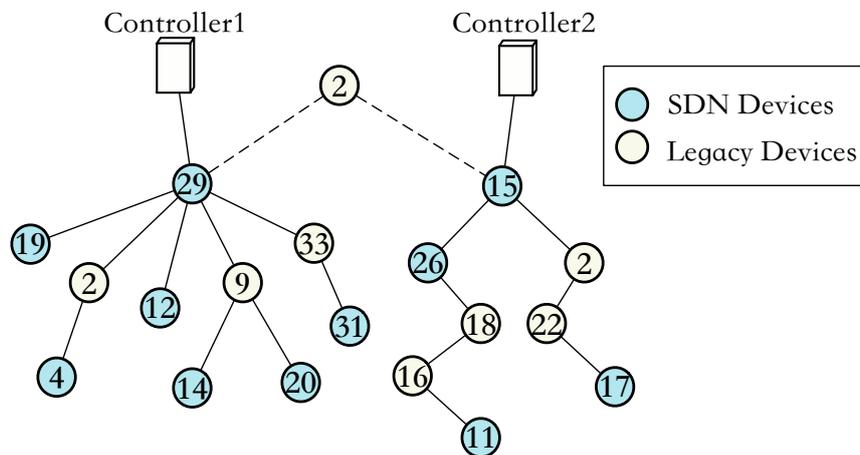


Figure 2. The control tree of Figure 1.

### 3.2. Problem Formulation

In this subsection, the controller assignment problem in hybrid SDNs is formulated. According to the problem description and analysis above, SFD-to-controller latency between SFDs and controllers in hybrid SDNs can be simplified to include two main parts: (1) the propagation latency on paths; and (2) queuing latency of the LFDs.

For the first part of SFD-to-controller latency in hybrid SDNs, the propagation latency of one available path  $\alpha$  can be defined as follows:

$$T_\alpha = \sum_{e \in \alpha} \eta_e d_e, \forall \alpha \in A, \tag{2}$$

where  $\eta_e$  is propagation latency of one available path  $e$  per kilometer and  $d_e$  is the distance of link  $e$ . The set of all latency-constrained available paths between the controllers and the SFDs is denoted by  $A$ .

For the second part of SFD-to-controller latency in hybrid SDNs, we firstly assume that the message arrival of LFDs is a Poisson process. It has an exponential distribution of the service time in the M/M/1 queue, and it has a parameter  $\mu$  where  $\frac{1}{\mu}$  is the mean service time. Given the message processing rate of an LFD is  $\lambda_v$ , its queuing latency can be denoted by:

$$Q_v = \frac{\rho_v}{\mu_v(1 - \rho_v)} - \frac{m_v \rho_v^{m_v}}{\mu_v(1 - \rho_v^{m_v})}, \forall v \in LN, \tag{3}$$

where  $\rho_v = \frac{\lambda_v}{\mu_v}$  and  $m$  is the estimated queue length.

Thus, if the control messages are successfully transmitted on path  $\alpha$ , the SFD-to-controller latency can be defined as:

$$R_\alpha = T_\alpha + \sum_{v \in LN} Q_v, \forall \alpha \in A. \tag{4}$$

However, when the capacity of LFDs is much less than the traffic load, some messages may be dropped because of the limited length of the queue in LFDs. The messages loss rate of an LFDs is denoted by:

$$\delta_v = \frac{1 - \rho_v}{1 - \rho_v^{m+1}} \rho_v^m, \forall v \in LN, \tag{5}$$

where  $\rho$  and  $m$  have the same definition with definition in (3).

If the control packages are dropped because of LFDs, they need to be re-sent from the source nodes. As usual, there should be time limitation, and the control messages would resend from the source nodes after this time. In this model, we only considered one-chance re-sending, because repeated re-sending would waste a large number of resources. In this case, more controllers or more SFDs should be considered to be added to the network to improve the situation of networks. Thus, considering the re-transmission of the dropped messages, the SFD-to-controller latency of one path  $\alpha$  can be defined as:

$$L_\alpha = R_\alpha + F(1 - \prod_{v \in LN} (1 - \delta_v)), \forall \alpha \in A, \tag{6}$$

where  $F$  is the maximum tolerant time. From this formulation, we can see that the loss rate of control messages in the LFDs is a very important metric to measure the performance of the controller assignment.

