Dynamic Wavelength Allocation and Analytical Model for Flow Assignment in Optical Packet and Path Integrated Networks

Onur Alparslan, Shin'ichi Arakawa, and Masayuki Murata

***Abstract*—A hybrid optical architecture combining path (circuit) and packet switching can be a good candidate for future optical networks because it exploits the best of both worlds. In this paper, we present a control framework called the Dynamic Optical Wavelength and Flow Allocation Framework (DOWFAF), which can dynamically change the ratio of path and packet wavelengths and the flow size threshold in the hybrid path–packet integrated networks in order to balance the utilization of path and packet subnetworks and maximize the ratio of large flows benefiting from the path switching. We propose an analytical model for calculating the flow size threshold and a feedback control for estimating the wavelength allocation ratio for varying traffic. DOWFAF can be implemented by software-defined networking, which is getting a lot of attention recently. We show that DOWFAF can greatly increase the goodput of large Transmission Control Protocol (TCP) flows for a wide range of traffic, while decreasing the cost and the power consumption.**

***Index Terms*—Blocking probability; Circuit switching; Optical network; Packet switching; Path switching; Wavelength-division multiplexing.**

I. Introduction

Wavelength-division multiplexing (WDM) is a promising solution to the high capacity, energy efficiency, cost, and quality-of-service (QoS) requirements of fast-growing Internet traffic. WDM provides different switching granularities, e.g., packet, burst, and path (circuit) switching, each satisfying these requirements to some degree. While optical packet switching benefits from high multiplexing gain and flexibility, it has disadvantages like high cost and high power consumption, as it needs fast switching fabric and fast header processing capability to achieve high granularity. However, its cost and power consumption is much higher than a path switch. Moreover, the current optical buffering technology is not mature enough to provide large buffering space, which may jeopardize the goodput (speed) of Transmission Control Protocol (TCP) flows because of the high packet drop rate in small buffers. In comparison, path switching has many advantages over packet switching, like low switch cost and power consumption, as its switching speed and frequency are lower. Moreover, it does not need optical buffering in the core nodes, as there is no contention of packets, so the flows can achieve higher goodput. Furthermore, it has an easier and more effective QoS support for flows with strict QoS requirements. However, establishing a path for each flow is not a good stand-alone solution for Internet traffic, because a short flow may not fully utilize the capacity in the dedicated channel due to TCP dynamics. Moreover, path switching needs prior reservation of channels, which adds an additional delay to the flow completion time and decreases the achievable utilization.

Recently, some hybrid architectures combining path and packet switching were proposed as a solution to these problems by exploiting the best of both worlds [1,2]. There are two main approaches in the literature for realizing a hybrid path and packet switching architecture. One of them is carrying both packet and path traffic on the same wavelength at the same time like in ORION [3] and OpMiGua [4], where all wavelengths are principally used by paths. The packet traffic is inserted into idle periods left from the path traffic on a wavelength. Another method is to use separate channels for path and packet switching and distribute the traffic between them. For example, the network can carry short flows over a packet switching subnetwork, while carrying the large flows on an independent path switching subnetwork as in CHEETAH [5]. The CHEETAH project proposes two different methods to provide its service. In the first method, end hosts must be equipped with second Ethernet network interface cards that are connected directly to a synchronous optical network based CHEETAH network. End hosts directly initiate end-to-end circuits. The authors state that this approach of creating dedicated virtual circuits between end servers located in enterprises and the closest core circuit/virtual circuit network switch is only feasible for small numbers of users, so they proposed one more method, which is a proxy-based internetworking architecture [6]. In this architecture, they define a system called a circuit-aware application gateway running modified Squid software to

Manuscript received June 27, 2016; revised January 20, 2017; accepted January 31, 2017; published March 31, 2017 (Doc. ID 268929).

The authors are with the Graduate School of Information Science and Technology, Osaka University, 1-5 Yamadaoka, Suita, Osaka 565-0871, Japan (e-mail: a-onur@ist.osaka-u.ac.jp).

https://doi.org/10.1364/JOCN.9.000304

interconnect connectionless IP-routed segments at the edges (regional/enterprise subnetworks) with two subnetworks consisting of a circuit switching subnetwork and an IP-routed subnetwork. A long-haul TCP connection is split into three concatenated connections. Gateways must run a complex Squid proxy server, cache data with disk buffering, and employ a special version of TCP to transport data between gateways in the optical network for each flow. However, considering that fiber speeds of over 1 Tbit/s are already achieved in commercial-grade hardware in a real-world environment, the processing speed may not be enough to employ as many functionalities at the gateways as are necessary for CHEETAH. Another hybrid architecture that uses separate channels for path and packet switching is OPCInet [7]. OPCInet uses waveband switching, where a waveband carries N wavelengths, so the number of ports required for packet switching fabric is decreased. When an optical path is established and the number of reserved path switching wavelengths in the border waveband reaches $N - 1$, a set of N packet switching wavelengths are converted to path switching. If all wavelengths in the path switching border waveband are unused and the number of used wavelengths in the previous path switching waveband is less than $N - 1$, the unused path switching border waveband is converted to packet switching [8]. The authors implemented a messaging framework for switch configuration based on OpenFlow [7,9]. However, the authors do not propose a control framework to decide which flows will be carried over the path subnetwork.

In this paper, we propose a control framework for an integrated hybrid path and packet switching network. We call it the Dynamic Optical Wavelength and Flow Allocation Framework (DOWFAF), as it can dynamically change the ratio of path and packet wavelengths. DOWFAF makes use of an optical path switching fabric for large flows, while carrying the short flows on an optical packet switching fabric. DOWFAF maximizes the ratio of large flows benefiting from dedicated path switching wavelengths by dynamically changing the ratio of path and packet wavelengths and the flow size threshold for using the path wavelengths according to the network conditions. Its objectives are the following:

1) increase the goodput of large flows by carrying them on dedicated path switching wavelengths;
2) not discriminate the short flows on packet switching wavelengths;
3) decrease the power consumption by carrying the large flows on a path switching fabric instead of a packet switching fabric, which consumes much more power;
4) decrease the manufacturing cost via decreasing the size of the packet switching fabric by using only path switching fabric for some wavelengths.

DOWFAF differs from the other works in the literature in the following aspects:

- The same fiber is used for both packet and path switching by allocating separate wavelengths for each switching paradigm, so it is completely different from ORION and OpMiGua, which use the same wavelength for both packet and path switching at the same time. Basically both ORION and OpMiGua use a path switching architecture, but they inject extra packets into the voids in path switching.
- OPCInet proposes only a switch hardware architecture and a messaging framework for a packet and circuit integrated switch. The authors do not specify either how to select the flows that will be carried over the path wavelengths or how to optimize the ratio of path and packet switching wavelengths. On the other hand, we propose an analytical model for calculating the flow size threshold for varying traffic that also optimizes the ratio of path and packet switching wavelengths. We do not propose a specific switch hardware architecture. DOWFAF can be applied to any hybrid switch architectures in the literature including OPCInet as long as they support dynamically changing the ratio of path and packet switching wavelengths.
- The optical network changes the ratio of path and packet switching wavelengths dynamically, so the network can adapt itself to the current traffic in order to maximize the ratio of large flows benefiting from path switching. On the other hand, CHEETAH uses fixed subnetworks for path switching and packet switching. Moreover, CHEETAH does not specify how to choose the capacity of these two subnetworks.
- The large flows are carried over path switching like CHEETAH, but we propose an analytical model for estimating the flow size threshold for selecting the large flows. CHEETAH's analytical model for estimating the flow size threshold is based on the analysis of TCP Reno's additive increase/multiplicative decrease congestion control [10]. However, recent operating systems are using much more aggressive TCP variants as the default, where the TCP analytical model in CHEETAH is no longer applicable. Unlike CHEETAH, our analytical model does not employ any equations specific to a TCP variant, so it is generic and also applicable in the future regardless of changes in TCP congestion control.
- The core network in DOWFAF is transparent to the end hosts (users). The only requirement is the application support, as the applications in the end hosts may request using the path subnetwork when sending their flows. If DOWFAF is available in the network, DOWFAF decides the subnetwork for the flow by using the flow size information.

This paper evaluates the performance of the proposed control framework for DOWFAF by a simulation study using CUBIC TCP [11], which has been the default TCP variant in Linux kernels since version 2.6.19. As CHEETAH does not specify the ratio of path and packet switching wavelengths and is based on an analytical model of TCP Reno congestion control, which is no longer the default in Linux kernels, direct comparison of DOWFAF and CHEETAH in terms of analysis or simulation is not possible. Therefore, a simulation study was carried out to evaluate the performance of DOWFAF and to show its improvement over a pure packet switching network. The simulation results revealed that DOWFAF is capable of increasing the goodput of large flows and dynamically optimizing the ratio of path/packet wavelengths in order to maximize the number of large flows benefiting from the path wavelengths without penalizing the short flows on packet wavelengths, while decreasing the power consumption and manufacturing cost of the routers.

The paper is organized as follows. Section II presents the architecture of the path/packet integrated architecture and proposes a model dynamically changing the path/packet wavelength ratio. Section III shows the simulation results and discusses the performance of the architecture. Section IV concludes the paper.

II. Model of Path/Packet Integrated Network

This section describes the architecture in detail.

A. Network and Node Architecture

The DOWFAF network consists of a packet switching subnetwork and a path switching subnetwork where separate wavelengths are allocated for each, as shown in Fig. 1. DOWFAF dynamically changes the ratio of wavelengths assigned to the path and packet subnetworks according to the feedback from the edge routers. The path switching subnetwork is used for the DATA packets of only selected large flows, while the remaining traffic is carried over the packet switching subnetwork. A possible node architecture is shown in Fig. 2 that was used in the simulations of this paper. As an example, a node with three wavelengths in red, blue, and green is plotted. It does not make sense to reserve a path for short flows like flows with a single packet size, so there should be at least one packet switching wavelength that is only connected to the packet switching fabric (OPS-SW), as shown by the red wavelength in the figure. Moreover, User Datagram Protocol (UDP) and TCP ACK packets require a packet switching wavelength. The blue wavelength is connected to both the OPS-SW and the path switching fabric (OCS-SW), so it is possible to assign it to the packet switching subnetwork or the path switching subnetwork. When it is assigned to the packet switching subnetwork, the small 1 × 2 switch at the entrance forwards the packets on this wavelength to the OPS-SW. Otherwise the packets are sent to the OCS-SW. The packets in the packet switching wavelengths require header processing, so after their headers are sampled, they are delayed for a short time for header processing before they enter the OPS-SW. The packets on the path switching wavelengths do not need any header processing, which requires a high amount of CPU power and optical/electrical/optical (O/E/O) conversion of sampled headers. The green wavelength is reserved for the path switching subnetwork, so it is not connected to the OPS-SW, which decreases the size and cost of the required OPS-SW fabric compared to a pure packet switching node with the same number of wavelengths.

