Performance of Optical Packet Switches Based on Parametric Wavelength Converters

Nattapong Kitsuwan, Roberto Rojas-Cessa, Motoharu Matsuura, and Eiji Oki

Abstract—In an optical packet switch (OPS), input fibers carry multiple wavelengths, each of which carries a packet to one output fiber. As several wavelengths from different inputs could be destined to the same output fiber, one wavelength can be connected and the others remain disconnected, losing the carried packets. Because of the multiple wavelengths available at an output fiber, wavelength conversion in the OPS of the unconnected wavelengths into those available can increase the number of connections. A parametric wavelength converter (PWC) provides multi-channel wavelength conversion where wavelengths can be converted to another. A PWC uses a pump wavelength that can be flexibly chosen to define which wavelengths can be converted, defining the so-called wavelength conversion pairs. However, it is unknown which set of pump wavelengths, and therefore the set of connection pairs, should be selected to improve the OPS performance. In this paper, the number of PWCs in the OPS is reduced. This paper proposes a pump wavelength selection policy for an OPS that uses different pump wavelengths, one for each PWC, within an arbitrarily selected interval. This policy is called variety rich (VR) policy. This paper also introduces a non-wavelength blocking OPS (NWB-OPS) to make full use of PWCs. The switch performance is evaluated through computer simulation. The results show that the proposed policy with different pump wavelengths achieves the highest performance when compared to another of similar complexity. Furthermore, the performance study shows that small sizes of the interval to select a pump wavelength are more beneficial than larger ones.

Index Terms—Optical packet switch, wavelength division multiplexing, wavelength converter, pump wavelength, scheduling algorithm.

I. INTRODUCTION

Optical switches that support high bandwidth for high-speed networks are required as the demand for link speed and bandwidth keeps increasing. In an optical packet network with optical packet switches (OPSs) interconnected with optical fibers running wavelength division multiplexing (WDM), packets can be transmitted from source to destination without any optical-electrical-optical conversion [1]-[5].

In an OPS, an optical fiber carries several wavelengths to transmit packets. An input fiber may carry as many packets as wavelengths that could be destined to one or several output fibers. In an OPS, output contention might occur as multiple packets from different input fibers might be destined to a single output fiber using the same wavelength at the same time. From the containing packets, only one is forwarded to the output fiber and the others are dropped. To perform the selection of the packet (and wavelength), a switch configured by a scheduler can be used. In an OPS, the set of aggregated wavelengths coming from the input fiber is optically demultiplexed into individual wavelengths and each wavelength is assigned to an input port of the switch fabric that connects outputs to inputs. The outputs of the switch fabric are assigned to one wavelength each. Therefore, each input port of the switch fabric can be connected to an output port of the same wavelength. The set of individual wavelengths of an output fiber are then aggregated before they egress the switch. For a switch with \( N \) input/output fibers and \( W \) wavelengths per fiber, an \( NW \times NW \) switch fabric is used. In such switch fabric the scheduler performs the decision of interconnecting different inputs and outputs for each wavelength by performing a matching process (i.e., the assignment of one input to one output). When several inputs of the same wavelength have packets for the same output, it is said to be output contention.

Several methods for resolving output contention have been proposed. One of them is the use of optical buffering through fiber delay lines (FDLs) [6] to delay a packet for a specified amount of time while waiting for a future transmission. However, because the delay times that FDLs can provide depend on the fiber length, the delay times that FDLs can feasibly provide are small when they are compared to the delay times that a packet would have to wait before being forwarded. Another option is the use of deflection routing [7], which can reduce the need for buffering packets. In this approach, one packet is sent to the destined output while the others are sent to any available output as alternative routes. However, the network control for such approach might be complex.

An alternative strategy is to use wavelength converter (WC) [8]. In this approach, the wavelength carrying a packet that cannot be switched to its destined output because of output contention is converted to another available wavelength for the same output fiber. A full wavelength converter (FWC), which can convert a wavelength to any other wavelength, has a self-tunable probe light source [9]. A network using FWCs achieves the highest performance known in terms of packet loss rate if the number of FWCs is increased [10]. However, this makes this approach costly for practical applications. Furthermore, in some actual traffic conditions, the FWCs in an OPS may not be used, making the OPS unnecessarily

Manuscript received April 30, 2010; revised June 8, 2010.
This work was supported in part by the Okawa Foundation and the Support Center for Advanced Telecommunications Technology Research (SCAT).
Nattapong Kitsuwan and Eiji Oki are with the Dept. of Information and Communication Engineering, The University of Electro-Communications, Tokyo, Japan. Email: {kitsuwan, oki}@ice.ucc.ac.jp.
Roberto Rojas-Cessa is with the Dept. of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, NJ 07102, USA. Email: rojas@njit.edu. He visited The University of Electro-Communications, Tokyo, Japan. He was a Fellow of Japan Society for the Promotion of Science (JSPS).
Motoharu Matsuura is with The Center for Frontier Science and Engineering, The University of Electro-Communications, Tokyo, Japan. Email: matsuura@awcc.uec.ac.jp.
expensive. To decrease the high cost of FWCs, inputs can share a WC, where each input uses one wavelength. Several architectures with shared WC have been proposed, such as the share per node (SPN) approach [10]-[12].

![Diagram](https://via.placeholder.com/150)

Fig. 1. Example of wavelength conversion pairs in a PWC.

