Scheduling algorithm for differentiated service on wavelength-division-multiplexed passive optical access networks

Maode Ma

School of Electrical and Electronic Engineering, Nanyang Technological University
Nanyang Avenue, Singapore, 639798

emdma@ntu.edu.sg

Mounir Hamdi

Department of Computer Science, Hong Kong University of Science and Technology
Clear Water Bay, Kowloon, Hong Kong

hamdi@cs.ust.hk

Received 6 August 2002; revised manuscript received 23 September 2002

Passive optical networks (PONs) will be the pervasive choice in the design of next-generation access networks. One possible solution to implementing passive optical access networks is to rely on wavelength-division multiplexing (WDM). Here we investigate the problem of providing real-time service to both hard and soft real-time messages in conjunction with a conventional best-effort service in WDM optical networks based on the single-hop passive-star coupler. We propose an adaptive scheduling algorithm to schedule and manage the message transmissions in the specified network. We have conducted extensive discrete-event simulations to evaluate the performance of the proposed algorithm.

OCIS code: 060.4250.

1. Introduction

Over the past 10 years, backbone networks have experienced substantial development. Optical fibers have been widely used as the main medium in the backbone network to increase bandwidth. However, the networks at the last mile, from the Internet providers to the end users, still experience insufficient resources. Recent studies[1,2] have shown that passive optical networks (PONs) are the feasible solution to the last-mile problem. With inexpensive passive optical components, the optical fiber could be brought to buildings and homes; thus we could expect great increases in the bandwidths of the access networks for meeting the demands of delivering multimedia services to end users.

One possible implementation of the PON is the single-hop passive-star-coupled topology[3] supported by wavelength-division multiplexing (WDM). WDM is an effective technique for utilizing the large bandwidth of an optical fiber. This technique, by allowing multiple messages to be transmitted in parallel on a number of channels, has the potential to improve significantly the performance of optical networks. Several topologies have been proposed for WDM optical networks.[4] Different from others, the single-hop passive-star-coupled topology, initially designed for local or metropolitan networks, can configure a WDM optical network as a broadcast-and-select network in which all the inputs from various nodes are combined by a WDM passive-star coupler, and the mixed optical information is broadcasted to all destinations. With this capacity, the single-hop passive-star-coupled

topology could be one of the solutions for the PON, which, logically, has point-to-point communications by nature.

Efficient medium-access-control (MAC) protocols and scheduling algorithms are needed to allocate the network resources optimally while satisfying the messages and system constraints. The medium access control protocols[5] in a single-hop passive-star-coupled WDM optical network environment can be divided into two main classes, namely, preallocation-based and on-demand adaptive protocols. Preallocation-based techniques [6,7] assign transmission rights to different nodes in a static and predetermined manner. Reservation-based techniques[8–12] allocate a channel as the control channel to transmit global information regarding messages to all nodes in the system. Once such information is received, all nodes invoke the same scheduling algorithm to determine when to transmit or receive a message and on which data channel. In this paper we focus on reservation-based techniques.

Many research results have been published to schedule variable-length messages.[8,10] These *variable-length* scheduling algorithms are more general than fixed-length scheduling algorithms and adapt better to various traffic characteristics (e.g., bursty). In addition, they perfectly fit the current research trend of IP-over-WDM and WDM burst switching.[13] We adopt the same strategy in this paper by allowing our scheduling algorithms to handle variable-length messages.

Our major contribution in this paper is that we develop a novel scheduling algorithm to provide differentiated services to messages with different time constraints for reservation-based MAC protocols in a single-hop WDM network, which has the potential to be adopted as a passive optical access network. The objective of the scheduling is to balance the network service for different kinds of messages while the messages' time constraints can be satisfied as much as possible. The advantage of the proposed algorithm is that the network resource could be efficiently used to balance different network services. This advantage is achieved by the principle that time-constrained messages have high priority for transmission while messages without time constraints could be transmitted earlier when real-time messages have been blocked as a result of transceiver tuning and unavailability of the destination nodes.

The remainder of this paper is organized as follows. Section 2 specifies the system mode of the WDM optical network. Section 3 presents our scheduling algorithm in detail. Section 4 shows the experimental results for evaluating the performance of the algorithm. Finally, Section 5 concludes the paper with a summary.

