Increasing Channel Utilisation using Segmentation based Channel Scheduling Algorithms in OBS Network

Abhilash Mandloi
Electronics Engineering Department,
S.V.N.I.T.
Surat, Gujarat, India

V. Mishra
Electronics Engineering Department,
S.V.N.I.T.
Surat, Gujarat, India

ABSTRACT

Optical burst switching (OBS) is emerging as the switching technology for next generation optical networks. Advantages of optical packet switching and circuit switching are combined in OBS and overcoming their limitations. Data (or payload) is separated from control packet. A control packet is sent before the payload to reserve the resources on the path to the destination of payload. When a control packet arrives at an intermediate node a wavelength scheduling algorithm is used by the scheduler to schedule the data burst on an outgoing wavelength channel. The required information to schedule a data burst are arrival time and duration of data burst, which are obtained from the control packet. On the other hand, scheduler keeps availability of time slots on every wave length channel and schedule a data burst in a channel depending upon the scheduling algorithm it uses. Different scheduling algorithms have been proposed in literature to schedule payload/ data burst. They differ in burst loss and complexity. Depending upon the channel selection strategy, they can be classified as Horizon and Void filling algorithm. Though these algorithms give less burst loss but channel utilization is very less. In this paper we introduce a new approach, which will give less burst loss and also utilize existing channels in efficient way. Also the performance of this proposed scheduling scheme has been analyzed and compared it with the existing void filling schemes. It is shown by simulations that the proposed scheme gives somewhat better performances compared to the existing schemes in terms of channel utilization and packet loss.

General Terms

Algorithms for channel scheduling in OBS networks

Keywords

Channel scheduling, channel utilization, horizon void filling, LAUC-VF, FFUC-VF, OBS.

1. INTRODUCTION

Optical Burst Switching (OBS) [1] is a new paradigm in optical networks that has got the advantages of packet switching as well as circuit switching. OBS architecture is composed of three types of node i.e. ingress node, egress node and core node. There are two main classification of scheduling algorithms [2-3]. They are Horizon Algorithms and Void filling scheduling schemes. Horizon algorithm considers the channels which have no scheduled data burst at or after current time t and the channels are called Horizon channels. Void filling algorithms consider the channels which have unused duration in between [4] two scheduled data bursts. These are called Void channels. The example of Horizon algorithms are FFUC, LAUC and Void filling algorithms are FFUC-VF, LAUC-VF and Min-EV. Horizon algorithms are easy to implement and burst loss ratio is high. Whereas burst loss ratio is lower in Void filling algorithms but complex switching are required to implement. Among the void filling algorithms, burst loss ratio is lower in LAUC-VF and Min-EV. LAUC-VF[5-6] schedule a data burst in a void channel such that the time difference between arrival data bursts starting time and previous scheduled data bursts end time is minimum[7-10]. Whereas Min-EV schedule a data burst in a void channel, such that the time difference between a scheduled data[11] bursts start time and arrival data bursts end time is minimum. Both, LAUC-VF and Min-EV consider only one side of a void. There may be a possibility, in which a smaller data burst will be scheduled in a larger void where as a bigger data burst will be dropped. This will lead to higher burst blocking and lower channel utilization. There has been growing interest in the realization of an optical Internet (IP over WDM)[12-14] to support these future applications. Optical burst switching (OBS) is one approach that can be used to efficiently transmit bursts of IP data over WDM networks. In an OBS network, a WDM link consists of multiple data channels to transmit the payload (data bursts) and one or more dedicated control channels to transmit the corresponding[15-17] burst header packets (BHP). The BHP is transmitted ahead of the burst in order to control the switches along the burst's route. The burst follows the header without waiting for an acknowledgment for the connection establishment. The header and the data burst are separated at the source, as well as subsequent intermediate nodes, by an offset time[18]. The offset time allows for the header to be processed at each node before the arrival of the burst; thus, no fiber delay lines are necessary at the intermediate nodes to delay the burst while the header is being processed. The control message may also specify the duration of the burst in order to let a node know when it may recon its switch for the next burst, a technique known as *Just-Enough-Time* (JET)[19]. One of the key issues in OBS is the scheduling of bursts on output data channels. Currently, contention resolution techniques used in scheduling are wavelength conversion and buffering. In wavelength conversion, if multiple bursts on the same wavelength compete for the same output port at the same time, then the bursts are shifted to different independent wavelengths. In optical buffering, FDLs are used to[20] provide limited delay to the data bursts, proportional to the length of fiber delay line, in order to resolve the contention. Current data channel scheduling algorithms that use wavelength conversion and FDLs include latest available unscheduled channel (LAUC), and latest available unscheduled channel with void filling (LAUC-VF) However, these techniques drop the burst completely if all of the data channels are occupied at the arrival time of the burst.

