Supporting Quality-of-Service Scheduling in a TT-FR WDM System

Panagiotis G. Sarigiannidis, *Member, IEEE*, Sophia G. Petridou, *Member, IEEE*, Georgios I. Papadimitriou, *Senior Member, IEEE*, Mohammad S. Obaidat, *Fellow, IEEE*, and Andreas S. Pomportsis

***Abstract*—In this paper, our recent work interval-based orderly scheduling strategy (IOSS) [1] is extended by adding quality-of-service (QoS) provisioning. A revised medium access control (MAC) protocol is introduced, which supports priority-based QoS in wavelength division multiplexing (WDM) networks with star topology. The proposed interval-based prioritized orderly scheduling strategy (IPOSS) operates in a distributed manner, and has the capability of handling channel collision and destinations conflicts in order to provide a collision-free scheduling scheme. Each node in the network is equipped with a tunable transmitter and a fixed receiver, assuring that the scheme is scalable with respect to the number of nodes and channels. Also, each node may accept high- and low-priority packet arrivals. IPOSS favors high-priority packets, without regarding packets' length or packets' destination. Moreover, the proposed scheme differentiates the packet's schedule order by prioritizing the long-length over the short-length packets. It is found that the adopted access control scheme achieves a critically high throughput-delay performance for real-time traffic. Furthermore, IPOSS presents a little bit improved throughput performance than IOSS scheme, since it handles in a different way the case in which two or more requests demand the same amount of transmission time. The traffic involved in the simulation results follows Bernoulli and Poisson distribution, regarding the arrival of the requests.**

***Index Terms*—Reservation, scheduling, quality-of-service (QoS), wavelength division multiplexing (WDM) star networks.**

I. Introduction

OPTICAL technology comes to satisfy the increasing number of Internet users along with the demanding network applications, since optical fibers offer radically higher bandwidth than alternative transmission media [2], [3]. More specifically, optical fiber technology can untie the capacity problem because of its great capabilities, such as the huge bandwidth, the low signal attenuation, the immunity to electromagnetic interferences, the high security of signal, the absence of crosstalk, the low-signal distortion, the low-power requirement, the low material usage, and the high electrical resistance [4], [5]. The future Internet may be analyzed as a three-level hierarchy, consisting of backbone networks, metropolitan area networks, and local area networks (LANs) [6]. LANs carry the data from and to the individual users. In particular, advanced LAN technologies such as Gigabit Ethernet, xDSL, and multi-channel packet switched wavelength division multiplexing (WDM) networks provide increasing amount of bandwidth. The designed packet-switched WDM metropolitan or local networks fall into two main categories: networks with a physical star topology and networks with a physical ring topology. Star networks come in two implementations, namely the broadcast-and-select single-hop networks based on a wavelength-insensitive passive star coupler (PSC) and wavelength-routing networks based on a wavelength-selective arrayed-waveguide grating.

A very convenient and effective way of realizing a local WDM optical network is the architecture of broadcast-and-select, according to which the connection of each node with each other is achieved through a PSC. A remarkable attribute of this class of networks is that they can easily provide broadcast, multicast and unicast services [7]. A likely form that the overall network might take in the future is a combination of broadcast-and-select LANs interconnected by a wavelength routing network. According to a broadcast-and-select architecture, each node transmits its data to the PSC through one of the available channels, using a transmitter. Each transmitted data from all the nodes of the network is combined in the PSC and sent forward to all the nodes of the network through a two-way optical fiber. Optical filters accept the transmitted data and select the desired signal sent by the star. The PSC is a broadcast device, so a signal that is inserted on a given wavelength from an input fiber port will have its power equally divided among all output ports on the same wavelength. Each node has at least one transmitter and one receiver. Both the transmitter and the receiver can be either fixed or tunable. The tunable transmitter can be tuned in any channel of the network in order to transmit the flow of data. Respectively, the tuned receiver can accept data from any channel of the network. On the other side, the fixed transmitter and the fixed receiver can send or receive data only in a predetermined transmission channel. In this paper, a single hop broadcast-and-select star LAN with one tunable transmitter and one fixed receiver (TT-FR) per node is realized as depicted in Fig. 1.

Various demanding applications (e.g., video-on-demand, direct banking data, urgent alarm signals, multimedia conferencing, etc.) require different type of services and need application specific quality of service (QoS) to meet the required standards [8]. To provide end-to-end QoS not only the backbone network but also the LAN must support some kind

Manuscript received April 24, 2008; revised September 16, 2008. First published December 09, 2008; current version published December 31, 2008.

P.G. Sarigiannidis is with the Department of Engineering Informatics and Telecommunications, University of Western Macedonia, 50100 Kozani, Greece.

S. G. Petridou, G. I. Papadimitriou, and A. S. Pomportsis are with the Department of Informatics, Aristotle University, 54124 Thessaloniki, Greece.

M. S. Obaidat is with the Department of Computer Science, Monmouth University, West Long Branch, NJ 07764 USA (e-mail: obaidat@monmouth.edu).

Digital Object Identifier 10.1109/JSYST.2008.2009207

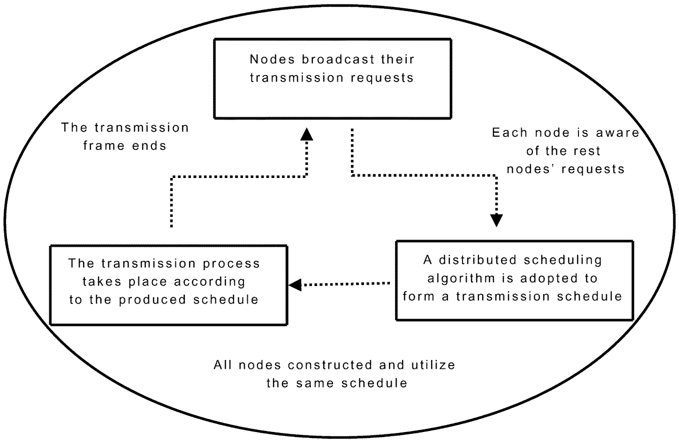


Fig. 1. TT-FR system nature.

of QoS. For that purpose, a MAC protocol is needed to support QoS in conjunction with a collision free scheme. In recent years, many MAC protocols have been proposed for WDM LAN based on star or ring. However, if we focus on a reservation based distributed network with a TT-FR architecture (absence of control channel) then we can observe that there is a lack of MAC protocols supporting QoS. In this paper, a revised scheduling scheme is introduced in order to meet the QoS requirements in a TT-FR optical WDM star network. The proposed interval-based prioritized orderly scheduling strategy (IPOSS) protocol falls in pretransmission coordination-based category, due to the fact that the transmission schedule is dynamic and depends on the network traffic. A scheduling algorithm is applied to construct the service order and the transmission program, known as transmission schedule.

The proposed algorithm manages to schedule real-time traffic, by prioritizing the real traffic data packets, which have high priority, over the non-real traffic data (normal data), which have low priority. In this way, an application such as an urgent alarm message or teleconference data generates high-priority packets, while a file transfer or a text message produces low priority ones. The new scheduling strategy comes as an improvement of the scheduling algorithm proposed in [1]. The novelty of IPOSS is that it supports QoS, realized with high- and low-priority packets. In practice, it differentiates packets' schedule order, since high-priority packets precedes the low-priority ones in transmission procedure. At the same time, IPOSS adopts the strategy of linear search with equality examination (ls-ee), which is an improvement of linear search for K elements (K-ls), proposed in [1].

According to ls-ee strategy, long-length requests are prioritized over the short-length ones, in such a way that a sorting process is avoided for complexity reasons. Furthermore, IPOSS determines the maximum length request of each reservation phase and then modifies the service order of each request, by giving priority to the most demanding requests. The new strategy also considers the case that two or more requests demand the same transmission time. In practice, while K-ls randomly selects among the competitive requests, the novel ls-ee considers two decision vectors, namely the node time vector (NTV) and the channel time vector (CTV). These vectors indicate the earliest time availability of each channel for the node whose request is being processed. The final choice among the competitive requests is based on these vectors. This schedule strategy allows a better usage of channels' utilization, since the possibility of finding an appropriate interval to schedule short-length requests is increased. Overall, IPOSS keeps the following scheduling order: high-priority long-length requests, high-priority short length requests, low-priority long-length requests and low priority short-length requests. If two or more equal-priority requests demand the same amount of transmission time, IPOSS exploits the information of NTV and CTV and schedules the request with the minimum scheduling delay. In other words, the request that can be scheduled earliest is favored among equal-priority requests.