The shared-per-node with parametric wavelength converters (SPN-PWC) [13] is an example of an SPN switch. This switch requires a smaller number of converters than that in a switch using single-channel WCs. A parametric wavelength converter (PWC) [14] allows multiple wavelengths to be simultaneously converted from one to another. To define the wavelengths that can be converted, a pump wavelength, \( \lambda_p \), is applied to the PWC. In the PWC, \( \lambda_p \) is set in the middle wavelength, between the wavelength to be converted from, \( \lambda_w \), and the one converted onto, \( \lambda_w' \). This is defined by

\[
\lambda_w' = 2\lambda_p - \lambda_w
\]

As a conversion wavelength pair comprises the set of wavelengths converted from and converted onto, each wavelength can play either role, the source wavelength or the converted wavelength in a time slot. For example, if \( \lambda_p \) is set to convert \( \lambda_a \) to/from \( \lambda_b \), where \( \lambda_a \) and \( \lambda_b \) are transmission wavelengths.

\( \lambda_a \) can be selected to be the original wavelength and \( \lambda_b \) to be the converted wavelength or vice versa, but just one direction at a time. Furthermore, each pair in the set can have independent conversion direction at a given time. Figure 1 shows examples of the wavelength conversion pairs in a PWC. There are seven wavelengths in this example, \( \lambda_1 \) to \( \lambda_7 \). When \( \lambda_p = \lambda_4 \), the conversion pairs in the PWC are \( \lambda_1 \leftrightarrow \lambda_7, \lambda_2 \leftrightarrow \lambda_6 \), and \( \lambda_3 \leftrightarrow \lambda_5 \), which can be used in either direction. The set of conversion pairs using the same pump wavelength is called conversion pattern. Setting different pump wavelengths obtains different conversion patterns.

PWC is considered becoming feasible. A number of simultaneous multi wavelength conversions of over 30 channels has been reported using fiber [15] and LiNbO\(_3\) waveguide [16]. A high-capacity field transmission experiment using multi wavelength conversion has also been demonstrated [17]. Moreover, parametric process is fully transparent to various types of advanced modulation formats such as differential phase shift keying (DPSK) [18] and quadrature phase shift keying (QPSK) [19]. These studies show that guard bands can be provided with a suitable channel spacing. With this in mind, this paper assumes that guard bands are provided in the adopted channel spacing, such that they are considered provisioned in the remainder of this paper.

It has been claimed that setting the pump wavelength in the middle of the lowest and highest transmission wavelengths can achieve high switching performance improvement because \( \lambda_p \) provides the maximum number of conversion pairs in a PWC [13]. This approach uses a pump selection policy where all PWCs in the OPS use the same pump wavelengths and all of the pump wavelengths attempt to be set as close as possible to the middle of the lowest and highest transmission wavelengths. If this approach is applied to all PWCs in the OPS and all of the pump wavelengths are fixed in the middle, each wavelength can be converted to another single wavelength. This is, the number of conversion pairs is the largest, but there is no variety as one wavelength can be converted to only another wavelength. The number of different conversion pairs is the smallest. This policy is called a number rich (NR) policy. Therefore, this raises the following questions: What are the pump wavelengths in multiple PWCs that can improve the switching performance of an OPS and how much is the performance improved, while keeping the number of PWCs small?

This paper takes on these questions and proposes a policy to set pump wavelengths with different values such that a variety of connection pairs is provided. This is called a variety rich (VR) policy. The VR policy in an OPS with multiple PWCs uses a different pump wavelength in each PWC. It thus provides a different set of conversion wavelength pairs per PWC. The performance improvement that the VR policy has over the NR policy on an optical packet switch is investigated. In the VR policy, a different pump wavelength per PWC is selected within a bounded interval of transmission wavelengths. To perform the configuration of the switch fabric used in the OPS, an iterative matching algorithm that considers wavelengths of fibers as separate ports is adapted from those used in electronic input-queued packet switches. The selections in the matching algorithm are random based for simplicity. However, other selection policies can be used to improve the performance of the scheduling scheme, and therefore, that of the OPS. The numerical results indicate that the performance of the VR policy is dramatically improved compared with that of the NR policy.

The remainder of this paper is organized as follows. Section II describes switch models which are the wavelength blocking (WB-OPS) and non-wavelength blocking (NWB-OPS) with PWCs. Section III describes a conventional number rich (NR) policy. Section IV introduces a proposed variety rich (VR) policy. Section V introduces the iterative scheduling algorithm. Section VI shows the performance results. Section VII presents the conclusions.

II. SWITCH MODELS

As shown in Figure 2, a switch consists of \( N \) input and output fibers and \( M \) PWCs. Each fiber carries \( W \) different wavelengths.

In Figure 2(a), a switch fabric without any demultiplexer after wavelength conversion from PWCs is used. A request with a converted wavelength is directly sent from a PWC to an output port of the requested output fiber without any demultiplexer. To do this, a PWC output forwards requests carrying the different wavelengths to only one output fiber.
Although output ports using the different wavelength on other output fibers are available. We call this switch as wavelength blocking OPS (WB-OPS) [13].

As shown in Figure 2(b), a non-wavelength blocking (NWB-OPS) switch differs from the previously presented switch, WB-OPS, (shown in Figure 2(a)) which no demultiplexer at the output of PWCs. The NWB-OPS uses a \( (NW + MW) \times (NW + MW) \) switch fabric, demultiplexers at the output of PWCs, where each demultiplexed wavelength is in a one-to-one correspondence with an input port of the switch fabric. The individual wavelengths, coming through individual input ports of the switch fabric, are grouped together by an optical coupler before being forwarded to an output fiber. In this way, each converted wavelength can be connected to any output, making this switch no wavelength blocking. For example, the two different wavelengths coming from the output PWC in the figure can then be connected to two different outputs, as shown by the solid and dashed lines traversing the switch fabric.