2. System Model and Service

We consider message transmission in a single-hop WDM network, whose nodes are connected via a passive-star coupler. The star coupler supports C channels and N nodes in the network. C channels, referred to as data channels, are used for message transmission. Another channel, referred to as the control channel, is used to exchange global information among nodes regarding the messages to be transmitted. The control channel is the basic mechanism for implementing the reservation scheme. Each node in the network has two pairs of transceivers. One pair of transceivers is fixed and tuned to the control channel; the other pair of transceivers is tunable to any of the data channels to access messages on those channels.[8]

The nodes are assumed to generate aperiodic messages with variable length, which can be divided into several equal-sized packets. The basic time interval on the data channels is the transmission time of one packet. In our model, we assume that the basic transmission unit is one message. The nodes are divided into two nondisjoint sets of source nodes s_i and destination nodes d_j. However, any node can be a source node as well as a destination node at the same time, because there is a transmitter and a receiver at each node. A queue for the messages waiting to be transmitted is assumed to exist at each source node s_i.

A time-division multiple-access (TDMA) protocol is used on the control channel for each node to access that channel. According to the TDMA protocol on the control channel, each node can transmit a control packet during a predetermined time slot. The basic time interval on the control channel is the transmission time of a control packet. N control packets make up one control frame on the control channel. Thus each node has a corresponding control packet in a control frame; when this is the case, that node can access the control channel. The length of a control packet depends on the number of messages through which each node is allowed to broadcast control information and on the amount of control information regarding each message, such as the address of the destination node, message length, time constraint, and so on. Figure 1 illustrates some of the basic concepts used in our model.

The transmitted messages in our WDM network can be described by the following parameters, which are also the control information contained in the control packet for each message to be transmitted. The source and destination nodes of a message are node addresses where the message is generated and where it should be received. The indication of hard or soft real time of a message is a sign to show whether the message should be dropped or still be transmitted when its time constraint is violated. The message without time constraint will be considered to be a non-real-time message with an infinite relative deadline. The relative deadline of a message is the time constraint of that message. A message will be considered to meet its time constraint when the transmission time of the message is less than its relative deadline.

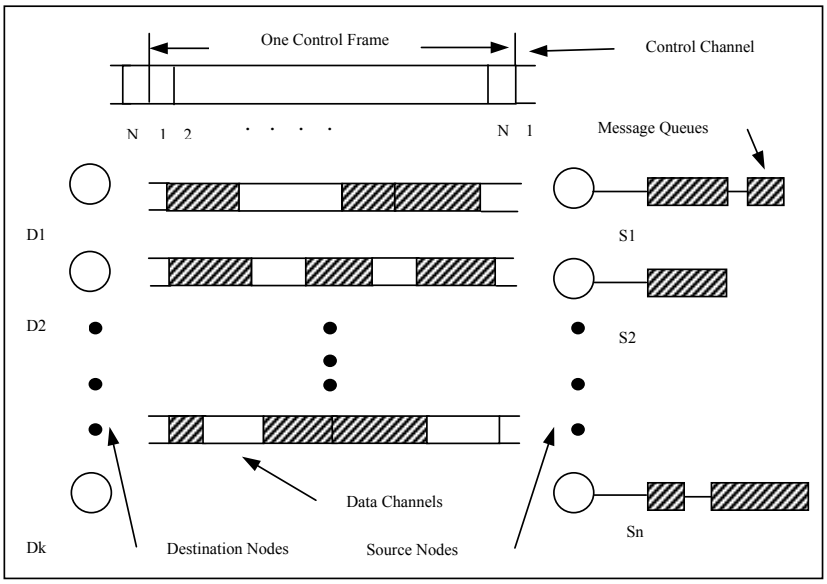

Fig. 1. Architecture of the WDM optical network.

The network service procedure can be summarized as follows. As the control channel is divided into time slots according to a TDMA scheme, each source node s_i can send a control packet during time slot i on the control channel to all other nodes. The control packet contains the information about one (at the head of s_i's message queue) or more messages it intends to transmit. After $R+F$ time units, where R is the round-trip propagation delay between a node and the star coupler and F is the time duration of a control frame, all the nodes in the network will have the information contained in a control frame regarding messages to be transmitted. At this point, an identical copy of a distributed scheduling

algorithm is invoked at each node, which assigns the messages represented in the control frame to the appropriate data channels and time slots so that the scheduled messages can be transmitted accordingly.