Instead of dropping the burst in its entirety, it is possible to drop only the overlapping parts of a burst. This technique is referred to as burst segmentation.

2. REVIEW OF VARIOUS SCHEDULING SCMES

2.1 Review of Horizon Scheduling Methods

Horizon scheduling algorithms consider the unscheduled channels to schedule a data burst. It does not consider the availability of void within a channel, which could otherwise be used in channel scheduling. For example consider the Figure 1. In this Figure there two data bursts a and b are scheduled on channel[4] *1* and data burst *c* on channel *2*. For horizon scheduling algorithms, channel *1* is available at time instant *t* and channel *2* is at *t'*. Suppose a data burst *x* arrives. Horizon scheduling algorithms will schedule the data burst *x* on channel *2* as shown in Figure 1. They do not consider the voids within a channel. In channel *1* there exist a void between data bursts *a* and *b* within which the data burst *x* could have been scheduled. Thus, horizon scheduling algorithms are not efficient in terms of channel utilization and gives rise to higher burst loss. On the other hand, void filling algorithms consider both unscheduled and void channel to schedule data bursts. For the scenario as shown in Fig. 1, void filling algorithms will schedule data burst x on channel. Thus, increases the channel utilization. Any data burst arriving between *t'* and *t* could be schedule on channel *2*, which otherwise could have been dropped in horizon algorithms. Thus, horizon scheduling algorithms are not efficient in terms of burst loss and channel utilization in comparison to void filling algorithms.

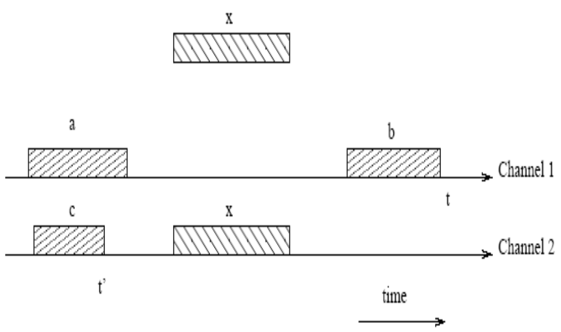

Fig 1. Scheduling by horizon scheduling technique

2.2 Scheduling by Void Filling Techniques

Though void filling algorithms are efficient than horizon scheduling algorithms, but they are not the optimal scheduling algorithms. The limitations of the void filling algorithms such as LAUC-VF and Min-EV [5] algorithms lies in the fact that they consider only one side of a void. LAUC-VF, consider the void created between incoming data bursts start time and previous scheduled data bursts end time. Whereas Min-EV, consider the void created between scheduled data bursts start time and incoming data bursts end time. Due to this smaller size data bursts may be scheduled in a larger void whereas bigger size data bursts may get blocked. In the following section, a brief description of the limitations in terms of blocking and channel utilization of LAUC-VF and Min-EV void filling algorithms is presented.

2.3 Scheduling by LAUC-VF Algorithm

In OBS data bursts are of variable lengths. If a smaller data burst arrive earlier than a larger size data burst then void filling algorithm may schedule[4] the smaller data burst on a larger void and the larger size data burst may be dropped due to unavailability of data channel. This can happens in void filling algorithms due to their consideration of one side of a void. For example consider the Figure 2. In this figure data burst b0 and b1 are schedule on channel 1, b2 and b3 on channel 2 and b4 and b5 on channel 3. On channel 1 the end time of data burst b0 is t1 and start time of data burst b1 is t2. Data burst b2 has end time of t3 and data burst b3 has start time of t4 on channel 2. Similarly, for data burst b4, t5 is the end time and for data burst b5, t6 is the start time. Suppose three data bursts B0, B1 and B2 arrive at a node. Arrivals of control packet for data bursts are shown in control channel. Control packet CB0 for data burst B0 has arrived first then CB1 for data burst B1, and finally CB2 for data burst B2arrived in that order. Start time and end time of data burst B0 is tb0 and te0, for data burst B1 is tb1 and te1 and for data burst B2 is tb2 and te2. Scheduling of the data burst onto a channel depend on the type of scheduling algorithm node is using. That is, whether node is using LAUC-VF or Min-EV algorithm. Two different cases have been presented. One for scheduling with LAUC-VF and the other for Min-EV algorithms. Since the data burst B0, B1 and B2 arrive in that order, the scheduler will schedule data burst B0 first, then B1 and followed by B2 in that order. LAUC-VF algorithm tries to schedule a data burst on a void, such that difference between the start time of a new data burst and the end time of a previous scheduled data burst whose end time is prior to the new data burst start time will be minimum. Data burst b0, b2 and b4 have their end time prior to data burst B0's start time. Differences between the start time of B0 and end time of b0, b2 and b4 are (tb0 − t1), (tb0 − t3) and (tb0 − t5) respectively. Of these LAUC-VF, schedule the data burst on a channel, that has the minimum difference. Difference between the start time of data burst B0 and end time of data burst b0 is minimum. That is (tb0 − t1) is the minimum value of the three values (tb0 − t1), (tb0 − t3) and (tb0 − t5). So LAUC-VF schedule the data burst B0 on channel 1. When the request CB1 for data burst B1 arrives, there is no available channel to schedule the data burst B1, hence B1 is dropped. Data burst B2 can be scheduled in channel 2. In Figure 2 the duration of void in channel *1*, *2* and *3* are ($t2-t1$), ($t4-t3$) and ($t6-t5$) respectively. Higher the fraction of void utilized higher will be channel utilization. Fraction of void utilized is the ratio of the data burst duration scheduled on the void to the void duration. In Figure 2, LAUC-VF schedules the data burst *B0* in the void of channel 1. The fraction of void utilized is ($te0 - tb0$)/ ($t2 - t1$). Of these the fraction ($te0 - tb0$)/ ($t2 - t1$) is smaller. Scheduling data burst *B0*, in channel *1*, *2* and *3*, the fraction of void utilized will be ($te0 - tb0$)/ ($t2 - t1$), ($te0 - tb0$)/ ($t4 - t3$) and ($te0-tb0$)/ ($t6-t5$) respectively. This is because ($t2- t1$) > ($t4 - t3$) > ($t6 - t5$). Thus scheduling data burst *B0* in channel 1, gives rise to lesser channel utilization. Moreover, this creates a void ($t2 -te0$) of considerable duration.