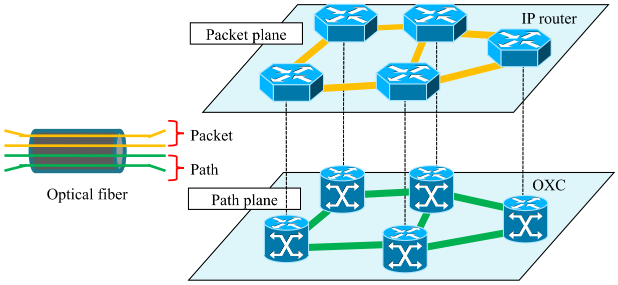


Fig. 1. Network architecture. The packet plane uses IP routers and the path plane uses optical cross connects (OXCs) for switching.

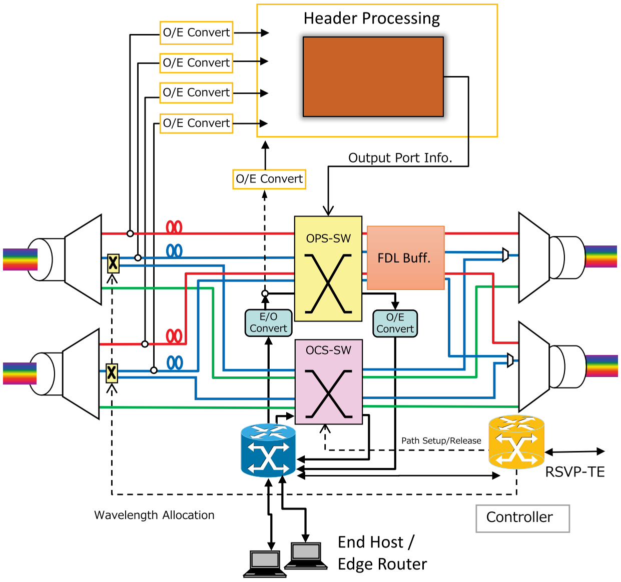


Fig. 2. Simulated node architecture.

Note that we do not propose a specific switch architecture for DOWFAF. DOWFAF can be applied to the hybrid switch architectures in the literature as long as they support dynamically changing the ratio of path and packet switching wavelengths. From a cost and power perspective, OPCInet may be a promising path–packet integrated architecture, because OPCInet minimizes the number of ports of the packet switching fabric by using waveband switching. However, waveband switching decreases the wavelength distribution granularity. In the simulations presented in this paper, we used the switch architecture shown in Fig. 2, which does not employ waveband switching, in order to carry out a more sensitive analysis of the wavelength allocation. While the cost of this node architecture is higher than OPCInet, as a larger optical switching fabric is required, it can be implemented by using an OSMOSIS [12] switch as the optical packet switching fabric. The OSMOSIS project demonstrated a 64 × 64 optical packet switch with a line rate of 40 Gbps per port and is scalable to 2048 ports by using multiple 64-port switch modules in a multi-stage fat tree topology.

B. Analytical Model

The primary objective of DOWFAF is improving the goodput of large flows, by forwarding them to the path subnetwork. As the full capacity of a path wavelength is dedicated to a single flow, the flows using the path subnetwork can get higher goodput than in the packet subnetwork. CHEETAH [5] proposes choosing the subnetwork for a new flow by estimating the goodput analytically in both subnetworks and choosing the one with the shortest transfer time. However, different variants of TCP using distinctively different congestion control algorithms are emerging, so it is difficult to propose a general analytical framework to estimate the goodput of TCP flows. CHEETAH's analytical model is based on TCP Reno's [10] congestion control. However, CUBIC TCP [11], which has been used by default in Linux kernels since version 2.6.19, applies a completely different congestion control in the congestion avoidance phase, so CHEETAH's analysis is no longer applicable. Instead of trying to estimate the transfer time of a TCP flow, we propose an analytical model that considers the utilization of wavelengths in order to calculate a size threshold for classifying a flow as large or short. In principle, we do not want to jeopardize the goodput of flows in the packet subnetwork when increasing the goodput of large flows. Therefore, we calculate the flow size threshold, which gives equal utilization at the problematic links in both packet and path subnetworks of DOWFAF and a pure packet switching network, while the utilization of other links is even lower in the packet subnetwork of DOWFAF than the path subnetwork and the pure packet switching network. Keeping the utilization of packet subnetworks equal to or lower than the path subnetwork is important. For example, let's say the traffic arrival rate to a fiber with eight wavelengths is 800 Mbytes/s. If all wavelengths are packet switching wavelengths and the traffic is distributed among wavelengths uniformly, the traffic per packet switching wavelength becomes 100 Mbytes/s. If we use an integrated architecture with four packet and four path switching wavelengths and forward only 200 Mbytes/s traffic (out of 800 Mbytes/s total traffic) to the path subnetwork, then the traffic per packet switching wavelength becomes 150 Mbytes/s, which causes higher utilization in the packet subnetwork compared to a pure packet switching network. As the utilization and the packet drop rate in the packet wavelengths increase, the goodput of flows in the packet wavelengths of the integrated architecture becomes worse than in the pure packet switching network, which we want to avoid.

In this paper, the flow size is assumed to have a bounded (truncated) Pareto distribution [13]. The bounded Pareto distribution possesses the heavy-tailed property of Internet traffic and has finite variance, so it is good for modeling the flow size distribution. The real flow size distribution in an operating network can also be used in the analysis by sampling the flow size distribution.

The bounded Pareto distribution has three parameters, α, k, and p, where α denotes the shape, k denotes the minimum value, and p denotes the maximum value. As given in Ref. [13], the mean of the bounded Pareto for $\alpha \neq 1$ is

$$E_{(k,p)} = \frac{k^{\alpha}}{1-(k/p)^{\alpha}} \cdot \frac{\alpha}{\alpha-1} \cdot \left(\frac{1}{k^{\alpha-1}} - \frac{1}{p^{\alpha-1}}\right). \tag{1}$$

Let the minimum and maximum flow sizes in the network be L and H, respectively. Then, the mean size of all flows in the network is $E_{(L,H)}$. Let t be the flow size threshold for carrying the flows over the path switching wavelengths. The flows shorter than t are sent to the packet switching wavelengths. The bounded Pareto distribution still holds for the size distribution of flows sent to both switching domains, so the mean size of flows sent to the packet switching wavelengths is $E_{(L,t)}$, while the mean size of flows sent to the path switching wavelengths is $E_{(t,H)}$.

First, we find the utilization of a pure packet switching network. TCP carries most of the traffic, so we estimate the traffic of TCP flows, which consists of DATA, ACK and TCP control packets (SYN and FIN).

1) DATA packets: Let λ be the arrival rate of TCP flows carrying DATA traffic in the forward direction, S_D be the average size of TCP DATA packets, C be the capacity of each wavelength, and W_{total} be the total number of wavelengths on a fiber. Calculating $C \cdot W_{\text{total}}$ gives the capacity of a fiber. As λ is the arrival rate of the flows carrying DATA packets, which has a mean size of $E_{(L,H)}$, calculating $\lambda \cdot E_{(L,H)}$ gives the incoming traffic rate of a fiber due to DATA packets. Dividing this by the fiber capacity $C \cdot W_{\text{total}}$ gives the utilization due to DATA packets, which is denoted by U_{all_D}, as

$$U_{\text{all}_D} = \frac{\lambda \cdot E_{(L,H)}}{C \cdot W_{\text{total}}}. \tag{2}$$

2) ACK and TCP control packets (SYN, FIN): SYN and FIN control packets are regarded as types of ACK packets in the analysis, as their only difference from ACK is the setting of a control flag in the TCP segment header. The calculation of utilization of a link due to ACK packets is a bit difficult. Delayed ACK strategy causes an ACK to be sent for every other full-size segment, so the number of generated ACK packets is around half of the number of DATA packets in a large file transfer. However, many short flows end up sending only one or two DATA packets, causing an ACK to be generated for each DATA packet. Moreover, a delayed ACK strategy may change in future variants of TCP. Let S_A be the average size of ACK packets and d be the ratio of the number of ACK and DATA packets on an outgoing link estimated from traffic stats after each control period. As the number of ACK packets is d times the number of DATA packets and the size ratio of ACK and DATA packets is (S_A/S_D), the ratio of the amount of traffic due to ACK and DATA packets is $d \cdot (S_A/S_D)$. Multiplying this ratio with $\lambda \cdot E_{(L,H)}$, which is the incoming traffic rate of an outgoing fiber due to DATA packets, gives the incoming traffic rate of an outgoing fiber due to ACK packets. Dividing this by the fiber capacity $C \cdot W_{\text{total}}$ gives the utilization due to ACK packets, which is denoted by U_{all_A}, as

$$U_{\text{all}_A} = \frac{d \cdot (S_A/S_D) \cdot \lambda \cdot E_{(L,H)}}{C \cdot W_{\text{total}}}. \tag{3}$$

The utilization of a fiber (and also a wavelength) in a pure packet switching network, which is denoted by U_{all}, can be calculated by summing the utilization due to DATA and ACK packets, calculated by Eqs. (2) and (3), as

$$U_{\text{all}} = U_{\text{all}_D} + U_{\text{all}_A}, \tag{4}$$

$$U_{\text{all}} = \frac{\lambda \cdot E_{(L,H)} + d \cdot (S_A/S_D) \cdot \lambda \cdot E_{(L,H)}}{C \cdot W_{\text{total}}}. \tag{5}$$