3. Scheduling Algorithm

The algorithm involved in our protocol should, first of all, choose one message among the messages at the head of each transmitter queue and then determine the time slots of a transmission channel, which can be selected from those available channels, to the selected message. Obviously, the differentiated service provided by the network, which is to transmit the messages within their time constraint without much sacrifice of throughput of non-real-time messages, can be implemented by an effective scheduling associated with the MAC protocol.

The technique of assigning data channels and transmission time slots to the selected messages may vary according to different WDM network models.[8] One of these techniques is called *earliest available time scheduling* (EATS). EATS is an efficient channel-assignment algorithm for selecting a channel and time slots to the transmitted messages. In our WDM network model, we adopt EATS as our basic channel-assignment mechanism. However, the choice of channel-assignment technique in our approach is independent of the other part of our scheduling algorithm. The basic idea of the EATS algorithm is to assign a message to a data channel that has the earliest available time slot among all other channels. Once the data channel is assigned, the message is scheduled to transmit as soon as that channel becomes available. To track channel and receiver usage and the network situations, two tables reside on each node, which are known as *receiver available time* array (RAT) and *channel available time* array (CAT). RAT records the unavailable time of the receiver of each node from the current time in the packet slot unit. CAT records the unavailable time of each channel from the current time. Both of them decrease dynamically with time units. With this global information, the distributed EATS works as follows: Transmit a control packet on the control channel; sort the channels on the basis of the CAT information; choose a channel with earliest available time slot, which is the channel with the smallest CAT value; calculate the transmission time of a message on the basis of the two tables; update the two tables according to the newly scheduled message.

The message transmission sequence is basically a policy to determine the order of message transmissions. There are various policies to assign priorities to the transmitted messages.[10–12] However, these priority schemes cannot be used effectively in a mixed-traffic differentiated service environment such as the one envisioned in this paper. Our idea is to design the scheduling algorithm such that the algorithm should first schedule the messages according to their time constraints. Then the messages without time constraints could be scheduled while the real-time messages are waiting for transmission.

To schedule the transmission of messages according to time constraints, we can adopt the *minimum laxity first* (MLF) scheme. By scheduling messages with minimum laxity first, the MLF algorithm could be expected to schedule and transmit tightly time-constrained messages first in order to reduce the messages' loss rate. To schedule the transmission of messages without time constraints, we have to seek suitable time periods during the scheduled real-time message transmission. We noted that a period of time—when the real-time messages are being blocked while waiting for their destinations to be free—could be used to schedule message transmission without or with time constraints. Since the RAT and the CAT, which contain global information on the states of the receivers and the channels, are available to every node, this idea is feasible and can easily be implemented.

With the transmission channel and time-slots assignment technique, we combine our real-time scheduling scheme with the insertion scheduling technique, which inserts non-real-time message transmissions in the tolerant time period, to form a new scheduling al-

gorithm, named the *MLF with time tolerance scheduling algorithm* (MLF-TTS). With this algorithm, the differentiated service could be provided by the MAC protocols in single-hop WDM networks. By use of the MLF-TTS, our initial objective of differentiated service could be achieved, which is to schedule the transmission of real-time messages to meet their time constraints as much as possible while the transmission of messages without time constraints could also benefit. Compared with the simple MLF scheduling algorithm[12] we can expect the average message delay for the messages without time constraints to decrease while the message loss rate or message tardy rate is kept as low as those of the MLF algorithm. In addition, unlike the scheduling algorithms,[10,11] which aim only to decrease the average message delay, the MLF-TTS can be expected to increase significantly the real-time performance of the WDM MAC protocol.