The remainder of this paper is organized as follows. Section III provides the network's description, while Section II presents related reservation protocols for WDM star networks. Section IV presents the proposed scheduling algorithm, while Section V discusses the simulation results. Conclusions are given in Section VI.

II. Related Reservation Protocols

Various scheduling schemes have been proposed for WDM star networks with tunable transmitters and fixed receivers [1], [9], [10], [11]–[14]. However, none of these protocols support QoS. Online interval-based scheduling (OIS) [9] is a simple and practical online scheduling algorithm, operating with simple functions in order to construct the schedule as quick as possible. OIS presents very low computational complexity, i.e., $O(nw^2k)$ (the lowest time complexity of TT-FR family protocols). According to OIS, each node maintains a list of time intervals that are available on every data channel. More specifically, the algorithm maintains two sets of intervals, one for each channel and another for the node whose reservation is currently being scheduled. The interval list per channel or current node represents the unallocated time on that channel or node. Although OIS has many advantages, e.g., very low time complexity, collision free, and simple hardware implementation, it lacks in efficiency.

Predictive Online Scheduling Algorithm (POSA) [10], which is essentially an extension to OIS, introduces a prediction system, which reduces the computation time of the scheduling process by means of predicting the requests of each node for the following frame. As a result of applying POSA, during the prediction period, the scheduling algorithm has more time to compute every schedule. However, POSA brings about some performance improvement, if the average duration of the control and data phases is at least equal to the time needed for predicting reservations and computing the corresponding schedule. Finally, it must be mentioned that POSA uses the same algorithm as OIS to construct the scheduling matrix and, therefore, it produces a schedule with low channel utilization and low schedule efficiency.

The interval-based orderly scheduling strategy (IOSS) [1] is an improved edition of OIS and POSA algorithms. It attempts to produce a shorter schedule, an action leading to an efficient schedule. POSA's prediction mechanism is optional. That means that IOSS could also function without a prediction mechanism, if, for example, it is difficult to predict the traffic prototype of the network (in case that nodes' requests appear at a totally random order) or if the prediction mechanism for a variety of reasons, cannot predict a great number of correct requests (i.e., if it has low performance and accuracy). The proposed IOSS protocol reorganizes the service order of the requests per node and per channel based on the value of each entry of the demand matrix, applying the K-ls strategy. Thus, IOSS uses K-ls strategy in order to put the requests in order, without the usage of a sorting process, using a set of linear searches only. The complexity of IOSS algorithm, using the strategy of linear search for k elements, is less than the complexity of OIS algorithm.

III. Network Description

It is assumed a single-hop WDM LAN with broadcast-and-select architecture, consisting of n nodes, which are connected in a PSC via a two-way optical fiber [15], [16], and w data channels (wavelengths), where $n \geq w$, which are of the same capacity [17]. Generally, in an n-node optical network [2] the most effective communication structure is achieved when each node has a tunable transmitter and a tunable receiver (TT-TR system) in combination with n channels, one per node [18]. However such a scheme has two critical demerits: the network may utilize only a few channels due to technological constraints, while the realization of a network consisting of equal number of nodes and channels does not co-exit with the financial standards. As a result, the solution of a TT-FR system seems to go along with the current technological and financial developments [1]. In the assumed TT-FR system time is slotted and the transmission is synchronous. Also, each packet is transmitted in time equal to a timeslot and all nodes in the network are assumed to be synchronized by using a common clock.

In the previously discussed TT-FR implementation, depicted in Fig. 2, transmission is organized into frames, where each frame consists of a reservation (or control) phase and a data phase [19]. The overall process is depicted in Fig. 3. More specifically, during the reservation phase, each source node is assigned a unique control slot for broadcasting its control packet to all channels by means of its tunable transmitter (TDM access). Control packets are received by all nodes on their corresponding home channel by means of their fixed receiver and are assumed to make reservations for the data phase. The n nodes include in their control packets the priority information of their data packets, i.e., $p_r = 1$ for high-priority packets and $p_r = 0$ for low-priority packets, while they also send their requests to the common data channels. Nodes' requests are formed as variable-length messages consisting of one or more fixed-length data packets and time is divided into timeslots (data slots), where each data packet is transmitted in time equal to a timeslot. The duration of data timeslots is larger compared to that of control timeslots. Real-time and non-real-time requests are recorded in the $n \times wD_h$ and D_l demand matrices, respectively, where $d_h(i,j)(d_l(i,j))$ element, $i = 1, \ldots, n$ and $j = 1, \ldots, w$, indicates the number of high-priority (low-priority) data packets at node u_i that are destined for channel λ_j. Based on D_h and D_l, D can be defined as $D_h + D_l$, where $d(i,j)$ element represents the total number of data packets at node u_i that are destined for channel λ_j. The proposed IPOSS

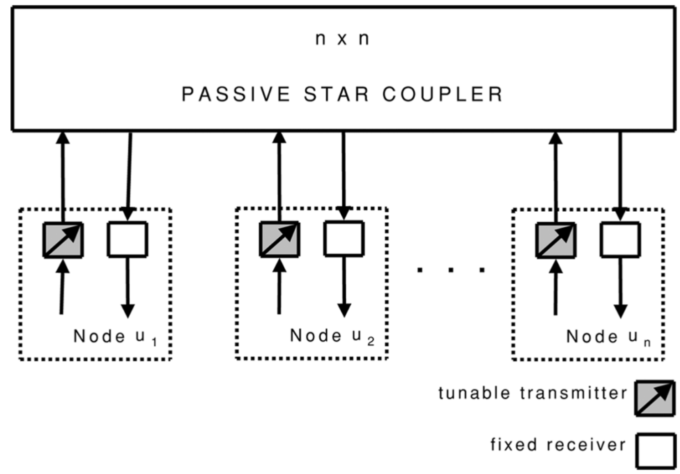


Fig. 2. Network topology.

operates in conjunction with a distributed scheduling algorithm and produces the $w \times t$ scheduling matrix S, where t denotes the length of the schedule in timeslots. Each $s(i,j)$ element, $i = 1, \ldots, w$ and $j = 1, \ldots, t$, represents the node that transmits on channel λ_i during the timeslot l_j. During the data phase the packets' transmission takes place according to the matrix S which was built during the reservation phase. At the end of reservation phase each node collects the total requests (for all channels). Then they run the same scheduling algorithm and calculate the exact transmission schedule. All the previous notation is summarized in Table I.

IV. Proposed Algorithm

IOSS scheme does not support QoS and, moreover, it randomly selects among equal-length requests, regarding the operation of K-ls strategy. The IPOSS scheduling scheme is proposed in order to improve the aforementioned IOSS algorithm, introducing a new improved scheduling strategy, named ls-ee. At the same time, QoS is provided.

In particular, IPOSS provides a new approach regarding the quite possible case that a number of requests may be of equal length. IPOSS adopts ls-ee, in order to handle these requests. According to ls-ee strategy, two vectors, namely the NTV and CTV vectors, one for each node and one for each channel are maintained. The $\mathrm{NTV}(i)$ element, $i = 1, \ldots, n$, indicates the earliest available time at which the node u_i will be available for transmission, while the $\mathrm{CTV}(j)$ element, $j = 1, \ldots, w$, denotes the earliest available time at which the channel λ_j will be available for transmission. NTV and CTV vectors are initialized at the start of each transmission frame (set equal to zero) and they are updated for each scheduled request. If two or more requests are of equal length, then IPOSS computes the maximum value between NTV and CTV, namely MaxV (max value) for each request and it finally chooses the request with the minimum MaxV value.