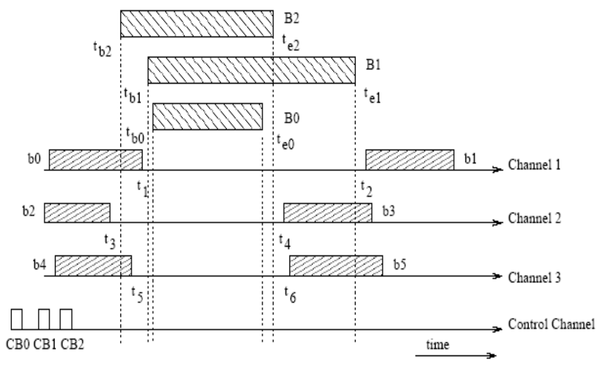

Fig 2. LAUC-VF and Min-EV Algorithms

2.4 Scheduling by Min EV Algorithm

In Min-EV scheduling algorithm, an incoming data burst is schedule on a channel, such that the start time of an already scheduled data burst and end time of an incoming data burst is minimum. Here only those schedule data bursts whose start time is after the end time of the incoming data burst are considered. In Figure 2, data bursts b1, b3 and b5 have start time after the end time of data burst B0. Difference between the end time of data burst B0, and the start time of data burst b1, b3 and b5 are (t2 − te0), (t4 − te0) and (t6 − te0) respectively. Of these (t4−te0) is the minimum. So the data burst B0 is schedule on channel 2. Similarly data burst B1 is schedule on channel 1. However, data burst B2 cannot be schedule as there is no wavelength channel is available.

2.5 Channel Utilization in LAUC-VF and Min EV Technique

In Fig 2 the duration of void in channel *1*, *2* and *3* are (*t*2−*t*1), (*t*4−*t*3) and (*t*6−*t*5) respectively. Higher the fraction of void utilized higher will be channel utilization. Fraction of void utilized is the ratio of the data burst duration scheduled on the void to the void duration. In Figure 3 LAUC-VF schedule data burst *B0* in the void of channel *1*. The fraction of void utilized is (*te*0 − *tb*0)/ (*t*2 − *t*1). Of these the fraction (*te*0 − *tb*0)/ (*t*2 − *t*1) is smaller. Scheduling data burst *B0*, in channel *1*, *2* and *3*, the fraction of void utilized will be (*te*0 − *tb*0)/ (*t*2 − *t*1), (*te*0 − *tb*0)/ (*t*4 − *t*3) and (*te*0−*tb*0)/ (*t*6−*t*5) respectively. This is because (*t*2− *t*1) > (*t*4 − *t*3) > (*t*6 − *t*5). Thus scheduling data burst *B0* in channel *1*, gives rise to inefficient channel utilization. Moreover, this creates a void (*t*2 −*te*0) of considerable duration. Min-EV algorithm schedule data burst *B0* in channel *2*. Fraction of void utilized is higher than that of scheduling on channel *1* and lower than scheduling on channel *3*. Scheduling *B0* in channel *2*, void of channel *3* remains utilized. Thus, it is observed that the channel utilization is lower in both LAUC-VF and Min-EV. This is because both algorithms consider only one side of a void i.e., either the start or end side of a void. Next we propose a new channel scheduling algorithm which considers both end of a void in scheduling and it utilizes void efficiently.