To simplify the description in the remainder of this paper, instead of referring to a wavelength in wavelength units (e.g., meters), the wavelengths are referred to by index numbers, where each index has a one-to-one correspondence to a wavelength, and wavelengths in the wavelength transmission interval are equally spaced. Therefore, all transmission wavelengths (i.e., the ones carrying data) and \( \lambda_p \) can be expressed by index numbers. Here, \( m \) and \( w \) are the indexes of PWCs and transmission wavelengths, respectively, where \( 1 \leq m \leq M \) and \( 1 \leq w \leq W \).

In the remainder of this paper, it is assumed that packets have fixed size and the time dedicated to switch packets from inputs to outputs has therefore fixed duration, which is called time slot. The OPS switch performs the connection of wavelengths from the input fibers to the output fibers, through connections between the input and output ports of the switch fabrics and through connections from the inputs to the PWCs and from PWCs to the output ports, each time slot. The configuration of the switch fabric can be updated at time-slot basis.

Several wavelengths, each carrying a packet, can be simultaneously matched, and those that lose contention can be converted to another wavelength by PWCs. To define the wavelengths that can be converted, a continuous pump wavelength, \( \lambda_{p_m} \), is applied to \( PWC_m \). In a PWC, \( \lambda_{p_m} \) can be set in the middle wavelength, between the wavelength to be converted from, \( \lambda_w \), and the one converted onto, \( \lambda_w' \). This is defined by

\[
w' = 2p_m - w
\]

For the general case, the set of pump wavelengths is defined as \( \Lambda_p = \{ \lambda_{p_1}, \lambda_{p_2}, \ldots, \lambda_{p_m}, \ldots, \lambda_{p_M} \} \), where \( \lambda_{p_m} \) is the pump wavelength of the \( m \)th PWC.

### III. Number Rich Policy

The number rich (NR) policy is a conventional policy, which was adopted from the existing description in [13]. The \( M \) PWCs use the same pump wavelength, \( \lambda_{p_m} \), placed in the middle between the lowest transmission wavelength, \( \lambda_1 \), and the highest transmission wavelength, \( \lambda_W \), as shown in Figure 3. This is \( \lambda_{p_m} = \lambda_{p_1} = \lambda_{p_2} = \ldots = \lambda_{p_M} \). It also means that \( \lambda_w \) is converted to \( \lambda_{W-(m-1)} \), for any \( w \) in the transmission wavelength set. The index of \( \lambda_{p_m} \) is

\[
p_m = \frac{1 + \frac{W}{2}}{2}
\]

\( \lambda_{\lambda_1} \): Lowest transmission wavelength
\( \lambda_{\lambda_W} \): Highest transmission wavelength

\( PWC_1 \)

\( \lambda_{\lambda_1} \)

\( \lambda_{\lambda_W} \)

\( PWC_2 \)

\( \vdots \)

\( \lambda_{\lambda_1} \)

\( \lambda_{\lambda_W} \)

Fig. 3. Setting \( \lambda_{p_m} \) for \( M \) PWCs for NR policy.

In the NR policy, although setting the pump wavelength in the middle of transmission wavelengths for in every PWCs achieves the maximum number of conversion pairs, the variety of conversion pattern is limited to only one. Some requests from input ports may be blocked, even though the desired output fiber has some available wavelengths. This can occur because none of wavelength conversion pairs can support. Therefore, it is desirable that packets that lose the contention process be able to utilize the available wavelengths.

### IV. Variety Rich Policy

The variety rich (VR) policy is a proposed policy, where each PWC has a different \( \lambda_{p_m} \). The VR policy increases a variety of conversion patterns while the NR policy has only one conversion pattern. Figure 4 shows an example of variety of conversion patterns in a set of different pump wavelengths for PWCs. There are eight wavelengths in this
example, $\lambda_1$ to $\lambda_8$. When $\lambda_{Pm} = \lambda_2$, the conversion pair in the PWC is $\lambda_1 \leftrightarrow \lambda_3$, which can be used in either direction. When $\lambda_{Pm} = \lambda_3$, the conversion pairs are $\lambda_1 \leftrightarrow \lambda_5$ and $\lambda_2 \leftrightarrow \lambda_4$. This offers more opportunity to wavelengths of contending packets to be converted to available wavelengths at the desired output fibers.

In VR policy, each PWC has a different $\lambda_{Pm}$, which takes a value of a transmission wavelength in a sub-interval within the transmission wavelength interval or between contiguous transmission wavelengths. The interval where the pump wavelength can be placed is called pump wavelength interval, which has a lowest and highest pump wavelengths, $\lambda_l$ and $\lambda_h$, respectively. The pump wavelength interval is centered in the middle of the transmission wavelength interval. The pump interval size is the difference between the indexes of $\lambda_h$ and $\lambda_l$, $h - l$. The largest size of the pump interval is $W - 1$ and the minimum size of the pump interval is $\lceil \frac{M - 1}{2} \rceil$, where $\lceil x \rceil$ is the smallest integer greater than or equal to $x$. Let’s denote $\lambda_{P1} = \lambda_l$ and $\lambda_{PM} = \lambda_h$. The rules for selecting the pump wavelengths after the pump interval size is arbitrarily selected in this policy are the following:

- The pump wavelengths with the smallest and the largest indexes in pump wavelength interval are $l = p_1 = \frac{1 + W}{2} - \frac{h - l}{2}$ and $h = p_M = \frac{1 + W}{2} + \frac{h - l}{2}$. The relationship of these wavelengths and those in the transmission interval are:

$$\frac{l + h}{2} = \frac{1 + W}{2} \quad (4)$$

- The remaining pump wavelengths are placed between the lowest and the highest pump wavelengths, or $\lambda_l \leq \lambda_{Pm} \leq \lambda_h$. The pump wavelength can be set at a transmission wavelength or in the middle between each consecutive pair of transmission wavelengths. Let’s denote as $\delta_p$ to the inter-index space (which also corresponds to the inter-wavelength space), which is defined as:

$$\delta_p = \frac{h - l}{M - 1} \quad (5)$$

which indicates that the inter-wavelength spaces are equal. Then $\lambda_p$ for $m = 2, \ldots, M - 1$ is $\lambda_{Pm} = \lambda_l + (m - 1)\delta_p$.

Figure 5 shows the setting of the pump wavelengths for the VR policy.

Fig. 5. Setting $\lambda_{Pm}$ for $M$ PWCs for VR policy.

**V. SCHEDULING ALGORITHMS**

There are three possible points where contention might occur in this switch model. The first contention point is at the output port of the switch as there could be more than one packet from the different input fibers with the same wavelength destined to the same output fiber and only one packet can be forwarded to an output. In the scheduling scheme, an input sends request to the output. An output selects a request by random selection. The requests not selected by the output port are sent to the PWC that can convert their wavelengths to another unmatched wavelengths at the output fiber.

The second contention point is at the inputs of the PWCs. There are two types of contentsions. First, since the requests rejected by the output port are sent to the PWC with the same original wavelength, the PWC selects only one request with the same original wavelength to convert to another wavelength within its wavelength conversion pairs. Second, if there are requests from different original wavelengths to request the same conversion pair, the PWC selects only one original wavelength out of two to be converted. The unmatched requests are sent to other PWCs that can possibly perform the conversion in the next iteration.

The third contention point is again at output ports of the switch, after the original wavelength of the requests are converted from PWCs. The PWC that performed the wavelengths conversions forward a request to the destined output port (the same destination of the original wavelength) of the switch. Only one request with the same converted wavelength from a
PWC is selected as described in the matching scheme and the inputs that are the sources of the converted wavelengths are notified of the matching results.

A. Definitions

The terminology used in this paper is as follows.

- **i** The *i*th input port of the switch fabric, where $1 \leq i \leq N$.
- **j** The *j*th output port of the switch fabric, where $1 \leq j \leq N$.
- **RO(i, w, j)** Request flag of the *i*th input fiber at the *w*th wavelength to the *j*th output fiber.
- **RC(i, m, w, j, w')** Request flag of the *i*th input fiber at the *w*th wavelength to the *m*th PWC for converting from wavelength *w* to wavelength *w'* and destined to the *j*th output fiber.
- **I(i, w)** Input port of the *i*th fiber and the *w*th wavelength.
- **O(j, w)** Output port of the *j*th fiber and the *w*th wavelength.
- **CI(m, w)** Input port of the *m*th PWC and the *w*th wavelength.
- **CO(m, w')** Output port of the *m*th PWC and the *w'*th wavelength.

B. Random-based Scheduling Algorithm for NWB-OPS

At the beginning of each time slot, input and output ports of the switch fabric are unmatched. The algorithm consists of two phases, phase I and phase II. In phase I, matching between input and output links without PWCs is performed to resolve the first contention point, i.e., at output ports of the switch. Remaining unmatched requests from phase I are considered for matching in phase II. In phase II, multiple iterations are considered. At each iteration, matching between input and output links using PWCs is performed to resolve the second and third contention points, which occur at the input ports of the PWCs and at output ports of the switch after the original wavelength of the requests are converted from PWCs, respectively. Remaining unmatched requests at the end of the iteration are considered for matching at the next iteration, until the specified number of iterations is reached or there is no unmatched request. The following steps describe the procedure of the selection process.

Phase I: Direct matching between input and output links.

- **Step 1**: Input port $I(i, w)$ sends a request $RO(i, w, j)$ to output port $O(j, w)$ if there is a request $RO(i, w, j)$ at $I(i, w)$.
- **Step 2**: Output port $O(j, w)$ randomly selects $RO(i, w, j)$ with a uniform distribution among all those received. $O(j, w)$ sends a grant, indicating the selected input, to the selected input. The grant includes the information of unmatched (available) wavelengths at the output.

Phase II: Matching between input and output links using PWCs.

**The first iteration**

- **Step 1**: An input port randomly selects a PWC out of those that can convert the *w*th wavelength to an available *w'*th wavelength at the destined output fiber to send the unmatched request $RO(i, w, j)$ in Phase I. The selection of PWCs is determined by the policy adopted.
- **Step 2**: The input port $I(i, w)$ sends request $RC(i, m, w, j, w')$ to the selected PWC at $CI(m, w)$ port.
- **Step 3**: The PWC randomly selects wavelength conversion direction if there are requests from different wavelengths request to the same wavelength conversion pair at the same PWC. The wavelength conversion direction is converting from the *w*th to the *w'*th wavelength or from the *w'*th to the *w*th wavelength.
- **Step 4**: Port $CI(m, w)$ randomly selects one request $RC(i, m, w, j, w')$ among all those received. The other requests are dropped.
- **Step 5**: The output port of the *m*th PWC with the converted wavelength, $CO(m, w')$, sends the request to the destined output fiber at $O(j, w')$.
- **Step 6**: The output port $O(j, w')$ randomly selects a request among those received and sends back the grant to $CO(m, w')$ of the *m*th PWC. The *m*th PWC forward the grant to $I(i, w)$ via $CI(m, w)$.