For in a hybrid SDN, there exist multi-paths for control messages forwarding. It would be wise to choose the path with minimum SFD-to-controller latency. The minimum latency between an SFD and a potential controller is given by the following formula:

$$\omega(u, c) = \min_{\alpha \in A_{uc}} L_\alpha, \forall u \in S, c \in C, \tag{7}$$

where  $A_{uc}$  is the set of latency-constrained available paths between SFD  $u$  and controller  $c$ .

The objective is to find a good match between the SFDs and the controllers to minimum the average SFD-controller latency:

$$\min_{\pi} \frac{1}{|S|} \sum_{u \in S} \sum_{c \in C} \omega(u, c) \pi_{uc}, \tag{8}$$

where the variable  $\pi_{uc}$  denoted whether the SDN device  $u$  is controlled by controller  $c$  or not.

For the controller assignment problem in hybrid SDNs, it should meet some constraints:

$$\sum_{u \in S} l(u) \pi_{uc} \leq B_c, \forall c \in C \tag{9}$$

$$\sum_{c \in C} \pi_{uc} = 1, \forall u \in S \tag{10}$$

$$\pi_{uc} \in \{0, 1\}, \forall u \in S, c \in C \tag{11}$$

where  $l(u)$  is the load of SFD  $u$  and  $B_c$  is the capacity of controller  $c$ . The constraint in Equation (9) means that the capacity of each controller should not be exceeded. The constraint in Equation (10) means that one SDN device is managed by exactly one controller. The constraint in Equation (11) means that variables  $\pi_{uc}$  are boolean.

#### 4. Evaluation

In this section, we first introduce the experimental environment and design. Then, extensive simulations are presented to evaluate the performance of the proposed approaches compared with the pure SDN controller assignment scheme.

#### 4.1. Experimental Design

As shown in Figure 3, topology TA2 from Survivable fixed telecommunication Network Design library (SNDlib) [32] was used for the performance evaluation. TA2 has 66 nodes and 108 links. Considering the convenience of control, the controllers were assumed to be connected with SFDs. As discussed in Section 3.1, in hybrid SDNs, there are some available paths, which can be used for routing control messages. The more paths that can be used in routing, the more chances there are to maintain stability and recover from link outage. Thus, in this experiment, the location of controllers was chosen greedily with the maximum number of paths between controllers and SFDs. About 30% of forwarding devices were selected randomly to be deployed with the SDN capability. The links between devices had the same transmission speed  $k_e = 5 \mu\text{s}/\text{km}$ . We assumed that LFDs had the same configuration with the service rate being  $\mu = 5 \text{ M packets}/\text{ms}$  and the estimated queue length being  $m = 40 \text{ M packets}$ . The solution of the dynamic assignment was computed on Intel Core i7 (2.9 GHz) with 16 GB memory. The Gurobi Optimizer, which is a programming solver, was chosen to solve this optimization problem in this work.

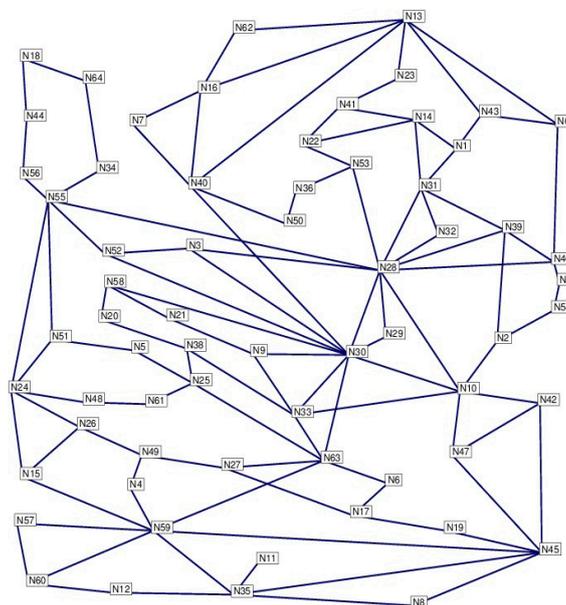


Figure 3. Topology of TA2 network.