2.6 BFVF Algorithm

Best Fit Void Filling (BFVF)[18], which attempts to maximize the channel utilization and minimize the burst loss. This algorithm first selects all possible void channels, on which the data burst can be scheduled. Then selects one of the possible void channel such that the void utilization factor is maximum. We calculate the void utilization factor as:

utilization n = (a × 100) / x (2) (1)

Where *a* is the data burst length and *x* is the void length. In Figure 3, data burst *B0* can be schedule any one of the channel *1*, *2* and *3*. Void utilization factor for *B0* on channel *1*, *2* and *3* are (*te*0 − *tb*0)/ (*t*2 − *t*1), (*te*0 − *tb*0)/ (*t*4 − *t*3) and (*te*0 − *tb*0)/ (*t*6 − *t*5) respectively. Void utilization factor for channel *3* is maximum, since (*t*6 − *t*5) < (*t*4 − *t*3) < (*t*6 − *t*5). So BFVF algorithms selects channel *3* to schedule the data burst *B0*. Similarly data burst *B1* is schedule on channel *1* and *B2* on channel *2*. BFVF algorithm all three data burst *B0*, *B1* and *B2* can be scheduled on channel *3*, *1* and *2* respectively as shown in Figure 3. Thus the channel utilization is higher and burst loss ratio is lower in our propose scheme than in LAUC-VF and Min-EV. Though BFVF has the highest Channel Utilization factor, it has got the limitations of dropping of burst in large number and the burst loss is high.

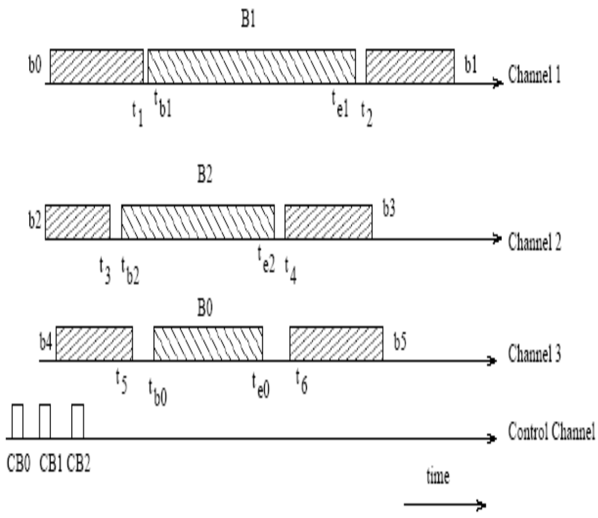

Fig 3. Scheduling by BFVF Algorithm

3. SEGMENTATION BASED BEST FIT VOID FILLING (MSBFVF) TECHNIQUE

This algorithm schedules the data bursts according to CUF but when it comes to segmentation the scheduling process is different as compared to BFVF scheme. Scheduling process is done without segmentation according to CUF when overlap of bursts with channels does not occur. But when overlap occurs among those bursts with channels then this algorithm first schedules the non-overlapping part of burst on suitable channels and then it segments the overlapping portion. Now it compares the segmented part with other arriving bursts as shown in Fig 4.

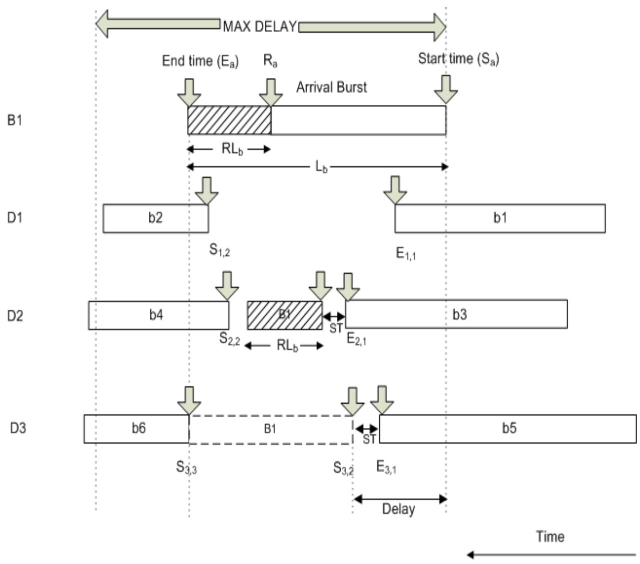

Fig 4. Scheduling by MSBFVF Algorithm

After comparing the segmented portion with other arriving bursts it selects the burst which has bigger length than that segmented part and it schedules that burst on suitable channel. As a result burst loss ratio will be less. Implementation of this algorithm is shown below with different cases.

The steps involved in SBFVF Method are as follows:

1: Enter number of bursts & channels.

2: Enter burst lengths & void lengths of channels.

3: If burst length is greater than start time of void & MAX_DELAY then entire burst is dropped.