More specifically, IPOSS algorithm consists of two main phases, as depicted in Algorithm 1. During the first phase, ls-ee strategy is applied and a reordered demand matrix is returned. This action is operated twice, one for the low-priority requests and two for the high-priority ones. Of course, if the network

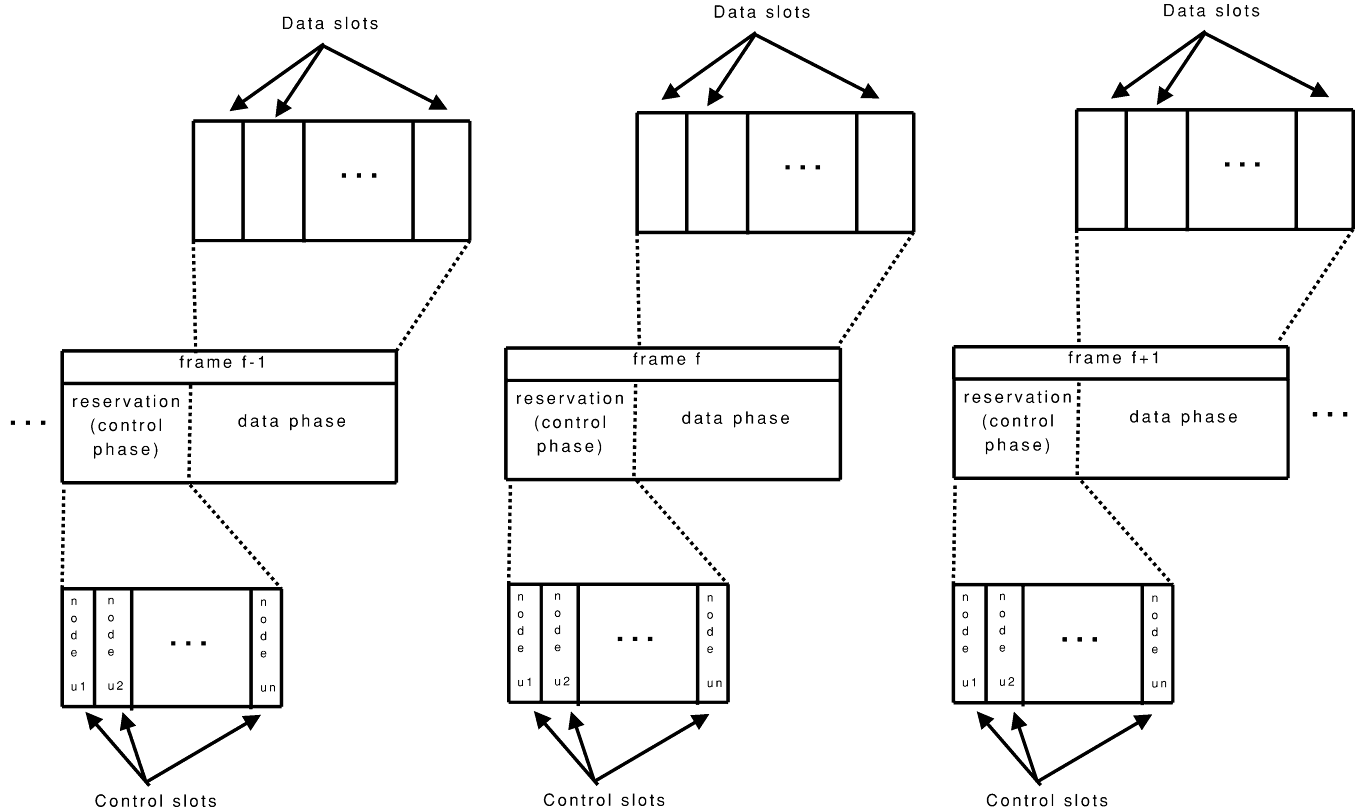


Fig. 3. Reservation and data phase.

TABLE I
BASIC SYMBOLS' NOTATION

Symbol	Description
n, w	Number of nodes and data channels
$U = \{u_1, \ldots, u_n\}$	Set of network's nodes
$\Lambda = \{\lambda_1, \ldots, \lambda_w\}$	Set of data channels
D	$n \times w$ demand matrix
D_h	$n \times w$ demand matrix of high-priority packets
D_l	$n \times w$ demand matrix of low-priority packets
NTV	Node time vector
CTV	Channel time vector
$maxV$	max value
t	Schedule's length in timeslots
$L = \{l_1, \ldots, l_t\}$	Set of timeslots
S	$w \times t$ scheduling matrix

configuration supports more priority levels then ls-ee is applied once for each level. Next, the schedule formation takes place. IPOSS constructs the transmission schedule according to the reordered demand matrices.

First of all, ls-ee strategy finds the request with the maximum length for each priority level. If K_h and K_l denotes the maximum length request for the high- and low-priority elements of the demand matrices D_h and D_l, respectively, then, ls-ee seek for the K_h or K_l different numbers just once (excepting zero). In other words, IPOSS performs $K_h + K_l$ linear searches to form the two reordered lists σ_h and σ_l. The search order is the following: initially the value of K_h (or K_l), which is the maximum entry of D_h (or D_l) is searched. The search returns all the values of K_h (or K_l) found in matrix D_h (or D_l) and more precisely, returns the position where the elements were found (the number of node and the number of channel of the referring request). So, if while searching for the value K_h (or K_l) the algorithm finds that the value is on the D_h (or D_l) matrix z times, where $z < nw$, then it returns the positions found, i.e., $D_h(i_1, j_1), D_h(i_2, j_2), \ldots, D_h(i_z, j_z)$ (or $D_l(i_1, j_1), D_l(i_2, j_2), \ldots, D_l(i_z, j_z)$), where $i_1, i_2, \ldots, i_z \in U$ and $j_1, j_2, \ldots, j_z \in \Lambda$. These values are stored in the σ_h (or σ_l). If two or more requests are of equal length then NTV and CTV vectors are computed and the request with the minimum $\max V$ is selected. Continuously, the selected request is stored in the reordered list σ_h (or σ_l) and is removed from the demand matrix D_h (or D_l).

Afterwards, the algorithm searches for the next biggest value, which is equal to $K_h - 1$ (or $K_l - 1$), in the same way. Next, the algorithm completes the process of linear searching for the requests with length equal to value 1. There is no point in searching for zero value, since it would not have practical interest on the final schedule matrix, because the specific node, on the specific channel does not have any packets to transmit. IPOSS algorithm completes the formation of the σ_h and σ_l lists in $O(nwK_h + nwK_l)$ time. It is obvious that the complexity of IPOSS algorithm, using the ls-ee strategy, is less than the complexity of OIS algorithm which is in $O(nw^2k)$ time, where k is the maximum value of $D_h + D_l$ matrix.

Algorithm 1: The IPOSS Flow Control

1: Begin frame
2: Initialize NTV and CTV vectors
3: Initialize w intervals (one for each channel)

Initialize n intervals (one for each node)
*/*Phase 1: ls-ee strategy begins*/*
Find the high-priority request with the maximum value, K_h
Reorder the high-priority requests and set the (reordered) service list σ_h
if two or more high-priority requests are of equal length **then**
 select the request with the minimum MaxV
end if
Find the low-priority request with the maximum value, K_l
Reorder the low-priority requests and set the (reordered) service list σ_l
if two or more low-priority requests are of equal length **then**
 select the request with the minimum MaxV
end if
*/*Phase 1: ls-ee strategy end*/*
*/*Phase 2: Scheduling formation begins*/*
if the σ_h list is not empty **then**
 Select the first entry of σ_h, let $D_h(i,j)$ be the first entry, $i \in U, j \in \Lambda$
 Find a suitable time space for the $D_h(i,j)$ request of node n_i on channel w_j, starting from the beginning of frame (timeslot 0), so that there is no other node scheduled for this space, for this channel and there is no schedule for node n_i on other channel
 Delete $D_h(i,j)$ from σ_h
 Update the interval of channel w_j
 Update the interval of node n_i
 Update NTV and CTV vectors
end if
if the σ_l list is not empty **then**
 Select the first entry of σ_l, let $D_l(i,j)$ be the first entry, $i \in U, j \in \Lambda$
 Find a suitable time space for the $D_l(i,j)$ request of node n_i on channel w_j, starting from the beginning of frame (timeslot 0), so that there is no other node scheduled for this space, for this channel and there is no schedule for node n_i on other channel
 Delete $D_l(i,j)$ from σ_l
 Update the interval of channel w_j
 Update the interval of node n_i
 Update NTV and CTV vectors
end if
*/*Phase 2: Scheduling formation ends*/*
End frame

For example, let us suppose that during a transmission frame three requests, namely the $d_l(1,3), d_l(2,1)$, and $d_l(3,3)$ have the same value of four data packets. In that case, nodes u_1, u_2 and u_3 compete each other for transmission on channels λ_3, λ_1, and λ_3, respectively. In that case, IOSS would randomly select one among the aforementioned requests. On the other hand, IPOSS applies the ls-ee strategy and computes the NTV and CTV vectors of each request. The $d_l(1,3)$ request is applied from node u_1 to channel λ_3, the $d_l(2,1)$ request comes from node u_2 and destines for channel λ_1, while $d_l(3,3)$ request comes from node u_3 and demands transmission on channel λ_3. Let us suppose that $\mathrm{NTV}(1) = 5, \mathrm{NTV}(2) = 4$, and $\mathrm{NTV}(3) = 3$, whereas $\mathrm{CTV}(1) = 3, \mathrm{CTV}(2) = 2$ and $\mathrm{CTV}(3) = 3$. It holds that IPOSS will select $d_l(3,3)$ for transmission, since this request has the minimum MaxV

$$\max V(d_l(1,3)) = \max(\mathrm{NTV}(1), \mathrm{CTV}(3)) = \max(5,3) = 5$$
$$\max V(d_l(2,1)) = \max(\mathrm{NTV}(2), \mathrm{CTV}(1)) = \max(4,3) = 4$$
$$\max V(d_l(3,3)) = \max(\mathrm{NTV}(3), \mathrm{CTV}(3)) = \max(3,3) = 3.$$

Finally, if two or more requests have the same value and the same minimum MaxV then ls-ee strategy choices the request with the minimum index (i.e., the minimum number of node or/and the minimum number of channel) [20].

