WFF: A High Performance Scheduling Algorithm for WDM Star Networks that Minimizes Idle Timeslots

P.G.Sarigiannidis, G.I.Papadimitriou, *Senior Member, IEEE,* and A.S.Pomportsis

Department of Informatics, Aristotle University, Box 888, 54124 Thessaloniki, Greece

A new media-access control (MAC) protocol is introduced in this paper. We consider a Wavelength Division Multiplexing (WDM) network with star topology. The protocol adopted is pre-transmission coordination-based, so the protocol coordinates nodes before the actual transmission. The protocol introduces a new method of computing the reservations of the demand matrix and brings some performance improvement, which is proven by simulations in terms of channel utilization, throughput and delay.

Introduction

If we suppose that the main advantage of optical switching is that it enables routing of optical data signals without the need for conversion to electrical signals then we can easily understand that it is the key factor of the future networks [1]. The WDM technique [2,3] brings this to reality by dividing the available optical bandwidth into multiple channels of lower bandwidth, which can be easily supported by the nodes' electronic circuits. The absence of the optical to electronic form conversion [4] offers Gigabit data rates; hence the multiplexing and the demultiplexing of the available channels (or wavelengths) are performed optically.

WDM Broadcast-and-Select star Networks belong to the WDM networks family and achieve a considerably high performance [5], using a passive star coupler in order to route all transmitted data from its inputs to its outputs. The presented protocol assumes a WDM broadcast-and-select LAN (Local Area Network) with N nodes and W wavelengths.

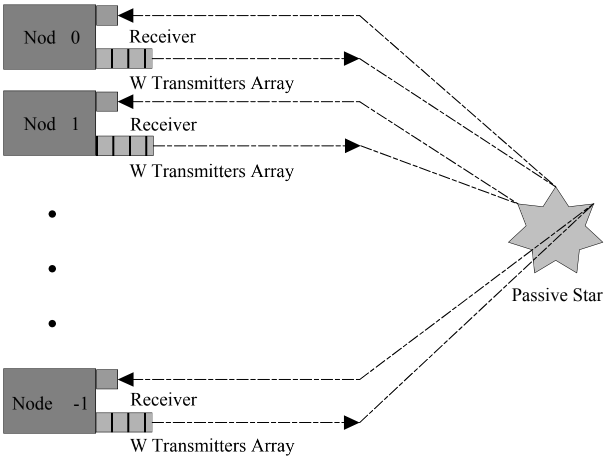

Fig.1. Broadcast and Select star network with tunable transmitter and fixed receiver per node. The network has N nodes and W channels.

Each node is equipped with an array of W fixed transmitters (or a tunable transmitter); hence each node can transmits using any channel. Also, a fixed receiver allows each node to receive in a particular channel, knows as its home channel (Fig. 1). Data transmission occurs sequentially in data frames, which are divided further in timeslots. In each frame the protocol examines the transmission requests of the network nodes and performs schedule processing in order to specify the order in which nodes will transmit in each channel. There is no restriction in the number of nodes that can transmit in each frame. Every node can send its transmission requests and have them assigned to a timeslot. In addition, every node has access to every transmission channel. However, there are two things that cannot be realized. First, a node cannot transmit simultaneously in two channels and second, two or more nodes cannot transmit simultaneously using the same channel.

Online Algorithms

As a typical online scheduling algorithm, OIS (online interval scheduling) [6] shares the advantage of online algorithms in that it does not require that the entire demand matrix becomes available before it begins schedule computation. Instead, OIS begins constructing the transmission schedule as soon as the requests of the first node are available. OIS functions as follows: assume that node n requests t_l timeslots for transmission in channel w and that OIS determines that there is a time period in the next frame during which the requested channel is available i.e. there exists a value t such that the channel is available from timeslot t to timeslot $t + (t_l - 1)$. Next, OIS examines the possibility of contentions. If the algorithm concludes that the scheduled transmission does not result in any collisions, it includes it in the scheduling matrix that is being constructed. As a result, at any given timeslot the request table (scheduling matrix) of OIS comprises of the nodes that are scheduled to transmit and the wavelengths they will transmit in.