4: If no, calculate CUF (Channel Utilization Factor) for all channels.

CUF= (a *100) / x where

a' is the data burst length and

'x' is the void length.

5: If CUF is less than 100% for the first arriving burst then select the channel which is closer to 100% and schedule the burst on that selected channel.

6: If CUF is greater than 100% (over laps) for arriving burst then select the channel which is greater & closer to 100%.

7: Then segment the burst of overlapping portion.

8: Now compare the segmented burst length with other arriving bursts & select the burst whose length is bigger than the segmented burst.

9: Calculate CUF for the selected burst.

10: If CUF is less than 100%, schedule the selected burst on suitable channel.

11: If CUF is greater than 100% go to step 6.

12: Stop

In this introduced scheme it does not require finding the minimum gap instead of it, finding the utilization factor of every channel and select the best utilization factor and scheduled the burst according to it (see Fig 4).

4. SIMULATION AND RESULTS

Simulation Set Up is shown in Fig 5. Simulation has been done for taking various conditions for different algorithms. Four void filling algorithms i.e. FFUC-VF, LAUC-VF, BFVF and SBFVF (newly introduced) have been simulated under similar conditions for different cases and channel utilization has been calculated accordingly.

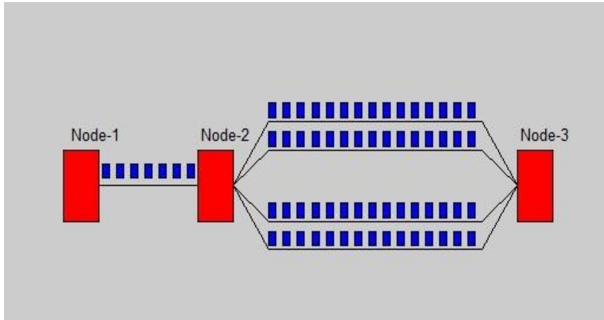

Fig. 5. Simulation set up

Case I: fixed number of channels & as burst no. increases with fixed burst length.

In this condition following values of input have been taken as mentioned below:

Number of Channels Taken: five

Maximum Delay permissible: 255 μs

Switching Time for each Bs: 10 μs

Void Length of 5 channels was taken as 60 μs, 50 μs, 20 μs, 30 μs, 50 μs and burst duration of each burst (incremented from 1 to 5) was taken 40 μs. Besides this for all five cases Switching time (ST) taken was = 10 μs Maximum Delay = 255 μs.. It is clear that SBFVF algorithm performs better in terms of channel utilization as compared to other three algorithms (see Fig 6 and 7). The main consideration for choosing fixed burst length was to investigate the performance of the algorithms constant bit rate (CBR).

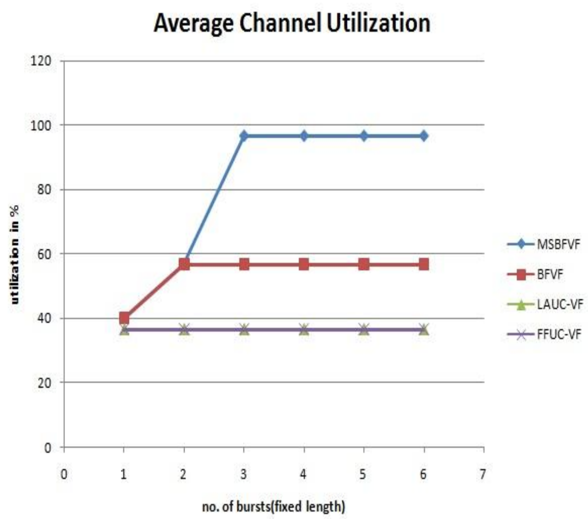

Fig 6. Channel Utilization for fixed number of channels as number of bursts (fixed size) are increased (case I).

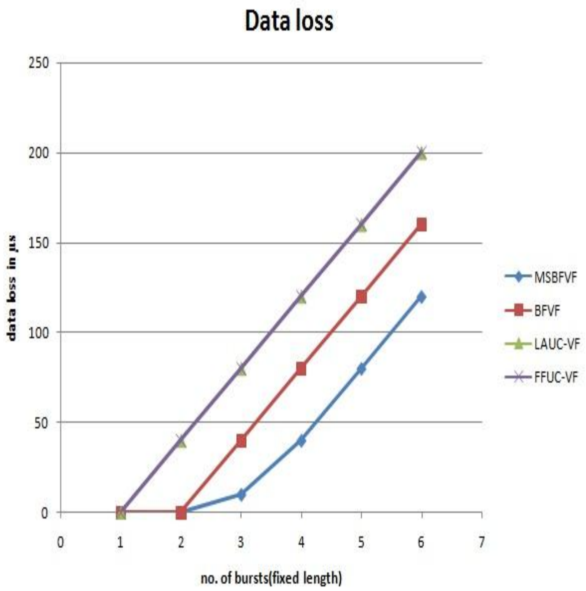

Fig 7. Burst Loss for fixed number of channels as number of bursts (fixed size) is increased (case I).