**The qth iteration** is performed if $2 \leq q \leq q_{max}$, where $q_{max} \leq WM$.

- Unmatched requests in the previous iteration goes through the process again, starting from Step 1 in Phase II.

The time complexity of both conventional NR and proposed VR policies are the same, since they use the same algorithm. It depends on the number of iterations. The time complexity is $O(q_{max})$, where $q_{max}$ is the number of iterations.

To clarify the scheduling algorithm for NWB-OBS, an example is illustrated in Figure 6. Three input and output fibers with two PWCs, $PWC_1$ and $PWC_2$, are simply assumed. Each fiber contains four wavelengths, $\lambda_1, ..., \lambda_4$. The VR policy is applied in this example. Pump wavelength of $PWC_1$ is set at the middle between $\lambda_2$ and $\lambda_3$. The request wavelengths are able to be converted between $\lambda_1$ and $\lambda_4$ and between $\lambda_2$ and $\lambda_3$. Pump wavelength of $PWC_2$ is set at $\lambda_2$. Its wavelength conversion pairs are only between $\lambda_1$ and $\lambda_3$.

At input fiber 1, a request using wavelength $\lambda_1$ requests output fiber 2, $RO(1, 1, 2)$, while requests using wavelength $\lambda_2$ and $\lambda_4$ request output fiber 1, $RO(1, 2, 1)$ and $RO(1, 4, 1)$ respectively. At input fiber 2, requests using wavelength $\lambda_1$ and $\lambda_4$ request output fiber 2, $RO(2, 1, 2)$ and $RO(2, 2, 2)$ respectively, while a request using wavelength $\lambda_3$ requests the output fiber 1, $RO(2, 4, 1)$. At input fiber 3, requests using wavelength $\lambda_1$ and $\lambda_2$ request output fiber 2, $RO(3, 1, 2)$ and $RO(3, 2, 2)$ respectively, while a request using wavelength $\lambda_4$ requests the output fiber 1, $RO(3, 4, 1)$.

All of the requests, first, are sent by input ports to output ports, $O(j, w)$, at the requested *j*th output fiber using the same
original wavelength $\lambda_w$ as in Figure 6(a). Each output port randomly selects one request among those received. $O(1, 2)$ selects $RO(1, 2, 1)$, $O(1, 4)$ selects $RO(1, 4, 1)$, $O(2, 1)$ selects $RO(3, 1, 2)$ and $O(2, 2)$ selects $RO(3, 2, 2)$. An input port that remains unmatched sends a request $RC(i, m, w, j, w')$ to selected $m$th PWC that can convert from $w$th wavelength to an available $w'$th wavelength at the requested output fiber. The available output ports at fiber 1 and 2 are $O(1, 1)$, $O(1, 3)$, $O(2, 3)$ and $O(2, 4)$. As in Figure 6(b), input port $I(1, 1)$, then, sends $RC(1, 1, 1, 2, 4)$ to $PWC_1$ at $CI(1, 1)$. Input port $I(2, 1)$ sends $RC(2, 2, 1, 2, 3)$ to $PWC_2$ at $CI(2, 1)$. Input $I(2, 2)$ sends $RC(2, 1, 2, 2, 3)$ to $PWC_1$ at $CI(1, 2)$. Input $I(2, 4)$ sends $RC(2, 1, 4, 4, 1)$ to $PWC_1$ at $CI(1, 4)$. Input $I(3, 4)$ sends $RC(3, 1, 4, 1, 1)$ to $PWC_1$ at $CI(1, 4)$. At $PWC_1$, requests to convert from $\lambda_2$ to $\lambda_4$ and from $\lambda_4$ to $\lambda_1$ are received. $PWC_1$ randomly selects to convert only from $\lambda_4$ to $\lambda_1$. Request $RC(1, 1, 1, 2, 4)$ is, then, dropped as shown in Figure 6(c). There are two requests, $RC(2, 1, 4, 1, 1)$ and $RC(3, 1, 4, 1, 1)$, at $CI(1, 4)$. The $CI(1, 4)$ randomly selects $RC(2, 1, 4, 1, 1)$ and $RC(3, 1, 4, 1, 1)$ is dropped, as shown in Figure 6(d). Each request with converted wavelength is sent from output port of the PWC, $CO(m, w')$, to output port of the switch, $O(j, w)$. In Figure 6(d), $RC(2, 2, 1, 2, 3)$ is converted from $\lambda_1$ to $\lambda_3$ and it is sent from $CO(2, 3)$ to $O(2, 3)$. $RC(2, 1, 2, 2, 3)$ is converted from $\lambda_2$ to $\lambda_3$ and it is sent from $CO(1, 3)$ to $O(2, 3)$. $RC(2, 1, 4, 1, 1)$ is converted from $\lambda_4$ to $\lambda_1$ and it is sent out from $CO(1, 1)$ to $O(1, 1)$, as in Figure 6(e). The $O(2, 3)$ output port has two requests, $RC(2, 2, 1, 2, 3)$ and $RC(2, 1, 2, 2, 3)$. It randomly selects $RC(2, 2, 1, 2, 3)$, as in Figure 6(f). Therefore, $RC(2, 1, 2, 2, 3)$ is dropped. Three unmatched requests, $RC(1, 1, 1, 2, 4)$, $RC(2, 1, 2, 2, 3)$ and $RC(3, 1, 4, 1, 1)$, participate in the matching between input ports and PWCs in the next iteration.