In order to decrease the delay that a ready node experiences while waiting for OIS to compute the schedule, POSA (predictive online scheduling algorithm) [7] attempts to minimize the duration of the schedule computation process by predicting the nodes' requests for the next frame. In this direction POSA makes use of a hidden Markov chain and attempts to predict the requests of the nodes for the subsequent frame based on their requests for the previous frame. Because the algorithm does not wait for the nodes to send their requests in order to compute the schedule but starts working based on the predictions, a significant amount of time is saved.

WFF algorithm

In this section we present a scheme called Wait For Fullness (WFF), that is based on the two previous algorithms that were discussed namely OIS [6] and POSA [7]. This new protocol attempts a synthesis of the main features of OIS and POSA and results in a performance improvement in terms of channel utilization and network throughput. WFF introduces a new schedule computation mechanism called the *cleanup mechanism*. The innovation of WFF lies in the way it modifies the scheduling algorithm so that it minimizes (or even eliminates) the number of idle timeslots thus significantly improving channel utilization while maintaining a high network throughput. WFF incorporates the same prediction mechanism as POSA which when combined with the pipelining of the schedule computation time and transmission time explains the significant performance improvement that is observed.

The innovative *cleanup mechanism* of WFF acts on the scheduling matrix. It is actually a procedure during which the timeslots that contain at least one idle channel are located and logically erased so that the total number of idle timeslots is minimized and the channel utilization is increased. The function of the *cleaning mechanism* can be divided into the following four steps:

1. *Locate the timeslots that contain at least one idle channel (referred to as idle timeslots).*
2. *Logically erase these timeslots and construct the scheduling matrix without these timeslots.*

3. *Reschedule the requests that were contained in the deleted timeslots.*
4. *At regular predetermined intervals perform the refresh function and schedule all stored (in queues) packets so as to put an upper bound on the incurred service delay.*

$$D_f=\begin{bmatrix}3..2\\4..1\\2..3\\1..2\end{bmatrix},\ D`_f=\begin{bmatrix}0..0\\2..0\\2..2\\0..0\end{bmatrix}$$

Frame 1. Demand matrixes for frame f

The discovery of idle timeslots is a pretty simple procedure. After constructing the scheduling matrix according to OIS or POSA, the lines of the matrix (corresponding to channels) are scanned one by one for all columns (i.e. for all timeslots). When a slot containing at least one idle channel is located it is logically erased which means that the transmissions it contains will be performed in one of the following frames. This means that the requests that were rescheduled will be added to the new requests that the nodes will send for the following frame and the actual data will continue to be stored in queues while their transmission is being scheduled.

timeslots

	0	1	2	3	4	5	6	7	8	9	10	11
w_0	N_0	N_0	N_0	N_1	N_1	N_1	N_1	N_2	N_2	N_3	idle	idle
w_1	N_1	N_3	N_3	N_0	N_0	idle	idle	idle	idle	N_2	N_2	N_2

Table 1. Scheduling matrix contructed by OIS or POSA

It is obvious that if the algorithm always functioned according to the *cleanup mechanism* the channel utilization would be approximately equal to 100% since the number of idle slots would be almost zero. This ideal level of channel utilization and the corresponding high network throughput come at the cost of a significant delay since the number of packets waiting in queues constantly increases and their scheduling is referred for later. For this reason the *cleaning mechanism* includes a process called *refresh function* during which the contents of the waiting queues are emptied (all waiting packets are scheduled for transmission). This *refresh function* is performed at regular frame intervals (i.e. every time a constant number of frames have elapsed). For the frame that the

timeslots

	0	1	2	3	4	5	6	7	8	9	10	11
w_0	N_0	N_0	N_0	N_1	N_1	~~N_1~~	~~N_1~~	~~N_2~~	~~N_2~~	N_3	~~idle~~	~~idle~~
w_1	N_1	N_3	N_3	N_0	N_0	~~idle~~	~~idle~~	~~idle~~	~~idle~~	N_2	~~N_2~~	~~N_2~~

Table 2. Identification of idle timeslots

timeslots

	0	1	2	3	4	5
w_0	N_0	N_0	N_0	N_1	N_1	N_3
w_1	N_1	N_3	N_3	N_0	N_0	N_2

Table 3. The final sheduling matrix after step 2 of cleanup mechanism

refresh function is performed, WFF functions as OIS or POSA, and the specific frame is named as refresh frame.