After all packets of the control frame reach all nodes, the scheduling algorithm, MLF-TTS, is called at each node to schedule the transmission of all messages represented in the current control frame. The algorithm will sort the real-time messages according to their time laxities and the non-real-time messages according to their message lengths. Then priorities are assigned to all sorted messages. The message with the highest priority will be the first considered to get its transmission channel and the time slots on that channel by our channel-assignment scheme. The channel-assignment algorithm, EATS, will assign the earliest available channel to this message. Then the delay to transmit this message will be evaluated by the algorithm. If the delay—including the time when the message is waiting for its assigned channel, tuning latency, and its destination to be available plus the transmission time which is the length of the message—is larger than its laxity, a hard real-time message will be dropped and a soft real-time message will be degraded to a non-real-time message with the lowest priority. The message with the next priority will then be chosen for scheduling. If the delay is less than its laxity, the scheduling is fixed to transmit the message thereafter. The algorithm will then consider arranging another message transmission by using the period of the waiting delay, t_{ws}, which is caused by correct tuning and the destination node of the scheduled message being available. The algorithm will further look for the current earliest available destination node and the candidate messages represented in the same control frame, either real time or non real time, with destination to this node. The algorithm will select one message from those candidates to schedule its transmission if the waiting time, t_{we}, for this message to get the earliest available destination plus its transmission time, is less than the waiting time t_{ws}. If this message can be found, it is scheduled for transmission on the earliest available channel to the earliest available node. After this is done, or no such message is found under this condition, the algorithm will continue to schedule other messages according to their priorities. After all the messages represented in the current control frame have been scheduled, the source nodes will then know on which channel to transmit which message at the heads of their message queues and at what time. The receiver nodes will also know to which channel they should tune and at what time to receive the appropriate message.

The MLF-TTS algorithm can be expressed in detail as follows: We assume that there are M nodes and C channels. The messages have variable lengths, l, following an exponential distribution. The real-time messages have time constraints with laxity, p, following an exponential distribution, too. The messages can be transmitted from source node i to destination node j, where $i \neq j$ and $i, j \in$ g. The RAT table can be expressed as an array of M elements, one for each node. $\mathrm{RAT}[i] = w$, where $i = 1, 2, \ldots, M$, means that node i will be free after w time slots. The CAT table can be expressed as an array of C elements, one for each channel. $\mathrm{CAT}[j] = v$, where $j = 1, 2, \ldots, C$, means that channel j will be available after v time slots.

MLF-TTS Algorithm

Begin:

Wait for a control packet on the control channel returning;

Sort the real-time messages represented in the control frame on the basis of their laxities;

Sort the non-real-time messages according to their lengths;

Assign the transmission priority to different messages with real-time messages;

Always have higher priorities;

S1:

Assign transmission channel to the current highest priority until no message left;

Search CAT[i] for a channel with the earliest available time;

Use the earliest available channel k, to transmit the selected message;

Calculate $r = \mathrm{CAT}[j] + T, t1 =$ max(CAT[k], T), $t2$= max($t1 + R$, r); where T is the transmitters' tuning time, R is the propagation delay;

Schedule the message transmission time at $t = t2 - R$;

If waiting time, $t_{\mathrm{ws}}, +$ transmission time (message length), $l_s + R >$ message laxity, p_s , drop it if it is a hard real-time message, degrade it to non-real-time message if it is a soft real-time message; return to S1 to schedule another message;

If waiting time, $t_{\mathrm{ws}}, +$ transmission time (message length), $l_s + R <$ message laxity, p_s update RAT[j] $= t2 + l_s$, CAT[k] $= t2 - R + l_s$; where $t_{\mathrm{ws}} = t -$current time;

Search for the current least visited node by min(RAT[j]);

Search for the candidates with destination to min(RAT[j]);

S2:

Select one message to schedule by testing availability of time tolerance until all considered;

If waiting time, $t_{\mathrm{we}}, +$transmission time (message length), $l_e + R >$ waiting time, t_{ws}, return to $S2$;

If waiting time, $t_{\mathrm{we}}, +$transmission time (message length), $l_e + R <$ waiting time, t_{ws}, assign the same channel to the message for the destination node min(RAT[j]), update RAT$[\min(\mathrm{RAT}[j])] = t_{\mathrm{we}} + l_e + R$; return to $S1$;

End.

The complexity of the MLF-TTS scheduling algorithm can be evaluated according to its operation. We find that the new algorithm has only one sequence procedure and three search procedures.

The sequence procedure is to sort the messages represented in one control frame according to their time constraints or message lengths. The first searching procedure is to search for a channel with the earliest available time among all the channels in the network, and the second procedure is to search for a node with the earliest available time among all the nodes in the network. The final procedure is to search for a set of messages with destination to the earliest available node.