C. Random-based Scheduling Algorithm for WB-OPS

At the beginning of each time slot, input and output ports of the switch fabric are unmatched. The algorithm consists of two phases in the same way as described in section V-B. Phase II for WB-OPS is different from that for NWB-OPS. The following steps describe the procedure of the selection process.

Phase I: Direct matching between input and output links.

- Step 1: Input port $I(i, w)$ sends a request $RO(i, w, j)$ to output port $O(j, w)$ if there is a request $RO(i, w, j)$ at $I(i, w)$.
- Step 2: Output port $O(j, w)$ randomly selects $RO(i, w, j)$ with a uniform distribution among all those received. $O(j, w)$ sends a grant, indicating the selected input, to the selected input. The grant includes the information of unmatched (available) wavelengths at the output.

Phase II: Matching between input and output links using PWCs.

The first iteration

- Step 1: An input port randomly selects a PWC out of those that can convert the $w$th wavelength to an available $w'$th wavelength at the destined output fiber to send the unmatched request $RO(i, w, j)$ in Phase I. The selection of PWCs is determined by the policy adopted.
- Step 2: The input port $I(i, w)$ sends request $RC(i, m, w, j, w')$ to the selected PWC at $CI(m, w)$ port.
- Step 3: The $m$th PWC randomly selects only one output fiber from the requests, $RC(i, m, w, j, w')$s. Requests destined for outputs that are not the selected $j$ are blocked. Notes that the wavelength conversion direction is ignored because the same wavelength conversion pair is not occurred since the $j$th output fiber is selected.
- Step 4: Port $CI(m, w)$ randomly selects one request $RC(i, m, w, j, w')$ among out of the requests that request to the selected $j$th output fiber.
- Step 5: The output port of $m$th PWC with the converted wavelength, $CO(m, w')$, sends the request to the selected $j$th output fiber at port $O(j, w')$. 

---

Fig. 6. An example of matching process of the scheduling algorithm for NWB-OPS.
Step 6: The output port \( O(j, w') \) randomly selects a request among those received and sends back the grant to \( m \)th PWC. The \( m \)th PWC forwards the grant to \( I(i, w) \) via port \( CI(m, w) \).

The \( q \)th iteration is performed if \( 2 \leq q \leq q_{\text{max}} \), where \( q_{\text{max}} \leq WM \).

- Unmatched requests in the previous iteration go through the process again, starting from Step 1 in Phase II.

The time complexity of both conventional NR and proposed VR policies are the same, since they use the same algorithm. It depends on the number of iterations. The time complexity is \( O(q_{\text{max}}) \), where \( q_{\text{max}} \) is the number of iterations.

![Diagram of PWCs and scheduling](image)

Fig. 7. An example process of the scheduling algorithm for WB-OPS.

Figure 7 shows an example of the scheduling algorithm for WB-OPS. The number of input and output fiber, transmission wavelengths, PWCs and the setting of the pump wavelength in each PWC are the same as in the example of NWB-OPS, in Figure 6. The difference is that the demultiplexer at each output port of PWC is not used. The requests after wavelength conversion from the same PWC are directly sent to only one selected output fiber.

The same requests as in the example for NWB-OBS, Figure 6, are assumed. All of requests are first sent by input ports to the requested \( j \)th output fiber with the same original wavelength as in Figure 7(a). Each output port that receives requests randomly selects only one request. These two steps are the same as in the example of NWB-OPS. Next, an input port that remains unmatched selects a PWC that is able to convert from \( \lambda_w \) to an available \( \lambda'_w \) at the requested output fiber. The input port \( I(i, w) \) sends a request \( RC(i, m, w, j, w') \) to the selected \( m \)th PWC at PWC’s input port, \( CI(m, w) \). \( RC(1, 1, 1, 2, 4) \) is sent from \( I(1, 1) \) to \( CI(1, 1) \), \( RC(2, 2, 1, 2, 3) \) is sent from \( I(2, 1) \) to \( CI(1, 2) \), \( RC(2, 1, 2, 2, 3) \) is sent from \( I(2, 2) \) to \( CI(1, 2) \), \( RC(2, 1, 4, 1, 1) \) is sent from \( I(2, 4) \) to \( CI(1, 4) \), and \( RC(3, 1, 4, 1, 1) \) is sent from \( I(3, 4) \) to \( CI(1, 4) \), as in Figure 7(b). The \( m \)th PWC randomly selects \( j \)th output fiber from among out of those requests, \( RC(i, m, w, j, w') \)s. In Figure 7(c), PWC \(_1 \) selects output fiber \( j = 1 \) and PWC \(_2 \) selects output fiber \( j = 2 \). Now, PWC \(_1 \) and PWC \(_2 \) is able to convert from \( w \)th wavelength to \( w' \)th wavelength that requests to \( j = 1 \) and \( j = 2 \) output fiber, respectively. Therefore, \( RC(2, 1, 4, 1, 1) \) and \( RC(3, 1, 4, 1, 1) \) are selected by PWC \(_1 \) and \( RC(2, 1, 2, 3) \) is selected by PWC \(_2 \). Since \( CI(1, 4) \) has two requests, PWC \(_1 \) selects only one request, randomly, which is \( RC(2, 1, 4, 1, 1) \) as shown in Figure 7(d). Converted wavelength of \( RC(2, 1, 4, 1, 1) \) is sent back to output fiber 1 and converted wavelength of \( RC(3, 1, 4, 1, 1) \) is sent back to output fiber 2, as Figure 7(e). Since there is no any conflict of wavelength at the output fiber, each output port sends back a grant to each input port \( I(i, w) \) along those selected PWC via \( CI(m, w) \).