As given in Ref. [13], in a bounded Pareto flow size distribution with minimum flow size L and maximum flow size H, the probability that the size of a flow is less than the threshold t is

$$\Pr(X < t) = \frac{1 - L^{\alpha} t^{-\alpha}}{1 - (L/H)^{\alpha}}. \tag{6}$$

The types of traffic carried over the packet subnetwork by DOWFAF and their utilization are as follows:

1) DATA packets of flows shorter than t: These packets are carried over the packet subnetwork. Let W_{packet} be the number of wavelengths assigned to the packet subnetwork at a time. The network changes W_{packet} according to the traffic, so it is not fixed. The probability that a flow is shorter than t is $\Pr(X < t)$, so the DATA traffic rate of flows shorter than t can be calculated by multiplying this probability with the overall arrival rate of DATA packets and their mean size: $\Pr(X < t) \cdot \lambda \cdot E_{(L,t)}$. The average utilization of packet wavelengths due to the DATA packets of flows shorter than t, which is denoted as U_{hyb_D}, can be calculated by dividing this traffic rate by the total capacity of the packet wavelengths:

$$U_{\text{hyb}_D} = \frac{\Pr(X < t) \cdot \lambda \cdot E_{(L,t)}}{C \cdot W_{\text{packet}}}. \tag{7}$$

2) ACK packets: The ACK traffic rate on a link was found previously in the pure packet switching architecture as $d \cdot (S_A/S_D) \cdot \lambda \cdot E_{(L,H)}$. As all ACK packets are carried in the packet subnetwork, the utilization due to ACK packets denoted by U_{hyb_A} can be found by dividing this traffic rate by the total capacity of the packet wavelengths:

$$U_{\text{hyb}_A} = \frac{d \cdot (S_A/S_D) \cdot \lambda \cdot E_{(L,H)}}{C \cdot W_{\text{packet}}}. \tag{8}$$

3) DATA packets of flows that are larger than t but directly forwarded to the packet subnetwork due to lack of flow size information: Let *req* be the probability that a flow requests to be carried over the path subnetwork. Then the probability that a flow is larger than t and does not request to be carried over the path network is $(1 - req) \cdot (1 - \Pr(X < t))$. Multiplying this probability with the arrival rate of DATA flows on this fiber, which is λ, and further multiplying with the mean size of flows larger than t, which is $E_{(t,H)}$, gives the traffic rate of DATA packets of flows that are larger than t but directly forwarded to the packet subnetwork due to lack of flow size information. Its utilization, denoted by $U_{\text{hyb}_{Di}}$, can be found by dividing this traffic rate by the total capacity of the packet wavelengths:

$$U_{\text{hyb}_{Di}} = \frac{(1 - req) \cdot (1 - \Pr(X < t)) \cdot \lambda \cdot E_{(t,H)}}{C \cdot W_{\text{packet}}}. \tag{9}$$

4) DATA packets of flows that are larger than t and contain the size information but failed in reserving a path wavelength and thus are assigned to a packet subnetwork: The architecture tries to keep the maximum reservation blocking rate in the path subnetwork around a threshold parameter denoted as T_B, which will be explained in Section II.C. The probability that a flow is larger than t and carries flow size information but fails reservation is $T_B \cdot req \cdot (1 - \Pr(X < t))$. Multiplying this probability with the arrival rate of DATA flows on this fiber, which is λ, and further multiplying with the mean size of flows larger than t, which is $E_{(t,H)}$, gives the traffic rate of DATA packets of flows that are larger than t and carry flow size information, but that failed reservation. Its utilization, which is denoted by $U_{\text{hyb}_{Df}}$, can be found by dividing this traffic rate by the total capacity of the packet wavelengths:

$$U_{\text{hyb}_{Df}} = \frac{T_B \cdot req \cdot (1 - Pr(X < t)) \cdot \lambda \cdot E_{(t,H)}}{C \cdot W_{\text{packet}}}. \tag{10}$$

5) Partial data sent to the packet subnetwork until the flows that are larger than t succeed in reserving a path: If this traffic is kept low by choosing a low T_B, retrial count, and back-off time, it can be omitted in the analysis.

6) Retransmissions due to packet losses in the packet subnetwork: The packet drop rate in the packet switching optical networks is expected to be low, so this extra traffic can be omitted in the analysis.

7) Other packet types, e.g., control packets like ARP, ICMP, and UDP traffic: The majority of Internet traffic is TCP. Unless other types of packets have a significant amount of traffic in the network, they can be carried in the packet subnetwork together with TCP traffic and omitted in the analysis. Otherwise, they may be carried on dedicated and fixed packet wavelengths that are not controlled by DOWFAF so that they do not affect the control framework.

The total utilization per wavelength in the packet subnetwork, which is denoted by U_{hyb}, can be estimated by summing the utilization due to these traffic types:

$$U_{\text{hyb}} = U_{\text{hyb}_D} + U_{\text{hyb}_A} + U_{\text{hyb}_{Di}} + U_{\text{hyb}_{Df}}. \tag{11}$$

We want to calculate the flow size threshold, which gives equal utilization on a link in the packet subnetwork of

DOWFAF and a pure packet switching network architecture. Therefore, we equalize the utilization of the packet subnetwork of DOWFAF and a pure packet switching network architecture by

$$U_{\text{all}} = U_{\text{hyb}}. \tag{12}$$

Equalizing the utilization of the packet subnetwork of DOWFAF and the pure packet switching network also implies equalizing them to the utilization of the path subnetwork. After expanding Eq. (12) and applying reduction, we get

$$\begin{aligned} W_{\text{packet}} \cdot E_{(L,H)} \cdot (1 + d \cdot (S_A/S_D)) \\ = W_{\text{total}} \cdot (\Pr(X < t) \cdot E_{(L,t)} + d \cdot (S_A/S_D) \cdot E_{(L,H)} \\ + (1 + req \cdot (T_B - 1)) \cdot (1 - \Pr(X < t)) \cdot E_{(t,H)}), \end{aligned} \tag{13}$$

which gives the flow size threshold t at a selected packet wavelength count W_{packet}. This equation does not depend on C and λ, so it is independent of the wavelength capacity and the flow arrival rate to a link. Moreover, the equation does not require the traffic matrix (traffic among source–destination pairs) in the network. The parameters in the equation can be estimated and used according to traffic statistics by sampling the packets passing through the link. In the case of difficulty in estimating the ACK traffic, the ACK packets may be carried on dedicated and fixed packet wavelengths so that they do not affect the analysis. In that case, d in Eq. (13) is set to zero.

By solving Eq. (13), t is calculated for each link, and the lowest one is applied to the network. The value of t decreases with decreasing *req* and increasing T_B and d. The TCP DATA packets are the main sources of traffic on the links, and the controller tries to keep the ratio of DATA traffic on the packet and path subnetworks the same at all links by applying the same t throughout the network. However, the ratio of ACK packets is higher on some links, which increases their d and causes them to have a lower t than other links. Moreover, the packet subnetwork of some links may be carrying more large flows that failed reservation due to high utilization, which causes them to have a lower effective t than other links. By choosing the lowest t among the t values calculated for each link, the network equalizes the utilization of the corresponding pure packet switching network and path and packet subnetworks of the link that has the highest ratio of extra traffic. As a result, the flows on the packet subnetwork of this link are not discriminated when compared with a pure packet switching network. On the other hand, when the lowest t is applied, the links with a higher calculated t will have a lower utilization in their packet subnetworks compared to a pure packet switch, so the flows on these links can get even higher goodput than in pure packet switching.

C. Dynamic Wavelength Allocation

When the t value calculated by the analysis decreases, this allows more flows to be carried on the path subnetwork and to benefit from the advantages of using a dedicated path. However, this decreases the average flow size in the path subnetwork. The packet switching wavelengths can achieve a high level of utilization independent of the flow size distribution, due to the high granularity multiplexing of packets of many flows on the same wavelength. However, the path switching suffers from wavelength reservation signaling delays and the slow convergence speed of TCP. The short flows decrease the utilization efficiency in the path subnetwork, because most of the wavelength capacity is wasted until a TCP flow increases its congestion window size high enough to achieve high utilization on a fast wavelength. Moreover, a wavelength becomes idle during the signaling and switch configuration when establishing a path reservation, so the utilization efficiency decreases as the reservation frequency increases due to shorter flows. This low efficiency may cause a high reservation blocking rate in the path subnetwork, even when the utilization is low. As a result, some of the flows larger than t may fail path reservation and end up using the packet subnetwork and further increase the utilization and the packet drop rate in the packet subnetwork, which decreases the goodput of flows. Therefore, it is necessary to find the optimum number of path switching wavelengths that can carry the maximum number of large flows while not passing the path blocking threshold and not increasing the utilization of the packet subnetwork more than the utilization of a pure packet switching network.

As stated before, it is difficult to estimate the goodput of TCP flows, as new TCP variants have different congestion control algorithms. Moreover, the goodput depends on many factors, like the packet drops that the flow experiences outside the optical domain and the processing speed of end hosts. Therefore, we apply a heuristic algorithm with feedback control in order to detect the maximum number of wavelengths that can be allocated to the path subnetwork. DOWFAF changes the ratio of wavelengths for path and packet subnetworks according to the recent reservation blocking statistics in the path subnetwork, so it can adapt itself to varying traffic without requiring the traffic matrix information. In the optical domain, all nodes use the same wavelength ratio and t, so a centralized control is easier to implement. In the network, one of the routers acts as a center node, which decides on both parameters. All nodes in the optical network send the path reservation success rate information to the center node by control packets, periodically. The architecture does not need the traffic matrix or the total traffic arrival rate information. Other parameters like the flow size distribution, the *req* parameter, and the ratio of forward DATA and reverse ACK packets on links are estimated from the traffic statistics on a link and sent to the center node only when there is a drastic change. The center node decides the wavelength ratio and the flow size threshold to be used network-wide in the next control period and informs the other nodes by control packets.