Since IOSS does not support real-time traffic, the proposed IPOSS comes to cover this issue, by applying a handling method in order to satisfy the demanding needs of real-time traffic. IPOSS supports QoS, meaning that it handles real-time traffic, co-existing with non-real-time traffic. In this manner data packets are generated in two modes: high-priority packets, carrying real-time traffic, and low-priority ones, carrying non-real-time traffic. While IOSS does not differentiates high- and low-priority packets, IPOSS schedules high-priority packets prior to low-priority ones. In other words, high-priority packets are scheduled first, independently of their length. Once the high-priority packets' schedule is completed, IPOSS begins to schedule the low-priority ones (at the same schedule matrix). It is clear that ls-ee strategy is executed twice; firstly, it is executed regarding the high-priority packets' schedule and, secondly, it is executed regarding the low-priority packets' schedule. Consequently, high-priority packets are scheduled prior to low-priority ones. If two or more equal-priority requests demand the same amount of transmission time, IPOSS selects the request with minimum $\max V$. In this way, a more effective schedule is produced and QoS is supported, giving priority to real-traffic packets.

A. Numerical Example

This section provides a numerical example which illustrates the IPOSS scheme and the way that it applies the ls-ee strategy and supports QoS. It is assumed that the WDM star network consists of $n = 4$ nodes, i.e., $U = (u_1, u_2, u_3, u_4)$ and $w = 2$ data channels, i.e., $\Lambda = (\lambda_1, \lambda_2)$. Given these parameters, the following 4×2 demand matrix D could represent the aggregate network traffic

$$D = \begin{pmatrix} 3 & 2 \\ 2 & 4 \\ 1 & 5 \\ 4 & 4 \end{pmatrix}$$

while the following 4×2 demand matrices D_h and D_l could describe the real- and non-real-time traffic, respectively. Obviously, it holds that $D = D_h + D_l$

TABLE II
SCHEDULING MATRIX S PRODUCED BY IPOSS

	Timeslots														
	l_1	l_2	l_3	l_4	l_5	l_6	l_7	l_8	l_9	l_{10}	l_{11}	l_{12}	l_{13}	l_{14}	l_{15}
λ_1	$\mathbf{u_4}$	$\mathbf{u_4}$	$\mathbf{u_1}$	u_1	u_1	$\mathbf{u_2}$	u_2	u_3		u_4	u_4				
λ_2	$\mathbf{u_3}$	$\mathbf{u_3}$	$\mathbf{u_3}$	$\mathbf{u_2}$	$\mathbf{u_2}$	$\mathbf{u_4}$	u_4	u_4	u_4	u_1	u_1	u_2	u_2	u_3	u_3

$$D = D_h + D_l = \begin{pmatrix} 1 & 0 \\ 1 & 2 \\ 0 & 3 \\ 2 & 1 \end{pmatrix} + \begin{pmatrix} 2 & 2 \\ 1 & 2 \\ 1 & 2 \\ 2 & 3 \end{pmatrix}.$$

Example 1: In the previous demand matrices, the fact that $d(4,2) = 4$ means that node u_4 requests four packets on channel λ_2. Three out of these four packets are of low-priority, since $d_l(4,2) = 3$, while the one rest packet is of high-priority, since $d_h(4,2) = 1$. □

As it has already been mentioned, IPOSS begins to construct the scheduling matrix handing high-priority packets, i.e., the matrix D_h. The ls-ee strategy is applied and at a first stage the following scheduling matrix is formed.

	Timeslots		
	l_1	l_2	l_3
λ_1			
λ_2	$\mathbf{u_3}$	$\mathbf{u_3}$	$\mathbf{u_3}$

It is clear that the three packets of node u_3 destined for channel λ_2 are favored among all high-priority packets. Then, IPOSS proceeds to the scheduling of $d_h(2,2)$ and $d_h(4,1)$ requests, which are of the same length. At this stage, a decision should be made about the previous packets' priority, which will be based on NTV and CTV vectors. For $d_h(2,2)$, it holds that, $\mathrm{NTV}(2) = 1$, since node u_2 becomes available for transmission at l_1 timeslot, while $\mathrm{CTV}(2) = 4$, because the channel λ_2 becomes available at l_4 timeslot. Thus, $\max V(d_h(2,2)) = 4$. Following the same logic, $\max V(d_h(4,1)) = \max(\mathrm{NTV}(4), \mathrm{CTV}(1)) = 1$. Given that the request $d_h(4,1)$ has the minimum $\max V$, it holds that it is scheduled prior to $d_4(2,2)$. At this stage, the scheduling matrix is updated as follows.

	Timeslots				
	l_1	l_2	l_3	l_4	l_5
λ_1	$\mathbf{u_4}$	$\mathbf{u_4}$			
λ_2	$\mathbf{u_3}$	$\mathbf{u_3}$	$\mathbf{u_3}$	$\mathbf{u_2}$	$\mathbf{u_2}$

The remaining high-priority requests, i.e., $d_h(1,1), d_h(2,1)$, and $d_h(4,2)$, are also of the same length and will be scheduled according to the following order: $d_h(1,1), d_h(2,1)$, and $d_h(4,2)$. This is due to the fact that $\max V(d_h(1,1)) = \max(\mathrm{NTV}(1), \mathrm{CTV}(1)) = 3, \max V(d_h(2,1)) = \max(\mathrm{NTV}(2), \mathrm{CTV}(1)) = 6$, while $\max V(d_h(4,2)) = \max(\mathrm{NTV}(4), \mathrm{CTV}(2)) = 6$. Thus, at the end of D_h processing, the scheduling matrix is formed as follows.

TABLE III
SCHEDULING MATRIX S PRODUCED BY IOSS

	Timeslots														
	l_1	l_2	l_3	l_4	l_5	l_6	l_7	l_8	l_9	l_{10}	l_{11}	l_{12}	l_{13}	l_{14}	l_{15}
λ_1	$\mathbf{u_4}$	u_4	u_4	$\mathbf{u_4}$	u_1	u_1	$\mathbf{u_1}$			$\mathbf{u_2}$	u_2	u_3			
λ_2	$\mathbf{u_3}$	u_3	$\mathbf{u_3}$	u_3	$\mathbf{u_3}$	$\mathbf{u_2}$	u_2	$\mathbf{u_2}$	u_2	$\mathbf{u_4}$	u_4	u_4	u_4	u_1	u_1

	Timeslots					
	l_1	l_2	l_3	l_4	l_5	l_6
λ_1	$\mathbf{u_4}$	$\mathbf{u_4}$	$\mathbf{u_1}$			$\mathbf{u_2}$
λ_2	$\mathbf{u_3}$	$\mathbf{u_3}$	$\mathbf{u_3}$	$\mathbf{u_2}$	$\mathbf{u_2}$	$\mathbf{u_4}$

Once the IPOSS handles the high-priority packets of D_h matrix, it proceeds to the scheduling of low-priority packets stored in D_l matrix. The matrix D_l has many equal-length requests for whose handing IPOSS employs the ls-ee strategy, which is extensively demonstrated in case of high-priority requests. Thus, following the aforementioned logic, the service order of low-priority requests is: $d_l(4,2), d_l(1,1), d_l(1,2), d_l(4,1), d_l(2,2), d_l(3,2), d_l(2,1)$, and $d_l(3,1)$. Given the previous information, the final scheduling matrix produced by IPOSS is shown in Table II.