The function of the *cleanup mechanism* is clarified with an example. Consider a network with 4 nodes (N_0, N_1, N_2 and N_3) that transmit data using two channels (W_0 and W_1). Assume that a total of 18 requests are submitted for frame f and that these requests are distributed, where D_f is the demand matrix for frame f (Frame 1.). From the matrix, it is evident that node N_0 requests three timeslots for channel W_0 and two timeslots for channel W_1, node N_1 requests four timeslots for channel W_0 and one timeslot for channel W_1, node N_2 requests two timeslots for channel W_0 and three timeslots for channel W_1 and node N_3 requests one timeslot for channel W_0 and two timeslots for channel W_1.

The application of OIS or POSA would result in the scheduling matrix of Table 1. At this point the *cleanup mechanism* is activated. The timeslots that include at least one idle channel are identified. These timeslots are 5, 6, 7, 8, 10, and 11 (Table 2.). The second step of the *cleanup mechanism* is to exclude these timeslots (or equivalently the corresponding transmission requests) from the scheduling matrix. This results in the scheduling matrix of Table 3. It must be pointed out that the requests included in the deleted timeslots are not overlooked. The third step of the *cleanup mechanism* is the rescheduling of these requests in the frame that follows (together with the nodes' actual requests for the next frame). Therefore two timeslots for node N_1 and two timeslots for node N_2 in channel W_0 and two timeslots for node N_2 in channel W_1 will be rescheduled. The new demand matrix that contains the cumulative nodes' requests for frame f+1, denoted as $D`_f$ is shown in Frame 1. Summarizing the schedules computed for frame f, it can be observed that the schedule length for OIS and POSA is equal to 12 timeslots while the schedule length for WFF is merely 6 timeslots. The number of idle slots for OIS and POSA is equal to 6 while there are naturally no idle timeslots for WFF.

$$D_{f+1}=\begin{bmatrix} 3..3 \\ 1..2 \\ 4..0 \\ 2..6 \end{bmatrix}, D`_{f+1}=\begin{bmatrix} 1..0 \\ 0..0 \\ 0..0 \\ 0..0 \end{bmatrix}$$

Frame 2. Demand matrixes for frame f+1

Assume that the demand matrix D_{f+1} for the next frame (i.e., frame f+1) is as shown in Frame 2. According to the *cleanup mechanism* the demand matrix for frame f+1 would be the result of the addition of tables $D`_f$+D_{f+1}. Therefore the new demand matrix for frame f+1 would be equal to

$$D`_f+D_{f+1}=\begin{bmatrix} 0..0 \\ 2..0 \\ 2..2 \\ 0..0 \end{bmatrix}+\begin{bmatrix} 3..3 \\ 1..2 \\ 4..0 \\ 2..6 \end{bmatrix}=\begin{bmatrix} 3..3 \\ 3..2 \\ 6..2 \\ 2..6 \end{bmatrix}$$

In this particular case, if we construct the scheduling matrix we will observe that contains only one timeslot with an idle channel. The rescheduled requests are contained in $D`_{f+1}$ (Frame 2.). In conclusion, for frame f+1, OIS and POSA construct a schedule of 16 timeslots while the schedule constructed by WFF has duration of 13 timeslots. There are no idle slots for WFF in contrast to OIS and POSA where the number of idle timeslots is 11.

