BEST FIT VOID FILLING SEGMENTATION BASED ALGORITHM IN OPTICAL BURST SWITCHING NETWORKS

A.K Rauniyar[1], A.S Mandloi[2]

[1]Department of Electronics and Communication, Advanced College of Engineering and Management, Kupondole, Lalitpur, Nepal

Email Address: amitu931@gmail.com

[2]Department of Electronics and Communication, Sardar Vallabhbhai National Institute of Technology, Surat, India

Email Address: asm@eced.svnit.ac.in

Abstract

Optical Burst Switching (OBS) is considered to be a promising paradigm for bearing IP traffic in Wavelength Division Multiplexing (WDM) optical networks. Scheduling of data burst in data channels in an optimal way is one of a key problem in Optical Burst Switched networks. The main concerns in this paper is to schedule the incoming bursts in proper data channel such that more burst can be scheduled so burst loss will be less. There are different algorithms exists to schedule data burst on data channels. Non-preemptive Delay-First Minimum Overlap Channel with Void Filling (NP-DFMOC-VF) and Non-preemptive Segment-First Minimum Overlap Channel with Void Filling (NP-SFMOC-VF) are best among other existing segmentation based void filling algorithms. Though it gives less burst loss but not existing the channel utilization efficiently. In this paper we propose a new approach, which will give less burst loss and also utilize existing channels in efficient way. Also analyze the performance of this proposed scheduling algorithm and compare it with the existing void filling algorithms. It is shown that the proposed algorithm gives some better performances compared to the existing algorithms.

Keywords: *OBS, Scheduling Algorithm, Void Filling Algorithm, NP-DFMOC-VF, NP-SFMOC-VF, Channel Utilization.*

1. Introduction

Optical burst switching (OBS) [1] 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 [2] is used by the scheduler to schedule the data burst on an outgoing wavelength channel. The required information to schedule a data burst is arrival time and duration of data burst, which are obtained from 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. 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 two scheduled data bursts. These are called Void channels. The example of non segmentation Horizon algorithms are FFUC, LAUC and non segmentation Void algorithms are FFUC-VF [3], LAUC-VF [4,5,6,7] and Min-EV [8]. The example of segmentation Horizon algorithms are Non preemptive Minimum

Overlap Channel (NP-MOC) [9], Non-preemptive Delay-First Minimum Overlap Channel (NP DFMOC) [9] and Non-preemptive Segment-First Minimum Overlap Channel (NP-SFMOC) [9]. And the example of non segmentation void filling algorithms are Non preemptive Minimum Overlap Channel with Void Filling (NP-MOC-VF) [9], Non-preemptive Delay-First Minimum Overlap Channel with Void Filling (NP-DFMOC-VF) [9] and Non-preemptive Segment First Minimum Overlap Channel with Void Filling (NP-SFMOC-VF) [9]. Horizon algorithms are easy to implement and burst loss ratio is high, where as burst loss ratio is lower in Void filling algorithms but complex switching are required to implement. All, LAUC-VF, Min-EV, NPMOC-VF, NP-DFMOC and NP-SFMOC-VF consider 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. In this chapter we propose a new channel scheduling algorithm which attempts to make efficient utilization of existing void within a channel. Thus, giving rise to higher channel utilization and lower blocking probability. Rest of the paper is organized as follows. Literature Review of the existing void filling algorithms are explained in Section 2.Methodology of the proposed best fit void filling algorithm is explained in scheme with Section 3. We compare our proposed scheme algorithm with NP-DFMOC-VF and NP-SFMOC-VF. Comparison and simulation results are presented in Section 4. Finally, some conclusions are drawn in Section 5.

2. Literature Review

In the following subsection a brief description of existing NP-DFMOC-VF and NP-SFMOC-VF void filling algorithms is presented.