From the previous matrix, where the high-priority packets are depicted in bold face, it is obvious that IPOSS achieves significantly low delay for these packets and, thus, it successfully supports QoS requirements. Tables III and IV represent the scheduling matrix S in case that the IOSS and POSA algorithms are employed, respectively. In these tables, the bold face nodes are randomly selected among the others to represent their high-priority packets, since IOSS and POSA do not handle real-time traffic.

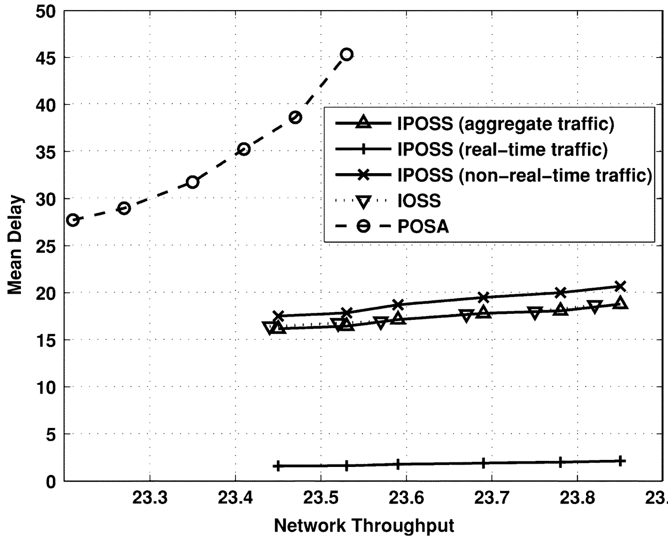
Based on the tables, the channel utilization providing by IPOSS and IOSS is 83.3% which is significantly improved in comparison to POSA whose channel utilization is 69.4%. In terms of the mean packet delay of aggregate traffic, the observed values are 6.1 timeslots for IPOSS, 6.2 timeslots for IOSS, and 7.8 timeslots for POSA. However, the contribution of the proposed scheme is clearly depicted by the mean delay of high-priority packets, since IPOSS cause mean delay of 2.3 timeslots, which is significantly improved in comparison with the previous values of delay caused by IOSS and POSA.

V. SIMULATION RESULTS

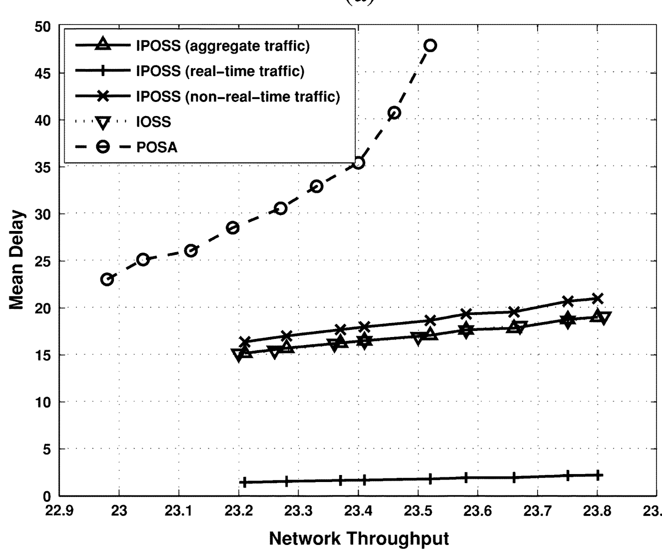
To evaluate the proposed algorithm, we carried out experiments, where we compared IPOSS with IOSS and POSA. More specifically, the experiments are conducted using a discrete-event simulator implemented in C environment. The performance of the compared algorithms is evaluated in terms of network throughput and mean packet delay. Network throughput represents the average number of bits transmitted per frame on each channel, while mean packet delay denotes the mean time in timeslots that packets wait at the queues till

TABLE IV
SCHEDULING MATRIX S PRODUCED BY POSA

	Timeslots																	
	l_1	l_2	l_3	l_4	l_5	l_6	l_7	l_8	l_9	l_{10}	l_{11}	l_{12}	l_{13}	l_{14}	l_{15}	l_{16}	l_{17}	l_{18}
λ_1	$\mathbf{u_1}$	u_1	u_1	u_2	$\mathbf{u_2}$	u_3	$\mathbf{u_4}$	u_4	$\mathbf{u_4}$	u_4								
λ_2				u_1	u_1	u_2	$\mathbf{u_2}$	$\mathbf{u_2}$	u_2	$\mathbf{u_3}$	u_3	$\mathbf{u_3}$	u_3	$\mathbf{u_3}$	u_4	$\mathbf{u_4}$	u_4	u_4

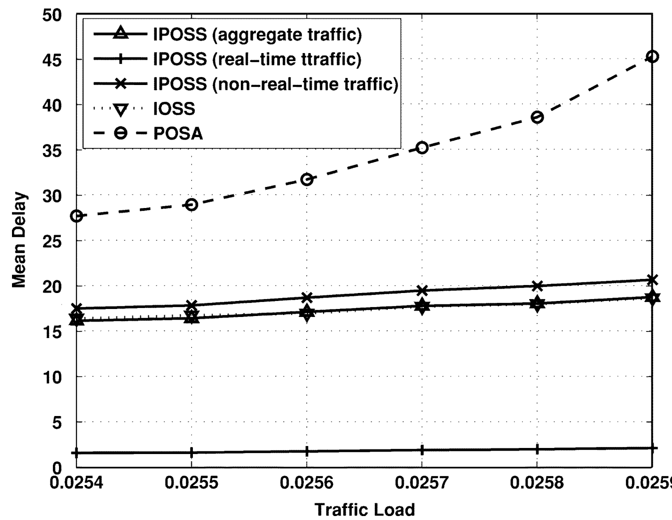


(a)

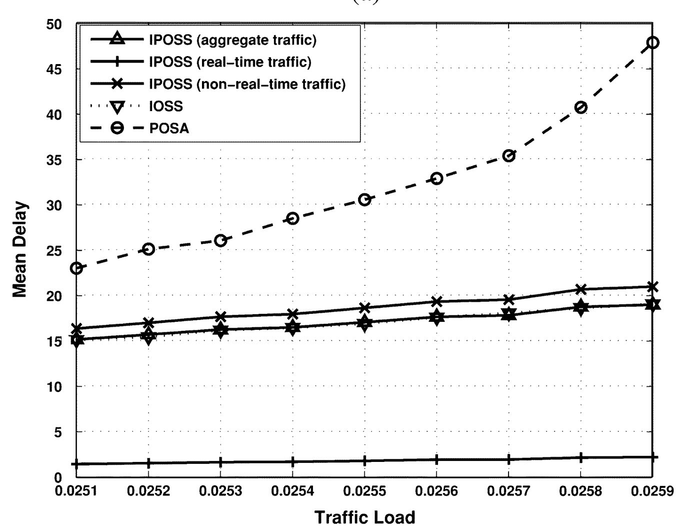


(b)

Fig. 4. Mean packet delay of aggregate, real-time, and non-real-time traffic as a function of network throughput for $w = 10$ channels. (a) Bernoulli traffic. (b) Poisson traffic.



(a)



(b)

Fig. 5. Mean packet delay of aggregate, real-time and non-real-time traffic as a function of network load for $w = 10$ channels. (a) Bernoulli traffic. (b) Poisson traffic.

the beginning of their transmission. It is crucial to keep the mean delay of high-priority packets low, in order to avoid long delays.

The experiments carried out are based on the following assumptions:

1) traffic pattern follows either the Bernoulli distribution with parameter p_1 or the Poisson distribution with parameter p_2;
2) line is defined at 3 Gb/s per channel and the tuning time is considered to be negligible;
3) outcome results from 10000 transmission frames;
4) shares of high- and low-priority packets is 25% and 75%, respectively.