The assignment scheme Legacy Ignored Assignment (LIA) [15] was used for comparison. It ignores the influence of the LFDs to the latency of control message. In this section, we focus on two metrics: the control message latency and the loss rate caused by the LFDs.

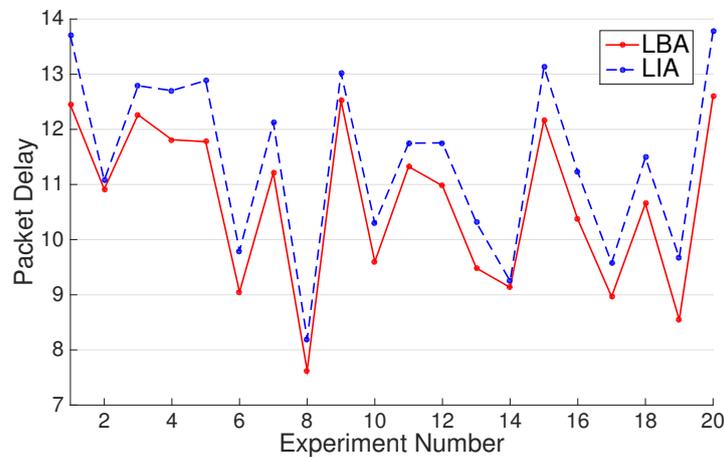
#### 4.2. Experimental Results

In this subsection, the results of the average SFD-to-controller latency and the loss rate of the messages from the SFDs to their assigned controllers are shown with different traffic load, number of controllers and SDN deployment ratio.

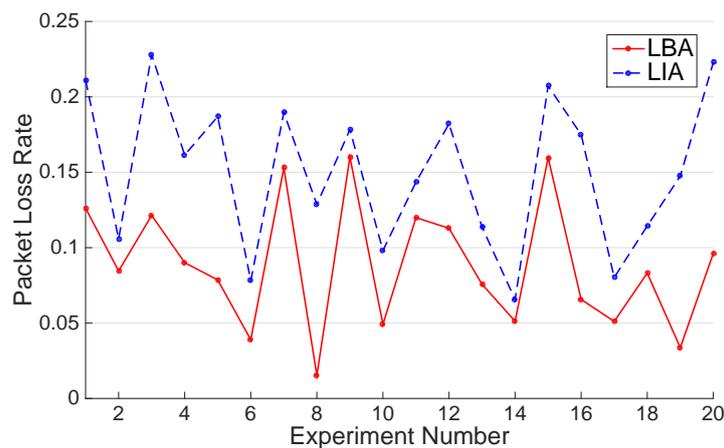
Figures 4 and 5 show the performance of 20 experiments that had different and random loads (arrive rates) of LFDs. Three controllers were deployed in this network. The characteristics of the experiment are shown in Table 2. The arrive rate  $\lambda$  of all LFDs was assumed to follow a normal distribution. The mean of the normal distribution was randomly selected from 4 to 5 M packets/ms and the standard deviation was from 0.5 to 1.5. Compared with the LIA, the proposed LBA had lower average SFD-to-controller latency and message loss rate in all experiments.

**Table 2.** Characteristics of experiments.

Link transmission speed	5 $\mu$ s/km	LFD service rate	5 M packets/ms
Queue length of LFD	40 M packets	Mean arrive rate of LFD	4–5 M packets/ms
Standard deviation of arrive rate	0.5–1.5		



**Figure 4.** Average SFD-to-controller latency with different traffic. SFD: SDN-enabled Forwarding Device. LBA: Legacy Based Assignment. LIA: Legacy Ignored Assignment.