We complete our example by considering a last set of demands D_{f+2} for frame f+2 (Frame 3.) According to the *cleanup mechanism* the demand matrix for frame f+2 would be the result of the addition of tables $D'_{f+1}+D_{f+2}$. Therefore the new demand matrix would be equal to:

$$D'_{f+1}+D_{f+2}=\begin{bmatrix}1..0\\0..0\\0..0\\0..0\end{bmatrix}+\begin{bmatrix}4..2\\3..1\\2..4\\3..1\end{bmatrix}=\begin{bmatrix}5..2\\3..1\\2..4\\3..1\end{bmatrix}$$

If we assume that *refresh function* is performed during frame f+2 we can draw some important conclusions from this example. The total number of nodes' requests for frames f to f+2 is equal to 59 (18 for the first frame, 21 for the second frame and 20 for the third). In order to service these requests OIS/POSA utilize a total of 41 timeslots (12 in frame f, 16 in frame f+1 and 13 in frame f+2) while WFF utilized a mere 32 timeslots (6 in frame f, 13 in frame f+1 and 13 in frame f+2). The total number of wasted timeslots for OIS/POSA is 23 while WFF only wasted 5 timeslots (all in the last frame when cleaning reset is performed).

$$D_{f+2}=\begin{bmatrix}4..2\\3..1\\2..4\\3..1\end{bmatrix}$$

Frame 3. Demand matrix for frame f+2

Simulation results

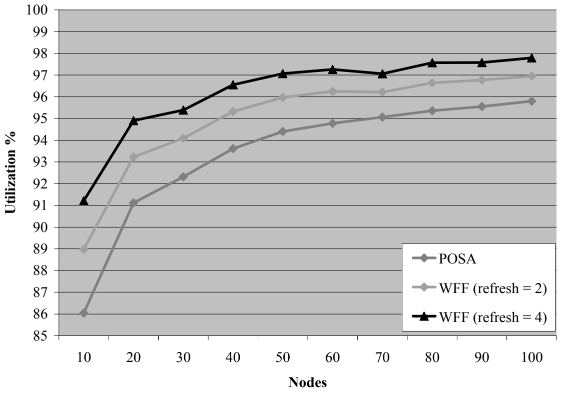

Fig.2. Channel Utilization (W = 2)

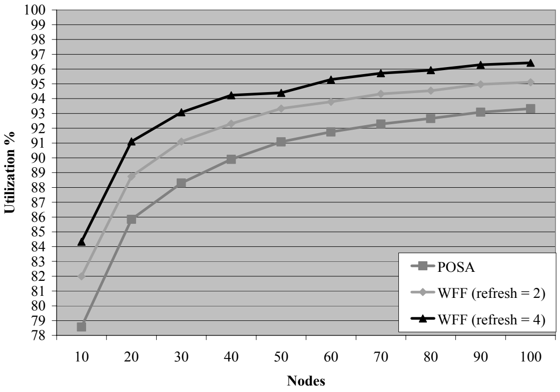

Fig.3. Channel Utilization (W = 3)

In the results of the simulation, it is assumed that N is the number of nodes; W is the number of the channels and K is the maximum value over all entries in the traffic matrix. POSA and WFF have been studied and analyzed under uniform traffic. The speed of the line has been defined at 2.4 Gbps. The metric, called refresh rate, means how often *refresh function* is executed. For example if refresh is equal to 2, then two normal frames is followed by one refresh frame.

The results from the comparison between POSA and WFF, in terms of channel utilization are shown in Figure 2 for two channels and in Figure 3 for three channels.

It is obvious that WFF remains constantly better than POSA for each number of nodes, either for 2 or for 3 channels. The results from the comparison between the two algorithms in terms of throughput are shown in Figure 4 for two channels and in Figure 5 for three channels.

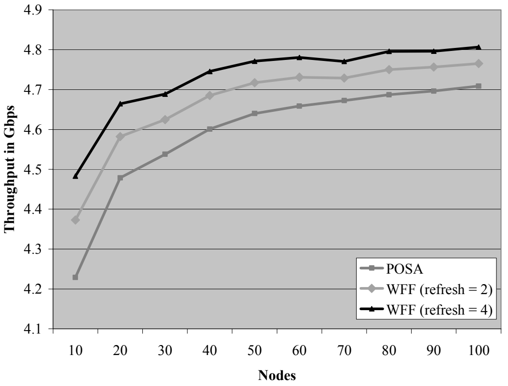

Fig.4. Network Throughput (W = 2)