Fig. 4(a) and (b) present the mean delay of aggregate, real-time, and non-real-time traffic as a function of network throughput, for $n = 30$ nodes and $w = 10$ channels, in case of Bernoulli and Poisson traffic, respectively. We can observe that IOSS and POSA have only one curve to present their mean delay, since they do not handle real-time traffic. On the other hand, IPOSS scheme supports priority-based QoS and as a result it achieves significantly lower levels of mean delay for

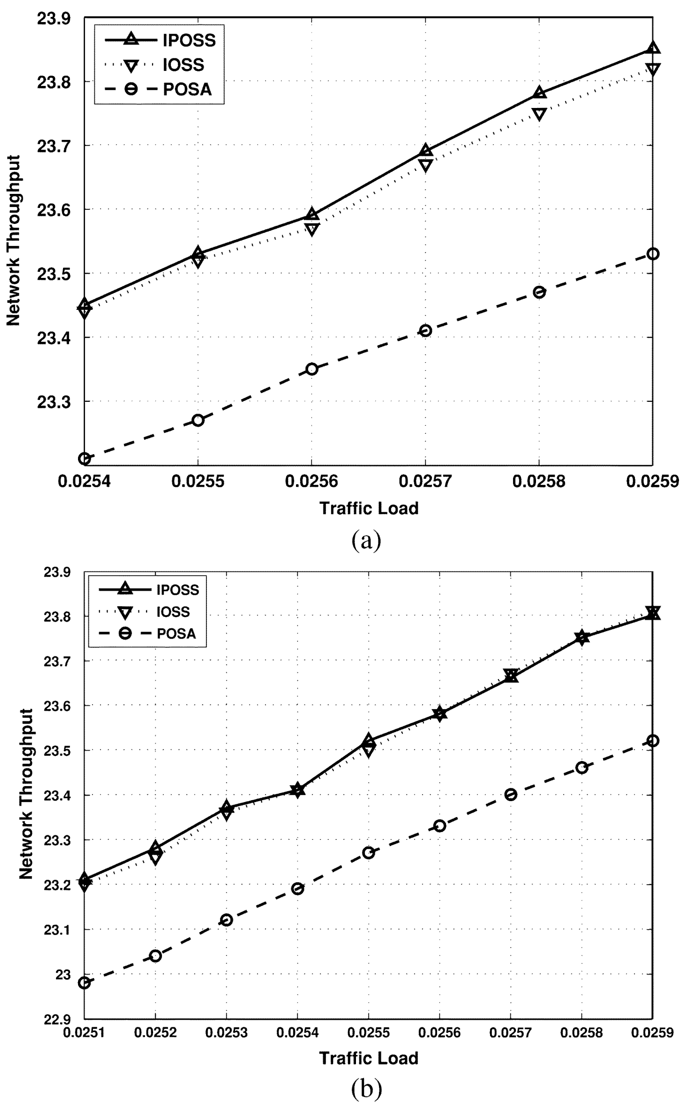


Fig. 6. Network throughput as a function of network load for $w = 10$ channels. (a) Bernoulli traffic. (b) Poisson traffic.

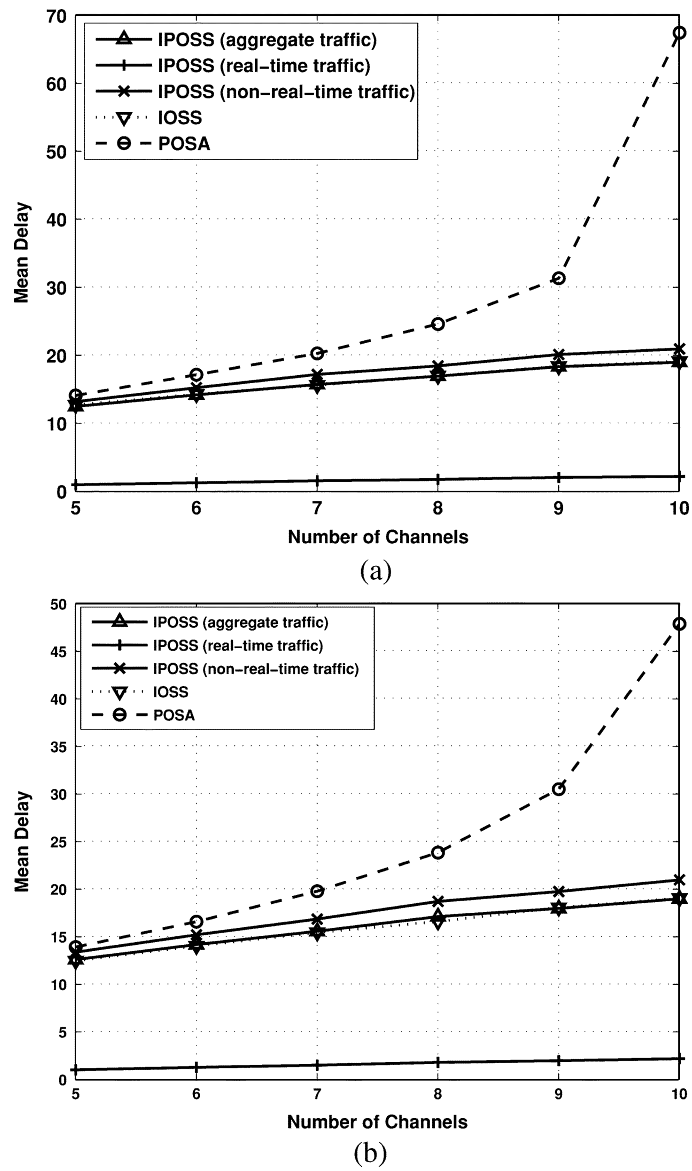


Fig. 7. Mean packet delay of aggregate, real-time and non-real-time traffic as a function of the number of channels. (a) Bernoulli traffic. (b) Poisson traffic.

high-priority compared to that of low-priority ones. Furthermore, IPOSS performance is clearly superior to that of POSA in terms of mean delay for both high- and low-priority packets, while it is marginally improved in comparison to IOSS in terms of aggregate traffic. This is expected, since IOSS and IPOSS are of the same logic and IOSS has already been superior to POSA. However, IPOSS's better performance is apparent under real-time traffic, where it achieves significantly lower delay compared to IOSS. This is due to the fact that high-priority packets have the privilege of being scheduled prior to low-priority ones. As shown in these figures, the mean delay of high-priority packets in case of IPOSS, for both Bernoulli and Poisson traffic, is up to 95.4% and 90.6% lower than the corresponding mean delay of POSA and IOSS, respectively. This significant improvement is not made in the cost of a high delay of low-priority packets, since, as depicted in Fig. 4(a) and (b), the non-real-time traffic curve of IPOSS is very close to that of the aggregate traffic.

Similar conclusions can be extracted from Fig. 5(a) and (b), where the mean delay is presented as a function of network load, for the aforementioned values of network's parameter i.e., $n = 30$ nodes and $w = 10$. In these figures, the values on the x axes correspond either to the values of the Bernoulli distribution's p_1 parameter [see Fig. 5(a)], or to the values of the Poisson distribution's p_2 parameter [see Fig. 5(b)]. Both p_1 and p_2 parameters denote the traffic load. For example, p_1 parameter expresses the possibility of generating one packet per node, per channel for each timeslot. Poisson p_2 parameter operates in line with $p(X; p_2) = (e^{-p_2} p_2^X)/(X!)$, which expresses the Poisson distribution. It is obvious that all protocols exhibit similar behavior independent of the traffic model, while their performance in terms of mean delay decreases as the network load increases. More specifically, the performance of IPOSS scheme is far superior compared to the performance of POSA independent of