**Figure 5.** Loss rate with different traffic.

Figures 6 and 7 show average latency and average message loss rate with 2–7 SDN controllers. For each case (number of controllers), we used 20 sets of random traffic, whose load of each switch was generated from the normal distribution with mean parameter 10 M packets/ms and standard deviation parameter 1.5. These two figures show the 95% confidence interval. In these cases, 20% of switches were assumed to be SDN randomly. Firstly, we can see that the LBA had and obviously lower average latency and message loss rate relative to the LIB. Secondly, in Figure 6, we can easily see that, with more SDN controllers deployed in the network, lower average latency can be obtained because deploying more controllers offers more choices to SFDs and the more chances to reduce the average SFD-to-controller latency. Besides, Figure 7 shows that more controllers deployed in hybrid SDNs have an overall reduction trend, but not for all cases. For example, when three controllers are deployed, the average packet loss rate is slightly increased relative to that when two controllers are deployed. The reason is that our controller assignment scheme and path scheduler consider the packet loss rate, but it is not the only factor in this problem. The objective of our model is to minimize the average SFD-to-controller latency. Thus, there are chances to choose paths with low latency, but with slightly higher packet loss rate.

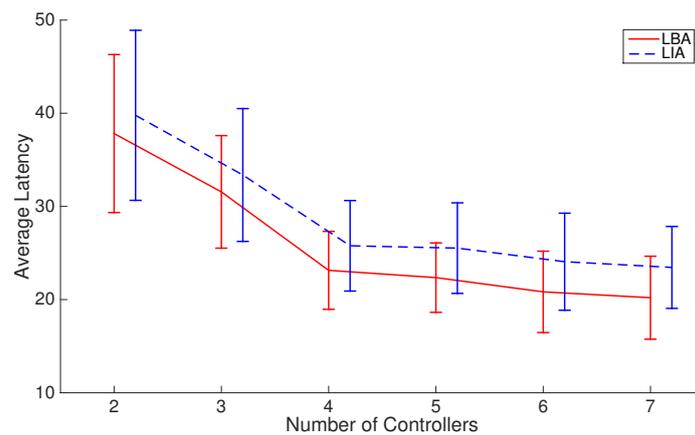


Figure 6. Average SFD-to-controller latency with different number of controllers.

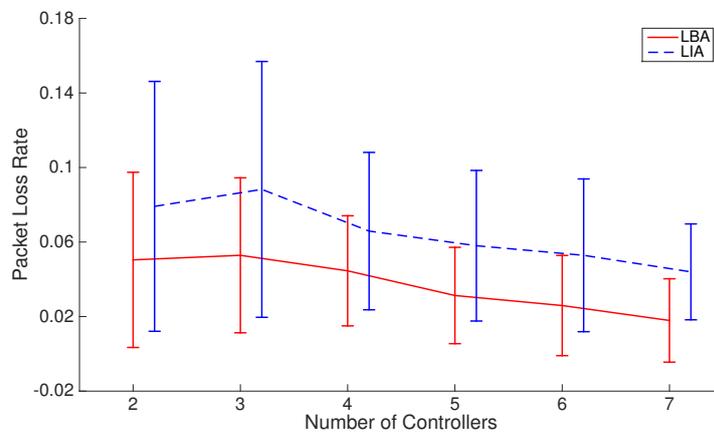


Figure 7. Loss rate with different number of controllers.

Figure 8 shows the average SFD-to-controller latency improvement compared with LIA. In this figure, two different topologies are used: the TA2 and the INDIA35. The traffic and deployment ratio are the same as Figure 6. The topology of INDIA35 is shown in Figure 1, which has 35 nodes and 80 links. On the one hand, in this figure, these two broken lines are the average latency improvement percentage compared to LBA and LIA of INDIA35 and TA2. The average improvement is from 4.5% to 14% for INDIA35, and from 2.8% to 9% for TA2. On the one hand, the box plot is a method for graphically depicting numerical dataset through their quartiles. In this figure, the box plots show latency improvement percentage data compared LBA and LIA of INDIA35 and TA2. This figure shows that, with more controllers deployed, the improvement is in an increasing trend overall. As INDIA35 has fewer nodes than TA2, with the same number of controller deployment, the latency improvement of INDIA35 is more obvious than TA2.