VI. SIMULATION AND RESULTS

A NWB-OPS with \( N = 16 \) and \( W = 32 \), and the NR and VR policies are modeled for computer simulation. The performance of the NWB-OPS switch with these policies is estimated in terms of the packet loss rate. The packet loss rate is defined as the rate of the number of unsuccessful packets to be transmitted to the destined output ports in reference to the number of incoming packets at input ports. The performance is measured under different traffic load (\( \rho \)) values, number of iterations (\( q \)) in the scheduling scheme, the pump interval size, and the number of PWCs (\( M \)). The NR policy has no interval size as there is only one pump wavelength. In the following discussion, the pump interval size takes the values of 1, 2, and 3 wavelengths, which are the smallest pump interval sizes for \( M = \{3, 5, 7\} \), respectively, unless otherwise stated.

We assume that packet arrival at \( N \) input ports follows a Bernoulli process. When input traffic load is \( \rho \), an incoming packet arrives with probability \( \rho \), and there is no arrival with probability \( 1 - \rho \). The incoming packets are distributed uniformly to all output ports. The input traffic is assumed to be homogeneous, and it is distributed uniformly to all input ports [21], [22].

Figure 8 shows the packet loss rate of the NWB-OPS and WB-OPS switch with execution of the scheduling scheme...
for two iterations. This figure shows that the VR policy in NWB-OPS switch achieves lower packet loss rate than the NR policy in the same switch for the tested number of PWCs, which is the set $M = \{1, 3, 5, 7\}$, where each of them has the corresponding minimum pump interval size according to $M$. Furthermore, the figure shows that for the VR policy in WB-OPS switch, the packet loss rate decreases as $M$ increases. In contrast, the packet loss rate of the NR policy in NWB-OPS remains almost unchanged as $M$ changes. The reason for this is that even if $M$ increases, the variety of conversion pairs remains the same. In WB-OPS, either changing of $M$ or using different policy remains almost the same packet loss rate. It also shows that the NBW-OPS has lower packet loss rate than the WB-OPS switch.

NWB-OPS does not improve after the use of three PWCs, while the performance increases by increasing the number of PWCs with the VR policy. The NR policy cannot provide high performance no matter how many PWCs are used.

Figure 10 shows the packet loss rate when $M$ is increased in the VR policy for NWB-OPS. Packet loss rate decreases until the saturation point. It saturates at $M$ is 13 when $\rho$ is 0.1 and 0.2.

Figure 11 shows the performance of the VR policy under different pump interval sizes for NWB-OPS. The packet loss rate of the VR policy is shown in function of the traffic load, from 1 to 30 wavelengths. These two policies are simulated under a traffic load ($\rho = 0.1$), with $M = \{3, 5, 7\}$ for $N = 16$. The figure shows that the packet loss rate of the VR policy is lower than that of the NR policy. It can also be observed that the packet loss rate of the VR policy decreases as the number of PWC increases. This is expected as the number of PWCs increases from $M = 3$ to 7 with $N = 16$, the number of connection pairs increases. The figure also shows that the packet loss rate increases as the interval size increases. This is because as the interval size increases, the size of the conversion pairs decreases and that at the same time the variety of conversion pairs decreases. For a very small $M$, as shown in the figure for $M = 3$ and interval size equal to and larger than 25 wavelengths, the packet loss rate of the VR policy is higher than that of the NR policy because in the VR policy, $\lambda_p$s close to the edges (i.e., $\lambda_1$ and $\lambda_W$) of the transmission wavelength interval have a size of the conversion pair close to 0.

Figure 12 shows the packet loss rate of the NWB-OPS using the VR policy in function of the number of iterations performed in the scheduling scheme. The tested loads are $\rho = \{0.2, 0.3, 0.4, 0.5\}$. The figure shows that the packet loss rate reaches the lowest saturation point at three iterations for all the tested loads, except for $\rho = 0.2$, which saturates after four iterations. These results present some similarities to those results obtained in input-queued electronic switches with random-based selection schemes [20].

Packet loss rates under uniform and non-uniform are
compared using the following examined traffic model. The VR policy outperforms the NR policy for our examined model. The uniform traffic is occurred if the destinations are uniformly distributed to all output ports [21], [22]. Otherwise, the traffic is non-uniform [23]. For uniform and non-uniform traffics, packets arriving at \(N\) input ports follows a Bernoulli process, and the input traffic is assumed to be homogeneous, and it is distributed uniformly to all input ports. The non-uniform traffic is generated using a parameter called unbalance probability, \(\alpha\) [24]. Considering offered input load for each \(i\)th input, \(\rho\), which is an incoming packet arrival probability, the traffic load from the \(i\)th input to \(j\)th output, \(\rho_{i,j}\), is given by [25],

\[
\rho_{i,j} = \begin{cases} 
\rho \left( \alpha + \frac{1-\alpha}{N} \right) & \text{if } i = j \\
\rho \frac{1-\alpha}{N} & \text{otherwise}.
\end{cases}
\]  

(6)

The traffic is uniform distributed when \(\alpha\) is zero and the traffic is completely unbalanced when \(\alpha\) is one. Figure 13 indicates that the VR policy outperforms the NR policy with any \(\alpha\).