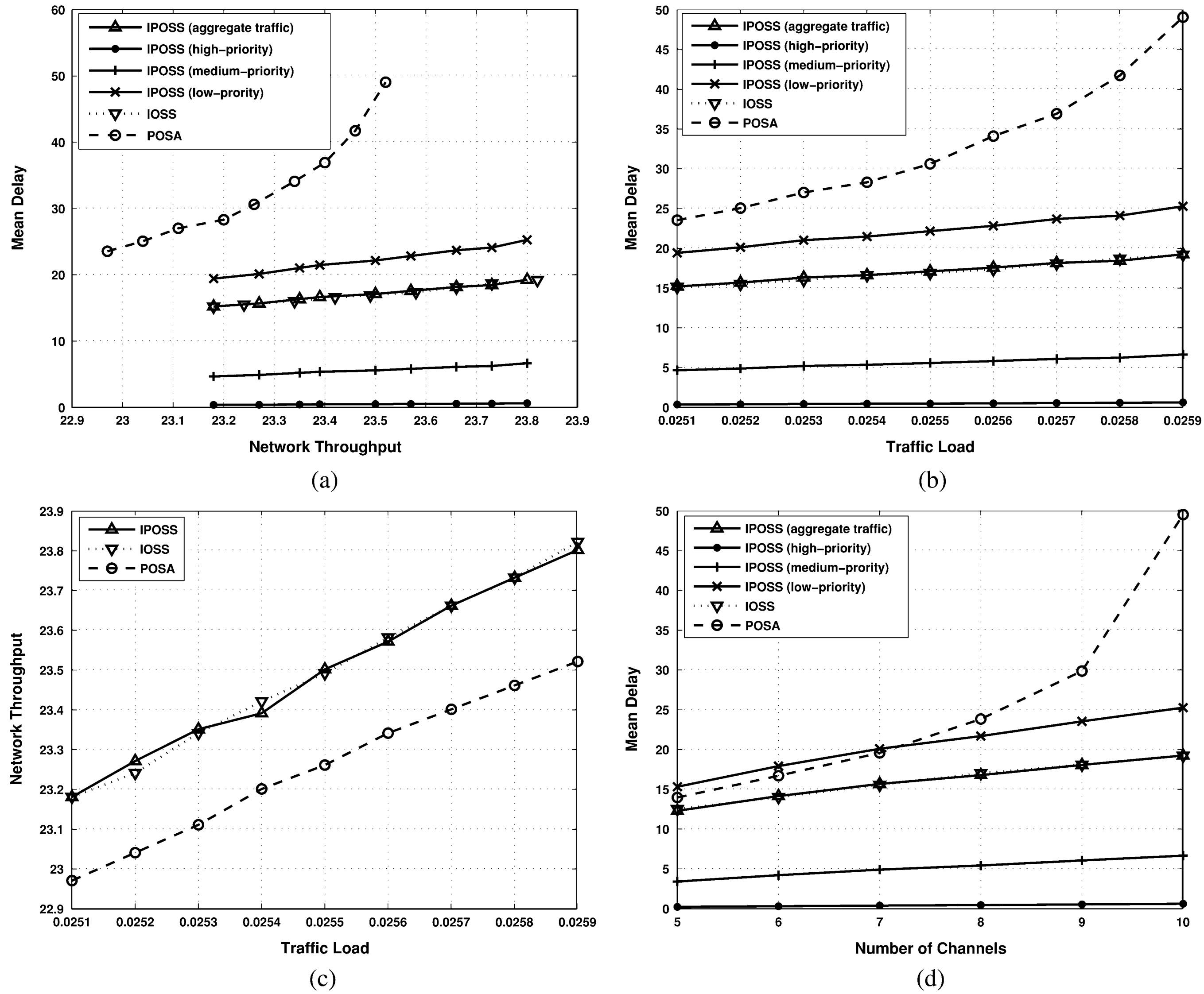


Fig. 8. More priority levels. (a) Mean packet delay as a function of network throughput for $w = 10$ channels. (b) Mean packet delay as a function of network load for $w = 10$ channels. (c) Network throughput as a function of network load for $w = 10$ channels. (d) Mean packet delay as a function of the number of channels.

packets' priority. On the other hand, what significantly differentiates IPOSS from IOSS is the mean delay of high-priority packets.

In Fig. 6(a) and (b), the network throughput is presented as a function of the network load for $n = 30$ nodes and $w = 10$ channels under Bernoulli and Poisson traffic, respectively. The throughput improvement that IPOSS provides over POSA scheme indicates that the use of the proposed algorithm leads to a significant reduction of the schedule's length. This is due to the fact that IPOSS apart from prioritizing the real-time traffic, it also provides to the long-length requests the privilege of being scheduled prior to the short-length ones and, thus, it allocates more free timeslots for the rest of requests. However, the improvement of IPOSS over IOSS is marginal, since the two schemes employ the same logic in handling the long- and short-length packets, which contribute to throughput performance.

Finally, Fig. 7(a) and (b) illustrate the mean delay of aggregate, real-time, and non-real-time traffic as a function of the number of channels for $n = 30$ nodes and traffic load equal to 0.026 and 0.0259, respectively. For this set of experimentation we set $w = 5, 6, \ldots, 10$. In line with the above results, it is apparent that the proposed scheme achieves significantly lower delay for high-priority packets. Thus, in Fig. 7(a), where the network's traffic follows the Bernoulli distribution, as the number of channels increases, IPOSS achieves from 93.3% to 96.8% lower delay in comparison to POSA and from 92.6% down to 88.9% lower delay in comparison to IOSS. On the other hand, in Fig. 7(b), where the network's traffic follows the Poisson distribution, as the number of channels increases, IPOSS achieves from 93% to 95.5% lower delay in comparison to POSA and from 92.1% down to 88.6% lower delay in comparison to IOSS.

A. *Experimentation With More Priorities*

On the above simulation results, data packets are considered to be of two priority levels: high- or low-priority level. However, the proposed scheme can be easily extended in order to handle more priority levels and thus to support applications which require a variety of QoS. In this section, we have conducted experiments where the IPOSS has been extended in order to handle three priority levels. More specifically, the shares of high-, medium-, and low-priority packets is 10%, 30%, and 60%, respectively.

In Fig. 8(a)–(d), the traffic pattern follows the Poisson distribution, while the number of nodes is $n = 30$. More specifically, in Fig. 8(a)–(c) the traffic load varies from 0.0251 to 0.0259 and the number of channels is fixed to $w = 10$, while in Fig. 8(d) the number of channels varies from 5 to 10 and the traffic load is equal to 0.0259. These figures show that independent of the network's parameters the addition of one priority level does not affect the performance of IPOSS, since the proposed scheme retains its superiority compared to POSA and IOSS.

In Fig. 8(a), where the mean packet delay is presented as a function of network throughput, it is apparent that IPOSS achieves lower levels of mean delay for high-priority packets compared to medium-priority ones, while the mean delay for medium-priority packets is significantly improved compared to that of low-priority ones. As it is expected, IPOSS performance is superior to that of POSA for all types of traffic, while it surpasses the IOSS performance for high- and medium-priority packets. The results of Fig. 8(b), where the mean packet delay is presented as a function of network load, are in line with that of Fig. 8(a). Fig. 8(c) illustrates network throughput as a function of network load and provides similar conclusions compared to Fig. 6(b), since network throughput is independent of the priority levels that IPOSS scheme handles. Finally, Fig. 8(d) confirms IPOSS performance under handling three priority levels presenting mean packet delay as a function of the number of channels. It is apparent that for any number on network channels high- and medium-priority packets has the privilege of being scheduled prior to low-priority ones and, thus, their mean delay is significantly lower compared to that of IOSS and POSA.

B. Major Observations

The following conclusions can be extracted from the simulation results presented in Fig. 4–8.

1) The proposed IPOSS scheme succeeds in handing real-time traffic and it is clearly superior to POSA and IOSS, since it creates a shorter schedule, which advances the network's throughput and significantly reduces the mean delay of high-priority packets. This is due to the fact that IPOSS exploits the priority information of data packets and, thus, it modifies their schedule by prioritizing high-priority packets over low-priority ones.
2) The proposed scheme can be easily extended to handle more priority levels and thus to support applications which require a variety of QoS. In our case, this extension handles three priority levels providing a superior performance for high- and medium-priority packets in terms of mean delay without sacrificing the performance of low-priority packets.

VI. Conclusion

A revised scheduling strategy for medium access in a optical star network is introduced. The proposed IPOSS supports concurrently two crucial issues: a collision-free reservation scheme and a QoS provision environment. In this manner, IPOSS keeps the following scheduling order: high-priority long-length requests, high-priority short length requests, low-priority long-length requests, and low priority short-length requests. Furthermore, IPOSS introduces a new scheduling strategy (ls-ee), which considers two decision vectors (NTV and CTV), indicating the earliest time availability of each channel for the node whose request is being processed. According to ls-ee strategy the final choice among the requests that demand the same amount of transmission frame is based on these vectors, allowing a better usage of channels' utilization, since the possibility of finding an appropriate interval to schedule short-length requests is increased. As a result, the proposed scheme obtains significant throughput-delay improvements for real time traffic compared to prior scheduling algorithms, under Bernoulli and Poisson traffic.

References

[1] P. Sarigiannidis, G. Papadimitriou, and A. Pomportsis, "A novel medium access control protocol for optical local networks based on data request ordering," *Opt. Laser Technol.*, vol. 40, no. 1, pp. 175–193, 2008.

[2] B. Mukherjee, *Optical WDM Networks*. New York: Springer, 2006.

[3] G. Papadimitriou, C. Papazoglou, and A. Pomportsis, "Optical switching: Switch fabrics, techniques, and architectures," *IEEE/OSA J. Lightw. Technol.*, vol. 21, no. 2, pp. 384–405, Feb. 2003.

[4] P. Green, "Progress in optical networking," *IEEE Commun. Mag.*, vol. 39, no. 1, pp. 54–61, Jan. 2001.

[5] J. McDonough, "Moving standards to 100 gbe and beyond," *IEEE Commun. Mag.*, vol. 45, no. 11, pp. 6–9, Nov. 2007.