Case II: Fixed number of bursts & as channel number increases (from 1 to 5) In the above case following values have been taken as input: Bursts Duration $B_{1=}$ 50 µs, B_2=50 µs, B_3=130 µs, B_4=120 µs, B_5=20 µs. Channel Lengths (variable Number of channels): C_1= 150 µs, C_2=140 µs, C_3=60 µs, $C_{4=}$20 µs, C_5=60 µs. Further MAX_DELAY and Switching Time were taken same as Case I. It is clear that MSBFVF algorithm performs better in terms of channel utilization and bursts loss. The burst loss is minimum and Channel Utilization is maximum for the Case II (See Fig 8 and 9).

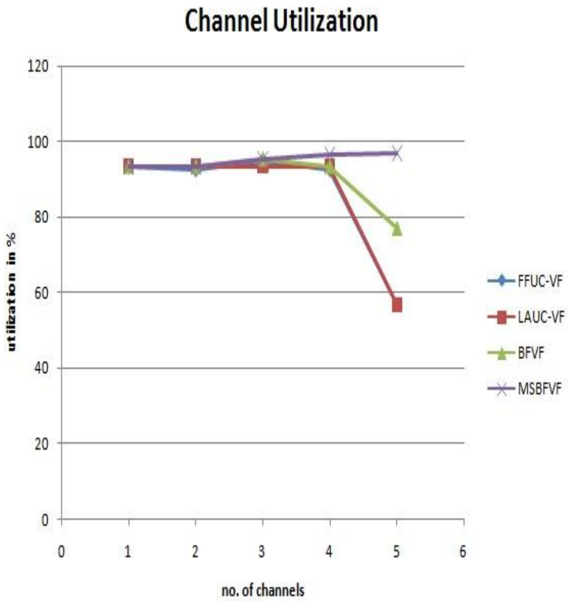

Fig 8. Channel utilization for case II

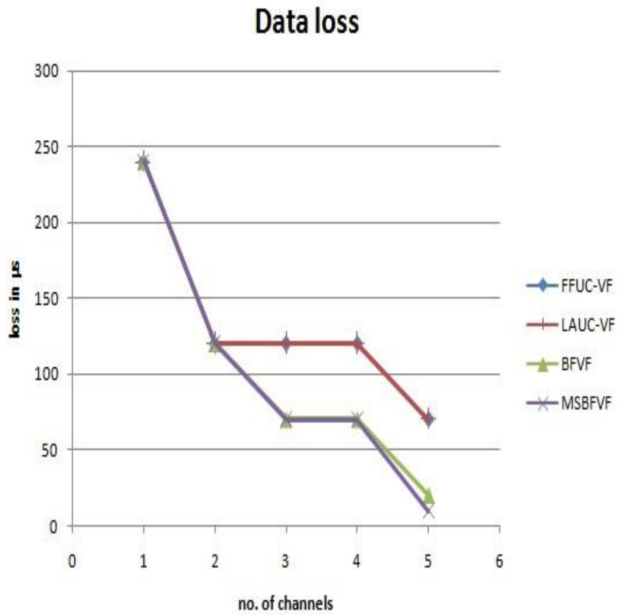

Fig 9. Packet Loss for Case II

Case III: as burst number increases of variable size & fixed number of channels with fixed channel length). The input parameters taken for the third case are: Void Length of all channels = 50 µs. Bursts lengths are 20 µs, 40 µs, 60 µs, 80 µs, 100 µs. Plots are shown in Fig 10 and Fig 11.