2.1 Non-preemptive Delay First Minimum Overlap Channel with Void Filling (NP-DFMOC-VF)

The NP-DFMOC-VF calculates the delay until the first void on every channel and then selects the channel with minimum delay. If a channel is available, the unscheduled burst is scheduled on the free channel with minimum gap. If all channels are busy and the starting time of the first void is greater than or equal to the sum of the end time, Ea, of the unscheduled burst and MAX _DELAY, then the entire unscheduled burst is dropped. Otherwise, the unscheduled burst is delayed until the start of the first void on the selected channel, where the non-overlapping burst segments of the unscheduled burst are scheduled, while the overlapping burst segments are dropped. In case the start of the first void is greater than the sum of the start time, Sa, of the unscheduled burst and MAX_DELAY, then the unscheduled burst is delayed for MAX_DELAY and the non-overlapping burst segments of the unscheduled burst are scheduled, while the overlapping burst segments are dropped. For example, consider Fig. 1. By applying the NP-DFMOC-VF algorithm, the data channel D1has the minimum delay, thus the unscheduled burst is scheduled on D1 after delaying the burst using FDLs. In this case, the overlapping segments of the burst are dropped though there is availability of channels D2 and D3 as shown in figure.

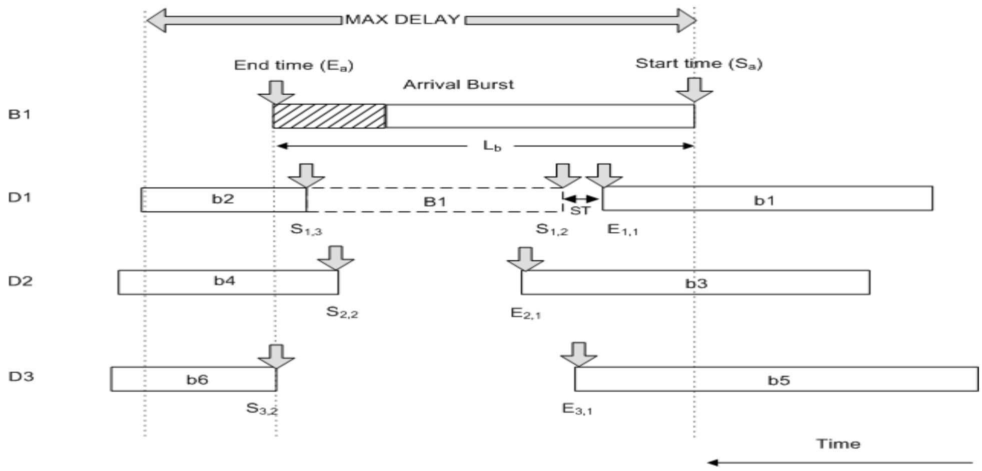

Fig 1 Illustration of NP-DFMOC-VF algorithm

Though there is presence of channels (D2 and D3) they can be only used for arrival of new bursts, the overlapping segments of the burst B1 are dropped and thus cannot be rescheduled which is the limitations of this algorithm. Hence to overcome this effect we move further to the next algorithms as discussed below.

2.2Non-preemptive Segmented First Minimum Overlap Channel with Void Filling (NP-SFMOC-VF)

The NP-SFMOC-VF algorithm calculates the loss on every channel and then selects the channel with minimum loss. If a channel is available, the unscheduled burst is scheduled on the free channel with minimum gap. If all channels are busy and the starting time of the first void is greater than or equal to the sum of the end time, Ea, of the unscheduled burst and MAX_DELAY, then the entire unscheduled burst is dropped. If the starting time of the first void is greater than or equal to the end time, Ea, of the unscheduled burst, the NP-DFMOC-VF algorithm is employed.