Figures 9 and 10 show average latency and packet loss rate with 20%, 40%, 60% and 80% SDN devices deployment ratios. The LBA has an obviously lower average SFD-to-controller latency and packet loss rate relative to the LIB in the same SDN deploy ratio. When the number of SFDs is the same as the number of controllers, the latency and packet loss rate decrease to zero for directly linked. Deploying more devices deployed with SDN allows more paths to be used to forward control messages, and thus reduces the chance they pass through the LFDs. Hence, a lower average latency and packet loss rate would be obtained. With more controllers deployed in the network, the curves in Figure 9 show a decreasing trend. For the same reason as Figure 7, some curves in Figure 10 have ups and downs. However, LBA still has lower packet drop rate relative to LIB.

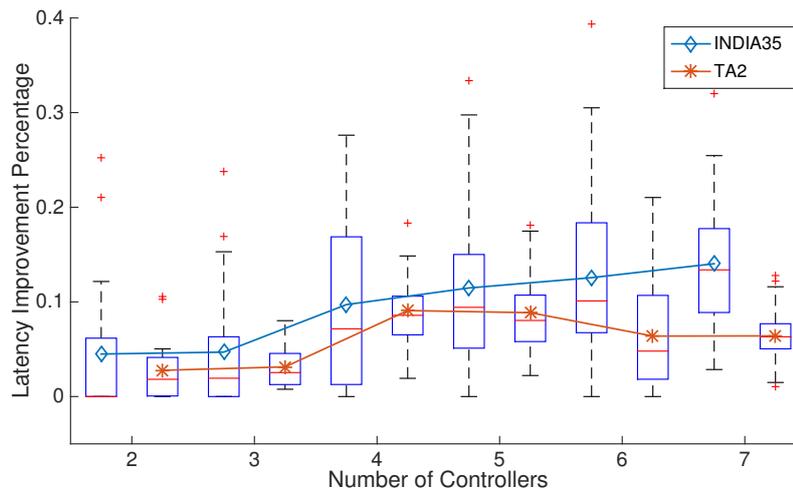


Figure 8. Average SFD-to-controller latency improvement compared with LIA in two different topologies.

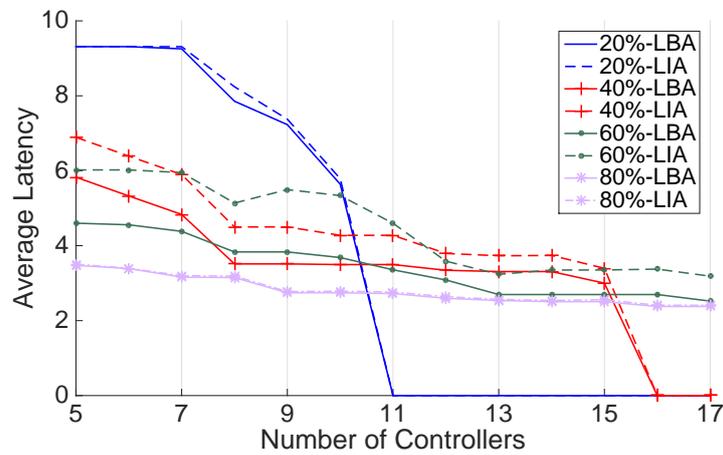


Figure 9. Average SFD-to-controller latency with different SDN deployment ratio.