If the center node decides to convert some packet switching wavelengths to path switching, the nodes move the flows assigned to these wavelengths to other packet switching wavelengths. If some path switching wavelengths are

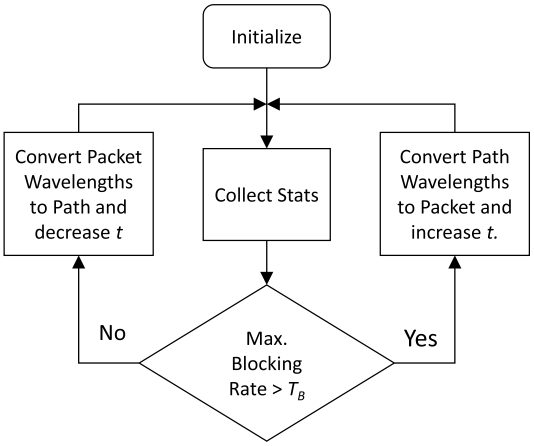

Initialize
Convert Packet Wavelengths to Path and decrease *t*
Collect Stats
Convert Path Wavelengths to Packet and increase *t*.
No
Max. Blocking Rate > T_B
Yes

Fig. 3. Flow chart of the dynamic wavelength allocation algorithm.

converted to packet switching, the nodes try to reserve new paths on the remaining switching wavelengths for the flows on the to-be-converted wavelengths. If the reservation succeeds, the remaining packets of a flow are sent to the new path over another wavelength. Otherwise, the flow is switched to one of the packet switching wavelengths. The router may retry path reservation for these flows. When there are no packets left in transit on the to-be-converted wavelengths, the routers inform the central node and start using these wavelengths according to the switching paradigm assigned by the center node after updating their switch configurations.

Unless the traffic matrix is varying too fast or the network has too many wavelengths, a heuristic trial-and-error algorithm can be enough to optimize the ratio of wavelengths assigned to the path and packet subnetworks dynamically by using the path blocking rate statistics collected periodically. Figure 3 shows the flow chart of the algorithm used for controlling the wavelength allocation. Initially, the ratio of packet wavelengths is maximized to be sure that the network can carry the current traffic. After a control period, all nodes start sending their stats, like path reservation success rate, to the center node by sending control packets. If the maximum blocking rate in the network is lower than a threshold T_B, the controller node increases the number of wavelengths assigned to the path subnetwork by converting one of the wavelengths in the packet subnetwork to the path subnetwork. Also, t is decreased according to the analytical model. If the highest blocking rate in the network is greater than T_B, one of the path switching wavelengths is converted to a packet switching wavelength and t is increased according to the analytical model.

D. Flow Switching

In order to prevent the packet reordering, which decreases the TCP goodput, the architecture employs flow-based switching, so all packets of a flow use the same wavelength regardless of the switching type. When a new TCP packet from an end host or an edge router arrives at the optical domain, first the hash of certain fields in the packet header is computed and compared with the hash table in the node. If the computed hash is available in the hash table, this means that the packet is a part of a currently active and known flow. If there is a path established for this flow according to the hash table info, the packet is sent directly to the next node over the path switching wavelength reserved for this flow. When there is packet contention in a packet switching wavelength, the contending packet is delayed by a fiber delay line (FDL) buffer until the wavelength is available, or dropped if the delay by the FDL is not enough to solve the contention.

If the hash is not available in the hash table and the packet type is SYN or SYN-ACK of TCP, then this means that the router received the first packet of a new TCP flow. The router forwards the large flows to the path switching subnetwork, so the router should know or detect whether this is a large flow or not. One possible solution to detect the large flows is to use the flow stats. The router may initially forward a flow to the packet subnetwork and record its stats. The router can classify a flow as a large flow and switch it to the path subnetwork if the total size of the packets of the flow carried in the network passes a threshold in a time window. However, in this method the flow ends up using the packet subnetwork until it transfers enough packets to pass the threshold, which increases the utilization of the packet subnetwork. Moreover, large flows may experience packet losses in the packet switching subnetwork, which may limit the growth of the TCP congestion window size and cause the flows to spend too much time in the packet subnetwork before they transfer enough data to pass the threshold. Therefore, we consider informing the routers about the size of the flows that are candidates for the path switching subnetwork. One way to inform the routers about the size of a flow can be that the application at the end host may write the flow size information to the TCP header option field or the TCP payload of the initial SYN or SYN-ACK packet. Moreover, some extra information like the link speed of the end host may be added to eliminate the end hosts that do not have a connection fast enough to achieve high utilization in a path wavelength. Section 3.4 of IETF RFC 793 allows data to be carried in the SYN packets [14]. For example, TCP Fast Open protocol, which was added to the Linux kernel between versions 3.6 and 3.7 and was turned on by default in 3.13 and supported by Google Chrome and Chromium browsers, proposes carrying initial data in the SYN and SYN-ACK packets. Another possible way to detect the flow size in Hypertext Transfer Protocol (HTTP) traffic is to check the Content-Length field in the HTTP header, but this is more processing intensive.

Another possible way to inform the routers about the size of a flow is to use software-defined networking (SDN) like OpenFlow [9], which is getting a lot of attention recently. If SDN is available, the agents at the end hosts can directly inform the SDN controller about the size of the flow and the preferred switching plane (path or packet switching) instead of writing the flow size information to the header or payload of SYN or SYN-ACK packets of the flow. The SDN controller can choose the switching plane and assign the flow based on this information using DOWFAF. Reference [7] reported that a messaging framework for resource requests and a centralized configuration based on OpenFlow have already been implemented on OPCInet.

When a new flow arrives at the network, the edge router tries to reserve a path for the flow inside the optical network if the flow size information is available and the size is higher than the current t. The initial TCP packet is buffered in the router until the reservation attempt concludes. If the reservation succeeds, the router adds the flow to the hash table with the path information and the initial TCP packet is sent over the established path. The ACK packets are carried over the packet subnetwork. If the reservation fails, the flow is sent to the packet subnetwork. The router may back off some time and retry the path reservation for the flow. If the retrial succeeds in reserving a path, the flow switches to the established path. Otherwise the flow uses the same wavelength on the packet subnetwork until it finishes.

The end host does not need to be directly connected to the optical network. A flow may also traverse other network domains and use the path subnetwork of the optical network on its route as long as it achieves high goodput. The end host is not informed about the switching paradigm used for its packets in the optical network, so the end host uses the same TCP connection during the data transfer regardless of whether the path or packet subnetwork is used. If the application does not want the flow to be a candidate for the path subnetwork, it does not write flow size information to the initial packet. The architecture directly assigns these flows to the packet subnetwork. It is possible that a path wavelength is reserved for a flow upon the request of the application, but the flow cannot achieve high utilization in the path wavelength due to a problem like processing limitations of the end host, behavior of the application at the end host, or a high packet drop rate on a link between the end host and the optical network. The edge node, where the packets of this flow enters the optical network, samples the incoming traffic rate and reassigns such flows to the packet subnetwork by tearing the path connection.

For path reservation, we used destination-initiated reservation (DIR), which is one of the most popular reservation algorithms in the literature [15]. Resource Reservation Protocol with extensions for traffic engineering (RSVP-TE) [16] signaling protocol in generalized multiprotocol label switching [17] networks uses DIR for wavelength reservation [18].

III. Evaluation

This section discusses the performance of DOWFAF by evaluating the transient wavelength allocation ratio, the goodput of TCP flows, and the cost and power consumption of the architecture.

A. Simulation Settings

We evaluated the performance of DOWFAF on the National Science Foundation Network (NSFNET) topology with 14 nodes and 21 bidirectional links, shown in Fig. 4, by using a packet-based simulator that we implemented. There were 80 wavelengths per link. Each wavelength had a speed of 1 Gbps, so the total capacity of a link was 80 Gbps. Although WDM allows much higher wavelength speeds, we used 1 Gbps due to simulation time restrictions. Link delay was 10 ms. Shortest-path routing was applied. The maximum and the average hop count were 3 and 2.1, so the maximum and average round trip time (RTT) were around 60 and 42 ms. The minimum flow size was 1000 bytes, and the maximum flow size was 50 Gbytes. The bounded Pareto shape parameter was 1.01. We applied the traffic demand matrix in [19] with shortest-path routing. The flows between each node pair arrived according to a Poisson process. The core links used FDL buffering with 30 delay lines and the FDL granularity was 500 bytes, so the maximum buffer size was 15 Kbytes per wavelength. If a packet contention was not solvable with the available buffering space, the contending packet was dropped. The maximum packet size was 1500 bytes. At this FDL configuration, the maximum achievable wavelength utilization was around 60% due to the voids occurring in FDLs. T_B was selected as 0.05. The total simulation time was 220 min. The maximum number of path reservation trials was three times. The back-off time between reservation retrials was 300 ms. We used the CUBIC TCP code from Ref. [11] written for ns-2 [20] and updated its code to use in our simulator.

The maximum flow size threshold was calculated by using the analytical model in Section II.B. Figure 5 shows the threshold for $req = 1$ and $req = 0.6$ values calculated by solving Eq. (13) using the Mathematica software package [21], which solved the equation almost instantly. The x axis is the number of path wavelengths and the y axis is t, which is the maximum flow size threshold, on a log scale. As seen in the figure, t decreases as req decreases.

In order to evaluate the transient wavelength allocation ratio in DOWFAF, the total traffic applied to the network was varied, as shown in Fig. 6. Initially, the total flow arrival rate was set to 400,000 flows/s. At 100 min, the total traffic was increased to 1,000,000 flows/s to simulate an instant traffic surge and network congestion. When 1,000,000 flows/s traffic was applied to a pure packet wavelength switching network, the expected utilization level of the most congested link in the network was around 30%. Most service providers operate their backbone networks at lower utilization [22]. Finally, the traffic decreased to a

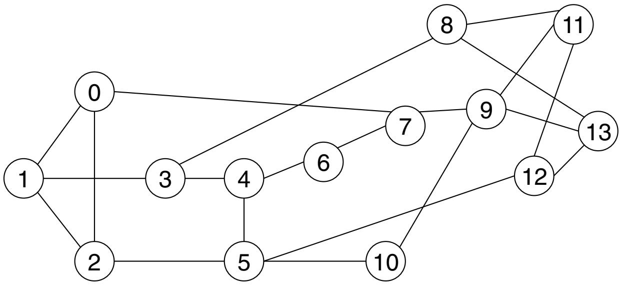


Fig. 4. NSFNET topology.