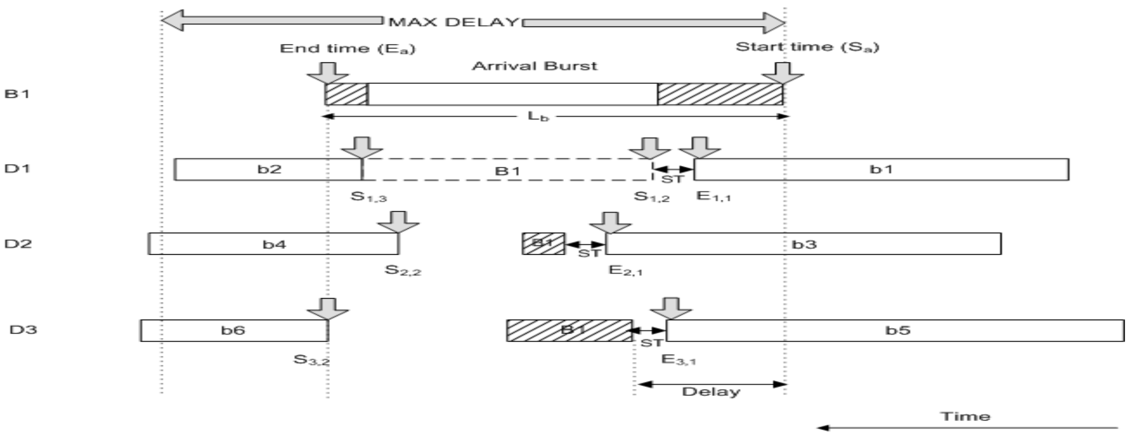

Fig 2 Illustration of NP-SFMOC-VF algorithm

Otherwise, the unscheduled burst is segmented (if necessary) and the non-overlapping burst segments are scheduled on the selected channel, while the overlapping burst segments are re-scheduled. For the rescheduled burst segments, the algorithm calculates the delay required until the start of the next void on every channel and selects the channel with minimum delay. The re-scheduled burst segments are delayed until the start of the first void on the selected channel. The non-overlapping burst segments of the re scheduled burst are scheduled, while the overlapping burst segments are dropped. In case the start of the next void is greater than the sum of the start time, Sa, of the unscheduled burst and

MAX_DELAY, the re-scheduled burst segments are delayed for *MAX_DELAY* and the non-overlapping burst segments of the rescheduled burst are scheduled, while the overlapping burst segments are dropped. For example, in Fig. 2, we observe that the data channel D1 has the minimum loss, thus the unscheduled burst is scheduled on D1, and the unscheduled burst B1 has both head overlapping and tail overlapping on which head overlapping re-scheduled burst segments are scheduled on D3 (as it incurs the minimum delay) and tail overlapping re-scheduled burst segments are scheduled on D2.

Though there is no loss of data bursts as shown in figure but for head overlapping and tail overlapping portion separate channels D3 and D2 respectively has been used which in turns to be expensive in terms of cost and looks un-effective as well. Thus the limitations of existing algorithms are both algorithms consider only one side of void. Next we propose a new channel scheduling algorithms which considers both end of a void in scheduling and also utilizes void efficiency and blocking probability of data burst is minimum.

3. Methodology

In this section we propose a new scheduling algorithm called Best Fit Void Filling (BFVF), which attempts to maximize the channel utilization and minimize the burst loss. Our propose 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:

$$\text{Utilization} = (a * 100) / x$$

Where 'a' is the data burst length and 'x' is the void length.

In figure 3, for first case, void utilization factor for B1 on channel D1, D2 and D3 are $(E_a\text{-}S_a)/((S_{1,2})\text{-}(E_{1,1}))$, $(E_a\text{-}S_a)/((S_{2,2})\text{-}(E_{2,1}))$, $(E_a\text{-}S_a)/((S_{3,3})\text{-}(E_{1,1}))$ respectively. If void utilization factor exceeds over 100 percent then the factor having close to 100 percent is considered. Here according to figure, using void utilization factor, it selects the channel D3 for the first case to schedule the portion of data burst B1. Since it cannot schedule all the portion of data burst B1 the overlapping portion of data bursts segments is reschedule. For that the remaining channel is D1 and D2 since channel D3 is already been used. For reschedule data burst segments that is for second case we again calculate the void utilization factor for remaining portion of data burst B1 which have to be rescheduled and calculated as$(E_a\text{-}R_a)/((S_{1,2})\text{-}(E_{1,1}))$, $(E_a\text{-}R_a)/((S_{2,2})\text{-}(E_{1,1}))$ where Ra is the start time for reschedule burst segment. In case, the void is greater than MAX_DELAY, the unscheduled burst is delayed for MAX_DELAY and the non overlapping burst segments of unscheduled burst is scheduled, while the overlapping burst segments are dropped. In this case, according to formula the data channel D2 is selected since its channel utilization factor for remaining reschedule burst segment is better than channel D1.