Let us assume that the number of nodes is always more than the number of channels. The number of times to invoke the sorting procedure is 1 only when all the messages represented in one control frame are scheduled. The complexity of a typical sorting algorithm is $O(n \log_2 n)$, where n is the number of nodes in the network. The complexity of a searching procedure is $O(n)$ for each message scheduling. The worst-case running time of the algorithm is $O(n \log_2 n) + 3nO(n)$. Finally, the complexity of the algorithm could be $O(n^2)$ for scheduling one batch of messages. The complexities for the MLF and shortest message first (SMF) algorithms can be evaluated on the basis of their operations. With both algorithms there is a sorting procedure for scheduling one batch of messages and one searching

procedure for each message scheduling. The worst-case running time of each algorithm is $O(n\log_2 n) + nO(n)$. And finally, their complexities also include $O(n^2)$, because the MLF-TTS algorithm has just two more searching procedures for each message scheduling.

4. Experimental Evaluation

In this section we present the results of a set of performance-comparison experiments to evaluate the performance of our proposed scheduling algorithm. In the experiments we study the performance of the network with the passive-star-coupler-based architecture when we have integrated traffic (including messages with or without time constraint) with varying message arrival rates.

4.A. Experiment Design

The parameters involved in our simulation include the number of nodes, which is set to 50, and the number of channels, which is set to 4. Tuning latencies are set to 0 time units in the experiments to focus the results on the salient features of the proposed scheduling algorithm. Round-trip propagation delay is set to 10 time units. However, we do not include this delay in the figures, because it is the same for all algorithms.

The channel-assignment strategy chosen for all candidate algorithms is the EATS algorithm. The candidate algorithms for the performance-comparison experiments are the SMF, the MLF, and the MLF-TTS scheduling algorithms. In comparing the SMF scheduling algorithm, we want to show that the MLF-TTS scheduling algorithm can improve the real-time performance of the network. The SMF algorithm sequences messages according to message length in order to achieve good performance in terms of average message delay. But it does not consider scheduling the transmission of messages with time constraint. In comparing the MLF scheduling algorithm, we want to show that the MLF-TTS scheduling algorithm can improve the network performance of messages without time constraint. The MLF algorithm schedules message transmission on the basis of the time constraint of messages in order to achieve good real-time performance in terms of message loss rate. However, we achieve real-time performance by sacrificing the transmission of non-real-time messages.

4.B. Experiment Results

The experimental results presented Figs. 2–4 show the relationship between the system performance and the system traffic load in terms of message arrival rate when real-time messages and non-real-time messages are transmitted.

Message length is a random variable following an exponential distribution. A Poisson message arrival rate across all nodes is considered that ranges from 0.002 to 0.005 messages per unit time as its mean for each node. Destination nodes for messages are chosen according to a uniform probability distribution.

The message time constraint is expressed by message laxity, which is a random variable following an exponential distribution. The behavior of the candidate algorithms is observed over a simulation period of 1,000,000 time units. Each point in the performance graphs is the average of 11 independent runs. The metrics of real-time performance in the experiments are the *message loss rate* for hard real-time message transmission. The *average message delay* and the *system throughput* are general performance metrics to describe the network performance. The average message delay is defined as the average time a message spends in the system, which is composed of message transmission delay, queuing delay, and propagation delay. The system throughput is defined as number of packets that are transmitted per unit of time. The specific message parameters used are as follows: The mean message arrival rate varies from 0.002 to 0.005; the mean message length is set to 20;

the mean message laxity is set to 25; and each type of message occupies 50% of the total population. For real-time messages, one half of them are hard real-time messages, and the other half are soft real-time messages; each occupies 25% of the total population.

For all the candidate algorithms, the principle of scheduling the transmission of the real-time messages is that any hard real-time message, which is later than its laxity, will be dropped; any soft real-time message, which is later than its laxity, will still be transmitted.