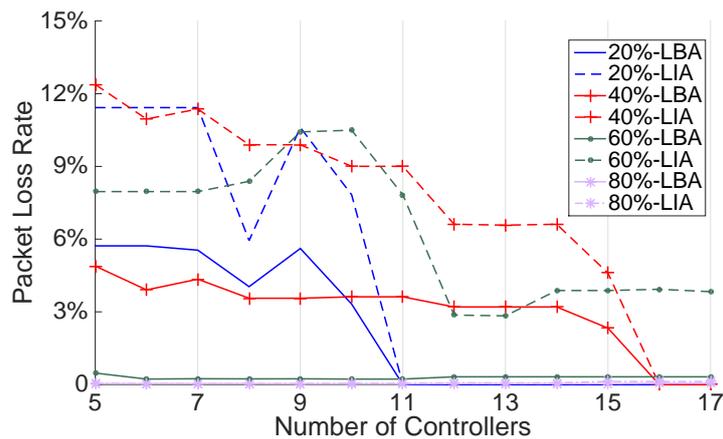


Figure 10. Loss rate with different SDN deployment ratio.

In conclusion, to achieve an optimal controller assignment with the lowest average SFD-to-control latency, the LFDs in the hybrid SDNs cannot be ignored. The performance evaluation shows that the proposed scheme has a better performance for both average SFD-to-control latency and message loss rate. It is obvious that, with more SDN controllers deployed in the network, a lower average latency can be obtained.

## 5. Conclusions

Hybrid SDN is an inevitable phase of the SDN transition, which consists of both SDN devices and legacy devices. A hybrid SDN WAN is usually partitioned into several domains by multiple in-band controllers, which may be located at different places. The dynamic SDN controller assignment in hybrid SDN WANs is an important problem, in which this influence of legacy devices has been ignored by previous studies. In hybrid SDNs, control messages may pass through LFDs, which would increase the SFD-controller latency because of queuing in LFDs.

In this paper, we propose a dynamic controller assignment scheme in hybrid SDNs, which is called the Legacy Based Assignment (LBA). This scheme can dynamically delegate each controller with a subset of SFDs in the hybrid SDNs, whose objective is to minimize average SFD-to-control latency. Firstly, we analyze the influence of LFDs of hybrid SDNs in SFD-to-control latency, which is caused by routing queuing in LFDs. Secondly, we formulate the dynamic controller assignment problem to be a 0–1 Integer Linear Programming. In the formulation, in addition to the impact of LFDs, the capacity limitation of the controllers and the available routing paths generated by the SDN partially deployment are also considered. Simulation results show that our solution has a better performance in decreasing the latency and packet loss rate, and we validate that the influence of LFDs cannot be ignored in solving this problem.

**Author Contributions:** Conceptualization, J.Y., X.L., T.Y., and E.L.; methodology, J.Y., X.L., T.Y., and E.L.; software, J.Y.; validation, J.Y., X.L., T.Y., E.L., C.K., and F.C.; formal analysis, J.Y., X.L., and T.Y.; investigation, J.Y., X.L., T.Y., and E.L.; resources, J.Y., X.L., T.Y., and E.L.; data curation, J.Y., X.L., T.Y., and E.L.; writing—original draft preparation, J.Y., X.L., T.Y., and E.L.; writing—review and editing, J.Y., X.L., and T.Y.; visualization, J.Y.; supervision, X.L.; project administration, C.K., F.C.; funding acquisition, X.L., E.L.

**Funding:** This work was supported in part by the project “Research on Security Architecture for Next Generation Distribution Automation System” of State Grid Corporation of China, and by the NSFC-General Technology Fundamental Research Joint Fund U1836215.