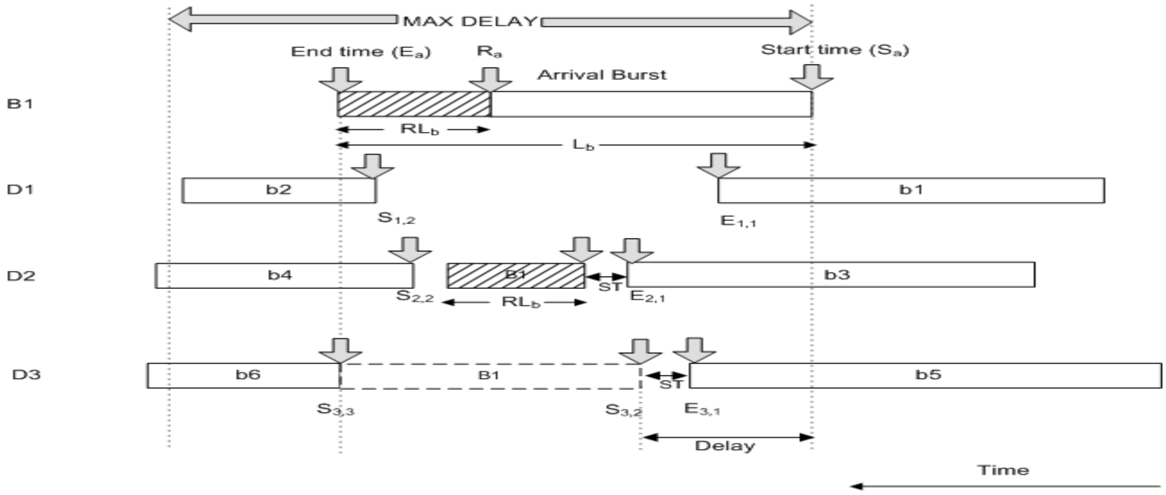

Fig 3 Illustration of BFVF segmented based algorithm

Hence, the reschedule data burst segment is scheduled on channel D2. And the data channel D1 which is free can be completely used for new arrival data burst. Thus the channel utilization is higher and burst loss ratio is lower in our propose scheme than in NP-DFMOC-VF and NP-SFMOC-VF algorithms. We work out an example to show void utilization on our proposed algorithm. We assume the following numerical values. For first case,

- $(S_{1,2}) - (E_{1,1}) = (220\text{-}140) = 80$ μs
- $(S_{2,2}) - (E_{2,1}) = (210\text{-}160) = 50$ μs
- $(S_{3,3}) - (E_{3,1}) = (230\text{-}145) = 85$ μs
- Length of data burst B1 (L_b) = (Ea-Sa) =110 μs
- Switching time (ST) = 10 μs
- Maximum Delay = 250 μs

Using channel utilization factor formula,

For D1, channel utilization = (110*100)/80 =137.5%

For D2, channel utilization = (110*100)/50 =220%

For D3, channel utilization = (110*100)/85 =129.4%

Here, we select the channel D3 since channel utilization of channel D3 is close to 100 percent as compare to channel D1 and D2. Note if the channel utilization had been less than 100 percent we go for channel utilization less than 100 percent instead of more than 100 percent.