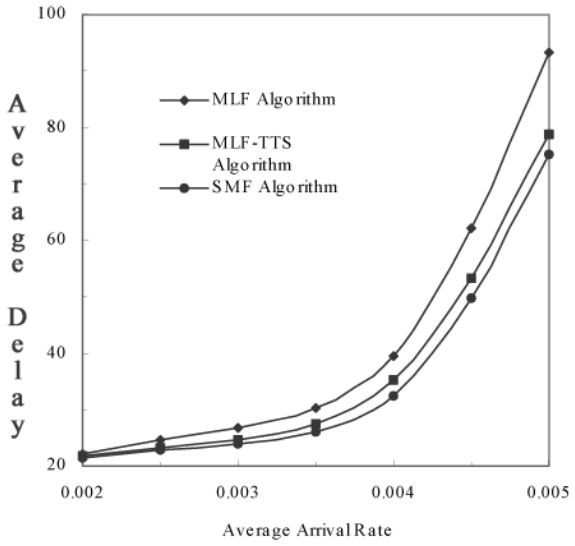

Fig. 2. Message delay versus message arrival rate.

Figure 2 presents the average message delay of all algorithms when the messages arrival rate changes. The figure shows that the SMF algorithm performs better than the other algorithms. The reason lies in that, unlike the MLF algorithm, the SMF algorithm always tries to arrange short messages to transmit first. This will decrease the waiting time of messages in the network. However, our new scheduling algorithm has achieved performance similar to that of the SMF algorithm. It has substantially improved the performance of the MLF algorithm in terms of average message delay. The improvement has reached more than 15%. What makes this achievement possible is that the MLF-TTS algorithm takes the time period when the scheduled message is waiting for its destination available to insert other message's transmission.

Figure 3 shows the relationship between the average message delay and the system throughput. From Fig. 3 we can see that the performance of the EATS algorithm and the MLF-TTS algorithm are better than that of the MLF algorithm in the sense that at a certain value of system throughput the average message delay of the system using the EATS algorithm or the MLF-TTS algorithm is always less than that of the system using the MLF algorithm. This is shown clearly at the point where the throughput approaches and exceeds 3.

Figure 4 presents the real-time performance of the system using different algorithms. The performance of the MLF algorithms is much better than that of the SMF algorithms and the MLF-TTS algorithm. The reason for this result is that the MLF algorithms always try to transmit the messages with short laxity first so that the message loss rate will be kept at the lower level. The SMF algorithm keeps the message loss rate at the highest level. The reason for this is that the SMF algorithm always schedules the shortest message first, ignoring the

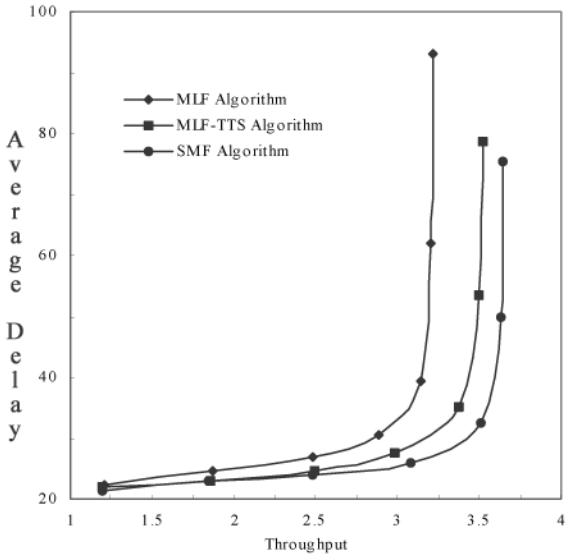

Fig. 3. Average message delay versus throughput.