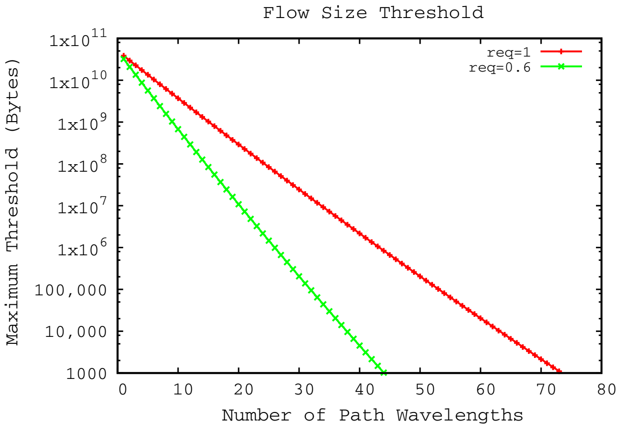


Fig. 5. Maximum flow size threshold.

mid-level of 600,000 flows/s at 160 min. In a single simulation, more than 8×10^9 TCP flows were carried in the network.

Two traffic scenarios were simulated. In the first scenario, all large flows carried the flow size information ($req = 1$). In the second scenario, only 60% of the flows carried the flow size information, so 40% of the flows were directly forwarded to the packet subnetwork ($req = 0.6$).

B. Simulation Results

All wavelengths were shared by the path and packet subnetworks, like the blue wavelength in Fig. 2, so the maximum required number of wavelengths for the OCS-SW and OPS-SW can be evaluated by looking at the transient wavelength allocation ratio. Figure 7 shows the number of wavelengths allocated to the path and packet subnetworks by DOWFAF for $req = 1$, denoted by *DOWFAF-1*, and for $req = 0.6$, denoted by *DOWFAF-0.6*. Initially, all wavelengths were allocated to the packet subnetwork, so the network operated like a pure packet switching architecture. At the end of each control period, the controller node received the path blocking rate information from all nodes.

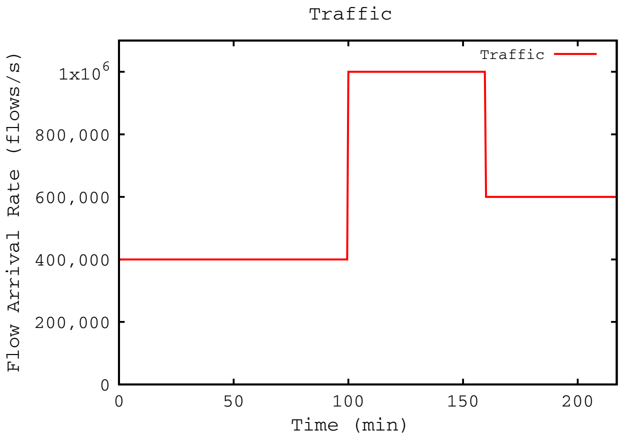
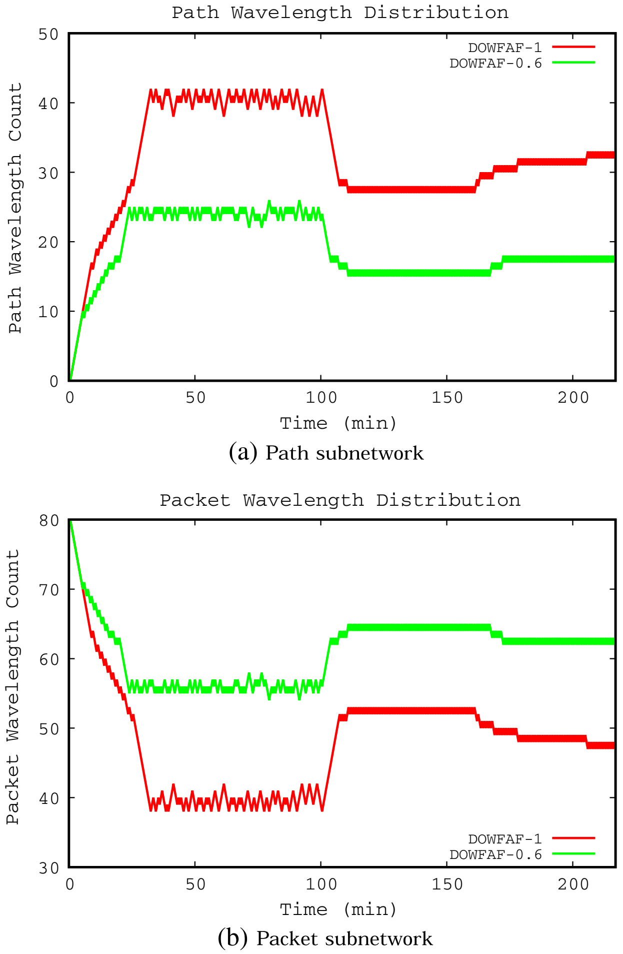


Fig. 6. Flow arrival rate at the network versus time.



(a) Path subnetwork

(b) Packet subnetwork

Fig. 7. Number of wavelengths allocated to the path and packet subnetworks by the integrated architecture versus time.

At first, the path subnetwork carried the assigned large flows with a low blocking rate because t was high. Figure 8 shows the value of t on a log scale versus time. As t was high, the congestion control algorithm of the assigned TCP flows had enough time and data to increase the congestion window size to fully utilize the path

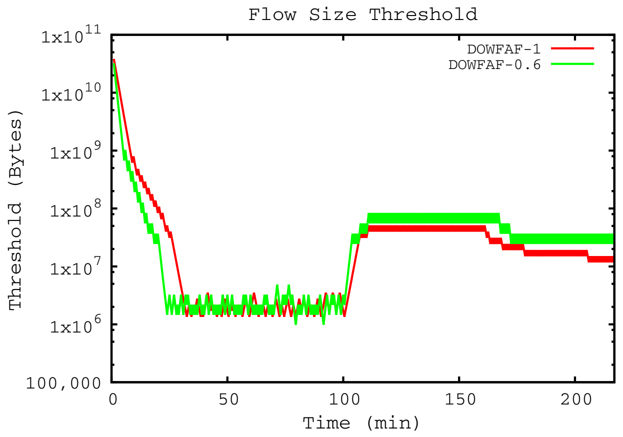


Fig. 8. Flow size threshold versus time.

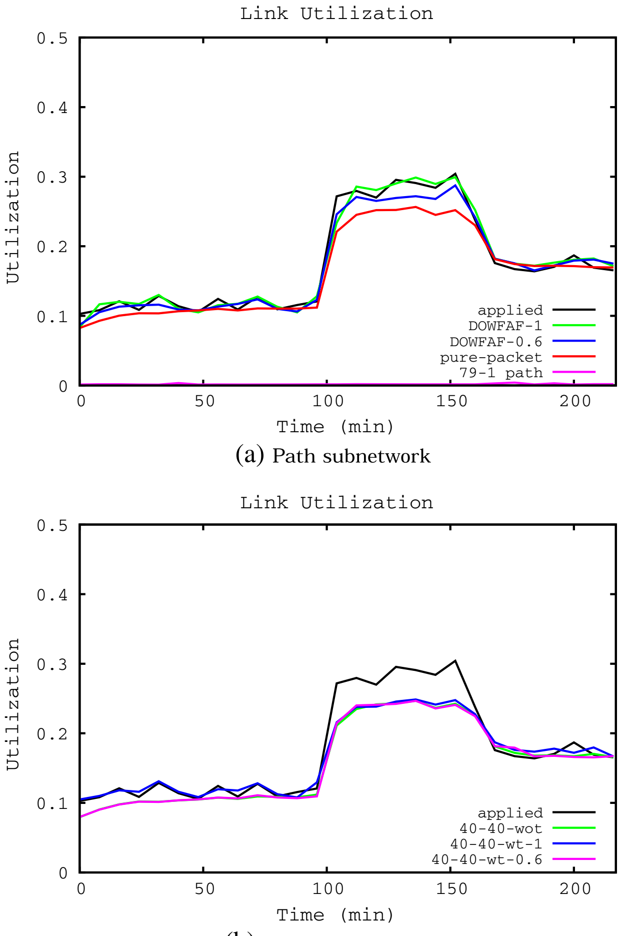
wavelengths. Moreover, the time wasted during the configuration and reservation in the path subnetwork was much lower than the flow duration, so the utilization efficiency was high. While the maximum blocking rate was below T_B, the architecture kept converting the wavelengths from packet switching to path switching. As the number of path wavelengths increased, t calculated by the analytical model decreased and allowed more flows to benefit from the dedicated path wavelengths. However, the efficiency of the path subnetwork decreased as t decreased. The capacity wasted during the TCP slow start and the path reservation phase became significant. Therefore, the path reservation blocking rate started to oscillate around the path blocking rate threshold T_B and the system stabilized. Figure 5 shows that t per path wavelength decreases with decreasing *req*. Lower t causes the wasted capacity to be higher when *req* is lower at the same number of path wavelengths. As a result, the path blocking rate reached T_B when there were fewer path wavelengths. Therefore, the number of path wavelengths was lower when $req = 0.6$, as seen in Fig. 7(a).

When the incoming traffic was increased to 1,000,000 flows/s after 100 min, the system started to send more flows to the path subnetwork, which caused the path blocking rate to pass T_B. The controller responded by decreasing the number of path switching wavelengths by converting them to packet switching wavelengths. As the number of path wavelengths decreased, the corresponding t and the average flow size in the path subnetwork increased. As larger flows utilized the path wavelengths more efficiently, the path subnetwork started to operate at higher utilization while satisfying the blocking rate threshold.

The congestion in the network was relieved by decreasing the traffic to 600,000 flows/s at 160 min. The utilization and the reservation blocking rate of the path subnetwork decreased, so the controller node again converted the packet switching wavelengths to path switching while decreasing t.

Figure 7 shows that when the traffic arrival rate was increased suddenly at the end of the first epoch, the biggest change in the wavelength distribution occurred with *DOWFAF-1*, where the system adapted to the new traffic and stabilized by increasing the number of packet wavelengths from 39 to 52 in only 7.5 min. In the worst-case scenario, if the number of packet switching wavelengths increases from zero to the maximum possible number of packet switching wavelengths, the minimum time required to change the wavelength distribution would be $0.5 \times 80 = 40$ min, when the control period is 0.5 min, the maximum number of packet wavelengths is 80, and the number of converted wavelengths at each control period is one. The convergence speed of the trial-and-error algorithm can be increased by decreasing the control period time and/or increasing the number of wavelengths converted after each period. If the traffic is varying too fast or the network has too many wavelengths, more advanced wavelength allocation algorithms may be employed. Designing such advanced wavelength allocation algorithms is left to future work.