For second case, (for RL_b)

Length of remaining data burst segment of B1, (RL_b)

- RL_b = (Ea-Ra) = 230-195 = 35 μs

Remaining channel D1 and D2

- $(S_{1,2}) - (E_{1,1}) = (220\text{-}140) = 80$ μs
- $(S_{2,2}) - (E_{2,1}) = (210\text{-}160) = 50$ μs
- Switching time (ST) = 10 μs

Using channel utilization factor formula for RL_b,

For D1, channel utilization = (35*100)/80 =43%

For D2, channel utilization = (35*100)/50 =70%

In this case, channel D2 is selected for reschedule the remaining data burst of B1 i.e. for RL_b. Also, the free channel D1 can be used for new arrival of data bursts. This shows that void utilization is higher in our proposed algorithm.

Table 1 Input data for channel scheduling of different algorithms

	CASE I (NPDFMOC-VF)			CASE II (NPSFMOC-VF)			CASE III (BFVF SEGMENTED)		
	B1	B2	B3	B1	B2	B3	B1	B2	B3
L_b=Ea-Sa (µs)	90-40=50	137-62=75	102-52=50	135-45=90	140-40=100	129-75=54	230-120=110	275-150=125	270-210=60
	(Si,j)-(Ei,j) µs			(Si,j)-(Ei,j) µs			(Si,j)-(Ei,j) µs		
D1	106-66=40			110-66=44			255-165=90		
D2	128-50=78			138-80=58			235-180=55		
D3	130-62=68			130-62=68			240-200=40		
D4	135-70=65			135-75=60			220-140=80		
D5	140-75=65			120-85=35			210-160=50		
D6	110-72=38			110-70=40			230-145=85		
D7				130-50=80			255-210=45		
D8				135-90=45			250-215=35		
D9				140-85=55			252-220=32		
ST (µs)	10			10			10		
W	6			9			9		
Maximum Delay (µs)	127			200			255		

Table 2 Output data for channel scheduling of different algorithms

	CASE I			CASE II			CASE III		
	NPDF MOC -VF	NPSF MOC -VF	BFVF Segme nted Based	NPDF MOC -VF	NPSF MOC -VF	BFVF Segme nted Based	NPDF MOC -VF	NPSF MOC -VF	BFVF Segme nted Based
Delay for non overlap ping burst B1	20 µs	20 µs	40 µs	27 µs	27 µs	25 µs	30 µs	30 µs	55 µs
Delay for non overlap ping burst B2	10 µs	10 µs	10 µs	36 µs	45 µs	32 µs	5 µs	25 µs	5 µs
Delay for non overlap ping burst B3	24 µs	30 µs	33 µs	15 µs	25 µs	15 µs	0 µs	10 µs	0 µs
Number of channel Used	3	6	4	3	9	6	3	9	6
Total packet loss	30 µs	0 µs	0 µs	10 µs	0 µs	0 µs	115 µs	0 µs	2 µs

4. Simulation and Results

We compare the performance of our proposed BFVF segmented based algorithm with that of NP-DFMOC-VF and NP-SFMOC-VF algorithm through simulation. For simulation proposed and to be more précised we take three cases for channel scheduling.

In each case we take three bursts B1, B2 and B3 which have to be scheduled by using different algorithms. W is the maximum number of outgoing data channels. According to given input data of table 1, we obtained an output as table 2 which is shown below. Considering a table II and its cases I, II and III we can see that in case I delay is more in our proposed algorithm as compare to NP-DFMOC-VF and NP-SFMOC-VF but in case II delay is less in our proposed algorithm than NP-DFMOC-VF and NP-SFMOC-VF where as in case III in our proposed algorithm delay is more for data burst B1 and less for data burst B2 and B3 as compare to NP-DFMOC VF and NP-SFMOC-VF.

Hence we can say that delay does not depend on type of algorithm we used but it depends on how the data bursts are schedule on the channels. Also from simulation of figure 4, 5 and 6 this can be seen.