[6] M. Herzog, M. Maier, and M. Reisslein, "Metropolitan area packet-switched WDM networks: A survey on ring systems," *IEEE Commun. Surveys*, vol. 6, no. 2, pp. 2–20, 2004.

[7] G. Papadimitriou, P. Tsimoulas, M. Obaidat, and A. Pomportsis, *Multi-Wavelength Optical LANs*. New York: Wiley, 2003.

[8] G. Bernstein, B. Rajagopalan, and D. Saha, *Optical Network Control: Architecture, Protocols, and Standards*. Boston, MA: Addison Wesley, 2003.

[9] K. Sivalingam, J. Wang, J. Wu, and M. Mishra, "An interval-based scheduling algorithm for optical WDM star networks," *J. Photon. Netw. Commun.*, vol. 4, no. 1, pp. 73–87, 2002.

[10] E. Johnson, M. Mishra, and K. Sivalingam, "Scheduling in optical WDM networks using hidden markov chain based traffic prediction," *J. Photon. Netw. Commun.*, vol. 3, no. 3, pp. 271–286, 2001.

[11] P. Sarigiannidis, G. Papadimitriou, and A. Pomportsis, "A high throughput scheduling technique, with idle timeslot elimination mechanism," *IEEE J. Lightw. Technol.*, vol. 24, no. 12, pp. 4811–827, Dec. 2006.

[12] P. Sarigiannidis, G. Papadimitriou, and A. Pomportsis, "Lena: An efficient channel eclectic algorithm for WDM optical networks," *Opt. Laser Technol.*, vol. 40, no. 1, pp. 39–51, 2008.

[13] P. Sarigiannidis, G. Papadimitriou, and A. Pomportsis, "A high performance scheduling priority scheme for WDM star networks," *IEEE Commun. Lett.*, vol. 11, no. 1, pp. 76–78, Jan. 2007.

[14] S. Petridou, P. Sarigiannidis, G. Papadimitriou, and A. Pomportsis, "Clustering based scheduling: A new approach to the design of scheduling algorithms for WDM star networks," presented at the 14th IEEE/CVT Ann. Symp. Commun. Veh. Technol. Benelux (IEEE SCVT), Delft, The Netherlands, 2007.

[15] I. Pountourakis, "Asynchronous transmission WDM networks with performance efficiency improvement," *IEEE Commun. Lett.*, vol. 7, no. 8, pp. 403–405, Aug. 2003.

[16] I. Pountourakis, "A new WDM protocol: Architecture and performance analysis," in *Proc. 10th IEEE Mediterranean Electrotechn. Conf. (IEEE Melecon)*, 2000, vol. 1, pp. 66–69.

[17] H.-C. Lin and P.-S. Liu, "Using dynamic control frames to reduce packet delays in transceiver array-based single-hop WDM networks," *IEEE J. Lightw. Technol.*, vol. 26, no. 7, pp. 742–755, Jul. 2008.

[18] B. Wang, J. Hou, and C. Han, "Design and analysis of a channel access protocol for WDM-based single-hop optical networks," *Photon. Netw. Commun.*, vol. 6, no. 3, pp. 289–308, 2003.

[19] R. Chipalkatti, Z. Zhang, and A. Acampora, "Protocols for optical star-coupled network using WDM: Performance and complexity study," *IEEE J. Sel. Areas Commun.*, vol. 11, no. 4, pp. 579–589, Apr. 1993.

[20] F. Jia, B. Mukherjee, and J. Iness, "Scheduling variable-length messages in a single-hop multichannel local lightwave network," *IEEE/ACM Trans. Netw.*, vol. 3, no. 4, pp. 477–488, Apr. 1995.

Panagiotis G. Sarigiannidis (S'05–M'07) received the Diploma and Ph.D. degrees in computer science from the Aristotle University of Thessaloniki, Thessaloniki, Greece, in 2001 and 2007, respectively.

He is currently an Adjunct Lecturer with the University of Western Macedonia, Macedonia, Greece. His research interests include optical networks and optical switching.

Sophia G. Petridou (M'08) received the Diploma and Ph.D. degrees in computer science from the Aristotle University of Thessaloniki, Thessaloniki, Greece, in 2000 and 2008, respectively.

Her research interests include clustering, optical, and wireless networks.

Georgios I. Papadimitriou (M'89–SM'02) received the Diploma and Ph.D. degrees in computer engineering from the University of Patras, Patras, Greece, in 1989 and 1994, respectively.

In 1997, he joined the faculty of the Department of Informatics, Aristotle University of Thessaloniki, Thessaloniki, Greece, where he is currently serving as an Associate Professor. His main research interests include optical networks and wireless networks. He is coauthor of three books published by Wiley. He is author or coauthor of 75 journal and 85 conference papers.

Prof. Papadimitriou is an Associate Editor of five IEEE publications.

Mohammad S. Obaidat (S'85–M'86–SM'91–F'05) received the M.S. and Ph.D. degrees in computer engineering with a minor in computer science from The Ohio State University, Columbus.

He is an internationally well known academic, researcher, and scientist. He is currently a full Professor with the Department of Computer Science, Monmouth University, West Long Branch, NJ. Among his previous positions are Chair of the Department of Computer Science and Director of the Graduate Program at Monmouth University and a faculty member at the City University of New York. He has received extensive research funding. He has authored or coauthored six books and over 380 refereed scholarly journal and conference articles. His research interests include wireless communications and networks, modeling and simulation, performance evaluation of computer systems, and telecommunications systems, security of computer and network systems, high performance computing–computers, applied neural networks and pattern recognition, security of e-based systems, and speech processing.

Dr. Obaidat was a recipient of a recognition certificate from IEEE and the distinguished Nokia Research Fellowship and the Distinguished Fulbright Award. He has served as a consultant for several corporations and organizations worldwide and is editor of many scholarly journals including being the Editor-in-Chief of the *International Journal of Communication Systems* published by Wiley. He is also an Editor of *IEEE Wireless Communications*. In 2002, he was the scientific advisor for the World Bank/UN Workshop on Fostering Digital Inclusion. He has made pioneering and lasting contributions to the multi-facet fields of computer science and engineering. He has guest edited numerous special issues of scholarly journals such as IEEE TRANSACTIONS ON SYSTEMS, MAN AND CYBERNETICS, *IEEE Wireless Communications*, IEEE SYSTEMS JOURNAL, *Elsevier Performance Evaluation*, *SIMULATION: Transactions of SCS*, *Elsevier Computer Communications Journal*, *Journal of C & EE*, and *Wiley, Security and Communication Network Journal*, and *Wiley International Journal of Communication Systems*, among others. He has served as the steering committee chair, advisory Committee Chair, honorary chair, and program chair of many international conferences. He is the founder of the International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS) and has served as the General Chair of SPECTS since its inception. Between 1994–1997, Obaidat has served as distinguished speaker/visitor of IEEE Computer Society. Since 1995 he has been serving as an ACM distinguished Lecturer. He is also and SCS Distinguished Lecturer. He is the founder of the SCS Distinguished Lecturer Program (DLP) and its present director. Between 1996 and 1999, he served as an IEEE/ACM program evaluator of the Computing Sciences Accreditation Board/Commission, CSAB/CSAC. Between 1995 and 2002, he has served as a member of the board of directors of the Society for Computer Simulation International. Between 2002 and 2004, he has served as Vice President of Conferences of the Society for Modeling and Simulation International SCS. Between 2004 and 2006, he has served as Vice President of Membership of SCS. He is currently the Senior Vice President of SCS. He has been invited to lecture and give keynote speeches worldwide. During the 2004–2005 academic, he was on sabbatical leave as the Fulbright distinguished Professor and Advisor to the President of Philadelphia University (Dr. Adnan Badran who became in April 2005 the Prime Minster of Jordan). He is a Fellow of the Society for Modeling and Simulation International SCS.

Andreas S. Pomportsis received the B.S. degree in physics, the M.S. degree in electronics and communications, and the Ph.D. degree in computer science from the University of Thessaloniki, Thessaloniki, Greece, the Diploma in electrical engineering from the Technical University of Thessaloniki, Thessaloniki, Greece.

Currently, he is a Professor with the Department of Informatics, Aristotle University, Thessaloniki, Greece. He is coauthor of the books *Optical Switching* (Wiley, 2007), *Multiwavelength Optical LANs* (Wiley, 2003), and *Optical Networks* (Wiley, 2003). His research interests include computer networks, computer architecture, parallel and distributed computer systems, and multimedia systems.