Figure 9 shows the average utilization rate of the wavelengths on the fiber from node 4 to 3, which is the most congested fiber in the network. As there are many simulation results, the results are split into two subfigures [Figs. 9(a) and 9(b)]. As a reference, the utilization estimated from the applied traffic matrix, which is denoted as *applied*, is plotted in both subfigures. It is calculated by dividing the sum of the sizes of the incoming flows in a 500 s time window by the fiber capacity. It assumes that incoming flows were carried instantaneously. Therefore, *applied* shows the optimum utilization that can be achieved. Figure 9(a) compares DOWFAF with two extreme cases of wavelength distribution. The first one is *pure-packet*, which shows the average utilization when the traffic pattern was simulated with a pure packet switching network. The second one is *79-1 path*, which shows the average utilization when 79 wavelengths were dedicated to path switching and a single wavelength was dedicated to packet switching. As control packets require a packet switching wavelength, it is not possible to simulate a pure path switching network, but a close result is obtained by simulating with 79 path wavelengths. As the blocking rate was high due to the low efficiency of the path wavelengths and a single packet wavelength was not able to carry all blocked flows, the



(a) Path subnetwork

(b) Packet subnetwork

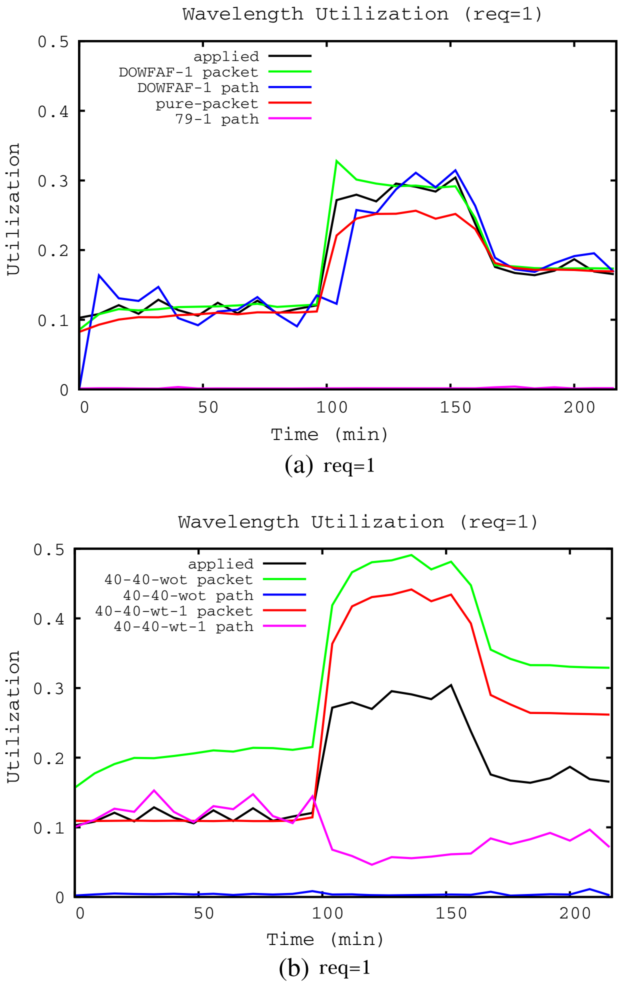
Fig. 9. Link utilization.

flows that failed reservation were deleted from the network in *79-1 path*. Only the control packets of DOWFAF and the ACK, SYN, and FIN packets of TCP flows were carried over the single packet wavelength.

When *pure-packet* and *applied* are compared, we see that *pure-packet* had a bit lower utilization at the beginning, but converged to *applied* at the end of the first epoch. The reason for slow convergence was the slow increase in the TCP congestion window of the large flows due to the packet drops in the packet switching network. Therefore, the utilization in the *pure-packet* architecture converged to *applied* only after many large flows accumulated on the link over a wide time span. The problem became more visible after the traffic increased in the second epoch. As the packet drop rate increased further, the congestion window of the TCP flows increased more slowly, so it could not converge to the utilization level of *applied* in the second epoch. In the third epoch, there were already many large flows that had accumulated from the previous epochs, so *pure-packet* converged faster in this epoch. When we check the result of DOWFAF, we see that the simulation with *DOWFAF-1* converged to *applied* in a short time. As DOWFAF carried most of the large flows on dedicated wavelengths, the large flows increased their congestion windows faster and finished sending their data in a short time, so the utilization closely followed the pattern of *applied*. Forty percent of the flows were directly forwarded to the packet subnetwork due to lack of flow size information in *DOWFAF-0.6*. Even though the packet drops degraded the performance of the large flows directly forwarded to the packet subnetwork, the majority of the large flows still benefited from the dedicated wavelengths, so the utilization in the second epoch was higher than *pure-packet*. Finally, *79-1 path*, which shows the average utilization when 79 wavelengths were dedicated to path switching and a single wavelength was dedicated to packet switching, gave a very low utilization with an average of 0.0018, so it is difficult to see in the graph. While DOWFAF controlled the ratio of wavelengths to keep the maximum blocking rate in the network around 0.05, the maximum blocking rate was around 0.995 when the path wavelength count was fixed to 79. Most of the flows that succeeded reservation could not utilize the capacity of path wavelengths due to their small size and caused extremely low utilization.

Figure 9(b) shows the average utilization rate of the wavelengths when the wavelength distribution is fixed to an equal separation of 40 path and 40 packet wavelengths. *40-40-wot* shows the case when there is no flow size threshold, so all incoming flows first try to reserve one of the 40 path wavelengths and use a packet wavelength in case the reservation fails. *40-40-wt-1* and *40-40-wt-0.6* show the cases when the flow size threshold calculated by DOWFAF for a 40–40 wavelength distribution is applied when 100% and 60% of the flows carry the flow size information, respectively. The only difference between *40-40-wt* and DOWFAF simulations is that the number of path and packet wavelengths is fixed to 40 in *40-40-wt*, so it shows the case when the wavelength distribution and the threshold are not updated according to the traffic. Again, *applied* is plotted as a reference. Figure 9(b) shows that none of the simulations could achieve the same level of utilization as *applied* and DOWFAF when the incoming traffic rate was 1,000,000 in the second epoch.

In order to show that DOWFAF can equalize the utilization of path and packet subnetworks by distributing the flows effectively, the transient utilization of path and packet wavelengths on the fiber from node 4 to 3 are plotted in Figs. 10 and 11 for $req = 1$ and $req = 0.6$, respectively. As this was the most congested fiber, DOWFAF was expected to equalize the utilization of this fiber in the path and packet subnetworks of DOWFAF and a pure packet switching network by solving Eq. (12) and applying the calculated t network-wide. The average utilization of the packet and path wavelengths of each simulation are denoted as *packet* and *path*, respectively. The *path* utilization is calculated by the total amount of data carried on the path wavelengths in a time window. Figure 10 shows the first scenario when all large flows carried the flow size information ($req = 1$). As there are many simulation results, again the results are divided into two subfigures [Figs. 10(a) and 10(b)]. Figure 10(a) reveals that *DOWFAF-1 path* and *DOWFAF-1 packet* by DOWFAF and *applied* had similar utilization. Both *DOWFAF-1 path* and *DOWFAF-1 packet*



(a) req=1

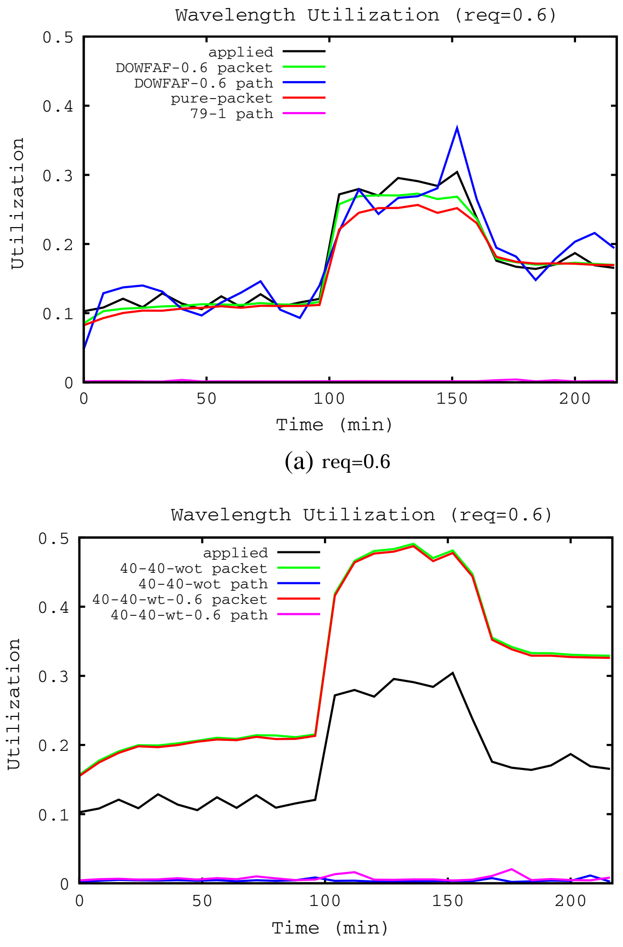
(b) req=1

Fig. 10. Comparison of wavelength utilization when $req = 1$.

closely followed *applied*, which is the optimum utilization, and both had a fast convergence to *applied* when the traffic rate changed at 100 and 160 min, showing that a good load balance was achieved by DOWFAF. The result denoted by *79-1 path*, which was the average utilization of path wavelengths when 79 wavelengths were dedicated to path switching and a single wavelength was dedicated to packet switching, shows that the utilization of path wavelengths was very low due to the high blocking rate because of the low efficiency of the short flows. On the other hand, DOWFAF kept the maximum blocking rate around T_B by controlling the ratio of path and packet wavelengths in order to maximize the ratio of path wavelengths and the ratio of large flows switched to the path subnetwork while balancing the utilization of path and packet subnetworks and preventing high blocking.