Again considering table 2, this time we consider total packet loss for different algorithms versus number of channel used for different algorithms. According to table we simulate the result for this as shown in figure 7, 8 and 9. We can see that packet loss for our proposed algorithm is zero for case I and II and in case III packet losses are very low and number of channel used is also less comparing to NPSFMOC-VF algorithm. In NPDFMOC-VF algorithm, though the number of channel used is less than NPSFMOC-VF and our proposed algorithm but the packet losses are very high in NPDFMOC-VF then NPSFMOC-VF and our proposed algorithm.

Also from figure 1, 2 and 3 we draw a table and conclude the comparison of burst loss and channel utilization as follows.

Table 3 Comparisons of different algorithm in terms of Burst Loss and Channel Utilization

Algorithm	Burst Loss	Channel Utilization
NPDFMOC-VF	High	High
NPSFMOC-VF	Low	Low
BFVF Segmented	Low	High

5. Conclusion

In this paper we discuss performance of horizon and void filling scheduling algorithm. It is found that the void filling scheduling algorithm performs better than the horizon scheduling algorithms. However, there are limitations to the existing void filling scheduling algorithms. This limitation is mainly due to that; the existing schemes consider either the start time of the new data burst or end time of the previously scheduled data burst or start time of previously scheduled data burst and the end time of the new data burst. They do not take into account the data burst length and void length. We proposed an algorithm called BFVF Segmented based algorithm, which takes the arrival data burst length and void length into account in scheduling. Proposed scheme calculates the void utilization factor, and schedule the new data burst into a void channel having maximum void utilization factor.

The proposed scheme is compared with NPDFMOC-VF and BFVF Segmented. It is found that the proposed scheme perform better in term of channel utilization, packet loss and number of channel used.