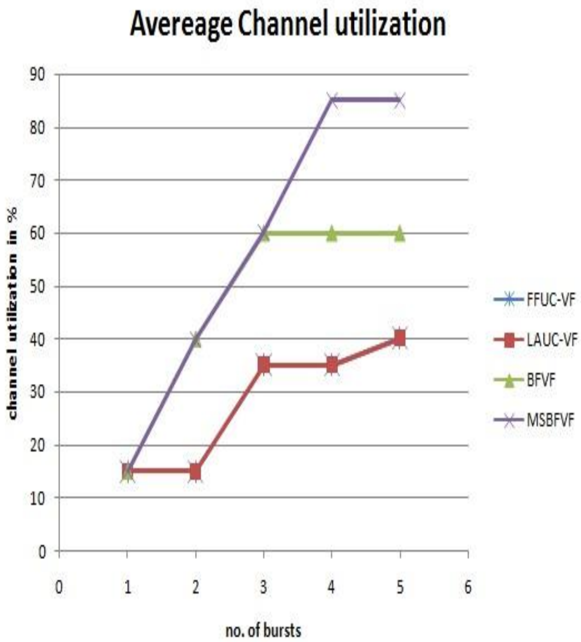

Fig 10. Channel Utilization` for different schemes for Case III

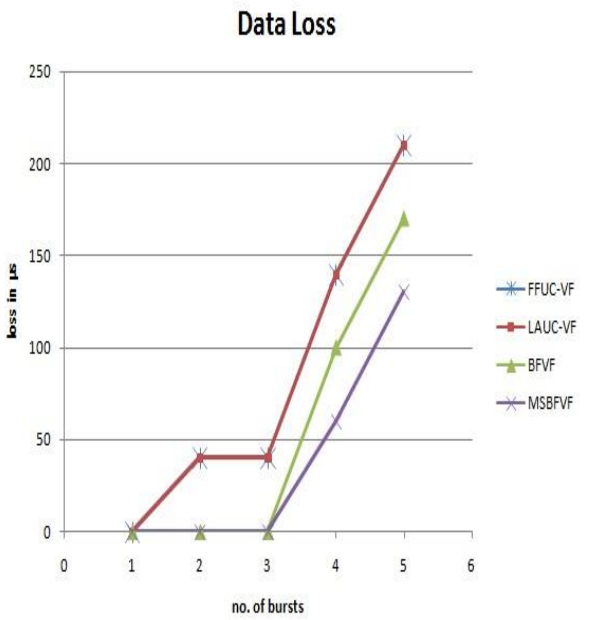

Figure11. Burst Loss for different schemes for Case III

Case IV: linear increment in burst size as channel no. is fixed. In this case both number of bursts an well as number of channels were increased. The input parameters were taken as follows: Burst lengths are 20 µs, 40 µs, 60 µs, 80µs, 100 µs. Channel lengths 50 µs, 40 µs, 70 µs, 60 µs. MAX_DELAY and switching time were taken same as previous cases. The plots of channel utilization and bursts loss are shown in Figure 12 and 13 respectively.

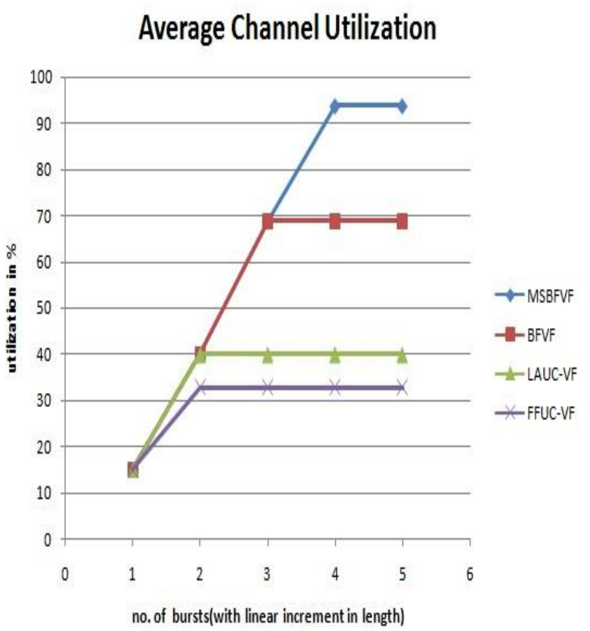

Fig 12. Channel utilizations for various schemes for case IV

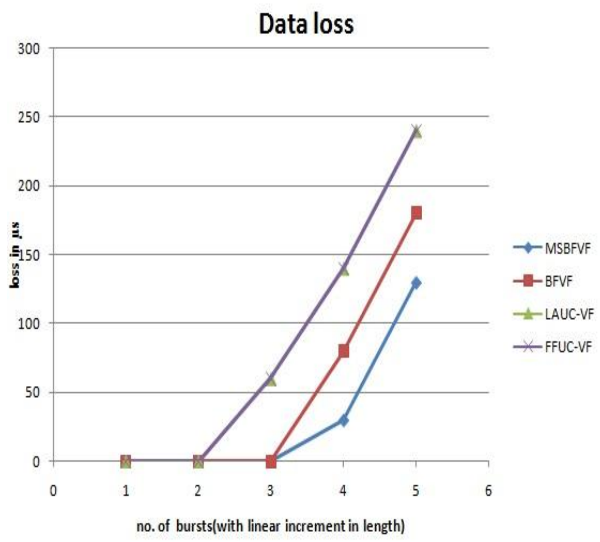

Fig 13. Packet loss for various schemes for case IV

5. CONCLUSION

Various scheduling scheme for Optical Burst Switching have been discussed and a new algorithm of channel scheduling in OBS (MSBFVF) has been introduced. Simulation results show that the MSBFVF Scheme was best in terms of bandwidth utilization; number of channels used or channel utilization and burst loss. It is found that the delay depends on the starting and ending times of the bursts and also the starting and ending time of void in the channels. This has been also evident by results. In this paper it is shown by simulations that this algorithm showed better performance in simulated cases as compared to the existing scheme. It can be concluded that proposed segmentation based scheme gave better result as compared to non segmentation based schemes.