Figure 10(b) shows that when the distribution of wavelengths was fixed to an equal distribution of 40 path and 40 packet switching wavelengths, the network could not balance the utilization of path and packet subnetworks. When there was no flow size threshold set, the path subnetwork experienced very low utilization (*40-40-wot path*) due to low efficiency, so most of the flows ended up being carried in the packet subnetwork, which caused very high utilization in the packet switching wavelengths (*40-40-wot packet*) compared to *applied*. When the flow size threshold was set according to the DOWFAF analysis at $req = 1$ and 40 path wavelengths, which was also the average number of path wavelengths that DOWFAF converged to in the first epoch, the path (*40-40-wt-1 path*) and packet (*40-40-wt-1 packet*) subnetworks achieved almost equal utilization in the first epoch as expected. However, as the flow arrival rate increased in the second epoch, by applying the same flow size threshold, the path subnetwork caused a high level of path reservation blocking. Blocked large flows ended up using the packet subnetwork, so the utilization of the packet subnetwork was much higher than *applied* in the second and third epochs.

Figure 11 shows the result of the scenario when 60% of the flows carried the flow size information ($req = 0.6$). When $req = 1$, there were few large flows in the packet subnetwork of DOWFAF, so *packet* was close to *path* and *applied*. When *req* is lower, it takes some time for the utilization of packet switching wavelengths to converge to a stable level due to the high number of large flows in the packet subnetwork, which increases their TCP congestion window size more slowly than in the path switching wavelengths due to packet drops. As expected, *DOWFAF-0.6 packet* in Fig. 11(a) was a bit lower than *DOWFAF-1 packet* in Fig. 10(a) due to slow convergence of large flows, while *DOWFAF-0.6 path* and *applied* were close to each other. Figure 11(b) shows that when the number of path and packet wavelengths were fixed to 40 and the flow size threshold was set to the value calculated by DOWFAF for 40 path wavelengths and $req = 0.6$ (*40-40-wt-0.6*), the path subnetwork got very low utilization with a high blocking rate, which caused high utilization in the packet subnetwork throughout the simulation. The reason was that at this traffic arrival rate, the number of packet wavelengths and t should be much higher, as shown in Fig. 7, in order to balance the utilization of path and packet subnetworks.

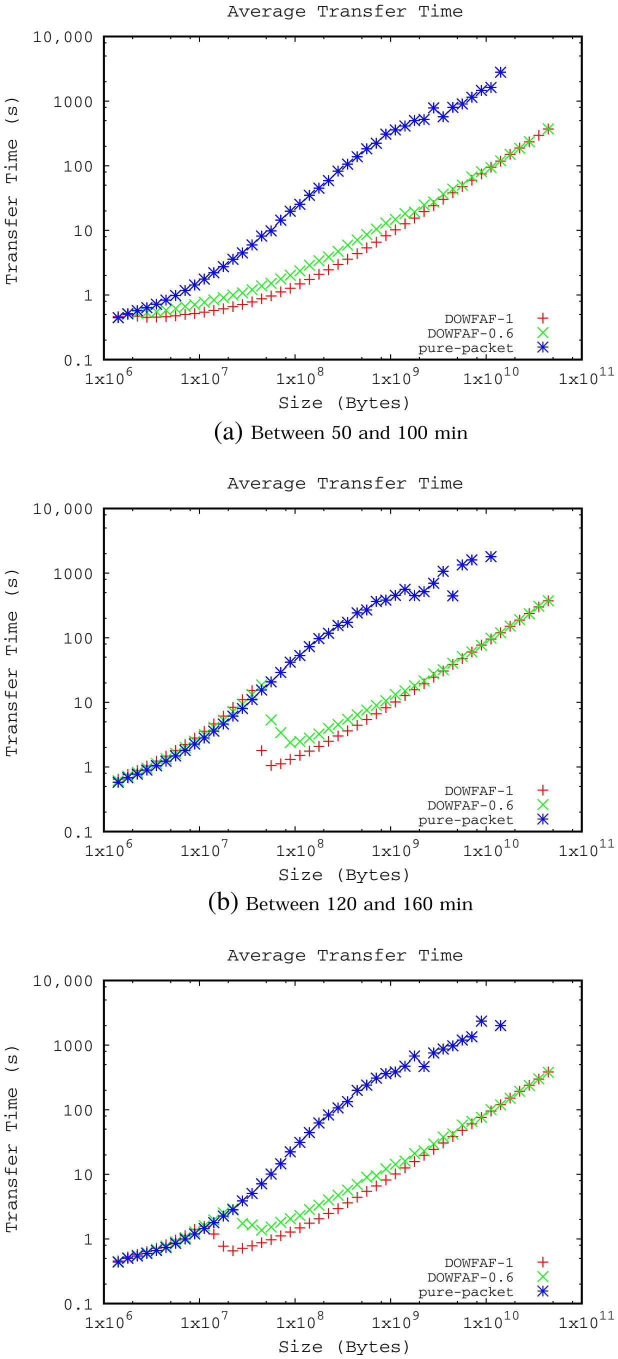


(a) req=0.6

(b) req=0.6

Fig. 11. Comparison of wavelength utilization when $req = 0.6$.

Figure 12 compares the average goodput of all TCP flows larger than 10^6 bytes. The figures are divided into three epochs between 50 and 100 min, 120 and 160 min, and 180 and 220 min. The first 50 min and also the 20 min after the traffic rate changed were ignored, because the wavelength allocation was still in transition. The x axis is the log-scale histogram of the flow size in bytes, while the y axis is the average transfer time in terms of seconds in log scale. In the first epoch, as the traffic was low, t was low in all DOWFAF simulations, as shown in Fig. 8. As a result, a wide range of flows were carried in the path subnetwork and their average transfer time was shorter than in the pure packet switching architecture when $t > 2 \times 10^6$, as shown in Fig. 12(a). Even though around 40% of the large flows were carried in the packet subnetwork when *DOWFAF-0.6*, the average transfer time of the large flows in DOWFAF was up to 40 times lower than *pure-packet*. In the second epoch, the high traffic arrival rate increased t, so fewer flows were carried in the path subnetwork of DOWFAF, as seen in Fig. 12(b). However, the flows in the path subnetwork continued to transfer much faster than in *pure-packet*. Moreover, the flows in the packet



(a) Between 50 and 100 min

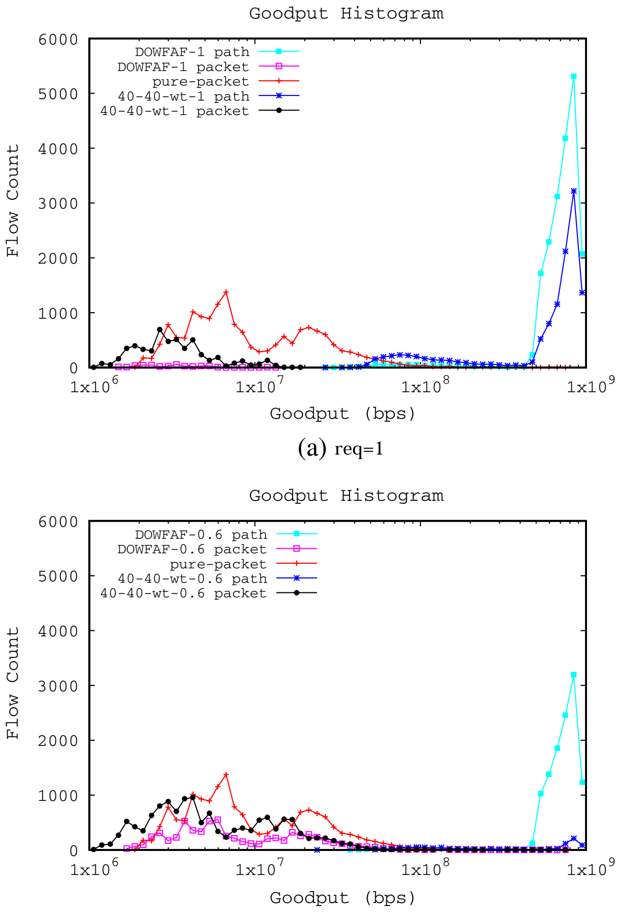
(b) Between 120 and 160 min

(c) Between 180 and 220 min

Fig. 12. Comparison of average transfer time of TCP flows in three epochs.

subnetwork achieved almost the same goodput as in *pure-packet*, which shows that DOWFAF succeeded in equalizing the utilization of the packet subnetwork, the path subnetwork of DOWFAF, and the pure packet switching architecture. Therefore, many large flows achieved up to 40 times higher goodput without discriminating the short flows, which achieved similar goodput as in the pure packet switching architecture. Figure 12(a) shows that when the flow arrival rate decreased in the third epoch, DOWFAF decreased t adaptively and more flows benefited from the path subnetwork.

Figure 13 shows the goodput histogram of the flows sized between 10^8 and 10^9 bytes in the second epoch between 120 and 160 min. The x axis is the flow goodput split into bins on a log scale and the y axis is the number of flows in each bin. Figure 13(a) shows that when $req = 1$, most of the large flows in DOWFAF achieved high goodput, as they were carried in the path subnetwork. Only a negligible number of large flows, which were carried in the packet subnetwork due to failed reservation in the path subnetwork, got lower goodput, but their goodput was in the range of the goodput of flows in *pure-packet*. When the wavelength distribution was fixed to 40–40 and the calculated flow size threshold was applied (*40-40-wt-1*), the number of the flows carried in the path subnetwork was lower than in DOWFAF due to the high blocking rate because of the high traffic in the second epoch and low flow size threshold, even though the number of employed path switching wavelengths was higher than in DOWFAF. Therefore, many large flows ended up being carried in the packet subnetwork, and their goodput was lower than in *pure-packet* due to the higher utilization and packet drop rate in the packet subnetwork than in *pure-packet*. Figure 13(b) shows



(a) req=1

(b) req=0.6

Fig. 13. Comparison of the average goodput distributions in the pure packet switching architecture and DOWFAF.

that when $req = 0.6$, the number of flows with high goodput in *DOWFAF-0.6 path* was lower than in *DOWFAF-1 path*, because 40% of the large flows were directly sent to the packet subnetwork. As DOWFAF equalized the utilization of the packet subnetwork of DOWFAF and the pure packet switching architecture, the flows carried in the packet subnetwork of DOWFAF achieved similar goodput to the flows in *pure-packet*. On the other hand, the goodput of the fixed wavelength distribution (*40-40-wt-0.6*) was much lower than that of DOWFAF, because the applied wavelength distribution was very different from the optimum distribution found by *DOWFAF-0.6 path*.