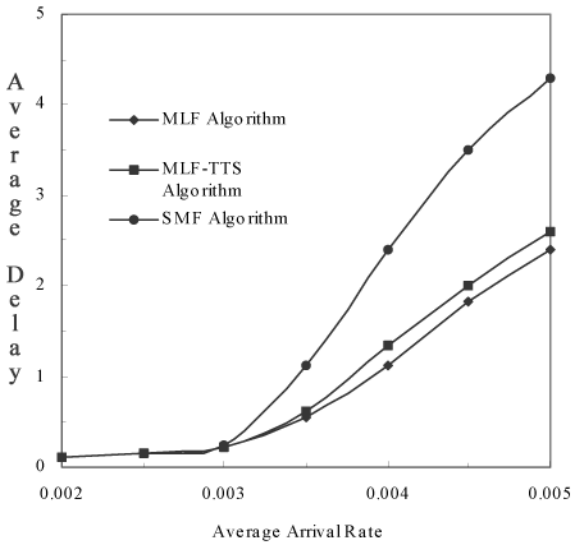

Fig. 4. Message loss rate versus message arrival rate

laxities of messages so that longer messages with tighter time constraints could not be scheduled in time. As a result, the real-time performance of the network will be impaired. However, although the message loss rate of the MLF-TTS algorithm is not exactly as low as that of the MLF algorithm, it has greatly improved the real-time performance of the network. The improvement in terms of the message loss rate by the MLF-TTS algorithm can reach up to 30% that of the SMF algorithm when the traffic is heavy. This is because the MLF-TTS algorithm follows the principle of the MLF algorithm. Moreover, the insertion of scheduling other messages will not violate the MLF principle. So it achieves almost the same real-time performance as that of MLF algorithm.

The following points summarize our overall results: (1) Our newly proposed algorithm, MLF-TTS, always performs better than the MLF algorithm in terms of average message delay and throughput. (2) Our newly proposed algorithm, MLF-TTS, performs better than the SMF algorithm in terms of message loss rate. (3) The MLF-TTS algorithm can balance both types of message transmission without too much mutual impairment.

5. Conclusion

In this paper we have proposed that the single-hop passive-star-coupler-based topology supported by WDM could be one of the solutions to realizing passive optical access networks. We have also proposed what to our knowledge is a novel reservation-based scheduling algorithm, named MLF-TTS, for providing differentiated transmission service to messages with and without time constraints in single-hop passive-star-coupler-based WDM optical networks. Using this scheduling algorithm, we can provide transmission service to either hard or soft real-time messages to meet their time constraints as much as possible, while the transmission of non-real-time messages can also be improved. The results of our experiments showed that, with our scheduling algorithm, almost the same real-time performance can be achieved as that of the simple real-time scheduling algorithm; however, the transmission performance of non-real-time messages can be improved simultaneously.

References and Links

1. G. Kramer and G. Pesavento, "Ethernet passive optical network (EPON): building a next-generation optical access network," IEEE Commun. Mag. (February 2002), pp. 66–73.
2. G. Kramer, B. Mukherjee, and G. Pesavento, "IPACK: a dynamic protocol for an Ethernet PON (EPON)," IEEE Commun. Mag. (February 2002), pp. 74–80.
3. K. Bogineni, K. M. Sivalingam, and P. W. Dowd, "Low-complexity multiple access protocols for wavelength-division multiplexed photonic networks," IEEE J. Sel. Areas Commun. **11,** 590–603 (1993).
4. B. Mukherjee, "WDM-based local lightwave networks. I. Single-hop systems," IEEE Netw. (May 1992), pp. 12–27.
5. B. Li, M. Ma, and M. Hamdi, "MAC protocols for WDM networks: survey and summary," in *Optical WDM Networks: Principles and Practice* (Kluwer, Boston, 2000).
6. A. Ganz and Y. Gao, "Time-wavelength assignment algorithms for high performance WDM star based systems," IEEE Trans. Commun. **42,** 1827–1836 (1994).
7. G. N. Rouskas and M. H. Ammar, "Analysis and optimization of transmission schedules for single-hop WDM networks," IEEE/ACM Trans. Netw. **3,** 211–221 (1995).
8. F. Jia, B. Mukherjee, and J. Iness, "Scheduling variable-length messages in a single-hop multichannel local lightwave network," IEEE/ACM Trans. Netw. **3,** 477–487 (1995).
9. N. Mehravari, "Performance and protocol improvements for very high-speed optical fiber local area networks using a passive star topology," J. Lightwave Technol. **8,** 520–530 (1990).
10. B. Hamidzadeh, M. Ma, and M. Hamdi, "Message sequencing techniques for on-line scheduling in WDM networks," J. Lightwave Technol. **17,** 1309–1319 (1999).

11. M. Ma, B. Hamidzadeh, and M. Hamdi, “An efficient message scheduling algorithm for WDM lightwave networks,” Comput. Netw. **31,** 2139–2152 (1999).
12. M. Ma, B. Hamidzadeh, and M. Hamdi, “Efficient scheduling algorithms for real-time service on WDM optical networks,” Photon. Netw. Commun. **1,** 161–178 (1998).
13. J. Strand, A. Chiu, and R. Tkach, “Issues for routing in optical networks,” IEEE Commun. Mag. (February 2001), pp. 81–88.