6. REFERENCES

[1] S. Amstutz, Burst switching - An update , IEEE Commun. Mag., pp.50–57, Sept. 2006.

[2] Kostas Ramantas and Kyriakos Vlachos, A TCP-Specific Traffic Profiling and Prediction Scheme for Performance Optimization in OBS Networks", Journal of Optical Communications and Networking, Vol. 3, Issue 12, pp. 924-936 (2011)

[3] L. Tancevski, A. Ge, G. Castanon, and L. Tamil, A New Scheduling Algorithm for Asynchronous, Variable Length IP Traffic Incorporating Void Filling", Proceedings, OFC'2006.

[4] Vinod Vokkarane, Guru P.V. Thodime, Venkata U.B. Challagulla, and Jason Jue, "Channel Scheduling Algorithms using Burst Segmentation and FDLs for Optical Burst-Switched Networks", submitted to, IEEE Conference, 2008

[5] V.M. Vokkarane, G.P.V. Thodime, V.B.T. Challagulla, and J.P. Jue. "Channel scheduling algorithms using burst segmentation and FDLs for optical burst-switched networks". In Proceeding, IEEE ICC, volume 2, May 2008.

[6] Kouji Hirata, Takahiro Matsuda, and Tetsuya Takine, Three-Domain Burst Scheduling in Optically Burst-Switched WDM Networks Journal of Optical

Communications and Networking, Vol. 2, Issue 6, pp. 332-343 (2010)

[7] M. Yoo and C. Qiao, Just-Enough-Time (JET): A High Speed Protocol for Bursty Traffic in Optical Networks, IEEE/LEOS Conf. on Technologies For a Global Information Infrastructure, pp. 26-27, Aug. 2005.

[8] Burmeister, E.F., Blumenthal, D.J., Bowers, J.E, A comparison of optical buffering technologies, Optical Switching and Networking Volume 5, Issue 1, March 2008, pp: 10-18

[9] A. Mandloi, V.Mishra A Segmentation Based Channel Scheduling Scheme in Optical Burst Switching Networks, International Journal of Computer Application (IJCA), August 2012

[10] A. Detti and M. Listanti, Application of Tell and Go and Tell and Wait Reservation Strategies in a Optical Burst Switching Network: a Performance Comparison, Proceedings of the 8th IEEE International Conference on Telecommunications (ICT 2007), Bucharest, June 2007.

[11] sungchang Kim, Namook Kim, Minho Kang, "Contention Resolution for Optical Burst Switching Networks Using Alternative Routing", Proceedings IEEE, ICC 2002, New York, NY, April-May 2006

[12] M. Duser and P. Bayvel, Performance of a dynamically wavelength-routed optical burst switched network IEEE Photonics Technology Letters, Volume: 14 Issue: 2 , Feb. 2006 pp. 239 -241.

[13] M. Düser and P. Bayvel, Analysis of Wavelength-Routed Optical Burst-Switched Network Performance, Optical Communication, ECOC '01, vol.1 , 2007, pp. 46 -47.

[14] Conor McArdle, Liam P. Barry, A two-moment performance analysis of optical burst switched networks with shared fibre delay lines in a feedback configuration, Optical Switching and Networking volume 9, Issue 4, November 2012, pp. 323–335

[15] L. Tancevski, A. Ge, G. Castanon, and L. Tamil, A New Scheduling Algorithm for Asynchronous, Variable Length IP Traffic Incorporating Void Filling, Proceedings, OFC’2006.

[16] K. Dolzer and C. Gauger, On burst assembly in optical burst switching networks - a performance evaluation of Just-Enough-Time", Proceedings of the 17th International Teletraffic Congress (ITC 17), Salvador, September 2007.

[17] Y. Xiong, M. Vanderhoute, and H.C. Cankaya, Control architecture in optical burst-switched WDM networks, IEEE Journal on Selected Areas in Communications, Vol. 18, No. 10, October 2006, pp. 1838-185.

[18] M. Nandi, A. K. Turuk, D. K. Puthal and S. Dutta, Best Fit Void Filling Algorithm in Optical Burst Switching Networks National Institute of Technology, Rourkela, India ICETET-200

[19] L. Xu, H.G. Perros, and G. Rouskas, Techniques for optical packet switching and optical burst switching, IEEE Communications Magazine, Vol. 39, No. 1, January 2001, pp. 136-142

[20] J.Y. Wei, J.L. Pastor, R.S. Ramamurthy, and Y. Tsai, Just-in-time optical burst switching for multiwavelength networks, Proceedings of the 5th IFIP TC6 International Conference on Broadband Communications (BC '99), Hong Kong, November 2005, pp. 339-352.