Finally, Fig. 14 shows the average of the packet loss rate of the flows in the packet subnetwork. Only the flows sized between 10^6 and 3×10^6 bytes were included. Larger flows were not included because they may span multiple epochs with different packet drop rates, and short flows were not included because they may finish before entering the congestion avoidance phase. The flows in the path subnetwork did not experience any packet drops inside the network, so they were not included either. The figure reveals that there were no sudden peaks in the packet loss rate when the wavelength allocation was still in transition when a new epoch started. As the packet subnetwork was underutilized, most of the packet drops were mainly due to packet contentions in the small-sized FDL buffers. Factors like synchronization of packet drops, traffic burstiness, and interactions among TCP flows have a big impact on the rate of packet drops in FDL buffers, so the shape of the packet drop rate in Fig. 14 is different from the flow arrival rate in Fig. 6. The packet drop rates of DOWFAF and *pure-packet* were very similar. Faster wavelength allocation with more aggressive algorithms may cause sudden increases in the packet loss rate, but their evaluation is left to future work.

Some types of traffic were omitted in the analysis based on the assumption that their traffic ratio is low enough to be omitted. Their traffic ratios were checked in the simulation. They are the following:

1) Partial data sent to the packet subnetwork until the flows that are larger than t succeed in reserving a path: In the simulations the maximum number of reservation trials was three and the back-off time was 300 ms. Therefore, the maximum duration a large flow may send partial data to the packet network until it succeeds in reserving a path is the back-off time spent for two failed reservation trials, which is 600 ms, plus the time spent during the last reservation control packet exchange, which is one RTT. As the maximum RTT is 60 ms, the maximum duration for sending partial data is 660 ms. The lowest number of packet wavelengths was in the first epoch, so the extra traffic had the highest impact in the first epoch of the simulation. Figure 12(a) shows that the flows carried around 10^6 bytes of data on average in the packet subnetwork in the first 700 ms. The average size of flows in the path subnetwork in the first epoch was around 2×10^7. Even if all the large flows carried in the path subnetwork had failed reservation twice before succeeding the reservation, they would have sent only around 5% of their initial data to the packet subnetwork. Therefore, this traffic was low enough to be omitted in the analysis.
2) Retransmissions due to packet losses in the packet subnetwork: The maximum average packet drop rate experienced by the flows in the packet subnetwork was 1.9%, so the ratio of retransmitted packets was low enough to be omitted in the analysis.

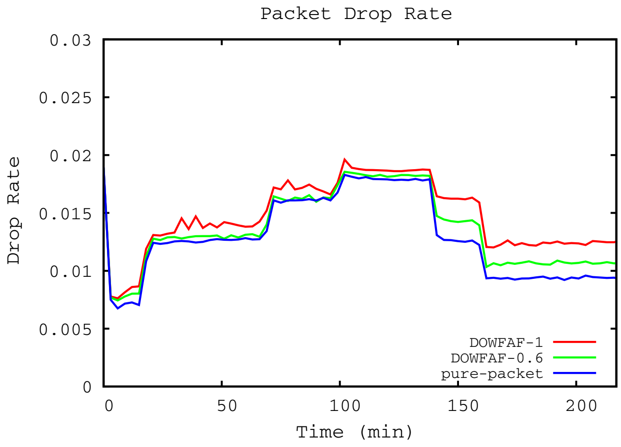


Fig. 14. Average packet drop rate.

C. Cost and Energy Consumption

DOWFAF not only increases the goodput of large flows but is also capable of decreasing the building cost and the energy consumption of the switch. Figure 7(b) revealed that DOWFAF required fewer packet switching wavelengths than a pure packet architecture by carrying some of the traffic over a path switching subnetwork. The cost of OCS-SW fabric is much lower than OPS-SW fabric, which requires much faster switching. Moreover, it is difficult to manufacture OPS-SW fabric with many ports, so even a small decrease in the number of ports of OPS-SW can greatly decrease the manufacturing cost of the switch. Furthermore, as OCS-SW has a much lower switching frequency than OPS-SW, carrying some of the traffic on the path subnetwork decreases the power consumption of the switch. The minimum flow transfer time in the path switching subnetwork was around 1 s. Compared to an OPS-SW with a 10^9 bps wavelength speed, which should be capable of switching packets with a duration on the order of 10^{-7} s, the OCS-SW can operate at a much lower granularity. The low granularity allows the use of a switching technology like in microelectromechanical systems, which is slower but consumes much less power and is much cheaper to manufacture [23]. As a result, DOWFAF can decrease the overall manufacturing cost and the energy consumption of a switch.

IV. Conclusions

In this paper, we proposed a control framework called DOWFAF for integrated optical architectures combining path and packet switching for future optical networks. We proposed an analytical model for calculating the flow size threshold and a feedback control for estimating the wavelength allocation ratio for varying traffic without requiring the traffic matrix information in the network. The simulation results on a mesh network revealed that DOWFAF adaptively changed the ratio of path and packet wavelengths in order to balance the utilization of path and packet subnetworks and to maximize the ratio of large flows benefiting from the path switching, which greatly increased the goodput of the large TCP flows. Moreover, the architecture decreased the power consumption of the switch by carrying part of the traffic over a low power path switching fabric and decreased the manufacturing cost by decreasing the size of the packet switching fabric.

In future work, we will work on real-world implementation issues. We will explore the effect of flow size distributions on the performance of the architecture. If the traffic is varying too fast or the network has too many wavelengths, more advanced wavelength allocation algorithms may be employed than the trial-and-error method. Designing such advanced wavelength allocation algorithms is left to future work.

Acknowledgment

This work was supported by the National Institute of Information and Communications Technology (NICT).

References

[1] H. Harai, “Optical packet and circuit integrated network system and testbed,” in *ECOC Co-Located Workshop*, 2010, invited talk.

[2] S. Arakawa, N. Tsutsui, and M. Murata, “A biologically-inspired wavelength resource allocation for optical path/packet integrated networks,” in *Proc. 15th Conf. Optical Network Design and Modeling*, 2011.

[3] E. Van Breusegern, J. Cheyns, D. De Winter, D. Colle, M. Pickavet, F. De Turck, and P. Demeester, “Overspill routing in optical networks: A true hybrid optical network design,” *IEEE J. Sel. Areas Commun.*, vol. 24, no. 4, pp. 13–25, 2006.

[4] S. Bjornstad, D. Hjelme, and N. Stol, “A packet-switched hybrid optical network with service guarantees,” *IEEE J. Sel. Areas Commun.*, vol. 24, no. 8, pp. 97–107, 2006.

[5] M. Veeraraghavan and X. Zheng, “A reconfigurable Ethernet/SONET circuit-based metro network architecture,” *IEEE J. Sel. Areas Commun.*, vol. 22, no. 8, pp. 1406–1418, 2004.

[6] X. Fang and M. Veeraraghavan, “Internetworking circuit and connectionless networks,” in *11th Int. Conf. Advanced Communication Technology*, 2009, pp. 63–68.

[7] H. Harai, H. Furukawa, K. Fujikawa, T. Miyazawa, and N. Wada, “Optical packet and circuit integrated networks and software defined networking extension,” *J. Lightwave Technol.*, vol. 32, no. 16, pp. 2751–2759, 2014.

[8] H. Furukawa, T. Miyazawa, N. Wada, and H. Harai, “Moving the boundary between wavelength resources in optical packet and circuit integrated ring network,” *Opt. Express*, vol. 22, no. 1, pp. 47–54, 2014.

[9] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, “OpenFlow: Enabling innovation in campus networks,” *Comput. Commun. Rev.*, vol. 38, no. 2, pp. 69–74, 2008.

[10] N. Cardwell, S. Savage, and T. Anderson, “Modeling TCP latency,” in *Proc. INFOCOM 19th Annu. Joint Conf. IEEE Computer and Communications Societies*, 2000, pp. 1742–1751.

[11] I. Rhee and L. Xu, “CUBIC: A new TCP-friendly high-speed TCP variant,” in *Proc. PFLDnet*, 2005.

[12] R. Hemenway, R. Grzybowski, C. Minkenberg, and R. Luijten, “Optical-packet-switched interconnect for supercomputer applications,” *J. Opt. Netw.*, vol. 3, no. 12, pp. 900–913, 2004.

[13] L. Zaninetti and M. Ferraro, “On the truncated Pareto distribution with applications,” *Central Eur. J. Phys.*, vol. 6, no. 1, pp. 1–6, 2008.

[14] “Transmission control protocol,” IETF RFC 793, Sept. 1981.

[15] X. Yuan, R. Melhem, R. Gupta, Y. Mei, and C. Qiao, “Distributed control protocols for wavelength reservation and their performance evaluation,” *Photon. Netw. Commun.*, vol. 1, no. 3, pp. 207–218, 1999.

[16] L. Berger, “Generalized multi-protocol label switching (GMPLS) signaling Resource ReSerVation Protocol-Traffic Engineering (RSVP-TE) extensions,” IETF RFC 3473, Jan. 2003.

[17] L. Berger, “Generalized multi-protocol label switching (GMPLS) signaling functional description,” IETF RFC 3471, Jan. 2003.

[18] O. Alparslan, S. Arakawa, and M. Murata, “Computing path blocking probability and delay in optical networks with retrial,” *J. Opt. Commun. Netw.*, vol. 5, no. 5, pp. 498–511, 2013.

[19] R. Ramaswami and K. Sivarajan, “Design of logical topologies for wavelength-routed optical networks,” *IEEE J. Sel. Areas Commun.*, vol. 14, no. 5, pp. 840–851, 1996.

[20] “The Network Simulator NS-2,” [Online]. Available: http://www.isi.edu/nsnam/ns/.

[21] Wolfram Research Inc., “*Mathematica 9.0*,” Champaign, Illinois: Wolfram Research 2012.

[22] X. Xiao, *Technical, Commercial and Regulatory Challenges of QoS: An Internet Service Model Perspective*. San Francisco, CA: Morgan Kaufmann, 2008.

[23] T.-W. Yeow, K. Law, and A. Goldenberg, “MEMS optical switches,” *IEEE Commun. Mag.*, vol. 39, no. 11, pp. 158–163, 2001.