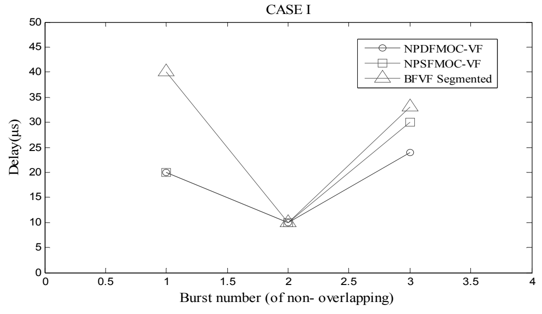

Fig 4 Delay vs. non overlapping burst for case I

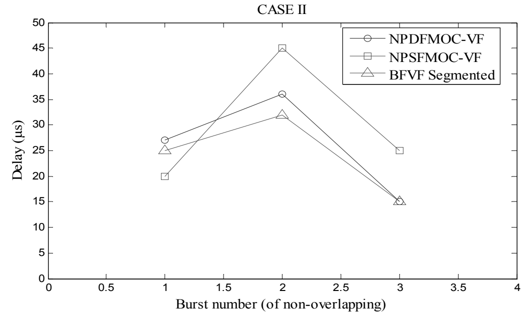

Fig 5 Delay vs. non overlapping burst for case II

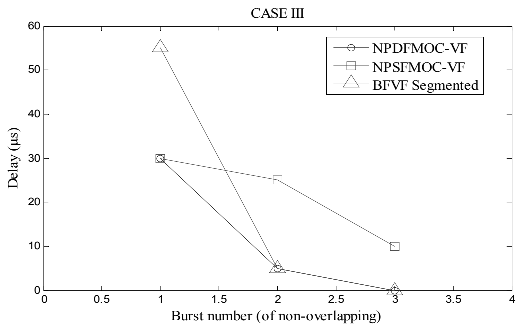

Fig 6 Delay vs. non overlapping burst for case III

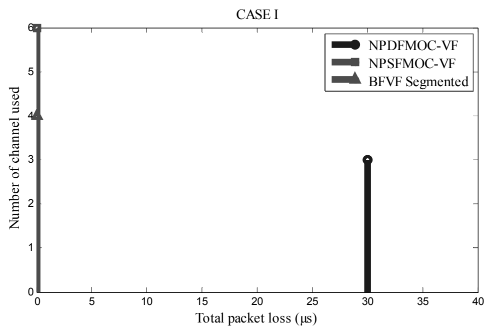

Fig 7 Number of channel used vs. Total packet loss for case I

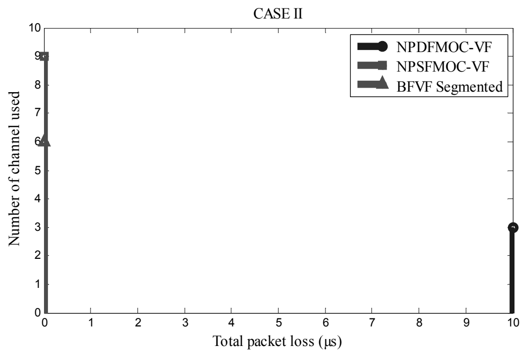

Fig 8 Number of channel used vs. Total packet loss for case II

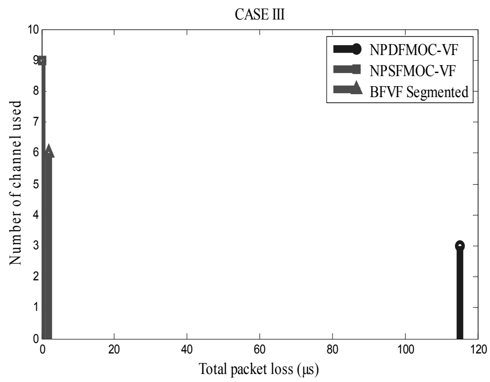

Fig 9 Number of channel used vs. Total packet loss for case III

References

1. K. Koduru, *"New Contention Resolution Techniques for Optical Burst Switching,"* Master's thesis, Louisiana State University, May 2005.

2. K. Dozer, C. Gauger, J. .Spath, and S. Bodamer, *"Evaluation of Reservation Mechanisms for Optical Burst Switching,"* AEU International Journal of Electonics and Communications, vol. 55, no. 1, January 2001.

3. M. Ljolje, R. Inkret, and B. Mikac, *"A Comparative Analysis of Data Scheduling Algorithms in Optical Burst Switching Networks,"* in Proceeding of Optical Network Design and Modeling, 2005, 2005, pp. 493 – 500.

4. Y. Xiong, M. Vandenhoute, and H. C. Cankaya*, "Control Architecture in Optical Burst Switched WDM Networks,"* IEEE JSAC, vol. 18, no. 10, pp. 1838 – 1851, October 2000.

5. M. Yoo, C. Qiao, and S. Dixit, *"QoS Performance of Optical Burst Switching in IP-Over-WDM Networks,"* IEEE Journal on Selected Areas in Communications, vol. 18, no. 10, October 2000.

6. W. M. Golab and R. Boutaba*, "Resource Allocation in User-Controlled Circuit-Switched Optical Networks,"* LNCS Spinger-Verlag, vol. 16, no. 12, pp. 2081 – 2094, December 1998.

7. M. Yang, S. Q. Zheng, and D. Verchere, *"A QoS Supporting Scheduling Algorithm for Optical Burst Switching DWDM Networks,"* In Proceeding of GLOBECOM 01, 2001, pp. 86 – 91.

8. J. Xu, C. Qiao, J. Li, and G. Xu, *"Efficient Channel Scheduling Algorithms in Optical Burst Switching Networks*," in Proceeding of IEEE INFOCOM, 2003, vol. 3, 2003, pp. 2268 – 2278

9. V.M Vokkarane and J.P Jue. *"Segmentation-based non preemptive scheduling algorithms for optical burst-switched networks."* In Proceedings, First Inter-national Workshop on Optical Burst Switching (WOBS), co-located with opti-Comm 2003, October 2003.