**Acknowledgments:** The authors would like to convey their heartfelt gratefulness to the reviewers and the editor for the valuable suggestions and important comments which greatly helped them to improve the presentation of this manuscript.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Nunes, B.A.A.; Mendonca, M.; Nguyen, X.N.; Obraczka, K.; Turletti, T. A survey of software-defined networking: Past, present, and future of programmable networks. *IEEE Commun. Surv. Tutor.* **2014**, *16*, 1617–1634. [[CrossRef](#)]
2. Farhady, H.; Lee, H.; Nakao, A. Software-defined networking: A survey. *Comput. Netw.* **2015**, *81*, 79–95. [[CrossRef](#)]
3. Hong, D.K.; Ma, Y.; Banerjee, S.; Mao, Z.M. Incremental Deployment of SDN in Hybrid Enterprise and ISP Networks. In Proceedings of the Symposium on SDN Research, Santa Clara, CA, USA, 14–15 March 2016; ACM: New York, NY, USA, 2016; p. 1.
4. Vissicchio, S.; Vanbever, L.; Bonaventure, O. Opportunities and research challenges of hybrid software defined networks. *ACM SIGCOMM Comput. Commun. Rev.* **2014**, *44*, 70–75. [[CrossRef](#)]
5. Caria, M.; Das, T.; Jukan, A.; Hoffmann, M. Divide and conquer: Partitioning OSPF networks with SDN. In Proceedings of the 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM), Ottawa, ON, Canada, 11–15 May 2015; pp. 467–474.
6. Salman, O.; Elhaji, I.H.; Kayssi, A.; Chehab, A. SDN controllers: A comparative study. In Proceedings of the IEEE 2016 18th Mediterranean Electrotechnical Conference (MELECON), Limassol, Cyprus, 18–20 April 2016; pp. 1–6.
7. Karakus, M.; Durresi, A. A survey: Control plane scalability issues and approaches in software-defined networking (SDN). *Comput. Netw.* **2017**, *112*, 279–293. [[CrossRef](#)]
8. Bannour, F.; Souihi, S.; Mellouk, A. Distributed SDN control: Survey, taxonomy, and challenges. *IEEE Commun. Surv. Tutor.* **2018**, *20*, 333–354. [[CrossRef](#)]