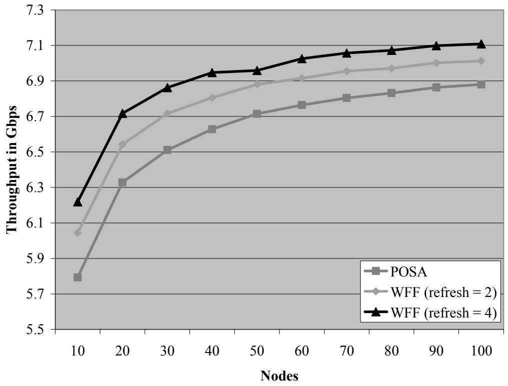

Fig.5. Network Throughput (W = 3)

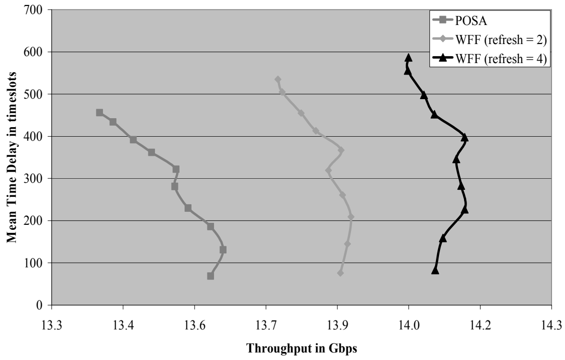

Fig.6. Throughput vs. Mean Time Delay (W = 2)

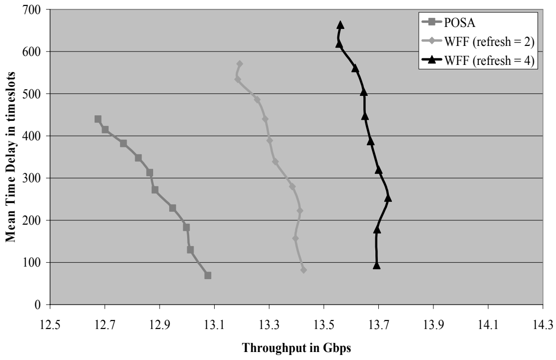

Fig.7. Throughput vs. Mean Time Delay (W = 3)

The results from the comparison between the two algorithms in terms of throughput vs. delay are presented in the Figure 6 for two channels and Figure 7 for three channels. It is obvious that for each value of the workload, WFF precedes POSA without a significant time delay.

Conclusions

This paper presented an improved protocol for WDM Broadcast-and-Select networks, which minimizes idle timeslots. The protocol eliminates the collisions by coordinating the demands of the nodes before the transmission with the help of a cleanup mechanism. According to simulation results it improves not only the schedule utilization and the throughput of the network, but also the mean time delay in relation to the throughput.

References

[1] G. I. Papadimitriou, Ch. Papazoglou, and A. S. Pompotrsis, “Optical Switching : Switch Fabrics, Techniques, and Architectures”, IEEE/OSA Journal of Lightwave Technology, vol. 21, no. 2, 2003, pp. 384-405.

[2] C.A. Brackett, “Dense wavelength division multiplexing network: Principles and applications”, IEEE J. Selected Areas Commun., vol. 8, 1990, pp. 948-964.

[3] T. E. Stern, and K. Bala, Multiwavelength Optical Networks, Addison-Wesley, 1999.

[4] P. Green, “Progress in optical networking, IEEE Communications magazine”, vol. 39, no. 1, 2001, pp. 54-61.

[5] G. I. Papadimitriou, P. A. Tsimoulas, M. S. Obaidat, and A. S. Pomportsis, Multiwavelength Optical LANs, Wiley, 2003.

[6] K. M. Sivalingam, J. Wang, J. Wu and M. Mishra, “An interval-based scheduling algorithm for optical WDM star networks”, Photonic Network Communications, vol. 4, no. 1, 2002, pp. 73-87.

[7] E. Johnson, M. Mishra, and K. M. Sivalingam, “Scheduling in optical WDM networks using hidden Markov chain based traffic prediction, Photonic Network Communications”, vol. 3, no. 3, 2001, pp. 271-286.