VII. CONCLUSIONS

This paper presented the improvement on the performance of an optical packet switch using parametric wavelength converters (PWCs) with pump wavelengths that produce a large variety of wavelength conversion pairs. Also, this paper introduced a non-wavelength blocking OPS, called NWB-OPS, where the PWCs can be fully utilized. This paper investigated how the switch performance, represented by the packet loss rate, is improved by the selection of the pump wavelengths for the PWCs and by the number of PWCs. This paper considered an existing number rich (NR) policy, where a single fixed pump wavelength, placed in the middle of the transmission wavelength interval, is used for all PWCs. This policy has the advantage of providing the largest number of conversion pairs, but also has the disadvantage of providing a single pattern of conversion pairs (or poor variety). The number of different conversion pairs is the smallest. To increase the conversion pair variety, this paper proposed a variety rich (VR) policy that uses different pump wavelengths, equally spaced in a defined pump wavelength interval, for each PWC. The size pump wavelength interval in this policy provides a larger variety of the conversion pairs but a smaller size of the conversion pairs.

Furthermore, to manage the switch configuration, the paper introduced an iterative scheduling algorithm used in combination with the pump selection policies to observe the switch performance. This performance was studied using computer simulation. The results show that the packet loss rate decreases as the number of PWCs increases using variety in the conversion pairs through the VR policy. The VR policy provides higher performance than the NR policy. In the VR policy, the packet loss rate is the lowest when the size of the interval to set the pump wavelength is the smallest, or \([\frac{M-1}{M}]\), where \(M\) is the number of PWCs, and the packet loss rate increases as this interval size increases. These results show that high performance switching can be obtained with a smaller number of PWCs using the VR policy than that with the NR policy. The performance increases by increasing the number of PWCs with the VR policy. However, the NR policy cannot provide high performance no matter how many PWCs are used. Therefore, the proposed approach in this paper shows a cost effective approach to the use of PWCs in an optical packet switch.
REFERENCES


Nattapong Kitsuwan received the B.E. and M.E. degree in Electrical Engineering (Telecommunication) from Mahanakorn University of Technology and King Mongkut’s institute of Technology, Ladkrabang, Thailand in 2000 and 2004 respectively. From 2002 to 2003, he was an exchange student at the University of Electro-Communications, Tokyo Japan where he did a research about packet switching. From 2003 to 2005, he was working for ROHM Integrated Semiconductor, Thailand company in a position of Information System Expert. He is currently working toward the Ph.D. degree in Information and Communications department at the University of Electro-Communications. His research focuses on optical networks, optical burst switching, optical packet switching, and scheduling algorithm.

Roberto Rojas-Cessa received the Ph.D. degree in Electrical Engineering from Polytechnic Institute of New York University, Brooklyn, NY. Currently, he is an Associate Professor at the Department of Electrical and Computer Engineering, New Jersey Institute of Technology. He has been involved in design and implementation of application-specific integrated-circuits (ASIC) for applications in biomedical equipment and high-speed computer communications, and in the development of high-performance and scalable packet switches and switch reliability. His research interests are high-speed switching and routing, fault tolerance, and quality-of-service networks. He enjoys serving for IEEE technical committees and as a panelist for U.S. National Science Foundation and the U.S. Department of Energy. He visited The University of Electro-Communications, Tokyo, Japan. He was a Fellow of Japan Society for the Promotion of Science (JSPS).

Motoharu Matsuura received the B.E., M.E and Ph.D. degrees all from the University of Electro-Communications, Tokyo, Japan. Now he is an assistant professor of the University of Electro-Communications. His research interests include optical signal processing and optical devices.
Eiji Oki is an Associate Professor of the University of Electro-Communications, Tokyo Japan. He received B.E. and M.E. degrees in Instrumentation Engineering and a Ph.D. degree in Electrical Engineering from Keio University, Yokohama, Japan, in 1991, 1993, and 1999, respectively. In 1993, he joined Nippon Telegraph and Telephone Corporation’s (NTT’s) Communication Switching Laboratories, Tokyo Japan. He has been researching multimedia-communication network architectures based on ATM techniques, traffic-control methods, and high-speed switching systems. From 2000 to 2001, he was a Visiting Scholar at Polytechnic University, Brooklyn, New York, where he was involved in designing tera-bit switch/router systems. He was engaged in researching and developing high-speed optical IP backbone networks with NTT Laboratories. He joined the University of Electro-Communications, Tokyo Japan, in July 2008. He has been active in standardization of Path Computation Element (PCE) and GMPLS in the IETF. He wrote more than ten IETF RFCs and drafts. He served as a Guest Co-Editor for the Special Issue on "Multi-Domain Optical Networks: Issues and Challenges;" June 2008, in IEEE Communications Magazine, Guest Co-Editor for Special Issue on Routing, "Path Computation and Traffic Engineering in Future Internet;" December 2007, in Journal of Communications and Networks, and Co-Chair of Technical Program Committee for Workshop on High-Performance Switching and Routing in 2006 and 2010, Track Co-Chair on Optical Networking for ICCCN 2009, and a Co-Chair of Technical Program Committee for International Conference on IP+Optical Network (iPOP 2010). Dr. Oki was the recipient of the 1998 Switching System Research Award and the 1999 Excellent Paper Award presented by IEICE, and the 2001 Asia-Pacific Outstanding Young Researcher Award presented by IEEE Communications Society for his contribution to broadband network, ATM, and optical IP technologies. He co-authored two books, “Broadband Packet Switching Technologies,” published by John Wiley, New York, in 2001 and “GMPLS Technologies,” published by RC Press, Boca Raton, in 2005. He is an IEEE Senior Member.