9. Hu, T.; Guo, Z.; Yi, P.; Baker, T.; Lan, J. Multi-controller based software-defined networking: A survey. *IEEE Access* **2018**, *6*, 15980–15996. [[CrossRef](#)]
10. Koponen, T.; Casado, M.; Gude, N.; Stribling, J.; Poutievski, L.; Zhu, M.; Ramanathan, R.; Iwata, Y.; Inoue, H.; Hama, T.; et al. Onix: A distributed control platform for large-scale production networks. *OSDI* **2010**, *10*, 1–6.
11. Wang, G.; Zhao, Y.; Huang, J.; Wang, W. The controller placement problem in software defined networking: A survey. *IEEE Netw.* **2017**, *31*, 21–27. [[CrossRef](#)]
12. Schiff, L.; Schmid, S.; Kuznetsov, P. In-band synchronization for distributed SDN control planes. *ACM SIGCOMM Comput. Commun. Rev.* **2016**, *46*, 37–43. [[CrossRef](#)]
13. Oktian, Y.E.; Lee, S.; Lee, H.; Lam, J. Distributed SDN controller system: A survey on design choice. *Comput. Netw.* **2017**, *121*, 100–111. [[CrossRef](#)]
14. Yao, G.; Bi, J.; Li, Y.; Guo, L. On the capacitated controller placement problem in software defined networks. *IEEE Commun. Lett.* **2014**, *18*, 1339–1342. [[CrossRef](#)]
15. Heller, B.; Sherwood, R.; McKeown, N. The controller placement problem. In Proceedings of the First Workshop on Hot Topics in Software Defined Networks, Helsinki, Finland, 13 August 2012; pp. 7–12.
16. Xiao, P.; Qu, W.; Qi, H.; Li, Z.; Xu, Y. The SDN controller placement problem for WAN. In Proceedings of the 2014 IEEE/CIC International Conference on Communications in China (ICCC), Shanghai, China, 13 October 2014; pp. 220–224.
17. Jimenez, Y.; Cervello-Pastor, C.; Garcia, A.J. On the controller placement for designing a distributed SDN control layer. In Proceedings of the 2014 IFIP Networking Conference, Trondheim, Norway, 2–4 June 2014; pp. 1–9.
18. Liu, B.; Wang, B.; Xi, X. Heuristics for sdn controller deployment using community detection algorithm. In Proceedings of the 2016 7th IEEE International Conference on Software Engineering and Service Science (ICSESS), Beijing, China, 26–28 August 2016; pp. 253–258.
19. Lange, S.; Gebert, S.; Zinner, T.; Tran-Gia, P.; Hock, D.; Jarschel, M.; Hoffmann, M. Heuristic approaches to the controller placement problem in large scale SDN networks. *IEEE Trans. Netw. Serv. Manag.* **2015**, *12*, 4–17. [[CrossRef](#)]
20. Dixit, A.; Hao, F.; Mukherjee, S.; Lakshman, T.; Kompella, R. Towards an elastic distributed SDN controller. *ACM SIGCOMM Comput. Commun. Rev.* **2013**, *43*, 7–12. [[CrossRef](#)]
21. Wang, T.; Liu, F.; Guo, J.; Xu, H. Dynamic SDN controller assignment in data center networks: Stable matching with transfers. In Proceedings of the IEEE INFOCOM 2016 35th Annual IEEE International Conference on Computer Communications, San Francisco, CA, USA, 10–15 April 2016; pp. 1–9.
22. Mostafaei, H.; Menth, M.; Obaidat, M.S. A Learning Automaton-Based Controller Placement Algorithm for Software-Defined Networks. In Proceedings of the 2018 IEEE Global Communications Conference (GLOBECOM), Abu Dhabi, UAE, 9–13 December 2018; pp. 1–6.
23. Wang, T.; Liu, F.; Xu, H. An efficient online algorithm for dynamic SDN controller assignment in data center networks. *IEEE/ACM Trans. Netw.* **2017**, *25*, 2788–2801. [[CrossRef](#)]
24. Dixit, A.; Hao, F.; Mukherjee, S.; Lakshman, T.; Kompella, R.R. ElastiCon; an elastic distributed SDN controller. In Proceedings of the 2014 ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS), Marina del Rey, CA, USA, 20–21 October 2014; pp. 17–27.
25. Ruiz-Rivera, A.; Chin, K.W.; Soh, S. GreCo: An energy aware controller association algorithm for software defined networks. *IEEE Commun. Lett.* **2015**, *19*, 541–544. [[CrossRef](#)]
26. Savas, S.S.; Tornatore, M.; Dikbiyik, F.; Yayimli, A.; Martel, C.U.; Mukherjee, B. RASCAR: Recovery-aware switch-controller assignment and routing in SDN. *IEEE Trans. Netw. Serv. Manag.* **2018**, *15*, 1222–1234. [[CrossRef](#)]
27. Suh, D.; Pack, S. Low-complexity master controller assignment in distributed SDN controller environments. *IEEE Commun. Lett.* **2017**, *22*, 490–493. [[CrossRef](#)]
28. Xu, H.; Li, X.Y.; Huang, L.; Deng, H.; Huang, H.; Wang, H. Incremental deployment and throughput maximization routing for a hybrid SDN. *IEEE/ACM Trans. Netw. (TON)* **2017**, *25*, 1861–1875. [[CrossRef](#)]
29. Huang, X.; Cheng, S.; Cao, K.; Cong, P.; Wei, T.; Hu, S. A survey of deployment solutions and optimization strategies for hybrid SDN networks. *IEEE Commun. Surv. Tutor.* **2018**, *21*, 1483–1507. [[CrossRef](#)]

30. Das, T.; Gurusamy, M. Resilient Controller Placement in Hybrid SDN/Legacy Networks. In Proceedings of the 2018 IEEE Global Communications Conference (GLOBECOM), Abu Dhabi, UAE, 9–13 December 2018; pp. 1–7.
31. He, K.; Khalid, J.; Gember-Jacobson, A.; Das, S.; Prakash, C.; Akella, A.; Li, L.E.; Thottan, M. Measuring control plane latency in sdn-enabled switches. In Proceedings of the 1st ACM SIGCOMM Symposium on Software Defined Networking Research, Santa Clara, CA, USA, 17–18 June 2015; p. 25.
32. SNDlib Library. Available online: <http://sndlib.zib.de> (accessed on 21 April 2019).



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).